Design Document

Dodo Done Right

Team #6: Dojun Lee, George Madathany, Aqil Kassam, Rayan Khoury

Date: November 13th, 2022

Table of Contents

Revision Record

Date	**Author**	**Comments**
September 25th, 2022	Team	Completed the preliminary design document as a team.
October 16th, 2022	Team	Updated the previous version to include the software subsystem, simulations, and more schematics
November 13th, 2022	Team	Revised mechanical, light, audio, and software design. Added more schematics.

Project Description

We are designing a system that involves a dodo bird that will communicate with a central director to play desired audio, while also engaging in at least one motion and activating the lights. The system will also have the ability to test its subsystems' functionality to ensure confidence in its operation.

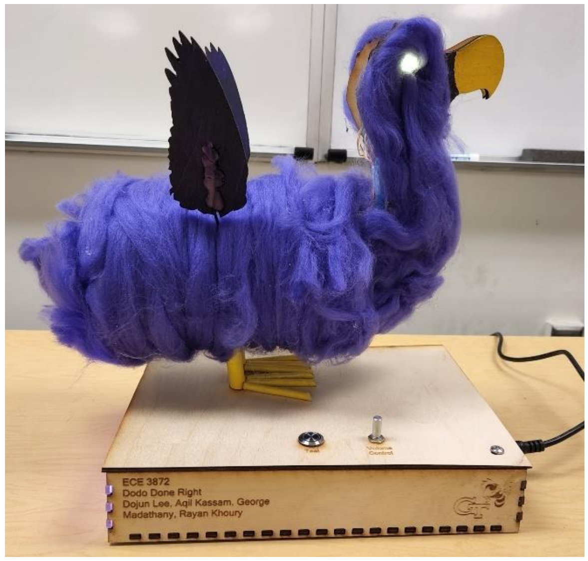

Figure 1: The "Dodo Done Right" system.

System Design

The design of the Dodo system involves five subsystems: mechanical, lights, audio, power, and processing. The system will take 120 VAC as an input, which will be converted into 5 VDC to be used by the subsystems. The processing subsystem will utilize an ESP32 microprocessor and will have user inputs ranging from buttons for the mechanical, audio, and light tests, as well as a power button that is part of the power subsystem. The mechanical system consists of the physical build of the system, as well as the servos used to move the dodo's wings.

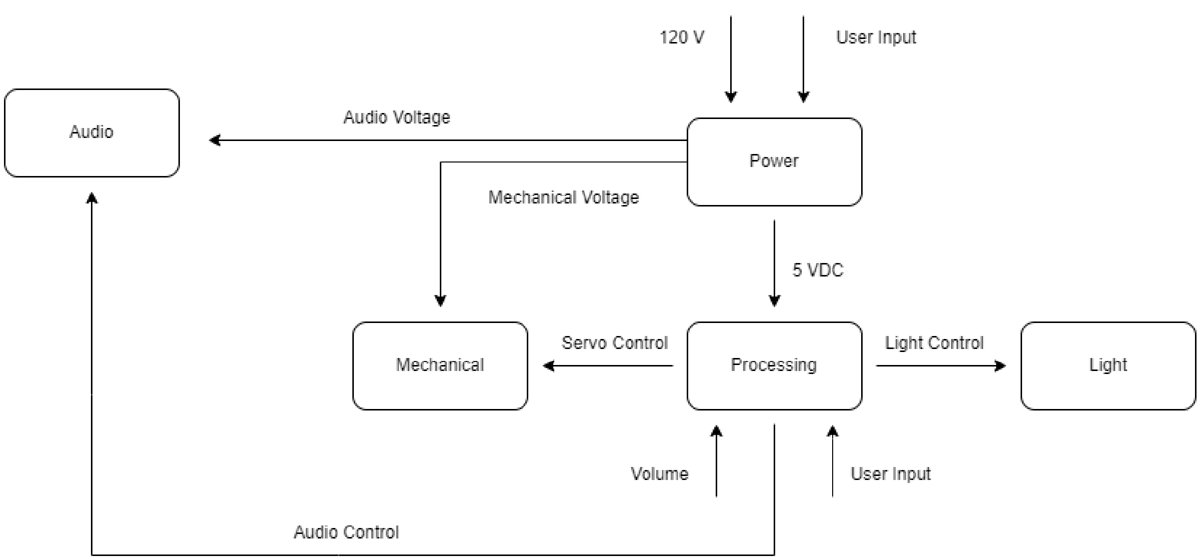

Figure 2: System Diagram

Interface	Source	Destination	Description
120 V Input	System Input	Power	120V AC
Mechanical Voltage	Power	Mechanical Subsystem	5V DC
Audio Voltage	Power	Audio Subsystem	5V DC for the amplifier
Processor Voltage	Power	Processing Subsystem	5V DC
Light Control	Processing Subsystem	Light	3.3V DC signal
Motor Control	Processing Subsystem	Mechanical Subsystem	PWM signal to control the motors
Audio Control	Processing Subsystem	Audio	Analog signal between 0V and 3.3V to output the voice lines
User Inputs	System Input	Processing Subsystem	3.3V DC voltage controlled by buttons
User Input	System Input	Power	5V DC voltage controlled by a button
Volume	System Input	Processing Subsystem	Analog signal between 0V and 3.3V to control noise decibel level

Table 1: Subsystem inputs and outputs

Software Design and Decomposition

The processing subsystem takes in 5V power, WiFi data, and digital inputs from buttons. One button is used to run a sequence that tests all the subsystems. 3.3V signals are supplied from the processor to the buttons. Digital output signals are sent to a motor controller and the LED subsystems respectively. A pulse-width modulator line is also reserved for the speaker output.

