MicroCART

Design Document

Team Number: 45
Client: Dr. Phillip Jones
Advisor: Dr. Phillip Jones

Team Members:
Connor Ryan - Project Manager
Austin Beinder - Simulation/Controls Lead
Emily Anderson - Telemetry/Backend Lead
Grant Giansanti - Client Interactions/Testing
Tyler Johnson - Physical Systems Lead
Cole Hunt - Git Master/Device OS
Gautham Ajith - Lead Youtuber/GUI

Team Email: sdmay23-45@iastate.edu
Team website: https://sdmay23-45.sd.ece.iastate.edu/#

Revised: 11/28/2022 - Version Final

Executive Summary

Development Standards & Practices Used

List all standard circuit, hardware, software practices used in this project. List all the Engineering standards that apply to this project that were considered.

- IEEE 1936.1-2021: IEEE Standard for Drone Applications Framework
- IEEE 1625-2008: IEEE Standard for Rechargeable Batteries for Multi-Cell Mobile Computing Devices
- Bluetooth Low Energy (BLE)
- Utilize a wifi standard within the 802.11 specification for wifi communication
- Waterfall methodology

Summary of Requirements

List all requirements as bullet points in brief.

The requirements for this project were given to us by our client Dr. Jones to create and develop drone resources that can be simple enough to be used in the EE488 and EE475 courses as well as be able to be used in research. The

Applicable Courses from Iowa State University Curriculum

List all Iowa State University courses whose contents were applicable to your project.

- CPRE488
- EE475
- CPRE 288
- CPRE 458
- EE 333
- COMS 309
- CPRE 308

New Skills/Knowledge acquired that was not taught in courses

List all new skills/knowledge that your team acquired which was not part of your Iowa State curriculum in order to complete this project.

- CAD knowledge
- PCB design
- Raspberry Pi development
- MATLAB/Simulink
    - Outside of Electrical Engineering, there are no required classes in ECpE that requires us to learn MATLAB and it Simulink environment
- QT GUI programming
    - In order to improve the GUI some of us had to learn QT, a program in which no classes we have taken taught us this.
- GitLab
    - All of our documentation is using GitLab and there are no specific classes that teach us so some of us have to learn how to use this software to push our code.

Table of Contents

List of Acronyms/definitions (This should be the similar to the project plan)

MicroCART- Microprocessor Controlled Aerial Robotics Team

Crazyflie- Refers to the Bitcraze Crazyflie which is an open source drone hardware and software which has been developed for prior MicroCart Projects

CPRE 488 - Computer Engineering 488 refers to the class Embedded Systems Design

EE 476 - Electrical Engineering 476 refers to the class Control System Simulation

PID - Proportional–Integral–Derivative which relates to the Proportional–Integral–Derivative controller

PCB - Printed Circuit Board

FPGA - Field-Programmable Gate Array

IMU - Inertial Measurement Unit

ESC - Electronic Speed Controller

PWM - Pulse Width Modulation

GPIO - General-Purpose Input/Output

GUI - Graphical User Interface

CLI - Command Line Interface

MP-4 - Machine Project 4 refers to the CPRE 488 class project that involves using the MicroCART project

SDMAY - Senior Design May

BLE - Bluetooth Low Energy

I2C - Inter-Integrated Circuit

XIAO SAMD21 - Low powered Arduino Controller

1 Team

1.1 Team Members

- Austin Beinder
- Cole Hunt
- Connor Ryan
- Emily Anderson
- Gautham Ajith
- Grant Giansanti
- Tyler Johnson

1.2 Required Skill Sets for Your Project

- Software
  - C/C++
  - Qt
  - Arduino
  - MATLAB
- Controls
- Simulation
- Video Editing
- CAD/3D Printing
- PCB Design
- Networking/Communication

1.3 Skill Sets covered by the Team

- Software
  - C/C++ (Cole Hunt, Grant Giansanti, Emily Anderson, Gautham Ajith)
  - Qt (Emily Anderson, Gautham Ajith)
  - Arduino (Grant Giansanti, Connor Ryan)
  - MATLAB (Austin Beinder, Gautham Ajith)
- Controls (Austin Beinder, Tyler Johnson)
- Simulation (Austin Beinder, Gautham Ajith)
- Video Editing (Gautham Ajith)
- CAD/3D Printing (Austin Beinder, Tyler Johnson)
- PCB Design (Connor Ryan, Tyler Johnson)
- Networking/Communication (Emily Anderson, Cole Hunt)

1.4 Project Management Style Adopted by the team

We have an assigned project manager who is to uphold with the help of the other team members a waterfall project management style. The design of the product requires each step to be done prior to beginning the next. Product development follows a linear sequence as shown through the example below:

Researching parts, ordering, assembling, and testing is a time/resource consuming process which limits the number of iterations allowed

With all of the files being handled by git, we use it to track the progress of our project. Last team's year used git issues to track progress and using the same strategy would improve cohesiveness and readability for future teams. Progress is being documented through Git Issues and well written commit messages following a commit template with relation to the corresponding issues. Gantt chart tracks our top-level deadline for our project.

1.5 Initial Project Management Roles

- Project Manager: Connor Ryan
- Git Master- Cole Hunt
- Physical Systems Lead: Tyler Johnson
- GUI Lead: Gautham Ajith
- Backend/Telemetry- Emily Anderson
- Device OS- Cole Hunt
- YouTube Lead: Gautham Ajith
- User Interaction/Testing: Grant Giansanti
- Simulation/Controls Lead: Austin Beinder

2 Introduction

2.1 Problem Statement

What problem is your project trying to solve? Use non-technical jargon as much as possible. You may find the Problem Statement Worksheet helpful.

Professor Jones needs an improved quadcopter from the current crazyflie model. The current model is hard to control and has little wiggle room available for deploying new hardware or software. The interaction with the drone though its GUI can be slow and troublesome. This is important because many classes and research groups are using the crazyflie and we want to limit the time for debugging the drone and its software. The simulation of the drone is also several years old and not adjusted for the newest drone models.

2.2 Intended Users and Uses

Who will use the product you create? Who benefits from or will be affected by the results of your project? Who cares that it exists? List as many users or user groups as are relevant to your project. For each user or user group, describe (1) key characteristics (e.g., a persona), (2) need(s) related to the project (e.g., a POV/needs statement), and (3) how they might use or benefit from the product you create. Please include any user research documentation, empathy maps, or other artifacts as appendices.

The main users for this project will be CPRE 488 students, EE476 Students and graduate students doing research on embedded systems and control systems. Any students that are involved in embedded systems, drones or control system programming will care that they can look forward to learning about this system. 488 Students will appreciate that this will be easier to use than the CrazyFlie. They would also appreciate the ability to find PID values in simulation. EE476 Students will appreciate that this has more control options. They will also appreciate the simulation aspect where they can test control algorithms in simulation. Additionally professors that do drone research can use it for their research. They will appreciate that it has more customizability on the hardware end, and they will care about the potential to tune their algorithms in simulation, and they will care about the potential to implement them on an FPGA should we get to do that.

- Persona 1
  - Demographic: Student, Engineer
  - Hobbies/Interests: Embedded Systems, Drones, Programming, Video Games
  - Motivation: GPA,Graduate,Learning,Wordle, Fly a Drone
  - Personality: No social Life, Depressed, Confused
  - Values: Quality Time, Words of Affirmation, Success
- Persona 2
  - Demographic: Graduate Student, Researcher
  - Hobbies/Interests: Embedded Systems,Drones, Programming,
  - Motivation: Research, Learn, Graduate Thesis
  - Personality: No social Life, Smart, Determined
  - Values: Quality Time, Words of affirmation,Success
- Persona 3
  - Demographic: 476 Student, Engineer
  - Hobbies/Interests: Control Systems, Drones, Programming, Autonomous Systems
  - Motivation: Research, Learn, Experience for Control Systems Career, Graduating
  - Personality: No social Life, Determined
  - Values: Quality Time, Success

2.3 Requirements & Constraints

List all requirements for your project. Separate your requirements by type, which may include functional requirements (specification), resource requirements, physical requirements, aesthetic requirements, user experiential requirements, economic/market requirements, environmental requirements, UI requirements, and any others relevant to your project. When a requirement is also a quantitative constraint, either separate it into a list of constraints, or annotate at the end of requirement as "(**constraint**)." Ensure your requirements are realistic, specific, reflective or in support of user needs, and comprehensive.

- Hardware changes: Brushless motor, Raspberry pi Zero for controls, small as possible frame, board for IMU and motor controls, modified test stand
- Software changes: write own code for drone control, new GUI for drones,
- Functional Requirements:
    - The drone communication should have a maximum latency of 100 ms
    - Should be able to maintain hovering in place for 5 seconds
    - The quadcopter should perform as well or better than the Crazyflie currently used
- Resource Requirements:
    - The quadcopter should communicate over Bluetooth Low Energy or the wifi network cards on lab computers
    - The quadcopter must use brushless motors and a Raspberry Pi Zero
    - The flight control software should be useable on lab computers without the need for administrative privileges
- Physical Requirements:
    - The dimensions of the quadcopter will be less than 125mm x 125mm.
    - The quadcopter will have a balanced layout to allow for stable flight.
    - The quadcopter will weigh less than 300g.
- User Experiential Requirements:
    - The documentation for the hardware and software should be easily comprehensible by senior level engineering students interested in controls or embedded systems.
    - The GUI needs to be easy to navigate and simplicity of the interface
    - The GUI should provide reliable performance when communicating with the mini quadcopter, crashing less than 2.5 times per 3 hour work session.
- Environmental Requirements:
    - The quadcopter should be safe for indoor use.
    - Student labs will often have multiple students present and the risk of physical harm to students should be low. No more than one incident should occur per semester.

2.4 Engineering Standards

What Engineering standards are likely to apply to your project? Some standards might be built into your requirements (Use 802.11 ac wifi standard) and many others might fall out of design. For each standard listed, also provide a brief justification.

- IEEE 1936.1-2021: IEEE Standard for Drone Applications Framework
    - Within this framework it describes the support of drones including the flight platform, the control systems,qualification of operators and insurance. We will need this to make sure that we are using proper safety when operating our drone
- IEEE 1625-2008: IEEE Standard for Rechargeable Batteries for Multi-Cell Mobile Computing Devices

  - This describes the standards for using lithium ion batteries within a computing device. Since we are power both the PIE and the drone from lithion battery we need to make sure we are following proper procedures to make sure the drone is safe and does not exceed extreme temperatures
- Bluetooth Low Energy (BLE)
  - This describes the protocols of BLE and we need this standard because we are planning on controlling the pie with our BLE
- Utilize a wifi standard within the 802.11 specification for wifi communication
  - This standard describes the protocols of wifi communication. We will be using the drone over wifi as well, therefore we will need to to follow these standards.

3 Project Plan

3.1 Project Management/Tracking Procedures

Which of agile, waterfall or waterfall+agile project management style are you adopting. Justify it with respect to the project goals.

The project workflow follows the waterfall management style. The design of the product requires each step to be done prior to beginning the next. Product development follows a linear sequence as researching parts, ordering, assembling, and testing is a time/resource consuming process which limits the number of iterations allowed.

What will your group use to track progress throughout the course of this and the next semester. This could include Git, Github, Trello, Slack or any other tools helpful in project management.

Progress throughout the project will be tracked using Git issues. This is a favorable option since past files are hosted on git. Former teams tracked issues in a similar fashion so using the same approach will result in material being readable and cohesive.

3.2 Task Decomposition

In order to solve the problem at hand, it helps to decompose it into multiple tasks and subtasks and to understand interdependence among tasks. This step might be useful even if you adopt agile methodology. If you are agile, you can also provide a linear progression of completed requirements aligned with your sprints for the entire project.

1. New Mini Quadcopter
   a. Hardware
      i. Research quadcopter hardware
      ii. Select and order premade parts (frame,motors,battery)
      iii. Design IMU/ESC PCB

    iv. Order and assemble IMU/ESC PCB
    v. Test basic operation of IMU/ESC Board
    vi. Test mini quadcopter with Raspberry Pi flight controller firmware (correlates with 1.b.iii)
    vii. Evaluate possible improvements to 1st hardware revision
  b. Software
    i. Convert Crazyflie flight controller firmware to Raspberry Pi
    ii. Test I/O pins for proper operation
    iii. Test firmware on mini quadcopter (correlates with 1.a.vi)
    iv. Operating System flight controller**
    v. Convert flight controller firmware to ZynqBerry**
2. Ground Station
  a. Understand current CLI/GUI implementation
  b. Explore issues found while running through MP-4 (correlates with 3.b)
  c. Improve User Experience
  d. Minimize crashes and data loss
  e. Implement WiFi option
3. Improve Lab Material
  a. Complete MP-4 to gain familiarity with the lab
  b. Address issues encountered during MP-4
  c. Improve lab document for readability
  d. Add information to CPRE488 wiki to help students debug issues
4. Improve Simulation
  a. Understand current Simulink based simulation
  b. Move from being mostly Simulink based to being entirely Matlab based
  c. Add the ability to use BLDC motors
  d. Change parameters to matching the new drone design
    i. Motor resistances and torque constants, inductance?
    ii. Moments of inertia and mass
    iii. Thrust of propellers
    iv. Battery capacity and discharge curve
    v. Calibrate IMU model to match our hardware
  e. Improve the simulation User Interface
  f. Make it easier to add new drone models - save drone configuration menu
  g. Determine PID values automatically

3.3 Project Proposed Milestones, Metrics, and Evaluation Criteria

What are some key milestones in your proposed project? It may be helpful to develop these milestones for each task and subtask from 2.2. How do you measure progress on a given task? These metrics, preferably quantifiable, should be developed for each task. The milestones should be stated in terms of these metrics: Machine learning algorithm XYZ will classify with 80% accuracy; the pattern recognition logic on FPGA will recognize a pattern every 1 ms (at 1K patterns/sec throughput). ML accuracy target might go up to 90% from 80%.

