Science2Practice

Volume 2 | Issue 1 November 2024

Soft Robotic Prosthesis: Sensory Development

Barbara Zent, Bruce Hesher, Ashley Spring, and Shawn Coulsey
Eastern Florida State College

Abstract

This experimental study hypothesized that a low-cost and open-source prosthetic hand can be developed to address the issue of accessibility. By utilizing advanced techniques including the integration of EMG sensory systems with robotics, establishing direct connections to the nervous system, and detecting muscle activity, the study created software that translates the inputs into precise prosthetic movements. This research aimed to demonstrate the feasibility and effectiveness of these approaches, ultimately enhancing prosthesis accessibility and expanding knowledge in the field.

Introduction

Prosthetic limbs can improve people's lives by enhancing mobility and independence in everyday activities. Robotic prostheses offer a natural and functional range of motion, making it possible to perform various tasks (Gallego et al., 2013). Prostheses can also improve people's mental well-being by increasing their confidence and self-esteem, which can help them reintegrate into society (Haverkate et al., 2016).

Presently, the prosthetic landscape boasts a rich tapestry of applications that underscore the pervasive integration of sensor technologies. In the domain of myoelectric prosthetic limbs, using electromyography (EMG) sensors has emerged as a transformative innovation. The first configurations of myoelectric limbs were created in 1944-1948 by Reinhold Reiter incorporating the design of EMGs for further prosthetic development (Parker et al., 2006). EMG sensors are meticulously affixed to the dermal surface and adeptly decode the complex electrical signals originating from residual muscles. Consequently, EMG sensors empower users with unprecedented precision and intuitive control which was previously considered unattainable (Grimm et al., 2016). However, despite their numerous benefits, many individuals find these robotic prostheses unaffordable because of their high costs (Gretsch et al.,

2016). This disparity underscores broader healthcare inequalities because people with limited financial means are often unable to access cutting-edge medical technologies.

EMG Boards for Nerve Signaling

EMG sensors play a crucial role in advanced prosthetic devices by providing real-time feedback from users, detecting and measuring electrical activity in forearm muscles to monitor muscle engagement and grip force during actions. This integration facilitates enhanced assessment and improvement of grip strength and control through visual and auditory cues. EMG sensors are invaluable for rehabilitation and scientific research, offering biofeedback and data collection on muscle activity, fatigue dynamics, and performance in grip-related tasks. Moreover, these sensors are integrated into assistive grip devices, allowing individuals with reduced hand strength or mobility to adjust grip strength programmatically, benefiting those with disabilities or the elderly (Farina et al., 2014). By closely monitoring muscle activity and facilitating precise adjustments to grip strength, EMG sensors aid in understanding the impact of grip techniques (Mastinu et al., 2019) and interventions on muscle activation and strength underscoring their significance in both research and practical assistance (Tiwana et al., 2012; Huang et al., 2011). A general EMG board with 27.3 mm x 27.3 mm dimensions was used. The compact size of the EMG board enhances its integration into the prosthetic design, resulting in a smaller, more compatible system.

Servo Motors for Movement

Servo motors are indispensable components in prosthetics due to their ability to precisely control motor output shaft positions, crucial for replicating natural limb movements (Gallego et al., 2013). These motors, typically operating at +5V, offer a torque of 2.5kg/cm and an operating speed of 0.1s/60°, making them ideal for prosthetic applications (Gallego et al., 2013). Equipped with integrated feedback systems such as encoders, these motors enable microcontrollers like the Arduino Nano to monitor and adjust the motor's position with exceptional accuracy in real time, ensuring the prosthetic limb accurately responds to the user's intentions (Au et al., 2008). Furthermore, their compact size and lightweight design, with a motor weight of only 9 g, facilitate seamless integration into prosthetic devices (Attenberger & Buchenrieder, 2014) without adding weight or bulk, enhancing the overall usability and comfort of the prosthetic limb (Haverkate et al., 2016). These specifications highlight the versatility and efficiency of servo motors in advancing prosthetic technology, offering precise control and enhanced functionality to users.

Arduino Nano Microcontrollers for Sensor Control

Prosthetics rely on sensors to enable precise control and functionality of the devices. The Arduino Nano (Fig. 1) is an effective tool for interfacing with these sensors, processing their data, and controlling prosthetic devices. It can adjust servo motors or actuators for responsive movement. Microcontrollers like the Nano can make real-time adjustments in the prosthetic's position and force based on sensor feedback (Unanyan & Belov, 2021). This enhances functionality and user experience by enabling adjustments in grip strength or providing haptic feedback (Unanyan & Belov, 2021).

Additionally, the Nano is relatively low-cost compared to other microcontrollers (Al-Madani et al., 2017), making it an appealing option for prosthetic development. This could reduce production costs and make prosthetic devices more accessible to a wider audience. The Arduino platform has a large and active open-source community. This means that developers can leverage a wealth of pre-existing code, libraries, and resources, which can expedite the development process.

