Automatic Drum Tuner

By

Joey Bacino

Jonathan Fejkiel

Max Wojtowicz

Final Report for ECE 445, Senior Design, Spring 2025

TA: Shenyang Liu

7 May 2025

Group No. 70

Abstract

To simplify the process of tuning a drum, our group has designed a device that can tune a drum for the musician. In order to play drums correctly and for a solid sound to be made, all lugs on the drum must produce the same pitch. Our automatic drum tuner hits the drum to hear the sound, processes how out-of-tune the drum is based on microphone readings, and turns the lug of the drum. This process repeats until the lug is tuned. The user can control the tuner through a simple website that showcases a wide selection of desired frequencies.The device is also powered with AA batteries for ease of accessibility. Through a lightweight and portable design, we hope to help beginning and aspiring musicians start playing drums.

Contents

1. Introduction

1.1 Problem

Playing instruments is a pastime enjoyed by millions of people across the world. A task that almost every musician must endure before playing is tuning their respective instrument. For many, this is done easily if they are of able body and have good pitch. However, turning lugs and listening for the right tune can be difficult if someone is weaker, such as a child or the elderly, or if they are inexperienced in discerning pitch. Drum tuning requires precise hand movements and finger strength, which can be a challenge for individuals with arthritis or limited dexterity. Many aspiring musicians are limited because instruments are not made with accessibility in mind. About 66% of Americans have played or know how to play an instrument. This number shows how important music is to the world, and physical ability should not be a limiting factor in self expression. With our senior design project, we aim to solve a limiting problem in the accessibility of playing the drums for anyone who may need it. We want to make playing the drums an inclusive form of art for those of all different skill levels and ages.

1.2 Solution

The solution we propose is an automatic tuner for drums that will adjust the drum's lugs until the desired pitch is reached. The device will strike the drum with a push pull solenoid and determine its pitch at the current lug. A microphone will capture the pitch of the drum after it is struck with our push-pull solenoid. This strike will be transmitted to the microcontroller, which will calculate the frequency of the drum. Our red LED will be lit if the drum is not tuned. Whether it determines if the drum is too low or high based on a preset pitch, the microcontroller will determine the direction a tuning motor will turn. A servo motor attached to the lug will rotate to tighten or loosen the drumhead. These steps will repeat until the drum has reached its desired pitch. Once the correct pitch is detected from the drum, the microcontroller will tell all subsystems to stop, and the green LED will turn on. At this point, the user can remove the device and place it on the next lug. The device will be connected together as one unit. The device will clip onto the lip of the drum and the servo motor will fit into the lug of the drum. The push pull solenoid will be adjustable to raise or lower so that it strikes the drum at the correct height without damaging. The microphone will be as close to the head as possible to ensure it only picks up tones from the drum and not any background noise.

