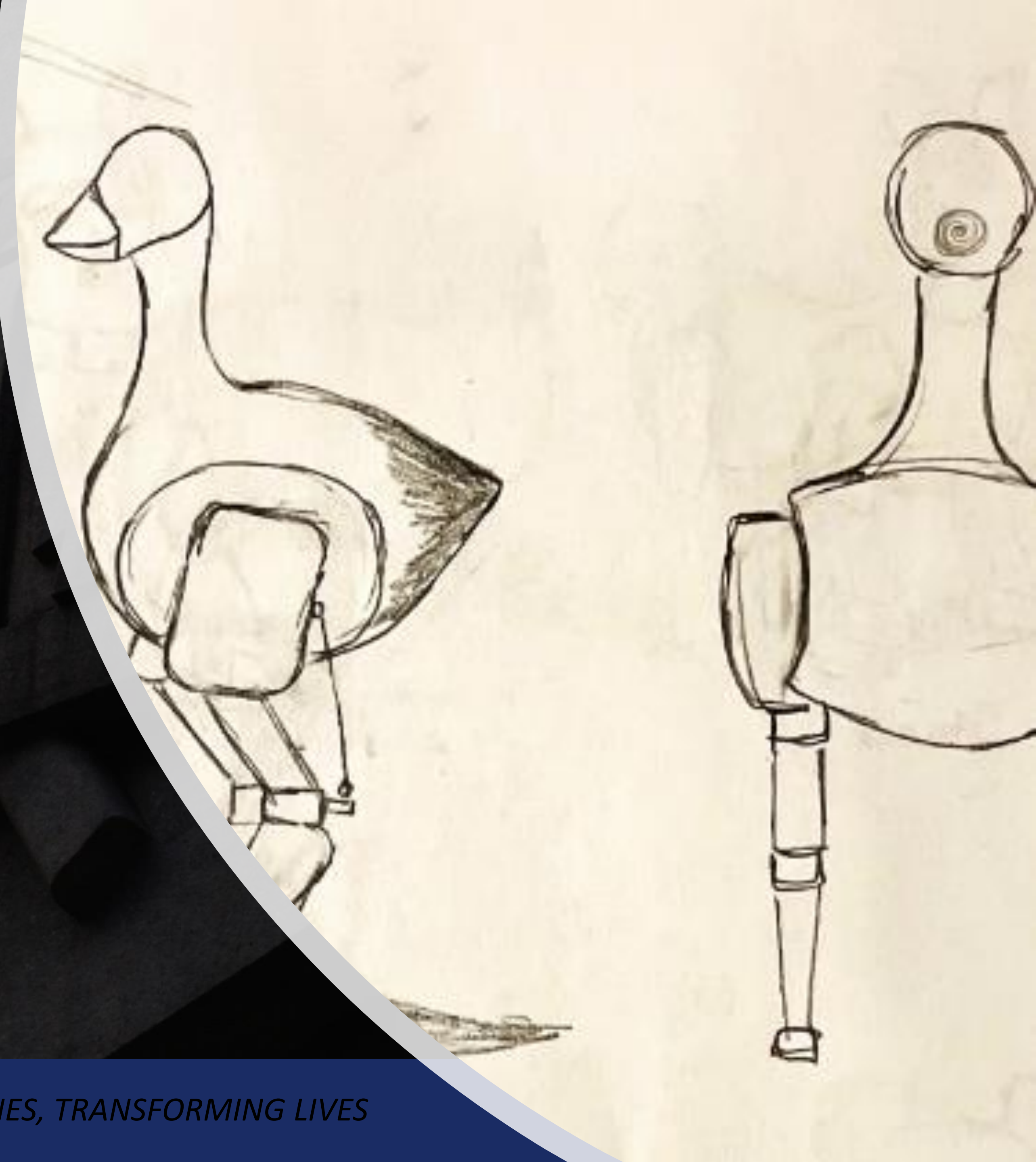
Robotics Studio MECE 4611
Spring 2023 Semester

Oluwatamilore Olushina

OTO2109

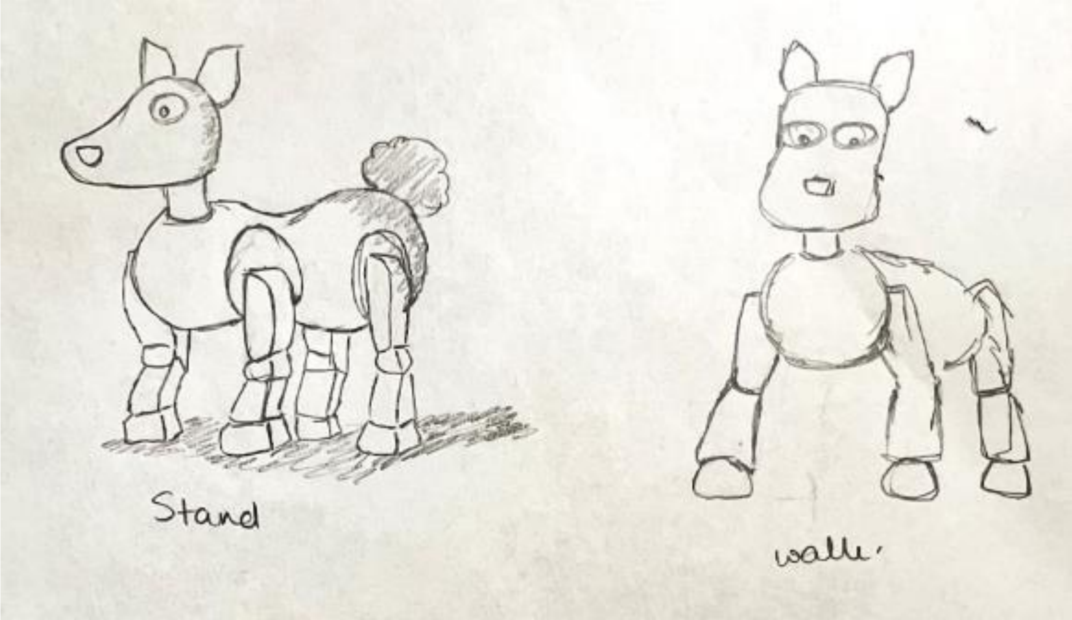
Assignment 1

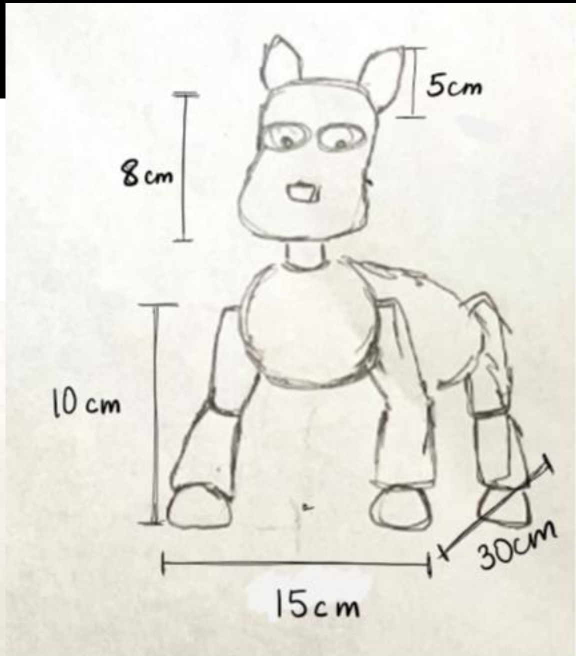
Submitted 28 Jan 2023, 19:23

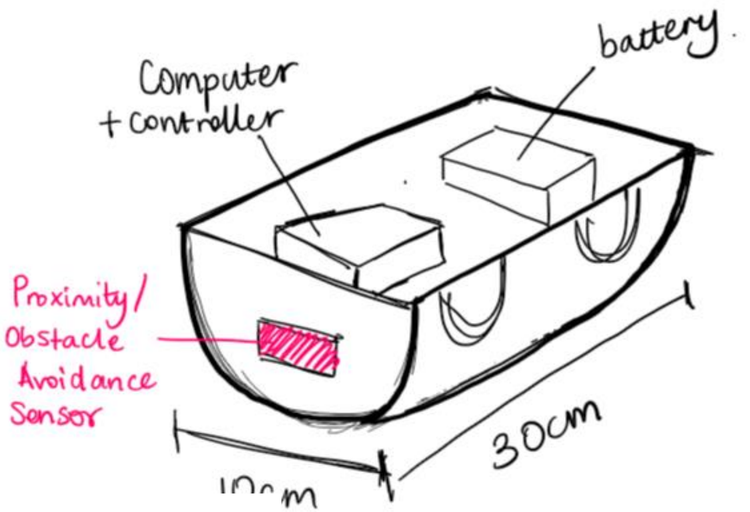


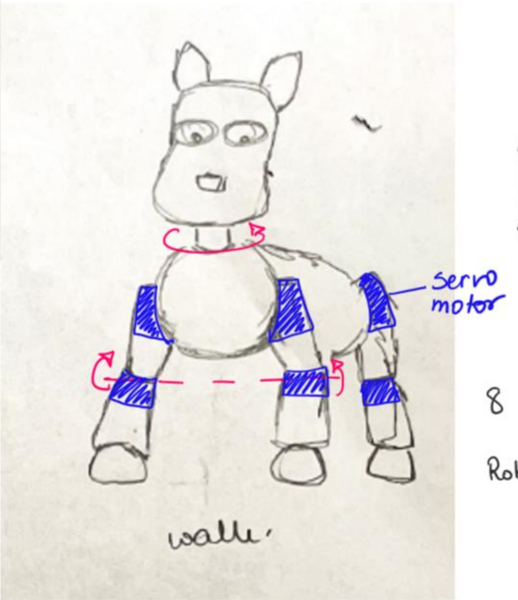
Concept 1 – PupBot

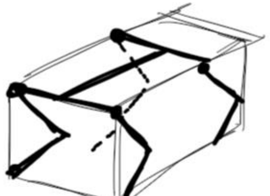
A poodle inspired dog design.











Approx. Power

8 motors at 6W per motor

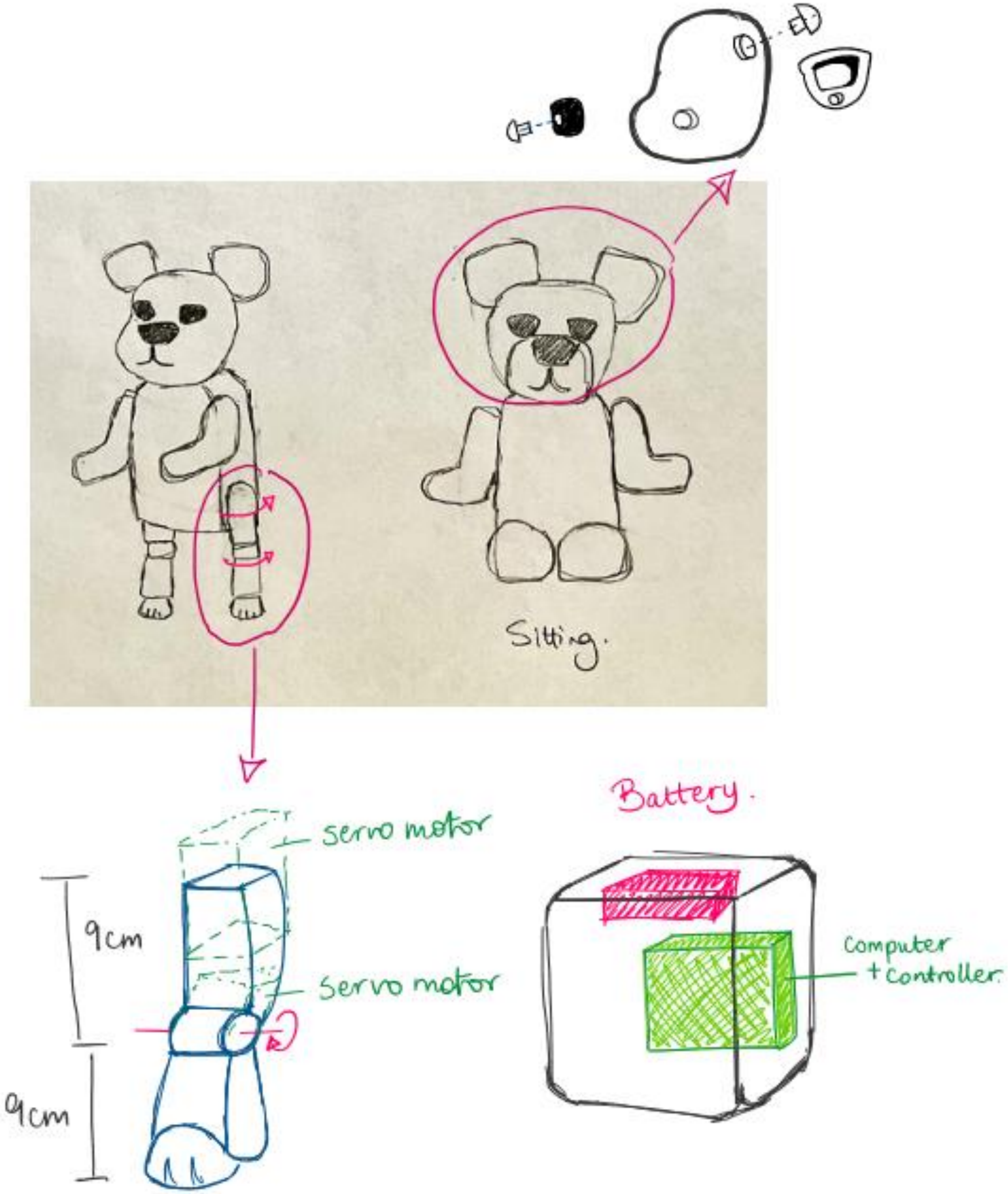
Robot runs for 38 mins.

$\frac{30W}{8 \times 6W} = 37.5$ mins.

Concept 2 – Teddy Bear

The initial idea was inspired by the idea of a robotic teddy bear.

- Most of the design is composed of fabric and wood to minimize their weight.
- The legs needed for a bipedal animal look more complex than you would expect on a teddy bear – the aesthetic does not match.



Approx. weight est.

$= (0.3+1+0.5)\text{kg} = 1.8\text{kg}.$

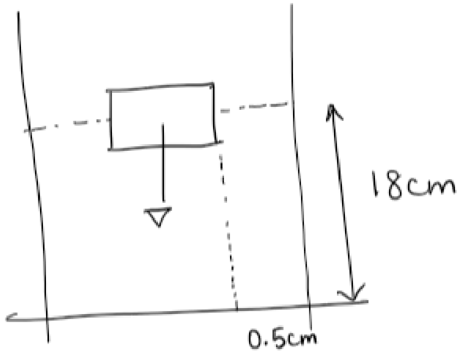
max flexion $= 30°$.

Static torque $= 1.8\text{kg} \times 18\text{cm} = 32.4\text{kgcm}$

→ Motor specification $\simeq 17\text{kgcm}$

• As $32.4 \gg 17$, these cannot support a single motor.