Prosthetic devices play a crucial role in enhancing the quality of life for individuals with limb loss, offering improved mobility (Jayaram et al., 2022), functionality, and psychological benefits (Tetyana et al., 2021). However, the high cost of many prosthetics (Gretsch et al., 2016), which can range from $5,000 to $50,000, significantly limits their accessibility. Components such as EMG boards, servo motors, and Arduino Nano microcontrollers are inexpensive yet effective tools for creating a successful prosthetic hand. This study hypothesizes that a low-cost and open-source prosthetic hand can be developed to address this issue. Utilizing advanced techniques such as the integration of EMG sensory systems with robotics (Huang et al., 2011), direct connections to the nervous system (Tetyana et al., 2021), and muscle activity detection (Huang et al., 2011), the software has been developed to translate these inputs into precise prosthetic movements. The result is a cost-effective, open-source prosthetic hand now available for educational and research purposes worldwide, promoting expanded knowledge and improving prosthesis accessibility.

Materials and Methods

The development of a robotic hand capable of measuring forces while ensuring smooth movement requires intricate circuitry integration to control EMG applications for both mechanical and electrical functionality. In Phase One, a prosthetic hand was created, and the model prosthesis was tested with connected external stimuli, such as a joystick, to deliver commands to the device. During Phase Two, testing of the prosthetic hand with response to human nerve impulses was tested.

Phase One

Designing Components for Modeling a Robotic Hand Prosthesis

The project's design was influenced by an open-source project from the E-nable Foundation (e-NABLE Phoenix Hand v3 Wiki, 2023), which was adapted to fit the specific needs of this project. The model was modified to smoothly integrate wired and electrical components. Which required changes in the design to enable the insertion of these components into the prosthetic model. This adaptation was crucial because the original model was not initially intended for electrical reconfiguration, highlighting the project's unique scope and requirements.

3-D Printing

The Stratasys F170 3-D printer boasts a maximum build area of 10 x 10 x 10 inches (254 x 254 x 254 mm) and a material bay containing one model and one support spool, providing ample capacity for increased output (Stratasys). This transition necessitated a software shift to GrabCAD for compatibility

with the F170 printer's specifications. The chosen materials for the initial tests were Acrylonitrile Butadiene Styrene (ABS) for its exceptional structural strength and Quick Soluble Resin (QSR) as dissolvable support (Kulich et al., 2000; Stansbury & Idacavage, 2016). ABS's high strength and durability make it ideal for structural components while its ease of processing allows for the seamless fabrication of intricate designs. Additionally, ABS's excellent impact resistance ensures the longevity of printed parts (Kulich et al., 2000).

QSR, on the other hand, serves as a dissolvable support material, enabling the creation of complex designs without compromising structural integrity (Stansbury & Idacavage, 2016). This dual-material approach eliminates the need for manual support removal, streamlining the post-processing workflow and ensuring the production of high-quality 3-D printed objects (Gretsch et al., 2016) (Fig. 2). Future project stages aim to explore composite materials to enhance flexibility, particularly for parts requiring a close fit on the wearer. The F170's larger build area and increased material capacity allow for greater flexibility in design and production. Additionally, the F170's AC power requirements, operating at 50/60 Hz with voltage options of 100-132 or 200-240 VAC and LAN compatibility, ensure seamless integration into existing workflows (Stratasys). The initial parts were produced on a smaller scale than planned, indicating the need for remodeling and adjustments in the final project phase.

Code

The Arduino code for the Phase One model is designed to control servo motors, directing them to rotate at specific angles. The rotation speed is adjusted via a joystick, offering enhanced control over motor speed and tension during testing before integrating the EMG board. The joystick serves as a precise input device for tension control, and its commands are successfully executed by the code (Fig. 3) and interpreted through the breadboard layout and the servo motors.

Phase Two

Setting up EMG Boards

The EMG boards for the device are arranged on a smaller breadboard to manage the -9 volts, +9 volts, and neutral ground connections. The sensor lead from the EMG board is connected to the A0 pin on the Arduino Nano (Fig. 4), while a ground pin from the EMG board is linked to a ground pin on the Arduino Nano.

Code

In Phase Two of integrating the EMG board, the code (Fig. 5) was designed to control the servo motors based on muscle contractions. When a specific muscle group contracts, the motors are activated to execute a 180-degree rotation, fully closing the hand. Conversely, when the muscle relaxes, the motors reverse, opening the hand.

Phase One: Results

The Phase One model of a custom-modified hand, made from ABS and fitted with thin fishing lines as tension cords for each finger, demonstrated promising functionality for future testing phases. These tension cords were attached to fingers through custom attachment points, allowing them to be wired through and secured. Lever systems connected to servo motors controlled the tension cords, with the motors linked to a breadboard alongside a joystick mechanism, all powered by one 9V battery. However, the joystick required a 5V power supply, leading to an issue where the breadboard's voltage was too high, potentially overwhelming the joystick's capabilities. To address this, the joystick's power was wired into the breadboard hosting the Arduino Nano, which also connected to each servo motor's input in the analog pins A0-A4, ensuring controlled movement and functionality (Fig. 1).

The Phase One model showcased the ability to perform simple gestures and create a closing grip that could be released using a joystick command (Fig 6.). The prosthetic model sustained grip for a minimum of 500 milliseconds and a maximum of 5000 milliseconds, with a contraction delay between the formation of the gripping gesture of 201 ± 157 milliseconds (mean ± SD, n = 12). The sustained time of functional grip of the human hand was significantly greater than the prosthetic model (1-way ANOVA, $F_{1,24}$=48.02, p<3.64E-7) (Fig. 8).

