Bluetooth Enabled Gloves for Controlling Music

ECE 445: Project Proposal

Team Members:

Saicharan Bandikallu (sb35)

Mehul Aggarwal (mehula3)

Oliver Johnson (owj2)

TA: Akshatkumar Sanatbhai Sanghvi

Professor: Viktor Gruev

Due: 2/9/2023

1 Introduction

1.1 Problem

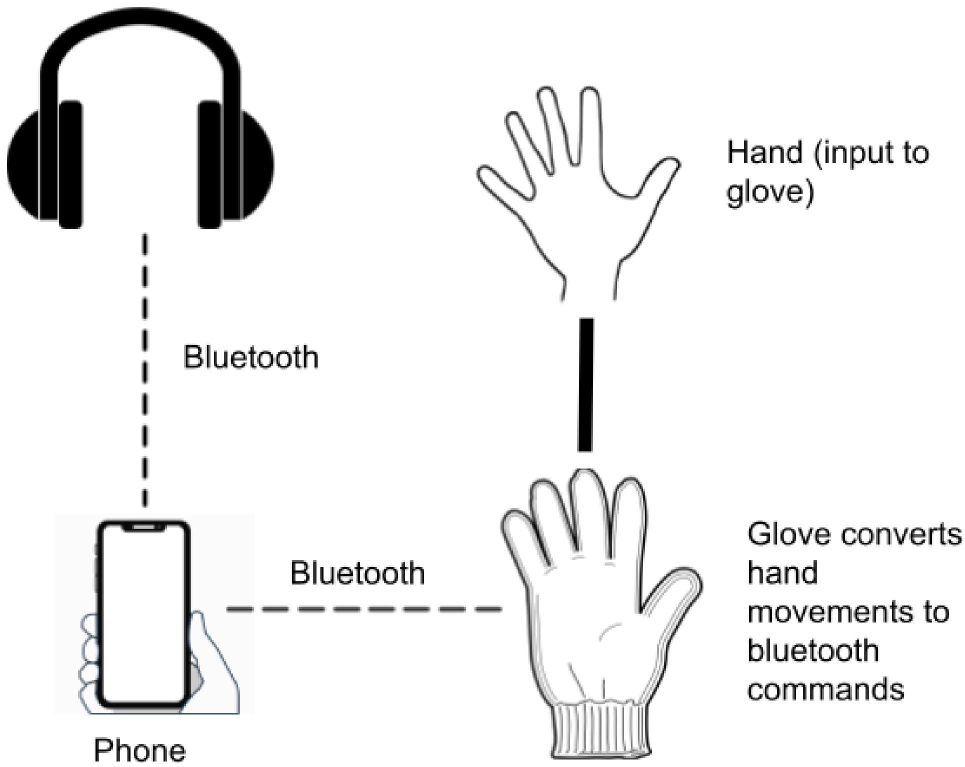
There are many situations in our lives where we may be wearing gloves. Activities such as construction, gardening, woodworking, and serving food require gloves. We also wear gloves when it’s cold outside. Since gloves are often bulky, wearing them makes it difficult for people to be able to control the music playback on their headphones and accept/decline calls. It’s hard to press buttons on your headphones, take your phone out of your pocket, or enter touchscreen commands with gloves on. Thus, people have to take their gloves off if they want to control their music playback and accept/decline calls. This is very inconvenient.

1.2 Solution

We will create a system of technological gloves that can be used to fix this problem. The gloves will be able to connect via bluetooth to a user’s phone and allow the user to control their music settings and accept/reject calls. The music settings the gloves will control are volume up/down, play/pause, and next/previous track. This will be possible through the use of flex sensors mounted on all five fingers and an IMU sensor that will be mounted on the hand. The IMU sensor will allow us to capture movement and changes in orientation at the wrist joint. The microcontroller will transform this sensor data into useful bluetooth commands, allowing the user to control their music/phone without having to remove their gloves.

1.3 Visual Aid

Figure 1: Visual Aid



1.4 High-Level Requirements List

In order to solve the problem in its entirety, we have three quantitative characteristics of a system that will accomplish our goals:

1.4.1 Correctly Identify Predetermined gestures From Sensor Readings

We need to be able to detect when the user provides input that matches any one of the seven predetermined gestures based on the sensor readings from the flex sensors and IMU sensor. In order to detect this, we will need to train a neural network model that identifies the correct command based on the gesture. We also want to limit the false positive gestures to prevent the user from giving unintended commands to the phone. This requirement will require us to create a neural network model with substantial training data.

