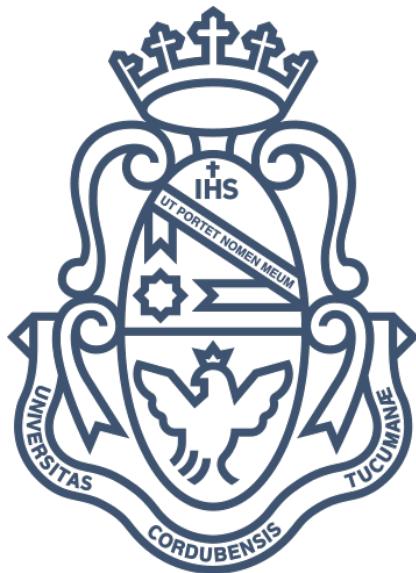


Universidad Nacional de Córdoba
Facultad de Ciencias Exactas, Físicas y Naturales
Facultad de Ciencias Médicas
Ingeniería Biomédica

**Control mioeléctrico y adaptabilidad de la prótesis
Gre-nable**



Alumnos:

Girard, Thibault – 23CI38066

Asesores:

Beltramone, Antonio, Diego



Córdoba, Julio de 2025

Índice

Resumen	vi
1 Introducción	1
2 Objetivos	2
2.1 Perfil del usuario final	2
3 Marco teórico	4
3.1 Anatomía del antebrazo	4
3.2 Anatomía de la mano y de los dedos	4
3.3 La prótesis	6
3.4 Modelo de prótesis	7
3.4.1 Modelo Hero Arm	7
3.4.2 Modelo Bebionic	8
3.4.3 Elastic Hand	9
3.4.4 Proyecto Gre-nable	10
3.4.5 Resumen	10
4 Presentación del proyecto	11
4.1 Modelo de base	11
4.1.1 La mano	12
4.1.2 El guante	13
5 Mecánica	17
5.1 Adaptación a mis objetivos	17
5.1.1 Sistema de ajuste	17
5.1.2 Caja trasera	18
5.1.3 Enlace entre el guante y la mano	21
5.1.4 Parte inferior	22
5.2 Impresión	23
5.2.1 Materiales	23
5.3 Impresoras y software	24
5.3.1 Parámetros importantes	25
5.3.2 Impresión con PLA	31
5.4 Impresión con TPU 95A	32
6 Electrónica	34
6.1 Descripción de los componentes	34
6.2 Circuito completo	39
7 Mantenimiento	42
8 Programación	43
8.1 Parte de control	43
8.1.1 Recuperación de la señal	43
8.1.2 Procesamiento de la señal	43
8.1.3 Detección de patrones	44

8.1.4	Modos de funcionamiento	45
8.2	Control de ejecución y motores	46
8.2.1	Gestión de tareas	46
8.2.2	Control de servomotores	47
9	Pruebas y validaciones	48
9.1	Pruebas	48
9.2	Validaciones	50
10	Perspectivas	53
10.1	Diseño Final	53
10.2	Fabricación y Comercialización	54
10.2.1	Moldeo por Inyección	54
10.3	Ejemplo: Producción de la Mano	56
10.3.1	Moldeos	56
10.4	Mejoras	57
10.4.1	Partes Mecánicas y Activación	57
10.4.2	Partes Electrónicas y Control	58
Apéndice A	Entrevista con Sandrine Calvet	62
Apéndice B	Montaje	64
Apéndice C	Arduino code	68

Índice de figuras

1	Tipos de amputaciones posibles	2
2	Anatomía del antebrazo	4
3	Movimientos de la mano	5
4	Movimientos de la mano	5
5	Prótesis cosméticas	6
6	Prótesis con tracciones	7
7	Prótesis Hero Arm	8
8	Prótesis Bebionic	8
9	Prótesis Bebionic con sistema de acoplamiento	9
10	Prótesis Elastic Hand	9
11	Diagrama de flujos	11
12	Modelo de base	12
13	Estructura de un dedo	13
14	Estructura del pulgar	13
15	Impresiones de mano	13
16	Pieza derecha y izquierda ubicadas entre la palma y el guante	14
17	Ubicación de las piezas entre el guante y la palma	14
18	Descripción del guante	15
19	Cajas sobre el guante	15
20	Ubicación de la rueda de ajuste	17
21	Manija visible al 100 % y al 70 %	18
22	Acople de la manija y sus estados: abierto/cerrado	18
23	Caja trasera	19
24	Primera extensión V1	19
25	Extensión V2, vista trasera y en 3/4	20
26	Ubicación del interruptor	20
27	Caja del receptor V1, vista en 3/4	21
28	Caja del receptor VF, impresa	21
29	Descripción del botón	21
30	Modelización de las tuercas	22
31	Ubicación de la tuerca	22
32	Modelización de la parte inferior	23
33	Movimiento de la parte inferior con el guante	23
34	Impresoras X1C y X1E	24
35	Interfaz del software Bambu Lab	25
36	Parámetros de resistencia en Bambu Studio	26
37	Relleno a 20 %, 40 % y 80 % con patrón hexagonal	26
38	Patrones de relleno a 80 % y 20 % (honeycomb)	27
39	Visualización de soportes en Bambu	28
40	Visualización de la mano con los huecos para los hilos y los parámetros de soporte	29
41	Soportes dentro de los huecos generados por los hilos	29
42	Parámetros avanzados de los soportes	30
43	Diferencia entre la activación del parámetro “max bridge length”	30
44	Visualización de la utilización del parámetro de calidad	31
45	Relleno de las partes flexibone	33
46	Nuevo posicionamiento para la impresión de las partes flexibone	33

47	Descripción de la placa micro Arduino	34
48	Servomotor	35
49	Sensor MyoWare	35
50	Ubicación óptima del sensor	36
51	Shield MyoWare	36
52	ESP32 (receptor)	37
53	Batería	38
54	Regulador	38
55	Interruptor y Led	39
56	Esquema de conexión	40
57	Esquema de soldadura	40
58	Componentes soldados	41
59	Líneas añadidas al código del receptor	43
60	Código de procesamiento de la señal	43
61	Patrón deseado a detectar	44
62	Patrón real de la señal a detectar	45
63	Código para activar o desactivar el bloqueo	45
64	Representación del estado de la prótesis: bloqueada o desbloqueada . .	46
65	Ejecución de tareas periódicas	46
66	Código para accionar los servomotores	47
67	Diseño final	53
68	2 tomas distintas	54
69	Creación de piezas por inyección	55
70	Etapas del proceso de inyección	55
71	Uso de un núcleo para crear un hueco	56
72	Esquema de creación de la palma por inyección	56
73	Vista de corte	57

Índice de cuadros

1	Criterios técnicos de las prótesis	3
2	Evaluación comparativa de prótesis	10
3	Criterios técnicos de las prótesis	10
4	Características del PLA	23
5	Características del TPU 95A	24
6	Comparación de patrones y porcentajes de relleno	27
7	Parámetros de impresión de cada pieza	32
8	Resultados de la prueba n°1 para objetos cilíndricos	50
9	Resultados de la prueba n°1 para objetos ortoédricos	50
10	Validación de la prueba N°2: peso máximo soportado por la prótesis .	51
11	Validaciones de la prueba N°3: funcionalidades programadas	51

Agradecimientos

A mi asesor, Diego Beltramone por su tiempo, comprensión, y sus consejos. Su visión y dedicación por los temas de rehabilitación me enseñaron mucho.

A mi padre y a los representantes y encargados del Laboratorio de impresión 3D de la empresa Continental que me permitieron usar sus impresoras.

A mis padres y a mi hermano por sus asistencias transmitidas constantemente y por sus apoyos incondicionales.

A Julieta Daiana Farfán para estar conmigo desde el inicio de mi doble titulación en Argentina hasta el fin. Por su tiempo y su dedicación a hacerme vivir y descubrir la vida argentina a su máximo.

A todos mis compañeros que encontré durante un año y medio con los cuales pude compartir los viajes, los mates y los asados, creando recuerdos inolvidables.

Resumen

El autor llegó a Argentina para realizar una doble titulación en Ingeniería Biomédica. En Francia, su formación previa era en Ingeniería Mecánica. Durante el segundo semestre, en la materia de Rehabilitación, participó en un proyecto grupal cuyo objetivo fue diseñar una prótesis que permitiera a personas con amputación transradial practicar esgrima. Este proyecto le permitió aplicar sus conocimientos en mecánica adquiridos en Francia. En la misma línea, para su proyecto integrador, decidió combinar los conocimientos adquiridos en ambas titulaciones para desarrollar una prótesis funcional.

Como base, tomó un proyecto francés realizado en 2018, que consistía en el desarrollo de una prótesis para una persona con amputación transmetacarpiana. A partir de este modelo, introdujo modificaciones para permitir el control del movimiento de los dedos mediante señales mioeléctricas. Tuvo acceso a los archivos del proyecto original y los adaptó para integrar las funcionalidades definidas en su anteproyecto. Entre estas, se destacó la posibilidad de que la prótesis pudiera ser utilizada por cualquier persona con amputación transradial y que permitiera la manipulación de objetos a partir de la activación muscular. Además, se propuso que la prótesis pudiera fabricarse con un presupuesto inferior a 200 €.

El desarrollo del proyecto se dividió en tres partes principales: una parte mecánica (impresión y montaje), una parte electrónica y una parte de programación. En la primera fase, se encargó de crear o modificar piezas. Diseñó tres nuevas piezas y modificó otras cuatro. Aumentó la longitud inicial de la prótesis y añadió una pieza inferior para que pudiera ajustarse a distintas longitudes del antebrazo.

Durante la segunda fase, llevó a cabo la impresión de las piezas. Para ello, fue necesario tener un conocimiento sólido de las técnicas de impresión 3D, así como de los parámetros que afectan la calidad de las piezas, tales como la orientación durante la impresión y la ubicación de los soportes.

En la tercera fase, trabajó en el circuito electrónico y la programación. Utilizó como base el circuito del proyecto francés, que ya contaba con componentes de actuación seleccionados y probados. Añadió un interruptor que no estaba presente en el diseño original y modificó la sección de control. Para esta parte, empleó un sensor mioeléctrico capaz de transmitir la señal de forma inalámbrica. Luego, programó el sistema en una placa Arduino para procesar adecuadamente las señales.

Finalmente, realizó el montaje completo de la prótesis y verificó que cumpliera con todos los requisitos establecidos. Al igual que el proyecto base, su trabajo se encuentra en formato open source, lo que permite que cualquier persona acceda a los archivos o al informe para modificar y mejorar la prótesis.

1 Introducción

En Francia, se estima que hay entre 100.000 y 150.000 personas con amputaciones, según un artículo consultado [1]. Cabe destacar que existen muy pocos datos o investigaciones sobre el número total de personas amputadas en cada país. Las causas que pueden llevar a una amputación incluyen accidentes, malformaciones congénitas y enfermedades.

De acuerdo con un artículo de Le Figaro [2], las personas con diabetes presentan un riesgo de amputación siete veces mayor que la población sin esta enfermedad.

Las amputaciones tienen consecuencias significativas en la vida de quienes las padecen. En una tesis realizada en Malí [3], se expone —a través de una investigación— que existe una relación entre la amputación y la depresión: el 75 % de la población estudiada presentó síntomas depresivos.

Según la misma tesis, el 40 % de los pacientes dejó de trabajar debido a la dificultad de reinserción laboral y la reticencia de los empleadores. Además, se observa que las personas más afectadas pertenecen a sectores de bajos recursos.

Un estudio realizado en Canadá [4] demuestra que las personas con bajos ingresos que padecen diabetes tienen tres veces más riesgo de sufrir una amputación, y que la atención hospitalaria presenta disparidades en función del nivel socioeconómico del paciente. A esto se suma el elevado coste de una prótesis, que en la mayoría de los casos supera los 10.000 € [5].

El alto coste de las prótesis comerciales representa una barrera considerable para muchas personas, especialmente en contextos de bajos recursos. Esta situación refuerza la necesidad de soluciones accesibles, funcionales y adaptables a distintos perfiles de pacientes. Una prótesis no solo debe reproducir el movimiento, sino también adaptarse cómodamente a las características físicas del usuario, ser resistente, liviana, y permitir una integración sencilla en la vida cotidiana.

Este proyecto tiene como objetivo diseñar y fabricar una prótesis de mano accesible, funcional y personalizable, utilizando tecnología de bajo coste. Para ello, se emplearán servomotores, hilos de tracción y una placa Arduino como unidad de control. Esta elección permite un desarrollo modular, económico y replicable, ideal tanto para la investigación como para aplicaciones reales.

La prótesis se basa en un sistema de tracción por hilos que simula los tendones naturales de la mano. Los hilos se tensan mediante servomotores, permitiendo el movimiento de los dedos. La relajación de los mismos se asegura mediante piezas flexibles impresas en TPU, que proporcionan la elasticidad necesaria para que los dedos vuelvan a su posición inicial.

2 Objetivos

Los objetivos del desarrollo de esta prótesis se comparten en 2 partes:

Objetivos generales: Diseñar y construir un prototipo de una mano que puede cerrar o abrir los dedos según la recepción de señales mioeléctricas.

Objetivo específicos :

- Diseñarla de manera que pueda ser adaptada a cualquier amputación transradial.
- Permitir la sujeción de objetos de dos maneras distintas.
- Flexionar los dedos y abrirlos.
- Construir una prótesis de bajo costo (<200\$).

2.1 Perfil del usuario final

El usuario final podría ser cualquier adulto que tenga una amputación transradial a nivel del brazo derecho. Con esas especificaciones, se puede imaginar una comercialización de un tipo de prótesis de bajo costo que permitiría reemplazar las funcionalidades básicas de una mano.



Figura 1: Tipos de amputaciones posibles

En la figura 1, se muestran todos los tipos de amputación para los cuales se puede utilizar la prótesis. Se distingua 3 casos:

- la amputación metacarpo-falángica (amputación de todos los dedos)
- la amputación transcarpal (amputación de la mano, preservando la muñeca)
- la amputación transradial (amputación entre el codo y la muñeca)

Son las 3 tipos de amputación para las cuales la prótesis puede adaptarse. En resumen, el perfil del usuario final debe ser un adulto que tiene una de las tres tipos de amputaciones mencionadas arriba. En la tabla 3, se especifica el rango de dimensiones posibles para el antebrazo de un usuario según el tipo.

Amputación	Diámetro muñón (mm)	Diámetro del antebrazo al nivel de la muñeca (mm)	Diámetro del antebrazo al nivel del codo (mm)
Metacarpo-falángica	ancho: [0–73] altura: [0–45]	ancho: — altura: [60–70]	longitud: —
Transcarpal/Transradial	X	ancho: — altura: —	longitud: —

Tabla 1: Criterios técnicos de las prótesis

3 Marco teórico

3.1 Anatomía del antebrazo

Desde el hombro hasta la mano, el brazo cuenta con 28 grados de libertad. Esta característica ya ha sido estudiada y documentada, como se detalla en la tesis de Dominique Boutet [6]. La parte que más contribuye es la mano, mientras que el antebrazo posee 2 grados de libertad (flexión/extensión y rotación).

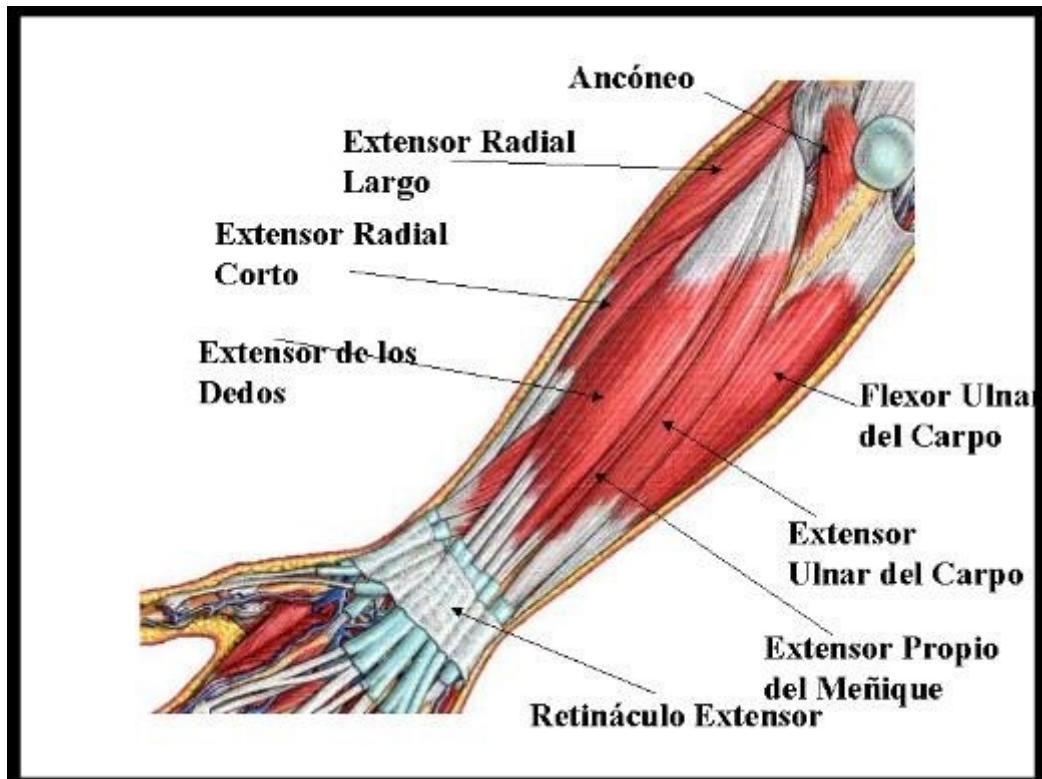


Figura 2: Anatomía del antebrazo

Los principales músculos responsables del movimiento de los dedos son: los músculos flexores y los músculos extensores, que pueden observarse en la figura 2 anatomía del antebrazo. En el caso de una amputación transradial, los grados de libertad del antebrazo deberían mantenerse, ya que el paciente conserva el codo.

Algunas prótesis funcionan leyendo la señal de los músculos; colocar los sensores a nivel de estos es una solución para controlar la prótesis.

3.2 Anatomía de la mano y de los dedos

La mano está articulada al brazo gracias a la muñeca. Esta articulación ofrece 3 grados de movimiento, mostrado en la figura 3:

- Flexión/extensión sobre el eje que atraviesa el ancho de la muñeca.
- Abducción/aducción sobre el eje perpendicular al anterior.
- Rotación alrededor del eje que es la extensión del antebrazo.

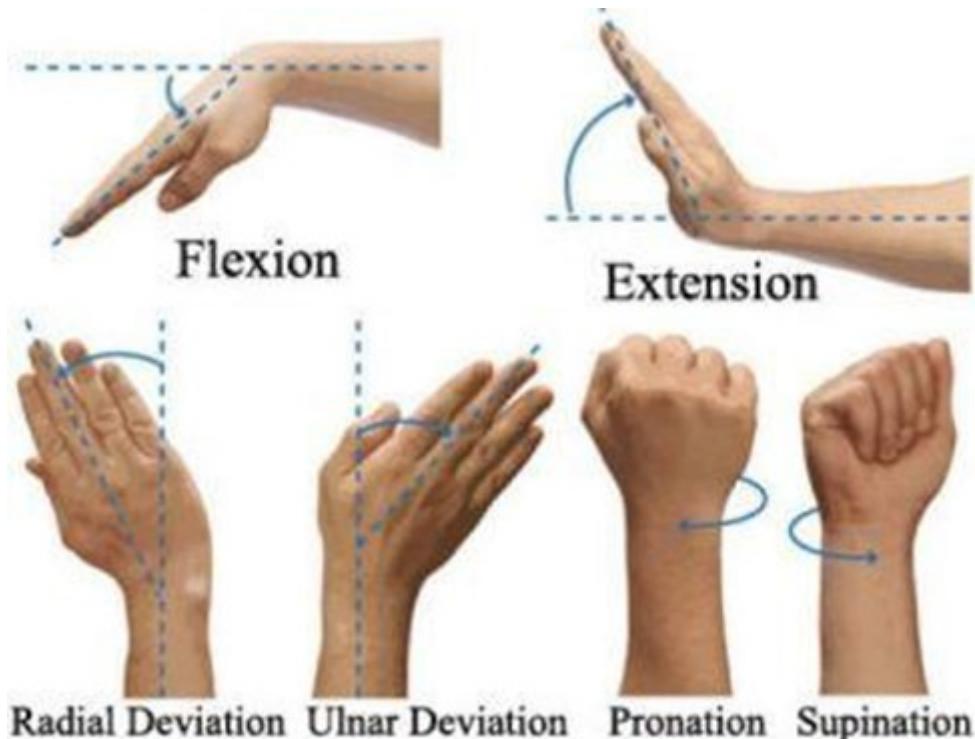


Figura 3: Movimientos de la mano

En cuanto a los dedos, cada uno (excepto el pulgar) cuenta con 4 grados de libertad. Por cada falange se considera una flexión/extensión. Se agrega la abducción/aducción a nivel de la base del dedo, lo que da un total de 4 grados de libertad. El pulgar cuenta con 5.

