Maestro Mittens

Team # 32
Team Members:
Shayna Kapadia
Tulika Gupta
Kushal Majmundar
TA:
Chi Zheng

Table of Contents

1. Introduction

1.1. Problem and Solution Overview

Learning to play an instrument is hard. It takes time, effort, and a lot of money to get any good. What is even more difficult is learning to play an instrument well. Not all people have a knack for picking out harmonic melodies, thus making learning new songs and freestyling even more difficult.

Our product, “Maestro Mittens” aims to bridge this knowledge gap by providing an easy way for people to “play” an instrument. In response to your movements, these mittens produce notes in the same key so you are never out of tune; no need to learn scales or music theory! The mittens will have different modes for different types of instruments.

For example, in “piano mode”, moving your hands to the right/left will produce higher or lower pitches. In “guitar mode”, bending your fingers in different combinations would produce different chords that are also in the same key. The strumming hand would determine the tempo of the sounds produced.

With Maestro Mittens, consumers can play various instruments easily even with limited musical abilities and skills. More so, our product eliminates the struggle of having to carry around heavy instruments wherever you want to play. All you need are your mittens and a computer!

1.2. Visual Aid

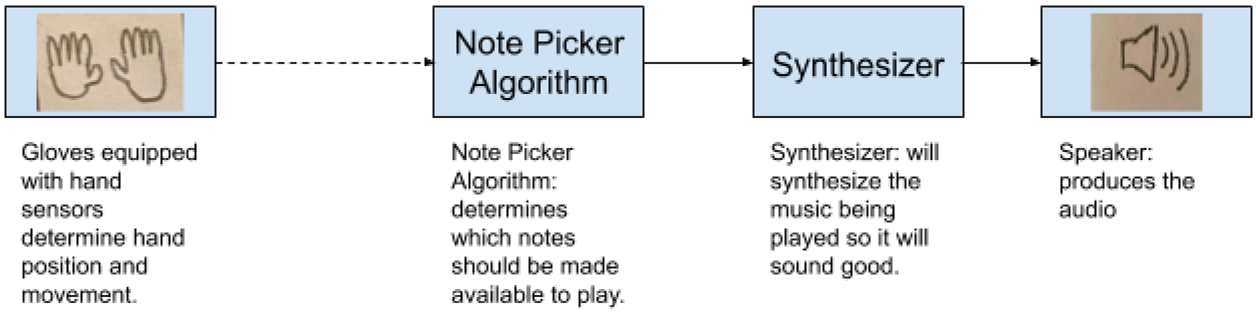

Figure 1: System Overview

1.3. High Level Requirements

1. The flex sensors and accelerator/gyroscope should be calibrated enough to detect finger, wrist and arm movement when they occur with enough precision that a note could be replicated by repeating the same movement. The flex sensor should be able to sense a 45° bend.
2. The signals from the sensors should be sent to our computer at rates of around 115.2kps so that the audible delay between the hand movement and sound is negligible to the human ear.
3. The computer should appropriately interpret the signals to produce the proper notes such that there are no dissonant notes played or abrupt unintended jumps in pitch.

