ParaPedal:
A Practical Product for Paraplegic Piano Players!

Engineers (Group 15):

Andre Llanos
Electrical Engineering

Benedict Eberhagen
Electrical Engineering

Jason Knaebel
Electrical Engineering (Manager)

Rafael Molina
Electrical Engineering

Coordinator:
Dr. Lei Wei

Reviewers:
Dr. Wei Sun
Dr. Saleem Sahawneh
Dr. Mark Maddox

Chapter 1: Executive Summary

Chapter 2: Project Description

Motivation

The idea of building a piano assistance device for people in wheelchairs was inspired by one of our group members, Benedict Eberhagen. He learned how to play the piano at a young age, however he stopped for a while. More recently, he has been picking it back up. However, he is now playing with one major change - he is currently in a wheelchair. His progress working on and improving his piano playing has been going great. However, since he is now in a wheelchair and incapable of using his legs, he is unable to play most of his piano pieces properly, he is unable to activate the piano pedals at the bottom of the piano. Although the usage of pedals isn't necessary to play a piece successfully, it does enhance the quality and sound of the piano piece. With that, our motivation is to help Benedict play the piano properly once more, as well as helping other people with physical disabilities - paraplegics in particular - to be able to activate the foot pedal in an easy and intuitive way. Our product will help our target customers reach the full potential and range of sounds that a piano has to offer. Benedict is an intermediate-level piano player with a decent background in understanding how to play, but he is not at the professional level. Our build will be focused around that factor as well. We will design our device to be more beginner-level and intermediate-level friendly, instead of making it overly complicated and designed for professional-level play.

Background

Our team consists of 4 Electrical Engineers, so our main focus on this project will be on the hardware of the remote-activated piano pedals, instead of the software. Two of our members, Andre and Jason, already have a strong background in designing PCBs having worked with them in their jobs and during research. Rafael has worked quite a bit with power distribution and overarching electrical designs during his internships. Benedict has a good amount of experience on the software side of designing devices, which he worked on during his internships.

Overall Goals

- Basic:
    - A user with no use of their legs should be able to operate a single piano pedal (the sustainer pedal) in an on and off fashion while playing the keyboard.
        - The user will press a button and the device will control an actuator to apply pressure to the piano pedal.
    - The device's operation will be intuitive.
        - A simple UI for the calibration app, combined with minimal components and quick response times from the system will accomplish this goal.

- The solution will have a low latency between a user's request and the actuation of the pedal.
  - Sufficient processor clock rates, wireless-communication response times, and motor movement speed will accomplish this goal.
- The setup process will allow for no permanent modifications to the existing piano.
  - The housing for the finished product will be simple to mount and dismount from most traditional pianos on the market.

- Advanced:
  - A user with no use of their legs should be able to operate three piano pedals, each with varying amounts of force depending on the user's control input.
    - A mouthpiece with 3 resistive pressure sensors for the user to press will accomplish this goal by allowing the user to activate all three pedals independently from one another.
  - The pedal actuator(s) will operate quietly, generally quieter than the piano itself.
    - If necessary, this will be implemented by selecting a noise-insulated material to house the motors in, as long as size and weight constraints stay met.
  - The device will calibrate to the user's specific input pressure(s) using an interface operated through a smartphone.
    - This will be implemented by developing an app interface that can prompt the user to individually press each sensor both gently and forcefully to set a range of pressures and correlate it to each pedal's range from "un-pressed" to "pressed".

- Stretch:
  - A sensor will be implemented that actuates the pedals through electrode-stimulation, without the use of physical force from the user.
    - This goal can be accomplished by having 3 electrodes that can sense when the user activates an attached muscle, and for how long it is activated, to activate the pedal respective to the electrode.
  - We design our own motor driver board.
    - An H-bridge circuit will be developed and implemented for greater force/speed control for pedal actuation.
  - An external haptic driver based on pedal position will be implemented for user feedback so they can feel the applied force more intuitively.
    - A haptic driver will be controlled based on a force feedback on the pedal.The user will feel the haptic response to know how much pressure is applied.
  - An alternative input system will be available for the user that operates using three tubes that the user blows into to activate the pedals, in lieu of the bite/tongue activated input system.

- The user will blow into three different tubes at different times, this creates a pressure differential that would be sensed with a pressure sensor and the value processed by a microcontroller to communicate the appropriate control signals.

Overall Functionality

The overall function of the project is to provide a person without use of their legs the ability to play the foot pedals of a Piano. The solution will be intuitive, comfortable to the player and have an easy setup process.

User input Functionality

The user will command the pedal to be pressed by using a lightweight and low profile 'bite sensor', where the pressure on the device is correlated to the intended pressure on the piano pedal. This signal communicates with a device near the foot pedal. The device then interprets the command and controls an actuator to move the piano pedal.

Actuator Functionality

The actuator must function to replicate a typical foot press and so should have the required force, speed and accuracy that is expected. The actuator will be low profile and have a small footprint so that it does not interfere with wheelchair accessibility.

Control Unit Functionality

The control unit will process the commands from the user and send the required drive signals to the actuator. The control unit will have a display that tells the user the operation mode and have a selector knob for mode selection. The control unit will be input sensor agnostic so that multiple methods of user commands may be used. The control unit will also have power handling capability through USB-C with an integrated battery system for mobile operation.

Existing products

There are already a couple of devices on the market that allow paraplegics to activate the piano pedals. None of them are mainstream/mass-produced; instead, they have been designed by relatively small teams, often for research purposes at different Colleges. The University of Victoria, the University of Heidelberg, as well as Resonate, a small company, are some of the few places that have designed and built a paraplegic piano assistance device. All of them were designed a bit differently, especially in regards to controls and how to smoothly activate the pedals. It's a bit hard to see how exactly each system was designed. However, all of them seem to have the core design of activating the pedals through the use of actuators. The key difference between each design is how they choose to activate the pedals. The University of Heidelberg has come up with 2 separate ways to activate the piano pedals. The first way is using Head movements, which is called the Bernd Schwankl Switch three tilt switches. Turning left activates the left pedal, turning right activates both pedals, and nodding activates the right pedal. The system can be seen in the two pictures below.

(4)

The second method that the University of Heidelberg designed is based on chewing. A teeth guard is inserted into the mouth where the user can chew on it. This will activate the actuators at different levels of pressure on the pedals depending on the amount of bite force.

(4)

The University of Victoria designed a similar system to the University of Heidelberg's first design through Canassist. Their design is also head-activated and uses a headband to detect the user's movements. This is done through a wireless sensor which measures the changes in its own position. These changes will promptly be communicated to the piano pedal device below to activate the wanted pedals for the user's desired duration.

Resonate, a private company that decided to build a paraplegic piano assistance device, used a similar method to the University of Heidelberg's second design. Resonate uses a mouth splinter to activate the foot pedals. The system is designed very simply - when the controller is bit on, it will activate the pedal. Resonate’s design is focused on simplicity and portability, although it's a bit hard to tell how exactly the design was made, as they weren't very specific about it’s design. It seems like their system can only activate one piano pedal. An example can be seen below.

(4)

Engineering Requirements

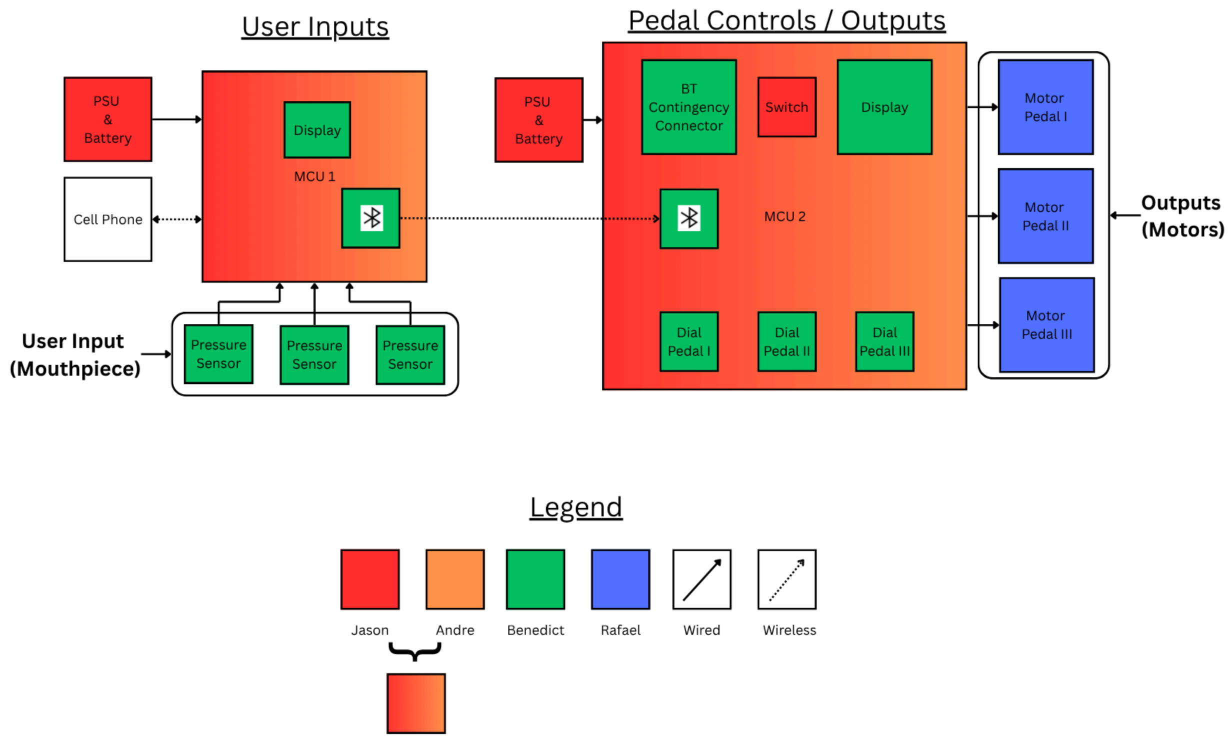
Subsystem Engineering Specifications:

- Motor Subsystem:
  - Contains motors that move the piano pedals based on user input.
  - Converts electrical control signals into mechanical motion.
- Control Subsystem:
  - Central microcontroller that interprets user inputs.
  - Manages control, feedback, and communication.
- Power Subsystem:
  - Provides regulated voltage to the system from a rechargeable power source.
  - Designed for at least 1 hour of continuous operation.
- User Input Subsystem:
  - Includes switches, dials (potentiometers), and buttons.
  - Allows the user to control pedal movement precisely and comfortably.
- Safety & Feedback Subsystem:
  - Monitors for abnormal operation or low pressure from sensors.
  - Includes emergency stop capabilities.
  - Communication Subsystem.
  - Provides Bluetooth connectivity for wireless setup or remote control.
- Specification Table:
  - Please refer to the next page for a table of engineering specifications and their quantitative measures.

Specification Table	
Motor System	
Motor Response Time	500-700 Milliseconds
Maximum Power Consumption	≤ 20 Watts
User Input (Button/Dial)	
Press Detection Threshold	5-7 Newtons
Button Recognition Time	≤ 500 Miliseconds
Control Unit	
Response Time	< 1 second
Voltage Input	3.3-5 Volts
Maximum Current Draw	< 500 MiliAmps
System Overall	
Device Weight	5-10 Kilograms
Battery Runtime	≤ 1 Hour
Communication Range	≥ 4 meters
Total Delay	≤ 1Second
Response Time	≤2 Second
Accuracy	75-85%
System Setup Time	≤ 2 Minutes

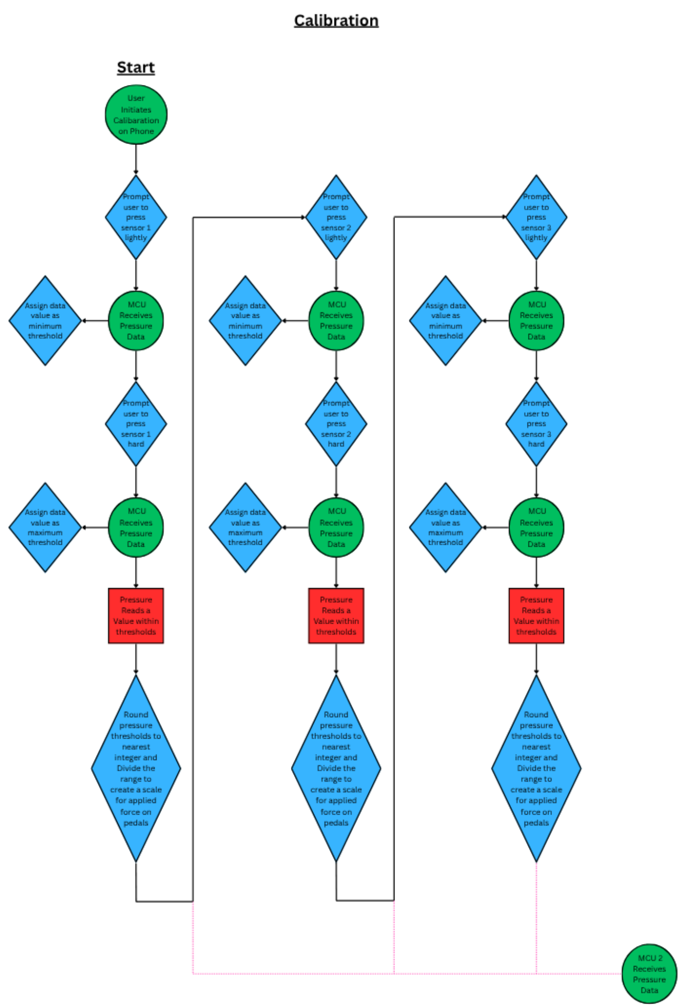
Hardware Connections

The electrical hardware for this product consists of 2 main boards, with their respective peripherals. The first main board is the User Input board. This board will be capable of calibrating to the user's individual pressure inputs through an external app on a smartphone, then receiving the user's inputs to the pressure sensors and transmitting the information via bluetooth to the second board for motor control. The second board, the Pedal Control/Output Board, will interpret the user's input transmissions from the User Input Board in order to control the motors (each motor respective to its associated pedal) to depress its associated pedal for the duration and intensity of the user's input. The boards will have variable settings that will be displayed and adjusted through use of a switch. Both boards will be powered by a rechargeable battery connected to a usb for charging. Please reference the Hardware connection diagram shown below.

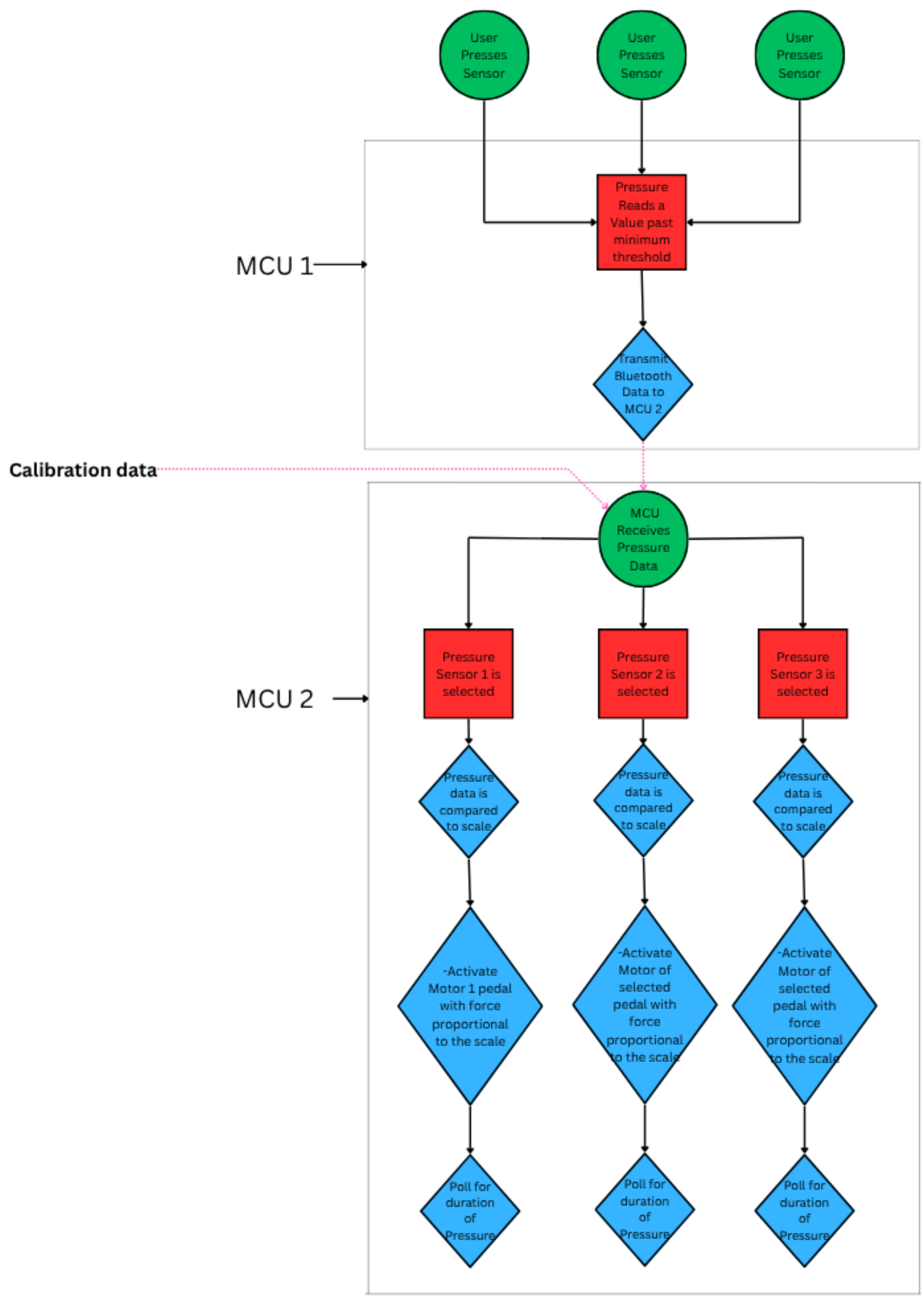


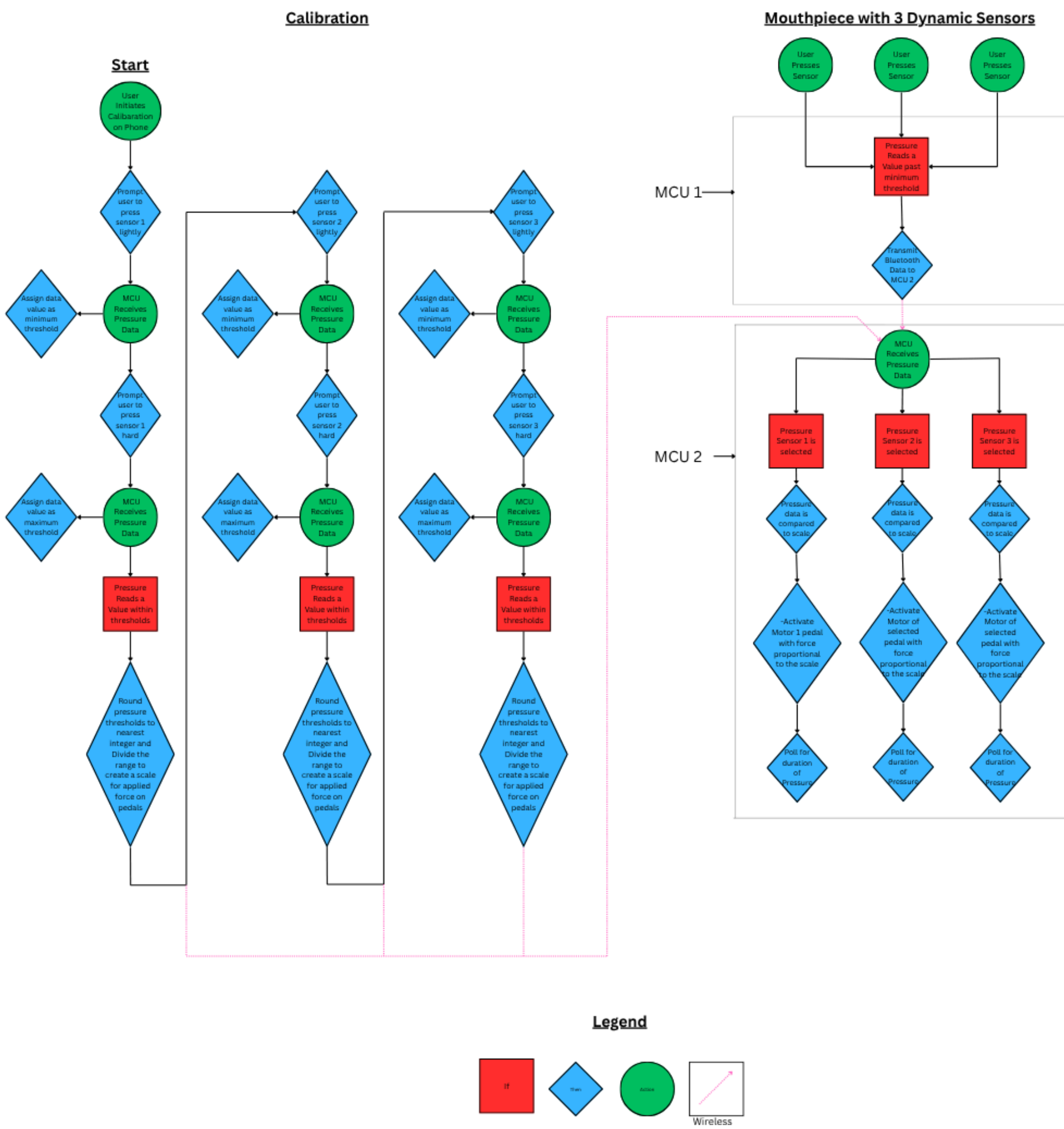
Software Summary and Pseudocode

In terms of software, the MCU will be capable of processing all required information in an efficient and effective manner. The User Input Board requires the Parapedal to be able to calibrate the pressure sensors within the mouthpiece to the user's specific pressure range. In order to do this, the code must be able to prompt the user to input their weakest and strongest preferred pressures in order to create a pressure gradient that the Motor Control/Output board can then use to determine how the motors will behave when actuating the piano pedals. Below are the individual pseudocodes for both processes, followed by an image of both pseudocodes in parallel.