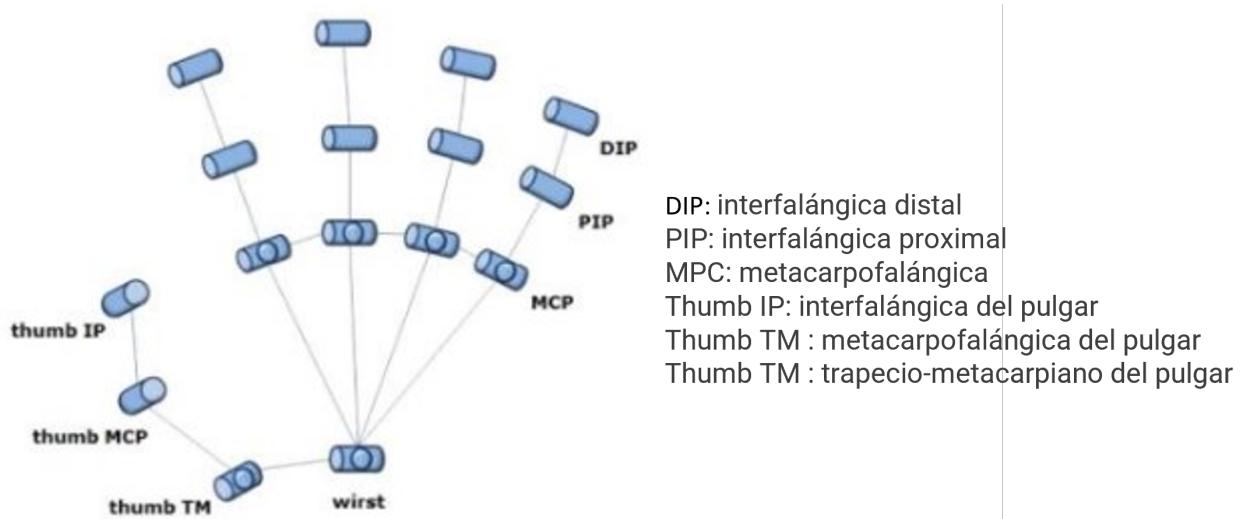


Figura 4: Movimientos de la mano

Como se observa en la figura 4, la estructura de la mano tiene una alta complejidad y es muy difícil de reproducir con la tecnología actual. Las prótesis mantienen la estructura de los dedos, pero los movimientos suelen ser dependientes entre sí, lo cual reduce considerablemente la cantidad de grados de libertad.

3.3 La prótesis

La tecnología de las prótesis se remonta hasta la época de los faraones en Egipto, donde los arqueólogos encontraron prótesis para los dedos del pie como está descrito en el artículo de Eric Roulot [7]. Posteriormente, las prótesis fueron evolucionando según las necesidades de cada época. Durante la Primera Guerra Mundial, ante la gran cantidad de soldados con heridas graves, la medicina tuvo que adaptarse y desarrollar nuevas prótesis con mayor comodidad y más grados de libertad. Actualmente, las prótesis utilizan sensores y motores para imitar de la manera más fiel posible el movimiento del órgano original. También se debe considerar la comodidad del usuario al mismo nivel que la funcionalidad de la prótesis, ya que será una parte del usuario de manera diaria. No se puede desarrollar algo que represente una incomodidad. Existen distintas tecnologías, con diversos materiales y diferentes formas de controlar el movimiento. En este proyecto se aborda el estudio de las prótesis de brazo. En general, se distinguen tres tipos de prótesis, descritas en la pagina web de Ottobock [8].

Prótesis cosméticas: El objetivo de estas prótesis es puramente estético, facilitando las interacciones sociales de las personas con amputaciones. Se hizo una entrevista con una usuaria de una, cuyo resumen esta en la anexa A. Subrayó la importancia de aspecto visual diciendo que no podría salir sin su prótesis. No presentan funcionalidades motoras o presentan solo funcionalidades mínimas. Esto destaca la importancia del aspecto visual que podemos observar en la figura 5.

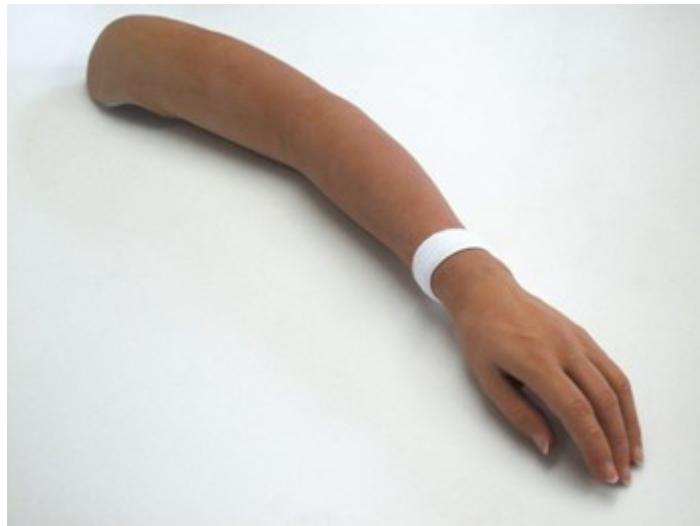


Figura 5: Prótesis cosméticas

Prótesis con tracciones: Este modelo de la figura 6 permite movimientos de la mano o del brazo mediante un sistema de cables accionados por el movimiento de otra parte del cuerpo, como el hombro o el torso. No utilizan electricidad, lo que otorga robustez y fiabilidad a lo largo de los años. Un aspecto importante a considerar, dado el uso diario, es el desgaste del sistema con el tiempo.



Figura 6: Prótesis con tracciones

Prótesis mioeléctricas: Son el tipo de prótesis que ofrece la mayor cantidad de funcionalidades. Combinan actuadores y sensores para reproducir el movimiento de la mano. Esta tecnología las convierte en las más costosas. El precio puede variar desde cientos hasta miles de dólares. En general, si se desea una prótesis personalizada, el precio puede alcanzar los 10.000 dólares. Algunas requieren intervenciones quirúrgicas para su instalación. Las que se conectan directamente a los nervios del cuerpo humano superan los 200.000 dólares según el artículo de Paola Da Silva [9]. Estas prótesis pueden reemplazar funcionalmente a un órgano, pero su alto precio limita el acceso a ellas.

Para reducir significativamente los costos, diversas personas comenzaron a utilizar la impresión 3D. El precio del material (por ejemplo, PLA) es relativamente bajo y permite imprimir modelos complejos en poco tiempo. Esto representa una revolución en el mundo de las prótesis mioeléctricas. Por unos pocos cientos de dólares, se puede obtener un modelo adaptado al usuario que ofrezca un número adecuado de grados de libertad y funcionalidad para el usuario.

3.4 Modelo de prótesis

Como se menciona en la introducción, ya existen prótesis comercializadas o modelos 3D disponibles en plataformas como E-nable. A continuación, se presentan cuatro modelos, 2 comerciales, un en libro servicio hecho por un particular y un proyecto universidad, con sus respectivas ventajas y desventajas.

3.4.1 *Modelo Hero Arm*

La prótesis Hero Arm, desarrollada por la empresa Open Bionics, es una de las más completas disponibles en el mercado. Puede encontrarla en el sitio de Open Bionics [10]. Utiliza un escáner 3D para obtener la forma del antebrazo, y luego se imprime en 3D una prótesis personalizada. Funciona con sensores mioeléctricos y proporciona retroalimentación al usuario mediante luces o vibraciones. Permite el control independiente de los dedos y dispone de seis modos de control. También es capaz de sujetar un objeto de hasta 8 kg con los dedos.. Durante el proceso de diseño, el usuario puede personalizar su apariencia. La empresa ofrece modelos tanto para adultos como para niños. Para una prótesis de alta complejidad, su precio es relativamente accesible (estimado en 11 000 USD), aunque sigue siendo elevado para muchos usuarios.



Figura 7: Prótesis Hero Arm

3.4.2 *Modelo Bebionic*

Bebionic es otra prótesis comercializada. Al igual que la anterior, se personaliza mediante escaneo 3D para adaptarse a la anatomía del usuario. La mano puede acoplarse al antebrazo mediante tres sistemas distintos: roscado, empotrado, o empotrado con sistema de bloqueo.



Figura 8: Prótesis Bebionic

Esta característica es interesante porque permite al usuario intercambiar la mano por una herramienta u otro accesorio utilizando el mismo sistema de acoplamiento.



Figura 9: Prótesis Bebionic con sistema de acoplamiento

La prótesis se controla mediante sensores mioeléctricos colocados sobre un músculo funcional del antebrazo, lo que representa una limitación para ciertos niveles de amputación. Además, la electrónica se encuentra dentro de la estructura del antebrazo, lo que dificulta su adaptación a amputaciones transradiales de distinta longitud. Su precio varía entre 30 000 y 40 000 USD, incluyendo servicios de mantenimiento.

3.4.3 Elastic Hand

El modelo Elastic Hand es posiblemente el más simple, ya que permite reproducir dos movimientos (abrir y cerrar la mano) sin motorización. Está disponible de forma libre en la plataforma E-nable. Se controla mediante los movimientos de la muñeca. Los materiales utilizados son económicos y de fácil acceso. Se sujetta mediante velcro a nivel de la muñeca, por lo que solo puede ser utilizada por personas con el antebrazo intacto hasta dicha articulación.



Figura 10: Prótesis Elastic Hand

3.4.4 Proyecto Gre-nable

El equipo del proyecto Gre-nable, conformado por cuatro estudiantes de una escuela francesa, desarrolló una prótesis personalizada de bajo costo para una usuaria con una amputación transmetacarpiana. La descripción de su proyecto esta en su pagina web [11]. Evaluaron distintos métodos de control, incluyendo sensores mioeléctricos, de fuerza y de deformación. Finalmente, optaron por un joystick colocado en la base del pulgar —la única articulación funcional— debido a la baja precisión de los sensores causada por el ruido. La prótesis permite el movimiento de cierre de los dedos mediante dos servomotores, ya sea con todos los dedos o solo con el índice. Para la sujeción utilizaron un sistema BOA Fit, que consiste en una rueda que enrolla un cable ajustando la prótesis al antebrazo. Esta explicado en la pagina web de matos velo [12]. Uno de sus objetivos fue mantener un costo inferior a 200 EUR, sin comprometer la robustez.

3.4.5 Resumen

Para cada criterio, se asignó una puntuación del 1 al 4 en la tabla 2, siendo 4 la mejor calificación.

Modelo	Precio	Robustez	Estética	Comodidad	Mantenimiento	Autonomía	Control
Hero Arm	2	1	4	3	1	1	4
Bebionic	1	2	3	4	2	2	4
Elastic Hand	4	4	2	2	4	Ø	1
Mano asistida eléctricamente	3	3	1	1	3	3	2

Tabla 2: Evaluación comparativa de prótesis

A continuación, se presenta un resumen de criterios y especificaciones técnicas que diferencian a los modelos evaluados en la tabla 3:

Modelo	Impresión 3D	Open source	Personalizable	Peso ligero
Hero Arm	✓	✗	✓	✓
Bebionic	✗	✗	✗	✓
Elastic Hand	✓	✓	✓	✓
Mano asistida eléctricamente	✓	✓	✗	✓

Tabla 3: Criterios técnicos de las prótesis

A partir de estos criterios, y considerando los recursos disponibles, puedo inspirarme en las ventajas de cada modelo. Para el diseño del proyecto, es necesario basarse en modelos de código abierto. Dado el objetivo de controlar la prótesis mediante activación muscular, es fundamental explorar diversos sensores antes de elegir el más adecuado.

4 Presentación del proyecto

Con los componentes presentados, tenemos un diagrama en la figura 11 presentando los distintos flujos incluidos en el proyecto.

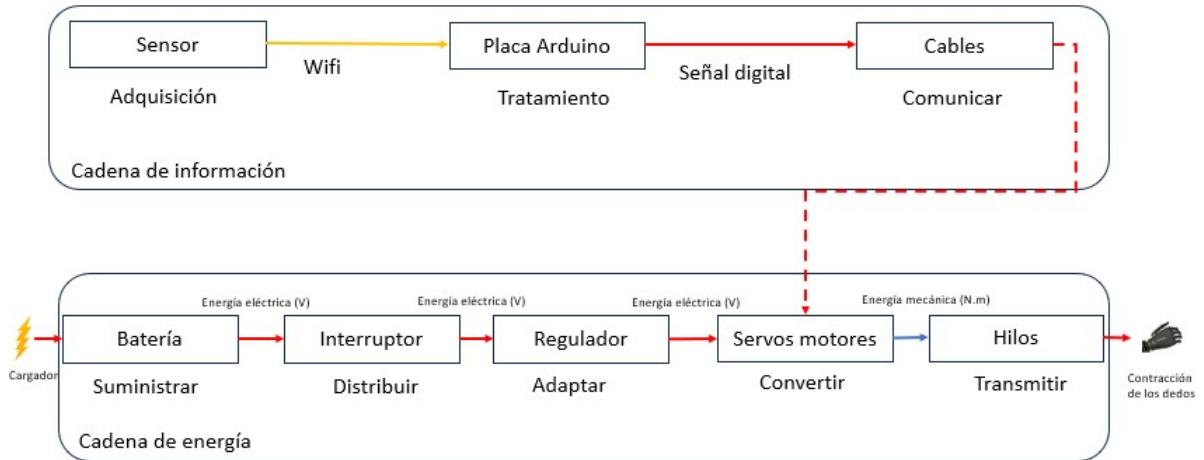


Figura 11: Diagrama de flujos

Se encuentra las distintas especificaciones de mis objetivos. Un control de la prótesis a través de un sensor que manda señales sin una conexión de hilos eléctricos. La prótesis debe poder funcionar con una batería y deberíamos controlar el suministro del sistema a través de un interruptor.

4.1 Modelo de base

El modelo hecho por Gre-nable sirvió como base del proyecto. Explicaron la disposición de las piezas y los componentes eléctricos usados en su proyecto de realizar una prótesis. El perfil de la usuaria final es una mujer que tiene una amputación transmetacarpian al nivel de la mano derecha.

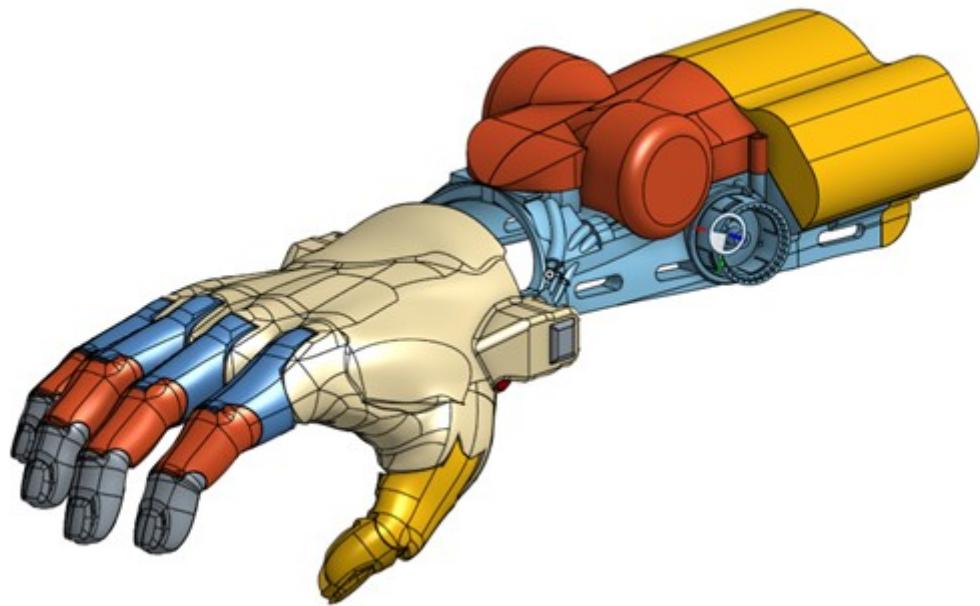


Figura 12: Modelo de base

Podemos ver en la figura ??, 2 principales partes. La mano y el guante.

4.1.1 *La mano*

La mano está compuesta por 5 dedos que están compuestos por 7 piezas cada uno (5 por el pulgar). Se observa las distintas partes que componen cada uno sobre las figuras 13 y 14. La parte dentro del dedo es llamada “flexibone”. Es la parte que permite la flexión del dedo porque es imprimida en un material con propiedad de elasticidad. Se inspiraron en un modelo ya hecho en el sitio Enable. El hilo pasa por huecos hechos al medio de la estructura de los dedos.



Figura 13: Estructura de un dedo



Figura 14: Estructura del pulgar

La parte donde se acoplan los dedos es la palma. Está vacía para que el muñón de la usuaria pueda ingresar en la palma y para disminuir el peso de la prótesis para dar más comodidad.

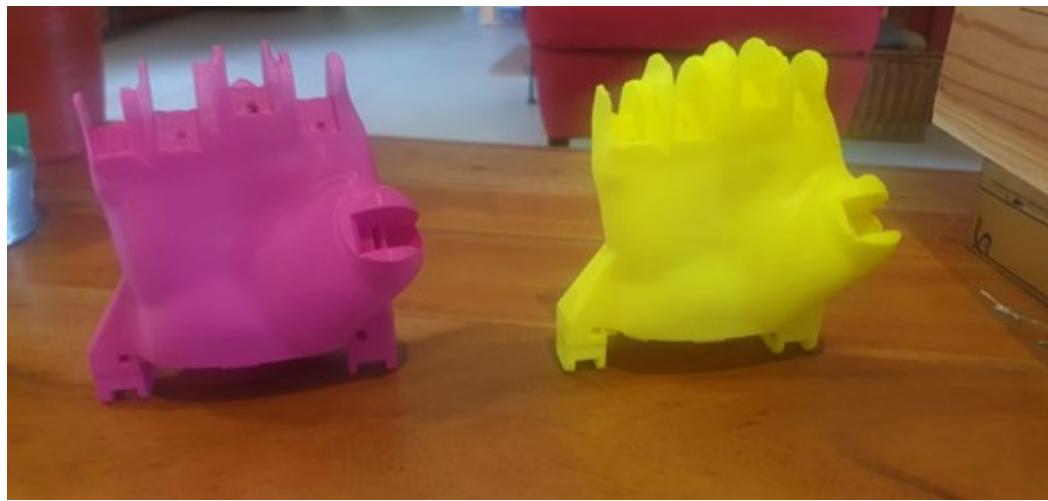


Figura 15: Impresiones de mano

4.1.2 El guante

La parte del guante es donde se coloca toda la parte electrónica de la prótesis. Esta parte se acopla con la mano al nivel de los huecos que se colocan al final de la palma. Diseñaron 2 piezas que permiten a los hilos pasar y de dejar un grado de rotación. En efecto, el usuario tenía su muñeca permitiendo los 2 grados de libertad que viene con ese.



Figura 16: Pieza derecha y izquierda ubicadas entre la palma y el gante

Los huecos son para hacer pasar los hilos de los dedos. El más gordo que vemos sobre la pieza derecha de la figura 16 sirve para los hilos que controlan el joystick, el modo de control de este proyecto.



Figura 17: Ubicación de las piezas entre el gante y la palma

Se observa que la palma no está fija y puede girar alrededor del eje formado por las piezas amarillas. A continuación, se encuentra el guante, diseñado con espacios específicos para alojar los componentes electrónicos, los cuales se describen en la figura 18.

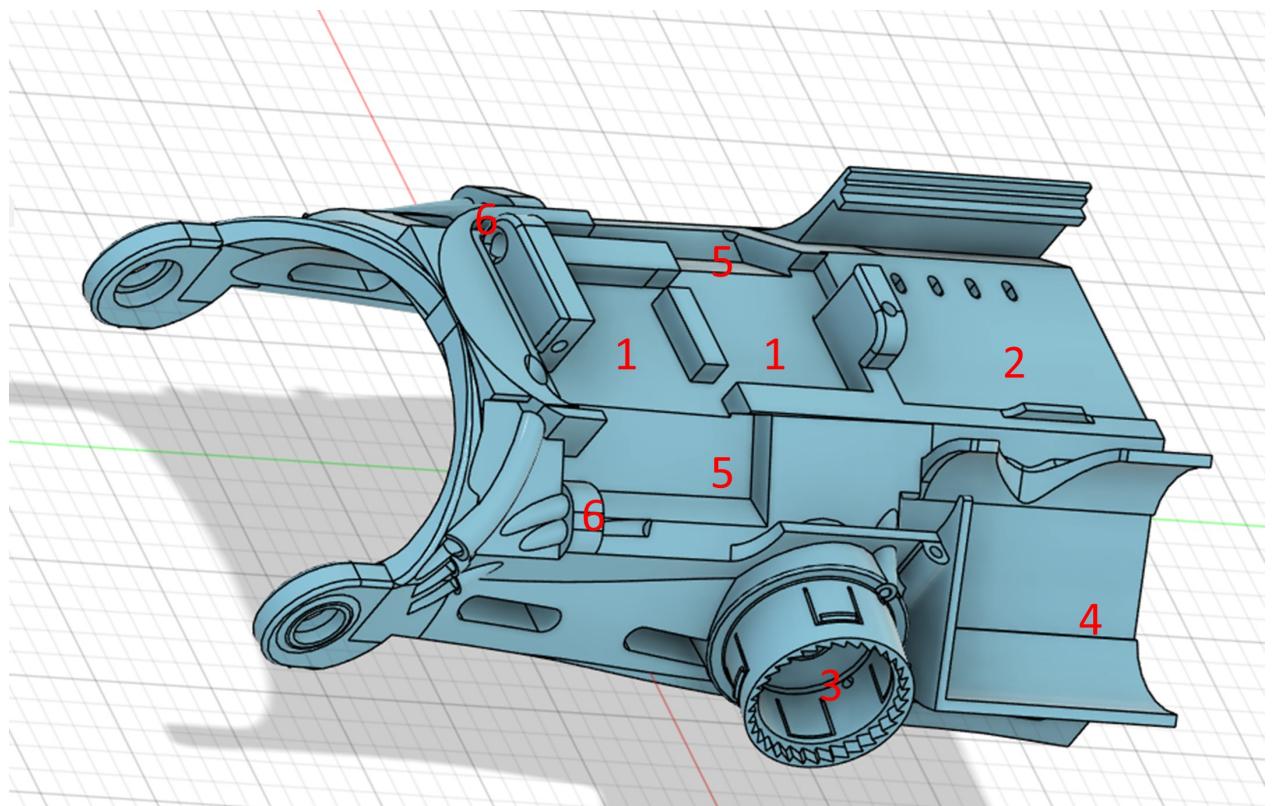


Figura 18: Descripción del guante

1: Ubicación de los dos servomotores 2: Ubicación del microcontrolador 3: Caja del sistema de ajuste 4: Caja de la batería 5: Ubicación de las poleas 6: Llegada de los hilos conectados a los dedos

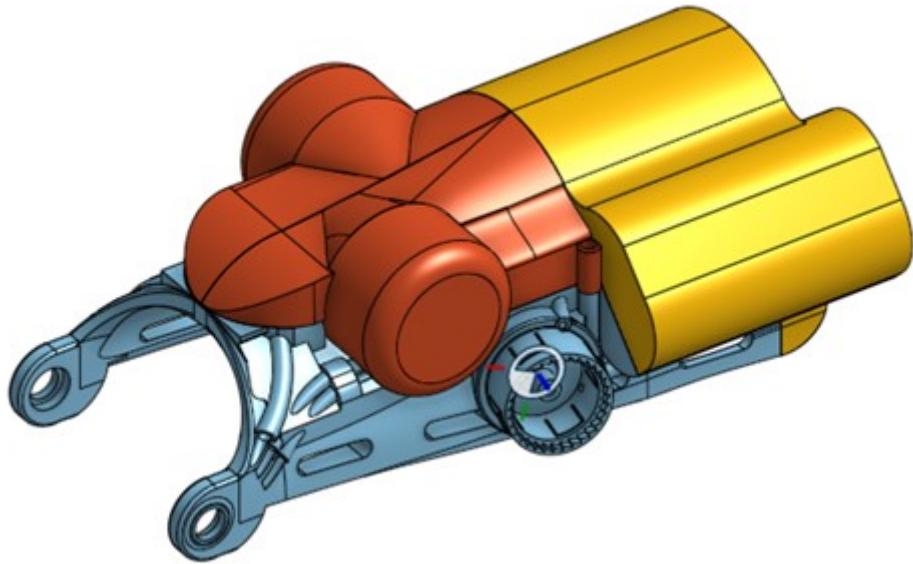


Figura 19: Cajas sobre el guante

Después, agregaron las cajas para proteger las piezas electrónicas. Son las partes rojas y amarillas sobre la figura 19. La parte amarilla tiene huecos atrás por hacer

pasar los cables que sirven para cargar la batería y conectar a la Arduino. La parte roja sirve a esconder y proteger los servomotores y las poleas que tensen los hilos.

