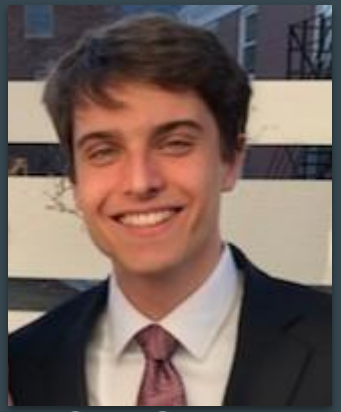
Microcontroller Controlled Aerial Robotics Team (MicroCART)

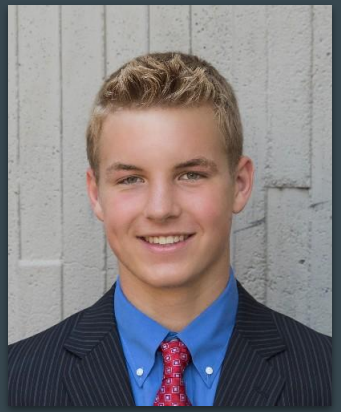
SDMY 23-45
Tyler Johnson, Connor Ryan, Cole Hunt, Austin Beinder,
Emily Anderson, Grant Giansanti, and Gautham Ajith
Advisor - Dr. Jones

Team 45



Grant Giansanti
User Interaction/Testing

Tyler Johnson
Physical Systems Lead

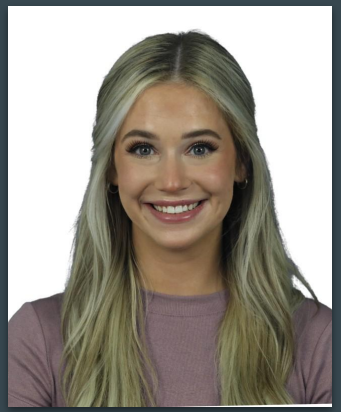


Connor Ryan
Project Manager

Austin Beinder
Simulation/Controls Lead

Gautham Ajith
YouTube/GUI Lead

Cole Hunt
Git Master/Device OS



Emily Anderson
Backend/Telemetry Lead

Agenda

- Overview of the MICROCART team
- Conceptual and Lab Overview
- Project Requirements and expected outcomes
- Proposed project plan
- Testing plan
- Conclusions

Growing need for Robotics and Drones at Iowa State

- MicroCART Project has been around since 1998.
- Previous designs
  - Lack of adaptability
  - Difficult GUI
  - Bad Software Design

- Iowa State Classes
  - CPRE 488 Embedded System Design
  - EE 476 Control System Simulation
- Research
  - Controls
  - Simulation to physical devices

Team 2009

Team 2015

Team 2021 *crazyflie

We are trying to create a quadcopter that can be easily adaptable and accessible for ISU students and researchers

This Years Vision

Our team will continue to increase the accessibility of quadcopter resources available at Iowa State University through the following:

- Develop quadcopter for control theory research requiring multi-core functionality
- Improve existing lab materials and documentation for CPRE488: Embedded System Design
- Create demonstration for potential students utilizing MATLAB quadcopter simulation
- YouTube tutorials explaining to undergraduates the toolsets and processes related to our quadcopters

