Spring 2023

Wireless Kick Pedal

Jacob Wise
jbw66@uakron.edu

Ryan Kinyo
rak113@uakron.edu

Bradley Toth
bst28@uakron.edu

Ian Zanath
iz9@uakron.edu

Wireless Kick Pedal

Jacob Wise – Honors Project Roll Description Statement

My technical duties for the wireless kick pedal consisted of the design and build of the Wireless Kick Pedal's base station unit. This included the development of the pedal's drive circuit which involved the sourcing of the actuating solenoid, using an N-Channel MOSFET transistor to switch the current flow through the solenoid, incorporating a wireless receiver to receive commands from the Wireless Kick Pedal's wearable device and drive the gate of the MOSFET to switch on the solenoid. I also determined how to deliver power to the base station so the solenoid strikes quickly and with as much force as possible to beat the drum head. I evaluated mounting considerations for the base station and determined how best to fit the base station to the test drum we used in the lab as well as how to fit the base station on other percussion instruments. Other technical tasks included support for the wearable device design through review of the PCBs manufactured for the wearable and designed by Ian Zanath and other engineering support in the manufacturing of the wireless charging circuit designed by Ryan Kinyo.

As project manager, my administrative duties included sourcing of necessary hardware components needed for the building phase of the project, submission of purchase requests for said components, being the main point of contact for the group between the project's coordinator and faculty advisor, scheduling and assignment of each team member for their specific subsystem design, as well as design and layout of the group's project poster and submission of reports or other documents required during the course of the project.

Wireless Kick Pedal

Project Design Report

DT01

Ryan Kinyo

Bradley Toth

Jacob Wise

Ian Zanath

Dr. Hamid Bahrami

04 / 03 / 2023

Table of Contents

List of Figures

List of Tables

Abstract

Playing an instrument is one of the best forms of self-expression. However, there is a limited number of accessibility options for aspiring musicians. The goal of this project is to change that by designing a wireless kick pedal for drummers who have an impairment of the leg or foot. This also creates a chance for multi-instrumentalists to incorporate new methods of playing in their work. The idea is to create two devices: a low-profile wearable device that will attach to the user, and a pedal base station that will receive signals from the wearable device to physically deliver the beat to the instrument. The wearable will be able to detect the user's motion, and wirelessly transmit a driving signal. That signal will then drive an actuator circuit that strikes the instrument. The wearable device runs off a rechargeable battery and the base station will receive power from a wall plug like those of guitar effects pedals. With this drum pedal system, the team hopes that more people will be interested in drumming and open the door to other novel instrument solutions. [JW]

- Does not require the use of a foot (but can still be foot operated).
- Allows user input detection up to 20ft away from the instrument.
- Rechargeable for prolonged use.
- Compatible with several percussion instruments.

1.0 Problem Statement

1.1 Need

For someone to play the drums, a drummer typically needs to employ the use of a kick pedal. The kick pedal allows the drummer to use their foot to play the bass drum of their kit. Other instrumentalists can employ the use of a kick pedal for other percussion instruments like a cajon. However, most (if not all) kick pedals are actuated through a mechanical linkage by a leg and foot. As such, a drummer with a leg or foot disability may have a difficult time playing their bass drum. In general, the music industry's accessibility options for musicians with a disability are few. A study conducted by Youth Music and the Take it Away Consortium showed that of music retailers surveyed in the UK, 63% were unaware of any musical instruments for disabled musicians. Out of those who were aware, only 38% knew where to find these instruments. The study also noted that 67% of disabled artists indicated that cost was a major barrier in playing an instrument [1]. Another study from the UK conducted by Attitude is Everything on artists from over 15 impairment groups found that 45% of disabled musicians ran into barriers while in the recording studio [2]. These studies indicate a need for more accessible, cost-effective, and robust instrument alternatives for disabled artists, such as an alternative kick pedal design. [JW, IZ]

1.2 Objective

The objective is to design an electronic kick pedal that can be actuated wirelessly. Actuation should be based on some sensible motion of the user. The sensor and the physical actuator to hit the drum shall be separate devices which communicate wirelessly. This will provide a more accessible option for using a kick pedal than with a mechanical design and allow a wide range of possibilities for musicians to incorporate the pedal into their playing, whether it is with a guitar or another instrument. [JW, IZ]

1.3 Background

1.3.1 Basic Theory

From the formation of the drum set, the kick pedal has allowed a drummer to have access to multiple parts of their kit at the same time. The original idea around a kick pedal comes from the idea of a "one man band" that allowed a musician to play drums with their feet while the rest of their body was able to play other instruments [3]. The modern drum set uses a kick pedal as the method for a drummer to play the bass drum. With the proposed kick pedal, the team hopes to encourage the spirit of being able to play multiple parts of a standard drum kit at once, even if the user does not have full range of movement in their limbs.

The general idea for the project is to develop two devices working in tandem to wirelessly hit a bass drum. One device will act as a compact sensor to track the motion of a user's body or appendage, and the second device will take the place of a traditional kick pedal. The two devices shall communicate wirelessly such that some motion of the detector will cause the kick pedal to actuate, striking the bass drum or the electronic bass-drum pad. There are many ways to achieve this, not limited to motors, linear actuators, pneumatics, springs, etc. The sensor could be realized as a simple gyro or accelerometer which is tuned to detect a change in direction, angle, or G force. The remote device should aim to be as small and light as possible to allow convenient attachment to a person and to reduce fatigue. As such, a small power supply must be devised in conjunction with a motion sensor, processor, and means of wireless communication. [IZ, JW]

1.3.2 Current Implementations

Currently there are no implementations that achieve actuation of a drum without the use of a foot or limb to physically connect with a pedal. Other designs of pedals do however achieve a goal like this project, which is making drums (more specifically the bass drum) easier to use. A patent by Ronald K. Hampton, named "Multi-trigger Electronic Drum Pedal" explains a design that makes it much easier to rapidly produce sounds while not having to exert as much energy as one would have to with a normal pedal. The design allows for one press of the pedal to generate two beats, one on press and the other on release. The patent even states previous pedals did not "provide multiple drumbeats with little additional effort or foot speed by the drummer" [4]. This design does not use a physical bass drum as our design proposes but instead uses a synthesizer or electronic drum set but it is worthy of note for the fact that it is attempting to make drums more accessible. [BT]

There are also implementations of electronic hobbyist instruments made with simple hardware like an Arduino development kit. In his book Digital Electronics for Musicians, Alexandros Drymonitis details the process of making a small digital drum set with 4 different pressure pads that can be played with the user's hands [5]. The drum includes a snare, a hi-hat, a bass drum, and a cymbal. This depicts a quick and easy way to combine music and electronics and is also accessible to hobbyists and musicians alike. And while this is not a full-size kit with all the additional parts, it does have all the instrument voicings necessary to play the drums. [JW]

1.3.3 Technical Limitations

Technical limitations for this project may include finding a way to integrate a small, lightweight motion sensor with a microcontroller and radio module in a way that does not consume large quantities of power. A device which only lasts 30 minutes per battery charge is not feasible for a musician to play even a small outing. As such, finding a reliable method of powering the device may prove challenging. Additionally, there are many ways to strike a drum. Pneumatics could be a reasonable first choice to electrically trigger motion with a lot of power, but it would require somewhat expensive and cumbersome infrastructure such as an air compressor to utilize. There is the possibility of using CO2 cartridges or scuba tanks to get around using an air compressor, but only being able to actuate a kick pedal 3 times before it needs its CO2 cartridge replaced is not feasible. Lastly, an air compressor is loud, a detriment to someone looking to produce and record music. Linear actuators are another way to electrically control motion, but they are typically slow. There is a tradeoff between speed and control of a linear actuator [6]. As such, a linear actuator may introduce too much delay or distortion through delay compensation. This is not ideal for a musician who wishes to play faster music. It does, however, solve the issue of large, cumbersome infrastructure and unwanted sound.

Another limitation for the project that comes from the usage of the product rather than its implementation comes from the limitations in the haptic response. A lot of what makes music so wonderful for both the listener and the player is the feeling of the music. For the scope of this project, the actual feeling in question is the connection that the artist makes with the instrument. From the book Musical Haptics by Papetti and Saitis, a musician physically interacts with their instrument, on the one hand, to generate sound and on the other hand, to receive and perceive the instrument's physical response [7]. This is what the authors refer to as the haptic channel. The proposed device will never be able to fully allow the physical connection between a musician (impaired or able-bodied) and the instrument as current kick pedals do. However, the proposed pedal may bring new methods and options of playing to a wide range of musicians. [IZ, RK, JW]

1.3.4 Comparison to Current Technologies

At the end of the day, the proposed design of the electronic kick pedal will still allow the user to strike the bass drum of a drum set. The pedal should operate nearly identically (if not identically) to the common mechanical kick pedal used by drummers today. The only difference is the mode of actuation of the pedal. While no alternatives to play the bass drum exist yet, there are alternatives to playing the actual drums.

With the dawn of digital and electronic music, musicians and programmers have developed several electronic musical applications using MIDI communication to create several digital alternatives to a variety of instruments. An article from Emusician.com entitled *Get Real!*

explains the rules and process of making a drum track using a MIDI sequencer. The article explains that a MIDI sequencer can create realistic and complex drum tracks that in some cases would be unachievable by a real drum set [8]. This of course is done without lifting a stick or kicking a pedal, like the goal of the proposed pedal. MIDI-sequencers also use electronics to work around the mechanical shortcomings of a regular drum set, much like the proposed pedal aims to do.

A similar application is in a gyro remote which uses the movement of the hand to emulate a standard computer mouse. This is a similar concept to the wireless kick pedal. Both must recognize and react to motion in a way that is repeatable, reliable, and predictable to be truly viable. That motion must then be processed into some form of signal for the device being controlled. In the example of the gyro remote, this is a computer. In the case of the wireless drum pedal, this is an actuator attached to a drum. Through this idea, a solid-state remote device with inertial sensors detects angular human motion [9] which is needed on the input end to transmit the signal to the pedal. **[**RK, JW, IZ]

1.3.5 Patented Technologies

While electronic actuation is a novel design for a kick pedal, there are some patents for kick pedals that attempt to improve the design of a pedal. Mark David Steele has several patents for an electronic kick pedal design. Steele uses a foot pedal to actuate a striking arm that then contacts electrical pads that translate into a beat on the bass drum. Several iterations of this pedal design have been filed, however, one of Steele’s more recent pedals allows the user to strike an assembly of two pads depending on whether the user raises or lowers their foot, making an electric version of a double bass pedal [10]. This patent shows that incorporating electronics into kick pedal design is the way of the future and that kick pedal design can benefit from the addition of electronics rather than being an entire mechanical unit. This pedal is designed specifically for electronic drum kits, however the wireless kick pedal proposed by the team intends to use electronics in the actuation of the pedal and be compatible with acoustic and electronic bass drums alike.

Another patent by Richard Sikra details an adjustable kick pedal. The pedal can be offset and configured in several orientations for the user to actuate the pedal [11]. While this pedal does not include any electronics, it does however stand as a successful attempt in the design of a kick pedal that can be shaped according to the user's needs. This pedal allows for change in its configuration that could benefit someone who may find it easier on their body if the pedal were off center a few degrees from a commercial kick pedal or needed the actual foot switch pad of the pedal to be a bit longer or shorter for their foot. This is significant to the design of the proposed wireless pedal as it is made in the attempt for a wider range of users to find a more comfortable and accessible mode of playing a kick drum pedal. [JW]

1.4 Marketing Requirements

1. The user shall be able to operate the drum pedal system without a foot.
2. The drum pedal system shall be actuated wirelessly.
3. The drum pedal system shall operate in real time with no noticeable delay.
4. The drum pedal system shall integrate seamlessly into both electric and acoustic drum kits.
5. The drum pedal system shall operate off readily available DC power sources.

2.0 Engineering Analysis

The team conducted research and analysis of the Wireless Kick Pedal's subsystems to further understand what should be required of the pedal and how the pedal must function to be a viable product for the user. The following describes the thought process for defining the kick pedal system's scope and the theory behind how it should be constructed [JW].

2.1 Circuits

2.1.1 Drive Circuit

One of the main elements of the wireless kick pedal design is the electronic actuator that will provide the means to strike the drum. This actuator requires a driving circuit for the pedal hammer to move. For the quick, forceful motion needed for the hammer of a kick pedal, one could use a solenoid or a servo motor. The drive circuit for a solenoid is a simple circuit, which uses a transistor to toggle current flow through the solenoid. As the solenoid can be viewed as a large inductor, it can store a fair amount of energy, which will have to be dealt with as back-EMF. This necessitates the use of the diode across the solenoid. A voltage divider is used on the gate of the drive MOSFET to protect it. A reference schematic for this driving circuit is given in Figure 1: [JW, IZ]

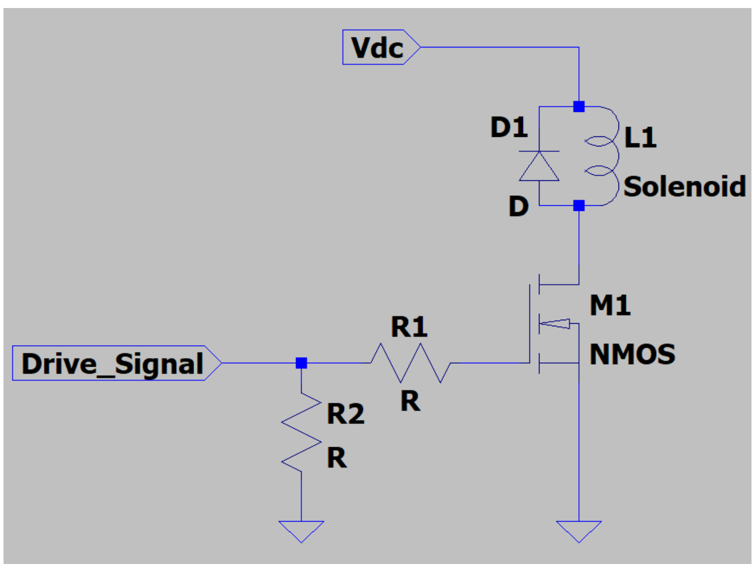

Figure 1: Generic Solenoid Driving Circuit

The drive circuit for a servo is more complex. Servos operate based on relative position and are controlled via a PWM signal. The PWM signal communicates whether the servo motor is driven forward or backward. This adds complexity to the drive circuit as it needs to be able to drive the servo in both directions, but not concurrently. The drive circuit consists of a timer IC and a network of switches and resistors to drive the servo either forward or backward depending on which switch is allowing current flow. [JW, IZ]

These circuits must be able to receive data and drive the actuator quickly to keep the beat. In each case, the drive transistor for the solenoid and the timer IC must be able to switch within a few microseconds to provide fast control of the actuator and must also reset the drum hammer quickly to deliver the next beat. Figure 2 depicts the default delay of one of the solenoids the team used in experimentation. From the figure, the actual delay of the solenoid is around 6.72ms. As the team had set a total transmission delay of 10ms, this value, while under specifications, does not account for the wireless latency, and therefore would be beneficial if the time could be reduced. [JW]

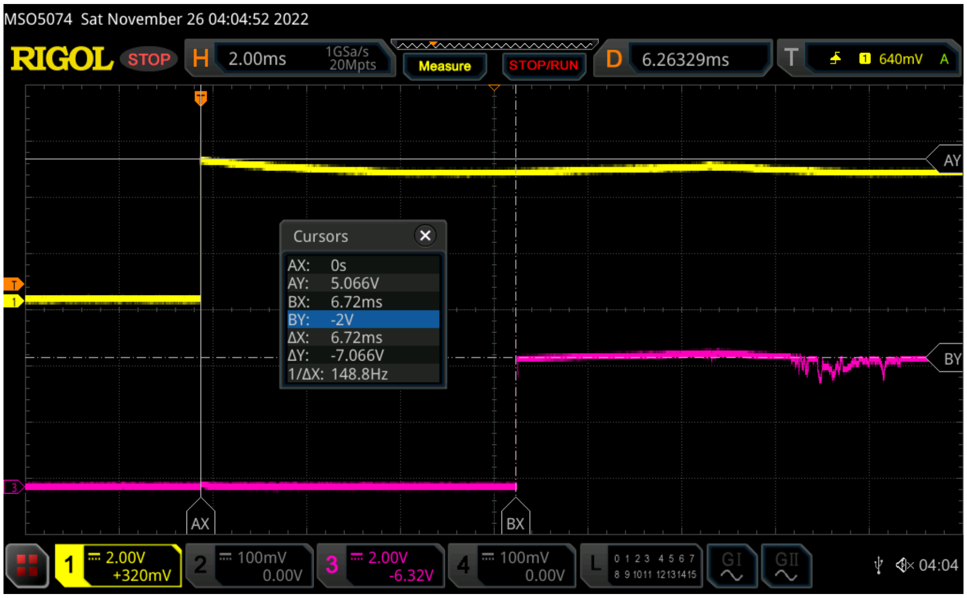

Figure 2: Oscilloscope Solenoid Delay Capture

2.1.2 Power

The overall drum pedal system will consist of two separate devices to facilitate wireless actuation. One device is to be a wearable band that attaches to the user and the other a base station with an actuator mechanism that receives user commands and delivers the beat. As such, the power requirements of each device are crucial to understand so that a user does not lose use of the system during a performance. The wearable device could be powered by a coin-cell battery due to its small size and convenience to obtain. An average coin-cell battery has a capacity of 150mAh. A typical musical performance lasts on average from 1.5 to 2 hours. The battery should last more than a single performance. A common voltage for a battery is 3.0V. The devices that will be powered from the source are an IMU, a microcontroller, and a wireless communication module. An average 6 axis gyro operates at 1.8V and consumes 6.1mW of power, this means the gyro draws roughly 3.4mA. The communication module operates with a voltage of 1.7V - 3.6V. The maximum current draw will be 14mA. The total current will be somewhere around 18mA. This points to a full operation power of the system at approximately 52mW.

