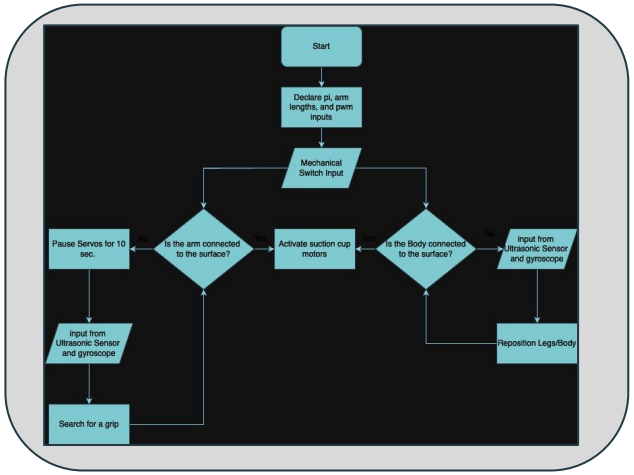
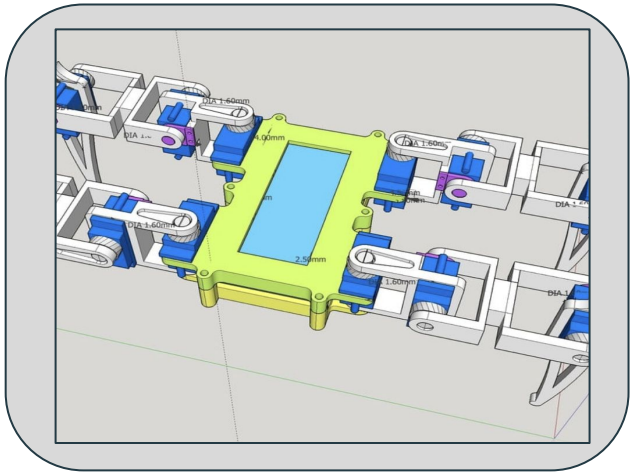


Simplified flow-chart of the program

```
servo_input.ino
1  //-----------------------------------------
2  #include<Servo.h>
3  const float pi = 3.14159;
4  Servo Femur;
5  Servo Joint;
6  Servo Tibia;
7  void setup() {
8
9  }
10 //-----------------------------------------
11 void loop() {
```

Photo of early experiments with code



The physical prototype is a 3D printed, online CAD design of a quadruped robot under the GNU license, which utilizes twelve SG90 micro servo motors for locomotion. Mechanical switches placed on the feet and underside of the robot are wired to the Arduino UNO processors.