In an agile development process, these milestones can be refined with successive iterations/sprints (perhaps a subset of your requirements applicable to those sprint).

- Quadcopter will communicate with the ground station over WiFi with a latency less than 10ms.
- After tuning PID values quadcopter will be able to maintain a controlled hover for more than five seconds.
- The flight controllers control loop will execute at the same speed or faster than the current crazyflie flight controller.
- A group will experience no more than 5 errors using the ground station when completing MP-4.

3.4 Project Timeline/Schedule

• A realistic, well-planned schedule is an essential component of every well-planned project

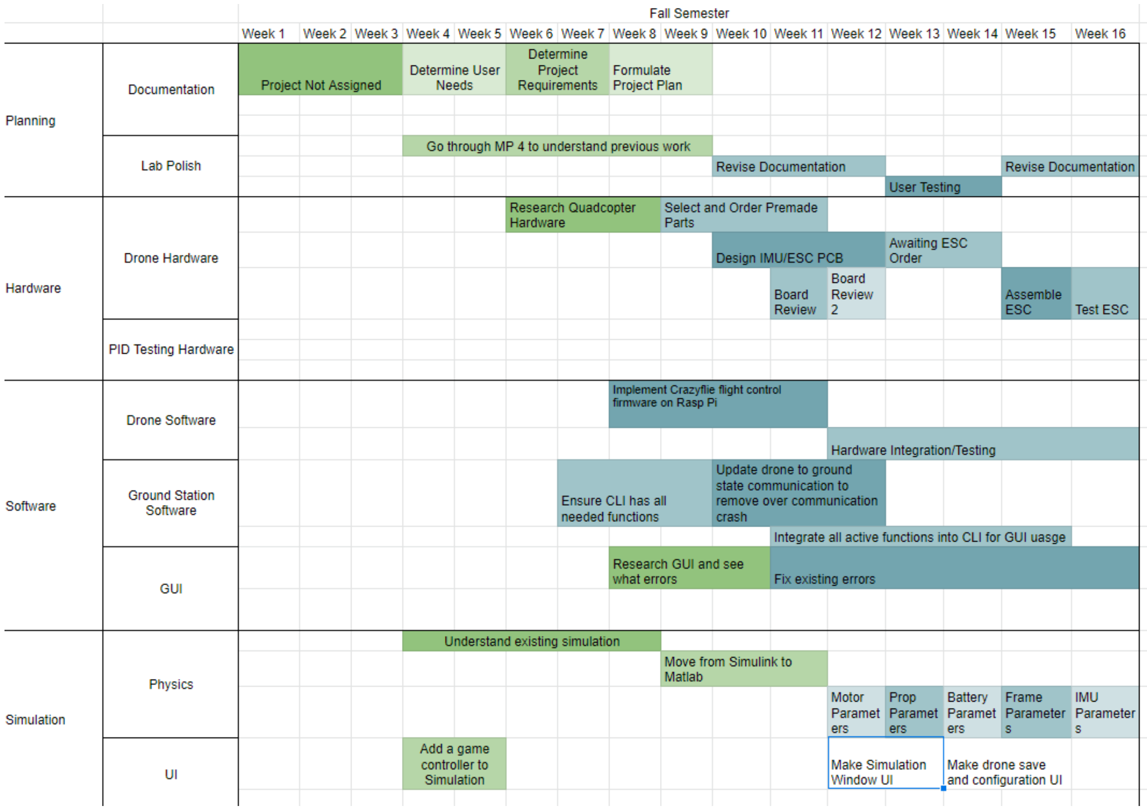
• Most scheduling errors occur as the result of either not properly identifying all of the necessary activities (tasks and/or subtasks) or not properly estimating the amount of effort required to correctly complete the activity

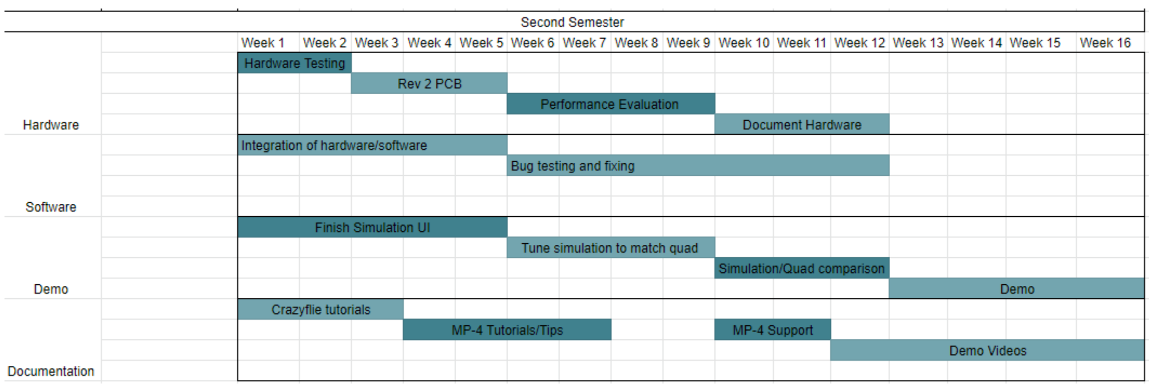
• A detailed schedule is needed as a part of the plan:

– Start with a Gantt chart showing the tasks (that you developed in 2.2) and associated subtasks versus the proposed project calendar. The Gantt chart shall be referenced and summarized in the text.

– Annotate the Gantt chart with when each project deliverable will be delivered

• Project schedule/Gantt chart can be adapted to Agile or Waterfall development model. For agile, a sprint schedule with specific technical milestones/requirements/targets will work.





3.5 Risks And Risk Management/Mitigation

Consider for each task what risks exist (certain performance target may not be met; certain tool may not work as expected) and assign an educated guess of probability for that risk. For any risk factor with a probability exceeding 0.5, develop a risk mitigation plan. Can you eliminate that task and add another task or set of tasks that might cost more? Can you buy something off-the-shelf from the market to achieve that functionality? Can you try an alternative tool, technology, algorithm, or board? Agile project can associate risks and risk mitigation with each sprint.

Quadcopter Hardware:

Risk	Probability	Impact	Mitigation

Ordered items arrive late	Moderate	Moderate	Use existing parts or order parts that would come quicker
Unable to purchase pi zero 2W	Moderate	Minor	Purchase pi zero W
Damage hardware during flight and testing	Minor	Minor-Severe	Order spare parts and replace hardware when possible
Mistake in PCB design	Moderate	Minor	Host board review to lower probability of mistakes. Design for easy modifications

Software:

Risk	Probability	Impact	Mitigation
Software contains bugs	High	Moderate	Extensive unit and functional testing
Software is hard to use	Moderate	Moderate	User interaction testing
Software crashes while drone is still in air	Minor	Severe	Implement failing gracefully techniques
Unable to communicate with the drone	Minor	Severe	Testing in safe environments
Sensors do not capture the data	Minor	Severe	Data testing and monitoring

Lab Documentation:

Risk	Probability	Impact	Mitigation
Incomplete	Minor	Minor	Use previous years software and documentation.
Revisions introduction inconsistencies in documentation	Moderate	Minor	Clarify inconsistencies and submit documentation changes later.

3.6 Personnel Effort Requirements

Include a detailed estimate in the form of a table accompanied by a textual reference and explanation. This estimate shall be done on a task-by-task basis and should be the projected effort in total number of person-hours required to perform the task.

Task	**Description**	**Time**
New Mini Quadcopter (Hardware)		
Research Quadcopters	We need to develop the best way to build this new	20 hours

	quadcopter keeping in mind cost, durability and payload size.	
Outsource Parts	We must buy the parts that we pick	25 hours
IMU/ESC PCB	Build the PCB that the controller will be controlling the quadcopter	75 hours
New Mini Quadcopter (Software)		
Convert Crazyflie flight controller firmware to Raspberry Pi	Embedded software and hardware to connect the drone to the controls on the computer. This will be the most intensive part of the project.	45 hours
Test I/O pins for proper operation	This is to test if the drone and the computer software are communicating with each other.	25 hours
Test firmware on mini quadcopter	This is to make sure that the firmware works	20 hours
Operating System flight controller	This is to create a way to be able to control the quadcopter	30 hours
Convert flight controller firmware to ZynqBerry	This is to change the firmware. This will be a hard part of the project	40 hours
Ground Station		
Understand current CLI/GUI implementation	This is to have a intuitive GUI that is easy to understand and to make sure the user can test the crazyflie	30 hours
Explore issues found while running through MP-4	Testing the previous implementation of the quad copter to gain a thorough understand of what can be improved	40 hours
Improve User Experience	This is to make sure if the ground station can be improved in any way such as the overall design to reduce friction or other variables…	30 hours
Minimize crashes and data loss	This is to make sure that there is a plan on what will happen if the drone loses a connection and falls from the air.	25 hours
Implement WiFi option	We need to implement a wifi option other than a radio signal to…	25 hours
Improve Lab Material		
Complete MP-4 to gain	Testing the previous implementation of the quad	84 hours

familiarity with the lab	copter to gain a thorough understand of what can be improved	
Address issues encountered during MP-4	Improve the flow and understandability of lab doc	40 hours
Improve lab document for readability Add information to CPRE488 wiki to help students debug issues	Add tips to help students progress through the lab	20 hours

3.7 Other Resource Requirements

Identify the other resources aside from financial (such as parts and materials) required to complete the project.

- Quadcopter Frame
- Raspberry Pi Zero 2W
- IMU/ESC PCB + Components
- Motors
- Propellers
- Battery

4 Design

4.1 Design Context

4.1.1 Broader Context

Describe the broader context in which your design problem is situated. What communities are you designing for? What communities are affected by your design? What societal needs does your project address?

Professor Jones needs an improved quadcopter from the current crazyflie model. The current model is hard to control and has little wiggle room available for deploying new hardware or software. The interaction with the drone though its GUI can be slow and troublesome. This is important because many classes including CPRE488 and EE476 and research groups are using the crazyflie and we want to limit the time for debugging the drone and its software. Reducing the time for debugging can allow for the users to be able to quickly conduct the research or learning that they need using this quadcopter.

Our project does need to adhere to the Public safety as these are drone and can be flown at very fast speed causing harm to the individuals in the lab however we will be reducing this risk by having several safety precaution

This project does have an economic impact because the current crazyflie drones that they are using cost around $195 and hopefully we can reduce that cost greatly. Additionally, we need to make sure that the drone we build is durable enough to last therefore future users don't have to spend a lot of money fixing or buying replacement parts.

List relevant considerations related to your project in each of the following areas:

Area	**Description**	**Examples**
Public health, safety, and welfare	How does your project affect the general well-being of various stakeholder groups? These groups may be direct users or may be indirectly affected (e.g., solution is implemented in their communities) - Dr. Jones and Students will need to be cautious of the drone causing physical damage to themselves or others and property	Increasing/reducing exposure to pollutants and other harmful substances, increasing/reducing safety risks, increasing/reducing job opportunities - The drone and its propellers are moving at high speeds - The new drone design is larger which will increase hazards and caution used
Global, cultural, and social	How well does your project reflect the values, practices, and aims of the cultural groups it affects? Groups may include but are not limited to specific communities, nations, professions, workplaces, and ethnic cultures. - The new drone design will allow for more research to be conducted	Development or operation of the solution would violate a profession's code of ethics, implementation of the solution would require an undesired change in community practices - Multi-core will allow for open-operating system usage - Controls research can still be conducted and relevant subjects
Environmental	What environmental impact might your project have? This can include indirect effects, such as deforestation or unsustainable practices related to materials manufacture or procurement. - Using battery powered drone	Increasing/decreasing energy usage from nonrenewable sources, increasing/decreasing usage/production of non-recyclable materials - Reduces the use of gas powered drone and allows for use indoors
Economic	What economic impact might your project have? This can include the financial viability of your product within your team or company, cost to consumers, or broader economic effects on communities, markets, nations, and other groups. - Cuts costs for materials with new drone design	Product needs to remain affordable for target users, product creates or diminishes opportunities for economic advancement, high development cost creates risk for organization - The drone needs to remain affordable and preferably cheaper than prior designs

4.1.2 Prior Work/Solutions

Include relevant background/literature review for the project

– If similar products exist in the market, describe what has already been done

– If you are following previous work, cite that and discuss the **advantages/shortcomings**

– Note that while you are not expected to "compete" with other existing products / research groups, you should be able to differentiate your project from what is available. Thus, provide a list of pros and cons of your target solution compared to all other related products/systems.