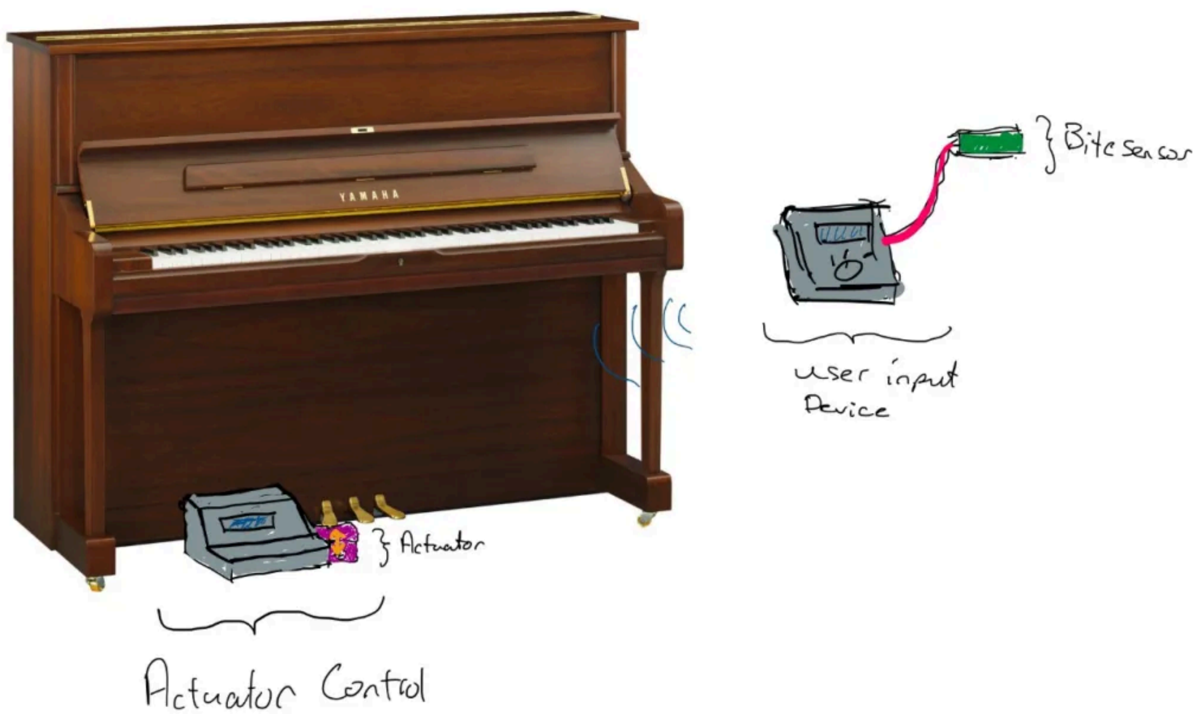
Mouthpiece with 3 Dynamic Sensors




Calibration
Start
User Initiates Calibaration on Phone
Prompt user to press sensor 1 lightly
MCU Receives Pressure Data
Assign data value as minimum threshold
Prompt user to press sensor 1 hard
MCU Receives Pressure Data
Assign data value as maximum threshold
Pressure Reads a Value within thresholds
Round pressure thresholds to nearest integer and Divide the range to create a scale for applied force on pedals
Prompt user to press sensor 2 lightly
MCU Receives Pressure Data
Assign data value as minimum threshold
Prompt user to press sensor 2 hard
MCU Receives Pressure Data
Assign data value as maximum threshold
Pressure Reads a Value within thresholds
Round pressure thresholds to nearest integer and Divide the range to create a scale for applied force on pedals
Prompt user to press sensor 3 lightly
MCU Receives Pressure Data
Assign data value as minimum threshold
Prompt user to press sensor 3 hard
MCU Receives Pressure Data
Assign data value as maximum threshold
Pressure Reads a Value within thresholds
Round pressure thresholds to nearest integer and Divide the range to create a scale for applied force on pedals
Mouthpiece with 3 Dynamic Sensors
User Presses Sensor
User Presses Sensor
User Presses Sensor
MCU 1
Pressure Reads a Value past minimum threshold
Transmit Bluetooth Data to MCU 2
MCU 2
MCU Receives Pressure Data
Pressure Sensor 1 is selected
Pressure Sensor 2 is selected
Pressure Sensor 3 is selected
Pressure data is compared to scale
Pressure data is compared to scale
Pressure data is compared to scale
-Activate Motor 1 pedal with force proportional to the scale
-Activate Motor of selected pedal with force proportional to the scale
-Activate Motor of selected pedal with force proportional to the scale
Poll for duration of Pressure
Poll for duration of Pressure
Poll for duration of Pressure
Legend
If
Wireless

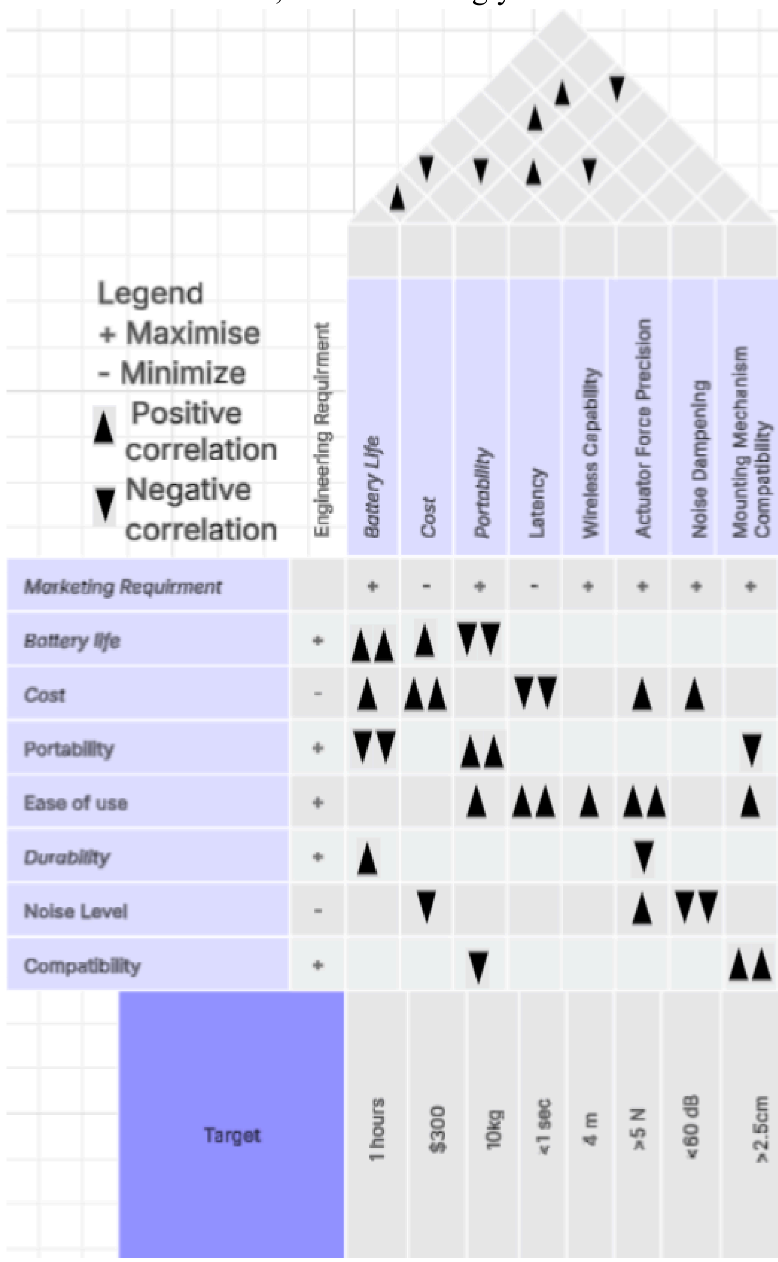
Sketch of Design

Below is our initial concept sketch for the Parapedal. While the final product may vary, it is an apt depiction of how the two boards will be separate and communicate with each other wirelessly via bluetooth.



House of Quality

This chart below shows how different parts of our design are connected to what users actually care about. It maps marketing goals like keeping costs low, making the device easy to carry, and ensuring it's simple to use. Also it shows the technical features we need to focus on, such as battery life, latency, noise level, and precision. The symbols help us understand whether we want to increase or decrease a certain feature, and how strongly it affects the user's experience.



Chapter 3: Research and Investigation

To select the components, we conducted research of applicable technologies and available devices for integration into our system. The system was split into six distinct subsystems with clearly defined functional requirements: the user intention sensing system, the sensor holder mouthguard, the microcontroller/communication module, power regulation, motor actuation , and display outputs . The research phase generated multiple options across all categories, which was then narrowed through a filtering process considering specifications, cost constraints, and integration complexity to arrive at a list of promising candidates.These candidates were then compared against each other and the design specifications to arrive at a singular option. The final component selection represents the optimal balance of individual subsystem performance and system-level integration feasibility.

User Input & Motor Output Batteries

Longevity and comfort are valuable characteristics for the user of this product. Whether it be piano practice or piano performance, the user must be capable of playing in comfort for a reasonable amount of time . With the user input device being a mouthpiece, the attached control unit must not put much, if any, weight on the mouth. That being said, the battery supply for the mouthpiece control unit must not be too heavy, but must still provide enough power to last the user for as long as needed. We have selected the following components as potential candidates to power the user input PCB.

For the Motor output, the power will be supplied from a battery pack in conjunction with a USB. With the battery being on the ground with the control unit, size and weight are not a significant factor here. The table below shows the parts selected for batteries and their associated packs during research.

- Requirements:
    - Not too bulky
    - Comfortable if mounted to user's clothing / placed in user's lap
    - Light
    - Enough power for a moderately long session

Device	18650 Battery Case Holder, 2 Slots	BAT-HLD-011-THM Lithium Coin Cell Holder	TEAYIR 3 AA Battery Case Box Holder 4.5-5V with Cover and On/Off Switch, with USB Cabl	DAIERTEK 4 AA Battery Holder with Switch 6V	Ecto® 5,000mah Battery
Price	$9.99 for 2	$12.90 for 10	$5.80 for 2	$8.39 for 3	$24.99
Dimensions	Awaiting Arrival	19mm - 23mm diameter	2.25in x 2.52in x 0.71in	68.7mm x 64.2mm x 2.5mm	2.6 inches x 3.9 inches x 0.7 inches
Weight	Awaiting Arrival	Negligible while empty, appx. 5.5g full	0.07 Pounds empty, 0.23 Pounds full	0.19 Pounds empty, 0.41 Pounds full	0.3 Pounds
Power Specs	3.7V - 6.4V	3.7V	4.5 ~ 5V, 0.2A	6V	5V, 2A
Pros	Enough power for long term use	lightweight, compact, enough power supplied for the input to be easily stepped down without consuming too much energy	Supplies enough power for input board	Supplies enough power for output board	Rechargeable, supplies enough power for both boards, bulky, expensive
Cons	None	None	Not directly rechargeable, bulky	Not directly rechargeable	
Decision	Final for Output	Final for Input	2nd choice	3rd choice	last choice

[1] BAT-HLD-011-THM Lithium Coin Cell Holder for LIR2450 Rechargeable Battery

Description:

This battery pack is specifically for the mouthpiece control unit. The battery will provide 3.7V, which will be stepped down to 3.3V for the pcb's required power. After comparing this battery pack to the other battery packs, this is the most ideal for the mouthpiece User Input Board as it is the smallest, lightest, and closest to the required power, which means less stepping down is required to meet power and heat constraints. It is cheap and extremely light, making it cost effective and also a comfortable option for the user. The lithium coin cell batteries can also be easily rechargeable. Due to these reasons, we have selected this battery pack as our final battery supply for the User Input Board.

[2] 18650 Battery Case Holder, 2 Slots

Description:

This battery pack holds two 18650 batteries. This battery pack was selected for the Motor Control/Output board due to its ability to provide up to 6.4V, which can be stepped down and regulated to the 5V, 3A required for the servo motors as well as the 3.3V required for the ESP32 and peripheral functions. This battery pack can also be easily configured to be rechargeable with peripheral devices. These reasons led us to conclude that this battery pack will be our final option for the Motor Control/Output Board.

[3] TEAYIR 3 AA Battery Case Box Holder 4.5-5V with Cover and On/Off Switch, with USB Cable

Description:

This battery pack is lightweight, compact, and affordable. It is cheap enough to easily replace, and it has a USB output rather than a fixed connector with a singular connector. These properties will allow us to adjust our PCB without adjusting the USB power supply, and allow both us and the user to easily replace faulty equipment. Having only 3 AA batteries, the input voltage will be easy to step down if necessary for the low-power systems we will be implementing. While it is not rechargeable (which would be best for the user), these reasons make this battery pack a great candidate for prototyping and testing. However, as the final choices for both boards' batteries were the 18650 battery and the Lithium coin cell, we did not choose this battery pack in our final search.

[4] DAIERTEK 4 AA Battery Holder with Switch 6V Battery Case Holder with Cover 4 Slots 1.5V AA Battery Storage Box with Wires Battery Connector

Description:

This battery pack having 4 AA batteries will allow it to last longer with a low current regulator. It is around ten cents cheaper than the TEAYIR 3AA battery, which keeps it within a very affordable price range. However, having 4 AA batteries makes it not only bulkier, but heavier, which will make it more difficult to mitigate user discomfort with the system it is supplying power to being in the user's mouth. This battery pack also has direct wiring, which can be worked with the design and testing process. Comparing the disadvantages of this battery pack to the aforementioned battery pack, this choice falls short in being selected for prototyping.

[5] Ecto® 5,000mah Battery

Description:

This meets nearly all of the requirements we are looking for. With a slender build and being only an inch longer than the TEAYIR 3 AA battery pack, while being only slightly heavier, it meets size and weight requirements for the user's comfort. Also comparable to the TEAYIR 3 AA battery pack, Ecto has usb connectors for its output. This will allow us to adjust our PCB without adjusting the power supply, and allow both us and the user to easily replace faulty equipment. Also, to the user's benefit, the Ecto is USB rechargeable, water resistant, has a battery level display, and overload / overcharge protection; so the user will have greater reassurance in our product's reliability. The only downside is the price per unit, so this may or may not end up being selected for the final product's user input battery supply.

Power Regulation

All components must receive their specific power constraints. With the board receiving a singular voltage, the power to each component must be stepped up or down in accordance with each individual component's power constraints. With the power requirements for the servo motors being 5V and 3A, power regulation will be necessary. The table below shows the parts selected for power regulation and monitoring.

- Requirements:
  - Must meet the 3.3V and 5V 3A power constraints for each board, respectively.

Device	LP2966IMM-1833/ NOPB	MP2338GTL-Z	XC6204B502MR-G
Price	$15.40 for 10	$11.30 for 10	4.06
Voltage - Input (Max)	7V	28V	10V
Voltage - Output	1.8V, 3.3V (Fixed)	0.5V - 16V	5V
Current Output: 150mA	150mA	3A	150mA
Pros	Drops any 1V -7V power supply down to 3.3V	>90% efficiency adjustable output voltage and current	Supplies the 5V for the servo motors
Cons	No 5V output	None	Does not suplly the require 3A for the servo motors
Decision	Final for User Input	Final for Motor Control/Output	Will not be used

[6] LP2966IMM-1833/NOPB

Description:

This Linear Low DropOut (LDO) regulator is capable of dropping all voltages ranging from 1V to 7V down to 3.3V, which will be capable of power regulation for all of our selected chips from both the wall supply and our selected battery options. However, with the power only stepping to a fixed 3.3V, the motors cannot run with this regulator, as our selected batteries supply more than 5V.

[8] MP2338GTL-Z

Description:

This regulator is specifically to drop the power from the Motor Control/Output Board's battery to control the motors. With a low price to buy in bulk, a 3A current max, and the maximum input voltage range being 28V and the output voltage range being from 0.5V to 16V, this switching regulator will be a versatile option for prototyping for both the user input and the motor output boards. Also, being a switching regulator, the MP2338GTL-Z has an efficiency greater than 90%, which will suffice as a replacement for the user-input regulator, the LP2966IMM-1833/NOPB LDO, whose efficiency varies with the power input.

[7] XC6204B502MR-G

Description:

This regulator is specifically to drop the power from the motor control PCB's battery to control the motors. The maximum voltage is set for 10V to allow for flexibility in case we select other batteries not yet listed on this report. While it supplies the motor's 5V requirement to run, it does not meet the 3A tolerance required for the motors to run.

Battery Recharging and Power Monitoring

Device	BQ25887 I2C 2 cell 2A Boost battery charger for USB input and integrated cell balancing	TPB4056A20-ES1R LINEAR BATTERY CHARGER	INA219AIDCNR Current Monitor
Price	$38.11 for 10	$5.97 for 10	$16.69 for 10
Number of Cells	2	1	NA
Pros	Easily implemented Integrated USB Works with both battery selections	Easily implemented Cheap	Useful for guaging battery level
Cons	Expensive	No USB integrated	Not a reasonable option when other recharging chips exist
Decision	Final for both boards	2nd recharging option	Not used

[9] BQ25887 I2C 2 cell 2A Boost battery charger for USB input and integrated cell balancing

Description:

This battery charger works best for interfacing our usb connection from the wall outlet power supply with the battery pack, which will be the primary source of power for both boards. This will allow the user to replace the rechargeable batteries if necessary, and also provide power to the servo motors. The integrated USB and intuitive interface lead to this board being our final decision.

[10] TPB4056A20-ES1R LINEAR BATTERY CHARGER

Description:

This battery charger is capable of charging both boards' battery packs both while the product is being used and when power is not being consumed. The TPB4056A20 is affordable, has numerous fault protection features, and meets our power requirements. However, due to the BQ25887's superior usb interface and useability, this is our second choice for recharging.

[11] INA219AIDCNR Current Monitor

Description:

This item may not be used in the prototyping process, but may be used in the final process to allow the user to gauge the battery level. This item was originally intended to be used to read the current level for non-rechargeable batteries, but ultimately we decided against that method.

Mouthpieces

One of the important components we need for our Parapedal is a proper control system. To activate the pedals in a fast and intuitive way, we decided to use a mouthpiece as our control system. This device will be inserted into the user's mouth, where the user can apply pressure with their tongue on a pressure sensor. The amount of force the user exerts on the pressure sensor will determine the amount of force exerted on the pedals. In order for the control system to be easy to use for the user, we need a way to keep the pressure sensors together in a compact controller that can be inserted into the mouth. Because of that, we decided on using a mouth guard as the base where the pressure sensors will be inserted. The mouthpiece will help give the controller a more intuitive feel as it should smoothly fit into any user's mouth. In order to determine which mouthpiece is the best fit, we decided to give the mouthpiece some requirements to determine which one more closely fits our optimal mouthpiece.

- Requirements:
  - Fits the pressure button
  - Easy wiring
  - Comfortable
  - Not noticeable
  - Light
  - Easy fit for the user

Mouthguard

Products	BBTO Sports Guard	GOYO Night Guard	KOHEEL Night Guard	SafeJawz
Dimensions	L: 2.56 inches W: 2.17 inches H: 0.59 inches	L: 1.3 inches W: 1.9 inches Biting W: 0.4 inches	L: 2.18 inches W: Adjustable Biting W: 1.33 inches	L: 2.28 inches W: 1.93 inches H: 0.79 inches
Price	9	17	17	16
Quantity	20	2	12	1
Noticeability	It can be clearly seen when smiling	Hidden away even when smiling	Hidden away even when smiling	It can be clearly seen when smiling
Weight	~3 ounces	~1 ounces	~1 ounces	~3 ounces
Comfort	Easy fit can be adjusted to user's mouth	Easy fit can be adjusted to user's mouth	Easy fit but can slide around a bit	Easy fit can be adjusted to user's mouth
Wiring	Fits wiring on the inside or outside the mouthpiece	No space to attach wiring	No space to attach wiring	Fits wiring on the inside or outside the mouthpiece

BBTO 20 Pieces Sports Mouth Guard Boxing Mouth Protection Athletic Mouthpiece for Football Basketball Hockey Karate Rugby Match

Description:

We decided to choose this mouthpiece as our first controller. We came to this conclusion as the BBTO mouthpiece seems to be the perfect choice for a prototype controller. It's cheap and it can be bought in bulk. This allows us to mess around on how to insert the pressure sensors while not being worried about destroying the mouthpiece, since we have enough backups. This allows us more liberty in terms of creativity to test the feasibility of our ideas. In terms of design, it's also in the middle of our size categories, not being too small nor too large. Based on how easy we can install the wiring and how it fits into the mouth, we can then decide whether we want to go for a large or smaller mouthpiece in our final design. The other mouthpieces are significantly more expensive, so we don't want to be too experimental on those mouthpieces and possibly risk damaging them beyond repair. Once we've decided on a proper controller setup and confirmed its feasibility, we are planning on switching to a better mouthpiece.

[6] GOYO Mouth Guard for Grinding and Clenching Teeth at Night, Sleep Night Guard, Grinding Protection, Clenching Protection, Teeth Bruxism Mouthpiece with Travel Hygiene Case

Description:

This is our optimal mouthpiece for the final project. It's small, sleek, and should universally fit everyone. The main challenge with this mouthpiece is its size. Since the Goyo mouth guard is so small, which is what makes it so attractive, it could also provide us with significant challenges in fitting everything we want on it. Although our pressure sensors are pretty small and should fit into the mouthpiece, there is still the wiring that we have to worry about. Depending on how well we can fit the sensors and wiring on our prototype will determine if we will end up using this mouthpiece or a larger one. The current idea for this mouthpiece would be to use something like a scalpel to cut a hole into the mouthpiece where we can insert the sensor. This will be pretty challenging to accomplish, but should be possible. Another challenge with this mouthpiece is that its main function is at the bottom of the mouth, not the top. Our goal is to have our sensors at the top of the mouth for easier control. This can be circumvented however, by either testing the controls out at the bottom or simply inverting the mouthpiece. Considering how flexible the mouthpiece is, that shouldn't be much of a challenge. Although it's hard to say whether or not we are able to use it as our mouthpiece effectively, the prototype design will help us in our understanding. If it does work, this mouthpiece would be the optimal solution.