Phase Two: Results

In Phase Two of testing, progress was made in achieving feedback from the EMG boards, demonstrating the ability to sustain full gripping motion for several seconds. The efficacy of the prosthetic's gripping capabilities was evaluated through meticulous timing captured via video recordings and data analysis conducted via the serial monitor within the Arduino application. The data collected during testing was graphed (Fig. 9) to visualize the attainment of a full grip and the duration it was maintained. Although there was a contraction delay of 282 ± 21.25 milliseconds (mean ± SD, n = 12), there was no significant difference in contraction time forming a fist for 900-4800 milliseconds between the human muscle contraction and the model contraction (1-way ANOVA, $F_{1,22}$=0.27, p=0.61). These results highlighted the successful integration of the EMG boards into the prosthetic system, showcasing the precision and reliability of the feedback mechanism. This achievement marks a substantial milestone in the development of the robotic hand, paving the way for further advancements in enhancing its functionality and usability.

Discussion and Conclusion

The Phase One model of the custom-modified hand demonstrates a step forward in prosthetic technology, particularly in its innovative use of ABS and fishing lines as tension cords. The integration of servo motors controlled by a joystick mechanism opens avenues for enhanced control and functionality. However, several challenges emerged during testing, notably the mismatch in voltage requirements between the joystick and the breadboard's power supply. The utilization of a 9V battery to power the breadboard and servo motors poses a technical hurdle due to the 5V power requirement of the joystick. This disparity potentially compromises the efficacy of the joystick, necessitating a strategic solution to

harmonize power distribution. By wiring the joystick's power directly into the Arduino Nano within the breadboard, alongside connections to each servo motor's input, the model's functionality is preserved, albeit with a temporary workaround. This highlights the importance of meticulous voltage management in future iterations to ensure seamless integration of diverse components. Despite these challenges, the Phase One model exhibits promising functionality, showcasing the ability to execute simple gestures and achieve a closing grip with a controlled release via joystick command. This achievement underscores the potential for further advancements in prosthetic design and functionality. Additionally, in the final stages of development, force sensors were integrated into the prosthesis to provide more detailed sensory feedback, culminating in a comprehensive and functional robotic hand.

The Phase One model of the custom-modified hand signifies a remarkable achievement in prosthetic innovation, exemplifying inventive design features and operational capabilities. While facing initial hurdles concerning voltage regulation and power allocation, the model illustrates the practicality of employing ABS and fishing lines in prosthetic fabrication, coupled with servo motors to achieve controlled movements. In Phase Two, emphasis is placed on integrating an EMG system as the principal control mechanism, harnessing user-generated electrical signals for heightened precision and intuitive control. This transition signifies an advancement in prosthetic technology, offering users enhanced autonomy and responsiveness. Notably, the results from Phase Two testing demonstrate the successful attainment and maintenance of full gripping motion for several seconds, as evidenced by meticulous timing captured via video recordings and data analysis conducted through the serial monitor within the Arduino application. The graphed data further illustrates the effectiveness of the EMG boards in facilitating grip control, affirming the viability of the prosthetic system's feedback mechanism.

A working, low-cost, open-access prosthetic hand was created, demonstrating the feasibility of using inexpensive components for effective prosthetic solutions. Future iterations will prioritize refining the design for enhanced comfort, reliability, and integration within a full-arm prosthetic system, while concurrently addressing cost-reduction measures to broaden accessibility for the general public. By minimizing or substituting current devices and materials with more cost-effective alternatives, the overall functionality and user experience may be improved. Additionally, the prosthesis will remain open source, allowing for educational use and fostering innovation within the prosthetic community. This approach emphasizes the democratization of prosthetic technology, empowering individuals with limb differences to have greater control over their prosthetic solutions. In summary, this research establishes a strong foundation for ongoing innovation in prosthetic design, providing low-cost options for research, education, and the advancement of prosthetic knowledge.

Acknowledgments

Korie Carter played a key role in identifying the necessary modeling software for this project by providing guidance and assistance in converting the project's 3-D models to a format compatible with AutoCAD transitioning from the original 3-D modeling system.

References

Al-Madani, F., Kassab, M., Sonmez, B. Z., & Solmaz, H. (2017). Design and development of a low-cost force feedback 3D printed myoelectric hand prosthesis. *2017 21st National Biomedical Engineering Meeting (BIYOMUT)*, i-iii. https://doi.org/10.1109/BIYOMUT.2017.8479096

Attenberger, A., & Buchenrieder, K. (2014). An arduino-simulink-control system for modern hand protheses. In L. Rutkowski, M. Korytkowski, R. Scherer, R. Tadeusiewicz, L.A. Zadeh, & J.M. Zurada, (Eds.), *Artificial Intelligence and Soft Computing, 8468*, 433-434. https://doi.org/10.1007/978-3-319-07176-3_38

Au, S., Berniker, M., & Herr, H. (2008). Powered ankle-foot prosthesis to assist level-ground and stair-descent gaits. *Neural Networks: The Official Journal of the International Neural Network Society*, *21*(4), 654-666. https://doi.org/10.1016/j.neunet.2008.03.006

e-NABLE Phoenix Hand v3 Wiki. (2023). e-NABLE Devices Catalog. https://hub.e-nable.org/s/e-nable-devices/wiki/e-NABLE+Phoenix+Hand+v3

Farina, D., Jiang, N., Rehbaum, H., Holobar, A., Graimann, B., Dietl, H., & Aszmann, O. C. (2014). The extraction of neural information from the surface EMG for the control of upper-limb prostheses: emerging avenues and challenges. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, *22*(4), 797-809. https://doi.org/10.1109/TNSRE.2014.2305111

