Affordable Universal Controller for Upper Limb Prosthetics

By

Kathleen Beetner
Leanne Lee
Minwoo Cho

May 2023

Abstract

For this project, we designed an affordable standalone EMG device that detects, processes, and interprets the EMG signal from digit flexion and extension into the corresponding movement of a wooden hand.

This final report starts with the motivations behind the project and then outlines the design process leading up to our final implementation. We then describe the requirements and verifications completed to prove functionality of our EMG device and review the costs of our project. Finally, we conclude with a discussion on the ethics and safety, challenges, and successes our project has achieved.

Contents

1. Introduction

1.1 Problem

Millions of people in the United States alone have lost a limb and this number is expected to double by 2050 [1]. Despite this growing population of people with limb loss, inexpensive prosthetic options in the market are largely unavailable. The average cost of myoelectric prosthetics amounts to almost $19,000 for a partial hand loss and reaches up to $62,000 for a loss up to the shoulder [2]. On top of the cost for the physical prosthetic, costs associated with repairs and replacement parts needed to maintain the structural integrity and compatibility between prosthetic components accumulate over the course of a lifetime, making prosthetics a lifelong financial burden. A financial burden so substantial that 9 out of 10 people worldwide can not access the prosthetics they need [3].

1.2 Solution

The goal of our project is to minimize the costs surrounding prosthetics by providing an affordable, standalone electromyography (EMG) device capable of being removed and used with various mechanical designs of transradial prosthetics. To do so, the execution of our project will prioritize use of inexpensive materials and a modular EMG design where the EMG signal detection, processing, and interpretation into movement can be completed by a single electrical component. Such design allows patients to select new prosthetics and prosthetic parts for replacement and repair without compromising quality function of the prosthetic's electrical component. Additionally, instead of buying an entirely new prosthetic with increasing age and variation in activity, a prosthetic-user only needs to buy and replace the mechanical component of a prosthetic, saving them thousands of dollars in the long-run.

1.3 High-Level Requirements

1. EMG device must be able to return the wooden hand to neutral orientation within 15 degrees of initialized position after at least 10 prosthetic movements.
2. Prosthetic arm should mimic digit extension/flexion with at least 70% accuracy.
3. EMG device must be able to operate with at least 2 unique subjects.

High-level requirement #1 aims to ensure lasting service of our EMG device if it was to be used with an actual prosthetic. Considering how often hands are used in day-to-day life, we need to ensure the EMG device can correctly interpret consecutive EMG signals into hand movements with high accuracy. We want to avoid imprecise movements of the hand due to insufficient signal detection, processing, and/or interpretation.

High-level requirement #2 prioritizes quality service of our EMG device if it was to be used with an actual prosthetic. We want our EMG device to interpret the EMG signal fast enough to minimize the delay between EMG signal detection to movement. An EMG device that consistently lags or buffers in movement can pose as an inconvenience more than a help for the prosthetic-user.

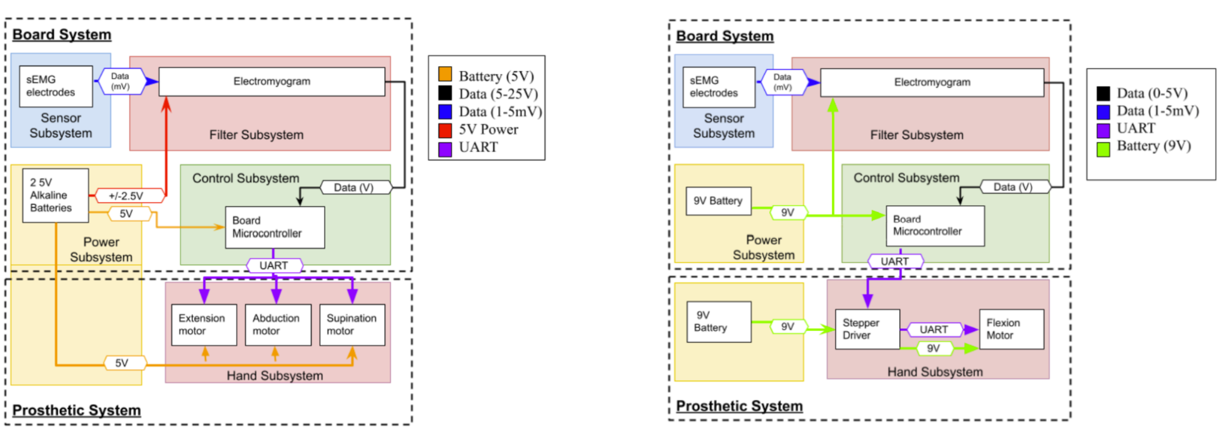
High-level requirement #3 emphasizes the importance of universality in our project. Since part of our goal for this project is to relieve the financial burden of the large population of prosthetic-users in the world, we need to ensure our device can be used across different people with different muscle composition which affects signal detection and processing. We want to avoid creating an EMG device that can be used only by one person with one type of muscle composition.

2. Design

As depicted in Figure 2.1, the block diagram of the current design consists of a board system and prosthetic system. In the board system, we have the sensor subsystem where 3 sEMG electrodes are used for EMG signal detection, the filter subsystem where an electromyogram circuit is responsible for amplification, filtration, and rectification, the control subsystem where an Arduino microcontroller interprets the processed EMG signal as either digit extension or flexion, and the power subsystem where two 9 V batteries supply necessary voltages to the board system. In the prosthetic system, we have the hand subsystem where the stepper motor drives the movement of a wooden hand according to the Arduino microcontroller interpretation of the EMG signal and the power subsystem where one 9 V battery supplies necessary voltages.

Observing both the original and current block diagram depicted in Figure 2.1, we can find 2 main changes: number of ranges of motion and voltage supplies. Originally, we wanted our wooden hand subsystem to be capable of 3 ranges of motion: digit flexion/extension, palmar flexion/extension, and palmar supination/pronation. To do so, we would have needed 7 sEMG electrodes, 3 electromyogram PCBs, and 3 motors attached to the wooden hand. After consulting with our TA and the Machine Shop, though, we concluded 3 ranges of motion would be too complicated for the scope of this course, especially as it requires a complex mechanical component. Therefore, we simplified the hand subsystem to 1 range of motion - digit flexion/extension - which requires 3 sEMG electrodes, 1 electromyogram PCB, and 1 motor, allowing us to focus more on the electrical aspect of the project than the mechanical aspect.