5 Mecánica

En el mundo, personas desarrollan diseños en modelos 3D y los publican en plataformas de libre acceso como free3D [13] y Onshape [14] permitiendo que otros los utilicen o los tomen como base para realizar mejoras. Para este proyecto, se llevaron a cabo investigaciones con el objetivo de encontrar un modelo digital de una mano.

5.1 Adaptación a mis objetivos

Se obtuvo acceso a los archivos STEP del proyecto Gre-nable en la pagina onshape [15], los cuales fueron modificados para adaptarlos a los objetivos específicos del presente trabajo. Un archivo STEP (Standard for the Exchange of Product model data) es un formato neutral utilizado para intercambiar modelos 3D entre distintos programas de diseño asistido por computadora (CAD) [16]. El modo de control de la prótesis se basa en la activación muscular, por lo que la parte de control mediante joystick del diseño original no resultaba útil. Asimismo, fue necesario desarrollar una solución que permitiera que la prótesis se adaptara a cualquier tipo de amputación transradial, es decir, que pudiera montarse en cualquier parte del antebrazo, por debajo del codo.

5.1.1 Sistema de ajuste

El archivo STEP obtenido no brindaba acceso al sistema completo de ajuste, sino únicamente a la caja dentada ya incorporada al guante. El sistema de ajuste es relativamente simple: dentro de una caja con dientes se coloca una rueda como en la figura 20 que enrolla un hilo. Esta rueda es accionada por una manija que el usuario manipula manualmente. La manija solo puede girar en un sentido, gracias a un mecanismo de trinquete.

Dado que el archivo ya incluía la caja con dientes visible en la parte del guante, fue necesario diseñar las piezas restantes para completar adecuadamente el sistema de ajuste según los requerimientos del proyecto. A la rueda se le incorporaron dientes que deben engranar con los de la manija para permitir su rotación. La rueda gira alrededor de un eje central, alineado con el eje de la caja dentada del guante, y se diseñó un orificio específico para permitir el paso del hilo.

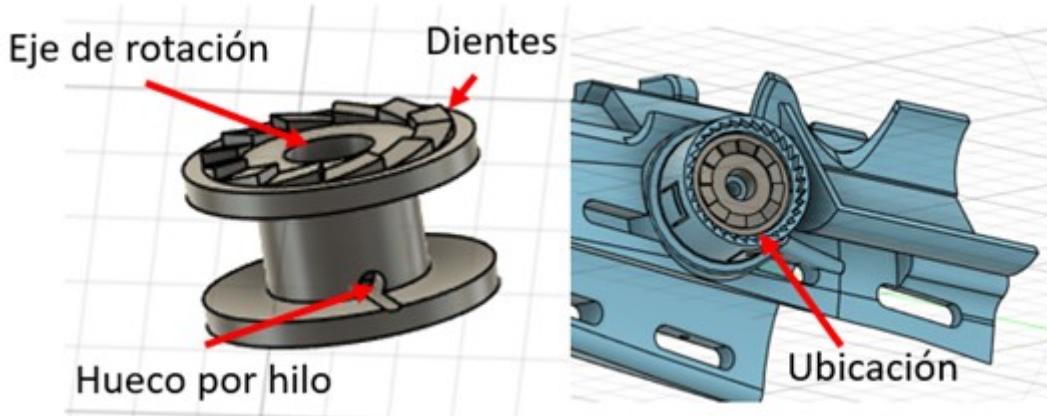


Figura 20: Ubicación de la rueda de ajuste

La manija es la parte que será accionada por el usuario. Tiene una sección con tres

dientes que se alinean con los de la caja. También presenta dientes que complementan a los de la rueda. Se encuentran dos ranuras espaciadas 3 mm que permiten a la manija tener dos posiciones. Su forma se ve en la figura 21

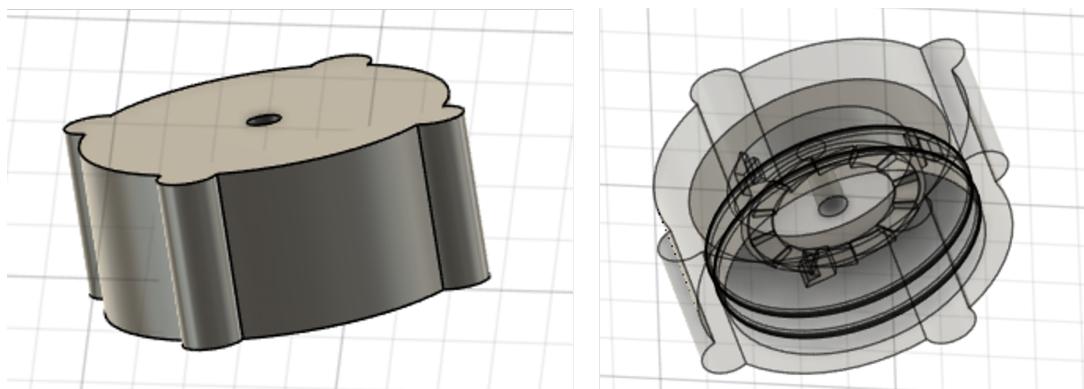


Figura 21: Manija visible al 100 % y al 70 %

La posición de ajuste, que puede ser observada en la figura 22, permite que todos los dientes se acoplen, haciendo girar la rueda en un solo sentido. En la posición de desajuste, ningún diente está en contacto, la manija queda libre y la prótesis se afloja con el movimiento del usuario.

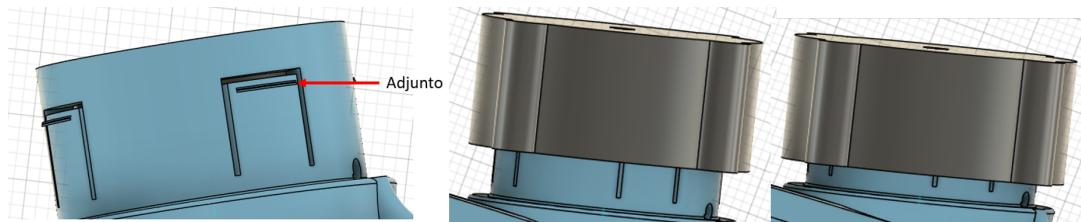


Figura 22: Acople de la manija y sus estados: abierto/cerrado

5.1.2 Caja trasera

En esta sección se analiza la caja que protege la placa Arduino y la batería de la prótesis. Inicialmente, la parte estaba como en la figura 23.

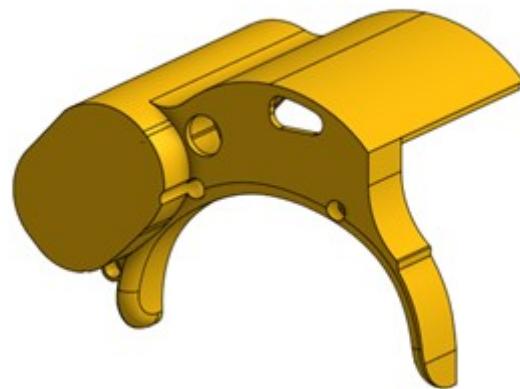


Figura 23: Caja trasera

Se pueden observar tres orificios hechos para acoplar esta pieza al guante, además de orificios de acceso a los componentes eléctricos de la prótesis. El objetivo inicial fue crear una pieza que permitiera controlar la longitud de mi prótesis. Por ello, una primera idea fue agregar una extensión, figura 24, que debía acoplarse con la parte ya existente mediante tornillos.



Figura 24: Primera extensión V1

Este primer prototipo presentó algunas dificultades. En primer lugar, no se contempló un espacio destinado a la incorporación de un detector. En segundo lugar, se introdujo una complejidad innecesaria al considerar el uso de tornillos muy finos y largos para acoplar esta parte al guante. Para resolver estos inconvenientes, se optó por no añadir una extensión adicional, sino por adaptar la pieza ya existente. Esta fue rediseñada con el objetivo de aumentar su longitud y de incorporar un espacio específico para alojar el receptor de señal del sensor utilizado. Las modificaciones se ven sobre la figura 25.

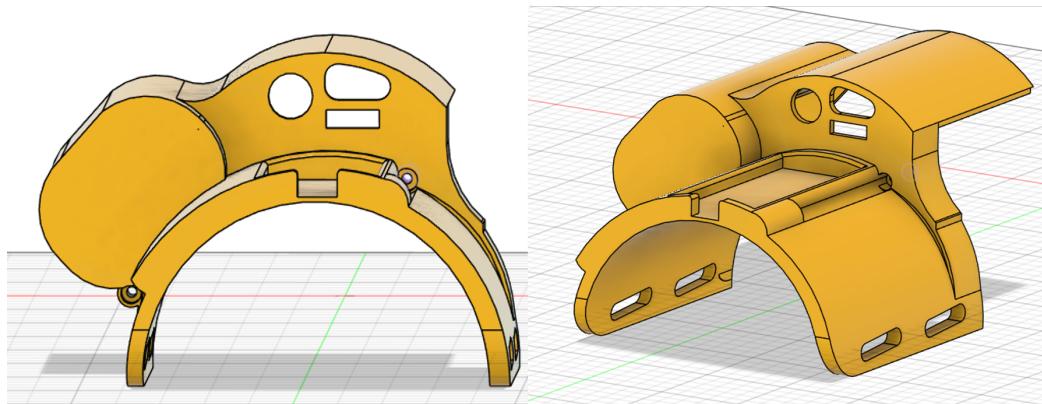


Figura 25: Extensión V2, vista trasera y en 3/4

También se modificaron los orificios destinados a los tornillos con el objetivo de simplificar el proceso de montaje. El tercer orificio fue diseñado específicamente para permitir el paso de los cables desde el receptor hasta la placa Arduino. Para finalizar el prototipo, se añadió un espacio destinado al interruptor, figura 26, el cual debía ser fácilmente accesible desde el exterior de la prótesis.

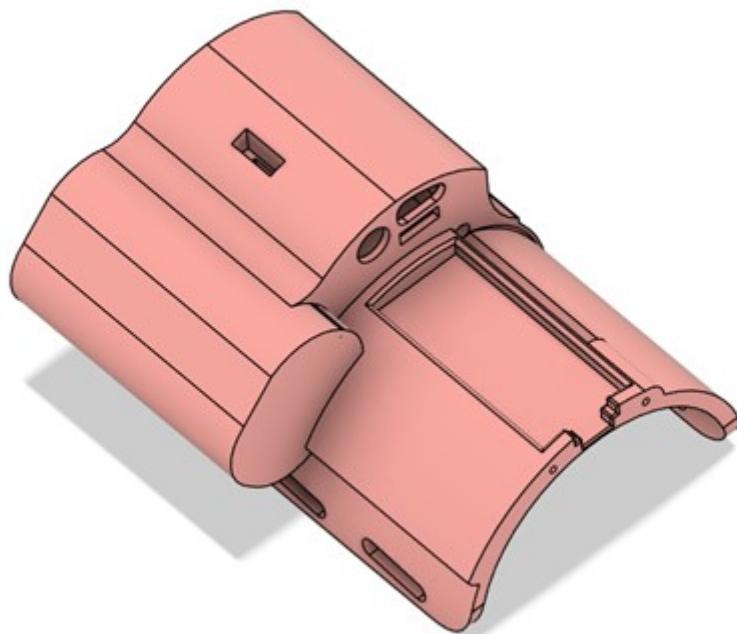


Figura 26: Ubicación del interruptor

Luego, fue necesario diseñar una caja de protección para el receptor. Inicialmente, se consideró crear una estructura simple que pudiera acoplarse a la pieza amarilla mediante dos tornillos.



Figura 27: Caja del receptor V1, vista en 3/4

En la figura 27 se muestra el primer prototipo. A partir de este modelo surgieron dos inconvenientes. El primero fue de carácter estético: los cables entre la placa Arduino y el sensor quedaban visibles, lo cual afecta la apariencia general del diseño, un aspecto importante para asegurar la comodidad y aceptación del usuario como lo confirmó la entrevista con una usuaria de prótesis cosméticas en la anexa A. El segundo fue funcional: para activar el receptor del sensor es necesario presionar un botón, y con esta disposición no era posible acceder a él de manera adecuada. Considerando estas observaciones, se diseñó una nueva caja, mostrada en la figura 28 con un botón incorporado.



Figura 28: Caja del receptor VF, impresa

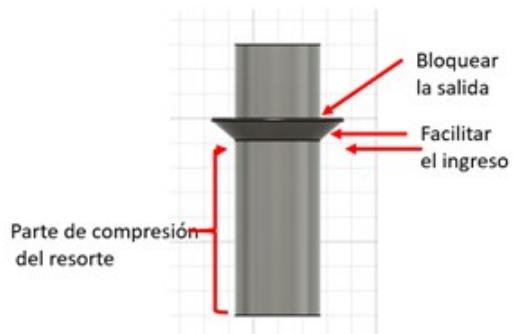


Figura 29: Descripción del botón

El botón, figura 29 está compuesto por un resorte que queda atrapado entre la parte móvil del botón y la carcasa. El botón puede ingresar a la caja, pero no puede salir.

5.1.3 Enlace entre el guante y la mano

Como se explicó en la presentación del proyecto base, existe un eje de rotación posible en la unión entre el guante y la mano. Sin embargo, si la persona presenta una amputación anterior a la muñeca, no podrá utilizar este movimiento. Por lo tanto, fue necesario desarrollar una solución que permitiera activar o desactivar este grado de libertad según las necesidades del usuario.



Figura 30: Modelización de las tuercas

La solución es permitir al usuario elegir si desea activar o no la rotación de la parte de la mano. Esta funcionalidad se logra mediante un sistema simple de tuerca, como se muestra en la figura 30. La tuerca se debe colocar en un espacio donde no moleste al usuario, figura 31.

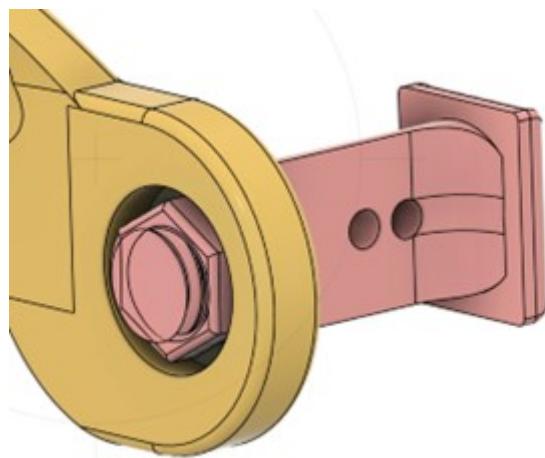


Figura 31: Ubicación de la tuerca

5.1.4 Parte inferior

Una de las condiciones establecidas para la prótesis es que pueda adaptarse a cualquier tipo de amputación transradial. Existe una dificultad asociada al desconocimiento del diámetro exacto del antebrazo y del punto preciso en que comienza la amputación. En consecuencia, una parte del guante podría llegar a ajustarse sobre un vacío. Para evitar esta situación, se incorporó una pieza inferior con el mismo diámetro que la parte superior. Se puede observar su diseño en la figura 32. De este modo, al realizar el ajuste máximo, la pieza inferior actúa como tope contra la superior, impidiendo un ajuste excesivo como se nota en la figura 33. Esta solución permite adaptar el sistema incluso en casos donde la longitud del antebrazo disponible es reducida.

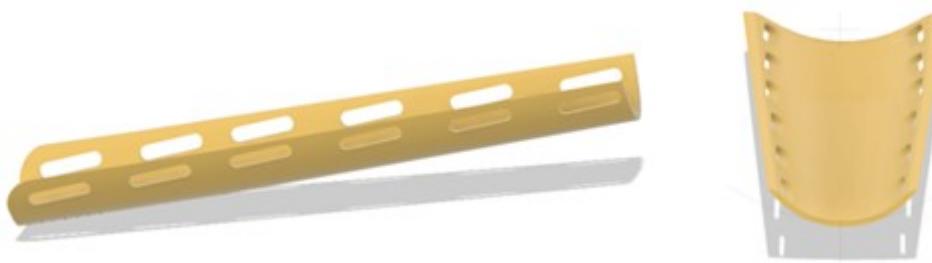


Figura 32: Modelización de la parte inferior

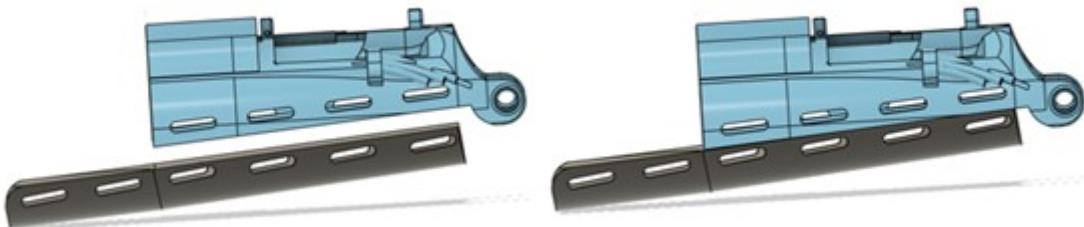


Figura 33: Movimiento de la parte inferior con el guante

5.2 Impresión

La impresión 3D permite crear piezas complejas con distintas funcionalidades, siempre que se controlen adecuadamente los parámetros de impresión.

5.2.1 *Materiales*

En este proyecto se utilizaron dos tipos de materiales: PLA y TPU 95A.

PLA:

El PLA es el material base de la impresión 3D. Se produce a partir de recursos renovables, lo que lo convierte en un material biodegradable. Es compatible con la mayoría de las impresoras. En la tabla 4 se presentan sus propiedades principales:

	Temperatura de extrusión (°C)	Masa volumétrica (t/m³)	Resistencia a la tracción (MPa)	Módulo de Young (GPa)
PLA	180–220	1.25	50–70	2.7

Tabla 4: Características del PLA

El PLA tiene una densidad media dentro del rango de los plásticos. Además, permite controlar la cantidad de material usado durante la impresión, y por ende, el peso del modelo. Es un material adecuado para impresión FDM. Sin embargo, presenta una débil

resistencia térmica: puede deformarse cuando la temperatura ambiente supera los 55°C. Aunque los componentes utilizados no alcanzan dicha temperatura, esta propiedad debe tenerse en cuenta durante el uso de la prótesis. Su módulo de Young elevado implica baja elasticidad a temperaturas normales, y la resistencia final depieude de la estrategia de impresión.

TPU 95A:

Desde 2017, el TPU 95A puede imprimirse como filamento, ofreciendo una elasticidad similar a la goma una vez impreso. Es el material que proporciona la flexibilidad necesaria para la flexión de los dedos. La tabla 5 muestra sus características.

	Elongación a la rotura (%)	Resistencia a la tracción (MPa)	Módulo de Young (MPa)
TPU 95A	55	39	40

Tabla 5: Características del TPU 95A

La elongación es la propiedad más relevante en este caso. El hilo conectado a los servomotores se tensa, creando una fuerza que activa los dedos a través de los “flexibone” presentes en su interior.

5.3 Impresoras y software

Se tuvo acceso a las impresoras X1C y X1E, de la empresa Bambu Lab (figura 35). Se trata de impresoras de alta tecnología, capaces de trabajar a gran velocidad y con alta precisión. Ambas fueron utilizadas para la fabricación de las distintas piezas del proyecto. La impresora X1E ofrece una calidad ligeramente superior, aunque la diferencia no es significativa. Esto se debe a parámetros adicionales, como la calefacción de la cámara, que contribuyen a una mejor forma final de las piezas al reducir las deformaciones durante la impresión.



