

PRÓTESIS BIÓNICA PARA MOVIMIENTO DE EXTREMIDADES SUPERIORES

Andrés Asprilla Pedroza & Jeisson Stiven Ospina Álvarez

**INSTITUCIÓN UNIVERSITARIA PASCUAL BRAVO -FACULTAD DE INGENIERÍA
TECNOLOGÍA SISTEMAS MECATRÓNICOS
MEDELLÍN 2021**

PRÓTESIS BIÓNICA PARA MOVIMIENTO DE EXTREMIDADES SUPERIORES

Trabajo de grado para optar al título de Tecnólogo en Mecatrónica

Asesor Técnico

Carlos Alberto Valencia Hernández Ingeniero en Instrumentación y Control

Andrés Asprilla Pedroza & Jeisson Stiven Ospina Álvarez

**INSTITUCIÓN UNIVERSITARIA PASCUAL BRAVO - FACULTAD DE INGENIERÍA
TECNOLOGÍA SISTEMAS MECATRÓNICOS
MEDELLÍN 2021**

Dedicatoria

Dedico principalmente este trabajo a todas aquellas personas, compañeros, profesores y amigos que estuvieron presentes en este largo proceso desde años atrás, a mi madre por su apoyo incondicional y a todo aquel amigo o compañero que ha influido en mí, el gusto y la curiosidad por la ciencia, tecnología e innovación.

Mismas herramientas que me han impulsado a la insaciable búsqueda constante de conocimiento, promoviendo la creatividad necesaria para implementar la misma, en soluciones para retos presentes y futuros presentados.

Contenido

<u>Introducción</u>	8
<u>Abstract</u>	9
<u>Glosario</u>	10
<u>1. Planteamiento del problema</u>	11
<u>1.1. Descripción.</u>	11
<u>1.1. Formulación.</u>	11
<u>2. Justificación</u>	12
<u>3. Objetivos</u>	14
<u>3.1. Objetivo general</u>	14
<u>3.2. Objetivos específicos</u>	14
<u>4. Marco teórico</u>	15
<u>4.1. Portéis de mano.</u>	17
<u>4.2. impresión 3D.</u>	17
<u>4.3. Arduino.</u>	18
<u>4.4. Servomotores.</u>	19
<u>4.5. Electromiografía de contacto.</u>	19
<u>4.6. Programación en Arduino.</u>	20
<u>5. Metodología</u>	21
<u>5.1. Tipo de proyecto.</u>	21
<u>5.2. Método.</u>	21
<u>5.2.1. Instrumentos de recolección de información.</u>	21
<u>6. Impresión 3D de la prótesis.</u>	22
<u>6.1 Búsqueda e investigación de prototipo.</u>	22
<u>6.2 parámetros y configuraciones de impresión</u>	27
<u>6.3 Impresión de Componentes</u>	29
<u>7. Ensamble de la prótesis</u>	32
<u>7.1 Arduino uno</u>	35
<u>7.2 Servos motores</u>	36
<u>7.4 Sensor MG.</u>	37

7.5 Control.	41
8.1 Diseño del algoritmo para reconocimiento de gestos y programación.	43
8.2 Diagrama de flujo.	53
9 conclusiones. -----	55

Lista de tablas y figuras de figuras.

Figura 1. prótesis de mano, emprendimiento que construye futuro.....	10
Figura 2. Impresora3D.....	11
Figura 3. Dedo Indice.....	15
Figura 4. pulgar parte superior.....	16
Figura 5. dedo anular.....	16
Figura 6. parte pulgar medio.....	16
Figura 7. tope final servo.....	16
Figura 8. palma de mano.....	16
Figura 9. parte inferior del pulgar.....	17
Figura 10. dedo meñique.....	17
Figura 11. parte superior de dedos.....	17
Figura 12. soporte servo.....	17
Figura 13. Hand.....	17
Figura 14. Muñeca.....	17
Figura 15. cura logo.....	19
Figura 16. impresora 3D.....	22
Figura 17. Arduino uno-.....	28
Figura 18. Servomotores.....	29
Figura 19. sensor MG.....	31
Figura 20. esquema conexión.....	33

Figura 21.	46
Figura 22. . PLA autoría propia	20
Figura 23. Wrist support_	21
Figura 24. parámetros	21
Figura 25. desplazamiento	21
Figura 26. Partes distales	22
Figura 27. Piezas modeladas	22
Figura 28. Base muñeca	22
Figura 29. Base servomotores	22
Figura 0. Piezas móviles	22
Figura 31. Piezas impresas	25
Figura 32. Mano desarmada	25
Figura 33. Mano ensamblada	25
Figura 34. Arduino uno	27
Figura 35. Sensor myoware	27
Figura 36. Pines	28
Figura 37. Board	28
Figura 38. Batería	28
Figura 39. Servo motor	28
Figura 40. Electrodos	28
Figura 41. Mano ensamblada	20
Figura 42. Definición pines	37
Figura 43. Voidsetup	37

Figura 44. Voidloop	38
Figura 45. Lectura seria.....	39
Figura 46. Lectura sploter.....	39
Figura 47. Librería.....	40
Figura 48. Vidsetup.....	41
Figura 49. Voidloop	41
Figura 50. Control servos	43
Figura 51. Ratio.....	43
Figura 52. Bluce	44
Figura 53. Programa	44
Figura 54. Montaje completo	45
Figura 55. control y conexiones	46

Introducción

Mucho se ha dicho respecto los avances de la ciencia y la tecnología en los tiempos contemporáneos, los cuales se hacen cada vez más tangibles debido al enorme potencial de transformar y simplificar la vida de las personas es por ello, que los aportes de la ingeniería juegan un papel fundamental y transversal en el desarrollo de la evolución humana y sus comodidades.

A través de este proyecto, el cual tiene como objetivo principal desarrollar una prótesis para la mano. combinando técnicas mecatrónicas e impresión 3D, se presenta una prótesis de bajo costo, la cual adopta los movimientos y funciones básicas de la extremidad.

Por medio de la plataforma Arduino, se realiza la programación de los comandos para la obtención de movimientos, a través de sensores superficiales electromiografico EMG, los cuales detectan pequeños pulsos eléctricos, producto de la actividad muscular, transformados en una señal analógica, capaz de interpretarlos, permitiendo el control del prototipo de forma sencilla y eficaz.

La impresión 3D, proporciona garantías en cuanto a la reducción de costos y tiempo de ejecución en comparación con las diferentes prótesis que ofrece el mercado; Las cuales están diseñadas con materiales costosos y difíciles de conseguir.

En la actualidad, los inminentes avances relacionados con la robótica y las técnicas modernas de ingeniería, favorecen los diversos campos del conocimiento, con el fin de generar aportes significativos en la sociedad.

Abstract

Much has been said regarding the advancement of science and technology in contemporary times, which a few decades ago seemed to be science fiction and today they become tangible, it is for them that the contributions of engineering play a fundamental and transversal role in the development of human evolution and its comforts.

Through this project, whose main objective is to develop a bionic prosthesis for the hand, mechatronic techniques are combined with 3D printing, to present a low-cost prosthesis that emulates the movements and basic functions of the limb.

Through the Arduino platform, the programming of the commands to obtain movements is carried out, through surface sensors EMG electromyography, which detect small electrical pulses product of muscle activity and transformed into an analog signal, the microprocessor is capable to interpret them allow the control of the prototype in a simple and effective way.

3D printing does not only allow obtaining a satisfactory result for the project, fulfilling the main objective of reducing costs and, in turn, represents implantation or execution times compared to the prostheses offered on the market.

Nowadays, the imminent advances related to robotics and modern control techniques have meant that these disciplines of knowledge can be used as an educational complement in the compulsory basic teachings.

Developing and implementing control techniques on different robotic elements locally or remotely is a clear example of academic activity that always causes additional motivation in Science, Technology or Engineering students. Thus, this document presents a novel and current electronic application such as the construction of a bionic prosthesis. At the same time, the different details of its implementation are discussed together with the experimental results obtained.

Glosario

Prótesis: Una prótesis es una extensión artificial que reemplaza o provee una parte del cuerpo que falta por diversas razones.

Mano: Parte del cuerpo humano que va desde la muñeca hasta la punta de los dedos

Robótica: es la rama de la ingeniería mecánica, de la ingeniería eléctrica, de la ingeniería electrónica, de la ingeniería biomédica y de las ciencias de la computación, que se ocupa del diseño, construcción, operación, estructura, manufactura y aplicación de los robots

Inteligencia artificial: la Inteligencia Artificial (IA) es la combinación de algoritmos planteados con el propósito de crear máquinas que presenten las mismas capacidades que el ser humano.

Impresión 3d: es un grupo de tecnologías de fabricación por adición donde un objeto tridimensional es creado mediante la superposición de capas sucesivas de material.

Electromiografía: es la técnica de registro gráfico de la actividad eléctrica producida por los músculos esqueléticos. Esta actividad eléctrica es conocida como el electromiograma o "EMG"

Señal Análoga: es un tipo de señal generada por algún tipo de fenómeno electromagnético; que es representable por una función matemática continua en la que es variable su amplitud y periodo en función del tiempo

Señal Digital: es un tipo de señal en que cada signo que codifica el contenido de la misma puede ser analizado en término de algunas magnitudes que representan valores discretos, en lugar de valores dentro de un cierto rango

Microcontrolador: Un microcontrolador es un circuito integrado programable, capaz de ejecutar las órdenes grabadas en su memoria.