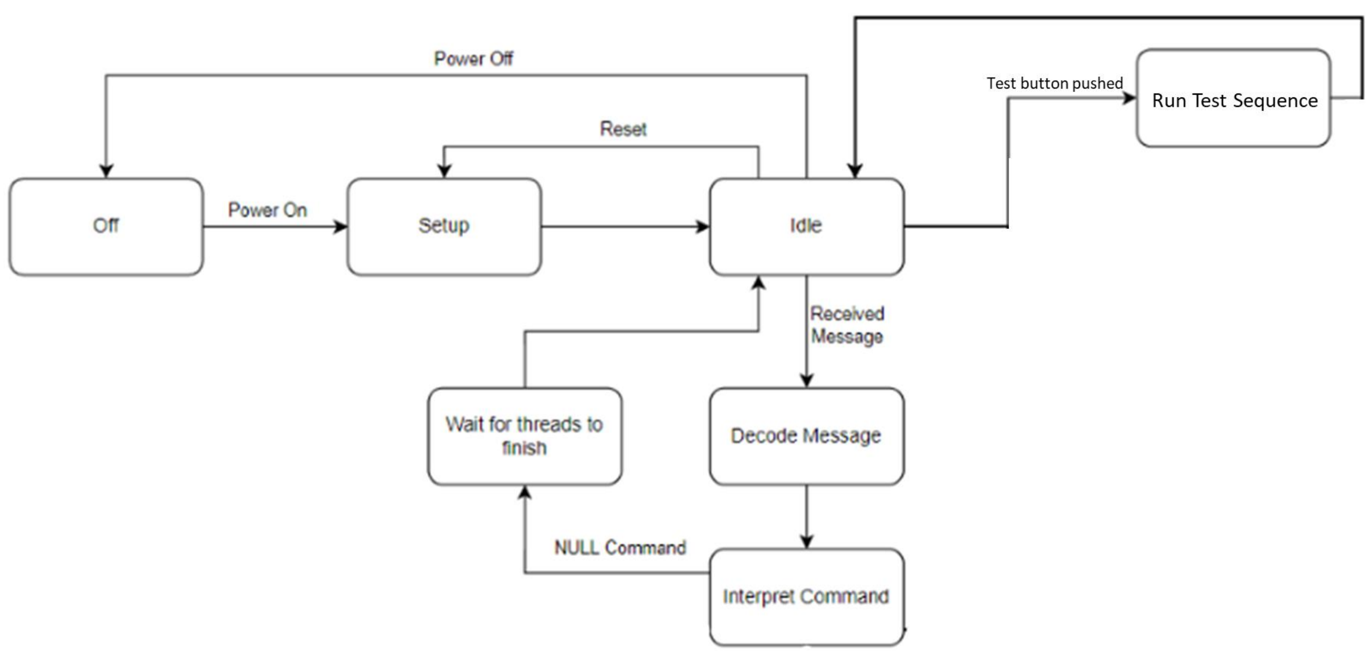

Figure 3: Software State Diagram

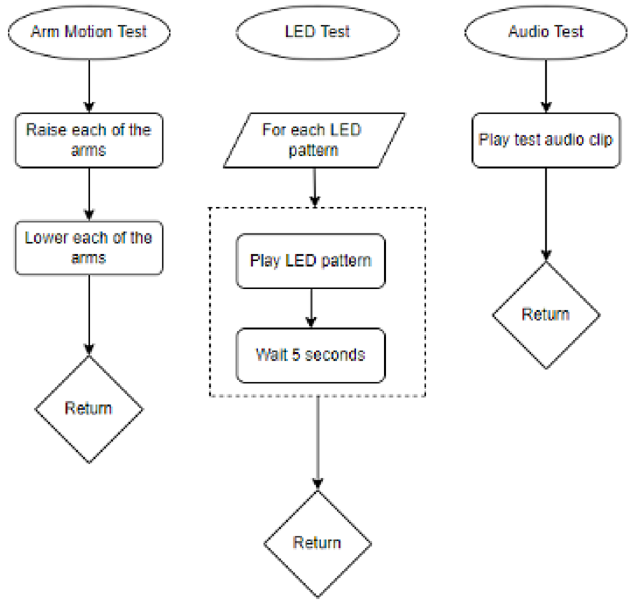

Figure 4: Software Test Functions

Subsystem Designs

Our design incorporates five subsystems: user interface, processor, motion, power, and audio. The audio will process the raw tone signals and convert it to sound, the motion will execute the movement of the figure, and the power will convert 120V AC to 5V DC. The processor will integrate all the subsystems by converting the user input to signals for the audio and movement systems. The LEDs and audio will activate once a file is received via Wi-Fi. Volume can be controlled by a potentiometer volume knob. The power subsystem will power the audio, mechanical and processing subsystems via conversion to 5 VDC.