Originally, the design utilized OPA376 operational amplifiers that require ± 2.5V and smaller motors that require 5V. But we noticed the torque of the motor was not enough to smoothly move the wooden hand and had to change our motor to a larger motor with higher torque requiring 9V. To simplify our voltage supplies between subsystems and eliminate the need for a voltage regulator, we switched our operational amplifiers from OPA376 to LM741, allowing everything to be powered with 9V batteries without any voltage regulators.



a. b.

Figure 2.1: (a) The block diagram for the original design is shown.
(b) The block diagram for the current design is shown.

2.1. Sensor Design

2.1.1 Design Procedure

We had 2 sensor options we seriously considered for our sensor subsystem: store-bought electrode buttons or copper electrode pads made by the Machine Shop. When determining which electrode option to use, we found the copper electrodes held slightly less resistance in comparison to the store-bought electrode buttons, thereby minimizing the loss of EMG signal during detection, and is cheaper in cost. With these considerations in mind, we decided to use the copper electrodes. Taking note, we also briefly considered the option of buying disposable, sEMG electrodes but we would have to buy these in bulk as they are for one-time-use only, resulting in higher costs. Since one of our primary goals is to reduce costs, we chose to continue using the copper electrodes.

2.1.2 Design Procedure

The sensor subsystem serves to acquire the sEMG signals of the muscles used for digit flexion and extension. As depicted in Figure 2.2, this subsystem will be implemented as 3 reusable electrodes held at fixed positions along the longitudinal midline of the muscle and will measure the raw EMG signal. The raw EMG signal will then be sent to the filter subsystem signal processing.

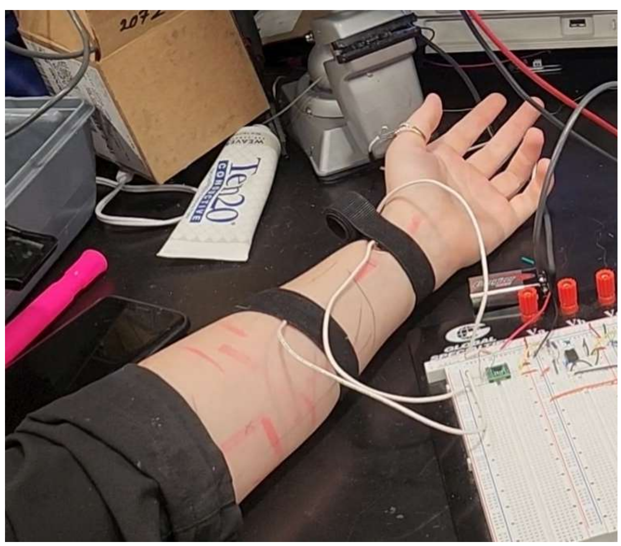


Figure 2.2: 3 copper electrodes held in place via velcro straps feed into the input of the filter subsystem via soldered white wires.

2.2. Filter Design

2.2.1 Design Procedure

There were three potential designs of our filter subsystem, which includes an amplifier, low-pass filter, high-pass filter, and rectifier. The first and original design of the filter subsystem used a differential amplifier to obtain a gain of 5000 V/V, passive first-order low-pass and high-pass filters to obtain cutoff frequencies at about 497 Hz and 4.97 Hz, respectively, and a full-wave rectifier circuit to convert the negative voltage values of the EMG signal to positive values. The issue with this design was the gain of the differential amplifier was too large. The Arduino microcontroller can only take up to 5 V. If a gain of around 5,000 V/V is applied to an input EMG signal expected to be up to 2.5 mV, the output EMG signal would potentially reach 10 V, a voltage too large for the ATmega328P microcontroller. Therefore to avoid damaging the microcontroller, we created a second potential design.

This design used the instrumentation amplifier INA819 to get a gain of 2 V/V, operational amplifiers OPA376 for active Sallen Key low-pass and high-pass filters each providing a gain of 50 V/V, and no rectifier circuit. The rectifier circuit was not needed for this design as the filters were biased with +2.5 V. We assumed that with the lower gain, our signal would be capped at +/- 2.5V. With the 2.5V bias, this would convert the signal to a 0 to 5V minimum and maximum, which is within the specifications of our microcontroller. While the overall gain of this design was now viable, we realized the Sallen Key filter designs with the simplifications we made could not achieve a gain as large as 50 V/V and the +2.5 voltage biasing of our filters resulted in fried operational amplifiers.

These complications led us to our third and final design which uses an instrumentation amplifier INA819 to acquire a gain of around 1516 V/V, operational amplifiers LM741 to create unity gain Sallen Key low-pass and high-pass filters with cutoff frequencies around 490 Hz and 7.23 Hz, respectively, and a full-wave rectifier. The final design allowed us to achieve a gain high enough to work with an EMG signal but low enough to not fry the microcontroller, and the rest of the circuit correctly attenuated frequencies outside of our desired frequency range, as well as prevent any negative voltages from going into our microcontroller. Additionally, changing the operational amplifier from OPA376 which needed ± 2.5 V to LM741 which can be supplied with ±9V, allowed us to simplify the voltages supplied across the entire fil`ter subsystem to just ±9V, eliminating the need for a voltage regulator.

2.2.2 Design Details

Generally, the function of each component of the filter subsystem is adjusted around the EMG signal. First, the amplifier circuit amplifies the EMG signal that has an expected voltage amplitude of around 0 mV - 2.5 mV by 100 V/V - 10,000 V/V [4]. Then, the low-pass and high-pass filters exclude any signal outside of the EMG signal frequency range of 5 Hz - 500 Hz [5]. Finally, the rectifier circuit converts all negative voltage outputs of the EMG signal to positive voltage values as the Arduino microcontroller can only take positive-valued voltages. To obtain desired values for the gain and cutoff frequencies in our current design, we use the following equations:

$$Gain = 1 + \frac{50k\Omega}{R_G} \qquad \text{Eq 1.}$$

Where R_G is gain resistor

$$2 * \pi * F = \frac{1}{\sqrt{R_1 R_2 C_1 C_2}} \qquad \text{Eq 2.}$$

Where F is the desired cutoff frequency, R_1 and R_2 are the resistor values used, and C_1 and C_2 are the capacitor values used

For the amplifier circuit, we set R_G to 33 Ω to obtain a gain of around 1516 V/V. While the capacitor values for both the low-pass and high-pass filters are set to 100 nF capacitors, the resistors for the low-pass filter are 3240 Ω while the resistors for the high-pass filter are 220 kΩ resistors. With these capacitor and resistor values, we obtain a high cutoff frequency at around 491.2 Hz and a low cutoff frequency at around 7.234 Hz.