Detail any similar products or research done on this topic previously. Please cite your sources and include them in your references. All figures must be captioned and referenced in your text.

The Micro cart team has been around for over 20 years trying to continuously improve this drone. There have been many different implementations of this over the years that we can take inspiration from. Each of these teams has a git repo and last year's team has a youtube channel that we can reference. One of the past projects has been a very large quadcopter that was too big to be used in the class SDMAY 15-28.

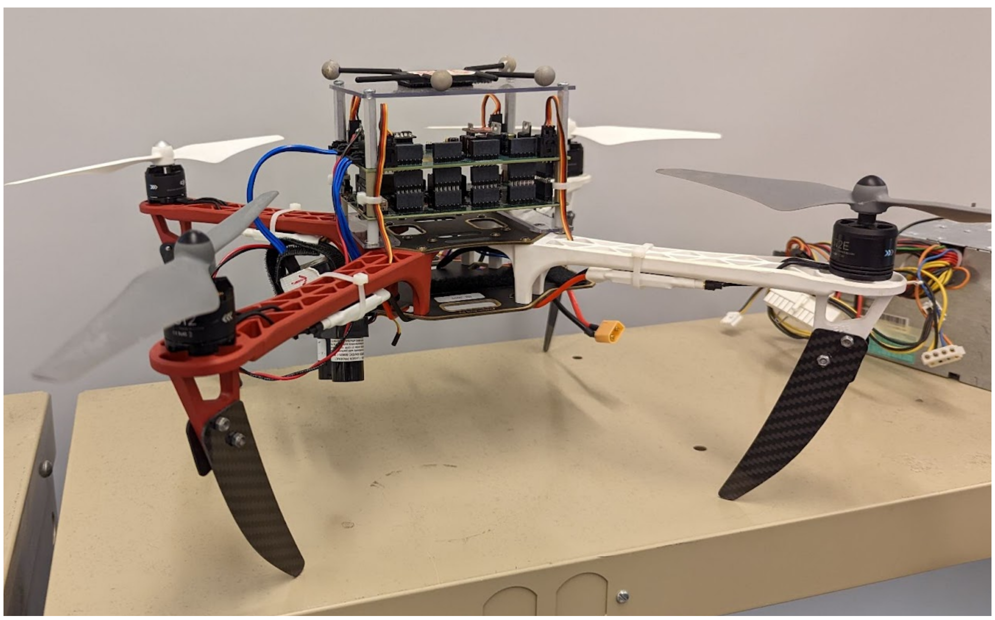


Figure 1: Team 15-28 microcart

As previously mentioned this previous implementation was very large and was not easily able to be controlled in addition to being a safety hazard. The succeeding teams sought to reduce the size and also the cost.

The previous team used a crazyflie as the drone of choice. Their model was to use a crazyflie as seen in Figure 2, and make their own software that students could use in the class.

Figure 2: Crazyflie

This implementation worked however feedback from students said it was very buggy and was hard to control. So therefore we would like to use this feedback to make our drone much easier to set up and create youtube videos to show how to set up the drone.

Finally we can also use other drone enthusiast work to help us on this project. There are many webinars that mathworks puts on that shows how to connect a drone to the simulations. These resources will help us to expedite our prototyping time.

4.1.3 Technical Complexity

Provide evidence that your project is of sufficient technical complexity. Use the following metric or argue for one of your own. Justify your statements (e.g., list the components/subsystems and describe the applicable scientific, mathematical, or engineering principles)

1. The design consists of multiple components/subsystems that each utilize distinct scientific, mathematical, or engineering principles –AND–
2. The problem scope contains multiple challenging requirements that match or exceed current solutions or industry standards.

This project consists of multiple components including, designing the simulation and building the drone, designing the 488 Lab along with other components. To build the drone we will need to be able to design and assemble a PCB that can hold the raspberry pi, connect the battery, drive the motors and connect to the computer to be controlled. Additionally we need to be able to simulate the drone using a type of software and this has to have similar characteristics to the real life drone. Finally we will need to construct a lab that 488 students can follow easily and be able to conduct their test on the drone without having major technical problems or a long debugging time.

This will also require redesigning the test stand to match our actual drone. Our problem has a multitude of different challenging requirements that need to exceed the current solution of a crazyflie.

4.2 Design Exploration

4.2.1 Design Decisions

List key design decisions (at least three) that you have made or will need to make in relation to your proposed solution. These can include, but are not limited to, materials, subsystems, physical components, sensors/chips/devices, physical layout, features, etc. Describe why these decisions are important to project success.

Our project is to design a quadcopter. Quadcopters have a limited number of physical components and these include a frame, flight computer, battery, motors, motor controllers, and propellers. Our first design decision was to select the components we would use in supporting the quadcopter. Our client specified that they would like us to either use a Raspberry Pi Zero or Zynqberry Zero as the main flight computer. We then also would require an IMU to support the flight computer in determining its position, and also a voltage regulator for stepping down the battery voltage. Knowing that we require these sets of components, we have spent some time in speccing out at least three different options for each of these and then selecting what we believe to be the best combination of them to make the hardware design of the drone. The selection of these components is important because if we select the wrong components the drone may be too heavy, or hard to fly correctly which will make the users of the drone have a less enjoyable experience and will also decrease its adoption rate by researchers.

The next design decision is based upon the boards being used. Because the Raspberry Pi and Zynq both have the same footprint, but both lack an IMU, a support board must be developed. This board needs to have the same footprint as these motherboards so that it can slot on top of it. This board also needs to have pins that connect to a lighthouse detection board that was used on last year's drone. This board will hold the IMU, and the voltage regulator. The decision of how to make this board is important because it will be necessary in order for the drone to fly, which means it will be blocking on all other components of the project.

Our third design decision was to focus on improving the functionality of the user interface on the software component of the project. Currently, there are bugs where the UI will randomly crash because messages are not being received correctly. We have not fixed these problems yet, but a large focus of our project will be to remedy these issues. The decision to focus on fixing these issues will increase the amount of success that students enjoy during the lab.

4.2.2 Ideation

For at least one design decision, describe how you ideated or identified potential options (e.g., lotus blossom technique). Describe at least five options that you considered.

To determine our different components, we found three options for the majority of sub components on the drone. We tried to find components that would be mostly compatible with each other. We did this to determine the range of options available. Below we have provided the information gathered about our different options. This process allowed us to learn in depth about the different components of a drone such that we could explore different ideas simply through finding the options available.

Frame

Specification	CinemAh	Cockroach	GEPRC CP 2"
Weight	68g	6.98g	50g
Size	150mm	75mm	115mm
Motor Size	14xx/20xx	0802	1104~1207
Propeller Size	3in/76mm	40mm	2in
Price	$89.99	$7.99	$34.99

Motors

Specs	**Option 1**	**Option 2**	**Option 3**
Weight	4.7g	7.2g	5.25g
Size	F1203	M1106	F1203
9x9 Mounting?	Says “Standard mounting pattern”	Yes	Yes
KV	3500KV, 5500KV	6000KV	7000KV
Configuration	9N12P	-	9N12P
Diameter	16.2mm	16mm	15.5mm
Shaft Diameter	1.5mm	1mm	1.5mm
Internal Resistance	-	-	198mOhm
Current Draw	1.2A @10V Idle	-	0.64A @10V
Power Draw	-	-	192W Max
Battery	3-4S	3-4S	2-3S
Recommended Propeller Size	Recommended for 2”-3” racing builds	“Toothpick builds”	-
Cost Per Motor	$13.99	$13.99	$14.99

Battery

Spec	Option 1	Option 2	Option 3
Style	3S	3S	4S
Nominal Voltage	11.1V	11.1V	14.8V
Capacity	450mAh	550mAh	650mAh
C rate	80C	70C	80C
Dimensions	14.5x32x57mm	17x31x56mm	26x28x59mm
Weight	47g	55g	80g
Price per battery	$8.99	$9.99	$14.17

Propellers

Option #1: 40 mm
Link:
https://www.getfpv.com/propellers/micro-quad-propellers/hqprop-40mm-2-blade-micro-whoop-propeller-1-5mm-shaft-set-of-4-gray.html

- Cost: $2.25

Option #2: 2in
Link:
https://www.getfpv.com/propellers/micro-quad-propellers/betafpv-gemfan-2020-4-blade-propeller-1-5mm-shaft-blue.html

- Cost: $2.50

Option #3: 3in
Link:
https://www.getfpv.com/propellers/micro-quad-propellers/emax-avia-th1609-3-2-blade-propeller-set-of-4.html

- Cost: $2.99

Regulator

Value	Regulator 1	Regulator 2	Regulator 3
Dual Voltage Output?	Yes	Yes	Yes
Max Current Output	3A/2A	3A/2A	3A/2A
Input Voltage Range	4.5~18V	4.5~18V	4.5V~18V
Output Voltage Range	0.76~7V	0.8~15V	0.76~7V
Dimensions	4.4x5mm	4x4mm	4x4mm
Adjustable?	Yes	Yes	Yes
Cost	$2.85	$3.03	$2.99

IMU

Field	IMU 1	IMU 2	IMU 3
Type	Accelerometer, Gyroscope, Temperature, 6 Axis	Accelerometer, Gyroscope, 6 Axis	Accelerometer, Gyroscope, Magnetometer, 9 Axis
Comm Busses	I²C, SPI	I²C, SPI	I²C, SPI, UART
Accelerometer Refresh Rate	12.5-1600Hz	1.6kHz	500 Hz
Accel Sensitivity	±2/±4/±8/±16 g full scale	+- 24 g	+- 8 g
Gyro Refresh Rate	12.5-1600Hz	1.6kHz	1 kHz
Angular Sensitivity	±125/±250/±500/±1000/±2000 dps full scale	±125/±250/±500/±1000/±2000 dps full scale	+- 2000 dps
ADC Bits	12?	16?	12-bit accel, 16-bit gyro
Supply Voltage	1.71~3.6V and 1.62V	1.2~3.6V	2.4V~3.6V
Dimensions	2.5x3x0.83mm	4.5x3mm	3.8x5.2x1.1mm
Current Consumption	1.25 mA	5mA	10 mA
Cost	$2.78	$10.70	$18.57 - 8 in stock

4.2.3 Decision-Making and Trade-Off

Demonstrate the process you used to identify the pros and cons or trade-offs between each of your ideated options. You may wish you include a weighted decision matrix or other relevant tool. Describe the option you chose and why you chose it.

The way we determined the best parts to use was by laying the options out next to each other as seen in the section above, along with listing the pros and cons of each item. We then chose the parts that best conformed to what we thought was beneficial, and conformed with the clients desires. We confirmed that they met our clients desires by showing them the options and picking items we all agreed on. Below is the full document showing the pros and cons, along with the options we used. The highlighted options are the choices we ended up deciding on.

Frames

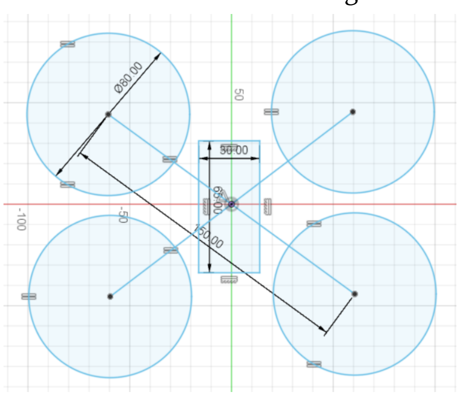
Option #1
https://newbeedrone.com/products/newbeedrone-cinemah

Pros

- Should have sufficient space

Cons

- Little too big
- Reviews claim it is fragile



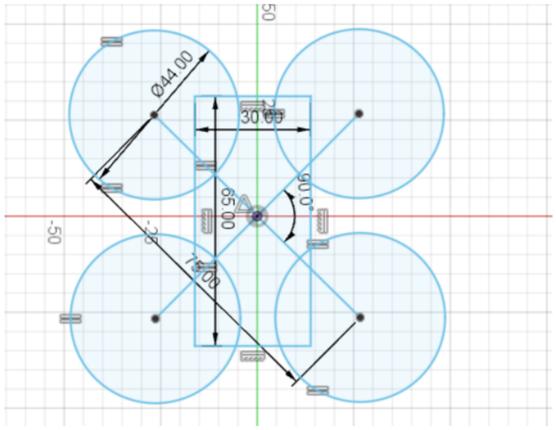
Option #2
https://newbeedrone.com/collections/whoop-fpv-drone-frames/products/cockroach75