Audio Subsystem

The audio system will take 5 Volts DC and receive an analog signal from the processor, then amplify it in through the 2N3904 amplifier and finally play it through the speaker.

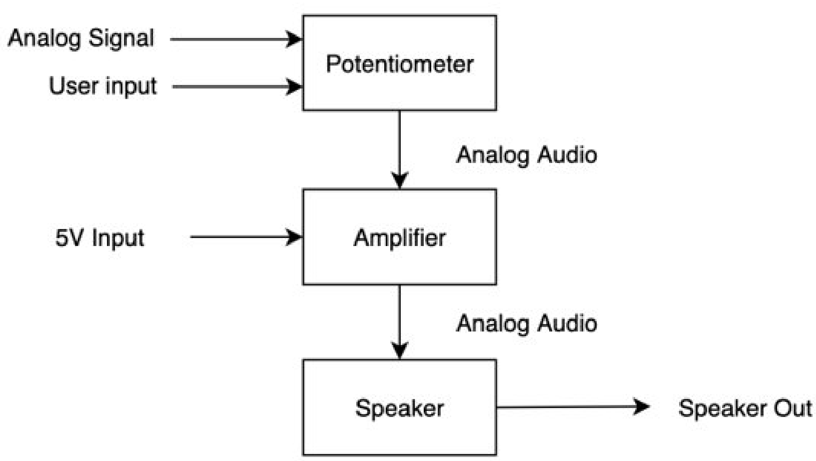

Figure 5: Audio Subsystem

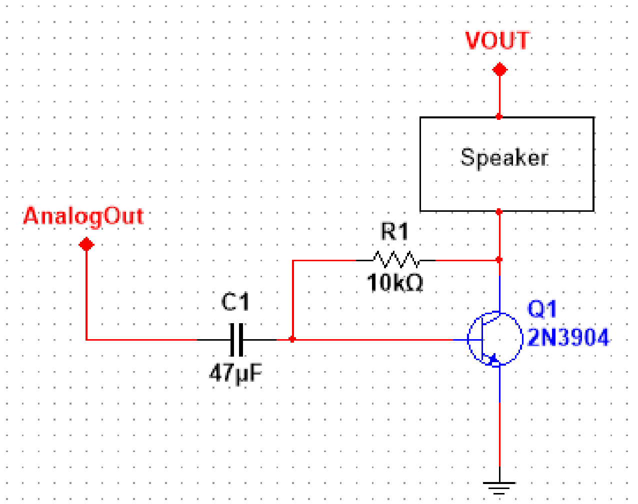

Figure 6: Audio Subsystem Schematic

As seen above, a schematic of the audio subsystem was created utilizing NI Multisim. The audio subsystem features a 2N3904 transistor, which is a common amplifier. The amplifier is also coupled with a capacitor in order to minimize oscillations. The speaker is then connected to the collector and VOUT of the microprocessor. Then, the circuit was implemented with an mbed microprocessor, as it has many of the same capabilities of the ESP32. A sample program was loaded onto the mbed, then tones of varying frequencies were played to validate the circuit design.

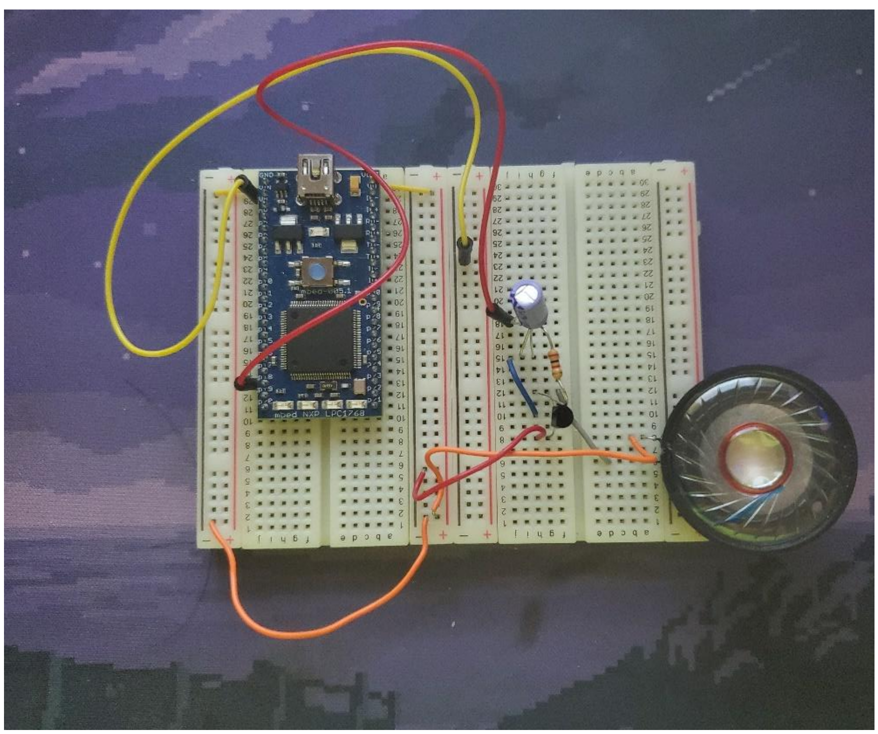

Figure 7: Audio Subsystem Simulation

N.B: As we were working on the project, we found out that the ESP32 audio library has been deprecated and that we were unable to use it for our bird.