1. Planteamiento del problema

1.1. Descripción.

En Colombia según un censo del 2005, se evidencia que existen 11.143 personas que No tienen manos, Mas de 1000 son niños que nacieron con malformaciones congénitas o presentaron un accidente; este trabajo de grado nace como una iniciativa que intenta ayudar con este problema ya que se pretende demostrar como a través de la tecnología enfatizada a la impresión 3D y programación ,se pueden desarrollar prótesis dirigidas a personas con movilidad reducida, las cuales mediante técnicas y materiales de baja inversión se pueden construir y en conjunto con la programación y la electrónica ser funcionales, dejando así un gran impacto mejorando la calidad de vida en estas personas.

1.1. Formulación.

¿Atraves de la programación e impresión 3D es posible construir una prótesis funcional y de accesibilidad económica?

2. Justificación

En un estudio hecho por el censo en el año 2005, Se demostró que Colombia hay 2'632.255 personas con discapacidad de los cuales 11.143 no tienen brazos.

Más de 1.000 son niños, este mismo estudio fue actualizado en el año 2015 y expuso que la situación no ha mejorado, debido a que el número total de personas con limitaciones Permanentes ascendió a 3'051.217 de los cuales el 29,6%, entre hombres mujeres y niños son personas que no pueden utilizar sus manos y brazos con proyectos como este podemos impactar calidad de vida, ya que con la aplicación de diversas tecnologías desarrolladas en los últimos año se hace posible fabricar prototipos con piezas complejas a muy bajo costo y con tiempos relativamente inmediatos. Si a ello se le suma la reducción de los tamaños y la aparición de actuadores eléctricos y sistemas embebidos, es posible obtener sistemas mecatrónicos complejos con dimensiones y pesos óptimos.

Con un coste total estimado de 160 dólares, se pueden comprar los elementos necesarios

Para el desarrollo y construcción de este proyecto-

2.1.Objetivo general

Desarrollar un prototipo de prótesis para la mano, mediante técnicas y tecnologías de impresión 3D, la cual pueda satisfacer las necesidades y funciones básicas para las personas con movilidad reducida a bajo costo.

2.2.Objetivos específicos

- 2.2.1. Imprimir, una prótesis de mano fabricada a través de técnicas de impresión 3D.
- 2.2.2. Ensamblar la prótesis impresa mediante la tecnología 3D e instalar servomotores.
- 2.2.3. Programar un algoritmo que facilite el movimiento, de servomecanismos funcionales que asemejen el comportamiento natural de la mano.

3. Marco teórico

Para llevar a cabo una prótesis de mano, es fundamental entender como naturalmente funciona la misma. La mano como órgano del cuerpo humano es sumamente compleja, permite realizar múltiples movimientos y es un miembro sensitivo, dado que brinda información del medio como la temperatura, tacto, dolor.

Está compuesta por más de 27 huesos, 8 en el carpo, 5 metacarpianos y un total de 14 falanges. En conjunto forman un canal de concavidad anterior por el que se deslizan los tendones de los músculos flexores de los dedos y de 40 músculos, con lo cual posee más de 20 grados de libertad. Existen músculos encargados del desplazamiento de la mano y otros que se encargan de limitar dichos movimientos. Es muy difícil definir donde empieza y termina cada músculo y tendón ya que se encuentra de forma entrelazada y se confunden.

Basados en la información anterior, las prótesis tienen como limitante la parte sensorial, por ello se restringen a suplir la necesidad funcional de los movimientos básicos de la extremidad, aparte de brindar armonía y estética al cuerpo, así como soporte y ayuda emocional a las personas que buscan mejorar su calidad de vida a través de las mismas.



Figura 1. Prótesis de mano, emprendimiento que construye futuro

Fuente: extraído de: <https://tec.mx/es/noticias/aguascalientes/emprendedores/emprendimiento-de-corazon-joven-hidrocalido-crea-protesis-3d>

3.1. Prótesis de mano.

Las prótesis de mano se definen como dispositivos, creados para mejorar la calidad de vida de las personas con amputaciones. También son definidas como, implante artificial implantado en el organismo. se pueden clasificar según el material, la funcionalidad, el mecanismo de actuación, la apariencia, el nivel de amputación o por el precio, En la actualidad existen varios tipos de estas como lo son: mecánicas, neumáticas, mioeléctrica y híbridas.

3.2. impresión 3D.

La impresión 3d es un grupo de tecnologías de fabricación por adición donde un objeto tridimensional es creado mediante la superposición de capas sucesivas de material, las herramientas que se utilizan para trabajar con esta tecnología son las impresoras, máquinas capaces de realizar réplicas de diseños en 3D, creando piezas o maquetas volumétricas a partir de un diseño hecho por ordenador. Surgen con la idea de convertir archivos de 2D en prototipos reales o 3D. Comúnmente se ha utilizado en el prefabricado de piezas o componentes.

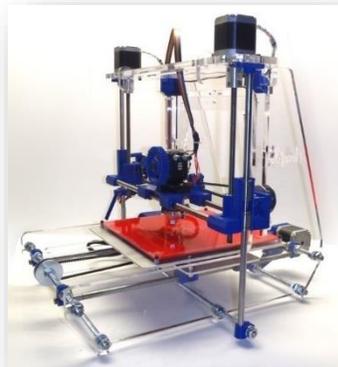


Figura 2. Impresora 3D.

Fuente: extraído de https://es.wikipedia.org/wiki/Impresora_3D

Nota: Impresora es precisa y la plataforma de construcción es de fácil acceso. Funciona con software de código abierto constantemente actualizado y firmware de código abierto calibrado a medida. Tomada de Impresora 3 (Wolf, 2012).

3.3. Arduino.

El proyecto nació en 2003, cuando varios estudiantes del Instituto de Diseño Interactivo de Ivrea, Italia, con el fin de facilitar el acceso y uso de la electrónica y programación. Lo hicieron para que los estudiantes de electrónica tuviesen una alternativa más económica a las populares BASIC Stamp, unas placas que por aquel entonces valían más de cien dólares, y que no todos se podían permitir.

El resultado fue Arduino, una placa con todos los elementos necesarios para conectar periféricos a las entradas y salidas de un microcontrolador, y que puede ser programada tanto en Windows como macOS y GNU/Linux. Un proyecto que promueve la filosofía 'learning by doing', que viene a querer decir que la mejor manera de aprender es cacharreando. Arduino a su vez es una plataforma de creación de electrónica de código abierto, la cual está basada en hardware y software libre, flexible y fácil de utilizar para los creadores y desarrolladores. Esta plataforma permite crear diferentes tipos de microordenadores de una sola placa a los que la comunidad de creadores puede darles diferentes tipos de uso, el hardware libre y el software libres son los dispositivos cuyas especificaciones y diagramas son de acceso público, de manera que cualquiera puede replicarlos. Esto quiere decir que Arduino ofrece las bases para que cualquier otra persona o empresa pueda crear sus propias placas, pudiendo ser diferentes entre ellas, pero igualmente funcionales al partir de la misma base, (YÚBAL FERNÁNDEZ,2020).

3.4. Servomotores.

Son dispositivos similares a un motor de corriente continua que tiene la capacidad de ubicarse en cualquier posición dentro de su rango de operación, y mantenerse estable en dicha posición

El servomotor es un motor eléctrico que lleva incorporado un sistema de regulación que puede ser controlado tanto en velocidad como en posición.

Es posible modificar un servomotor para obtener un motor de corriente continua que, si bien ya no tiene la capacidad de control del servo, conserva la fuerza, velocidad y baja inercia que caracteriza a estos dispositivos, Está conformado por un motor y un circuito de control. También potencia proporcional para cargas mecánicas. Un servo, por consiguiente, tiene un consumo de energía reducido. (Leal, Silvia.2015).

3.5. Electromiografía de contacto.

Es la técnica de registro gráfico de la actividad eléctrica producida por los músculos esqueléticos.

Esta actividad eléctrica es conocida como el electromiograma o “EMG”.

El EMG puede ser monitoreado a través de electrodos insertados dentro de los músculos (electrodos intramusculares) o a través de electrodos en la superficie de la piel sobre el músculo (electrodos superficiales).

El EMG es usado por científicos para estudiar el sistema neuromuscular, por médicos para el diagnóstico de enfermedades neuromusculares, y por fisioterapeutas para monitorear la activación de músculos de un paciente, especifica mente para nuestro proyecto, aplicaremos la electromiografía de superficie a través de un sensor muscular, el registro de las señales eléctricas extracelulares generadas por las fibras musculares y transmitidas a través de los tejidos. Será leída e capturada por este sensor el cual, la enviará la señal al Arduino y este a su vez hará actuar los servomotores montados en cada dedo, de la prótesis.

La electromiografía analiza la integridad de todo el sistema motor, que incluye las neuronas motoras superiores e inferiores, la unión neuromuscular y el músculo (Iemman y Ritchie,1979).

3.6. Programación en Arduino.

es la programación de un microcontrolador, Programar Arduino consiste en traducir a líneas de código las tareas automatizadas que queremos hacer leyendo de los sensores y en función de las condiciones del entorno programar la interacción con el mundo exterior mediante unos actuadores.

Arduino proporciona un entorno de programación sencillo y potente para programar, pero además incluye las herramientas necesarias para compilar el programa y “quemar” el programa ya compilado en la memoria flash del microcontrolador. Además el IDE nos ofrece un sistema de gestión de librerías y placas muy práctico. Como IDE es un software sencillo que carece de funciones avanzadas típicas de otros IDEs, pero suficiente para programar.

3.7 Formato STL

Es un formato de archivo informático de diseño asistido por computadora (CAD) que define geometría de objetos 3D, excluyendo información como color, texturas o propiedades físicas que sí incluyen otros formatos CAD. Fue creado por la empresa 3D Systems, concebido para su uso en la industria del prototipado rápido y sistemas de fabricación asistida por ordenador, formato STL es ampliamente utilizado por el software de control de máquinas de impresión 3D.

