**An EMG based cost effective Prosthetic hand**

**Rajdeep Saha, Abhijit Biswas, Saradwat Sen and Sayani Dhali**

Department of Electronics and Communications Engineering, Supreme Knowledge Foundation Group of Institution, 1, Khan Road, Mankundu, West Bengal, Hooghly 712139, India

**Abstract** In the era of assistive technology, the need for affordable and accessible prosthetic solutions has been roaming around forever. We introduce the solution by making an inexpensive prosthetic arm controlled by electromyographic (EMG) signals obtained from the upper arm muscles of amputees. This project addresses a major global issue which is, in an estimate over 100 million individuals worldwide require prosthetic limbs due to limb damage or amputation. Unfortunately, a large portion of these individuals (especially those from lower economic backgrounds) face problems accessing due to the high cost of traditional prosthetic options, which can range from Rs70,000 to Rs3.5 lakh for imported prosthetic hands. But this project is Remarkably cost-effective. The overall system costs around a few thousand rupees significantly reducing expenses compared to market alternatives. This project solves the solution by using cost-effective technologies in the most efficient ways. The key component in this prosthetic hand is the Muscle EMG sensor. The noise reduction and the amplification of the Bio-Potential signal is done in the IC of the EMG sensor, which helps in precise signal acquisition. By integrating this sensor into the prosthetic hand design, the component can accurately capture muscle signals and translate them into commands for prosthetic hand movements. The brain of this prosthetic hand is a microcontroller, which serves as the central processing unit for signal processing as Band-Pass Butterworth IIR digital filter and responsiveness. Through confined algorithms and real-time data analysis, the microcontroller reads EMG signals captured by the sensor, enabling full control of the prosthetic hand. This integration of hardware and software makes the project's goal of affordability without compromising performance possible. Beyond the technical aspects, this project solves the topmost prioritized problem which is affordability with efficiency. In third world countries where a large portion of people are suffering through poverty, surgery on amputation is nowhere near possible, let alone installing prosthetics. This cheap but effective alternative which costs around 1-3% of the market's product cost, can be affordable to the mass population with disabilities and enhance their quality of life.

**Keywords:** EMG Signal, Bio-Potential Amplifier, Band-Pass Butterworth IIR digital filter

**1. Introduction**

Prosthetics are artificial devices that compensate for the lack of existence of limbs throughout the body. A prosthetic arm replaces the amputated or inoperative upper limbs. This offers individuals who have lost limbs a chance to regain mobility and independence. Now unlike traditional prosthetic arms, a Myoelectric Prosthetic arm is controllable by myoelectric (EMG) signals expediently. This is made achievable by the fact that amputees' neuro-muscular systems continue to function normally even after amputation. After appropriate processing, the leftover signals are utilized and are sufficient to control the arm's movement. [16]. The technical approach of Sudesh Garg, Kiran Rathi, and Amit Kr. Bansal is concentrated on creating a 3D-printed bionic arm that prioritizes human control via EMG sensors, affordability, and usefulness. It is possible to fabricate objects that are lightweight and customizable by using PLA and 3D printing technology. Evaluating for necessary hand movements guarantees usability and imitation of natural hand movements. The objective is to offer amputees an affordable and easily obtainable option that prioritizes controllability and enhanced quality of life. However, noise filtering methods are not specifically mentioned [9]. Furthermore, to separate muscle signals from outside interference in EMG-based systems, noise filtering techniques are crucial for precise bionic arm control as well as improved user experience and safety.

The research aim focuses on decreasing external noise interference that occurs during signal transmission from the brain to the skin of the targeted muscle location. We successfully reduced this interference by utilizing the Butterworth digital filter, which guarantees more precise and more accurate signal capture. This advanced filtering system is indispensable to improve the prosthetic device's overall reliability and effectiveness.

The merit of the following research lies within the cost factor of the prosthetic arm. It has been estimated that there are roughly 0.62 amputees in India per thousand population [17]. With that number of amputees, we are aware that the traditional EMG-controlled prosthetic’s pricings are way beyond affordable for the lower-income groups. This cheap but efficient controllable prosthetic arm will be viable for every individual regardless of their monetary situation.

In addition, we have added a manually operated radical feature that makes it quite effortless for users to hold a pen and write. With the help of this unique feature, people with upper limb disabilities can now write with greater confidence and ease.

**2. Methodology**

The initial step involves using dry surface electrodes to gather EMG data from the patient's hand. For EMG sensors, three electrodes are employed: two for recording the signal and one as a reference. When targeting specific muscles like the forearm, one electrode is placed at the muscle's center, and the other at a distance of 1.5 inches. The reference electrode is positioned on the bony side or the back side of the forearm.

To minimize electrode-to-skin impedance, the electrodes are pre-gelled. Before applying the electrodes, the skin is sanitized. The collected EMG data are then analyzed to detect forearm movement and distinguish between hand states such as flexion and extension.[15]

**2.1 Acquisition of signal using Muscle BioAmp Band**

Three electrodes are used here to capture the EMG signal from the muscle of the forearm. Two of them are bipolar surface electrodes and one is a reference electrode. Then the signal is sent to the Muscle BioAmp Candy through BioAmp Cable and then to the Arduino Nano for further processing.



Figure 1 Electrodes placement for signal acquisition

**2.2 Control of The Prosthetic Hand**

The prosthetic hand was designed to assist the hand's elbow motions by using signals derived from wrist motions. The EMG signals were processed and then delivered as control signals to an Arduino Nano. Wrist flexion is the signal that is obtained if the person closes their hand or flexes their forearm muscle.

This results in the motor rotating in an upward and downward direction, enabling the hands of individuals who are amputated to open and close the hand, and grasp objects. The Arduino Nano's PWM ports are connected to the motor's control input, which is powered by an external source.

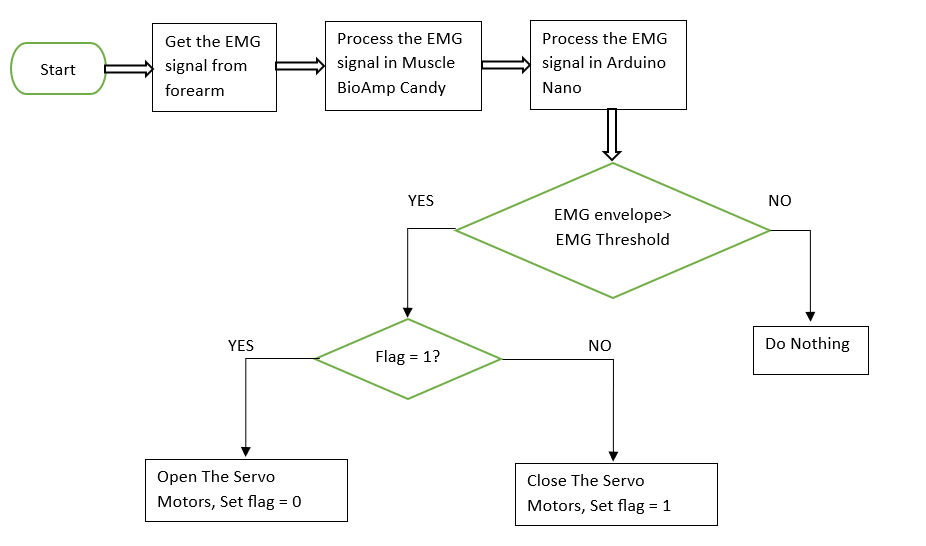
Consequently, the prosthetic hand was successfully operated using EMG signals that were recorded from the healthy hand. [15]