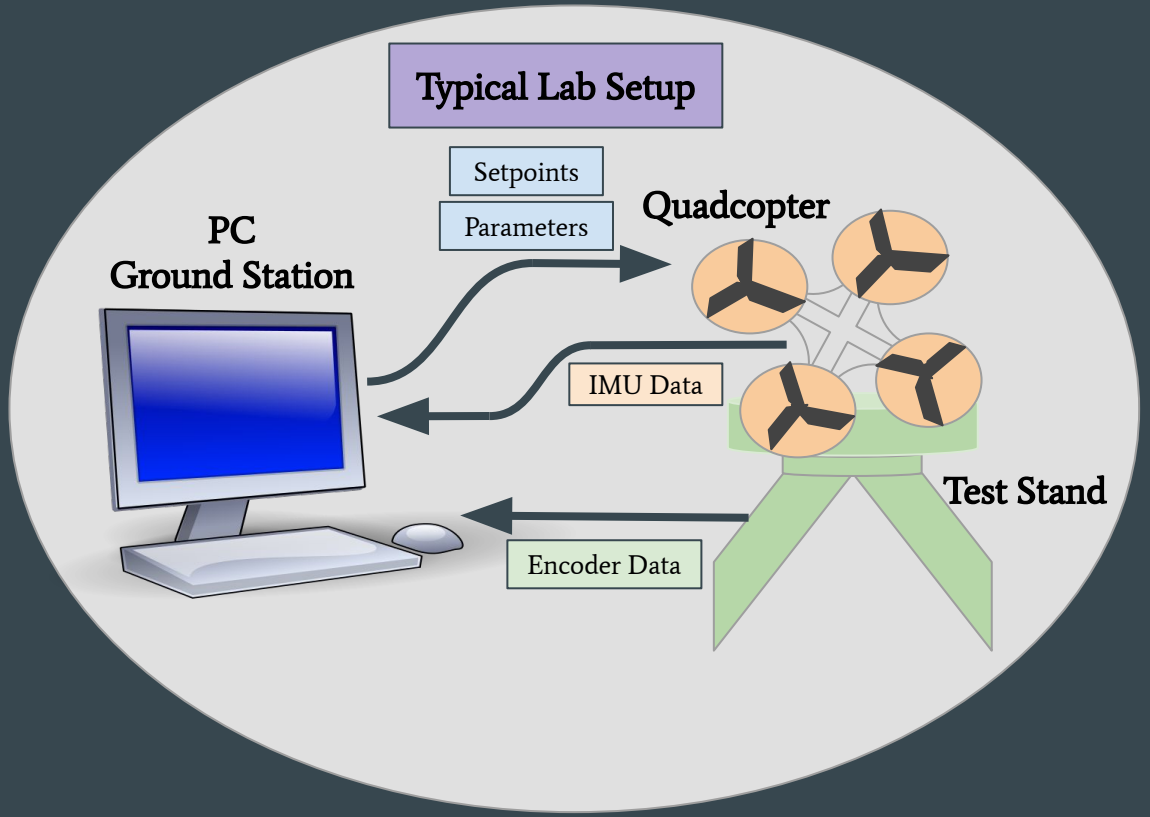
Conceptual Overview

Quadcopter: Runs flight controller software and logs data to PC.

Ground Station: Application communicating with quadcopter, sending commands, and displaying data.

Test Stand: Different source of rotational acceleration/velocity data

Simulation: Will simulate the quadcopters behavior for test flights



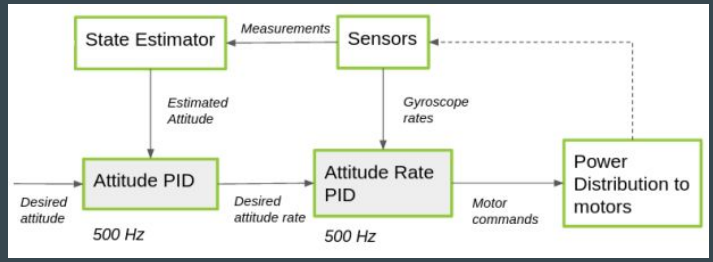
CPRE488 Lab

Overview

- Quadcopter uses a cascading PID flight controller
- First, students tune the PID values
- Next, students implement their own PID flight controller software

Feedback from students

- Connection to ground station is unreliable
- Test Stand communicates poorly with GUI
- UI is not always intuitive



Cascading PID Diagram, *sdmay22-43*

Requirements

- Functional Requirements
  - Drone: Communication latency of max 100 ms
  - Drone: 5 second stable hover
- Resource Requirements
  - Drone: Bluetooth Low Energy or the Wifi Network Cards
  - Labs: Flight Control Software on Lab PCs
- Physical Requirements
  - Drone: Size - 125mm x 125mm | Weight - 300g | Balanced COM
- User Experiential Requirements
  - Labs: Documentation for the Hardware and Software senior level engineering students
  - Labs: GUI needs simple navigation for interfacing with the Drone
  - The GUI should provide reliable performance with reduced errors and error logs
- Environmental Requirements
  - Drone: Safe for indoor use
  - Drone: Student safety in lab with no extreme injury incidents

Design Approach

- Waterfall/Aglie
- Brainstorming and Constraints
- Comparison Tables
- Git Tasks

Part Selection

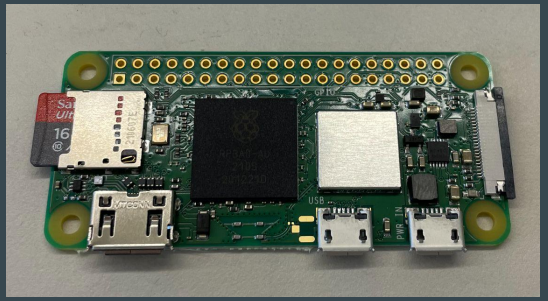
Field	IMU 1	IMU 2	IMU 3
Type	Accelerometer, Gyroscope, Temperature, 6 Axis	Accelerometer, Gyroscope, 6 Axis	Accelerometer, Gyroscope, Magnetometer, 9 Axis
Comm Busses	I²C, SPI	I²C, SPI	I²C, SPI, UART
Accelerometer Refresh Rate	12.5-1600Hz	1.6kHz	500 Hz
Accel Sensitivity	±2/±4/±8/±16 g full scale	+- 24 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