1) $P = V \cdot I$

2) $= 3V * 17.4mA = 52mW$

3) $I = I1 + I2$

4) $= 3.4mA + 14mA = 17.4mA$

The base station will be powered with 120VAC in from a wall outlet, so it leaves the option open for either a servo motor or solenoid for the hammer to strike the drum. In each case, a DC wall adapter will need to be added after the source to convert from 120VAC to 12/24VDC for the control voltage. This serves the purpose of making the power design simpler on the base station as well as making the pedal like other industry standard electronic devices such as a guitar effects pedal. [RK]

It was suggested that the team determine the feasibility of adding a charging circuit to the wearable device to prevent the user from needing to replace a coin cell after only a few shows due to the possibility of a high-power draw from the wearable's electrical hardware. To approach this, the team researched and employed the use of lithium chemistry batteries (particularly single cell batteries) and the circuits needed to charge them properly. A lithium chemistry battery usually has a working or nominal voltage it is rated for, somewhere around 3.3 to 3.7V depending on the chemistry (lithium ion, lithium polymer, lithium iron phosphate). When the battery cell is charged, typically by an IC that handles the charging, the battery voltage will rise at constant applied current. Once the peak charge voltage is reached, 4.2V in a single cell Li-ion battery, the voltage is then held constant while the current decreases. The charge terminates once the current reaches a set level, for example 3% of its rated current.

Because lithium is a sensitive and sometimes volatile material if handled improperly, there are some rules for the discharging of a lithium chemistry battery. Namely, a lithium battery should never be fully discharged to 0V. This will cause significant damage to the cell, and it will be incapable of recharging again. To protect against this, a set of MOSFET switches can be used to prevent a battery from operating too far below its working voltage. A lithium battery should also never be overcharged or have too much current put through it. This could cause the cell to rupture and / or catch fire. Overcharge and overcurrent (short circuit) should also be monitored and protected against by other battery protection ICs. [JW]

2.2 Electronics

2.2.1 Communication Device

A wireless radio will be required in both the base station and the wearable. The unit in the base station can be as powerful as necessary, however the one in the wearable must be optimized for space and power efficiency. This may come at the cost of range, though this device is set up such that the range is not the biggest priority. The range for the pedal should translate to a room size of about 20ft by 20ft.

Two types of system can be implemented for the wireless interface: analog or digital. Analog has benefits as it is very simple to implement and can have a large range due to a low frequency. Digital is moderately more complex, though it has better noise immunity, and allows for more complicated data transmission. For example, settings could be transmitted between the wearable and the base station through a digital protocol. Doing the same with an analog radio is still doable, though more complex. One additional drawback in digital signals is common range. Since these signals tend to utilize a higher frequency, they do not tend to go as far. [IZ]

2.2.2 PCB Design

For the purposes of this project, a PCB design will be implemented for the wearable device. This will allow the wearable to be smaller and more practical for the user. The base station including the hammer will only require basic hardwired connections, so the main PCB focus of the project will be on the wearable's circuit. This board should be roughly the same size as a typical wristwatch or other wrist or leg mounted device. This is to keep the wearable device from becoming too bulky and having an unsecure fit on the user. To support this, the circuit board should be no less than four layers to accommodate tighter trace routing and any (relative to the board) large components that will be populated on the PCB. The PCB should have enough room to include terminal connectors for a commercial battery, such as a coin cell, and a USB programming port, if necessary. A PCB antenna should also be included on the board to connect to the base station and deliver the actuation signal from the user. [JW]

2.3 Signal Processing

2.3.1 Gyro Beat Detection

The key sensor used in the wearable device is some form of an inertial measurement unit (IMU). Through some experimentation with a 6-axis gyro/accelerometer, the most critical measurement device is an accelerometer. This will detect the upwards and downwards acceleration of a user, giving a peak along a graph whenever the user wants the drum to be played. Additionally, the magnitude of this peak can be used to determine how intense the user wishes the drum to be played. The higher the peak of the accelerometer data, the more intensely the drum should be played.

Additional concerns regarding the IMU are accuracy, power consumption, and polling rate. The most critical of these being accuracy. The IMU used should present no false positives to successfully detect both strong and weak beats. Additionally, power consumption plays a huge role in the uptime and reliability of the wearable device. The lower the power consumption, the longer the device will last. This principle can also be found in the wireless interface and microcontroller for the wearable. Finally, polling rates will have some impact on accuracy and detection. If the gyro polls once per second, it will be proven useless. A huge polling rate has a higher power draw and additional cost, so a balance must be reached between power draw, polling rate, cost, and accuracy.

Figure 3 shows a sample program utilizing the accelerometer for basic beat detection. All axes are summed together in this sample to make the device direction agnostic. The line in blue represents the raw IMU input, whilst the line in red represents smoothed input utilizing a moving average. The line in green indicates whether a peak has been detected, going high when the difference between the raw and smoothed inputs is greater than some configurable percentage. This percentage is the sensitivity of the algorithm. A lower value means that smaller movements are detected as beats, and a higher value means that larger values are required to be detected as a peak. The team found that this implementation of the algorithm was moderately accurate at filtering out the movement of an individual as they walked around the lab. [IZ]

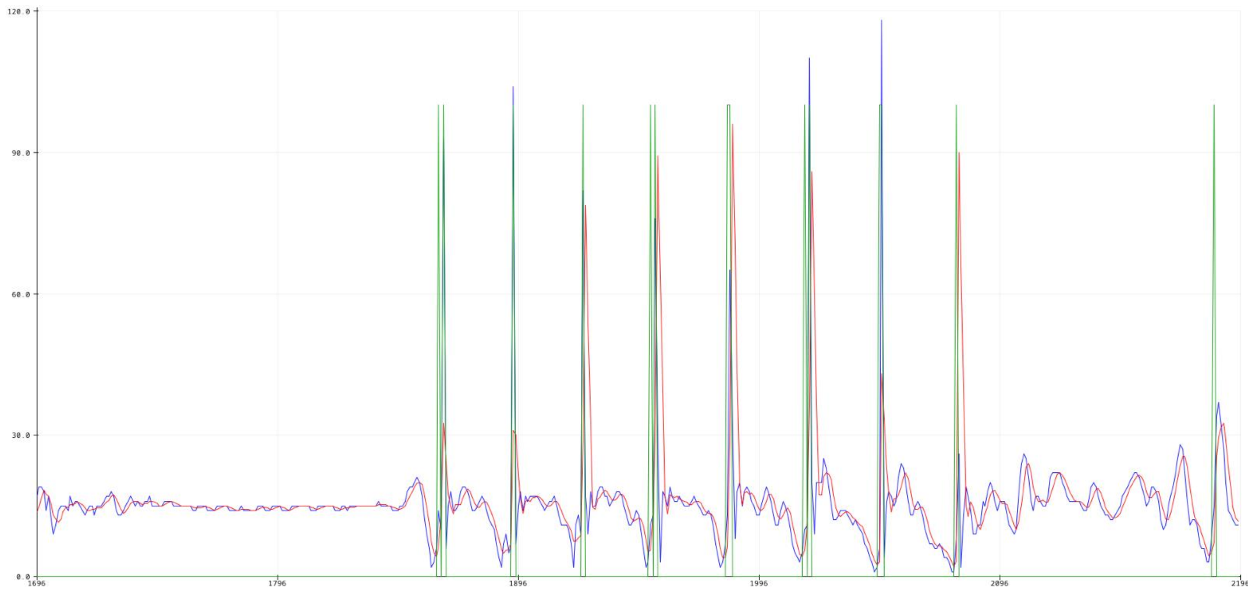

Figure 3: Beat Detection Sample

2.4 Communications

2.4.1 Communication Protocol

Some digital protocol shall be used to link the base station and wearable. This results in enhanced noise immunity and more data versatility. Analog radio is easier, but it is generally bulkier and less immune to noise. By adopting a digital protocol, additional data such as intensity of hit or user configured settings such as input sensitivity can be included. Additionally, certain microcontrollers have these digital radios built in, which increases space efficiency. This may come at the cost of power draw, though there are many options which offer low power consumption and exceptional range. [IZ]

Table 1: Wireless Protocol Comparison

Bluetooth Low Energy (BLE)	Wi-Fi
Pros: • Tiny SoCs available • Low power consumption	Pros: • Quick and reliable • Ease of implementation • Lower latency than BLE

Cons: • More difficult to implement • Higher latency	Cons: • More power draw than BLE • Larger footprint SoCs

Table 1 shows a basic breakdown of the two wireless protocols under investigation by the team. Bluetooth Low Energy offers the lowest power consumption of those the team has investigated; however, it provides a less stable latency, which may prove to be unsuitable for a device with the timing requirements of a drum pedal. Wi-Fi provides a lower and more consistent latency; however, it comes at the cost of power draw/consumption. One solution to this problem may be the sleep modes of these radios, which may allow the average power draw to be significantly decreased while the radio is not needed. [IZ]

Below are some oscilloscope captures that were taken to depict graphically what kind of transmission latency would be present for several protocols. The team tested the latency of Bluetooth LE communication, Wi-Fi communication and a proprietary transmission protocol available for use on the wireless communication devkits from Nordic semiconductor the team procured. Figure 4 depicts the latency of Bluetooth LE communication, Figure 5 shows the latency of the proprietary wireless devkit communication protocol, and lastly Figure 6 shows the latency of Wi-Fi communication. These concur with the expected results of Table 1 and show the pros and cons of each protocol as well as what the latency of something in between would look like.

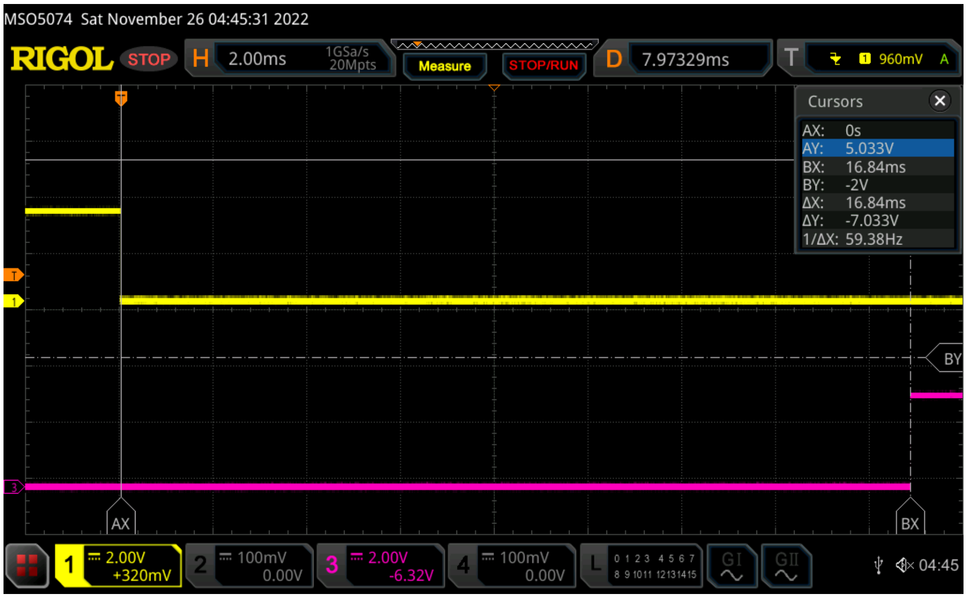

Figure 4: Bluetooth LE Latency (2.0ms/div)

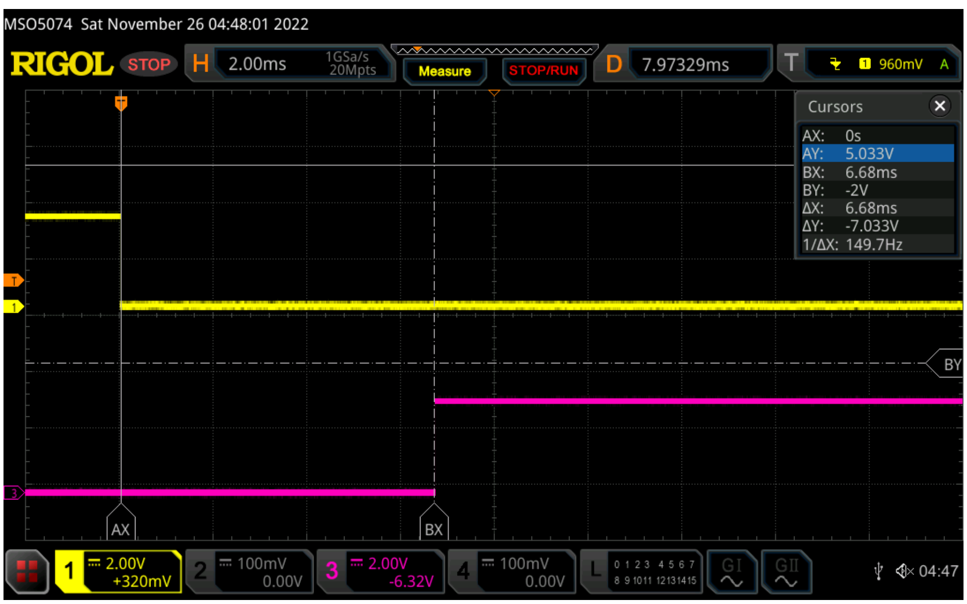

Figure 5: NRF Proprietary Protocol Latency (2.0ms/div)

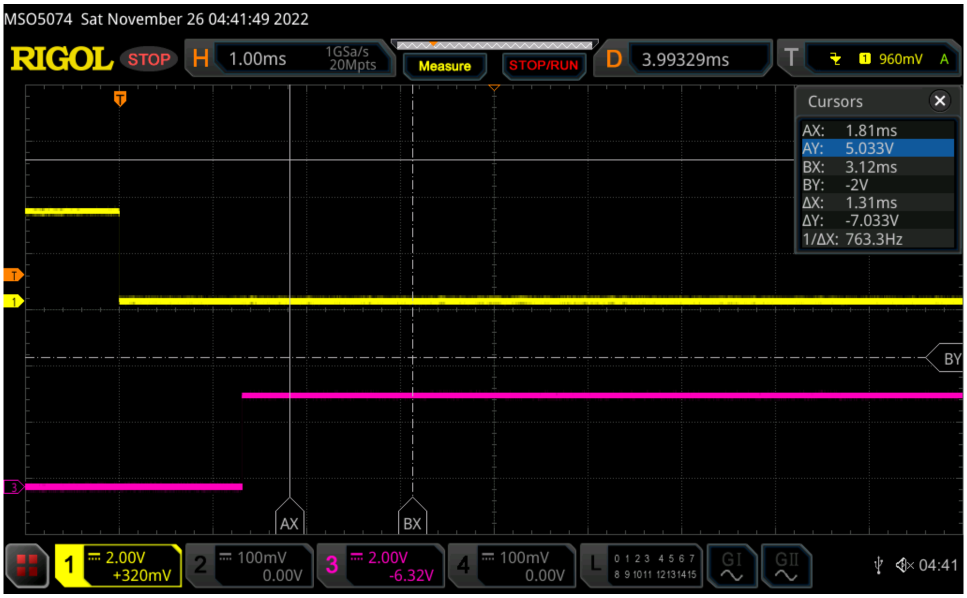

Figure 6: Wi-Fi Latency (1.0ms/div)

2.5 Electromechanics

2.5.1 Pedal Actuator

As mentioned previously, the drum pedal system should make use of an electronic actuator like a solenoid or a servo motor. At 120 BPM (beats per minute), playing eighth notes on a kick drum requires the drum to be hit 4 times a second or rather, the drum must be hit every 250ms to keep the maximum rhythm as defined in the requirements. Based on a lead study by Claudio Salvalaio, an average drummer exerts three levels of kicking force while playing the bass drum, characterized by a maximum force of 45N (approx. 10lbs.-force), a medium or standard force of 20N (approx. 4.5lbs.-force), and a minimum force of 5N (approx. 1lbs.-force). These values are summarized in Table 2 [12]:

Table 2: Average Force of Varying Kick Intensities

Kick Intensity	Force Generated
Heavy	45N (approx. 10lbs.-force)
Medium	20N (approx. 4.5lbs.-force)
Light	5N (approx. 1lbs-force)

(Adapted from Table 3 of Qualitative Evaluation of Physical Effort in Bass Drum Pedal Drive by Thermography by C. Salvalaio et. Al.)