(StereoLithography,2011)

4. Metodología

4.1. Tipo de proyecto.

La metodología que se utilizará para el desarrollo de esta prótesis será práctico experimental con una exhaustiva investigación y ejecución de algoritmos, diseño e impresión, que permita el desarrollo del prototipo mecatrónico de la prótesis .

4.2. Método.

El método a implementar es el deductivo, porque será parte de una teoría estipulada para solucionar un problema que se está presentado con las prótesis de manos, su funcionalidad, alto costo y poca eficiencia.

4.2.1. Instrumentos de recolección de información.

4.2.1.1. Fuentes primarias.

Videos, artículos, blogs, páginas web y librerías de procesamiento de imágenes.

4.2.1.2. Fuentes secundarias.

Tesis especializadas, libros de procesamiento de imágenes y revistas técnicas de información general.

5. Impresión 3D de la prótesis.

La impresión 3D, consiste en producir objetos a través de la adición de material en capas que corresponden a las sucesivas secciones transversales de un modelo 3D, el desarrollo de nuestra prótesis se basó en los siguientes pasos para ejecutarse:

- Búsqueda e investigación de prototipo
- Ajustes de parámetros, para impresión
- Impresión de Componentes

5.1 Búsqueda e investigación de prototipo.

El diseño de nuestro prototipo, es minimalista he inspirado por la estructura ósea de la mano humana, este diseño fue investigado y descargado en la plataforma (Thingiverse) realizado por el usuario **Bribo** el 12- 24 de marzo de 2018, consta de 30 Stl en las cuales podemos encontrar, los dedos, pulgar, índice, corazón, anular, meñique, la palma de la mano y la muñeca. Adicional algunas armaduras para el montaje de servo motores y circuitería.

Algunos de los Diseños, se puede evidenciar a continuación:

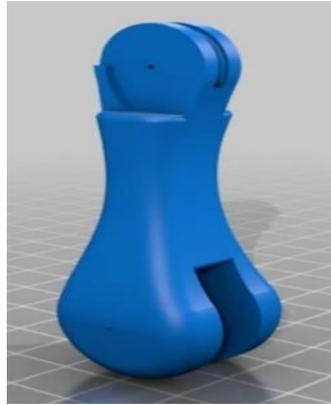


Figura 3. Dedo

Fuente: <https://www.thingiverse.com/thing:2838239>

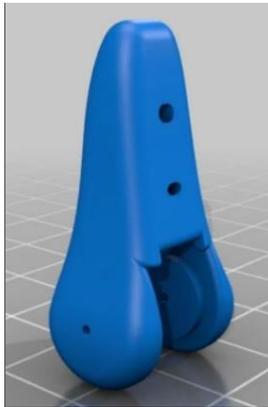


Figura 4. Pulgar parte superior.

Fuente: <https://www.thingiverse.com/thing:2838239>



Figura5. Dedo anular.

Fuente: www.thingiverse.com/thing:2838239

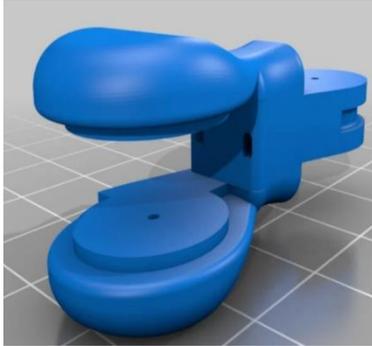


Figura 6. Pieza mitad pulgar

Fuente: <https://www.thingiverse.com/thing:2838>



Figura 7. Tope final servo

Fuente: <https://www.thingiverse.com/thing:2838>

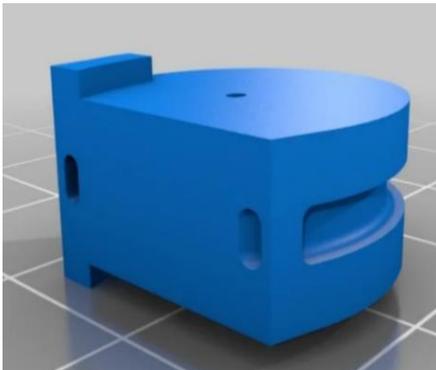


Figura 8. Parte inferior del pulgar

Fuente:

<https://www.thingiverse.com/thing:2838>



Figura 9. Dedo meñique

Fuente: <https://www.thingiverse.com/thing:2838>

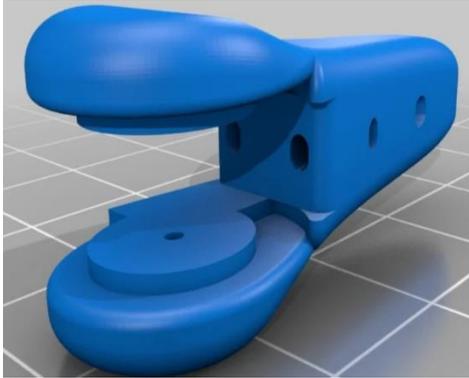


Figura10. parte superior de dedo

Fuente: imagen extraída de (BriBro12, 2018)
<https://www.thingiverse.com/thing:2838>

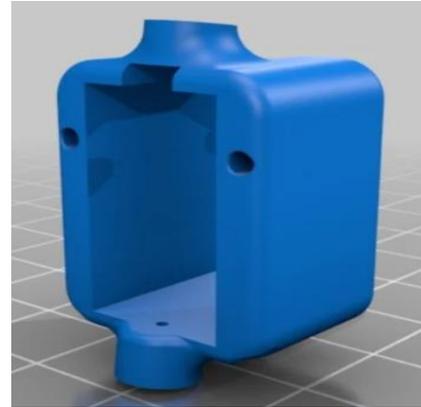


Figura11. soporte servo

Fuente: imagen extraída de (BriBro12, 2018)
<https://www.thingiverse.com/thing:2838>

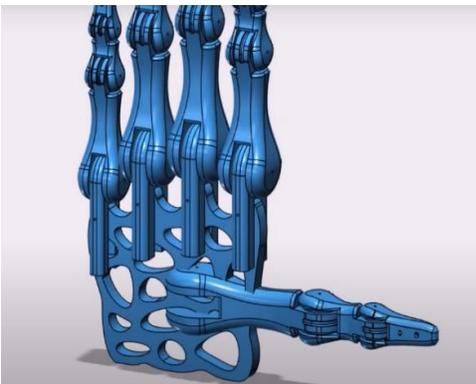


Figura 12 hand

Fuente: extraída de (BriBro12, 2018)
<https://www.thingiverse.com/thing:2838>

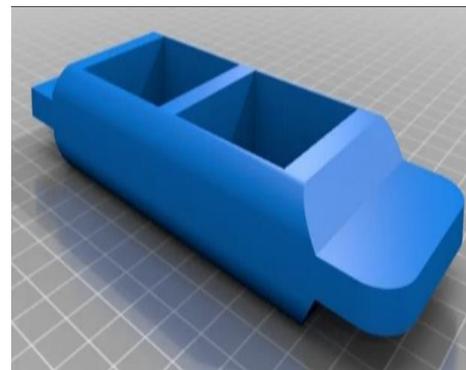


Figura13. muñeca

Fuente: imagen extraída de (BriBro12, 2018)
<https://www.thingiverse.com/thing:2838>

: imagen

Este diseño fue realizado en el software, fusión 360 de Autodesk.

El material utilizado para imprimir, los componentes para armar esta prótesis, fue filamento de PLA este filamento ácido poliláctico, es un termoplástico fabricado a base de recursos renovables como el almidón de maíz, raíces de tapioca o caña de azúcar. A diferencia de otros materiales de la industria hechos principalmente a base de petróleo.

El PLA fue creado en la década de 1930 por el químico estadounidense Wallace Carothers, más reconocido por el desarrollo del nailon y el neopreno en la empresa química DuPont. Pero no fue hasta la década de los 80 que el PLA fue finalmente producido para su utilización en nuestra vida cotidiana.

Elegimos este material debido a que es económico, efectivo, biodegradable y tolera temperaturas de fusión de hasta 180°C, adquirimos un royo de in 1kg de este material por un coste accesible en nuestra ciudad este rondaba unos 21,72 dólares.



Figura 14. PLA
Fuente: diseño propio

Para el proceso operativo de la fabricación de cada una de las partes que componen La prótesis, utilizamos el Software Cura , Esta es una aplicación diseñada para impresoras 3D, en la que se pueden modificar los parámetros de impresión y después transformarlos a código G, el código G es un lenguaje mediante el cual las personas pueden decir a máquinas (herramienta controladas por computadora) qué hacer y cómo hacerlo, están definidos mayormente por instrucciones sobre a dónde moverse, cuán rápido moverse y qué trayectoria seguir. Las máquinas típicas que son controladas con G-code son fresadoras, cortadoras, tornos e impresoras 3D.

Su funcionamiento, consta en dividir el archivo del modelo de un usuario en varias capas, generando un código G para la impresora 3D. Cuando ya se haya finalizado el proceso, el código G se envía a la impresora por medio de un dispositivo de almacenamiento o vinculando la impresora con el software por medio de una conexión, para que después la impresora fabrique el objeto físico.