[7]

KOHEEL Adjustable Mouth Guard for Teeth Grinding and Clenching, Comfort-Fit Night Guard, One Size Fits All, Stop Bruxism Dental Guard

Description:

The KOHEEL mouthpiece is an alternative to the Goyo mouthpiece. It is very similar in design and would function in a similar way with basically the same setup as the Goyo. Our goal would be to cut a space into the mouthpiece that can fit the pressure sensor in; however, it's facing similar issues to the Goyo. Finding space for wiring and making sure that the sensors fit properly will be the biggest challenge. Compared to the Goyo, it's also slightly smaller, making it a bit more difficult to fit the pressure sensor. Unlike the Goyo mouthpiece, the KOHEEL mouthpiece is designed for the upper part of the mouth, not the lower. The KOHEEL is, however, designed with it being on the outside of the mouth, not the inside. This will make it a bit challenging to fit the pressure sensor properly; however, similarly to the Goyo, it is very flexible, which means simply inverting it shouldn't be too difficult. Due to this challenge and in the reduction of space, it is our second alternative for a smaller mouthpiece design after the Goyo mouthpiece.

[8]

SafeJawz Sports Mouthguard, Dual Layer Premium Mouthguard with Case

Description:

In terms of size, the SafeJawz is a middle-ground between smaller Goyo and KOHEEL mouthpieces and the larger mouthpieces like Difiney and Honex. The SafeJawz mouthpiece is medium in size and attaches to the top jaw. It's not as invisible as the Goyo and KOHEEL. However, while the mouth is closed, it is a lot harder to see, unlike the Difiney and Honex. It is very similar in design to BBTO, where it's large enough to fit the pressure sensors comfortably. If the prototype BBTO is the perfect size for our Parapedal, then we should transition over to the SafeJawz for our final model, as it is built for a lot more comfort while not sacrificing any of the functionality of the BBTO.

[9]

Anti-Snoring Device

Products	Difiney Anti-Snore	Honex Anti-Snore
Dimensions	L: 42 mm W: 61mm Diagonal: 30mm	L: 42 mm W: 61mm Diagonal: 30mm
Price	40	14

Quantity	1	1
Noticeability	Leaves the mouth open which makes it visible even if the mouth is closed	Leaves the mouth open which makes it visible even if the mouth is closed
Weight	~5 ounces	~5 ounces
Comfort	Easy Fit, large size that cant slide around	Easy Fit, large size that cant slide around
Wiring	Perfect gap for wiring between upper and lower mouthpiece	Perfect gap for wiring between upper and lower mouthpiece

Difiney Anti Snoring Device: FDA-Cleared Anti Snoring Mouthpiece Mouth Guard - Effective Anti Snore Solution for Men and Women to Reclaim Peaceful Sleep

Description:

The Difiney is on the other side of the spectrum from the Goyo and KOHEEL design. If in our prototype design, we come to the conclusion that proper wiring is particularly difficult and more space is necessary, then we will need to go with an anti-snoring device. These are much larger and bulkier than the Goyo and KOHEEL counterparts. The Difiney anti-snoring mouthpiece extends to both the top and bottom teeth and almost completely covers them. Due to it being an anti-snoring device, it also has a gap between the top and bottom rows of teeth. This gap is the perfect place for us to insert wiring, as it can be perfectly stored away. There is also a lot more room in general to fit the pressure sensor onto the mouthpiece. Unlike the Goyo and KOHEEL mouthpiece, it is, however, a lot more rigid and not as flexible. This stiffness can make the installation of pressure sensors a lot harder, at least if we want it to be inserted inside the mouthpiece, not on the outside. The biggest issue with the Difiney anti-snoring device, as our mouthpiece, is mainly visual. Although initially this might not be considered a major issue, we do want our device to be used as much as possible. Since our Parapedal is built for the artistic and performance community, visual appeal is very important and closely behind the devices practicality. Because of that, if at all possible, we want our device to be as small, sleek, and unnoticeable as possible, which the Difiney and almost all anti-snoring devices are not as good in. Therefore, our goal is to go for a smaller mouthpiece like the Goyo, KOHEEL, or BBTO. As we build our prototype, we will be able to judge how much space we truly need for our mouthpiece. Therefore, if our prototype BBTO mouthpiece is simply not big enough to house everything, we will transfer our design over to this mouthpiece, as it is by far the most practical design. With a lot of space as well as a perfect gap that can tuck away wiring safely without interacting with the user.

[10]

Honex Mouthpiece Universal Size

Description:

An alternative to the Difiney anti-snoring mouthpiece is the Honex Mouthpiece. This is also an anti-snoring device with a similar design. As mentioned before, these larger anti-snoring devices are alternatives to the smaller mouthpieces like the Goyo, KOHEEL, and BBTO, in case more space is required to fit our pressure sensors as well as wiring. The Honex mouthpiece has many advantages over the Difiney anti-snoring device. First of the Honex mouthpiece is smaller in size, making it less visibly disruptive and sleeker in design. Even with its smaller size, the Honex mouthpiece still has plenty of space to tuck away wires in the gap between the top and bottom mouthpieces. It is also cheaper in price and, based on the material, seems to be a bit more flexible as well. The rest of the design is still very similar, with it still being a pretty solid, inflexible mouthpiece, being attached to both the top and bottom jaw, as well as being pretty large and noticeable. Depending on how much more space will be necessary for our pressure sensor, the Honex mouthpiece is a great, smaller alternative to the Difiney anti-snoring mouthpiece.

[11]

Summary

Mouthpiece	BBTO Sports Guard	SafeJawz	GOYO Night Guard	KOHEEL Night Guard	Honex Anti-Snore	Difiney Anti-Snore
Cost	9	16	17	17	14	40
Amount	20	1	2	12	1	1

Noticeability	It can be clearly seen when smiling	It can be clearly seen when smiling	Hidden away even when smiling	Hidden away even when smiling	Leaves the mouth open which makes it visible even if the mouth is closed	Leaves the mouth open which makes it visible even if the mouth is closed
Wiring	Fits wiring on the inside or outside the mouthpiece	Fits wiring on the inside or outside the mouthpiece	No space to attach wiring	No space to attach wiring	Perfect gap for wiring between upper and lower mouthpiece	Perfect gap for wiring between upper and lower mouthpiece
Comfort	Easy fit can be adjusted to user's mouth	Easy fit can be adjusted to user's mouth	Easy fit can be adjusted to user's mouth	Easy fit but can slide around a bit	Easy Fit, large size that cant slide around	Easy Fit, large size that cant slide around
Conclusion	Prototype	Potential Final design	Optimal Final design	Backup	Potential Final design	Unnecessary
Reason	Large cheap quantity and middle of the road for size. The key issue is that we want to go for limited noticeability so depending on the prototype we	Better quality than BBTO and if the BBTO is the optimal size for the mouthpiece this will be our go to for the final design.	Amazing in terms of limited visibility which if the wiring needs less space this is the optimal mouthpiece.	Backup for the Goyo if its not a good fit	If our Prototype needs better wire management this anti snoring device will be the go to option for our final design.	Too expensive while providing no real benefit over the Honex

	can make our finial decision on our mouthguard.					

Wire safety

As part of building a safe and comfortable controller for the Parapedal, we need to decide on how to build our mouthpiece. This obviously requires us to choose a mouthpiece and pressure sensor; however, it's important not to forget about proper wiring. Exposed wires can be dangerous to the user, especially considering that the controller is inside the user's mouth. This means that the environment is sensitive and wet, which only adds to the complexity. Because of that, it's important for us to properly seal the wires. We found a variety of items that we can use as a sealant for exposed wires with different advantages and disadvantages. Depending on how close the advantages of our different items match our requirements will help determine what we will be using.

- Requirements:
  - Insulates cables
  - Safe
  - Looks neet
  - Easy to apply over an uneven surface

Products	Liquid Electric Tape	Heat Shrink Tubing	Electric Vinyl Tape
Material	Vinyl Rubber	Polyolefin	Vinyl Glue
Price	9	8	9
Insulation	Can perfectly cover any wire or joints while still liquid and after its dried	Great for covering the wire but struggles at the end where the wire intersects with other devices	It's great for joints where multiple devices meet
Clean Look	Looks well sealed and orderly	Looks well sealed and orderly	Looks well sealed and orderly

Ease of use	Easy application of liquid after which you just need to let it dry	Easy to slide over wire and shrinks with heat ray	Can be used like normal tape to cover sensitive areas
Safe	Toxic material which in the mouth can be very harmful	Non toxic material	Non toxic material
Uneven surfaces	In liquid form can be easily applied everywhere, regardless of how uneven a surface is	It's a flexible material however is limited in size as it can only really cover wires	Can easily be applied around joints however a bit more complicated for uneven surfaces as you need to make sure it covers everything

STAR BRITE Liquid Electrical Tape with Applicator Brush Cap Black 4 OZ.

Description:

Our first concern when choosing how to seal open wires is to make sure that the wires can be sealed completely, even at joints between devices. Due to that, we started looking for the best sealant for uneven surfaces. One of the options we found was STARBRITE Liquid Electrical Tape. This would be perfect for sealing different connections as it can easily be applied regardless of what shape the connections are, as well as being labeled as water-resistant. When we started doing a bit more research into it, we discovered a major concern with Liquid Electrical tape - it is toxic while it is still in liquid form. While it is said to not be toxic when dried, we considered the wet and sensitive environment of our device, and concluded this would still be dangerous. Although STARBRITE's Liquid Electrical Tape is one of the best sealants in terms of versatility, it's simply not safe for us to use as it could lead to poisoning the user, which automatically disqualifies the Liquid Electrical Tape as an option to be used for Wire safety.

[12]

Waterproof Heat Shrink Tubing Kit, Large Marine Dual Wall Adhesive Shrinkable Wire Wrap Tube, Insulation Sealing Wear-Resistant Cable Protector by YUKSY

Description:

A great option we found to properly seal our wiring is Waterproof Heat Shrink Tubing. Heat Shrink Tubing works by pulling the tubing over our wiring and then heating up the tubing, which will shrink it around our wiring, sealing it off. This method allows for easy covering of the wire by length. The one concern of using Heat Shrink tubing is that it's not as great in sealing off joints between wiring and different devices. However, with the use of some other sealants, we feel confident that we can properly help close off these gaps. We were thinking of either using small plastic bags, which can be placed around the joint but inside the tubing before shrinking it to seal it, or using Electrical Vinyl tape, which is described below. With its versatility and ability to be combined with other sealing methods, Waterproof Heat Shrink Tubing is one of our preferred items to use for wiring safety. As we discovered that Liquid electrical tape is toxic, we have looked into Waterproof Heat Shrink Tubings' safety as well. The tubing is made of Polyolefin, which is non-toxic to Humans and a great fit for our Parapedal Controller’s location in a wet environment.

[13]

Electrical Tape Vinyl Black - 3/4 IN 65 FT Professional Flame Retardant Waterproof PVC Electrical Insulation Tape For All Weather Under 600V Home Outdoor Multipurpose Electric Insulating Wire Wrapping

Description:

As mentioned in the Waterproof Heat Shrink Tubing, Electric Vinyl Tape would be a great addition to help seal the required wiring. Vinyl tape is great for covering smaller uneven areas like the connection between wires and devices; however it isn’t as good for covering larger areas along the length of the wire. Tubing, on the other hand, is great at covering longer distances along the wire; however can pose some difficulty covering the ends and connection between the wire and the device. The two items seem to work perfectly together and cover each other's gaps. Therefore, we are planning on using Electric Vinyl Tape alongside Waterproof Heat Shrink Tubing for Wire safety. As both help cover each other's weaknesses, are water resistant, and non-toxic, making it a very reliable combination for wire safety.

[14]

Summery

Requirement	STAR BRITE Liquid Tape	Heat Shrink Tubing	Electrical Vinyl Tape
Price	9	8	9
Insulation	Can perfectly cover any wire or joints while still liquid and after its dried	Great for covering the wire but struggles at the end where the wire intersects with other devices	It's great for joints where multiple devices meet
Safety	Toxic material which in the mouth can be very harmful	Non toxic material	Non toxic material
Clean look	Looks well sealed and orderly	Looks well sealed and orderly	Looks well sealed and orderly
Uneven surfaces	In liquid form can be easily applied everywhere, regardless of how uneven a surface is	It's a flexible material however is limited in size as it can only really cover wires	Can easily be applied around joints however a bit more complicated for uneven surfaces as you need to make sure it covers everything
Conclusion	Not usable	Optimal Choice	Optimal Choice
Reason	Although it really checks all our boxes in terms of utility, it is toxic, which in an environment like our mouth makes it unusable	It's an optimal voice when it comes to covering wires over long distances, but has issues insulating joints and uneven surfaces. Works well with Electrical Vinyl Tape	It’s an optimal choice when covering uneven areas, but not good for longer pieces. Works well with Heat Shrink Tubing

Display

As part of our Parapedal device, we need our device to switch between different modes. An active mode that will press the piano pedal based on the pressure, put on the pressure sensor, and a passive mode that will press the piano pedal down to the maximum regardless of pressure. Our Parapedal also needs to be initially configured in order to properly fit every single piano. These actions, along with battery life and other features, will need to be displayed. Therefore, our device will need to implement a functioning display. This display will need to seamlessly integrate into our device and be compatible with all our other components. One of our most important requirements is that the device needs to be able to run on Arduino. With that as the main requirement, we choose our optimal display based on how closely it matches our other requirements.

- Requirements:
  - Working display
  - Hardware Compatible
  - Software Compatible
  - Cheap
  - Large

Products	ESP32 Board	1602 LCD	SPI TFT	Waveshare 2 LCD
Dimensions	L: 65mm W: 29mm	L: 80mm W: 36mm	L: 53mm W: 45mm	L: 55 mm W: 35 mm
Pins	30	16	24	8
Price	12	10	12	15
Resolution	128x64	16x2 Character LCD	128x160	240x320
Specs	Dual Core RAM:16KB Memory: 520KB Bluetooth	AIP31066 controller 32 characters I2C serial interface	ST7735 controller chip 1.8-inch SPI TFT LCD Display	ST7789 Interface 262K color
Voltage	3.3V	5V	3.3V	3.3 / 5 V

Hardware Compatible	Can fit into our design as it meets all the spec requirements however is a PCB that we don't really need	Fits all size requirements and can be connected to our PCB's	Fits all size requirements and can be connected to our PCB's	Fits all size requirements and can be connected to our PCB's
Software Compatible	Can be used with Arduino and would be able to match our code	Can be used with Arduino and would be able to match our code	Can be used with Arduino and would be able to match our code	Can be used with Arduino and would be able to match our code

ESP32 Development Board WiFi & Bluetooth Dual Cores Wireless Module CH340 Driver Support Integrated PCB Antenna with 0.96" OLED Display for Arduino/Micropython, Micro USB Interface, 34GPIOs

Description:

The first display we take a look at is a chip that already has a display integrated into it. This will make it very easy in terms of ensuring compatibility, as it is already integrated into the system. It will also be more reliable and easier to get the maximum performance out of. At the end, however, we have decided not to go with a display-integrated chip due to certain criteria. The display provided is very small, and it will make it hard to read for the user. The main issue however, is that we already have a chip that we agreed upon, which meets requirements much better than this chip. Simply having a display attached to it doesn't make up for the shortcomings of this chip. Therefore, having an unwanted chip attached to a small display is simply not what we need for our display choice.

[15]

1602 LCD Display Module with IIC I2C Serial Interface Adapter LCM Black Backlight Screen Red Font AIP31066 Controller 16X2 LCD for Arduino, Raspberry pi, Tinkerboard, Nano pi, Banana pi, stm32, DIY

Description:

The 1602 LCD Display is simple, cheap, but has a pretty low-resolution display. Even with its low resolution, we actually decided on this display for our prototype. The main reason behind this is due to its simplicity and our familiarity with it. During our Junior design class, we already did some basic programming on this display, and we still have a leftover display from the class. Since we already own it and understand how it functions, there are few reasons for us not to use this display. Based on its performance and usefulness of the display in our prototype Parapedal, we can decide on what exact specifications we want out of our display. This display will most likely not end up as our final display, but as for now, it perfectly fits our requirement for our prototype.

[16]

3pcs 1.8 Inch Serial SPI TFT LCD Screen Display Module PCB Adapter IC 128x160 Dot Matrix 3.3V 5V IO Interface Compatible LCD1602

Description:

The TFT Display is a great option in terms of what we want for our project. Although we already decided on a display for our prototype, this screen is a great option for our final display. The TFT Display runs Arduino and is compatible with our software requirements. In regard to our hardware requirements, it also fits our desired specs and relies on 3.3 Volts. The fact that it is cheap and comes in sets of 3 is also great because it allows us to make possible errors while installing it on our device without having to worry about extra costs or time wasted on shipping. The resolution is also better than the 1602 LCD Display, while still being straightforward to program. When it comes to our selection, if we're looking for a small upgrade to our resolution but nothing too fancy, this Display is the perfect option for our Parapedal. It fits our requirements, is simple to program, and is cheap.

[17]

Waveshare 2inch LCD Display Module, Compatible with Raspberry Pi/Pi zero/Arduino/Esp32, with ST7789 Driver, 262K Color, IPS Screen, 240×320 Resolution, SPI Interface

Description:

The Waveshare LCD display is our fanciest option in terms of displays. It has the highest resolution and has color display options. If during our prototype design, we decide that the display is an integral part of the Parapedal and we want to invest in a higher quality display, then we will likely implement this option. Just like the other displays, it is able to run Arduino and meets our software requirements. In regards to hardware requirements, it is also a great match, being able to integrate with most boards currently being sold on the market. In terms of price, it is the most expensive option; however, it is still well within our specification requirements.

[18]

Requirement	ESP32 Board	1602 LCD	SPI TFT	Waveshare 2 LCD
Cost	12	10	12	15
Resolution	128x64	16x2 Character LCD	128x160	240x320
Voltage	3.3V	5V	3.3V	3.3 / 5 V
Hardware Compatible	Can fit into our design as it meets all the spec requirements however is a PCB that we don't really need	Fits all size requirements and can be connected to our PCB's	Fits all size requirements and can be connected to our PCB's	Fits all size requirements and can be connected to our PCB's
Software Compatible	Can be used with Arduino and would be able to match our code	Can be used with Arduino and would be able to match our code	Can be used with Arduino and would be able to match our code	Can be used with Arduino and would be able to match our code
Conclusion	Disqualified	Prototype	Backup	Optimal Final Design

Reason	Although it is a great display we decided to not go with a display attached to a PCB since we will be designing our own PCB	This is the Cheapest Display and we already own 3 of them and are pretty familiar having used them in other classes. It has however the lowest resolution so were not planning on using it for the final design	Second best Display in terms of Resolution and great choice if our primary option fails to work	Best resolution with the most capabilities making it the optimal choice for the final design

Actuator and Servo Motor Selection

In the design of the piano pedal assist device, actuation plays a crucial role. The system must be responsive, quiet, lightweight, and powerful enough to press down a piano pedal, requiring between 10 to 40 Newtons of force.

Requirements:

- Satisfactory output force
- Fast Actuation speed
- Satisfactory voltage and current
- Low size and weight
- Low Noise level
- Compatibility with selected electronic devices
- Ease of integration
- Affordable Cost
- Safety features

Actuators:

Actuator			
Device	RVMARINEPAT Actuator	20N Linear Actuator	VEVOR Actuator
Price	39.99	27.99	33.99
Force/Torque	1000 N	20 N	1500 N
Stroke/Range	100 mm	30 mm	100 mm
Speed	10 mm/s	30 mm/s	10 mm/s
Voltage	12 V	12 V	12 V DC
Noise	Low	Low	Low
Limit Switches	Built-in	None	Built-in
Waterproof	Yes	Not specified	IP54
Material	Not specified	Lightweight plastic	Aluminum alloy
Size/Bulk	Bulky	Compact	Bulky
Control	Direct DC	Direct DC	Direct DC
Pros	High force, waterproof, built-in switches	Fast, compact, low-cost	High force, durable, includes brackets
Cons	Bulky, slow, high power draw	No limit switches, lower durability	Overkill, slower, heavier
Decision	6th choice	4th choice	5th choice

[19] RVMARINEPAT Linear Actuator

Description:

This actuator is a powerful solution that exceeds the mechanical requirements for our piano pedal device, which typically only requires 10–40 N. The RVMARINEPAT delivers 1000 Newtons, meaning it can easily actuate even heavy pedals. Despite its slow full-stroke time, the short stroke needed can be achieved in under 2 seconds, which satisfies our response time target. It is also low-noise, waterproof, and includes built-in limit switches for safety. However, due to its bulk and price, we believe it's better suited as a backup or testing option.

[20] 20 N Linear Actuator

Description:

This actuator is a great fit for our piano pedal assist device because it provides just the right amount of force (20N), which matches the typical 10–20N needed to press a piano pedal, without being overly powerful like larger models. It has a high speed of 30mm/s, ensuring quick response and allowing even a 30mm movement to complete in about 1 second. Its affordability makes it a strong candidate for the final design, especially if force requirements stay below 20 N, but is more expensive than the motor options.

[21] VEVOR Linear Actuator

Description:

The VEVOR actuator offers 1500 N of force, which far exceeds what's needed for our piano pedal assistance. However, its robust build and included mounting brackets make it mechanically convenient and stable. With 10 mm/s, it's fast enough for the small stroke distance we need, allowing for acceptable response times. Built-in limit switches ensure it stops automatically at the ends. This actuator is not a top choice due to its price and speed.

Motors:

This table compares four common types of motor technologies—Servo Motor, Stepper Motor, DC Brushed Motor, and Brushless DC (BLDC) Motor—across several key performance and design aspects.