Our Solution: Hardware

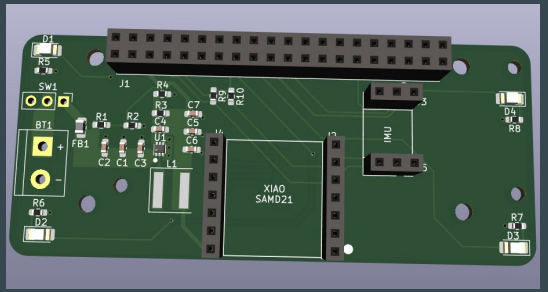
- Raspberry Pi Zero 2W for increased computing power and multi-core

- PCB interfaces processor with quadcopter hardware (IMU, ESC/Motors, Expansion Decks, etc.)

- Brushless motors used for improved performance



Raspberry Pi Zero 2W



PCB 3D Rendering

Our Solution: Software

- Firmware: Bare-metal Application and ROS
- Support integration with simulation environment
- Improved Ground Station control software and GUI

Prototype/Implementation

- The main parts of the quadcopter have been selected and ordered
- First PCB revision has been ordered and assembled
- Test software has been created to interface with the motors

Challenges

Technical Problems:

- Learning new Software(GIT and QT)
- Redesign board because of space constraints.
- Current MicroCART Software contains a lot of bugs

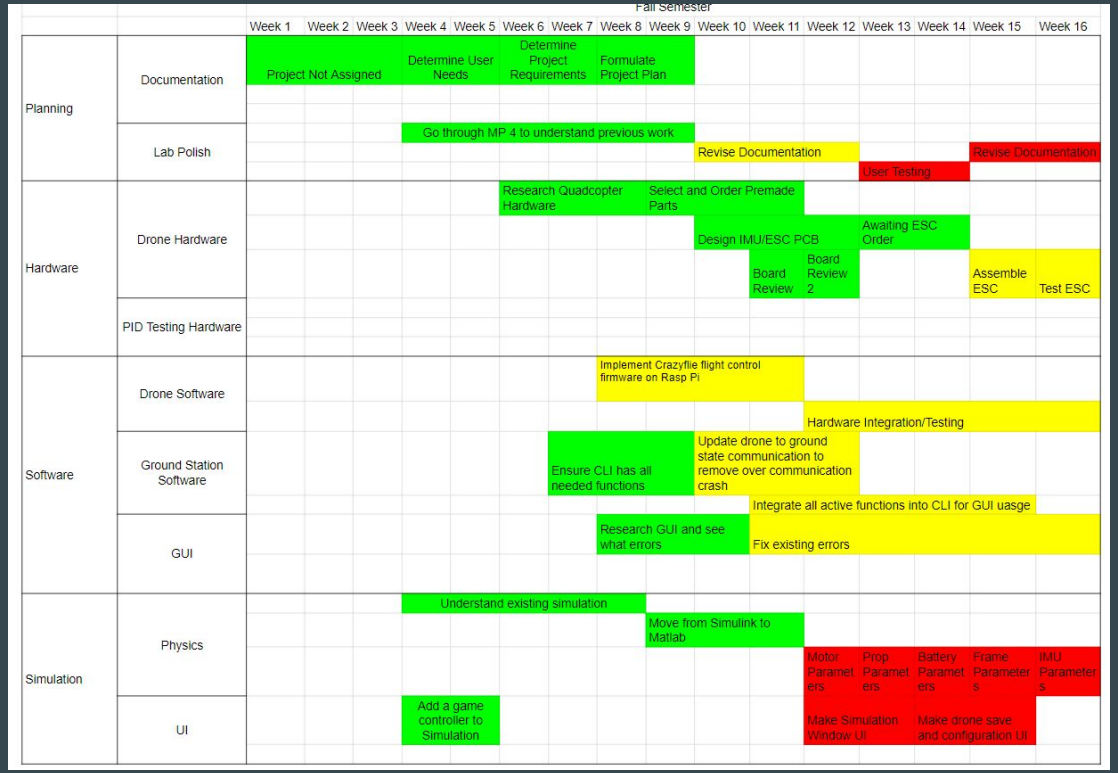
Non-Technical:

- Project management
- Time Management
- Organization
- Supply chain issue

Project Plan-Schedule

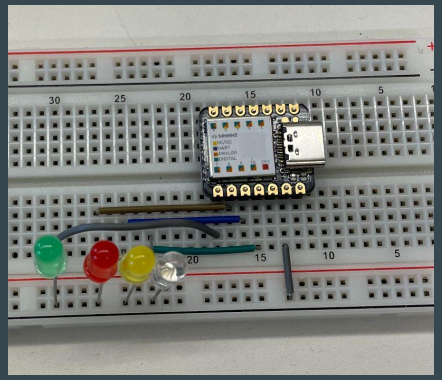
Done In progress Not Complete

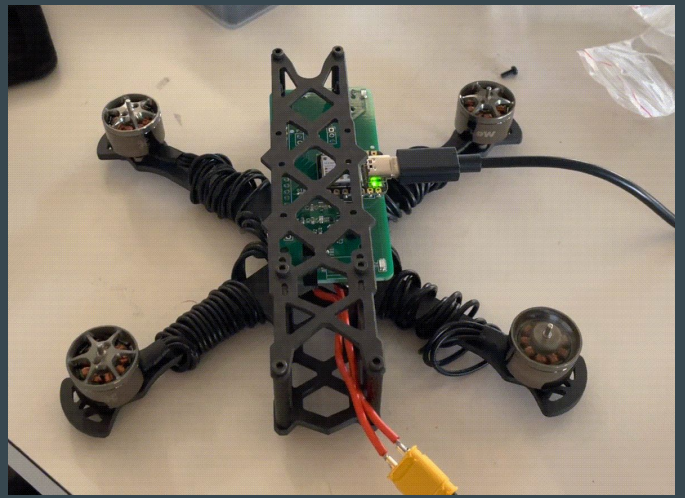
- Working through MP4
- PCB has been ordered
- Still continuing to fix and improve the GUI



Testing Plan

- Unit Testing
  - PWM expansion with LEDs
- Interface Testing
- Integration Testing
- System Testing
  - PWM expansion with motors on the frame
- Regression Testing
- Acceptance Testing





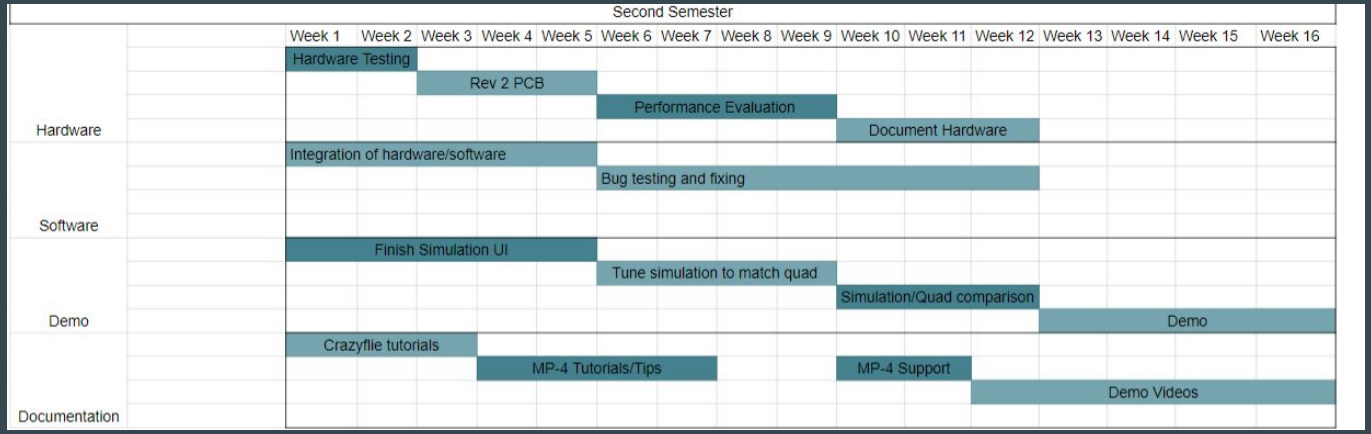
Contributions

Significant Individual Contributions for Fall 2022

Team Member	Contributions	Total Hours
Austin Beinder	• Converted a lot of the simulation to matlab • Specced out a lot of the parts we ended up using	66
Cole Hunt	• Updated crazyflie firmware for flight testing • Setup Raspberry Pi Zero W with environment for hardware testing	37
Connor Ryan	• Designed Main PCB • Evaluated parts and frames • Website	54
Emily Anderson	• Worked primarily on learning GUI software in order to make functionality improvements and diagnose current bugs	38
Gautham Ajith	• Converted the Actuation to MATLAB • Setting up documentation process for GUI issues • Helped around related to Design Documentation	37
Grant Giansanti	• Designed and tested PWM code on PWM expansion module • Setup a headless ubuntu image on pi zero 2w • Examined ways to reverse engineer crazyflie software to raspberry pi	40
Tyler Johnson	• Designed IMU Breakout Board • Helped a lot with Design Documentation	41

Project Roadmap

- We are currently slightly behind in our schedule
- Next semester plans:
  - Continue fixing bugs GUI
  - Test drone
  - REV 2 of PCB
  - Match simulation to our quad



Questions?

- We are currently slightly behind in our schedule
- Next semester plans:
  - Continue fixing bugs GUI
  - Test drone
  - REV 2 of PCB
  - Match simulation to our quad