Gallego, J. A., Rocon, E., Belda-Lois, J. M., Koutsou, A. D., Mena, S., Castillo, A., & Pons, J. L. (2013, July). Design and validation of a neuroprosthesis for the treatment of upper limb tremor. *Annual International Conference of the IEEE Engineering in Medicine and Biology Society,* 3606-3609. https://doi.org/10.1109/EMBC.2013.6610323

Gretsch, K. F., Lather, H. D., Peddada, K. V., Deeken, C. R., Wall, L. B., & Goldfarb, C. A. (2016). Development of novel 3D-printed robotic prosthetic for transradial amputees. *Prosthetics and Orthotics International*, *40*(3), 400-403. https://doi.org/10.1177/0309364615579317

Grimm, F., Walter, A., Spüler, M., Naros, G., Rosenstiel, W., & Gharabaghi, A. (2016). Hybrid neuroprosthesis for the upper limb: combining brain-controlled neuromuscular stimulation with a multi-joint arm exoskeleton. *Frontiers in Neuroscience*, *10*, 367. https://doi.org/10.3389/fnins.2016.00367

Haverkate, L., Smit, G., & Plettenburg, D. H. (2016). Assessment of body-powered upper limb prostheses by able-bodied subjects, using the Box and Blocks Test and the Nine-Hole Peg Test. *Prosthetics and Orthotics International*, *40*(1), 109-116. https://doi.org/10.1177/0309364614554030

Huang, C. J., Wang, Y. W., Huang, T. H., Lin, C. F., Li, C. Y., Chen, H. M., Chen, P.C., & Liao, J. J. (2011). Applications of machine learning techniques to a sensor-network-based prosthesis training system. *Applied Soft Computing*, *11*(3), 3229-3237. https://doi.org/10.1016/j.asoc.2010.12.025

Jayaram, A. K., Pappa, A. M., Ghosh, S., Manzer, Z. A., Traberg, W. C., Knowles, T. P. J., Daniel, S., & Owens, R. M. (2022). Biomembranes in bioelectronic sensing. *Trends in Biotechnology*, *40*(1), 107-123. https://doi.org/10.1016/j.tibtech.2021.06.001

Kulich, D. M., Gaggar, S. K., Lowry, V., & Stepien, R. (2003). Acrylonitrile–butadiene–styrene (ABS) polymers. *Kirk-Othmer Encyclopedia of Chemical Technology*. https://doi.org/10.1002/0471238961.01021911211209.a01.pub2

Mastinu, E., Clemente, F., Sassu, P., Aszmann, O., Brånemark, R., Håkansson, B., Controzzi, M., Cipriani, C., & Ortiz-Catalan, M. (2019). Grip control and motor coordination with implanted and surface electrodes while grasping with an osseointegrated prosthetic hand. *Journal of Neuroengineering and Rehabilitation*, *16*, 1-10. https://doi.org/10.1186/s12984-019-0511-2

Parker, P., Englehart, K., & Hudgins, B. (2006). Myoelectric signal processing for control of powered limb prostheses. *Journal of electromyography and Kinesiology*, *16*(6), 541-548. https://doi.org/10.1016/j.jelekin.2006.08.006

Stansbury, J. W., & Idacavage, M. J. (2016). 3D printing with polymers: Challenges among expanding options and opportunities. *Dental Materials*, *32*(1), 54-64. https://doi.org/10.1016/j.dental.2015.09.018

Tetyana, P., Shumbula, P. M., & Njengele-Tetyana, Z. (2021). Biosensors: design, development and applications. In *Nanopores*. IntechOpen. https://doi.org/10.5772/intechopen.97576

Tiwana, M. I., Redmond, S. J., & Lovell, N. H. (2012). A review of tactile sensing technologies with applications in biomedical engineering. *Sensors and Actuators A: physical*, *179*, 17-31. https://doi.org/10.1016/j.sna.2012.02.051

Unanyan, N. N., & Belov, A. A. (2021). Design of upper limb prosthesis using real-time motion detection method based on EMG signal processing. *Biomedical Signal Processing and Control*, *70*. https://doi.org/10.1016/j.bspc.2021.103062

Figure 1

Arduino nano pinout (Ground Power LED Internal Pin SWD Pin Digital Pin Analog Pin Other Pin Microcontroller's Port Default, n.d.)