Motor Technologies

Device	Servo Motor	Stepper Motor	DC Brushed Motor	Brushless DC (BLDC) Motor
Control Method	PWM	Step pulses (via driver)	Variable DC voltage	Electronic controller (ESC)
Precision	High (built-in angle control)	High (discrete steps)	Low (unless encoder added)	High (with encoder)
Speed	Fast	Moderate	Very fast	Very fast
Torque Output	Moderate	Moderate to high	Variable	High
Voltage Range	Low (4.8–7.2V)	Typically 5–24V	Wide (1.5–24V)	Wide (7–48V)
Power Consumption	Low to moderate	Moderate	Varies (depends on load)	Efficient (low power loss)
Feedback Capability	Yes (internal)	Open-loop (optional closed-loop)	No (unless external encoder added)	Yes (often includes Hall sensors)
Integration Complexity	Simple (direct microcontroller)	Moderate (requires driver)	Simple to moderate	More complex (needs ESC)
Common Applications	RC control, robotics, automation	CNC, 3D printers, precise positioning	Fans, toys, small appliances	Drones, e-bikes, high-performance tools
Form Factor	Compact	Compact to medium	Compact	Compact to medium
Decision	✓			

Motors			
Device	MG995 Servo Motor	MG996R Servo Motor	DFRobot 12V DC Motor
Price	~$4.00 each (4-pack)	~$5.75 each (4-pack)	$16.50
Torque	10 kg·cm	13 kg·cm	45 kg·cm
Stroke/Range	0–180°	0–180°	N/A
Speed	0.16 s/60°	0.17 s/60°	83 RPM
Voltage	4.8–7.2 V	4.8–7.2 V	12 V
Noise	Quiet	Quiet	Low
Limit Switches	N/A (Servo Control)	N/A (Servo Control)	None
Waterproof	No	No	Not specified
Material	Metal gears	Copper gears	Metal gear + encoder
Size/Bulk	Small	Small	Medium
Control	PWM (Positional)	PWM (Positional)	DC + Encoder
Pros	Budget-friendly, Arduino-ready, easy control	High torque, reliable, fast, best performance	High torque, encoder feedback, good control
Cons	Requires rotary-to-linear linkage	Requires linkage, higher peak current	Needs controller, higher voltage
Best Use	Basic prototyping or low-cost build	Final design: balance of cost, speed, and control	Good for complex feedback-controlled designs
Decision	2nd choice	✓	3rd choice

[22] MG995 Servo Motor

Description

The MG995 servo motor with a mechanical linkage is one of the best choices for our project because it provides precise positional control, fast response times, and sufficient torque. Unlike linear actuators, which require additional circuitry for position feedback, the servo has built-in angle control, making it ideal for both static mode and dynamic mode. Additionally, servos operate quietly and efficiently at 3.3–7.4V. The servo's 0.16s/60° speed ensures quick pedal presses, meeting our ≤ 500ms response time requirement, and its 10kg-cm torque is strong enough to depress piano pedals without overheating under normal use. Though it needs a mechanical linkage to convert rotary to linear motion, its cost and reliability make it a strong candidate for the final design and prototype.

[23] MG996R Servo Motor – High Torque Option

Description:

The MG996R servo stands out as the optimal choice for our project due to its excellent balance of torque, speed, reliability, and cost-effectiveness, surpassing both linear actuators and the MG995. Delivering 13 kg-cm of torque, it is 30% stronger than the MG995's 10 kg-cm and is significantly more powerful than most compact linear actuators. With a fast response time of 0.17s per 60°, it enables responsive musical pedal actions while maintaining peak current draw of 2.5–3A. At just $5.75 per unit, the MG996R is also more affordable than both

MG995 servos and the linear actuators previously considered. Additionally, it offers the flexibility of programmable speed control through Arduino, making it easy to fine-tune responsiveness to user preferences. Though it needs a mechanical linkage to convert rotary to linear motion, its cost and performance make it a strong candidate for the final design and prototype.

[24] DFRobot 12V DC

Description:

The DFRobot 12V DC motor with encoder is a powerful option for our project, especially in cases where more torque and accurate position tracking are needed. It provides high torque, making it capable of pressing even stiff piano pedals without strain. Unlike servo motors, it doesn't include built-in angle control, but it comes with an encoder that allows us to track movement precisely using software. This makes it useful for future designs where dynamic feedback or advanced calibration might be added. However, the motor runs on 12V and requires a separate driver, which makes it more complex to integrate. Still, its strength, durability, and encoder support make it a solid choice for projects that demand more force and detailed control.

Motor vs Actuator

Device	Motors	Actuators
Motion Type	Rotary	Linear
Control Type	PWM (Pulse Width Modulation)	Direct DC or Relay control
Precision	High	Moderate
Speed	Fast	Moderate to slow
Force/Torque Output	Moderate	High
Voltage Range	Low voltage (5–7V)	Medium voltage (typically 12V)
Power Consumption	Low to moderate	Higher
Integration Complexity	Low (simple microcontroller setup)	Moderate (driver + physical switches)
Position Feedback	Built-in angle feedback	Usually none (unless sensors added)
Typical Applications	Robotics, automation, RC devices	Furniture, machinery, industrial tools
Response Time	Very fast (milliseconds)	Moderate (hundreds of ms to seconds)
Noise	Low	Low
Form Factor	Compact and lightweight	Bulky and heavy
Decision	✓	Not used for final design

Final Selection and Justification

After careful evaluation, we selected the MG996R servo motor as the optimal motor for the piano pedal due to its superior balance of performance, cost, and integration ease. With 13 kg·cm of torque, it provides sufficient force (10–20N) to reliably press piano pedals, while its 0.17s/60° speed ensures quick, musically responsive actuation and is faster than most linear actuators. The servo's PWM-based positional control simplifies implementation for both static and dynamic modes, eliminating the need to use additional feedback circuits required by linear actuators. At $5.75 per unit, it is significantly more cost-effective than linear actuators ($20–$40), and its compact size, quiet operation, and 4.8–7.2V power requirements make it ideal for portable, battery-powered designs. While linear actuators (20N models) were considered, their strong force, slower response times (1s full stroke), and complex integration made them less suitable for the design. The MG996R's copper gears also ensure durability, and its Arduino compatibility allows control without additional drivers. A simple rotary-to-linear linkage will convert its motion into pedal presses, offering a lightweight, low-cost, and high-performance solution.

USER INPUT PRESSURE SENSOR

The main solution to the user input problem is to signal to the device the user's intentions, with minimal impact on the playing of the piano itself. To accomplish this we have decided that the most effective way for the user to signal their intentions is through sensing the tongue's pressure on the interior of a mouthguard device.

Design Goals

The system requires a user input method that enables intention signaling with minimal interference to the piano performance. The solution must be:

- Non-intrusive to the musical performance
- Intuitive and responsive
- Reliable in its operational environment

Sensor Requirements

- **Size constraints:** 1mm to 10mm diameter (Dictated my mouthguard size)
- **Force Range:** 1N to 10N (Dictated by the average tongue force)
- **Actuation Area:** Approximately 20 mm in length (Dictated by the tongue movement accuracy)
- **Environmental:** Use non toxic material or encapsulate (Dictated by the users health)
- **Performance:** High life cycle and low unit cost (Dictated by economic viability)

- **Electrical:** Safe voltage/current levels,easy MCU interface (User health and ease of development

Technology selection

Resistive force sensors were chosen over some other alternatives such as capacitive, piezoelectric, and optical sensors based on:

- Simple analog output compatible with standard ADC inputs
- Linear response characteristics over the force range
- Minimal signal conditioning
- Cost-effectiveness

Sensor Option analysis

Requirement	Specification	Option 1: RP-S5	Option 2: BF350-AA	Option 3: RP-C10-S
Size	1mm to 10mm diameter	4.75mm x 5mm	4mm x 7mm	10mm x 19mm
Force Range	1N to 10N	0.1N to 10N	No reasonable change in range	0.2N to 20N
Actuation Area	~20mm length coverage	5mm length	7mm length, complex mounting	19mm length, but l coverage
Material Safety	Biocompatible / Encapsulatable	Phenolic Resin (Biocompatible)	Phenolic Resin (Biocompatible)	Polyester
Performance	High Cycle Life & Low Cost	$4.70 cost	$0.70, but poor performance	$2.50 cost, 1M cyc
Electrical Interface	Easy MCU Integration	Linear Resistance (300Ω-50Ω)	Difficult Strain Gauge Integration	Linear Resistance (

RP-S5 Resistive force sensor

This sensor is widely available and has good reviews from other purchasers. It is available on amazon. This sensor is currently the prime candidate and preliminary testing shows that it meets the electrical design goals and fits well into the mouthguard.

https://www.amazon.com/dp/B07N1JKSGP?ref=ppx_yo2ov_dt_b_fed_asin_title

BF350-AA strain gauge

This option was also purchased due to its low cost and tested for electrical compatibility, the strain gauge arrangement makes it difficult to integrate with the device and the sensors did not have a reasonable resistance change over the expected force range https://www.amazon.com/dp/B0CNJTY2F3?ref=ppx_yo2ov_dt_b_fed_asin_title

RP-C10-ST

This sensor meets the electrical requirements but may necessitate physical design changes so that it can fit in the mouthguard.

MICROCONTROLLER (USER INPUT & MAIN CONTROLLER)

System Architecture

The device has a dual controller architecture made up of:

- **Peripheral Controller-** Reads the pressure sensor in the mouth-guard and wirelessly communicates it to the Main controller.
- **Main Controller-** Receives sensor data, controls servo rotation.

Peripheral controller requirements

- **ADC Inputs:** 3 channels minimum for force sensors
- **Wireless communication:** low latency protocol support (BLE or ESPNOW)
- **Power consumption:** low power for battery operation
- **Form factor:** Compact size for user comfort

Main controller requirements

- **PWM outputs:** 3 channels minimum for motor control
- **Processing power :** Real time motor control capability
- **Wireless communication:** low latency protocol support (BLE or ESPNOW), WIFI for WEB interface control
- **Power management:** Efficient operation for extended use

MCU Option analysis

Category	Requirement	ESP32-S3	ESP32-H2	Nordic nRF52832	ST STM32WB55
Potential Role	Main or Peripheral	Main Controller	Peripheral Controller	Peripheral Controller	Peripheral Controller

Analog Inputs	3+ ADC Channels	8 Channels (12-bit)	8 Channels (12-bit)	8 Channels (12-bit)	16 Channels (12-bit)
Motor Outputs	3+ PWM Channels	36 Pins	36 Pins	12 Pins	3 Pins
Wireless	Low Latency (BLE / ESP-NOW)	Integrated	Integrated	Integrated BLE	Integrated BLE
	WiFi for Web Interface	Integrated	Not Supported	Not Supported	Not Supported
Power Profile	Low Power for Battery Use	Higher Consumption	Lower Consumption	As low as 0.3uA	Low Power
Processing Power	Real-Time Control Capability	Dual Core 240MHz	RISC 96MHz	Single Core 64MHz	Dual Core 32/64MHz
Price	Per Unit	$5.06	$3.58	$4.00	

ESP32

The ESP32 comes in a variety of configurations with all having both wireless capability and a variety of analog to digital converters (ADC) and General purpose input output pins (GPIO) for sensor reading and external device control. The major categories are the D0WDQ6 , S3, C6 and H2. The H2 device is viable as a peripheral controller as it supports BLE and ESPNOW but does not support WIFI so it will not work for hosting the calibration and setup webpage. It does however boast lower power consumption. The C6 has a higher power consumption but is low cost and has WIFI compatibility. The S3 is the most powerful with a faster processor, higher cost and full wireless capability. A viable approach therefore is to either use an S3 for both the peripheral and the main controller for design simplification. The optimal approach, however, is likely to have a hybrid system where the peripheral controller is a low power H2 and the main controller is an S3 to utilise their relative strengths.

- Pros:
  - Integrated wireless communication (BLE & ESPNOW)
  - Complete feature set for both controllers
  - Extensive software support and development tools
  - Cost effective
  - Built in calibration for Vref in ADC
- Cons:
  - Limited ADC range, 0 to 1.1V
  - Higher power consumption (multiple sleep modes available)

Nordic semiconductor nRF52832

- Pros:
    - Integrated bluetooth
    - Sufficient peripherals for both controllers
    - Good power efficiency as low as 0.3uA
    - Compact form factor of 6x6mm in QFN

- Cons:
    - No integrated WIFI, so external module needed for web interface
    - Lower processing power
    - Potentially higher developmental complexity due to smaller user base

The nRF52832 is more focused on low power and BLE functionality and is acceptable as a peripheral controller. Due to its low power processor and lack of support for WIFI, it will not work as a web page host on the main controller. The development process is also not as widely documented as the ESP32 series of controllers.

STMicroelectronics STM32WB55

The STM32 competes directly with the nRF52832 with BLE capability but no WIFI support. This makes it challenging to implement as the main controller but makes it a viable alternative as the peripheral controller. It also boasts low power but has a similar disadvantage to the ESP32 series as the development process is not as well documented.

- Pros:
    - Integrated wireless communication (BLE
    - Complete feature set for both controllers
    - Uses the STM32CUBE ecosystem for easier development
    - Extensive software support and development tools
- Cons:
    - Higher cost
    - No WIFI compatibility

Chapter 4: Standards and Design Constraints

Throughout the development of our assistive piano pedal system, we followed a variety of established engineering standards and worked within several important design constraints. These standards helped guide our decisions around PCB layout, motor control, and system communication, ensuring that our design would be reliable, safe, and easy to integrate.. This chapter breaks down the standards we used and explains how each design constraint shaped our final implementation.

PCB Design Standards (IPC-2221)

The IPC-2221 standard provides general design rules for PCBs, helping ensure they meet both reliability and electrical performance requirements. For our custom PCB, this standard was a key reference throughout the design process.

We applied IPC-2221 in several important ways:

- Trace width and spacing were carefully calculated to handle the transient currents generated by the servo motors, minimizing voltage drops and preventing excess heat buildup.
- For thermal management, we used large copper pours and placed thermal vias near voltage regulators to keep temperatures within safe limits.
- Material selection was based on durability under thermal cycling and mechanical vibration
- We followed component placement guidelines to ensure good signal integrity and reduce the risk of EMI, especially around sensitive areas like the ESP32 microcontroller.
- Design for manufacturability was another key focus; we avoided overly tight trace spacing around connectors and kept pad-to-hole alignment clean to simplify assembly.
- Lastly, we adhered to proper electrical clearance, aligning with IPC Class 2 standards to prevent shorts.

Communication and Interface Standards

For our project, we made it a priority to follow the established communication and interface standards. These standards help ensure that all parts of the system work well together. They define how devices talk to each other, how power is delivered, and how data moves between components. For our assistive piano pedal system, we focused on the following standards as the foundation for making smart decisions in both the electrical design and how the system communicates wirelessly.

USB Type-C

USB Type-C is a modern connector standard designed for both power and data transfer. Its reversible design makes it easier for users to plug in without worrying about orientation. One of its best features is the reversible design, where you can plug it in either of two ways, which makes it super user-friendly. It can carry a lot of power and is already standard in most phones, laptops, and chargers today.

USB Type-C is likely to be used as the main power input for our system. Since it supports higher current levels and is already found in most chargers. Its compact size also helps keep the device portable and easy to set up near a piano.

USB Power Delivery (USB PD)

USB Power Delivery is a protocol that works alongside USB Type-C to negotiate how much power a device receives. Instead of sending a fixed voltage, USB PD allows a device and charger to communicate and agree on the best voltage and current levels. Its benefits include higher power delivery and preventing overloads or undervoltage conditions

Although our current project may not require full USB Power Delivery support, following the basic principles of this standard can prepare the system for future upgrades we may have in mind. Including USB PD support would also improve our device's compatibility with different adapters.

Bluetooth

Bluetooth is a widely used wireless communication standard for short-range connections. It's ideal for small devices that need to communicate quickly without using too much power. Bluetooth Low Energy in particular is designed for low-latency and energy-efficient performance.

Bluetooth would likely be used to allow wireless control of the pedal system, such as triggering modes or activating pedals possibly through a smartphone.

I²C (Inter-Integrated Circuit)

I²C is a simple communication protocol that uses two wires to connect multiple digital devices: one line for the clock signal (SCL) and one for data (SDA). I²C is designed for simplicity and efficiency; it allows a single master to communicate with several slave devices, such as sensors or displays, over a shared bus.

The SDA line is bidirectional, which means that data can flow in both directions depending on whether the master is reading from or writing to a device. The master is responsible for initiating communication, synchronizing data transfer using the clock signal, and determining when transfers begin and end. Communication starts when the master issues a start condition, signaled by the SDA line transitioning from high to low while SCL is high.

Each device on the I²C bus is assigned a unique address and the master uses this address to target a specific slave. This allows multiple devices to share the same lines without conflict.

Electrical-wise, the bus is designed as open-drain. That means devices can pull the signal low, but they cannot drive it high. Instead, pull-up resistors are used to bring the lines high when no device is pulling them low, ensuring proper voltage levels.

Speed classifications within the I²C standard:

- Standard mode (up to 100 kHz)
- Fast mode (up to 400 kHz)
- Fast Mode Plus (up to 1 MHz)
- High-Speed mode (up to 3.4 MHz)

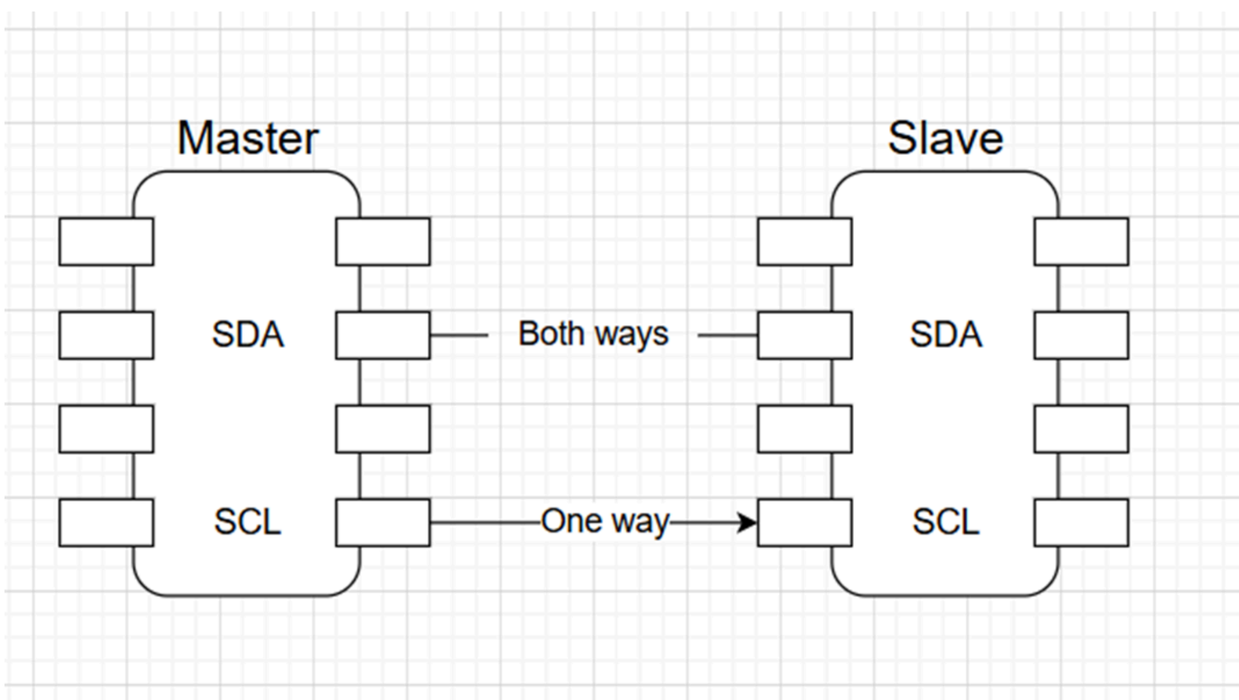


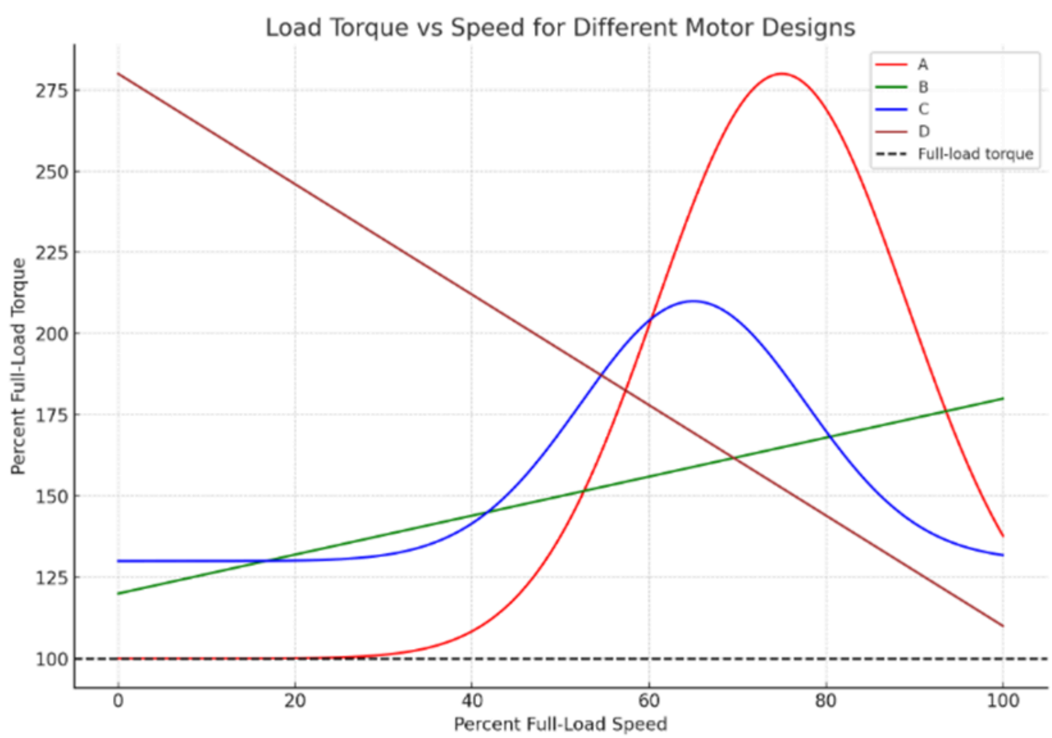
Figure 4.2

Motor Control (NEMA ICS 2-1996 ; IEC 60034)

To ensure safe and reliable motor operation, we follow these two relevant standards: NEMA ICS 2-1996 and IEC 60034.

