*ECE 1000 Final Report: Robot Arm*

Steven Crocker, Cole Wilson   
Electrical and Computer Engineering (ECE)  
*Tennessee Technological University*Cookeville, TN; United States  
sicrocker42@tntech.edu, clwilson47@tntech.edu

*Abstract*—ECE1000 Robot Arm Final Project. We are doing this project to demonstrate competencies with topics for our class like circuit design. We were successful in moving the arm using a joystick, though it has a currently nonfunctional claw.

Keywords—Robotics, Servos, Raspberry Pi Pico, 3D-printing

# Introduction

Steven Crocker is a Computer Engineering student and Cole Wilson is an Electrical Engineering student. We chose to construct a robotic arm because we both find robots interesting and had previous experience with servo motors. We were supplied with three 9g servos and a Raspberry Pi Pico by our professor, and we were told to 3D print a simple arm and program the Pi.

# Background

First, we normally begin a report by talking about what sources and literature we used to develop the project. For example, in this section you would state what articles, journals, forums, YouTube videos, or people you used to help formulate your code and overall process for the project. Please cite all your sources; plagiarism is a serious concern, and as engineers, we must always give credit where credit is due! Also, these sources might be useful for you in the future, so please make sure to note where your team drew information.

Cole found a design for a 3D-printed arm on Thingiverse [1] that we used as a chassis for our servos to move, and we had the design approved for this project by our professor Dr. Bhattacharya. We differed from the original design a bit in that we used a joystick rather than potentiometers to move the servos as that is what we were supplied. Due to the joystick’s limitations, we could only move 3 servos.

Steven found a YouTube video by Paul McWhorter [2] describing how to code the Raspberry Pi Pico using a joystick. We also talked with our Teaching Assistant John Caleb Williams who helped us a lot with our code due to our limited experience with Python. He got us started with the code for the X- and Z- directions [3] while Steven figured out the Y- direction on his own. He also helped us format our template for the GitHub using his example [4].

# Project Description and formulation

We assembled the servos to the 3D printed chassis that whenever the servo horns moved a part of the arm would move. There were 3 servos total, with 1 rotating the base and another 2 moving the 2 sections of the arm. We chose not to use a 4th servo on the clamp due to time and our sensor’s (joystick’s) constraints. We connected the servo’s power connections to the breadboard receiving power from the Raspberry Pi Pico. The joystick was separate from breadboard and connected directly to the Pi.

A diagram of a breadbox

Description automatically generated

Figure 1 – Simple circuit diagram of power connections. Resistor values are placeholders for internal resistances.

The servos’ signal wires are connected to the GP0, GP1, and GP2 pins according to the order they appear on the robot from bottom to top. These pins receive information from pins GP26, GP27, and GP16 receiving information from the joystick’s X-position, Y-position, and Button (Z) respectively.

A diagram of a circuit

Description automatically generated

Figure 2 – Simple circuit diagram for logic connections

When the joystick is moved, the X-Y positions are shown on the computer screen and those same values go through the Pi code to tell the servo to change its position accordingly. The button (Z) is a bit different since it only has an on and off feature, so we coded it to go from its min to the halfway when the button is pressed, then return to the halfway point when the button is released.

A white tower on a blue table with wires

Description automatically generated

Figure 3– picture of arm, Pi, and computer

# IV. Discussion and results

We were able to move the 2 segments of the arm and rotate it about the base. Even though the claw was not functional, were pleased that the arm itself worked and had a good range of movement. If we were to continue working on this project in the future, we may get more or different sensors to be able to control four servos at once to have a functional claw. We would also like to increase the range of our third topmost servo to cycle from min to max position, but we were unable to complete that due time constraints.

Our team enjoyed assembling the arm and testing it. It was pretty neat seeing everything come together mechanically and electronically, and when we were able to see our code do physical work it was exciting. Learning how to code the Raspberry Pi and seeing how all the pins work was neat.

Steven worked primarily on the coding portion and wiring connections of the project. Cole worked primarily on the 3D-printing and mechanical assembly of the arm portions of the project. Documentation was done primarily by Cole while initial testing and the video was done primarily by Steven.

# V. Conclusion

Steven and Cole learned how to research sources and cite them as applicable for engineering projects. We also learned more about how Python works when applied to running servo motors and using a joystick. We learned more about how servo motors themselves work and how to connect them to a microcontroller and how their rotation affects physical hardware. The movement of our arm using the 3 servos was satisfactory for us.

##### References

[1] Thingiverse.com, “Simple Robot Arm Using Standard Servos, No Hardware by chrisingham05,” *Thingiverse*. https://www.thingiverse.com/thing:4316282 (accessed Apr. 20, 2024).

‌[2] “Raspberry Pi Pico W LESSON 58: Determine Angular Position of a Joystick in MicroPython,” *www.youtube.com*. https://youtu.be/KpDIv0i41Tw (accessed Apr. 20, 2024).

[3] JCWilliams1003, “ECE-1000-Spring-2024-Final-Project-Insert-Project-Name/Example Micropython Codes/ECE\_1000\_Joystick\_Servo\_Example.py at main · JCWilliams1003/ECE-1000-Spring-2024-Final-Project-Insert-Project-Name,” *GitHub*, 2024. https://github.com/JCWilliams1003/ECE-1000-Spring-2024-Final-Project-Insert-Project-Name/blob/main/Example%20Micropython%20Codes/ECE\_1000\_Joystick\_Servo\_Example.py (accessed Apr. 22, 2024).

[4] J.-C. (JC) Williams, “JCWilliams1003/ECE-1000-Spring-2024-Final-Project-Insert-Project-Name,” *GitHub*, Apr. 15, 2024. https://github.com/JCWilliams1003/ECE-1000-Spring-2024-Final-Project-Insert-Project-Name/tree/main (accessed Apr. 22, 2024).

‌

‌

‌