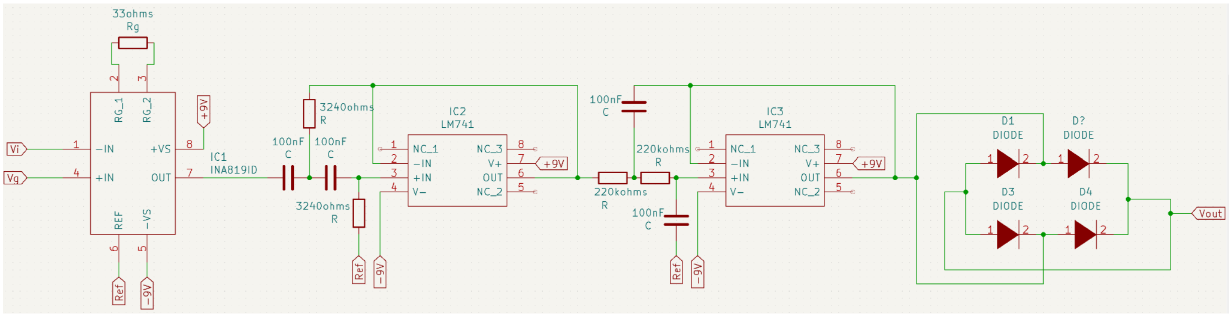


Figure 2.3: Detailed design of the current implementation of the filter subsystem circuit.

2.3. Power Design

The power subsystem for the project uses two 9V batteries to supply +/-9V to the EMG device and a separate 9V battery for the motor that powers the prosthetic hand. Because the prosthetic motor drains the battery significantly faster than the EMG device, keeping this battery system separate allows for easier replacement of the motor battery and keeps the system modular.

The power subsystem also includes a 9V → 5V voltage step down circuit. Although many elements are powered by the +/-9V rails of the battery, both the microcontroller and the stepper driver require 5V input voltage. The step down proved the best solution to this challenge due to its consistent output voltage as opposed to a voltage divider design.

Originally, the OP376 was used in the bandpass filter which necessitated an additional 9V → 2.5V step down. However, these amplifiers were later switched to new amplifiers that ran on the same +/- 9V rails as the instrumentation amplifier and the 2.5V step down was no longer required.

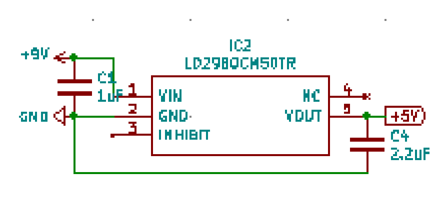


Figure 2.4: Schematic of 9V -> 5V voltage step down

2.4. Control Design

The control system uses an Arduino Uno in combination with the A4988 stepper driver to analyze the input data and send actions to the prosthetic-mounted stepper motor. The stepper driver is current-limited, which means it has built-in protection to prevent the motor from being harmed should something malfunction in the circuit. The A4988 driver allows the stepper motor to be controlled with just two wires from the Arduino: one for the direction of rotation, and another wire pulses to turn the motor one step.

```
void openHand()
{
  // if hand is already open, do nothing
  if(handPos == 1)
    return;
  // set direction of rotation
  digitalWrite(dirPin, LOW);
  // pulse the step pin 50 times with 4.5 ms delay
  for(int x = 0; x < stepsPerRevolution; x++)
  {
    digitalWrite(stepPin, HIGH);
    delayMicroseconds(del);
    digitalWrite(stepPin, LOW);
    delayMicroseconds(del);
  }
  // update position status of hand
  handPos = 1;
}
```

Figure 2.5: The A4988 control algorithm can be seen in the openHand() function. The closeHand() is practically identical except for the 'dirPin' and 'handPos' values.

In Figure 2.5. The 'dirPin' set to low turns the motor counterclockwise, opening the prosthetic hand, and the pin set to high turns the motor clockwise and closes the prosthetic hand. The 'stepPin' is pulsed every 4.5 ms (which is set by 'del') to turn the motor 1.8 degrees, or 1/200th of a full rotation. Each opening and closing motion is 50 pulses (set by 'stepsPerRevolution'), which is equivalent to 90 degrees.

The openHand() and closeHand() functions are called by the main loop. The main loop reads the output from the EMG circuit and keeps a running average of the 50 most recent readings. The value of the analog input pin on an Arduino is an integer between 0 and 1024. 0 means the voltage at the pin is 0V and value of 1024 means the voltage is 5V. And all voltages in between are linearly distributed along the range. The loop function we designed is relatively simple and low on calculations and it runs at roughly 50 Hz, which is a sufficient speed for an EMG controller.

The signal at the output is a steady 0V when the arm is at rest, and spikes to 1-2V when the user's hand is closed, which the Arduino reads as a value at 200-400. By the end, the circuit we built was better than we imagined and the logic to interpret the signal was as simple as a rolling average with a couple thresholds. The thresholds were found manually to be optimal through trial and error and the values at the end can be seen in Figure 2.6.

```
void loop()
{
  // read voltage
  sensorVal = analogRead(sensorPin);
  // update running total by adding most recent value
  // and subtracting oldest value
  total += sensorVal;
  total -= valArr[oldPointer];
  // update oldest value with newest value
  valArr[oldPointer] = sensorVal;
  // calculate running average
  avg = total/50;
  Serial.println(avg);
  // check if running average meets thresholds
  if(avg > 100){
    closeHand();
  }
  else if(avg < 10){
    openHand();
  }
  // update position pointer for oldest value
  oldPointer++;
  oldPointer %= 50;
}
```

Figure 2.6: Main control loop that calculates the running average and opens and closes the prosthetic hand based on the value

2.5. Hand Design

The prosthetic hand system was made by connecting a precision stepper motor to a wooden model of a hand with the help of a bracket. The whole contraption was made by the ECE machine shop. A stepper motor is mounted on the hand to provide the movement required to close and open the fingers. The hand/motor design is built completely separate from the EMG device because the focus of the project is to build a standalone controller that can be used with any number of prosthetics with different designs. Likewise, as the prosthetic hand was only to prove the concept of the design, there was minimal emphasis put into usability of the design as an actual prosthetic, and it only needed to provide visual movement. The motor in Figure 2.7 is the larger motor mentioned in the design procedure section. The new motor has a torque of 302 mNm which was enough to drive the resistive joints in the model hand. The motor turns a bracket that holds fingers together and they are able to mimic the movement of a hand closing and opening with just one stepper motor.