Figure 2 Flow diagram of the process

**2.3 Circuit Diagram**

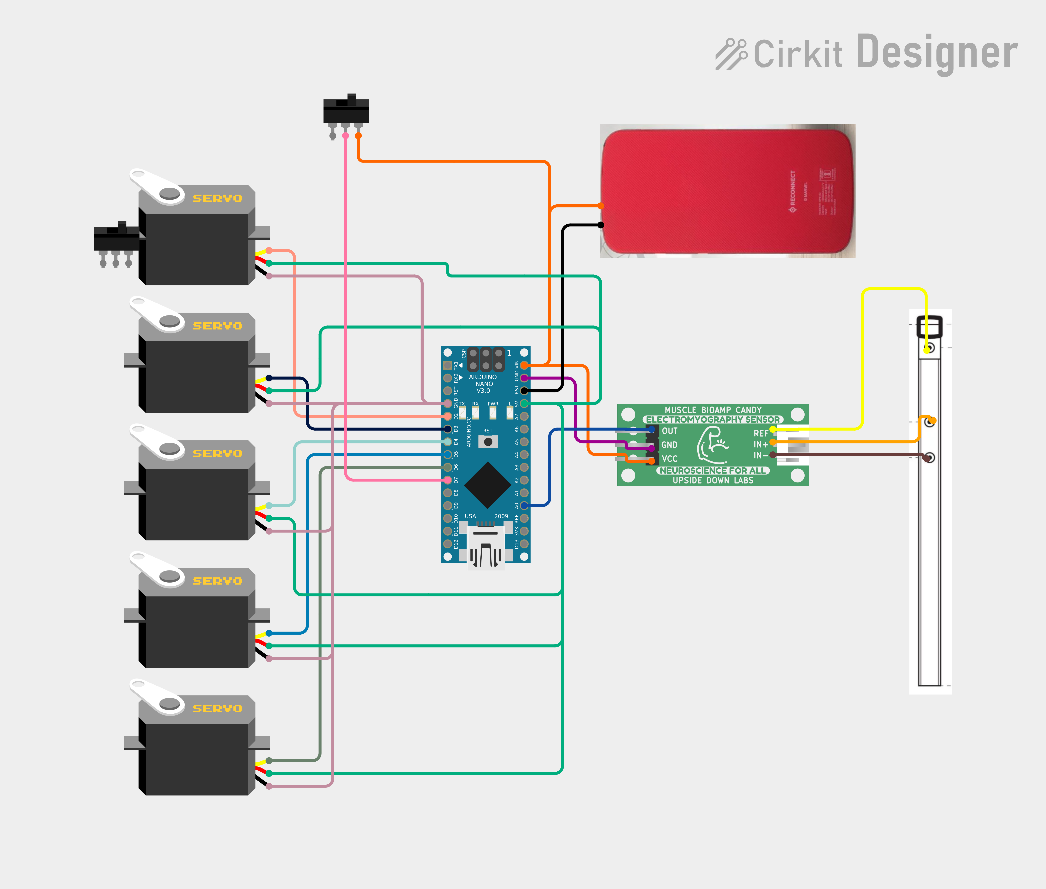


Figure 3 Circuit Diagram of The Prosthetic Hand

Now lets discuss about the circuitry and power management system utilized in our project. The whole system has 5 servo motors for accurate finger movement, an EMG sensor for muscle signal acquisition, and a power bank for portable operation. Each component's connections to the Arduino Nano microcontroller are discussed below in this project.

**Servo Motors (Thumb, Index, Middle, Ring, Pinky):**

* + Each servo motor is connected to the Arduino Nano for control.
  + Signal wires (usually PWM) from each servo are connected to digital pins (D2, D3, D4, D5, D6) of the Arduino Nano.
  + Power for each servo is drawn from the 5V port of the Arduino Nano.
  + Ground connections for each servo are made to the GND port of the Arduino Nano.

**Power Management (Power Bank):**

* + The power bank used is a "Reconnect Power Hub 10000 mAh Power Bank Series 100".
  + The power bank is connected to the Arduino Nano via its USB port, providing power to the entire system.

**Write Switch:**

* + One toggle switch is connected from Vin to D7.
  + By turning it on the hand will enable the user to hold a pen and write with it.

**Muscle BioAmp Candy(EMG Sensor):**

* + The EMG sensor used is a "Muscle BioAmp Candy".
  + The output side of the EMG sensor is connected to the Arduino Nano.
    - The analog output (Out) of the EMG sensor is connected to analog pin A0 of the Arduino Nano.
    - Ground (GND) of the EMG sensor is connected to the GND port of the Arduino Nano.
    - VCC (power) of the EMG sensor is connected to the Vin port of the Arduino Nano.
  + The input side of the EMG sensor is connected to the EMG band.
    - REF, IN+, and IN- ports of the EMG band are connected accordingly to the REF, IN+, and IN- ports of the EMG sensor.

This circuit diagram effectively illustrates the interconnection of components within the

EMG-controlled prosthetic hand system

.

**3. Components and Parts Used**

**3.1 Muscle-BioAmp-Candy**

Muscle-BioAmp-Candy is a candy-sized Electro Myography (EMG) sensor designed to capture precise muscle bio-potential signals. With a compact size and functionality, this device can transform the way of monitoring muscle activity.

It has a fixed gain of x2420 and a Band Pass filter spanning from 72Hz to 720Hz, Muscle-BioAmp-Candy ensures signal clarity and accuracy.