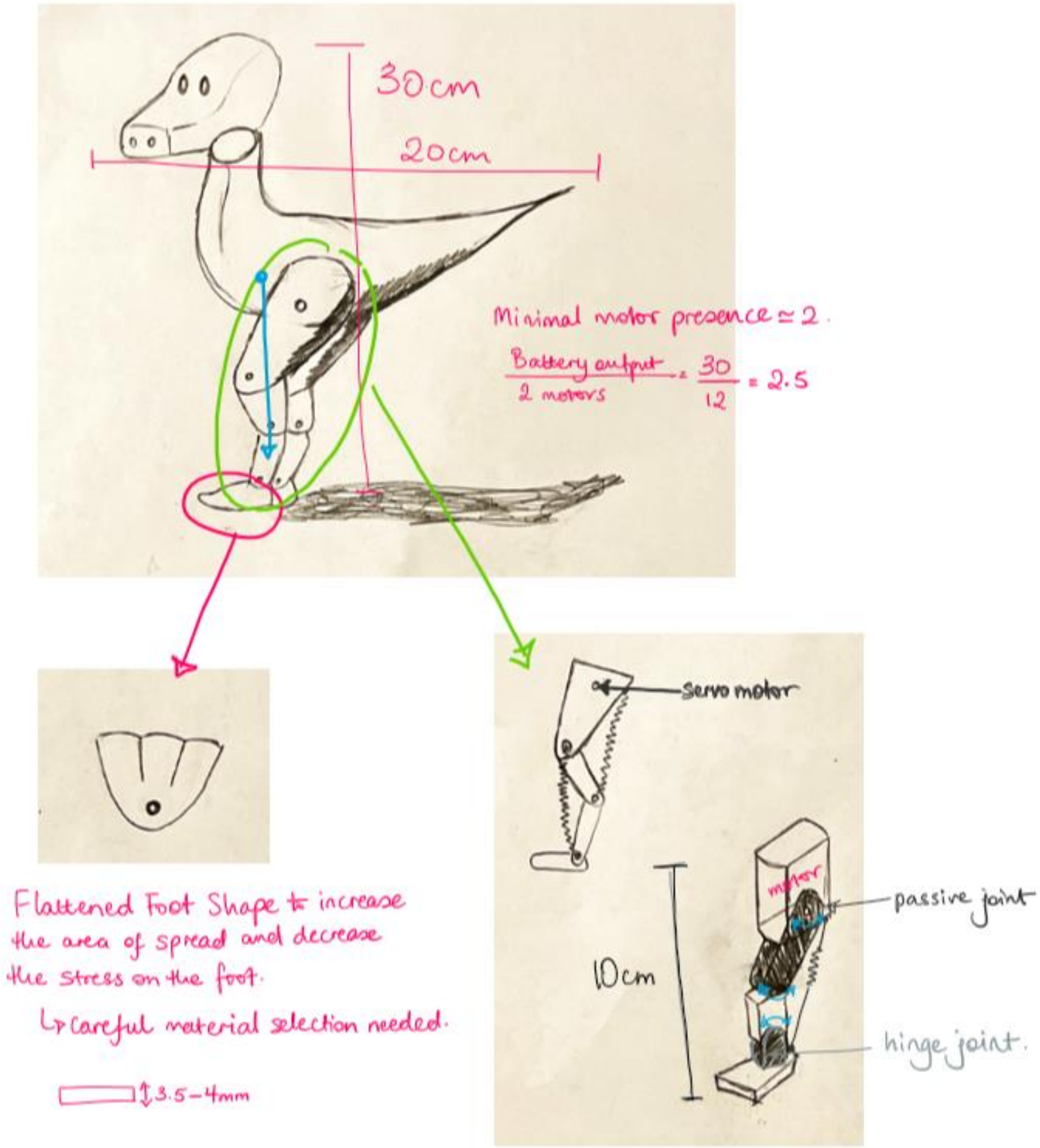
→ Reduce weight to approx 0.9kg to work within the safety limits.



Concept 3 – Dinosaur

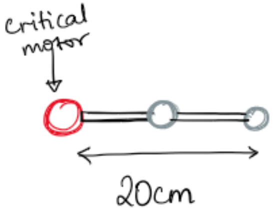
This idea is inspired by the dinosaur toy in the movie series Toy Story.

- The aesthetic tries to follow the general shape of the T-Rex as closely as possible.
- The extension of the tail is necessary to balance out the weight of the head.



Concept 4 – Ostrich (Ogongo)

The refinement of the idea is based on the concept that dinosaur legs evolved into birds. Since ostriches are flightless birds, they serve as the primary inspiration.



Assume volume of one link = $\pi(1cm)^2 \times 10cm$ =

density of HDPE pipe = $0.95 g/cm^3$

Mass $\simeq$ 5.97g.

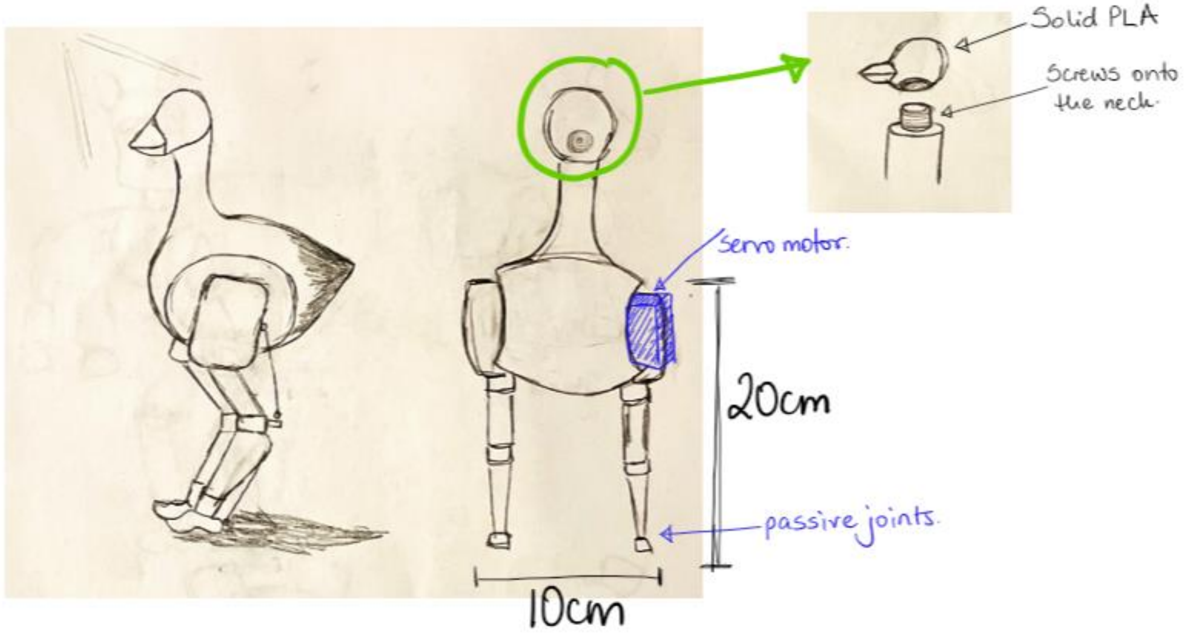
Static Torque for motor 1

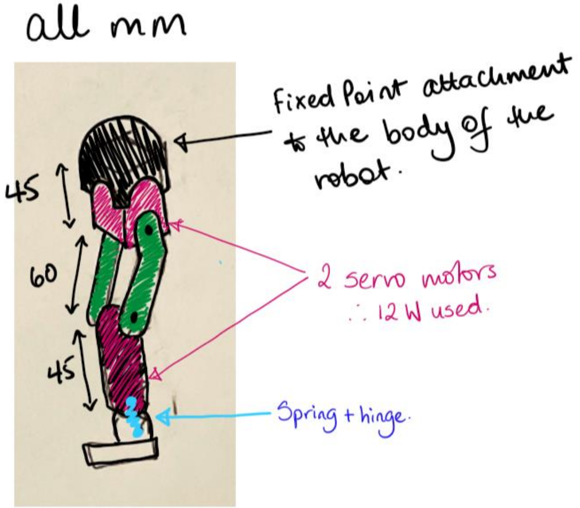
$= [(52+5.97)g \times 9.81 \times 10] + [(52+5.97) \times 9.81 \times 20]$

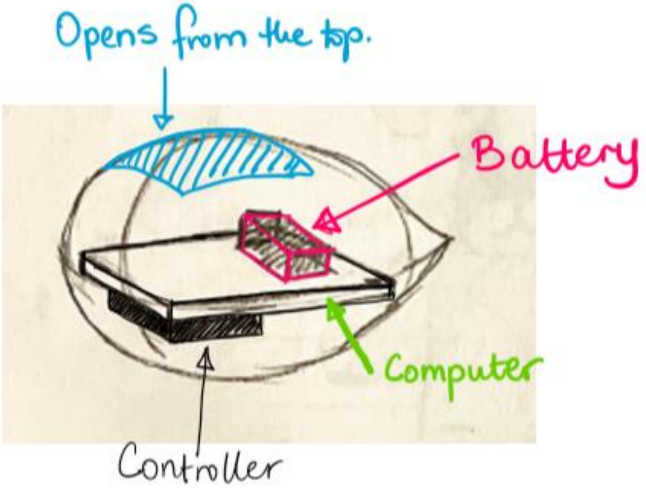
$= 17.1$ N.cm

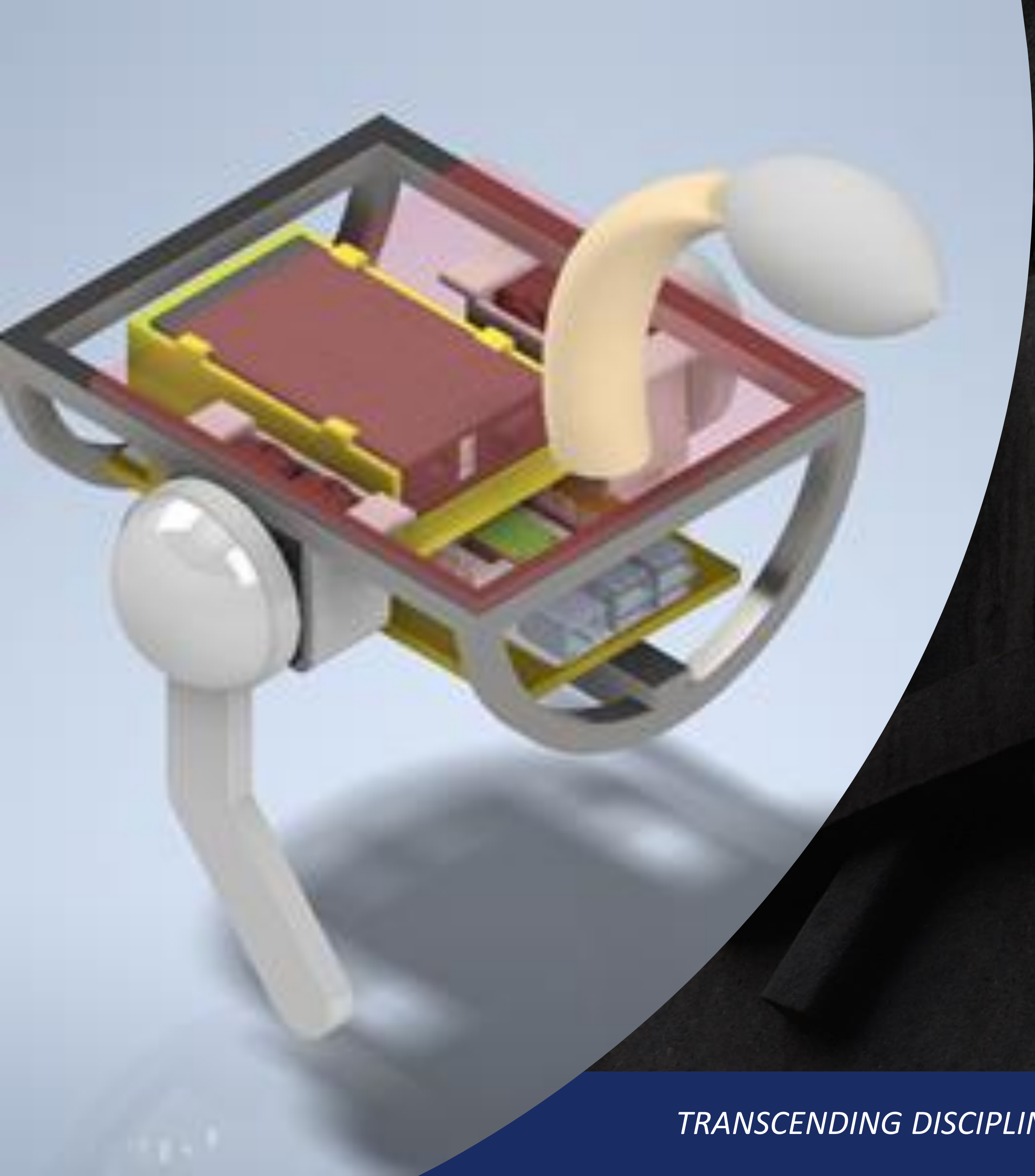
Motor spec. is 17 kg.cm $\simeq$ 166.7N.cm