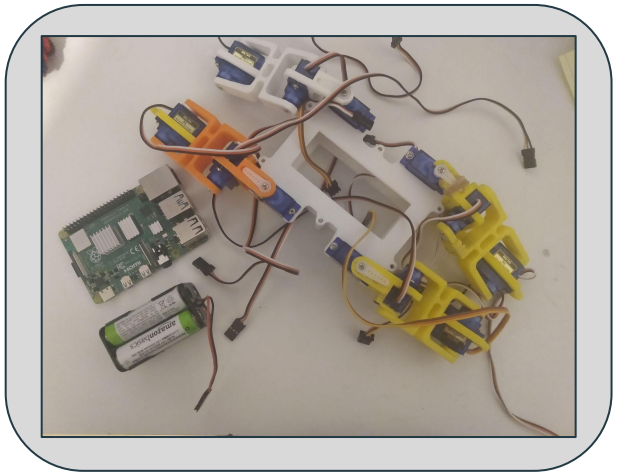
SG90 Servo

Kwadropus Controls

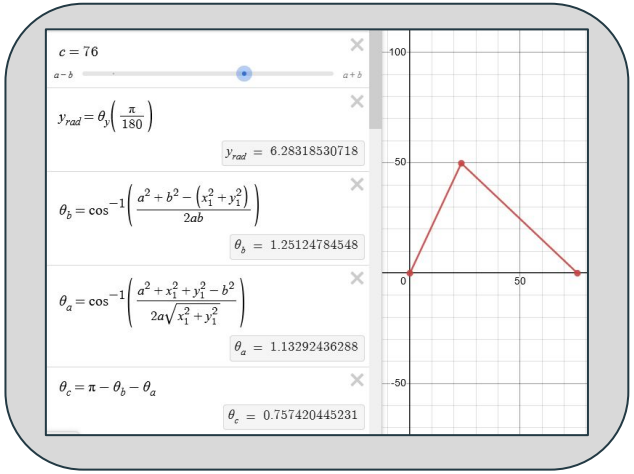
Saint John Valley Tech Center

Mr Mitchell Daigle

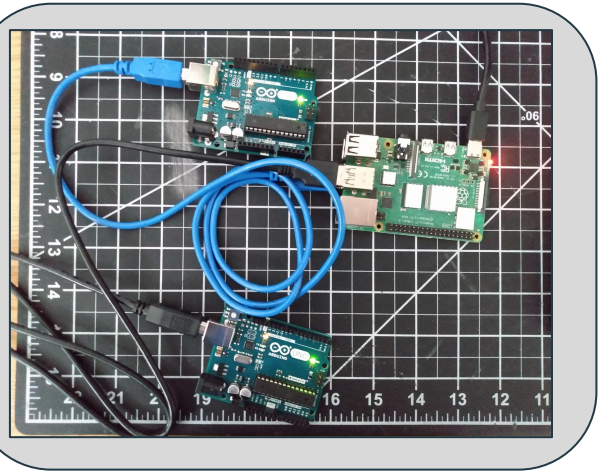
By: Aiden Daigle



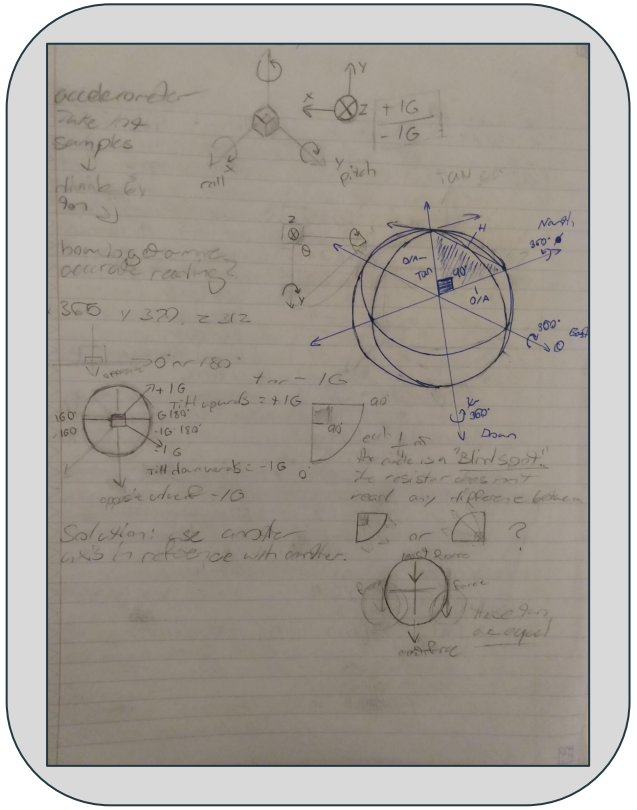
Since the controls for the Kwadropus robot must employ hive programming, this design uses a Raspberry Pi 4 microcomputer, and two Arduino UNO microcontrollers together.



Graphing an inverse kinematic function to solve for servo angles



Microcontroller communication and other kinematics research



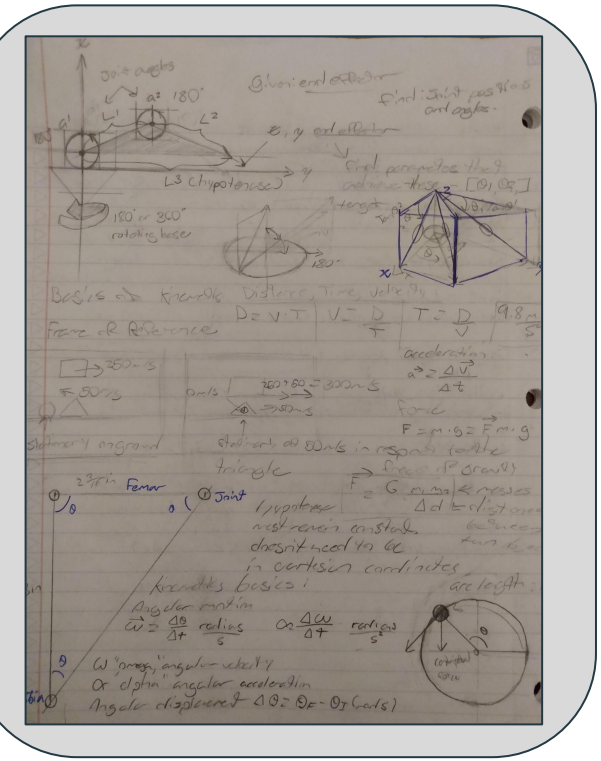
Accelerometer/Gyroscope research and code

Functions for 3 Dimensional coordinate systems

$$r = \sqrt{x^2+y^2+z^2}$$

$$\phi = \begin{cases} \tan^{-1}\left(\frac{y}{x}\right) & (x>0) \\ \tan^{-1}\left(\frac{y}{x}\right)+\pi & (x<0, y>0) \\ \tan^{-1}\left(\frac{y}{x}\right)-\pi & (x<0, y<0) \\ \frac{\pi}{2} & (x=0, y>0) \\ -\frac{\pi}{2} & (x=0, y<0) \end{cases}$$

$$\theta = \begin{cases} \tan^{-1}\left(\frac{\sqrt{x^2+y^2}}{z}\right) & (z>0) \\ \tan^{-1}\left(\frac{\sqrt{x^2+y^2}}{z}\right)+\pi & (z<0) \\ \frac{\pi}{2} & (z=0) \end{cases}$$



```
int X, Y, Z; // declare cordiantes
const int sample = 10; // take 10 samples
void setup() {
Serial.begin(9600); // open serial
}

void loop() {
for (int i=0; i<sample; i++) { // loop thru ten times, adding each value recieved to the cordinates
  X += analogRead(A0);
  Y += analogRead(A1);
  Z += analogRead(A2);
}
//---------------------------------
  // divide by the amount of samples
  X /= sample;
  Y /= sample;
  Z /= sample;
//---------------------------------
//print x y z values
  Serial.print("X: "), Serial.print(X);
  Serial.print("\t"); Serial.print("Y: "); Serial.print(Y);
  Serial.print("\t"); Serial.print("Z: "); Serial.println(Z);
  delay(100);
}
```