Compatible with standalone ADCs such as the ADS1115 or any microcontroller development board equipped with an ADC, Muscle-BioAmp-Candy offers seamless integration into your existing setup. [1]

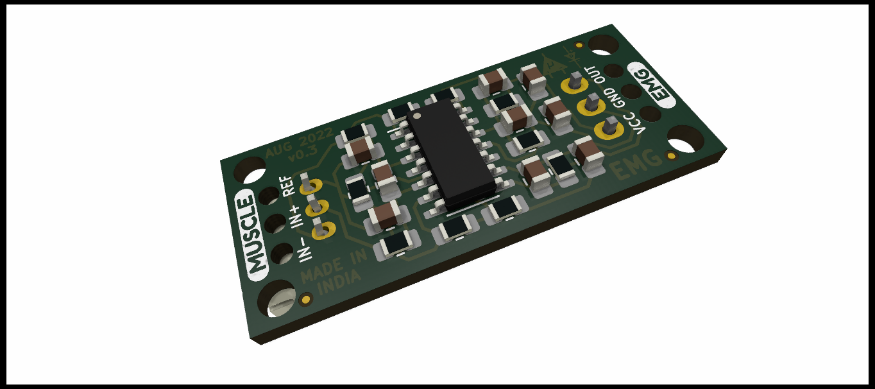


Figure 4 Muscle-BioAmp-Candy [1]

**Schematic Diagram-**

Figure 5 Schematic diagram of Muscle-BioAmp-Candy [1]

**3.2 Muscle BioAmp Band (EMG Band)**

Stretchable in nature, the Muscle BioAmp Band (EMG Band) can be linked to the Muscle BioAmp-Candy via a BioAmp Cable that uses dry electrodes. It makes it simple for you to capture your muscle signals.[14]

**Dimensions:**

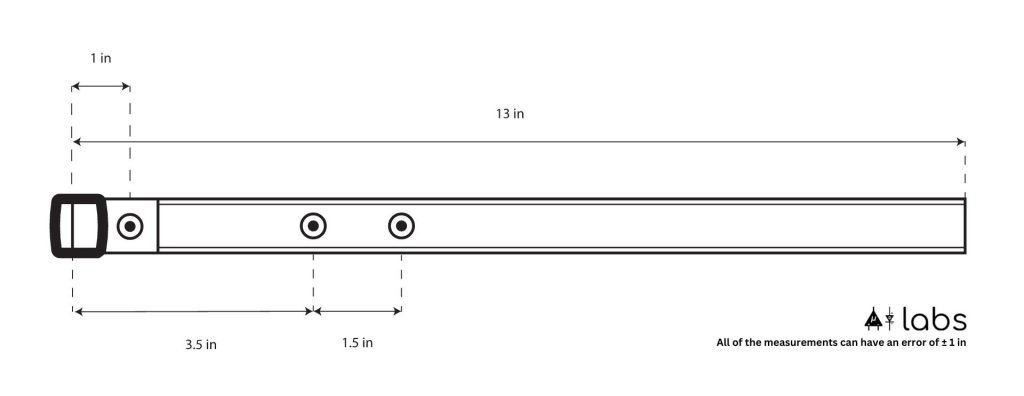


Figure 6 Dimensions of the Muscle BioAmp Band (EMG Band) [14]

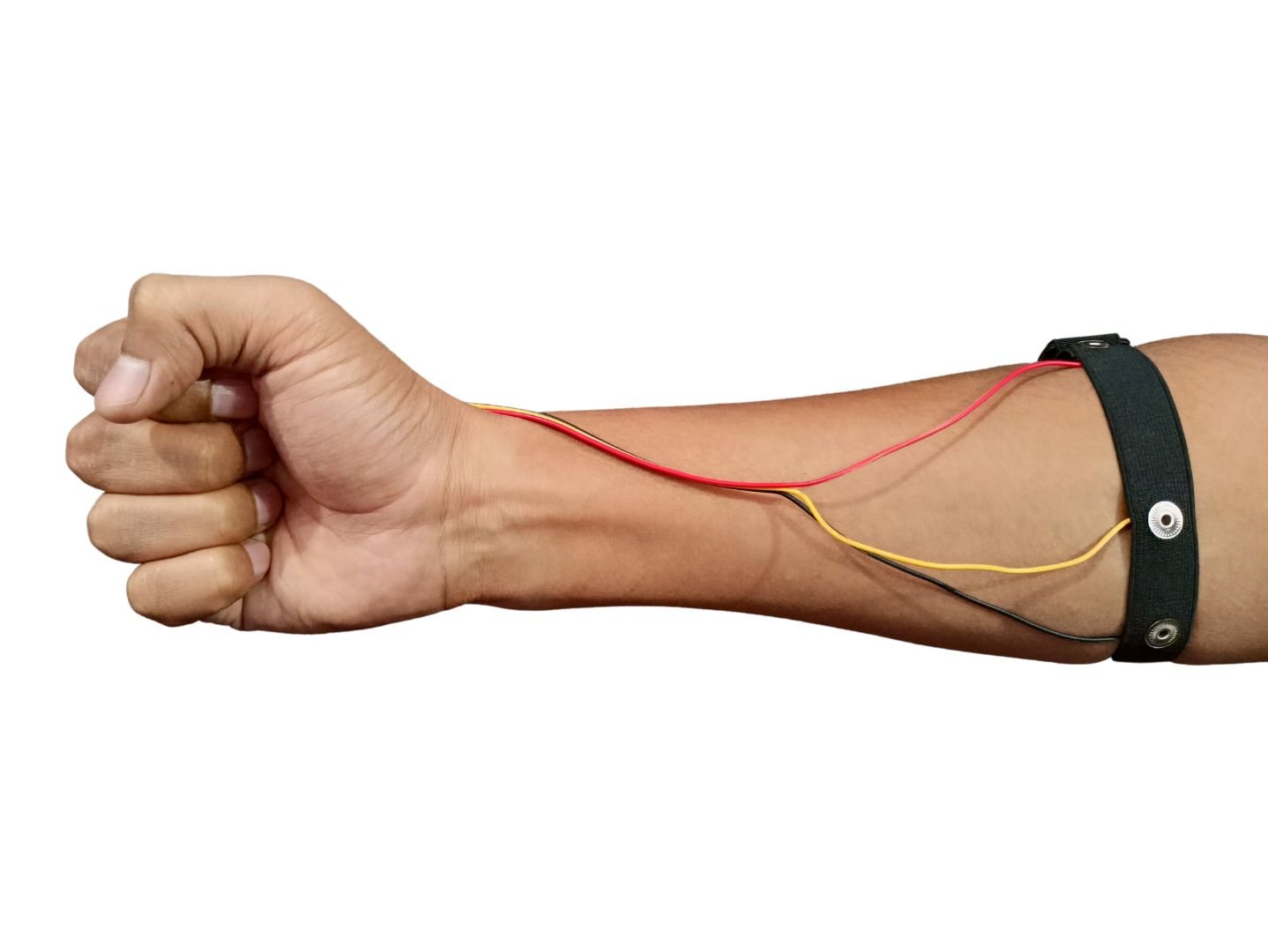


Figure 7 Electrode Placement Example [14]

**3.2** **Arduino Nano**

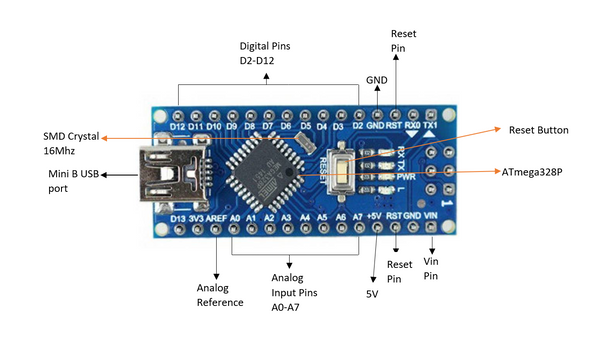


Figure 8 Arduino Nano Board [2]