Pros

- Ideal size
- Different Color Options
- Infinite Warranty

Cons

- Does not have enough space



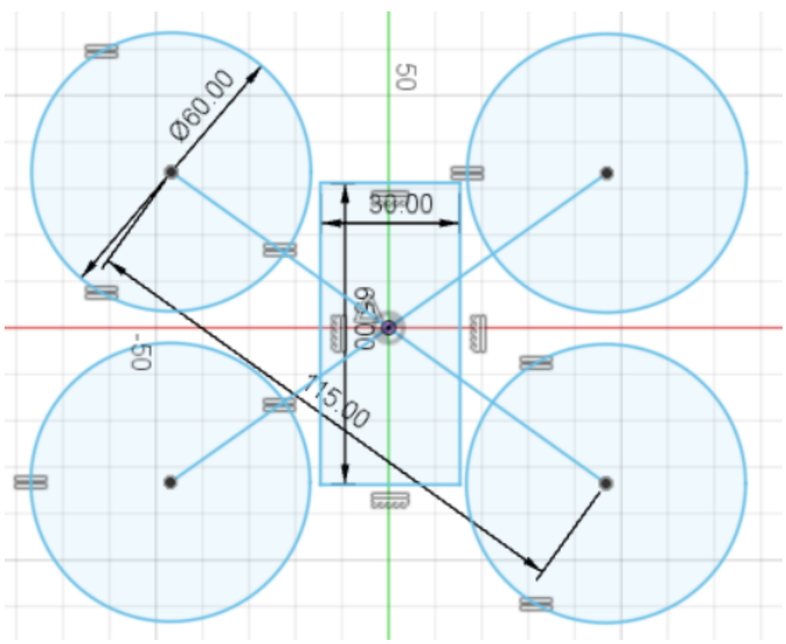
Option #3
[https://www.getfpv.com/geprc-cp-2-frame-kit.html?gclid=CjwKCAjwhNWZBhB_EiwAPzlhNqbRsqp0qXHjO5L130LDAjFPiIHzGB3xWbvz65zwj-5GQwNiQD_opB0Coko QAvD_BwE](https://www.getfpv.com/geprc-cp-2-frame-kit.html?gclid=CjwKCAjwhNWZBhB_EiwAPzlhNqbRsqp0qXHjO5L130LDAjFPiIHzGB3xWbvz65zwj-5GQwNiQD_opB0CokoQAvD_BwE)

Pros

- “Minimal” size to fit pi zero
- Looks like butterfly

Cons

- Lots of pieces, requires assembly
- Too heavy?
- Recommends 4S battery
- H-frame



Specification	CinemAh	Cockroach	GEPRC CP 2"
Weight	68g	6.98g	50g
Size	150mm	75mm	115mm
Motor Size	14xx/20xx	0802	1104~1207
Propeller Size	3in/76mm	40mm	2in
Price	$89.99	$7.99	$34.99

We ultimately chose option 3 because we believe if represented the smallest yet most robust drone frame, at the best affordable price. Furthermore, it was the only design we really believed that would actually fit the footprint of the Pi Zero without interfering with the propellers.

Motors

Motor selection will be dependent on frame selection. These motors assume the selection of the butterfly frame.

Option #1:
Link:
https://www.getfpv.com/motors/micro-quad-motors/flywoo-nin-v2-1203-pro-motor-3400kv-4850kv-5500kv-11500kv.html

Specs	**Values**
Weight	4.7g
Size	F1203
9x9 Mounting?	Says “Standard mounting pattern”
KV	3500KV, 5500KV
Configuration	9N12P
Diameter	16.2mm
Shaft Diameter	1.5mm
Internal Resistance	-

Torque Constant	-
Current Draw	1.2A @10V Idle
Power Draw	-
Battery	3-4S
Recommended Propeller Size	Recommended for 2”-3” racing builds
Cost Per Motor	$13.99

Pros

- Customers seem happy
- Max battery power
- More power in small frame
- Light

Cons

- Slowest option
- High Idle power consumption

Option #2
Link:
https://www.getfpv.com/motors/micro-quad-motors/t-motor-m1106-micro-motor-6000kv.html

Specs	**Values**
Weight	7.2g
Size	M1106
9x9 Mounting?	Yes
KV	6000KV
Configuration	-
Diameter	16mm
Shaft Diameter	1mm
Internal Resistance	-
Torque Constant	-

Current Draw	-
Power Draw	-
Battery	3-4S
Recommended Propeller Size	"Toothpick builds"
Cost Per Motor	$13.99

Pros

- Allows for 4S
- Fastest option

Cons

- Heaviest option

Option #3

Link: https://www.getfpv.com/motors/micro-quad-motors/t-motor-f1203-7000kv-motor.html

Specs	**Values**
Weight	5.25g
Size	F1203
9x9 Mounting?	Yes
KV	7000KV
Configuration	9N12P
Diameter	15.5mm
Shaft Diameter	1.5mm
Internal Resistance	198mOhm
Torque Constant	-
Current Draw	0.64A @10V
Power Draw	192W Max
Battery	2-3S

Recommended Propeller Size	-
Cost Per Motor	$14.99

Pros

- More specs are given
- Smallest option

Cons

- Only 2-3S

Options Laid Out

Specs	**Option 1**	**Option 2**	**Option 3**
Weight	4.7g	7.2g	5.25g
Size	F1203	M1106	F1203
9x9 Mounting?	Says "Standard mounting pattern"	Yes	Yes
KV	3500KV, 5500KV	6000KV	7000KV
Configuration	9N12P	-	9N12P
Diameter	16.2mm	16mm	15.5mm
Shaft Diameter	1.5mm	1mm	1.5mm
Internal Resistance	-	-	198mOhm
Current Draw	1.2A @10V Idle	-	0.64A @10V
Power Draw	-	-	192W Max
Battery	3-4S	3-4S	2-3S
Recommended Propeller Size	Recommended for 2"-3" racing builds	"Toothpick builds"	-
Cost Per Motor	$13.99	$13.99	$14.99

We chose option 1 because it was affordable, and because it was the lightest option.

Batteries

Our frame is capable of holding a 3S-4S battery. The boards require 5V DC. It seems to be generally recommended for drones to use LiPo batteries due to their lighter weight and higher maximum current over Li-Ion batteries.

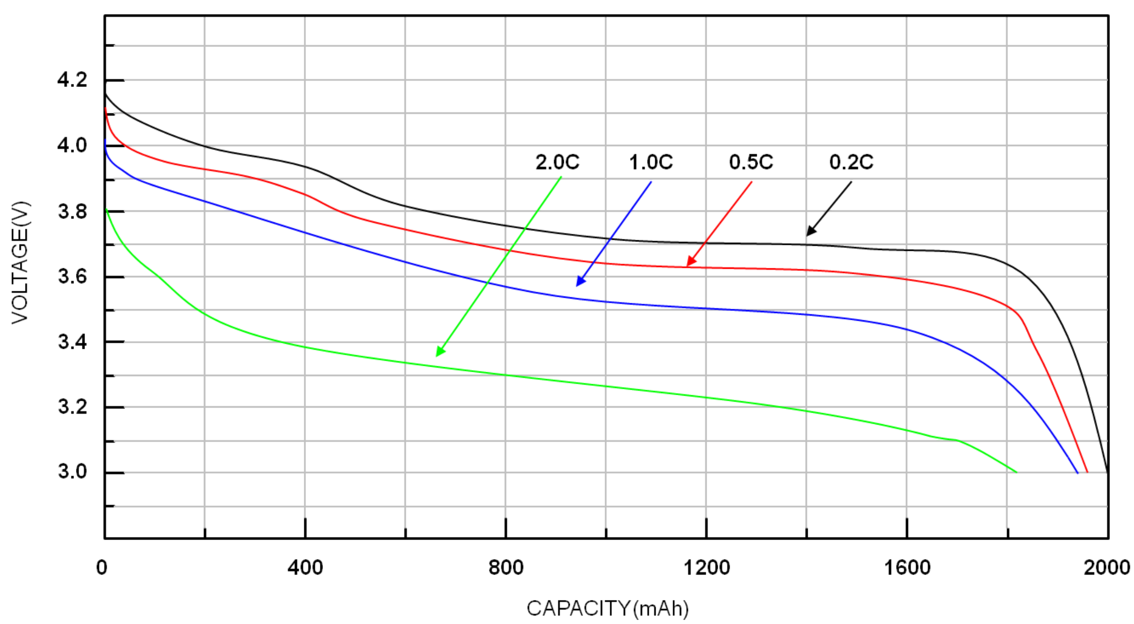
If our battery were 3S-4S our input voltage range would be:

Battery Type	Undervolt	Nominal Voltage (V)	Overvolt
3S	9.6	11.1	12.6
4S	12.8	14.8	16.8

The exact undervolt and overvolt specifications are usually not listed, but it seems that between 3.2 volts and 4.2 volts is commonly held as the usable voltage range, with many listing down to 3.0 volts.

LiPo batteries follow the below discharge curve, with varying voltages under different loads due to internal resistance and RC characteristics.

Battery selection will be dependent on the frame and motor selection. These batteries assume the selection of the butterfly frame.



As both the Pi Zero and Zynqberry parts depend on 5V USB power, it will probably be necessary to regulate the voltage supply to the boards.

Option #1
Link:
https://www.racedayquads.com/products/rdq-450mah-3s-fpv-battery-for-2-and-3-quads-80c

Spec	**Value**
Style	3S
Nominal Voltage	11.1V
Capacity	450mAh
C rate	80C
Dimensions	14.5x32x57mm (Fits butterfly well)
Weight	47g
Price per battery	$8.99

Pros
- Smallest dimensions
- Lightest option
- Cheapest option

Cons
- Smallest Capacity
- The butterfly might be able to hold more capacity
- Wide

Option #2
Link:
https://www.racedayquads.com/products/rdq-550mah-3s-fpv-battery-for-2-and-3-quads-80c-xt30

Spec	**Value**
Style	3S
Nominal Voltage	11.1V
Capacity	550mAh

C rate	70C
Dimensions	17x31x56mm
Weight	55g
Price per battery	$9.99

Pros

- Larger Capacity than option 1
- Less long than option 3

Cons

- Wide
- Lower C rating than options 1 and 3

Option #3
Link:
https://www.racedayquads.com/products/3-pack-of-rdq-series-14-8v-4s-650mah-80c-lipo-micro-battery-xt30

Spec	**Value**
Style	4S
Nominal Voltage	14.8V
Capacity	650mAh
C rate	80C
Dimensions	26x28x59mm (Fits butterfly)
Weight	80g
Price per battery	$14.17

Pros

- Large Capacity

Cons

- Heavy

- We don't know the exact dimensions of the butterfly battery area, and this might be a little tight.

Spec	Option 1	Option 2	Option 3
Style	3S	3S	4S
Nominal Voltage	11.1V	11.1V	14.8V
Capacity	450mAh	550mAh	650mAh
C rate	80C	70C	80C
Dimensions	14.5x32x57mm	17x31x56mm	26x28x59mm
Weight	47g	55g	80g
Price per battery	$8.99	$9.99	$14.17

We chose option 2 because it wasn't considerably more expensive or heavier than option 1, and the increases in battery capacity represent a larger gain than the increase in weight. It also has a less long and wide geometry than option 1 which allows us some more confidence when placing it in the butterfly frame.

Propellers

Every propeller was specced with 1.5mm shaft. Should we choose the motor with a 1mm shaft we will need to find different props. Each propeller was specced to the size of the different frames outlined above. The propeller will be specific to the frame we choose.

Option #1: 40 mm
Link:
https://www.getfpv.com/propellers/micro-quad-propellers/hqprop-40mm-2-blade-micro-whoop-propeller-1-5mm-shaft-set-of-4-gray.html

- Cost: $2.25

Option #2: 2in
Link:
https://www.getfpv.com/propellers/micro-quad-propellers/betafpv-gemfan-2020-4-blade-propeller-1-5mm-shaft-blue.html

- Cost: $2.50

Option #3: 3in
Link:
https://www.getfpv.com/propellers/micro-quad-propellers/emax-avia-th1609-3-2-blade-propeller-set-of-4.html

- Cost: $2.99

We chose option two because they were a compatible size with the butterfly frame, and a compatible shaft size with the motor we selected.

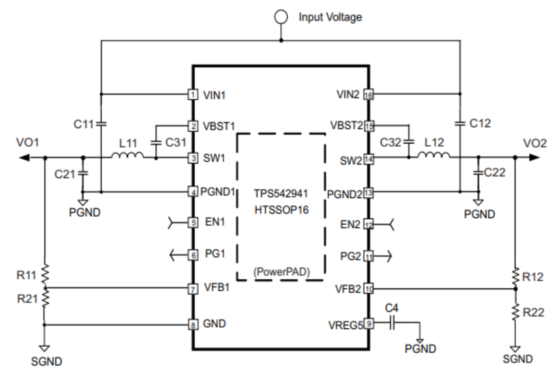
Voltage Regulator

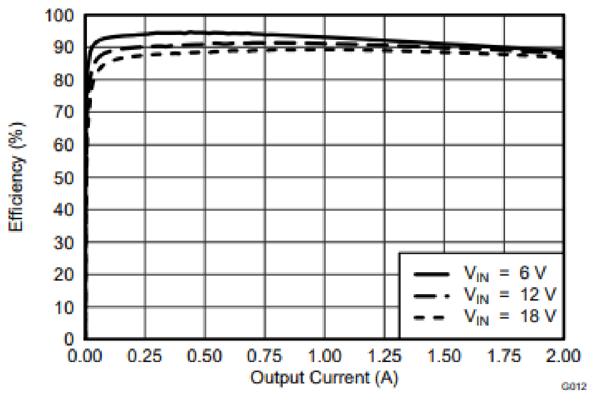
The raspberry pi zero and the zynqberry both require 5V DC. We can design our motor control and IMU board to run on a mixture of battery voltage and 5V DC. We should assume the motors will be running on the battery voltage and require no regulation, while the flight control will require 5V. As such a voltage regulator needs to be able to supply the power requirements of both such boards. It seems reasonable to take the power requirements of the pi zero, the zynqberry and the IMU and multiply by two in order to derate the regulator and provide room for expansion.