Figure 2.7: The prosthetic hand made of a mounted wooden art model and a stepper motor as fabricated by the ECE machine shop

3. Verification

3.1. Sensor Verification

The sensor system serves to acquire the sEMG signals of the muscles used for digit flexion and extension. However, these signals are expected at a voltage of 1-5mV.

As demonstrated in Figure 3.1. below, although the electrodes were able to accurately detect the input signal, the amount of noise was much more of an issue at smaller voltages. Because the electrodes are passive, the length of the wire between the electrode plate and the circuit is directly correlated with noise. Ideally, this length would be as close to zero as possible to increase the accuracy of the system.

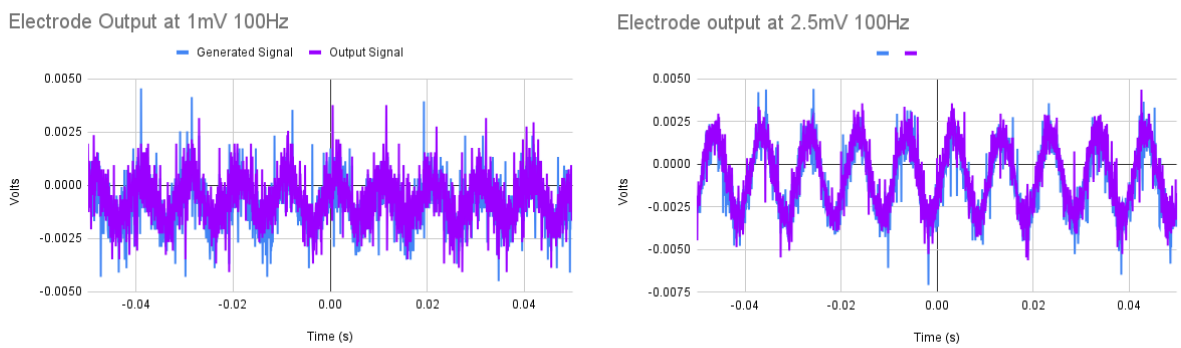


Figure 3.1: (a) Sensor verification at 1 mV
(b) Sensor verification at 2.5 mV

3.2. Filter Verification

For the filter subsystem, the requirements and verifications depicted in Table A.2 of Appendix A consists of 3 primary requirements: gain of 1516 V/V provided by the instrumentation amplifier, cutoff frequencies of 491.2 Hz and 7.234 Hz from the low-pass and high-pass filters respectively, and a unity gain contributed by the filters.

When determining the actual gain of the instrumentation amplifier circuit, we obtained the data organized in Table 2. From this table, we can determine the actual gain of the instrumentation amplifier circuit averages to 1137 V/V which is within ± 500 V/V of the desired gain of 1516 V/V.

Input Sinusoidal wave	Mean Input Amplitude (mV)	Mean Output Amplitude (V)	Gain (V/V)
2.0 mVpp	4.52	5.50	1216
2.1 mVpp	4.72	5.49	1163
2.2 mVpp	4.80	5.50	1145
2.3 mVpp	4.88	5.47	1120
2.4 mVpp	4.99	5.46	1094
2.5 mVpp	5.07	5.49	1082

Table 2: Gains computed for the instrumentation amplifier circuit when sinusoidal waves of varying amplitudes are inputted.

To find the actual cutoff frequencies of our low-pass and high-pass filters, we found the data shown in Table 3. From this table, we find the actual cutoff frequencies where gain is 0.707 in magnitude are 474 Hz for the low-pass filter and 11.3 Hz for the high-pass filter.

Input Voltage (V)	Output Voltage (V)	Ratio of Output to Input (V/V)	Frequency (Hz)
1.96	1.93	0.984	50
1.96	1.82	0.929	25
1.96	1.60	0.816	15
1.96	1.42	0.724	12
1.96	1.39	0.709	11.3
1.92	1.34	0.684	11

a.

Input Voltage (V)	Output Voltage (V)	Ratio of Output to Input (V/V)	Frequency (Hz)
1.96	1.56	0.796	400
1.96	1.49	0.760	450
1.96	1.40	0.714	460
1.96	1.38	0.704	474
1.96	1.34	0.684	480
1.96	1.32	0.673	490

b.

Table 3: (a) Gains of the high-pass filter determined at various frequency values. (b) Gains of the low-pass filter at various frequency values.

While the low-pass filter cutoff frequency is not within 15 Hz of 500 Hz, based on the bode plot of our bandpass filter as seen in Figure 3.2 the roll-off from the cutoff frequencies is not as abrupt as an ideal filter. Therefore having a low cutoff frequency of 11.3 Hz that is slightly above the desired cutoff frequency of 5 Hz and a high cutoff frequency of 474 Hz that is slightly below the desired cutoff frequency of 500 Hz is permissible for our purposes.