1.3 Visual Aid

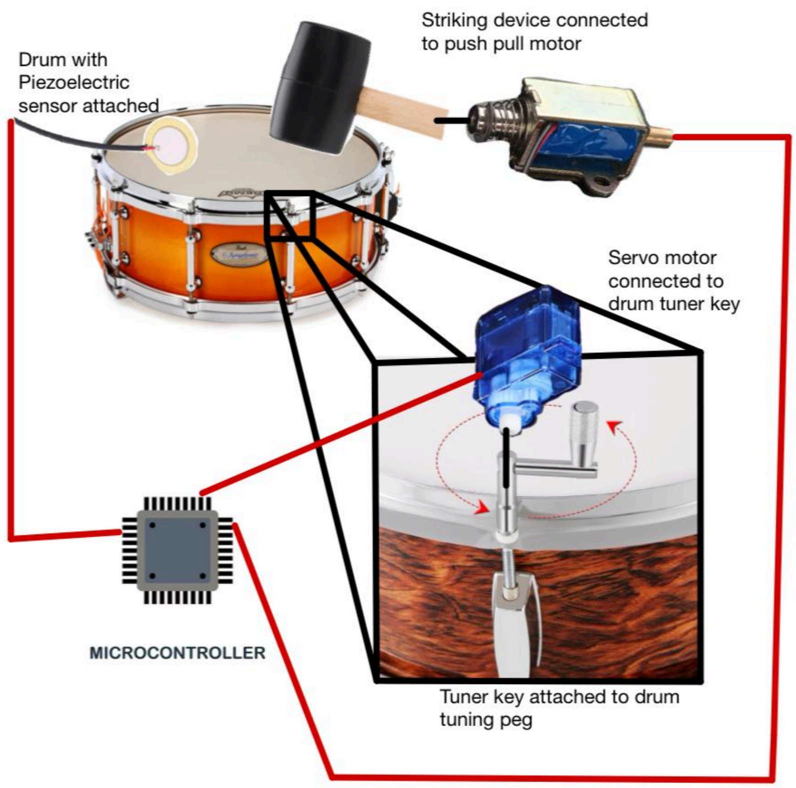

Figure 1: Visual Aid

1.4 High Level Requirements

1. The first requirement for our project is that the mechanism strikes the drum correctly and for the same amount of time and intensity for each cycle as needed. It must be strong enough to create frequency that can be picked up from the microphone and repeated if more tuning is needed. The motor rod should make contact with the drumhead for less than 10 milliseconds to maximize resonance for the drum.
2. The tuning mechanism should be able to apply the correct amount of torque needed to turn the lug to reach the desired pitch/frequency. We want to avoid any cases of the motor under or over shooting to keep the entire process streamlined and efficient, without unnecessary corrective turns. The motor should be able to produce at least 8-10 kg-cm of torque to be able to tighten the lugs enough to reach higher frequencies.

3. Finally, the pitch detection subsystem should accurately find the pitch of the drum after the solenoid has struck. It should only pick up sound from after the hit of the drum and not any softer background audio. To consider the lug to be tuned, we have a tolerance of 5Hz for each pitch. Based on the frequency for notes in the range of the second octave, 5Hz is an adequate tolerance to create a unified sound among all the lugs of the drum.

2. Design

2.1 Physical Design

Figure 2: Final Physical Design

The final physical design of our device went through many changes throughout the semester. We initially wanted our motor underneath our device to provide more support for the housing apart from just the clips attaching to the rim, but due to some difficulties we ended up mounting it on the side of the housing. We also wanted to 3D print our own clips to attach to the housing, but we couldn't find filament flexible enough to be able to snap around the rim without cracking, so we purchased drum-rim clips that were already on the market.

