





## A LA ACADEMIA DE MECATRÓNICA

TITULACIÓN INTEGRAL MEDIANTE TESIS

### “INTERFAZ EASYVR 3 PARA CONTROL EMBEBIDO DE PRÓTESIS”

PARA OBTENER EL TÍTULO DE:

**INGENIERO EN MECATRÓNICA**

SUSTENTANTE:

**DAVID OSVALDO FRIAS TREVIZO**

**No. CONTROL: 16610192**

DIRECTORES DE TESIS:

**M.C. NANCY IVETTE ARANA DE LAS CASAS**

**DR. DAVID SÁENZ ZAMARRÓN**

**CD. CUAUHTÉMOC, CHIH., MÉXICO. MARZO DE 2021**



## **Agradecimientos y dedicatoria.**

Le agradezco principalmente a Dios por permitirme concluir con mis estudios de la mejor manera.

Le agradezco a mis padres por brindarme su apoyo incondicional, motivarme a seguir adelante, brindarme los recursos económicos y enseñarme a trabajar por mis metas que por más difíciles que parezcan, con trabajo arduo se pueden lograr. A mi hermano por ser parte importante de esta etapa y de mi vida.

Agradezco a mis compañeros por su amistad durante mi estancia en la institución, por el compañerismo y trabajo en equipo para lograr con los objetivos y retos presentados.

Agradezco a todos los docentes que fueron parte importante para mi formación profesional, aportando sus conocimientos y compartiendo sus experiencias laborales a lo largo de cada curso impartido.

Al Instituto Tecnológico de Cd. Cuauhtémoc, por permitirme desarrollarme profesionalmente y poder realizar mi tesis en la institución.

Le agradezco a mis directores de tesis, la MC. Nancy Ivette Arana de las Casas y el Dr. David Sáenz Zamarrón por brindarme su apoyo para realizar el proyecto.

Mil gracias.

## **Resumen.**

En la actualidad, las prótesis robóticas han adquirido gran importancia para la rehabilitación de personas discapacitadas que han padecido la pérdida de alguna extremidad superior permitiéndoles retomar actividades por sí mismas, sin embargo, debido al costo, incluyendo la falta de opciones alternas a un precio accesible y funcionalidad óptima, lo vuelve algo inalcanzable para la mayoría.

Debido a la relevancia del tema, en el presente informe de tesis se plantea un proyecto de investigación el cual consiste en crear una prótesis de mano con tecnología de impresión 3D, para luego integrar un control por reconocimiento de voz con módulo EasyVR 3, con el fin de proponer una alternativa de menor costo y facilitar el manejo al ejecutar movimientos específicos.

El objetivo es utilizar la prótesis de mano controlada por comandos de voz para manipular un ratón de ordenador incluyendo todas las acciones requeridas para interactuar con el software, tales como clic, clic derecho, clic sostenido, sujetar ratón, entre otras acciones.

Para lograrlo se analiza cada una de las partes que conforma la prótesis electromecánica, incluyendo un sistema mecánico articulado por tendones artificiales y un sistema eléctrico para la manipulación de los mismos incorporando servomotores (actuadores) capaces producir el movimiento en ángulos necesario para la flexión de cada uno de los dedos.

Luego se incluyen las acciones para el control de un ratón de ordenador definiendo comandos de voz cortos para facilitar el reconocimiento y adquirir un nivel de respuesta favorable. Se analiza el software del módulo de reconocimiento de voz y el de la placa Arduino para trabajarlos en conjunto e integrar los servomotores a las órdenes del microcontrolador y módulo EasyVR 3.

Enseguida se realizan pruebas y ajustes modificando las partes que pudieran ser optimizadas para un mejor funcionamiento. La observación y análisis son dos factores importantes para que la prótesis de mano pueda ejecutar las acciones asignadas de la mejor manera.

En general el proyecto realizado explora el funcionamiento de una prótesis electromecánica controlada por voz y se integran varias disciplinas de la ingeniería para complementar su funcionamiento al manipular un ratón de ordenador. Los resultados más relevantes, se mencionan enseguida:

- a) Es posible crear una prótesis de mano controlada por reconocimiento de voz y servo actuada que pueda ejecutar acciones para el control de un ratón de ordenador, con la posibilidad de generar una alternativa prometedora a futuro que permita la rehabilitación a personas discapacitadas por amputación transradial a un costo menor.
- b) Al controlar la prótesis por comandos de voz se observa que la recepción de los comandos se muestra beneficiada al estar en un entorno con el menor ruido posible, sin embargo, al ser palabras cortas estas pueden trabajar en un entorno con posibilidad de ser escuchadas aun si existe un factor externo, el cual pueda interferir.
- c) Tomando en cuenta otros tipos de control prostético, el módulo para el reconocimiento de comandos de voz suele ser simple al interactuar con la persona discapacitada ya que solo se requiere de entrenar comandos para detectar y ejecutar la acción, sin importar las condiciones de los músculos del muñón.

## Contenido.

Capítulo 1 . Introducción. ....	1
Capítulo 2 . Justificación. ....	3
Capítulo 3 . Objetivos. ....	5
3.1. Objetivo General. ....	5
3.2. Objetivos Específicos.....	5
3.3. Hipótesis. ....	6
Capítulo 4 . Caracterización del área de participación. ....	7
Capítulo 5 . Planteamiento del problema. ....	8
5.1. Definición del problema. ....	9
Capítulo 6 . Alcances y limitaciones. ....	10
Capítulo 7 . Fundamento teórico. ....	11
7.1. Mano humana. ....	11
7.1.1. Anatomía de la mano. ....	12
7.1.2. Movimientos de la mano. ....	15
7.1.3. Especificaciones físicas de la mano. ....	16
7.2. Amputación y principales causas. ....	17
7.2.1. Niveles de amputación para extremidad superior.....	18
7.2.2. Rehabilitación por amputación de extremidad superior. ....	20
7.3. Prótesis de mano para extremidad superior. ....	20
7.3.1. Antecedentes de prótesis.....	21
7.3.2. Clasificación de prótesis de mano.....	23
7.3.3. Prótesis controlada por reconocimiento de voz. ....	25
7.4. Control de prótesis. Módulo EasyVR 3, microcontrolador y servomotor. ....	26
7.4.1. Módulo EasyVR 3. ....	26
7.4.2. Microcontrolador, Arduino UNO. ....	28
7.4.3. Servomotores. ....	28
7.5. Impresión 3D, Fabricación aditiva.....	29
7.5.1. Impresora 3D. ....	30
7.5.2. Materiales de impresión 3D.....	31

Capítulo 8 . Metodología. ....	32
8.1. Diseño de prototipo prótesis de mano. ....	32
8.2. Recopilación de información e instalación de módulo EasyVR 3. ....	32
8.3. Designar movimientos de operación de prótesis de mano. ....	33
8.4. Análisis de software “EasyVR Commander”. ....	33
8.5. Programación para control de prótesis a través de comandos de voz. ....	33
8.6. Integrar módulo EasyVR 3 a prótesis de mano. ....	33
8.7. Correcciones de Software/Hardware y pruebas finales. ....	33
8.8. Informe de tesis. ....	34
8.9. Metas. ....	34
Capítulo 9 . Resultados. ....	35
9.1. Diseño de prótesis de mano. ....	35
9.1.1. Evidencias de diseño mano alzada. ....	35
9.1.2. Evidencias de diseño asistido por computadora. ....	37
9.1.3. Evidencias de prototipo impreso en 3D. ....	39
9.1.4. Evidencias de diseño eléctrico. ....	40
9.2. Control de prótesis por reconocimiento de voz. ....	42
9.2.1. Acciones asignadas para la prótesis de mano. ....	42
9.2.2. Programación de software EasyVR Commander. ....	43
9.2.3. Modificación de código en software Arduino IDE. ....	47
9.4. Pruebas realizadas. ....	53
9.4.1. Pruebas y correcciones del sistema electromecánico. ....	53
9.4.2. Pruebas finales de respuesta de los comandos de voz. ....	55
9.4.3. Pruebas para manipular el ratón de ordenador. ....	56
Capítulo 10 . Conclusiones y recomendaciones. ....	58
Capítulo 11 . Bibliografía. ....	59
Capítulo 12 . Anexos. ....	61

## Lista de figuras.

Figura 7.1 Cara palmar y dorsal de la mano y dedos.....	11
Figura 7.2 Huesos de la mano y dedos vista posterior.....	12
Figura 7.3 Músculos y tendones de la palma de la mano. ....	14
Figura 7.4 Movimientos del pulgar. ....	15
Figura 7.5 Flexión, extensión abducción y aducción dedos trifalángicos. ....	16
Figura 7.6 Causas de amputación, México 2013. ....	17
Figura 7.7 Tipos de amputación para miembros superiores.....	19
Figura 7.8 "El dedo gordo del Cairo" prótesis de Egipto .....	21
Figura 7.9 Pierna de bronce de Capua, Italia. ....	21
Figura 7.10 Prótesis de madera, Renacimiento. ....	22
Figura 7.11 Prótesis estéticas de mano y dedo.....	23
Figura 7.12 Prótesis: a) Mecánicas) Eléctricas, c) Neumáticas.....	23
Figura 7.13 Módulo EasyVR 3.....	26
Figura 7.14 Especificaciones técnicas EasyVR 3.....	27
Figura 7.15 Arduino UNO, microcontrolador. ....	28
Figura 7.16 Especificaciones técnicas, servomotores MG995. ....	29
Figura 7.17 Etapas de la fabricación aditiva.....	29
Figura 7.18 Impresora MakerBot Replicator +.....	30
Figura 9.1 Tendón artificial para flexión y extensión de cada dedo.....	35
Figura 9.2 Mano y antebrazo, primeras ideas de diseño.....	36
Figura 9.3 Diseño dedos, a)índice, medio, anular y meñique b)pulgar. ....	37
Figura 9.4 Diseño de la palma.....	37
Figura 9.5 Diseño del antebrazo. ....	38
Figura 9.6 Ensamble de diseño de prótesis de mano. ....	38
Figura 9.7 Recopilación fotografías de impresión 3D, prototipo de prótesis. ....	39
Figura 9.8 Prototipo de prótesis de mano armado. ....	39
Figura 9.9 Posición de cada servomotor MG995. ....	40
Figura 9.10 Diseño y pruebas. Servomotores, Arduino y fuente externa. ....	40
Figura 9.11 EasyVR 3 módulo y SHIELD.....	41
Figura 9.12 Prótesis de mano y circuito eléctrico de control. ....	41
Figura 9.13 Entorno principal de EasyVR Commander.....	43
Figura 9.14 Conexión modo PC de microcontrolador y módulo EasyVR 3. ....	43
Figura 9.15 Ubicación grupo Trigger o grupo 0. ....	44
Figura 9.16 Botón "Add Command", añadir comando.....	44
Figura 9.17 Nombrar primer comando, "ENCENDER". ....	44
Figura 9.18 Botón "Train Command", entrenar comando de voz. ....	45
Figura 9.19 Ventana "Command Training", entrenar comando. ....	45
Figura 9.20 Comandos de voz agregados al Grupo 1.....	46
Figura 9.21 Botón "Generate Code", generar código IDE. ....	46

Figura 9.22 Guardar archivo .INO "Código VOICE.....	46
Figura 9.23 Ventana principal de Arduino IDE. ....	47
Figura 9.24 Cargar archivo "CodigoVOICE" a software Arduino IDE. ....	47
Figura 9.25 Variables para los servomotores. ....	48
Figura 9.26 Variables del código. ....	48
Figura 9.27 Pines de salida. ....	48
Figura 9.28 Comando "ENCENDER". ....	49
Figura 9.29 Modificación al comando "SOSTENER". ....	49
Figura 9.30 Modificación de comando "SOLTAR". ....	50
Figura 9.31 Modificación de comando "CLIC". ....	50
Figura 9.32 Modificación de comando "CLICDER". ....	50
Figura 9.33 Modificación de comando "DOBLECLIC". ....	51
Figura 9.34 Modificación de comando "CLICSOSTENIDO". ....	51
Figura 9.35 Modificación de comando "SOLTARCLIC". ....	51
Figura 9.36 Modificación de comando "APAGAR". ....	52
Figura 9.37 Modificación de código para la señal cuadrada de servomotores. ....	52
Figura 9.38 Posición de la articulación para la flexión y extensión. ....	53
Figura 9.39 Reubicación de articulaciones para la flexión y extensión. ....	54
Figura 9.40 Vista general de circuito eléctrico. ....	54
Figura 9.41 Ubicación del micrófono para pruebas de respuesta. ....	55
Figura 9.42 Acciones de la prótesis de mano y comandos ejecutados. ....	55
Figura 9.43 Postura para apagado, encender y soltar con ratón de ordenador. ....	56
Figura 9.44 Postura de prótesis para comandos de voz "2" y "8". ....	56
Figura 9.45 Postura de prótesis para comandos de voz "4", "6" y "7". ....	57
Figura 9.46 Postura de prótesis para comandos de voz "5". ....	57

## Lista de tablas.

Tabla 7.1 Mano, longitudes de falanges y metacarpianos. ....	16
Tabla 7.2 Amputación miembro superior, clasificación topográfica Schwartz. ....	18
Tabla 8.1 Cronograma de actividades para proyecto de investigación. ....	34
Tabla 9.1 Acciones, comandos de voz y ángulos de momento.....	42

## **Glosario de términos.**

- ABS El acrilonitrilo butadieno estireno «ABS» es un polímero amorfo procedente de la emulsión o polimerización en masa de acrilonitrilo y estireno en presencia de polibutadieno. Es resistente a los impactos y presenta dureza entre sus propiedades.
- CAD Computer-Aided Design «CAD; diseño asistido por computadora «DAC»
- CAM Computer-Aided Manufacturing «CAM»; fabricación asistida por computadora «FAC»
- PLA Siglas de poliácido láctico, un polímero termoplástico biodegradable, derivado del maíz utilizado como material para impresión 3d.

## **Capítulo 1 . Introducción**

La mano es uno de los elementos más extraordinarios y complejos del cuerpo humano, ya que se considera un instrumento versátil con gran precisión para interactuar con el entorno. Distintiva anatomía de esta extremidad superior en la que el pulgar y el índice pueden hacer una pinza permite fabricar distintos elementos, además de utilizar diversas herramientas con facilidad.

La pérdida de una extremidad tan fundamental para el desarrollo humano ha ocasionado que durante siglos se busque la elaboración de distintas prótesis que replacen su funcionalidad, pero en los primeros diseños solo se cumplía con capacidades limitadas de control y movimiento, lejanas a ser útiles para actividades que demandan precisión.

En los últimos años el desarrollo en dispositivos prostéticos ha sido prometedor, considerando popular el control por impulsos eléctricos a través de nervios residuales, sin embargo, en ocasiones estos se encuentran dañados por lo cual se buscan alternativas de control como el reconocimiento de voz.

El objetivo del proyecto es crear una prótesis de mano robótica con características ergonómicas y movilidad fundamental para el desempeño de actividades como controlar un ordenador mediante el uso del ratón. Para ello, el principal aspecto a estudiar es el control, por lo que se incorpora un módulo EasyVR 3 capaz de interactuar con el usuario discapacitado al activar comandos de voz específicos.

El capítulo 1 incluye la introducción destacando la importancia de evolución en las prótesis y se describe el objetivo del proyecto. Además, es planteada una breve descripción de la estructura del informe técnico de la tesis.

En el segundo capítulo se redacta la justificación donde se menciona la razón por la que se realiza el proyecto al identificar la necesidad de una prótesis de mano robótica la cual integre un control simple de reconocimiento de voz, capaz de realizar movimientos necesarios para manipular el ratón de un ordenador.

El capítulo 3 contiene el objetivo general y objetivos específicos los cuales buscan dar a conocer los puntos clave a lograr en el proyecto, es decir, el desarrollo de una prótesis de mano incorporando un control por comandos de voz. También la hipótesis es incluida como posible explicación al problema a resolver.

En el capítulo 4 se describe el área de participación para realizar la tesis, en este caso el Departamento de División de Estudios de Posgrado e Investigación dentro del Instituto Tecnológico de Cd. Cuauhtémoc.

Posteriormente, se da a conocer el planteamiento del problema, el cual se encuentra en el capítulo 5 donde se describe la situación de amputaciones en aumento en México, falta prótesis robóticas a un costo accesible y simple control.

El capítulo 6 da a conocer alcances y limitaciones del proyecto en donde se describen los puntos importantes a desarrollar en contraste con los obstáculos.

Dentro del capítulo 7 se plantea el fundamento teórico, incluyendo temas importantes para el diseño de prótesis de mano robóticas y el control por reconocimiento de voz.

Luego, en el capítulo 8 se define la metodología a emplear describiendo el procedimiento a realizar para cumplir con los objetivos del proyecto.

El capítulo 9 muestra los resultados donde se recopilan las evidencias obtenidas a lo largo del proyecto, tales como el armado de la prótesis, diseño del circuito, la implementación del módulo de reconocimiento de voz, programación y pruebas.

En el capítulo 10 se incluye la conclusión del proyecto tomando en cuenta los resultados obtenidos, puntos positivos y negativos, además de recomendaciones que pudieran beneficiar al dar seguimiento a la investigación.

## **Capítulo 2 . Justificación**

En México se considera la discapacidad motriz como el principal tipo de discapacidad padecida, ya sea por accidentes laborales, enfermedades vasculares como diabetes, cáncer, entre otras causas. La presencia de casos por amputaciones de extremidades ha aumentado en la población mexicana. Con ello surge una demanda enorme en prótesis y una falta de abastecimiento considerando diversos factores económicos o la utilidad de acuerdo a las expectativas del usuario.

Esta situación ha limitado a gran parte de personas discapacitadas por amputación en la posibilidad de adquirir una prótesis, con ello, surge gran dificultad para desempeñarse día con día, en especial a aquellas que aún se encuentran en edad productiva y necesitan realizar actividades específicas.

Para la evolución tecnológica que en los últimos años se ha presenciado, una de las actividades más relevantes y necesarias es la interacción con un ordenador, ya sea con fines educativos o laborales, es fundamental para la persona discapacitada contar con una prótesis que le permita ejecutar los movimientos correctos y precisos para cubrir esta necesidad.

De acuerdo a lo anterior mencionado, en este proyecto de investigación se busca crear una prótesis robótica de mano implementada específicamente en amputación transradial, es decir en el área del antebrazo, considerando el control como uno de los elementos más importantes, por lo que en este caso se pretende utilizar un reconocimiento de voz con módulo EasyVR 3 el cual debe ser capaz de ejecutar en el momento indicado los movimientos de la prótesis al interactuar con el ordenador mediante el ratón.

Las razones principales por las que se decidió utilizar un reconocimiento de voz con módulo EasyVR 3 se mencionarán a continuación. En primera estancia es evidente la reducción del costo en comparación a otros tipos de control, lo que permitiría fundamentar una alternativa con la capacidad de contribuir a que más personas discapacitadas por amputación transradial puedan adquirir una prótesis.