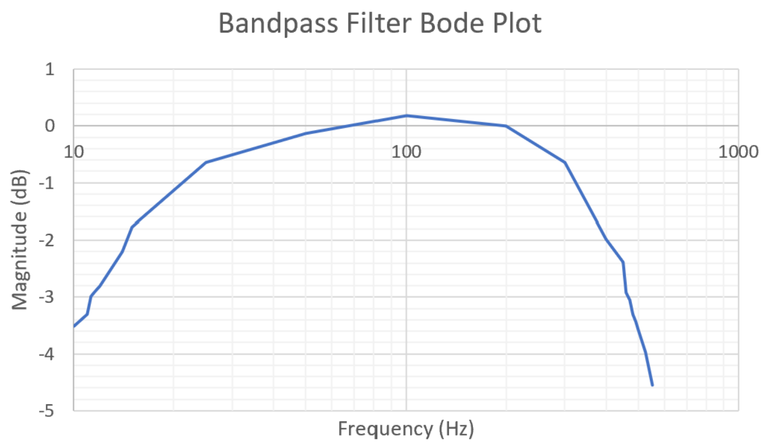


Figure 3.2: Bode plot of the low-pass and high-pass filter.

To find the actual gains of the low-pass and high-pass filters, we gathered the data in Table # and determined the average gain of the low-pass and high-pass filter to be 0.968 V/V and 1.03 V/V respectively, resulting in a bandpass filter gain of 0.994 V/V which is within ± 0.10 V/V of the desired gain of 1 V/V.

a. **Low-Pass Filter**

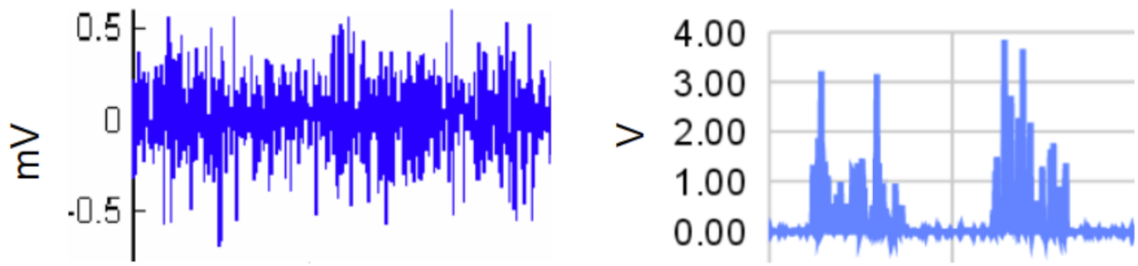
Input Sinusoidal wave (Vpp)	**Mean Input Amplitude (V)**	**Mean Output Amplitude (V)**	**Gain (V/V)**
1.5	1.48	1.38	0.932
2.0	1.96	1.90	0.969
2.5	2.52	2.38	0.944

b. **High-Pass Filter**

Input Sinusoidal wave (Vpp)	**Mean Input Amplitude (V)**	**Mean Output Amplitude (V)**	**Gain (V/V)**
1.5	1.44	1.48	1.03
2.0	1.96	1.97	1.01
2.5	2.40	2.50	1.04

Table 4: (a) Gains of the low-pass filter across 3 different amplitudes of the input sinusoidal wave. (b) Gains of the high-pass filter across3 different amplitudes of the input sinusoidal wave.

While a verification for this component of the filter subsystem is not outlined in the requirements and verifications table, the rectifier circuit had to be tested to ensure it is properly working. To do so, we simply observed the input and output waves of the rectifier circuit using an oscilloscope. If the full-wave rectifier circuit is functioning, we should expect positive and negative-valued voltages of the input EMG signal and all positive-valued voltage values of the output EMG signal during digit flexion. Note there is an EMG signal during digit extension, but due to power dissipation of the diodes used in the rectifier circuit a voltage drop of the EMG signal occurs eliminating the EMG signal during digit extension to. By observation of Figure 3.3, we can see the positive and negative-valued voltages of the EMG signal input and only see positive-valued voltage values of the output EMG signal as desired.



a. b.

Figure 3.3: (a) Raw input EMG signal. (b) Rectified output EMG signal.

3.3. Power Verification

Within the power subsystem, only the 5V step down was used to modify the voltage of the 9V battery inputs. This system needed to have a voltage of 5 +/- 0.5 V with a current of less than 10mA in order to prevent damage to the microcontroller or stepper driver. In Figure 3.4, it is shown that the step down circuit was much more accurate than expected and has a low output current.

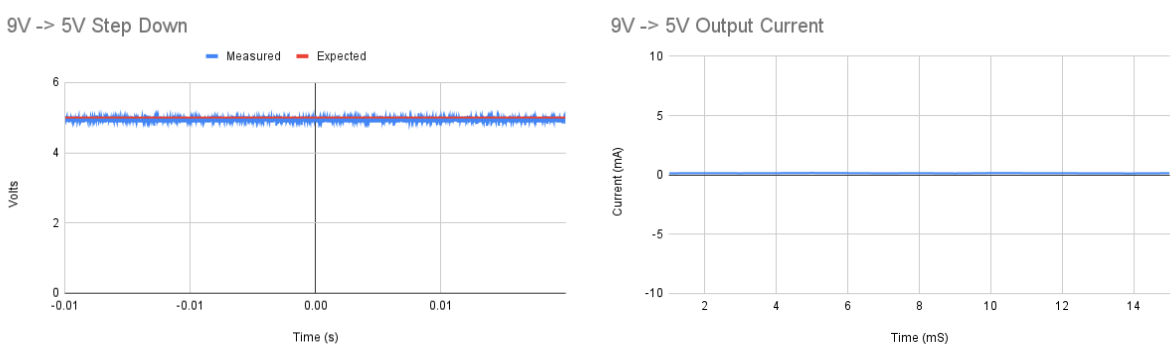


Figure 3.4: (a) Voltage accuracy of the 5V step down circuit
(b) Current output of the 5V step down circuit

3.4. Control Verification

The main purpose of the control system is to correctly classify input signals as open or closed positions of the hand. Essentially, this means determining a threshold between flexion and extension of the forearm muscles.

In Figure 3.5, it can be seen that there is a distinct difference between flexion and extension of the forearm muscles as detected by the EMG circuit. Because the software uses a rolling average with a window of 500mS, the threshold between upper and lower was set at 3mV so any activity registers as the hand closed position. This selection of window width for the average also ensures the program classifies movement within the 2 desired seconds set out by Appendix A.