Figura 34: Impresoras X1C y X1E

El software para preparar las impresiones también se llama Bambu Lab. A continuación se presenta su interfaz principal en la figura 35

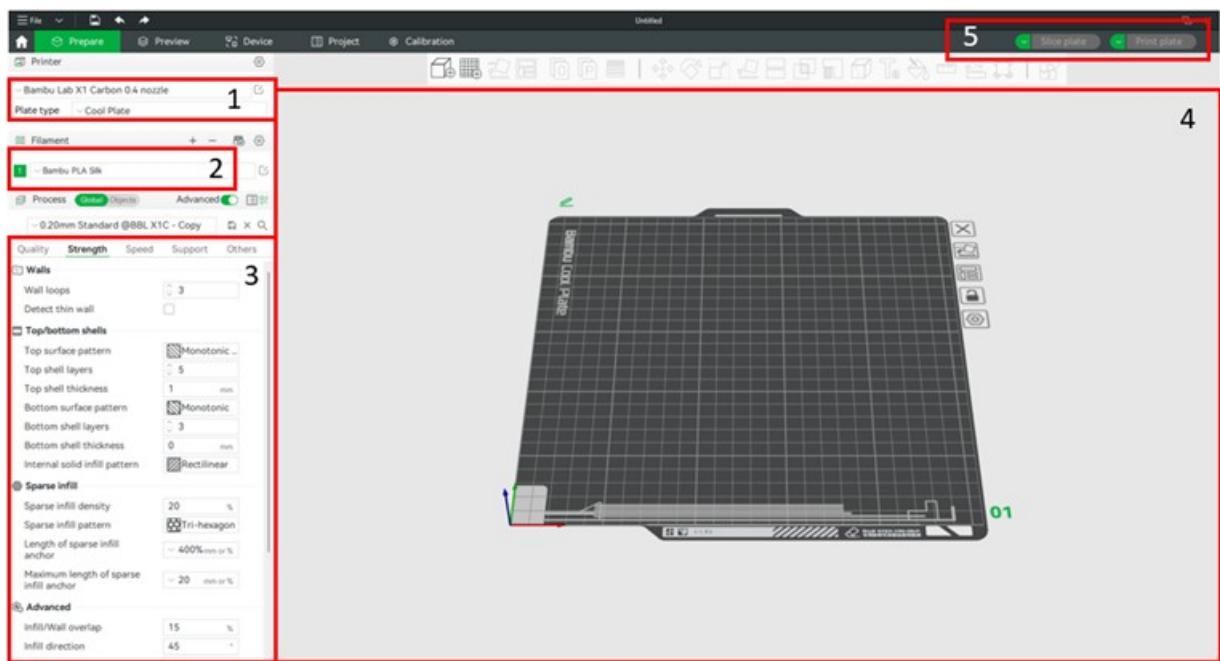


Figura 35: Interfaz del software Bambu Lab

- 1: Selección de la impresora y la placa de impresión
- 2: Selección del material
- 3: Parámetros de impresión
- 4: Placa de impresión donde se ubican las piezas
- 5: Botón de exportación de archivos

5.3.1 Parámetros importantes

Conocer los parámetros de impresión permite fabricar piezas más complejas con buena calidad. Además, permite controlar la resistencia y la composición del objeto impreso. Se presenta los parámetros de fuerza en la figura 36.

Fuerza:

El parámetro “wall loop” determina el número de capas exteriores que se imprimirán antes de llenar la pieza.

Los otros parámetros influyen en la primera y la última capa, pero no fueron determinantes en este diseño.

Los dos últimos parámetros son cruciales: permiten controlar la densidad y el patrón del relleno.

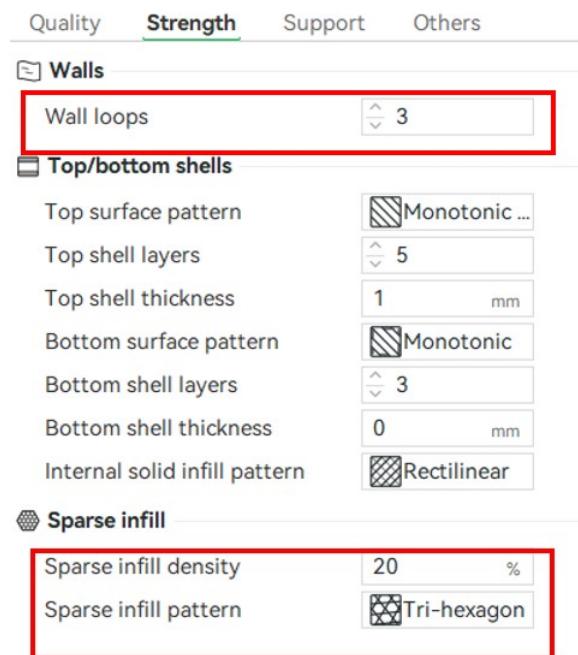


Figura 36: Parámetros de resistencia en Bambo Studio

En la figura 37, se muestra la preparación para imprimir la caja posterior con distintos porcentajes de relleno (20 %, 40 % y 80 %), usando un patrón hexagonal. También se observan tres paredes alrededor del relleno (una naranja y dos amarillas).

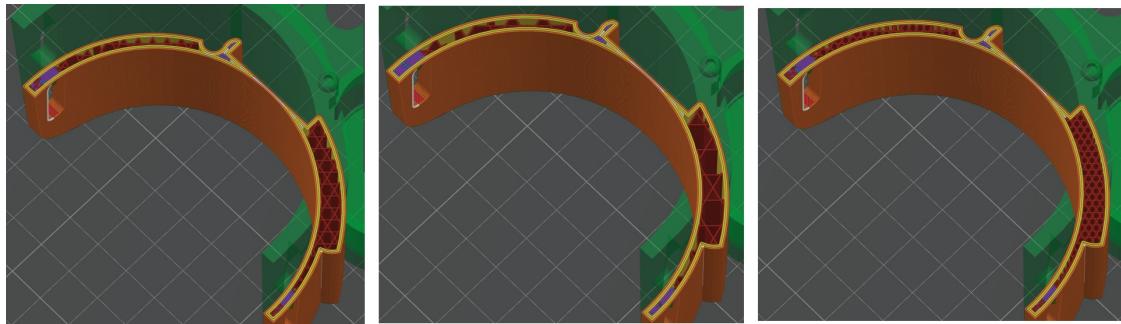


Figura 37: Relleno a 20 %, 40 % y 80 % con patrón hexagonal

Si no se necesita una resistencia especial, un 15–20 % de relleno es suficiente. Aumentar este porcentaje incrementa significativamente el tiempo de impresión. El patrón de relleno influye más en la resistencia mecánica que en el consumo de material o el tiempo. Durante pruebas, se hizo una comparación de los tiempos de impresión y la cantidad de material usado según el tipo de patrón y el porcentaje de relleno. Un resumen de las pruebas esta mostrada en la tabla 6.

Porcentaje de relleno	Patrón	Tiempo	Cantidad (g)
80 %	Tri-hexagonal	4h28	127
20 %	Tri-hexagonal	3h55	109
80 %	Grid	4h37	128
20 %	Grid	3h58	109
80 %	Rectilinear	4h28	129
20 %	Rectilinear	3h55	110
80 %	Honeycomb	5h38	130
20 %	Honeycomb	4h06	110

Tabla 6: Comparación de patrones y porcentajes de relleno

El tiempo de impresión fue un factor en la elección de un porcentaje de relleno bastante bajo. Se aumentó el relleno cuando

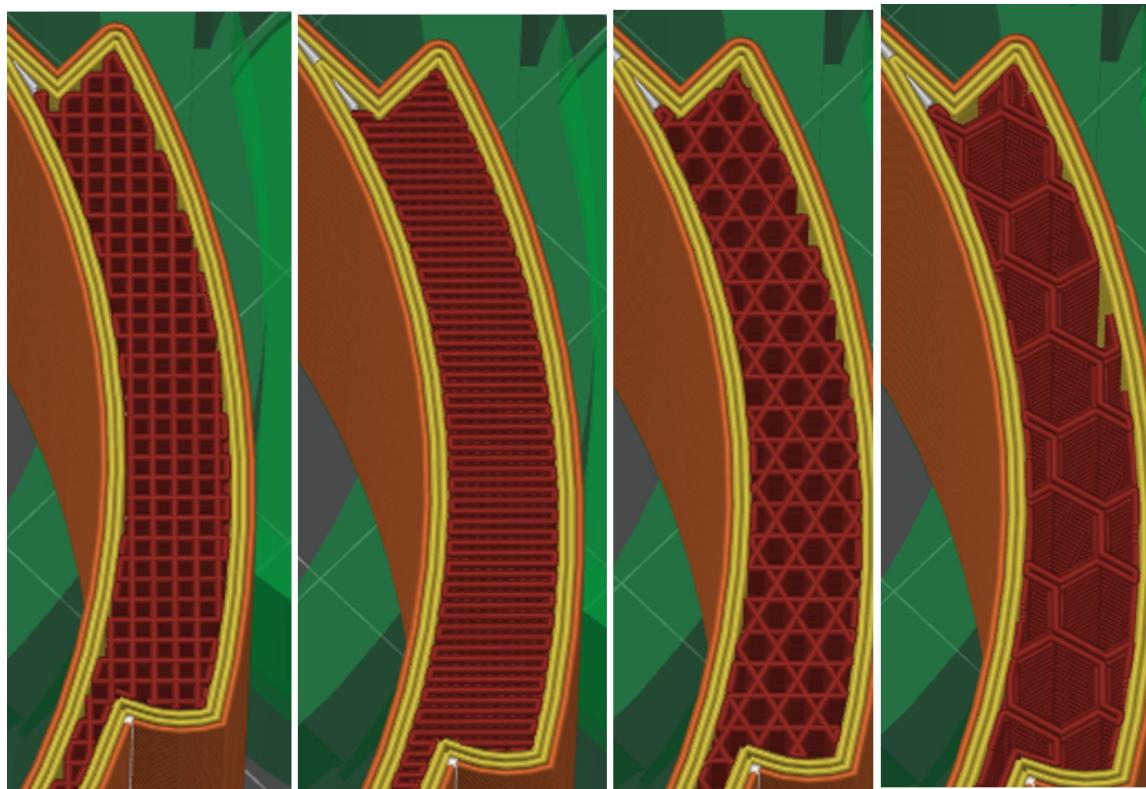


Figura 38: Patrones de relleno a 80 % y 20 % (honeycomb)

Después de varias pruebas, los resultados muestran que no hay diferencias significativas en tiempo o consumo a 20 %. Por tanto, la diferencia se basa en la resistencia. El patrón tri-hexagonal es el más resistente a esfuerzos multidireccionales, ideales para una prótesis sometida a choques y uso intensivo. Por eso, se eligió este patrón.

Soportes

Los soportes son elementos que sostienen las partes de una pieza que no se podrían imprimir directamente, ya que estarían en el vacío.

Existen dos tipos de soportes: normales y tipo árbol. En mis primeras impresiones utilicé los normales, pero consumían demasiado material, eran difíciles de retirar y reducían la calidad de las superficies de contacto. Además, no ofrecían tantas opciones de personalización como los soportes tipo árbol. Se muestra los distintos tipos de soportes en la figura 39

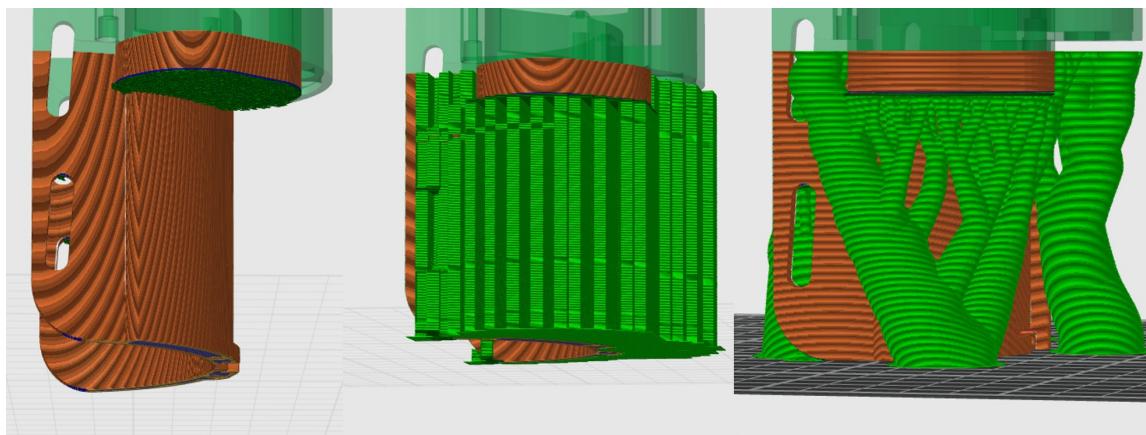
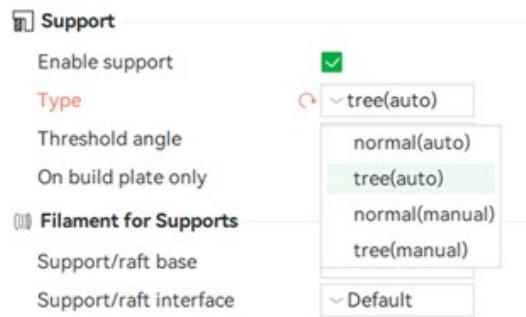


Figura 39: Visualización de soportes en Bambu

El principal problema identificado con los soportes surgió al imprimir piezas que contenían huecos. El software de impresión colocaba automáticamente soportes en el interior de estos espacios, lo que hacía imposible retirarlos posteriormente.

La primera solución implementada consistió en modificar la orientación de las piezas para que fueran más rectas. Sin embargo, en diseños complejos como el de la mano, no fue posible encontrar una posición que garantizara una orientación adecuada para todos los huecos.



Figura 40: Visualización de la mano con los huecos para los hilos y los parámetros de soporte

En la figura 40, se observan los huecos (en sombra) que deben generarse. Incluso configurando los parámetros adecuados, seguían apareciendo soportes internos. El “threshold angle” es el ángulo máximo que puede tener una superficie con respecto al suelo sin que necesite soporte. Las impresoras Bambu pueden llegar hasta un ángulo de 70°. Por seguridad, elegí 60°. También seleccioné la opción de realizar los soportes solo sobre la placa y únicamente en las zonas críticas. Sin embargo, aún quedaban soportes dentro de los huecos, como se muestra en verde en la figura 41.

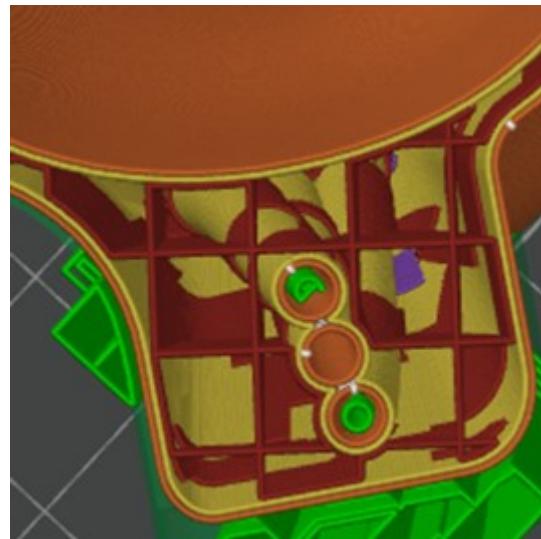


Figura 41: Soportes dentro de los huecos generados por los hilos

Fue necesario acceder a los parámetros avanzados del software para eliminar los soportes de manera definitiva. En particular, se identificó un parámetro denominado “max bridge length”, que por defecto se encuentra configurado en 0.

Este parámetro define la distancia máxima que la impresora intentará imprimir en el aire sin necesidad de soporte, antes de que el slicer agregue refuerzos adicionales. En casos donde no se requiera una alta precisión, este valor puede ajustarse entre 10 y 20 mm.



Figura 42: Parámetros avanzados de los soportes

Con este ajuste se pudo imprimir las manos sin que se generaran soportes dentro de los huecos.

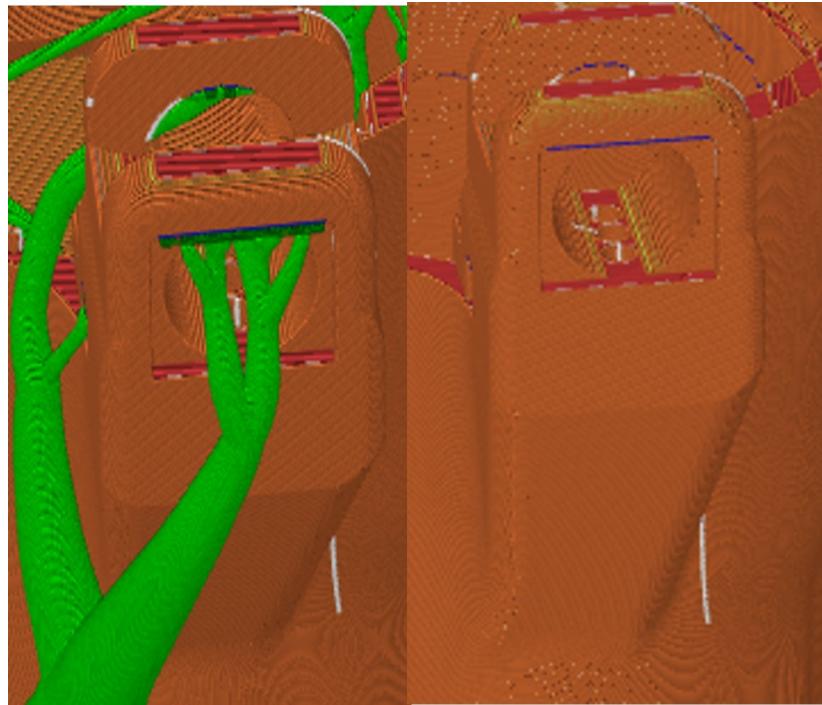


Figura 43: Diferencia entre la activación del parámetro “max bridge length”

En la figura 43, vemos la misma pieza, pero con diferentes configuraciones de soportes. Al final, la calidad de ambas es la misma.

Existen otras maneras de retirar los soportes, como elegir un material específico para los mismos y luego utilizar una solución química para eliminar solo este material

de la pieza. Sin embargo, esto implica costos adicionales y se utiliza especialmente cuando la calidad de la pieza es crítica a gran escala.

Calidad de impresión

Otra característica importante del software es la posibilidad de imprimir múltiples piezas en una sola placa, controlando los parámetros de cada una. Además, se puede mejorar la calidad de impresión de una parte específica de una pieza.

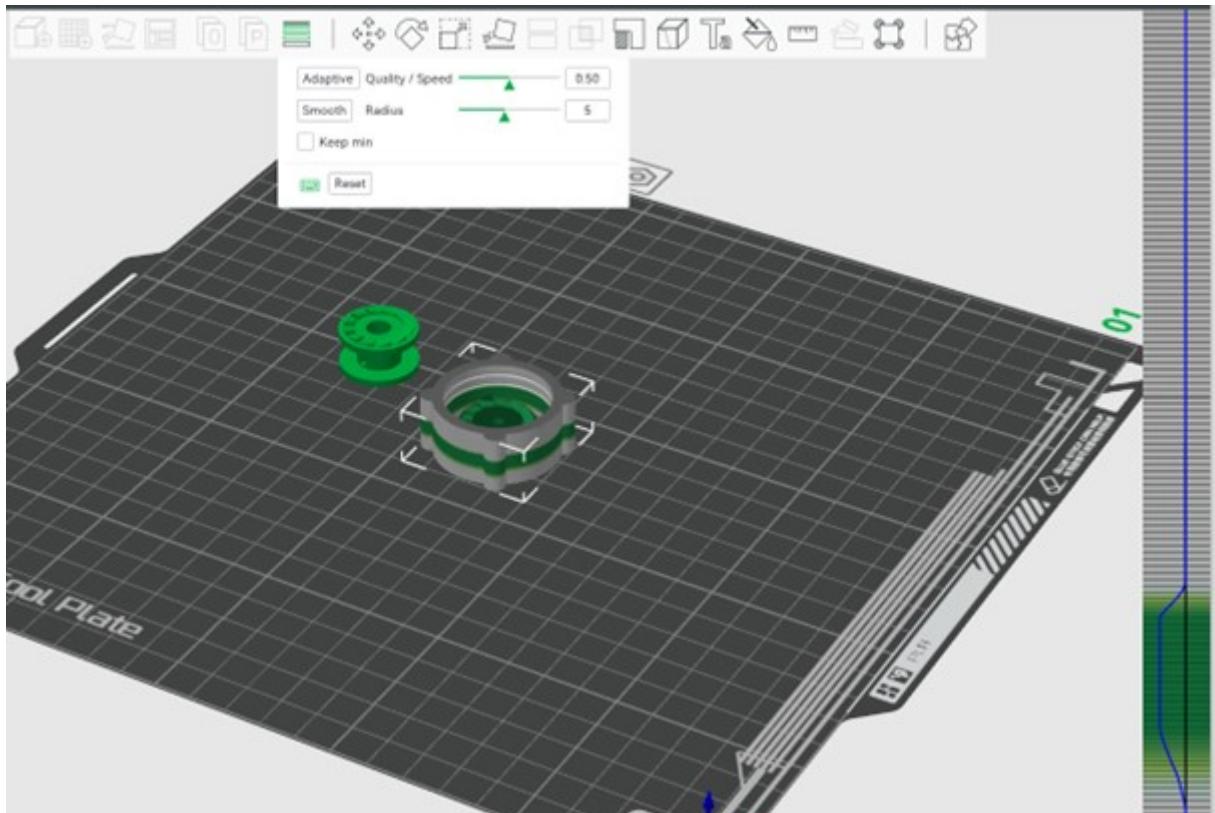


Figura 44: Visualización de la utilización del parámetro de calidad

En la figura 44, los dientes del sistema de ajuste deben ser precisos para garantizar la durabilidad del sistema. Al aumentar la calidad, se reduce el juego entre los dientes. El resto de la pieza mantiene una calidad estándar.

5.3.2 Impresión con PLA

La mayoría de las piezas están fabricadas en PLA. A continuación, se presenta la tabla 7 con los diferentes parámetros utilizados en las impresiones. Las piezas fueron impresas varias veces debido a diversas razones:

- Imposibilidad de retirar los soportes.
- Cambio de diseño.
- Resistencia insuficiente.
- Calidad de la superficie o funcionalidad deficiente.

Nombre	Patrón	Relleno (%)	Soportes	Calidad localizada	Nº de impresiones
Mano	Tri Hexagonal	20	Árboles + modificación de los parámetros iniciales	No	4x
Gante	Tri Hexagonal	20	Árboles + modificación de los parámetros iniciales	Sí, en el sistema de ajuste	3x
Dedos	Grid	13	No	No	1x
Parte de abajo	Tri Hexagonal	30	Árboles	No	1x
Poleas	Tri Hexagonal	15	Árboles	Sí, en el acoplamiento con los servomotores	2x
Tensiómetros	Tri Hexagonal	13	No	No	2x
Tuercas	Tri Hexagonal	25	No	Sí, en la rosca	2x
Caja de atrás	Tri Hexagonal	15	Árboles	No	4x
Caja del receptor	Tri Hexagonal	30	No	No	3x

Tabla 7: Parámetros de impresión de cada pieza

5.4 Impresión con TPU 95A

Las piezas denominadas “flexibone” cumplen un rol fundamental en el sistema de activación de la prótesis, ya que permiten el movimiento de flexión de los dedos y aportan la elasticidad necesaria para su extensión. Se especificó una elongación elástica del 55 %, lo que implica que estas piezas pueden estirarse hasta 1,5 veces su longitud original y retornar a su forma inicial una vez liberadas.