2.2 Block Diagram

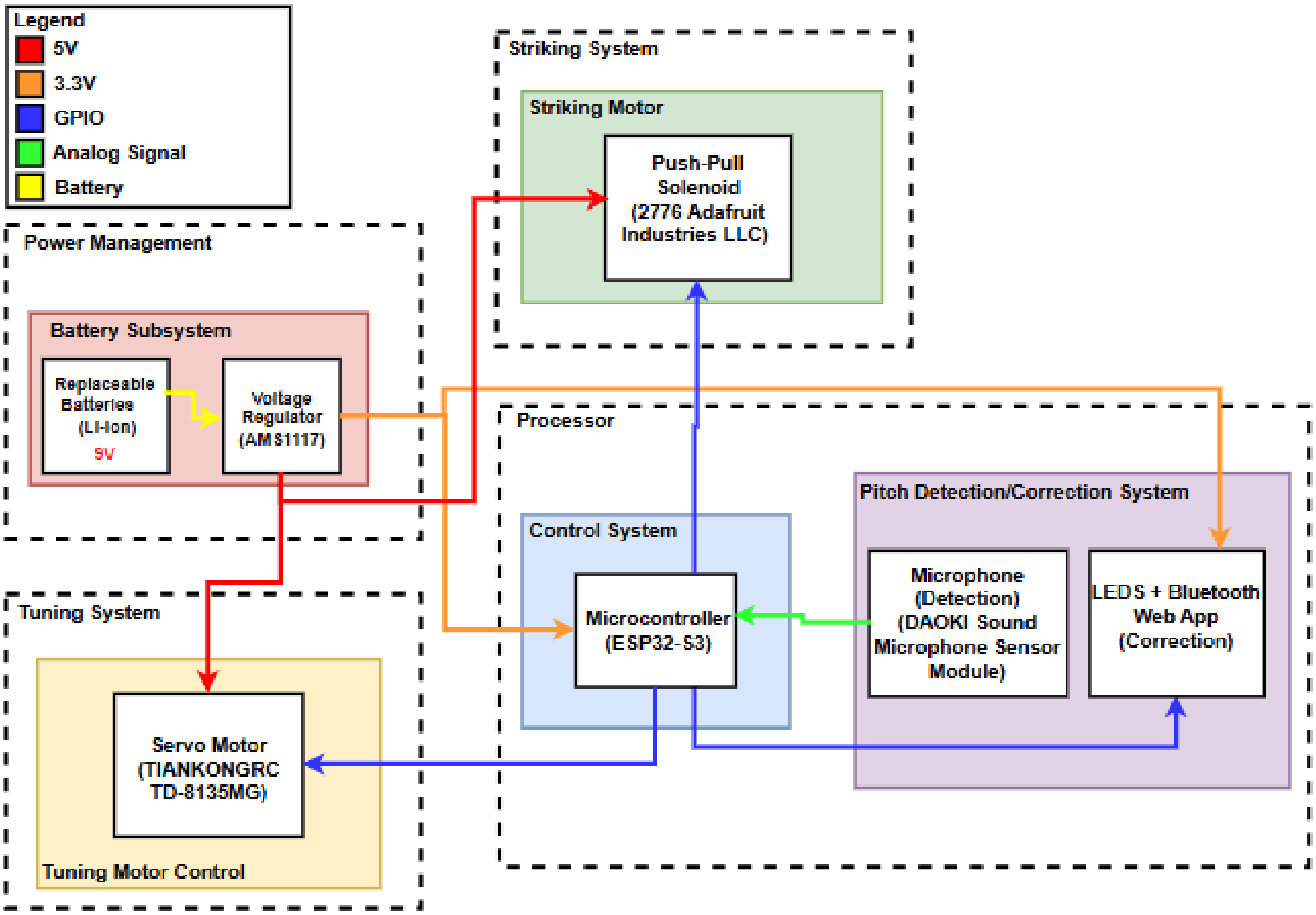

Figure 3: Block Diagram

2.3 Functional Overview and Block Diagram Requirements

2.3.1 Power Subsystem

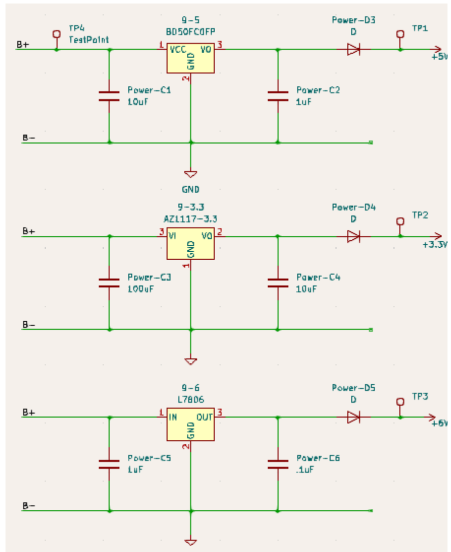

Figure 4: Power Subsystem Schematic

The power subsystem is responsible for stepping down to and maintaining the various voltage levels that the other subsystems and devices require to run. The power subsystem must step down from 9V and continuously supply 6V ± 0.1V to the tuning motor system, supply 5V ± 0.1V to the striking motor, and also provide 3.3V ± 0.1V to the microcontroller and pitch correction LEDs. A LM7805 voltage regulator will step down 9V to 5V for powering the push-pull solenoid, with 100µF and 10µF capacitors included for ripple reduction. The AMS1117 voltage regulator steps down 9V to 3.3V to supply power to the ESP32 microcontroller, using 10µF and 1µF capacitors for ripple filtering. Additionally, a LM7806 voltage regulator steps down 9V to 6V for the servo motor, incorporating 1µF and 0.1µF capacitors for ripple reduction. This setup ensures stable voltage levels within the required tolerances.

2.3.2 Striking Subsystem

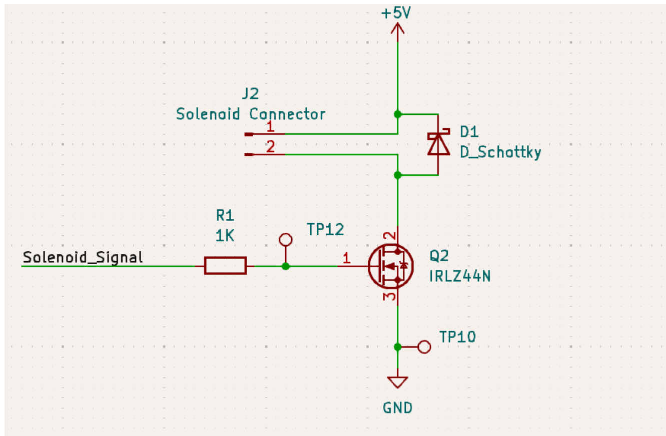

Figure 5: Striking Subsystem Schematic

The striking subsystem is responsible for striking the drum head near each lug to create sound that can be detected by our microphone. The striking subsystem must reliably receive a signal from the microcontroller to initiate a strike, ensuring precise timing for accurate pitch detection. It must also operate consistently when instructed, as failure to strike at the correct moment would lead to incorrect pitch readings from the detection subsystem. The system uses a 2776 Adafruit push-pull solenoid motor to execute the