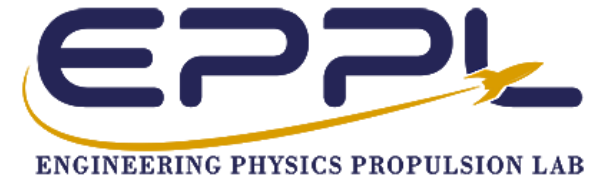


CubeSat Control Platform + ACTIV

10/17 Meeting

EMBRY-RIDDLE
Aeronautical University



[Justin]: Working on

Progress completed this past week

-Software:

EulerAngleAcquisition class for all platforms

Learning a lot about OOP in Python

Creating PID controllers utilizing OOP to control pendulum

.

Goals for next week

Choose motors for ACTIV pending Drakunov's advice

Create program to control pendulum in compartmentalized fashion (OOP)

.

.

Anticipated Challenge

Making progress remotely

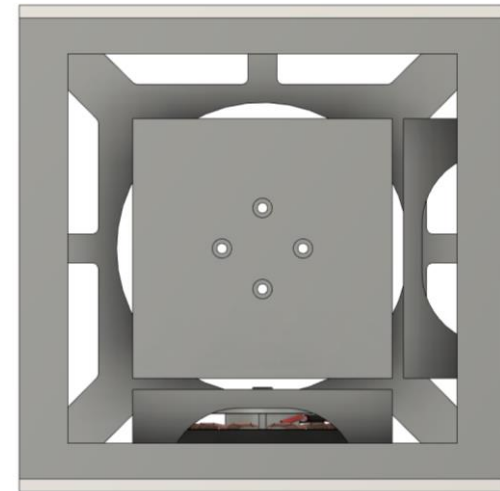
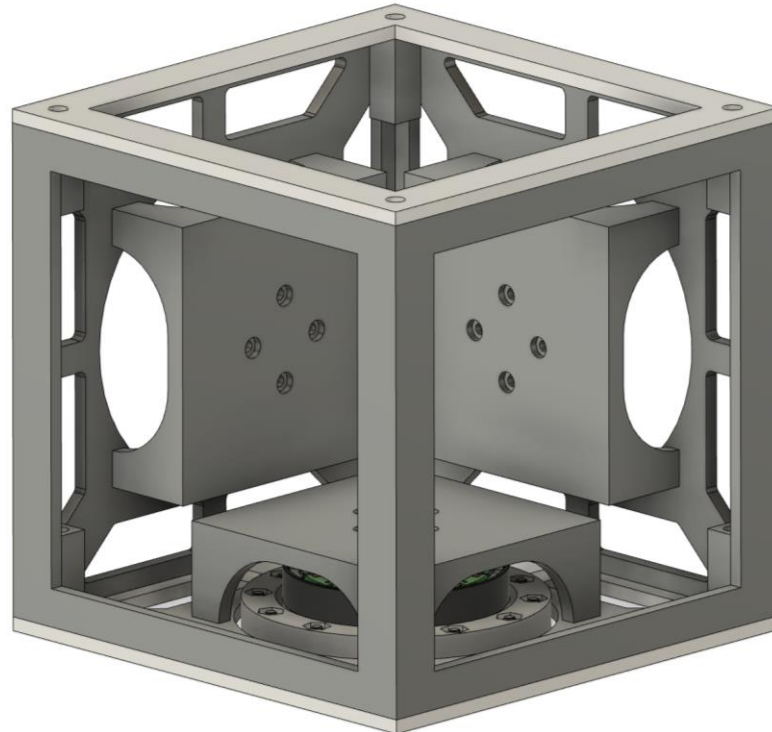
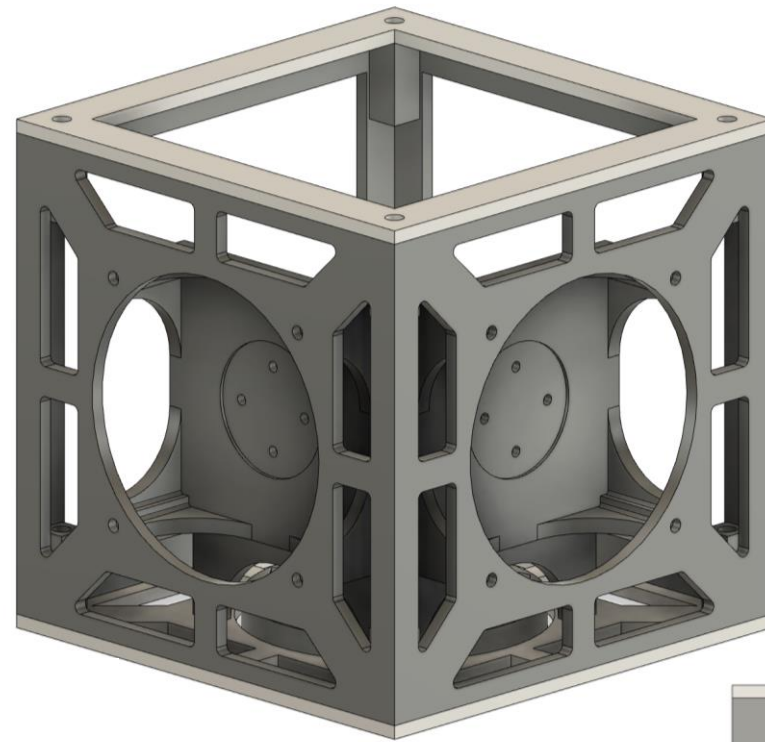
[Isaac] 2DOF and 3DOF CubeSats

