Navigation System MC

- The Raspberry Pi 4B will utilize the Ubuntu operating system which is a Linux-based OS.
- The Raspberry Pi 4b will utilize a Python 3 IDE Pycharm in order to be programmed both on the device and via VNC remote connections.
- The Raspberry Pi 4B will utilize ROS Noetic in order to calibrate and process data pulled from navigation sensors by creating "shells" within the ROS software and utilizing available open-source algorithms.
- The Raspberry Pi 4B will utilize Python 3 in order to develop HD Camera Flash and Image/Video Capture protocols through the use of open-source OpenCV libraries.
- The Raspberry Pi 4B will communicate control signals to the Arduinos used in the Mechanical Movement Subsystem through the ROS software.
- The Raspberry Pi 4B will utilize Python 3 libraries to communicate via LAN connections to the receiver/controller module by inputting and outputting control signals.

Mechanical Movement MCU

- The Arduino Nano Every will receive input from the Raspberry Pi 4 in the Nav subsystem via Serial Communication in the micro USB port on the Arduino.
- The information delivered will be instructions on which way the robot needs to go.
- The Arduino will interpret those instructions and then deliver the directions to the L298N motor controller.
- The Arduino will use PWM to set the speed of the motors.
- The Arduino will receive input from the hall sensors on the motors in order to provide an error detection system for the drive train.
- That error detection adheres to the strategies identified after the FMEA video was made, which arose from the recommendation of Mr. Roberts

Moisture Probe MCU

- The Raspberry Pi will trigger the Arduino nano through a GPIO pin
- The nano will extend the arm until the proximity sensor triggers
- The nano will extend the linear actuator
- Resistance measurements will be taken through ADC
    - R1 will be the first active voltage divider resistor
    - If the reading is too small R2 will be turned on instead
    - The process will continue through all 5 resistors
    - If an edge case is found it will be tested on both sides
- Linear actuator will be lowered
- Arm will be lowered
- Recorded data will be averaged and sent to Raspberry Pi through SPI

Humidity MCU

- The Raspberry Pi will trigger the Arduino micro through a GPIO pin
- The Arduino will record sensor data through I2C for a second
- Recorded data will be averaged and sent to Raspberry Pi through SPI

Temperature MCU

- The Raspberry Pi will trigger the Arduino micro through a GPIO pin
- The Arduino will record sensor data through I2C for a second
- Recorded data will be averaged and sent to Raspberry Pi through SPI

Power

No programing will be necessary for power as it will be a purely