Preliminary Design Document

The Chunky Caterpillar

Team C09: Ray Daw, Madi Gajula, Ibrahim Islam, Hongze Khor

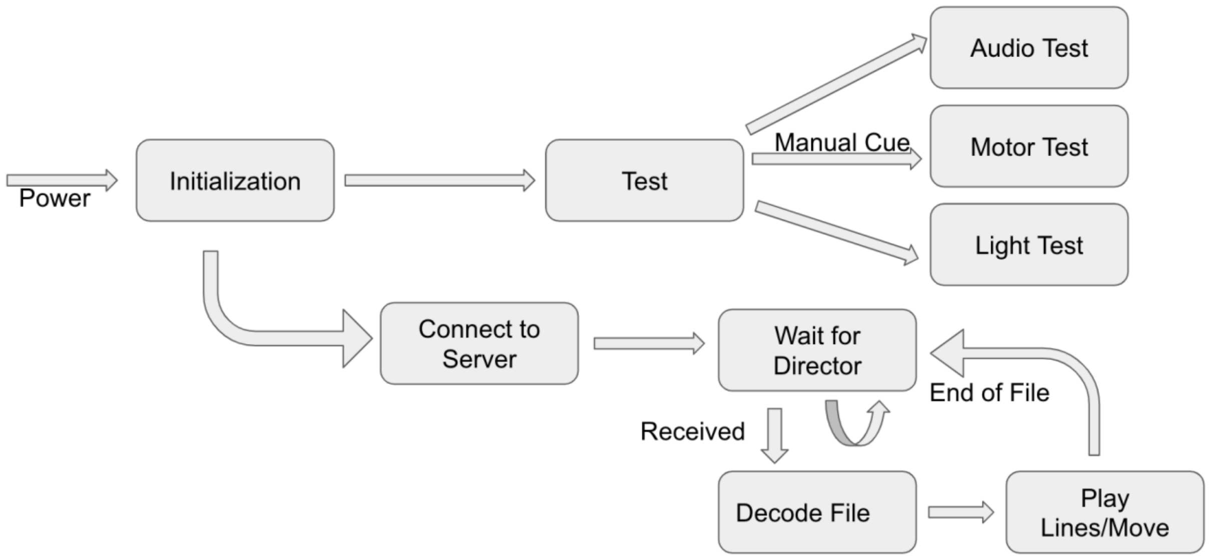
Date: September 24, 2022

Project Description

Our team's project is an implementation of the Caterpillar from Alice in Wonderland. We created Caterpillar to communicate with the director via an Internet connected server as well as to "smoke" out of a hookah using a smoke machine. The Caterpillar sits on a movable mushroom stage. This allows the Caterpillar to move up and down and ensures the circumference of the mushroom stage to light up.