Part	**Estimated Maximum Current Consumption**
Pi Zero W	1 A (this is overkill)
Pi Zero 2W	2.5 A (recommended by datasheet)
Zynqberry	500 mA (per USB 2.0 spec)
IMU Option 1	1.25 mA
IMU Option 2	5 mA
IMU Option 3	10 mA

This means the regulator at most needs to power 3A, and because the IMU's operate on between 1.2 and 3.6 volts, we should probably try to find a regulator that supplies both 5 Volt and 3 Volt lines, preferably one intended for powering the USB 2.0 standard.

Option #1
Link: https://www.digikey.com/en/products/detail/texas-instruments/TPS542941PWPR/3548336





Pros

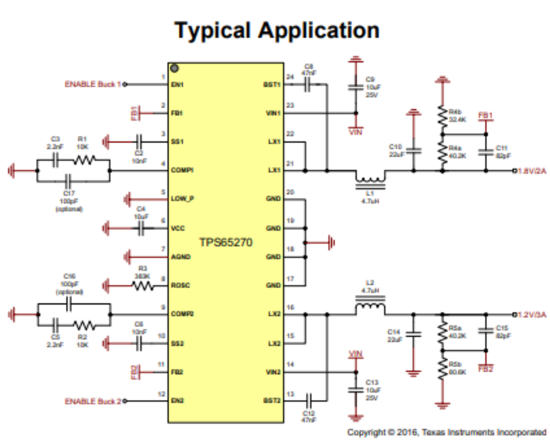
- Most affordable
- Requires few passive components.

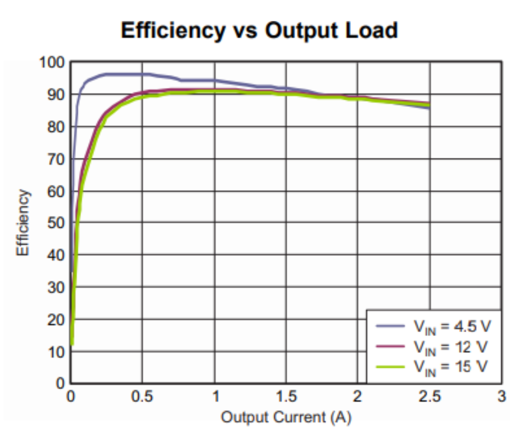
Cons

- Pad on bottom requires reflow… but so do the other two chips

Option #2

Link: https://www.digikey.com/en/products/detail/texas-instruments/TPS65270RGER/2798812





Pros

- Largest output voltage range
- Smallest footprint

Cons

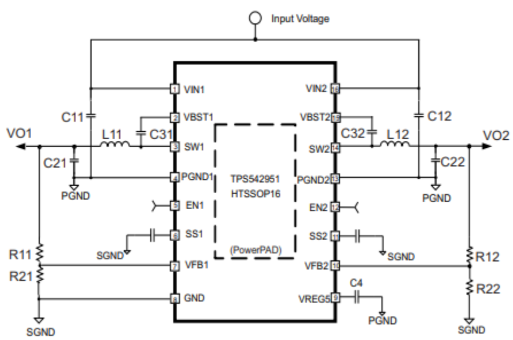
- Most expensive
- Requires the most external passive components

Option #3

Link:

https://www.digikey.com/en/products/detail/texas-instruments/TPS542951RSAR/4288723

https://www.ti.com/lit/ds/symlink/tps542951.pdf?HQS=dis-dk-null-digikeymode-dsf-pf-null-wwe&ts=1665119468248&ref_url=https%253A%252F%252Fwww.ti.com%252Fgeneral%252Fdocs%252Fsuppproductinfo.tsp%253FdistId%253D10%2526gotoUrl%253Dhttps%253A%252F%252Fwww.ti.com%252Flit%252Fgpn%252Ftps542951



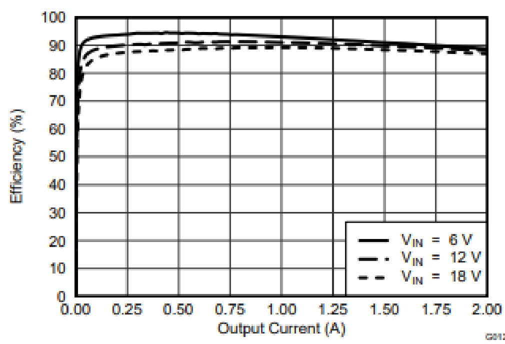


Figure 12. VO1 = 3.3V, Efficiency vs Output Current

Pros

- Nothing different than option 1, same chip smaller package
- Requires few passive components.

Cons

- Nearly identical to option 1 but more expensive

Value	**Regulator 1**	**Regulator 2**	**Regulator 3**
Dual Voltage Output?	Yes	Yes	Yes
Max Current Output	3A/2A	3A/2A	3A/2A

Input Voltage Range	4.5~18V	4.5~18V	4.5V~18V
Output Voltage Range	0.76~7V	0.8~15V	0.76~7V
Dimensions	4.4x5mm	4x4mm	4x4mm
Adjustable?	Yes	Yes	Yes
Cost	$2.85	$3.03	$2.99

We chose option one because it did basically everything the other options did but at a lower price.

IMU

According to hobbyists, a 9DOF Accelerometer that includes a 6-axis accelerometer, gyroscope, and a magnetometer is best.

Option #1
Link: https://www.digikey.com/en/products/detail/stmicroelectronics/LSM6DS3TR/5180552

Pros

- Cheapest
- Seems to be comparable with other IMU's
- Smallest form factor

Cons

- Cheapest may mean worst
- Not a 9-axis

Option #2
Link: https://www.digikey.com/en/products/detail/bosch-sensortec/BMI088/8634936

Pros

- Highest gravitational sensitivity
- Used on the previous drone so the software would still be compatible

Cons

- Not a 9-axis

Option #3
Link: https://www.ceva-dsp.com/wp-content/uploads/2019/10/BNO080_085-Datasheet.pdf
https://www.digikey.com/en/products/detail/ceva-technologies-inc/BNO085/9445940

Pros

- 9 axis
- Seems to be the most intended for a drone
- Arm processor on chip

Cons

- Only 8 in stock
- Price

Field	**IMU 1**	**IMU 2**	**IMU 3**
Type	Accelerometer, Gyroscope, Temperature, 6 Axis	Accelerometer, Gyroscope, 6 Axis	Accelerometer, Gyroscope, Magnetometer, 9 Axis
Comm Busses	I^2C, SPI	I^2C, SPI	I^2C, SPI, UART
Accelerometer Refresh Rate	12.5-1600Hz	1.6kHz	500 Hz
Accel Sensitivity	±2/±4/±8/±16 g full scale	+- 24 g	+- 8 g
Gyro Refresh Rate	12.5-1600Hz	1.6kHz	1 kHz
Angular Sensitivity	±125/±250/±500/±1000 /±2000 dps full scale	±125/±250/±500/±1000 /±2000 dps full scale	+- 2000 dps
ADC Bits	12?	16?	12-bit accel, 16-bit gyro
Supply Voltage	1.71~3.6V and 1.62V	1.2~3.6V	2.4V~3.6V
Dimensions	2.5x3x0.83mm	4.5x3mm	3.8x5.2x1.1mm
Current Consumption	1.25 mA	5mA	10 mA
Cost	$2.78	$10.70	$18.57 - 8 in stock

We chose option two because it is the same as the IMU on the previous drone which allows us to not change that portion of the software. We believe that by following this process, we were able to come to the best possible solution.

4.3 Proposed Design

4.3.1 Overview

Provide a high-level description of your current design. This description should be understandable to non-engineers (i.e., the general public). Describe key components or subsystems and how they contribute to the overall design. You may wish to include a basic block diagram, infographic, or other visual to help communicate the overall design.

Professor Jones needs an improved quadcopter from the current crazyflie model. The current model is hard to control and has little wiggle room available for deploying new hardware or software. The interaction with the drone though its GUI can be slow and troublesome. This is important because many classes including CPRE488 and EE476 and research groups are using the crazyflie and we want to limit the time for debugging the drone and its software. Reducing the time for debugging can allow for the users to be able to quickly conduct the research or learning what they need using this quadcopter. Furthermore, the processor that is currently on the drone is limited in performance which makes it less attractive to researchers.

A new design can be used to solve these problems, as well as enhance the drone's capabilities. Our new design will make the following improvements while keeping the capabilities of the previous model. We will make a small form factor drone that is robust to falling or crashing into things. This drone will also be capable of more complex computation by using a full processor like that of a Raspberry Pi Zero or Zynqberry Zero. These quadcore solutions will enable researchers and students to run their software on a more sophisticated operating system. It will also use brushless dc motors rather than brushed dc motors which will improve the dynamic performance of the drone as it flies.

Beyond just a new quadcopter design, we will also be enhancing the existing software and simulation software. For the main PC side software, we will be finding various GUI, reliability and speed improvements that we can make as time goes on. For the drone side, we will be trying to run the drone using two CPUs that have a high level linux OS, and two more bare metal CPU's. A reach goal for the project is to run the Robot Operating System (ROS) on one of the cores to help abstract the design.

For the simulation, we will be moving to being able to more easily switch between models of drones, and flight environments. We will also be moving the simulation to a better environment for researchers to use by refactoring it from simulink to matlab. Eventually, the goal will be that students in 488 could tune their PID parameters for the real life drone via the simulation. Should we implement ROS on the drone, we can also implement ROS in the simulation.

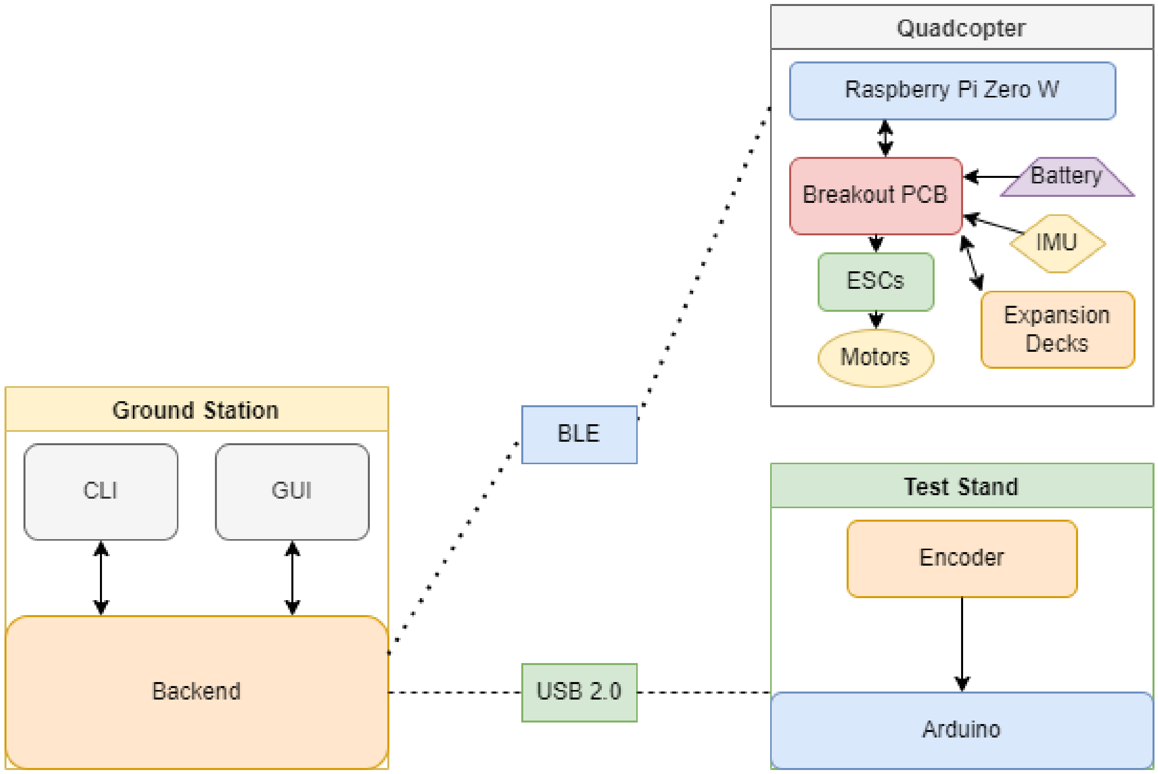


Figure 1: Proposed Design

The diagram above shows a high-level diagram of the system we are implementing. The PC runs the ground station software using either a CLI or GUI to send commands. The commands are sent over bluetooth to the Raspberry Pi Zero W mounted on the quadcopter. The microprocessor is running the flight controller using inputs and outputs from a circuit board our team will design. When calibrating the quadcopter it is mounted to a test stand which uses an encoder and Arduino nano to log position and rotation data back to the PC. This data is useful for tuning different control parameters in the quadcopters firmware.

4.3.2 Detailed Design and Visual(s)

Provide a detailed, technical description of your design, aided by visualizations. This description should be understandable to peer engineers. In other words, it should be clearly written and sufficiently detail such that another senior design team can look through it and implement it.