Progress completed this past week

- Finished pizza table
- Fixed the mounts for three main sides on cubesat body for new pizza table

Goals for next week

Anticipated challenges



[Ryan] + Software Refactoring

Progress completed this past week

- Decided on a structure for software library

Goals for next week

- Get basic functionality of all critical classes for orientation software and incorporate into main control program

Anticipated challenges

- Inexperience may lead to incomplete structure at first

```
SatelliteSystem/  
├── SatelliteOrientationEstimator/  
│   ├── data_collection/  
│   ├── orientation_estimation/  
│   ├── filtering/  
│   ├── storage/  
│   ├── user_interaction/  
│   └── __init__.py  
├── MotorControl/  
│   └── ...  
├── Controller/  
│   └── ...  
└── main.py
```

```
SatelliteOrientationEstimator/  
├── data_collection/  
│   ├── __init__.py  
│   ├── sensor.py  
│   ├── lsm9ds1.py  
│   └── ...  
│   # Abstract base class for sensors  
│   # Specific implementation for LSM9DS1  
│   # Other sensor implementations as needed  
├── orientation_estimation/  
│   ├── __init__.py  
│   ├── estimator.py  
│   ├── quest_estimator.py  
│   ├── madgwick_estimator.py  
│   └── ...  
│   # Base class for orientation estimators  
│   # QuEST algorithm implementation  
│   # Madgwick filter implementation  
│   # Other estimators as needed  
├── filtering/  
│   ├── __init__.py  
│   └── filters.py  
│   # Different filtering methods  
├── storage/  
│   ├── __init__.py  
│   └── database.py  
│   # SQLite database functions and operations  
├── user_interaction/  
│   ├── __init__.py  
│   └── cli.py  
│   # Command line interface methods  
└── main.py  
    # Main execution script
```

[Dylan] + [Inverted Pendulum Integration]