Mechanical Subsystem

The mechanical subsystem will have two servos which will receive PWM inputs from the microprocessor, which are attached to each of the dodo's wings. The servos will be powered by the 5V input from the power subsystem and will receive an input once an audio file is received from the Director.

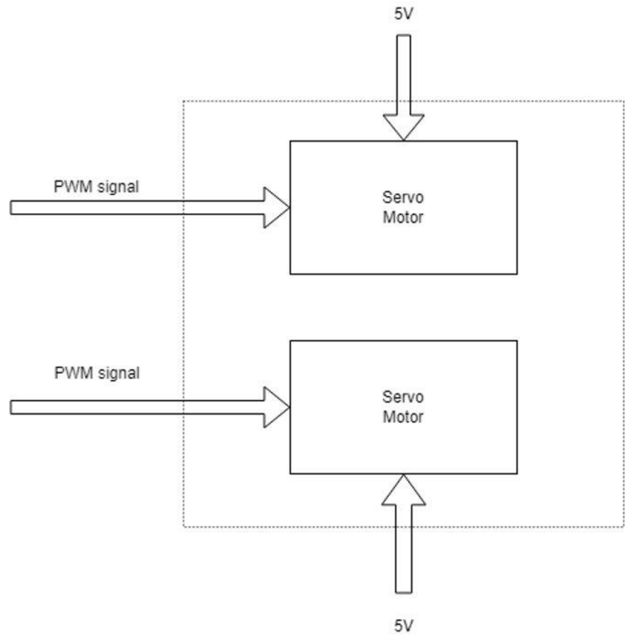

Figure 7: Mechanical Subsystem

The base of the Dodo was created by laser cutting wooden and acrylic panels to form a box for the body of the Dodo to rest on top of. First, a general SVG file was generated from Boxes.py, which is an open-source box generator. After creating the general structure of the box, the SVG file was then opened in Adobe Illustrator to add finer features. The following was the result:

Figure 8: Mechanical Subsystem

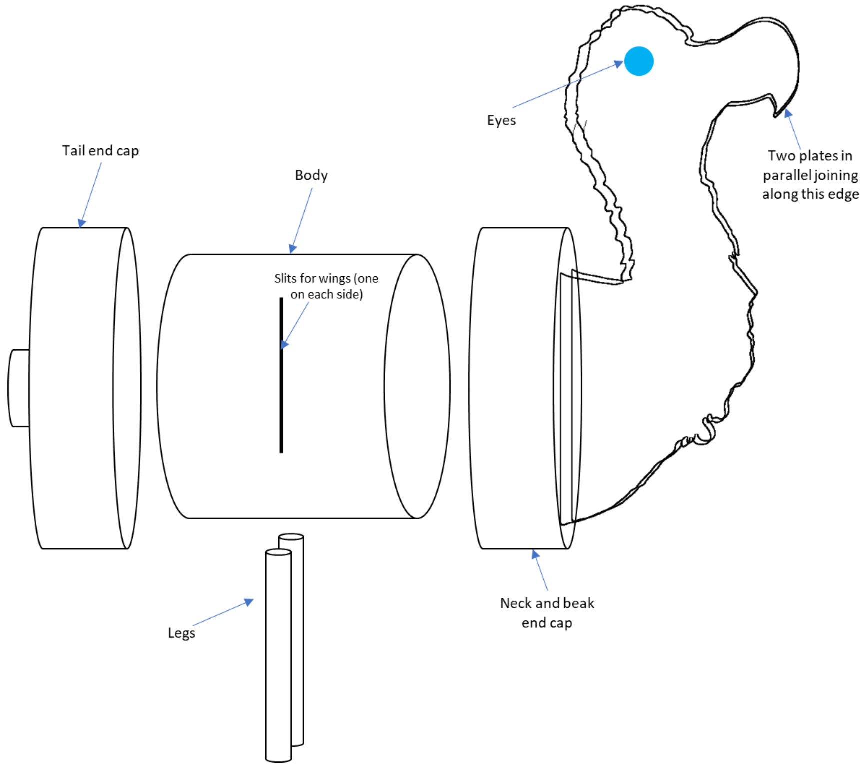

Figure 9: Mechanical Assembly

The bird itself is made up of multiple parts as shown in Figure 9: the body, the legs, the tail end cap, the neck and beak end cap, and the eyes. The tail and neck/beak end caps fit onto the body perfectly, the legs fit through precisely drilled holes on the bottom of the body and then through the precisely drilled holes on the top of the box, and the eyes (LEDs) fit through precisely drilled holes in the neck/beak piece.