striking action. Additionally, the striker must not remain in contact with the drum for too long, as prolonged contact would alter the drum's pitch and cause faster sound dissipation. To ensure proper operation and longevity of the solenoid, the circuit must be designed to provide a quick voltage on-and-off signal without damaging the solenoid, incorporating necessary components such as flyback diodes and MOSFETs to prevent voltage spikes.

2.3.3 Pitch Detection Subsystem

The pitch detection subsystem is responsible for translating the time-domain signals from the microphone into frequency-domain information. This will be done using the FFT algorithm through the arduinoFFT library. Once we have the FFT, we can get the pitch from the last strike. We will compare the most dominant pitch from the FFT to our desired pitch, and this comparison will dictate if our tuning subsystem will either tighten or loosen the current lug the device is on. The ESP32-S3 must accurately identify the most prominent frequency from the DAOKI microphone sensor module. The microphone will be placed as close to the striking subsystem as possible, ensuring pitch readings only come from the target lug. The microcontroller must correctly determine whether the detected pitch is higher or lower than the desired pitch and subsequently send the appropriate signals to the tuning system. Additionally, it must control LED indicators to provide real-time feedback to the user, red for too low or too high, green for within tolerance, ensuring clear progress tracking during the tuning process.

Our original design used a piezoelectric sensor in place of a microphone. Our goal was to see if a sensor of this type is sensitive enough to detect exact frequencies from a struck drum.

