# Assistive Piano Pedal

Group Number: 10

Group Members: Alex Naylor, Dre’ Hodges, Karl Shaver, Matthew Hwang, Matthew Lombardoni

Professor: Dr. Guilherme A. S. Pereira

Date Submitted: 5/2/22

# Table of Contents

# 1. Overview

Music is a foundation of society and is an important part of many people's lives. However, it often involves fine motor skills, something that the disabled and the elderly often have difficulty with or lack entirely. In particular, playing the piano while using a foot pedal can be a challenging task for those with limited function in their lower body. To help these individuals stay engaged in the world of music, we have designed a device that solves this unique problem a disabled or elderly piano player may face without the use of either their legs or arms.

# 2. Background

Music is such an essential part of our society, and many people enjoy playing, performing, as well as listening since the beginning of time. It used to be the case where if someone became disabled, partially paralized for example, they wouldn't be able to play their instrument anymore.

There were a couple projects that took a look at this issue, the first attempt I saw was designed for a man suffering from paraplegia, it was a bar that went across the front of the keys by the players wrists where they could push the bar down to lower the pedal. The device was portable and could be attached to almost any piano. This device was far from perfect however, pushing down on the bar constantly heavily restricts the motion of your arms, which could be crucial when playing piano.

A much better attempt I saw was a project that was done by a group called CanAssist. Their device looks like it goes behind the wood backdrop in front of the players knees, and is a large pedal that presses down from the opposite direction as the foot. This device was controlled by a bluetooth sensor that was placed on the player's head. The woman in the video could raise and lower the pedal by doing the same with the sensor on her forehead.

# 3. Objectives

The objectives of this project were to design and build an assistive mechatronic device that would enable a piano player with the inability to utilize their legs to still play a piano, while the assistive device pushes the pedal down when the player bites down on a button sensor placed in their mouth. The device should be able to press down on the foot pedal on command quickly and accurately enough where it does not hinder the user’s piano playing experience. The device should also be able to release the foot pedal on command as soon as the piano player wills it. This implies the motor has to be of a certain quality and precision, it cannot not push the pedal beyond its limit. The device could be wireless, and rigged with a 5V battery to supply power to the stepper motor, as well as a bluetooth button in the player’s mouth connected to the arduino.

# 4. Project Description

The initial idea was to design a motor attached to a wooden base that has a lever arm that would be able to rotate with enough force to push down onto the piano foot pedal once a button has been pressed. This button would be able to be pressed with the use of the mouth to enable the hands and legs to be free. Once the button is released, the motor should rotate back to its initial position. Initially it was planned for the piano pedal to be attached to the base by carving the wood to have shelves to prevent the pedal from moving, but it was later changed to the use of a top wooden shelf screwed to the bottom base. If the force from one motor wasn’t enough, the design would be able to accommodate for a second motor. The motor would be attached to the base with the use of a bracket. This initial design can be seen in **Figure 1**, and the SOLIDWORKS model for this design can be seen in **Figure 2**. The lever arm was designed in SOLIDWORKS to have a hole in the shape of the motor shaft. This model of the lever arm can be seen in **Figure 3.**