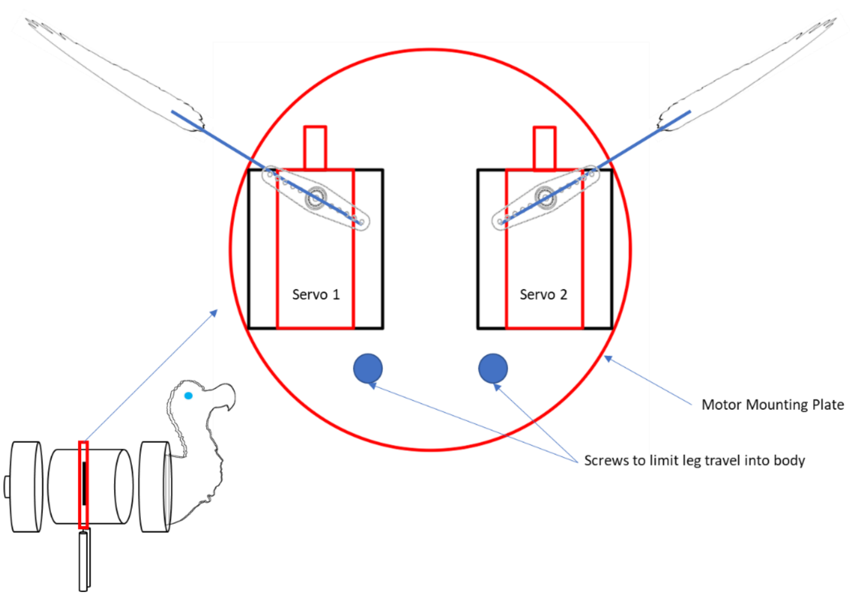

Figure 10: Design (left) and implementation (right) of motor mount plate with wings attached.

To mount the motors, we designed a motor plate to fit inside the PVC pipe. This motor plate friction fits into the area of the body where the legs enter and has spots for the motors to be screwed in and screws to limit the travel of the legs within the body. The wings are attached to metal rods which are attached to a servo horns. To assemble this, the servo horns are passed through the slits in the side of the body where they can simply clip onto the servo motor. Figure 10 is a cross section of the part of the body that has the motor plate and wings.

Light Subsystem

The light subsystem is composed of two sources of light – the lights in the eyes of the bird and the LED strip for the cutout of the box.

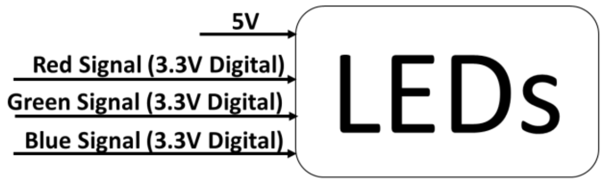

Figure 11: Light Subsystem

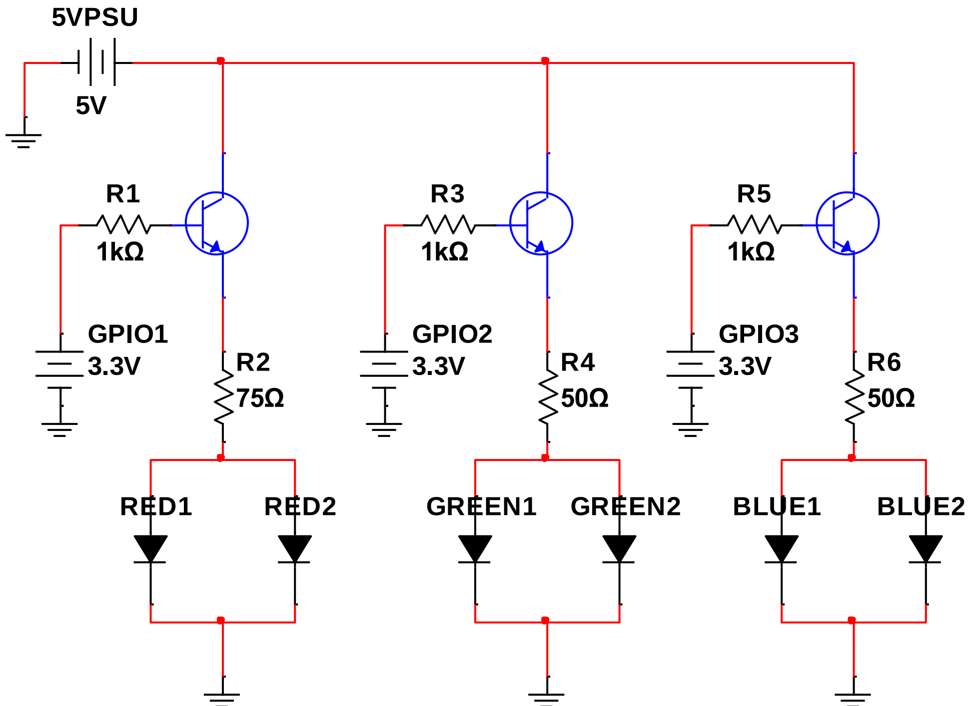

Figure 12: Eye LEDs circuit schematic

The lights in both eyes of the bird are RGB and combined, each color requires more current than the ESP32 can supply from 2 pins. For this reason, we designed a circuit based around NPN BJTs to power the LEDs in parallel using our 5V power source. Figure 11 is a schematic of this circuit – 5V is supplied to the transistor and the base of the transistor is modulated by a signal from our ESP32 to operate the transistor in either saturation or cutoff mode.

