CubeSat Control Platform + ACTIV

11/28 Meeting

EMBRY-RIDDLE Aeronautical University

EasyControls.org

EPPL ENGINEERING PHYSICS PROPULSION LAB

General Updates/Reminders

Coursework is wild right now, so lab productivity will decrease

However, the following is a list of open-items for the various projects

ENGINEERING PHYSICS PROPULSION LAB

2

CubeSat

- Continuing expanded 1U development
  - Determining electronics placement
  - Setting CubeSat up for multi-motor testing
- Print gimbal rings
  - Design gimbal ring locking mechanism
- Configure for ACTIV 1-DOF testing
- Refactor O-Drive Can control code
- Setup database for CubeSat trials

ENGINEERING PHYSICS PROPULSION LAB

[Justin]: Database integration, Fusion management, and more

Progress completed this past week

Goals for next week

Anticipated Challenges

ENGINEERING PHYSICS PROPULSION LAB

[Isaac] +[2DOF/3DOF Cube Sats]

Progress completed this past week

-made CAD for electronics panel and large resistors

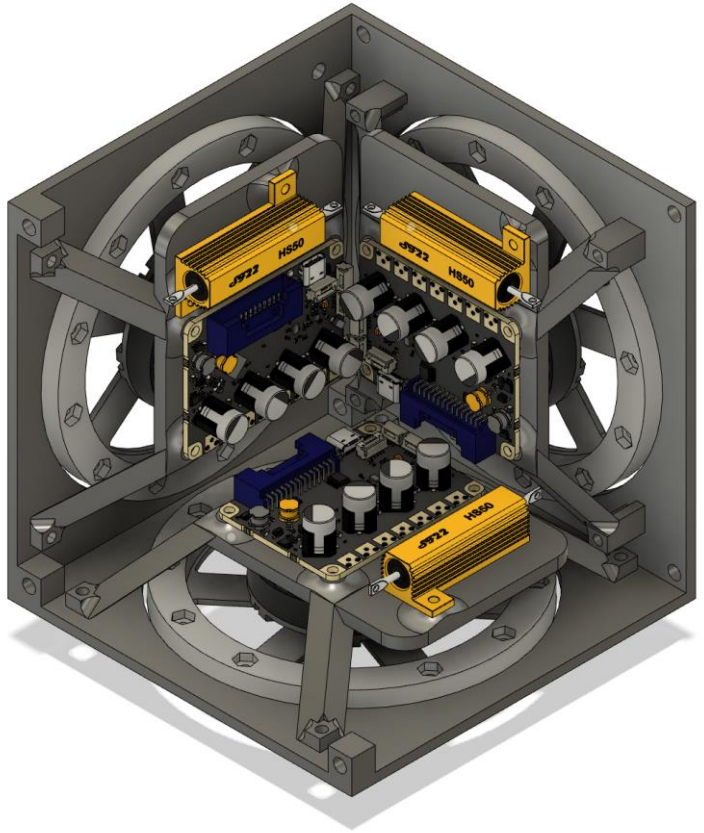
-got cubesat printed

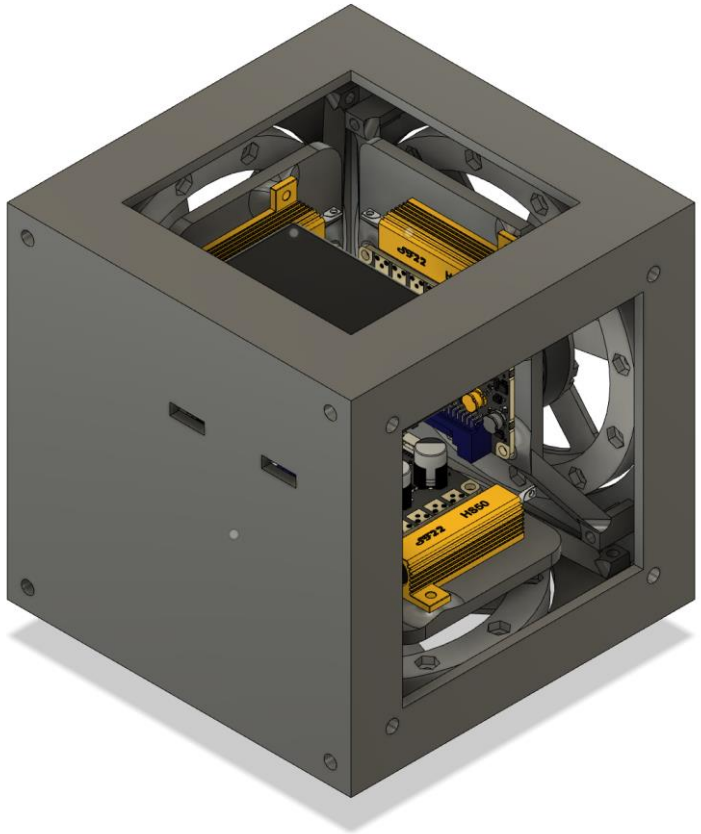
Goals for next week

-assembling the cubesat

Anticipated challenges

-disassembling the old cubesat





ENGINEERING PHYSICS PROPULSION LAB

[Ella] + [The Clawwwww]

Progress completed this past week

-Claw for 1.5U CubeSat

Goals for next week

-SRS and Trep (I hope I can make it)

Anticipated challenges

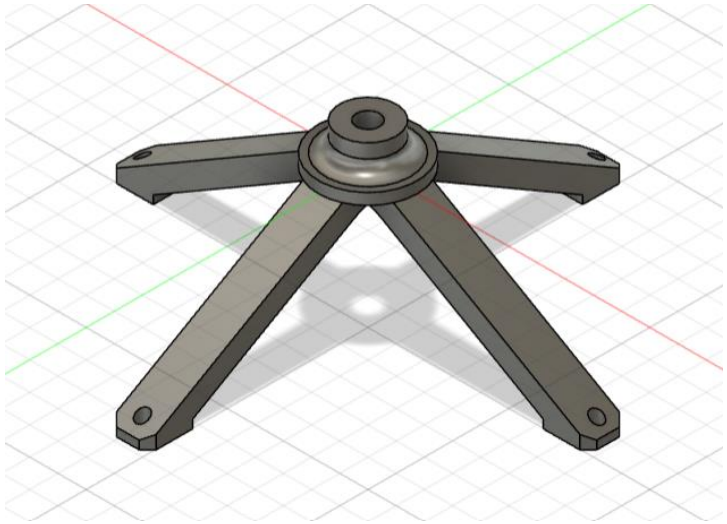
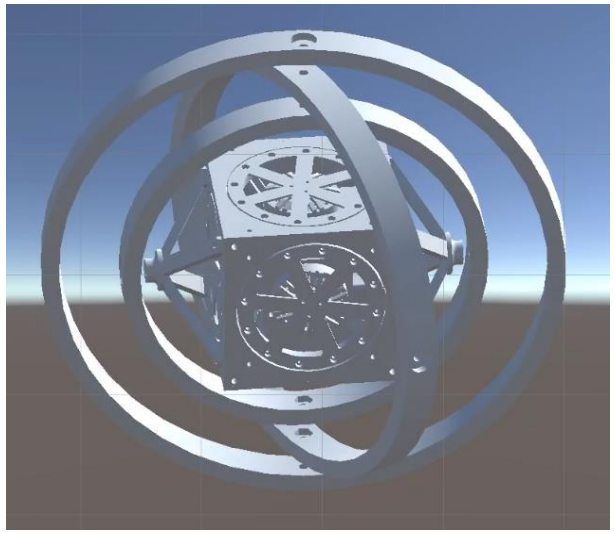