1.4.2 Correctly Convert Predetermined Gestures into Correlated Bluetooth Commands

Once the gloves detect one of the seven predetermined gestures, they must provide the correct bluetooth command to the phone. The particular bluetooth protocol we will be using is the HFP protocol to accept and reject phone calls. To control music playback, we will use the AVRCP protocol. The goal of this requirement is to correctly use these protocols to send the specified command to the phone based on the gestures obtained from the previous requirement. We will know if the correct bluetooth command is sent based on the change we see on the phone.

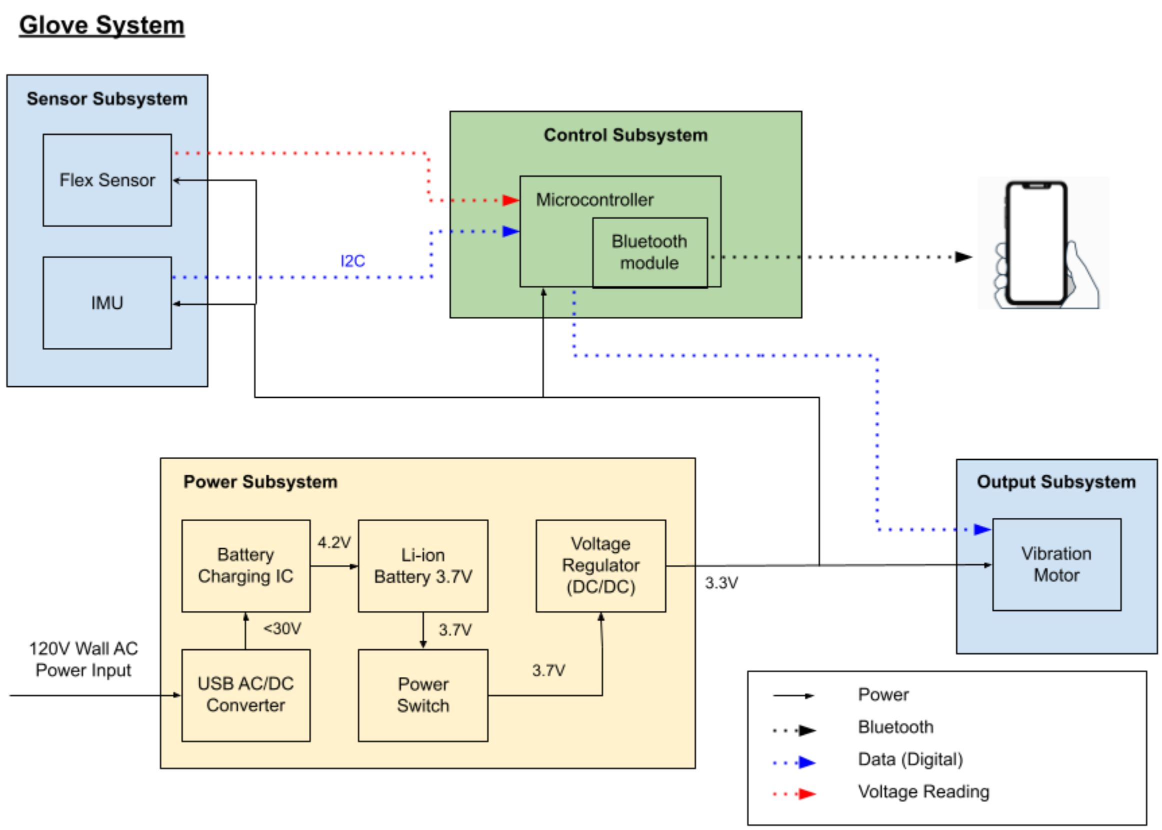
1.4.3 Glove Has Small Form Factor and Battery Life Sustains a Session of Use

We want to also ensure that the user has a final product that fits comfortably on the hand in a small form factor, while also having enough battery life to last a season's worth of wearing. In our specified use case, we have set a target of 3 hours battery life for the glove. This will be plenty of time for a glove session. In order to test the battery life, we will hook up our system to an oscilloscope and measure voltage across a shunt resistor, triggering the oscilloscope when the current goes to 0.