2. Design

2.1. Block Diagram

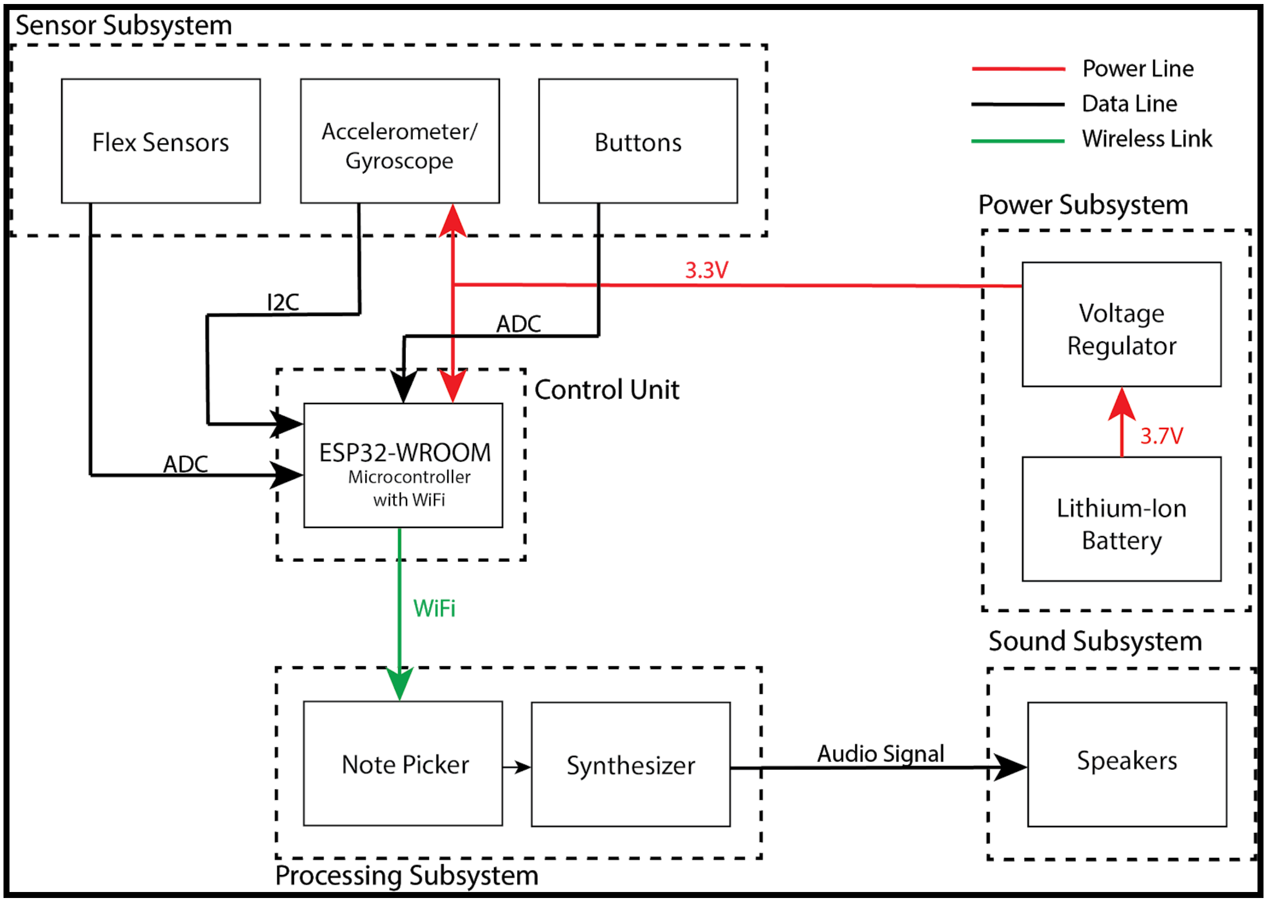

Figure 2: Block Diagram

2.2. Physical Design

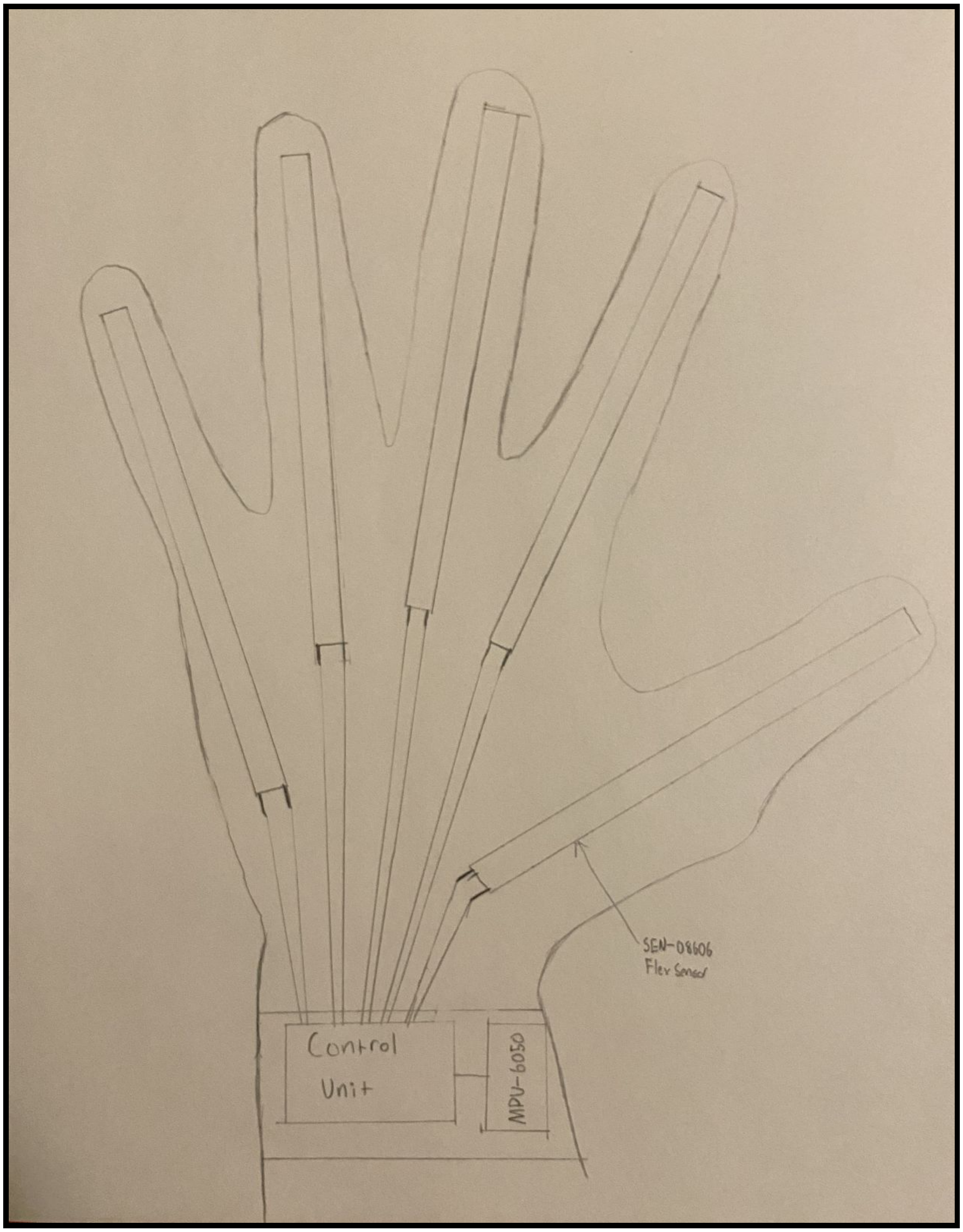

Figure 3: Physical design of gloves

2.3. Sensor Subsystem

The sensor subsystem is responsible for detecting the user's hand motions and sending that information to the control unit. Specifically the sensor subsystem needs to:

(1) sense when a finger has been pressed and
(2) sense the direction and speed of the user's hand movements and
(3) detect which instrument is set through the buttons.

The subsystem is composed of ten flex sensors, which are used to detect if a finger has been bent. An accelerometer/gyroscope on each hand to detect the user's hand movements. And it also has two push buttons located on the control unit. Each sensor in the Sensor Subsystem will send its data to the Control Unit for processing.

2.3.1. Flex Sensor

There will be ten 905-1070-ND flex sensors attached to each of the ten fingers of the gloves. These sensors will detect when a finger has been pressed down and send that information to the control unit. This is done via a flex sensor subcircuit (fig. 4), which will have the output voltage drop by ~1V when the flex sensor is bent ~45°.

Requirement	Verification
When bent to about 45° the flex sensors subcircuit will output a voltage approximately 1V less than when the flex sensors were unbent.	1. Build the flex sensor subcircuit (fig. 4) and bend the flex sensor to about 45° 2. Attach a voltmeter across the output and see if the voltage drops a volt

Table 1: Flex Sensor R&V

2.3.2. Accelerometer/Gyroscope

There will be one accelerometer/gyroscopes- attached to each glove that will be used to sense the user's hand movements. They will detect the speed and direction of the user's hand movements and relay the information to the control unit.