The description should include a high-level overview written for peer engineers. This should list all sub-systems or components, their role in the whole system, and how they will be integrated or interconnected. A visual should accompany this description. Typically, a detailed block diagram will suffice, but other visual forms can be acceptable.

The description should also include more specific descriptions of sub-systems and components (e.g., their internal operations). Once again, a good rule of thumb is: could another engineer with similar expertise build the component/sub-system based on your description? Use visualizations to support your descriptions. Different visual types may be relevant to different types of projects, components, or subsystems. You may include, but are not limited to: block diagrams, circuit diagrams, sketches/pictures of physical components and their operation, wireframes, etc.

As mentioned above this project is mainly about improving the quadcopter resources that are currently available to 488 students

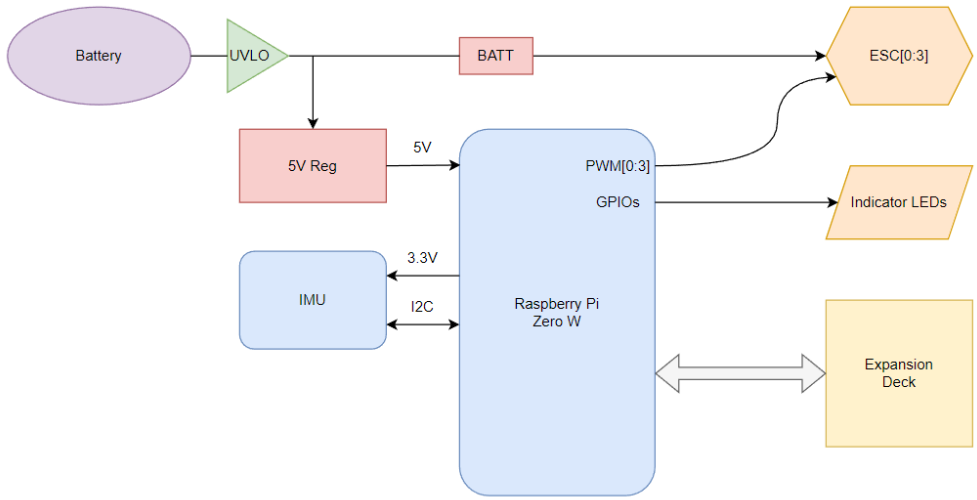


Figure 2: High Level PCB design

As you can see from above, our proposed quadcopter design uses a Raspberry Pi Zero W as the flight controller. The Raspberry Pi does not have the capabilities to operate a quadcopter on its own so we will design a printed circuit board to allow it to interface with an Inertial Measurement Unit (IMU), electronic speed controllers (ESC), and various expansion decks. The IMU will be used to gather feedback about the quadcopters rotation and tilt. The flight controller will use the data to adjust the motors accordingly through the ESC. The opportunity for expansion decks is also available to allow for the addition of more sensors for various applications.

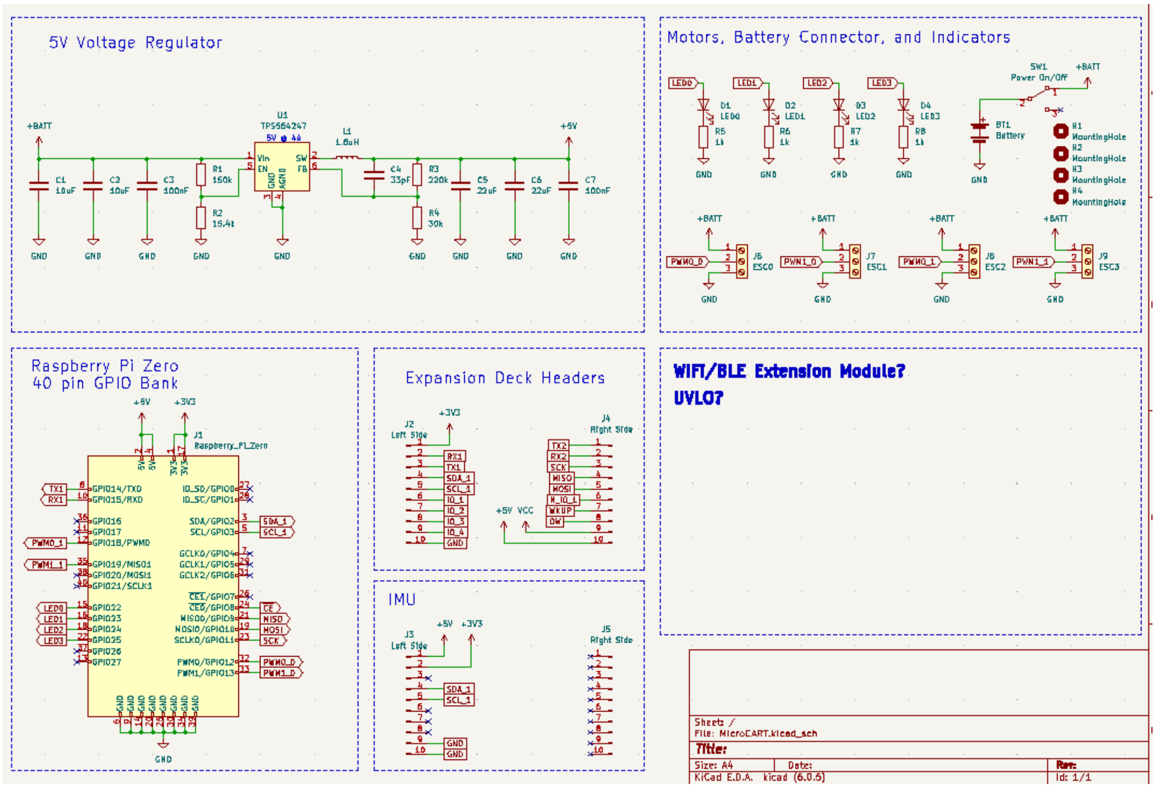


Figure 3: Preliminary Schematic

Shown above is a preliminary schematic of the breakout PCB including the power management, Pi connections, external connections, and an inertial measurement unit.

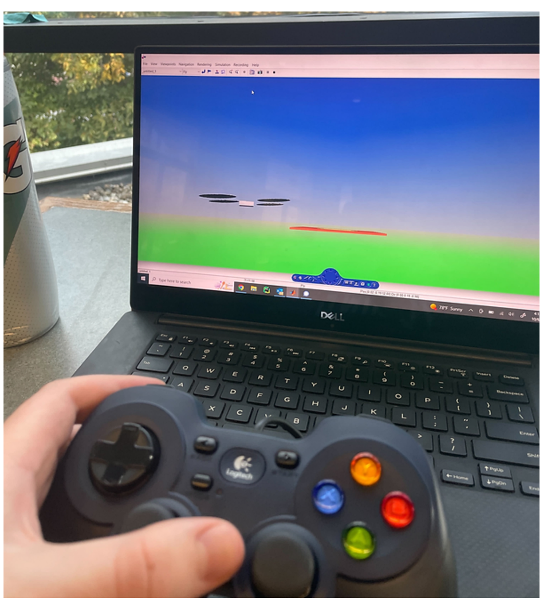


Figure 4: simulation of the drone

Figure 5: crazyflie drone

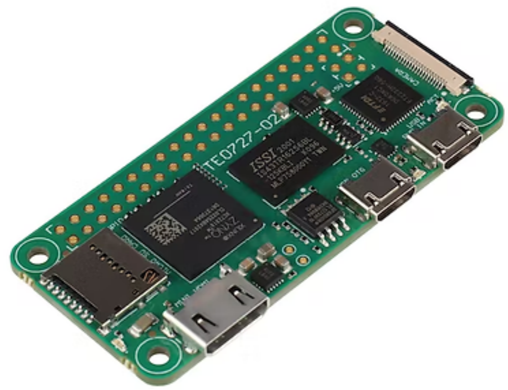


Figure 6: Raspberry Pi

4.3.3 Functionality

Describe how your design is intended to operate in its user and/or real-world context. What would a user do? How would the device/system/etc. respond? This description can be supplemented by a visual, such as a timeline, storyboard, or sketch.

As mentioned above our design is specifically targeting researchers, CPRE 488 Students, and EE476 students. It is intended as a very easy way for students to gain an opportunity to work on a real world application of embedded systems and to combine their hardware and software skills. Students will have the opportunity to put programs on the drone to directly control the functionality of the drone. They can control the drone with a radio and test it on the test stand. Additionally they can simulate it on MATLAB and test out their program before applying it to the drone. This can all be done on an easily accessible GUI that we will develop.

So a Student in CPRE 488 might first start with our simulation to simulate different PID values of the quad copter and to see what happens when you adjust these values. After mastering this part they will then move on to the drone, booting their software onto the drone using our intuitive GUI. This process will hopefully be very easy to follow for the students and make a lot of sense.Then the students can test their PID values or other research on the test stand to make sure that the drone is working properly. Finally they are able to control the quad copter using a joystick through the air. There are a couple of other functionalities however we are trying to make this as functional and as seamless as possible for students.

4.3.4 Areas of Concern and Development

How well does/will the current design satisfy requirements and meet user needs?

Based on your current design, what are your primary concerns for delivering a product/system that addresses requirements and meets user and client needs?

What are your immediate plans for developing the solution to address those concerns? What questions do you have for clients, TAs, and faculty advisers? We believe that based on the background our client has given us, and our continued meetings with them that our proposed design will satisfy the vast majority of user needs. Our understanding of user needs includes a better functioning 488 lab, a drone that offers more fine grained control, as well as greater computational power, and a simulation environment that would enable students and researchers to test their control algorithms in a low stakes environment.

To meet these expectations, we have delegated our members to specific areas of development from the GUI that students interact with, to the hardware that will enable greater computation and control, to the Simulation environment.

Our biggest concern when it comes to this is to stay on our timelines to ensure these tasks actually get done. We tend to be a pretty busy group, and so making sure that everyone actually finds time to work on the project has been a bit of a challenge. We do also have some pretty regular questions for our clients mainly focusing on what types of components they wish for us to use. We believe that by regularly meeting, asking questions and by continuing to address issues as we see them, we can surpass any and all of these issues and build a successful project.

4.4 Technology Considerations

Describe the distinct technologies you are using in your design. Highlight the strengths, weakness, and trade-offs made in technology available. Discuss possible solutions and design alternatives.

The design will use a Raspberry Pi Zero 2W for the main computing unit on the quadcopter. This is a lightweight computer with a Quad Core processor and easy GPIO breakout and control. The small scale of the board allows for mounting on the quadcopter frame. The Pi Zero only has 2 hardware PWM channels and 4 are needed to control the quadcopter, however the variable usage GPIO pins can be used to create software based PWM channels which would allow for full control of the motors. Currently there is a potential setback depending on the performance difference between the software and hardware PWM channels. If this is the case a PWM expander IC will be needed to control all 4 motors.

4.5 Design Analysis

Discuss what you have done so far, i.e., what have you built, implemented, or tested? Did your proposed design from 4.3 work? Why or why not? Based on what has worked or not worked (e.g., what you have or haven't been able to build, what functioned as expected or not), what plans do you have for future design and implementation work? For example, are there implications for the overall feasibility of your design or have you just experienced build issues?

We have started the beginning design phases of implementation for our MicroCART project. We are able to conduct testing of our design through simulation, specifically using simulink. This has allowed us to test remotely and test our code thoroughly. We are still designing our boards and waiting to order them. In the meantime we are conducting research about what boards work best and what is the preference for our project. Our plan in the future is to get everything we need ordered and then focus on testing and iterating through design. We want to get the board right the first time so we are focusing on making sure everything is correct in the earlier stages so we do not have to go back to a mistake. We are testing by working through the given labs for the class and marking down any errors or problems we run into along the way. Our proposed design is feasible at this current point in the project.

5 Testing

Testing is an **extremely** important component of most projects, whether it involves a circuit, a process, power system, or software.

The testing plan should connect the requirements and the design to the adopted test strategy and instruments. In this overarching introduction, given an overview of the testing strategy and your team's overall testing philosophy. Emphasize any unique challenges to testing for your system/design.

In the sections below, describe specific methods for testing. You may include additional types of testing, if applicable to your design. If a particular type of testing is not applicable to your project, you must justify why you are not including it.

When writing your testing planning consider a few guidelines:

- Is our testing plan unique to our project? (It should be)
- Are you testing related to all requirements? For requirements you're not testing (e.g., cost related requirements) can you justify their exclusion?
- Is your testing plan comprehensive?
- When should you be testing? (In most cases, it's early and often, not at the end of the project)

5.1 Unit Testing

What units are being tested? How? Tools?