Este software es de código abierto ,compatible con la mayoría de las impresoras 3D de escritorio que puedan trabajar con archivos en formatos de 3D más comunes como STL, OBJ, X3D, 3MF, entre otros. También pueden trabajar formatos de imagen como BMP, GIF, JPG y PNG



Figura 15. cura logo

Fuente extraído de:

[https://es.wikipedia.org/wiki/Cura_\(software\)#/media/Archivo:Ultimaker_cura.png](https://es.wikipedia.org/wiki/Cura_(software)#/media/Archivo:Ultimaker_cura.png)

5.2 parámetros y configuraciones de impresión

- Resolución: 0,2
- Relleno: 25
- Temperatura de impresión: 200
- Temperatura de la placa de impresión:60
- Impreso a 2 conchas perimetrales, con material PLA de color blanco.
- Velocidad :50mm/s

a continuación, se pueden observar en las siguientes imágenes:

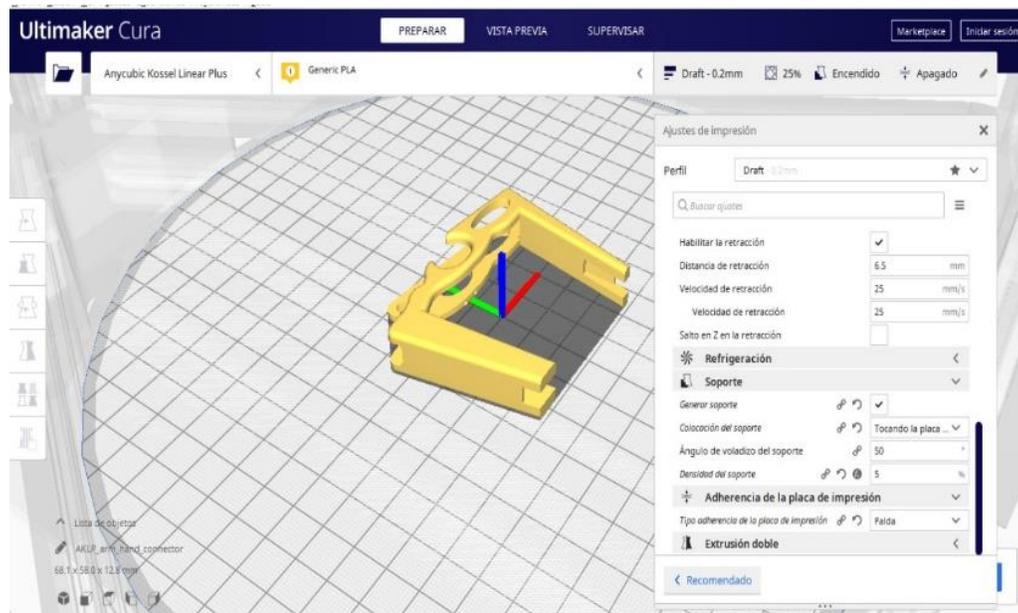


Figura 16. Wrist support
Fuente: autoría propia

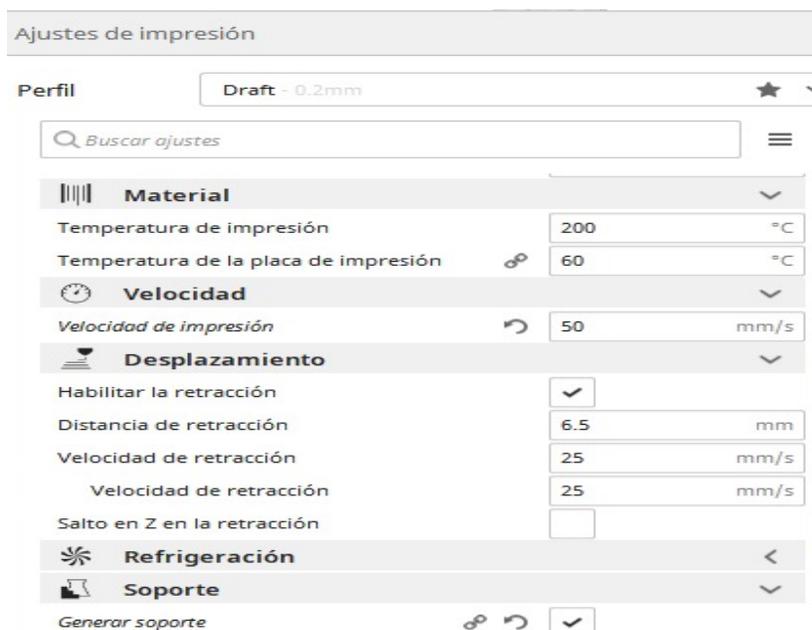


Figura 17 Desplazamiento y refrigeración.

Fuente: soporte de muñeca, parámetros de impresión, imagen de elaboración propia



Figura 18. parámetros

Fuente: Evidencia de desplazamiento parametrizado para la impresión de pieza, parte superior Dedo pulgar, imagen de captura de pantalla del software cura, elaboración propia.

Piezas en cama para impresión:



Figura 19 Partes distales,
Fuente: imagen de autoría propia.

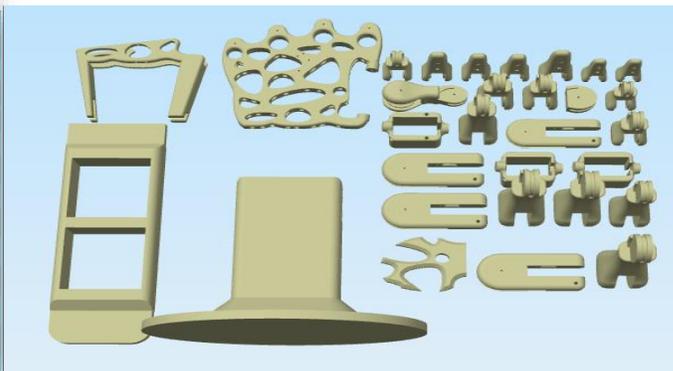


Figura 20. Piezas modeladas
Fuente: imagen de autoría propia.

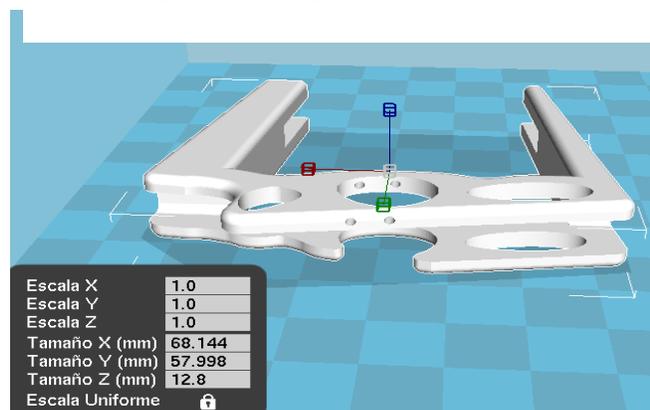


Figura 21. Base muñeca
Fuente: imagen de autoría propia.

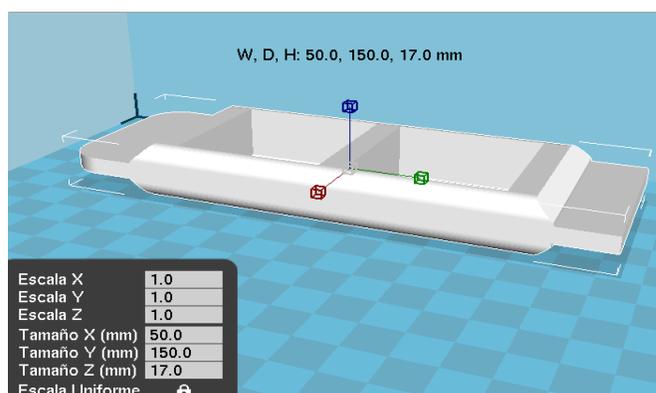


Figura 22. Base servomotores
Fuente: imagen de autoría propia.



Figura 23. Piezas móviles
Fuente: imagen de autoría propia.

5.3 Impresión de Componentes

La impresora utilizada fue la Ender 3, esta permite imprimir de forma asequible numerosos tipos de filamento: PLA, ABS, PETG, flexible, entre otros en un volumen de impresión por encima del estándar, Tiene un diseño innovador, y sobre todo es muy fácil de manipular para realizar cualquier cambio de pieza o mejora, o simplemente para trabajar con ella.



Figura 24. impresora 3D

Fuente extraído de <https://artecosas.comercian.co/impresora-3d-creality-cr-6-se>

Características:

- Impresora 3D Open source (permite cualquier tipo de modificación)
- Filamentos de terceros.

- Materiales: PLA, ABS, PETG, Flexible 95A, composites, etc.

Especificaciones técnicas:

- Tipo de extrusión: FDM (Sistema Bowden).
- Volumen de construcción: 220 x 220 x 250 mm.
- Diámetro filamento: 1,75 mm.
- Diámetro boquilla: 0.4 mm.
- Espesor de capa: 0.1 – 0.35 mm.
- Precisión: +/-0.1 mm.
- Temperatura máxima extrusor: 255 °C.
- Temperatura máxima de cama: 110 °C.
- Velocidad máxima de trayecto: 180 mm/s.
- Formatos modelo 3D: STL, OBJ, G-Code
- Conectividad: Tarjeta SD y cable USB.
- Sistemas operativos compatibles: Windows/ Mac/ Linux.
- Chasis: Perfiles V-Slot de aluminio.
- Dimensiones de la impresora 3D: 440 x 410 x 465 mm.
- Peso: 8.6 Kg.
- Peso neto: 10 Kg.

Las piezas impresas de la prótesis se pueden evidenciar a continuación:



Figura 25. Foto de Piezas impresas

Fuente: imagen de autoría propia.

La mano está constituida por 30 piezas las cuales constituyen los 5 dedos (pulgares, índice, corazón, anular, meñique, la palma de la mano y la muñeca), Sin embargo, las piezas para el pulgar y otras partes presentaron Warping por lo que al momento de armarlas

Y articularlas con el nailon, se rompían y era necesario reimprimirlas, y aun así a algunas de estas les sucedía lo mismo, sumado a las complicaciones, presentadas en la ciudad por el COVID 19, no se pudieron volver a imprimir piezas para dedo pulgar, por ende, el prototipo y ensamblaje de la prótesis se realizó solo con 4 dedos, los cuales iban amarrados desde la parte superior de cada uno respectivamente Por sus orificios que conducían hasta la muñeca y de ahí a los servos, para el control de contracción y retorno de la mano lo que se traduce en cerrar y abrir la misma.