Image stolen from
Dylans cool video



ENGINEERING PHYSICS PROPULSION LAB

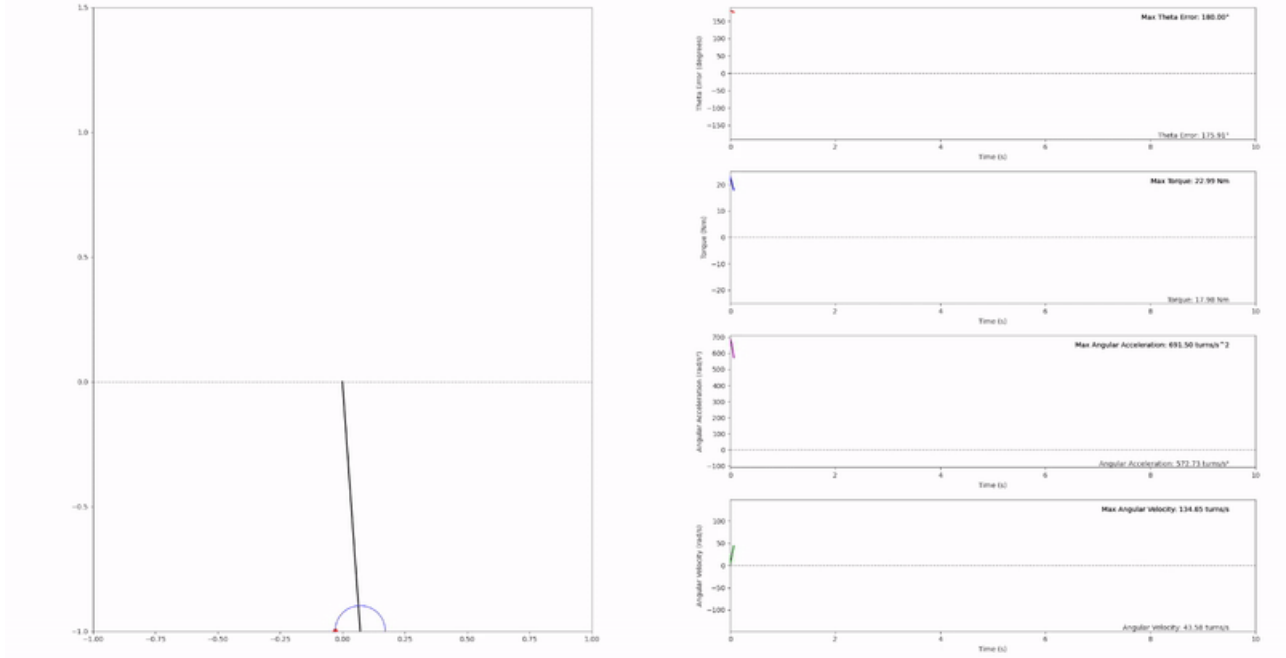
[Dylan] + [Simulations!]