A solenoid is a device that takes in electrical current to produce a magnetic field and generate linear motion from that magnetic field. Depending on the solenoid and the power given to it, a solenoid may actuate in a few milliseconds or over several seconds. Some commercial solenoids are available with a speed of 10ms or 30ms, with some being as quick as 2ms at voltages ranging from 3VDC to almost 400VDC. The other choice of actuator is a servo motor. A servo motor is a device that uses a pulse width modulation signal and a potentiometer to quickly drive the position of a small DC motor. Servo motors are also available in many sizes and configurations including full 360° rotation or linear motion along a single axis. They can operate on voltages ranging from 5VDC to 480VDC. However, when it comes to servos, the stroke length becomes a big player in determining the speed of the servo. A servo's speed is typically measured in inches per second to define how quickly a servo can reach its full stroke length. While a servo used to hit the drum need not have a long stroke length, the servo should be able to reach the drum and retract in a few milliseconds which may cause issues if the PWM signal needed to drive the servo already has pulses that are several milliseconds long. [JW]

2.6 Embedded Systems

2.6.1 The Base Station

The embedded system for the stationary kick pedal should be simple. A microcontroller will connect to the wireless interface and receive data about the user's playing. When some filter condition is met, the microcontroller will send a signal to actuate the pedal and strike the drum. Broken down to the simplest parts, this is a signal in, some signal processing, and a signal output. The output signal could simply drive a (number of) MOSFET(s) which cause an actuator to strike the drum. The power for this device can come from a standard wall outlet, as most drum sets are played where there is power readily available. This also decreases complexity and prevents users from having to manage heavy and expensive batteries. As such, the power drawn by this microcontroller is much less important. [IZ]

2.6.2. The Wearable

The embedded system for the wearable is likely to be more complex. A good microcontroller can handle all the communication and IO for the base station, however with power being a prime concern for the wearable, dedicated solutions for each part become a necessity. For example, some monolithic systems have a large current draw while operating. By utilizing a lower power microcontroller and dedicated components, battery life can be improved dramatically as each component has a particular use. This results in a three-IC design, where one is a microcontroller, another is an IMU (Inertial Measurement Unit), and the last is a wireless radio. Since all three devices have a dedicated purpose, the bloat can be cut from the system, and the power draw will decrease. One drawback to this, however, is a larger area taken up by ICs, requiring more layers in the PCB stack up, or more complicated trace routing.

In contrast, a monolithic system such as the ESP32 integrates the microcontroller and the wireless radio, in a small package, but it has a much higher power draw that is unsuitable for a wearable device. One preferable idea is using both an ESP32 and a dedicated wireless radio such that the ESP32 can be run at full power for debugging, then be put into sleep modes to conserve power, offloading communications to the dedicated wireless radio. This adds a fair amount of bulk to the overall design, however [IZ]

2.6.3 Microcontroller

A microcontroller is necessary to receive user input commands from the IMU and send a signal to the wireless module that the pedal needs to deliver a kick to the drum. Determining the force of the kick on the base pedal station side may also require a microcontroller. For the purposes of this project, a microcontroller with fast execution speeds is necessary to deliver kicks in time without noticeable delay. Standard drumming does come with some inherent delays. Namely, a drummer has about a 3ms delay between kicking the bass drum and their ears hearing it. The human ear can compensate for a latency of up to 10ms until a rhythm or note being played will sound off time. To ensure the system can execute a kick without a listener or musician hearing a delay in the playing, the microcontroller should be quick in executing its instructions. The faster the microcontroller, the faster an actuation signal is sent to the base station and the actuator can be triggered on time. Figure 7 below depicts the total latencies of the system once a kick is initiated by the user for 3 of the wireless protocols researched by the team. [JW]

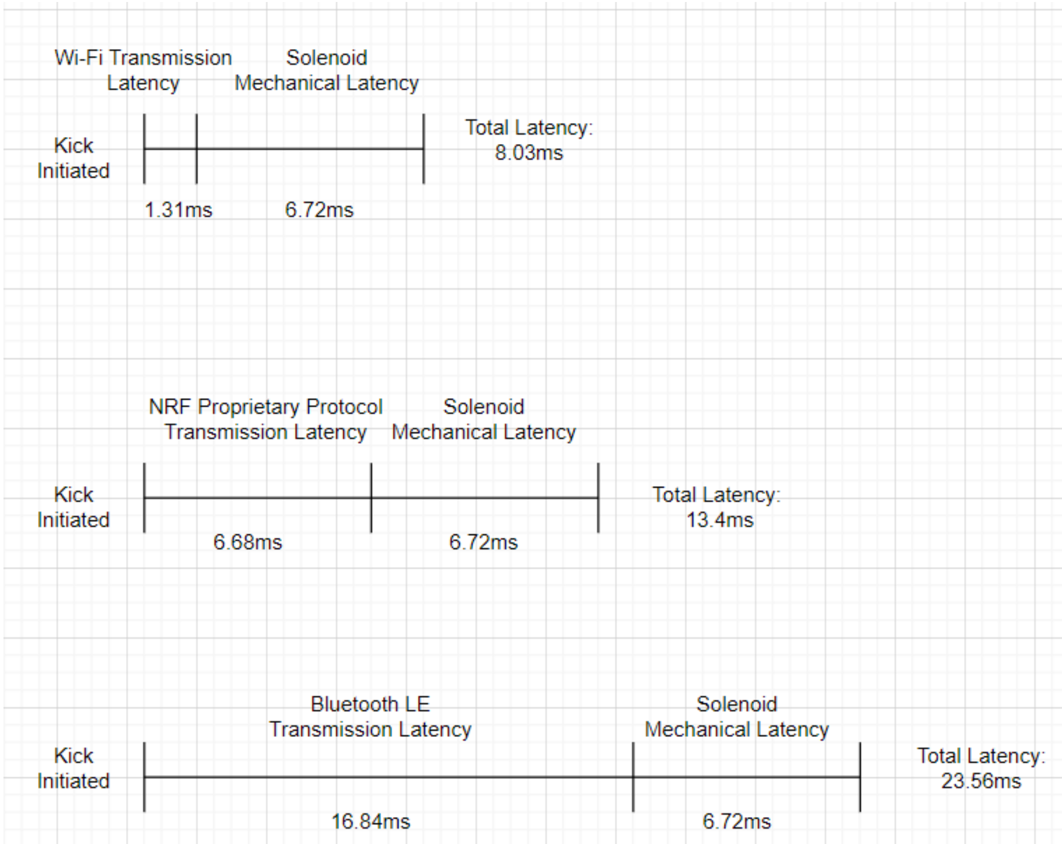

Figure 7: Timing diagrams for 3 wireless protocols researched

Microcontrollers will also be necessary for signal processing and interfacing on both the wearable and the base station. As these two devices are communicating wirelessly over a digital protocol, hardware is needed to communicate with the transmitters and receivers for the chosen protocol. Additionally, the IMU in use commonly requires a digital protocol to communicate. Analog IMUs may exist, though that adds extra complexity to a board design to keep RF noise down in the analog sections of a PCB. Finally, this microcontroller will be used for the processing of the signal output by the IMU. When the appropriate event is detected by the IMU, the microcontroller will send a signal wirelessly to the base station, causing the base station to hit the drum. [IZ]

Table 3: Wireless Module Comparison

NRF5X SoC	ESP32
Pros: • Ultra-compact • Fast • Enables Bluetooth LE integration • Ultra-low power draw • Supports multiple 2.4GHz transmission protocols, • automatic power saving modes	Pros: • Cheap and readily available • Bluetooth, BLE, and Wi-Fi available in a single package • Programmer set power saving modes
Cons: • Higher latency than Wi-Fi • Small size may be difficult to solder • Expensive	Cons: • High power draw • Smallest package significantly larger than NRF

Two candidates for the microcontrollers for this project are the NRF5x by Nordic Semiconductor and the ESP32 by Espressif Systems. Both are compact and highly integrated devices which incorporate a microcontroller and a wireless radio module into one package. This aids with the ease of implementation and keeps the size of the finished device small and compact. From the discussion in Table 3, The NRF5x has the lower power draw of the two options. The ESP32 has the higher power draw, but it supports both the Wi-Fi and Bluetooth Protocols. [IZ]

3.0 Engineering Requirements Specification

Table 4: Engineering Requirements

Marketing Requirements	Engineering Requirements	Justification
1, 2, 4	1. The drum pedal system shall be comprised of 2 separate devices. A user wearable device and a base station affixed to the instrument (drum).	For the system to operate wirelessly, the system must attach to a user and have a separate station to deliver the kick to the drum.

3, 4	2. The drum pedal system shall be able to deliver a kick every 250ms.	To play eighth notes at 120BPM, a kick must be delivered 4 times a second or once every 250ms.
2, 3	3. The drum pedal system shall have a latency of no more than +/- 10ms from the desired rhythm being played.	The human ear can detect latencies of 10ms from the desired rhythm.
1, 2, 3	4. The user shall be able to deliver a kick to the pedal from 20ft away.	A 20ft-by-20ft area accommodates a wide area for a musician not sitting at a drum kit to play in. This mimics a small stage or an average size room for practicing or gigging.
4	5. The pedal base station shall be able to deliver a kick with a force of 10lbs.-force to mimic the strongest kick of an average drummer.	An intense kick from a drummer on average has a force of roughly 45N.
5	6. The wearable user device shall be powered by a low voltage rechargeable power source.	Low voltage is commonly used for a rechargeable wearable product.
5	7. The pedal base station shall be powered by a readily available DC wall power plug of no more than 24V.	The pedal base station should be powered like a guitar effects pedal that takes a common barrel plug as its power input coming from a DC wall adapter.
4	8. The drum pedal system shall not cause damage to either the user or the instrument with which the device will be attached.	This ensures the safety of the user and the user's investment. Mechanical protection on both the wearable and the base station side should be in place such that the stress on the system is identical to the stresses introduced by a normal kick pedal.
1, 2, 3	9. The wearable device shall accurately detect intentional user input.	This ensures that the drum pedal system is not too sensitive or playing out of time.
1, 2, 4, 5	10. The wearable device shall have dimensions around 51mm in diameter by 13.5mm thick and include an adjustable wearable band.	This allows more comfortability for the user so there is not a bulky device attached to the musician.

Marketing Requirements:

1. The user shall be able to operate the drum pedal system without a foot.
2. The drum pedal system shall be actuated wirelessly.
3. The drum pedal system shall operate in real time with no noticeable delay.
4. The drum pedal system shall integrate seamlessly into both electric and acoustic drum kits.
5. The drum pedal system shall operate off readily available DC power sources.

Table 4 includes the engineering requirements set for the project. These requirements provide specific goals that can be validated numerically through measurements of the final developed product. These requirements also stem from the marketing requirements the team developed to gauge what a user would hope to get out of the system as a product and justification was provided to explain why such requirements were included. Several of the engineering requirements fulfill the needs of multiple marketing requirements. [JW]

4.0 Engineering Standards Specification

Table 5 lists the major engineering standards expected to be used for the project's duration. The team has determined standards for safety, communication, design methodology, programming language, and connectors. These standards are used to maintain the scope and general overall design of the project as the team moves forward with the design of the wireless kick pedal. [JW]

Table 5: Engineering Standards Specification

	Standard	Use
Safety	NEC 310-16	Wire gauges and current throughput safety
Communication	RS232 Bluetooth / Wi-Fi USB	RS232 may be used for programming. Bluetooth / Wi-Fi will link both the base station and wearable device USB for programming
Data Format	N/A	No data to be stored as of the present device.
Design Methods	PCB manufacturing 3D Printing	A PCB Shall be fabricated for the wearable device The wearable can likely be 3D printed
Programming Languages	C	C code runs on most microcontrollers
Connector Standards	USB	USB-C for programming connector

Communication within this project will involve Bluetooth, USB, and potentially RS323 serial. Bluetooth / Wi-Fi will handle the wireless communication between base and wearable, while the USB interface will handle the programming. RS232, I2C, and/or SPI will be used internally to communicate with our chosen microcontroller and to communicate with the IMU.

No data is presently expected to be stored on the device, however if data storage is required, it will be stored in EEPROM as an array holding integers or floating-point values to save user configuration options. Another option for this is status registers, setting registers high or low to act as configuration bits.

The design and manufacturing methods used for this device will include PCB fabrication and 3D printing. Creating a custom printed circuit board for the wearable device will help keep the device tightly integrated, and professional looking. Combining this with 3D printing allows for a protective outer casing to be created which keeps the electronics safe and prevents short circuits. This plastic case will also provide convenient points to use when affixing the device to a user.

The programming language of choice is C, due to its efficiency and prevalence. Additionally, most microcontrollers on the market run C code, so there is no real need to use anything more complicated or newer.

Finally, the only connectors found on the product will be that of USB-C and a DC Barrel Jack. USB-C will provide the ability for developers to interface with the product and flash new code, while the DC Barrel jack will allow for the base station to be powered. [IZ]

5.0 Accepted Technical Design

The following figures and tables break down the major subsystems of the wireless kick pedal. The hardware and software theories of operation will be discussed via functional requirements tables that explain the team's hardware and software block diagrams. The block diagrams begin at a Level 0 and progress in depth to Level N (in the case of hardware and software, Level 2). These describe the overall flow and innerworkings of the wireless drum pedal and how it is expected to operate. [JW]

5.1 Hardware Design

5.1.1 Level 0 Hardware Block Diagram

Beginning with the Level 0 hardware diagram, the wireless kick pedal is viewed as a single black box and shows the most important inputs and outputs of the system. These are the motion of the user, and input power, and then actuation of a pedals hammer as the final output and overall goal of the drum pedal system. The Level 0 block diagram is given in Figure 8 below. [IZ, JW]

Figure 8: Level 0 Hardware Block Diagram

Table 6: Level 0 Hardware Block Diagram

Module	Drum Pedal
Designer	Ryan Kinyo, Bradley Toth, Jacob Wise, Ian Zanath
Inputs	User Motion: From the leg, thigh, arm or other limb Power
Outputs	Pedal Actuation: Kick pedal hammer strikes the drum
Description	The Wireless Kick Pedal will take the motion of a user as an input to the system. The motion will be sent wirelessly as an actuation signal to strike a drum with the pedal hammer. Power will be delivered to the Wireless Kick Pedal from readily available sources such as wall power or a battery.

Table 6 shows the functional requirements for the overall drum pedal system with the critical inputs and outputs and how the team expects it to function. The system will have a wearable device powered by a small battery. This device uses an IMU to detect a movement that would be intended as a strike. This signal will then be transmitted wirelessly to the base station that contains the actuator to deliver the strike on the drum. [IZ, JW]

5.1.2 Level 1 Hardware Block Diagram

The design proceeds to the Level 1 hardware diagram that breaks the Level 0 diagram down to its major subsystems, Motion Detection, Wireless Transmission, Power Supply and Drum Striking, the intricacies of which are then broken down in the next level. The Level 1 block diagram is given in Figure 9. The tables that follow detail the functional requirements for each of the major subsystems. Table 7 explains the User Motion Detection system in place to gather a motional request from the user to strike the drum. Table 8 describes the necessary operation of wireless communications to deliver the request from the wearable to the base station. Table 9 details the requirements of the pedal drive mechanism that finally does the striking of the drum or other percussion instruments. Lastly, Table 10 provides an overview of what kind of voltages should be delivered to the wireless kick pedal's peripheral systems. [JW, IZ]

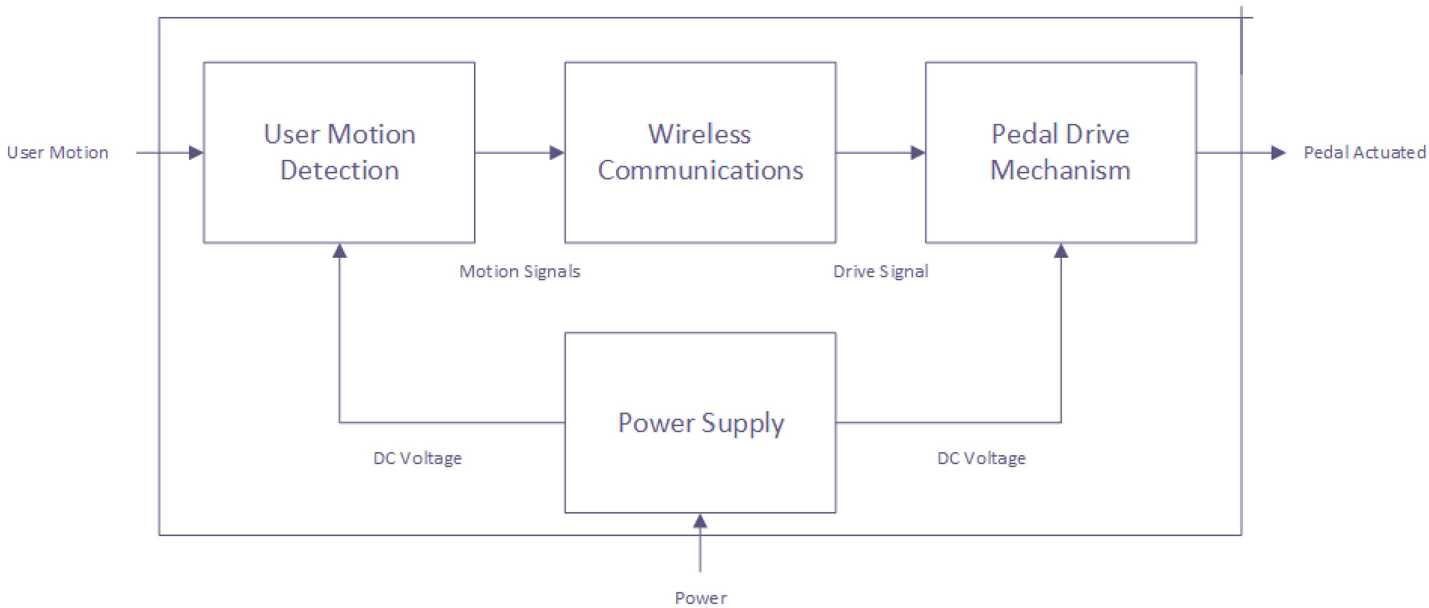

Figure 9: Level 1 Hardware Block Diagram

Table 7: Level 1 User Motion Detection

Module	Motion Detection
Designer	Bradley Toth, Ian Zanath
Inputs	User Motion: From the leg, thigh, arm, or other limb Power: On board device battery 3.0V
Outputs	Motion Signals: motion data sent to show user wants to deliver a kick
Description	Measures the acceleration of a user's movement and outputs motion signals that are sent wirelessly