6. Ensamble de la prótesis

Para el ensamble de todas las partes de la prótesis, fue necesario pulir cada una de estas piezas, lijarlas, lubricarlas con cloruro de sodio, y posteriormente con un bisturí con el fin de quitarle los excesos.

Una vez las piezas estaban listas, se procedió a armar cada dedo según el acople de su pieza encajando cada parte a hasta formar los 4 dedos de la mano, para articularlos se usó nailon transparente ya que este tiene resistencia a la fatiga y al desgaste y cuenta con muy buena propiedad al deslizamiento. El procedimiento fue el siguiente.

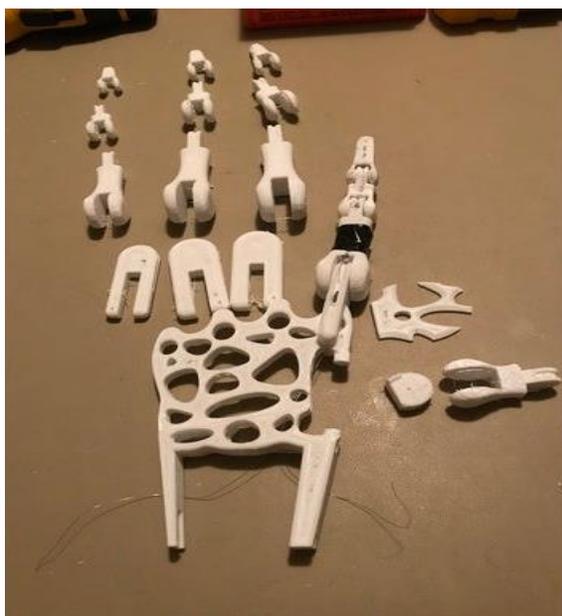


Figura 26. Mano desarmada ,
Fuente: imagen de autoría propia.

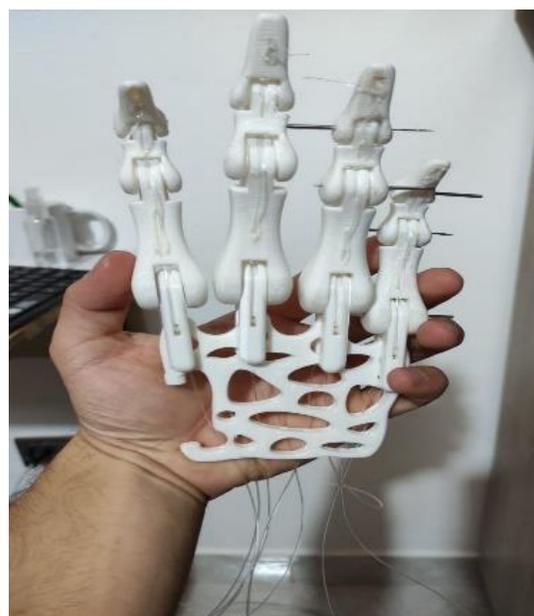


Figura 27. Mano ensamblada
Fuente: imagen de autoría propia.

En la parte electrónica, se utilizaron los siguientes componentes

- 4 servos motores
- 1 batería (24v)
- 1 Arduino Uno
- 1 sensor muscular
- 3 electrodos musculares
- X Cables, protoboard, pines.

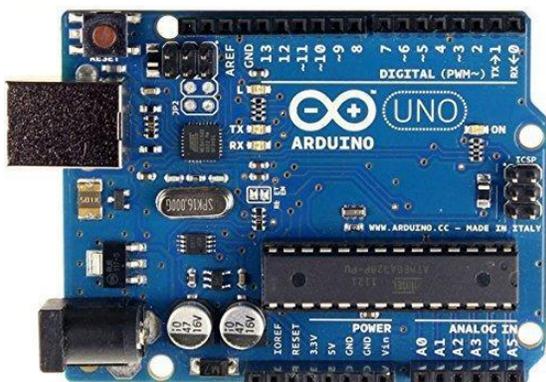


Figura 28 . Arduino uno

Fuente: <https://articulo.mercadolibre.com.co/MCO->



Figura 29. Sensor myoware

Fuente: imagen de autoría propia.

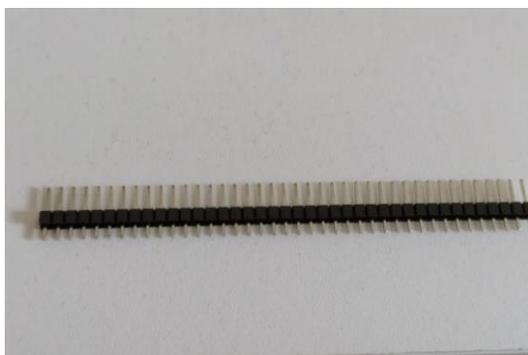


Figura 30 Pines

Fuente: imagen de autoría propia.

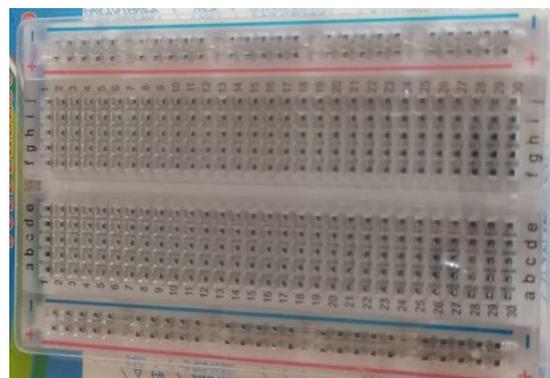


Figura 31. Board

Fuente: imagen de autoría propia.

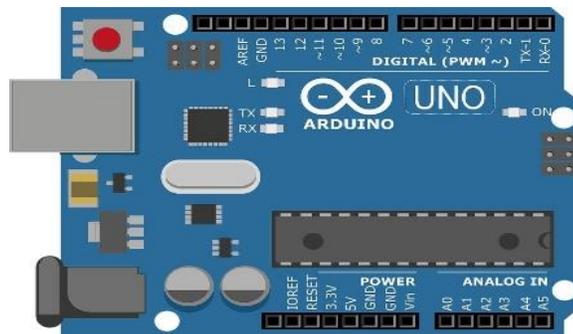


Figura 35. Arduino uno

Fuente: <http://cursoarduino.proserquisa.com/2016/10/05/introduccion/>

En nuestra prótesis este actúa como el cerebro, de todos los procesos que, se ejecutan. La señal análoga captada por el sensor EMG, es transmitida a este dispositivo el cual controla Los servomotores asignados, para que la mano reaccione, abriendo o cerrando los dedos, según el movimiento hecho por el individuo.

6.2 Servos motores

los servos motores utilizados en este proyecto, fueron 4 de la referencia. MG995 es un servo digital de un gran par y 180°, hasta 12 kg x cm y con una extremada resistencia gracias a que sus engranajes son metálicos para soportar mayores esfuerzos.

Este es ya un servo con una buena capacidad y que viene con todos los horns que acompañan a habitualmente a este tipo de motores, estos dispositivos se encargan del control de

los dedos índice, medio, anular y meñique.



Figura 36. Servomotores, (electronilab, 2020)

Nota: El servo MG995 destaca por su buen torque (11Kg), engranajes metálicos y gran robustez. Es utilizado principalmente en proyectos de robótica, como brazos robóticos y robots bípedos. Puede rotar aproximadamente 180 grados (90° en cada dirección). Imagen tomada de (electronilab, 2020)

Fuente: <https://coldfire-electronica.com/esp/item/26/6/servomotor-towerpro-mg995-15kg-cm>

6.3 Características del MG995 Tower Pro.

- Tipo de Interfaz: Analógica
- Dimensiones (L x W x H): 40.6 x 19.8 x 42.9 mm (1.60 x 0.78 x 1.69 pulgadas)
- Peso: 55 gramos
- Batería Par a 4.8 V: 10.00 kg/cm.
- Voltaje de operación: 4.0 a 7.2 volts
- Velocidad de giro a 4.8 volts: 0.2 sec / 60 °.

6.4 Sensor MG.

Específicamente El sensor muscular MyoWare de Advancer Technologies, mide, filtra, rectifica y amplifica la actividad eléctrica de un músculo, y produce una señal análoga de salida que puede ser fácilmente leída por un microcontrolador, lo que permite una novedosa interfaz controlada por un músculo.

Con el fin de adherirse a la piel, el sensor necesita tres electrodos los que se ponen a presión en los conectores (tipo broche) del sensor, lo que facilita conectar y desconectar los electrodos. Dos de los conectores se encuentran en la PCB, y el tercero se encuentra en el electrodo de referencia.

La mejor forma de alimentar el MyoWare es con el mismo voltaje lógico que el resto del sistema (por ejemplo, 5 V cuando se usa un microcontrolador de 5 V). Si enciende el sensor con una batería u otra alimentación de voltaje más alto, la salida puede ser tan alta como el voltaje de alimentación (Vs), se recomienda tomar las precauciones (como ajustar la ganancia) para asegurarse de que la señal no exceda lo que el resto del sistema puede manejar.