Sin embargo, esta propiedad solo se conserva cuando la pieza está fabricada íntegramente en TPU 95A, con un 100 % de relleno. En la práctica, imprimir con este nivel de densidad no es habitual, ya que incrementa considerablemente el tiempo de fabricación y la fuerza que deben ejercer los servomotores para activar los dedos.

Por este motivo, se produjeron tres versiones de flexibones con rellenos del 20 %, 50 % y 75 % para evaluar su comportamiento. Los resultados no fueron los esperados: la elasticidad resultó insuficiente para permitir la apertura completa de la mano. Esto se atribuye a la forma en que fueron impresas las piezas, ya que, como se observa en la figura 45, las articulaciones no fueron adecuadamente llenadas.

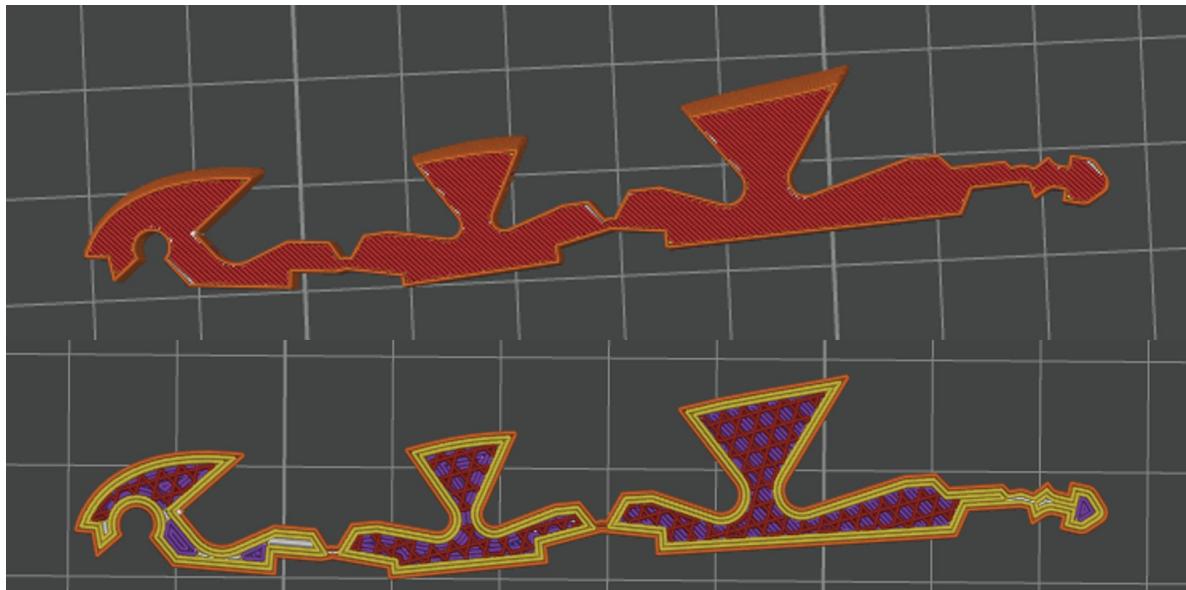


Figura 45: Relleno de las partes flexibone

Como se puede observar, en las articulaciones no hay filamentos adentro. Esto se debió a que durante el proceso de computación, el software no rellenó esas áreas. Esto provocó que el material no se comportara como se esperaba y, además, se perdió resistencia, lo que hizo que las piezas fueran fáciles de romper. Para solucionar esto, cambié la orientación de la impresión, lo que mejoró la resistencia y elasticidad en las articulaciones, aunque afectó la calidad superficial.

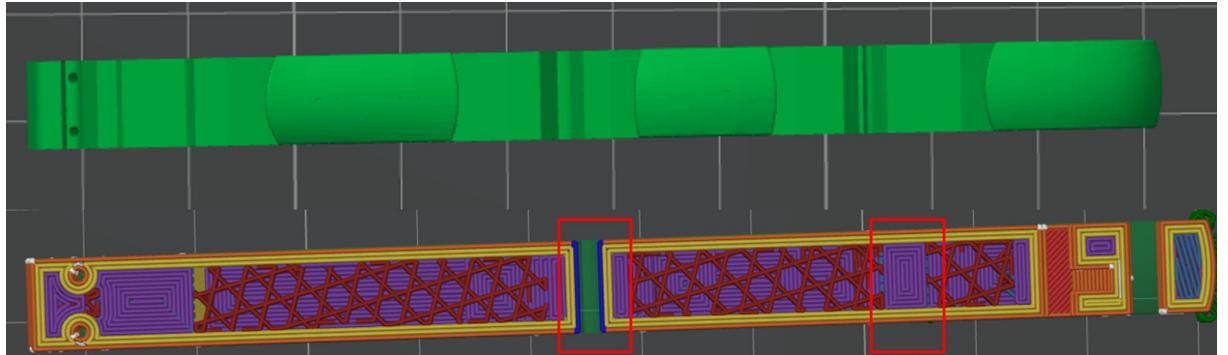


Figura 46: Nuevo posicionamiento para la impresión de las partes flexibone

De esta forma, vemos en la figura 46 se pasó de tener 2 capas a 8, aumentando el grosor de 0.4 mm a 1.6 mm. Además, el patrón de relleno pasó a ser rectangular, lo que permite tener fibras transversales al movimiento de flexión, incrementando la fuerza necesaria para doblar los dedos y disminuyendo el riesgo de deformación de las fibras, lo que permite que las piezas vuelvan a su forma inicial.

6 Electrónica

6.1 Descripción de los componentes

Módulo de control

Para controlar el sistema fue necesario utilizar una placa programable. Las placas Arduino presentan múltiples ventajas: son de fácil uso, cuentan con una aplicación gratuita que permite cargar el código directamente en la placa, y utilizan el lenguaje de programación C++. Aunque algunas placas Arduino pueden ser voluminosas, la empresa ha desarrollado versiones más compactas, como la Arduino Micro. Esta placa ofrece características similares a las de la Arduino Uno, incluyendo pines analógicos y digitales, y se alimenta mediante un conector USB micro. A continuación, se detallan sus principales características:

Generales:

- Voltaje de entrada: 7V – 12V
- Memoria: 32 KB



Suministro:

- USB micro: 5V
- Pin Vin: hasta 12V (para fuentes externas)
- Pin RAW: hasta 12V

Salidas:

- 5V, 500mA
- 3.3V, 50mA

Figura 47: Descripción de la placa micro Arduino

No hay conector hembra JACK en este tipo de placa en comparación a una placa arduino clásica. Este tipo de conector está usado para suministrar la corriente. Es decir, que para suministrar la placa es necesario crear un circuito donde conectamos una batería, el pin Vin y la Arduino micro. La Arduino micro puede leer las señales de entrada con un pin analógico (si es una señal analógica). El ADC (convertidor analógico-digital) puede codificar el valor sobre 10 bits.

$$2^{10} = 1024 \quad (1)$$

La entrada de la placa en un pin analógico va a leer un valor entre 0 (0V) y 1023 (5V). Si es una señal de 3.3V, obtendremos valores entre 0 (0V) y 675 (aproximadamente 3.3V).

Servomotores

La elección de los servomotores se hizo porque son componentes que permiten alcanzar un ángulo deseado en poco tiempo y con un torque proporcional al precio/tamaño. El torque debe ser lo suficientemente alto para resistir la fuerza de retroceso del TPU 95A cuando se contrae. Esto explica la elección de los servomotores Servomoteur Hitec

HS85BB mostrado en la figura 66



Figura 48: Servomotor

Sus características son:

- Velocidad: $60^\circ/\text{s}$
- Torque: $3,5 \text{ kg/cm}$
- Precio: 40 \$ (estadounidenses)
- Voltaje de entrada: 6 V

Para controlarlos, tenemos 3 cables: 2 para el suministro y tierra, y un último para el control. Se puede enviar una señal PWM donde el ancho del pulso representa el ángulo que debe alcanzar el servomotor.

Sensor

La parte superior de la placa contiene el circuito y los conectores macho. La parte inferior de la placa tiene los conectores hembra a presión. Se observa en la figura 49.



Figura 49: Sensor MyoWare

En la parte inferior, conectamos electrodos para medir la diferencia de potencial que ocurre durante la actividad de los músculos. Se cuenta con un electrodo de referencia (REF) que debe conectarse a una zona externa al músculo en actividad. Otro es MID, que debe ser conectado al músculo en su medio. Y finalmente, tenemos el electrodo END, que debe ser conectado al extremo del músculo [17]. La ubicación óptima está mostrada en la figura 50.

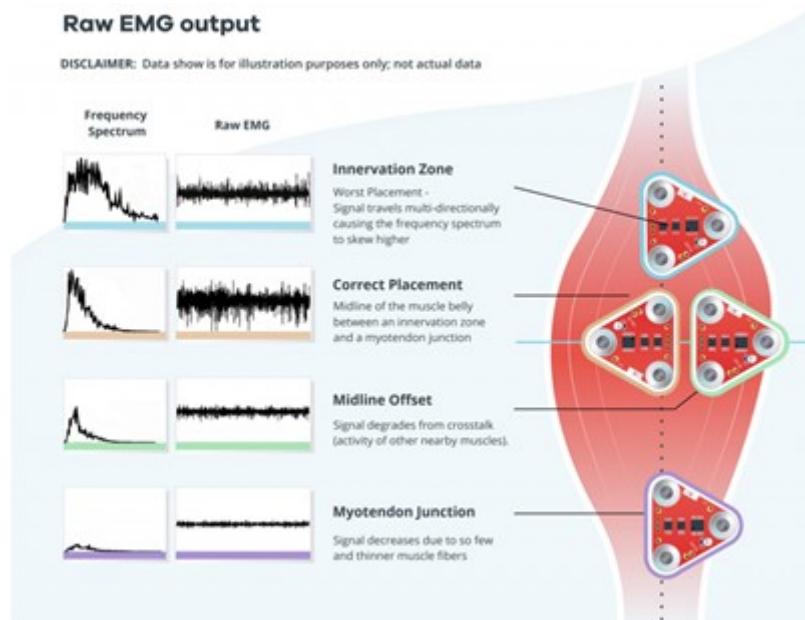


Figura 50: Ubicación óptima del sensor

La alimentación proviene de la parte superior. Los 3 conectores macho sirven para alimentar el sensor. En nuestro caso, están conectados al "Wireless shield" de la figura 51.

Wireless shield

El wireless shield es la parte que alimenta el sensor y que recoge los valores para enviarlos.

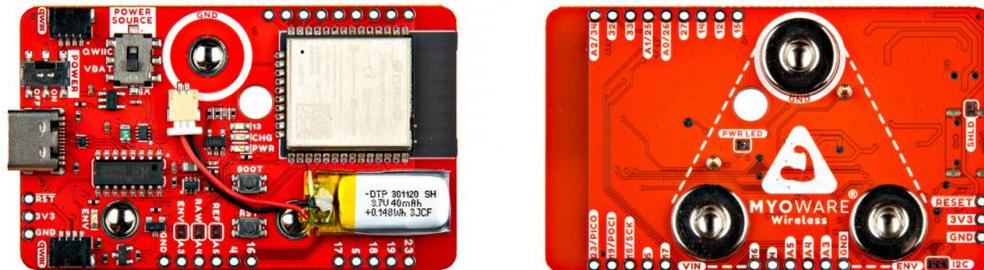


Figura 51: Shield MyoWare

Se conecta al sensor a través de los 3 conectores hembra. El sistema de alimentación puede variar. El que se usa es la batería LiPo, que alimenta todo el sistema y se recarga con el conector USB-C. Otra opción es la alimentación QWIIC. Podemos agregar una batería más eficiente y duradera con este tipo de suministro. Para cambiar la fuente de alimentación del sistema, solo necesitamos activar el interruptor en la parte superior de la placa.

El módulo encargado de enviar los valores es la placa ESP32 WROOM ubicada a la derecha de la placa. La información es recibida por la misma placa ubicada en el receptor. Sale 26.95\$ (estadounidenses) en el sitio sparkfun [<empty citation>]

Receptor

La placa receptora que se observa en la figura 52 tiene el único objetivo de hacer la conexión entre la placa Arduino y el Wireless shield. Se puede configurar sus pines en modo DAC. El DAC (Convertidor Digital a Análogo) permite crear una salida analoga con valores digitales. Las informaciones que obtengo al activar los bíceps varían entre 0 y 3870. Para enviarlas de manera que sean visibles para la placa Arduino, se debe convertir a un rango de valores mucho más pequeño [0,255], que es el rango de valores que el receptor puede enviar con un suministro de 3.3V. Sale 24.95\$ en el sitio sparkfun [18].



Figura 52: ESP32 (receptor)

Batería

Para alimentar todo el sistema, se utilizo una batería de 7.4V con una capacidad de 5800 mAh. Esto significa que si el sistema consume 1000 mAh, la batería puede durar 5h48min. Es suficiente para alimentar los servomotores utilizando un regulador para reducir la tensión. Con la misma batería, también se puede alimentar la placa micro Arduino con su pin Vin. Esta misma placa puede, a su vez, alimentar el receptor con su salida de 3.3V. Para poder cargar la batería, se agregó una boquilla conectada al interruptor, formando otro circuito. Es en este otro circuito donde se conecta el cargador de la batería de la figura 53.



Figura 53: Batería

Regulador

El regulador es el componente que nos permite disminuir la tensión que llega a los servomotores. La batería suministra con 7.4V mientras se requiere 6V a la entrada de los servomotores. El rango de tensión de entrada del regulador debe admitir 7.4V y el de la salida debe admitir 6V. Además debe hacer pasar un corriente suficiente para hacer funcionar los servomotores. El modelo Pololu S9V11MA, que se ve en la figura 54 responde a estas distintas características:

- Tensión de entrada: 2V - 16V
- Tensión de salida: 2.5V – 9V
- Corriente de salida: 1.5 A



Figura 54: Regulador

Interruptor y led

Para garantizar la seguridad y comodidad del usuario, se incorporó un interruptor

que permite commutar entre circuitos al ser activado. En una posición, la batería está conectada al circuito de la prótesis, y en la otra está conectada de manera que pueda cargarse gracias a la boquilla agregada. Una led se enciende para indicar cuando el sistema esta encendido. Ambos elementos estan mostrados en la figura 55.



Figura 55: Interruptor y Led

Problema de interferencia

Durante las pruebas del sensor, se observó que la micro Arduino no estaba leyendo correctamente los valores. Cuando los músculos no estaban activados , la placa estaba leyendo valores diferentes de 0. El pin tenía un comportamiento de antena, captando interferencias electromagnéticas. La solución fue agregar una resistencia de $1\text{ M}\Omega$ entre el pin y la tierra para que cuando no enviamos información, la entrada siempre esté a 0.

6.2 Circuito completo

A continuación, se muestra en la figura 56 el circuito completo con todos los componentes presentados anteriormente.

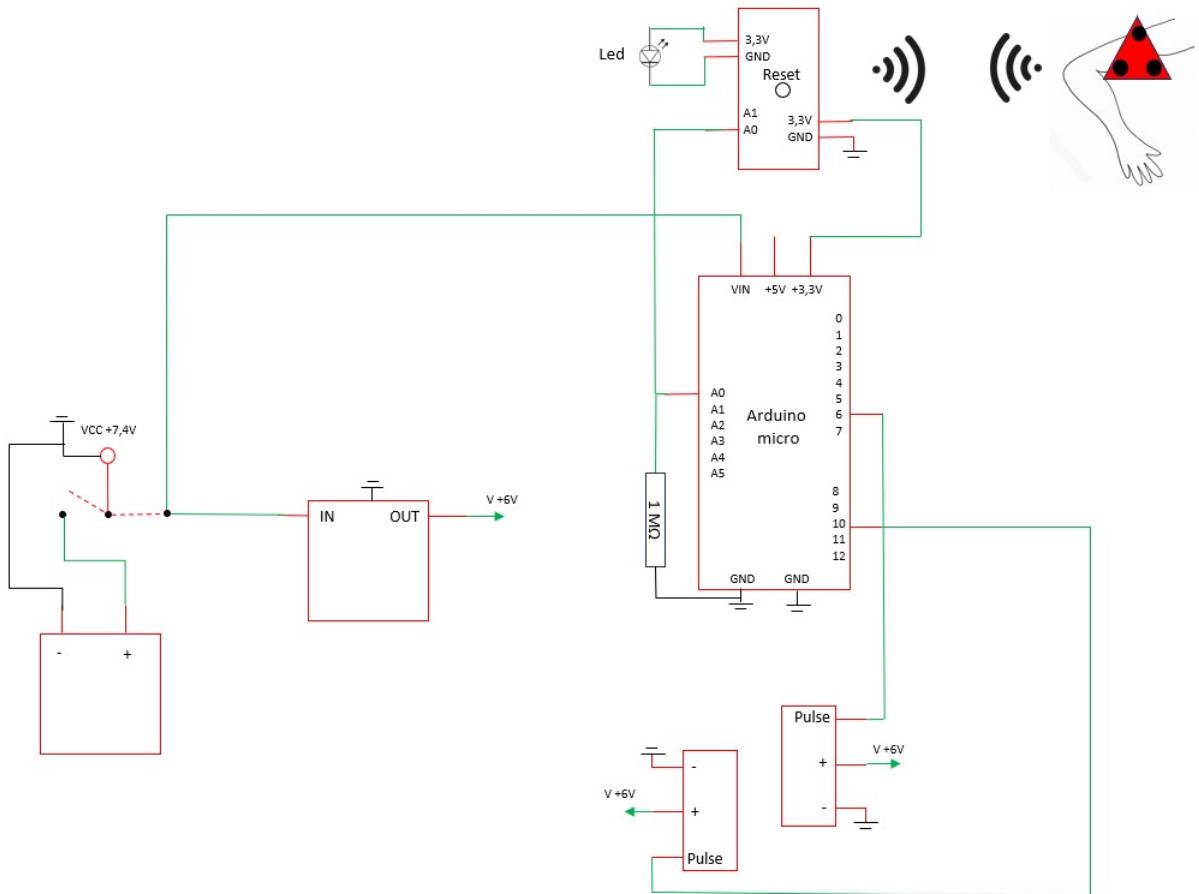


Figura 56: Esquema de conexión

Este circuito está soldado sobre una protoboard según el siguiente esquema que se observa en la figura 57 y la figura 58:

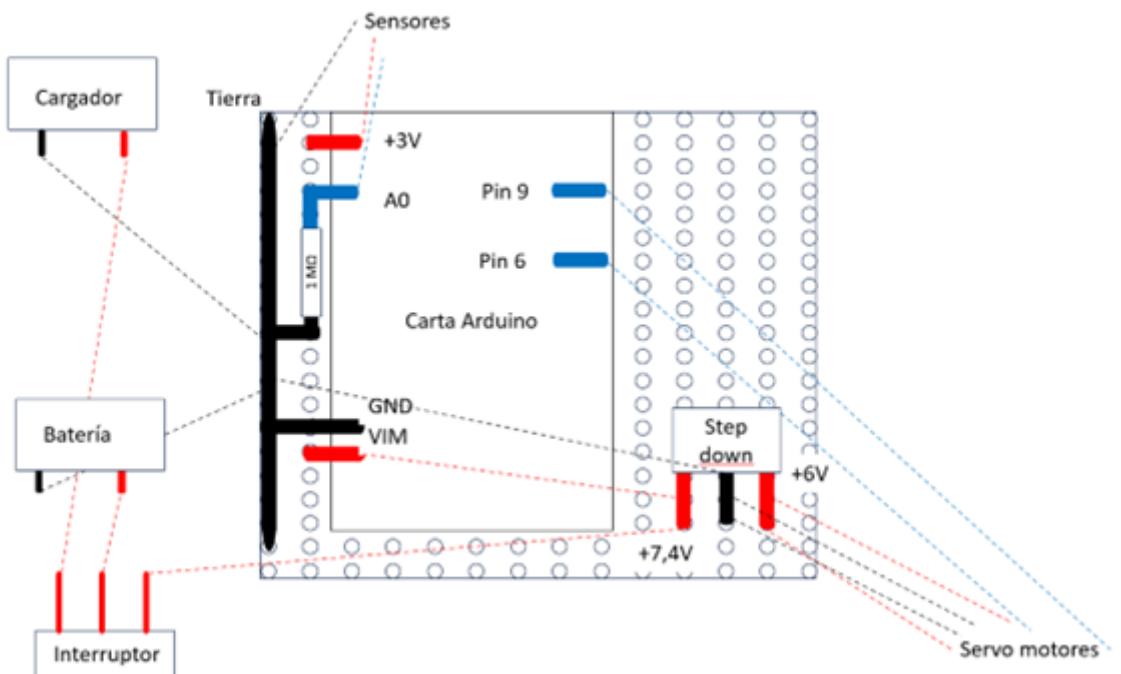


Figura 57: Esquema de soldadura

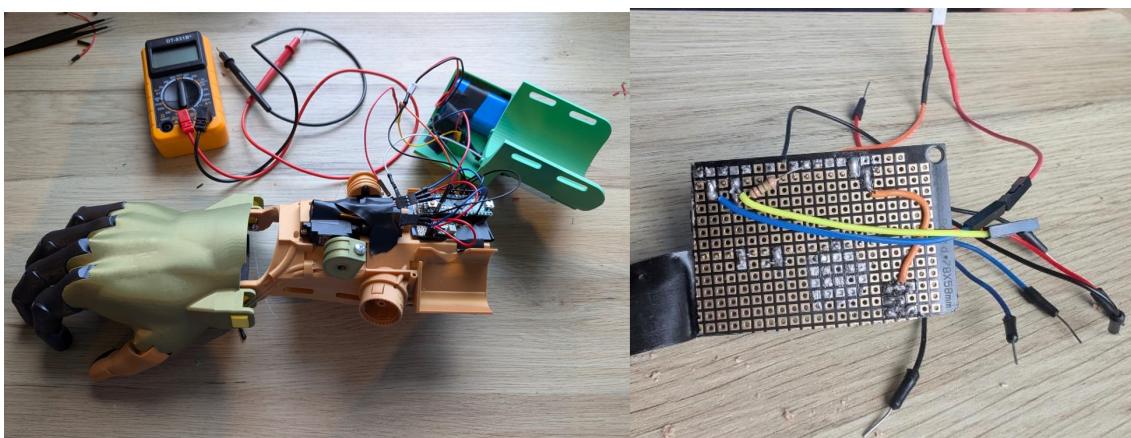


Figura 58: Componentes soldados

7 Mantenimiento

El mantenimiento es un aspecto importante de una prótesis. Es un problema recurrente que señalan los usuarios.