The LED strip for the cutout of the box was a much simpler venture. 5V from the power supply is connected to the red, green, and blue pins of the strip and the inline controller of the strip gives us some interesting patterns. This LED strip serves a dual purpose – it acts as a power indicator, turning on anytime power is connected to the whole system, and it improves the user's visibility of our beautiful circuitry inside the box.

Power Subsystem

The power subsystem will take 120 Volts AC as an input and output 5 Volts DC. It will route power through a 25 transformer, then through a user input switch and a fuse before distributing it to the other systems.

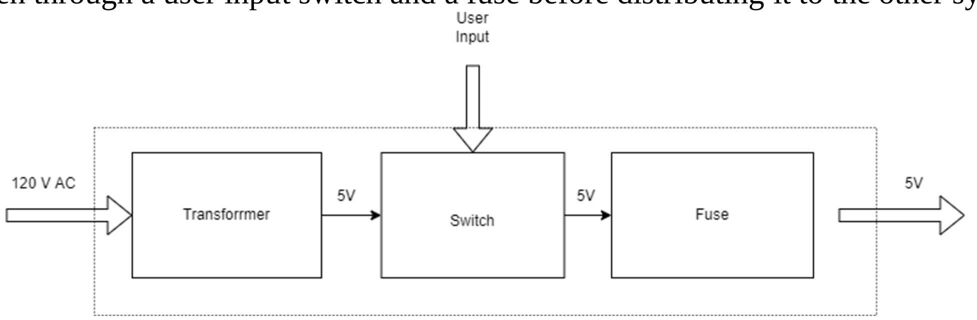

Figure 13: Power Subsystem

Processor Subsystem

The processor subsystem takes input from the user in forms of both digital and analog inputs, and outputs signals for the motor, audio, and the LEDs. The motor, audio, and cutout lighting are externally powered so those outputs are treated as merely logic signals, but the LEDs in the eyes will be powered by this signal.

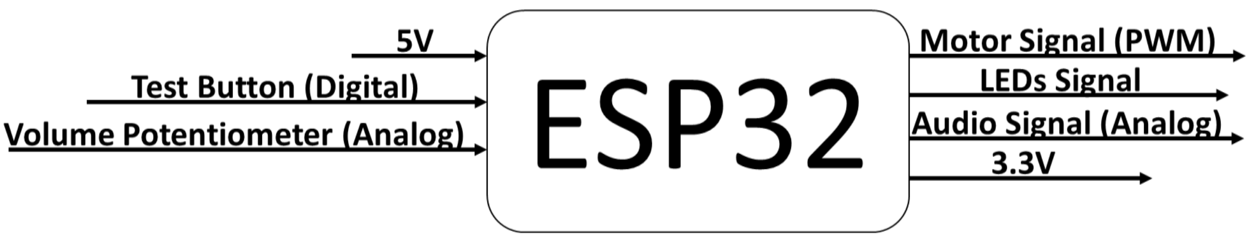

Figure 14: Processor Subsystem

Electronic Design

Our electronic design is centered around the ESP-32 processor, a central processing printed board, and a User Interface printed circuit board all made in Autodesk EAGLE. During assembly, we realized the LED strip only needed a 5V and ground connection, so we used a spare 5V pin and "white wired" ground. We also consolidated the two test buttons to one button, thus some of the connections on the boards were unused.