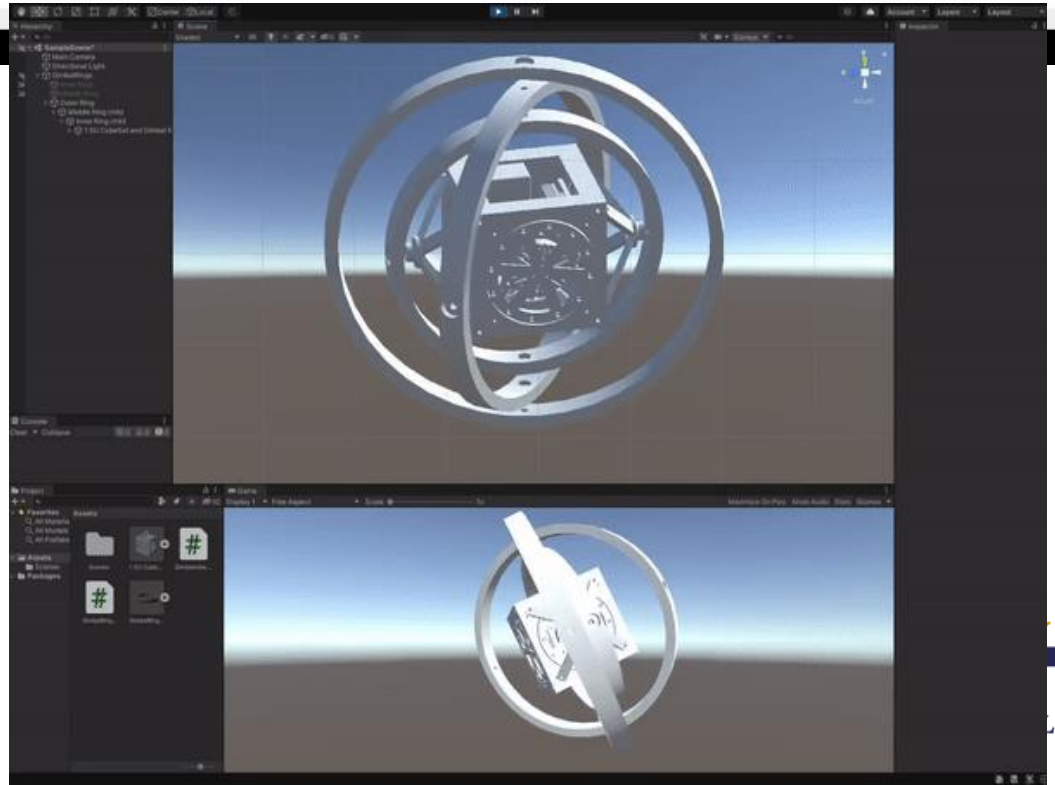
Progress completed this past week

- 3D print and assembly ISSAC 3DoF CubeSat
- TREP expo poster / SRS poster
- Worked on IP python simulation
- Build simple Unity sim with 3 gimbal rings & cubesat

Goals for next week

- ~~Implement simple PID on Inverted Pendulum w/ Justin~~
  - ~~Pending new Inverted Pendulum Arm~~
- ~~Get preliminary design for 3DoF CubeSat Electronics~~
- ~~Try and get 3D print going of the new gimbal rings~~
- ~~Want to work more on livestreaming data with WebSocket~~
- Going to Vegas on Thursday for F1 Race then flying home for thanksgiving.
- Will work more on python simulation while I am gone





[Jacob S] + [Optical Attitude Determination]

Progress completed this past week

- Became familiar with camera calibration for CV. Got some good insight into MQTT over the break and became more familiar working with the Raspberry Pi.

Goals for next week

- Survive.
- Get the camera calibration software working with the pi camera.

Anticipated challenges

- The latest edition of the aruco library has changed a lot of their functions.

COPY

EasyControls.org

Dylan Ballback, Justin Hartland, Ella Cheatham, Adam Duke, Jacob Romeo, Jacob Salazar, Isaac Stitt, Ryan Taylor, Vishal Ramisetty, Vishwam Rathod, Jacob Raynor, Kyle Fox

COMPANY PURPOSE

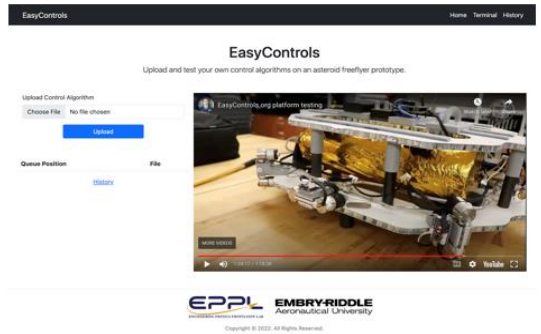
EasyControls.com allows for students and researchers to accelerate and expand their knowledge of control systems and algorithms. This educational space gives people the ability to complete hardware-in-the-loop testing in real-time on physical hardware through the internet.

PROBLEM

- Learning Controls can be overwhelming with complex mathematics and equations, and it is very hard to visualize and connect the dots between controlling a system in real life and the abstract mathematical equations/formulas.

SOLUTION

- EasyControls.org will allow for anyone online can upload their own controls algorithm and watch a live stream of how their algorithm performs on hardware in real-time.

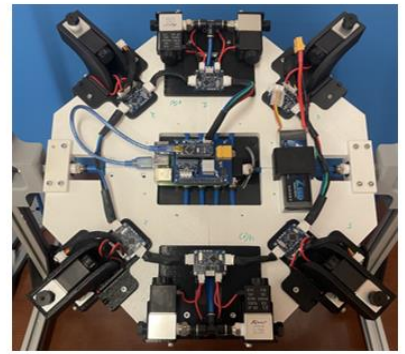


PRODUCT

- Monetizing the use of EasyControls.org
  - Able to charge $/min of runtime

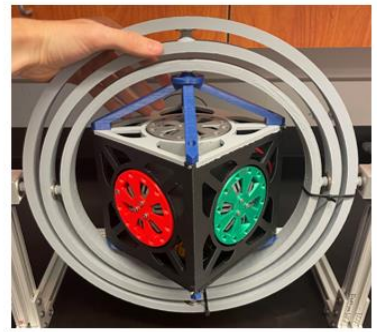
- Educational Kits (Raw Material Cost)