Arduino Nano is a small, breadboard-friendly development board based on an ATmega 328P SMD package microcontroller and offers the same connectivity and specs as Arduino Uno in a small package. In our project, we are using it because the size is the main factor. After all, the microcontroller needs to be fit into the 3d printed hand. We are supplying the power to the board with a Mini B USB port present on it. [2]

**3.3 SG90 9 g Micro Servo**

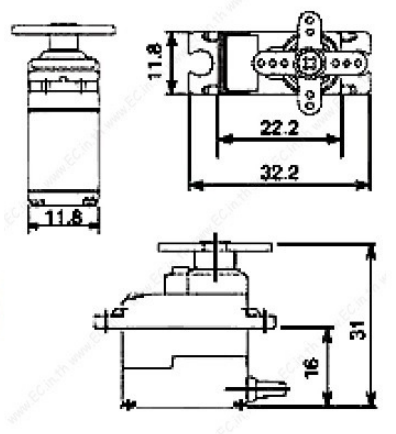
****

Figure 9 SG90 9 g Micro Servo [3] Figure 10 Orthographic view of SG90 9 g Micro Servo [3]

SG90 9 g Micro Servo is tiny and lightweight with high output power. The servo motors are used for the movement of the fingers in the hand. The servo motors usually provide control over the 180° range. This angular position control is controlled by the PWM technique so by varying its duty cycle we can control the angular position of the motor. This servo motor can lift a maximum of 1.6 kg when suspended at a 1cm distance from the shaft. [3]

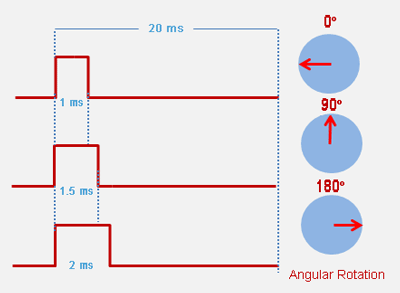
Every 20 ms, the servo examines the pulse. The servo can be rotated to zero degrees by a pulse with a width of 1 ms (1 millisecond), to ninety degrees (the neutral position) by a pulse of 1.5 ms, and to 180 degrees by a pulse of 2 ms.. [13]

Figure 11 Rotation control of SG90 9 g Micro Servo [13]

**3.4 Power Bank**



Figure 12 Reconnect Power Hub 10000 mAh Power Bank Series 100 [4]

Here we are using Reconnect Power Hub 10000 mAh Power Bank Series 100. It has a capacity of 10000 mAh, so it can be used for longer intervals. As the microcontroller, servo, and Muscle-BioAmp-Candy draw very little power on a full charge it lasts for about 10hrs. [4]

**3.5 Fabrication of 3D Printed Hand**

Here we used PLA because PLA is a common 3D printing material to print components. PLA is widely used because of its ease of availability, ecological footprint, and affordability.   
PLA is a biodegradable substance derived from food crops like sugarcane, corn, and jowar. Here are some key characteristics of PLA and the 3D printer [9]:

**Table 1:** Features of PLA[9][10]

|  |  |  |
| --- | --- | --- |
| **S.N.** | **Property** | **Value** |
| 1 | Melting Point | Low (150°C and 180°C) |
| 2 | Thermal Expansion | Low (68 µm/m-K ) |
| 3 | Layer Adhesion | Moderate |
| 4 | Heat Resistance | Low(55–65 °C) |
| 5 | Tensile Strength | High(39.9 MPa to 52.5 MPa) |
| 6 | Compressive Strength | High (48.2 MPa to 62.0 MPa) |
| 7 | Dimensional Accuracy | High |

**Table 2:** Features of 3D Printer[11]

|  |  |
| --- | --- |
| **Model** | **Ender 3** |
| Physical dimensions | (w) 40 cm x (d) 22 cm x (h) 46 cm |
| Maximum printing area | (w) 20 cm x (d) 22 cm x (h) 22 cm |
| Wire diameter | 0.2mm |
| Nozzle diameter | 0.4mm |
| Platform temperature | ~100 °C |
| Nozzle printing temperature | ~200 °C |
| Cooling method | Air Cool |
| Motor drive | Stepper Motor Drive |

Now let us discuss the specification of the 3D Printed Hand

**Table 3:** Technical Specification of 3D Printed Parts[12]

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Name of the parts** | **No. of joints/parts** | **Length (cm)** |
| 1 | Thumb finger | 2 | 5.6 |
| 2 | Index finger | 3 | 7.5 |
| 3 | Middle finger | 3 | 8.2 |
| 4 | Ring finger | 3 | 6.8 |
| 5 | Pinky finger | 3 | 5.4 |
| 6 | Palm | 1 | 9 |
| 7 | Wrist | - | 11.5 |
| 8 | The diameter of the end of the wrist | - | 9.3 |
| 9 | The total length of the arm | - | 28.7 |

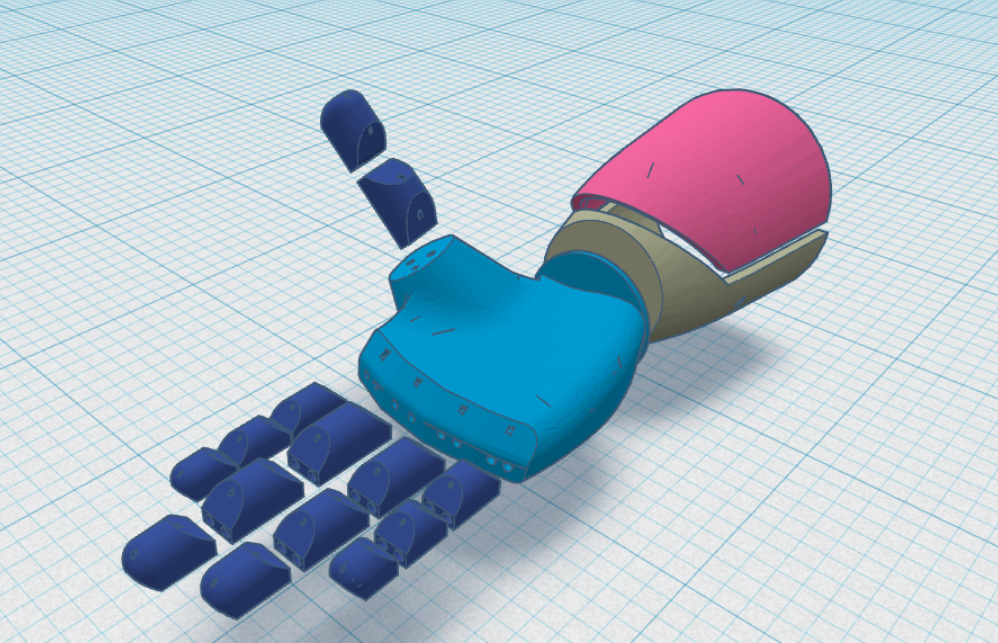
****

Figure 13 Isometric projection of the 3D Printed Hand [11]



Figure 14 After assembling all the parts of the hand

**4 Algorithms and Their Detailed Analysis**

**4.1 Algorithm of the Arduino Code**

**Initialization:**

* + Set up the system parameters including the header files, sampling rate, baud rate, and pin configurations.
  + Initialize the communication interfaces (e.g., Serial) for monitoring and debugging purposes.