Progress completed this past week

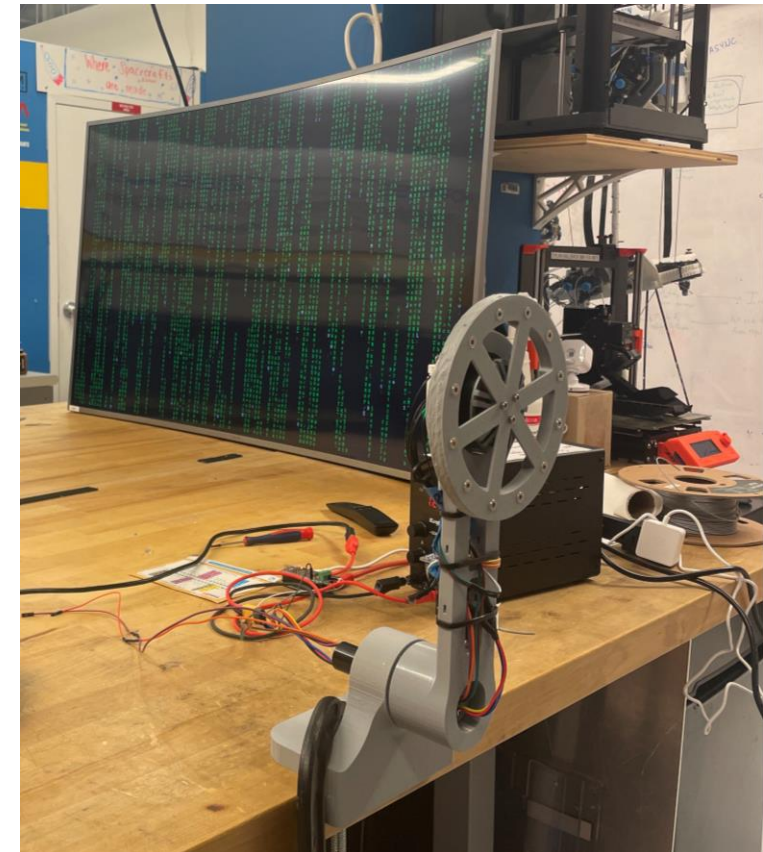
- 3D printed and Assembled Inverted Pendulum
- O-Drive Control using CAN
 - Velocity Control
 - Reading Encoder Position & Velocity
- Reading reliable IMU Roll Data on Inverted Pendulum
- Was able to get all the following running at once using Threading on the Pi
 - Velocity Motor Commands
 - Reading Encoder Position & Velocity
 - Reading IMU Roll Data (Issue with speed of imu data)
 - Printing all to Terminal