Otra de las razones de gran relevancia es la simplicidad de interacción con el usuario discapacitado utilizando comandos simples previamente entrenados y definidos para ejecutar acciones específicas, sin entorpecer el nivel de respuesta por daño en el área amputada como nervios importantes para censar la actividad que dependen de ellos para controlar la extremidad artificial adecuadamente.

En el presente proyecto considerar aspectos como eficiencia, resistencia y capacidad de movimiento como fundamentales para evaluar la capacidad de operación. Además, incluir tecnologías tan populares en la actualidad como impresión 3D permiten reducir considerablemente el costo de elaboración y brindar la capacidad de fabricar mecanismos esenciales a menor tiempo y la posibilidad de mejorar si es necesario.

Una de las características particulares para la elaboración de este proyecto consiste en aplicar conocimiento de distintas áreas como mecánica, programación, electrónica, diseño y modelado asistido por computadora, adquiridos durante el estudio de ingeniería, con el propósito de controlar un prototipo de prótesis de mano utilizando comandos de voz y contribuir en un tema que pretende mejorar la calidad de vida a personas discapacitadas.

## **Capítulo 3 . Objetivos**

### **3.1. Objetivo General.**

El objetivo del proyecto es controlar una prótesis de mano para una amputación transradial en el área del antebrazo, utilizando un microcontrolador y un módulo EasyVR 3, capaz de reconocer comandos específicos de voz que permitan la movilidad de cada uno de los dedos designados para una actividad específica.

Se desea adquirir el control de un ordenador mediante ratón y movimiento de la prótesis de mano, permitiéndole al usuario discapacitado desempeñar actividades necesarias, tales como dar clic derecho, clic izquierdo y sujetar con firmeza el ratón para la movilidad de este.

### **3.2. Objetivos Específicos.**

Para cumplir con el objetivo general, es fundamental cumplir con una serie de objetivos específicos, los cuales son mencionados a continuación:

- a) Recopilar información concreta que definan los principios de diseño y funcionamiento de una prótesis de mano.
- b) Diseñar y construir una prótesis de mano con mecanismos específicos y un sistema mecánico efectivo para transmitir movimiento entre articulaciones.
- c) Recopilar información relevante del módulo de reconocimiento de voz EasyVR 3 y diversos componentes electrónicos a utilizar.
- d) Vincular módulo de reconocimiento de voz y ordenador, para designar comandos de operación que efectúen movilidad en la prótesis de mano.
- e) Brindar movilidad a una prótesis de mano utilizando un microcontrolador, diversos componentes electrónicos, módulos de reconocimiento de voz y fuente de alimentación correcta.

- f) Desarrollar la programación adecuada que cumpla con comandos de voz requeridos para el control de un ratón de ordenador utilizando movimientos específicos de una prótesis de mano.
- g) Establecer una interacción efectiva entre la persona discapacitada y la prótesis de mano mediante comandos de voz claros y concretos.

### **3.3. Hipótesis.**

De acuerdo con los objetivos anteriormente definidos, se ha creado una hipótesis la cual pretende cuestionar si los resultados obtenidos al final del proyecto coinciden con las suposiciones planteadas. Esta se menciona a continuación:

- a) Resultará funcional incorporar un microcontrolador y el reconocimiento de voz con módulo EasyVR 3 para que al usuario discapacitado le sea posible controlar una prótesis de mano al pronunciar comandos de voz previamente entrenados, con el objetivo de ejecutar acciones que permitan controlar el ratón de un ordenador y así interactuar con la interfaz.

#### **Capítulo 4 . Caracterización del área de participación**

Se define como área de participación el departamento de División de Estudios de Posgrado e Investigación en las instalaciones del Instituto Tecnológico de Cd. Cuauhtémoc con el fin de apoyar en el desarrollo de un proyecto de investigación asignado por la M.C. Nancy Ivette Arana De Las Casas y el Dr. David Sáenz Zamarrón.

Para llevar a cabo la tesis se cuenta con un cubículo, el cual cuenta con cuatro ordenadores y acceso a internet de lunes a sábado en horario de 9:00 a 18:00 para realizar el proyecto de investigación en un plazo de cuatro meses y con la finalidad de cumplir con los objetivos propuestos.

Además, es fundamental acceder a el área de diseño en la cual la institución proporciona impresoras 3D para la elaboración del prototipo de prótesis de mano con plástico PLA y con la posibilidad de reimprimir si es necesario.

Con el propósito de elaborar un prototipo de prótesis de mano para posteriormente controlarlo con un módulo EasyVR3 con comandos específicos de voz con fines de investigación y cumplir con ciertos objetivos establecidos, se formulan las siguientes preguntas.

- a) ¿Es posible crear una prótesis de mano con un microcontrolador y módulo EasyVR 3 con capacidad de reconocer comandos específicos para accionar movimientos predeterminados, la cual permitan al usuario discapacitado tomar el control al manipular un ratón para interactuar con el ordenador?
- b) ¿Es una necesidad desarrollar una prótesis de mano robótica a menor costo que, a su vez aumente la posibilidad de rehabilitar y mejorar la calidad de vida a personas que sufren de una amputación transradial?

## Capítulo 5 . Planteamiento del problema

El proyecto a realizar se enfoca principalmente en el control por comandos de voz de una prótesis de mano aplicada en una amputación transradial en área de antebrazo, la cual busca desempeñarse en la manipulación de un ratón de ordenador. Se consideró llevarlo a cabo, luego de observar problemas en la productividad y desempeño de personas discapacitadas que necesitan controlar una prótesis para diversos fines, sean educativos o laborales principalmente.

Durante el desarrollo del proyecto se busca atender 3 problemas principales, de los cuales se propone resolver de manera prioritaria.

- a) Dificultades para el control. Ya que la mayoría de los dispositivos prostéticos se basan en impulsos eléctricos de nervios, que son propensos a estar dañados en casos particulares, en este proyecto se plantea un control por comandos de voz facilitando el entendimiento y simplicidad de interacción con el usuario.
- b) Limitaciones en el diseño. Los grados de libertad suficientes y la ejecución de posiciones específicas son limitadas de acuerdo al costo de las prótesis, es por ello que se propone crear un diseño que cumpla con las características de flexión y extensión de cada dedo para manipular el ratón al interactuar con el ordenador.
- c) Alto costo. Es uno de los aspectos más comunes por los que no es posible adquirir una prótesis robótica para rehabilitación, en este caso al usar elementos de menor costo, pero que cumplan adecuadamente sus funciones se podrá cubrir esta necesidad volviéndola más accesible.

## **5.1. Definición del problema.**

La cantidad de amputaciones reportadas en los últimos años en México es preocupante, solo lo que respecta en el año 2010, INEGI considero 785 000 personas carentes de una o varias extremidades, sin embargo, en 2014 diversas fuentes indicaron que la cantidad de amputados fue alrededor de 935 000.

Ya sea por causas traumáticas, degenerativas o de origen congénito, la discapacidad motriz es una de las principales en el país, por lo que en consecuencia disminuye la calidad de vida y productividad de la persona discapacitada, en este caso se considerando específicamente a aquellos con amputación transradial de miembro superior.

El sistema médico en el país no incluye dentro de los planes de salud, la adaptación de una prótesis robótica a una persona que haya perdido un miembro; ya que, respecto a las altas tasas de personas con alguna amputación y altos costos de una prótesis, se considera poco factible atender esta necesidad, por lo que solo existe la posibilidad de optar por un gancho o símil que no es funcional para realizar actividades que demandan precisión.

Sin embargo, es de gran dificultad adquirir una prótesis robótica con la capacidad para desempeñar tareas específicas, es decir, actividades como interactuar con un ordenador utilizando el ratón para desenvolverse en el ámbito laboral o educativo que permita lograr una reinserción social productiva, es una situación limitada tomando en cuenta el costo elevado y la capacitación que esto conlleva para ser usada adecuadamente.

La Dra. Martha Hajar Medina, Secretaria Técnica del Consejo Nacional para la Prevención de Accidentes, estima que: “sólo 1 de cada 10 personas con miembros amputados se rehabilita, y sólo el 30% de los rehabilitados saben usar una prótesis”. Lo cual define dos problemas importantes; en primera la rehabilitación no se lleva a cabo debido a los altos costos de una prótesis robótica funcional, sin embargo, en caso de ser adquirida el problema se presenta en la complejidad de control ante las condiciones del muñón, entre otras características específicas.

## **Capítulo 6 . Alcances y limitaciones**

Los alcances del proyecto se definen en la creación de una prótesis de mano aplicada en la amputación transradial de miembro superior en el área del antebrazo, considerando principalmente aspectos ergonómicos, funcionalidad y desempeño al interactuar con un ordenador mediante el uso del ratón.

Uno de los temas más destacables en el proyecto es la interacción eficiente entre el usuario discapacitado y la prótesis de mano, por ello se ha considerado utilizar un módulo EasyVR 3 capaz de reconocer comandos de voz que controlen directamente los movimientos en conjunto como sujeción y clics que el ratón necesita para interactuar con el ordenador.

Sin embargo, las limitaciones como el tiempo y falta de disposición de ciertos recursos son evidentes ante el desarrollo del proyecto, debido a la situación actual en el mundo ante la emergencia sanitaria generada por el virus SARS-CoV2 (COVID-19), por lo que se busca atender solo las partes esenciales para que la prótesis de mano cumpla con actividades prioritarias que le permitan ejecutar movimientos específicos.

Otro aspecto que limita a el proyecto es el desempeño de sistemas de reconocimiento de voz respecto a un entorno sonoro, considerando que entre más elementos interfieren en la recepción de las ordenes, menor será la capacidad de accionar la prótesis. Una posible solución sería disponer de comandos previamente grabados por un dispositivo externo y cercano al micrófono.

## Capítulo 7 . Fundamento teórico

Con el propósito de explorar conceptos fundamentales para el desarrollo del presente proyecto de investigación, se realizó una recopilación de información con temas de interés para la elaboración de una prótesis electromecánica controlada por reconocimiento de voz.

### 7.1. Mano humana.

La mano es la parte del cuerpo humano unida a la extremidad del antebrazo y que comprende desde la muñeca inclusive hasta la punta de los dedos (RAE, Definición de mano, 2021). Es el órgano compuesto por cara palmar y dorsal dedicado a la prensión, ya que por su anatomía dispone de una posición particular del pulgar que se opone a los demás dedos (fig. 7.1). Se considera una de las mejores herramientas del ser humano para ejecutar innumerables acciones con precisión, fuerza y agilidad.



**Figura 7.1 Cara palmar y dorsal de la mano y dedos.**

Fuente: Anatomía humana, Ruiz Liard, Latarjet 2005.

Diversas características fisiológicas de la mano permiten adoptar posiciones precisas para desempeñar diversas acciones, además de la recepción sensorial que proporciona datos imprescindibles que influyen en la toma de decisiones al trabajar en conjunto con el cerebro. A continuación, se describe la composición de la extremidad superior a estudiar y movilidad respectivamente.

### 7.1.1. Anatomía de la mano.

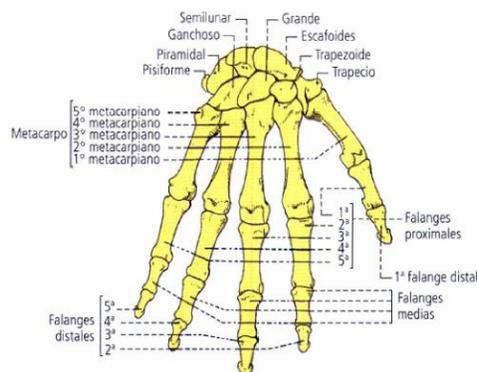
La anatomía de la mano se subdivide en tres partes; muñeca, metacarpo y dedos. Los cinco dedos se componen de pulgar ubicado en forma lateral y cuatro dedos mediales a éste, llamados índice, medio, anular y meñique. En reposo se posicionan de forma de arcada flexionada, donde el meñique es el más flexionado y el índice es el que menor flexión presenta.

Las superficies se identifican como anterior ubicado en la palma y dorsal considerando el dorso de la mano. La movilidad se define un eje longitudinal ubicado en el dedo medio para evaluar la aducción al mover en dirección al eje de referencia y abducción para movimiento contrario a este. Con lo que respecta al pulgar en una posición de 90° al eje longitudinal para un movimiento perpendicular en relación a los demás dedos. (Richard L. Drake, 2015).

A continuación, se definen las características anatómicas específicas del mano clasificado en huesos, articulaciones, músculos y tendones de la extremidad.

#### 7.1.1.1. Huesos y articulaciones.

El esqueleto de la mano está compuesto por veintisiete huesos repartidos en tres grupos: el carpo, el metacarpo y los dedos. A continuación, se describen a detalle y se representan en la figura 7.2. (Ruiz Liar Alfredo, 2011)



**Figura 7.2 Huesos de la mano y dedos vista posterior.**

Fuente: Anatomía humana, Ruiz Liard, Latarjet 2005.

El carpo comprende 8 huesos en dos filas transversales, superior e inferior. La fila superior (proximal o primera fila) se compone de huesos escafoides, semilunar, piramidal y pisiforme, mientras que la fila inferior (distal o segunda fila) comprende de lateral a medial, los huesos trapecio, trapecoide, grande y ganchoso.

Los huesos del metacarpo forman el esqueleto de la palma y dorso de la mano. Son 5 huesos llamados metacarpianos, articulados de manera proximal con los huesos de la segunda fila del carpo y denominados de primero a quinto metacarpiano del pulgar hasta el meñique.

Los huesos de las falanges (dedos) están distribuidos de manera independiente, articulados con los metacarpianos y se componen de tres falanges llamados proximal, media y distal posicionados en ese orden a excepción del pulgar, el cual solo cuenta con dos falanges, proximal y distal.

Las articulaciones de la extremidad superior se dividen en radiocarpiana, intercarpianas, carpometacarpianas, metacarpofalángicas y las interfalángicas. Se realiza una breve descripción a continuación (Ruiz Liar Alfredo, 2011).

La articulación radiocarpiana ubicada en la muñeca permite el movimiento de la mano en dos ejes, tales como abducir, aducir, flexionar y extender. Luego en las articulaciones intercarpianas se localizan entre los huesos del carpo. El movimiento es limitado, pero contribuyen para el desplazamiento de la mano.

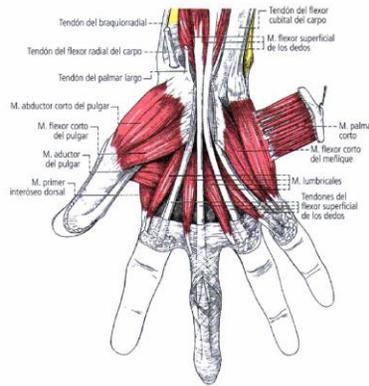
Entre los huesos metacarpianos y fila distal del hueso del carpo se localizan cinco articulaciones carpometacarpianas, numeradas de primera a quinta en orden desde pulgar hasta el meñique. La articulación carpometacarpiana del pulgar suele ser más flexible que la de los cuatro dedos restantes.

Las articulaciones metacarpo-falángicas están entre las cabezas distales de los metacarpianos y falanges proximales de los dedos, para la flexión, extensión, abducción, aducción y rotación. Por último, las articulaciones interfalángicas de la mano son de tipo bisagra, reforzadas por los ligamentos colaterales medial y lateral, así como ligamentos palmares para movimientos de flexión y extensión.

### 7.1.1.2. Músculos y tendones.

La mano cuenta con 19 músculos divididos de la siguiente manera y señalados en la figura 7.3. (Ruiz Liar Alfredo, 2011) .

- a) Músculos de eminencia tenar compuesta por el abductor corto, el oponente, el flexor corto y el aductor del pulgar.
- b) Músculos de eminencia hipotenar comprendiendo el palmar corto, el aductor, el flexor corto y el oponente del meñique.
- c) Músculos de la parte media de la palma formada lumbricales. Músculos interóseos compuesto por los palmares y dorsales.



**Figura 7.3 Músculos y tendones de la palma de la mano.**

Fuente: Anatomía humana, Ruiz Liard, Latarjet 2011.

Los músculos y tendones de la mano trabajan en conjunto con el antebrazo, definido por tres regiones importantes; el compartimiento antebraquial anterior, compartimiento antebraquial posterior y porción lateral (anexo A). (Ruiz Liar Alfredo, 2011)

El compartimiento antebraquial anterior, establece 8 músculos divididos en 4 planos, desde el superficial hasta el profundo. Para el compartimiento ante braquial posterior se definen dos partes, es decir, 8 músculos en dos capas (superficial y profunda) y lateral. Y la porción lateral se ubican cuatro músculos, el braquiorradial, el extensor radial largo del carpo, el extensor corto del carpo y el supinador, ordenados desde superficial a profundo respectivamente.

### 7.1.2. Movimientos de la mano.

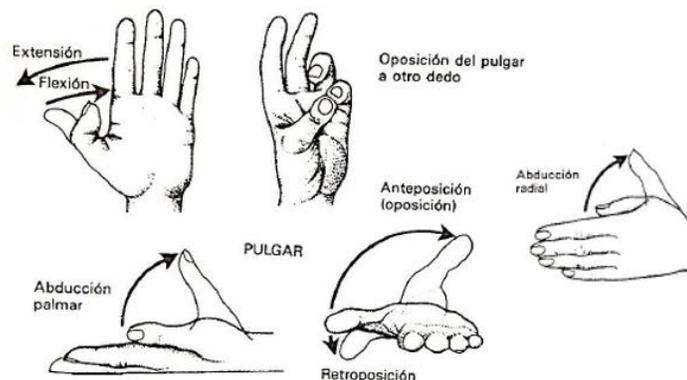
La mano es susceptible a realizar movimientos de flexión, extensión y de inclinaciones laterales (cubital y radial), con respecto al antebrazo, con la habilidad de ejecutarlos sucesivamente. (Rouvière H. Delmas, 2005)

La flexión de la mano con respecto a la muñeca está definida al dirigir la palma hacia la cara anterior del antebrazo. El movimiento inverso se le denomina extensión. La movilidad delimita 180° entre distancias máximas de cada acción.

Las inclinaciones laterales (cubital y radial) de la muñeca se efectúan alrededor del eje anteroposterior común, que pasa por el centro de la cabeza del hueso grande. Los movimientos están limitados por tensión de los ligamentos colaterales.

Los movimientos para los 5 dedos se definen enseguida. El primero es el pulgar, el cual cuenta con una independencia y movilidad única, ante la flexión y extensión se posiciona el primer metacarpiano hacia la palma, o hacia el dorso de la mano.

La abducción radial y palmar lo separa de los otros metacarpianos, mientras que la aducción es la acción contraria. Respecto a la oposición, ocurre al girar la cara palmar del pulgar hacia la cara palmar de los otros dedos, asociando la aducción con la flexión. Se muestra en la figura 7.4 los movimientos previamente descritos.