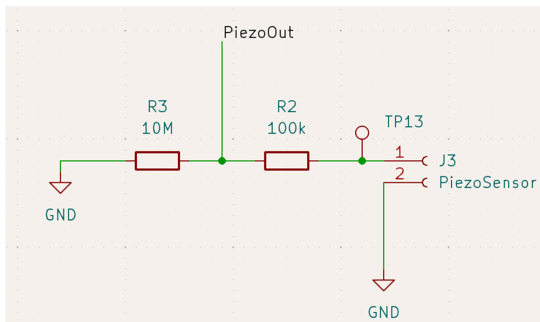

Figure 6: Striking Subsystem Schematic

After testing, we realized it was not possible to use the piezoelectric sensor. Above we have the original schematic for the piezo sensor. The tests showed spikes in voltage output when the drum was struck, yet the FFT library was unable to correctly determine a frequency. Different resistor values were used in the schematic above as well, yet there was no success.

2.3.4 Tuning Subsystem

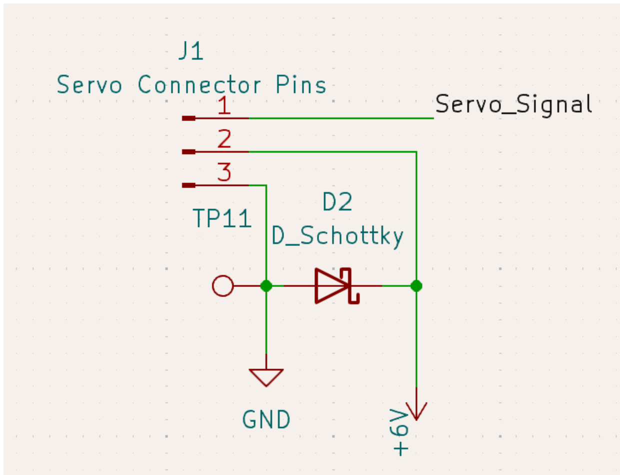

Figure 7: Striking Subsystem Schematic

The tuning subsystem is responsible for either tightening or loosening the lugs depending on how the current pitch compares to the desired pitch. The microcontroller must translate pitch deviation signals into commands that the servo motor controller can recognize, ensuring proper tuning adjustments. The motor controller will then direct the servo motor to rotate clockwise or counterclockwise to tighten or loosen the drum lugs accordingly. The servo motor will be controlled via PWM through a GPIO port on the ESP32-S3. After the drum is struck and is resonating, the tuning subsystem must continuously process pitch correction signals and adjust the lugs in real-time until the desired pitch is achieved. The system will use a TIANKONGRC servo motor to execute the tuning rotations effectively.

3. Design Verification

3.1 Pitch detection subsystem

Our pitch detection system is a vital part of our device. Without accurate pitch detection, our device won't be able to tell whether to tighten or loosen the lug to get to the desired pitch. Verifying this subsystem had two parts. Firstly, we needed to ensure that our microphone with its FFT backing was working properly and detecting the right pitch. We verified this by playing pure tones into the microphone, looking for a match between the frequency we played and the frequency our device read. Once we had this verification, we had to make sure we could accurately get the dominant frequency from the strike of a drum. We verified this part by comparing the output of our device after a drum strike to market available instrument pitch detectors, reaching an accuracy of over 80%. This accuracy was lower than we desired, but still enough to tune each lug in a reasonable amount of time.

3.2 Tuning subsystem

The tuning is done using a servo motor connected to the outside enclosure of our device. Here it is connected to the lug of the drum by a 3D printed adapter. Once the microcontroller detects a signal from the microphone and determines what the frequency is, it finally decides if we need to loosen or tighten the lug. Once it makes that determination, it sends a signal to the servo motor, telling it to turn clockwise for tightening, which makes the frequency increase, and counterclockwise for loosening, which makes the frequency decrease.

After initial testing to make sure we could turn the motor we purchased a full 360 degrees, we began testing different turning time intervals. Initially we began with having the motor turn based on how far it was from the wanted frequency, so if it was farther away, it would turn faster and for a longer time. But some complications arose with that, as it made the tuning not as precise, so ultimately we decided to make the motor turn in very short intervals of 1-2 seconds. After testing using these new intervals, we found that the tuning became very efficient and ultimately more accurate.