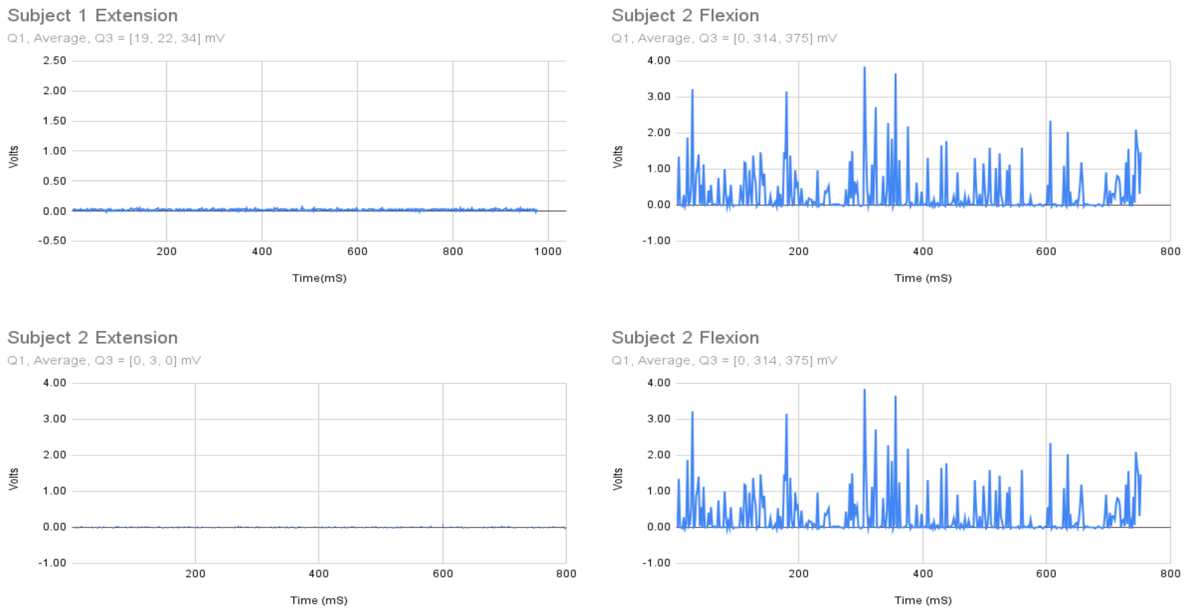


Figure 3.5: Left: Extension samples
Right: Flexion samples

3.5. Hand Verification

The prosthetic is expected to move within 2 seconds and within 15 degrees of the desired angle. In Figure 3.6, it can be seen that both the open, when the direction goes from low to high, and close, when the direction goes from high to low, are performed within a stepper activity window average of 0.28 seconds, well under 2 seconds.

From the datasheet [7], it is noted that each step takes approximately 7000 uS to complete with an accuracy of 5%.

$$\text{Expected step activity} = \frac{7000\ uS}{step} \times 40 \text{ steps} = 0.28\ \ +/- \ 0.06 \text{ seconds} \qquad \text{Eq. 3}$$

Because a full revolution is 200 steps, 40 steps is the same as 72 degrees. A displacement of 15 degrees would be the equivalent of 21% error while the average stepper activity shown in Figure 3.6 is below 10%. This indicates the prosthetic system functions fully as intended.

Figure 3.6: Time it takes to open and close the prosthetic

4. Costs

Description	**Manufacturer**	**Part Number**	**Quant.**	**Retail Price**	**Department Cost**
Wooden hand model	Blick	21650-1004	1	*$9.80*	*$9.80*
100nF Capacitors	Kyocera AVX	04023C104KAT2A-62	4	*$0.74*	*Free*
3.2k Ω Resistor	Vishay	TNPW08053K20BEA	2	*$0.04*	*Free*
320k Ω Resistor	KOA Speer Electronics	RN73H2ATTD3203F100	2	*$0.05*	*Free*
Operational Amplifier	Texas Instruments	LM741	2	*$1.26*	*Free*
Stepper Motor	Pololu	2183-2267-ND	1	*$20.63*	*$20.63*
9V Battery	Duracell	3046-AA-MN1500-ND	3	*$11.09*	*$11.09*
Arduino Uno	Arduino	1050-1024-ND	1	*$24.99*	*Free*
33 Ω Resistor	Vishay	MBA02040C1002FC100	1	*$0.03*	*Free*
Instrumentation Amplifier	Texas Instruments	INA819ID	1	*$6.96*	*$6.96*
PCB			10	*$5.00*	*Free*
Total				***$80.59***	***$41.52***

Table 7: Itemized List of Components and Costs

Additional Costs

The ECE Machine Shop built the prosthetic hand for our project as well as the sEMG electrodes. This included fabricating various metal plates, cutting wooden blocks, and screwing everything together. The estimated time for building all the components is around 5 hours.

We used various equipment and components in the lab including solder, soldering iron, wires, and more to build the PCB as well as test the circuits using the oscilloscopes, function generators, and power supplies provided by the ECE Department for debugging and verifying systems.

Total Cost of Project

UIUC Grainger College of Engineering reported that the average starting salary for ECE graduates was around $80k - $105k [6]. This comes out to about $43 per hour wage. We have put in an estimated 12 hours per week working on this project for 12 weeks. With 3 people working on this project, the total labor cost was $13,932, as shown by Eq. 4.

$$\$43/hour \;*\; 12\; hours/week \;*\; 12\; weeks * 3\; members = \$18{,}576 \qquad \text{Eq. 4}$$

However, there may have been parts of the project that required more work than we estimated and increased the number of hours put in. Putting this factor as 2.5 (worked 30 hours a week), the maximum labor cost could be $55,728.