As per the engineering requirements, the system must be able to detect and process user motion to strike the drum. This block outputs a high signal when the appropriate user motion is detected, causing the drum to be struck after the signal is communicated through the system. [IZ, JW]

Table 8: Level 1 Wireless Communications

Module	Wireless Communication
Designer	Bradley Toth, Ian Zanath
Inputs	Motion Signals
Outputs	Drive Signal: control signal that engages the pedal hammer
Description	Sends motion data from the wearable to engage the pedal hammer of the base station

Wireless Communication is part of the namesake of this project. The wireless communication system receives a signal from the motion detection system, transmits a packet to the receiver, which outputs a pulse to the pedal drive mechanism to deliver actuation. [IZ, JW]

Table 9: Level 1 Pedal Drive Mechanism

Module	Drive Mechanism
Designer	Ryan Kinyo, Jacob Wise
Inputs	Drive Signal
Outputs	Pedal Actuation
Description	Takes the drive control signal, engages the base station's pedal hammer and delivers a kick to the drum

The pedal actuation will begin from a wireless signal received from the wearable device. Once this signal has been received, the receiver will then deliver the signal to the actuator to enable the strike. [IZ, RK]

Table 10: Level 1 Power Supply

Module	Power Systems
Designer	Ryan Kinyo, Jacob Wise
Inputs	120VAC (base station)
Outputs	3.0VDC (wearable) 24VDC (base station)
Description	Readily available power sources that deliver DC Voltage to each of the drum pedal system peripherals

The power supply for the wearable device will be a single cell rechargeable battery. The voltage supplied from the battery will be 3.0VDC. Due to the current draw of the actuator to produce a strike of at most 45N, the converter must be rated for at least 72W of power. This is to accommodate a commonly available 24VDC wall adapter with an output of up to 3A. [RK]

5.1.3 Level 2 Hardware Block Diagram

Finally, the Level 2 hardware diagram, as mentioned, depicts the separation of the drum pedal system into two peripheral devices, the wearable and the base station. The wearable consists of a microcontroller, inertial measurement unit, wireless transmitter, and a power source. The base station consists of a microcontroller, mechanical actuator, drive circuit, and power source. The Level 2 block diagram with both devices shown is given in Figure 10. Table 11 describes the system device that does the actual detection of the user's motion. Table 12 details the microcontroller, which intakes the user motion data and processes it into a signal that can be sent to the rest of the system. Table 13 defines the requirements of the wireless modules that will transmit from the wearable device and be received by the base station. Table 14 shows the drive circuitry that will take in a received drive command and make the actuator move. Lastly, Table 15 explains the actuator the pedal system will use to physically strike the drum once the user's motion to actuate the pedal is received by its target. [IZ, JW]

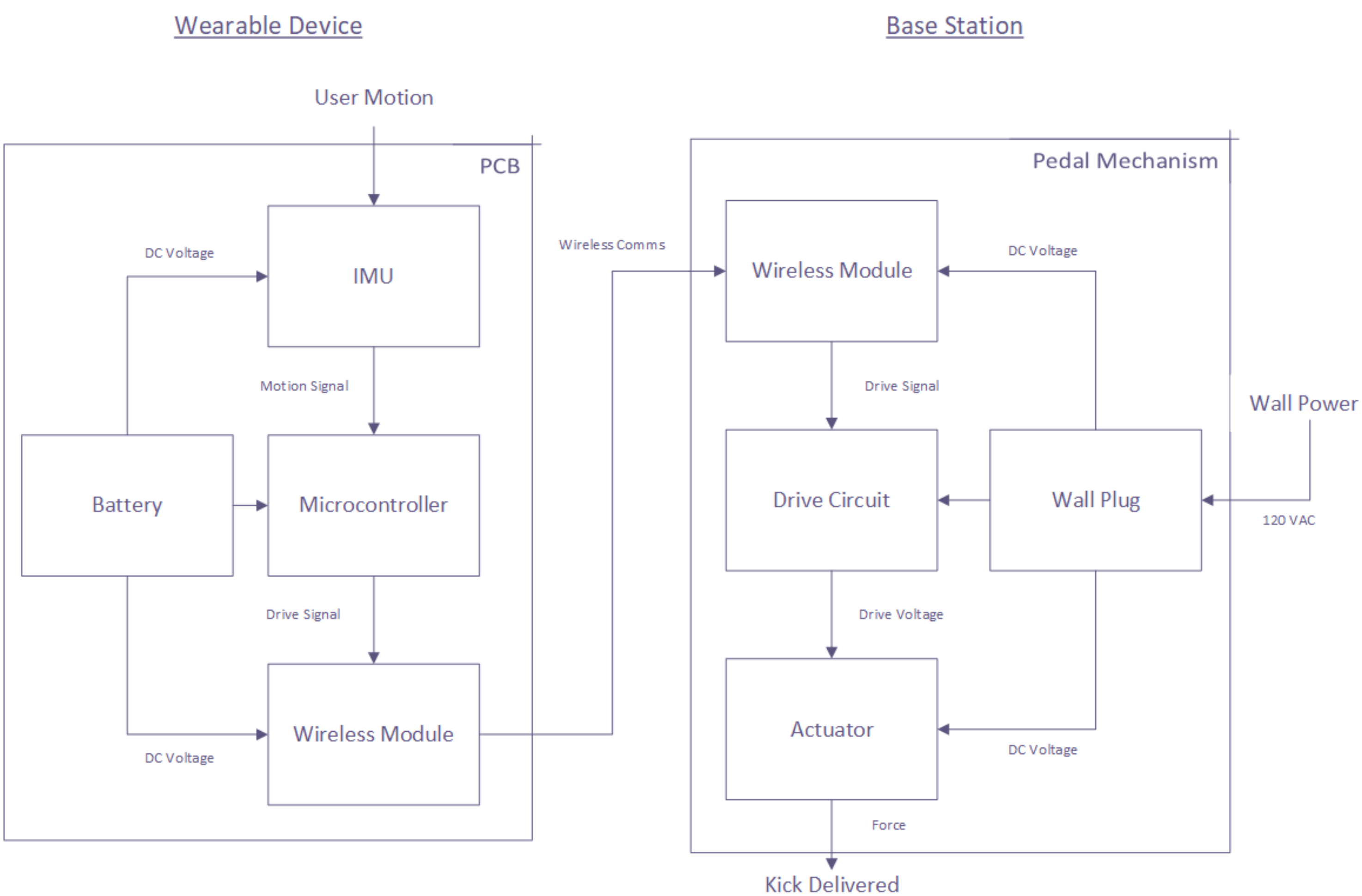

Figure 10: Level 2 Hardware Block Diagram

Table 11: Level 2 Inertial Measurement Unit

Module	IMU
Designer	Bradley Toth, Ian Zanath
Inputs	User Motion
Outputs	Motion Data (accelerometer peaks)
Description	Detects the acceleration of a user and sends motion data to microcontroller to be interpreted

The inertial measurement unit is used to detect user motion. This takes in power, and outputs motion data in terms of directional axes to a microcontroller for processing. [IZ]

Table 12: Level 2 Microcontroller

Module	Microcontroller
Designer	Bradley Toth, Ian Zanath
Inputs	Motion Data
Outputs	Motion Signal
Description	Accepts motion data and confirms that kick is within range before sending to base station

The microcontroller takes input from the IMU and detects the appropriate input signal, forwarding a signal to the wireless module which will be used to actuate the striker. [IZ]

Table 13: Level 2 Wireless Modules

Module	Wireless Radio
Designer	Bradley Toth, Ian Zanath
Inputs	Motion Signal
Outputs	Wireless Communication Signal
Description	Sends motion signals from wearable device to pedal base station.

The wireless modules communicate with each other and with the microcontrollers. When an appropriate signal is detected, a signal is transmitted to the base station which will cause the receiving microcontroller to send a signal through the drive circuit, striking the drum. [IZ]

Table 14: Level 2 Drive Circuit

Module	Drive Circuit
Designer	Jacob Wise
Inputs	Wireless Communication Signal
Outputs	Drive Signal
Description	Receives wireless motion signal from the wearable to drive the actuator

There will be a wireless signal generated from the wireless device that is transmitted to the base station. Once this signal is received, the actuator will be engaged and strike the drum. The basic design for this circuit is shown in Figure 1. [RK]

Table 15: Level 2 Pedal Actuator

Module	Actuator
Designer	Ryan Kinyo, Jacob Wise
Inputs	Drive Signal
Outputs	Pedal Actuation: strikes drum and delivers kick
Description	Completes striking of the drum

The pedal actuator will be the electromechanical device that will deliver the strike on the drum. The strike provided by the actuator will have to closely mimic the force of a human kicking the drum. Part of the force of the actuator is determined by the distance between the actuator arm and the striking surface. The closer the arm is to the striking surface, the harder the striking force will be. [RK]

Table 16: Level 2 Wall Plug

Module	Base Station Power
Designer	Ryan Kinyo
Inputs	120VAC: wall power from a socket
Outputs	24VDC, 72Wmax
Description	Converts 120VAC down to 24VDC to power the base station electronics

The base station device will be powered by a wall plug. The AC voltage from the wall will be converted to a DC voltage. The power needed to provide a force of at most 45N from the actuator will need to be a minimum of 72W. [RK]

5.1.4 Drive Circuit Subsystem Design

For the drive circuit design, the team acquired two different solenoids and built a simple MOSFET switching circuit around them, very similar to the schematic in Figure 1. These solenoids were rated for 12V and 24V to test the speed of actuation for different voltages. After testing, it was found that the 24V solenoid at its rated current of 4A was faster to actuate than the 12V solenoid. The remainder of the design and testing was done using the 24V solenoid.

To begin, the team created a simple test circuit in the browser-based circuit simulator Falstad. This program allows the user a quick and free way to test the basic operation of a circuit and the behavior of elements within it in an ideal or theoretical context. The Falstad circuit simulation schematic is given in Figure 11. The incoming drive voltage from the wireless signal and the base station's microcontroller was modeled as a simple 3.3V voltage source at the gate terminal of the NMOS that was flipped manually during the simulation. The simulation depicting current flow through the inductor when the gate is high is shown by the yellow waveform in Figure 11. The green waveform depicts the constant supplied voltage of 24V at the inductor's first terminal. This simulation helps give a basic understanding of how the MOSFET drives the solenoid by allowing current to flow through it. The resistors shown in the simulation are to ensure the gate is pulled to ground and not floating as well as to limit the current to the gate terminal. The team was also able to determine by using an RLC meter that the inductance of the 24V solenoid was approximately 12µH and is depicted as such in the simulation.

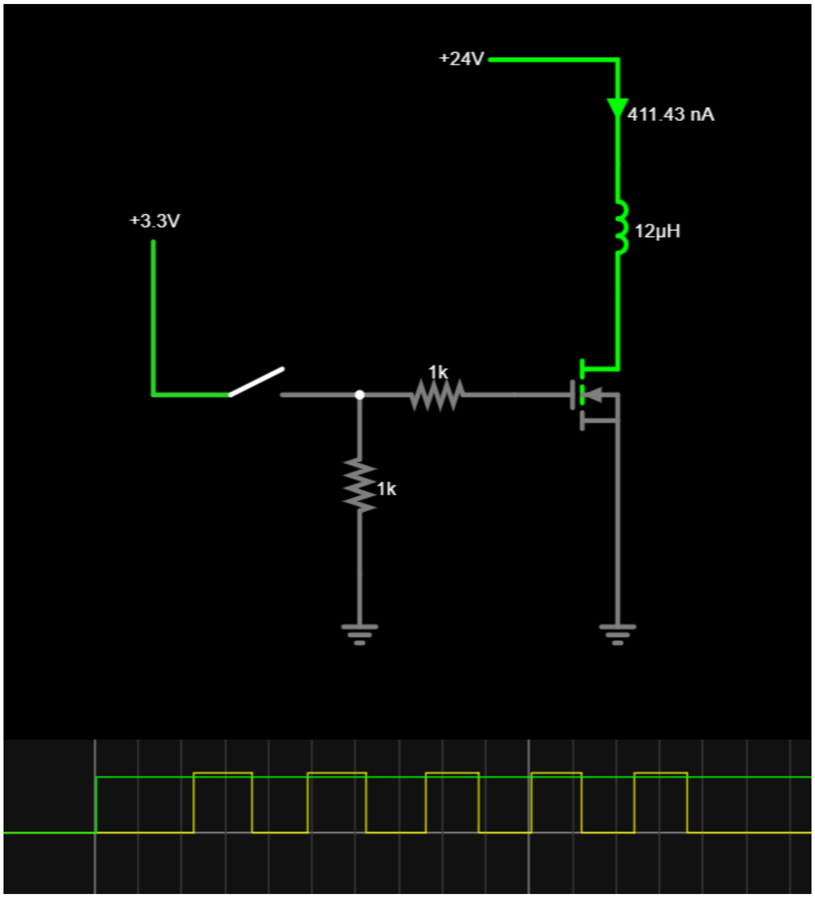

Figure 11: Falstad simulation for drive circuit

The team then used KiCAD design software to generate the schematic for the drive circuit. The final drive schematic is given in Figure 12. It was decided to use a 1kΩ resistor for current limiting and 10kΩ resistor to pull the gate to ground (a weaker pulldown). The SE15N50FRA N-Channel MOSFET used was provided by the team for testing of the circuit and is capable of handling up to 60A (pulsed), much more than is expected to be used in the base station. Two pictures of the realized drive circuit are shown in Figure 13.