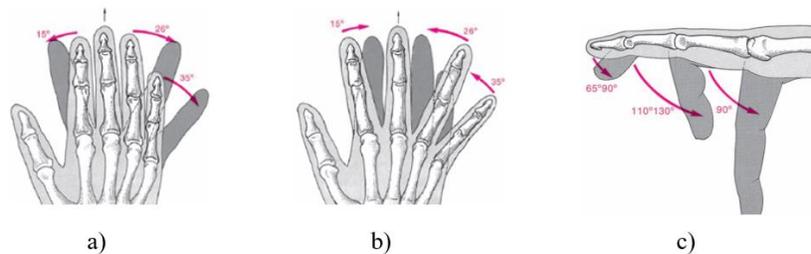


**Figura 7.4 Movimientos del pulgar.**

Fuente: La mano. Sociedad Americana de Cirugía, Churchill Livingstone, s.f.

Con respecto a los 4 dedos: índice, medio, anular y meñique, se observa la flexión y extensión alrededor de un eje transversal. El eje de la mano y de los dedos pasa por el tercer hueso metacarpiano y por el dedo medio.

La abducción de los dedos (fig. 7.5 a) se aleja del eje, mientras que la aducción (fig. 7.5 b) los aproxima. La amplitud del movimiento es mayor cuando los dedos se hallan en extensión. El movimiento de abducción alcanza 60° para el dedo índice y 45° para los dedos anular y meñique. La falange proximal del dedo se encuentra inicialmente en extensión, en la prolongación del hueso metacarpiano. Cuando se produce una extensión provocada y pasiva, la falange se sitúa en un plano posterior a su posición original (fig. 7.5 c). (Rouvière H. Delmas, 2005).



**Figura 7.5 Flexión, extensión abducción y aducción dedos trifalángicos.**

Fuente: Anatomía humana, descriptiva topográfica y funcional, 2005.

### 7.1.3. Especificaciones físicas de la mano.

Para elaborar un modelo de prótesis de mano es necesario conocer las medidas antropométricas de dicha extremidad, por lo tanto, se muestra la siguiente tabla con las longitudes estándar de falanges y metacarpianos.

**Tabla 7.1 Mano, longitudes de falanges y metacarpianos.**

Dedo	Falange Distal (mm)	Falange Media (mm)	Falange Proximal (mm)	Metacarpo (mm)
Pulgar	21.67 ± 1.60	-	31.57 ± 3.13	46.22 ± 3.94
Índice	15.82 ± 2.26	22.38 ± 2.51	39.78 ± 4.94	68.12 ± 6.27
Corazón	17.40 ± 1.85	26.33 ± 3.00	44.63 ± 3.81	64.60 ± 5.38
Angular	17.30 ± 2.22	25.65 ± 3.29	41.37 ± 3.87	58.00 ± 5.06
Meñique	15.96 ± 2.45	18.11 ± 2.54	32.74 ± 2.77	53.69 ± 4.36

Fuente: (Sánchez Calderón, 2016).

## 7.2. Amputación y principales causas.

Amputación se considera del latín “amputatio”, se define como la separación de un miembro o parte del mismo, de una extremidad del cuerpo; específicamente se conoce como la operación quirúrgica de cortar circularmente un miembro por continuidad del hueso o huesos extirpando una parte o su totalidad. (Vv, 2009)

La amputación es un acontecimiento vital que lleva implícito una incapacidad física al remover una extremidad. Sin embargo, aunque su propósito es curativo, supone una agresión a la integridad física y constituye una nueva situación a la que el amputado debe adaptarse física y emocionalmente. (Periago, 2009) Para su estudio, la Secretaria de Salud de México en 2013 realizó una clasificación de tipo de amputación, frecuencia y causas principales respecto a el sistema automatizado de egresos hospitalarios y procedimientos médicos (fig. 7.6).



**Figura 7.6 Causas de amputación, México 2013.**

Fuente: (Vela Sánchez, 2016)

Las principales causas de una amputación son congénita, traumática y vascular. En México las causas vasculares como diabetes, son las que generan el mayor número de amputaciones. El patrón descrito por la Secretaria de Salud se observa un incremento a partir de los 40 años de edad, sin embargo, en los grupos de entre los 50 y 64 años de edad, ocurre el mayor número de casos.

Las amputaciones traumáticas son producidas por accidentes generalmente en un entorno laboral, por lo que afecta con mayor frecuencia a la población en edad productiva como jóvenes entre 15 y 24 años de edad y del sexo masculino los que presentan el mayor número de casos en razón de 5:1.

Los casos de amputación respecto a problemas degenerativos, como cáncer, presentan una distribución diferenciada según grupo de edad y sexo, con un incremento en grupos de 40 a 64 años de edad y predominio del sexo masculino. Mientras que las amputaciones de origen congénito se concentran en los menores de 5 años de edad con mínimas diferencias por sexo (Vela Sánchez, 2016).

### 7.2.1. Niveles de amputación para extremidad superior.

El nivel de amputación es fundamental para obtener un muñón útil y adoptar una prótesis. Se considera la amputación con mayor eficacia en el área de menor masa existente y la longitud más adecuada de un muñón es aquella que conserva mejor la comodidad, la función y la estética. (Oxolón Salvador, 2018)

La amputación de miembro superior se clasifica de acuerdo a la tabla topográfica Schwartz (tabla 7.2). Al final se observa cada tipo en la figura 7.7.

**Tabla 7.2 Amputación miembro superior, clasificación topográfica Schwartz.**

MIEMBROS SUPERIORES
Interescapulotorácico
Desarticulación de Hombro
Amputación por arriba de codo
Desarticulación de codo
Amputación por abajo de codo
Desarticulación de muñeca
Amputaciones parciales de mano

Fuente: (Oxolón Salvador, 2018)

La amputación interescapulotorácica o cintura escapular, es la de menor frecuencia dentro de las amputaciones de miembro torácico. Debido a la extirpación de la clavícula, omóplato, húmero, es totalmente antiestético y presenta graves problemas funcionales. El muñón es prácticamente fijo.

La desarticulación de hombro mantiene la pinza omo-clavicular. La movilidad es mínima solo con movimientos escapulares, con ello se facilita el uso de un sistema protésico, pero con posibles dificultades de adaptación. Luego, la amputación por arriba de codo, mantiene una ligera libertad de movimiento a nivel escapulo humeral, sobre todo para la flexo-extensión y la abducción de brazo, también es posible realizar ligeros movimientos de rotación.

Para la desarticulación de codo se considera una amputación de poca utilidad. Se realizan con mayor amplitud los movimientos de rotación. Conserva, por tanto, la longitud total del humero. La amputación por abajo de codo dependiendo del nivel, será la funcionalidad que el muñón adquiera. A niveles determinados se permitirán movimientos de flexo extensión y prona supinación.

La desarticulación de muñeca se establece como una posibilidad muy importante, ya que se conserva el porcentaje de función del brazo y el antebrazo, por permitir los movimientos de codo y la pronosupinación del antebrazo en todo su arco. Las amputaciones parciales se dividen en Transcarpiana, Transmetacarpiana y parcial de dedos. La Transcarpiana mantiene la función total del antebrazo. La Transmetacarpiana con un muñón antiestético, pero permite tener adecuada función del antebrazo. La amputación parcial de la mano comprende la presencia de por lo menos uno de los dedos o la amputación parcial de cualquiera de ellos.



**Figura 7.7 Tipos de amputación para miembros superiores.**

Fuente: (Londoño, 2013)

### **7.2.2. Rehabilitación por amputación de extremidad superior.**

La amputación de una extremidad superior tiene como consecuencia una discapacidad funcional y de índole psicológico que la rehabilitación protésica trata de disminuir. Para la elección de una prótesis de miembro superior es muy importante llevar a cabo una valoración con las expectativas de la persona discapacitada, ya que es esencial para determinar el tipo de prótesis.

Los resultados de la evaluación para la rehabilitación dependen, en primer lugar, del nivel o de la extensión de la amputación y, en segundo lugar, del tipo de prótesis elegida, ya sea una prótesis de tipo funcional o una prótesis estética (Médico-Quirúrgica, 2007).

Todas las etapas para la rehabilitación deben ser controladas desde la etapa de cicatrización cutánea comprendido desde día 0 al día 30, etapa pre protésica del día 30 al día 45, etapa de adaptación de la prótesis del día 45 al día 75, etapa de instalación definitiva del día 75 al día 90, renovación cada 5 años.

Para que el usuario con amputación elija el uso de prótesis para su rehabilitación debe recibir una información objetiva, neutra y documentada que le permita una elección adaptada y considerando que el apoyo psicológico en esta etapa es primordial (Periago, 2009).

### **7.3. Prótesis de mano para extremidad superior.**

Una prótesis es una pieza o aparato empleados para sustituir un órgano o un miembro del cuerpo (RAE, Definición de prótesis, 2021). Es un elemento artificial creado con el fin de mejorar o reemplazar una función, una parte o un miembro completo del cuerpo humano afectado, en este caso una extremidad superior, por lo que, una prótesis para la persona discapacitada colabora directamente en el desarrollo psicológico del mismo, creando una percepción de totalidad al recobrar movilidad y adquirir mejoras en su condición estética.

### 7.3.1. Antecedentes de prótesis.

La historia del desarrollo protésico cuenta con múltiples antecedentes, desde comienzos primitivos, las propuestas actuales y proyecciones futuras. A continuación, se mencionan los sucesos más importantes.

La civilización del antiguo Egipto es pionera en el desarrollo de prótesis. Elaboraban sus extremidades protésicas rudimentarias con fibras. Recientemente científicos descubrieron en una momia egipcia el primer dedo del pie protésico, que parece haber sido funcional. Se observa en la figura 7.8. (Norton, 2007).



**Figura 7.8 "El dedo gordo del Cairo" prótesis de Egipto**

Fuente: (Lisandro Puglisi, s.f.)

Comprendiendo el año 424 a. C. al 1 a. C. numerosos descubrimientos se han revelado. En Italia, una pierna artificial que data de aproximadamente 300 a. C. elaborada con hierro y bronce, y un núcleo de madera, perteneciente a un amputado por debajo de la rodilla (fig. 7.9). También, un general romano de la Segunda Guerra Púnica (218-210 a. C.) a quien le amputaron el brazo derecho, se le colocó una mano de hierro para que sostuviera el escudo. (Norton, 2007)



**Figura 7.9 Pierna de bronce de Capua, Italia.**

Fuente: (Lisandro Puglisi, s.f.)

Con respecto a la Alta Edad Media (476 a 1000) se establecieron pocos avances como el gancho y la pata de palo. Pero en la época del Renacimiento, surgieron nuevas perspectivas para la medicina. Se retomaron descubrimientos médicos relacionados con la protésica de griegos y romanos lo que produjo un renacer en prótesis elaboradas con hierro, acero, cobre y madera (fig. 7.10) (Norton, 2007).



**Figura 7.10 Prótesis de madera, Renacimiento.**

Fuente: (Norton, 2007)

En los siglos XVII al XIX, se presentaron avances en prótesis de miembro inferior, donde se intentó incorporar la articulación de rodilla de acero y pie, desde una articulación controlada por cuerdas que cumplían la función de tendones (Pierna de Anglesey) . Luego surgieron prótesis más avanzadas, con encaje de succión, una rodilla policéntrica y un pie multi articulado, de menor peso. (Norton, 2007)

En el siglo XX se implementó el aluminio con alternativa en el desarrollo de prótesis (1912), lo que aligero el peso considerablemente. Además, conflictos como la guerra civil, la primera y segunda guerra mundial, hicieron que los casos de amputación aumentaran, lo que obligo a países como Estados Unidos a especializarse en el área de la protésica creando varios organismos dedicados a mejorar el funcionamiento y producción de prótesis modernas (Norton, 2007).

En la actualidad las prótesis se caracterizan por ser elaboradas principalmente con plásticos, aluminio y materiales compuesto para dispositivos más funcionales. Además, se consideran dispositivos hechos a la medida del paciente, el advenimiento de los microprocesadores, los chips informáticos y la robótica en los dispositivos actuales han permitido que los amputados recuperen su estilo de vida, en lugar de solo proporcionar una funcionalidad básica (Norton, 2007).

### 7.3.2. Clasificación de prótesis de mano.

En el siguiente apartado se describen los tipos de prótesis de extremidad superior específicamente en la mano, de acuerdo a características funcionales definidas por diferentes tecnologías y conocimiento.

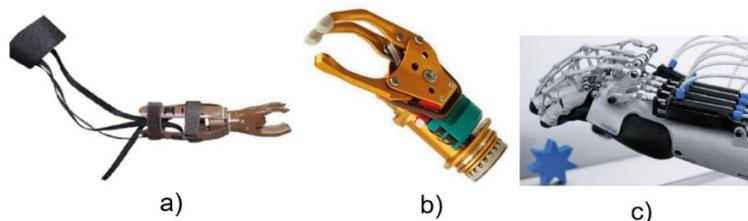
Las prótesis estéticas se clasifican también como prótesis pasivas. Sirven para restablecer el aspecto exterior y satisfacen las exigencias del paciente con respecto a su composición, peso, confort y aspecto. Los materiales más son PVC rígido, látex flexible o silicona. Estos materiales son más livianos y requieren de poco mantenimiento. Se observan en la figura 7.11. (Rivera, 2018).



**Figura 7.11 Prótesis estéticas de mano y dedo.**

Fuente: (Rivera, 2018)

Sin embargo, existen prótesis que proveen de cierta movilidad para ser más funcionales con el propósito de suplir algunas acciones del miembro natural amputado. La clasificación prostética depende de su tecnología, es decir, mecánicas, eléctricas, neumáticas (fig. 7.12), hasta mioeléctricas e híbridas.



**Figura 7.12 Prótesis: a) Mecánicas, b) Eléctricas, c) Neumáticas.**

Fuente: (Rivera, 2018)

Las prótesis mecánicas (fig. 7.12 a) ejecutan movimientos básicos como apertura y cerrado de la mano, limitando el agarre de objetos grandes y movimientos imprecisos. La señal mecánica para el control se obtiene de otro miembro del cuerpo como codo o el hombro. Se implementa un arnés colocado en la espalda el cual generará la movilidad de la prótesis a través de una liga. (Jofre L. Brito, 2013)

Se consideran prótesis eléctricas (fig. 7.12 b) a aquellas que usan como fuente de energía implementos electrónicos como motores eléctricos en el dispositivo terminal, muñeca o codo con una batería para su alimentación. Entre los tipos de control se observa el uso de servomotores, botón pulsador o botón con interruptor de arnés. Sin embargo, cuenta con desventajas como que son costosas, el peso es considerable y es necesario de cuidado ante un medio húmedo (Rivera, 2018).

Las prótesis neumáticas (fig. 7.12 c prótesis Festo) comprenden a dispositivos que utilizan aire comprimido como fuente energética para ejecutar movimiento. Se implementan actuadores por músculos neumático. Son de gran rapidez y brindan mayor fuerza, pero el costo es alto respecto al mantenimiento (Rivera, 2018).

Las prótesis mioeléctricas son en la actualidad las de mayor aplicación debido a su grado de precisión, fuerza y distintiva estética. El medio de control se basa en señales musculares, las mismas que son obtenidas mediante el uso de electrodos que permiten la extracción de la señal que es amplificada, procesada y filtrada para el control de la prótesis (Jofre L. Brito, 2013).

Las prótesis híbridas es una combinación entre la acción del cuerpo con el accionamiento por electricidad. Este concepto mayormente utilizado en las prótesis transmurales (amputación por encima del codo), donde por lo general el codo es accionado por el cuerpo y el dispositivo terminal (gancho o mano) es de accionamiento mioeléctrico (Jofre L. Brito, 2013).

### **7.3.3. Prótesis controlada por reconocimiento de voz.**

La gran diversidad en el control de una prótesis de mano ha permitido la rehabilitación a gran cantidad de personas con alguna amputación para una reinserción social efectiva. El uso de impulsos eléctricos y nervios residuales para el control hoy en día es de las tecnologías más implementadas, sin embargo, al encontrarse considerablemente dañado el muñón no hay forma de adaptabilidad de los electrodos que censan el movimiento.

De acuerdo al panorama descrito, se proponen distintas alternativas de control, por lo que para esta investigación se considera accionar los mecanismos y electrónica de una prótesis implementando un reconocimiento de voz, el cual suele ser menos invasivo y fácil de usar para el usuario discapacitado.

Un reconocimiento de voz es la técnica que permite almacenar información de manera digital de una palabra o frase pronunciada analógicamente. Es el conjunto de aplicaciones que permiten interactuar con un agente externo para proporcionar ordenes de manera directa (Villacis, 2008).

Para que una prótesis pueda interactuar con un reconocimiento de voz, es importante considerar convertir una señal de voz humana hacia una señal digital, para ello se debe convertir la señal acústica emitida a una señal eléctrica al captarla por un micrófono y se aplica un periodo de muestreo considerando la codificación de la señal para evitar distorsión. (Vinicio D. Méndez, 2015)

La señal digital emitida requiere de una alimentación de carga como baterías esenciales al comunicarse con la etapa de potencia la cual se compone por servomotores articulados a la prótesis y sistema mecánico, para producir el control y ejecución de movimientos según sea programado por el microcontrolador.

Un proyecto de la UNAM dedicado a el diseño de mecanismos de voz para controlar una prótesis (2013), menciona que “el uso de reconocimiento de voz podría reducir el costo a una quinta parte, lo que permitiría ponerla al alcance de más personas”. Es decir, la rehabilitación prostética sería más accesible.

## 7.4. Control de prótesis. Módulo EasyVR 3, microcontrolador y servomotor.

### 7.4.1. Módulo EasyVR 3.

El módulo de reconocimiento de voz EasyVR 3 (fig. 7.13), es capaz de reconocer comandos específicos de voz para distintos propósitos brindando versatilidad en sus aplicaciones. Es posible utilizarlo con cualquier host con una interfaz UART alimentada a 3.3V - 5V, como placas PIC y Arduino (Vear, 2016).



**Figura 7.13 Módulo EasyVR 3**

Fuente: (Vear, 2016)

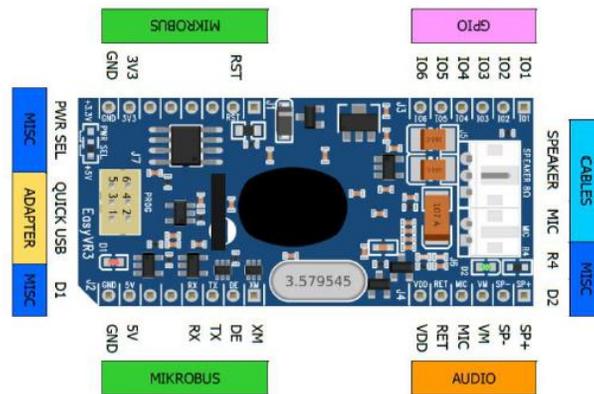
Las características principales del módulo EasyVR 3 son las siguientes:

- a) 28 vocabularios de comandos personalizados de Speaker Independen (SI).
- b) 64 comandos definidos por el usuario dependientes del altavoz (SD) o de verificación del altavoz (SV), divididos en un máximo de 16 grupos.
- c) Hasta alrededor de 21 minutos de sonidos o habla pregrabados.
- d) 100 segundos de grabación y reproducción de mensajes en vivo.
- e) Capacidad de sincronización de labios en tiempo real.
- f) Salida de audio diferencial que admite directamente altavoces de 8  $\Omega$ .
- g) Interfaz gráfica de usuario fácil para programar comandos de voz y audio.
- h) 6 líneas E/S de uso general que se pueden controlar por comandos UART.