Figura.37 sensor MG

Fuente: <https://www.sigmaelectronica.net/producto/sen-13723/>

Características

- Voltaje de funcionamiento: 2,9 V a 5,7 V
- Voltaje de salida análogo: 0 V a +Vs (voltaje de alimentación)
- Ganancia ajustable
- Salidas disponibles: EMG envelope y EMG bruto
- Conectores de electrodos embebidos, los electrodos encajan directamente en MyoWare (alternativamente, se pueden conectar cables de electrodos externos)
- Indicadores ledes: un LED de potencia, y otro LED se ilumina cuando se flexiona el músculo
- Interruptor de alimentación
- Protector de voltaje inverso
- Dos orificios de montaje (apto para tornillos M3 ó # 4)
- Tamaño: 53,3 mm x 20,3 mm (sin el cable del electrodo de referencia, que es de ~76,2 mm de largo)
- Peso: 7,5 g (sin electrodos)

Especificaciones generales:

- Voltaje mínimo de funcionamiento: 2,9 V
- Voltaje máximo de funcionamiento: 5,7 V
- Protección de voltaje reverso: s
- Corriente de alimentación típico: 9mA

6.5 Control.

El cuerpo humano es capaz de generar señales eléctricas en sus músculos, a estas señales se les conoce como señales bioeléctricas. Las señales bioeléctricas se dividen en diferentes tipos dependiendo de la procedencia de dicha señal.

Las señales mioeléctricas son las generadas por la contracción de algún músculo de cualquier extremidad como lo son los brazos y las piernas, pueden medirse con un equipo adecuado y así utilizar la información que estas nos proporcionan en el diseño de prótesis, La señal muscular o mioeléctrica (SME) utilizada para la activación de las prótesis tiene las siguientes características amplitud de entre $50 \mu\text{V}$ y 5 mV , y un rango de frecuencia de 20 Hz a 500 Hz . Por lo que para la elaboración del prototipo de prótesis primero se registra la SME mediante electrodos de plata -cloruro de plata.

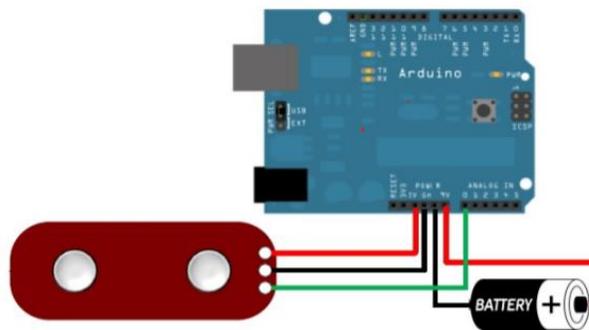


Figura 38, Esquema conexión sensor MG.

Fuente: <https://shop.pimoroni.com/products>

El sensor muscular MyoWare es un sensor de electromiografía (EMG) de Advancer Technologies. La placa MyoWare actúa mediante la medición de la actividad eléctrica filtrada y rectificadora de un músculo; la salida de 0-V Volts dependiendo de la cantidad de actividad en el músculo seleccionado, donde V_s representa el voltaje de la fuente de alimentación. Para colocarlo a la piel es necesario tres electrodos de almohadillas que se conectan al sensor por medio de broches lo que hace fácil de poner y quitar. Dos de los electrodos son colocados directamente a la placa del sensor mientras que el tercero se coloca al final de una extensión de cable que trae.

Ensamble físico.

Una vez teniendo todos los componentes electrónicos a utilizar y todos los objetos del prototipo impreso, se procede a ensamblar cada uno de ellos, para hacer las pruebas y realizar el funcionamiento, Los servos motores fueron posicionados vertical mente, en la muñeca de la mano, en los oficios Rectangulares que traía por defecto el diseño de la misma.



Figura 39. Mano ensamblada
Fuente: imagen de autoría propia

7.programación y control de prótesis.

71 Diseño del algoritmo para reconocimiento de gestos y programación.

Para poder realizar el movimiento propio de la mano, es necesario la realización de un código, que plantee la actuación o el trabajo que se va a realizar, Inicialmente debíamos comprender cómo es la estructura del entorno. Podemos determinar que se divide en tres secciones:

1. Declarativa: En esta primera sección estructural del código, debemos incluir las librerías que se harán uso, debemos definir los periféricos de entrada y salida de nuestro proyecto y declarar a qué pines del microcontrolador se han conectado estos periféricos.

2. Inicialización: La segunda área de trabajo comienza en Arduino a partir de la llamada void setup { }. En este punto, se especifica el sistema de arranque del programa, es decir, se determina el comando a ejecutar al comienzo. Una vez se inicializa el programa 86 esta sección, no se ejecuta más, es decir, tan solo se ejecuta una vez.

3. Ejecución en bucle: Esta tercera parte del código comienza con la llamada void loop { } y es en este punto en el cual, se introducen los comandos que conforman la función principal del programa

Para transformar la detección muscular en un movimiento mecánico realizado por un servomotor, debemos establecer qué relación encontramos entre la información que nos aporta el sensor y cómo podemos manejar dicha información dentro del editor. El sensor al final nos

ofrece señales que las recoge el microcontrolador. A partir de aquí el editor transforma estos datos y trabaja con la tasa de Baudio. Esta tasa se refiere a la velocidad de conmutación que se realiza por segundo, es decir la variación de voltaje o de frecuencia que se encuentra en la señal. Esta se expresa en símbolos por segundo. Por tanto, cada vez que se detecta dicho cambio se genera una variación del baudio.

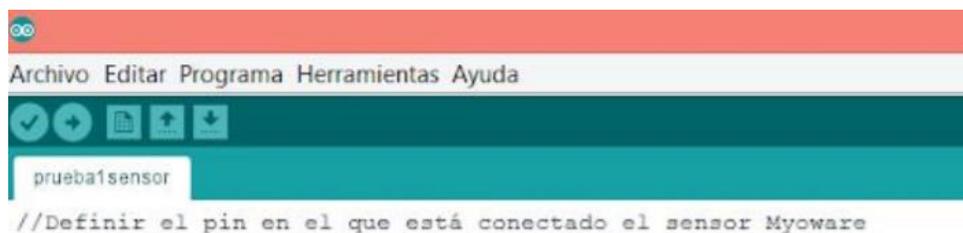


Figura 40. Definición pines
Fuente: imagen de autoría propia

Luego, en la estructura de inicialización, declararemos el ratio de baudio al que queremos trabajar. Es importante más tarde que cuando ejecutemos el programa, coloquemos el monitor serial en el mismo ratio de baudio que hemos declarado.

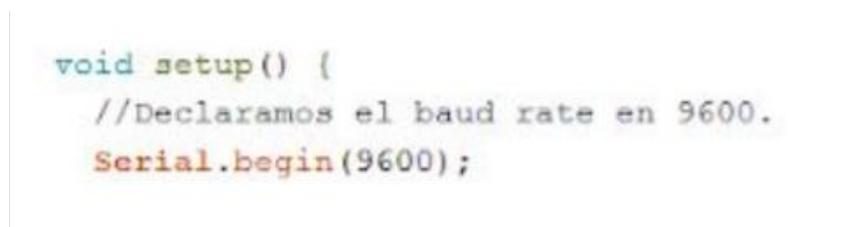


Figura 41. Void setup
Fuente: imagen de autoría propia.

En la estructura bucle, y aquí lo que queremos conseguir es leer los datos recogidos por el sensor. La idea es voltear los datos ofrecidos por el sensor a una variable entera que en nuestro código llamaremos “value”. Y finalmente para que podamos ver los datos leídos, imprimimos en

pantalla dichos valores.

```
void loop() {
  //El valor entero "Value" lee los diferentes valores peidos por el pin analógico conectado al sensor
  int value = analogRead(EMG_PIN);
  //Podemos observar los valores en el monitor serial.
  Serial.println(value);
}
```

Figura 42. Void loop

Fuente: imagen de autoría propia

Para comprobar que funciona debemos compilar, y si todo está correcto volcamos el programa al microcontrolador. De manera provisional alimentaremos el conjunto del proyecto a partir de la energía que nos ofrece el portátil. Dicha energía y toda la información que vuelca al microcontrolador se realiza a través de un cable USB. Por supuesto, antes de poder volcarlo en el microcontrolador es necesario, realizar las diferentes conexiones del sensor con el dispositivo Arduino.

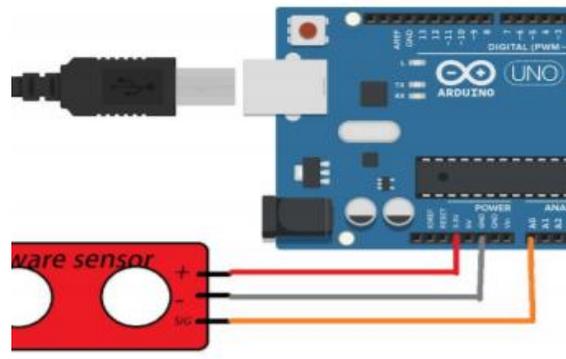


Figura 43 Conexión sensor.

Fuente: imagen de autoría propia

7.2 Validación del algoritmo en conjunto con el sensor.

Se realizan pruebas con el sensor sobre el músculo Flexor Carpi para corroborar el funcionamiento del algoritmo programado y determinar una tasa de aciertos, Para la visualización de los valores en pantalla, tenemos dos opciones que nos ofrece el editor esto fue previamente explicado por nuestro asesor asignado. La primera opción es el monitor serial, este lo que nos ofrece directamente es el número leído en la ratio de baudio, es decir, si al valor leído y que se ha transformado para poder comprenderlo es un valor entero de 280, en el monitor serial nos aparecerá impreso el número 280 en cascada o en serie.