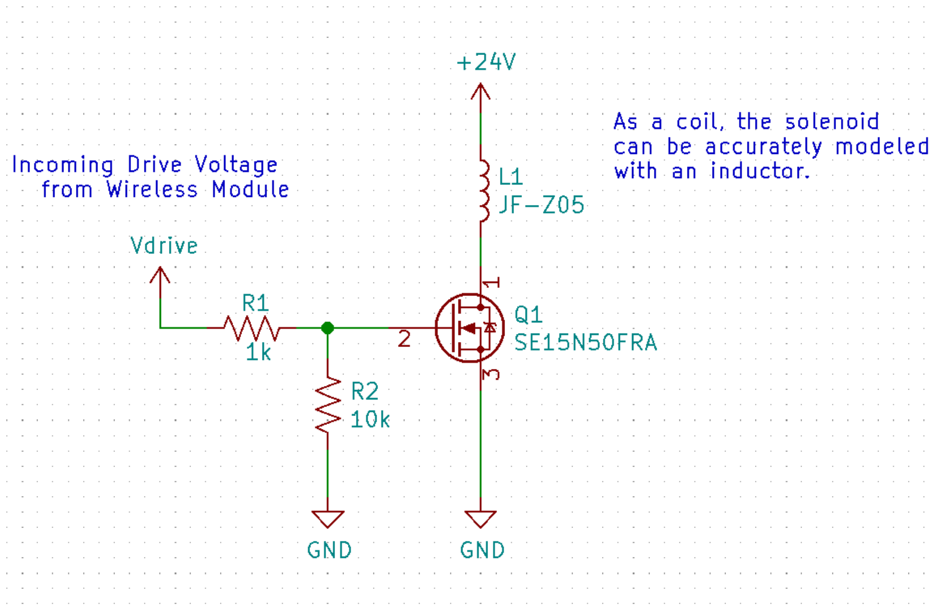

Figure 12: KiCAD Drive Circuit Schematic

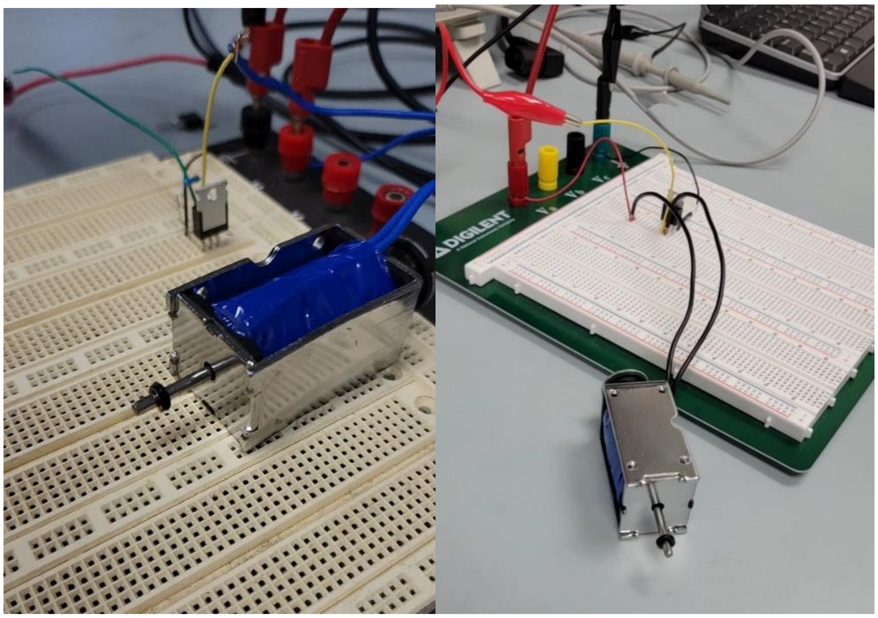

Figure 13: Realized Drive Circuit on bread board

5.1.5 Wireless Charging Subsystem Design

A wireless charging circuit was also added to be more user friendly and prevent the user from having to change batteries very often and also to provide more power to the microcontroller and IMU for longer. A basic wireless charging setup requires a power transmitter or a charging pad and then the actual device that receives charge from the transmitter. The transmitter generates an EMF by applying a periodic voltage to an LC tank circuit composed of a capacitor and an inductor in parallel. This EMF resonates at a desired frequency (in the team's case 141kHz) set by the value of the inductor and capacitor on the transmitter and receiver side. In particular, the transmitter includes a 6.3µH coil and a 200nF capacitor. The team designed the transmitter to use a NE555S-13 timing IC to control the gate of a MOSFET switch and thus turn a provided DC input of 12V to a periodic voltage for transmission. This power is also regulated and decoupled to provide a supply voltage for the timing IC and tuned with a potentiometer to ensure the correct resonant frequency is achieved. Using a 500Ω between reset and discharge and tuning the potentiometer to 260Ω sets the 555 timer to oscillate at 141kHz. The datasheet also recommends 10nF decoupling capacitors on the V_{CC}, CONT, and TRIG lines. The schematic for the wireless charging transmitter is provided in Figure 14. [RK, JW]

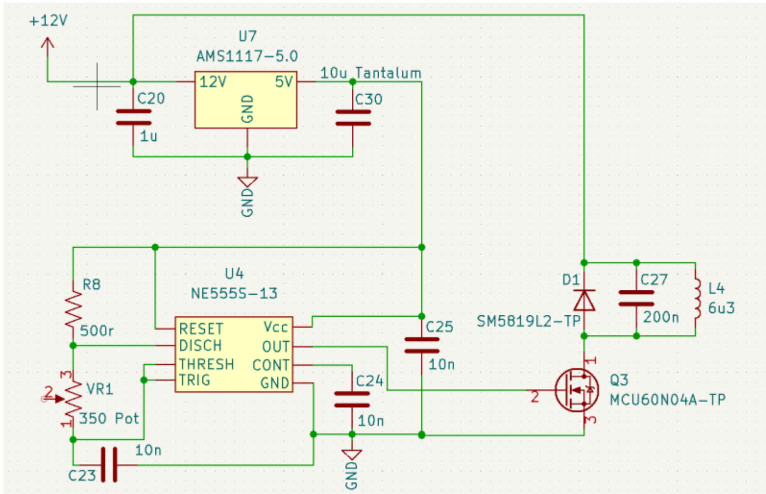

Figure 14: Wireless Charging Transmitter Schematic

Then, in order for the wearable to use this induced EMF for charging, the wearable must include a receiver circuit to take in this induced voltage, rectify it, and regulate it to be useful for charging. Another LC tank is constructed on the receiver side to match the resonant frequency and includes an 8.32µH and 152nF capacitor. It is then passed through a bridge rectifier circuit to turn the periodic voltage back into usable DC voltage to charge the wearable battery. This voltage is then regulated from 12V to 5V and connected to the rest of the battery charging circuitry. The wireless receiver circuit is depicted in Figure 15. A VBUS connection is also added with the intent of allowing the battery to receive charge over USB as well. [RK, JW]

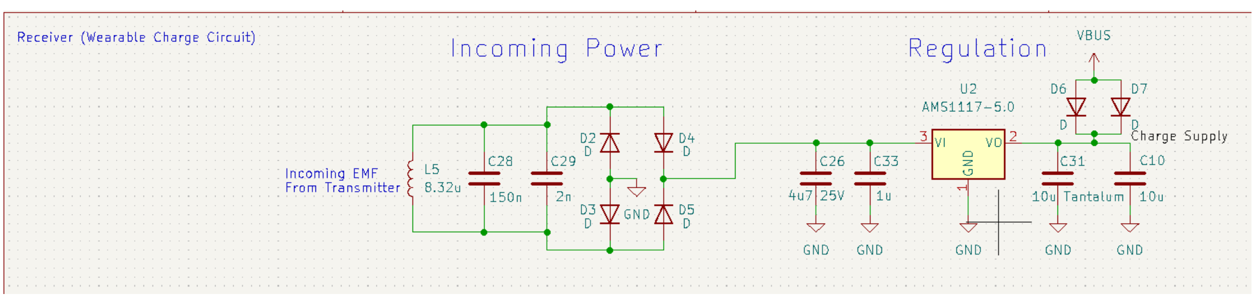

Figure 15: Wireless Charging Power Receiver Schematic

For the charging circuitry, the team decided to build around the TP4056 battery charging chip, which is a very common battery charging IC. This device takes in the supplied voltage from the wireless charging receiver and charges the battery at the desired current. A 1.5kΩ resistor between the IC's programming pin and ground sets the charge current to 780mA. This current will allow the charging of the 1000mAh battery to be more controlled and safer. The status of the battery's charging progress is also shown with two LEDs controlled internally by the IC. The battery charging circuit is provided in Figure 16. A DW01A battery protection IC is then used in conjunction with the TP4056. This chip can detect overcharge and over discharge. If the battery is over charged or discharged, the protection IC switches a dual MOSFET package that connects the battery power to the load. Depending on the scenario, the DW01A disconnects the battery from being drawn by the load or cuts the ability of the TP4056 to further charge the battery. A 1kΩ resistor connects the current sense pin (CS) to ground for protection in the case of an over discharge scenario. The protection IC circuit and the dual MOSFET switch package used with the protection IC is given in Figure 17. The wearable device may also be powered off by a simple toggle switch that is accessible by the user. [RK, JW]

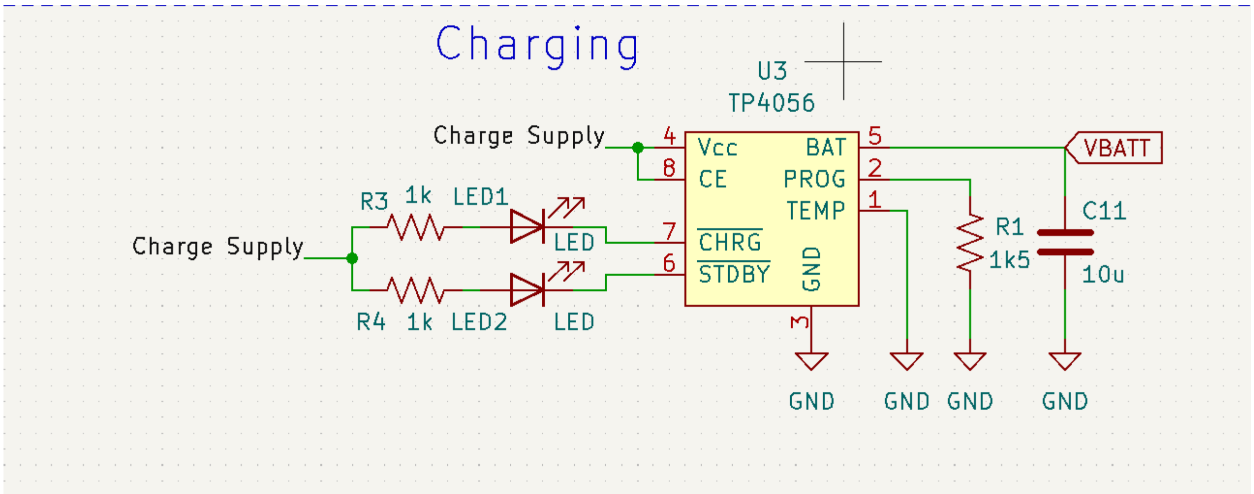

Figure 16: Battery Charging IC Schematic

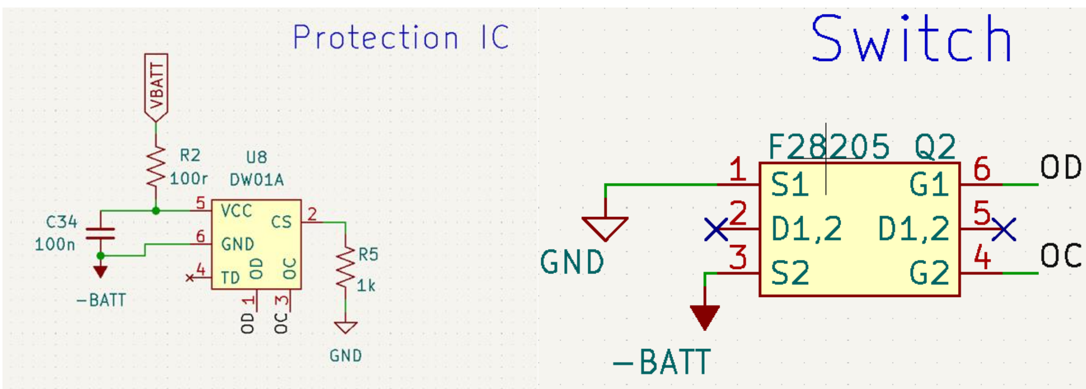

Figure 17: Battery Protection Schematic

5.1.6 Microcontroller Subsystem Design

The charging circuit mention will then be used to power the NRF52833 microcontroller. This microcontroller can handle wireless communications as well as receiving inputs from the IMU unit and providing the control for the base station drive circuit. The microcontroller circuit along with its peripherals, including a USB-C programming port, constituent power filtering capacitors, an impedance matching circuit for RF transmission and reception, and external oscillator are provided in Figure 18. This circuit is used on the wearable and the base station. To communicate with the IMU for beat detection on the wearable, the I2C lines on pins 14 and 15 are appropriated. For the base station control, pin 20 is used as a GPIO output for the received drive signal from the wearable. The impedance matching circuit, power filtering, and resonator circuit are derived from the NRF52833 datasheet. [IZ, JW]

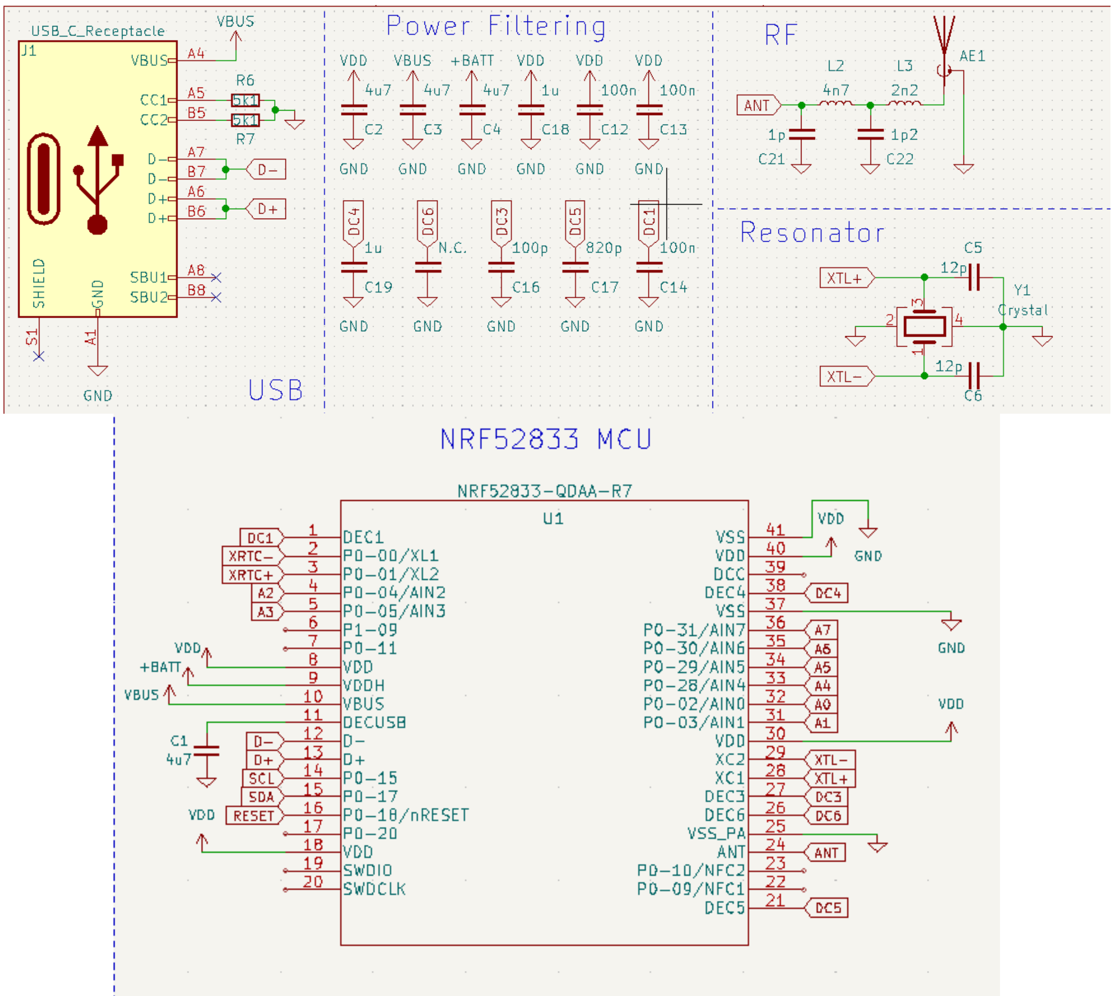

Figure 18: Microcontroller Schematic

5.1.7 IMU Subsystem Design

Finally, the LIS2DW12 IMU chip handles the reading of the user's motion data on the wearable and communicates over I2C to the NRF microcontroller to send the control signal for the base station. Apart from a few decoupling capacitors on the voltage supply lines, a 10kΩ pulls up the chip select (CS) pin to ensure that I2C communication is enabled. Neither of the interrupt pins are necessary and are thus unconnected. The reset pin is also connected to ground per the device's datasheet. The IMU also requires 1.8V to operate, which is regulated from the main battery supply voltage using a 1.8V regulator with its own decoupling network. The schematic for the IMU circuit is given in Figure 19. [IZ, JW]

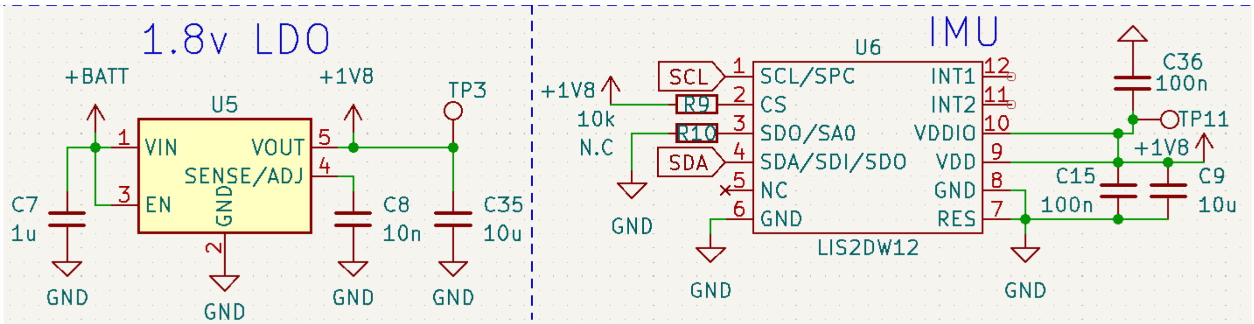

Figure 19: IMU and 1.8V Power Supply Schematic

5.1.8 Hardware Parts Lists

Table 17: First Request Parts List

Qty.	Refdes	Part Num.	Description
2	U1	NRF52-DK	Bluetooth LE Development Kit
1	L1	JF-Z05	24v Solenoid
1	L2	FJ-Z05	12v Solenoid
1	U2	ADXL345	3Axis Digital Accelerometer(+-16g)