- Each component used on the boards will have both their inputs and outputs tested. This ensures that on the component level everything will work. Then we can test the microcart on a system level. Because our project is very broad and contains many different subsystems we have to test many different components to make sure the whole system works.
- For the PWM expansion deck, we can send a "fake" packet to then see if it gets correctly parsed to each pin and value. We can also see if the correct PWM outputs are set. We will be using a usb with it connected to ubuntu to be sending the packets.
- IMU can be tested by reading the outputs based on the positioning of the imu. This can be done through the terminal on the pi.
- Telemetry can be tested by logging the values the device experiences within the device itself, and also logging the values received over telemetry and comparing them. We can also use timestamps to test its frequency. This can also be tested on the pi.
- Main PCB will be tested by first checking the resistance across the power rails to ensure the power supply will not be shorted. The power supply will then be attached to the board and the 5V supply will be measured using Pin 14 of the XIAO SAMD21 slot. The voltage level should be 5 ± 0.05V. Once the 5V supply has been verified the Raspberry Pi is attached. The 3.3V supply can then be tested using Pin 1 of the IMU slot. Additionally ERC and DRC checks with the PCB will make sure that our board is correct.
- Our overall drone hardware and ESC can be tested by implementing a crazyflie BigQuad board with the drone frame and motors to test that the software and external hardware works.

- GUI can be tested by doing extensive usability testing.
- The simulation will be tested by creating unit test demo files for each individual component, from low level through system integration. We will then also test many individual parameters that define the drone. We will plug these parameters into the simulation and compare the simulated values to telemetry values to determine the effectiveness of the match. We will use the same control algorithm for the drone in the simulation as we do in real life.

5.2 Interface Testing

What are the interfaces in your design? Discuss how the composition of two or more units (interfaces) are being tested. Tools?

I2C is used between the IMU and the Pi and the Pi and the PWM expansion. To test this we will run a program between the IMU and the PI to make sure that they interfacing correctly with one another. These tests will confirm functionality of these two modules within the entire system.

The ground station software and bluetooth are being used to connect the GUI to the drone itself. This will be tested by making sure data is being received from the drone and it constraints the correct information. Some of the data that we receive is the sensor data, the gyro and the accelerometer on the IMU.

Additionally there are many modules that communicate with each other. All of these will need to be thoroughly tested.

- Commands will be sent to the Raspberry PI Zero II W, over BLE, to blink test LEDs in order. The test will be repeated over WiFi to confirm both BLE and WiFi communication.
- The drone will have its gyroscope data logged over bluetooth. The drone will be placed on the test stand and rotated to confirm valid values are being read. The test stand has different configurations allowing the reading of the x,y,z axis of rotation.
- IMU data to the breakout board then the PWM breakout board and then the PI
- The interface between the Raspberry PI Zero II W and the ESC is the PWM board. The XIAO SAMD21 acts as the interface and reads I2C data to set the output of 4 PWM channels. The interface will be tested by increasing the duty cycle to rev the motors.

5.3 Integration Testing

What are the critical integration paths in your design? Justification for criticality may come from your requirements. How will they be tested? Tools?

- Integration of the PI and the PWM expansion will need to be tested. We will send a command over I2C to set PWM0 to 50% duty cycle. The duty cycle of the waveform will then be measured using a Picoscope. The duty cycle will then be set to 75% and validated with the Picoscope, then repeated at 25% duty cycle. The duty cycle will then be swept from 0-100% duty cycle. The test will be repeated for the remaining three PWM channels.
- Integration of the IMU and the PI will need to be tested. The PI will be configured with a test program streaming gyroscope and accelerometer data back to the PC. The IMU will be placed on a flat surface so it can calibrate and the gyroscope and accelerometer should read near zero values for all data besides the accelerometers z value representing the force of gravity. The IMU

will be placed on its side so gravity is completely in the x direction, then placed on the other side for an acceleration completely in the y direction. The gyroscope will then be tested by rotating the IMU around the board's z-axis. The data will be checked to validate a positive value for counterclockwise rotation and a negative value for clockwise rotation. The test will be replicated for the x- and y-axis.

- Integration of the ground station software and the drone will need to be tested. The compatibility of the ground station software and the drone will be tested by completing the MP-4 lab experiment using the new drone. The lab experiment requires setting parameters and setpoints on the drone and will thoroughly test the combination of the two systems.

5.4 System Testing

Describe system level testing strategy. What set of unit tests, interface tests, and integration tests suffice for system level testing? This should be closely tied to the requirements. Tools?

- The drone must be able to hover in place for 10 seconds with a minimal acceptable amount of moving. This will be used to test the control loop, the state estimation algorithm, the motor control. Additionally this will test how easily the drone can be controlled through the GUI. We want to make sure there is no lag from our commands to the drone movement.
- The drone must be able to maneuver correctly through a more complicated specified series of setpoints repeatedly when configured with the correct PID values. (10 times in a row without issues). This can test how easily our simulation can be transferred to the drone as well as testing the autonomy of the drone without user input.
- The ground station must be able to successfully and consistently communicate with the drone. This shall be specified by experiencing no issues for 5 battery life cycles in a row, on multiple drones.
- The drone should be able to be programmed with PID values on the ground station with minimal issues. It should also be able to be commanded from the ground station with minimal issues during this time. This shall be tested by tuning a few PID values, getting and setting each parameter, and applying setpoints at each iteration. This test shall be run on multiple drones, on multiple laptops.
- We can test our safety features by hovering, then hitting the kill switch, removing the battery while the drone propellers are still moving and testing the kill switch on the PCB. During an autonomous flight there will also be checks to make sure if a waypoint is outside of a safe range it will not allow the drone to go that far.

5.5 Regression Testing

How are you ensuring that any new additions do not break the old functionality? What implemented critical features do you need to ensure they do not break? Is it driven by requirements? Tools?

With all of these testings we will be using GIT for all the software to make sure that we can go back to older versions if we end up changing something. Additionally for the hardware for the initial PCB we have many detachable modules and test points that will make debugging and fixing and functionality within these components.

The requirements that are critical to our project are:

- The communication latency must be under 100ms for 10,000 telemetry messages in a row.
- The drone must be able to hover in place for 10 seconds.
- The control loop must respond in an expected manner to the inputs given.
- The state estimation algorithm must determine the state of the drone within an acceptable range of error.
- We can set up a series of automated unit tests that tests much of the functionality outlined above for software changes. We can run these tests as time goes on to confirm continued functionality.

Without these features our drone will not meet the specifications defined by our client as well as the drone will not be able to fly correctly.

5.6 Acceptance Testing

How will you demonstrate that the design requirements, both functional and non-functional are being met? How would you involve your client in the acceptance testing?

Through our project acceptance testing will be conducted as our last step of our design process. This will be conducted on a large scale to see if our drone's system will meet the requirements of our client, Dr. Jones. Dr. Jones can evaluate our drone by flying it, Using the gui, using the simulation tools and talking with students who will be using our drones within 488. After Dr. Jones surveys students and we get feedback on the problems they experienced in the lab and how our solution can be improved. Finally we can evaluate how well we increase the documentation for drone technology at Iowa state by the amount of people who find our tutorial videos helpful. These videos will be demonstrating an overview of our implementation and will communicate with our users on the structure of our design and its performance.

5.7 Security Testing (if applicable)

For our project, security testing is not applicable, because we are using the drones only inside in a controlled environment and not in the public. We do not have to worry about security.

5.8 Results

What are the results of your testing? How do they ensure compliance with the requirements? Include figures and tables to explain your testing process better. A summary narrative concluding that your design is as intended is useful.

While we currently have only done testing on the PI and the PCB schematic, the expected results of our testing should align with our functional requirements. Our drone should be able to hover in place for at least 5 seconds. Also, the latency for the communication should be less than 100 ms. On top of that, we are expecting the performance of this drone to be better than the crazyflie based on speed and reliability. Our implementation of the GUI should be more reliable and durable meaning that it should not crash often. 488 Students should be able to reliably use the drone without breaking or having to adjust our base code. With these results, we can expect less frustrated students working with the drones because it will be more intuitive and reliable.

Therefore, the development process should go more smoothly and more quickly. Plus, it will give time for students to dive into more capabilities with the drone.

6 Implementation

Describe any (preliminary) implementation plan for the next semester for your proposed design in 3.3. If your project has inseparable activities between design and implementation, you can list them either in the Design section or this section.

We are currently creating new schematics for our future work. There are no finalized designs to share at this time.

7 Professional Responsibility

This discussion is with respect to the paper titled "Contextualizing Professionalism in Capstone Projects Using the IDEALS Professional Responsibility Assessment", *International Journal of Engineering Education* Vol. 28, No. 2, pp. 416–424, 2012

Area of Responsibility	Definition	NSPE Canon	Ethics (IEEE)	Importance	Team Performance
Work Competence	Perform work of high quality, integrity, timeliness, and professional competence.	Perform services only in areas of their competence; Avoid deceptive acts.	Maintain and improve technical competence, and only take on tasks that you are qualified for.	High	Medium
Financial Responsibility	Deliver products and services of realizable value and at reasonable costs.	Act for each employer or client as faithful agents or trustees.	Reject any bribery	Medium	Low
Communication Honesty	Report work truthfully, without deception, and understandable to stakeholders.	Issue public statements only in an objective and truthful manner; Avoid deceptive acts.	Look for and listen to honest feedback on work, and correct errors.	Medium	High
Health, Safety, Well-Being	Minimize risks to safety, health, and well-being of stakeholders.	Hold paramount the safety, health, and welfare of the public.	Uphold the safety, health, and welfare of the public	High	Medium
Property Ownership	Respect property, ideas,	Act for each employer or	Improve the understanding	Medium	Medium

	and information of clients and others.	client as faithful agents or trustees.	by person of the capabilities of new technology		
Sustainability	Protect environment and natural resources locally and globally.			Low	N/A
Social Responsibility	Produce products and services that benefit society and communities.	Conduct themselves honorably, responsibly, ethically, and lawfully so as to enhance the honor, reputation, and usefulness of the profession.	Treat everyone respectfully and fairly and avoid injuring others.	Medium	Medium

7.1 Areas of Responsibility

Pick one of IEEE, ACM, or SE code of ethics. Add a column to Table 1 from the paper corresponding to the society-specific code of ethics selected above. State how it addresses each of the areas of seven professional responsibilities in the table. Briefly describe each entry added to the table in your own words. How does the IEEE, ACM, or SE code of ethics differ from the NSPE version for each area?

For the most part the IEEE code of ethics does not differ from the NSPE version for each area; however, for financial responsibility, it focuses more on motives behind work rather than the waste of money. Also, for property ownership, IEEE wants people to understand that there is new work emerging rather than focusing on giving credit to the individual's work.

7.2 Project Specific Professional Responsibility Areas

For each of the professional responsibility area in Table 1, discuss whether it applies in your project's professional context. Why yes or why not? How well is your team performing (High, Medium, Low, N/A) in each of the seven areas of professional responsibility, again in the context of your project. Justify.

Work Competence is high for our project because we are producing a product that is buggy at the moment, but the goal is for it to be used for other classes and researchers, so our work should be at the highest level since others will be using our product. Also, we are representing Iowa State, so we should show the quality of work produced out of Iowa State. Financial Responsibility is medium because we do not want the drones to be very costly since there will be more than one drone being produced; however, it is not a huge focus in our project. Communication Honesty is high because we are working with an advisor who has been on this team for many years, so he has a lot of good advice to give us. Also, it is important to keep other team members informed as well as our advisor/client. Health, safety, and well-being has a high importance because we are working with drones that are considered a flying hazard. We do not want to have a drone that may fly into someone and injure them. Property Ownership is medium because we want to give credit to open source code, but we are also working on moving away from open source to using our own code. Sustainability is low because the drones we are building will be used in a classroom setting. Social Responsibility is of medium importance because we are building something that will benefit the Iowa state ICpE community, but other than that, there are no current plans for this to be used outside of Iowa State Classrooms.

Currently, our team's work competence is performing at a medium level, mainly because we were slow to start working because we had to wait for parts to come. We are slowly getting more involved with our work so the level is getting close to being considered high. We are not very conscious about our financial responsibilities so it is low. Our communication honesty is high because we meet with our team a lot and with our advisor weekly. Also, we have a discord set up to allow for virtual communication. We understand that drones are dangerous so the level is medium. Our property ownership is medium just because we have not started developing any code yet, but we have reviewed all of the previous developed code from years past. Sustainability is N/A since this has not been a concern for our team. Lastly, our social responsibility is at a medium mainly because we know the purpose of our project, but some of us have not taken the class that these drones are used for so we do not know the current challenges in the class.

7.3 Most Applicable Professional Responsibility Area

We think that our communication honesty has been critical to our project, and our team has been doing a relatively good job with it. This is important because this project has been ongoing for many years, so there is a lot to learn about it, and there is a lot to communicate about with our advisor who is also our client. Our communication has been really good with our advisor as well as with our team which has led to being able to solve problems quickly while also being able to move forward more quickly.

8 Closing Material

8.1 Discussion

Discuss the main results of your project – for a product discuss if the requirements are met, for experiments oriented project – what are the results of the experiment, if you were validating a hypothesis – did it work?

As of right now, the results of our project are a bit underwhelming. We have yet to fly our new drone design. By the end of this project, we must successfully demonstrate a functioning drone that can be easily programmed by other students compared to the crazyflie drone last year. While we have yet to do this, we are on track to do so by the 488 lab next semester.

8.2 Conclusion

Summarize the work you have done so far. Briefly re-iterate your goals. Then, re-iterate the best plan of action (or solution) to achieving your goals. What constrained you from achieving these goals (if something did)? What could be done differently in a future design/implementation iteration to achieve these goals?