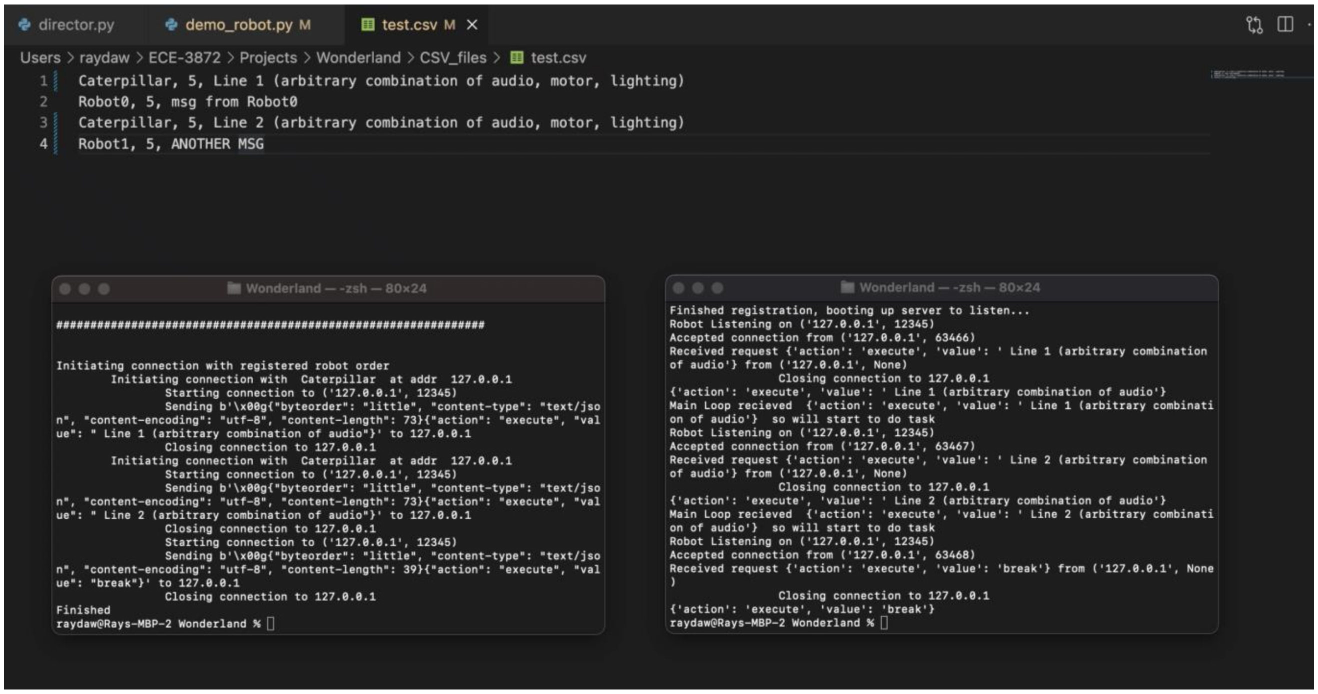
Software Design

Upon turning on the robot, the robot will “initialize.” Right now, this simply looks like getting power to the microcontroller and all the peripherals (lights, servos, audio system, etc.), and the controller connecting to the peripherals. After this there will be some sort of physical user interface on the robot that allows for the user to enter a testing mode, and individually toggle and test the audio, motor, and lighting systems on the robot. When the robot is toggled for “showtime,” the pi will connect to the server, which is connected to the director that will have our script uploaded in csv files. The Pi will wait for the director, and once a file is received, decode that file, play the corresponding lines, movements, and lighting patterns, and then go back to waiting for the director. This loop will continue until the robot is shut off.



Software Simulation

To simulate the communication of the director and the robot, we edited demo_robot.pi director.pi with the appropriate host and port info and edited the test CSV file to send a few commands to the renamed robot. The messages sent to the robot from the CSV file are placeholders for the audio, lighting, and motor commands that will be coded using the GPIOZero python library. The simulation (terminal pictured below) shows that the robot can correctly connect to the director, receive commands specifically for the correct robot name, execute those commands, and then disconnect from the director.


director.py
demo_robot.py M
test.csv M
Users > raydaw > ECE-3872 > Projects > Wonderland > CSV_files > test.csv
Wonderland — -zsh — 80×24
Wonderland — -zsh — 80×24

```
Caterpillar, 5, Line 1 (arbitrary combination of audio, motor, lighting)
Robot0, 5, msg from Robot0
Caterpillar, 5, Line 2 (arbitrary combination of audio, motor, lighting)
Robot1, 5, ANOTHER MSG
```

```
################################################################


Initiating connection with registered robot order
        Initiating connection with  Caterpillar  at addr  127.0.0.1
                Starting connection to ('127.0.0.1', 12345)
                Sending b'\x00g{"byteorder": "little", "content-type": "text/jso
n", "content-encoding": "utf-8", "content-length": 73}{"action": "execute", "val
ue": " Line 1 (arbitrary combination of audio"}' to 127.0.0.1
                Closing connection to 127.0.0.1
        Initiating connection with  Caterpillar  at addr  127.0.0.1
                Starting connection to ('127.0.0.1', 12345)
                Sending b'\x00g{"byteorder": "little", "content-type": "text/jso
n", "content-encoding": "utf-8", "content-length": 73}{"action": "execute", "val
ue": " Line 2 (arbitrary combination of audio"}' to 127.0.0.1
                Closing connection to 127.0.0.1
                Starting connection to ('127.0.0.1', 12345)
                Sending b'\x00g{"byteorder": "little", "content-type": "text/jso
n", "content-encoding": "utf-8", "content-length": 39}{"action": "execute", "val
ue": "break"}' to 127.0.0.1
                Closing connection to 127.0.0.1
Finished
raydaw@Rays-MBP-2 Wonderland % 
```

```
Finished registration, booting up server to listen...
Robot Listening on ('127.0.0.1', 12345)
Accepted connection from ('127.0.0.1', 63466)
Received request {'action': 'execute', 'value': ' Line 1 (arbitrary combination
of audio'} from ('127.0.0.1', None)
                Closing connection to 127.0.0.1
{'action': 'execute', 'value': ' Line 1 (arbitrary combination of audio'}
Main Loop recieved  {'action': 'execute', 'value': ' Line 1 (arbitrary combinati
on of audio'}  so will start to do task
Robot Listening on ('127.0.0.1', 12345)
Accepted connection from ('127.0.0.1', 63467)
Received request {'action': 'execute', 'value': ' Line 2 (arbitrary combination
of audio'} from ('127.0.0.1', None)
                Closing connection to 127.0.0.1
{'action': 'execute', 'value': ' Line 2 (arbitrary combination of audio'}
Main Loop recieved  {'action': 'execute', 'value': ' Line 2 (arbitrary combinati
on of audio'}  so will start to do task
Robot Listening on ('127.0.0.1', 12345)
Accepted connection from ('127.0.0.1', 63468)
Received request {'action': 'execute', 'value': 'break'} from ('127.0.0.1', None
)
                Closing connection to 127.0.0.1
{'action': 'execute', 'value': 'break'}
raydaw@Rays-MBP-2 Wonderland % 
```