Mantenimiento mecánico

Las piezas mecánicas que pueden romperse pueden ser fácilmente reemplazadas, siempre que se cuente con los archivos CAD de las piezas. Los archivos del proyecto serán abiertos al público. Lo más inconveniente es tener acceso a impresoras 3D. Cada montaje puede ser desmontado, los tornillos son accesibles, no es necesario romper partes para alcanzar otros componentes.

Mantenimiento electrónico

En cuanto al mantenimiento electrónico, el objetivo era permitir un reemplazo sencillo de los componentes. Para eso, se incorporaron componentes que permiten desconectar los elementos entre sí. Primero, se utilizaron cables con conectores hembra que salen de la placa electrónica que contiene el micro Arduino. Para el conector que permite cargar la batería, se usó una regleta de conexiones. Sin embargo, durante el montaje de la prótesis, la regleta ocupaba demasiado espacio, por lo que se soldaron otros cables hembra al nivel del conector. De este modo, con estos tipos de conexiones, solo es necesario contar con repuestos para cambiar un componente defectuoso.

8 Programación

El código se divide en dos partes: una de control y otra de activación. La parte de control se encarga de recuperar y analizar la señal, mientras que la parte de activación acciona los motores.

8.1 Parte de control

En esta etapa intervienen tres componentes: el sensor mioeléctrico, el receptor y la placa Arduino.

8.1.1 Recuperación de la señal

El sensor debe configurarse para enviar la señal al receptor. Para ello, puede descargarse un código de ejemplo para Arduino [19]. Además, es necesario instalar las bibliotecas disponibles en el sitio SparkFun [20], específicamente diseñadas para este propósito.

Posteriormente, la señal del receptor debe ser captada por la placa Arduino. Utilicé el código proporcionado con el receptor, que permite visualizar los valores en la computadora. Modifiqué ese código para configurar un pin DAC en A0, como se muestra en la figura 59.

```
212     const double sensorValueDac = map(sensorValue, 0,3900,0,255);
213     dacWrite(readValues, sensorValueDac);
```

Figura 59: Líneas añadidas al código del receptor

8.1.2 Procesamiento de la señal

El sistema incluye filtros que eliminan el ruido generado durante la detección muscular. Los servomotores se controlan en función de la intensidad de la señal. Para evitar activaciones no deseadas, se definió un valor umbral que corresponde a la diferencia mínima entre dos señales consecutivas. Los valores de la señal, que oscilan entre 0 y 675, se convierten a un rango de 0 a 180 para ser compatibles con los servomotores. Estos valores se escalan cada vez que se transmiten.

```
97     int tratamientoSenal(int sV) {
98         int mapValor = map(sV, 0, 675, minAngulo, maxAngulo);
99         if (abs(ultimaMapValor - mapValor) > anguloTecho){
100             ultimaMapValor = mapValor;
101             return mapValor;
102         }
103     }
```

Figura 60: Código de procesamiento de la señal

En la figura 60 se observa la función de tratamiento de señal, que transforma los valores del sensor en ángulos útiles para los motores. Se compara la diferencia entre lecturas consecutivas y, en función de ella, se determina si se debe actualizar el valor.

8.1.3 Detección de patrones

Con un solo sensor, el objetivo es distinguir dos tipos de movimientos: uno que involucre todos los dedos y otro con solo el índice y el pulgar. Para lograrlo, se diseñó una función que identifica un patrón de contracción con forma de señal PWM, como se muestra en la figura 61. Para evaluar el comportamiento del detector, se realizaron pruebas prácticas. Se activó el sistema y se observaron las respuestas generadas ante distintos gestos musculares, variando la intensidad y duración de las contracciones. Se prestó especial atención a la estabilidad de la señal, la sensibilidad del sistema y su capacidad para diferenciar entre señales voluntarias y ruidos o movimientos no deseados. Estas pruebas permitieron identificar tanto la fiabilidad general del detector como sus limitaciones en situaciones de uso real.

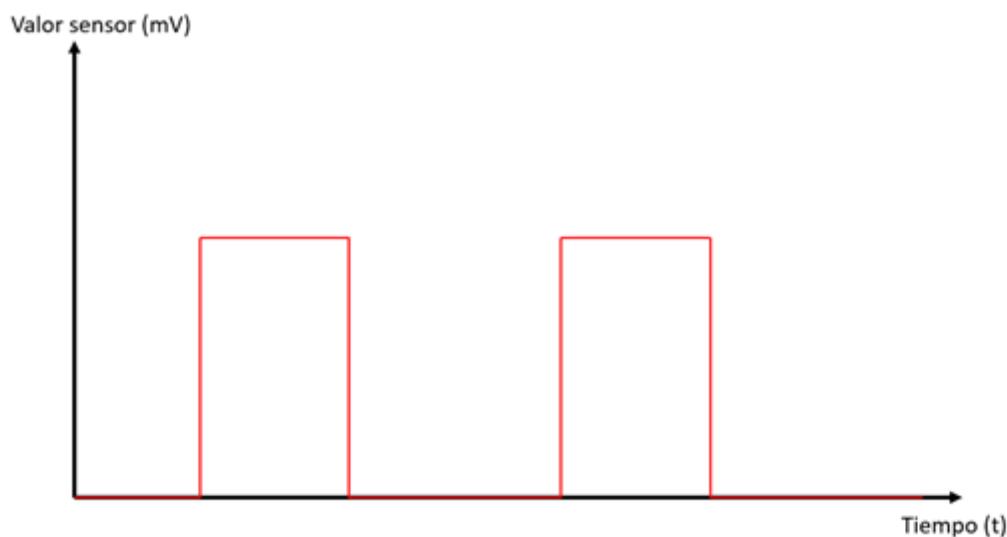


Figura 61: Patrón deseado a detectar

Este patrón se detecta mediante flancos ascendentes que superan un valor de 600, correspondiente a una contracción máxima. La contracción debe mantenerse durante un tiempo mínimo, y relajarse luego para completar el patrón.

Para mejorar la fiabilidad, se clasificaron los valores en tres rangos: 0–100 (relajación), 100–600 (transición) y 600–675 (contracción intensa). La detección de flancos se produce al pasar de una zona a otra.

Las pruebas anteriores indicaron que mantener una contracción fuerte durante 0 a 3 segundos es cómodo. Para evitar activaciones accidentales, se requiere una segunda contracción después de 2 segundos. El patrón real esperado se muestra en la figura 62. Durante la entrevista del anexo A, se habló sobre este tipo de función para cambiar de modo. Se mencionó que no sería una función de uso muy frecuente, ya que el modo 1 se utilizaría con mayor frecuencia en comparación con el modo de agarre con dos dedos. El entrevistado lo explicó en relación con necesidades básicas, como sujetar el manillar de una bicicleta, sostener un libro o mantener un objeto mientras la otra mano realiza una acción más precisa. De este modo, confirmó que, aunque el cambio de modo no sea completamente fluido, puede utilizarse sin mayores inconvenientes.

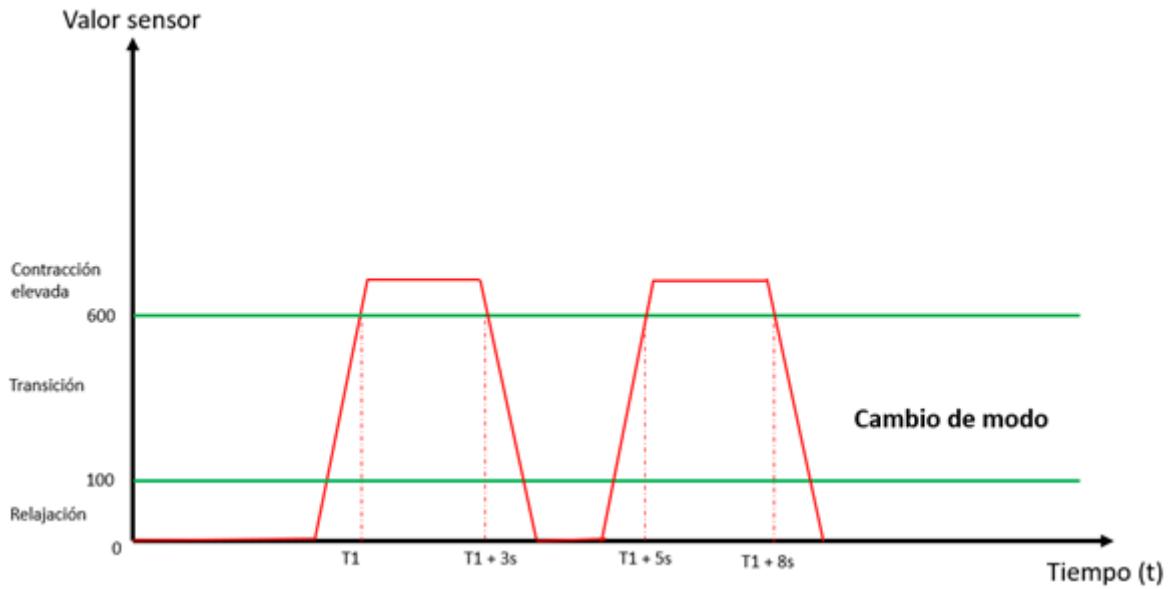


Figura 62: Patrón real de la señal a detectar

8.1.4 Modos de funcionamiento

La detección de patrones permite cambiar entre modos de control. El modo inicial activa todos los dedos, y el segundo solo el índice y el pulgar. Esta lógica está contenida en la función `controlFuncion` de la figura 63, que también permite bloquear la posición de los dedos para mantener un objeto sin esfuerzo muscular constante.

El valor umbral para activar el bloqueo es de 650. Una vez superado, se fijan los motores en su posición. Para desbloquear, la señal debe caer por debajo del umbral y luego superarlo nuevamente, como se muestra en la figura 64.

```

224 void controlFuncion(int sV, int mSV) {
225     if (ultimaValor - umbral1 <= 0 && sV - umbral1 >= 0) { //detectamos cuando sobre pasamos el umbral
226         cambioEstado = true; // es un cambio de estado
227         sobreLimita += 1; //permite de saber si estamos bloqueado o no
228     }
229     else{
230         cambioEstado = false;
231     }
232     if (cambioEstado){ // Si cambiamos de estado
233         if (sobreLimita % 2 == 0){ //si es par, es que debemos desbloquear
234             bloqueo = false;
235         }
236         else { //si es impar debemos bloquear
237             bloqueo = true;
238         }
239     }
240     ultimaValor = sV; //Guardamos la ultima valor

```

Figura 63: Código para activar o desactivar el bloqueo

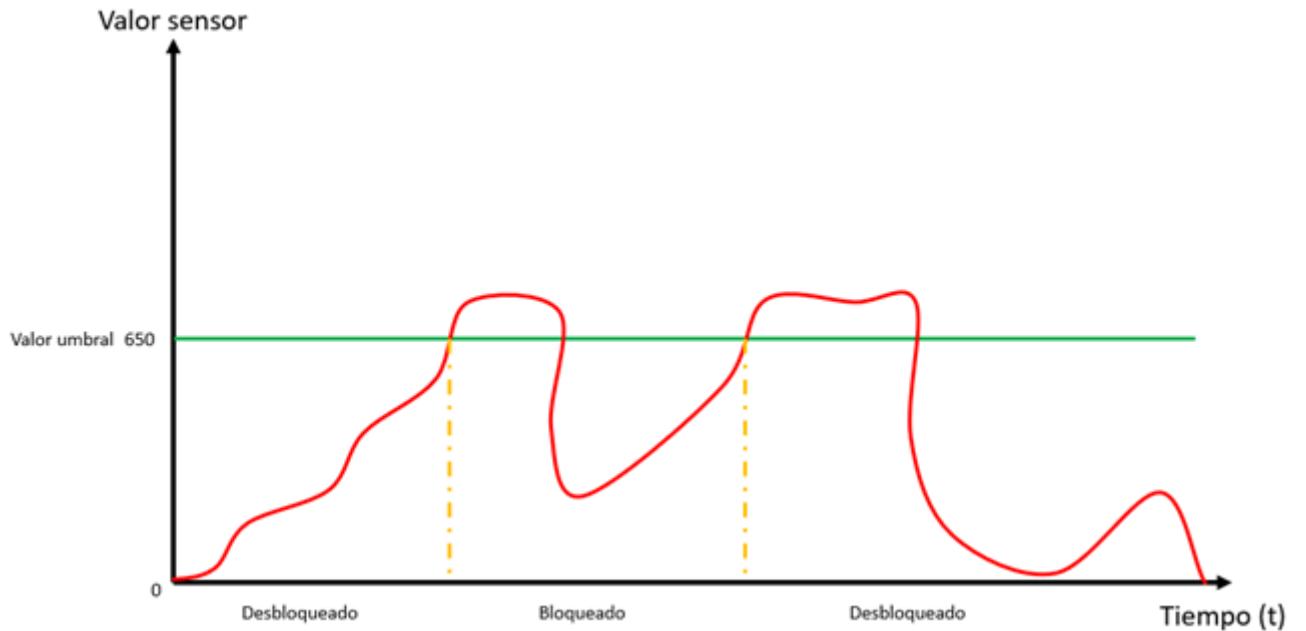


Figura 64: Representación del estado de la prótesis: bloqueada o desbloqueada

8.2 Control de ejecución y motores

Esta parte gestiona las tareas periódicas y controla los servomotores.

8.2.1 Gestión de tareas

La función `loop` de Arduino que aparece en la 65 se ejecuta indefinidamente. Para evitar sobrecargar la placa, se implementó un sistema de temporización basado en milisegundos. Se crearon dos variables (`os20` y `os100`) que permiten ejecutar ciertas tareas cada 20 y 100 ms, respectivamente. Esta estrategia distribuye la carga de procesamiento y evita la sobrecarga del sistema. Para no perder información y mantener un tiempo de reacción rápido, cada 20 ms se realiza la lectura y el procesamiento de la señal EMG. Dado que la detección de patrones opera en escalas de tiempo mayores (en segundos), esta se ejecuta cada 100 ms, momento en el que también se envían las instrucciones a los motores.

```

64 void loop() {
65
66     contador = millis();
67     // ##### Tarea de operaciones #####
68     // Tarea cada 100 ms
69     if (contador % os100 == 0) {
70         deteccion(); //Detectamos la paterna
71         controlFuncion(valorSensor, mapSensorValor); //actualizamos los valores dados a los motores
72     }
73
74     // Tarea cada 20 ms
75     if (contador % os20 == 0) {
76         valorSensor = analogRead(sensorPin); //leemos la valor del detector
77         mapSensorValor = tratamientoSenal(valorSensor); //tratamos la señal
78     }

```

Figura 65: Ejecución de tareas periódicas

8.2.2 Control de servomotores

El control de los servos se basa en el ángulo calculado por la función `controlFuncion`. El motor se mueve hasta alcanzar ese ángulo según el código de la figura 66.

```

100 void motorDerecha(int ObjAngD) {
101     if (posicionServoDerecha < ObjAngD) {
102         // Serial.println("on est rentré à droite d'un angle inférieur");
103         posicionServoDerecha++;
104         ServoDerecha.write(posicionServoDerecha);
105     }
106     else if (posicionServoDerecha > ObjAngD) {
107         //Serial.println("on est rentré à droite d'un angle supérieur");
108         posicionServoDerecha--;
109         ServoDerecha.write(posicionServoDerecha);
110     }
111 }
112
113 void motorIzquierda(int ObjAngI) {
114     if (posicionServoIzquierda < ObjAngI) {
115         //Serial.println("on est rentré à gauche d'un angle inférieur");
116         posicionServoIzquierda++;
117         ServoIzquierda.write(posicionServoIzquierda);
118     }
119     else if (posicionServoIzquierda > ObjAngI) {
120         //Serial.println("on est rentré à gauche d'un angle supérieur");
121         posicionServoIzquierda--;
122         ServoIzquierda.write(posicionServoIzquierda);
123     }
124 }
```

Figura 66: Código para accionar los servomotores

La velocidad de movimiento depende del valor asignado a la temporización.

Por ejemplo, si la variable de la velocidad es 20, el motor gira un grado cada 20 ms, lo que permite completar una rotación de 180° en aproximadamente dos tercios de segundo.

9 Pruebas y validaciones

Se realizaron diversas pruebas para evaluar las funcionalidades de la prótesis. A través de cuadros se presentan las características de los objetos que puede sujetar, así como el peso máximo soportado. Aunque no fue posible contar con un usuario real para probar directamente las funcionalidades vinculadas al código, se llevaron a cabo ensayos para evaluar la respuesta general del sistema.

9.1 Pruebas

Prueba n°1: Diámetro y forma de los objetos que pueden ser sujetados

Se utilizaron ocho objetos: cuatro con forma cilíndrica y cuatro con forma ortoédrica. El tamaño aumenta progresivamente para deducir un perfil de los objetos que la prótesis puede manipular.

- Etapa 1: alinear los objetos por forma, colocándolos sobre una mesa
- Etapa 2: acercar la prótesis hasta que los dedos rodeen el objeto
- Etapa 3: contraer el músculo para activar el cierre de la mano
- Etapa 4: realizar movimientos laterales y horizontales para comprobar si el objeto permanece en la mano
- Etapa 5: efectuar rotaciones con la prótesis
- Etapa 6: repetir las etapas anteriores para cada objeto

Se considera que un objeto supera satisfactoriamente la prueba si no se produjo ninguna falla durante todas las etapas.

Prueba n°2: Peso máximo soportado por la prótesis

Tras la primera prueba, se seleccionaron los dos objetos que la prótesis pudo sujetar con mayor eficacia. El objetivo de esta prueba es determinar el peso máximo que la prótesis puede sostener.

- Etapa 1: colocar el objeto vacío sobre una mesa
- Etapa 2: acercar la prótesis hasta que los dedos rodeen el objeto
- Etapa 3: contraer el músculo para sujetar el objeto
- Etapa 4: añadir 100 g de agua al objeto
- Etapa 5: esperar 5 segundos y observar si el objeto permanece sujeto
- Etapa 6: repetir las etapas 4 y 5 hasta que la prótesis no pueda sostener el peso adicional
- Etapa 7: repetir la prueba con el segundo objeto, de forma diferente

Estas pruebas permiten estimar con mayor precisión el peso máximo que la prótesis puede manipular de forma segura.

Prueba n°3: Validación de funcionalidades programadas

Esta prueba tiene como objetivo verificar el correcto funcionamiento de las funcionalidades programadas en el microcontrolador (Arduino), relacionadas con la interpretación de las señales EMG y la activación de los modos de agarre.

- Etapa 1: encender el sistema y asegurarse de que el modo inicial sea el de reposo
- Etapa 2: realizar una contracción simple y observar si se activa el primer modo de agarre
- Etapa 3: contractar el músculo al máximo para bloquear los dedos
- Etapa 4: repetir la etapa 3 para desbloquear los dedos
- Etapa 5: verificar visualmente que los dedos actúan según el patrón esperado
- Etapa 6: realizar una doble contracción para cambiar de modo
- Etapa 7: observar si el cambio de modo se ejecuta correctamente
- Etapa 8: repetir el procedimiento 10 veces para verificar la consistencia del reconocimiento de señales y la respuesta mecánica
- Etapa 9: registrar posibles errores, tiempos de respuesta lentos o contracciones mal interpretadas

Se evaluaron tres aspectos principales:

- **Reconocimiento de señales musculares:** se comprobaron los valores leídos por el sensor EMG, evaluando la diferencia entre el estado de reposo y la contracción voluntaria. Se verificó que la señal procesada supere un umbral establecido y active la secuencia correspondiente.
- **Cambio de modo de agarre:** se probó la funcionalidad que permite alternar entre distintos modos de cierre de la mano (por ejemplo, pinza lateral, agarre cilíndrico, o reposo). El cambio se realiza mediante una doble contracción rápida del músculo. Se midió la fiabilidad del reconocimiento de esta doble contracción y la respuesta del sistema.
- **Respuesta de los servomotores:** se evaluó la precisión del movimiento de los dedos y del pulgar, asegurando que los ángulos alcanzados correspondan con los programados para cada modo. También se observó el tiempo de reacción del sistema desde la señal EMG hasta el movimiento completo del motor.

Se consideró que el sistema funcionaba correctamente si:

- La señal EMG diferenciaba de manera estable entre reposo y contracción.
- Los modos se cambiaban de forma intencionada, sin falsas activaciones.

- Los movimientos ejecutados por los servomotores eran consistentes con el modo activo y no presentaban bloqueos o comportamientos erráticos.

Los resultados demostraron que, bajo condiciones de prueba controladas, la prótesis responde adecuadamente a las señales musculares y ejecuta los movimientos esperados. Aun así, se recomienda ajustar el umbral de activación y el tiempo de detección de la doble contracción para mejorar la sensibilidad y evitar activaciones no deseadas.

9.2 Validaciones

Diámetro del objeto	3 cm	5 cm	7 cm	10 cm
Resultado	✗	✓	✓	✗
Observaciones	No logra ajustarse al objeto	✓	Requiere apertura manual adicional de los dedos	La apertura de la mano no es suficiente para rodear el objeto

Tabla 8: Resultados de la prueba n°1 para objetos cilíndricos

En la tabla 8 se presentan los resultados de la prueba n°1 con objetos cilíndricos. Se deduce un rango óptimo de diámetros entre 5 cm y 7 cm aproximadamente. Se observó que la movilidad del pulgar es el factor más determinante para sujetar objetos: si el pulgar no alcanza el objeto o no ofrece una fuerza de oposición suficiente, la prótesis falla en su función. En algunos casos, los objetos con grandes diámetros tienden a deslizarse, por lo que se aplicó magnesia en las puntas de los dedos para mejorar la adherencia.