NEMA ICS 2-1996 is a standard published by the National Electrical Manufacturers Association that defines requirements for motor controllers. This standard provides voltage ratings and thermal protection for DC motors.

IEC 60034 was published by the International Electrotechnical Commission, and focuses on rotating electrical machines. It classifies motor duty cycles into eight categories (S1–S8), describing how motors are expected to operate in terms of load, duration, and thermal equilibrium.



(4)

S1 – Continuous Duty: Motor runs continuously under constant load long enough to reach thermal stability.

S2 – Short-Time Duty: Motor runs under constant load for a short time, not long enough to reach thermal equilibrium, followed by a full cooling-off period.

S3 – Intermittent Periodic Duty: Sequence of identical cycles of operation and rest, where the motor does not reach thermal stability and starting currents are negligible.

S4 – Intermittent Periodic Duty with Starting: Similar to S3, but each cycle includes motor starting, where the current contributes to temperature rise.

S5 – Intermittent Periodic Duty with Electric Braking: Like S4, but includes braking in each cycle.

S6 – Continuous Operation with Intermittent Load: Continuous operation without rest, alternating between periods of load and no-load.

S7 – Continuous Operation with Electric Braking: Like S6, but includes periodic braking.

S8 – Continuous Operation with Variable Load and Speed: Complex cycles that include different loads and speeds without rest periods.

The MG996R servo motors used in this system follow design considerations based on NEMA motor standards, even though they aren't officially classified under a NEMA motor frame. To guide decisions on power and thermal design, we referred to NEMA ICS 2-1996 for general safety practices and compared our motor usage patterns to the IEC 60034 duty cycle classifications.

Since these motors only activate briefly with rest periods in between, their behavior closely matches Duty Cycles S3 and S4 in the IEC standard, which describe intermittent operation without reaching full thermal equilibrium.

Design Constraints

To make sure the system works reliably in real-world situations, we set a few important design constraints. These are based on what users actually need, the limits of our hardware, safety considerations, and how we want the system to perform.

Motor Response Time and Power Constraints

The main goal of this project relies on simulating a human foot pressing a piano pedal in real time. To achieve it, the motor must respond within a narrow time window, typically 500 to 700 milliseconds, which corresponds to a natural human reaction. If the motor is too slow, the system will feel unresponsive and throw off musical timing. On the other hand, if it reacts too quickly, it might overshoot or behave too aggressively, making the experience uncomfortable for the user.

The system is powered by a USB source, meaning available electrical power is limited at around 5V and 500–900 mA depending on the port. This limits the types of motors that can be used and constrains the current draw during both startup and continuous operation. Overdrawing current could lead to system resets. Therefore, balancing power consumption and fast motor response makes it a key design challenge that can be solved with rechargeable batteries as the main supply for the motors.

Input Recognition and Press Threshold

This project is meant to be accessible for users who have physical limitations, such as reduced strength and dexterity. That means the input system has to be forgiving, it should recognize slower bite inputs just as reliably as a strong and deliberate press. A conventional input device wouldn't cut it; it would risk missing valid signals from users who can't exert much force or respond quickly.

Just like with motor response, timing is critical here too. If there's too much delay between when a user bites down and when the pedal moves, anything over about 500 milliseconds, it can throw off their rhythm and make the system feel disconnected. In a musical setting where timing is everything, even small delays can disrupt the rhythm, making the user feel frustrated. At the same time, we can't make the system overly sensitive. It needs to filter out unintentional movements, brief jaw twitches, or electrical noise that might otherwise trigger false inputs.

Finding this middle ground puts pressure on both the hardware and the software. The sensor has to be sensitive enough to detect low-force input but robust enough to avoid reacting to background activity. On the software side, signal processing algorithms must quickly and accurately distinguish between real and accidental inputs, often in noisy environments.

Communication Range and Reliability

The project relies on wireless communication to keep things flexible and user-friendly. This not only helps reduce clutter but also makes sure users aren't tied down to one spot, something especially important for our target consumers.

However, going wireless brings its own set of challenges. Signals can be blocked or weakened by common obstacles like metal piano frames, chairs, or even assistive equipment. These kinds of interferences can lead to delays, dropped connections, or unpredictable behavior.

We established a minimum reliable range of 4 meters to make sure the system works well across a variety of real-world scenarios - whether it's a stage performance, a music classroom, or a home studio. To meet this 4m range, we had to carefully select a strong, compact antenna and optimize the signal path. If the connection drops or a command gets missed, the servo won't respond, which will make the system unreliable.

Since consistency is so important in our project, uninterrupted communication became one of the core design challenges. The system needs to maintain a steady link under different conditions, without forcing the user to worry about where they're standing or how close they are to the controller. Many users may not be able to move freely, and asking them to reposition themselves repeatedly would defeat our goal of accessibility and ease of use.

PCB Design Constraints

The printed circuit board (PCB) must be designed to meet both electrical performance needs and the physical limitations of the chosen manufacturing process. Services like JLCPCB make prototyping affordable, but they come with restrictions on things like how thin your traces can be, how small holes can get, and how close different elements can sit. These limitations shape how we place components and route signals on the board.

The PCB also has to do a lot in a small space. It needs to carry enough current to power the motor, while also supporting delicate analog inputs and communication signals, all without interference. If the layout isn't done properly, we risk overheating, electrical noise, or even failure during regular use, making the system feel unstable for the user.

Component Size and Mechanical Fit

The entire system must physically integrate with a piano, ideally in a discrete manner, without drawing attention or getting in the user's way. This demands compactness in both the PCB and motor mounting hardware. The enclosure, wiring, and connectors must all fit within a tight space, while still allowing for airflow for cooling, access to buttons and ports, and making sure the system is physically stable during use.

This tight form factor rules out using larger modules, which are often easier to handle but take up too much space. Instead, we had to rely on smaller surface-mount parts and low-profile connectors. While these help us save space, they come with tradeoffs: limited power capacity, trickier assembly, and tighter layout tolerances.

Practical Constraints

Time Constraints

Time is one of the most significant constraints affecting the execution of our project. Unlike other teams who may have started their planning phases in the spring or fall prior to Senior Design I, our group did not begin with a pre-defined idea. Instead, we started brainstorming and forming our core idea during the early weeks of Senior Design I, which put us at a disadvantage compared to teams who might have developed their ideas semesters prior to Senior Design I.

A major factor in this delay was team uncertainty. Since our group consisted entirely of four electrical engineering students, there was early concern that we might be reassigned or split up based on the distribution of majors in the class. This uncertainty led us to hesitate on committing to a single idea, as we weren't sure if our team would remain intact. For the first two weeks of the course, we explored several project options in parallel, which made it harder to focus our efforts on one main idea.

Eventually, once it became clear we would stay together as a team, we finalized our project idea. However, by that time, we had already lost some precious time. This late start placed us at a disadvantage compared to teams that had locked in their design choices before the semester even began.

Being enrolled in the summer session of the Senior Design I, we had a more condensed academic calendar and fewer weeks overall, and thus had less time than most groups that take this course in the fall and spring to move from concept to completion. While the typical senior design timeline is two full semesters, our group had less time and fewer scheduled class weeks to develop, test, and refine our system, which posed greater challenges in terms of development time.

Because of this late start, we knew we had to work harder to stay on pace. We made a conscious effort to accelerate our development timeline by initiating prototyping and subsystem integration during Senior Design I, something that many teams wait until the second half of the course to implement. This entails building circuits, testing sensors, programming firmware, and iterative design testing. This challenge required us to have a strong coordination and extra commitment from all of us.

The reduced timeline left little room for setbacks. Issues like delayed component shipments, debugging errors in our servo control logic, or mechanical alignment problems had the potential to significantly delay our progress. As a result, we were forced to prioritize essential functionality and leave out optional and more complex features aside.

Overall, the time constraint was critical for nearly every decision we made. While we were able to deliver a functional and reliable prototype by the end of the course, it took more effort and tighter coordination. This experience showed us the critical importance of early planning and how valuable it would have been to begin technical development even before the course started.

Sustainability Constraint

Although the assistive piano pedal system is not intended for large-scale manufacturing or harsh environments, sustainability considerations still influence design choices since we are aiming for perfection (or at least as close as we can get).

- Component Lifespan: Our system is intended for long-term use by individuals with disabilities. We made sure that Components such as servo motors and switches withstand repeated actuation over months or years. Therefore, the parts we selected were not only considered for performance but also for its durability. Although cheaper alternatives were available that used aluminum windings or lower-quality materials, we opted for a slightly more expensive MG996R servo with copper windings. Copper provides superior electrical conductivity and thermal efficiency, leading to longer motor lifespan and less power loss during operation.
- Material Waste Minimization: We design our custom PCB compactly to minimize material usage and avoid unnecessary board space.
- Power Efficiency: Our system operates entirely on a 5V USB power source, consuming under 20W even during actuation. This low power aligns with sustainability goals by minimizing energy usage during operation.
- Repairability and Modularity: Our design includes accessible connectors and headers, allowing easy replacement of faulty components without discarding the entire system.

Ethical Sourcing Constraints

Ethical sourcing was an important consideration throughout the development of this project. As a team, we made a conscious effort to ensure that all components used in our system were purchased from reputable suppliers.

While cost was a major constraint, we deliberately avoided purchasing parts from vendors with questionable reputations regarding labor exploitation or environmental irresponsibility. Instead, we prioritized suppliers known for ethical manufacturing standards, such as Digikey, Mouser, and SparkFun which are companies that publicly disclose compliance with regulations like RoHS.

We understood that engineering decisions extend beyond technical performance. As future professionals in the field, we believed it was important to acknowledge the social and ethical impacts of the products we create. This meant choosing suppliers that could guarantee responsible sourcing, even when cheaper alternatives were available from unverified or overseas vendors with limited visibility into their labor practices.

Health and Safety Constraints

Health and safety were fundamental considerations in our design, especially given that it's intended to be used as an assistive device. Since users may have limited mobility, we decided it was critical for the system to be both electrically and mechanically safe under all expected operating conditions. One major concern was electrical safety. Although the system operates at a relatively low voltage and draws limited current through USB power, we still had to ensure that all circuits were properly protected. Components such as the servo motor and bite sensor interface were enclosed to prevent accidental contact with wiring or moving parts.

To ensure lab safety during testing, we handled all live components on non-conductive surfaces, and we wore protective eyewear when soldering or drilling enclosures. Wire strain relief was carefully managed to prevent accidental disconnections or shorts.

Because this is an assistive device, we also took care to prevent overheating, noise, or unexpected motion that could startle or harm the user. We ensured that the servo remained within safe temperature limits and that system sounds were minimal and non-disruptive. All final components were secured inside a fully enclosed case, with no exposed wires or metal parts.

Manufacturability

Manufacturability was a key constraint in this project. Since we are college students with no advanced equipment for soldering and fabrication, we had to rely on university lab facilities; unlike professional engineering teams that may have access to CNC machines whenever they want, we were limited to only the university lab equipment.

Our custom PCB was fabricated externally through JLCPCB, which allowed us to produce our board. However, this external fabrication made us strictly follow manufacturability design rules, including minimum trace width, via size, and spacing constraint.

On the mechanical side, we addressed manufacturability constraints through careful part selection. Instead of fabricating brackets, frames, or housings from raw material, we sourced standardized components online, such as servo brackets and adjustable mounts. This allowed us to reduce the amount of mechanical fabrication required and focus our limited tools on small modifications.

Ultimately, manufacturability shaped not only the component selection and layout, but also how we designed the system to be built, tested, and repaired using only the tools and resources available to us undergraduate students.

Chapter 5: Application of ChatGPT and other Similar Platforms

Introduction

AI has become a very controversial tool nowadays, with its massive capabilities being able to greatly speed up your workflow however if misused can result in faulty writings and misleading information as AI can often make facts up. Even though Artificial Intelligence like ChatGPT, Gemini and Deepseek are controversial they have still become a part of our daily life and with the amount of benefit it can provide should be used for a Senior design project. During our time working on our project we have used AI for various purposes however have been careful and aware of its flaws and mainly used it sparingly. If AI was used we made sure to double check its writing to make sure the information we present is truthful.

We used AI for:

Coding:

One of the tasks we used AI for is coding. Our group is made up of 4 electrical engineering majors. Due to this, the focus of our project is much more focused on the physical component of our Parapedal instead of the coding component. That being said, AI helped a good amount in creating our current code. We used both Deepseek and ChatGPT in order to make our project function correctly. This involved creating both a Master and a Slave protocol. The Master and Slave are programmed on two different boards. The Slave protocol focuses on converting the analog input of the pressure sensor into digital information and transmitting it to the Master board. There, the program will use this data in order to move the servo motors accordingly. We used AI to help create a baseline for this code. This means we gave it all the prompts we wanted it to include, like reading the analog data, converting it to digital data, and then transferring it over Bluetooth from the slave board to the master board, and finally using this data to move the servo motors accordingly. AI helped a lot here as it gave us structure on how to set up the code and all the different variables we need to use in C. It also helped us understand all the variables and how the different functions work. Using these explanations enabled us to make some tweaks, since we noticed some mistakes and wrong inputs made by the AI. After setting up the code with its basic functionalities, we also wanted to create a simple website. Using AI, we created a code that would enable the master board to host an online website. In order to create a website, the AI created a hotspot you could connect to on the master board, and when using the IP address, it would bring you to a website that the AI set up. Once we made sure the program worked, we added a couple of additional features and safety measures to our code to ensure that the servo motor, as well as the piano, would not be overstrained.

AI Advantages in coding

Using AI for coding has many advantages. AI is great in helping to organize your thoughts and put them into a functioning code. We already knew what we wanted our code to do; however, structuring and creating a plan on how to implement these features into code is complicated, as there are a lot of variables to take into account when creating a coding blueprint. Since AI works pretty fast and can pop out a functioning code nearly instantly while also taking all tasks into consideration, it is great for creating a source of structure for your code. After entering our constraints, the AI gave a functioning, well-structured code that we could use.

The initial code was pretty large and a lot to take in at first glance. There was a wall of code that you were seeing for the first time, with a lot of new variables and functions that we hadn't seen before. This can be kind of overwhelming at the beginning however, AI is great in these situations. After receiving the code, we used the AI to explain to us step by step what it did to meet our parameters. With that, we were able to break down each task and explain what variables and functions were responsible for and how they all came together to create this code. I find this teaching aspect of AI especially helpful as it really helps you learn and understand code, as you can ask the AI any question and it can give you a great explanation in as much or as little detail as possible. The new code also utilized a lot of new functions and libraries that I had not used or seen before. Here, AI helped break down exactly what each function does and how it contributes to solving the task given to it. This was very helpful since, for this code, we knew we had to use a variety of different functions that we hadn’t learned about yet. Not only did AI help us understand new functions, but it also helped us find the optimal functions to use in our program. Understanding functions is hard enough, but finding exactly what you're looking for is even harder, as there are a million different ways you can write a code, and most of them are not that efficient. Although AI isn’t perfect, it does help create a decently optimized code within the parameters given to it.

The biggest advantage of using AI, however, which was mentioned previously, is speed. With the help of AI, tasks that would have taken hours, if not days, can now be completed in a matter of minutes. We think this is one of the greatest features of AI. The speed that AI provides helps us create a much bigger and more complex project. Especially in an environment like senior design, where we are tasked with creating a meaningful creation in a relatively short time frame. Using AI helps increase the possibilities and scope of projects, as you are no longer as limited by time constraints and can work on more goals. With the help of AI, we are able to work on our advanced goal of the Parapedal project and even start testing out some stretch goals to implement in our project. AI doesn’t just help in increasing the scope of the project by helping reduce the time constraint, but also helping by lowering the barrier to entry.

Coding is pretty complex, and for a project like ours, although we don't have to be experts, we still need to have extensive knowledge of coding. As mentioned previously, our group is made up of 4 electrical engineering majors. Although we learned about coding and the C language, none of us are experts in this field. AI helps quite a bit with this as it fills the gaps in our knowledge and lowers the barrier of coding complexity by implementing it, refining it, and allowing us to further learn from it.

Once our code is finished and everything is implemented, a key advantage of AI is using it to help troubleshoot issues. When errors pop up in our code, we can use AI to identify our mistakes and have it break down exactly what the issue with our code is and how to fix it. This is particularly helpful since the error message is often very cryptic and it's sometimes hard to tell what the exact issue of our code is. We have been using AI for exactly that purpose, and it has been of great help in finding errors and explaining and recommending possible solutions.

That being said, I feel like AI really offers a lot of advantages, especially in the context of our project and coding. AI helped us create a baseline and structure for our code, it helped optimize and explain the intricate parts of the code, and showed us new, efficient functions we didn't know about previously. Most importantly, it helped speed up the coding process, enabling us to perfect our project and work on more advanced and stretch goals in our project. Finally, as a group of Electrical engineers, it also enabled us to code much more complicated code than what we were taught and what we are used to.

AI Disadvantages in coding

AI has been very helpful so far for coding; however, having used it, AI does have some challenges and disadvantages that come with it. One of the main disadvantages being the overwhelming amount of code and data that AI can throw at you. Since we didn't code the baseline code, there is a lot of new information that came at us all at once. This is quite a lot of new code, whose structure and functions can be very confusing. With some time and effort, as well as having AI explain it step by step, we are more than able to overcome these issues. However, it can definitely be overwhelming at first.

Another big issue with AI is that although it helps create a functioning code, it's not as good at optimizing the code. Most AI code, although not terrible, does come with some questionable code that can be simplified or coded more efficiently. This definitely isn't a major issue, but it is something important to keep in mind when using AI. After AI creates code, it's important to review the code and see where the AI had some efficiency issues. This takes a bit of time to find and fix, but in the long run it is well worth it for the integrity of the code.

Although AI is great at understanding a large variety of individual devices, it doesn't always get its compatibility right, especially if the user doesn't specify correctly what the specifications need to be. With that, there is a high chance that AI might have issues properly configuring devices. Even though the code might function perfectly, it could still not work with other devices simply because it was configured to the other device's specifications. This can often be mitigated by giving more specific configurations on tasks, but it is still very important to keep in mind when asking AI to write code for you, and it should be one of the first things to check when the code isn't functioning.

Sometimes, even with the right specifications, AI might still get the code wrong. That is one of the biggest issues with using AI. Finding these issues can be quite hard since, on the surface, everything seems to be right. Although it doesn't happen often, AI does occasionally get things completely wrong. When you are dependent on AI, this is particularly challenging as it will keep giving you wrong information on the issue or task at hand. This can lead to an endless loop of error testing. Due to this, it is extremely important to understand what the code is doing and what it isn't doing in order to spot the errors the AI is making. Not becoming overly reliant on AI is one of the most important things to keep in mind when using AI.

Our biggest challenge with AI was the fact that the digital information gathered from the pressure sensor wasn't configured correctly. Due to this, our server motor wasn't running, since the signal sent to it was incorrect. This took us a while to figure out, since the AI wasn't able to help with the issue. Therefore, we tracked down the error ourselves using our understanding of the code. Once we figured out the source of the issue it was clear what we needed to do, and fixing the issue wasn't that big of a challenge.

Wire Sealing Recommendations

We used AI to give us some general advice on item selection as well. In particular, we were researching ways to properly seal wiring. We didn't know what to use, as we needed a sealant in a sensitive area, the mouth. There, it is important for the sealant to be effective and non-toxic for consumption. Knowing this, we prompted the AI to give us some suggestions on what to use. The AI recommended using heat-shrinking tubing to cover our wires. This is since heat-shrinking tubing is typically made out of Polyolefin, which is a non-toxic substance. Therefore, helping us solve one of the problems we had when planning out the design for our Parapedal.

Pictures

One of the different ways we used AI was for picture generation. This is one of the most interesting ways AI is being used today as it brings a variety of different opportunities to the user. We used AI picture generation to help illustrate what our final product would look like and help give a visual representation of all the research and concepts we have come up with so far. At the beginning we didn't have our Parapedal device built yet, and a picture is a great way to visually present the project. Since our Parapedal had to be approved first we used AI picture generation to help showcase how our final project might look.

This usage of AI has a lot of advantages. The biggest being the ease of use and low barrier of entry for picture generation. To be able to draw a good picture takes years of practice and experience to do successfully, and actually drawing the picture also takes an extensive amount of time. As our team is made up of 4 Electrical Engineers, and none of us is particularly talented in drawing, our ability to draw a good-looking representation of our schematic is incredibly low and requires a hefty time investment. For such a minor part of our project presentation, it is simply not worth it. With the help of AI, however, we are able to produce presentable pictures in a matter of minutes. All we have to do is input some prompts on what we want our picture to look like and make some minor modifications. This will create a good picture that we can add to our presentation of the Parapedal.

Using AI pictures does have some disadvantages. First off, not all pictures turn out well and a good amount of them, especially the first drafts, are often not to the specifications asked for. Some tweaking and adjustment comments can often help in making the image look closer to what you are imagining. Secondly, AI pictures do an injustice to the impressiveness of authentic, human-created pictures. If you decide to spend a good amount of time drawing a picture, it is no longer recognized for its effort as much, since an AI picture, which took a couple of minutes to generate, will look similar in quality. This also leads to the third disadvantage, which is that it makes life as an artist a lot more difficult since now our team, for example, doesn't need to hire an artist for pictures, which is beneficial to us but quite harmful for the economy of artists.

What we thought and how we used it

Benedict Eberhagen

My name is Benedict Ebehragen. Personally, I find AI to be one of the best tools for our senior design project, when used correctly. AI is such a great learning tool that can help create practice problems and explain any issue you have step by step in detail. For our project, I found it very helpful, especially for coding. It helped create a nice baseline for our code, which greatly contributed to our project as it gave us a point we could start from in regards to coding our project. AI was also able to explain to us every line of the code and how it contributed to the overall task. It also introduced a lot of new functions I've never heard of before, which greatly simplified our code. I do, however, find it important not to be over reliant on AI. Although most of the code was great, there were still some glaring issues that AI made in the code - like sending the wrong signal to the servo motors. However, as long as you use AI responsibly and check through and understand the code, it can help in truly optimizing the coding section of our project. Therefore, AI has been a great help to me during our Senior design project.