Hierarchical Design

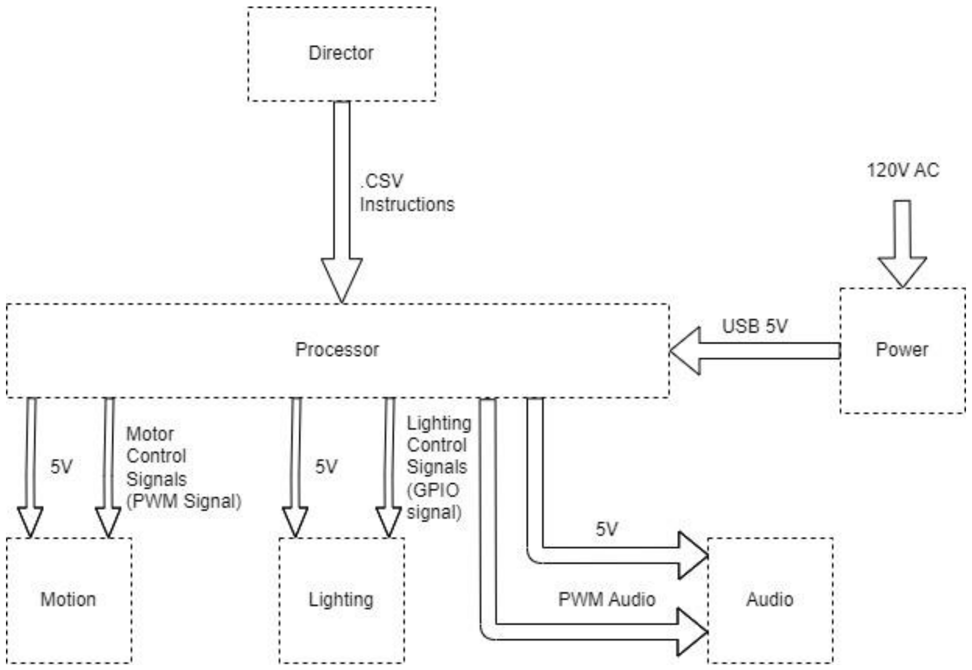


Figure 1. System Diagram

Sub-System Design

There are five sub-systems contained in our design: power, audio, motion, lighting, and software. The power will supply power to the processor, the audio will output pre-recorded audio lines, the motion will control the movement of the caterpillar, the lighting subsystem will control the LEDs and lighting patterns, and the software serves to communicate instruction to the rest of the sub-systems.

Power Sub-System

The power sub-system will take 120V in from the wall and output 5V to the Raspberry Pi. It will go through a USB adapter that will have an inline switch. There will also be a fuse to protect the components.