17.1 N.cm << 166.7N.cm









Robotics Studio MECE 4611
Spring 2023 Semester

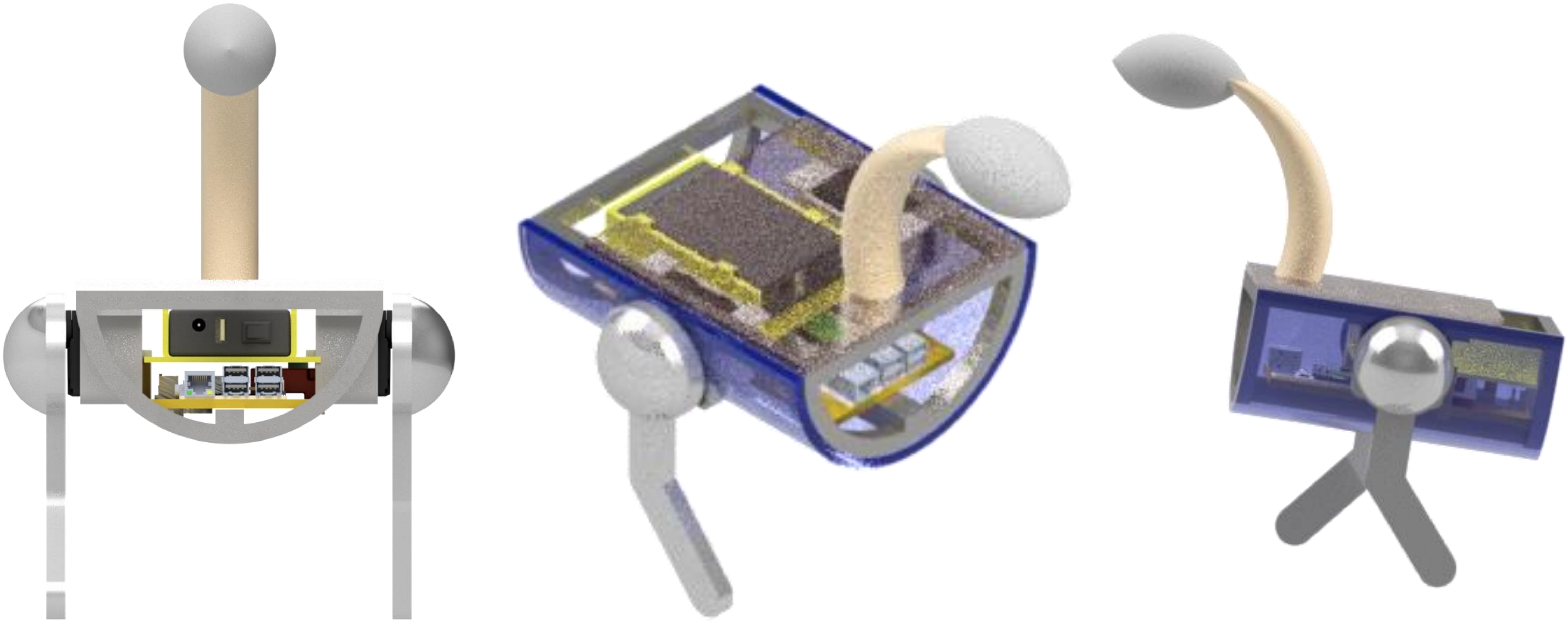
Assignment 2

Oluwatamilore Olushina
OTO2109

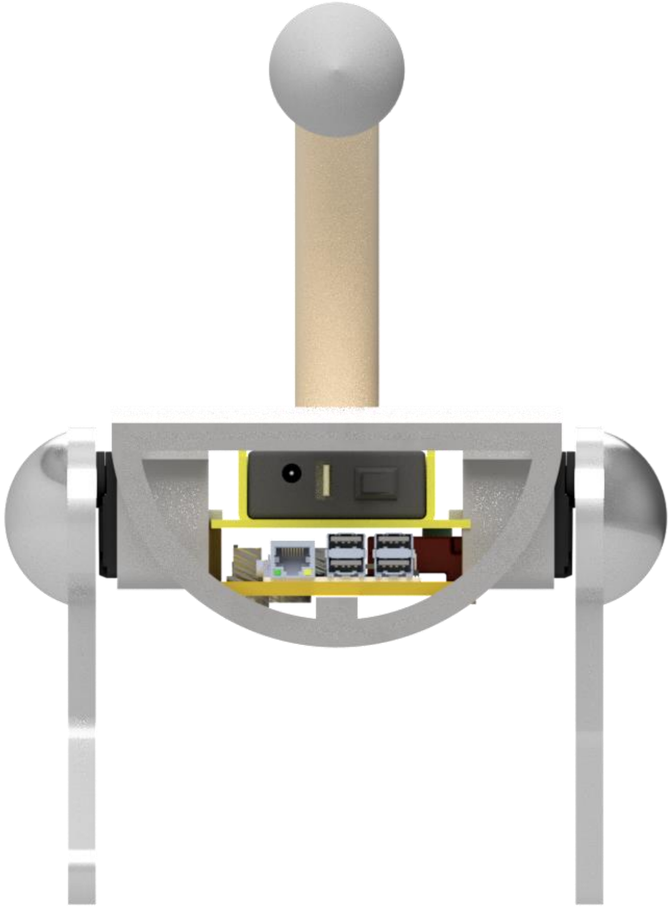
Submitted 5 Feb 2023 14:35

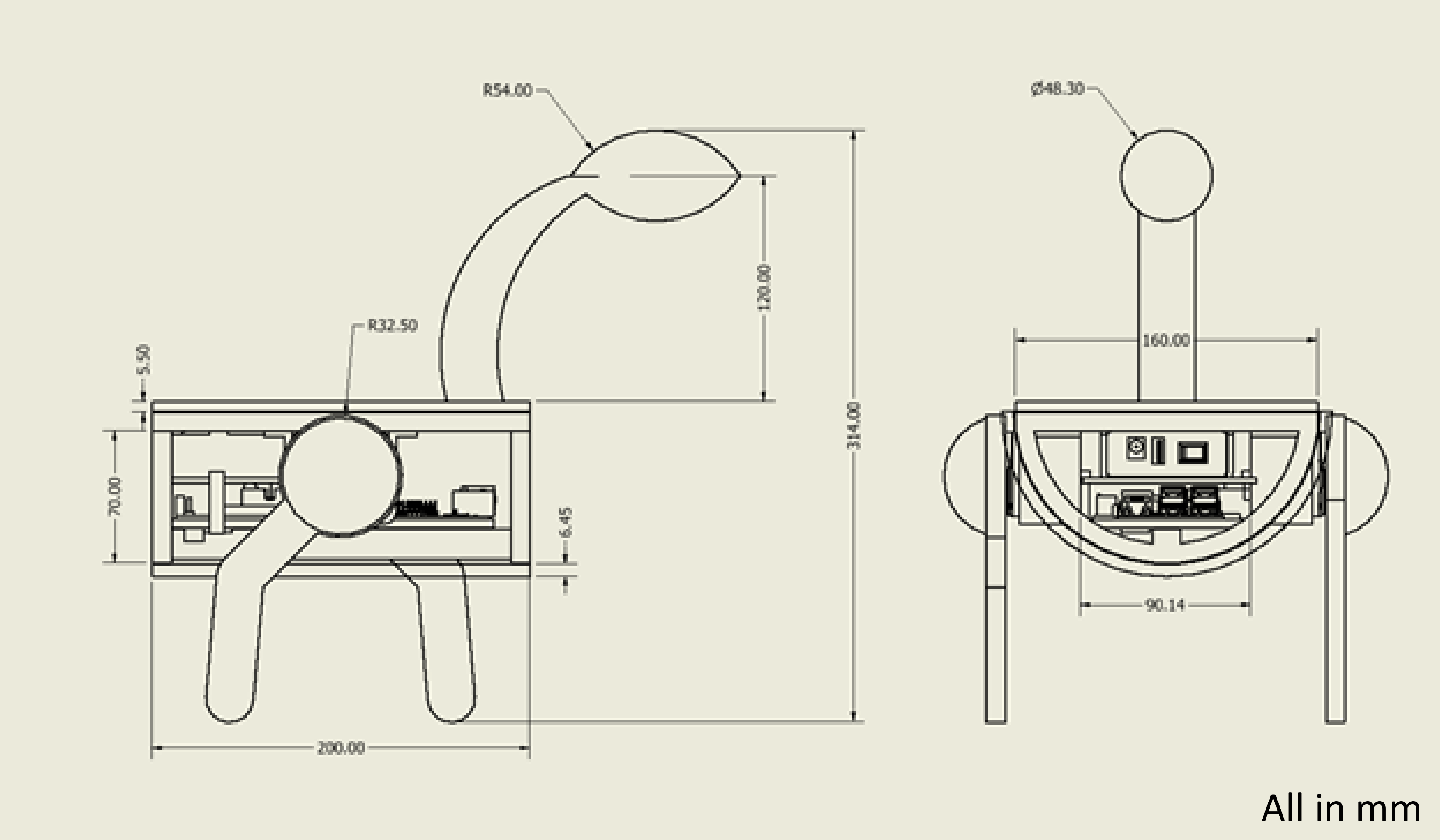
Birdy-Bot

Different Views of the Birdy-Bot Rendering

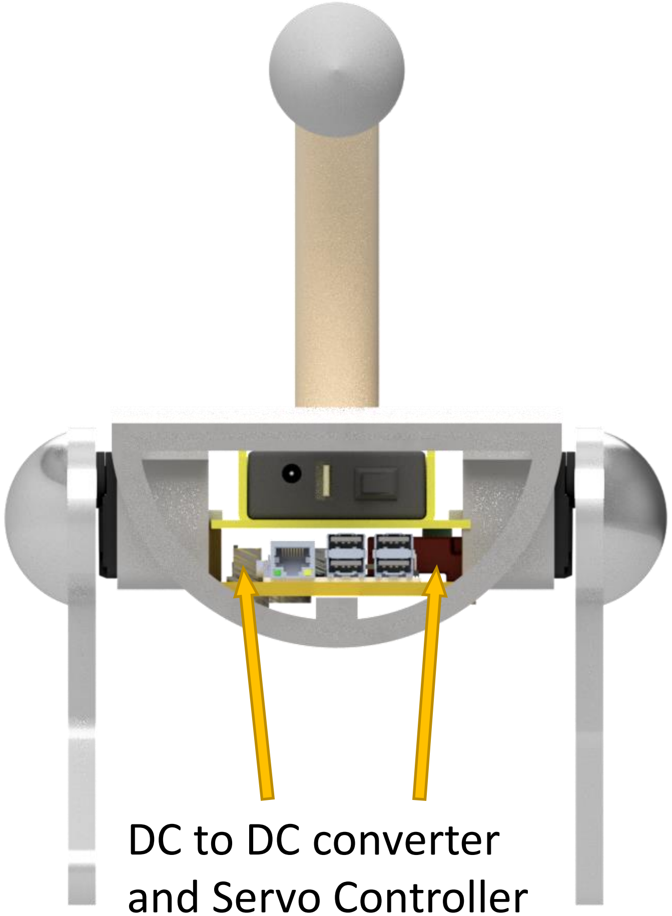


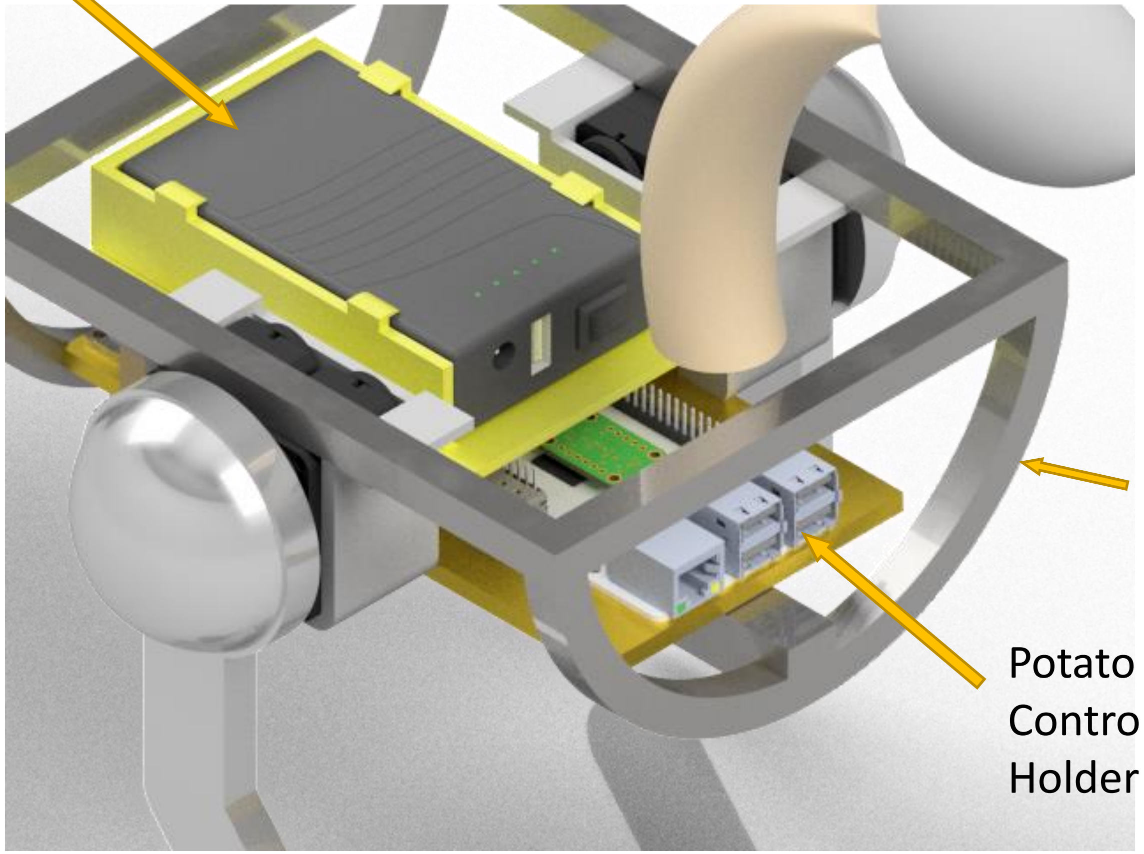
Main Dimensions of the Birdy-Bot





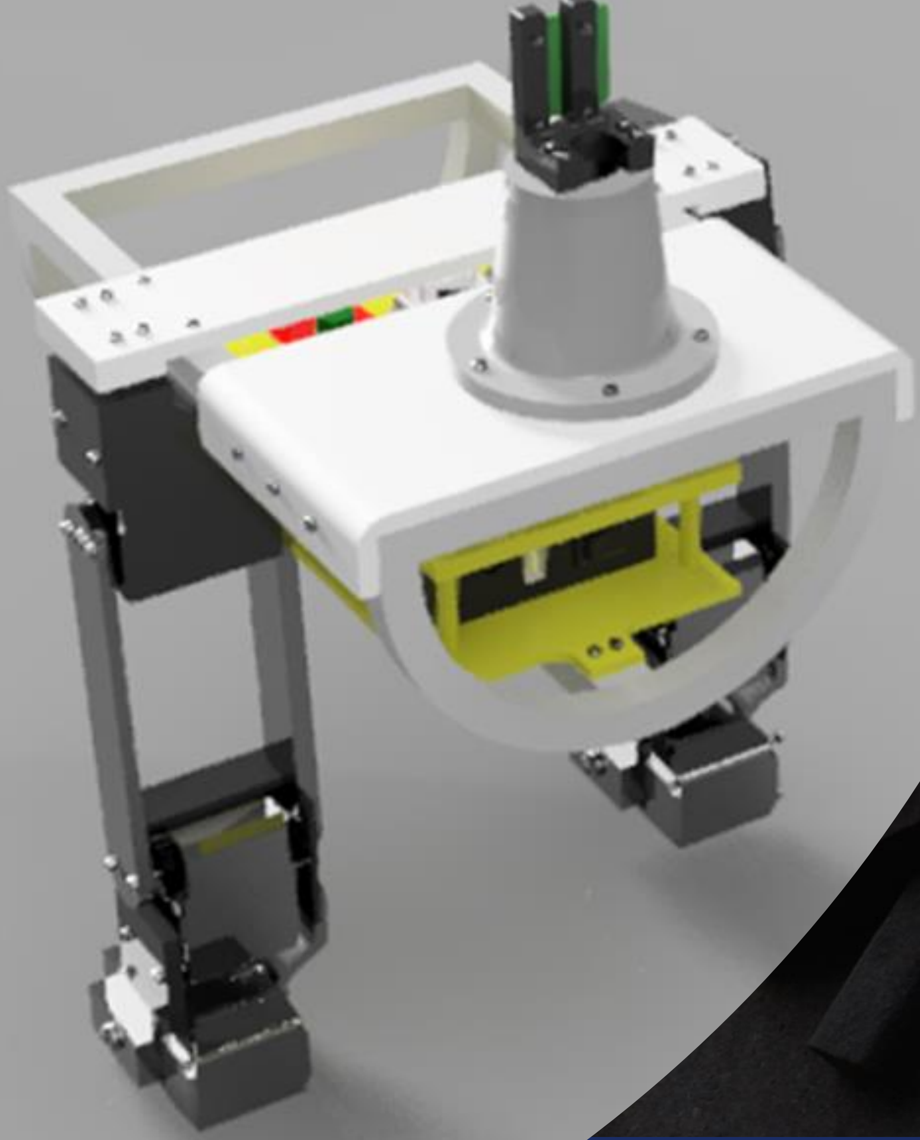
View of the Birdy-Bot Rendering





Context Rendering with Animation

[Animation of Leg movement using concept Leg design]