2 Design

2.1 Block Diagram

Figure 2: Block Diagram



2.2 Subsystem Overview

2.2.1 Sensor Subsystem

The sensor subsystem will consist of the IMU sensor and five flex sensors. The IMU sensor will detect orientation and movement at the wrist joint and arm through the use of the accelerometer and gyroscope. The flex sensor will indicate which fingers were flexed at the knuckle. In conjunction, these sensors will indicate the current gesture of the user's hand. The IMU and flex sensor will connect to the data I/O pins of the ESP32.

2.2.2 Control Subsystem

The control subsystem consists of the microcontroller. The microcontroller is responsible for taking the sensor input from the flex sensor and IMU and detecting whether any of the seven predetermined gestures are provided by the user. The microcontroller will be programmed such that when turned on, the bluetooth component enters pairing mode. This way, the user can pair the glove to their phone just by turning it on. If a gesture is detected, the bluetooth module embedded in the ESP32 will translate this gesture into a bluetooth command that is sent to the user's phone. The microcontroller will be connected to the IMU sensor, flex sensor, and vibration motor. The microcontroller will get its power from the power subsystem.

2.2.3 Power Subsystem

The power subsystem will be responsible for providing power to all the components in our design: the vibration motor, microcontroller, IMU sensor, and flex sensors. The power will come from a portable Li-ion battery the user can charge via a USB charging port.

2.2.4 Output subsystem:

The output system will consist of a vibration motor. The vibration motor will get output from the microcontroller to control its functionality and will be biased with power from the power subsystem. The purpose of the vibration motor is to provide haptic feedback to the user to alert them that their gestures have been recognised and sent as a command to their phone.

2.3 Subsystem Requirements

2.3.1 Sensor Subsystem

The sensor subsystem will consist of the five flex sensors and the IMU sensor. The IMU will be biased to 3.3V and the output data will directly connect to the I/O pins of the ESP32 microcontroller, with I^2C interface. The flex sensor can take a range of input voltages up to 5V, so we will bias the flex sensors to 3.3V as well. The output will also connect to the microcontroller. Since the flex sensor is just a variable resistor, we will construct a simple voltage divider circuit with a fixed resistance and read the voltage between the two resistors. This voltage value will change depending on the resistance value of the flex sensor. The more "flexed" the sensor is, the greater the resistance value. The sensor subsystem will communicate with the microcontroller (Control subsystem) to send gesture data and get power from the power subsystem.

Requirement 1: Since we will have a 3.3V power rail from our battery pack, we will supply 3.3V to the flex sensor. Since flex sensors are simply variable resistors, we will also need to add a shunt resistor to detect changes in the resistance value of the flex sensor based on changes in the output voltage reading. The flex sensor resistance ranges from 25k ohms (unflexed) to 100k ohms (most flexed).

Requirement 2: The IMU we will be using requires a 3.3V power supply from our power rail. Since we will be reading data using the I^2C protocol we will also require pull up resistors connected to the clock and data lines, which will connect to 2 I/O pins of the microcontroller.

2.3.2 Control Subsystem

The control subsystem contains the microcontroller and the bluetooth module, which is already embedded in the microcontroller we are using: ESP32-WROOM. The microcontroller will connect to the sensor subsystem and combine the flex sensor and IMU data to detect the current gesture of the user. It will also connect to the vibration motor to send the signal to buzz when a gesture is recognized. The controller subsystem will be powered through a connection to the power subsystem. The bluetooth subsystem will connect to the end user's phone and send the command associated with the gesture.

Requirement 1: Communicate with IMU using I^2C protocol with 2 I/O pins and read changes in voltage reading for the flex sensor connections to detect the degree of bend for the flex sensor.

Requirement 2: Train neural network model and use model to correlate sensor data to predetermined gestures.

Requirement 3: Receive 3.3V from the power subsystem.

Requirement 4: Translate identified gestures into bluetooth commands using HFP and AVRCP protocols and send commands to the user's phone.

Requirement 5: Output high signal to output subsystem when a gesture is recognized so the motor can vibrate. Output low signal to output system in all other cases.