4. Costs

Description	Manufacturer	Quantity	Price	Link
Servo motor	TIANKONGRC	1	$27.99	https://www.amazon.com/dp/B08JCVLSCK?ref=ppx_yo2ov_dt_b_fed_asin_title
Microphone	DAOKI	5	$6.49	https://www.amazon.com/dp/B00XT0PH10?ref=ppx_yo2ov_dt_b_fed_asin_title
Push Pull Solenoid	AdaFruit	1	$4.95	https://www.adafruit.com/product/2776
6 AA Battery Pack Holder Enclosure	NEWZOLL	1	$6.59	https://www.amazon.com/dp/B01N2INSBR?ref=ppx_yo2ov_dt_b_fed_asin_title
MCU Dev Board	Espressif Systems	1	16.53	ECE Supply Center , HiLetgo ESP-WROOM-32 ESP32 ESP-32S Development Board
6V Voltage Regulator	STMicroelectronics	1	.96	https://www.digikey.com/en/products/detail/stmicroelectronics/L7806ABD2T-TR/585696
PLA Filament for Enclosure	Elegoo	1	14.99	https://www.amazon.com/ELEGOO-Filament-Dim

				ensional-Accuracy-Printers/dp/B0D421Q2Q2?source=ps-sl-shoppingads-lpcontext&ref_=fplfs&smid=A2WWHQ25ENKVJ1&gQT=1&th=1
		Parts Total	$78.50	

Table 1: Cost of Parts

All other components are available in the ECE self service shop and do not need to be paid for and ordered. This includes other voltage regulators, passive components, USB port, and the microcontroller itself.

Name	Rate	Hours	Total (($/hour) x 2.5 x hours to complete)
Joey Bacino	40$	80	$8,000
Jonathan Fejkiel	40$	80	$8,000
Max Wojtowicz	40$	80	$8,000
Total Labor Cost			$24,000
Total Project Cost (Parts + Labor)			

Total Project Cost (Parts + Labor)	$24,078.50

Table 2: Labor Cost Analysis With Total Project Cost

5. Conclusion

5.1 Accomplishments

The first accomplishment we achieved was the correct powering of the device. Through our power subsystem, we were able to create outputs of 6V, 5V, and 3.3V. The power supply that we used supplied 9V, however we also tested with a 12V AC-DC charger. This also maintained correct voltages across all subsystems. The pitch detection subsystem was a success as it accurately read the pitch of the current lug with an accuracy greater than 80%. This was enough for the lug to be tuned in reasonable time. The turning of the lug was a success as the device was powerful enough for tightening or loosening the lug across a variety of frequencies. This was improved with the component change of the servo motor.

Finally, the user interface and bluetooth connection from the companion website to the device was an additional success for our project. The site was immensely helpful in testing and also proved to be an accessible way for the user to interact with the tuner.

5.2 Uncertainties

The most important uncertainty we have in the operation of the device is the pitch detection. This is caused by the combination of the mechanical structure of the actuator, the device's attachment to the drum, and how the drum produces a frequency. The actuator is made of metal, so when it is given power and extends and retracts, it hits its own housing and produces a sound that interferes with the drum's sound. Additionally, the device was not fully secure to the drum. This caused variation in the way the solenoid hits the drum and the variation caused some slight difference in the output sound. Finally, the way a drum produces an even tone is quite complex. When some lugs are tightened or loosened much more than other lugs, the tone produced becomes very uneven. Normally, these lugs should all be tightened a little bit at a time. Our device turns one at a time, so when this lug is very tight or loose compared to the others, the pitch detection is not as accurate.

5.3 Ethical Considerations

Since our ‘Automatic Drum Tuner’ is a fairly simple device, it has minimal ethical and safety concerns. However, we will formally adhere to the IEEE Code of Ethics [1] and remain mindful of any potential risks throughout the development and testing process. We will also actively consider and implement feedback from our TA and professors to ensure that our device meets the highest safety and ethical standards. Every team member will be treated with respect, and open communication will be prioritized to address any concerns that may arise during development.