Robotics Studio MECE 4611
Spring 2023 Semester

Assignment 4

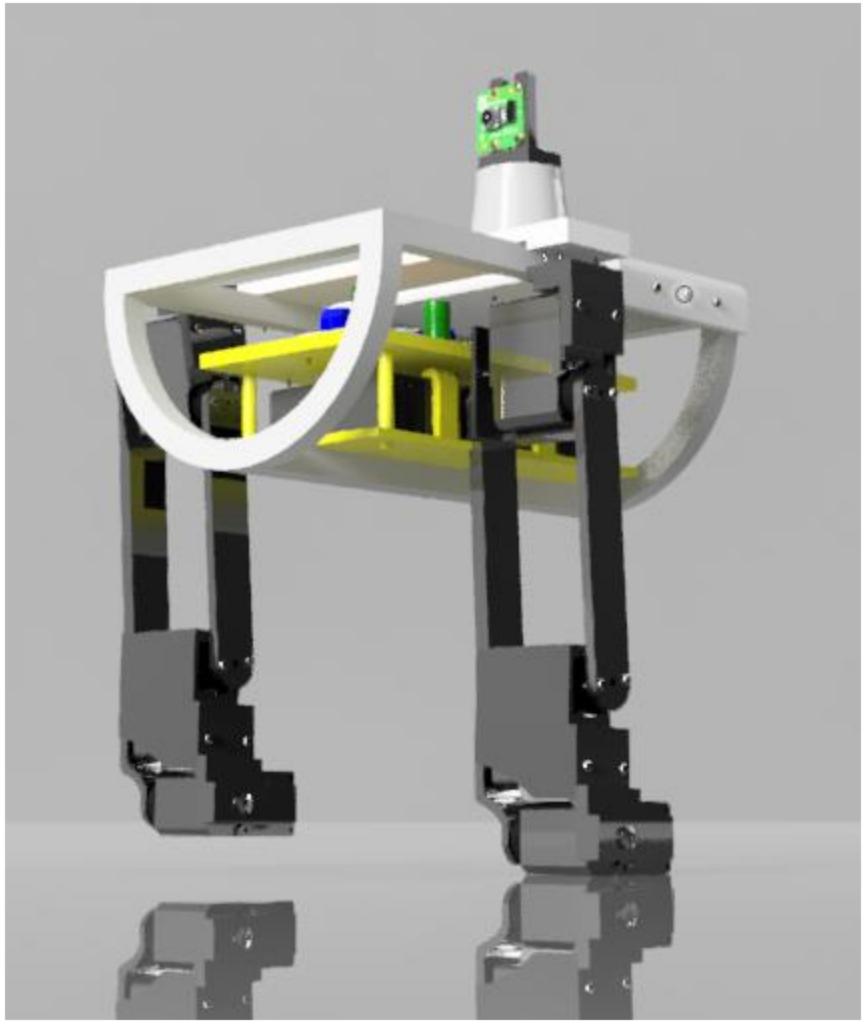
Oluwatamilore Olushina

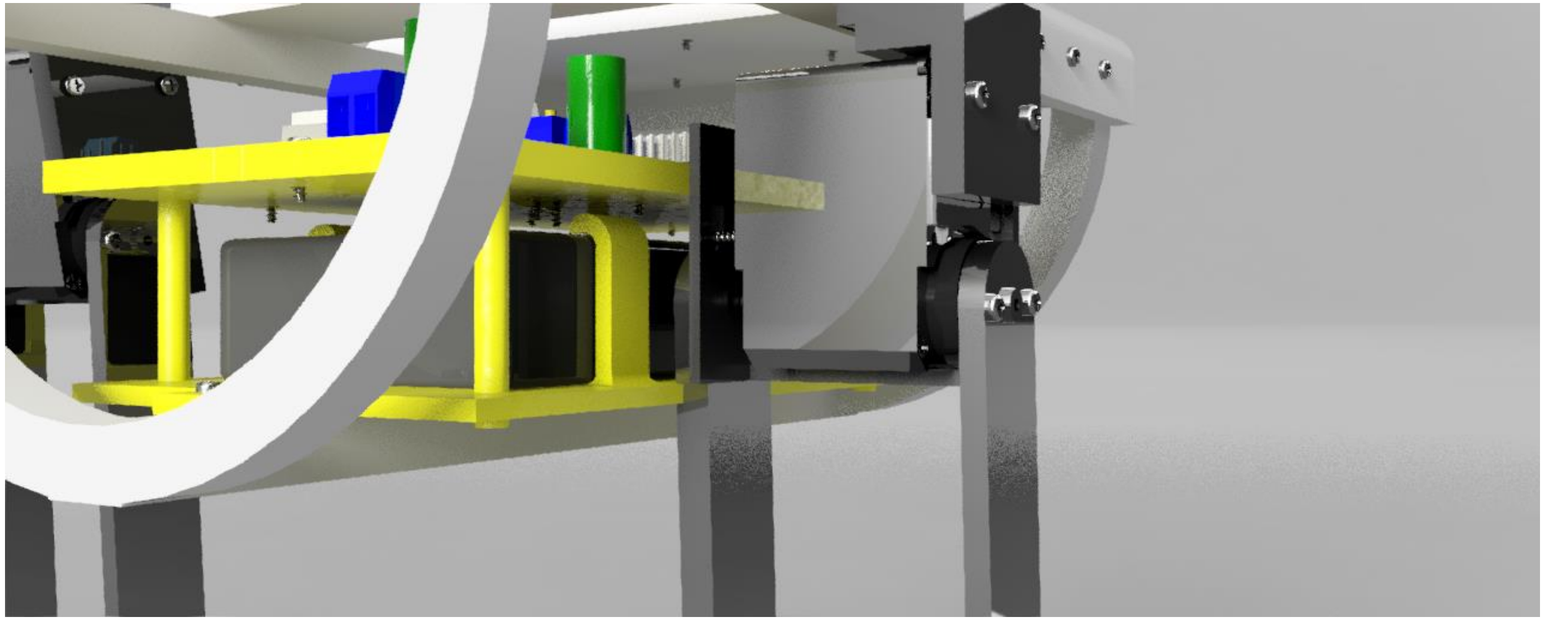
OTO2109

Submitted 5 March 2023 10:03

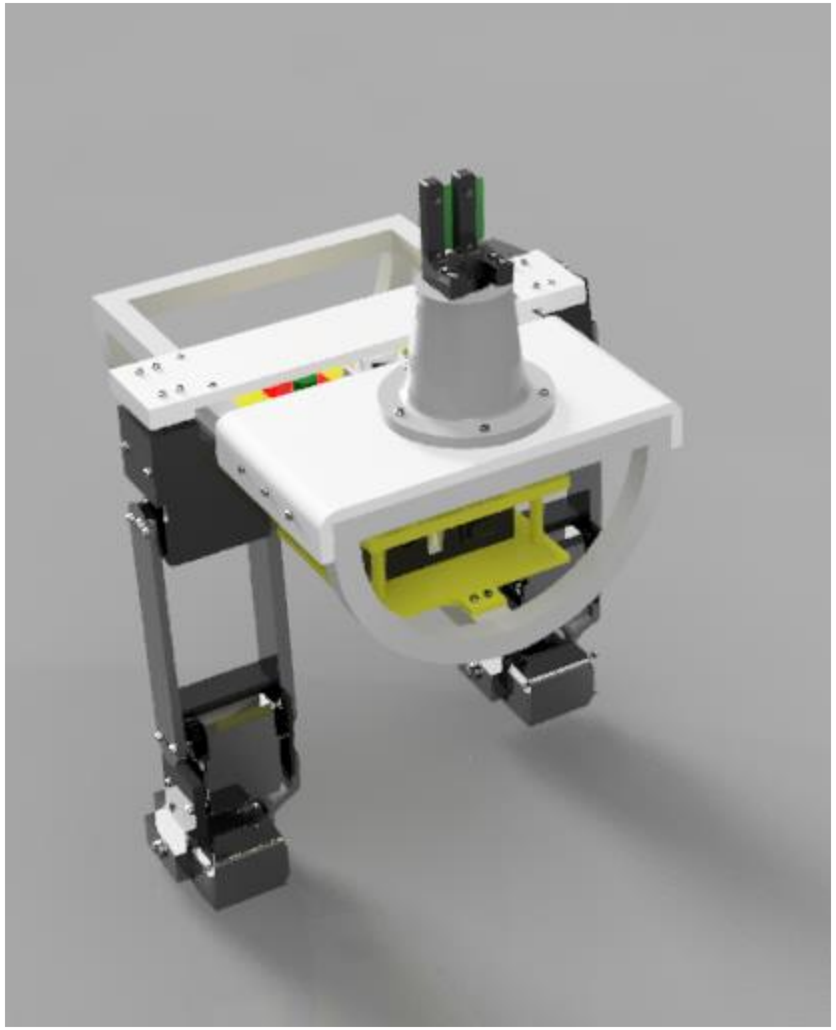
Birdy-Bot

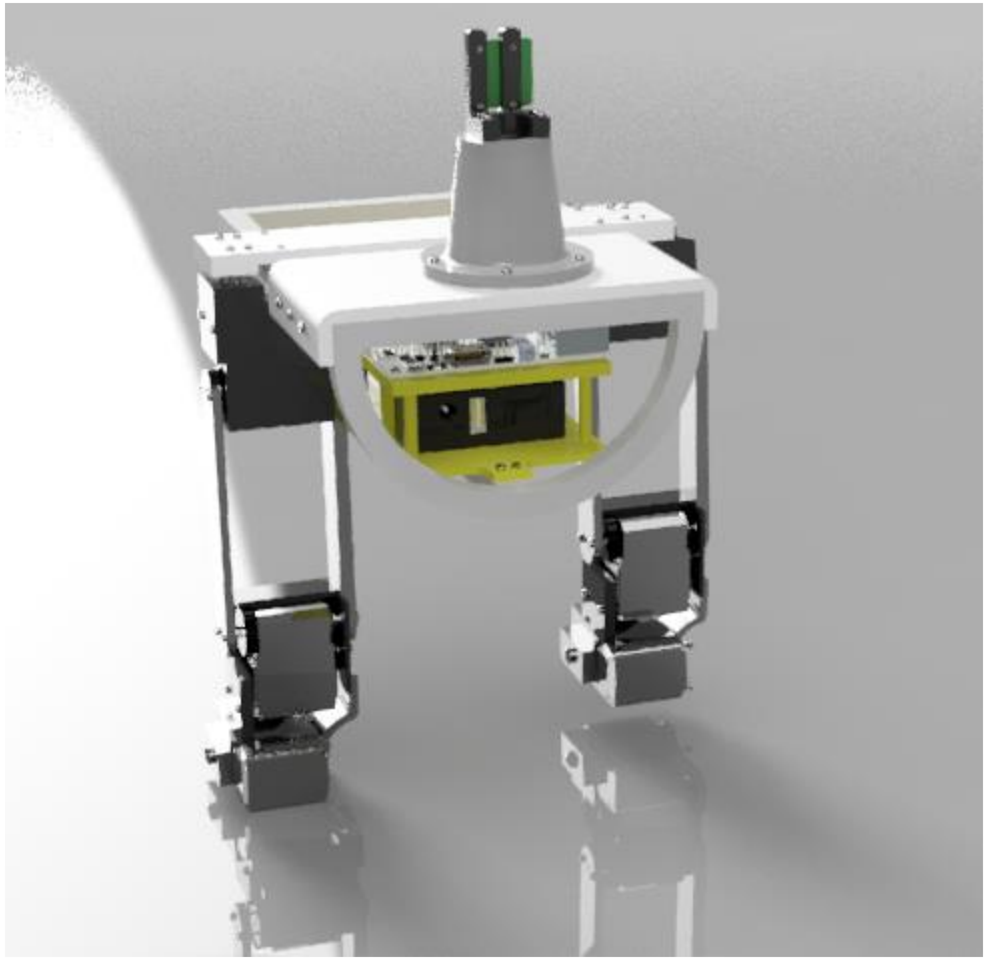
Different Views of the Birdy-Bot Rendering