El módulo Easy VR 3 cuenta con un Shield 3 el cual proporciona encabezados de orificios pasantes separados para conectar el Shield a las placas Arduino y otras placas basadas en microcontroladores que adoptan una interfaz compatible con Arduino. Para más características del Shield 3 consultar anexo B. (Vear, 2016)

Las especificaciones técnicas se describen por encabezados (fig. 7.14). En los encabezados J1 y J2 están conectores de interfaz mikroBus, que proporcionan de 3.3V/ 5V, al módulo y líneas de I/O digitales traducidas por voltaje, que incluyen: líneas de recepción / transmisión UART y pines de control. En el encabezado J3 se incluyen líneas de expansión de I/O configurables (entradas con pull-up interno débil por defecto), alimentadas con el voltaje lógico interno VDD.

El encabezado J4 proporciona las principales señales analógicas, como la de micrófono y salidas DAC amplificadas, también disponibles en conectores internos de ángulo recto J5 y J6. El conector de programación está en J7 (Vear, 2016).



**Figura 7.14 Especificaciones técnicas EasyVR 3.**

Fuente: (Vear, 2016)

Para el uso del módulo EasyVR 3 es necesario realizar los siguientes pasos:

- Conectar el micrófono a la toma de 2 pines MIC (J6).
- Conectar el altavoz de 8 ohm a la toma de 3 pines SPEAKER (J5).
- Conectar el cable QuickUSB a la toma de 3X2 pines (J7).
- Conectar el extremo del USB al PC.
- Si la instalación es exitosa, se observará un puerto COM virtual (COM3) en administrador de dispositivos.
- Iniciar Software EasyVR Commander, elegir puerto COM y clic en conectar.

Las características técnicas de EasyVR 3 se observan en anexo C del documento.

### 7.4.2. Microcontrolador, Arduino UNO.

Arduino ofrece ventajas como ser libre, extensible, con entorno de programación gratuito y multiplataforma, además es indispensable para trabajar con el módulo de reconocimiento de voz al conectarlo con el Shield 3 del módulo EasyVR 3.

Se considera el uso de Arduino UNO (fig. 7.15) el cual se describe como una placa de microcontrolador basada en ATmega328P. Cuenta con 14 pines de entrada/salida digital (6 pines se pueden usar como salidas PWM), 6 entradas analógicas, un resonador cerámico de 16 MHz, una conexión USB, un conector de alimentación, un encabezado ICSP y un botón de reinicio (Arduino, 2021).



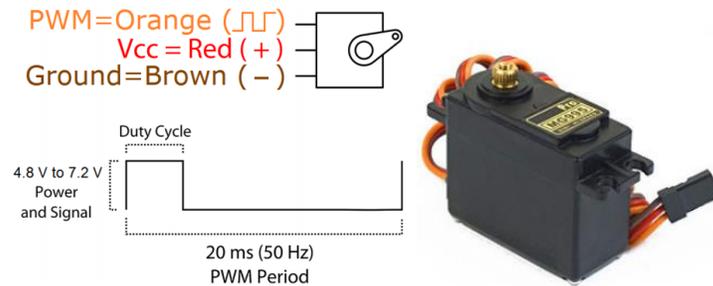
**Figura 7.15** Arduino UNO, microcontrolador.

Fuente: (Arduino, 2021)

### 7.4.3. Servomotores.

Los servomotores son dispositivos con la capacidad de ser controlados, tanto en velocidad como en su posición del eje principal dentro de un rango de operación y mantenerse estable. Se compone por un motor, una caja reductora y un circuito de control que pueden operarse por una señal codificada, con un peso reducido, además brindan fiabilidad, estabilidad y versatilidad a baja tensión. (Andres, 2020)

El servomotor MG995 brindan alta velocidad y se puede usar cualquier código, hardware o biblioteca para controlarlos. Sus longitudes y peso son ideales para una posición compacta. se destaca por su torque (11 Kg), engranajes metálicos y gran robustez. Especificaciones técnicas en fig. 7.16 (Electronicos Caldas, s.f.).



**Figura 7.16 Especificaciones técnicas, servomotores MG995.**

Fuente: (Electronicos Caldas, s.f.)

### 7.5. Impresión 3D, Fabricación aditiva.

La impresión 3D es el proceso mediante el cual se crea un objeto físico sobre la base de un modelo digital. El objeto digital que se ha de imprimir se deconstruye utilizando programas laminadores y se reconstruye mediante una impresora 3D en un objeto físico. El nombre técnico que engloba la tecnología de impresión 3D es fabricación aditiva. (Educar Sociedad del Estado, s.f.)



**Figura 7.17 Etapas de la fabricación aditiva.**

Fuente: (Nemcansky, 2017)

La fabricación aditiva se divide en 3 etapas: modelo digital en archivo CAD y la fabricación desde la división de capas hasta la construcción de cada una (CAM) (fig. 7.17). Esta tecnología comprende todos los grupos de procesos que se basan en añadir material de forma selectiva, formando un objeto por la superposición de capas sucesivas de material a partir del modelo digital. Este proceso de fabricación es muy útil debido a la facilidad de personalización en procesos de diseño y geometrías complejas (Rivera, 2018).

### 7.5.1. Impresora 3D.

La impresora 3D es una máquina controlada por ordenador capaz de fabricar cualquier forma o geometría por muy compleja que sea, lo que significa que es capaz de fabricar objetos complejos que no requieren de pasos posteriores de ensamblaje, moldes o herramientas de corte, facilitando la manufactura de objetos en gran medida (Jorquera Ortega, 2016).

En el presente proyecto se comprende el uso de una impresora 3D MakerBot Replicator+ (fig. 7.18), la cual dispone de una resolución de capa de material a 100 micrones y con diámetro de material a 1.75 mm, con motores con ángulo de laso de 1.8° y micro pasos de 1/16. Cuenta con la compatibilidad de archivos STL para su funcionamiento (MakerBot, 2021).



**Figura 7.18 Impresora MakerBot Replicator +.**

Fuente: (MakerBot, 2021)

Los avances en la manufactura han sido revolucionarios de acuerdo a el uso de las impresoras 3D, ya que son rápidas, baratas y fáciles de usar comparado a otras tecnologías de fabricación por adición de material. Además, cuenta con la capacidad para imprimir partes y montajes con diferentes propiedades físicas y mecánicas con un simple proceso de montaje, aunque dependiendo la calidad de la impresora, este definirá su desempeño. (Fernando Bordigon, 2018)

El uso de distintos plásticos para la elaboración de capas depende de las características que se buscan en el modelo a realizar. Entre los plásticos más usados están ABS y PLA, los cuales se describirán a continuación.

### **7.5.2. Materiales de impresión 3D.**

En la actualidad existe gran variedad de insumos para una impresión aditiva por deposición de material. Es importante considerar los diámetros estándar del filamento y funcionamiento de la impresora, con dos medidas estándar, 1.75 y 3 mm. Entre los filamentos más utilizados se encuentran ABS y PLA debido a sus características y funcionalidad para diversos propósitos. Los cuales se describen a continuación (Fernando Bordigon, 2018).

El filamento ABS (por sus siglas en inglés de acrilonitrilo butadieno estireno) es un termoplástico derivado del petróleo, utilizado en gran variedad de productos cotidianos, entre los más conocidos las piezas de LEGO. Este material cuenta con una resistencia al calor a un punto de fundición de 190 °C, por lo que se recomienda trabajarlo a una temperatura de extrusión de 220 y 250 °C, ideal para piezas que se someten a altas temperaturas. Con respecto a su rigidez, cuenta con la resistencia y fuerza óptimas para múltiples aplicaciones, además de cierta flexibilidad para llevar a cabo un buen acabado o posibilidad de aplicar un tratamiento posterior (pintura, lijado, etc.) (Fernando Bordigon, 2018).

El filamento PLA (por sus siglas en inglés de ácido poliláctico), es un plástico biodegradable derivado del almidón, el cual ha adquirido popularidad en el ámbito doméstico. Una de sus ventajas es que el plástico PLA no emite olores tan fuertes como el plástico ABS por lo que es de preferencia para el mercado doméstico. La temperatura ideal para laborar con PLA es 180 a 230 °C para el extrusor y 60 °C a 70 °C la temperatura de la cama. Se considera una opción ecológica debido a que es un plástico biodegradable, cuenta con una rapidez de impresión mayor por lo que brinda un consumo hasta 30% menor de electricidad y sufre una menor deformación en piezas grandes. Sin embargo, presenta algunas desventajas como la resistencia al calor al exponerse a 60 °C, y es más propenso a fracturarse en caso de aplicarle algún acabado (Fernando Bordigon, 2018).

Para crear la prótesis de mano se propone el uso de plástico PLA debido a las características que benefician al desarrollo del modelo.

## **Capítulo 8 . Metodología**

### **8.1. Diseño de prototipo prótesis de mano.**

En la primera actividad del presente proyecto realizar la recopilación de información para el diseño de una prótesis de mano, tomando en cuenta investigaciones semejantes en donde se implemente la ergonomía y dimensiones respecto a la anatomía de la mano.

Enseguida de acuerdo a datos recopilados, crear un diseño mecánico asistido por computadora de la prótesis de mano con ayuda de software SolidWorks, en donde se incluyan los mecanismos esenciales para para satisfacer las necesidades del proyecto. Luego realizar la impresión en 3D de la prótesis, ensamblar y verificar la movilidad adecuada.

Integrar un diseño eléctrico con los componentes necesarios para brindar el control de los movimientos de la prótesis, tales como servomotores y conexión a un microcontrolador (Arduino), considerando la alimentación eléctrica en general.

### **8.2. Recopilación de información e instalación de módulo EasyVR 3.**

Para aplicar el control por medio de comandos de voz, se requiere investigar acerca del módulo EasyVR 3 empezando por la instalación del software “EasyVR 3 Commander” en el ordenador y estudiar acerca de la conectividad con el Arduino mediante un “SHIELD”.

Es importante leer el manual de usuario de EasyVR 3 brindado por el fabricante ya que ofrece características técnicas con instrucciones precisas para la programación y lograr utilizar el módulo correctamente.

### **8.3. Designar movimientos de operación de prótesis de mano.**

Analizar las posiciones que la prótesis de mano debe adoptar durante la manipulación del ratón con movimientos específicos como clic izquierdo, clic derecho, doble clic, mantener clic (sostenido), soltar clic, sujetar ratón y soltarlo. Plantear en una tabla cada acción con respecto al comando de voz y ángulos de posición para cada servomotor a la flexión o estiramiento de cada dedo.

### **8.4. Análisis de software “EasyVR Commander”.**

Estudiar el software “EasyVR Commander” con ayuda del manual de usuario EasyVR 3 para conocer los conceptos básicos y herramientas fundamentales necesarias durante la programación del módulo de reconocimiento de voz y el control de la prótesis de mano robótica.

### **8.5. Programación para control de prótesis a través de comandos de voz.**

Incluir comandos de voz al software “EasyVR 3 Commander”, previamente asignados en la tabla de acciones al controlar un ratón de ordenador. Enseguida entrenar dichos comandos y determinar el nivel de respuesta modificándolos si es necesario. Luego generar el código, el cual se asigna al microcontrolador para incluir modificaciones que permitan el control de los servomotores.

### **8.6. Integrar módulo EasyVR 3 a prótesis de mano.**

Colocar el módulo de reconocimiento de voz al prototipo de prótesis de mano, considerando la ubicación del micrófono y nivel de recepción de los comandos de voz. Posteriormente calibrar el sistema electromecánico para la transmisión de movimiento y obtener un nivel de respuesta óptimo.

### **8.7. Correcciones de Software/Hardware y pruebas finales.**

A través de pruebas de funcionamiento del prototipo de prótesis de mano en conjunto con módulo de reconocimiento de voz, identificar fallas en software y/o hardware presentadas para aplicar las mejoras correspondientes, asegurando un funcionamiento óptimo y estable.

### 8.8. Informe técnico de la tesis.

Con el fin de documentar cada una de las etapas del proyecto de investigación, realizar un informe técnico de acuerdo a las condiciones establecidas en la guía. La documentación debe ser constante, recopilando información relevante para la investigación, además de incluir resultados de diseño, armado y control de una prótesis de mano mediante reconocimiento de voz.

### 8.9. Metas.

Al diseñar una prótesis de mano, integrando un control por comandos de voz específicos, es necesario administrar el tiempo para cada actividad por lo que se elabora un cronograma, el cual se muestra a continuación.

**Tabla 8.1 Cronograma de actividades para proyecto de investigación.**

ACTIVIDADES		SEMANAS															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	Diseño de prótesis de mano.	■	■	■													
2	Recopilación de información sobre Easy VR3.			■													
3	Vinculación entre Easy VR3 y ordenador.				■												
4	Designar movimientos de operación de prótesis de mano.					■											
5	Análisis de software para programación y selección de comandos de voz.						■	■									
6	Programación en software para el control de la prótesis de mano incluyendo comandos.							■	■	■							
7	Vincular módulo de reconocimiento de voz con la prótesis de mano.										■	■					
8	Fase de pruebas.												■				
9	Corrección de errores de hardware.												■	■			
10	Corrección de errores de software.												■	■			
11	Pruebas finales.															■	
12	Realizar el reporte formal.				■	■	■	■	■	■	■	■	■	■	■	■	■

## Capítulo 9 . Resultados.

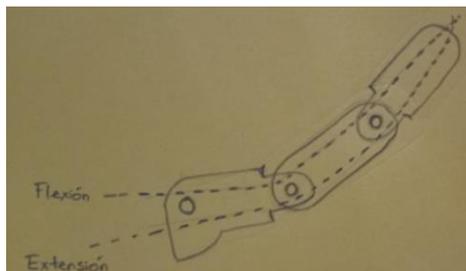
### 9.1. Diseño de prótesis de mano.

La elaboración del prototipo de prótesis de mano se dividió en tres etapas, es decir: diseño mano alzada, diseño mecánico y diseño eléctrico. A continuación, se describe cada una incluyendo evidencias y se explican las acciones realizadas.

#### 9.1.1. Evidencias de diseño mano alzada.

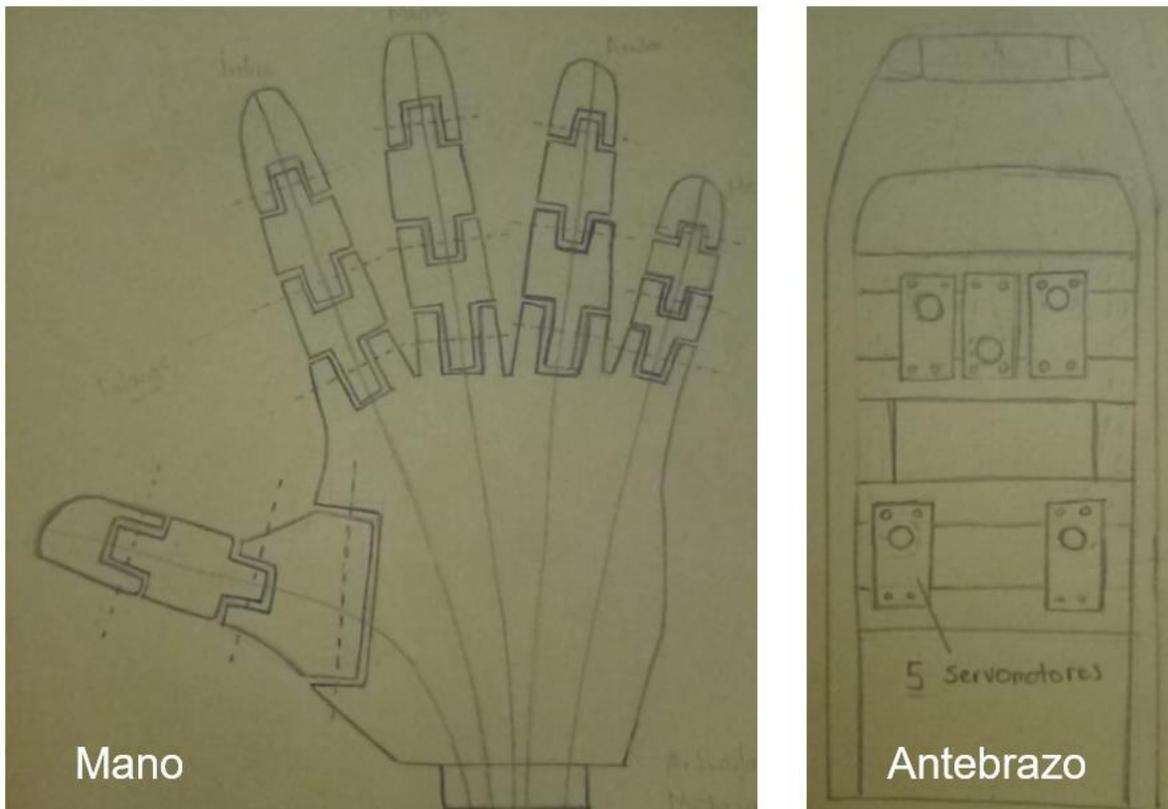
Para elaborar el diseño mecánico se consideró información de distintas prótesis fabricadas con impresión en 3D. Los diseños estudiados se tomaron de la empresa InMoov la cual crea prótesis de código abierto y una tesis propuesta por Mahdi Elisayed Hussein que describe la elaboración de un brazo prostético mioeléctrico con impresión 3D.

Con respecto a lo observado, se crearon dibujos en los que se ilustraron ideas para la mano, antebrazo y las articulaciones. Se considero prioridad analizar acciones como la flexión y la extensión utilizando un tendón artificial para cada dedo controlados por servomotores como se muestra en la figura 9.1.



**Figura 9.1 Tendón artificial para flexión y extensión de cada dedo.**

Fuente: Diseño propuesto.



**Figura 9.2 Mano y antebrazo, primeras ideas de diseño.**

Fuente: Diseño propuesto.

El dibujo a mano alzada de la mano artificial (fig. 9.2 izquierda) se creó con ayuda del contorno de la mano derecha. Se definieron dieciséis piezas, tres para cada dedo, con tres grados de libertad y la palma con la guía de los tendones artificiales y dedos, ensamblándose en la parte inferior con el antebrazo.

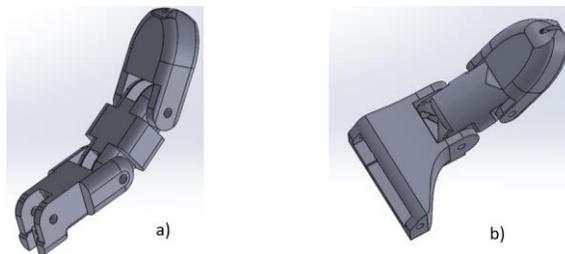
En el antebrazo (fig. 9.2 derecha) se encuentra la distribución de los cinco servomotores, los cuales son articulados por el tendón artificial a cada dedo correspondiente. Los dos servomotores localizados en la parte de abajo corresponde al pulgar y meñique. Para los tres servomotores localizados en la parte de arriba se conectan a dedos anular, medio e índice.