Kit Type	Raw Material:	Sale Price	Profit
SpaceCraft	$ 1,200.00	$ 2,000.00	$ 800.00
CubeSat	$ 900.00	$ 1,500.00	$ 600.00
Inverted Pendulum	$ 400.00	$ 600.00	$ 200.00
Totals	$ 2,500.00	$ 4,100.00	$ 1,600.00

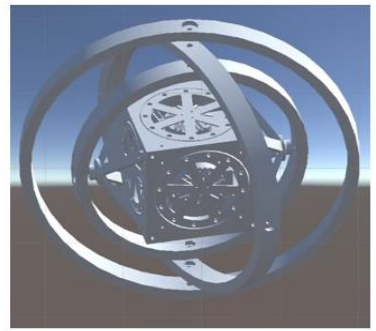


1DoF Spacecraft Prototype

Inverted Pendulum Prototype



1U 3DoF CubeSat Prototype



1U 3DoF CubeSat Prototype

Join us at Easy Controls, and let's accelerate space technology education together. Because when it comes to reaching for the stars, the only limit should be the breadth of your imagination, not access to cutting-edge educational resources.

MARKET POTENTIAL/COMPETITION

- Controls professors being able to use this as a teaching aid while in class or for homework/project assignments.
- Able to sell educational controls courses with a focus on using EasyControl.org to learn.
- Existing educational controls platforms on the market:
  - Balancing Robot Kit ($85)
  - HiWonder Robotic Arm and Car ($600)
  - Quanser Inverted Pendulum ($5000)
  - EyasSat Classroom Satellite ($12350)

Attitude Control Testbed In Vacuum

Justin Hartland, Dylan Ballback, Ella Cheatham, Jacob Romeo, Jacob Salazar, Isaac Stitt, Ryan Taylor, Vishal Ramisetty, Vishwam Rathod

ABSTRACT

Attitude Control Testbed in Vacuum (ACTIV) will be designed to simulate a microgravity environment with the use of 3 gimbal rings with embedded electric motors in each rotational axis to provide an opposing torque to those produced by gravity and friction.

This controlled gyroscope will be designed to test spacecraft ranging from 1U to 6U (as supplied by the CubeSat Control Platform project) in size so that varying spacecraft designs may be tested.

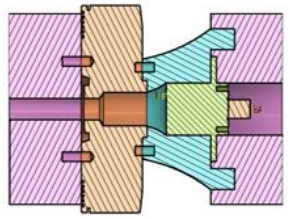
ACTIV

General System Architecture

Comprised of 3 gimbal rings, each ring will be mounted on a high-torque, hollow-shaft R80 motor. Shafts will be built into the rings to allow wires to pass. An external electronics bay will house the power source, the Raspberry Pi, ODrive controllers, and other electronics.

Rotary Union Point

Electrical continuity throughout rings will be accomplished using hollow-shaft motors in conjunction with slip rings.



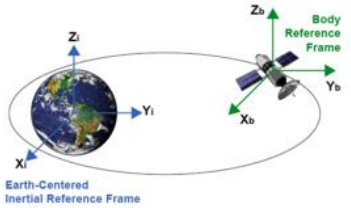
ACTIV method to maintain continuity with R80 motor and slip ring

6U CubeSat 3DOF Stability in ACTIV

PROJECT GOALS

Develop 3 degree of freedom, actively driven gimbal ring system to counteract gravitational and friction torques produced microgravity effect.

Develop an inverted pendulum, and 1, 3, and 6U CubeSat Control Platforms all of which utilize reaction wheels to change attitude.



EASY CONTROLS

EasyControls.org allows for students and researchers to accelerate and expand their knowledge of control systems and algorithms by allowing hardware-in-the-loop testing in real-time on physical hardware through the internet.

ACTIV, along with the inverted pendulum and CubeSat platforms, will be integrated into the website.

CURRENT STATE

General

Achieved multi-motor control using CAN; to be applied to both ACTIV and CubeSat platforms.

Reporting IMU data, sending motor commands, and storing data in SQ Lite database using threading.