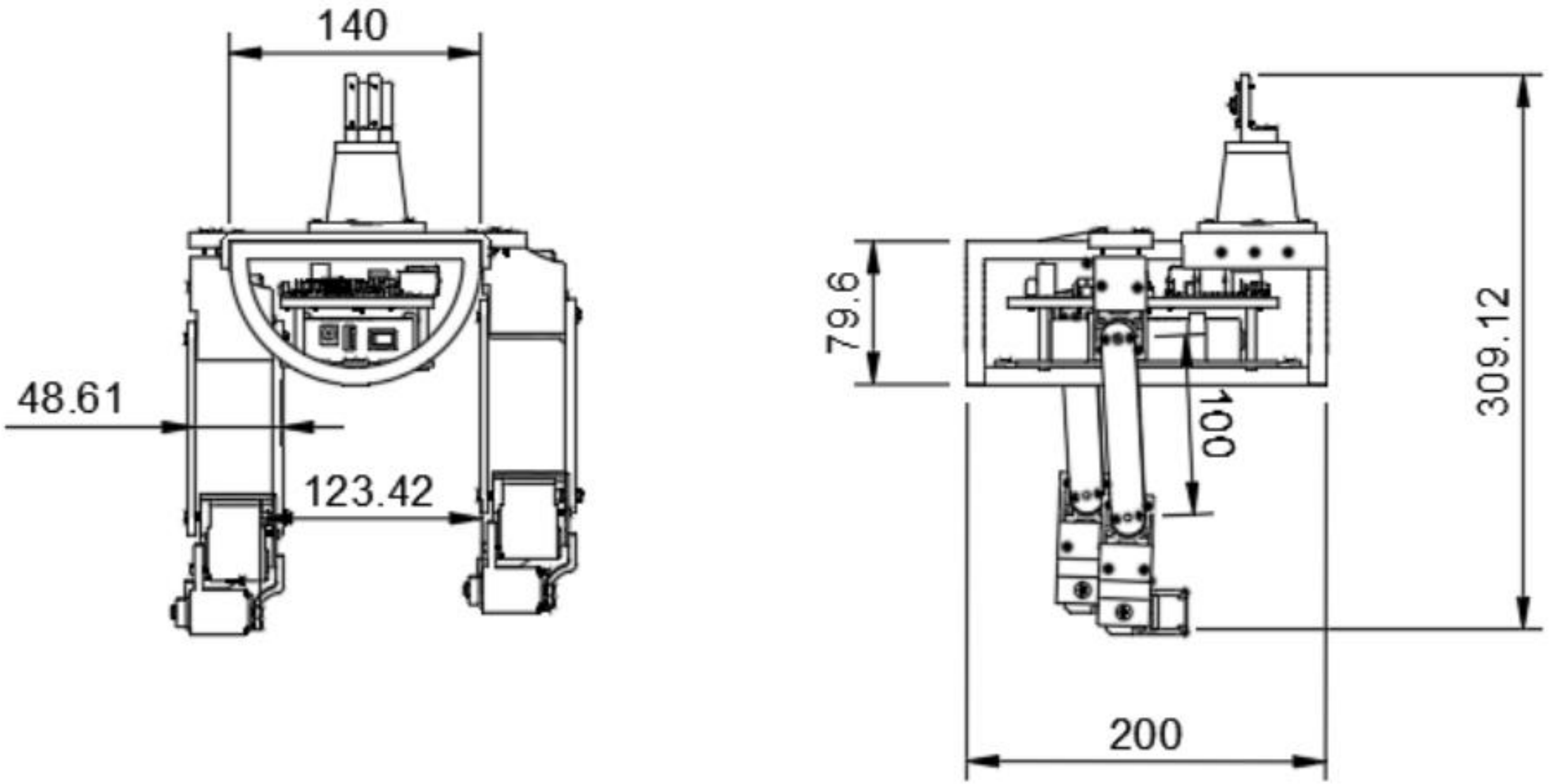


Different Views of the Birdy-Bot Rendering



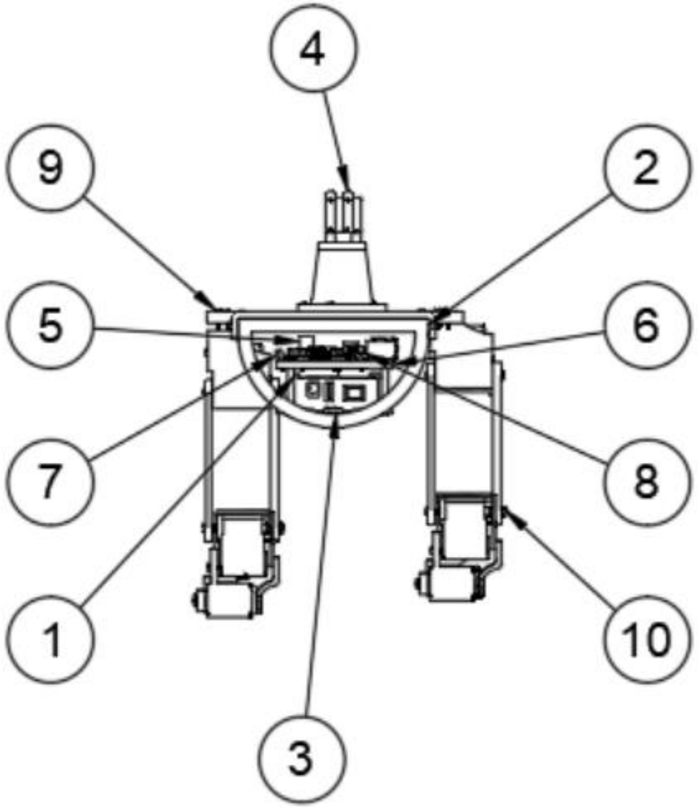


Main Dimensions of the Birdy-Bot



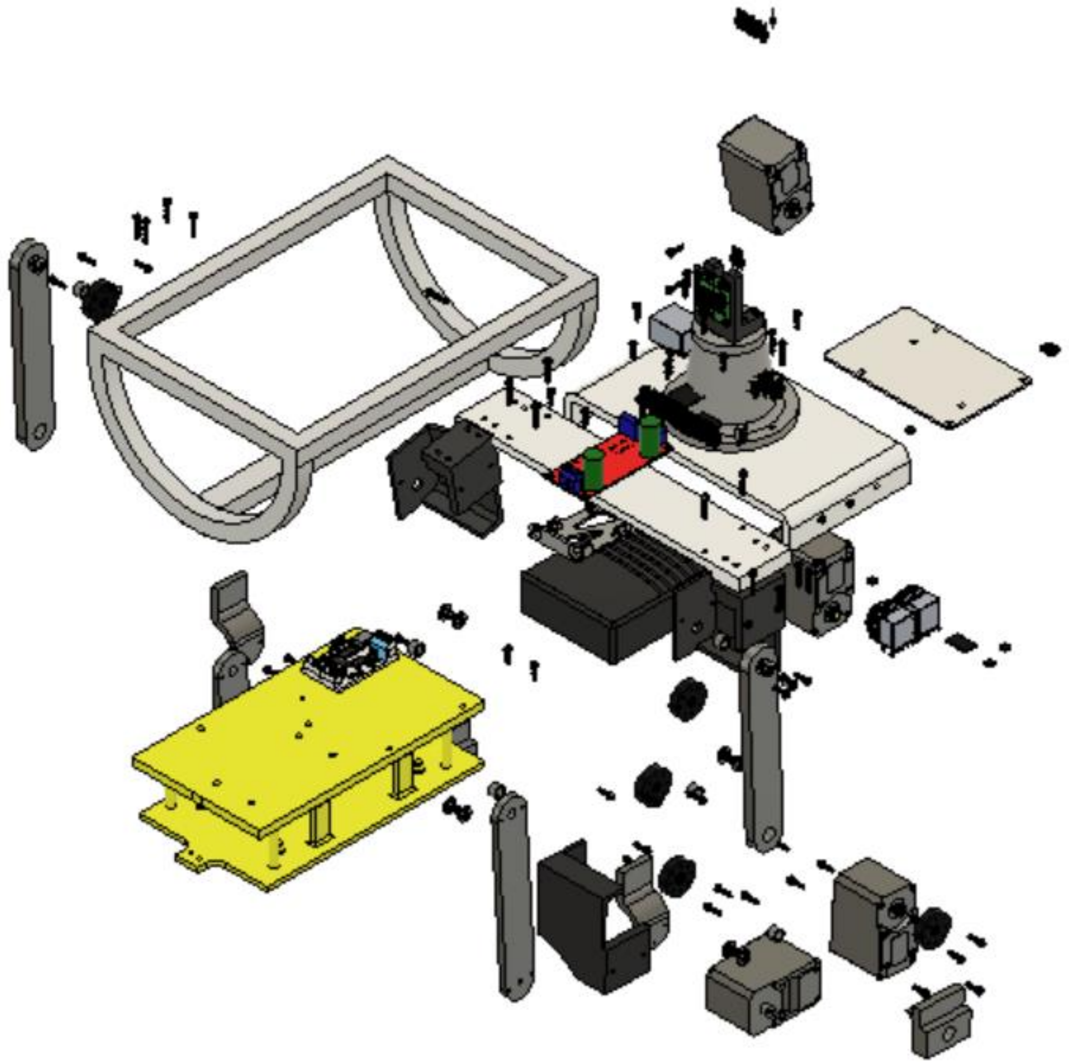
All in mm

Key Parts List Birdy-Bot Rendering

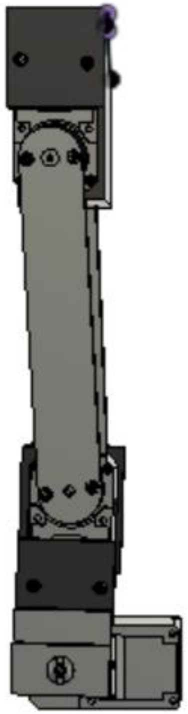


Item	Qty	Part Number
1	1	TalentCell Rechargeable Battery
2	1	Chasiss V1
3	12	M2 8mm 99461A922_Phillips Rounded Head
4	1	Ostrich Neck
5	1	10A Synchronous Step-Down Voltage Regulator DC-DC 4-30V to 1.2-30V 12V
6	1	Controller Board
7	1	Le Potato
8	1	Bus Linker Snap Fit
9	25	M2 12mm Long 99461A924_Phillips Rounded Head
10	2	Leg Assembly

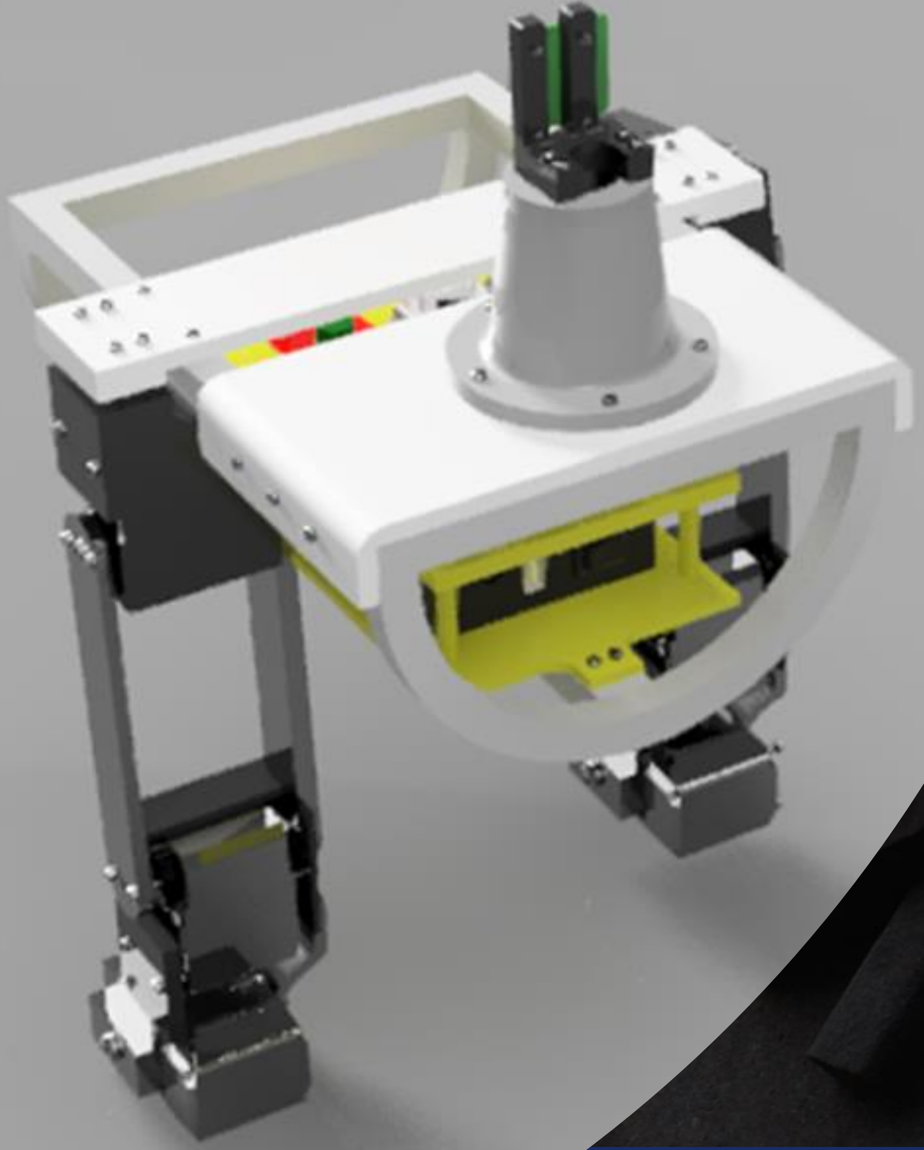
Exploded View Birdy-Bot Rendering



Context Rendering with Animation



[Animation of leg with detailed CAD Design]



Robotics Studio MECE 4611
Spring 2023 Semester

Assignment 4

Oluwatamilore Olushina

OTO2109

Submitted 4 March 11:54 PM

Birdy-Bot

Leg in Motion – Photos and Video

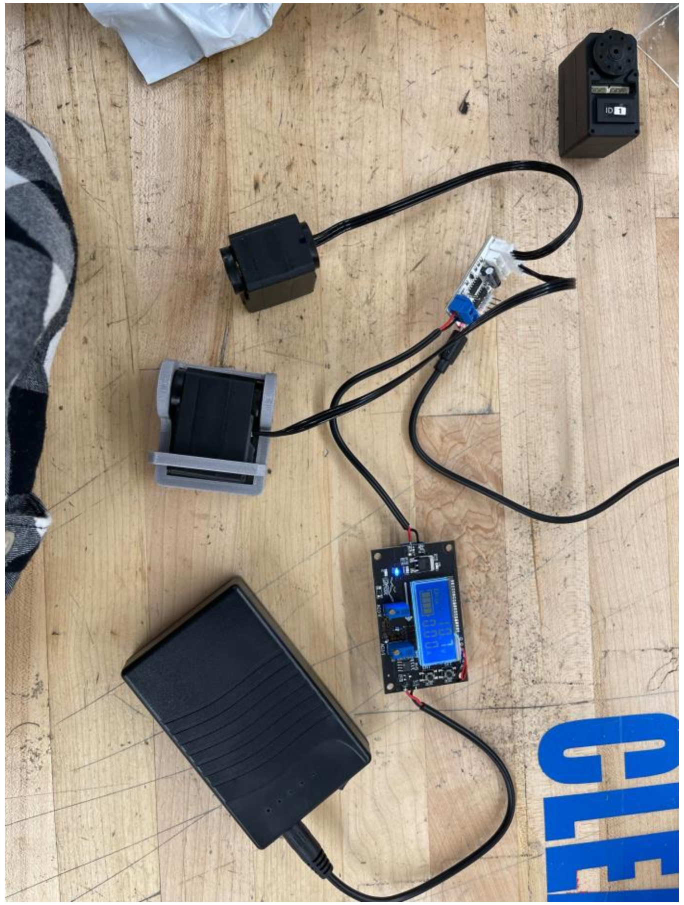
Although the current design allows easy swapping of parts, some redesigns based on this idea are needed.

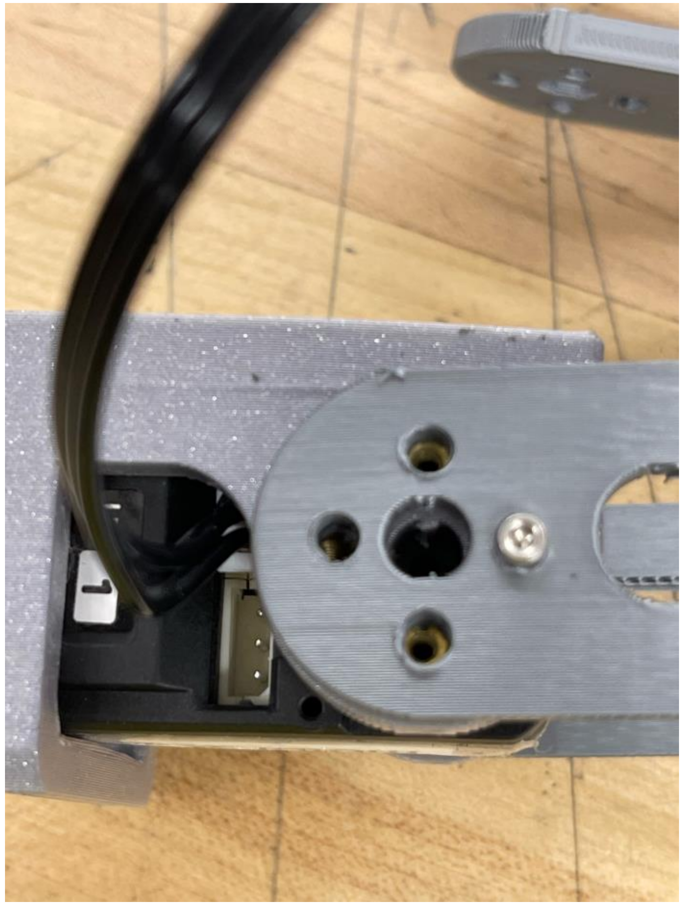
https://youtube.com/shorts/5pFsCOnPuxM?feature=share

Form Issues Identified

The shelf used to hold the motor at the knee in place did not have the correct tolerances or hole dimensions. The linkage was redesigned with an integrated shelf using previously found dimensions.

Form Issues Identified





The foot was redesigned and reprinted.

Reasons

- Considerable difficulty aligning the screws between the linkages.
- The motor has limited movement.

Form Issues Identified

A simpler foot was used to create an easy-to-build base, but more edits may need to be prevent interference with the ankle to foot linkage.

Robotics Studio MECE 4611 Spring 2023 Semester

Assignment 5

Oluwatamilore Olushina

OTO2109

Submitted 27 March 2:22 AM

Birdy-Bot

Leg in Motion – Photos and Video

https://youtu.be/ux9bLoVPfnQ

Form Issues Identified

The initial chassis design fails in shear when bolting motors in place.

Reasons

- Considerable difficulty aligning the screws meant holes needed to be increased. Drilling caused shear failure.

Form Issues Identified

Space available in the current design limits the linkage servo motor movement.

- The redesign allows both modular foot linkage point and free motor movement.

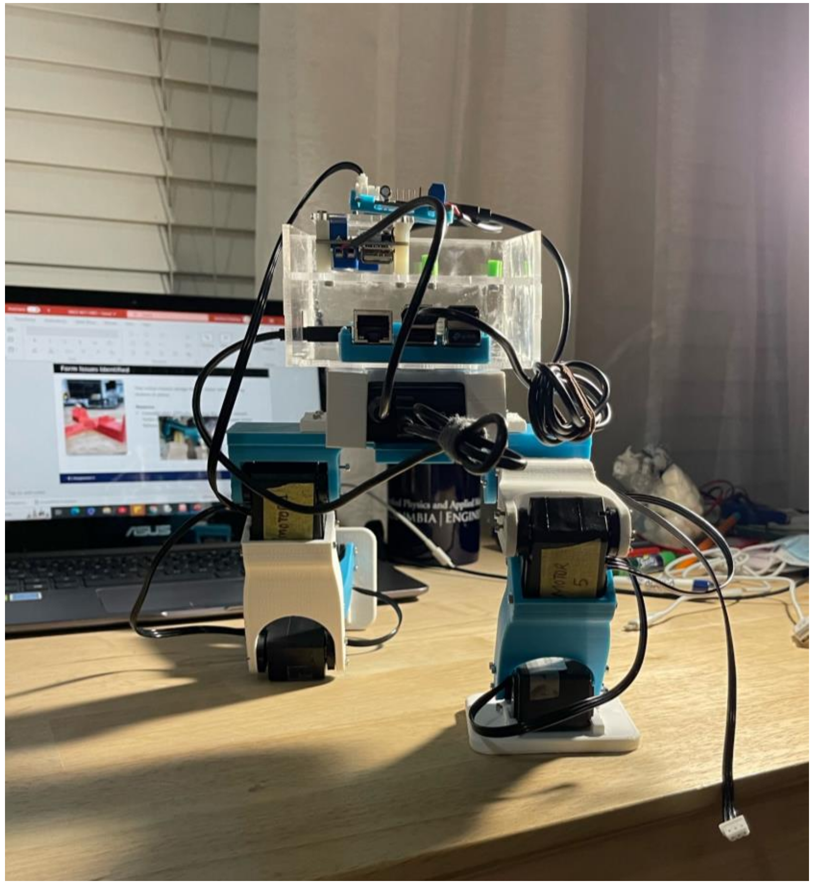
Form Issues Identified

Multiple foot designs used on the foot. Alignment was difficult, so but more edits may need to be prevent interference with the ankle to foot linkage.

The redesigned foot allows maximum stability.