The total cost for the project will then be $55,728 for labor added with $75.59 for materials, which is $55,803.59

4.1 Schedule

Week	Task	Team
February 20 - February 26	Ordered parts for prosthetic	Minwoo, Kathleen
	Began PCB design	Leanne, Minwoo, Kathleen
	Outlined comprehensive requirements	Leanne, Minwoo
February 27- March 5	Manufactured dry electrodes	Minwoo, Leanne, Kathleen
	Evaluated subcircuits in LTSpice	Minwoo
March 6 - March 12	Finalized PCB design	Leanne
	Assembled breadboard subcircuit	Minwoo, Leanne
	Finished prosthetic assembly	Kathleen, Minwoo
March 13 - March 19	Passed KiCad audit	Leanne, Kathleen, Minwoo
	Created first round sketch of Arduino program	Kathleen
March 20 - March 26	Added revisions to PCB design	Leanne
	Began initial testing for circuit	Kathleen, Minwoo
	Began testing the prosthetic	Kathleen, Leanne, Minwoo
March 27 - April 2	Revisions to PCB design	Kathleen
	Test prosthetic with new motor	Kathleen, Leanne, Minwoo
	Finished writing Arduino program	Minwoo
April 3 - April 9	Revisions to PCB design	Leanne
	Finalize all subsystem verifications	Kathleen
April 10 - April 16	Finalize whole circuit with prosthetic	Kathleen, Minwoo
April 17 - April 23	Prepare final touches for demonstration	Leanne, Kathleen, Minwoo

5. Conclusions

5.1 Accomplishments

Over the course of 12 weeks, we have designed, tested, redesigned, and verified a functional electromyography controller for a prosthetic hand. The high level requirements were met: the precise stepper motor was able to replicate movements dozens of times without drifting from the initialized position, the controller correctly classified the signal with a near 100% accuracy, and the design worked on two people. The final design cost less than $40 in components to build the EMG controller. This includes a retail Arduino Uno; if we redesigned to use the ATmega328p as originally planned, this would bring the cost down to about $25 without losing any functionality. Our ultimate goal was building an affordable option for possible prosthetics and we believe this final cost a great achievement.

5.2 Uncertainties

While we were able to fully implement our project on the breadboard, we were unsuccessful in implementing a functional PCB design. The biggest reason we failed to create a functioning PCB design is insufficient simulation using software like LTspice and breadboard testing. Being a bit behind schedule, we were in a hurry to complete the PCB design which includes the second, potential design as described in 2.2.2 design details of this document. A few of the problems with this potential design, as previously explained in 2.2.2 design details of this document, is the voltage biasing and unattainable gain of the filters. Had we extensively tested the potential filter subsystem design before ordering the PCB, we would have been able to take note of these problems earlier and made adjustments accordingly. Still, granted we now have a fully functioning breadboard implementation of our current filter subsystem, we are confident that if given more time we would have been able to create a functional PCB.

5.3 Future Work

Despite the success we have found with this project, we could always improve the EMG device by integrating a calibration function, implementing a PCB for the filter subsystem, and creating a casing holding all components of our design. To enhance the universality of our project, we could integrate a calibration function where the user wearing the electrodes will take a few moments to perform digit extension and flexion. During this time, the Arduino microcontroller will receive the EMG signal and be able to adjust the voltage thresholds which indicate when the prosthetic hand performs digit extension or digit flexion. This way the user would not have to manually input their EMG signal voltage thresholds and the EMG device can more easily be used across different prosthetic users. Also, as mentioned in the previous section, we could implement the PCB of our current design to better professionality and present ability of our EMG device. Lastly, we can 3D print or invest in some casing to hold the filter subsystem PCB and Arduino microcontroller. This would help organize everything into one component, further emphasizing a modular EMG design.

5.4 Ethical and Safety Considerations

We recognize the importance of technologies and their capability to affect lives throughout the world. Thus, we agreed to commit ourselves to follow the Code of Ethics adopted by the IEEE and uphold the highest standards of integrity, responsible behavior, and ethical conduct while making our design.

I. **We did not store nor share any patient data as to protect the privacy of others and to prevent any conflicts of interest [13].**

With online privacy becoming a bigger concern in recent history, we ensured that no personal information was saved. This was not only ethical but it also aligned with the strict Illinois privacy laws. By not storing information, we were not able to sell any data as a way to fund the project.

II. **We accepted criticism of our work, and were honest in stating claims or estimates regarding our device [13].**

The design and building process revealed flaws in our system and talking with more experienced people, they provided feedback and offered suggestions. We accepted all criticism and tweaked our design to mitigate any weaknesses or oversights on our part. When testing and verifying our system, we were truthful with the data and included all relevant results, even one time outliers as a part of the analysis in the capability of our design.

III. **We treated all persons fairly and strived to ensure the code of ethics is upheld by colleagues [13].**

We all came to this group with different skills and it took our combined efforts to create a successful project. We divided workloads fairly to each member, taking into consideration what we specifically excelled at. We were kind to each other and all persons that we interacted with. We were also each other's witnesses to make sure that the code of ethics was followed and respected.

IV. **We hold paramount the health and welfare of the public [13]**

The prosthetic controller was made affordable with the idea that all people who can benefit from a prosthetic will be able to receive them without the burden of heavy costs.

Safety

One of the most dangerous parts to our design is the battery. They are dense in energy and if used improperly, they could lead to explosions or severe burns. We minimized the chance of injuring others by using dry cell batteries instead of a lithium ion battery as the device is attached to the user. Lithium batteries may overheat and burn the user or even catch fire and cause even more severe injuries. Dry cell batteries are not immune to these issues but greatly decrease the risk of such accidents. To further increase the safety of our design, we also ensured the batteries cannot be short circuited in any case which prevented any excessive wear on the components.

We also avoided using motors that were more powerful than necessary. The prosthetic hand will be close to the user and powerful motors could tear the build apart in the event of malfunction and injure people around it. As our project was only for demonstration purposes, we used weaker motors that can move the necessary components but are not powerful enough to break any other components or cause harm to any body parts.

Our design utilizes low voltages and although the sensors are located on the person's arm, there is no other interaction that may cause serious dangers. The sensors are made of copper which are harmless and overall, our design includes minimal safety concerns. Every member of the group completed the safety training and therefore was knowledgeable in appropriate behaviors in the lab and interacting with equipment, electronic components, and colleagues. Every member was also trained in how to act in the case of possible emergencies.

6. References

[1] "Amputee statistics you ought to know," *Advanced Amputee Solutions*. [Online]. Available: https://advancedamputees.com/amputee-statistics-you-ought-know. [Accessed: 04-Feb-2023].

[2] S. Hubbard and D. K. Blough, "Journal of Rehabilitation Research & Development (JRRD)," *Prosthetic cost projections for service members with major limb loss from*, 15-Jun-2010. [Online]. Available: https://www.rehab.research.va.gov/jour/10/474/Blough.html. [Accessed: 09-Feb-2023].

[3] "Who standards for prosthetics and orthotics," *World Health Organization*. [Online]. Available: https://www.who.int/publications-detail-redirect/9789241512480. [Accessed: 09-Feb-2023].