Figura 44. Lectura monitor serial
Fuente: imagen de autoría propia

La segunda opción que nos ofrece el editor es el serial plotter. Esta función lo que nos permite es ver de forma gráfica esa lectura de datos

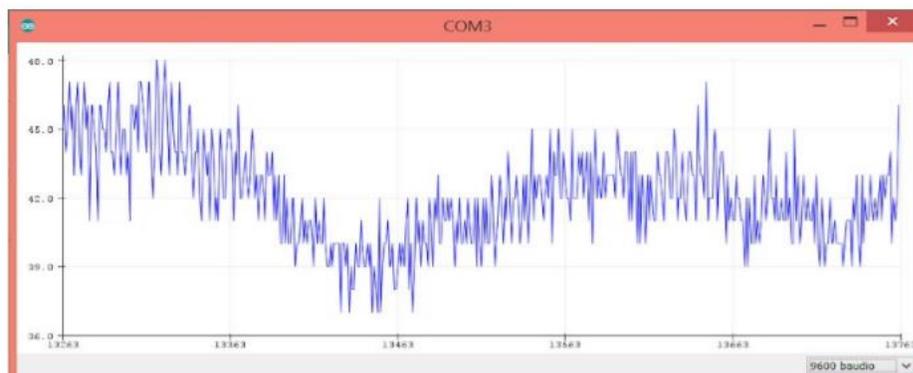
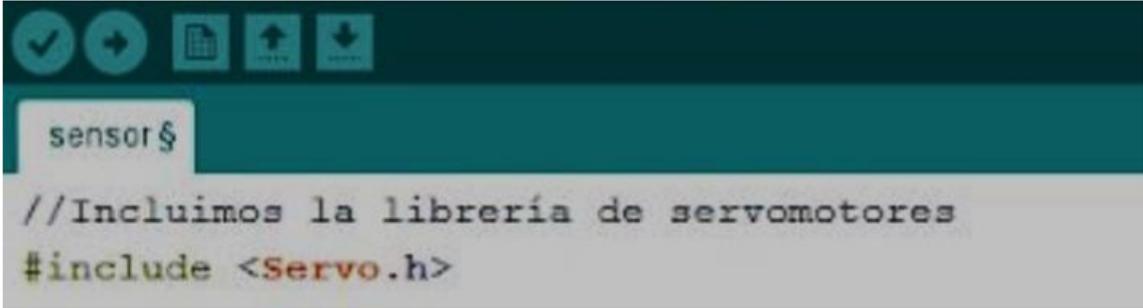


Figura 45. Lectura sploter
Fuente: imagen de autoría propia

Viendo como es el funcionamiento del sensor, debimos plantear cómo plasmar la idea de movimiento de los dedos, a partir de la señal captada. Se llegó a la conclusión de que lo más factible era, obtener un valor umbral mediante la observación de la señal muscular, que nos determine que a partir de ese punto el usuario está realizando la acción de fuerza, para poder mover los dedos

Para inicializar el código, lo primero que haremos es hacer uso de una librería que contiene porciones de códigos, que nos permite hacer 91 más sencillo el programa. En este caso lo que queremos incluir es la librería referente al servomotor.

A screenshot of a code editor window with a dark teal background. At the top, there is a toolbar with icons for a checkmark, a right arrow, a document, an upload arrow, and a download arrow. Below the toolbar, a search bar contains the text "sensor \$". The main code area shows two lines of C++ code: "//Incluimos la librería de servomotores" and "#include <Servo.h>".

```
sensor $  
  
//Incluimos la librería de servomotores  
#include <Servo.h>
```

Figura 46. Librería

Fuente: imagen de autoría propia

Seguidamente definiremos los periféricos de entrada y salida que se usarán, así como el pin al que están conectados. Como bien desarrollamos en la prueba del sensor, definimos el mismo sensor y a qué pin está conectado. Mantendremos el pin analógico 0. A continuación, definimos el nombre del servomotor que usaremos y declaramos a qué pin digital está conectado que en este caso será el 3. Al plantear la idea de crear un límite que nos determine cuándo ha de realizarse el movimiento del servomotor, definiremos una variable que llamaremos “THRESHOLD” y diremos cuál es el límite. Debemos tener muy presente esta definición, dado que cuando queramos hacer uso del programa, previo a usarlo debemos realizar una calibración.

Esta calibración se debe a que el sensor se colocará cada vez que hagamos pruebas en una zona en concreto, pero nunca será el mismo punto, por tanto, los datos de lectura que tendremos serán diferentes y por consiguiente el límite que marquemos también variará.

```

//Se establecerá un límite en concordancia con el valor máximo y mínimo del sensor.
#define THRESHOLD 200

//Definimos el número del Pin donde estará conectado el sensor. En este caso es el pin analógico 0.
#define EMG_PIN 0

//Declaramos el número del pin donde estará conectado el servomotor. Será el pin digital 3 (Digital PWM 3)
#define SERVO_PIN 3

//Definimos el servomotor
servo SERVO_1;

```

Figura 47. Void setup

Fuente: imagen de autoría propia

Posteriormente implementamos en la función de arduino void setup(). Tal y como planteamos en el funcionamiento del sensor, en este punto estableceremos la inicialización de operar el puerto serie a la velocidad que deseemos, a través del comando Serial.begin, e indicaremos que el servomotor que hemos definido como SERVO_1 irá conectado al pin que hemos declarado el void loop().

```

void setup() {

  //declaramos el baud rate en 9600, remember it to set monitor serial properly.
  Serial.begin(9600);

  //Conectamos la definición del servomotor con el pin donde estará conectado (digital pin 3)
  SERVO_1.attach(SERVO_PIN);
}

```

Figura 48 Parámetros

Fuente: imagen de autoría propia

En este punto lo que realizaremos es la condición de que si la señal leída por el sensor, supera el umbral que se ha definido, el servomotor debe girar 179 grados y si no se supera, el servomotor se mantendrá en su posición original. Mantendremos la declaración de la variable entera “value”, ya usada para el código de lectura de la señal del sensor. Usaremos las funciones condicionales if y else, para realizar el movimiento del servomotor si el valor leído supera el “Threshold” o que no se mueva si no lo supera. Y finalmente, mantenemos la impresión por pantalla de los valores leídos por el sensor.

```

/*----- void loop -----*/
void loop()
{
  //El valor entero "Value" lee los diferentes valores leídos por el pin analógico conectado al sensor
  int value = analogRead(EMG_PIN);

  //Si el valor del sensor supera el límite que hemos establecido, queremos que el servomotor rote 180 grados.
  if(value > THRESHOLD){
    SERVO_1.write(179);
  }

  //Si no supera el límite establecido no rota.
  else{
    SERVO_1.write(0);
  }

  //Podemos observar en el monitor serial, los valores para poder ajustar el límite deseado mientras abrimos y cerramos la mano.
  Serial.println(value);
}

```

Figura 49 Void loop

Fuente: imagen de autoría propia

Mantuvimos el sensor con su mismo pin de conexión que la entrada analógica 0. Tras este paso, definiremos los periféricos de salida que serán los cuatro servomotores, que definiremos del mismo modo que realizamos para la definición de un dedo. Debemos aportar a cada servomotor un nombre diferente e introducir a que pin estará conectado.

```

//Declaramos el numero del pin donde estará conectado el servomotor.
#define SERVO_PIN 3
#define SERVO2_PIN 4
#define SERVO3_PIN 5
#define SERVO4_PIN 6

//Definimos el servomotor
Servo SERVO_1;
Servo SERVO_2;
Servo SERVO_3;
Servo SERVO_4;

```

Figura 50 . Control servos

Fuente: imagen de autoría propia

En el bloque setup(), donde declararemos las líneas de código a ejecutar al inicio del programa. Por tanto, como en el programa para un solo dedo declaramos el ratio de Baudio y seguidamente debemos introducir la conexión de cada servomotor con su pin correspondiente

```

/*----- void setup -----*/
void setup(){

//declaramos el baud rate en 9600, remember it to set monitor serial properly.
Serial.begin(9600);

//Conectamos la definición del servomotor con el pin donde estará conectado (digital pin 3)
SERVO_1.attach(SERVO_PIN);
SERVO_2.attach(SERVO2_PIN);
SERVO_3.attach(SERVO3_PIN);
SERVO_4.attach(SERVO4_PIN);

}

```

Figura 51. Ratio

Fuente: imagen de autoría propia

En el bloque bucle, en el que mantendremos las líneas de código incorporadas en el programa de movimiento de un dedo, pero introduciendo en la función condicional las 96

acciones de los otros servomotores. Mantendremos los grados de giro para todos los servomotores en 80

```

----- void loop -----
void loop(){
//El valor entero "Value" lee los diferentes valores leídos por el pin analogico conectado al sensor
int value = analogRead(EMG_PIN);

//Si el valor del sensor supera el límite que hemos establecido, queremos que el servomotor rote 180 grados.
if(value > THRESHOLD){

    SERVO_1.write(80);
    SERVO_2.write(80);
    SERVO_3.write(80);
    SERVO_4.write(80);

}

//Si no supera el límite establecido no rota.
else{
    SERVO_1.write(0);
    SERVO_2.write(0);
    SERVO_3.write(0);
    SERVO_4.write(0);
}

//Podemos observar en el monitor serial, los valores para poder ajustar el límite deseado mientras abrimos y cerramos la mano.
Serial.println(value);
}

```

Figura 52. Bluce

Fuente: imagen de autoría propia

```

void loop() {
    lectura = analogRead(A0); //se asigna el valor de la entrada del sensor
    voltaje = ((lectura/1023)*5); //se ajusta y se escala.
    Serial.println(voltaje);
//dependiendo del valor de del sensor, si mide la mano cerrada y no tiene la bandera de seguridad activa, giramos los servos durante el tiempo suficiente para cerrar los dedos de la mano, dactivamos la bandera
if(voltaje>=3.0 && dir==0){
    dir=1;
    dir2=0;

    servo1.write(180);
    delay(300);
    servo1.write(88);
    servo2.write(180);
    delay(500);
    servo2.write(90);
    servo3.write(0);
    delay(600);
    servo3.write(90);
    servo4.write(0);
    delay(600);
    servo4.write(90);
    servo3.write(180);
}
}

```

Figura 53. Programa

Fuente: imagen de autoría propia

Dependiendo del valor de del sensor, si mide la mano cerrada y no tiene la bandera de seguridad activa, giramos los servos durante el tiempo suficiente para cerra los dedos de la mano, activamos la bandera de seguridad y desactivamos la bandera de seguridad para el giro inversa.