Requirement	Verification
The accelerometer/gyroscope should be able to detect when the user moves their hand to the right or left.	1. Display the accelerometer readings on the computer 2. Place the accelerometer on hand and move hand back and forth about 2 ft 3. Record and analyse the accelerometer's reading for positive and negative x-axis motion

Table 2: Accelerometer/Gyroscope R&V

2.3.3. Buttons

There will be two push buttons located on the control unit that will be used to set the gloves in different instrument modes. One button will be for piano-mode and the other will be for guitar-mode.

Requirement	Verification
When a button is pressed the control unit should receive the signal.	1. Connect out push buttons to our control unit and push the button 2. Verify the control unit has received the signal

Table 3: Button R&V

2.4. Control Unit

The Control Unit Subsystem is tasked with
(1) receiving all the data from the Sensor Subsystem,
(2) filtering out the noise from the flex and accelerometer sensors, and
(3) sending this data to the Processing Subsystem.

2.4.1. Microcontroller

Requirement	Verification
1. Can send and receive data over Wi-Fi above 115kbps to prevent audible delay	1. a. Connect ESP-32 to power and upload arduino program to send a 30 byte packet to computer b. Receive packets on computer end c. Confirm that the data rate is above 115kbps
2. Can receive data over I2C at a minimum rate of 400kps: the max data rate I2C on the accelerometer/gyroscope	2. a. Connect microcontroller to a arduino via I2C b. Program arduino to send microcontroller a 0.5Mb packet of data, and time how long it takes. c. Program microcontroller to echo back data d. Check RTT and data integrity upon receiving data back to arduino.
3. Can filter out noise from accelerometer/gyroscope and flex sensor readings	3. a. Set up tested flex sensor and gyroscope circuits and connect to control unit b. Output from microcontroller to two LEDS. One for accelerometer/gyroscope readings and one for flex sensor readings c. Bend fingers less than 45 degrees and check to see if LED lit d. Slightly shake hand side to side and check if LED is lit

Table 4: Microcontroller R&V

2.5. Power Subsystem

The power subsystem is responsible for providing and regulating 3.3V of power for the Microcontroller (ESP32)and Accelerometer/Gyroscope. The power subsystem is comprised of a 3.7V lithium-ion battery attached to a voltage regulator circuit. A 3.7V lithium-ion battery is not necessarily a consistent power source, as its voltage can vary from 2.5V-4.5V[8]. Our the accelerometer/gyroscope and ESP32 require a voltage between 3.0V-3.46V therefore a voltage regulator circuit is necessary to safely provide our subsystems with a consistent voltage of ~3.3V. The total current draw for all the components needing power falls into the range 500.5mA - 503.9 mA . These calculations are based on the power consumption of the ESP-32 (500 mA), and Accelerometer/Gyroscope (500 µA - 3.9 mA) datasheets. The minimum is a summation of their idle current draw, and the maximum is a summation of the active current draw.

Requirement	Verification
Voltage supply should be within 3.1-3.7 volts	1. Connect the voltage regulator circuit to a power supply at 2.5V. 2. Use a voltmeter to monitor the voltage. 3. Ensure the output voltage remains within 3.1-3.7V 4. Repeat 1-3 with power supply at 4.2 V
Current draw should remain between 45.2mA-54.5mA	1. Connect the battery to all components (Accelerometer/Gyroscope, ESP-32) and connect ammeter to battery output. 2. Plot current draw over time to

	ensure it is within the required values.

Table 6: Power Supply R&V

2.6. Processing Subsystem

The Processing Subsystem is in charge of taking in the sensor data from the Control Unit and outputting an audio signal to the Speaker Subsystem. Based off of the values given by the Control Unit, the note picker will pick out a note (or multiple notes) and send these values to the synthesizer. The synthesizer will then produce an audio signal based on the values given and send it to the Speaker Subsystem.