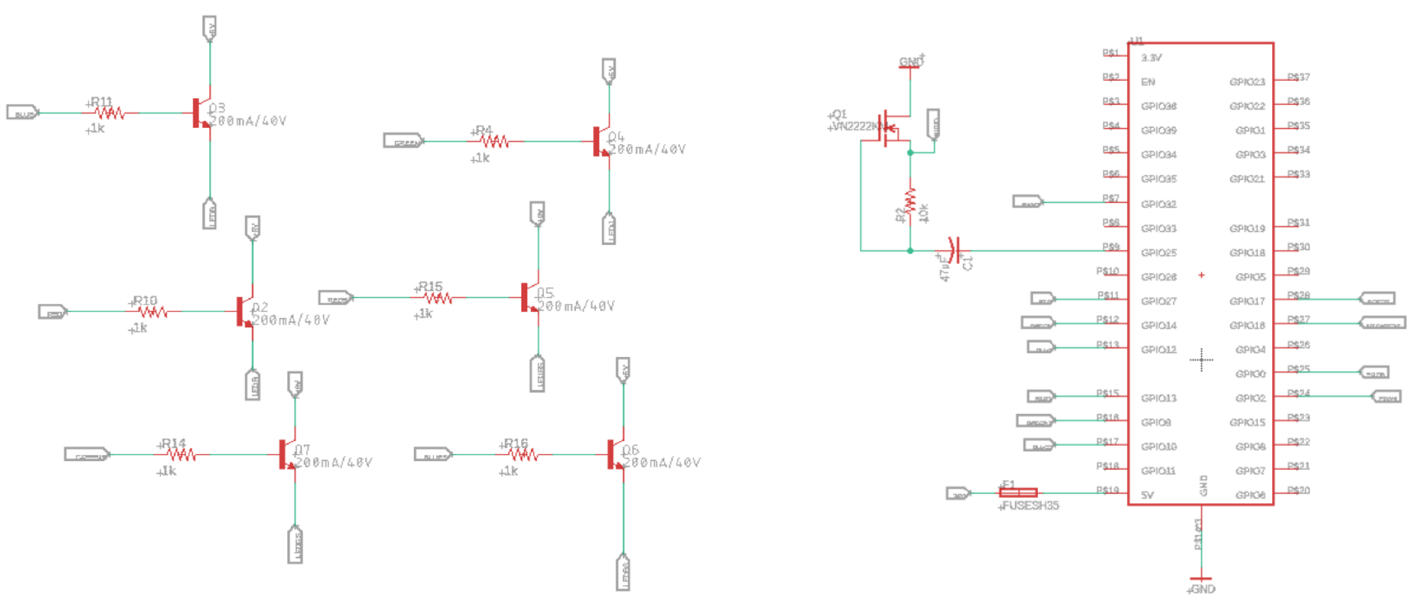

Figure 15a: Central Processing PCB Schematic

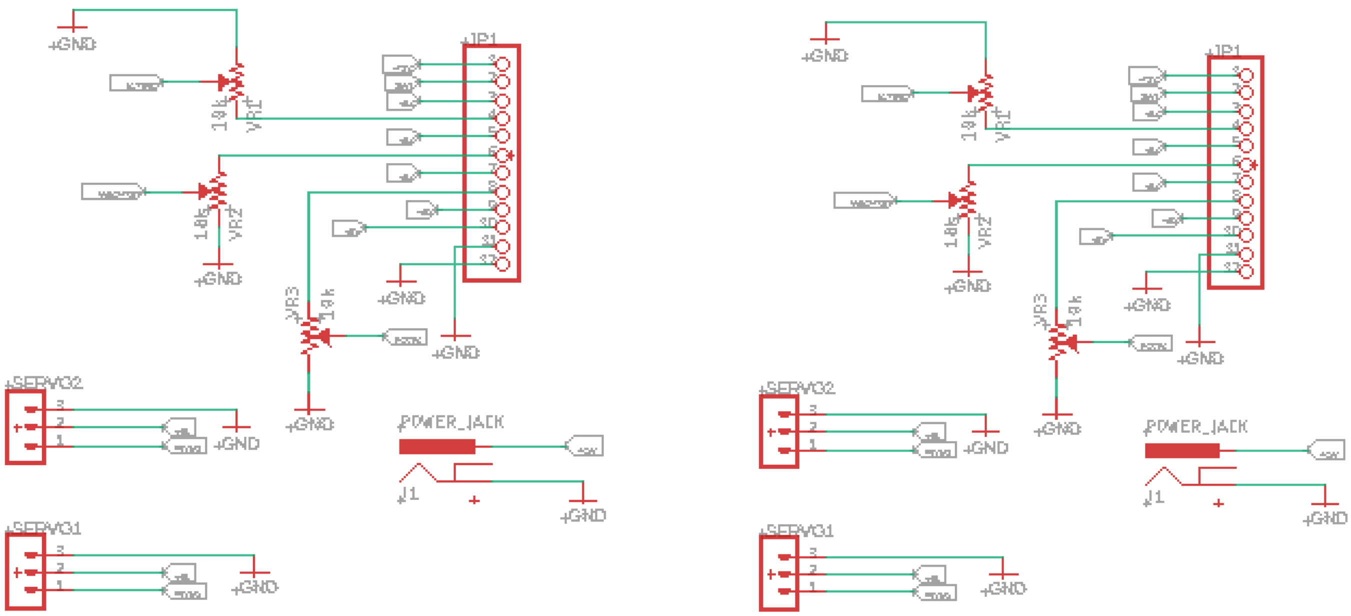

Figure 15b: Central Processing PCB Schematic (Cont)