Inverted Pendulum

Designed and assembled 1 degree of freedom inverted pendulum; currently tuning PID controller.

CubeSat

Assembled 1 degree of freedom CubeSat with ODrive controller, serving as testbed for IMU yaw drift mitigation.

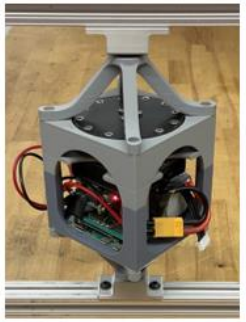
Iterating upon 3 degree of freedom CubeSat design.

ACTIV

Torque estimations determined ideal motor which has been used to design rotary point.

Manufacturing 1 degree of freedom configuration for initial testing prior to manufacturing full system.

Multi-motor control by utilizing CAN



1 DoF CubeSat with ODrive motor controller

Upright inverted pendulum

O-Drive CAN Commands

CMD ID	Name	Sender	Start Byte	Name	Type	Description
0x000	Get_Version	ODrive_Axis0	0	Protocol_Version	uint8	Always reported as 2
			1	Hw_Version_Major	uint8	hw_version_major
			2	Hw_Version_Minor	uint8	hw_version_minor
			3	Hw_Version_Variant	uint8	hw_version_variant
			4	Fw_Version_Major	uint8	fw_version_major
			5	Fw_Version_Minor	uint8	fw_version_minor
			6	Fw_Version_Revision	uint8	fw_version_revision
			7	Fw_Version_Unreleased	uint8	fw_version_unreleased
0x001	Heartbeat	ODrive_Axis0	0	Axis_Error	uinit32	<axis>.active_errors
						<axis>.disarm_reason
			4	Axis_State	uint8	<axis>.current_state
			5	Procedure_Result	uint8	<axis>.procedure_result
			6	Trajectory_Done_Flag	uint8	<axis>.controller.trajectory_done (0: Flase, 1: True)
0x002	Estop	Master	Empty Payload	-	-	ESTOP_REQUESTED
0x003	Get_Error	ODrive_Axis0	0	Active_Errors	uinit32	active_errors
			4	Disarm_Reason	uinit32	disarm_reason
0x004	RxSdo	None	0	Opcode	uint8	0: Read, 1: Write
			1	Endpoint_ID	uint16	Endpoint ID as found in flat_endpoints.json
			3	Reserved	uint8	-
			4	Value	uinit32	Data type and length depend on endpoint ID
0x005	TxSdo	None	0	Reserved0	uint8	-
			1	Endpoint_ID	uint16	Endpoint ID as found in flat_endpoints.json
			3	Reserved1	uint8	-
			4	Value	uinit32	Data type and length depend on endpoint ID
0x006	Set_Axis_Node_ID	Master	0	Axis_Node_ID	uinit32	node_id
0x007	Set_Axis_State	Master	0	0	uinit32	requested_state
0x009	Get_Encoder_Estimates	ODrive_Axis0	0	Pos_Estimate	float32	<axis>.pos_vel_mapper.pos_rel <axis>.pos_vel_mapper.pos_abs Depends on: ODrive.Controller.Config.absolute_setpoints
			4	Vel_Estimate	float32	<axis>.pos_vel_mapper.vel
0x00b	Set_Controller_Mode	Master	0	Control_Mode	uinit32	control_mode
			4	Input_Mode	uinit32	input_mode
0x00c	Set_Input_Pos	Master	0	Input_Pos	float32	input_pos
			4	Vel_FF	uint16	input_vel
			6	Torque_FF	uint16	input_torque
0x00d	Set_Input_Vel	Master	0	Input_Vel	float32	input_vel
			4	Input_Torque_FF	float32	input_torque
0x00e	Set_Input_Torque	Master	0	Input_Torque	float32	input_torque
0x00f	Set_Limits	Master	0	Velocity_Limit	float32	vel_limit
			4	Current_Limit	float32	current_soft_max
0x011	Set_Traj_Vel_Limit	Master	0	Traj_Vel_Limit	float32	vel_limit
0x012	Set_Traj_Accel_Limits	Master	0	Traj_Accel_Limit	float32	accel_limit
			4	Traj_Decel_Limit	float32	decel_limit
0x013	Set_Traj_Inertia	Master	0	Traj_Inertia	float32	inertia
0x014	Get_Iq	ODrive_Axis0	0	Iq_Setpoint	float32	Iq_setpoint
			4	Iq_Measured	float32	Iq_measured
0x015	Get_Temperature	ODrive_Axis0	0	FET_Temperature	float32	<axis>.motor.fet_thermistor.temperature
			4	Motor_Temperature	float32	<axis>.motor.motor_thermistor.temperature
0x016	Reboot	Master	Empty Payload	-	-	reboot()
0x017	Get_Bus_Voltage_Current	ODrive_Axis0	0	Bus_Voltage	float32	vbus_voltage
			4	Bus_Current	float32	Ibus
0x018	Clear_Errors	Master	Empty Payload	-	-	clear_errors()
0x019	Set_Absolute_Position	Master	0	Position	float32	set_abs_pos()
0x01a	Set_Pos_Gain	Master	0	Pos_Gain	float32	pos_gain
0x01b	Set_Vel_Gains	Master	0	Vel_Gain	float32	vel_gain
			4	Vel_Integrator_Gain	float32	vel_integrator_gain
0x01c	Get_Torques	ODrive_Axis0	0	Torque_Target	float32	effective_torque_setpoint
			4	Torque_Estimate	float32	torque_estimate