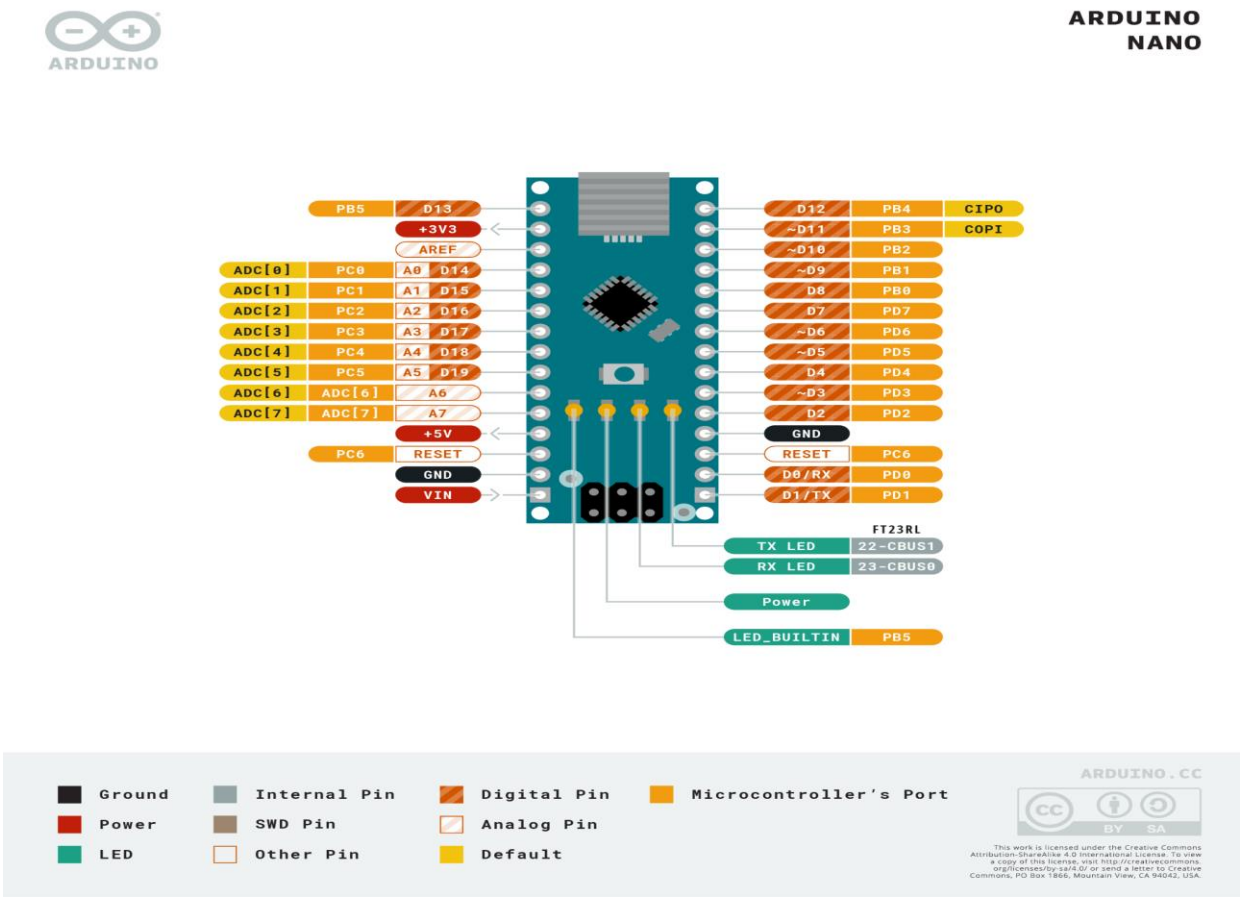
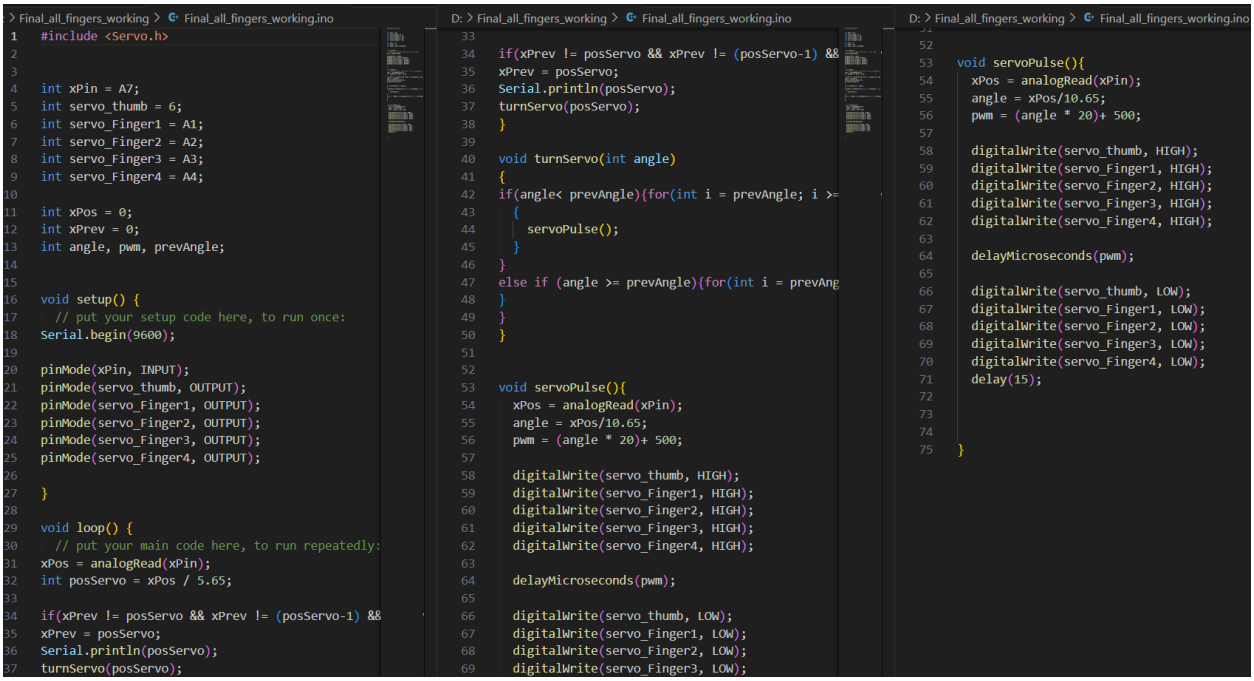