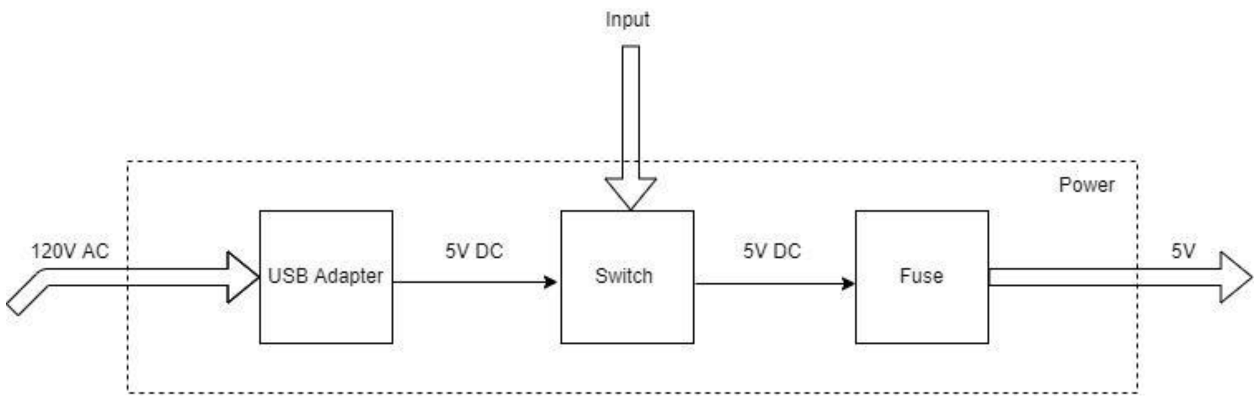


Figure 2. Power Sub-System Diagram

Audio Sub-System

The audio sub-system will take in a PWM signal from the raspberry pi and amplify the signal using a class D amplifier and output it through a speaker.

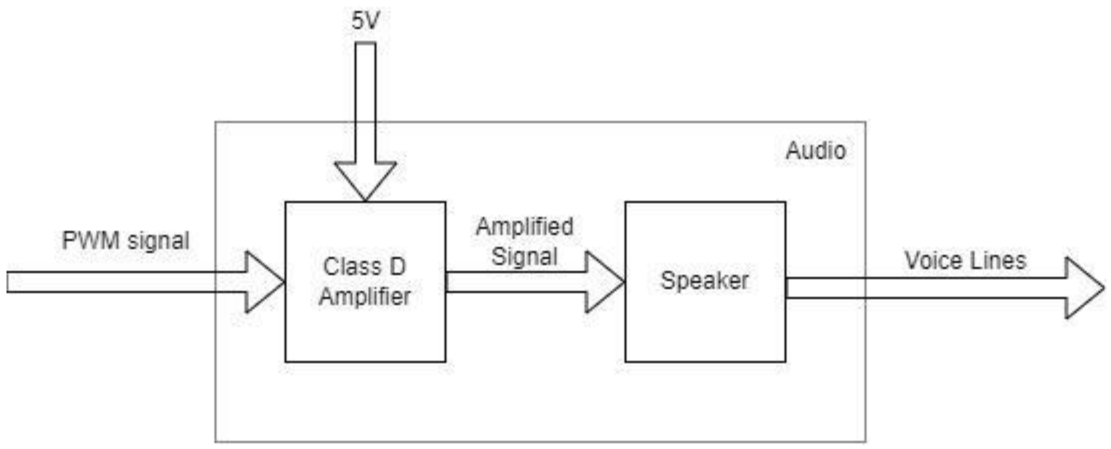


Figure 3. Audio Sub-System Diagram

Motion Sub-System

The motion sub-system will control a servo to allow the caterpillar to move vertically up and down.

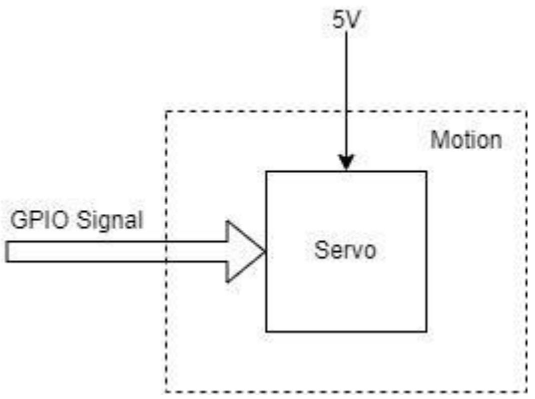


Figure 4. Motion Sub-System Diagram

Lighting Sub-System

The lighting sub-system will take direction from the raspberry pi and output different lighting patterns.

