ECE 4950 - Project 1
The Big Lebowski: Autonomous Bowling Robot

Team 19

Corey Golec, John Phillips, Jacob Criswell, Avi Patel, Josue Morales-Sanchez

Date: 9/19/25

Summary

The objective of Project 1 was to show that we could connect and control a sensor and motor using MATLAB Simulink software and an Arduino, while also gaining experience with 3D modeling. We used a potentiometer, which is a dial that changes voltage, to send information into Simulink. The software displayed this information as both numbers and a graph. We then connected a small servo motor, which was first controlled by a simple on/off switch and later set up to turn automatically once the dial passed a certain point.

In addition to the hardware demonstration, a custom 3D part was designed in Autodesk Inventor with approximate dimensions of 4” x 2” x 1” with our team number embossed. This introduced the basic skills of manufacturing and showed how CAD tools can be used for our future projects. Through these milestones, our team confirmed that we were able to communicate with the Arduino through MATLAB and Simulink to acquire sensor data and control actuators in real time while also introducing us to the process of making rapid prototype tools in 3D modeling software.

Materials and Methods for Sensor/Actuator Demonstration

In order to demonstrate our ability to interface with a sensor and actuator using Simulink and Arduino, we used a potentiometer and a servo motor in combination with an Arduino MEGA 2560 and Simulink. The potentiometer was used to vary an analog input signal going into the Arduino, which was then compared to a threshold value in Simulink. If the input value went past a set threshold, the servo was sent a signal that spun it 180 degrees. Once the input signal dropped back below the threshold, the servo would spin back to the original 0-degree position.

Equipment used included a 5 kΩ potentiometer component, a Towerpro MG92B servo motor, the Arduino MEGA 2560, various jumper wires, and MATLAB Simulink. Figure 1 below shows the constructed circuit. The potentiometer was energized with the 5V pin from the Arduino, and the wiper pin on the potentiometer was connected to pin A0 on the Arduino. We powered the servo motor using the 5V pin on the Arduino and connected the signal wire to PWM pin 9 on the Arduino. All the grounding wires were connected back into the Arduino’s GND pin. With regards to the software, the circuit and program were tested on both MATLAB 2025a and MATLAB 2021b with success.

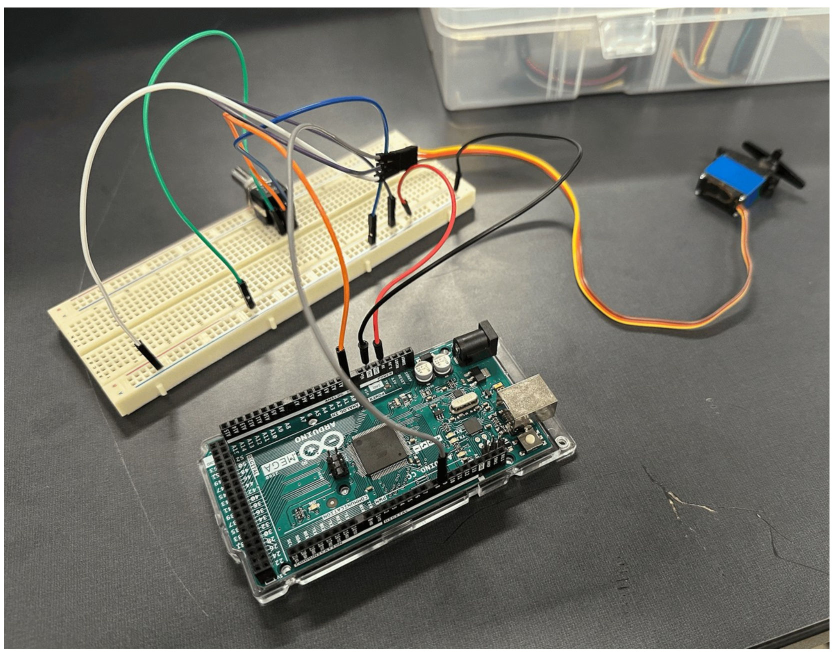

Figure 1. Final constructed circuit for controlling the servo with a potentiometer

With the circuit constructed, we next designed the Simulink program, which would allow us to control the servo motor using the potentiometer output. Figure 2 shows the Simulink program used to accomplish this task. The analog input from the potentiometer coming from A0 is fed into a switch, which compares the input to a preset threshold value. When the analog input rises above the threshold value, the switch will send a constant signal of 180 to a data conversion block, which converts the constant to a number that can be read by the servo write block. The servo write block then reads the constant value and turns the servo to the position specified by the constant block, which in this case is 180°. Once the input falls back below the threshold value, the other constant will be sent to the servo write, returning it to 0°.