La ergonomía es uno de los aspectos más importantes para elaborar el diseño, ya que es fundamental para una prótesis de mano tener un apego físico al cuerpo del usuario amputado para facilitar el control y adaptabilidad al entorno.

### 9.1.2. Evidencias de diseño asistido por computadora.

El diseño asistido por computadora fue elaborado con ayuda de SolidWorks, un software creado para moldear componentes y crear ensambles mecánicos sólidos. Las ideas en los dibujos de mano alzada y las medidas de falanges y metacarpianas estándar previamente consultada en la anatomía de la mano.

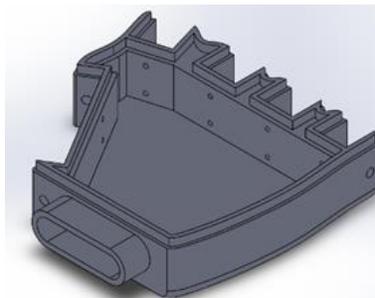
Los dedos índices, medio, anular y meñique (fig. 9.3 a) se crearon de manera semejante, sin embargo, se cambiaron las longitudes con respecto a su posición. El pulgar se creó similar (fig. 9.3 b), pero con distintas características de acuerdo a las funciones que desempeña. Cuenta con tres piezas unidas por pines de alambre galvanizado y con doble guía que atraviesa desde la punta para colocar el tendón artificial, el cual estira y flexiona de acuerdo a su movimiento.



**Figura 9.3 Diseño dedos, a) índice, medio, anular y meñique b) pulgar.**

Fuente: Diseño propuesto.

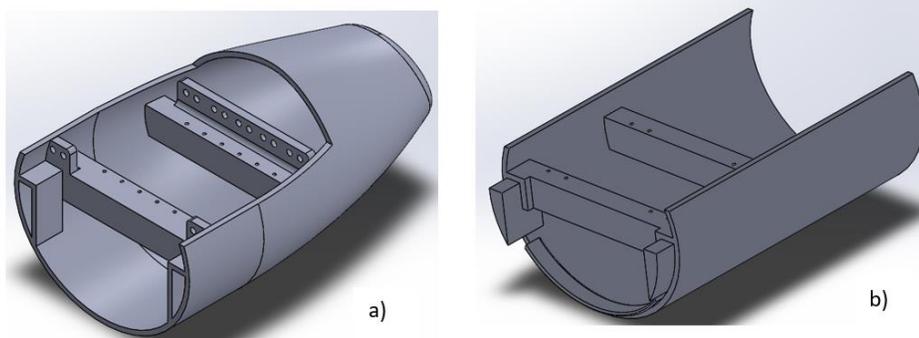
En la palma se ensambla cada dedo según corresponda. Se conecta mediante un eje de alambre galvanizado y se conecta al antebrazo por la parte inferior.



**Figura 9.4 Diseño de la palma.**

Fuente: Diseño propuesto.

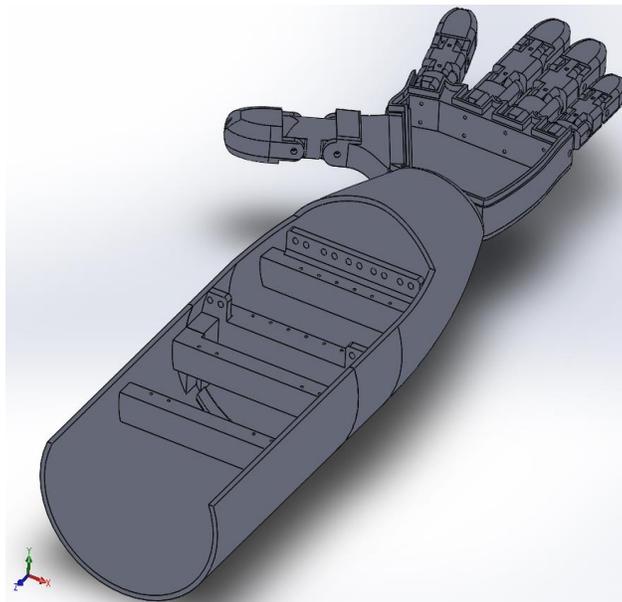
El antebrazo se elaboró con dos piezas, la superior (fig. 9.5 a) con las bases para tres servomotores correspondientes a dedo medio, índice y anular, y la parte inferior (fig. 9.5 b) con bases para el pulgar y meñique.



**Figura 9.5 Diseño del antebrazo.**

Fuente: Diseño propuesto.

Se realizó el ensamblaje de las piezas diseñadas en software SolidWorks (fig. 9.6) y se generaron los archivos .STL. La impresora 3D es compatible con estos archivos, los cuales deben ser cargados para elaborar las piezas con plástico PLA.

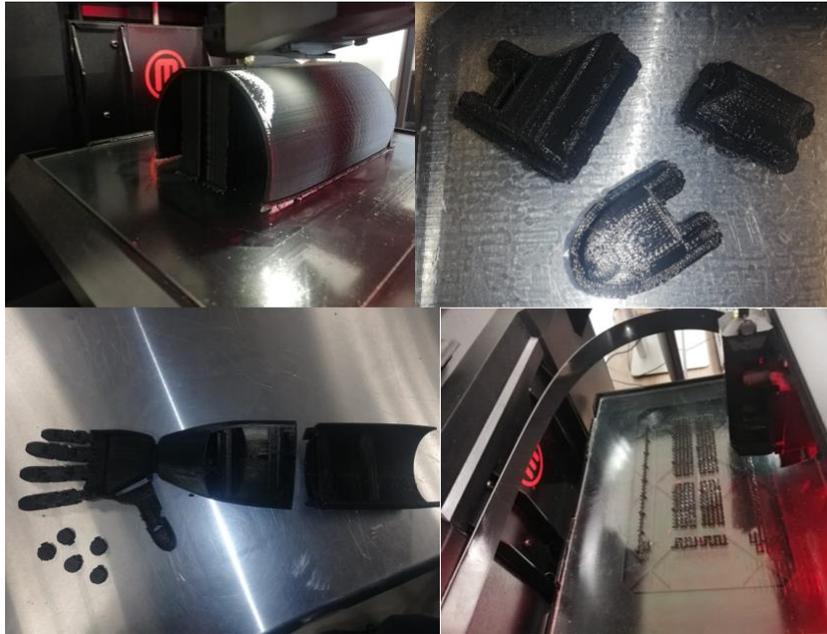


**Figura 9.6 Ensamble de diseño de prótesis de mano.**

Fuente: Diseño propuesto.

### 9.1.3. Evidencias de prototipo impreso en 3D.

Se tomaron fotografías de la impresión 3D del prototipo de prótesis como evidencias. A continuación, se muestra una breve recopilación (fig. 9.7).



**Figura 9.7 Recopilación fotografías de impresión 3D, prototipo de prótesis.**

Fuente: Diseño propuesto.

Posteriormente, luego de imprimir las piezas se realizó el procedimiento de lijado para el ensamble correcto y articular con hilo de pescar desde la punta de cada dedo hasta el servomotor que corresponda. El prototipo se observa en fig. 9.8.

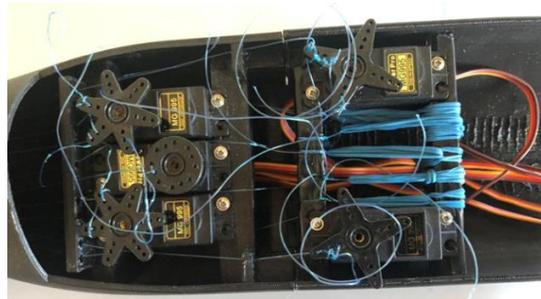


**Figura 9.8 Prototipo de prótesis de mano armado.**

Fuente: Diseño propuesto.

#### 9.1.4.- Evidencias de diseño eléctrico.

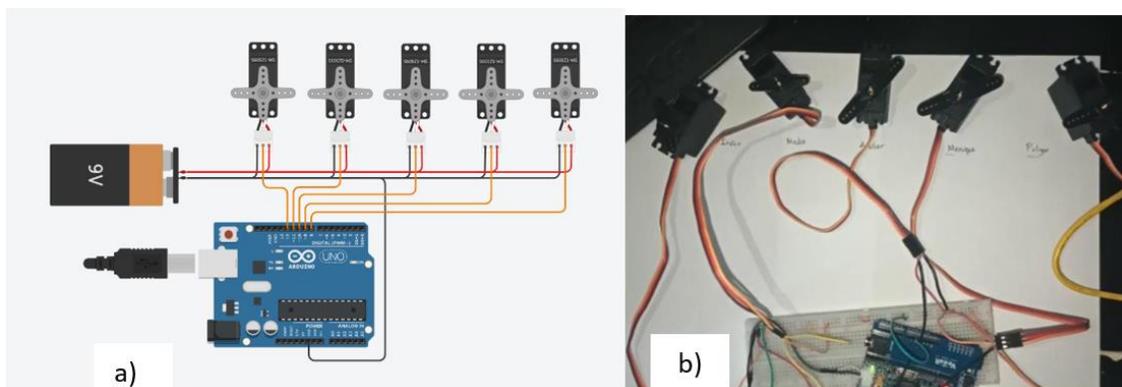
Para dar movilidad al diseño mecánico se colocaron 5 servomotores MG995 correspondientes a cada dedo de la prótesis de mano. Cada servomotor cuenta con tres cables; un par de alimentación compuesto de color café (tierra) y color rojo (positivo), y un cable de color naranja para la señal PWM. En la siguiente figura se observa la posición de cada servomotor (fig. 9.9).



**Figura 9.9 Posición de cada servomotor MG995.**

Fuente: Diseño propuesto.

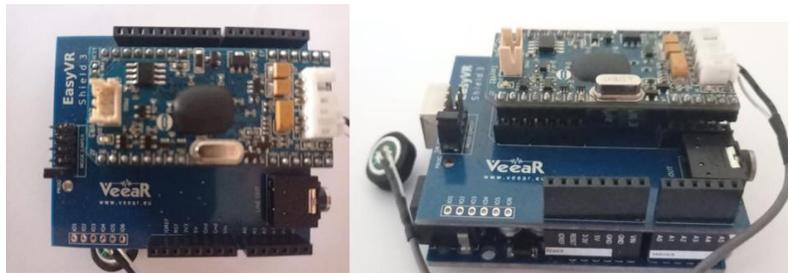
El control para los servomotores se realizó mediante el uso de un microcontrolador Arduino UNO. Los cables de señal PWM se conectaron a los pines 8, 9, 10, 11 y 12 del Arduino. La alimentación eléctrica se obtiene de una fuente externa por lo que el cable de tierra de cada servomotor, el Arduino y la fuente externa se comunican mutuamente. El diseño con las conexiones (9.10 a) y pruebas efectuadas (9.10 b) se muestra en la siguiente figura.



**Figura 9.10 Diseño y pruebas. Servomotores, Arduino y fuente externa.**

Fuente: Circuito propuesto.

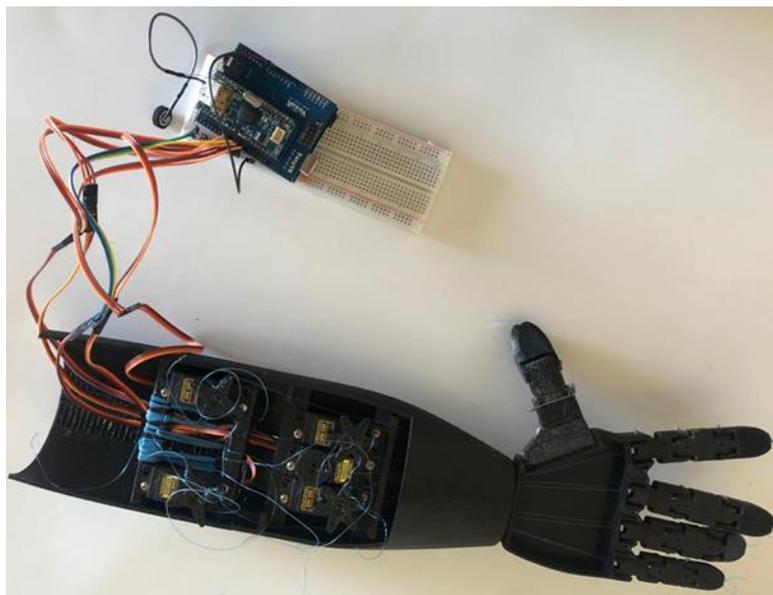
Para la recepción de los comandos de voz se agregó un módulo EasyVR 3 al circuito, con ayuda de un EasyVR SHIELD 3 (se muestra a la izquierda de la figura 9.11) para ensamblarse en la parte superior del Arduino (como se observa a la derecha de la figura 9.11). No es necesaria otra conexión ya que el SHIELD permite seguir utilizando los pines para los servomotores en la misma posición.



**Figura 9.11 EasyVR 3 módulo y SHIELD.**

Fuente: Circuito propuesto.

Con ayuda de una protoboard se armó el circuito de control incorporando todos los elementos necesarios con la prótesis de mano para iniciar con la programación de movimientos. La prótesis y el circuito armado se observan en la figura 9.12.



**Figura 9.12 Prótesis de mano y circuito eléctrico de control.**

Fuente: Diseño propuesto.

## 9.2. Control de prótesis por reconocimiento de voz.

Para el control de la prótesis primero se consultó el objetivo, el cual busca controlar una prótesis de mano para que manipule un ratón de ordenador. Por lo tanto, se analizaron los movimientos y posturas que la prótesis debía ejecutar para distintas acciones que el ratón de ordenador debe adquirir al interactuar con la interfaz. Con ello se establecieron las variables de programación a los comandos de voz. Luego se trabajó con el software EasyVR 3 Commander y Arduino IDE.

### 9.2.1. Acciones asignadas para la prótesis de mano.

Se analizaron las posiciones y movimientos que la prótesis debe realizar para el control de ratón de ordenador, enlistándolas y asignándoles un número para el comando de voz, ya que es más sencillo para el módulo de voz captar palabras cortas, omitiendo los que no eran efectivos como el 3 que se confundía con el 6. Los ángulos de momento para los servomotores se obtuvieron durante las primeras pruebas de flexión y estiramiento de cada dedo al ser ajustada la articulación artificial. La tabla generada fue la siguiente (tabla 9.1).

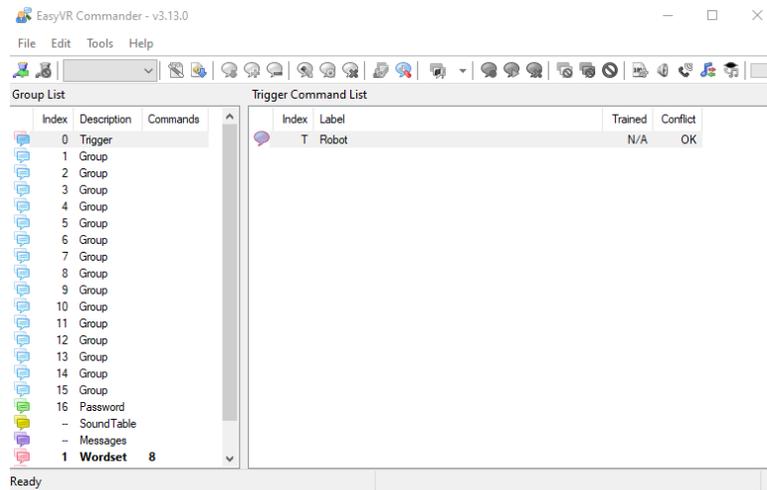
**Tabla 9.1 Acciones, comandos de voz y ángulos de momento.**

Acción	Comando voz	Ángulo de momento de servos				
		Pulgar	Índice	Medio	Anular	Meñique
Apagado	0	0°	0°	0°	0°	0°
Encender	1	--	--	--	--	--
Sostener	2	180°	30°	80°	180°	180°
Clic izquierdo	4	180°	50° 30°	80°	180°	180°
Clic derecho	5	180°	30°	120° 80°	180°	180°
Doble Clic	6	180°	50° 30°	x2 80°	180°	180°
Clic sostenido	7	180°	50°	80°	180°	180°
Soltar clic	8	180°	30°	80°	180°	180°
Soltar	9	0°	0°	0°	0°	0°

Fuente: Creación propia.

### 9.2.2. Programación de software EasyVR Commander.

EasyVR Commander (fig. 9.13) es un software utilizado para configurar fácilmente el módulo EasyVR 3 al conectarlo a la PC con ayuda de una placa host o microcontrolador Arduino.



**Figura 9.13 Entorno principal de EasyVR Commander.**

Fuente: Creación propia.

En el software se definieron los grupos de comandos y se generó una plantilla de código básica para manejar funciones del reconocimiento de voz. A continuación, se describe el procedimiento realizado para la programación de comandos de voz.

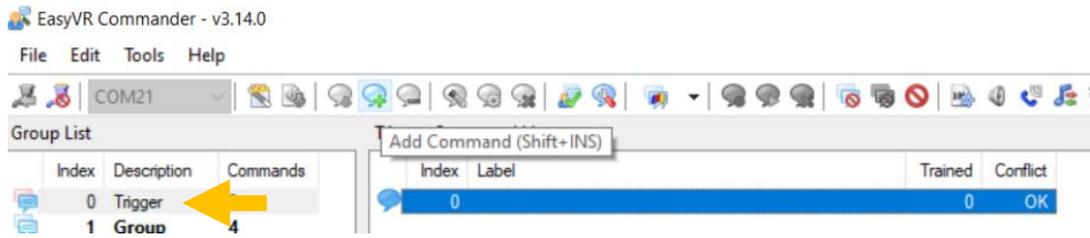
Primero se conectó al ordenador el microcontrolador con el módulo EasyVR 3 en modo PC (fig. 9.14), luego se verificó el puerto COM en el software EasyVR Commander (en este caso COM3).



**Figura 9.14 Conexión modo PC de microcontrolador y módulo EasyVR 3.**

Fuente: Creación propia.

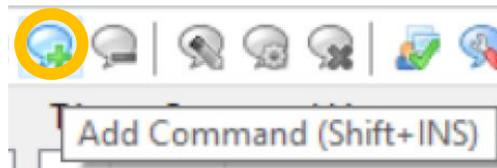
Enseguida se configuró el grupo "Trigger" (grupo 0), el cual permite ingresar a otros grupos a través de comandos de voz. Se encuentra en la parte superior izquierda de la venta del programa EasyVR Commander como señala fig. 9.15.



**Figura 9.15 Ubicación grupo Trigger o grupo 0.**

Fuente: Creación propia.