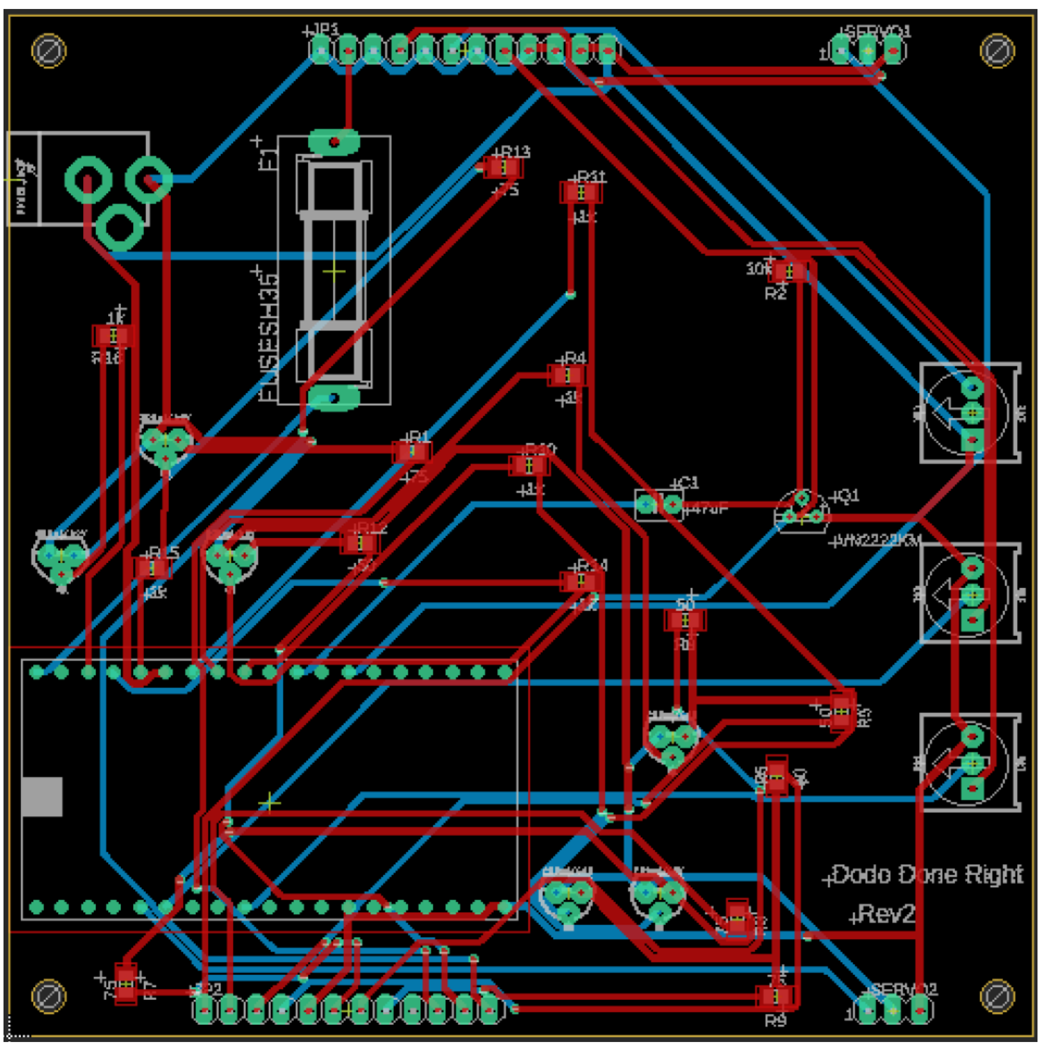

Figure 16: Central Processing PCB Board

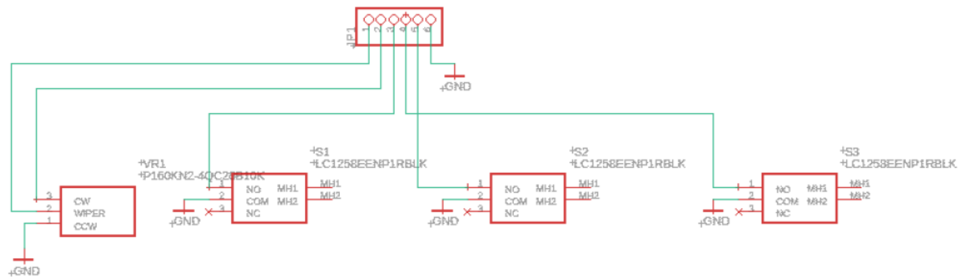

Figure 17: User Interface Schematic

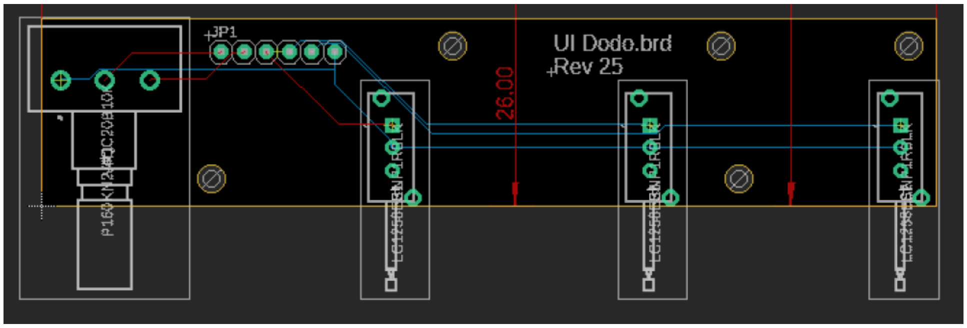

Figure 18: User Interface Board