Andre Llanos

Using the standard LLM style models has been useful for both engineering design and documentation. On the engineering design side, I found it helpful in speeding up the time it takes to prove a certain concept or troubleshoot issues. For example, I did not have much experience creating a web interface that would run on a microcontroller and was not sure if this was a viable direction we should take as a way for the user to interact with the device. However, after only half an hour using the Google Gemini model (2.5 pro), I had a foundation for a web interface that ran on the device for the calibration processes. Although it was not perfect, it showed that the concept was viable in a fraction of the time that it would have taken me to prove otherwise. The downside, however, is that when a problem does arise (as the code was not written by me); it usually takes longer to diagnose and fix. I would say that AI can be double sided depending on the task. On one hand, it allows me to implement features that at one point I would have thought may have been too time consuming otherwise, and thus it gives me greater insight into other fields. However, the insight gained tends to be surface level and it pushes me to be even more specialized in the things that I am good at, leaving other tasks to the AI - which can be detrimental, as I value being a well rounded engineer in lieux of having one specific specialisation.

For documentation, I find it helpful to paste a completed section of writing into the context of one of the AI models and ask for an outside audience perspective to help me balance the writing being both technical and approachable. I then may or may not make edits to help achieve this goal.

Rafael Molina

During this project, I used ChatGPT as a supportive tool to help organize and refine my work. It was particularly helpful in recreating visual diagrams and technical figures, allowing me to represent concepts more clearly without relying on copyrighted materials. Additionally, I used ChatGPT for grammar checks, technical writing assistance, and formatting support. This is particularly helpful when drafting formal documentation, as it helped ensure that my ideas were conveyed clearly and professionally. Overall, ChatGPT served as a writing assistant, helping me to rephrase content, expand technical explanations, and improve the overall quality of my written work.

Jason Knaebel

During my research for the power supply, power regulation, and battery recharging systems, I used ChatGPT to help narrow down my parts selection when my Digikey searches came up short. While ChatGPT was helpful, some of the parts did not meet certain constraints (i.e. output current), it did help me find sections of Digikey that I had not yet searched. ChatGPT also helped with performing the Resistor, Capacitor, and Inductor values for a switching regulator to meet output voltage and output current constraints. Overall, ChatGPT helped expedite my parts research and basic design calculations.

Chapter 6: Hardware Design

The intended solution has both mechanical and electrical considerations. The mechanical design is primarily made up of the pedal actuation system that translates the rotational servo motor motion to the linear motion needed to move the piano pedal. Included in the mechanical consideration will be the enclosures and mounting for the electronic components. The electrical design consists of two major subsystems, the User Input device that senses the users intentions and the Pedal control device that receives the commands and actuates the servo motors. Both of these systems have Battery management and power subcircuits with a USB power input as well as Microcontrollers with both input and output ports.

Motor Interface Subsystem

The motor interface subsystem is responsible for user input into the controlled mechanical motion that depresses the piano's pedal.

This subsystem contains:

- MG996R servo motor
- A mechanical rotary-to-linear linkage
- A PWM-based control circuit from the ESP32 microcontroller

Motor interface block diagram

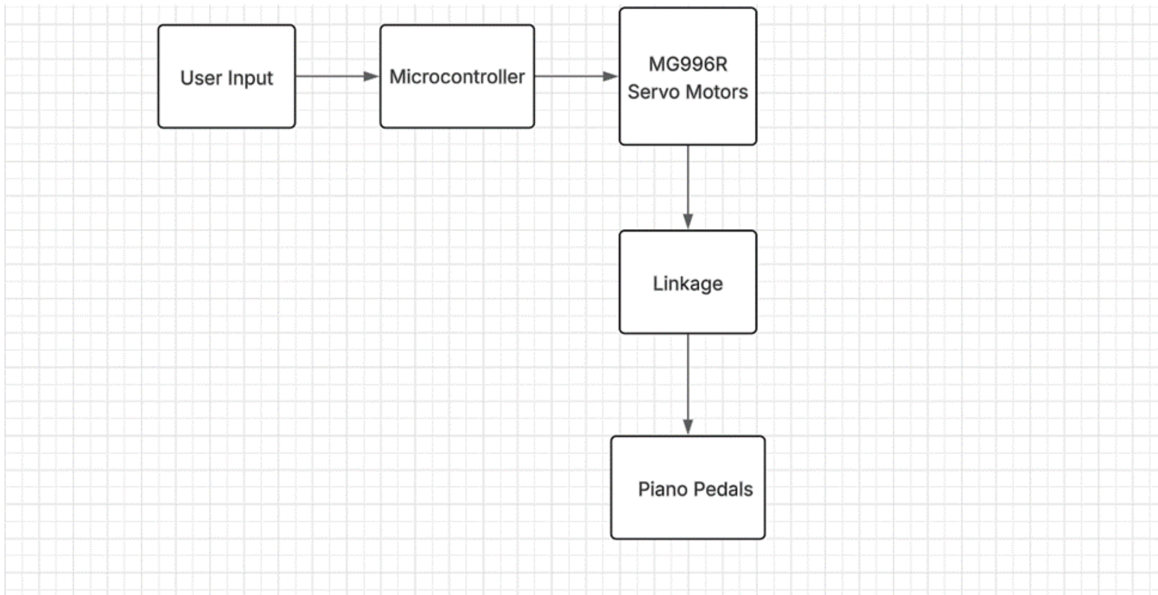


Figure 6.1

Servo Motor Control and Electrical Schematic

Electrical Schematic

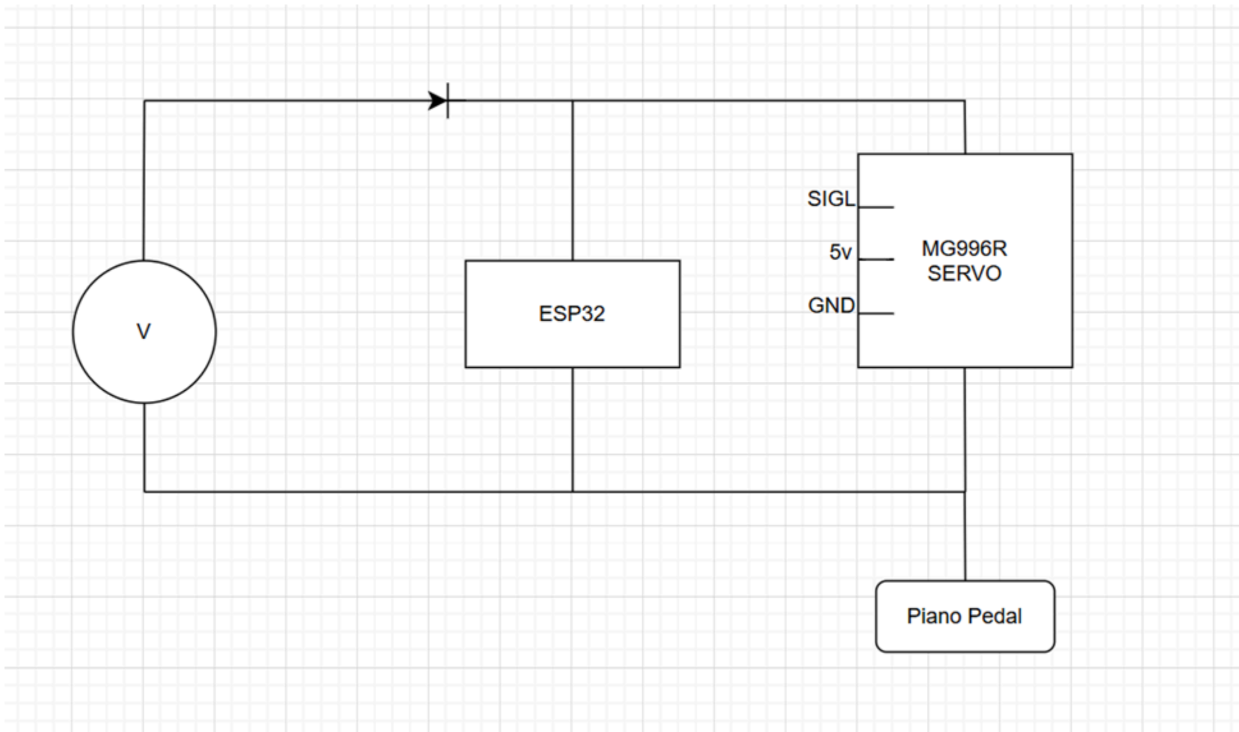


Figure 6.2

This system uses three MG996R servo motors, each assigned to one of the piano's pedals. These servo motors are controlled by PWM signals generated by an ESP32 microcontroller.

Each servo pin is connected as follows:

- Signal (SiGL): A dedicated GPIO pin from the ESP32 sends a PWM signal to control the servo's position.
- VCC (Red wire): Each servo is powered by an external 5V DC supply capable of delivering sufficient current.
- GND (Brown wire): The ground of the power supply is shared with the ESP32 to maintain a common voltage reference.

To protect the system against voltage spikes caused by the servo's inductive load, a flyback diode is placed across the power terminals of each servo.

PWM Signal and Control Logic

The servo motor position is controlled using Pulse Width Modulation (PWM), for setting angular position. The ESP32 microcontroller generates a 50 Hz PWM signal, where the pulse width determines the degree of rotation and the extent of pedal press.

The system supports two control modes based on how the user interacts with the mouthguard sensor:

Static Mode

In this mode, a bite detected by the mouthguard sensor always triggers a fixed PWM pulse width, resulting in a consistent servo movement and pedal response. This is for users who want the pedal to move the same way every time they bite, without having to think about how hard they bite. For example, a bite above a preset threshold may always send a pulse, corresponding to a medium pedal press.

Dynamic Mode

In Dynamic Mode, the ESP32 uses calibrated bite force levels to determine the pedal press. The analog value from the sensor is compared against three user-defined thresholds obtained during calibration:

- Soft bite = slight press
- Medium bite = medium press
- Hard bite = full press

Mechanical Linkage Design

The rotational motion of the servo horn is converted into linear downward motion using a mechanical linkage system. This is achieved by attaching an arm or pushrod to the servo horn, which physically presses the piano pedal when rotated.

Design characteristics:

- Adjustable linkage length to fine-tune pedal throw
- Lightweight but rigid materials
- Mounting bracket to fix the servo position relative to the pedal

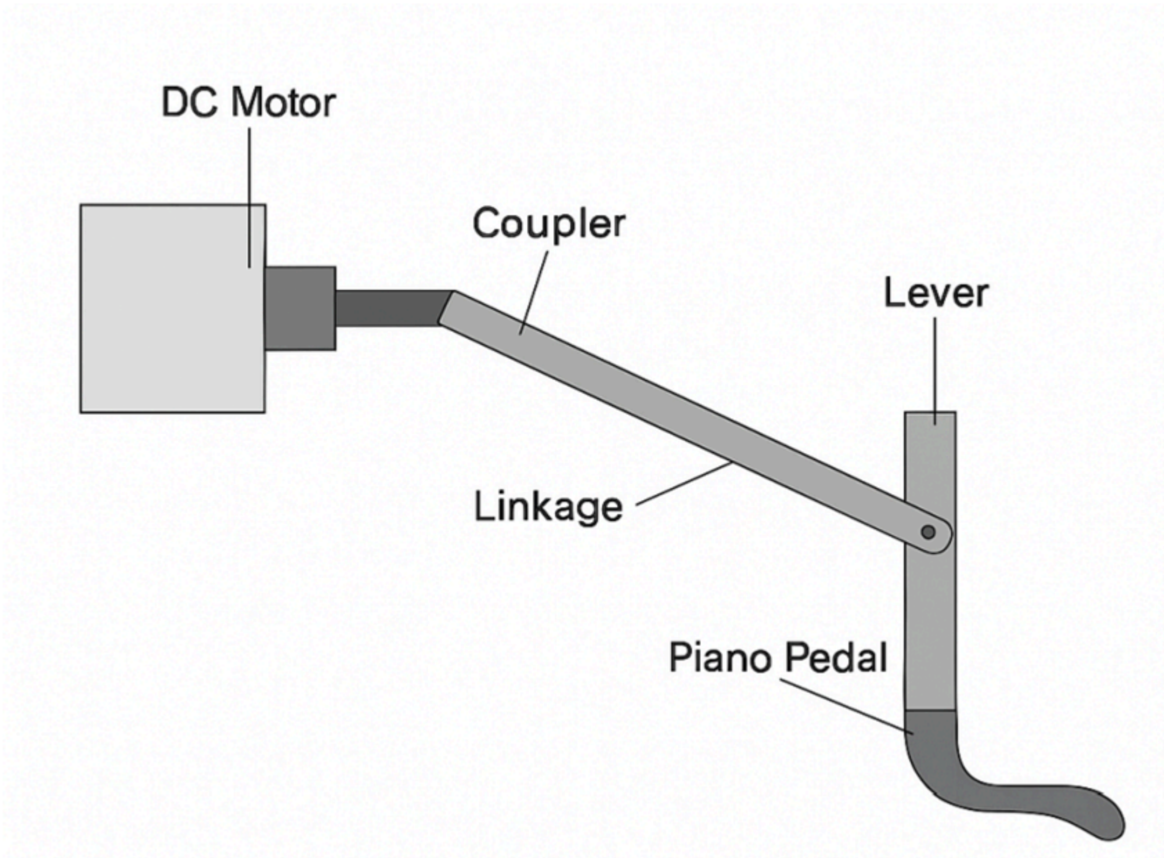


Figure 6.5. Mechanical Linkage Diagram (OpenAI)

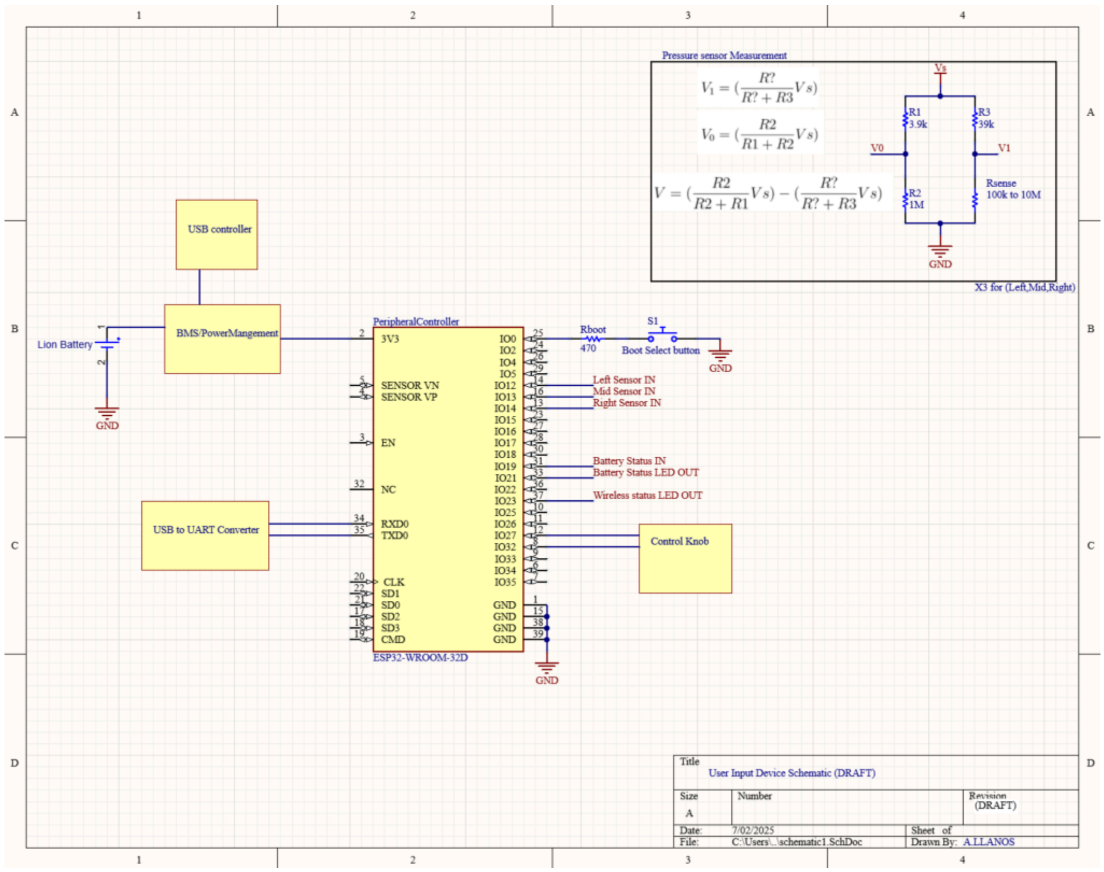
Sensor-Based Calibration

To translate user bite strength into pedal control, a pressure sensor is integrated into a custom mouthguard and connected to the ESP32 microcontroller's analog input. This forms a sensor-actuator feedback loop that is calibrated at setup. The calibration process defines hardware thresholds that trigger each of the three servo motors assigned to the piano pedals.

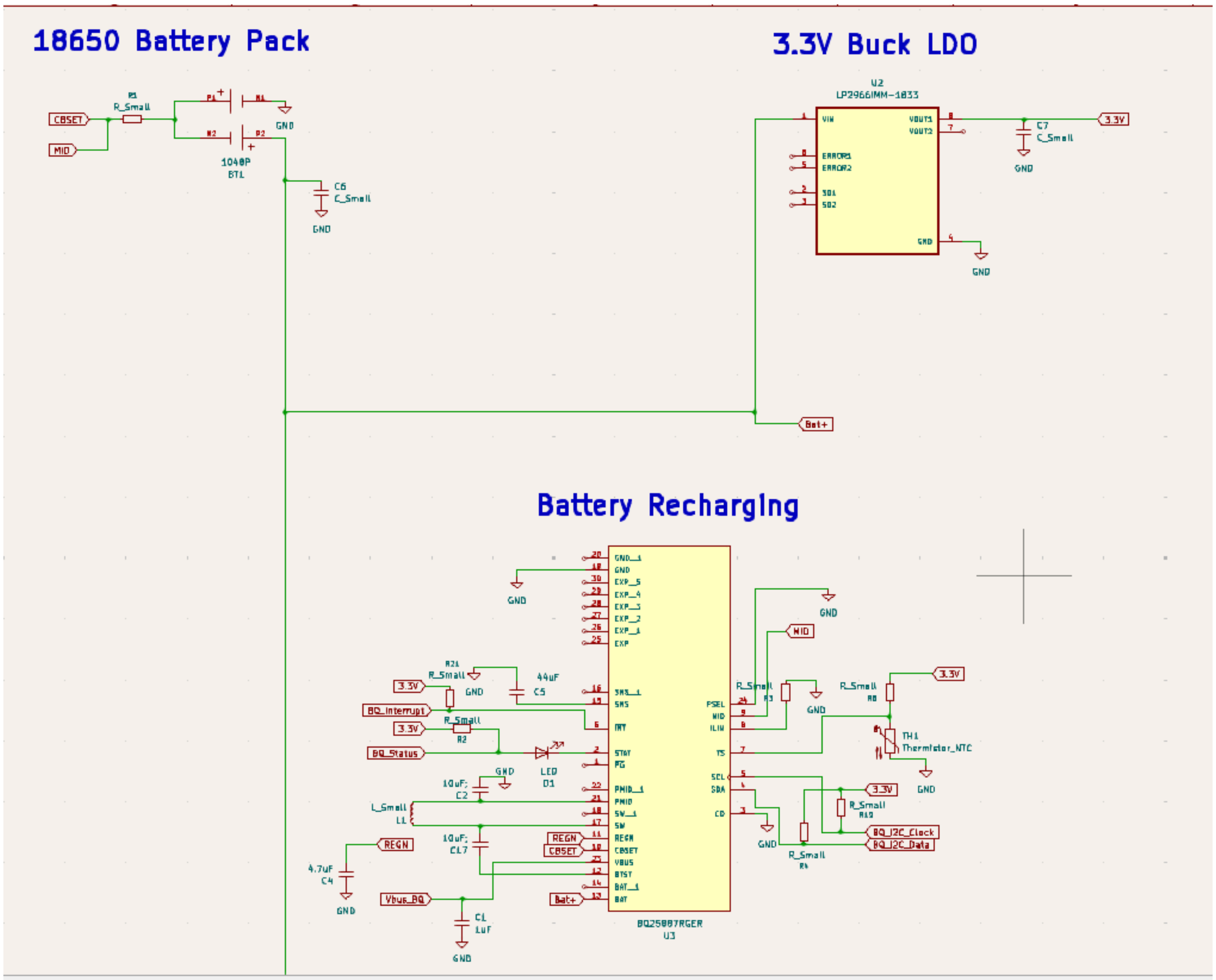
The hardware calibration system includes:

- A bite-sensitive pressure sensor, integrated into a soft mouthguard.
- An analog signal connected to one of the ESP32's GPIO pins.
- An ESP32 microcontroller responsible for processing the sensor values and generating corresponding PWM signals to drive three MG996R servo motors.