2.3.3 Power Subsystem

The power subsystem contains the appropriate circuitry for a rechargeable battery system that delivers 3.3V to the other subsystems. The AC/DC converter takes a voltage input from an AC power outlet (120V), and converts it to a DC voltage. The battery charging IC then takes that DC voltage input and outputs a single power output with a steady current to charge the battery. We have a Li-ion battery (3.7V and 3Ah) as our rechargeable battery to store power, which is then connected to a power switch, allowing us to control the device. We then connect the battery to a voltage regulator to step down the 3.7V battery output to 3.3V.

The purpose of this subsystem is to provide the correct power to all the other subsystems for a substantial period of time. This subsystem delivers 3.3V to all of the other subsystems in order to power those components.

We will charge our battery using a USB interface, where our AC/DC converter will be a USB AC/DC power converter. We will only use the power line of the USB, which is 5V. This will directly connect the battery IC.

Requirement 1: The battery must have a capacity of at least 2.39 Ah (in order to have a battery life of at least 3 hours) and a nominal output voltage of 3.7V.

Requirement 2: The USB charging port should correctly deliver 5V to the battery charging IC to charge the battery.

Requirement 3: The battery charging IC must output a safe and steady supply current to charge the battery.

Requirement 4: The power switch must safely disconnect and reconnect the battery to the DC/DC converter.

Requirement 5: The DC/DC converter must output a voltage of 3.3V ± 5% to the rest of the subsystems.

2.3.4 Output Subsystem

The output subsystem contains a vibration motor in order to provide haptic feedback to the user once a command has been registered and sent. This is important because it alerts the user of the fact that their hand gesture has been correctly interpreted. This subsystem obtains power from the power subsystem at 3.3V in order to run the vibration motor. There is also a data line connection from the control subsystem to the vibration motor in order to control when the motor is turned on and off.

Requirement 1: The vibration motor should turn on or off when the microcontroller sends an appropriate on or off signal.

2.4 Tolerance Analysis

One risk to successful completion of the project is battery life. The goal is for the gloves to be able to run for 3 hours. Using some approximations with real components on the market, we can determine if this is doable. The components we're using that require power are the 5 flex sensors, 1 IMU, 1 vibration motor, and 1 ESP32 microcontroller. Looking at datasheets of real components on the market, we can look at the maximum current draw from each component, add it up, and use this as the average current draw of the glove from the battery (1). This way, we avoid underestimating the power consumption of the glove. Then, this approximate current average and the battery capacity can be used to find the battery life (2).