Table 18: Second Request Parts List

Qty.	Refdes	Part Num.	Description
5	D1	1N3209	100V, 15A, DO-5 Diode
1	DZ1	1N2811B	13V, 50W, 5%, TO-3 Zener Diode
2	U3	LM747	Dual 741 Op Amp

Table 19: Final Request Parts List

Qty.	Refdes	Part Num.	Description
5	U1, U2	NRF52833	NRF52 MCU
100	C1,C2,C3,C4	V475K0402X5R100NCT	4u7 Capacitor, 10V Rated
100	C5, C6	CC0402JRNPO9BN120	12pf Capacitor
50	C7, C8, C9, C10,	CL05A106MP5NUNC	10uF Capacitor
100	C12, C13, C14, C	TCC0402X7R104M160AT	100nF Capacitor
100	C16	TCC0402C0G101J500AT	100pF Capacitor
100	C17	0402B821K500NT	820pF Capacitor
100	C18, C19, C20, C	CL05A105KP5NNNC	1.0uF Capacitor
50	C21	V1R0A0402HQC500NBT	1pF Capacitor
50	C22	V1R2A0402HQC500NBT	1.2pF Capacitor
20	L2	LQW15AN4N7B00D	4.7nH Inductor
20	L3	LQG15HS2N2B02D	2.2nH Inductor
5	Y1	X201632MKB4SI	32MHz Resonator
5	Y2	Q13FC1350000400	32.768kHz Resonator
10	U5	BCT2019EXKV18-TR	1.8v LDO
5	U6	LIS2DW12TR	IMU
5	J1, J2	Type C-31-M-12	USB-C Jacks
5	U3	TP4056X-42-ESOP8	TP4056 Battery Charging Module
4	U4	NE555S-13	555 Type, Timer/Oscillator (Single) IC 500kHz 8-SO
15	D1, D2, D3, D4, D	SM5819L2-TP	Diode Schottky 40 V 1A
1	L4	WT303012-12F2-ID	6.3µH Wireless Charging Coil Transmitter
1	L5	WR303050-12F5-ID	8.23µH Wireless Charging Coil Receiver
2	BT1	ASR00012	3.7 V Lithium-Ion Battery
4	Q1, Q2	MCU60N04A-TP	N-Channel 40 V 60A
5	U7, U8, U9	AMS1117-5.0	12V to 5V Linear Regulator 1A
3	LED3	APHHS1005LQBC/D-V	Blue 465nm LED
3	LED1	APHHS1005LSECK/J3-PF	Red 625nm LED
10	LED2	16-219AUTD/S3151/TR8	White LED
100	R1, R3, R4, R10	CR0402JF0102G	1k resistor
100	R5	0402WGJ0152TCE	1.5k resistor
10	Q3	FS8205A	Dual NMOS package
10	U10	DW01A	Battery protection IC
100	R6, R7	FRC0402J512 TS	5k1 resistor
5	C23, C24, C25	0402YC103KAT2A	0.01uF Capacitor
2	R8	TNPU0603500RAZEN00	500 ohm resistor
2	VR1	3214W-1-501E	350 ohm potentiometer
1	S1	MFS201N-16-Z	Slide Switch
10	R2	RC0805JR-0710KL	10k Resistor
2	C26	GRM155C61E475ME15J	4.7uF Capacitors , 25V Rated
2	C27	CAA572C0G2J204J640LH	200nF Capacitor
2	C28	C1206C154F5JAC7800	150nF Capacitor
2	C29	GCM1885C1H202FA16J	2nF Capacitor
10	C30, C31, C32	TPSR106K006R1500	10uF Tantalum Capacitor
3	R9	RCA0402100RFKEDHP	100ohm

5.2 Software Design

5.2.1 Level 0 Software Block Diagram

The software block diagrams start with the Level 0 wearable flow of control. After the wearable has gone through a boot up procedure it will attempt to establish a wireless connection with the base station. Once a connection is established, the wearable will send a signal to the base station to actuate the mechanism that strikes the drum. [BT]

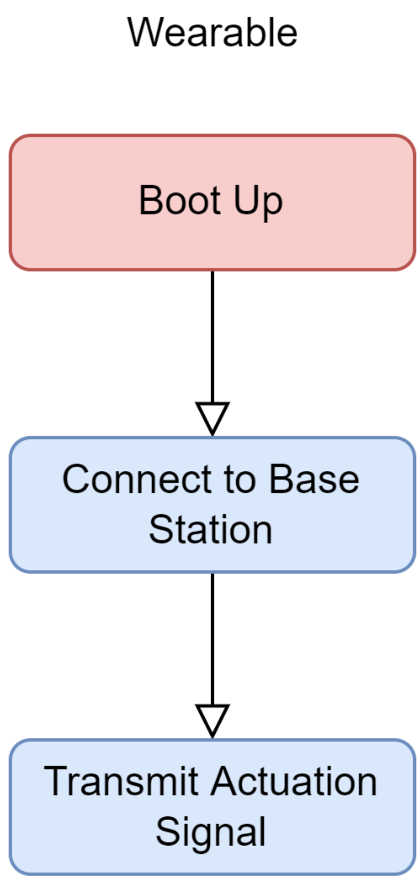

Figure 20: Level 0 Wearable Software Diagram

Table 20: Level 0 Software Wearable Block Diagram

Module	Wearable
Designer	Bradley Toth, Ian Zanath
Inputs	User Motion: change in momentum of the body part that the wearable is attached to
Outputs	Data Packet: Signal to actuate drum mechanism
Description	The wearable will detect user motion and send a signal to the base station

The base station at Level 0 has a similar flow of control to the wearable. It begins with a boot up sequence, then broadcasts out to attempt establishing a wireless connection. Once a connection is established the base station waits until it receives a signal, whenever a signal is received the drum will be struck. [BT]

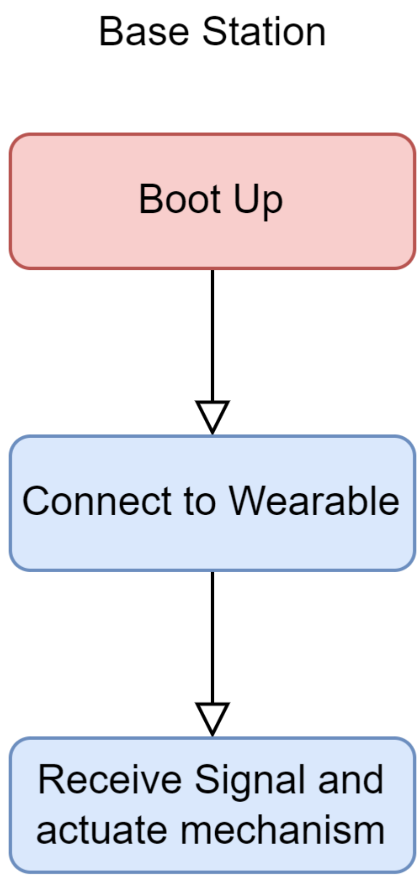

Figure 21: Level 0 Base Station Software Diagram

Table 21: Level 0 Software Base Station Block Diagram

Module	Base Station
Designer	Bradley Toth, Ian Zanath
Inputs	Data Packet: Signal to actuate drum mechanism
Outputs	Sound: Mechanism hits drum
Description	The base station will receive a signal from the wearable and hit the drum to produce sound

5.2.2 Level 1 Software Block Diagram

The Level 1 software block diagram explains more on how user input is interpreted by the device. It begins the same way as the Level 0 block diagram, with a boot up and establishing connection with the base station, after the initial setup it enters a loop to check for a connection. The wearable then waits for a change in input from the gyro sensor. When a change occurs the signal from the gyro sensor will be manipulated in some calculations to determine if the value is one that warrants a strike upon the drum. If the signal is not above the desired threshold the data is discarded and the next cycle of the loop will begin. If it is above the threshold the wearable sends a signal. [BT]

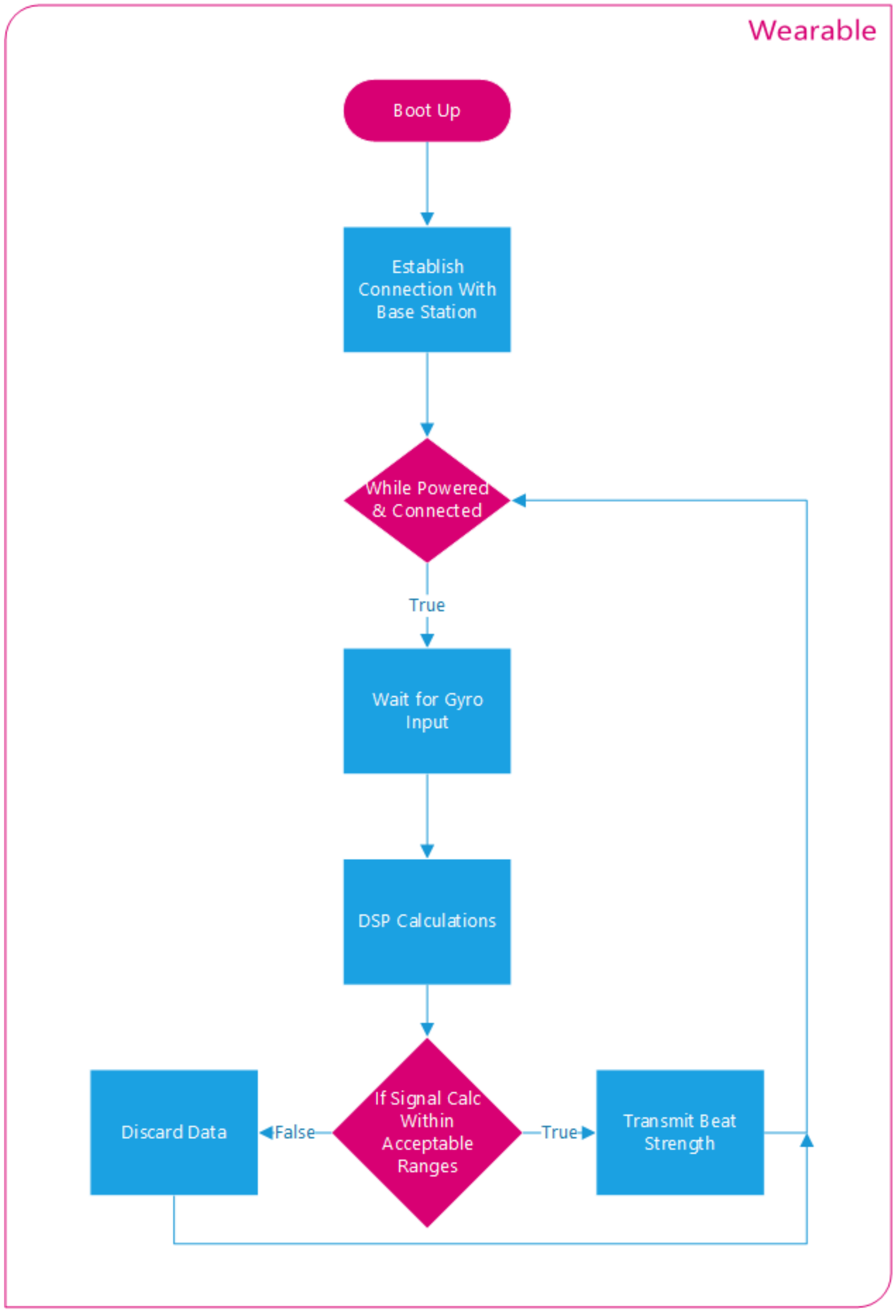

Figure 22: Level 1 Wearable Software Diagram

Table 22: Level 1 Software Wearable Block Diagram

Module	Wearable
Designer	Bradley Toth, Ian Zanath
Inputs	User Motion: change in momentum of the body part that the wearable is attached to
Outputs	Data Packet: Sent to the Base Station
Description	The wearable will send a data packet telling the base station to strike

The base station in the Level 1 block diagram has the same setup as the wearable. The base station also enters a loop that checks for connection after initial set up. Inside the loop the base station simply waits for a packet to be received. Once a packet is received and the data is extracted, the base station sends a signal to the striking mechanism. [BT]

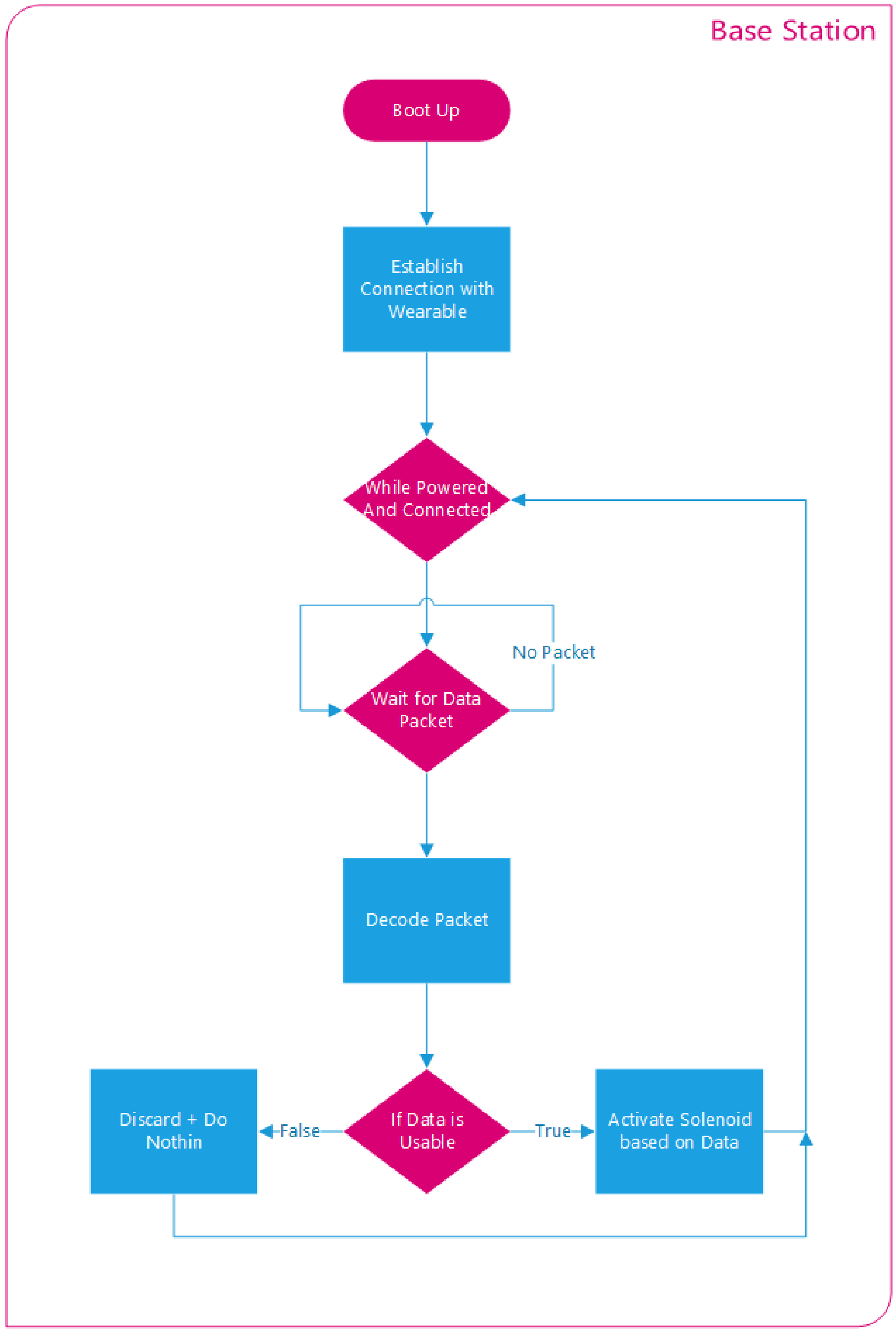

Figure 23: Level 1 Base Station Block Diagram

Table 23: Level 1 Software Base Station Block Diagram

Module	Base Station
Designer	Bradley Toth, Ian Zanath
Inputs	Data Packet: Received from Wearable
Outputs	Sound: Mechanism hits drum
Description	The Base Station will wait for data packets from the wearable, it will then hit the drum

5.2.3 Level 2 Software Block Diagram

In the Level 2 diagram, setup is the same for the wearable, the main difference in this level is describing how the signal processing is achieved. At the beginning of each iteration of the main loop, input from the gyro sensor is gathered. This data is then added into a set of previous gyro inputs and used to calculate the moving average. If the average changes above a threshold amount, it is determined that the value warrants a strike of the drum. The average is then checked against a high-powered threshold, if it is above that then a signal for a strong strike is sent, otherwise a signal for a normal strike is sent. [BT]