Figure 2

Prosthetic hand created in this research

Figure 3

Code for Phase One



```
#include <Servo.h>



int xPin = A7;
int servo_thumb = 6;
int servo_Finger1 = A1;
int servo_Finger2 = A2;
int servo_Finger3 = A3;
int servo_Finger4 = A4;

int xPos = 0;
int xPrev = 0;
int angle, pwm, prevAngle;


void setup() {
  // put your setup code here, to run once:
Serial.begin(9600);

pinMode(xPin, INPUT);
pinMode(servo_thumb, OUTPUT);
pinMode(servo_Finger1, OUTPUT);
pinMode(servo_Finger2, OUTPUT);
pinMode(servo_Finger3, OUTPUT);
pinMode(servo_Finger4, OUTPUT);

}

void loop() {
  // put your main code here, to run repeatedly:
xPos = analogRead(xPin);
int posServo = xPos / 5.65;

if(xPrev != posServo && xPrev != (posServo-1) &&
xPrev = posServo;
Serial.println(posServo);
turnServo(posServo);
```

```

if(xPrev != posServo && xPrev != (posServo-1) &&
xPrev = posServo;
Serial.println(posServo);
turnServo(posServo);
}

void turnServo(int angle)
{
if(angle< prevAngle){for(int i = prevAngle; i >=
  {
    servoPulse();
  }
}
else if (angle >= prevAngle){for(int i = prevAng
}
}
}


void servoPulse(){
  xPos = analogRead(xPin);
  angle = xPos/10.65;
  pwm = (angle * 20)+ 500;

  digitalWrite(servo_thumb, HIGH);
  digitalWrite(servo_Finger1, HIGH);
  digitalWrite(servo_Finger2, HIGH);
  digitalWrite(servo_Finger3, HIGH);
  digitalWrite(servo_Finger4, HIGH);

  delayMicroseconds(pwm);

  digitalWrite(servo_thumb, LOW);
  digitalWrite(servo_Finger1, LOW);
  digitalWrite(servo_Finger2, LOW);
  digitalWrite(servo_Finger3, LOW);
```

```


void servoPulse(){
  xPos = analogRead(xPin);
  angle = xPos/10.65;
  pwm = (angle * 20)+ 500;

  digitalWrite(servo_thumb, HIGH);
  digitalWrite(servo_Finger1, HIGH);
  digitalWrite(servo_Finger2, HIGH);
  digitalWrite(servo_Finger3, HIGH);
  digitalWrite(servo_Finger4, HIGH);

  delayMicroseconds(pwm);

  digitalWrite(servo_thumb, LOW);
  digitalWrite(servo_Finger1, LOW);
  digitalWrite(servo_Finger2, LOW);
  digitalWrite(servo_Finger3, LOW);
  digitalWrite(servo_Finger4, LOW);
  delay(15);



}
```

Figure 4

Arduino® Nano. (n.d.). (*https://docs.arduino.cc/resources/datasheets/A000005-datasheet.pdf*)

5.1 Analog

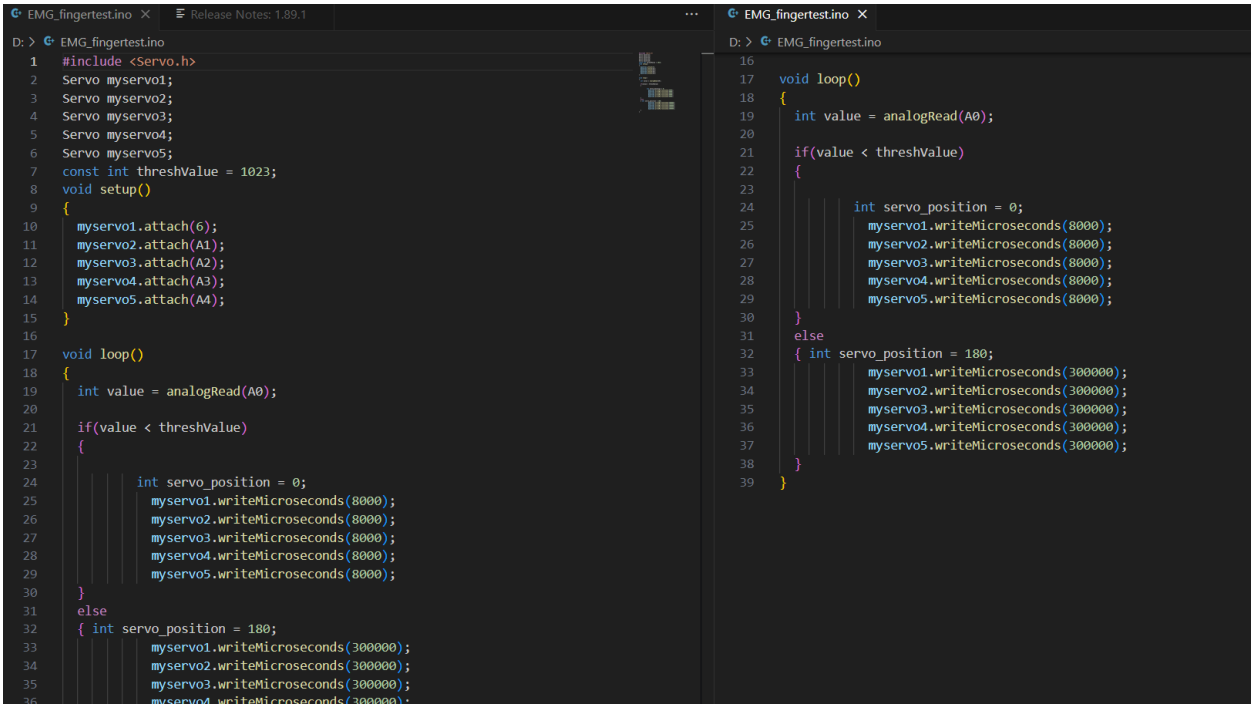
Pin	Function	Type	Description
1	+3V3	Power	5V USB Power
2	A0	Analog	Analog input 0 /GPIO
3	A1	Analog	Analog input 1 /GPIO
4	A2	Analog	Analog input 2 /GPIO
5	A3	Analog	Analog input 3 /GPIO
6	A4	Analog	Analog input 4 /GPIO
7	A5	Analog	Analog input 5 /GPIO
8	A6	Analog	Analog input 6 /GPIO
9	A7	Analog	Analog input 7 /GPIO
10	+5V	Power	+5V Power Rail
11	Reset	Reset	Reset
12	GND	Power	Ground
12	VIN	Power	Voltage Input