**Setup:**

* + Attach the servo motors to their respective pins and set their initial positions.
  + Define the WritePin
  + Configure the EMG sensor input pin and any auxiliary pins required for system operation.
  + Define the threshold voltage (different for each person)

**Main Loop:**

* + Check WritePin is High or Low

If High, Rotate the thumb, index, and middle finger as such it can hold a pen for writing purposes

If Low, Continue to the next steps

* + Continuously sample the EMG signal at the defined sampling rate.
  + Apply a band-pass Butterworth IIR digital filter to the raw EMG signal to extract relevant muscle activity within the desired frequency range.
  + Compute the EMG envelope using an envelope detection algorithm to estimate the magnitude of muscle activation.

**Gesture Recognition:**

* + Compare the normalized envelope value with a predefined threshold to determine muscle activation and gesture recognition.
  + Implement a hysteresis mechanism to prevent rapid toggling of the hand due to noise or minor fluctuations in muscle activity.
  + Define specific thresholds for opening and closing gestures based on individual user characteristics and preferences.

**Servo Control:**

* + Based on the detected gesture:
    - If the muscle activation exceeds the closing threshold:
      * Close the hand by rotating the servo motors to the predefined closed position.
    - If the muscle activation falls below the opening threshold:
      * Open the hand by rotating the servo motors to the predefined open position.
  + Implement a gesture delay to prevent rapid and unintended toggling of the claw in response to minor fluctuations in muscle activity. [1]

**4.2 Algorithm for Envelope Detection-**

1. Subtract the previous EMG signal value from the sum.
2. Add the absolute value of the current EMG signal to the sum.
3. Store the absolute value of the current EMG signal in the circular buffer at the current index.
4. Update the data index to point to the next position in the circular buffer, wrapping around to the beginning if necessary.
5. Compute the average of the EMG signal values in the circular buffer by dividing the sum by the buffer size.
6. Multiply the average by 2 to scale the envelope signal.
7. Return the computed envelope signal.

**4.3 Algorithm of EMG Band Pass Filter**

Algorithm for the Band-Pass Butterworth IIR digital filter:

1. **Initialization**:
   * Initialize the state variables **z1** and **z2** for each filter section to zero.
2. **Filtering Process**:
   * For each input sample:
     + For each filter section:
       - Calculate the intermediate variable **x** using the difference equation of a second-order IIR filter.
       - Update the output using the calculated **x** value and the previous state variables.
       - Update the state variables **z1** and **z2** for the next iteration.
3. **Output**:
   * Return the filtered output.

Here's a breakdown of the steps within the filtering process:

* For each filter section:
  1. Calculate the intermediate variable **x** using the difference equation:

x = input - a1 \* z1 - a2 \* z2

where **input** is the current input sample, **z1,** and **z2** are the previous state variables, and **a1** and **a2** are the filter coefficients.

* 1. Update the output using the calculated **x** value and the previous state variables:

output = b0 \* x + b1 \* z1 + b2 \* z2

where **b0**, **b1**, and **b2** are the filter coefficients for the output.

* 1. Update the state variables **z1** and **z2** for the next iteration:

z2 = z1 z1 = x

Repeating these steps for each input sample, obtained the filtered output of the Band-Pass Butterworth IIR digital filter. [1] [5] [6]

**4.4 General Difference Equation of** **Band-Pass Butterworth IIR digital filter**

The code implements a Band-Pass Butterworth IIR digital filter using second-order sections (biquads). Each biquad represents a second-order IIR filter section. Let's break down the mathematical expression for each biquad.

The general difference equation for a second-order IIR filter is:

*y*[*n*]=*b*0​*x*[*n*]+*b*1​*x*[*n*−1]+*b*2​*x*[*n*−2]−*a*1​*y*[*n*−1]−*a*2​*y*[*n*−2]

Where:

* *y*[*n*] is the output at time *n*
* *x*[*n*] is the input at time *n*
* *x*[*n*−1] and *x*[*n*−2] are the previous input samples
* *y*[*n*−1] and *y*[*n*−2] are the previous output samples
* *b*0​,*b*1​,*b*2​ are the feedforward (numerator) coefficients
* *a*1​,*a*2​ are the feedback (denominator) coefficients

Let's express each biquad in the code as difference equations:

**First Biquad:**

z1\_1[n] = x[n] - 0.05159732 \* z1\_1[n-1] - 0.36347401 \* z2\_1[n-1] z2\_1[n] = z1\_1[n-1] y\_1[n] = 0.01856301 \* z1\_1[n] + 0.03712602 \* z2\_1[n] + 0.01856301 \* z2\_1[n-1]

**Second Biquad:**

z1\_2[n] = y\_1[n] - (-0.53945795 \* z1\_2[n-1] - 0.39764934 \* z2\_2[n-1]) z2\_2[n] = z1\_2[n-1] y\_2[n] = 1.00000000 \* z1\_2[n] - 2.00000000 \* z1\_2[n-1] + 1.00000000 \* z2\_2[n-1]

**Third Biquad:**

z1\_3[n] = y\_2[n] - (0.47319594 \* z1\_3[n-1] - 0.70744137 \* z2\_3[n-1]) z2\_3[n] = z1\_3[n-1] y\_3[n] = 1.00000000 \* z1\_3[n] + 2.00000000 \* z1\_3[n-1] + 1.00000000 \* z2\_3[n-1]

**Fourth Biquad:**

z1\_4[n] = y\_3[n] - (-1.00211112 \* z1\_4[n-1] - 0.74520226 \* z2\_4[n-1]) z2\_4[n] = z1\_4[n-1] y\_4[n] = 1.00000000 \* z1\_4[n] - 2.00000000 \* z1\_4[n-1] + 1.00000000 \* z2\_4[n-1]

Where:

* *z*1*i*​[*n*] and *z*2*i*​[*n*] are the state variables for the *i*-th biquad at time *n*
* *x*[*n*] is the input at time *n*
* *yi*​[*n*] is the output of the *i*-th biquad at time *n*

This set of equations describes the mathematical expression for the given Band-Pass Butterworth IIR digital filter implemented in the provided code. Each biquad contributes to the overall filter response, and the output of one biquad serves as the input to the next biquad in the chain.[5] [6] [7][8]

**5. Testing and Results**

**5.1 Simulation of BPF in MATLAB**

Here we have simulated the Band-Pass Butterworth IIR digital filter in MATLAB. First we have shown the unfiltered EMG signal acquired from the hand which consists of different frequencies (45Hz, 60Hz, 90Hz, 100Hz and 160Hz) and noise.