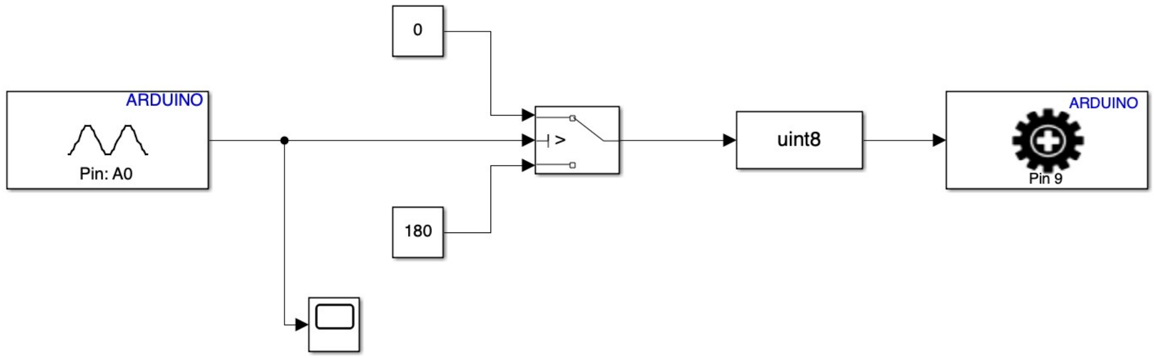

Figure 2. Simulink diagram which controls servo motor based on potentiometer output

In order to test our circuit, we first connected a scope to the analog input in order to verify the signal coming into the Arduino. An example of the potentiometer input can be seen in Figure 3. After verifying the analog input was being read by the Arduino, we then ensured that the servo motor was spinning after the threshold of 450 was reached. This action was successfully verified on two different computers.

Results and Discussion

In Figure 3 below, we illustrated that we were able to read analog inputs via the analog A0 pin on the Arduino. This was shown on the graph displaying ADU (Analog-to-Digital Units) vs time. As the potentiometer was rotated clockwise, the ADU values increased; when rotated counterclockwise, the values decreased. The observed range was from 0 to 1023, confirming that the ADC operates at 10 bits, where 0 corresponds to 0 V and 1023 corresponds to 5 V.

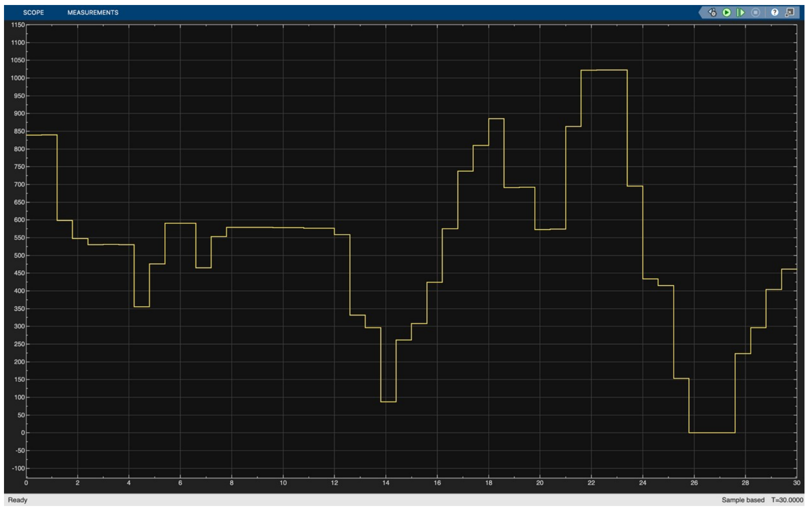

Figure 3. Example of analog input received by Arduino from a potentiometer

The next experiment involved manually commanding the servo to turn on and off. This was achieved using a Manual Switch block and a buffer. The Manual Switch toggled between 0 (off) and 1 (on), feeding into a buffer set to 90, which represented the desired servo angle in degrees. The output signal was sent through Pin 9, successfully rotating the servo to 90°. Figure 4 shows the Simulink structure used to implement this control.