5.2 Digital

Pin	Function	Type	Description
1	D1/TX1	Digital	Digital Input 1 /GPIO
2	D0/RX0	Digital	Digital Input 0 /GPIO
3	D2	Digital	Digital Input 2 /GPIO
4	D3	Digital	Digital Input 3 /GPIO
5	D4	Digital	Digital Input 4 /GPIO
6	D5	Digital	Digital Input 5 /GPIO
7	D6	Digital	Digital Input 6 /GPIO
8	D7	Digital	Digital Input 7 /GPIO
9	D8	Digital	Digital Input 8 /GPIO
10	D9	Digital	Digital Input 9 /GPIO
11	D10	Digital	Digital Input 10 /GPIO
12	D11	Digital	Digital Input 11 /GPIO
13	D12	Digital	Digital Input 12 /GPIO
14	D13	Digital	Digital Input 13 /GPIO
15	Reset	Reset	Reset
16	GND	Power	Ground

Figure 5

Code for Phase Two



```
#include <Servo.h>
Servo myservo1;
Servo myservo2;
Servo myservo3;
Servo myservo4;
Servo myservo5;
const int threshValue = 1023;
void setup()
{
  myservo1.attach(6);
  myservo2.attach(A1);
  myservo3.attach(A2);
  myservo4.attach(A3);
  myservo5.attach(A4);
}

void loop()
{
  int value = analogRead(A0);

  if(value < threshValue)
  {

        int servo_position = 0;
          myservo1.writeMicroseconds(8000);
          myservo2.writeMicroseconds(8000);
          myservo3.writeMicroseconds(8000);
          myservo4.writeMicroseconds(8000);
          myservo5.writeMicroseconds(8000);
  }
  else
  { int servo_position = 180;
          myservo1.writeMicroseconds(300000);
          myservo2.writeMicroseconds(300000);
          myservo3.writeMicroseconds(300000);
```

```
void loop()
{
  int value = analogRead(A0);

  if(value < threshValue)
  {

        int servo_position = 0;
          myservo1.writeMicroseconds(8000);
          myservo2.writeMicroseconds(8000);
          myservo3.writeMicroseconds(8000);
          myservo4.writeMicroseconds(8000);
          myservo5.writeMicroseconds(8000);
  }
  else
  { int servo_position = 180;
          myservo1.writeMicroseconds(300000);
          myservo2.writeMicroseconds(300000);
          myservo3.writeMicroseconds(300000);
          myservo4.writeMicroseconds(300000);
          myservo5.writeMicroseconds(300000);
  }
}
```