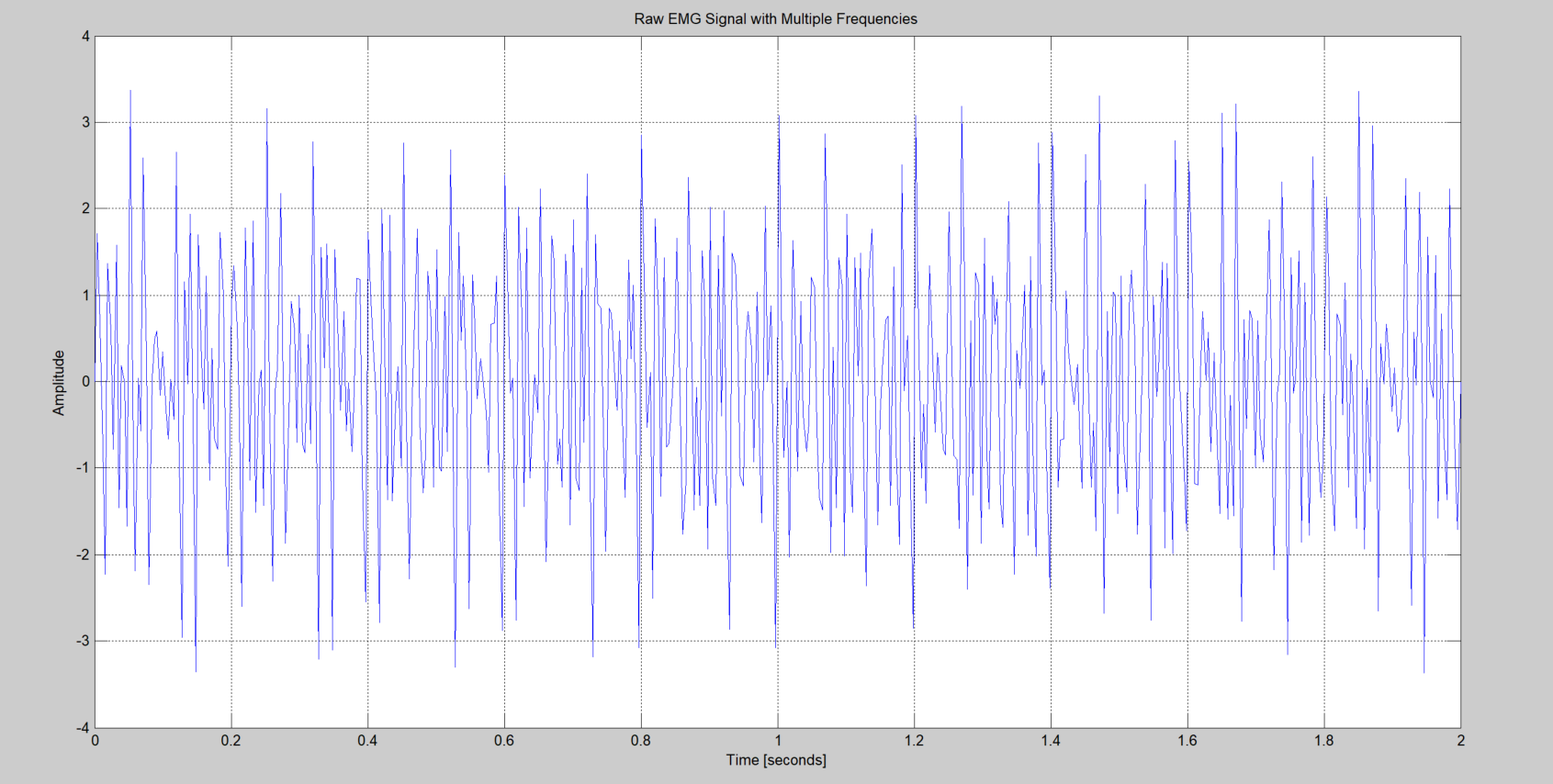


Figure 15 Raw EMG signal with multiple frequencies of 45Hz, 60Hz, 90Hz, 100Hz and 160Hz [5][6] [7][8]

And here we have shown the EMG signal after passing through the Band-Pass Butterworth IIR digital filter.