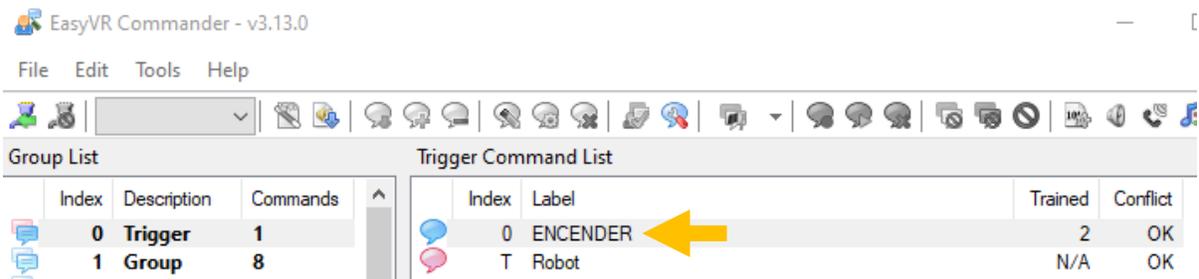
Luego de ubicar el grupo Trigger, se da clic en globo contextual con un símbolo verde para añadir un comando. Se muestra en la cinta de opciones (fig. 9.16).



**Figura 9.16 Botón "Add Command", añadir comando.**

Fuente: Creación propia.

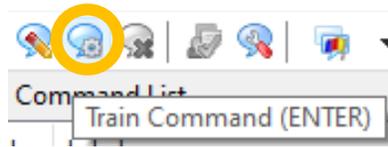
Posteriormente se colocó el nombre "ENCENDER" al primer comando (fig. 9.17).



**Figura 9.17 Nombrar primer comando, "ENCENDER".**

Fuente: Creación propia.

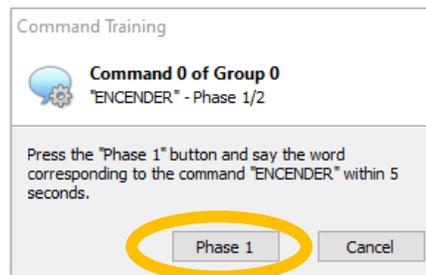
Luego se entrenó el comando “ENCENDER”, dando clic a el icono en forma de globo textual con un engranaje (Train Command) como muestra la figura 9.18.



**Figura 9.18 Botón "Train Command", entrenar comando de voz.**

Fuente: Creación propia.

Aparece una ventana (fig. 9.19) en la cual se pronunció el comando de voz definido en la tabla de acciones para la prótesis de mano. Para “ENCENDER” se grabó el comando de voz “UNO”. A la ventana “Command Training” se le da clic en “Phase 1” para grabar el comando de voz y “Phase 2” repitiendo el procedimiento anterior tratando de sonar lo más semejante posible.

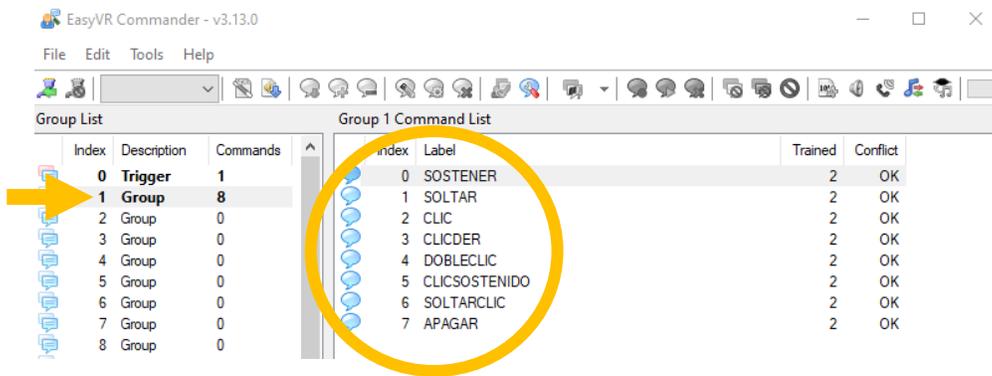


**Figura 9.19 Ventana “Command Training”, entrenar comando.**

Fuente: Creación propia.

Si la recepción del comando de voz es correcta la ventana “Command Training” desaparecerá del entorno. Se repite la grabación del comando si existe una diferencia entre el “Phase 1” y “Phase 2”.

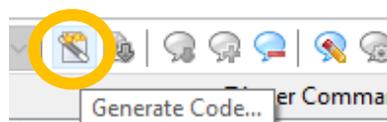
De manera semejante se configuró el grupo 1, agregando los ocho comandos de voz restantes, es decir, “cero” para “APAGADO”, “dos” para “SOSTENER”, “cuatro” para “CLIC”, “cinco” para “CLICDER”, “seis” para “DOBLECLIC”, “siete” para “CLIC SOSTENIDO”, “ocho” para “SOLTAR CLIC” y “nueve” para “SOLTAR”. Los comandos de voz agregados al grupo 1 se observan en la figura 9.20.



**Figura 9.20 Comandos de voz agregados al Grupo 1.**

Fuente: Creación propia.

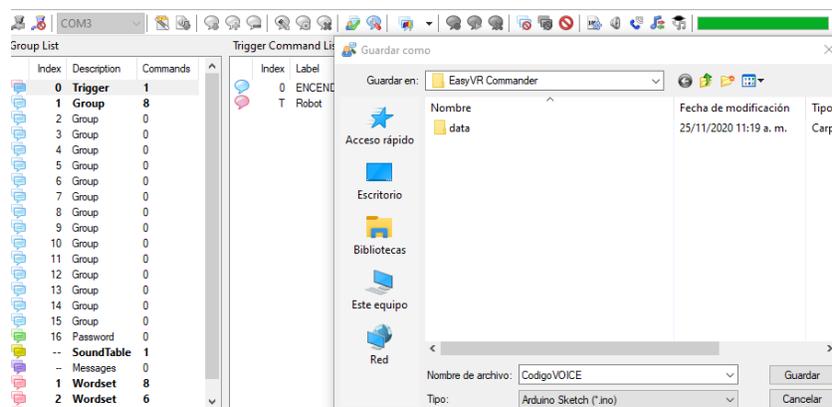
Una vez finalizada la configuración de grupo 0 y grupo 1, se procedió a generar el código IDE (fig. 9.21) el cual permite la edición en Arduino IDE para incluir los servomotores de la prótesis de mano y ser accionados por los comandos de voz.



**Figura 9.21 Botón “Generate Code”, generar código IDE.**

Fuente: Creación propia.

El archivo IDE con extensión .INO se guardó en una carpeta (fig. 9.22), para luego ser abierto por el software Arduino IDE y modificar el código de programación.



**Figura 9.22 Guardar archivo .INO “Código VOICE.**

Fuente: Creación propia.

### 9.2.3. Modificación de código en software Arduino IDE.

El software Arduino de código abierto (IDE) “Integrated Development Environment” (Entorno de Desarrollo Integrado) cuenta con un conjunto de herramientas como un editor de código, un compilador, un depurador y un constructor de interfaz gráfica. La ventana principal de Arduino IDE se muestra en la figura 9.23.

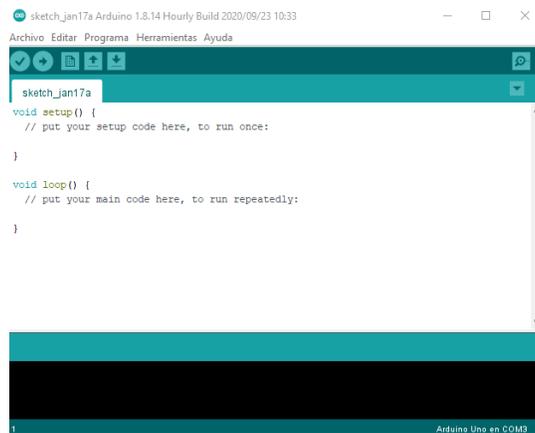


Figura 9.23 Ventana principal de Arduino IDE.

Fuente: Creación propia.

Se cargó el archivo .INO con el nombre “CodigoVOICE” al software Arduino IDE, tal como se muestra en la figura 9.24.

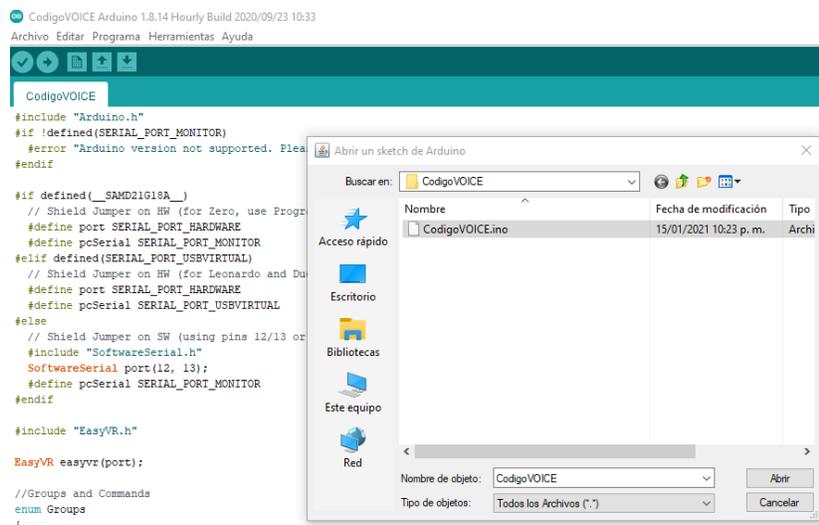


Figura 9.24 Cargar archivo "CodigoVOICE" a software Arduino IDE.

Fuente: Creación propia.

Luego de que el código fue cargado, se modificaron partes específicas para el control de la prótesis de mano a través de los servomotores. A continuación, se describen los pasos realizados. El código completo se encuentra en el anexo D.

Primero se añadieron cinco variables de nombres “índice”, “medio”, “anular”, “menique” y “pulgar” las cuales representan los servomotores, y se les da los valores “8”, “9”, “10”, “11” y “12” respectivamente, dichos valores representan los pines del microcontrolador a los que se conectan los servomotores (fig. 9.25).

```
int indice = 8; //Pin de servomotor
int medio = 9; //Pin de servomotor
int anular = 10; //Pin de servomotor
int menique = 11; //Pin de servomotor
int pulgar = 12; //Pin de servomotor
```

### Figura 9.25 Variables para los servomotores.

Fuente: Creación propia.

Enseguida se agregaron ocho variables más, las cuales son utilizadas para el funcionamiento del programa y el uso se observará más adelante. En la figura 9.26 se muestran dichas variables.

```
int tiempoan; //Tiempo que representa al ángulo
int dindice = 0; //Habilitar/deshabilitar movimiento del dedo índice
int dmedio = 0; //Habilitar/deshabilitar movimiento del dedo medio
int dmenique = 0; //Habilitar/deshabilitar movimiento del dedo meñique
int dpulgar = 0; //Habilitar/deshabilitar movimiento del dedo pulgar
int danular = 0; //Habilitar/deshabilitar movimiento del dedo anular

int indiceant; //Ángulo anterior del dedo índice
int medioant; //Ángulo anterior del dedo medio
```

### Figura 9.26 Variables del código.

Fuente: Creación propia.

Dentro de “setup ()” del código de Arduino se declararon las variables de los servomotores como salidas, tal como se muestra en la figura 9.27.

```
void setup()
{
  pinMode(indice,OUTPUT); //Pin de salida
  pinMode(medio,OUTPUT); //Pin de salida
  pinMode(anular,OUTPUT); //Pin de salida
  pinMode(pulgar,OUTPUT); //Pin de salida
  pinMode(menique,OUTPUT); //Pin de salida
}
```

### Figura 9.27 Pines de salida.

Fuente: Creación propia.

Para el comando “void action ()” se programó la acción que el código debe ejecutar en caso de ser detectado. Por default el programa primero busca los comandos del grupo 0, donde se encuentra el comando “ENCENDER”. Se modificó la acción de dicho comando para que al activarse cambie al grupo 1, permitiendo al módulo EasyVR 3 detectar los comandos pertenecientes (fig. 9.28).

```
case GROUP_0:
  switch (idx)
  {
    case G0_ENCENDER: //Al actuar el comando "ENCENDER"
      group = GROUP_1; //Se cambia de grupo al grupo 1
      break;
  }
  break;
```

### Figura 9.28 Comando “ENCENDER”.

Fuente: Creación propia.

Enseguida para el comando “SOSTENER”, los servomotores para dedo anular, meñique y pulgar se mueven a una posición de 180°, 30° y 80° respectivamente. El movimiento de los tres dedos se realiza dos veces para que puedan completar su trayectoria. Las variables “indiceant” y “medioant” guardan el ángulo de los dedos medio e índice para regresar a dicho ángulo luego de una acción. Las variables “dindice”, “dmedio”, “danular”, “dmenique” y “dpulgar” dan la orden de movimiento a cada dedo al recibir el valor de “1” y asignando el ángulo de momento del servomotor. Al terminar la orden su valor cambia a “0” (fig. 9.29).

```
case G1_SOSTENER:
  danular = 1; //Movimiento de anular
  dmenique = 1; //Movimiento de meñique
  dpulgar = 1; //Movimiento de pulgar
  movera(180); //A un ángulo de 180°
  danular = 1; //Movimiento de anular
  dmenique = 1; //Movimiento de meñique
  dpulgar = 1; //Movimiento de pulgar
  movera(180); //A un ángulo de 180°

  dindice = 1; //Movimiento de índice
  movera(30); //A un ángulo de 30°
  dmedio = 1; //Movimiento de medio
  movera(80); //A un ángulo de 80°

  indiceant = 30; //Ángulo actual del índice
  medioant = 80; //Ángulo actual del medio
  break;
```

### Figura 9.29 Modificación al comando "SOSTENER".

Fuente: Creación propia.

Luego se modificó el comando “SOLTAR” donde todos los servomotores regresan a posición 0°, tal como se muestra en el fragmento de código de la figura 9.30.

```
case G1_SOLTAR:
    dindice = 1; //Movimiento de índice
    dmedio = 1; //Movimiento de medio
    danular = 1; //Movimiento de anular
    dmenique = 1; //Movimiento de meñique
    dpulgar = 1; //Movimiento de pulgar
    movera(0); //A un ángulo de 0°

    indiceant = 0; //Ángulo actual del índice
    medioant = 0; //Ángulo actual del medio
    break;
```

### Figura 9.30 Modificación de comando "SOLTAR".

Fuente: Creación propia.

Para el comando “CLIC”, el dedo índice realiza un movimiento de 50° y regresa a su posición anterior, la modificación se observa en figura 9.31.

```
case G1_CLIC:
    dindice = 1; //Movimiento de índice
    movera(50); //A un ángulo de 50°
    dindice = 1; //Movimiento de índice
    movera(indiceant); //Al ángulo anterior
    break;
```

### Figura 9.31 Modificación de comando “CLIC”.

Fuente: Creación propia.

Se modificó el comando “CLICDER” (fig. 9.31) para que el dedo medio realice un movimiento a 120° y regresa a su posición anterior.

```
case G1_CLICDER:
    dmedio = 1; //Movimiento de medio
    movera(120); //A un ángulo de 120°
    dmedio = 1; //Movimiento de medio
    movera(medioant); //Al ángulo anterior
    break;
```

### Figura 9.32 Modificación de comando “CLICDER”.

Fuente: Creación propia.

Para el comando “DOBLECLIC” el dedo índice realiza un movimiento a 50° y regresa a su posición anterior, repitiendo el proceso una vez más (fig. 9.33).

```
case G1_DOBLECLIC:
  dindice = 1; //Movimiento de índice
  movera(50); //A un ángulo de 50°
  dindice = 1; //Movimiento de índice
  movera(indiceant); //Al ángulo anterior
  dindice = 1; //Movimiento de índice
  movera(50); //A un ángulo de 50°
  dindice = 1; //Movimiento de índice
  movera(indiceant); //Al ángulo anterior
  break;
```

### Figura 9.33 Modificación de comando “DOBLECLIC”.

Fuente: Creación propia.

Al detectar el comando “CLICSOSTENIDO” el dedo índice se mueve a un ángulo de 50° y mantiene esa posición, la modificación de código se muestra en fig. 9.34.

```
case G1_CLICSOSTENIDO:
  dindice = 1; //Movimiento de índice
  movera(50); //A un ángulo de 50°
  break;
```

### Figura 9.34 Modificación de comando “CLICSOSTENIDO”.

Fuente: Creación propia.

Para el comando “SOLTARCLIC” la acción a ejecutar es el índice debe moverse a la última posición guardada en la variable “indiceant” tal como describe fig. 9.35.

```
case G1_SOLTARCLIC:
  dindice = 1; //Movimiento de índice
  movera(indiceant); //Al ángulo anterior
  break;
```

### Figura 9.35 Modificación de comando “SOLTARCLIC”.

Fuente: Creación propia.

Enseguida se modificó el código para que al activar el comando “APAGAR” todos los servomotores cambien a un ángulo de 0° y se regresa al grupo 0 (fig. 9.36). En esta condición los comandos del grupo 1 no se pueden detectar hasta que se realice un nuevo cambio de grupo activando el comando “ENCENDER”.

```

case G1_APAGAR:
    dindice = 1; //Movimiento de índice
    dmedio = 1; //Movimiento de medio
    danular = 1; //Movimiento de anular
    dmenique = 1; //Movimiento de meñique
    dpulgar = 1; //Movimiento de pulgar
    movera(0); //A un ángulo de 0°
    indiceant = 0; //Ángulo actual del índice
    medioant = 0; //Ángulo actual del medio
    group = GROUP_0; //Se cambia de grupo al grupo 0
    break;
}
break;

```

**Figura 9.36 Modificación de comando “APAGAR”.**

Fuente: Creación propia.

Por último, se modificó la parte final del código. Esta modificación permite crear una señal cuadrada la cual es enviada a los servomotores para su funcionamiento, por lo que se recibe un ángulo aplicando una formula con la cual se obtiene el tiempo que debe durar en alto la señal cuadrada. Para la generación de la señal cuadrada se activa una señal de 5V a los servomotores que se vayan a controlar y dura el tiempo obtenido anteriormente a partir del ángulo, después se les da una señal de 0V durante el tiempo que falte para completar 20 milisegundos, este proceso se realiza 50 veces por lo que en total la señal dura un segundo. El código modificado se encuentra en la fig. 9.37.

```

void movera(int anguloentrada)
{
    tiempoan = (100 * anguloentrada); //Cambio de ángulo a tiempo
    tiempoan = tiempoan / 9; //Cambio de ángulo a tiempo
    tiempoan = tiempoan + 500; //Cambio de ángulo a tiempo
    for(int hz = 0; hz < 50; hz++) //Ciclo que dura un segundo
    {
        if(dindice == 1) digitalWrite(indice,HIGH); //Señal de 5V al índice en caso de valer 1
        if(dmedio == 1) digitalWrite(medio,HIGH); //Señal de 5V al medio en caso de valer 1
        if(danular == 1) digitalWrite(anular,HIGH); //Señal de 5V al anular en caso de valer 1
        if(dpulgar == 1) digitalWrite(pulgar,HIGH); //Señal de 5V al pulgar en caso de valer 1
        if(dmenique == 1) digitalWrite(menique,HIGH); //Señal de 5V al meñique en caso de valer 1
        delayMicroseconds(tiempoan); //Temporizador en microsegundos correspondiente al ángulo
        if(dindice == 1) digitalWrite(indice,LOW); //Señal de 0V al índice en caso de valer 1
        if(dmedio == 1) digitalWrite(medio,LOW); //Señal de 0V al medio en caso de valer 1
        if(danular == 1) digitalWrite(anular,LOW); //Señal de 0V al anular en caso de valer 1
        if(dpulgar == 1) digitalWrite(pulgar,LOW); //Señal de 0V al pulgar en caso de valer 1
        if(dmenique == 1) digitalWrite(menique,LOW); //Señal de 0V al meñique en caso de valer 1
        delayMicroseconds(20000 - tiempoan); //Temporizador en microsegundos restantes para completar 20 milisegundos
    }
    delay(500); //Temporizador de 500 milisegundos
    dindice = 0; //Desactivar movimiento de índice
    dmedio = 0; //Desactivar movimiento de medio
    danular = 0; //Desactivar movimiento de anular
    dmenique = 0; //Desactivar movimiento de meñique
    dpulgar = 0; //Desactivar movimiento de pulgar
}

```

**Figura 9.37 Modificación de código para la señal cuadrada de servomotores.**

Fuente: Creación propia.