Se estableció la siguiente condición, realizando para el parámetro de las banderas en cada ciclo:

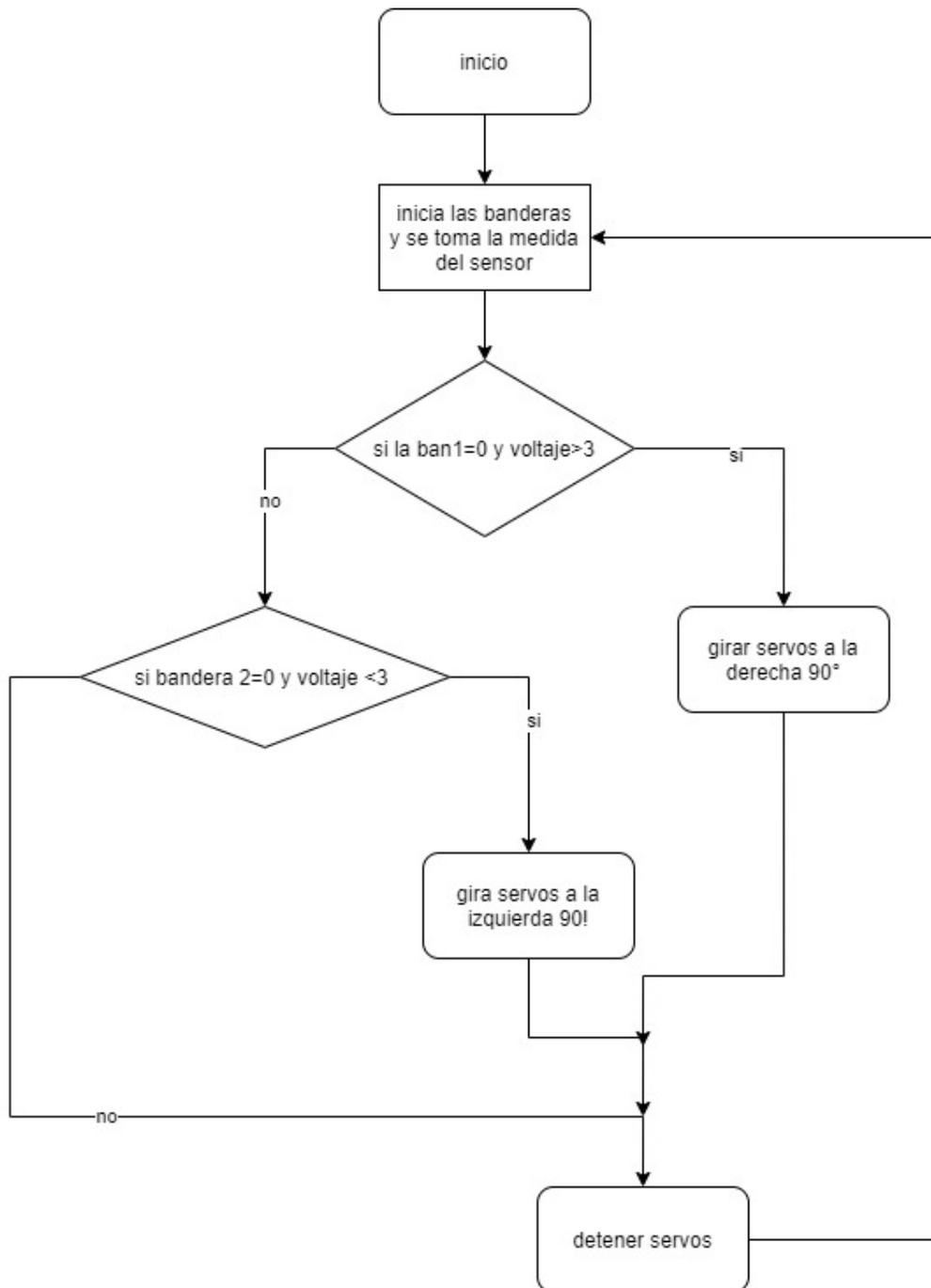
- `if(voltaje>=3.0 && dir==0){`
- `dir=1;`
- `dir2=0;`

Lo anterior significa que si el voltaje era mayor a ≥ 3 v la mano cerraría, para que esto sucediera, era necesario empuñar la mano, para que el sensor tratara de leer la señal, y actuara una vez des tensionemos nuestra mano y el voltaje este en 0, está la bandera reversa, servomotores a su estado original.



Figura 54. Montaje completo
Fuente: imagen de autoría propia

7.3 Diagrama de flujo de código.



Finalmente realizamos la prueba con los servomotores y comprobamos que, en el momento de detección de contracción muscular, los servomotores realizaban su movimiento de giro en el ángulo determinado y en el orden que queríamos que trabajaran. Es cierto, que a medida que se realizaban las diferentes pruebas, hubo complicaciones al encontrar fallos como la existencia de servomotores que no se movían en algunas ocasiones y en otras sí. Se decidieron hacer diferentes cambios como pudo ser por ejemplo usar otros servomotores o cambiar estos de posición, pero seguían existiendo estos fallos. Tras seguir indagando no podíamos encontrar el error hasta que se decidió hacer una prueba excepcional. Se realizaron pruebas individuales de cada uno de los servomotores y todos funcionaban correctamente cuando se conectaban de forma individual, después se hacían pruebas por parejas y también funcionaban correctamente. Con tres servomotores seguían funcionando correctamente, pero fue al introducir cuatro cuando empezaron los mismos problemas en los que a veces funcionaban y en otras ocasiones no funcionaban todos. Debido a la falta de tiempo, materiales y condiciones no se pudo comprobar, bien este punto, pero si se pudo verificar que la prótesis funcionaba y el control de esta estaba relativamente bien.

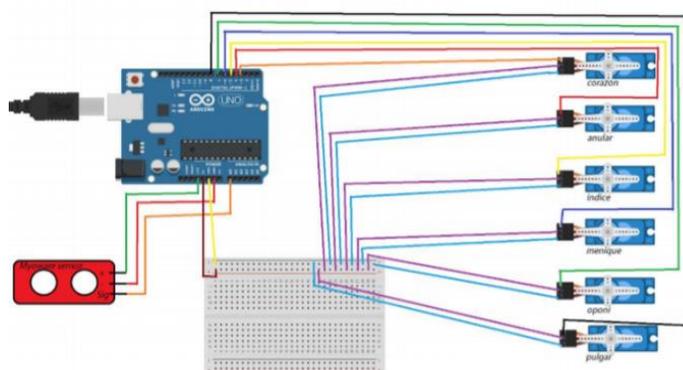


Figura 55. Control y conexiones

Fuente: imagen de autoría propia

8. Conclusiones.

Fue posible la construcción de la mano impresa en 3D, en material PLA de manera rápida con una buena velocidad, sin embargo, esta presentó complicaciones al momento de armarse debido a que muchas piezas presentaban Warping, lo que hizo necesaria la reimpresión de las mismas, Con ajustes paramétricos, los cuales ayudaron a tener una estructura sólida y efectiva para su posterior montaje electrónico.

Se utilizó la información de las señales en el dominio del tiempo para implementar el algoritmo de reconocimiento de gestos, pues el análisis en el dominio de la frecuencia no aportó resultados beneficiosos para el avance del proyecto hacia la construcción de un prototipo.

Los servomotores fueron posicionados con éxito y conectados a una fuente de alimentación de 24v, estos pudieron operar correctamente, en la parte del control, se utilizó como procesador el famoso Arduino uno, esta placa fue programada para hacer mover cada uno de los motores junto con el sensor Muscular precedido de un gesto estipulado, lo cual tuvo éxito, siendo este el resultado más importante de todos ya que responde a la pregunta principal mente planteada, que fue : “ ¿Atreves de la programación e impresión 3d es posible construir una prótesis de fácil adaptabilidad y accesibilidad económica?”

Se demostró que con recursos muy accesibles que en total facturaron un coste estimado de 160 dólares. Fue posible el desarrollo de una prótesis tecnológica que permite el movimiento articular de una extremidad superior.

9. Bibliografía.

Romero, I. P., & Aarón, M. (2011). Análisis cinemático e implementación de una manorobótica servo-articulada aplicable como prótesis (Doctoral dissertation).

Brito, J. L., Quinde, M. X., Cuzco, D., & Calle, J. I. (2013). Estudio del estado del arte de las prótesis de mano.

González, J. M. D. (2005). Robótica y prótesis inteligentes.

Burgos, C. Q., & Albán, O. A. V. (2010). Diseño y construcción de una prótesis robótica de mano funcional adaptada a varios agarres. *Popayán*.

J. Fajardo, A. Lemus and E. Rohmer. " Galileo bionic hand: sEMG activated approaches for a multifunction upper-limb prosthetic ". IEEE Central American and Panama Convention (CONCAPAN 2015). IEEE Explore Digital Library. Tegucigalpa, Honduras. November '15, pp. 1 to 6.

Ismail, M. Ariyanto, W. Caesarendra and A. Nurmiranto. " Development of robotic hand integrated with SimMechanics 3D animation ". International Seminar on Intelligent Technology and Its Applications ISITJA '16. IEEE Explore Digital Library. 28-30 July 2016, Lombok, Indonesia; pp.: 629 to 634.