[Building the Robot]

Stability Configurations

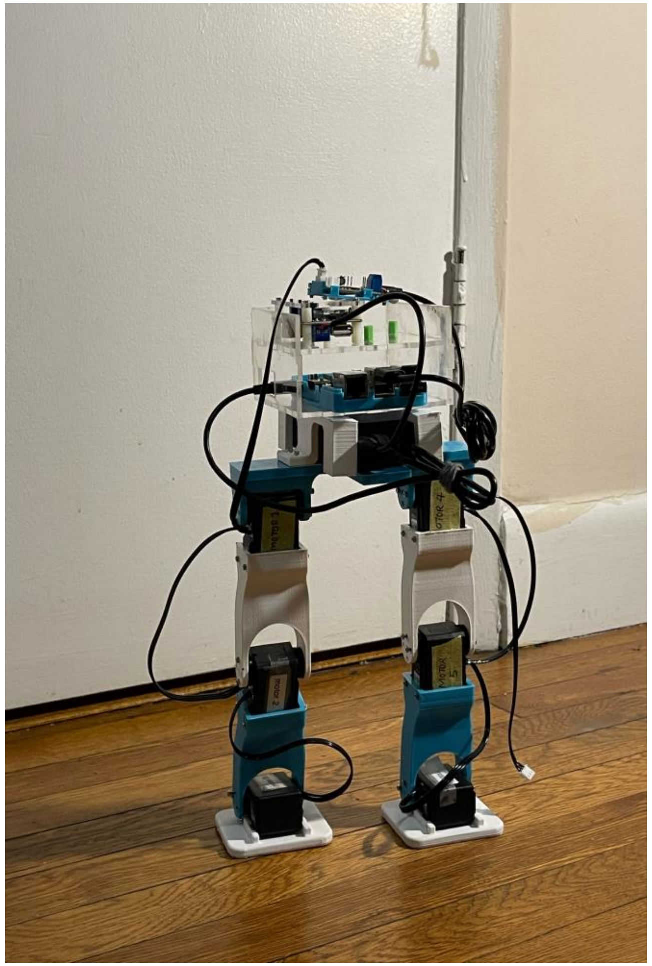


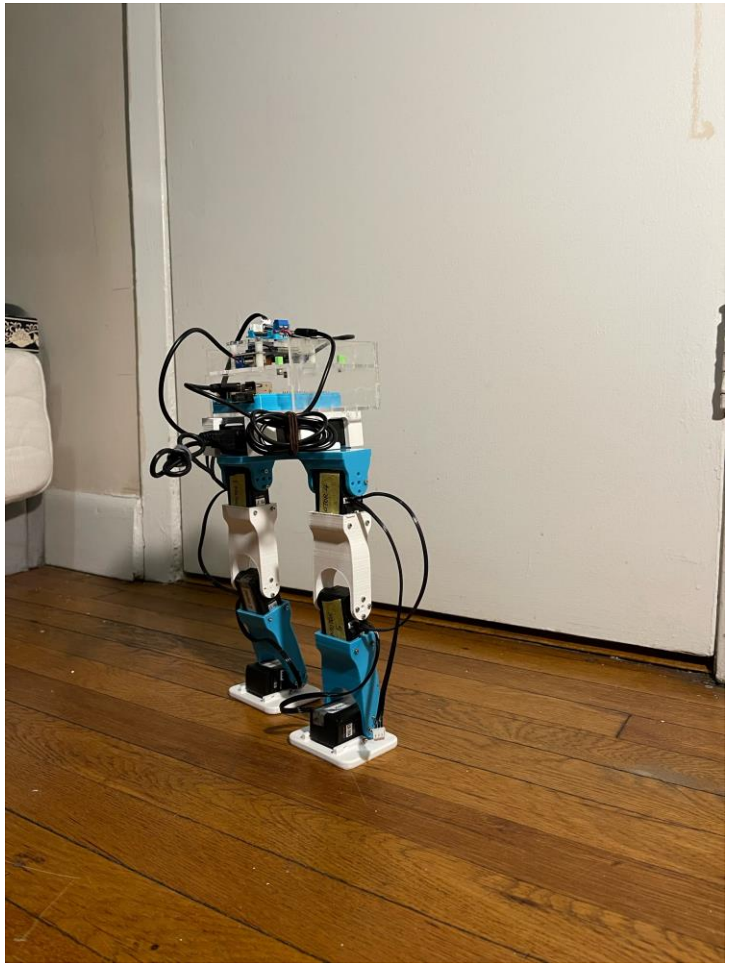
Codes for Boot Test and Homing Test Routine

C: > Users > User > OneDrive > Desktop > walking_test.py > ...

```python

LX16A.initialize("/dev/ttyUSB0", 0.1)

#set limits and functions
def over_temp():
    if LX16A.get_temp()<40:
        return True
    else:
        return False

def over_volt():
    if LX16A.get_vin()<5000:
        return True
    else:
        return False


#check for errors in each motor
for x in range (1,1,6):
    Lx16A(x).servo_mode()
    LX16A(x).enable_torque()
    #errors set
    LX16A(x).set_led_error_triggers(over_temp(), over_volt(), LX16A(x).is_torque_enabled())

#get homing position
    LX16A(x).get_physical_angle() == 120
    while False:
        LX16A(x).move(10,10)

```

Glamour Photo





Robotics Studio MECE 4611 Spring 2023 Semester

Assignment 6 – Baby Steps

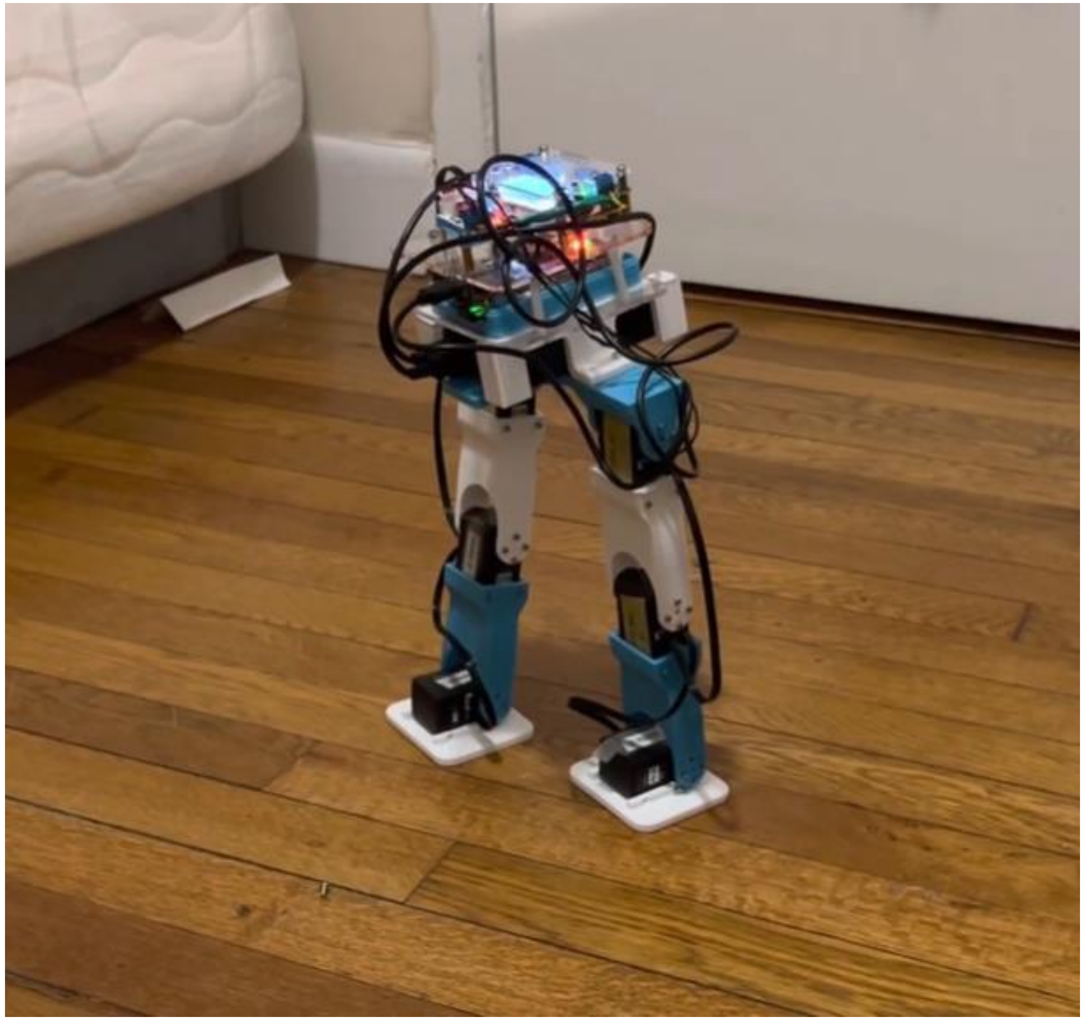
Oluwatamilore Olushina

OTO2109

Submitted April 12, 2023 at 00:57

Birdy-Bot

Leg in Motion – Photos and Video



Approximate speed

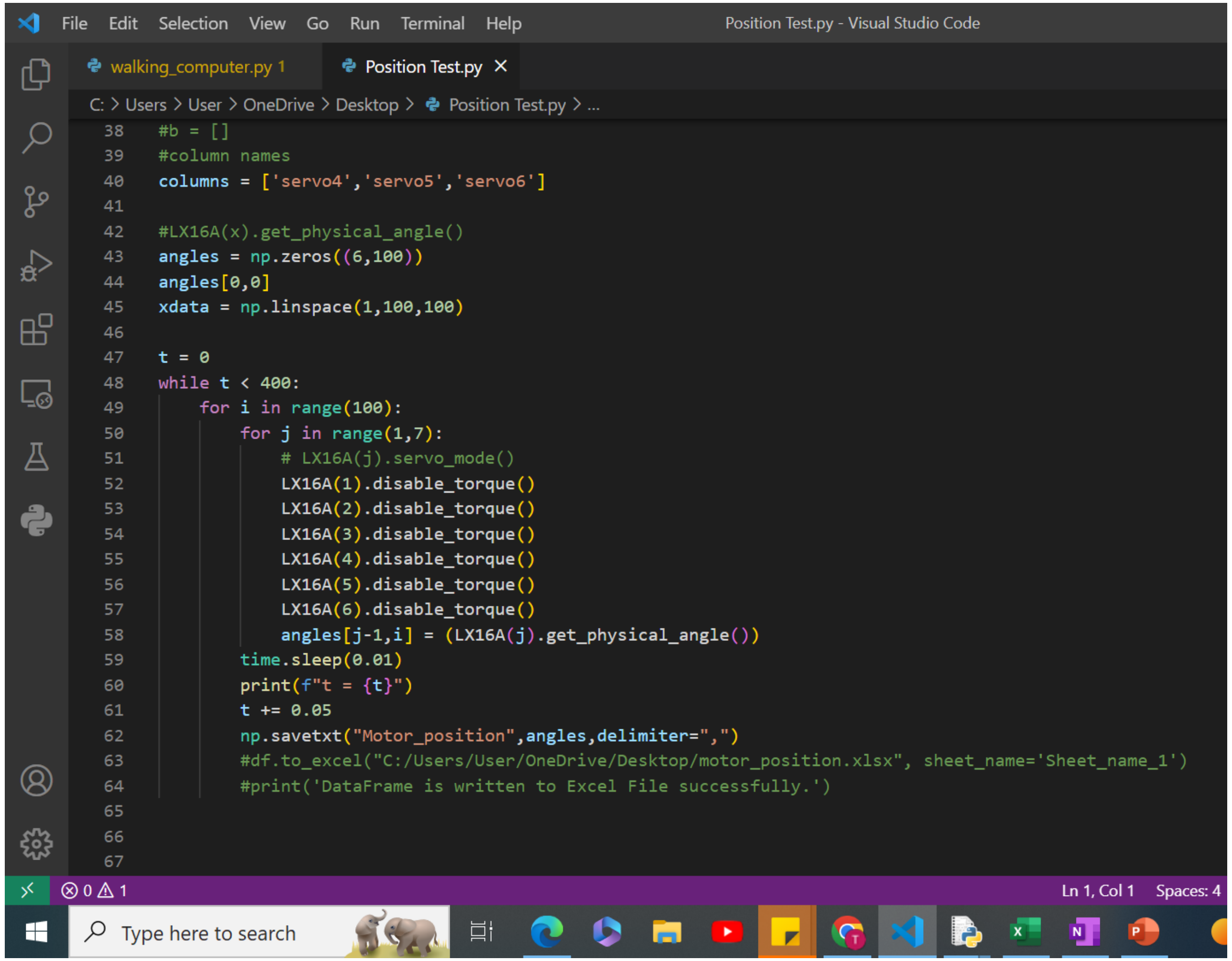
15mm/8 seconds
= 0.001875 m/s

Baby Steps

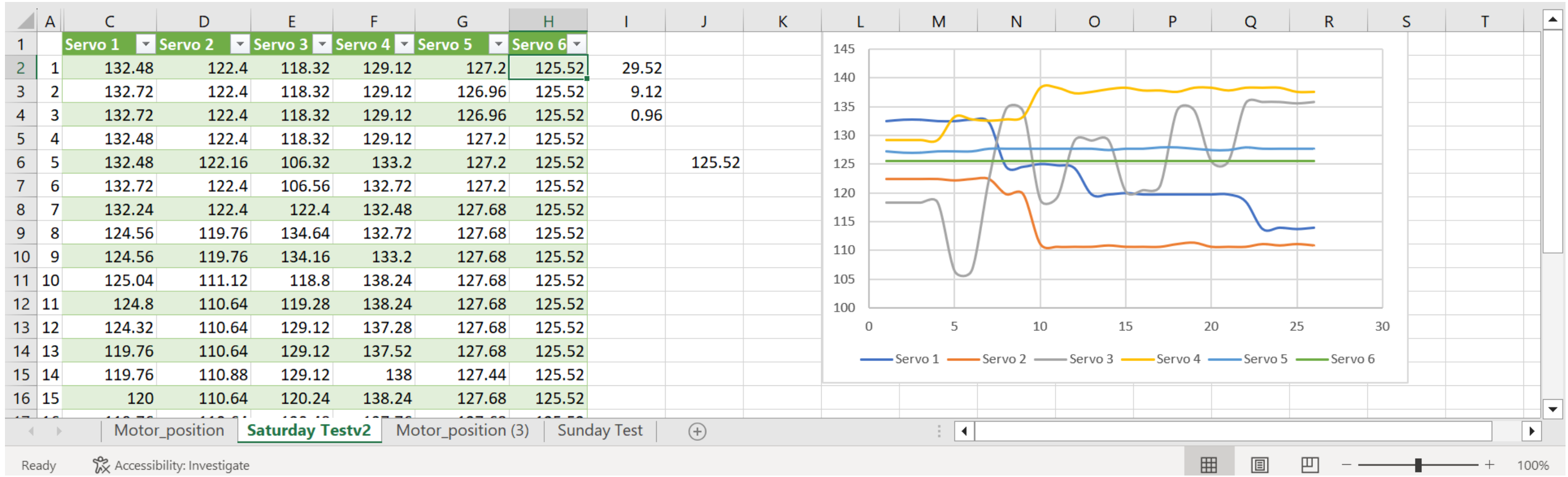
Motor Angles Plotted as a Function of Time

A combination of **keyframing and hand coding** was used to refine the movement of the robot

The Python script Position Test.py was written and used to produce a .csv script containing all the angles produced as the legs moved.