Ancho del objeto	3 cm	5 cm	8 cm	10 cm
Resultado	✓	✓	✓	✗
Observaciones	A diferencia de los objetos cilíndricos, el pulgar tiene mejor apoyo en objetos ortoédricos	✓	Requiere apertura manual adicional de los dedos	La apertura de la mano no es suficiente para rodear el objeto

Tabla 9: Resultados de la prueba n°1 para objetos ortoédricos

La tabla 9 resume el perfil de los objetos ortoédricos que la prótesis puede manipular eficazmente. En todos los casos, es indispensable que el pulgar esté en contacto con el objeto para garantizar un agarre firme. Por lo tanto, la altura del objeto debe ser suficiente para alcanzar el pulgar.

Forma del objeto	Peso máximo soportado	Observaciones
Cilíndrico	530 g	El objeto comienza a deslizarse con pesos superiores. Se recomienda mejorar el agarre o reducir la velocidad de apertura.
Ortoedro	800 g	Mejor distribución de la fuerza y mayor superficie de contacto.

Tabla 10: Validación de la prueba N°2: peso máximo soportado por la prótesis

El cuadro de resultados 10 muestra el peso máximo que la prótesis puede soportar. Se estima que, con un objeto de forma cónica, sería posible levantar un peso mayor, ya que su geometría ayudaría a prevenir el deslizamiento. No obstante, las dimensiones de dicho objeto cónico deberían mantenerse dentro del rango de diámetros determinado en la primera prueba.

En el cuadro 11, se nota sobre 10 ensayos, como reaccionó la prótesis según la funcionalidad pedida: cambiar de modo, bloquear, desbloquear y mover los dedos en modo desbloqueado.

Funcionalidad	Éxito (sobre 10)	Observaciones
Cambiar de modo	8/10	En dos ocasiones, las contracciones fueron demasiado cercanas entre sí, lo que llevó al sistema a interpretar erróneamente una contracción para mover los dedos como un cambio de modo
Bloquear	10/10	La función respondió correctamente en todos los intentos
Desbloquear	10/10	La prótesis reaccionó con precisión a la orden de desbloqueo
Mover los dedos (modo desbloqueado)	10/10	Los movimientos fueron realizados correctamente según lo programado

Tabla 11: Validaciones de la prueba N°3: funcionalidades programadas

La prueba de la programación y actuación de la prótesis puede considerarse un éxito. En cuanto al cambio de modo, es importante tener en cuenta que no debe intentarse activarlo inmediatamente después de una acción de bloqueo o desbloqueo. Es necesario

esperar al menos 2 segundos antes de iniciar una nueva contracción siguiendo el patrón correspondiente al cambio de modo, con el fin de evitar errores de interpretación por parte del sistema.

Las pruebas realizadas permitieron evaluar eficazmente las capacidades mecánicas y funcionales de la prótesis. La primera prueba evidenció que puede sujetar objetos de forma cilíndrica u ortoédrica con dimensiones específicas, destacando el rol fundamental del pulgar en la sujeción. La segunda prueba confirmó que la prótesis puede soportar pesos de hasta 530 g para objetos cilíndricos y 800 g para ortoedros, con mejores resultados cuando la forma del objeto limita el deslizamiento. Por último, la tercera prueba validó la fiabilidad del sistema de control por EMG, con una tasa de éxito alta en todas las funciones, aunque se identificó la necesidad de espaciar las contracciones para evitar errores de interpretación. En conjunto, estas pruebas confirman la funcionalidad básica de la prótesis y brindan información clave para futuras mejoras.

10 Perspectivas

La prótesis final está disponible de forma libre sobre la plataforma github [21]. Se puede encontrar los archivos stl de las piezas a imprimir, los códigos y el informe final del proyecto. Significa que cualquier persona puede recuperar el proyecto, mejorarlo, cambiar el diseño o el programa.

10.1 Diseño Final

Se presenta el diseño final de la prótesis con sus movimientos. El material utilizado entre la prótesis y la piel del usuario es etilvinilacetato (EVA), conocido como goma eva o foamy, lo cual proporciona un contacto cómodo entre la piel y la prótesis. La ventilación se mantiene gracias a los huecos en el sistema de ajuste. Finalmente, al juntar todas las piezas y seguir las instrucciones de montaje según se especifica en el Anexo B, se obtiene el diseño final.

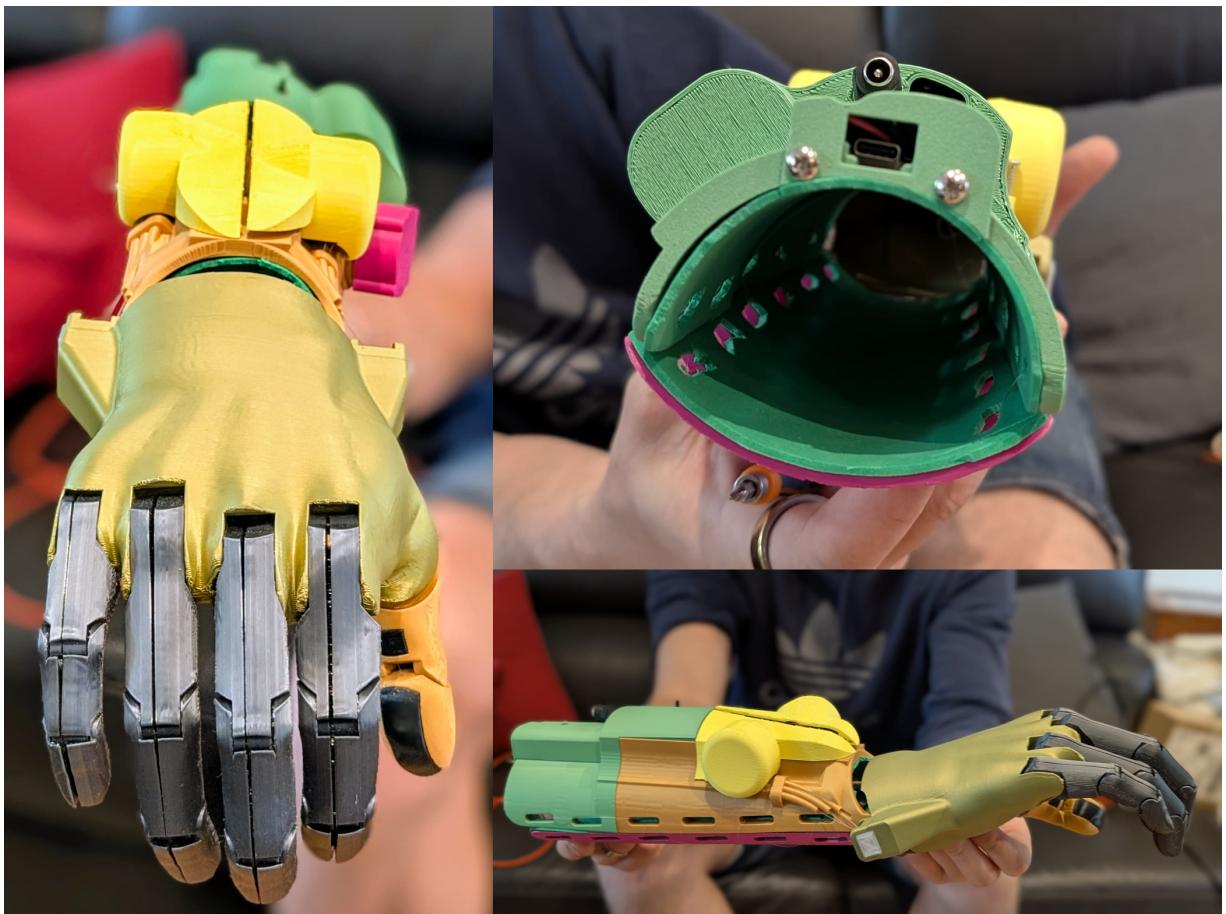


Figura 67: Diseño final

El sistema se activa al encender el interruptor y presionar el botón de la caja del receptor. Posteriormente, el sensor, ubicado sobre el brazo o cualquier otro músculo, permite controlar la prótesis.



Figura 68: 2 tomas distintas

La prótesis cumple satisfactoriamente con los objetivos funcionales propuestos. Es capaz de sujetar una variedad de objetos con diferentes formas y tamaños, especialmente aquellos con geometrías rectangulares, donde ofrece una sujeción firme y estable. En cuanto a la capacidad de carga, logró soportar hasta 800 gramos, lo cual representa un excelente desempeño considerando su diseño actual. Además, las funcionalidades programadas fueron validadas con éxito: el cambio de modo, el bloqueo y desbloqueo de los dedos, así como el movimiento individual en modo desbloqueado, respondieron correctamente en la gran mayoría de los ensayos. El sistema mostró una buena reactividad y fiabilidad, lo que permite una interacción fluida entre las señales musculares y la respuesta mecánica. En conjunto, la prótesis representa una solución funcional, versátil y prometedora, con un rendimiento más que satisfactorio en escenarios de uso realista.

10.2 Fabricación y Comercialización

El diseño de esta prótesis está completamente realizado mediante impresión 3D. Es necesario contar con impresoras de buena calidad, ya que algunas piezas son complicadas de fabricar. Algunas impresoras no permiten retirar los soportes en los huecos o no tienen las mismas características de impresión que las que usé. La idea con este prototipo es verificar primero si esta prótesis se adapta a cualquier persona. En ese caso, el objetivo sería producirla en serie utilizando otras técnicas de fabricación que sean más eficientes y rápidas.

10.2.1 Moldeo por Inyección

El moldeo por inyección es una tecnología de fabricación ampliamente conocida. Permite producir piezas complejas en serie con alta capacidad de producción. Representa el 40 % de las piezas plásticas producidas a nivel mundial [12]. El proceso se realiza en tres etapas [13]:

Etapa 1: Creación del Moldeo

El moldeo es la parte en la que se inyecta el plástico en su estado líquido. Se lleva a cabo utilizando un software de CAO (como los que usamos para crear las piezas del proyecto). El moldeo se realiza en dos piezas de aluminio, generalmente.

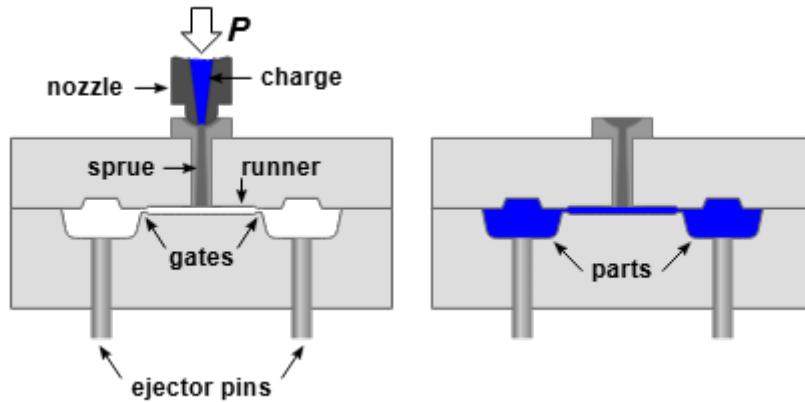


Figura 69: Creación de piezas por inyección

Etapa 2: Inyección

El molde se coloca en la máquina de inyección. Dos barras metálicas mantienen el molde firmemente en su lugar. El material elegido para la pieza se calienta hasta su punto de fusión antes de ser inyectado en el molde con alta presión. Luego, se espera que el material se enfríe para obtener su forma final.

Etapa 3: Expulsión y Acabado

Después de la segunda etapa, se libera el sistema de ajuste del molde. Se utiliza el molde para producir otras piezas. Se realizan acabados sobre la pieza con herramientas industriales, se retiran las partes que sirvieron para la inyección y se lleva a cabo un tratamiento de la superficie.



Figura 70: Etapas del proceso de inyección

Resumen de las tres etapas principales de este método: Para los huecos dentro de las piezas, agregar núcleos dentro de los moldes permite evitar que el plástico llene esos espacios, creando así los huecos deseados.

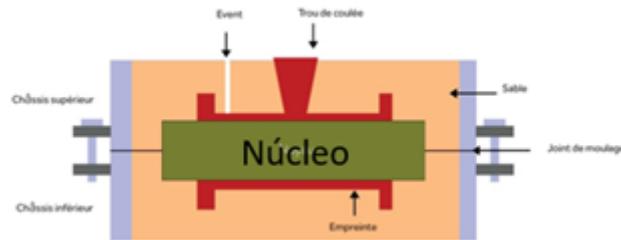


Figura 71: Uso de un núcleo para crear un hueco

10.3 Ejemplo: Producción de la Mano

La mano que contiene los dedos es una pieza compleja con diferentes ángulos y huecos para los hilos. La primera etapa consiste en crear los moldes a partir del software de modelización.

10.3.1 Moldeos

En la figura 75, se muestran las dos partes del molde. La parte superior contiene la boquilla, que es la encargada de hacer llegar el plástico fundido a la pieza. También se observa la primera parte del núcleo central (la mano está vacía). El núcleo se mantiene en su posición porque está acoplado con la base del molde, como se muestra en la siguiente imagen.

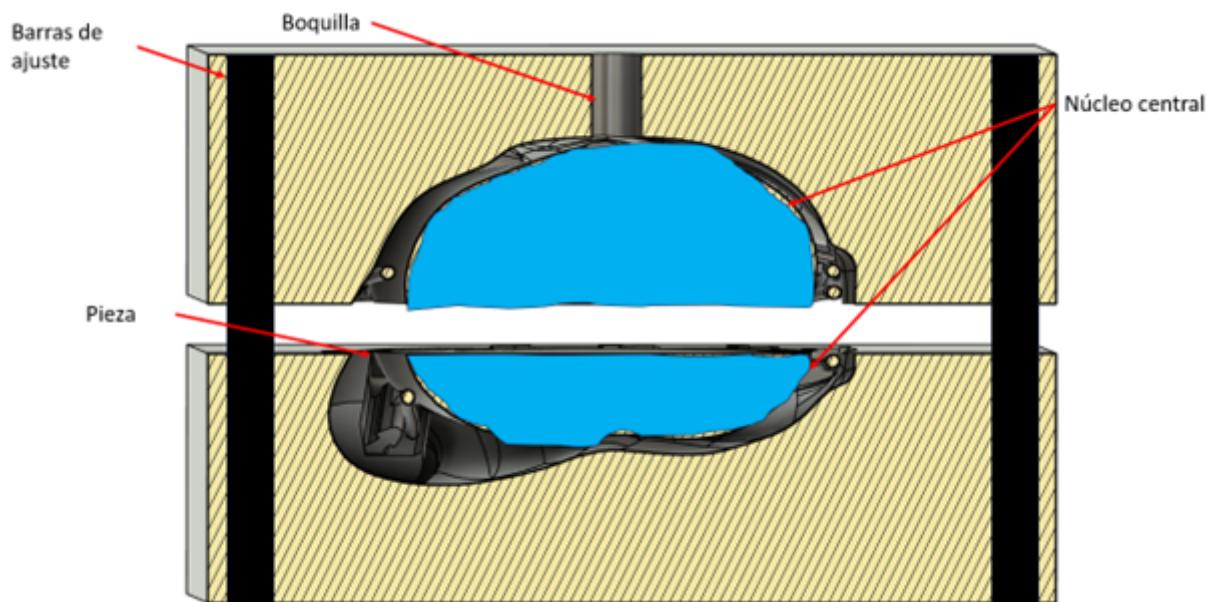


Figura 72: Esquema de creación de la palma por inyección

El núcleo en la imagen está dividido en dos partes para mostrar el espacio que ocuparía el núcleo cuando el molde esté cerrado.

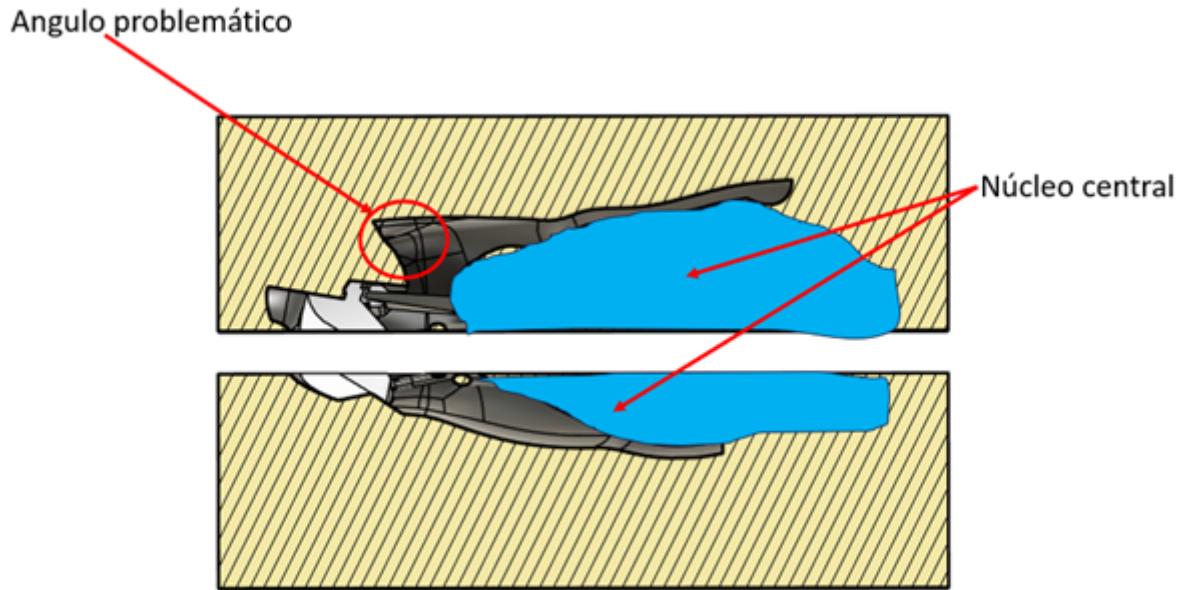


Figura 73: Vista de corte

Debido a los núcleos en los huecos atravesados por los hilos, se sugiere hacer los huecos más rectos para simplificar la producción. También es importante evitar los ángulos que podrían dificultar la extracción de la pieza una vez que se haya enfriado. Para solucionar esto, es necesario planificar cuidadosamente la forma del molde, teniendo en cuenta el plano de corte y evitando estos ángulos durante el diseño de la pieza.

10.4 Mejoras

Durante el diseño y las pruebas realizadas, se observaron varias mejoras que podrían ser implementadas en la prótesis. El proyecto está en 'open source', lo que permite que cualquier persona tenga acceso a los archivos STL y al informe.

10.4.1 Partes Mecánicas y Activación

El material utilizado ofrece varias ventajas en la fabricación de las piezas, pero presenta algunas limitaciones en cuanto a resistencia y estética.

Sistema de Ajuste

Las distintas partes que componen el sistema de ajuste son las que sufrirán la mayor deformación en las áreas de los dientes. El material se deteriorará rápidamente debido a su falta de elasticidad y baja resistencia a la fatiga. Esto implica que será necesario aumentar el grosor de las piezas, aunque incluso así no será suficiente para asegurar su buen funcionamiento a largo plazo. Comprar un sistema BOA podría ser una mejor solución a lo largo del tiempo. Se puede encontrar en el sitio vestiaire pro a 14,90 [vestiaire]. Este sistema se encuentra en muchos tipos de equipos, tanto de calzado como de deportes, y permite un ajuste preciso sin esfuerzos. Las piezas cuentan con garantía de por vida.

Diseño

El diseño está optimizado para ubicar las diferentes piezas de manera eficiente y re-

ducir la fricción de los cables que accionan los dedos. En algunos casos, la resistencia de la prótesis depende del grosor y relleno de las piezas. Aumentar estas propiedades incrementa el tamaño y peso de la prótesis, lo que afecta también su aspecto visual. Lo ideal sería contar con un material que ofrezca una buena resistencia con una densidad baja. El aluminio sería el mejor compromiso en este caso, ya que permitiría reducir el tamaño de la prótesis y mejorar su apariencia. Además, la elección de los componentes electrónicos podría mejorarse para hacer el sistema más compacto. Una placa electrónica con los componentes integrados permitiría reducir el uso de cables, lo que proporcionaría un ahorro significativo de espacio.

Agarramiento

La principal limitación de la prótesis es su capacidad de agarre, ya que el movimiento de los dedos no está completamente logrado. En particular, el pulgar juega un papel fundamental en la funcionalidad de la mano, ya que su oposición y movilidad permiten realizar agarres más precisos y estables. Por esta razón, es crucial optimizar su movimiento para mejorar la eficacia general de la prótesis. Una posible mejora sería controlar el pulgar de forma independiente respecto a los demás dedos, lo que permitiría una mayor variedad y precisión en los movimientos y agarres. Además, la incorporación de una articulación rotular (rótula) en el diseño podría proporcionar una mayor libertad y naturalidad en el movimiento del pulgar, facilitando diferentes tipos de agarres y mejorando la adaptabilidad a objetos de variadas formas y tamaños. Para avanzar, es necesario realizar más pruebas con los materiales actuales de los flexibones o explorar otros materiales que ofrezcan un equilibrio óptimo entre flexibilidad, resistencia y durabilidad.

10.4.2 Partes Electrónicas y Control

Suministro del Sensor

El sensor es el punto débil en términos de autonomía de la prótesis. Su batería suministra energía al sistema del sensor durante 30 minutos antes de perder eficiencia. Es posible mejorar el circuito para alimentarlo con la batería, pero esto presenta algunas desventajas. Se perdería potencia debido a los servomotores, y se agregarían cables que tendrían que llegar hasta el músculo encargado del control de la prótesis. Otra opción sería crear un brazalete que albergue tanto el sensor como la batería. Esto permitiría un mejor ajuste estético y una mayor solidez, y el brazalete podría colocarse sobre cualquier músculo del cuerpo.

Programación y Detección de la Señal

La placa Arduino puede ser reprogramada en cualquier momento, lo que permite modificar la manera en que se trata la señal, cambiar de modo o ajustar el tiempo de operación. Una mejora sería agregar un segundo sensor y más servomotores para controlar los dedos de manera independiente. Según la modificación del prototipo, se podrían añadir dos servomotores adicionales: uno para el dedo índice y otro para el dedo medio. El segundo sensor permitiría diferenciar los movimientos de la mano.