2.6.1. Note Picker Algorithm

- Upon start up, note picker picks a key and filters out notes that are not in that key
- Note picker takes in values from the control unit
- Compare values from before to see if pressed fingers have changed, unless the saved value is null, then it will calculate the notes regardless
- If pressed fingers have changed, calculate right hand notes
    - If the speed of the right hand is positive, shift right hand window by specified amount
    - Else if speed of the right hand is negative, shift right hand window by specified amount
    - For each finger pressed
        - Pick one note from window and add it to the notes_to_play array
- If pressed fingers have changed, calculate right hand notes
    - If the speed of the left hand is positive, shift right hand window by specified amount
    - Else if speed of the left hand is negative, shift right hand window by specified amount
    - For each finger pressed
        - Pick one note from window and add it to the notes_to_play array
- Send notes to play to the synthesizer

Algorithm 1: Note Picking

2.7. Schematics

The flex sensor subcircuit is designed to provide an output voltage drop of about 1V when the flex sensor has been bent to about 45. In other words, the voltage difference when the flex sensor is flat and when it is bent to 45° is about 1V. The output voltage can be determined by:

$$Vo = 3.3 \bullet \left(\frac{R2}{R2 + R1} \right)$$

Where: R1 is the flex sensor resistance and R2 is the other resistor seen in figure 4.

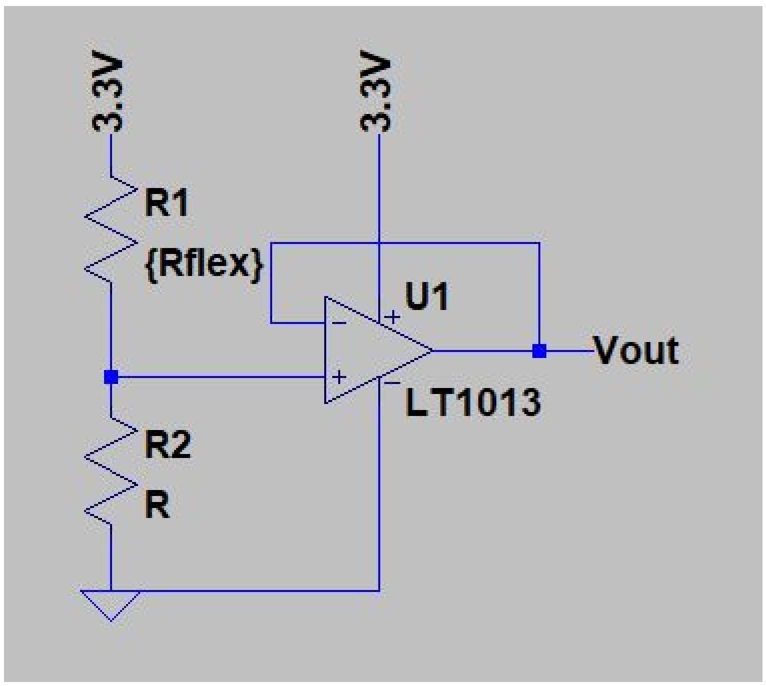

Figure 4: Flex Sensor Subcircuit Schematic
(note: we will be using an LM358 op amp)

2.8. Tolerance Analysis

An important tolerance that must be maintained would be the tolerance of the sensors in the sensor subsystem. We want the sensors to be sensitive enough to detect and relay all intentional user movement. However, we do not want our sensors to relay information on any minor hand movements. It is not practical to keep one's hand perfectly stiff, as such, the user's hands will slightly move about when they are intending to keep them still. The unintentional movements will still be picked up by our sensors, therefore there needs to be certain tolerances set so we can ignore these minor movements. Our goal is to have the sensors tolerances set so they can register the user's intentional hand movements, while ignoring any unintentional hand movements.