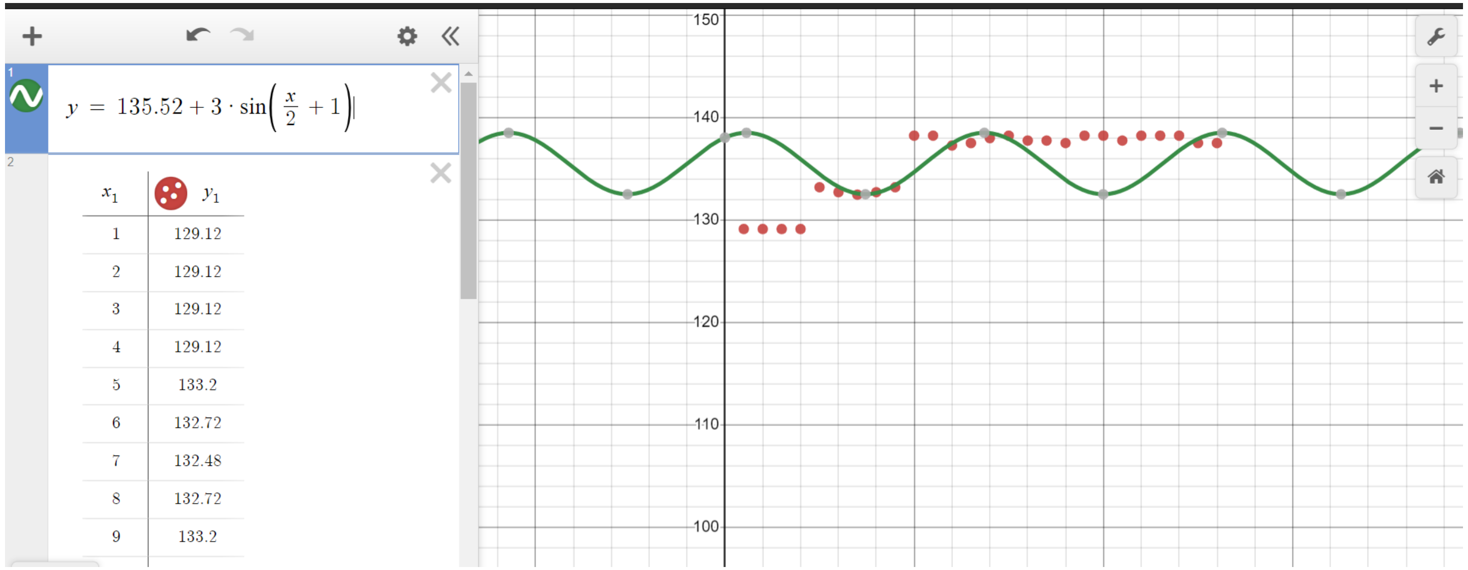
Motor Angles Plotted as a Function of Time



	A	Servo 1	Servo 2	Servo 3	Servo 4	Servo 5	Servo 6	I	J
2	1	132.48	122.4	118.32	129.12	127.2	125.52	29.52	
3	2	132.72	122.4	118.32	129.12	126.96	125.52	9.12	
4	3	132.72	122.4	118.32	129.12	126.96	125.52	0.96	
5	4	132.48	122.4	118.32	129.12	127.2	125.52		
6	5	132.48	122.16	106.32	133.2	127.2	125.52		125.52
7	6	132.72	122.4	106.56	132.72	127.2	125.52		
8	7	132.24	122.4	122.4	132.48	127.68	125.52		
9	8	124.56	119.76	134.64	132.72	127.68	125.52		
10	9	124.56	119.76	134.16	133.2	127.68	125.52		
11	10	125.04	111.12	118.8	138.24	127.68	125.52		
12	11	124.8	110.64	119.28	138.24	127.68	125.52		
13	12	124.32	110.64	129.12	137.28	127.68	125.52		
14	13	119.76	110.64	129.12	137.52	127.68	125.52		
15	14	119.76	110.88	129.12	138	127.44	125.52		
16	15	120	110.64	120.24	138.24	127.68	125.52		

Due to my limited Python proficiency, I double-checked all data using Excel and the Desmos graphing calculator.

When movements were not as smooth as expected, hand coding was employed in the walking script.

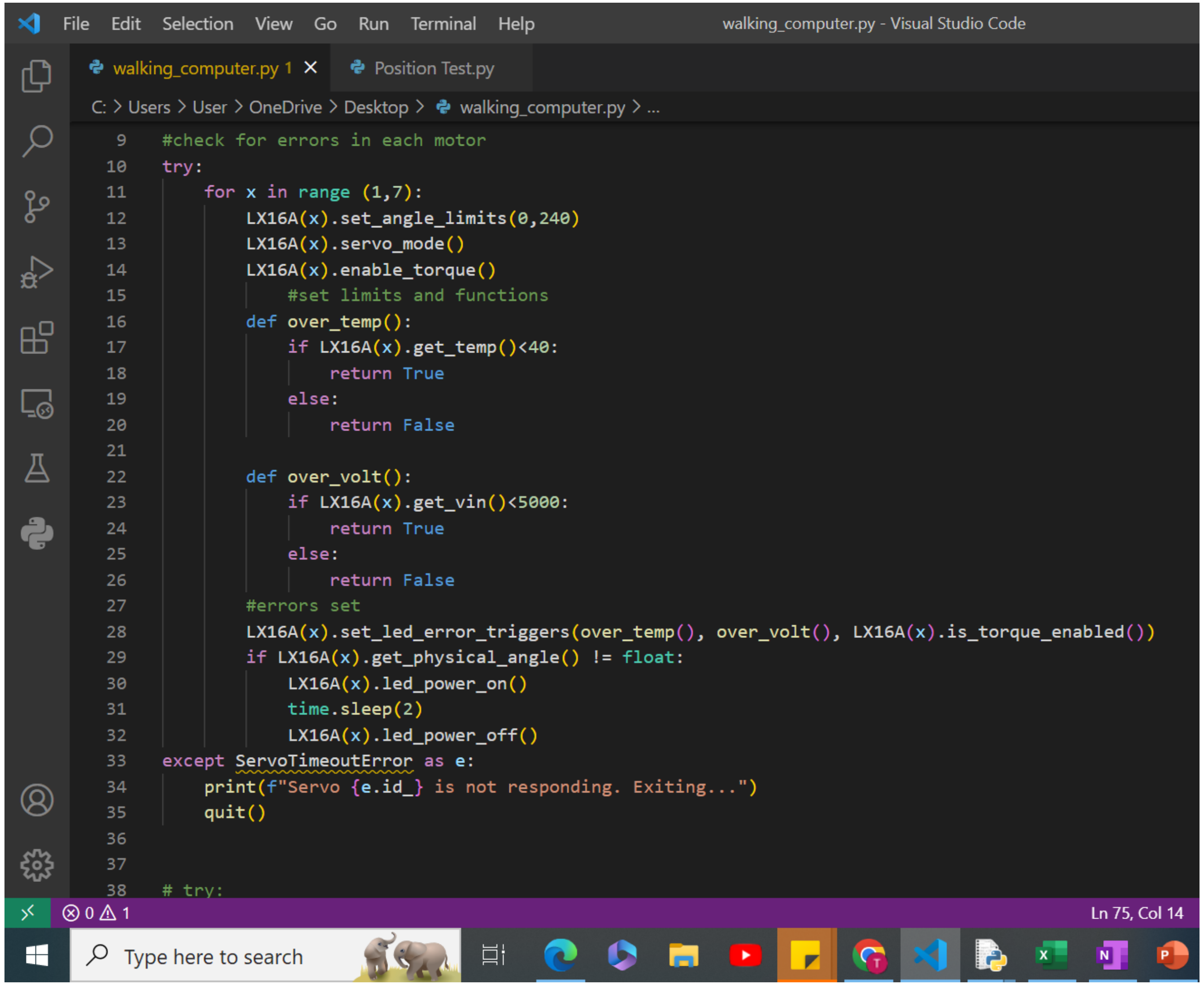


$$y = 135.52 + 3 \cdot \sin\left(\frac{x}{2} + 1\right)$$

x_1	y_1
1	129.12
2	129.12
3	129.12
4	129.12
5	133.2
6	132.72
7	132.48
8	132.72
9	133.2

The speed of the robot was also tweaked as part of each refinement

Ongoing Health Test



A health test routine is implemented at startup and during every rotation taken by the motors.

Python Code

Optimise Line.py - C:\Users\User\OneDrive\Desktop\Optimise Line.py (3.10.11)

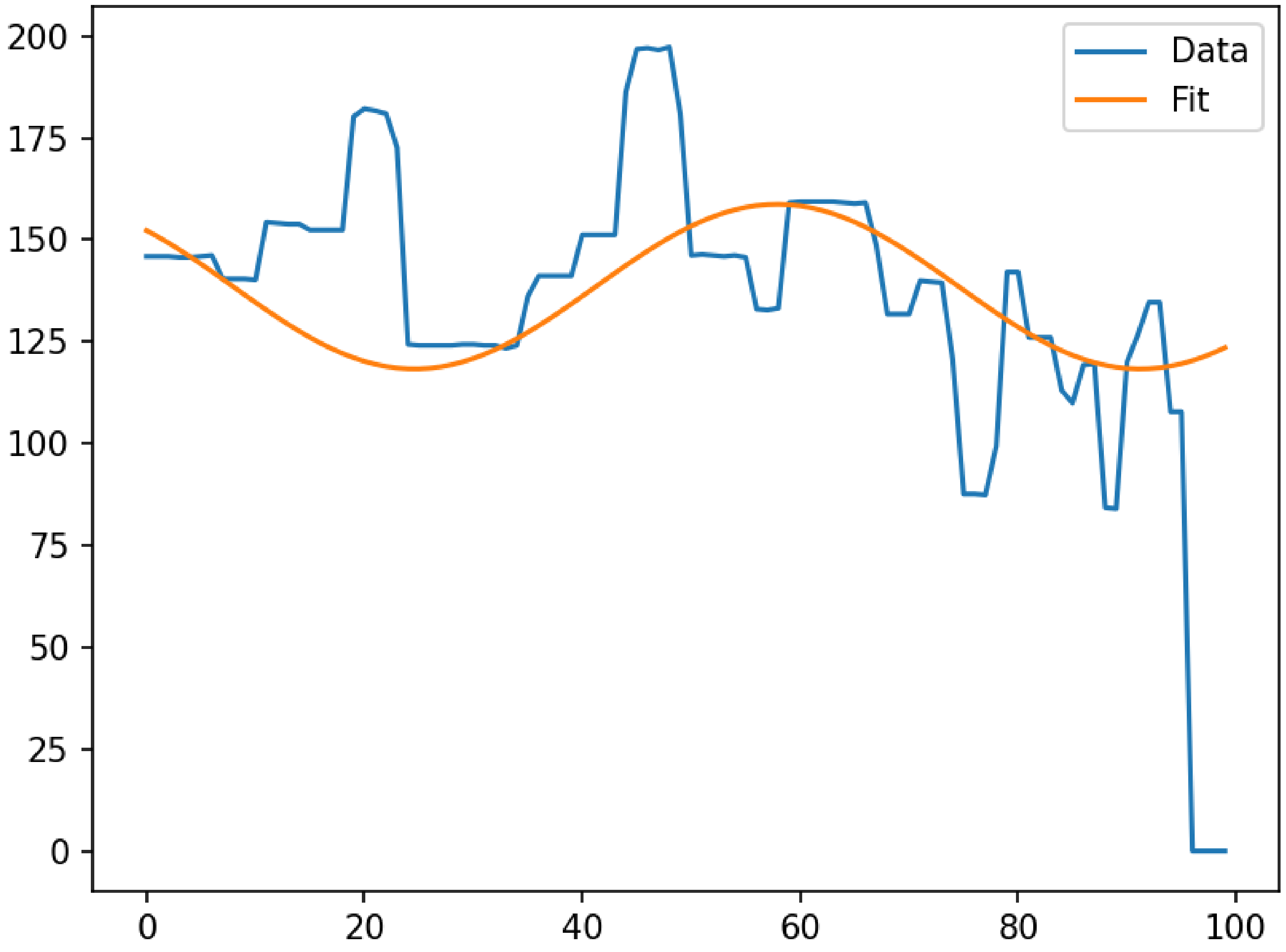
File Edit Format Run Options Window Help

```
import numpy as np
from scipy.optimize import curve_fit
import csv

# Load data from CSV file
with open('C:\\Users\\User\\OneDrive\\Desktop\\Motor_position', 'r') as csvfile:
    csvreader = csv.reader(csvfile)
    header = next(csvreader)
    for i in range(2):
        next(csvreader)
    row_of_data = next(csvreader)


# Convert data to numpy arrays
x_data = np.array(range(len(row_of_data)))
y_data = np.array(row_of_data, dtype=float)

# Define the sine wave function to fit
def sine_func(x, A, B, C, D):
    return A * np.sin(B * x + C) + D

# Fit the sine wave to the data
initial_guess = [15, 0.1, 0, 120]
fit_params, _ = curve_fit(sine_func, x_data, y_data, p0=initial_guess)

# Print the fit parameters
print(fit_params)
print(fit_params[0],"*sin(",fit_params[1],"x+",fit_params[2],")+",fit_params[3])

# Plot the data and fitted sine wave
import matplotlib.pyplot as plt

plt.plot(x_data, y_data, label='Data')
plt.plot(x_data, sine_func(x_data, *fit_params), label='Fit')
plt.legend()
plt.show()
```

Sci Py Optimize Results

Motor 1 – Hip Motor

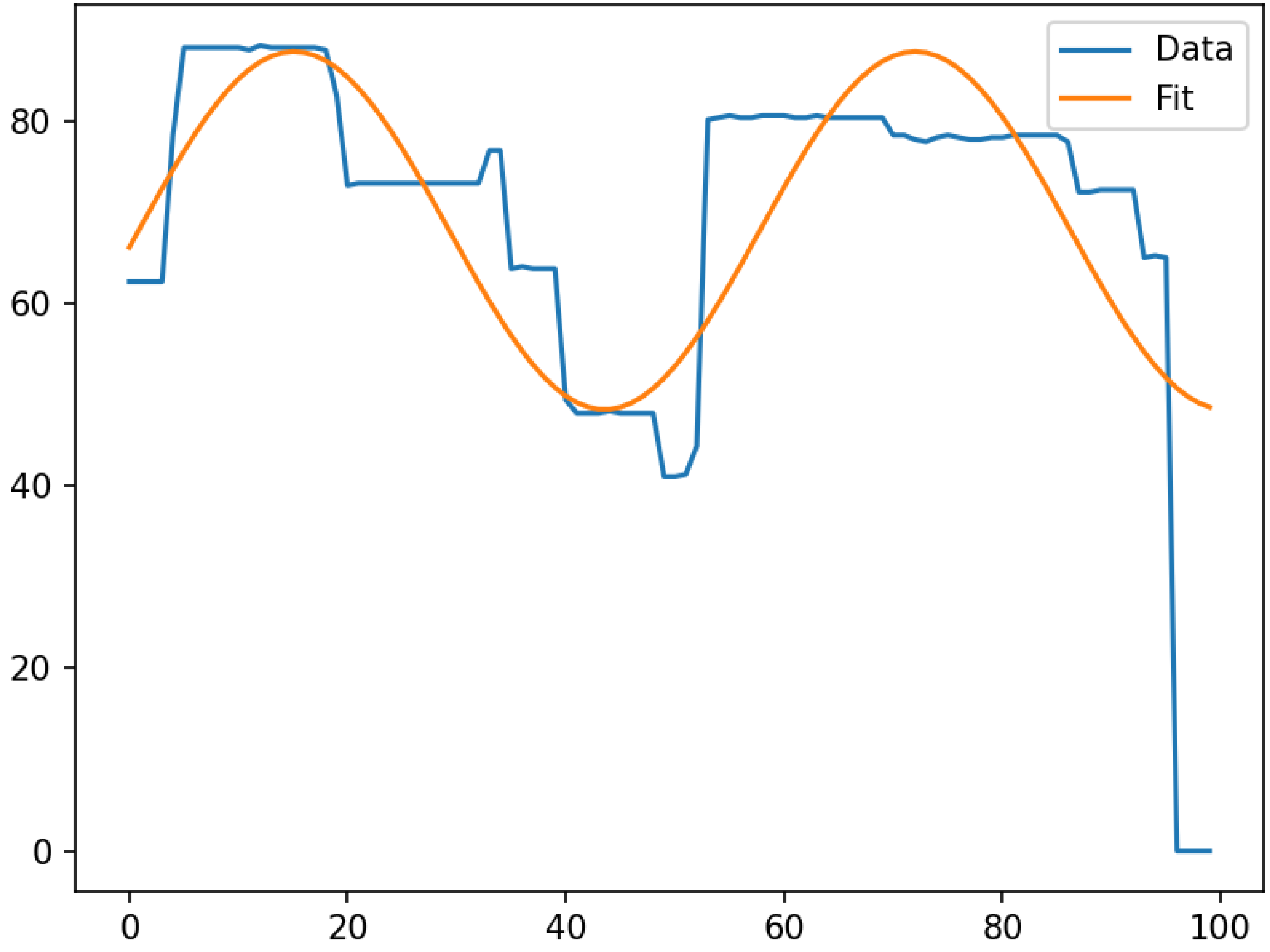
-20.206848310504668 *sin(0.0943411337278227 x+ -7.032845666604736)+ 138.4992143537333 8