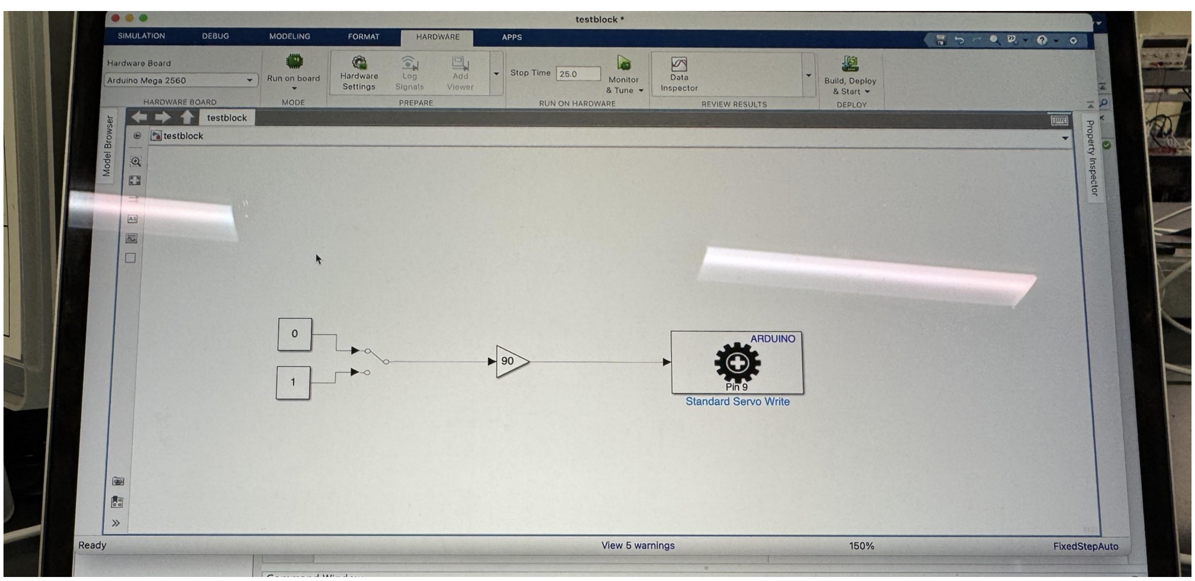

Figure 4: Manual Switch Simulink

After successfully testing all input and output connections and verifying that the system can reliably receive signals from the potentiometer, we combined both experiments (Figure 2) and updated the Simulink file. This file includes a switch block that continuously monitors the analog input until it crosses a threshold of approximately 480 ADU . Once this threshold is exceeded, the Arduino sends a digital signal that commands the servo to rotate to 180°. When the input falls below 480 ADU, the servo returns to 0°.