$$I_{avg} = \Sigma\, I_{component} \tag{1}$$

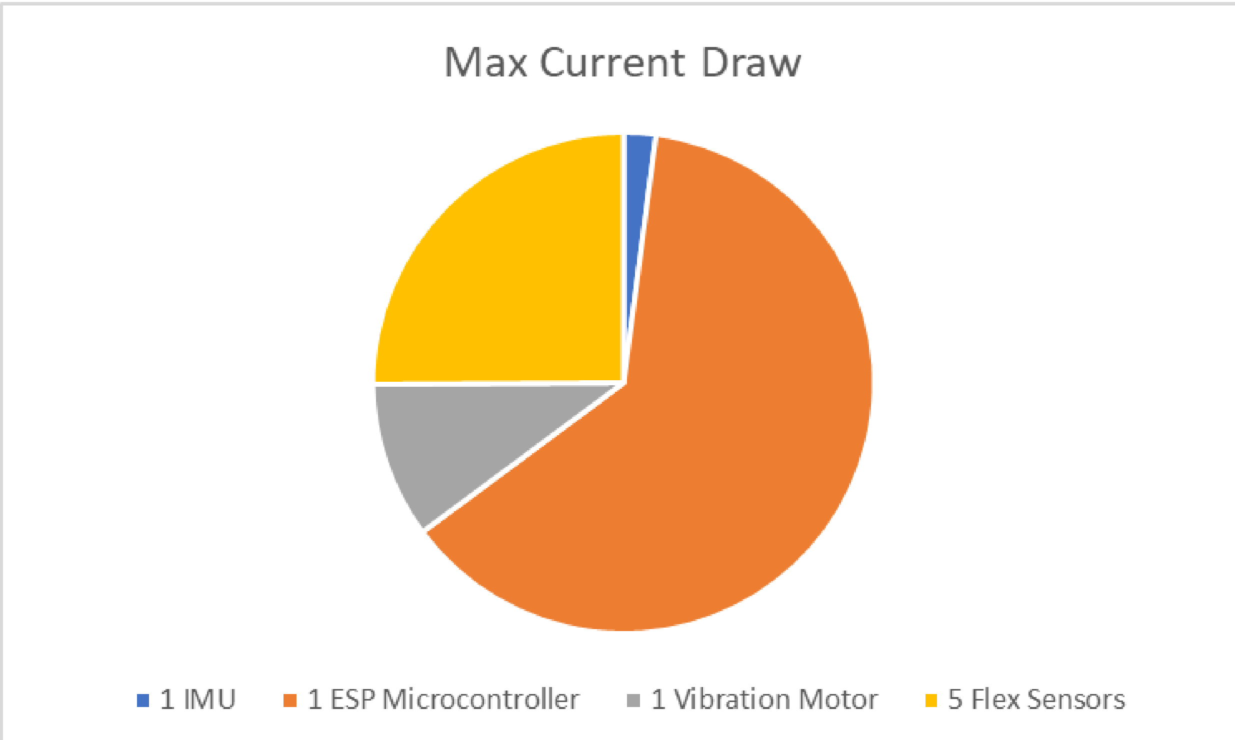
$$T_{bat} = C_{bat} / I_{avg} \tag{2}$$

Table 1: Max Current Draw of Each Component

Component	Max Current Draw (mA)
1 IMU	16.5 [4.5]
1 ESP Microcontroller	500 [4.4]
1 Vibration Motor	80 [4.3]
5 Flex Sensors	200 [4.2]

Total	796.5

Figure 3: Current Draw Component Breakdown



Looking at the pie chart, it seems the flex sensors take up around a quarter of the total power. For power optimization, we could add a feature that turns the glove off when no motion is detected for a certain amount of time. Since the battery has a capacity of 3 Ah [4.1], the approximate minimum battery length is 3 hours, 46.2 minutes. In order to achieve our goal of 3 hours, we would need an average current draw of 1 A. This calculation was done assuming the highest possible current draw from all the components, so we have some wiggle room in terms of power.

3 Ethics and Safety

For section I of the IEEE code of ethics [4.6], our project doesn't have too many issues. The gloves don't connect to the internet, so privacy shouldn't be an issue. The only thing that can theoretically be done maliciously is messing with a person's music, volume, or calls. However, this shouldn't be an issue because any established bluetooth connection must be consented to by the user of the phone. Our group will also uphold a high standard of safety, integrity, and responsible behavior in the process of working on the project.

For section II of the code of ethics, there are a couple safety concerns. Once concern is the risk of an accidental rapid increase in volume. This could cause ear damage to the user or disorientation that could lead to an accident. We should be able to remedy this using a feedback system, which will notify the user when the gloves are actively taking commands. Another safety issue is the microcontroller. The controller can heat up, which can cause burns to the user if it comes in contact with the user's skin. To prevent this, we can minimize the risk of skin-microcontroller contact. We also plan to treat each other as well as other groups fairly and with respect.

For section III, we plan to support and hold to account ourselves as well as other groups in following the IEEE code of ethics. Effective teamwork is essential in order for us to be able to successfully create our Bluetooth Enabled Gloves for Controlling Music. Thus, we will all ensure to act in a professional manner whilst not engaging in any discrimination and treating all people fairly and with respect.

4 Citations

[4.1] US Electronics, “Li-Io Polymer Battery Specification,” USE-104060-3K-PCBJST, 10/08/2021

[4.2] Spectra Symbol, “Flex Sensor,” 2014

[4.3] Pololu, Shaftless Vibration Motor 8x3.4mm, 1637

[4.4] Espressif Systems, “ESP-WROOM-32 Datasheet,” 3/9/2017

[4.5] Epson, “IMU (Inertial Measurement Unit) ,” M-V340PD, 3/2015

[4.6] *IEEE code of Ethics* (no date) *IEEE*. Available at: https://www.ieee.org/about/corporate/governance/p7-8.html (Accessed: February 4, 2023).