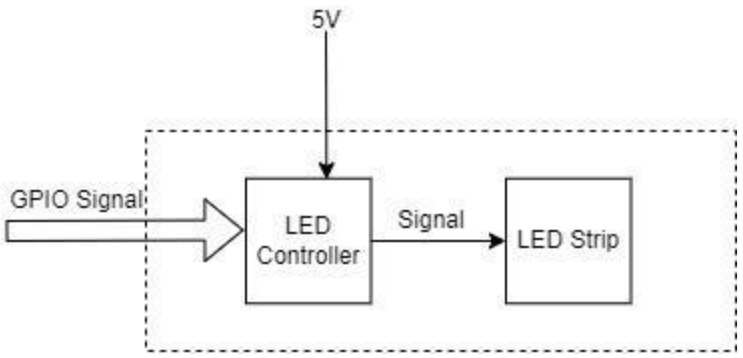


Figure 5. Lighting Sub-System Diagram

Electronic Design

Our electrical design is based on a Raspberry Pi 4B microcontroller and a custom PCB. The PCB was designed to fit onto the header of the Pi board, so there were not any excess connection outside of the wires directly routed to the speaker, servo, LED strip, and control buttons/potentiometer.

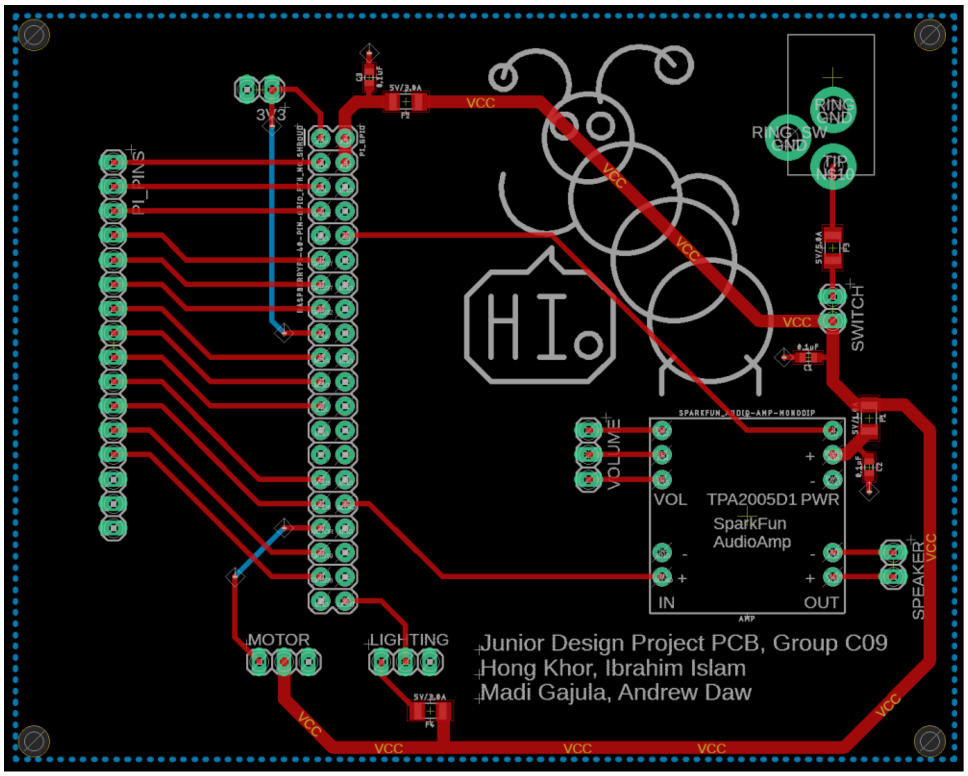


Figure 6. Eagle PCB schematic that includes header connection to be directly attached to the Pi 4 PCB.

The two-layer PCB design used includes traces to embed the Class D amp and Raspberry Pi. The PCB design also included traces to headers for the servo, lighting, and speaker. Additionally, the routing for the buttons was included on the PCB so the power switch, control buttons, and potentiometer could neatly be attached to the PCB and send signals to the Pi.