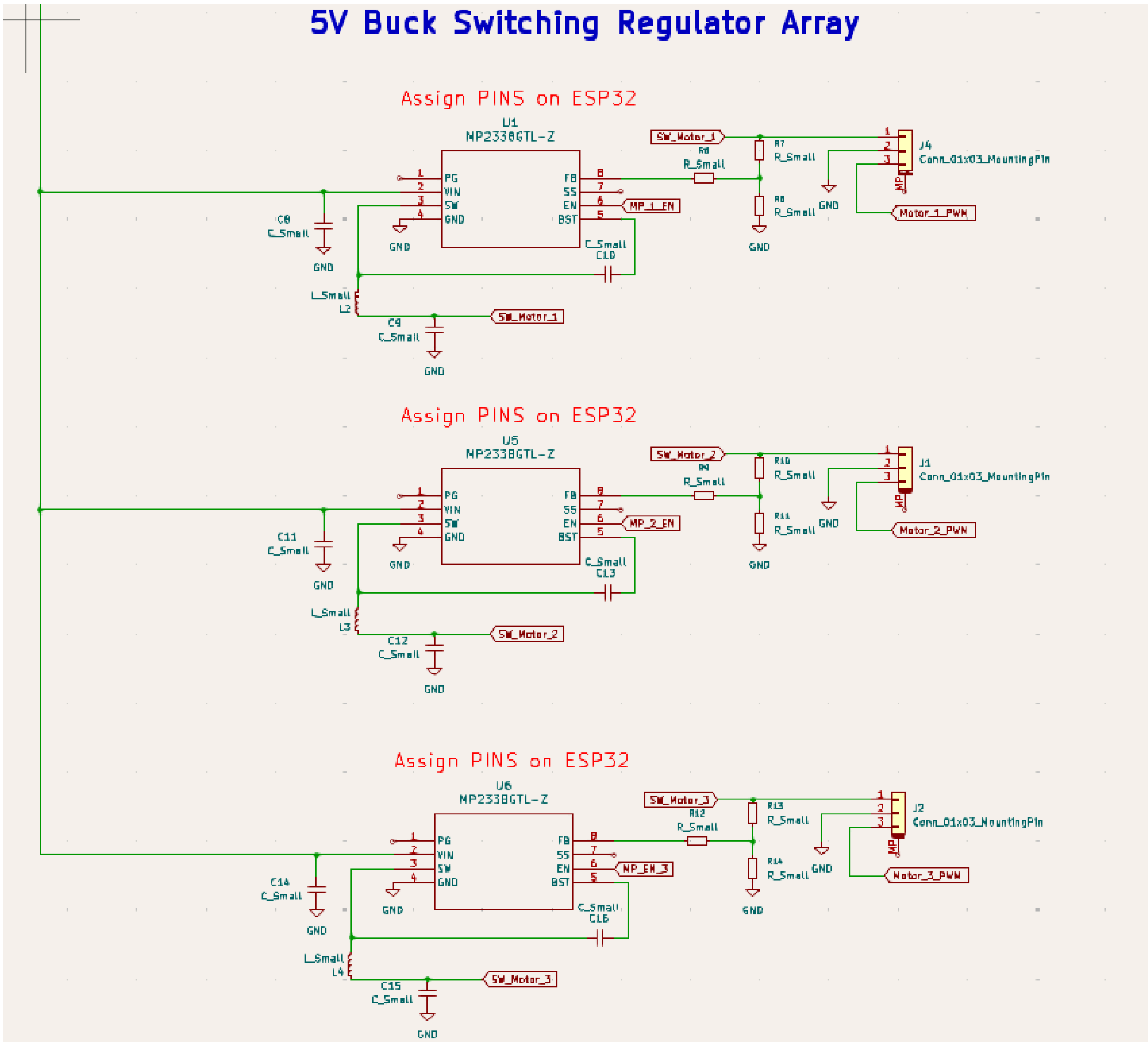
Electrical Design



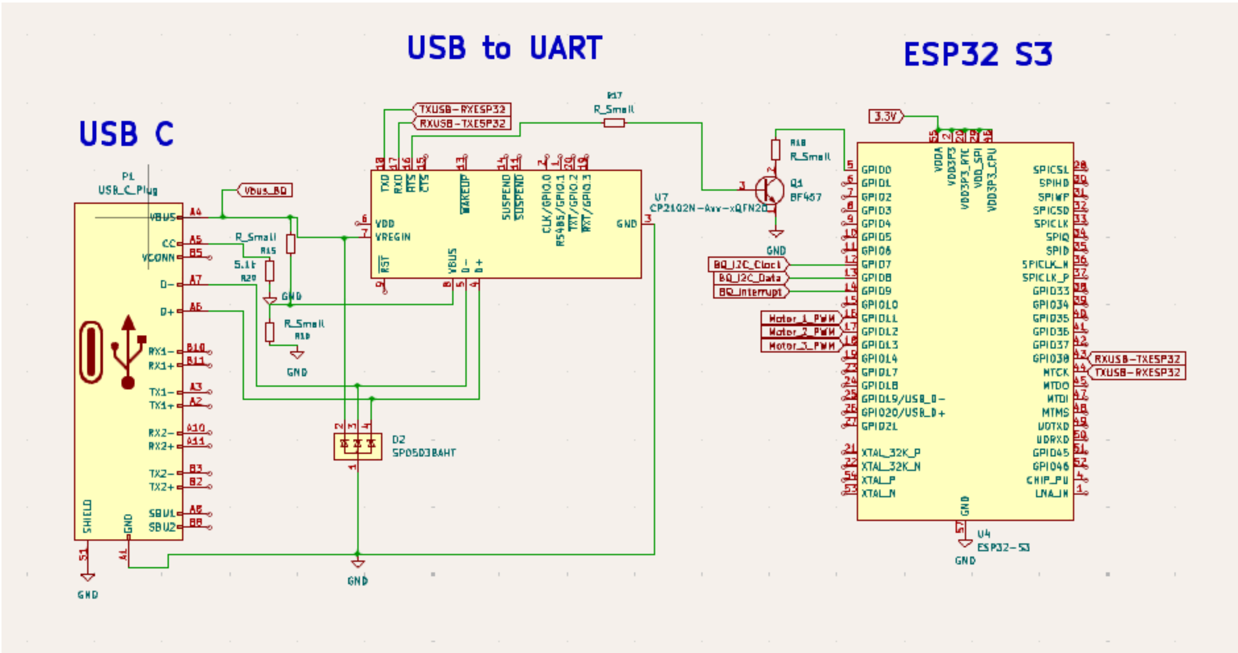
Schematic for the User Input Device(DRAFT)



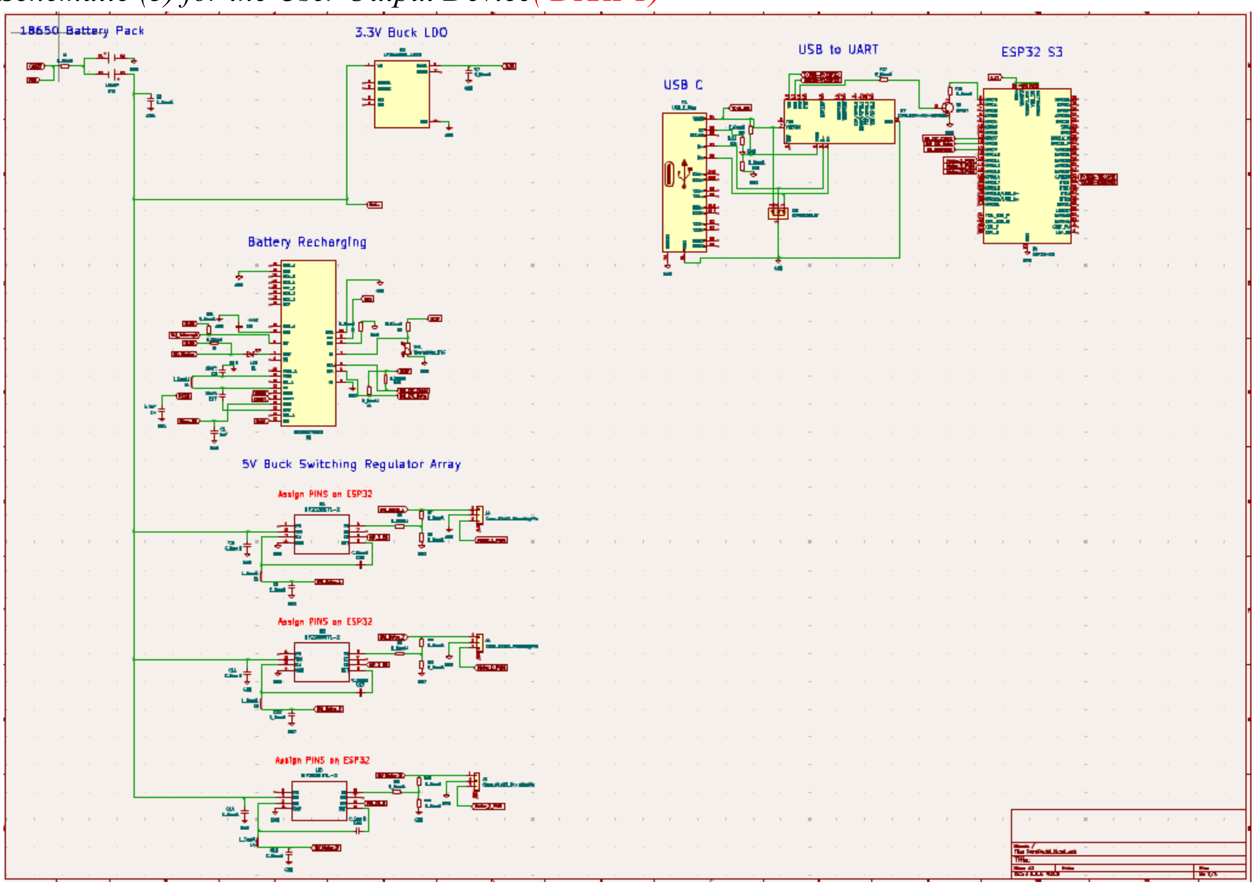
Schematic (1) for the User Output Device(DRAFT)



Schematic (2) for the User Output Device(DRAFT)



Schematic (3) for the User Output Device(DRAFT)



Overall Schematic for the User Output Device(DRAFT)

User Input Electrical Design

The User input can be broken into submodules, the heart of the device is the Microcontroller (MCU) which is an ESP32-WROOM. This device reads the measurements taken by the resistive pressure sensors and transmits them wirelessly to the Main controller. The MCU requires 3.3V for operation and is intended to be battery powered, so a power conditioning and protection module is integrated in the circuit to supply this.

The power module controls the charging and discharging of a lithium ion cell. It is charged through a USB connector (+5V supply) which is regulated to charge the 2.7V to 4.2V battery. The battery is then fed through an Low drop-out regulator to provide the 3.3V to the MCU.(Chip selection to be decided)

To program the MCU, a USB to UART converter is connected to both the USB connector and the TX/RX pins on the controller. (Chip selection to be decided)

The resistance change from the pressure sensor is measured through the use of a wheatstone bridge which is designed so that the output voltage is between 0V and 0.93V depending on the applied pressure. (Bridge to be modified to use only 3xADC instead of 6xADC in differential)

The battery voltage will be read by the ADC to alert the user when it charging is needed and display this state of charge using an LED connected to a GPIO pin

The status of the wireless connection will be communicated to the user through an LED connected to a GPIO pin.

When programming the controller the boot switch needs to be pressed.

Output/Motor Control Electrical Design

The electrical design of the main controller is also centered around an ESP32 microcontroller. The main power source is a lithium ion pack that is regulated to a 5V bus for the servo motors and a 3.3V bus for the microcontroller. The battery is charged by a 5V USB source. To control the servo, a 50Hz PWM signal is generated by the MCU.

In finer detail, we are powering the Output/Motor Control board using a 2-cell 18650 battery pack that will be recharged through a BQ25887RGER using a usb-c wall-connected power supply. The battery power will lead to two separate components; one being an LP2866IMM-1833 LDO that will drop the voltage to 3.3V to power the ESP32 and the CP2102N USB to UART conversion chip, and the other being an array of MP2338GTL-Z switching regulators that will supply the 5V 3A required power.

The ESP32 chip will be processing the signals from the User Input Board via bluetooth to control the motors using the aforementioned 50Hz PWM signal. It will also have the ability (this is a tentative design) to read the battery level from the charger using I2C communication.

Chapter 7: Software Requirements

The Parapedal is a device that focuses on helping people with movement impairment to play the piano better, by helping them activate the foot pedals when and how they want. As part of our Parapedal, there are a variety of different components and tasks we need to program. Our device will be made of two separate devices that need to be programmed, a Master PCB and a Slave PCB. The Master PCB will have most of the functionalities as it is in charge of moving the servo motor, receiving data from the slave PCB, display control, different mode selection, as well as hosting a website to enable fine-tuning of certain settings. The Slave PCB will be in charge of collecting data from the analog pressure sensor and transferring that data over Bluetooth to the Master PCB.

Devices:

- Pressure Sensor
- Servo Motor
- Display
- Mode switch button

Programs

- Setup mode
- Passive mode
- Active mode
- Website readings

Pressure Sensor

For our Parapedal, the pressure sensor is one of the most important components as it is used as the input for the entire device. For the pressure sensor we will be using is a film pressure sensor. The PCB will continuously read the analog signals it gets based on the pressure applied to the film pressure sensor. This analog signal will be sent to the main PCB, which in the prototype will be the ESP32 and our own designed PCB for the final Parapedal. Once the pressure sensor signal arrives, we need to convert the analog signal into a digital signal. For this conversion, we will be using a built-in function, which is the Analog-to-Digital Converter, ADC. Analog to Digital Conversions work through GPIOs 32-39 and function by basically taking measurements at different points of the analog signal and dividing it into different levels, which can then be used as a digital signal.

Once we have our signal converted to digital data, we will translate the different pressure signals over to the servo motor to move the pedals of the piano accordingly. There are many different ways to set up the connection between the PCBs that we will be using. We can use either Bluetooth or wifi.

Wifi communication is a really interesting way to have two boards communicate with each other. As the name suggests, wifi communication can require you to have a functioning router that is transmitting wifi. The two boards both need to connect to the central access point of the router to be able to communicate with each other. Another alternative is to use direct connect for wifi. This is a peer-to-peer mode where devices can connect to each other without a router. This works by having one of the devices act like a soft access point to which all the other devices can connect. This is done through IPs where the soft access point PCB shares its IP address, enabling the other devices to connect to its signal. This is one of the methods we are looking at to connect our two PCBs to each other. We don't want our device to have to rely on a router since we want the Parapedal to be a portable device that can be used anywhere. That means that we don't want our users to carry around an additional router and want our device to be used anywhere, regardless of infrastructure and electricity ports. Wifi connection has some key advantages and disadvantages that make it especially appealing for us to use over the Bluetooth alternative.

Wifi’s advantages are:

- Longer range, which is around 50-100m indoors
- Faster data rate transmission
- Ability to handle multiple devices
- Lower Latency

Wifi’s Disadvantages are:

- Higher power consumption
- More complex protocols
- It requires more processing power

These functionalities can be very helpful for our design. One of the key features our Parapedal needs to have is low latency and fast data transfer. While playing piano, it's important that the device reacts fast to the pressure sensor, as missing the activation can lead to errors in playing a piano piece. We do need to watch out for wifi’s higher power consumption and higher processing requirement however, if we can get our device to overcome these limitations, the speed advantages it would provide would be a great benefit.

In comparison, Bluetooth functions a bit differently from wifi. Bluetooth relies on either point-to-point or mesh networking as its communication method. In a point-to-point Bluetooth setup, we would be programming one board as the master board and the other as the slave board. The master board initiates the connection, which the slave accepts. This point-to-point method is limited to two devices that are connected to each other. As another alternative, we can use a Bluetooth mesh network, which focuses on having multiple devices connected to each other.

This works by transmitting data through multiple devices connected to each other and using them as nodes to reach your final destination device. Bluetooth and its capabilities are a great alternative to wifi due to its various advantages and disadvantages.

Bluetooth's advantages are:

- Lower power consumption
- Easier pairing method
- A variety of support features
- Low processing requirement
- Great for short bursts of information

Bluetooth's disadvantages are:

- Short range 10-30m
- Lower data transfer
- Limited possible connections
- Slower setup for connections

These features can be quite advantageous to use. The lower processing requirements and lower power mean we can build our Parapedal smaller which is important for its portability. The lower power consumption also means that while the Parapedal is on its own battery power, it is able to last longer. Bluetooth's disadvantages of higher latency and lower data transfer do hold it back a bit since fast reaction speed is key for proper piano playing, which is Parapedals most important function.

Both wifi and Bluetooth are good options for our Parapeda device as they have their own advantages and disadvantages. Based on the data and key advantages, we found wifi to be a more suitable way of communicating. This is due to its higher data transmission and lower latency, which is so important for the Parapedal, which needs to react quickly to the constant change in pressure. Therefore, during our prototype, we will be focusing on using the Wi-Fi signal. If it turns out to be too demanding on our processor and power, we will switch over to Bluetooth as it is nonetheless a great alternative.

Servo motor

Our servo motor is also one of the most important components of our device, as it is the part that actually allows us to exert pressure on the different piano pedals. As mentioned in the description of the pressure sensor, the servo motor will receive digital data from the PCB that was provided to it from the analog pressure sensor. The higher the pressure exertion on the sensor, the more the servo motor will tighten the string. All this will be done on the main PCB on the ground. In order to ensure that the servos are operated properly, there are some features that we need to program into the system in order to ensure the safety of the piano and Parapedal device, as well as ensuring a smooth operation of tasks.

One feature we need to install is setting up limiters around the servo motors' movements. That means that we won't always push the servo motor to its limit of 180 degrees. Instead set a limiter at around 175 and 5 degrees in order to ensure the system doesn't strain the servo motor too much. This will limit the servo motor's capabilities however, greatly increase its durability as it prevents overexertion and possible error if wrong data tells the servo to move a specific amount of degrees it is simply not able to move to.

.

If we leave the program to be a direct translation of the data flow from the pressure sensor, then the resulting servo movement will be very rigid as well. In order to prevent this from occurring, we will implement a movement limiter. That means the servo motor can only gradually increase the pressure to a certain extent and won't be able to jump from 10 degrees all the way to 160 degrees and then straight back down to 10 degrees. This is done to ensure the safety of the servo motor, as such rigid and sudden movement can internally damage the servo motor. Of course, we need to be careful with this feature as we don't want the movement to be too slow to the extent where it will affect the quality of playing the piano. This means we really need to strike a careful balance with a degree of change that will limit the strain on the servo motor while still being able to respond quickly enough to a change in pressure.

With this limit on large movement, we also want to limit the servo movement for really small pressure changes. Being able to maintain the same constant pressure on a sensor while playing a piano can be very challenging, as a slight decrease in pressure will result in an instant movement of the servo motor. In order to limit this issue, we will set a limit that requires the servo motor to receive a large enough pressure change before it will actually move the motor. Similarly, we do want to be careful with this limit as we want to make sure that the Parapedal still reacts to pressure changes by the user properly, while ignoring minor shifts in pressure that are unintentionally made by the user.

These modifications are crucial to maintain a smooth functioning system with long durability. Our programming for these limitations will be complicated as we will need to ensure the code doesn't interfere with each other or the user input. More importantly and challenging however, is as mentioned before, to strike a perfect balance. We want these limits to improve the feel and durability of the Parapedal, and not for it to limit the user's ability to play fast and complicated piano pieces.

Display

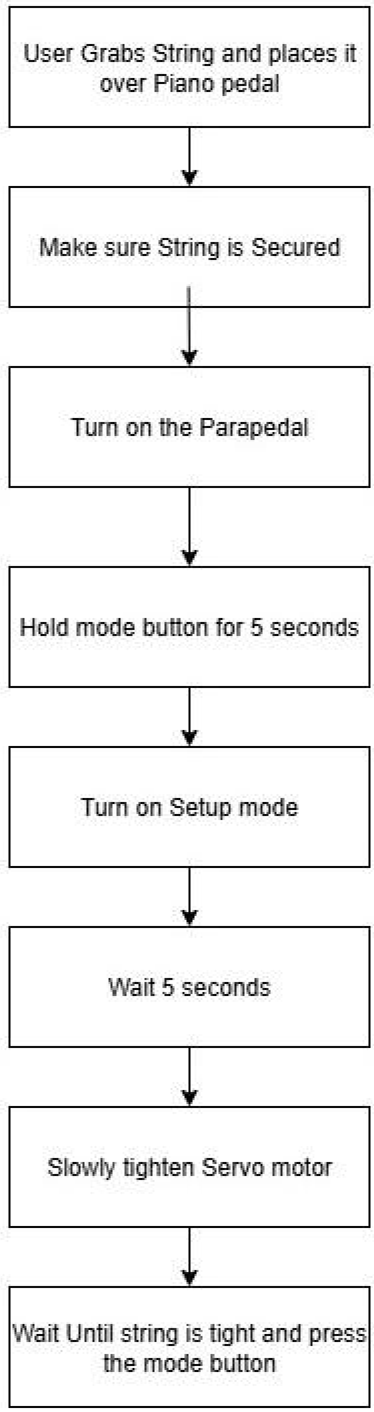
As part of our Parapedal device, we need to implement a display for multiple purposes. First, we need to display the different modes that the Parapedal is currently set to. We have a Setup, active, and passive mode that the Parapedal will be rotating through. We also need a way to display the current energy level as well as its current status on the website connection. These are some of the basic functionalities that the display needs to have, although we may add new features as our project continues.

Mode switching

As mentioned before, the Parapedal will have 3 different modes that we need to toggle between. In order to accomplish that, we will be using a button to toggle between the different modes. There will be a Setup, Passive, and Active mode - in that order - through which the user will be able to toggle between. This button will be integrated into the PCB and will need to be programmed. In order to ensure that we are not toggling multiple times per press, we will also need to implement a debouncing system for the mode switching.

Setup Mode

One of our 3 modes will be the Setup mode. Its purpose is pretty self-explanatory, as it will be activated before you start playing in order to properly connect the Parapedal to the piano pedals. In order to set up the Parapedal, you will need to put the string that is connected to the servo motor over the pedal and then turn the Parapedal on and select the setup mode. To be able to activate the Setup mode, you will need to hold the button down for 5 seconds until the Display indicates that it is in setup mode. In this mode, the servo motors will be turned on and slowly tightened the string to make sure it has a baseline for what a fully extended pedal “looks like”.


User Grabs String and places it over Piano pedal
Make sure String is Secured
Turn on the Parapedal
Hold mode button for 5 seconds
Turn on Setup mode
Wait 5 seconds
Slowly tighten Servo motor
Wait Until string is tight and press the mode button

Servo Motor Activation

In order for the servo motor to properly complete the setup phase, it will need to pull the string that's connected to the piano pedal until tension is built. In order for it to be able to tell when it has reached tension, we are planning to either implement a code that will automatically be able to tell the tension level and stop once it has reached a certain level of resistance, or have the Parapedal manually be told that it has reached its wanted tension by pressing the button one more time.