To further enhance safety, we will document all procedures necessary to handle the device properly. A lab safety document will be created to outline best practices for battery handling, electrical safety, and general lab safety protocols. This will ensure that all team members are aware of potential hazards and know how to respond to emergencies.

Although our device does not involve high-voltage components, or direct human/animal interfaces, we will conduct thorough testing to ensure it does not pose unforeseen risks. Additionally, if any safety concerns arise during the development or testing phases, we will reassess our design and implement mitigation strategies to protect both users and developers. By maintaining a proactive approach to safety and ethics, we can ensure that our project remains both responsible and reliable.

5.4 Future Work

Since our final design’s clamp design wasn’t strong enough on its own, we would want to greatly improve it by implementing a few different possible designs. One design would end up attaching across multiple points around the rim of the drum, this would allow a very secure and stable connection to the drum. All of these connection points would be connected together to a circle that would be sitting above the rim of the drum. Then we would add tuning components along the rim of this device at each tuning lug on the drum. Finally we could attach a striking device to the middle of the device, where it would be right above the middle of the drum. This would allow us to strike the drum, and having a microphone and motor at each lug, we could adjust each lug at the same time, as if for example one lug is too low or too high compared to the rest, it would cause the overall pitch detection to be thrown off.

Another key aspect we could work on in future design is better microphone filtering. With a bit more adjustment on the microphone, we would be able to possibly eliminate all unwanted noise right from the start so that way we would never get any unnecessary noise coming through. This could be done through filtering implementation on the pcb itself, or through some code implementation.

References

[1] *IEEE - IEEE Code of Ethics*, www.ieee.org/about/corporate/governance/p7-8.html. Accessed 13 Feb. 2025.

[2] GPI International Ltd. (2017). *Safety Data Sheet for Alkaline 9V battery (Model: 1604A)*. https://data.kleintools.com/sites/all/product_assets/documents/msds/klein/SDS-GP-battery.pdf

Appendix A: Requirement and Verification Table

System Requirements and Verifications		
Requirement	**Verification**	**Verification status**

		(Y or N)
1. The power subsystem must step down from 9V and provide 5V ± 0.1V for the striking motor.	Measure the 5V output with a multimeter under load to confirm it remains within ±0.1V tolerance.	Y
2. The AMS1117 regulator must step down 9V to 3.3V ± 0.1V for the ESP32 and LEDs.	Measure the 3.3V output with a multimeter and confirm voltage stability within ±0.1V under normal operation.	Y
3. The LM7806 regulator must step down 9V to 6V ± 0.1V for the servo motor.	Measure the 6V output with a multimeter to ensure it remains within ±0.1V under typical load conditions.	Y
4. The striking subsystem must receive a signal from the microcontroller to initiate a strike.	Use an oscilloscope or logic analyzer to verify that the control signal from the microcontroller is correctly sent when a strike is required.	Y
5. The motor must always operate when instructed by the microcontroller.	Send multiple strike signals and observe the solenoid's response to ensure consistent operation. We'll use an oscilloscope to measure the current draw to confirm activation.	Y
6. The microcontroller must correctly identify the most prominent frequency from the microphone sensor readings.	We will use a pitch detecting app to compare the microcontrollers output to the expected values.	Y
7. The microcontroller must translate higher or lower pitch signals into commands recognizable	We will monitor the output signals from the ESP32-S3 using an oscilloscope or logic analyzer to	Y

by the servo motor controller.	ensure correct PWM signal generation based on pitch translation from the FFT.	
8. The motor controller must direct the servo to rotate clockwise to tighten and counterclockwise to loosen based on the microcontroller's signals.	We will send test signals to the motor controller and verify that the servo rotates in the correct direction for both tightening and loosening actions.	**Y**

Table 3: Requirements and Verifications