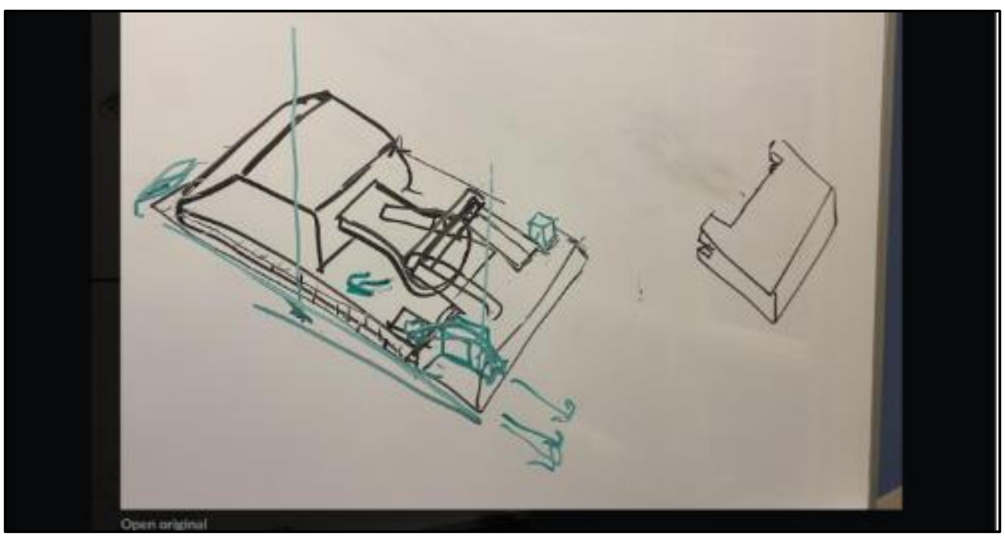


Figure 1: The initial design with lever arm

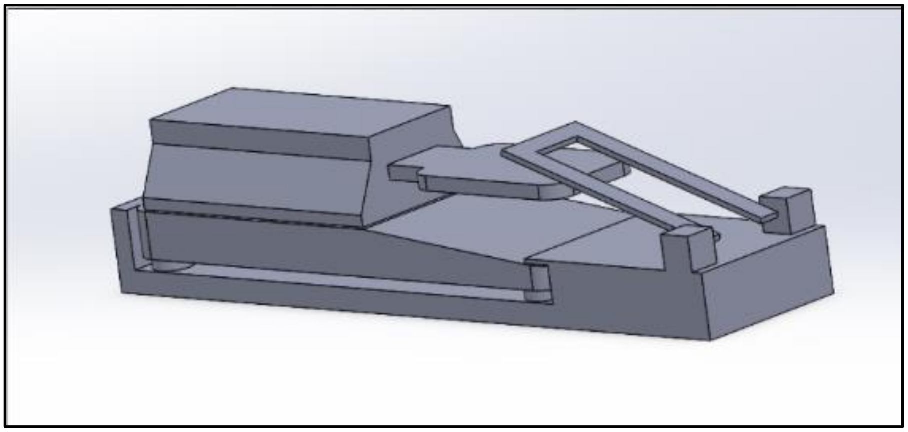

Figure 2: SOLIDWORKS model for the pedal, base, and lever arm

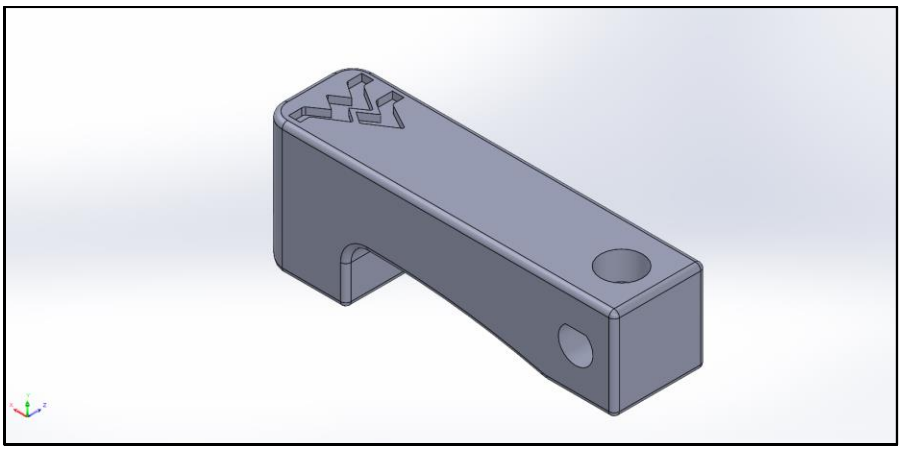

Figure 3: SOLIDWORKS model for the lever arm to be 3D printed

It was also decided that the lever arm would be as short as possible due to the torque being lower at the tip when the length is longer. Plans were also made to add initial weight to the arm at the tip in some way so that the pedal would be on the verge of being pushed down, with the force from the motor being enough to finally create enough total force to enable the pushing motion. As soon as the motor would rotate the other direction once the button was released, the piano pedal should also move up to its original position.

There was also a second design that was not chosen but was in the talks when the group was brainstorming designs. This design would use a motor attached to a base above the foot pedal to extend a rod and push down onto the foot pedal directly. However, concerns over how quickly the rod could extend prevented this design from being chosen. This design can be seen in **Figure 4**.