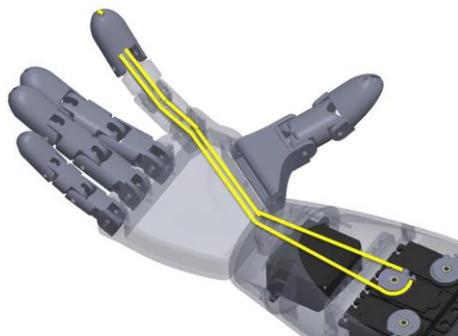
#### **9.4. Pruebas realizadas.**

Después de trabajar con el diseño de la prótesis de mano y programación correspondiente, se realizaron pruebas de funcionamiento en tres aspectos; el sistema electromecánico, nivel de respuesta ante la recepción de comandos de voz y aplicación de la prótesis para manipular un ratón de ordenador. A continuación, se describen las evidencias generadas.

##### **9.4.1. Pruebas y correcciones del sistema electromecánico.**

La prótesis de mano primero fue armada incorporando las articulaciones artificiales colocadas en cada dedo comunicado con el servomotor correspondiente, además se agregó el circuito eléctrico para las conexiones adecuadas entre los elementos lo cual define el sistema electromecánico.

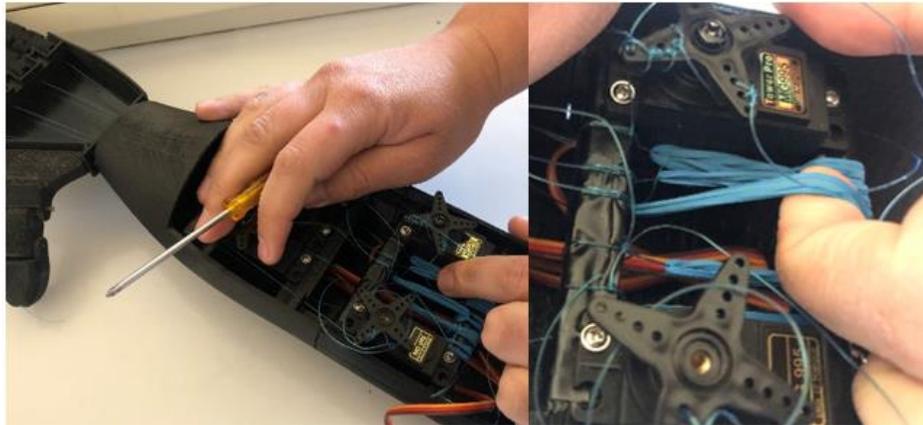
Se llevaron a cabo observaciones para el ajuste de las articulaciones artificiales. Inicialmente las articulaciones de flexión y extensión se extendían por el servomotor correspondiente para cada dedo, como se muestra en la figura 9.38.



**Figura 9.38 Posición de la articulación para la flexión y extensión.**

Fuente: Diseño propuesto.

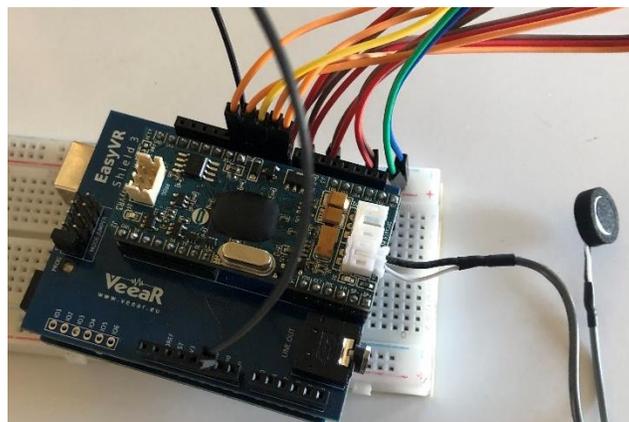
Sin embargo, luego de distintas pruebas y falta de firmeza al ejecutar los movimientos, se optó por reubicar la articulación encargada de la extensión de cada dedo a una liga tensa en la base trasera de la prótesis como se muestra en fig. 9.39. Para la flexión se conservó la articulación en el servomotor.



**Figura 9.39** Reubicación de articulaciones para la flexión y extensión.

Fuente: Diseño propuesto

También se verifico que cada servomotor fuera controlado correctamente por el microcontrolador y el módulo de voz, observando que todos los pines estuvieran en el lugar indicado y el nivel de respuesta fuera el correcto (fig. 9.40).



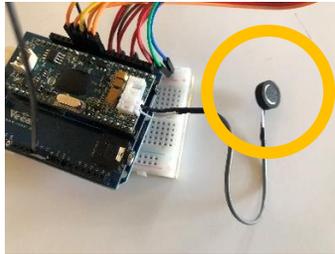
**Figura 9.40** Vista general de circuito eléctrico.

Fuente: Diseño propuesto

La alimentación eléctrica se obtuvo de una fuente de poder para un ordenador Dell Optiplex 320, solo utilizando una de las terminales con 5 volts de salida para conectar los servomotores a través de una protoboard. Sin embargo, se considera implementar a futuro pilas de 9 volts recárgales para optimizar el circuito.

### 9.4.2. Pruebas finales de respuesta de los comandos de voz.

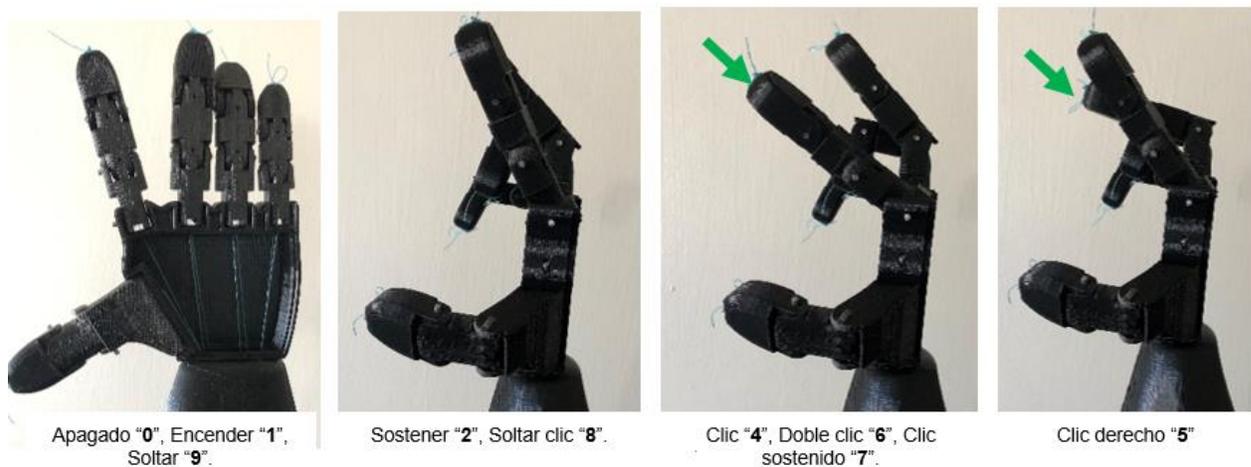
Luego de cargar el código modificado al microcontrolador, el cual ya contaba con el shield y el módulo de voz en funcionamiento, se realizaron pruebas de recepción de comandos de voz en un entorno interior donde los ruidos que interfieren no son un problema grave para el funcionamiento del micrófono. Para las pruebas se colocó el micrófono cerca del circuito como muestra la figura 9.41.



**Figura 9.41 Ubicación del micrófono para pruebas de respuesta.**

Fuente: Creación propia.

En general se obtuvo nivel de respuesta bueno ante los distintos comandos designados en la tabla de acciones, ya que al elegir números en lugar de acción por su nombre facilitan la recepción. Se tuvo que omitir el comando “tres” por qué se confundía con el comando “seis”. A continuación, se muestra una recopilación de acciones ejecutadas de la prótesis con distintos comandos de voz (fig. 9.42).



**Figura 9.42 Acciones de la prótesis de mano y comandos ejecutados.**

Fuente: Creación propia.

### 9.4.3. Pruebas para manipular el ratón de ordenador.

Al comprobar que las acciones se ejecutaban correctamente y se adoptaban las posturas adecuadas, se hicieron pruebas con un ratón de ordenador inalámbrico. Primero se sobrepuso la prótesis de mano en el ratón, esta postura se utiliza para comandos de voz “0”, “1” y “9” correspondientes a apagado, encender y soltar. La postura se muestra en la figura 9.43.



**Figura 9.43 Postura para apagado, encender y soltar con ratón de ordenador.**

Fuente: Creación propia.

Luego se probó el funcionamiento con los comandos “2” y “8”, los cuales corresponden a “sostener” y “soltar clic” (es la postura adoptada al desactivar clic sostenido). La acción de la prótesis se observa en figura 9.44.



**Figura 9.44 Postura de prótesis para comandos de voz "2" y "8".**

Fuente: Creación propia.

Enseguida se llevó a cabo la ejecución de los comandos "4", "6" y "7", correspondientes a las acciones "clic", "doble clic" y "clic sostenido". Aunque la postura observada en la figura 9.45 se captura de la misma manera, la ejecución es distinta ya que para "clic" el dedo índice se flexiona y vuelve a su posición original, en "doble clic" se ejecuta dos veces la acción y para "clic sostenido" solo se flexiona permaneciendo en la misma posición. La primera posición para los servomotores es de 30° y para flexionar el dedo índice se considera 50°.



**Figura 9.45 Postura de prótesis para comandos de voz "4", "6" y "7".**

Fuente: Creación propia.

Por último, se ejecutó el clic derecho con el comando de voz "5" donde se observó la postura del dedo medio, primero con el servomotor a 80°, 120° al clic y se regresa a la posición inicial. La figura 9.46 muestra la posición del dedo medio al tener el comando en funcionamiento.



**Figura 9.46 Postura de prótesis para comandos de voz "5".**

Fuente: Creación propia.

## Capítulo 10 . Conclusiones y recomendaciones

- a) Considerando los resultados obtenidos del presente proyecto de investigación, se concluye que la hipótesis formulada es aceptada ya que, si es posible incorporar un microcontrolador y el reconocimiento de voz para el control de una prótesis de mano al ponerla a prueba para manipular un ratón de ordenador, observando que las acciones se ejecutaban sin mayor problema y la interacción entre usuario y prótesis era sencillo.
- b) El control por reconocimiento de voz es popular en la actualidad para interactuar con distintas tecnologías, sin embargo, para el desarrollo prostético no ha sido un tema relevante por lo que no existen tantas alternativas como esta para la rehabilitación prostética, la cual podría reducir el costo considerablemente.
- c) La calidad de vida aumenta para una persona discapacitada al implementar una prótesis electromecánica controlada por comandos de voz, ya que al permitirle realizar actividades como manipular un ratón de ordenador aumenta su productividad y rehabilitación en gran medida. Además, al aplicarlo a otros ámbitos como tomar una taza o recoger un objeto integrando más comandos de voz podría aumentar sus habilidades.

Como recomendación, es importante considerar que la prótesis creada para esta investigación fue el primer prototipo, por lo tanto, existen aspectos en el diseño que pudieran ser mejorados como la firmeza en los movimientos ajustando el sistema mecánico proponiendo guías para los tendones artificiales que producen la flexión y extensión de cada uno de los dedos.

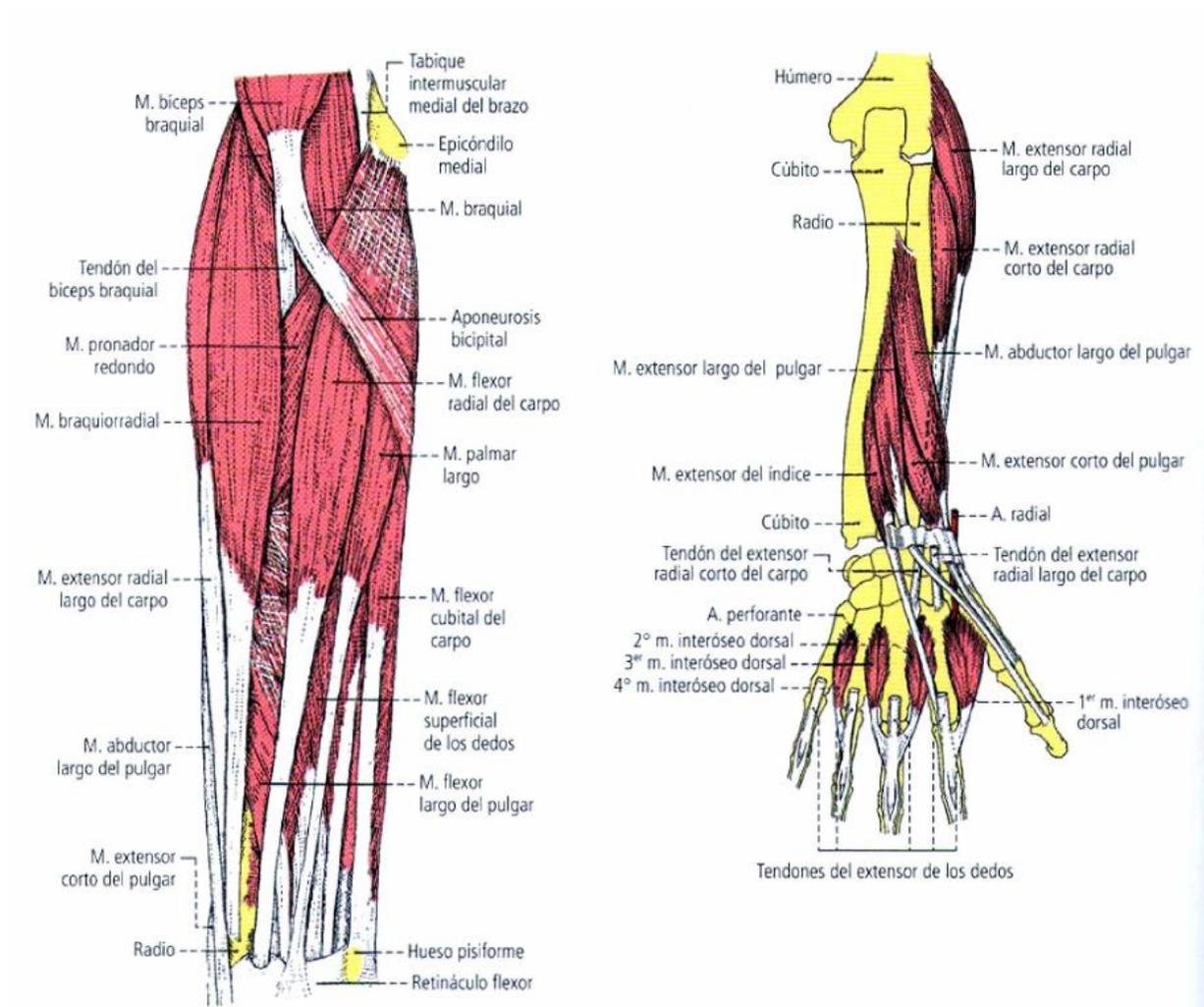
## Capítulo 11 . Bibliografía

- Andres, C. P. (2020). *Desarrollo de un sistema para el reconocimiento de comandos de voz implementados en una cabeza robótica*. Quito, Ecuador: Instituto Superior Tecnológico Vida Nueva.
- Arduino. (11 de Enero de 2021). *Arduino UNO Rev3*. Obtenido de Arduino Store: <https://store.arduino.cc/usa/arduino-uno-rev3>
- Educar Sociedad del Estado. (s.f.). *Conceptos básicos de la impresión 3D*. Argentina: Ministerio de Educación, Cultura, Ciencia y Tecnología.
- Electronicos Caldas. (s.f.). *MG995 High Speed Metal Gear Dual Ball Bearing Servo*. Obtenido de Datasheet MG995: [https://www.electronicoscaldas.com/datasheet/MG995\\_Tower-Pro.pdf](https://www.electronicoscaldas.com/datasheet/MG995_Tower-Pro.pdf)
- Fernando Bordigon, A. A. (2018). *Diseño e impresión de objetos 3D*. Ciudad Autónoma de Buenos Aires: Unipe Editorial Universitaria.
- Jofre L. Brito, M. X. (2013). Estudio del estado del arte de las prótesis de mano. *Ingenius Revista de ciencia y tecnología*, 57-64.
- Jorquera Ortega, A. (2016). *Fabricación Digital: Introducción al modelo e impresión 3D*. España: Ministerio de Educación, Cultura y Deporte-Área de Educación.
- Lisandro Puglisi, H. M. (s.f.). Prótesis Robóticas. *Departamento de Automática*, 1-10.
- Londoño, M. P. (2013). *Diseño y Construcción de una Prótesis para Ciclistas con Amputación*. Bogotá: Universidad de los Andes.
- MakerBot. (11 de Enero de 2021). *MakerBot Replicator +*. Obtenido de MakerBot: <https://www.makerbot.com/3d-printers/replicator/>
- Médico-Quirúrgica, E. (2007). *Kinesiterapia, medicina física*. París: Elsevier Masson.
- Nemcansky, K. (2017). *Manual básico FDM i3D*. Buenos Aires, Argentina: INTI Diseño Industrial.
- Norton, K. (2007). Un breve recorrido por la historia de la prótesis. *inMotion*, 1-5.
- Oxolón Salvador, M. P. (2018). *Tratamiento fisioterapéutico en amputados de miembro superior*. Lima, Perú.
- Periago, R. Z. (2009). *Prótesis, ortesis y ayudas técnicas*. España S.L.: Elsevier Masson.
- RAE, R. A. (06 de Enero de 2021). *Definición de mano*. Obtenido de Diccionario de la lengua española: <https://dle.rae.es/mano?m=form>

- RAE, R. A. (05 de Enero de 2021). *Definición de prótesis*. Obtenido de Diccionario de la lengua española: <https://dle.rae.es/pr%C3%B3tesis?m=form>
- Richard L. Drake, A. W. (2015). *GRAY Anatomía para estudiantes*. Barcelona, España: Churchill Livingstone, Elsevier.
- Rivera, D. L. (2018). *Análisis y mejora del diseño de una mano protésica*. Madrid, España: Industriales.
- Rouvière H. Delmas, V. D. (2005). *Anatomía Humana Descriptiva, topográfica y funcional*. París: Masson, S.A.
- Ruiz Liar Alfredo, L. M. (2011). *Anatomía humana*. Buenos Aires, Argentina: Editorial Médica Panamericana.
- Sánchez Calderón, D. H. (2016). *Consideraciones anatómicas en el diseño de una mano artificial*. Bogotá, Colombia: Universidad Militar Nueva Granada.
- Universitaria, C. (23 de Octubre de 2013). *DISEÑAN MECANISMO DE VOZ PARA DAR MOVIMIENTO A PRÓTESIS DE ANTEBRAZO*. Obtenido de Boletín UNAM-DGCS-634: [https://www.dgcs.unam.mx/boletin/bdboletin/2013\\_634.html](https://www.dgcs.unam.mx/boletin/bdboletin/2013_634.html)
- Veear. (02 de Enero de 2016). *Manual de Easy VR 3 plus*. Obtenido de Fortebit: <https://fortebit.tech/docs/manuals/easyvr-3/>
- Vela Sánchez, E. V. (2016). *Los amputados y su rehabilitación. Un reto para el Estado*. México: Intersistemas, S.A. de C.V.
- Villacis, J. M. (2008). *Sistema de control electrónico por comandos de voz para prototipo de robot compañero*. Riobamba, Ecuador: ESPOCH.
- Vinicio D. Méndez, D. O. (2015). *Diseño e implementación del sistema de movimiento direccional de una silla de ruedas para ser controlada por reconocimiento de un patrón de voz*. Quito, Ecuador: Universidad Politécnica Salesiana.
- Vv, A. (2009). Tema 26 Fisioterapia en ortopedia, Amputación de miembros. En A. Vv, *Fisioterapeutas, Servicio de Salud de la Comunidad de Madrid* (pág. 20). España: Adams.