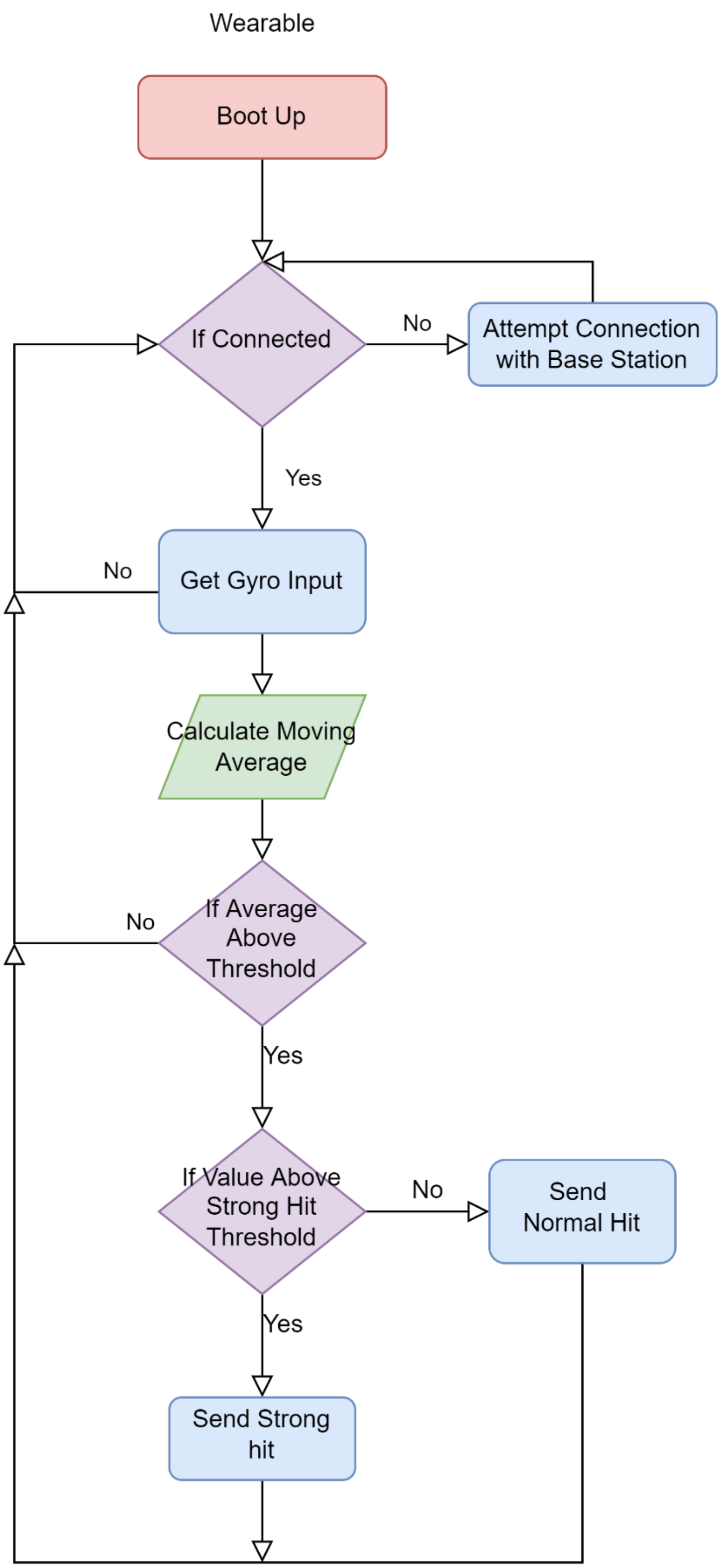

Figure 24: Level 2 Wearable Software Block Diagram

Table 24: Level 2 Software Wearable Block Diagram

Module	Wearable
Designer	Bradley Toth, Ian Zanath
Inputs	User Motion: change in momentum of the body part that the wearable is attached to
Outputs	Data Packet: Received by the Base Station
Description	The wearable will calculate the moving average of the output of the gyro sensor and output a drive command to the base station via wireless protocol.

For the base station, the Level 2 diagram digs deeper into the two striking powers provided by the drum pedal system, normal and strong. The setup and packet handling are the same as Level 1. When it comes to determining which level to strike the drum at, it will be a simple comparison of a Boolean. [BT, JW]

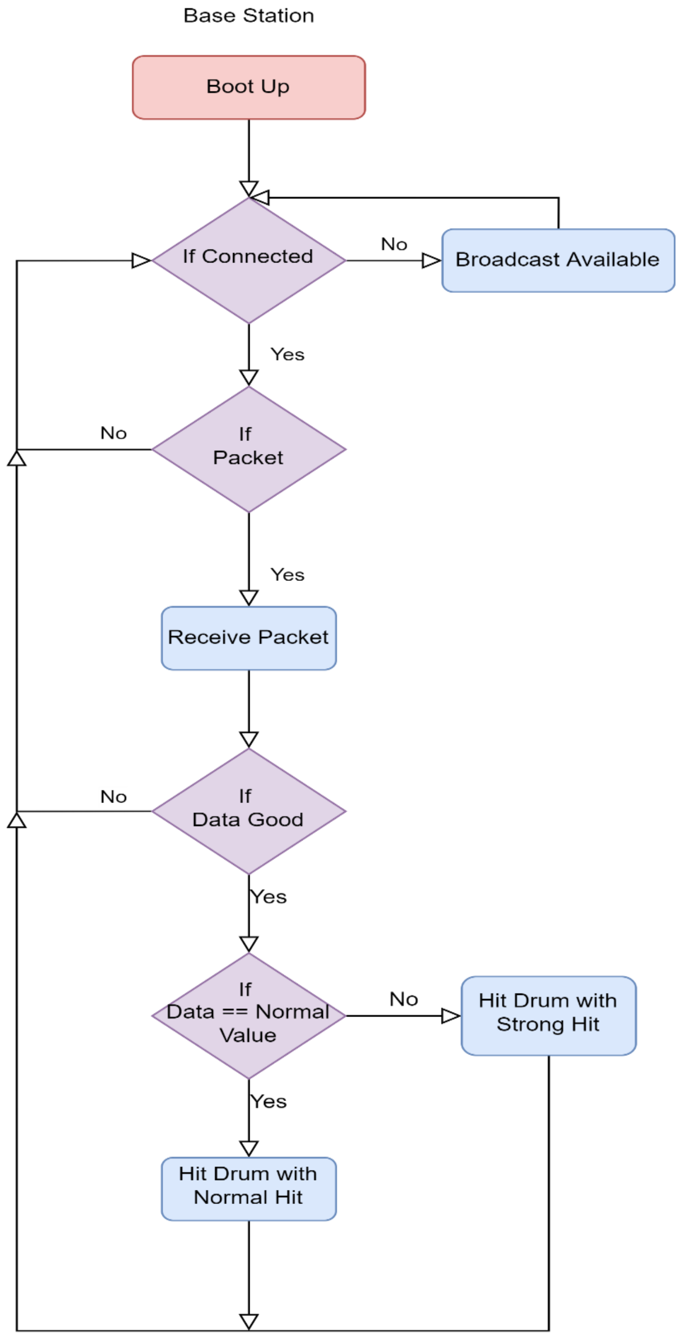

Figure 25: Level 2 Base Station Block Diagram

Table 25: Level 2 Software Base Station Block Diagram

Module	Base Station
Designer	Bradley Toth, Ian Zanath
Inputs	Data Packet: Received from Wearable
Outputs	Sound: Mechanism hits drum
Description	The base station will wait for data packets from the wearable, it will then hit the drum via command to switch on the drive circuit depending on power level

5.2.4 Software Design

5.2.4.1 Beat Detection Sample

The following code implements the IMU based peak detection as shown in Figure 3. [IZ]

```
// Libraries for InvenSense MPU6050
#include <Adafruit_MPU6050.h>
#include <Adafruit_Sensor.h>
#include <Wire.h>
#include <PeakDetection.h>

// Create IMU and Moving Average objects
Adafruit_MPU6050 mpu;
PeakDetection peakdetection;

void setup(void) {
Serial.begin(115200);

// initialize IMU
if (!mpu.begin()) {
Serial.println("Failed to find MPU6050 chip");
while (1) {
delay(10);
}
}

// Configure moving average for 4 samples.
peakdetection.begin(4, 2, 1);
```

```
// Configure IMU in 16G range
mpu.setAccelerometerRange(MPU6050_RANGE_16_G);
mpu.setGyroRange(MPU6050_RANGE_250_DEG);
mpu.setFilterBandwidth(MPU6050_BAND_21_HZ);
delay(100);
}

void loop() {
// Acquire new readings from the IMU
sensors_event_t a, g, temp;
mpu.getEvent(&a, &g, &temp);

// Sum all Accelerometer readings into one data point
int datapoint = abs(a.acceleration.x)+abs(a.acceleration.y)+abs(a.acceleration.z);

// Add the above data point to the moving average.
peakdetection.add(datapoint);

// Print data to serial monitor as a nice graph.
Serial.print(datapoint);
Serial.print(",");
Serial.print(peakdetection.getFilt());
Serial.print(",");
Serial.print('0');

// Logic to detect peaks. If RAW is more than 1.8*smoothes, set the third variable to 100.
// Else, set the third variable to 0.
if(datapoint > peakdetection.getFilt()*1.8){
Serial.print("100");
for (int i=0; i < 5; i++){
mpu.getEvent(&a, &g, &temp);
datapoint = abs(a.acceleration.x)+abs(a.acceleration.y)+abs(a.acceleration.z);
peakdetection.add(datapoint);
}
}

// Finalize print and wait 10ms.
Serial.println();
delay(10);
}
```

5.2.4.2 Wi-Fi Transmission Sample

The following code implements the server of our WiFi protocol testing. ESPNow provides a simple and convenient method of transmitting a few bytes of data between two WiFi equipped microcontrollers. This protocol’s biggest drawback is its high power consumption, measured around 135mA while operating. The benefit is it is ultra-low latency at just 1.3ms as shown in Figure 6. [IZ]

```
// Client Code
#include <esp_now.h>
#include <Arduino.h>
#include <WiFi.h>

// Structure example to receive data
// Must match the sender structure
typedef struct struct_message {
  uint8_t bop;
} struct_message;

// Create a struct_message called myData
struct_message myData;

// callback function that will be executed when data is received
void OnDataRecv(const uint8_t * mac, const uint8_t *incomingData, int len) {
  memcpy(&myData, incomingData, sizeof(myData));
  Serial.println(myData.bop);
  if (myData.bop == 1){
    Serial.println("Inside the else-if");
    digitalWrite(2, 1);
    delay(500);
    digitalWrite(2, 0);
  }
}

void setup() {
  // Initialize Serial Monitor
  Serial.begin(9600);
  pinMode(2, OUTPUT);

  // Set device as a Wi-Fi Station
  WiFi.mode(WIFI_STA);

  // Init ESP-NOW
  if (esp_now_init() != ESP_OK) {
    Serial.println("Error initializing ESP-NOW");
    return;
  }
```

```
  // Once ESPNow is successfully Init, we will register for recv CB to
  // get recv packer info
  esp_now_register_recv_cb(OnDataRecv);
}

void loop() { }
```

5.2.4.3 Wi-Fi Receiving Sample

The following code implements the client of our WiFi protocol testing. Pin 27 is the button on an ESP32 development kit. Pressing this button transmits a ‘1’ to the receiving server, which causes the LED attached to pin 2 to illuminate. [IZ]

```
// Server Code
#include <esp_now.h>
#include <WiFi.h>

// Receiver MAC Address
uint8_t broadcastAddress[] = {0xE0, 0xE2, 0xE6, 0x70, 0x28, 0x08};

// Structure example to send data
// Must match the receiver structure
typedef struct struct_message {
uint8_t bop;
} struct_message;

// Create a struct_message called myData
struct_message myData;

esp_now_peer_info_t peerInfo;

// callback when data is sent
void OnDataSent(const uint8_t *mac_addr, esp_now_send_status_t status) {
Serial.print("\r\nLast Packet Send Status:\t");
Serial.println(status == ESP_NOW_SEND_SUCCESS ? "Delivery Success" : "Delivery Fail");
}

void setup() {
// Init Serial Monitor
Serial.begin(9600);
pinMode(27, INPUT_PULLUP);
// Set device as a Wi-Fi Station
WiFi.mode(WIFI_STA);
```

```
// Init ESP-NOW
if (esp_now_init() != ESP_OK) {
Serial.println("Error initializing ESP-NOW");
return;
}

// Register CB to get the status of Trasnmitted packet
esp_now_register_send_cb(OnDataSent);

// Register peer
memcpy(peerInfo.peer_addr, broadcastAddress, 6);
peerInfo.channel = 0;
peerInfo.encrypt = false;

// Add peer
if (esp_now_add_peer(&peerInfo) != ESP_OK){
Serial.println("Failed to add peer");
return;
}
}

void loop() {
// Set values to send
myData.bop = 1;
if(digitalRead(27) == 0){
esp_now_send(broadcastAddress, (uint8_t *) &myData, sizeof(myData));
delay(250);
}
}
```

5.2.4.4 NRF Gazell Host Sample

The following code was used to test the Gazell wireless protocol for latency. Gazell is a proprietary protocol created by NRF like Bluetooth low energy. It is more energy efficient than other protocols but sacrifices some speed. The latency for sending and receiving a signal averaged about 6.7mS of latency as seen in Figure 6. This section specifically modeled the base station with outputs being the four LEDs on the devkit. [BT]

```
#include <zephyr/kernel.h>
#include <zephyr/settings/settings.h>
#include <dk_buttons_and_leds.h>
```

```
#include <nrf_gzll.h>
#include <gzll_glue.h>
#include <gzp.h>

/* Pipes 0 and 1 are reserved for GZP pairing and data. See gzp.h. */
#define UNENCRYPTED_DATA_PIPE      2

/* RXPERIOD/2 on LU1 = timeslot period on nRF5x. */
#define NRF_GZLLDE_RXPERIOD_DIV_2  504

static void led_output(uint8_t value)
{
   int err;

   err = dk_set_leds(value & DK_ALL_LEDS_MSK);
}

void main(void)
{
   /* Debug helper variables */
   uint32_t length;

   /* Data and acknowledgment payloads */
   uint8_t payload[NRF_GZLL_CONST_MAX_PAYLOAD_LENGTH];

   int err;
   bool result_value;

   settings_subsys_init();

   err = dk_leds_init();
   if (err) {
      return;
   }
   /* initizlize status leds to show borad is on */
   led_output(DK_LED1_MSK | DK_LED2_MSK);
   /* Initialize Gazell Link Layer glue */
   result_value = gzll_glue_init();
   if (!result_value) {
      return;
   }

   /* Initialize the Gazell Link Layer */
   result_value = nrf_gzll_init(NRF_GZLL_MODE_HOST);
   if (!result_value) {
      return;
```

```
  }

  result_value = nrf_gzll_set_timeslot_period(NRF_GZLLDE_RXPERIOD_DIV_2);
  if (!result_value) {
    return;
  }

  /* Initialize the Gazell Pairing Library */
  gzp_init();

  result_value = nrf_gzll_set_rx_pipes_enabled(nrf_gzll_get_rx_pipes_enabled() |
                  (1 << UNENCRYPTED_DATA_PIPE));
  if (!result_value) {
    return;
  }

  gzp_pairing_enable(true);

  result_value = nrf_gzll_enable();
  if (!result_value) {
    return;
  }

  for (;;) {
    gzp_host_execute();
    /* If a Host ID request received */
    if (gzp_id_req_received()) {
      /* Always grant a request */
      gzp_id_req_grant();
    }
    length = NRF_GZLL_CONST_MAX_PAYLOAD_LENGTH;
    if (nrf_gzll_get_rx_fifo_packet_count(UNENCRYPTED_DATA_PIPE)) {
      if (nrf_gzll_fetch_packet_from_rx_fifo(UNENCRYPTED_DATA_PIPE,
                  payload,
                  &length)) {
        led_output(payload[0]);
      }
    }
  }
}
```