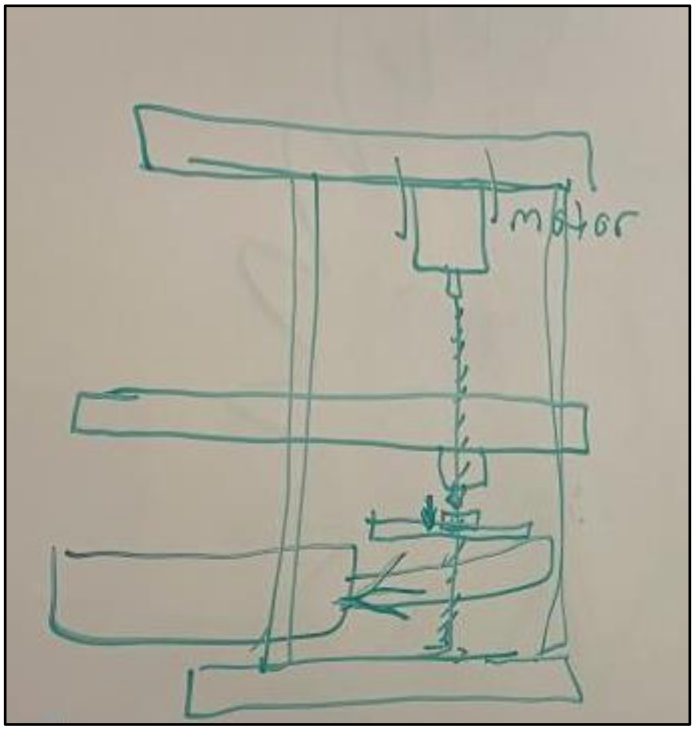


Figure 4: The second design with an extending rod

Two additional design choices that were not implemented due to time constraints and debugging issues were the ability for the button to connect with the Arduino via Bluetooth, and for the device to be powered by battery. The Bluetooth capability wasn’t implemented due to time constraints, and the battery powered Arduino wasn’t implemented due to debugging issues that could not be solved before the due date. Additionally, since this is just a proof of concept, the button was not designed to be put into the mouth, and is simply attached to the breadboard and pressed down on. If this project were to be actually implemented, it would be simple enough to design a comfortable mouthpiece for the button and also implement bluetooth functionality. Finally, the final design we had come up with which can be seen in **Figure 5** had to be scrapped since we discovered too late that the motor did not provide enough torque to push down on the pedal, and was instead stalling and losing steps. Instead, we disassembled a smaller piano pedal and pushed down onto the switch directly.

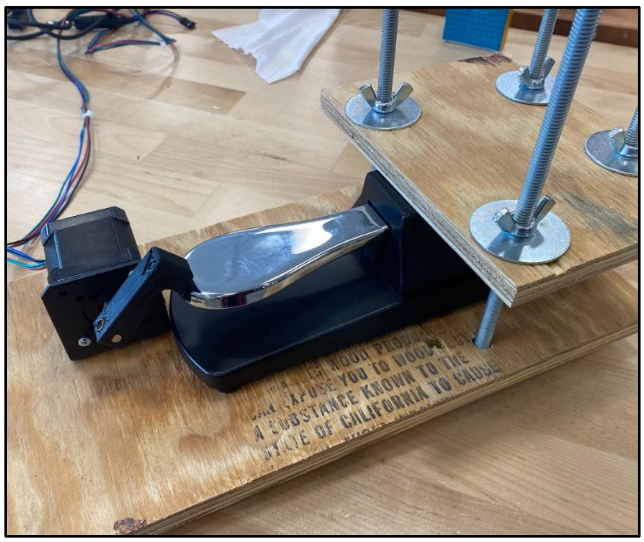

Figure 5: The intended final design prototype

The code for the Arduino was designed to be easily adjusted with variables and with console print commands for easy debugging. The setup would initialize and reset the pins and motor. The loop would constantly check to see if the button was pressed, and then rotate the motor clockwise or counterclockwise depending on whether the current angle is above or below the minimum or maximum angle variable. This would be done by calling a custom helper function that takes in a rotation parameter and handles both directions of rotation. The helper function works by keeping track of the current step variable, making sure that it can only be a value of 0, 1, 2, or 3, resetting to 0 if overflow occurs, or to 3 if the value falls below 0. The current angle is also adjusted during this step. The function finishes by having multiple if statements that would rotate the motor by adjusting the pins depending on which step that the motor is currently on. A delay of 2 milliseconds in the function was added since it was found through testing that a delay any shorter would result in glitches where the motor would rotate past the initial position when rotating back. This code was initially done in MATLAB, but was redone in the Arduino IDE and can be seen in **Appendix A**.