[5] G. Yang, Z. Sun, Z. Wang, and G. Sun, "Detection and identification of the human surface ELECTROMYOGRAM signal," *IOP Conference Series: Earth and Environmental Science*, vol. 440, no. 2, p. 022016, 2020. [Accessed: 09-Feb-2023].

[4] "(PDF) important factors in surface EMG Measurement." [Online]. Available: https://www.researchgate.net/publication/308365391_Important_factors_in_surface_EMG_measurement. [Accessed: 09-Feb-2023].

[5] "IEEE code of Ethics," *IEEE*, 2020. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 09-Feb-2023].

[6] "Salary averages," *Electrical & Computer Engineering | UIUC*. [Online]. Available: https://ece.illinois.edu/admissions/why-ece/salary-averages. [Accessed: 23-Feb-2023].

[7] "Stepper Motor: Bipolar, 200 Steps/Rev, 42×38mm, 2.8V, 1.7 A/Phase." Pololu, https://www.pololu.com/product-info-merged/2267. Accessed 3 May 2023.

Appendix A: Requirements and Verifications Tables

Table A.1: Sensor Requirements and Verifications

Requirements	Verifications
Requirement 1: Electrodes should be able to measure EMG signals with amplitudes ranging from 1 mV to 2.5 mV.	Using the function generator, input a sinusoidal signal wave with an amplitude of 1mV to the electrode. Connect the electrode to the oscilloscope and note the amplitude of the wave. Repeat this procedure but change the amplitude of the function generator to 2.5 mV. Verify the wave has an amplitude of 1 mV ± 0.1 mV and 2.5 mV ± 0.1 respectively.

Table A.2: Filter Requirements and Verifications

Requirements	Verifications
Requirement 1: The gain of the instrumentation amplifier should be 1516 V/V ± 500 V/V.	Using a function generator, prepare a sinusoidal signal with a frequency of 250 Hz, an amplitude of 2 mVpp, an offset of 0V, and phase of 0°. Using an oscilloscope, connect the sinusoidal signal from the function generator into channel 1 and the input of the amplifier circuit and connect the output of the amplifier circuit to channel 2. Observe and obtain the amplitude of the outputted sinusoidal wave from the oscilloscope. Repeat the procedure for amplitude to 2.1 mVpp, 2.2 mVpp, 2.3 mVpp, 2.4 mVpp, and 2.5 mVpp. Verify the average gain for the range of amplitudes is 1516 V/V ± 500 V/V.
Requirement 2: The bandpass filter must have a low cutoff frequency of 5 Hz ± 15 Hz and a high cutoff frequency of 500 Hz ± 15 Hz.	Using a function generator, prepare a sinusoidal signal with an amplitude of 2.0 Vpp, an offset of 0V, and phase of 0°. Using an oscilloscope, connect the sinusoidal signal from the function generator into channel 1 and connect the output of the amplifier circuit to channel 2. Input the sinusoidal signal into the bandpass filter circuit. To determine the low cutoff frequency, set the frequency of the sinusoidal input signal to 50 Hz. Decrease the frequency of the sinusoidal input wave until $\frac{amplitude\ of\ output\ (mV)}{amplitude\ of\ input\ (mV)}$ has a magnitude of 0.707. Verify the low cutoff frequency is 5 Hz ± 15 Hz. To determine the high cutoff frequency, set the frequency of the sinusoidal input signal to 450 Hz. Increase the frequency of the sinusoidal input

	wave until $\frac{amplitude\ of\ output\ (mV)}{amplitude\ of\ input\ (mV)}$ has a magnitude of 0.707. Verify the low cutoff frequency is 500 Hz ± 15 Hz.
Requirement 3: The high pass filter and low pass filter must provide a gain of 1 V/V ± 0.10 V/V	Using a function generator, prepare a sinusoidal signal with a frequency of 250 Hz, an amplitude of 0.5 Vpp, an offset of 0V, and phase of 0°. Using an oscilloscope, connect the sinusoidal signal from the function generator into channel 1 and the high-pass filter and connect the output of the high pass filter circuit to channel 2. Observe and obtain the amplitude of the outputted sinusoidal wave from the oscilloscope. Repeat for amplitudes 2.0 Vpp and 2.5 Vpp. Verify the high pass filter provides a gain of 1 V/V ± 0.10 V/V. Repeat the procedure for the low pass filter.

Table A.3: Power Requirements and Verifications

Requirements	Verifications
Requirement 1: Voltage of 2.5 +/- 0.25V at less than 10 mA should be supplied from the 2.5V step down circuit	Supply 9V to the 2.5V step down circuit using the digital power supply and measure the output pin on the oscilloscope to verify the voltage output is 2.5 +/- 0.25V. Verify the output current using the digital multimeter.
Requirement 2: Voltage of 5V+/-0.5V at less than 10 mA should be supplied from the 5.0V step down circuit	Supply 9V to the 5V step down circuit using the digital power supply and measure the output pin on the oscilloscope to verify the voltage output is 5 +/- 0.5V. Verify the output current using the digital multimeter.

Table A.4: Control Requirements and Verifications

Requirements	Verifications
Requirement 1: Program should output the hand movements within 2 seconds.	Use the clock of the ATmega328P to measure how long it takes to classify the input signal to hand/wrist movement in software.
Requirement 2: Program should correctly interpret hand movement with at least 70% accuracy.	Use the function generator to alternate between a 0mV and 2mV 100 Hz sine input to the EMG 10 times. Verify and video record the system response to each change in the input signal.

Table A.5: Hand Requirements and Verifications

Requirements	Verification
Requirement 1: The hand must be able to move from one position to another in 2 seconds.	Hard write a open/close loop within software to prompt the prosthetic to change positions 10 times and record the movement with a camera. Use the video to analyze how long it took for the hand to change positions.
Requirement 2: The motors must move the prosthetic hand within 15° of the desired angle.	Hard write an open/close loop within software to prompt the prosthetic to change positions 10 times. Set a camera in a fixed position facing the range of motion. Record the wooden hand moving from neutral position to desired positions. After each completed movement, use the app Angulus to measure the angle of the position. Verify the angle position is within 15° of desired position.