The sensor subsystem should only register a "key press" if the user's fingers have been significantly (about 45°) and ignore any other minor finger movements. This can be accomplished by designing our flex sensor subcircuit (fig. 4) to have an output voltage drop of approximately 1V when a "key press" is detected. First we need to obtain the average flex sensor resistance when flat and when at a 45° bend. Next we can use the voltage divider formula to see how the flex sensor resistance will affect the output voltage. Where: Vi = 3.3V, R1 is the flex sensor resistance, and R2 is the second resistor.

$$Vo = Vi \bullet \left(\frac{R2}{R2 + R1} \right)$$

If we want our output voltage to experience a 1V drop when our flex sensor is bent we need to choose an R2 value to satisfy the equation:

$$Vo(flat) - Vo(45^{\circ}) \approx 1V$$

$$Vi \bullet \left(\frac{R2}{R2 + R1(flat)} \right) - Vi \bullet \left(\frac{R2}{R2 + R1(45)} \right) \approx 1V$$

$$\left(\frac{R2}{R2 + R1(flat)} \right) - \left(\frac{R2}{R2 + R1(45)} \right) \approx \frac{1}{3.3}$$

This will give us an R2 resistance value that will cause a voltage drop of 1V when the flex sensor is bent to 45°. Thus satisfying our tolerance of only registering a significant finger bend.

Similarly the accelerometer/gyroscope should be able to detect the user's hand movements, while ignoring any unintended, minor motions. First we place the accelerometer on our hand and perform minor hand shakes, as to simulate minor movements. Then we display the accelerometers readings on a computer and observe the range of accelerometer reading values that are to be ignored. This range of values will be the threshold the accelerometer's reading needs to surpass in order to be registered as intentional movement. Our gloves will also have a starting hand position, as the user will begin with their hands in front of them. This gives our accelerometers readings a point of reference as to where the user's hands start.

3. Cost Analysis

3.1. Labor

- Labor: (For each partner in the project)
  - A Computer Engineering Major from the University of Illinois at Urbana Champaign makes an average salary of $84,250 a year[7]
  - This is about $40.5 per hour, which is our chosen labor cost
  - Estimated work time is 5 hours/week for 15 weeks, which is 75 hours to complete.
  - With the given equation, cost per hour x 2.5 x hours to complete = TOTAL:
    - $40.5/hour x 2.5 x 75 hours = **$ 7593.7**
- Labor for all the partners in the project:
  - $ 7593.7/partner x 3 partners = $22,781.25
- So, our final estimated cost of labor is **$22,781.25**

3.2. Parts

Part	Manufacturer	Part #	Quantity	Total Cost ($)
Flex Sensor 4.5	DigiKey	905-1070-ND	10	$80.90
Accelerometer /Gyroscope	DigiKey	1428-1007-1-ND	2	$16.62
Printed Circuit Board (PCB)	PCBWay	N/A	2	$0
ESP32 WiFi Module	Espressif Systems	1904-1025-1-ND	2	$9

Low dropout linear voltage regulator	Infineon Technologies	726-TLS850B0 TEV33ATM	2	$3.14
				TOTAL: **$109.66**

Table 7: Cost Analysis

3.3. Total Costs

- Total Costs
  = Labor + Parts
  = $22,781.25 + $109.66
  = **$22,890.91**

4. Schedule

Number	Week of Date	Kushal	Shayna	Tulika
1.	2/24/20	Research and understand PCB design requirements and specifications	Research and understand PCB design requirements and specifications	Research and understand PCB design requirements and specifications
2.	3/02/20	Prepare PCB design for conversation with machine shop	Prepare PCB design for conversation with machine shop	Prepare PCB design for conversation with machine shop
3.	3/09/20	Finalize design and correct any errors for PCBWay Order	Finalize design and correct any errors for PCBWay Order	Finalize design and correct any errors for PCBWay Order
4.	3/16/20	Spring Break	Spring Break	Spring Break
5.	3/23/20	Test and integrate voltage regulator from power supply.	Initial test for confirmation of sensors' requirements and	Initial test for confirmation of sensors' requirements and

			verification.	verification.
6.	3/30/20	Program microcontroller to send acquired data to computer	Learn usage of SuperCollider for synthesizer program	Get sensor data and detect user movement and key presses to be sent to microcontroller
7.	4/06/20	Integrate the sensor subsystem, the control subsystem and the power subsystem	Complete programming synthesizer to receive and transition between notes	Complete Note Picker program to send notes to synthesizer
8.	4/13/20	Accept data into computer from ESP-32, parse accepted data into required format	Continue previous week's deliverables	Continue previous week's deliverables
9.	4/20/20	Refine and test, catch up if fallen behind schedule	Refine and test, catch up if fallen behind schedule	Refine and test, catch up if fallen behind schedule
10.	4/27/20	Final Demo, work on final presentation	Final Demo, work on final presentation	Final Demo, work on final presentation
11.	5/04/20	Final Presentation	Final Presentation	Final Presentation