# 5. Team Management

This team did not have an assigned or elected leader. Team members communicated through a combination of phone group text messages and with a popular messaging software called Discord. Discussions were done to find a time that would work best for all members to meet up and work on the project together.

Table 1: Team member roles and responsibilities

| **Team Member** | **Roles/Responsibilities** |
|---|---|
| Alex Naylor | Preliminary Designs, Technical Report (Background, Objectives), Gantt Chart |
| Dre’ Hodges | Final SolidWorks Design, Manufacturing, Assembly |
| Karl Shaver | Ideation, planning & budgeting, mechanical apparatus design & fabrication, lever arm design & modeling, parts compatibility, inventory & ordering |
| Matthew Hwang | Initial idea/design, foot pedal SOLIDWORKS, MATLAB and Arduino IDE coding, Arduino circuiting, Technical Report (Objectives, Project Description, Team Management, Video) |
| Matthew Lombardoni | Preliminary designs/ideas, Technical Report (Overview, Objectives, Budget, Bonus and Constraints) |

Expected Timeline:

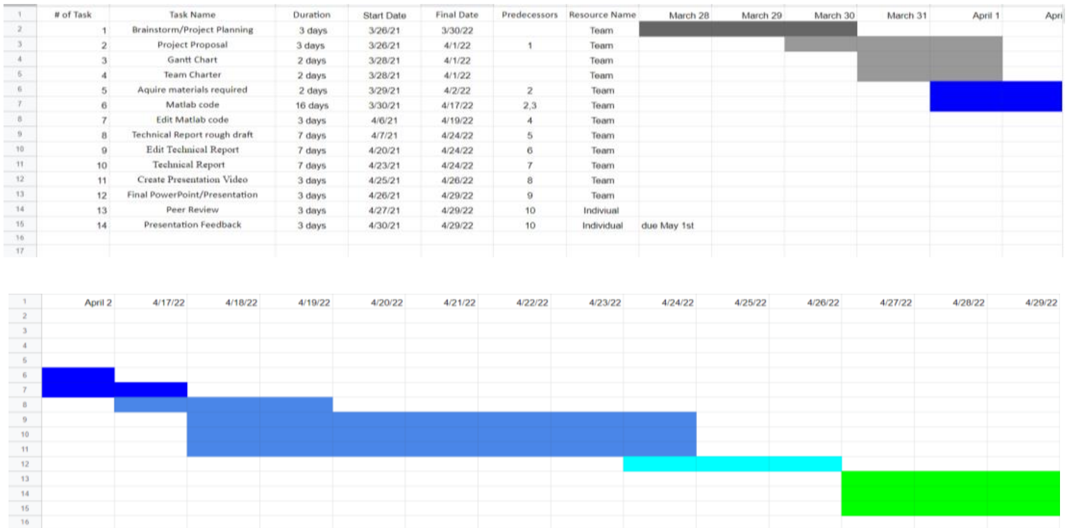


Figure 5: Gantt chart

# 6. Budget

Table 2: Budget

| **Component** | **Cost** |
|---|---|
| Arduino UNO & Shield | $48 |
| Single Stepper Motor | $14 |
| Wood Bases | $13 |
| Mounting Hardware | $16 |
| 3D Printed Lever Arm | $0 |
| Wiring | $8 |
| USB Cable | $6.5 |
| Interface Buttons | $5 |
| **Total** | $110.5 |

# 7. Bonus and Constraints

The team made sure to follow all guidelines set forth in the “Final Project Description” on the WVUEcampus.

There were three main constraints set for this project. The first constraint was the project must be an original thought or work, and if not, similar projects must be cited. The second major constraint was that the group must be able to provide proof of concept of our design. This can be found in our prototype. The third major constraint of this project was the budget cap. The design, including the Arduino Board and 3D printing costs, could not cost more than 125 dollars.

# 8. Conclusion

As evident from the research stated previously, not only is there an important reason for our assistive mechatronic device, there is a market for this device as well. The group was able to design a device that satisfies all the criteria and constraints. The device was completed under the budget, our design was our own idea and our device shows that it can be applied in practical situations for an every-day user.