Passive Mode

The passive mode will be one of the two modes you can set the Parapedal to while playing the piano. The passive mode will be printed on the display once you have selected it. In this mode, the Parapedal will compress the Piano pedal to its maximum range on the activation of the pressure sensor, regardless of the pressure applied. This means that regardless of how much pressure is put onto the pressure sensor, the motor will always press the pedal fully once the user exerts any substantial pressure on the sensor.

Pressure Sensors

The coding for the Pressure sensors in this mode will be very straightforward. Once the Pressure sensor feels a certain level of pressure, it will fully activate the piano pedal. In order to make sure the sensor isn't too sensitive and the user doesn't accidentally press it, we will place the pressure requirement at a decent level where it will not require too much pressure, but also not too little.

Servo Motor Activation

The servo motor will need to be programmed to press the piano pedal down to its maximum angle. This action will be performed once the pressure sensor is activated and sends the signal to the PCB, and then to the servo motor. To ensure the servo motor doesn't exert too much power onto the Piano pedal, we will implement maximums and minimums to the servo motor to ensure it, as well as the piano pedal, aren't damaged.

Active mode

The active mode is a bit more complex than the Passive mode. Once you toggle the button and the display is printing the message Active mode, the Parapedal will switch to a more dynamic pedal activation mode. In this mode, depending on the pressure exerted on the pressure sensor, the Parapedal will alter the angle at which the Piano pedal is compressed, too.

Pressure Sensor

The pressure sensors will be fully utilized in this mode. When pressure is put on the pressure sensor, it will convert the pressure measured to an analog signal that will be sent to the PCB. To be able to properly read this data and to further send it to the servo motor, we first need to convert it to digital data. To ensure that the digital data is correct based on the pressure exerted, we will assign it a digital value between 0 to 180, which will be sent to the servo motor for proper activation. To make sure the sensor isn't hyper-sensitive either, we will implement a baseline pressure that is necessary to activate the servo motor; this will ensure it won't randomly activate. This baseline will be fairly low at 5 degrees, which is lower than the passive mode; however, not too low so as to make it hypersensitive either.

Servo Motor Activation

The servo motor will activate based on the digital signal it receives from the PCB. The higher the value, the higher the angle at which the piano pedal will be pressed down. This will need to be a dynamic transition based on how the pressure changes and how much the servo needs to move. In order not to make the movement too sporadic, we will be implementing a bit of a gradual increase and decrease of the pedal angle. This means it will not always be at the exact pedal angle related to the pressure on the sensor, but will gradually reach that level. As piano playing is still pretty fast, this gradual increase will still be pretty fast, just not instantaneous.

Website Reflection

Our PCB will also create a website to showcase Parapedal's current status. This is to enable the user to receive data about the Parapedal on their phone and not just through the display. As it is on a phone screen, it will also allow us to display more data than on the display, which has significantly less space. None of our features are final yet, although we already have some features we want to showcase.

Current status

One thing we want our website to show is the full current condition of our Parapedal. This will be similar to what the display will be showing, however much more in-depth as the website isn't limited by size, unlike the built-in Parapedal display. Showing the status means it will be displaying first off, in what mode the current system is in. This means that it will state whether it is currently in setup, passive, or active mode. This feature can help the user if the display were to break, and give the user another way to view what mode it is currently in. Secondly the website will also show the current status of the different devices connected to the PCB. It will print the current pressure on any of the three pressure sensors as well as indicate at which angle all of the servo motors are currently turned to. This will be helpful feedback to double-check if the Parapedal is working correctly and help with maintenance in case any issues occur. It will also display how much battery the system has left. Depending on what we are able to program onto the display, this battery display will always be on the website, giving you a constant view of when you need to charge the Parapedal device.

Set up mode specifications

Setup mode is one of the hardest modes to program directly and is where the most possible errors can occur. With the fact that it is also one of the most important modes, as it is what enables the Parapedal to properly interact with the piano, it's important for us to have a backup setup specification. As already described before, as well as shown in the diagram, the setup mode will be activated through a button hold and then go through its setup procedures. Regardless of how well we design this feature, there is always a possibility of failure, which in this case will block you from properly syncing with the piano, resulting in the Parapedal not functioning correctly. In order to overcome this possible error, the website will have an alternate setup function where you can manually input the tightness of the string to set up the Parapedal. On the press of the setup mode button on the website, it will prompt you for inputs on what degree you want the baseline servo motors the be set to. This will allow you to manually change the degrees until you find the proper tightness of the string. This feature will set the minimum possible degree that the servo can slacken to. Therefore, even if you don't put any pressure on the sensor, the Parapedal string connected to the piano pedal will remain tight and ready to quickly respond to your input.

There are two different ways we can host this website, either through HTML or through JSON-REST API. We have looked at both options with HTML's endless programming possibilities and great UI, as well as JSON-REST API, which is a lot more efficient, but relies on a separate frontend. Both of them are great options that can bring a variety of advantages or disadvantages to our design choices for the website.

Html

HTML functions by having the PCB run a lightweight web server. This works by having the PCB store the HTML in its flash memory. When a forge device like a phone connects to it, the PCB will then forward that file to the phone. This content can also be dynamic, like for our Parapedal current status messages that need to be continuously updated. Me and my senior design group have already used HTML in embedded code in various different classes. We all feel very familiar with this kind of code and are comfortable using it. That being said when deciding on how to set up your website, we need to consider more factors than just familiarity in our decision making. Some of the key advantages and disadvantages of using HTML are:

HTML advantages:

- Diverse UI with a lot of functionality
- No additional Client
- Can be used offline
- Intuitive human interaction

HTML disadvantages:

- Requires more Processing power
- Slower communication
- Security risks

These features can be quite helpful when it comes to designing a website, especially as they allow us a large variety of modifications. The fact that it is very user-friendly is also great since our goal is to make the Parapedal as easy and accessible as possible to anyone who wants to use it. It is slower in terms of communication to the PCB however, in this context, it doesn't matter how fast the website is, since it doesn't show anything that is time sensitive.

JSON-REST API

JSON-REST API is another alternative to HTML. It works by having the PCB host a lightweight REST API and listening. When a separate device, like a phone, sends an HTTP request to the PCB, the PCB will respond with JSON data. This method of hosting has some different advantages and disadvantages compared to HTML.

JSON-REST API advantages:

- Requires less processing power and bandwidth
- Better for M2M
- More flexible in terms of interacting with different services like apps
- Lower memory usage

JSON-REST API advantages:

- No built-in UI
- Uses HTTP

JSON-REST API has some important advantages for our device. It requires less memory and bandwidth, which allows us to design a cheaper and smaller device, which can be very helpful by making our device more portable. It is, however, somewhat limited in its capabilities and less user-friendly, which in a device like the parapedal is quite important. Nevertheless, it is still a great option for our website.

Both JSON-REST API and HTML have a variety of advantages and disadvantages that are quite helpful in our design of the Parapedal. JSON-REST API advantages, like lower processing power, are very helpful however, we found the advantages that HTML and the UI functions provided to be a bit more beneficial. Although it requires more processing power, the differnce doesn't seem to be big enough to warrant a different PCB device, and the user friendliness that HTML provides is very important for users to use the Parapedal to its optimal capabilities. Due to this, during our prototype, we will be implementing HTML code for our website. If HTML has some major issues that come up during the prototype, we are going to try out JSON-REST API for our final design.

Chapter 10: Administrative

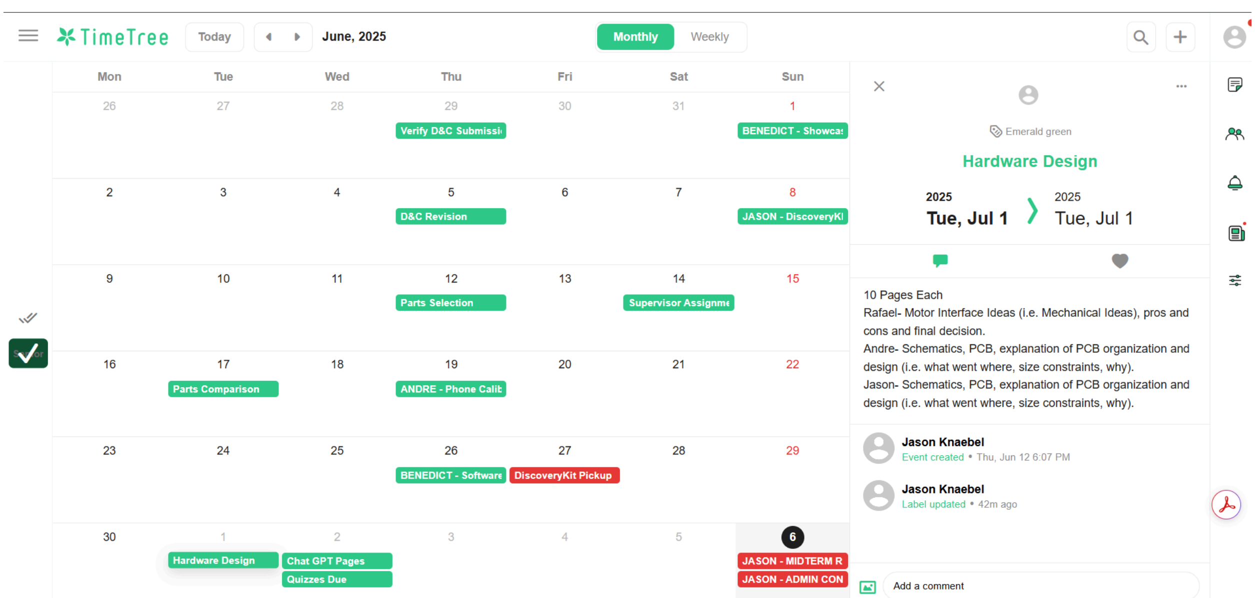
Budget and Bill of Materials

Below is the Bill of Materials (BOM), a table of parts we intend to use when developing both PCBs and their prices. In the bottom right of the BOM is the cumulative total of the cost of parts, which will be our total budget.

Comment	Description	Designator	Footprint	LibRef	Quantity	Manufacturer Part Number	Cost
MP2338GTL	Battery to 5V Buck				10		$11.30
MS621FE-FL11E	MS Lihium Rechargeable Battery	Lion Battery?	SEIK-MS621FE-FL11E-2_V	CMP-00029-00000-1	2		$13.29
ESP32-WROOM-32D	WIFI MODULE 32MBITS SPI FLASH	PeripheralController	FP-ESP32-WROOM-32D-MFG	CMP-194065-000003	1	ESP32-WROOM-32D	$5.38
Boot Select button	Push Button Normally Opened	S1		Push Button NO	1		$1.94
CP2102-GMR	Single-Chip USB to URAT Bridge	U?	SLAB-QFN-28-3125x3125TP_M	CMP-2000-04924-1	1		$8.88
LP2966IMM-1833/NOPB	Dual 150mA Ultra Low-Dropout Regulator	U?	MUA08A_N	CMP-0062-02215-2	10		$15.40
BQ25887	Battery Manager				2		$10.00
18650 Battery Case					2		$9.99
18650 Batteries					2		$3.00
Satin String Spool					1		$6.99
Servo Pulley					20		$3.99
MG996R	Servo Motor				4		$18.00
RP-S5	Resistive Pressure Sensors				2		$9.50
Mouthguard					20		$8.99
						Total Budget	$122.66

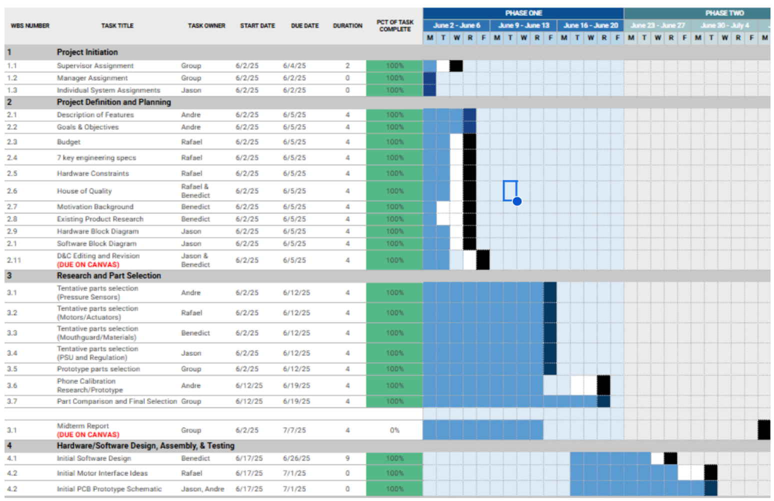
Project Milestones

We are using an app called TimeTree to track our tasks and milestones. TimeTree is primarily for tasks with hard deadlines assigned on Canvas. A picture of some of the scheduling we have made so far is provided below:



Task Delegation

For individual assignments, I have been consulting with and assigning everyone tasks along with deadlines to ensure ample progress throughout the semester. I track the individual deadlines' progress through the Gantt chart shown in the screenshot below. Shown in the Gantt chart are each team member's tasks, the timeframes for their tasks, and the dates of completion.



WBS NUMBER	TASK TITLE	TASK OWNER	START DATE	DUE DATE	DURATION	PCT OF TASK COMPLETE
1	**Project Initiation**					
1.1	Supervisor Assignment	Group	6/2/25	6/4/25	2	100%
1.2	Manager Assignment	Group	6/2/25	6/2/25	0	100%
1.3	Individual System Assignments	Jason	6/2/25	6/2/25	0	100%
2	**Project Definition and Planning**					
2.1	Description of Features	Andre	6/2/25	6/5/25	4	100%
2.2	Goals & Objectives	Andre	6/2/25	6/5/25	4	100%
2.3	Budget	Rafael	6/2/25	6/5/25	4	100%
2.4	7 key engineering specs	Rafael	6/2/25	6/5/25	4	100%
2.5	Hardware Constraints	Rafael	6/2/25	6/5/25	4	100%
2.6	House of Quality	Rafael & Benedict	6/2/25	6/5/25	4	100%
2.7	Motivation Background	Benedict	6/2/25	6/5/25	4	100%
2.8	Existing Product Research	Benedict	6/2/25	6/5/25	4	100%
2.9	Hardware Block Diagram	Jason	6/2/25	6/5/25	4	100%
2.1	Software Block Diagram	Jason	6/2/25	6/5/25	4	100%
2.11	D&C Editing and Revision (DUE ON CANVAS)	Jason & Benedict	6/2/25	6/5/25	4	100%
3	**Research and Part Selection**					
3.1	Tentative parts selection (Pressure Sensors)	Andre	6/2/25	6/12/25	4	100%
3.2	Tentative parts selection (Motors/Actuators)	Rafael	6/2/25	6/12/25	4	100%
3.3	Tentative parts selection (Mouthguard/Materials)	Benedict	6/2/25	6/12/25	4	100%
3.4	Tentative parts selection (PSU and Regulation)	Jason	6/2/25	6/12/25	4	100%
3.5	Prototype parts selection	Group	6/2/25	6/12/25	4	100%
3.6	Phone Calibration Research/Prototype	Andre	6/12/25	6/19/25	4	100%
3.7	Part Comparison and Final Selection	Group	6/12/25	6/19/25	4	100%
3.1	Midterm Report (DUE ON CANVAS)	Group	6/2/25	7/7/25	4	0%
4	**Hardware/Software Design, Assembly, & Testing**					
4.1	Initial Software Design	Benedict	6/17/25	6/26/25	9	100%
4.2	Initial Motor Interface Ideas	Rafael	6/17/25	7/1/25	0	100%
4.2	Initial PCB Prototype Schematic	Jason, Andre	6/17/25	7/1/25	0	100%

Table of Work Distributions

Below is the table of work distributions. While these are each individual's primary and secondary tasks, all members are actively communicating and working on every aspect of development in order to learn about topics we are weak in and teach others about topics within our strong areas. The Gantt chart is the most accurate depiction of our work distributions.

Name	Primary Tasks	Secondary Tasks
Jason	Manager, Power	PCB
Andre	PCB Design	Coding
Rafael	Motors	PCB
Benedict	Coding	PCB

Appendix

Appendix A References

1. CanAssist. "Head-Activated Piano Pedal." *CanAssist*, University of Victoria, www.canassist.ca/EN/main/programs/our-technologies/museum/piano-arts.html. Accessed 25 May 2025.
2. Steingraeber & Söhne. "Steingraeber Pedal Systems for Wheelchair Users." *Steingraeber*, www.steingraeber.de/en/innovationen/pedal-devices-for-pianists-in-wheelchairs/. Accessed 25 May 2025.
3. Resonate Adaptive. "Resonate. Make Music Accessible." *Resonate Adaptive*, www.resonateadaptive.com/. Accessed 25 May 2025.
4. OpenAI. *ChatGPT*. OpenAI, GPT-4o, chat.openai.com/chat
5. Google. *Gemini*. Google, 2.5 Pro, https://gemini.google.com/app
6. “BBTO 10 Pieces Transparent Mouthguard for Sports Teeth Protection Athletic Dental.” *Amazon*, Amazon.com, 2024, www.amazon.com/BBTO-Pieces-Transparent-Protection-Athletic/dp/B07VNDPSR8.
7. “Grinding Mouth Guard for Clenching, Teeth Night Guard for Grinding & Clenching, Teeth Protection for Bruxism, Dental Hygiene.” *Amazon*, Amazon.com, 2024, www.amazon.com/Grinding-Clenching-Protection-Bruxism-Hygiene/dp/B0BZPDCH2R.
8. "KOHEEL Mouth Guard for Grinding Teeth, Adjustable Night Guard for Teeth Grinding & Clenching, Comfort-Fit Dental Guard." *Amazon*, Amazon.com, 2024, www.amazon.com/KOHEEL-Adjustable-Grinding-Clenching-Comfort-Fit/dp/B0DK1GLQKN.
9. "Safe Jawz 596634 Fangz Mouthguard." *Amazon*, Amazon.com, 2024, www.amazon.com/Safe-Jawz-596634-Fangz-Mouthguard/dp/B00QS6ZXH0.
10. "Difiney Anti Snoring Device, FDA Cleared Stop Snoring Mouthpiece for Sleeping." *Amazon*, Amazon.com, 2024, www.amazon.com/Difiney-Anti-Snoring-Device-FDA-Cleared/dp/B0F48ZW2MZ.
11. "Snoring Mouthpiece, Universal Snore Stopper for You & Partner to Sleep Better & Wake Refreshed." *Amazon*, Amazon.com, 2024, www.amazon.com/Mouthpiece-Universal-Partner-Better-Refreshed/dp/B0F2N2Y1S7.
12. "Star brite Liquid Electrical Tape, 4 oz, Black." *Amazon*, Amazon Services LLC, 2024, www.amazon.com/Star-brite-Liquid-Electrical-Tape/dp/B0000AXNOD.
13. "3/4 inch Waterproof Wire Connectors, Shrinkable Insulation Sleeve, Wear-Resistant Wire Terminal." *Amazon*, Amazon Services LLC, 2024, www.amazon.com/inch-Waterproof-Shrinkable-Insulation-Wear-Resistant/dp/B07VQ6VRG8.

14. "Electrical Tape, Vinyl Tape 10-Pack (3/4 inch x 66 ft, Black) - Professional Grade Wire Insulation Tape." Amazon, Amazon Services LLC, 2024, www.amazon.com/Electrical-Tape-Vinyl-Black-Professional/dp/B08N668YQS.
15. "Development Board with Bluetooth and 1.28 Inch LCD Display Integrated, Supports Micropython Programming and USB-C Interface." *Amazon*, Amazon Services LLC, 2024, www.amazon.com/Development-Bluetooth-Integrated-Micropython-Interface/dp/B0DFH77DYF.
16. "SPI Interface 1.14 Inch IPS LCD Display with Backlight, PCB Controller Board Compatible with Raspberry Pi and Tinkerboard." *Amazon*, Amazon Services LLC, 2024, www.amazon.com/Interface-Backlight-Controller-Raspberry-Tinkerboard/dp/B0D2LBRXHM.
17. "AOICRIE 1.28 Inch LCD Display with Adapter Board, SPI Interface for Raspberry Pi (Compatible with Multiple Models)." *Amazon*, Amazon Services LLC, 2024, www.amazon.com/AOICRIE-Display-Adapter-Inerface-Cmmpatible/dp/B0D2TN2P6N.
18. "2inch LCD Module with 320x240 Resolution, SPI/I2C Communicating Interface for Raspberry Pi and Arduino." *Amazon*, Amazon Services LLC, 2024, www.amazon.com/2inch-LCD-Module-Resolution-Communicating/dp/B081NBBRWS.
19. RVMARINEPAT Linear Actuator Electric Waterproof 12V. Amazon, Amazon Services LLC, 2024. https://www.amazon.com/RVMARINEPAT-Linear-Actuator-Electric-Waterproof/dp/B0CX8Y49JN
20. Electric Linear Actuator for Robotics and Automation. *Amazon.* Amazon Services LLC, 2024. https://www.amazon.com/Actuator-Electric-Actuators-Robotics-Automation/dp/B0D17MCVYK
21. VEVOR Linear Actuator with Mounting Bracket, IP54 Waterproof. *Amazon.* Amazon Services LLC, 2024. https://www.amazon.com/VEVOR-Actuator-Mounting-Bracket-Protection/dp/B0CWNBRKWB
22. Deegoo MG995 Metal Gear Servo Motor for Arduino and FPV. *Amazon.* Amazon Services LLC, 2024. https://www.amazon.com/Deegoo-FPV-Servo-MG995-Metal-Gear/dp/B07NQJ1VZ2
23. MG996R High-Torque Digital Servo Motor with Copper Gears. *Amazon.* Amazon Services LLC, 2024. https://www.amazon.com/MG996R-Copper-Digital-Helicopter-Control/dp/B0DK1V5ZSM
24. DFRobot 12V DC Geared Motor with Encoder (83 RPM, 131:1, 45 kg·cm). *Amazon.* Amazon Services LLC, 2024. https://www.amazon.com/DFRobot-12V-Motor-83RPM-Encoder/dp/B01B0IPZTQ