5.2.4.5 NRF Gazell Client Sample

The following code was used to model the wearable as a client using the Gazell wireless protocol. The inputs were the four buttons representing the gyro input which is transmitted to the host device. [BT]

```
#include <zephyr/kernel.h>
#include <zephyr/settings/settings.h>
#include <dk_buttons_and_leds.h>
#include <nrf_gzll.h>
#include <gzll_glue.h>
#include <gzp.h>

/* Pipes 0 and 1 are reserved for GZP pairing and data. See gzp.h. */
#define UNENCRYPTED_DATA_PIPE      2

/* RXPERIOD/2 on LU1 = timeslot period on nRF5x. */
#define NRF_GZLLDE_RXPERIOD_DIV_2  504

/* Ensure that we try all channels before giving up */
#define MAX_TX_ATTEMPTS
(NRF_GZLL_DEFAULT_TIMESLOTS_PER_CHANNEL_WHEN_DEVICE_OUT_OF_SYN
C * \
          NRF_GZLL_DEFAULT_CHANNEL_TABLE_SIZE)

static K_SEM_DEFINE(tx_complete_sem, 0, 1);

static bool gzp_tx_is_success;

static void gzp_tx_result_checker(bool success,
              uint32_t pipe,
              const nrf_gzll_device_tx_info_t *tx_info)
{
   if (pipe == UNENCRYPTED_DATA_PIPE)  {
      gzp_tx_is_success = success;
      k_sem_give(&tx_complete_sem);
   }
}

void main(void)
{
   int err;
   bool result_value;
```

```
    bool tx_success = false;
    enum gzp_id_req_res id_req_status = GZP_ID_RESP_NO_REQUEST;

    /* Data and acknowledgment payloads */
    uint8_t payload[NRF_GZLL_CONST_MAX_PAYLOAD_LENGTH];

    err = dk_buttons_init(NULL);
    if (err) {
        return;
    }

    settings_subsys_init();

    /* Initialize Gazell Link Layer glue */
    result_value = gzll_glue_init();
    if (!result_value) {
        return;
    }

    /* Initialize the Gazell Link Layer */
    result_value = nrf_gzll_init(NRF_GZLL_MODE_DEVICE);
    if (!result_value) {
        return;
    }

    nrf_gzll_set_max_tx_attempts(MAX_TX_ATTEMPTS);

    result_value = nrf_gzll_set_timeslot_period(NRF_GZLLDE_RXPERIOD_DIV_2);
    if (!result_value) {
        return;
    }

    /* Erase pairing data. This sample is intended to demonstrate pairing after every reset. */
    gzp_erase_pairing_data();

    /* Initialize the Gazell Pairing Library */
    gzp_init();

    gzp_tx_result_callback_register(gzp_tx_result_checker);

    result_value = nrf_gzll_enable();
    if (!result_value) {
        return;
    }

    for (;;) {
```

```
        payload[0] = ~dk_get_buttons();

        /* Send packet as plain text. */
        if (nrf_gzll_add_packet_to_tx_fifo(UNENCRYPTED_DATA_PIPE,
                                            payload,
                                            GZP_MAX_FW_PAYLOAD_LENGTH))  {
            k_sem_take(&tx_complete_sem, K_FOREVER);
            tx_success = gzp_tx_is_success;
        }

        /* Check if data transfer failed. */
        if (!tx_success) {

            /* Send "system address request". Needed for sending any
             * user data to the host.
             */
            if (gzp_address_req_send()) {
                /* Send "Host ID request". Needed for sending
                 * encrypted user data to the host.
                 */
                id_req_status = gzp_id_req_send();
            } else {
                /* System address request failed. */
            }
        }

        /* If waiting for the host to grant or reject an ID request. */
        if (id_req_status == GZP_ID_RESP_PENDING)  {
            /* Send a new ID request for fetching response. */
            id_req_status = gzp_id_req_send();
        }
    }
}
```

6.0 Mechanical Sketches

6.1 Wearable Device Sketch

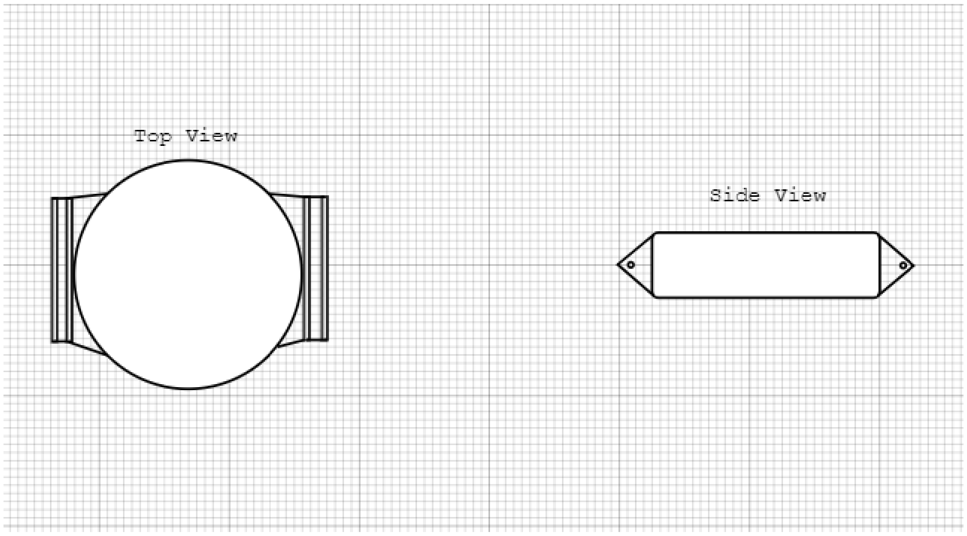

Figure 26: Wearable Device Mechanical Sketch

6.2 Base Station Sketch

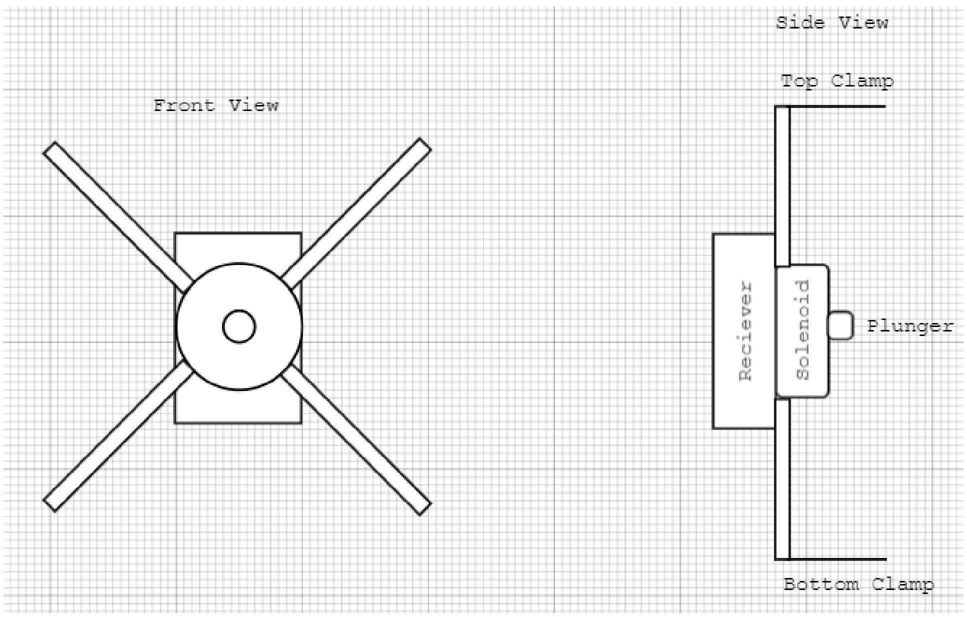

Figure 27: Base Station Mechanical Sketch

7.0 Team Information

I. Project Manager - Jacob Wise, Electrical Engineering. ESI (Y)

II. Engineering Data Manager - Ryan Kinyo, Electrical Engineering. ESI (Y)

III. Hardware Manager - Ian Zanath, Computer Engineering.

IV. Software Manager - Bradley Toth, Computer Engineering.

8.0 Budget Information

Table 26: First Request Budget Information

Qty.	Part Num.	Description	Unit Cost	Total Cost
2	NRF52-DK	Bluetooth LE Development Kit	$39.00	$78.00
1	JF-Z05	24v Solenoid	13.79	13.79
1	FJ-Z05	12v Solenoid	15.99	15.99
1	ADXL345	3Axis Digital Accelerometer(+-16g)	10.90	10.90
			Total	$118.68

Table 27: Second Request Budget Information

Qty.	Part Num.	Description	Unit Cost	Total Cost
5	1N3209	100V, 15A, DO-5 Diode		
1	1N2811B	13V, 50W, 5%, TO-3 Zener Diode		
2	LM747	Dual 741 Op Amp		

Table 28: Final Request Budget Information

Qty.	Part Num.	Description	Unit Cost	Total Cost
5	NRF52833	NRF52 MCU	$4.87	$24.37
100	V475K0402X5R100NCT	4u7 Capacitor, 10V Rated	$0.01	$0.50
100	CC0402JRNPO9BN120	12pf Capacitor	$0.00	$0.10
50	CL05A106MP5NUNC	10uF Capacitor	$0.01	$0.50
100	TCC0402X7R104M160AT	100nF Capacitor	$0.00	$0.08
100	TCC0402C0G101J500AT	100pF Capacitor	$0.00	$0.09
100	0402B821K500NT	820pF Capacitor	$0.00	$0.09
100	CL05A105KP5NNNC	1.0uF Capacitor	$0.00	$0.18
50	V1R0A0402HQC500NBT	1pF Capacitor	$0.01	$0.41
50	V1R2A0402HQC500NBT	1.2pF Capacitor	$0.01	$0.41
20	LQW15AN4N7B00D	4.7nH Inductor	$0.03	$0.69
20	LQG15HS2N2B02D	2.2nH Inductor	$0.04	$0.84
5	X201632MKB4SI	32MHz Resonator	$0.10	$0.51
5	Q13FC1350000400	32.768kHz Resonator	$0.21	$1.06
10	BCT2019EXKV18-TR	1.8v LDO	$0.05	$0.48
5	LIS2DW12TR	IMU	$0.81	$4.06
5	Type C-31-M-12	USB-C Jacks	$0.23	$1.14
5	TP4056X-42-ESOP8	TP4056 Battery Charging Module	$0.20	$1.01
4	NE555S-13	555 Type, Timer/Oscillator (Single) IC 500kHz 8-SO	$0.40	$1.60
15	SM5819L2-TP	Diode Schottky 40 V 1A	$0.28	$4.14
1	WT303012-12F2-ID	6.3µH Wireless Charging Coil Transmitter	$5.94	$5.94
1	WR303050-12F5-ID	8.23µH Wireless Charging Coil Receiver	$5.45	$5.45
2	ASR00012	3.7 V Lithium-Ion Battery	$9.95	$19.90
4	MCU60N04A-TP	N-Channel 40 V 60A	$0.71	$2.84
5	AMS1117-5.0	12V to 5V Linear Regulator 1A		
3	APHHS1005LQBC/D-V	Blue 465nm LED	$0.61	$1.83
3	APHHS1005LSECK/J3-PF	Red 625nm LED	$0.53	$1.59
10	16-219AUTD/S3151/TR8	White LED	$0.32	$3.18
100	CR0402JF0102G	1k resistor	$0.00	$0.04
100	0402WGJ0152TCE	1.5k resistor	$0.00	$0.04
10	FS8205A	Dual NMOS package	$0.06	$0.60
10	DW01A	Battery protection IC	$0.04	$0.37
100	FRC0402J512 TS	5k1 resistor	$0.00	$0.04
5	0402YC103KAT2A	0.01uF Capacitor	$0.10	$0.50
2	TNPU0603500RAZEN00	500 ohm resistor	$2.88	$5.76
2	3214W-1-501E	350 ohm potentiometer	$3.07	$6.14
1	MFS201N-16-Z	Slide Switch	$1.77	$1.77
10	RC0805JR-0710KL	10k Resistor	$0.03	$0.31
2	GRM155C61E475ME15J	4.7uF Capacitors , 25V Rated	$0.25	$0.50
2	CAA572C0G2J204J640LH	200nF Capacitor	$0.25	$0.50
2	C1206C154F5JAC7800	150nF Capacitor	$4.58	$9.16
2	GCM1885C1H202FA16J	2nF Capacitor	$0.26	$0.52
10	TPSR106K006R1500	10uF Tantalum Capacitor	$0.52	$5.16
3	RCA0402100RFKEDHP	100ohm	$0.28	$0.84
			Total	$115.22

- Budget: $600

- First Request: $118.68
- Second Request: $0
- Final Request: $136.04

- Final Balance: $345.28

9.0 Project Schedule (Final Design Gantt Chart)

Project Poster	14 days	Tue 10/11/22	Tue 10/25/22		
Final Design Report	47 days	Tue 10/11/22	Sun 11/27/22	3	
Abstract	47 days	Tue 10/11/22	Sun 11/27/22	3	
Hardware Design: Phase 2	**47 days**	**Tue 10/11/22**	**Sun 11/27/22**	**3**	
Modules 1…n	**47 days**	**Tue 10/11/22**	**Sun 11/27/22**	**3**	
Simulations	45.38 days	Tue 10/11/22	Fri 11/25/22	3	Ryan Kinyo,Jake Wise
Schematics	45.38 days	Tue 10/11/22	Fri 11/25/22	3	Jake Wise,Ryan Kinyo
Software Design: Phase 2	**47 days**	**Tue 10/11/22**	**Sun 11/27/22**		
Modules 1…n	**47 days**	**Tue 10/11/22**	**Sun 11/27/22**		
Code (working subsystems)	45.38 days	Tue 10/11/22	Fri 11/25/22	3	Brad Toth,Ian Zanath
System integration Behavior Models	45.38 days	Tue 10/11/22	Fri 11/25/22	3	Ian Zanath,Brad Toth
Parts Lists	**47 days**	**Tue 10/11/22**	**Sun 11/27/22**		
Parts list(s) for Schematics	47 days	Tue 10/11/22	Sun 11/27/22	3	
Materials Budget list	47 days	Tue 10/11/22	Sun 11/27/22	3	
Proposed Implementation Gantt Chart	47 days	Tue 10/11/22	Sun 11/27/22	3	
Conclusions and Recommendations	47 days	Tue 10/11/22	Sun 11/27/22	3	
Parts Request Form for Subsystems	32 days	Wed 9/21/22	Sun 10/23/22	41	
Subsystems Demonstrations Day 1	0 days	Wed 11/9/22	Wed 11/9/22		
Subsystems Demonstrations Day 2	0 days	Wed 11/16/22	Wed 11/16/22		
Parts Request Form for Spring Semester	0 days	Fri 12/2/22	Fri 12/2/22	44	

Figure 28: Final Design Gantt Chart

10.0 Conclusions and Recommendations

In conclusion, the team will create two separate devices to strike a drum or other percussion instrument. The first device will be an electromechanical actuator which will deliver a quick force to mimic the traditional action of a drummer’s kick pedal. The second device will sense and process user motion, wirelessly telling the first device to actuate when a sufficient movement is detected. This should allow those with impairments or loss of limb(s) to continue to play the drums in full. The team will use a wireless protocol such as Wi-Fi or Bluetooth to communicate between the two devices and will design custom hardware for both. [IZ]

The team has begun realizing the subsystems that will be a part of the finished Wireless Kick Pedal. The team has circuits that can actuate the solenoid and deliver that final kick necessary to strike the drum as well as a path to integrate a charging circuit that will add practicality to the wearable device as well as make it more familiar to a player in the modern age of charging an electronic device. The first software has been written in order to detect the user's intent to deliver a kick as well as the means of communicating that intent between the two major devices. These will further the team's goal to fully integrate these subsystems and realize a usable Wireless Kick Pedal. The team hopes that once the integration of each piece is completed, the final product will prove to be something not just useful and accessible, but something that will change the way that people look at music and the way it is made. [JW]

11.0 References

[1] V. Wai, "Disabled Musicians Face 'Significant Barriers' to Music Education, New Study Shows,"*edm.org*, 17-Oct-2020. [Online]. Available: https://edm.com/industry/uk-study-shows-barriers-for-disabled-people-music-access. [Accessed: 26-Jan-2022].

[2] J. Hall, "Is Technology Finally Making Life More Adaptable for Musicians with Disabilities?" highsnobiety.com, 2020. [Online]. Available: https://www.highsnobiety.com/p/disabled-musicians-technology/. [Accessed: 26-Jan-2022].

[3] Brennan, M. (2020). *Kick It*. Oxford University Press. https://doi.org/10.1093/oso/9780190683863.001.0001

[4] Hampton, R. K. (2005). *Multi-trigger electronic drum pedal* (US Patent No. US6979770B2). U.S. Patent and Trademark Office.

https://patents.google.com/patent/US6979770B2/en?q=electronic+drum+pedal&scholar&oq=electronic+drum+pedal+

[5] Drymonitis, A. (n.d.). *Digital Electronics for Musicians*. Apress. https://doi.org/10.1007/978-1-4842-1583-8

[6] Swartz, R.A.. In: Proceedings of the 2011 American Control Conference American Control Conference (ACC), 2011. :2516-2521 Jun, 2011; IEEE Language: English, Database: IEEE Xplore Digital Library

[7] Papetti, S. and Saitis, C. (eds.), *Musical Haptics*, Springer Series on Touch and Haptic Systems, https://doi.org/10.1007/978-3-319-58316-7_1

[8] Get Real! (2021). *Electronic Musician*, 37(4), 22–26.

[9] Ballard, M. *Multimedia Schools* , Mar/Apr2003, Vol. 10 Issue 2, p48-49, 2p, Database: Education Full Text (H.W. Wilson)

[10] Steele, M. D. (2009). *Electronic drum pedal* (US Patent No. US 7531733B2). U.S. Patent and Trademark Office. https://patents.google.com/patent/US7531733B2/en?oq=US+7%2c531%2c733+B2

[11] Sikra, R. A. (2017). *Drum pedal with adjustment features* (US Patent No. US9589546B2). U.S. Patent and Trademark Office. https://patents.google.com/patent/US9589546B2/en?oq=US+9%2c589%2c546+B2

[12] Salvalaio, C. L. et al. (2011) *Science and Technology*, 1(1), 1-6.