Agregar Sensores para la Sensación de Tacto

Una opción que se encuentra en las prótesis comercializadas es la retroalimentación para el usuario mediante sensores ubicados en la punta de los dedos. La idea es colocar sensores de deformación que, al detectar que el usuario ha agarrado un objeto, generen

una vibración en la prótesis.

Referencias

- [1] *LES DIFFERENTES CAUSES DE L'AMPUTATION.* URL: <https://milienmodestin.wixsite.com/tpeprothese1ers2/les-differentes-causes?>.
- [2] AFP agence. *Près de 8000 amputations dues au diabète chaque année.* 2015. URL: <https://sante.lefigaro.fr/actualite/2015/11/10/24298-pres-8000-amputations-dues-diabete-chaque-annee>.
- [3] Aissatou Koné. *Impact psychologique et socioprofessionnel de l'amputation chez les diabétiques hospitalisés dans le service de médecine et d'endocrinologie de l'hôpital du Mali.* 2023. URL: <https://www.biblio.sante.ml/handle/123456789/12569>.
- [4] Institut canadien d'information sur la santé. *Une analyse de l'équité révèle les populations les plus à risque de subir une amputation liée au diabète.* 2024. URL: <https://www.cihi.ca/fr/nouvelles/une-analyse-de-lequite-revele-les-populations-les-plus-a-risque-de-subir-une-amputation-liee-au>.
- [5] Amputee care center. *Quel est le coût d'une prothèse ?* 2024. URL: <https://amputeeccarecenter.com/fr/blog/quel-est-le-cout-dune-prothese>.
- [6] Dominique Boutet. "Une morphologie de la gestualité : structuration articulaire". Tesis de mtría. Université EVE - UMR, 2008. URL: <https://hal.science/hal-00607593/document#:~:text=L'avant%2Dbras%20poss%C3%A8de%20deux,paume%20ouverte%20vers%20l'avant..>
- [7] Eric Roulot. *L'anatomie fonctionnelle de la main : Les mouvements et les articulations.* Accessed: 2025-03-02. 2020. URL: <https://eric-roulot.com/main-poignet/anatomie-main-poignet/#:~:text=L'anatomie%20fonctionnelle%20de%20la,1'on%20appelle%20la%20pronosupination..>
- [8] Ottobock. *Bras et mains prothétiques.* Accessed: 2025-03-03. URL: <https://www.deltamedicalpro.com/la-prothese-medicale-de-l-antiquite-a-aujourd'hui/>.
- [9] Paola Da Silva. "Bras bionique : un troisième patient français opéré en mars 2023". En: *Harmonie santé* (2023). URL: <https://www.harmonie-sante.fr/sante-soins/innovations/bras-bionique-troisieme-patient-francais-opere-mars-2023>.
- [10] Open Bionics. *Meet the Hero Arm – a prosthetic arm for adults and children.* URL: <https://openbionics.com/hero-arm/>.
- [11] Team Gre-Nable. *Main Assistée Electriquement Flexibone.* 2019. URL: <https://www.gre-nable.fr/main-assistee-electriquement-flexibone/>.
- [12] Guillaume ROBERT. *Tout savoir sur le système de serrage Boa.* 2016. URL: <https://www.matosvelo.fr/index.php?post/1861/tout-savoir-sur-le-systeme-de-serrage-boa>.
- [13] Free3D. *Free3D.* URL: <https://free3d.com/fr/3d-models/>.
- [14] Onshape. *The Perfect CAD PDM Solution for Education.* URL: <https://www.onshape.com/en/>.
- [15] Onshape CAO. URL: <https://cad.onshape.com/documents/9b0b2ccc0b8e81e82a015503/w/49b84c15c0dc949d471b8de1/e/fc7681d7907e35a0d61a9aad>.

- [16] *Industrial automation systems and integration — Product data representation and exchange — Part 21: Implementation methods: Clear text encoding of the exchange structure.* International Organization for Standardization, 2016.
- [17] MyoWare. *MyoWareBLEPeripheral.ino*. Accessed: 2025-05-28. URL: <https://github.com/AdvancerTechnologies/MyoWare-Arduino-Library/blob/main/examples/MyoWareBLEPeripheral/MyoWareBLEPeripheral.ino>.
- [18] SparkFun Electronics. Accessed: 2025-05-28. URL: <https://www.sparkfun.com/>.
- [19] Sparkfun. *Code for MyoWare Central*. 2020. URL: <https://github.com/AdvancerTechnologies/MyoWare-Arduino-Library/blob/main/examples/MyoWareBLECentral/MyoWareBLECentral.ino>.
- [20] Sparkfun. *Installing board for MyoWare*. 2020. URL: <https://learn.sparkfun.com/tutorials/installing-board-definitions-in-the-arduino-ide/installing-a-third-party-board-definition>.
- [21] Thibault Girard. *Protesis-Base-proyecto-Gre-nable-*. 2025. URL: <https://github.com/Thib89-167/Protesis-Base-proyecto-Gre-nable-/tree/main>.

Apéndice A: Entrevista con Sandrine Calvet

Usaria con agenesia de brazo derecho

Perfil de la usuaria

- Agenesia congénita – antebrazo derecho
- Muñón de 5 cm, difícilmente adaptable según el centro de prótesis
- Primera prótesis a los 14 años

Uso actual de prótesis

- Tipo: Prótesis cosmética (revestimiento de guante de PVC, uñas postizas)
- Frecuencia de uso: Usa la prótesis todo el tiempo (nunca sale sin ella)
- Criterio prioritario: Estética (forma de la mano, apariencia realista)
- Peso: Se probó una versión móvil (630 g), que provocó dolores importantes en la espalda
- Mantenimiento:
 - Cambio del guante: aproximadamente 1 hora
 - Frecuencia: 2 prótesis cada 5 años
 - Lavado: Actualmente posible con algunos modelos, pero los guantes de silicona son muy frágiles y no están cubiertos por la seguridad social

Intentos y vista sobre otros tipos de prótesis

- Prótesis mioeléctrica: demasiada cara y no proponen una buena adaptación
 - "La mayor ventaja que veo en una protesis myoélectrica es de sujetar fácilmente objetos, por ejemplo el manillar de una bicicleta"
 - "Debe ser fácil y fluida de utilizar, ya que está pensada para un uso diario"
- Tipo: Encaje de silicona con funda de silicona sobre el muñón y vástago rígido
 - Inconveniente: El codo triplicaba su volumen, lo que impedía el uso de ropa

Limitaciones específicas

- Morfología particular:
 - Brazo muy delgado
- Dolores de espalda relacionados con el peso de la prótesis

Aspectos prioritarios identificados

1. Estética y realismo (apariencia de la mano, uñas, guante resistente)
2. Ligereza, para evitar dolores de espalda
3. Adaptación a la morfología específica (brazo delgado + mano larga)
4. Facilidad de mantenimiento y materiales duraderos

Apéndice B:

Montaje

Lista de las piezas Hacemos la lista de cada componente necesario para el montaje de cada módulo de las piezas. Podemos encontrar un mismo componente en 2 montajes. Estará especificado.

Pulgar	
Hueso distal	x1
Hueso proximal	x1
Proximal cáscara izquierda	x1
Proximal cáscara derecha	x1
Distal cáscara izquierda	x1
Distal cáscara derecha	x1
Hilo de pescar 0.4mm*1m	x1

Index	
Index flexibone	x1
Media cáscara izquierda	x1
Media cáscara derecha	x1
Proximal cáscara izquierda	x1
Proximal cáscara derecha	x1
Distal cáscara izquierda	x1
Distal cáscara derecha	x1
Hilo de pescar 0.4mm*1m	x1

Index	
Medio flexibone	x1
Media cáscara izquierda	x1
Media cáscara derecha	x1
Proximal cáscara izquierda	x1
Proximal cáscara derecha	x1
Distal cáscara izquierda	x1
Distal cáscara derecha	x1
Hilo de pescar 0.4mm*1m	x1

Anular	
Anular flexibone	x1
Media cáscara izquierda	x1
Media cáscara derecha	x1
Proximal cáscara izquierda	x1
Proximal cáscara derecha	x1
Distal cáscara izquierda	x1
Distal cáscara derecha	x1
Hilo de pescar 0.4mm*1m	x1

Meñique	
Meñique flexibone	x1
Media cáscara izquierda	x1
Media cáscara derecha	x1
Proximal cáscara izquierda	x1
Proximal cáscara derecha	x1
Distal cáscara izquierda	x1
Distal cáscara derecha	x1
Hilo de pescar 0.4mm*1m	x1

Sistema de ajuste	
Hilo 0.4mm*2m	x1
Rueda	x1
Muñeca	x1
Caja de ajuste (parte del gante)	x1

Gante	
Caja servomotor izquierda	x1
Caja servomotor derecha	x1
Gante	x1
Circuito electrónico	x1

Circuito electrónico	
Carta micro arduino	x1
Sensor	x1
Interruptor	x1
Regulador	x1
Servomotor	x2
Receptor	x1
Batería	x1

Caja de atrás	
Caja de atrás	x1
Tornillos M2x30	x2
Caja del receptor	x1
Botón	x1
Resorte	x2
Receptor (mismo que en el circuito)	x1
Tornillos M1*20	x2

Palma	
Dedos	x4
Pulgar	x1
Tuerca	x2
Palma	x1

Etapa 1: Recuperar todas las piezas que componen un dedo. Debe haber 7 piezas, 6 en PLA y una en TPU.	
Etapa 2: Acoplar el flexibone con el proximal cáscara, después el media y por fin el distal cáscara. Hacer pasar el hilo a través de los huecos.	
Etapa 3: Cerrar el dedo con las cáscaras en el mismo orden que los primeros.	
Etapa 4: Ubicar el dedo sobre la mano y forzar el ingreso en el hueco hasta que no pueda salir más.	
Montaje final	

Noticia: montaje de los dedos

Etapa 1: Colocar el receptor en el espacio de la pieza.	
Etapa 2: Conectar los cables del circuito principal al receptor.	
Etapa 3: Usar los tornillos M1x20 para fijar la caja del receptor.	
Etapa 4: Verificar que el botón está bien centrado con el botón del receptor.	

Noticia: montaje de la caja de atrás

Etapa 1: Asegurarse que el circuito está completo y no hay discontinuidad sobre entre los cables.	
Etapa 2: Centrar los huecos del gante con los huecos de la caja de atrás. Usar los 2 tornillos M2x20 para fijar las 2 partes.	
Etapa 3: Verificar que el sistema funciona correctamente activando el interruptor.	

11.1.7. Noticia de montaje de la caja de atrás con el gante

Etapa 1: Alinear los huecos de la mano con los huecos del gante.	
Etapa 2: Hacer pasar las tuercas en los huecos. Hacer pasar los hilos de los dedos a través de los huecos hechos en las tuercas.	
Etapa 3: Según la voluntad de mantener el grado de libertad de la muñeca, ajustar las tuercas.	
Etapa 4: Hacer pasar los hilos en los huecos previstos en el gante.	

Apéndice C: Arduino code

Here are the main parts of the code use to produce the various results in this documents.

```
1 #include <Servo.h>
2
3 Servo ServoDerecha;
4 Servo ServoIzquierda;
5
6 //Inicializacion hardware
7 int servomotorDerechaPin = 6;
8 int servomotorIzquierdaPin = 10;
9 int sensorPin = A0;
10 int posicionServoDerecha = 5;
11 int posicionServoIzquierda = 180;
12
13 unsigned long contador = 0;
14 int velocidadD = 1;
15 int objetivoAnguloD = 0;
16 int velocidadI = 1;
17 int objetivoAnguloI = 0;
18 int ultimaValor = 0;
19 int modo = 0;
20 int valorSensor = 0;
21 int mapSensorValor = 0;
22 int anguloTecho = 30; //valor minima de diferencia entre un angulo y el
→ objetivo para que los motores funcionan
23 int ultimaMapValor = 0;
24 int maxAngulo = 180;
25 int minAngulo = 5;
26
27 // bloqueo
28 int ultimaValor = 0;
29 bool cambioEstado = false;
30 bool bloqueo = false;
31 int sobreLimita = 0;
32 int umbral1 = 610;
33
34 //Operacional task a 20 y 200 ms
35 int os20 = 20;
36 int os100 = 100;
37
38
39 //Control de la detección de señal
40 unsigned long T0 = 0;
41 unsigned long T1 = 0;
42 unsigned long T2 = 0;
43 int contraccion = 0;
44
45 // Detection de Patern et changement de mode
46 int calibrationDuree1 = 3000;
47 int calibrationDuree2 = 2000;
48 int stepDetection = 0;
```

```

1
2 void setup() {
3     // inicializacion de los motores y del pin de entrada de la señal
4     Serial.begin(115200);
5     ServoDerecha.attach(servomotorDerechaPin);
6     ServoIzquierda.attach(servomotorIzquierdaPin);
7     pinMode(sensorPin,INPUT);
8
9     // Posicion inicial de los motores a 0 grados
10    //ServoDerecha.write(posicionServoDerecha);
11    //ServoIzquierda.write(posicionServoIzquierda);
12    digitalWrite(A0, LOW);
13 }
14
15 void loop() {
16
17     contador = millis();
18     // ##### Tarea de operaciones #####
19     // Tarea cada 20 ms
20     if (contador % os20 == 0) {
21         valorSensor = analogRead(sensorPin); //leemos la valor del detector
22         mapSensorValor = tratamientoSenal(valorSensor); //tratamos la señal
23     }
24
25     // Tarea cada 100 ms
26     if (contador % os100 == 0) {
27         deteccion(); //Detectamos la paterna
28         controlFuncion(valorSensor, mapSensorValor); //actualizamos los valores
29             → dados a los motores
30     }
31
32     // ##### Pilotage Motor #####
33     if (contador % velocidadD == 0) {
34         //Serial.println("DROIT --> " + String(objetivoAnguloD) + " POS " +
35             → String(posicionServoDerecha));
36         motorDerecha(objetivoAnguloD);
37     }
38
39     if (contador % velocidadI == 0) {
40         //Serial.println("GAUCHE -->" + String(objetivoAnguloI)+ " POS " +
41             → String(posicionServoIzquierda));
42         motorIzquierda(objetivoAnguloI);
43     }
44
45     int tratamientoSenal(int sV) {
46         int mapValor = map(sV, 0, 675, minAngulo, maxAngulo);
47         if (abs(ultimaMapValor - mapValor) > anguloTecho){
48             ultimaMapValor = mapValor;
49             return mapValor;
50         }
51     }

```

```

1
2 void motorDerecha(int ObjAngD) {
3     if (posicionServoDerecha < ObjAngD) {
4         // Serial.println("on est rentré à droite d'un angle inférieur");
5         posicionServoDerecha++;
6         ServoDerecha.write(posicionServoDerecha);
7     }
8     else if (posicionServoDerecha > ObjAngD) {
9         //Serial.println("on est rentré à droite d'un angle supérieur");
10        posicionServoDerecha--;
11        ServoDerecha.write(posicionServoDerecha);
12    }
13 }
14
15 void motorIzquierda(int ObjAngI) {
16     if (posicionServoIzquierda < ObjAngI) {
17         //Serial.println("on est rentré à gauche d'un angle inférieur");
18         posicionServoIzquierda++;
19         ServoIzquierda.write(posicionServoIzquierda);
20     }
21     else if (posicionServoIzquierda > ObjAngI) {
22         //Serial.println("on est rentré à gauche d'un angle supérieur");
23         posicionServoIzquierda--;
24         ServoIzquierda.write(posicionServoIzquierda);
25     }
26 }
27
28 void deteccion(){
29 /*if (contador%2000 == 0){
30     Serial.println("Contraccion info " + String(contraccion)+ " Pattern
31     ↪ detection step --> " + String(stepDetection) + "Le mode est "
32     ↪ String(modo) + "valeur sensor " + String(valorSensor));
33 }*/
34 int ultimaContraccion = contraccion;
35 boolean flancoAscendente = false;
36 boolean flancoDescendente = false;
37
38 if (valorSensor < 100){ //attribuamos 3 valores a contraccion segun la
39     ↪ intensidad
40     contraccion = 0;
41 }
42 else if (valorSensor > 600){
43     contraccion = 2;
44 }
45 else {
46     contraccion = 1;
47 }
```

```

1 // ##### detection fronts #####
2 if (ultimaContraccion == 1 && contraccion == 2){ //si estabamos en un
→ estado de transicion y que detectamos una contraccion superior a 600
3 flancoAscendente = true; //detectamos un frente subiendo de un PMW
4 //Serial.println("on a détecté un front montant");
5 }
6 if (ultimaContraccion == 1 && contraccion == 0){ //si estabamos en un
→ estado de transicion y que detectamos una contraccion inferior a 100
7 flancoDescendente = true; //detectamos un frente bajando de un PMW
8 //Serial.println("on a détecté un front descendant");
9 }
10 // ###### Patern detection #####
11 if (stepDetection == 0 && flancoAscendente) { //si somos a la primera
→ etapa de la detección y que detectamos un frente subiendo
12 T0 = contador; //Guardamos el "tiempo" al momento que detectamos
13 stepDetection = 1; //Pasamos a la segunda etapa
14 Serial.println("on a le premier front montant à " + String(T0) +
→ "contraction " + String(valorSensor) + "temps " + String(contador));
15 }
16 //-----
17 if ((stepDetection == 1) && (contador - T0 > calibrationDuree1)) { // Si
→ estamos a la segunda etapa pero que el tiempo entre la proxima etapa y
→ la ultima es demasiado largo
18 stepDetection = 0; //Volvemos a la primera etapa
19 Serial.println("on réénitialise premiere etape trop longue car " +
→ String(contador-T0));
20 }
21 //-----
22 if ((stepDetection == 1 && flancoDescendente) && (contador - T0 <
→ calibrationDuree1)) { //Si somos a la segunda etapa de la detección en
→ el buen tiempo y que detectamos un frente bajando
23 T1 = contador; //Guardamos el "tiempo" al momento que detectamos
24 stepDetection = 2; //Pasamos a la tercera etapa
25 Serial.println("on a le premier front descendant à " + String(T1) +
→ "contraction " + String(valorSensor) + "temps " + String(contador));
26 }
27 //-----
28 if ((stepDetection == 2 && flancoAscendente) && (contador - T1 <
→ calibrationDuree2)) { //Si somos a la tercera etapa de la detección en
→ el buen tiempo y que detectamos un frente subiendo
29 T2 = contador; //Guardamos el "tiempo" al momento que detectamos
30 stepDetection=3; //Pasamos a la tercera etapa
31 Serial.println("on a le deuxième front montant");
32 }

```

```

1  if ((stepDetection == 2) && (contador - T1 > calibrationDuree2)) { // Si
2    ↵ estamos a la tercera etapa pero que el tiempo entre la proxima etapa y
3    ↵ la ultima es demasiado largo
4    stepDetection = 0; //Volvemos a la primera etapa
5    Serial.println("on réinitialise deuxieme etape trop longue car " +
6      ↵ String(contador-T1));
7  }
8  //-----
9  if ((stepDetection == 3 && flancoDescendente) && (contador - T2 <
10   ↵ calibrationDuree1)) { //Si somos a la quartacera etapa de la deteccion
11   ↵ en el buen tiempo y que detectamos un frente bajando
12   stepDetection = 0; // alcanzamos la ultima etapa, inicializamos de nuevo
13   ↵ la deteccion
14   Serial.println("on a le deuxieme front descendant : contraction" +
15     ↵ String(valorSensor) + "temps " + String(contador));
16   if (modo >= 1){
17     modo = 0;
18   }
19   else {
20     modo++;
21   }
22   Serial.println(String(modo));
23 }
24 //-----
25 if ((stepDetection == 3) && (contador - T2 > calibrationDuree1)) {
26   // Patern non reconnu -> reset
27   stepDetection = 0;
28   Serial.println("on réinitialise troisieme etape trop longue car" +
29     ↵ String(contador-T2));
30 }
31
32 void controlFuncion(int sV, int mSV) {
33   Serial.println(String(sV));
34
35   if (ultimaValor - umbral1 <= 0 && sV - umbral1 >= 0) { //detectamos
36     ↵ cuando sobre pasamos el umbral
37     cambioEstado = true; // es un cambio de estado
38     sobreLimita += 1; //permite de saber si estamos bloqueado o no
39   }
40   else{
41     cambioEstado = false;
42   }
43   if (cambioEstado){ // Si cambiamos de estado
44     if (sobreLimita % 2 == 0){ //si es par, es que debemos desbloquear
45       bloqueo = false;
46       Serial.println("débloqué");
47       Serial.println(String(sV));
48     }
49   }
50 }
```

```

1     else { //si es impar debemos bloquear
2         bloqueo = true;
3         Serial.println("bloqué");
4         Serial.println(String(sV));
5
6     }
7 }
8 ultimaValor = sV; //Guardamos la ultima valor
9 if (modo == 0) { //primer modo : todos los dedos se contractan
10    velocidadI = 20;
11    velocidadD = 20;
12    if (bloqueo){ //si estamos en un estado bloqueado
13        objetivoAnguloI = 180 - maxAngulo; //Contractamos a los maximos los
14        ↵ dedos de la parte izquierda
15        objetivoAnguloD = maxAngulo; //Contractamos a los maximos los dedos de
16        ↵ la parte derecha
17    }
18    else {
19        objetivoAnguloI = 180 - mSV;//contractamos con una valor que
20        ↵ corresponda a la intensidad de la contractaccion
21        objetivoAnguloD = mSV;//contractamos con una valor que corresponda a
22        ↵ la intensidad de la contractaccion
23    }
24    if (modo == 1) {
25        velocidadI = 20;
26        velocidadD = 20;
27        if (bloqueo){//si estamos en un estado bloqueado
28            objetivoAnguloI = 180 - maxAngulo;//no activamos los dedos de la parte
29            ↵ izquierda
30            objetivoAnguloD = minAngulo; //Contractamos a lo maximum los dedos de
31            ↵ la parte derecha (index y pulgar)
32        }
33    else {
34        objetivoAnguloI = 180 - mSV;//no activamos los dedos de la parte
35        ↵ izquierda
36        objetivoAnguloD = minAngulo; //contractamos con una valor que
37        ↵ corresponda a la intensidad de la contractaccion
38    }
39 }
40 }

```