So far, the team has completed the PCB, made progress on working with the raspberry pi, converted most of the simulation to raw matlab code, and specced and ordered parts for the rest of the drone.

The goals for our project are to improve the 488 lab experience, and to build a platform that would be attractive to researchers seeking to conduct studies on quadcopters. To improve the experience, we will improve the physical design of the drone, update the user-interfaces of the GUI, ground station and drone software. We also seek to improve the current simulation support to better match both our drones, and researchers' needs.

Our plan of action for the rest of the year is to continue working on the project. Our board has just arrived and now that it has, there is much less blocking the team from rapidly ramping up and getting our drone flying reliably. The team will now be working on building the test stands that the pid values will be calibrated with, and with getting the software of previous teams uploaded and adapted to our drone. Once this is done, we will be focused on making sure that the platform is working reliably for the lab, and will also be seeing just how far and how useful we could take the simulation.

Early on we ran into supply chain issues when ordering physical portions of our projects. However, this only resulted in delays and we were able to hit most of our deadlines related to the physical changes on the drone. Underestimating the complexity of the software of the ground station, GUI, and has also taken a bit longer than anticipated.

8.3 References

List technical references and related work / market survey references. Do professional citation style (ex. IEEE).

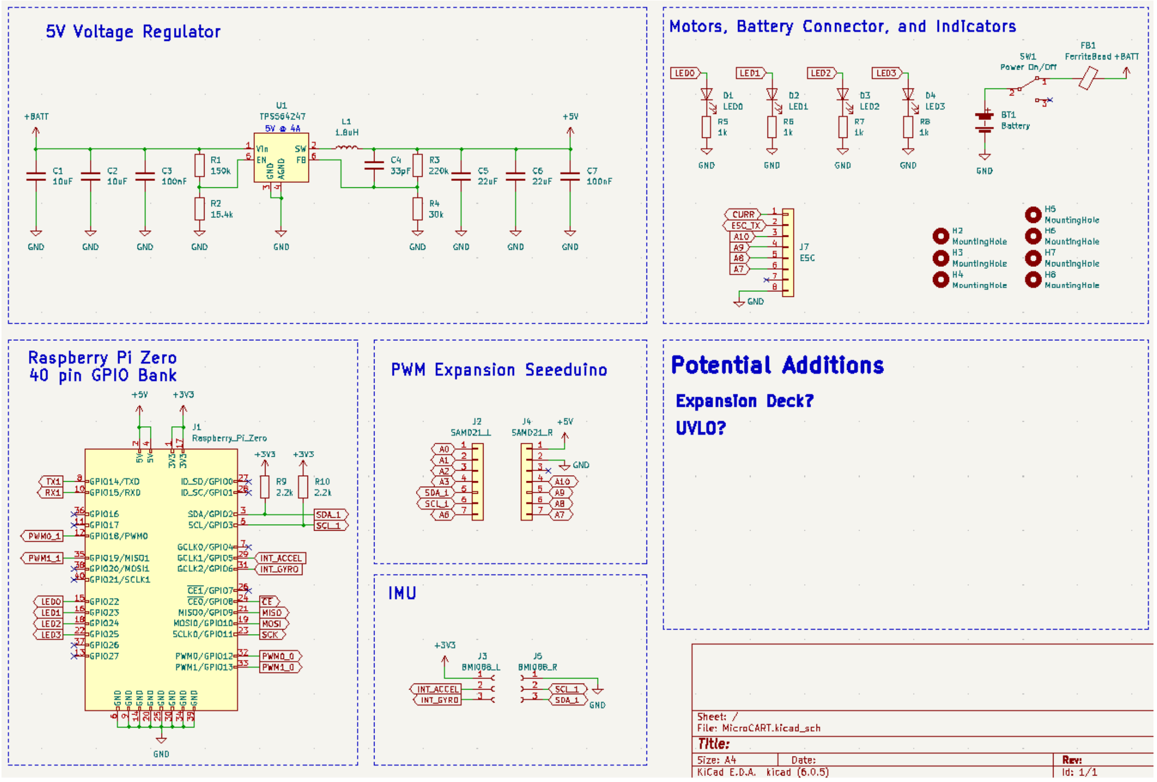
Schaub, H., & Junkins, J. L. (2018). *Analytical Mechanics of Space Systems*. American Institute of Aeronautics and Astronautics, Inc.

Rich, M. (2012). *Model Development, system identification, and control of a quadrotor helicopter* (thesis). Iowa State University Digital Repository.

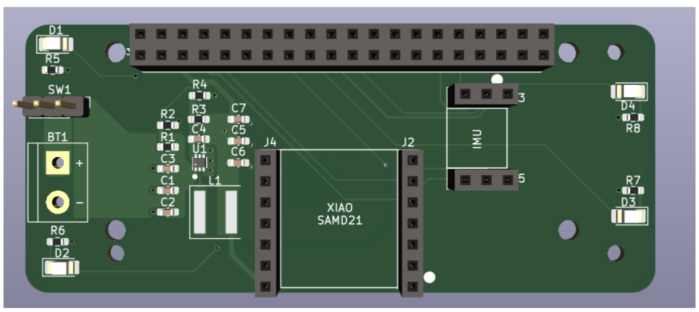
8.4 Appendices

Any additional information that would be helpful to the evaluation of your design document.

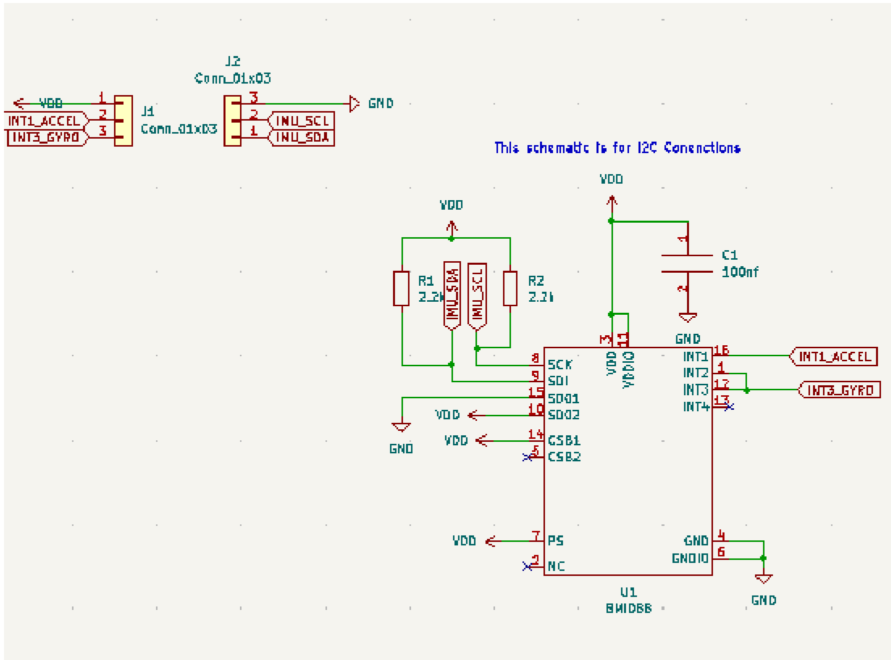
If you have any large graphs, tables, or similar data that does not directly pertain to the problem but helps support it, include it here. This would also be a good area to include hardware/software manuals used. May include CAD files, circuit schematics, layout etc,. PCB testing issues etc., Software bugs etc.



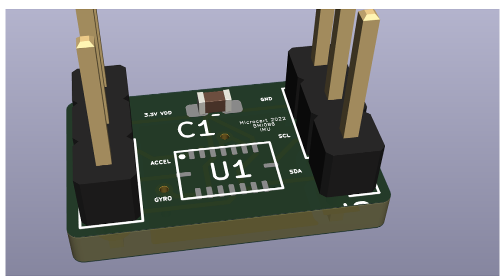
Breakout Board Schematic



Breakout Board 3D View



IMU Breakout Board Schematic



IMU Breakout 3D View

8.4.1 Team Contract

Team Members:

1) ___________Gautham Ajith________ 2) ________Connor Ryan ____________

3) ___________Cole Hunt ___________ 4) ________Tyler Johnson____________

5) ___________Austin Beinder _______ 6) ________Emily Anderson__________

7) ___________Grant Giansanti_______ 8) _______________________________

Team Procedures

1. Day, time, and location (face-to-face or virtual) for regular team meetings:
    a. Advisor Meeting: Weekly at 1pm on Fridays, Coover 3050
    b. In-Person Workdays: Thursdays 12-3PM, Coover 3050
    c. Virtual or In-Person Workdays: Sunday 1-4PM, Monday 9-12PM

2. Preferred method of communication updates, reminders, issues, and scheduling (e.g., e-mail, phone, app, face-to-face):
    a. Discord
    b. Email
3. Decision-making policy (e.g., consensus, majority vote):
    a. 5/7 vote for big decisions
    b. Majority of those present will make smaller decisions
4. Procedures for record keeping (i.e., who will keep meeting minutes, how will minutes be shared/archived):
    a. Meeting Minutes
        i. Scribe will be Connor Ryan
    b. Weekly Status Report
        i. Everyone will be expect to contribute to the status report

Participation Expectations

1. Expected individual attendance, punctuality, and participation at all team meetings:
    a. Everyone is expected to attend the advisor weekly meetings unless with a valid excuse
    b. If a meeting is missed it's up to you to review the meeting minutes of the missed meeting
2. Expected level of responsibility for fulfilling team assignments, timelines, and deadlines:
    a. Minimum of 3 hours/week, at least 2 hours should be completed during workdays
    b. Attendance to one hour of in-person workday is expected
3. Expected level of communication with other team members:
    a. Responses are expected with 1.5 business days
4. Expected level of commitment to team decisions and tasks:
    a. Team members should follow team decisions or express their disapprovement to the group.

Leadership

1. Leadership roles for each team member (e.g., team organization, client interaction, individual component design, testing, etc.):
    a. Project Manager - Connor Ryan
    b. Simulation/Controls Lead- Austin Beinder
    c. Telemetry/Backend Lead- Emily Anderson
    d. Client Interactions/Testing - Grant Giansanti
    e. Physical Systems Lead- Tyler Johnson
    f. Git Master/Device OS - Cole Hunt
    g. Lead Youtuber/GUI - Gautham Ajith
2. Strategies for supporting and guiding the work of all team members:
    a. Members take ownership of a particular part of the project and delegate work accordingly at weekly in-person meetings.
    b. Roadblocks should be communicated in the discord so the team can help support the issue.
3. Strategies for recognizing the contributions of all team members:
    a. Weekly contributions will be recorded in weekly status reports.

Collaboration and Inclusion

1. Describe the skills, expertise, and unique perspectives each team member brings to the team.

Team Member	Skills	Interests
Austin Beinder	Signals, simulation, hardware, software, motors, modeling	Controls, Motors
Cole Hunt	C\C++, Git	Embedded System, Physical Systems
Connor Ryan	High-speed design, leadership	RC
Emily Anderson	software	Flight systems
Gautham Ajith	Software, video editing	Embedded software, simulation
Grant Giansanti	C,VHDL,Management	Embedded System, Drones
Tyler Johnson	CAD,Verilog	FPGA, PCB, VLSI

2. Strategies for encouraging and supporting contributions and ideas from all team members:
    a. positive reinforcement
    b. having a inclusive environment
3. Procedures for identifying and resolving collaboration or inclusion issues (e.g., how will a team member inform the team that the team environment is obstructing their opportunity or ability to contribute?)
    a. A channel will be established in the discord where issues among the team can be expressed. If the issue can not be resolved within the team, Dr. Jones will be asked to mediate.

Goal-Setting, Planning, and Execution

1. Team goals for this semester:
    a. Update documentation of the 488 Lab and improve lab resources
    b. Develop a new quadcopter using pi-zero/zynqberry-zero microprocessor
    c. Expand on the capabilities established by previous MicroCART teams
2. Strategies for planning and assigning individual and team work:
    a. Communicate team members' workloads and tasks to be completed/need assigned during weekly meetings.
3. Strategies for keeping on task:
    a. Weekly Status Reports will establish a timelines and priorities for issues

Consequences for Not Adhering to Team Contract

1. How will you handle infractions of any of the obligations of this team contract?
    a. 1-2 infractions will be a will be a verbal warning
    b. On the 3rd infraction we will communicate it as a team during the team meeting with the advisor
2. What will your team do if the infractions continue?
    a. Bring it up to Dr. Jones and Professor Fila to help mediate

****

a) I participated in formulating the standards, roles, and procedures as stated in this contract.

b) I understand that I am obligated to abide by these terms and conditions.

c) I understand that if I do not abide by these terms and conditions, I will suffer the consequences as stated in this contract.

1) ___Tyler Johnson________________________________ DATE ___9/15/2022________
2) ____Grant Giansanti ______________________________ DATE ___9/15/2022________
3) ____Gautham Ajith________________________________ DATE ___9/15/2022________
4) ____Emily Anderson_______________________________ DATE ___9/15/2022________
5) _____Connor Ryan ________________________________ DATE ___9/15/2022________
6) ______Cole Hunt__________________________________ DATE ___9/22/2022________
7) ______Austin Beinder______________________________ DATE ___9/22/2022________
8) ____ DATE ____________________