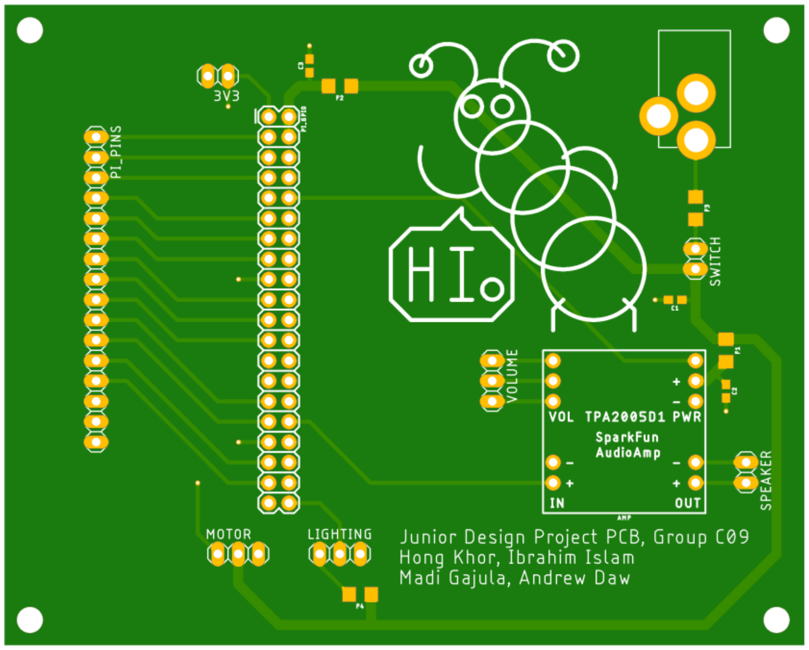


Figure 7. Eagle PCB model. This is an Eagle generated model of what the PCB will look like once it is fabricated