Table 8: Schedule

5. Ethics and Safety

5.1. Ethics

We believe our project complies with all of IEEE code of ethics. Specifically, code number 5 states that developed technology should "improve the understanding by individuals and society of the capabilities and societal implications of conventional and emerging technologies, including intelligent systems". Our project aims to improve individuals' understanding of music through the use of this technology. Thus we believe that our project doesn't undermine any of the IEEE Code of Ethics.

5.2. Safety

There are a few safety concerns in using a lithium-ion battery. If the battery were to fail or overheat, this would result in "thermal runaway which is a reaction within the battery causing internal temperature and pressure to rise at a quicker rate then can be dissipated". Once a battery goes into thermal runway, it can cause enough heat to induce thermal runway in other batteries ultimately resulting in a fire. These fires are more difficult to put out and thus make this uniquely dangerous[6]. Other safety concerns could arise from open or uncovered wires that could potentially cause electric shock to the wearer of the device[5]. This is why our design will have all wires covered and away from the user's skin; thus adhering to the first rule of the IEEE code of ethics by ensuring the safety of the user[3].

6. Risk Analysis

A big risk for the project is the sensors. Sensor readings can be quite unreliable at times, and we will be using multiple sensors which will all affect a single note value.

Our system could be calibrated to cancel out noise that is not relevant to the program. That way minor motions aren't incorrectly interpreted as playing a note. We could observe the sensor values at rest, when there are no movements being made, as well as implement the project to require somewhat exaggerated gestures to guarantee a note being played right.

The sensors of course are integral to the user experience. The user would like to see every movement they perform being translated into sound. Too much noise and unexpected behavior could be detrimental to the overall experience. It is quite important that we avoid this issue.

7. Citations

[1] Sparkfun, 'FLEX SENSOR DATASHEET', [Online]. Available: https://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/FLEXSENSOR REVA1.pdf. [Accessed: 20- Feb- 2020]

[2] wpengine.netdna-cdn.com, 'MPU-6000-Datasheet1', [Online]. Available: http://43zrtwysvxb2gf29r5o0athu.wpengine.netdna-cdn.com/wp-content/uploads/2015/02/MPU-6000-Datasheet1.pdf. [Accessed: 20- Feb- 2020]

[3] ieee.org, 'IEEE Code of Ethics', 2020. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html.[Accessed: 20- Feb- 2020]

[4] "XBee Buying Guide," XBee Buying Guide - SparkFun Electronics. [Online]. Available: https://www.sparkfun.com/pages/xbee_guide. [Accessed: 24-Feb-2020].

[5] "Lithium Ion Battery Safety for Consumers," Lithium Ion Battery Safety for Consumers. NFPA, 2017. Available: https://www.nfpa.org/-/media/Files/Public-Education/Resources/Safety-tip-sheets/LithiumIonBatterySafety.ashx

[6] "The Hazards of Lithium Batteries", Viceprovost.tufts.edu.(2020). [Online]. Available: https://viceprovost.tufts.edu//ehs/files/The-Hazards-of-Lithium-Batteries.pdf [Accessed: 24-Feb-2020].

[7] E. I. T. S. Services, “Salary Averages,” Salary Averages :: ECE ILLINOIS. [Online]. Available: https://ece.illinois.edu/admissions/why-ece/salary-averages.asp. [Accessed: 27-Feb-2020].

[8] AnalogDevices, “It Just Got Easier to Convert Lithium-Ion Battery Voltage”, [Online]. Available: https://www.analog.com/en/technical-articles/it-just-got-easier-to-convert-lithium-ion-battery-voltage-to-33v.html. [Accessed: 27-Feb-2020]