O-Drive CAN Python Class

Started on the code layout:

- Maybe need some help with this code!

```python
import board
import can


class ODriveCAN:
    """
    A class for setting up O-Drive motor controllers using CAN comunincation

    Attributes:
        Specifically for setting up CAN comunication between Raspberry Pi and CAN Communication Type:
            canBusID (String): Can Bus ID should be default "can0" but if you have muilitiple can buses
            on your device you can modify here

            canBusType (String): python-can package CAN communication type we by default us "socketcan"

        O-Drive Controller Specific Attributes:
        nodeID (integer): The node ID can be set by the
    """
    def __init__(self, canBusID, canBusType, nodeID):
        self.canBusID = canBusID
        self.canBusType = canBusType
        self.nodeID = nodeID


    def setAxisNodeID(self):
        """
        Sets Axis NodeID for an O-Drive Controller through CAN BUS

        Set_Axis_NodeID: 0x06
        """

        pass

    def setAxisState(self):
        """
        Set Axis State for an O-Drive Controller through CAN BUS


        CAN Set_Axis_State: 0x07
            Axis_Requested_State:
                Undefined:                          0x0
                Idle:                               0x1
                Startup_Sequence:                   0x2
                Full_Calibration_Sequence:          0x3
                Motor_Calibration:                  0x4
                Encoder_Index_Search:               0x5
                Encoder_Offset_Calibration:         0x6
                Closed_Loop_Control:                0x7
                Lockin_Spin:                        0x8
                Encoder_DIR_Find:                   0x9
                Homing:                             0xA
                Encoder_Hall_Polarity_Calibration:  0xB
                Encoder_Hall_Phase_Calibration:     0xD
        """
        pass
```

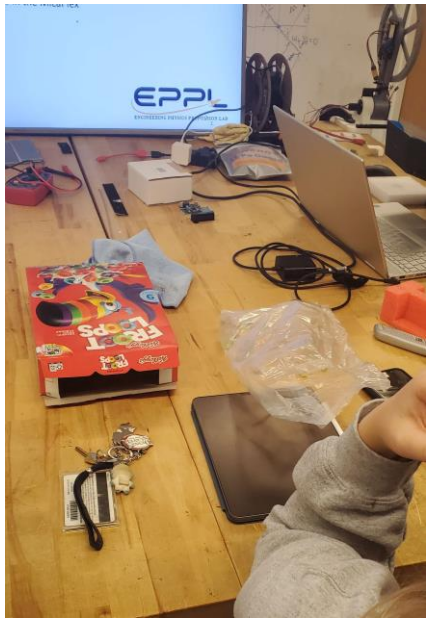
```python
def setControllerMode(self):
        """
        Set the O-Drive Controller Mode type

        Attribute:
            CAN Set_Controller_Mode: 0x0B
                    Control_Mode:
                        Voltage_Control:   0x0
                        Torque_Control:    0x1
                        Velocity_Control:  0x2
                        Position_Control:  0x3

                    Input_Mode:
                        Inactive:      0x0
                        Passthrough:   0x1
                        VEL_Ramp:      0x2
                        Pos_Filter:    0x3
                        Mix_Channels:  0x4
                        Trap_Traj:     0x5
                        Torque_Ramp:   0x6
                        Mirror:        0x7
                        Tunning:       0x8
        """
        pass



def getAxisEncoderEstimates(self):
    """
    Get Encoder Estimates for specific O-Drive Controller Axis through CAN BUS

    CAN Get_Encoder_Estimates: 0x09
                - Pos_Estimate
                - Vel_Estimate

    Attributes:
        Axis_ID


    Returns:
        Pos_Estimate
        Vel_Estimate
    """
    pass
```