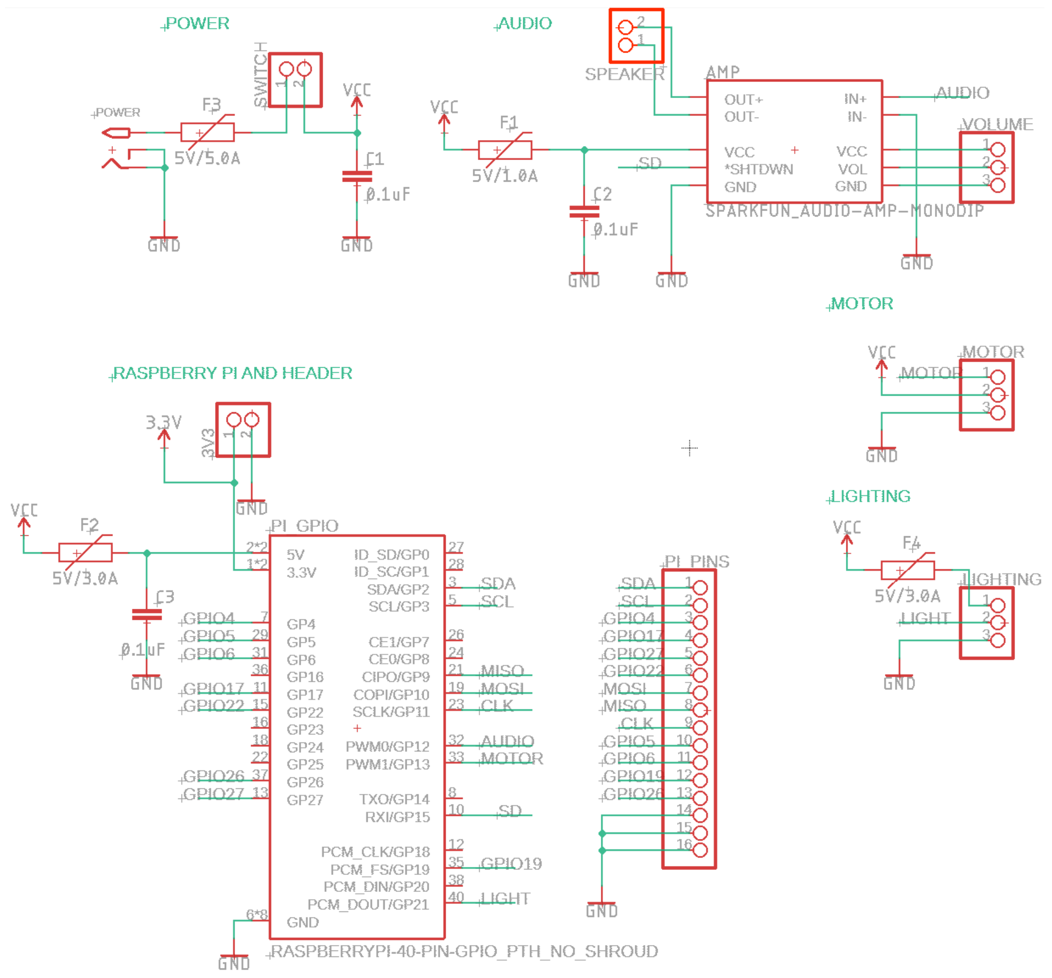


Figure 8. Eagle PCB Schematic based are initial software architecture and hardware

This PCB design includes several headers and integrations (Pi 4 header) that helped to avoid clutter in the overall design of the robot. This also made components easily removeable for testing purposes.

Mechanical Design

Our mechanical design featured a single servo driving the movement of a caterpillar that is set on top of it. The servo can be controlled with a PWM signal that rotates according to the duty cycle that is set.

The figure below shows the model used to laser cut the plywood box that contained the hardware for the robot.

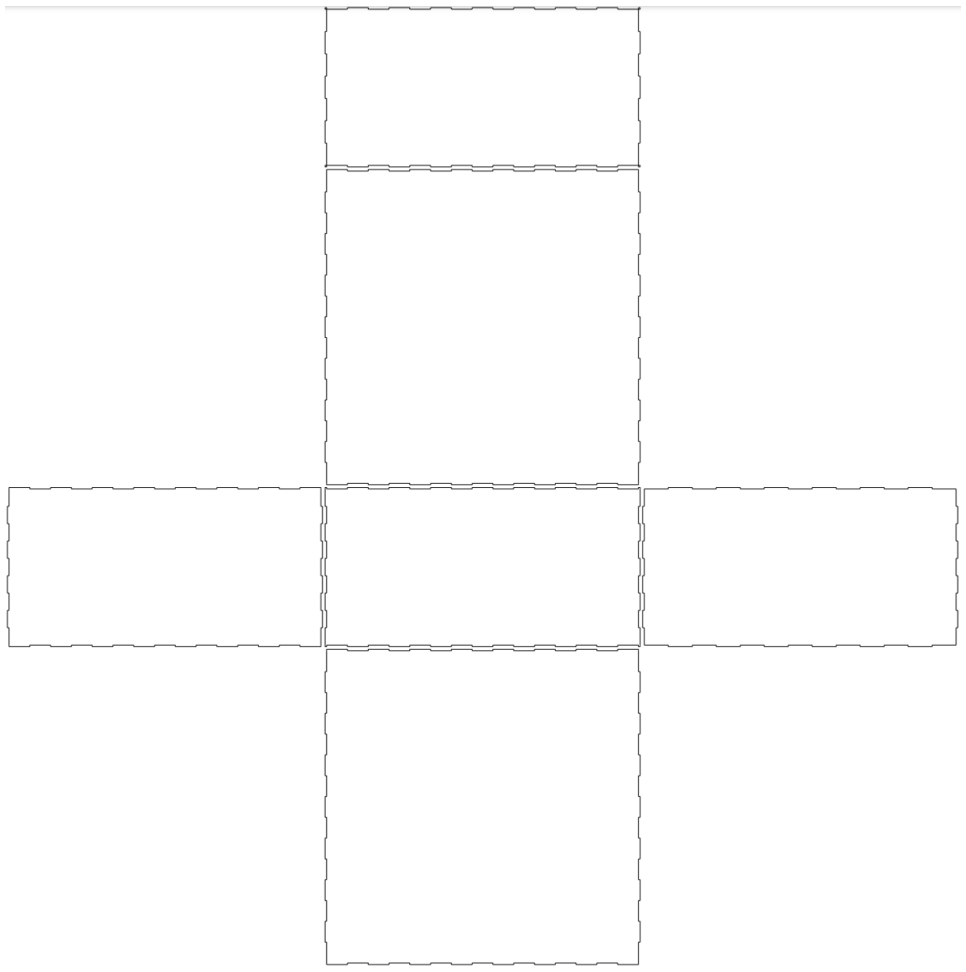


Figure 9. Box schematic from Adobe illustrator.