Sci Py Optimize Results

Motor 2 – Knee Motor

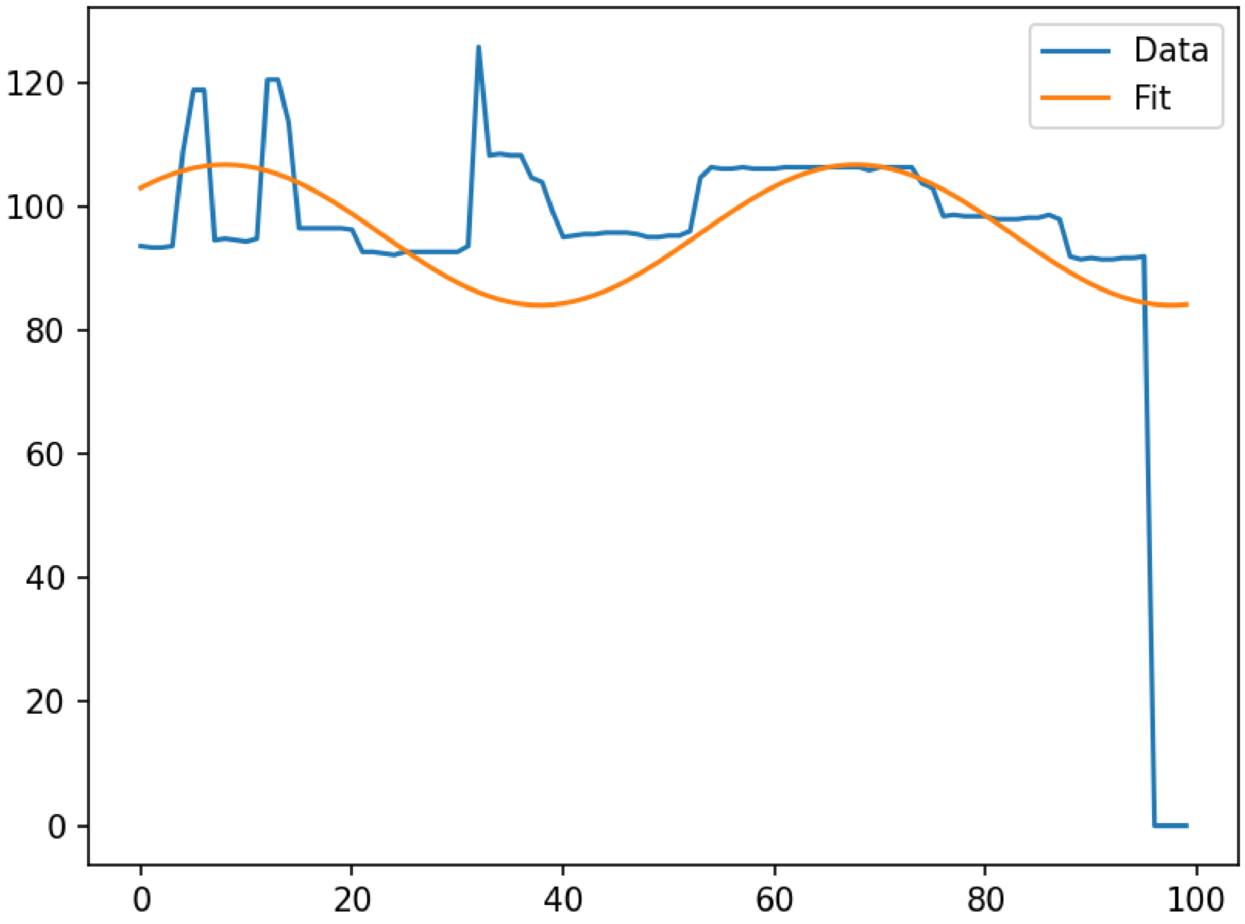
19.63320084540252 *sin(0.11037670352674774 x+ -0.094332233500900 8)+ 68.02379761377823



Sci Py Optimize Results

Motor 3 – Ankle Motor

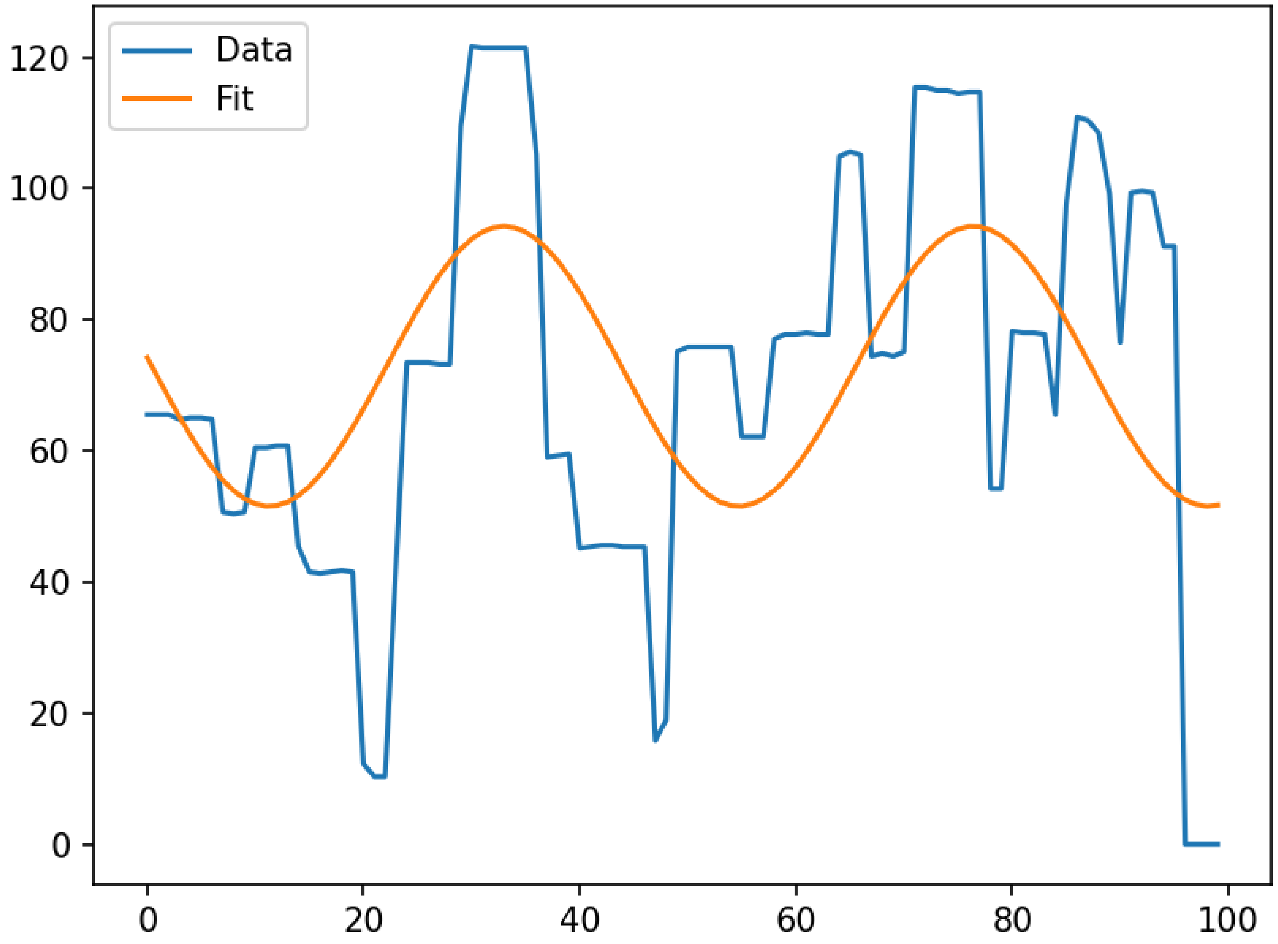
11.35623024189049 *sin(0.10517627325050459 x+ 0.7332456670128624)+ 95.39245192991137



Sci Py Optimize Results

Motor 4 – Hip Motor

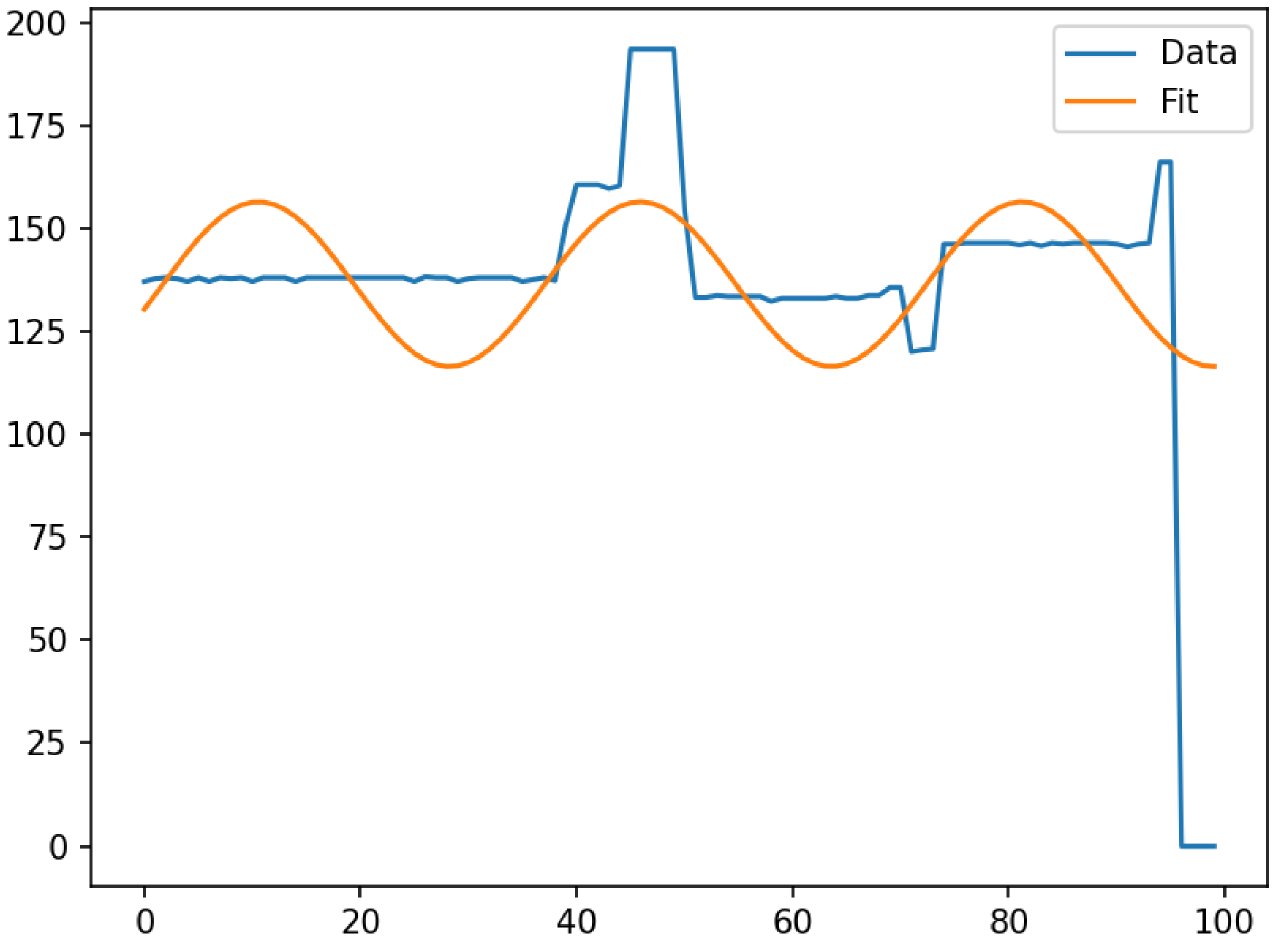
21.356551598817852 *sin(0.1447353184250192 x+ -3.2040717665581293)+ 72.90704232746138



Sci Py Optimize Results

Motor 5 – Hip Motor

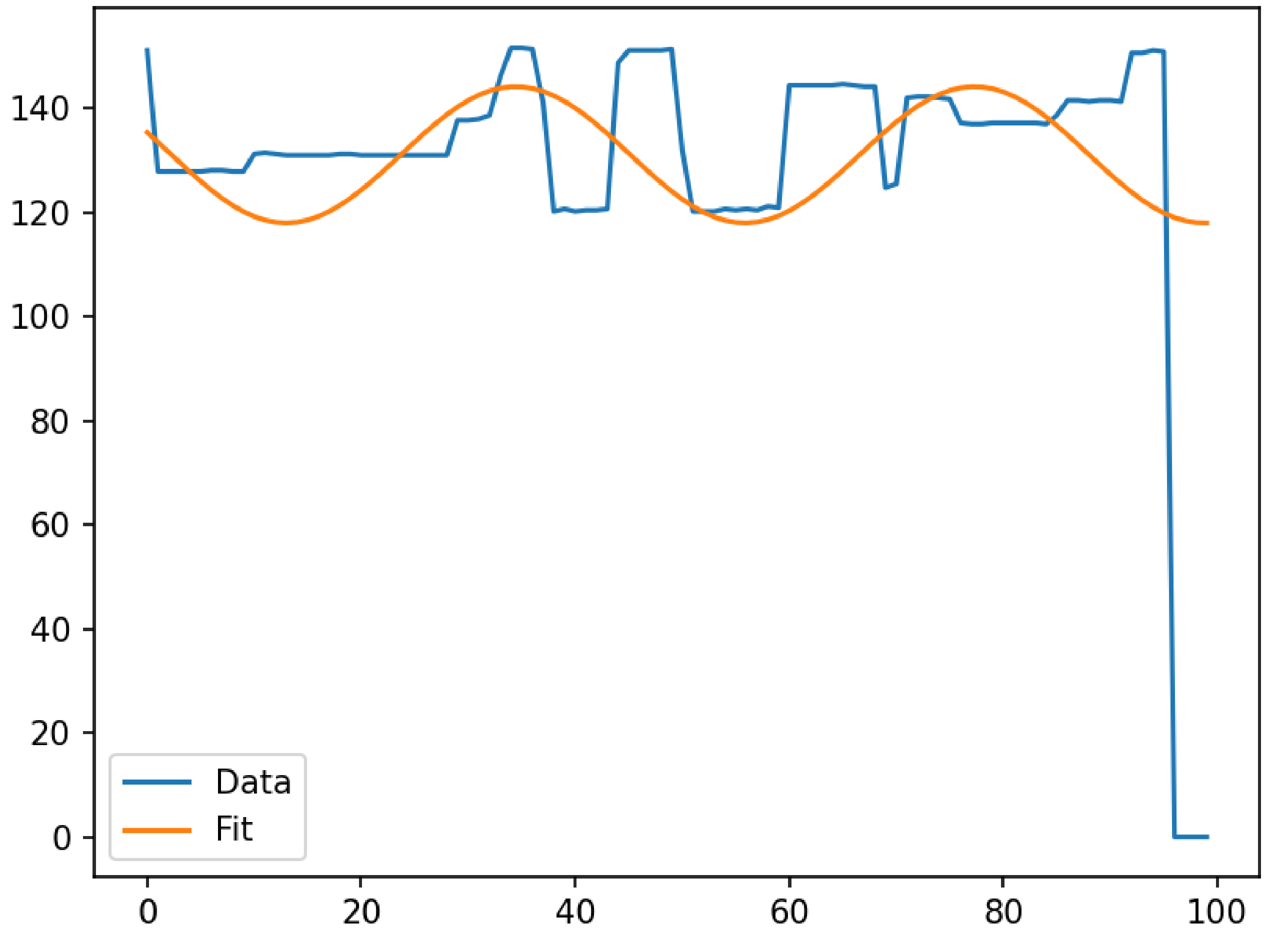
-20.00531735532487 *sin(0.1778011203739185 7 x+ -3.448642069908140 6)+ 136.411906533484 7



Sci Py Optimize Results

Motor 6 – Knee Motor

13.110680916155452 *sin(0.14656961334043 8 x+ - 3.475082762529182)+ 131.1328124662851 5



Walking Speed

11 cm/s

Journey Video