However, there are several changes we would make to our design if given the chance to make a second prototype. Originally, we designed the motor/lever assembly to fit in front of a standard piano pedal. While this would make the device practical and easy to implement for each user, the stepper motor chosen to press down the lever did not generate enough torque to press down the heavily weighted pedal. To improve this design, we would need to use a Brushless DC motor which generates more torque than the stepper motor originally chosen. Alternatively, to demonstrate proof of concept for our device, we switched out the conventional pedal for a smaller electronic pedal that did not have as much weight to the internal spring. To further ensure that our lever arm could activate the internal switch, the top cover of the pedal was removed. This allowed the lever arm to easily press the switch.

This would give us options on how to build the second prototype. We could implement a one-size-fits-all approach and mount a larger motor/assembly behind the pedal so that it could be implemented on all pianos, i.e Grand, Baby Grand, Upright and Electric Keyboards. The other approach would be to abandon applying the device to "real" pianos and implement the design to electric piano pedals. This would restrict the number of pianos the user of the device could use to just electric keyboards, but considering the types and selections of keyboards available, this is not a bad alternative either. Even though there is room for this idea to grow, there is potential for this design to make an impact on the lives of musicians with physical limitations everywhere.

Video Pitch: https://www.youtube.com/watch?v=dE_TFL2Dm4c

# Bibliography

*Head-activated piano pedal*. https://www.canassist.ca. (n.d.). Retrieved April 29, 2022, from https://www.canassist.ca/EN/main/programs/our-technologies/museum/piano-arts.html

Pereira, G. A. S. (n.d.). Final Project Description. Morgantown, WV; West Virginia University.

Tom, T., Tipsare: Tom is working as a church musician and piano teacher, V. all tips by. (n.d.). *Piano Assistive Device*. Spinalistips. Retrieved April 29, 2022, from https://spinalistips.se/en/tip-piano-assistive-device-1349#:~:text=The%20device%20is%20used%20to,digital%20organs%2C%20pianos%20and%20keyboards

# Appendix A

```
float button_state = 0;
const float step_angle = 1.8;
float current_angle = 0;
int current_step = 0;
const int a_direction = 12;
const int b_direction = 13;
const int a_power = 3;
const int b_power = 11;
const float desired_angle = 45.0;

void setup() {
  pinMode(7, INPUT); // Button Pin
  pinMode(a_power, OUTPUT); // A Power
  pinMode(b_power, OUTPUT); // B Power
  pinMode(a_direction, OUTPUT); // A Direction
  pinMode(b_direction, OUTPUT); // B Direction

  // Initial Reset
  digitalWrite(a_direction, HIGH);
  digitalWrite(a_power, HIGH);
  digitalWrite(b_power, LOW);
  digitalWrite(b_direction, LOW);

  Serial.begin(9600); // Use this line for printing purposes
}

void loop() {
  button_state = digitalRead(7);
  Serial.println(button_state);
  if (button_state <= 0) { // Not pressed
    // Serial.println("not pressed");
    if (current_angle > 0) {
      rotate(0);
    }
  } else if (button_state >= 1) { // Pressed
    // Serial.println("pressed");
```

```
    if (current_angle < desired_angle) {
      rotate(1);
    }
  }
}

// 1 for clockwise, 0 for counterclockwise
void rotate(int rotate_direction) {
  if (rotate_direction == 1) {
    current_step = (current_step + 1) % 4;
    current_angle += step_angle;
  } else {
    current_step -= 1;
    if (current_step < 0) {
      current_step = 3;
    }
    current_angle -= step_angle;
  }
  if (current_step == 0) {
    digitalWrite(a_direction, HIGH);
    digitalWrite(a_power, HIGH);
    digitalWrite(b_power, LOW);
  } else if (current_step == 1) {
    digitalWrite(b_direction, HIGH);
    digitalWrite(a_power, LOW);
    digitalWrite(b_power, HIGH);
  } else if (current_step == 2) {
    digitalWrite(a_direction, LOW);
    digitalWrite(a_power, HIGH);
    digitalWrite(b_power, LOW);
  } else { // current_step == 3
    digitalWrite(b_direction, LOW);
    digitalWrite(a_power, LOW);
    digitalWrite(b_power, HIGH);
  }
  delay(2);
}
```