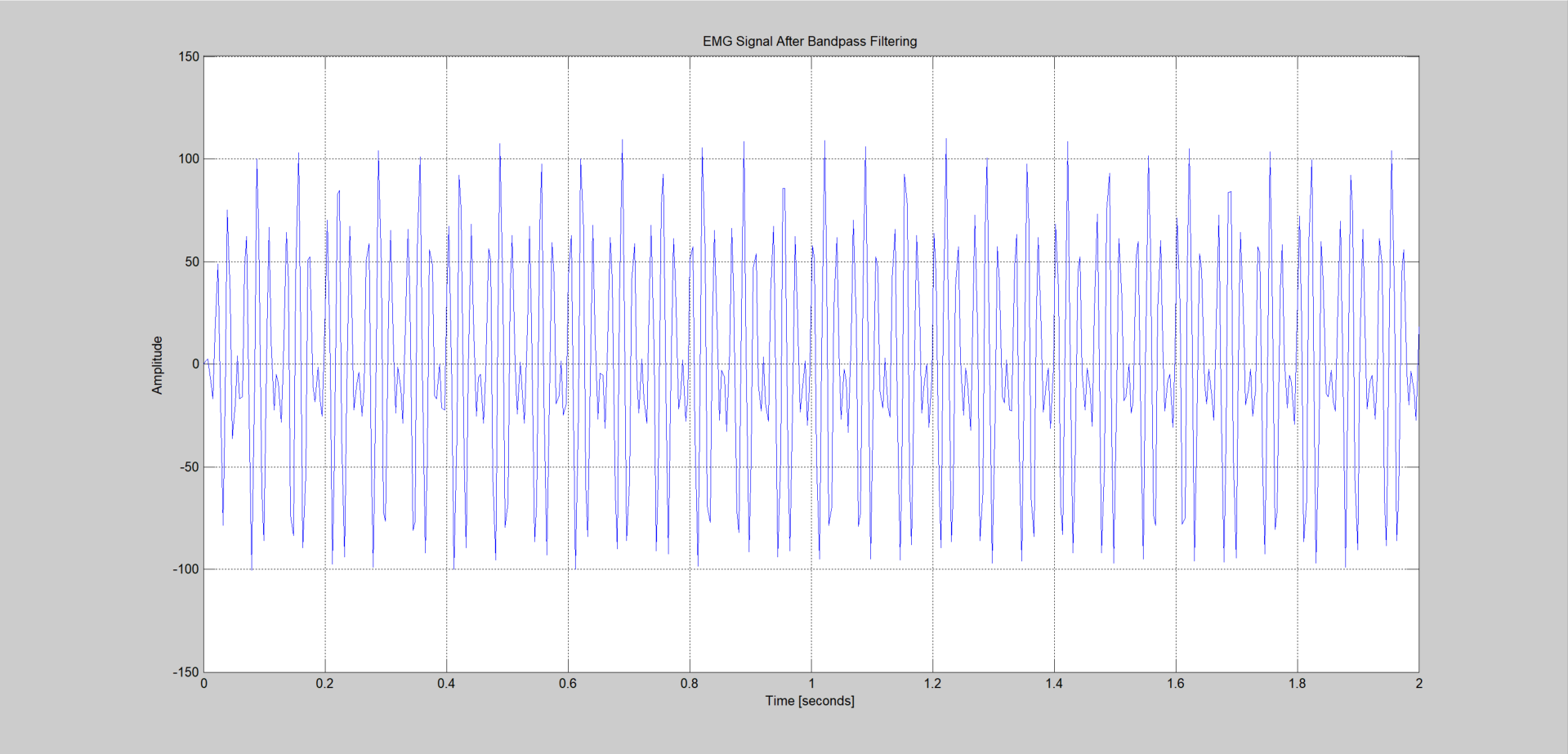


Figure 16 Filtered EMG Signal after Band Pass Filter consisting of frequencies 90Hz and 100Hz [5][6][7][8]

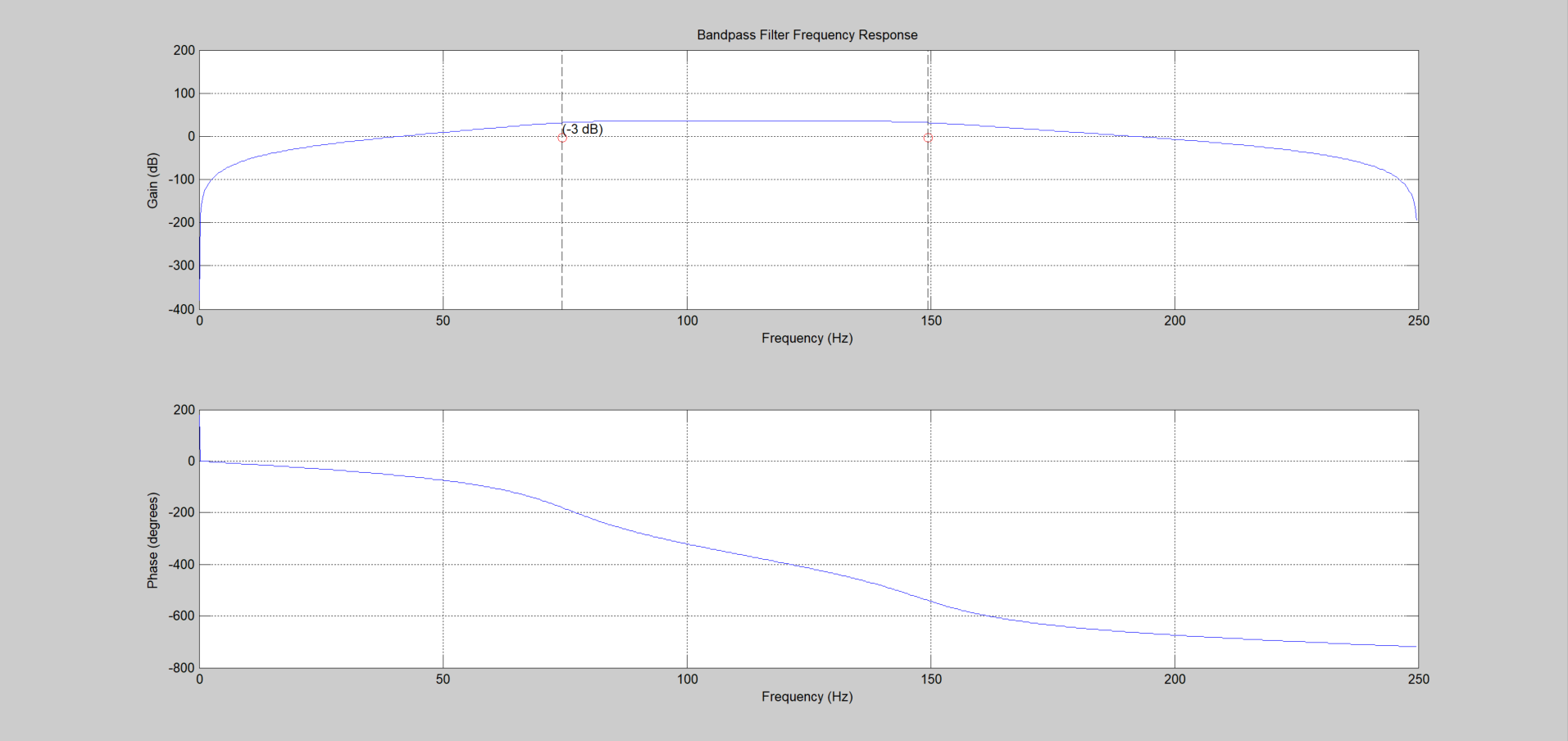
And this is the frequency response of the Band-Pass Butterworth IIR digital filter with -3db point which has a sampling rate: 500.0 Hz, cutoff frequency: [74.5, 149.5] Hz[5][6] [7][8]

Figure 17 Frequency response of the Band-Pass Butterworth IIR digital filter with -3db point [5][6] [7][8]