Goals for next week

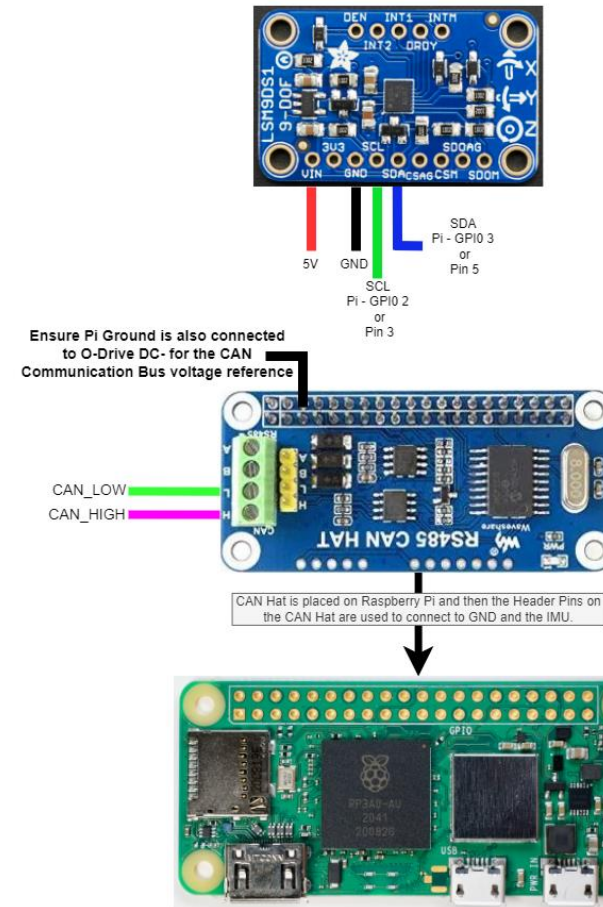
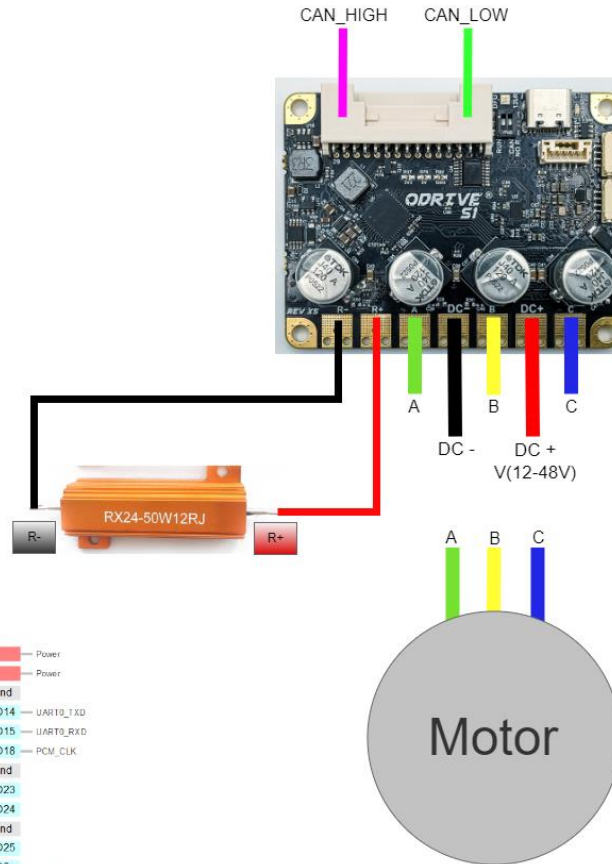
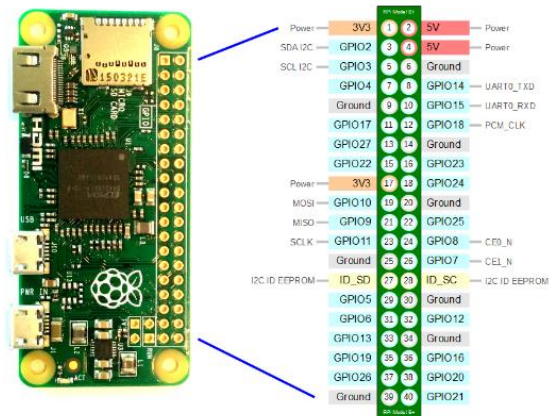
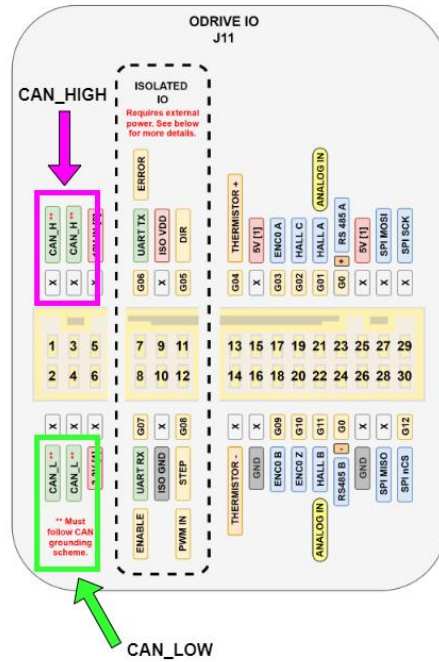
- Get CAN Control
 - Torque Control
 - Controlling multiple motors at once
- Create Database to store
 - IMU data
 - Motor Data
- Create and Implement simple PID on Inverted Pendulum with Justin