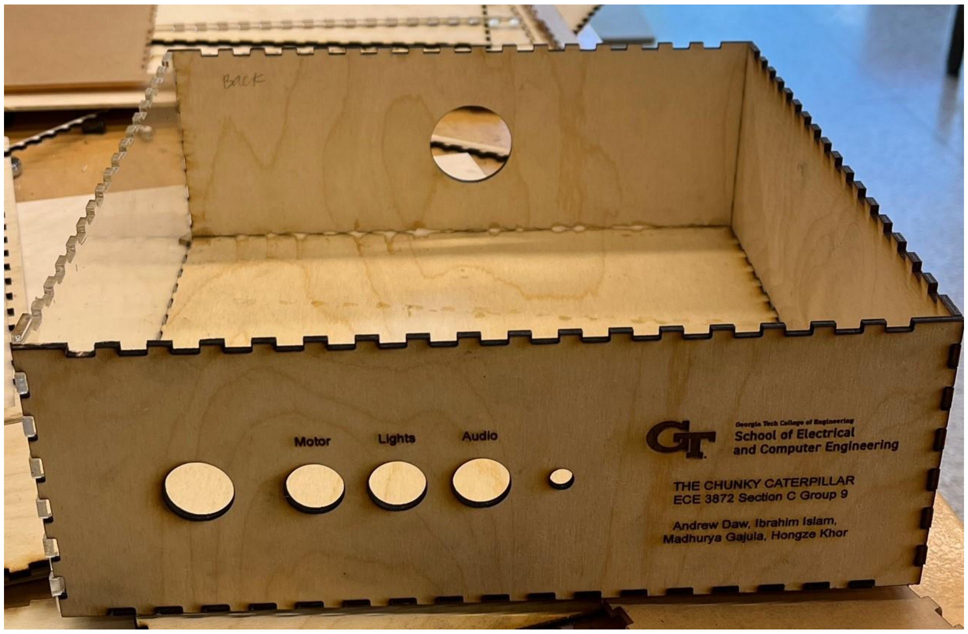

BACK
Motor
Lights
Audio
Georgia Tech College of Engineering
School of Electrical and Computer Engineering
THE CHUNKY CATERPILLAR
ECE 3872 Section C Group 9
Andrew Daw, Ibrahim Islam, Madhurya Gajula, Hongze Khor

Figure 10. Laser cut box with acrylic side.

The box has cut outs for the power cord, buttons, and potentiometer. The name of the robot and team details were also etched into the front of the design.

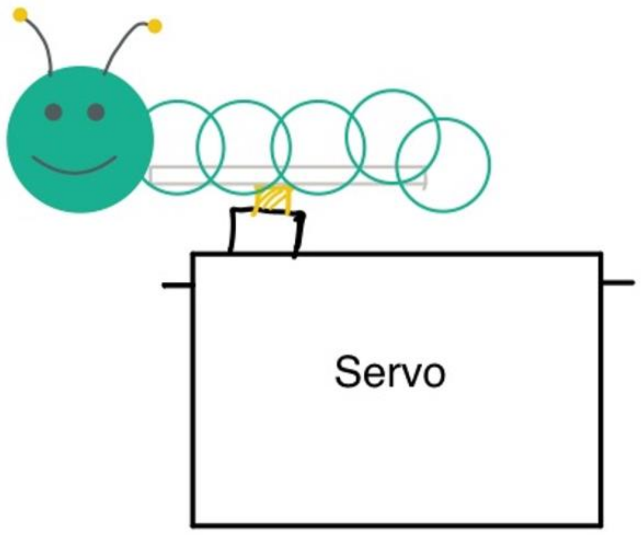


Figure 11. Servo mounting sketch.

The servo is concealed inside the body of the plush toy that is on top of the design. A wooden arm is attached to the drive gear of the servo to provide more robust movement.