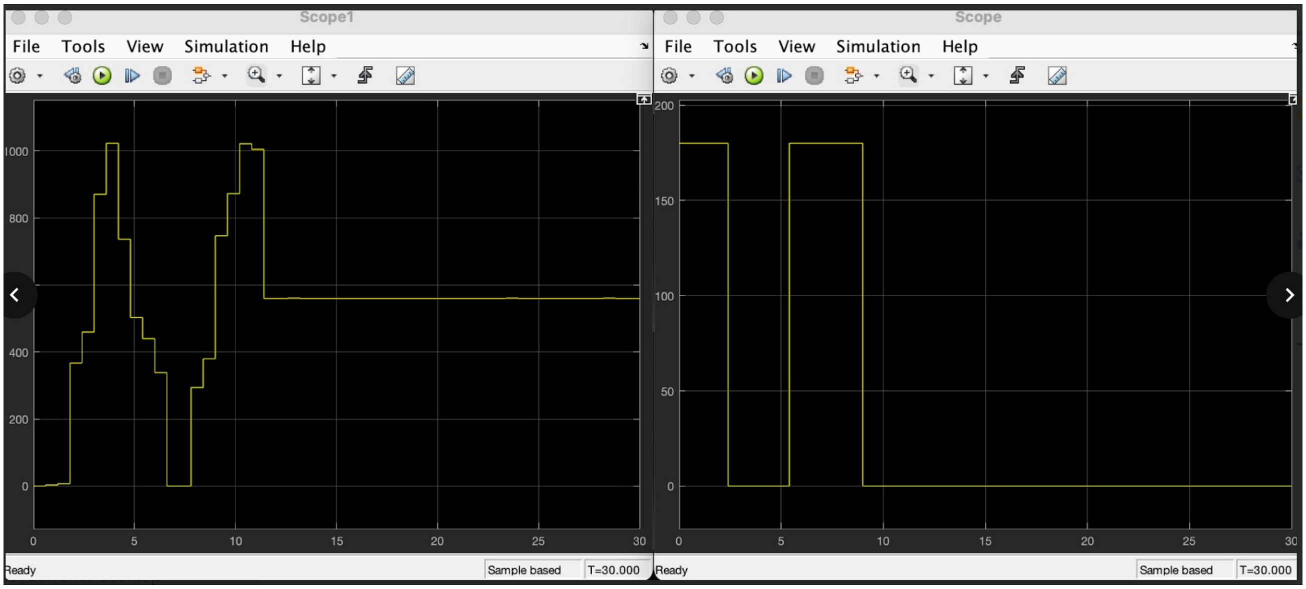

Figure 5: Input and Output Side By Side

From the graph in Figure 5, we can observe that once the input signal crosses above 480 ADU value, the output signal to the servo becomes set to 180. Once we are below that 480 value, the servo returns to 0. The graph shows this relationship clearly, with the output signal being a cleaner version of the input signal. However, there is a delay of 1 second. Note that both the threshold value and the degree that the servo turns to can be modified on Simulink.

Conclusions

We recommend the use of MATLAB Simulink for interfacing with Arduino hardware. The combination correctly read values from the potentiometer and successfully controlled the servo if the threshold was crossed. For learning how components communicate with each other and the design prototype stage, this setup works rather well because Simulink is an extremely high-level modeling environment with an easy to learn block diagram interface. Parameters can easily and quickly be adjusted to test many situations. However, there are some limitations to consider for the later stages of projects. The servo response has a one-second delay, which could increase depending on your block diagram setup. This could affect applications where speed and precision are critical. Also, the Arduino has a limited output current and processing power, so larger or more advanced projects would require stronger hardware or a dedicated controller.

The 3D printer has a wide range of uses for future projects. It can be used to rapidly create designed parts like motor mounts and housings for all stages of the project. This allows us to validate shapes and dimensions before committing to expensive or time consuming manufacturing methods. We can also add details like team numbers or logos, which makes the parts both functional and customized for our project.

Working through the project helped us learn and reinforce how to brainstorm and prototype as a newly formed team. Beginning with research on how our components functioned and testing them individually made it easy to confirm each part of the circuit worked before adding another component. This approach helped us find issues that arose early on and made the final design very reliable. Working as a team emphasized how important it is to communicate and establish roles to keep the project on track and complete it in a timely manner.

Score	Pts		
	15	**General Format - Professional Looking Document/Preparation (whole document)** a) Fonts, margins (11pt, times new roman, single spaced. 1" margins on all sides). b) Spelling and grammar are correct c) Layout of pictures – all figures need numbers and captions and must be referenced in the text d) Follows the page limitations below. e) References. Use IEEE reference format. f) This grading sheet is included as the final page.	**O3-SA1:1**
	15	**Page 1: Title, Group Name, Group Members, and Date** **Executive Summary** (~1/3 of the page) Provide a summary of the whole project. Use language that targets **a non-technical audience**. An important skill for an engineer is to communicate complex technical information to a general audience that may be involved in decision making, e.g. marketing. Important criteria: a) Can a non-technical audience (~ high-school degree) read this section and understand your goals, procedures, and conclusions? b) Use simple words and graphics to help explain	**O3-SA1:1**
	30	The next sections of the report follow the standard **laboratory report format**. **Page 2: Materials and Methods for the Sensor/Actuator Demonstration** (1 page) You are establishing the credibility and usefulness of your results by providing all the details so that someone else could repeat your experiment. As an example, MATLAB 2021a may behave differently than MATLAB 2021b – the software version information which would be required to reproduce your result should be included. This section should answer the following: a) What equipment is used, include software versions. **b)** How were the experiments conducted? How is the equipment connected and used? Describe the instrumentation, cables, connections, and experiments using diagrams and photos. You should have drawings (pin connection and connector part numbers) **Pages 3-4: Results and Discussion for the Experiments** (~1 page) Describe what you have done. Include plots (from MATLAB, not photos of the Target screen) for each of the three experiments and a brief discussion of how you interpret the result. Did you demonstrate (through your documentation) that the equipment has been configured and used correctly?	**O2-SA2:3** **O1-SA5:4**
	5	**Page 4-5: Conclusions and References** (~ 1 page) a) Based on this experiment, do you recommend this equipment for use in a robot control project? What are the possible limitations? Your results and observations should be the basis for your conclusions. (~1/2 page) b) What are the possible uses for the 3D Printer in your projects? (~1/4 page) c) What did you learn from the process of determining your final project design? **Page 6: This Grading Sheet**	**O3-SA1:1**
	5	**3D Designed Part** Grading based on: a) Does the design meet the requirements? b) Originality and creativity	**O7-SA1:1**
	10	**Demo Slide Show Presentation** (~3 slides) a) What did you do to complete this project b) How did all the team members get involved c) Photos of the designed 3D model	