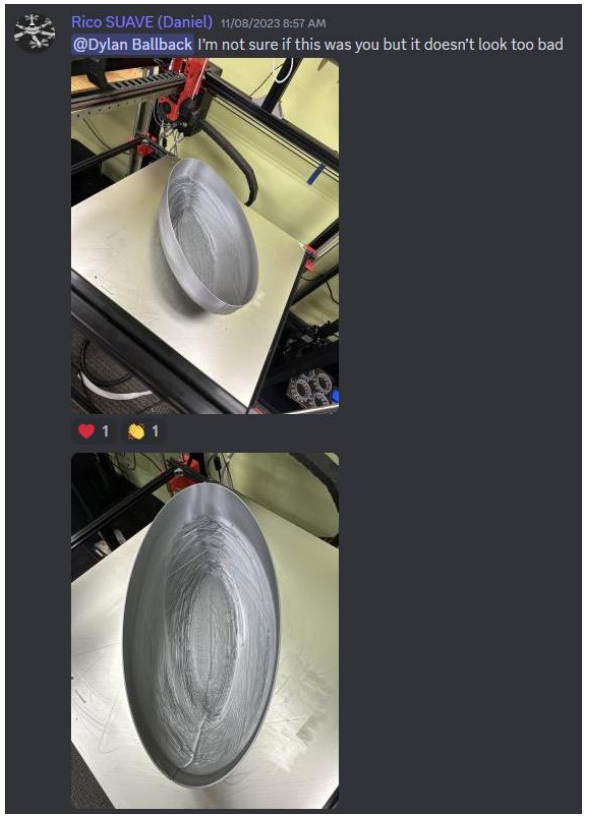
Visual System Updates

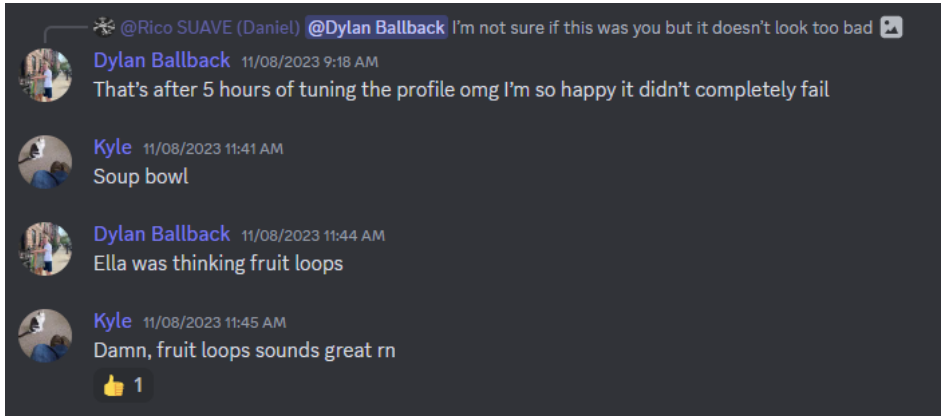
Froot Loops

The result of talking about fruit loops in discord.

Purchasing Fruit Loops.







Ryan + Sensors

Progress completed this past week

-Got bored over break and wrote some sqlite stuff

-Was easy to implement (assuming it works)!

Goals for next week

-Test program

-Get with Justin for merge

Anticipated challenges

-Coursework, it sounds like

[Relevant photos if needed]

ENGINEERING PHYSICS PROPULSION LAB

[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]

COPY