Anticipated challenges

- Going home for fall break and will not be able to work :(

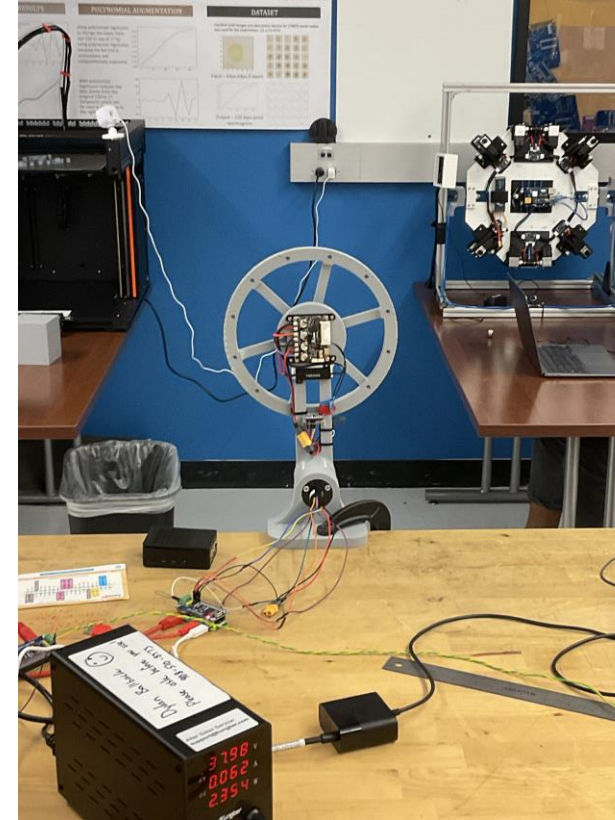
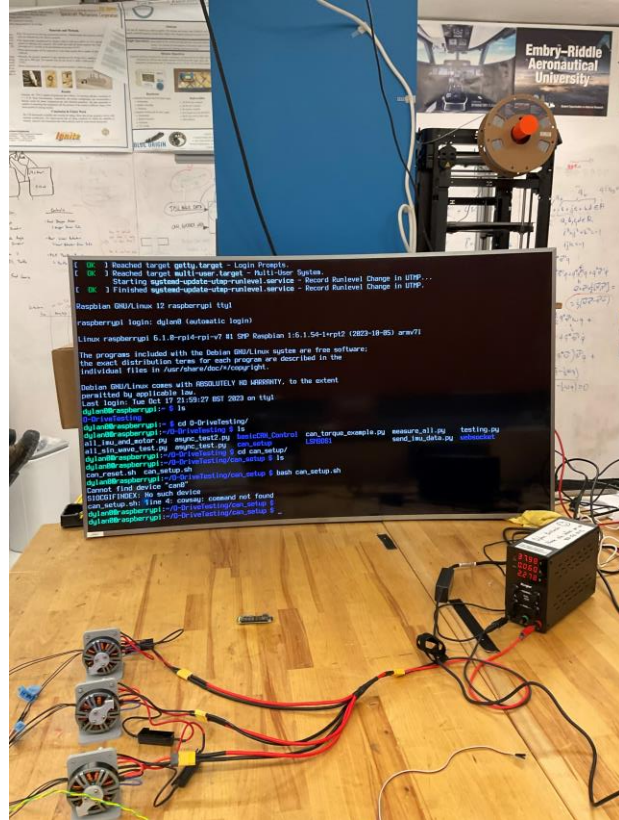


Inverted Pendulum Electrical Wiring Diagram



By: Dylan Ballback
Rev 1: 10/16/23

Visual System Updates



[Assignee] + [Task Title]

Progress completed this past week

-[Discuss progress]

-Highlight based on: Complete, >50%, <50%

Goals for next week

-[Discuss goals]

Anticipated challenges

-[discuss challenges, request assistance if needed]



[Relevant photos if needed]