**5.2 EMG signal, Detected Envelope**

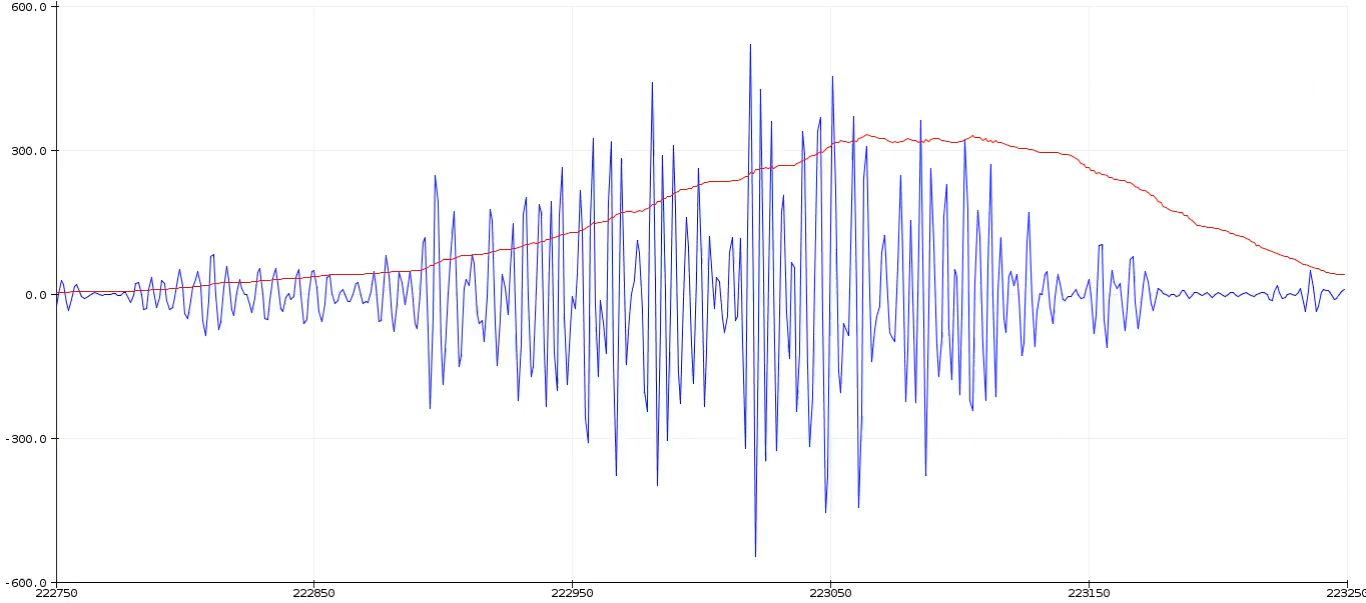


Figure 18 EMG Signal After Filter with The Detected Envelope [9]

**5.3 Testing of the Prosthetic Hand**

We tested the prosthetic hand with 10 people who have healthy hands and we got these accurate results. We tested 3 things here first how accurately it can grab an object then how accurately it can close and open the hand.

To have better gripping capability we used hot glue to have a better gripping capability in the most probable contact points when trying to grab any object.

Here are some images that shows how the prosthetic hand looks when at rest and flexed.

Figure 20 Hand at Flexed (Close) Position



Figure 19 Hand at Rest (Open) Position

To test the accuracy of grabbing any object we have used a round object and a cylindrical object as shown in the Figure 22 and Figure 21.



Figure 22 Grabbing one Cylindrical Object

Figure 21 Grabbing one round object

Now here are the results that we got after testing it on 10 people with heathy hands.

Figure 23 Accuracy of Closing and Opening the Prosthetic Hand

Figure 24 Accuracy of Grabbing Any Object With the Prosthetic Hand

**6. Conclusion**

**References**

1. GitHub - upsidedownlabs/Muscle-BioAmp-Candy: A candy size ElectroMyoGraphy (EMG) sensor for precise muscle bio-potential signals recording. (n.d.). GitHub. <https://github.com/upsidedownlabs/Muscle-BioAmp-Candy>

2. Arduino Nano  - Soldered (Without Cable). (n.d.). QuartzComponents. <https://quartzcomponents.com/products/arduino-nano-v3-0-soldered-without-cable?_pos=2&_sid=2bf3b65a9&_ss=r>

3. SG90 Servo Motor Datasheet. (n.d.). https://components101.com/. Retrieved March 24, 2024, from <https://components101.com/sites/default/files/component_datasheet/SG90%20Servo%20Motor%20Datasheet.pdf>

4. Reconnect Marvel Spiderman 10000mAh PowerBank, Li-Po battery, 1 USBB output port, Micro USB & Type C dual input, BIS approved, Mobile Accessories- DPB102 SM Online at Best Prices in India - JioMart. (n.d.). JioMart. <https://www.jiomart.com/p/electronics/reconnect-marvel-spiderman-10000mah-powerbank-li-po-battery-1-usbb-output-port-micro-usb-type-c-dual-input-bis-approved-mobile-accessories-dpb102-sm/590041746>

5. scipy.signal.butter — SciPy v1.12.0 Manual. (n.d.). <https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.butter.html>

6. Iyer, S. V. (2020, November 30). Digital Filter Design using Python for Power Engineering Applications. Springer Nature. <http://books.google.ie/books?id=jx8MEAAAQBAJ&pg=PA125&dq=https://docs.scipy.org/doc/scipy/reference/tutorial/signal.&hl=&cd=2&source=gbs_api>

7. Verma, N. K., & Ghosh, A. K. (2018, July 31). *Computational Intelligence: Theories, Applications and Future Directions - Volume I*. Springer. <http://books.google.ie/books?id=i_pmDwAAQBAJ&pg=PA94&dq=https://en.wikipedia.org/wiki/Butterworth_filter&hl=&cd=2&source=gbs_api>

8. *Digital filter*. (2023, December 11). Wikipedia. https://en.wikipedia.org/wiki/Digital\_filter#Direct\_form\_II

9. Garg, S., Rathi, K., & Bansal, A. K. (2023, April 1). Development of 3D Printed Bionic Arm Controlled by EMG Sensor. SKIT Research Journal, 13(1), 13–18. <https://doi.org/10.47904/ijskit.13.1.2023.13-18>

10. Gao, G., Xu, F., Xu, J., & Liu, Z. (2022, October 10). Study of Material Color Influences on Mechanical Characteristics of Fused Deposition Modeling Parts. *Materials*, *15*(19), 7039. https://doi.org/10.3390/ma15197039

11. Hsueh, M. H., Lai, C. J., Wang, S. H., Zeng, Y. S., Hsieh, C. H., Pan, C. Y., & Huang, W. C. (2021, May 27). Effect of Printing Parameters on the Thermal and Mechanical Properties of 3D-Printed PLA and PETG, Using Fused Deposition Modeling. *Polymers*, *13*(11), 1758. https://doi.org/10.3390/polym13111758

12. Sakib, N., & Islam, M.K. (2019). Design and Implementation of an EMG Controlled 3D Printed Prosthetic Arm. 2019 IEEE International Conference on Biomedical Engineering, Computer and Information Technology for Health (BECITHCON), 85-88.

13. *What is a Servo Motor? - Understanding the basics of Servo Motor Working*. (n.d.). <https://circuitdigest.com/article/servo-motor-working-and-basics>

14. *Muscle BioAmp Band (EMG Band) | Upside Down Labs Store*. (2024, March 18). Upside Down Labs Store. https://store.upsidedownlabs.tech/product/muscle-bioamp-band/

15. (2021). EMG sensor based control strategy for hand function movements using 3D prosthetic hand . *Journal of Cardiovascular Disease Research*, *12*(03), Article 0975–3583, 0976–2833. https://jcdronline.org/admin/Uploads/Files/624418536e9b14.69166253.pdf

16. Jun-Uk Chu, Member, IEEE, Inhyuk Moon,” A Supervised Feature-Projection-Based Real-Time EMG Pattern Recognition for Multifunction Myoelectric Hand Control“, IEEE/ASME Transactions on Mechatronics, vol. 12, no. 3, June 2007, p. 282-290

17. *A Report on Amputees in India | O&P Virtual Library*. (n.d.). <https://www.oandplibrary.org/op/1986_01_016.asp>