Figure 6

Phase One testing movement of the model with a joystick

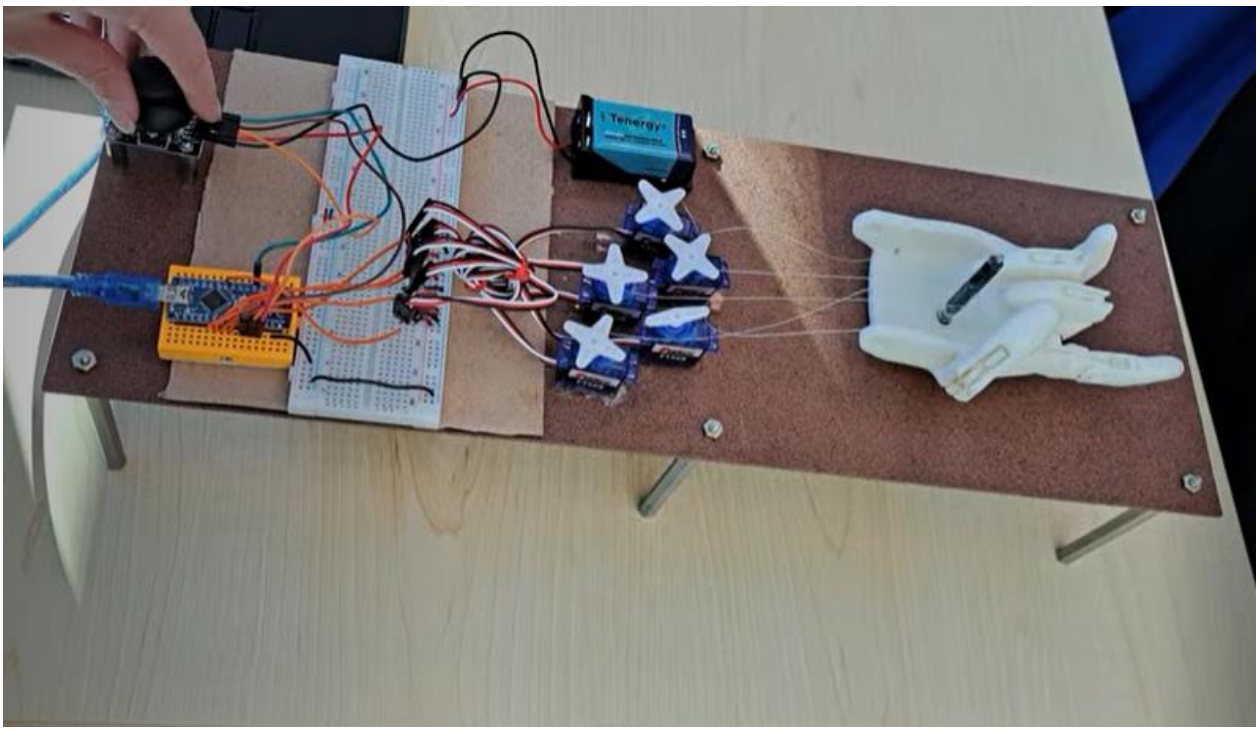


Figure 7

EMG board testing

Figure 8

Phase One graphed results of prototype functionality

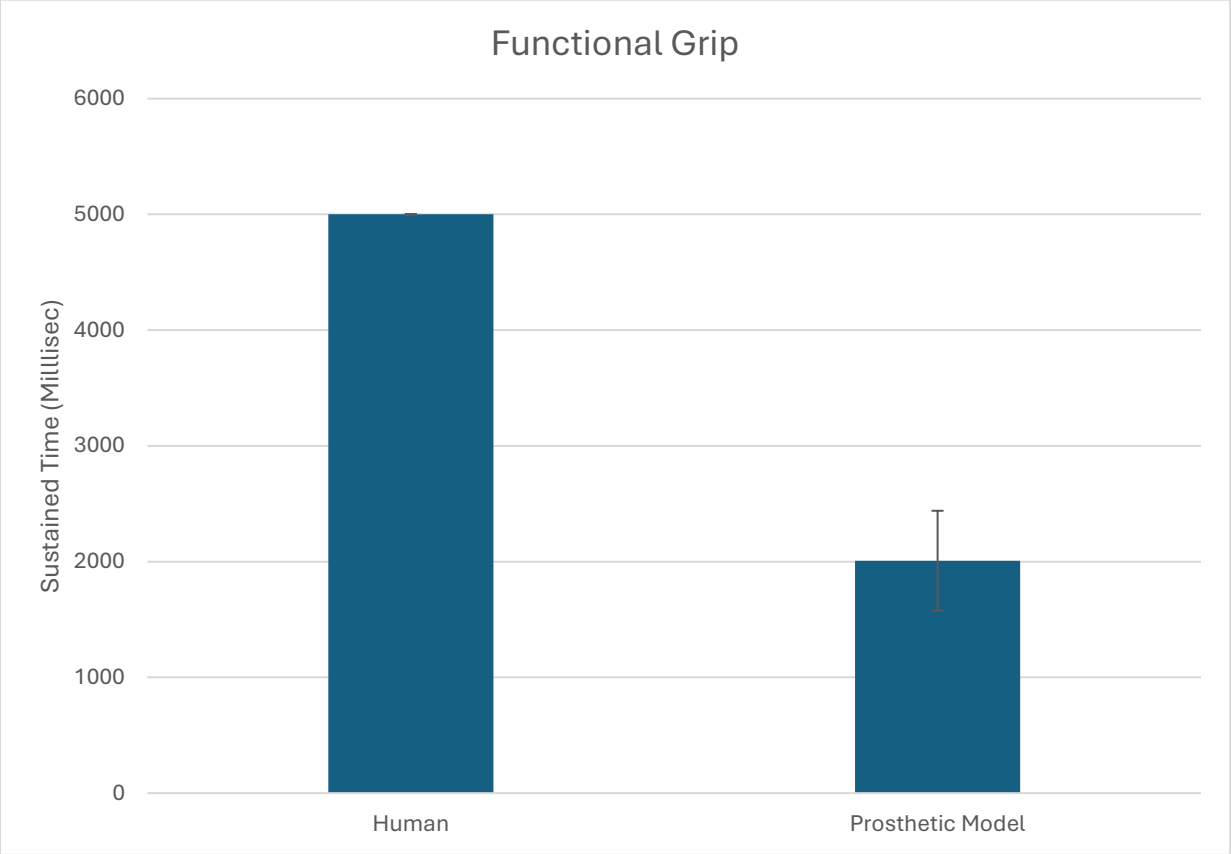


Figure 9

Delay reaction times between human contractions and model reactions