## Capítulo 12 . Anexos

### Anexo A Músculos del compartimiento antebraquial anterior, porción superficial y músculos extensores radiales, abductor largo, extensores del pulgar.



Fuente: Anatomía humana, Ruiz Liard, Latarjet 2011.

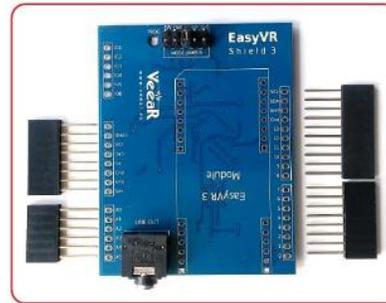
## Anexo B Especificaciones técnicas de EasyVR Shield.

### Descripción del producto

EasyVR Shield 3 es una placa adaptadora para el módulo EasyVR 3, diseñada para simplificar su uso entre la comunidad Arduino.

El Shield es compatible con cualquier placa Arduino que utilice encabezados Shield UNO-R3, funcionando a niveles de 3,3 V o 5 V, utilizando el pin IOREF para seleccionar el voltaje de funcionamiento EasyVR.

También es compatible con versiones anteriores de las placas Arduino anteriores que no tienen el pin IOREF, que utilizan niveles de E / S de 5 V de forma predeterminada.



Si su placa no tiene el pin IOREF pero está funcionando a 3.3V, aún puede operar el EasyVR Shield 3 correctamente si conecta manualmente los pines IOREF y 3V3 juntos, por ejemplo, con un cable de puente.

La placa viene con encabezados apilables Arduino separados para la interfaz Shield. El módulo EasyVR 3 también se proporciona por separado.

**Nota:** ¡El módulo EasyVR 3 y todos los encabezados apilables deben soldarse antes de su uso!

### Características de EasyVR Shield 3

- Compatible con placas Arduino que tienen la interfaz 1.0 Shield (UNO R3) que incluye, pero no se limita a:
  - Arduino cero
  - Arduino Uno
  - Arduino Mega
  - Arduino Leonardo
  - Arduino Due
- Admite placas principales de 5V y 3.3V a través del pin IOREF (por defecto es 5V si este pin está ausente) Admite la conexión directa a la PC en las placas principales con un chip USB / serie separado y un "modo puente" especial controlado por software en las placas con solo interfaz USB nativa, para un fácil acceso y configuración con EasyVR Commander
- Permite diferentes modos de conexión en serie y también actualizaciones flash del módulo EasyVR integrado (a través del puente de modo)
- Admite la reasignación de pines en serie utilizados por Shield (en modo SW)
- Proporciona un conector de salida de audio de 3.5 mm adecuado para auriculares o como salida de línea



EasyVR Shield 3 completamente ensamblado

Fuente: (Veeear, 2016)

## Anexo C Características técnicas de módulo EasyVR, asignación de pines.

Grupo	Nombre	Número	Affiler	Tipo	Descripción
● MIKROBUS	J1	1	-	-	(No conectado)
		2	RST	yo	Restablecimiento asincrónico bajo activo (pull-up interno) (No
		3-6	-	-	conectado)
		7	3V3	yo	Entrada de alimentación de 3,3 V CC
		8	GND	-	Tierra
	J2	1	XM	yo	Selección de arranque (menú desplegable interno)
		2	Detaware	O	(Reservado)
		3	TX	O	Transmisión de datos en serie
		4	RX	yo	Recepción de datos en serie
		5-6	-	-	(No conectado)
		7	5V	yo	Entrada de alimentación de 5,0 V CC
		8	GND	-	Tierra
	● GPIO	J3	1	IO1	E/S
2			IO2	E/S	propósito general (niveles lógicos VDD) E/S de propósito
3			IO3	E/S	general (niveles lógicos VDD) E/S de propósito general
4			IO4	E/S	(niveles lógicos VDD) E/S de propósito general (niveles
5			IO6	E/S	lógicos VDD) E/S de propósito general (niveles lógicos
6			IO6	E/S	VDD)
● AUDIO	J4	1	SP+	O	Salida de audio diferencial (puede conducir directamente un altavoz de 8 Ω)
		2	SP-	O	
		3	VM	O	Potencia del micrófono (para admitir micrófonos
		4	MIC	yo	personalizados) Entrada de audio para micrófono
		5	RETIRADO	-	Retorno de micrófono (tierra analógica) Voltaje lógico
		6	VDD	O	Interno (solo como referencia)
● CABLES	J5	1	SP-	O	Salida de audio diferencial (puede conducir directamente un altavoz de 8 Ω)
		3	SP+	O	
		2	-	-	(No conectado)
	J6	1	MIC	yo	Entrada de audio para micrófono
		2	RETIRADO	-	Retorno de micrófono (tierra analógica) Cable de
● ADAPTADOR	J7	1	RX_P	O	programación recepción de datos en serie
		2	RTS_P	yo	Solicitud de cable de programación para enviar (reinicio / control de arranque)
		3	GND	-	Tierra del cable de programación
		4	5V_P	yo	Cable de programación Salida de alimentación de 5 V CC Cable de
		5	TX_P	yo	programación transmisión de datos en serie
		6	CTS_P	O	Cable de programación claro para enviar (atado a tierra)

Fuente: (Veear, 2016)

## Anexo D Código de programación para el control de la prótesis de mano.

```
#include "Arduino.h"
#if !defined(SERIAL_PORT_MONITOR)

#endif

#if defined(__SAM21G18A__)
  // Shield Jumper on HW (for Zero, use Programming Port)
  #define port SERIAL_PORT_HARDWARE
  #define pcSerial SERIAL_PORT_MONITOR
#elif defined(SERIAL_PORT_USBVIRTUAL)
  // Shield Jumper on HW (for Leonardo and Due, use Native Port)
  #define port SERIAL_PORT_HARDWARE
  #define pcSerial SERIAL_PORT_USBVIRTUAL
#else
  // Shield Jumper on SW (using pins 12/13 or 8/9 as RX/TX)
  #include "SoftwareSerial.h"
  SoftwareSerial port(12, 13);
  #define pcSerial SERIAL_PORT_MONITOR
#endif

#include "EasyVR.h"

EasyVR easyvr(port);

//Groups and Commands
enum Groups
{
  GROUP_0 = 0,
  GROUP_1 = 1,
};

enum Group0
{
  G0_ENCENDER = 0,
};

enum Group1
{
  G1_SOSTENER = 0,
  G1_SOLTAR = 1,
  G1_CLIC = 2,
  G1_CLICDER = 3,
  G1_DOBLECLIC = 4,
  G1_CLICSOSTENIDO = 5,
  G1_SOLTARCLIC = 6,
  G1_APAGAR = 7,
};
```

```
//Grammars and Words
enum Wordsets
{
    SET_1 = -1,
    SET_2 = -2,
    SET_3 = -3,
};

enum Wordset1
{
    S1_ACTION = 0,
    S1_MOVE = 1,
    S1_TURN = 2,
    S1_RUN = 3,
    S1_LOOK = 4,
    S1_ATTACK = 5,
    S1_STOP = 6,
    S1_HELLO = 7,
};

enum Wordset2
{
    S2_LEFT = 0,
    S2_RIGHT = 1,
    S2_UP = 2,
    S2_DOWN = 3,
    S2_FORWARD = 4,
    S2_BACKWARD = 5,
};

enum Wordset3
{
    S3_ZERO = 0,
    S3_ONE = 1,
    S3_TWO = 2,
    S3_THREE = 3,
    S3_FOUR = 4,
    S3_FIVE = 5,
    S3_SIX = 6,
    S3_SEVEN = 7,
    S3_EIGHT = 8,
    S3_NINE = 9,
    S3_TEN = 10,
};
```

```

// use negative group for wordsets
int8_t group, idx;

int indice = 8;    //Pin de servomotor
int medio = 9;    //Pin de servomotor
int anular = 10;  //Pin de servomotor
int menique = 11; //Pin de servomotor
int pulgar = 12;  //Pin de servomotor

int tiempoan;    //Tiempo que representa al ángulo
int dindice = 0; //Habilitar/deshabilitar movimiento del dedo índice
int dmedio = 0;  //Habilitar/deshabilitar movimiento del dedo medio
int dmenique = 0; //Habilitar/deshabilitar movimiento del dedo meñique
int dpulgar = 0; //Habilitar/deshabilitar movimiento del dedo pulgar
int danular = 0; //Habilitar/deshabilitar movimiento del dedo anular

int indiceant;   //Ángulo anterior del dedo índice
int medioant;    //Ángulo anterior del dedo medio

void setup()
{
  pinMode(indice,OUTPUT); //Pin de salida
  pinMode(medio,OUTPUT);  //Pin de salida
  pinMode(anular,OUTPUT); //Pin de salida
  pinMode(pulgar,OUTPUT); //Pin de salida
  pinMode(menique,OUTPUT); //Pin de salida
  // setup PC serial port
  pcSerial.begin(9600);
bridge:
  // bridge mode?
  int mode = easyvr.bridgeRequested(pcSerial);
  switch (mode)
  {
  case EasyVR::BRIDGE_NONE:
    // setup EasyVR serial port
    port.begin(9600);
    // run normally
    pcSerial.println(F("Bridge not requested, run normally"));
    pcSerial.println(F("---"));
    break;

  case EasyVR::BRIDGE_NORMAL:
    // setup EasyVR serial port (low speed)
    port.begin(9600);

```

---

```

// run normally
pcSerial.println(F("Bridge not requested, run normally"));
pcSerial.println(F("----"));
break;

case EasyVR::BRIDGE_NORMAL:
// setup EasyVR serial port (low speed)
port.begin(9600);
// soft-connect the two serial ports (PC and EasyVR)
easyvr.bridgeLoop(pcSerial);
// resume normally if aborted
pcSerial.println(F("Bridge connection aborted"));
pcSerial.println(F("----"));
break;

case EasyVR::BRIDGE_BOOT:
// setup EasyVR serial port (high speed)
port.begin(115200);
pcSerial.end();
pcSerial.begin(115200);
// soft-connect the two serial ports (PC and EasyVR)
easyvr.bridgeLoop(pcSerial);
// resume normally if aborted
pcSerial.println(F("Bridge connection aborted"));
pcSerial.println(F("----"));
break;
}

// initialize EasyVR
while (!easyvr.detect())
{
pcSerial.println(F("EasyVR not detected!"));
for (int i = 0; i < 10; ++i)
{
if (pcSerial.read() == '?')
goto bridge;
delay(100);
}
}

pcSerial.print(F("EasyVR detected, version "));
pcSerial.print(easyvr.getID());

if (easyvr.getID() < EasyVR::EASYVR3)
easyvr.setPinOutput(EasyVR::IO1, LOW); // Shield 2.0 LED off

```

```

if (easyvr.getID() < EasyVR::EASYVR)
  pcSerial.print(F(" = VRbot module"));
else if (easyvr.getID() < EasyVR::EASYVR2)
  pcSerial.print(F(" = EasyVR module"));
else if (easyvr.getID() < EasyVR::EASYVR3)
  pcSerial.print(F(" = EasyVR 2 module"));
else
  pcSerial.print(F(" = EasyVR 3 module"));
pcSerial.print(F(", FW Rev."));
pcSerial.println(easyvr.getID() & 7);

easyvr.setDelay(0); // speed-up replies

easyvr.setTimeout(5);
easyvr.setLanguage(0); //<-- same language set on EasyVR Commander when code was generated

group = EasyVR::TRIGGER; //<-- start group (customize)

}

void loop()
{
  if (easyvr.getID() < EasyVR::EASYVR3)
    easyvr.setPinOutput(EasyVR::IO1, HIGH); // LED on (listening)

  if (group < 0) // SI wordset/grammar
  {
    pcSerial.print("Say a word in Wordset ");
    pcSerial.println(-group);
    easyvr.recognizeWord(-group);
  }
  else // SD group
  {
    pcSerial.print("Say a command in Group ");
    pcSerial.println(group);
    easyvr.recognizeCommand(group);
  }

  do
  {
    // allows Commander to request bridge on Zero (may interfere with user protocol)
    if (pcSerial.read() == '?')
    {
      setup();
    }
  }
}

```

---

```

    return;
}
// <<-- can do some processing here, while the module is busy
}
while (!easyvr.hasFinished());

if (easyvr.getID() < EasyVR::EASYVR3)
    easyvr.setPinOutput(EasyVR::IO1, LOW); // LED off

idx = easyvr.getWord();
if (idx == 0 && group == EasyVR::TRIGGER)
{
    // beep
    easyvr.playSound(0, EasyVR::VOL_FULL);
    // print debug message
    pcSerial.println("Word: ROBOT");
    // write your action code here
    // group = GROUP_X\SET_X; <-- jump to another group or wordset
    return;
}
else if (idx >= 0)
{
    // beep
    easyvr.playSound(0, EasyVR::VOL_FULL);
    // print debug message
    uint8_t flags = 0, num = 0;
    char name[32];
    pcSerial.print("Word: ");
    pcSerial.print(idx);
    if (easyvr.dumpGrammar(-group, flags, num))
    {
        for (uint8_t pos = 0; pos < num; ++pos)
        {
            if (!easyvr.getNextWordLabel(name))
                break;
            if (pos != idx)
                continue;
            pcSerial.print(F(" = "));
            pcSerial.println(name);
            break;
        }
    }
    // perform some action
    action();
    return;
}

```

---

```

}
idx = easyvr.getCommand();
if (idx >= 0)
{
    // beep
    easyvr.playSound(0, EasyVR::VOL_FULL);
    // print debug message
    uint8_t train = 0;
    char name[32];
    pcSerial.print("Command: ");
    pcSerial.print(idx);
    if (easyvr.dumpCommand(group, idx, name, train))
    {
        pcSerial.print(" = ");
        pcSerial.println(name);
    }
    else
        pcSerial.println();
    // perform some action
    action();
}
else // errors or timeout
{
    if (easyvr.isTimeout())
        pcSerial.println("Timed out, try again...");
    int16_t err = easyvr.getError();
    if (err >= 0)
    {
        pcSerial.print("Error ");
        pcSerial.println(err, HEX);
    }
}
}

void action()
{
    switch (group)
    {
    case GROUP_0:
        switch (idx)
        {
        case G0_ENCENDER: //Al actuar el comando "ENCENDER"
            group = GROUP_1; //Se cambia de grupo al grupo 1
            break;
        }
    }
}

```

```

break;
case GROUP_1:
switch (idx)
{
case G1_SOSTENER:
danular = 1; //Movimiento de anular
dmenique = 1; //Movimiento de meñique
dpulgar = 1; //Movimiento de pulgar
movera(180); //A un ángulo de 180°
danular = 1; //Movimiento de anular
dmenique = 1; //Movimiento de meñique
dpulgar = 1; //Movimiento de pulgar
movera(180); //A un ángulo de 180°

dindice = 1; //Movimiento de índice
movera(30); //A un ángulo de 30°
dmedio = 1; //Movimiento de medio
movera(80); //A un ángulo de 80°

indiceant = 30; //Ángulo actual del índice
medioant = 80; //Ángulo actual del medio
break;
case G1_SOLTAR:
dindice = 1; //Movimiento de índice
dmedio = 1; //Movimiento de medio
danular = 1; //Movimiento de anular
dmenique = 1; //Movimiento de meñique
dpulgar = 1; //Movimiento de pulgar
movera(0); //A un ángulo de 0°

indiceant = 0; //Ángulo actual del índice
medioant = 0; //Ángulo actual del medio
break;
case G1_CLIC:
dindice = 1; //Movimiento de índice
movera(50); //A un ángulo de 50°
dindice = 1; //Movimiento de índice
movera(indiceant); //Al ángulo anterior
break;
case G1_CLICDER:
dmedio = 1; //Movimiento de medio
movera(120); //A un ángulo de 120°
dmedio = 1; //Movimiento de medio
movera(medioant); //Al ángulo anterior
break;

```

```

case G1_DOBLECLIC:
    dindice = 1; //Movimiento de índice
    movera(50); //A un ángulo de 50°
    dindice = 1; //Movimiento de índice
    movera(indiceant); //Al ángulo anterior
    dindice = 1; //Movimiento de índice
    movera(50); //A un ángulo de 50°
    dindice = 1; //Movimiento de índice
    movera(indiceant); //Al ángulo anterior
    break;
case G1_CLICSOSTENIDO:
    dindice = 1; //Movimiento de índice
    movera(50); //A un ángulo de 50°
    break;
case G1_SOLTARCLIC:
    dindice = 1; //Movimiento de índice
    movera(indiceant); //Al ángulo anterior
    break;
case G1_APAGAR:
    dindice = 1; //Movimiento de índice
    dmedio = 1; //Movimiento de medio
    danular = 1; //Movimiento de anular
    dmenique = 1; //Movimiento de meñique
    dpulgar = 1; //Movimiento de pulgar
    movera(0); //A un ángulo de 0°
    indiceant = 0; //Ángulo actual del índice
    medioant = 0; //Ángulo actual del medio
    group = GROUP_0; //Se cambia de grupo al grupo 0
    break;
}
break;
case SET_1:
    switch (idx)
    {
    case S1_ACTION:
        // write your action code here
        // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands
        break;
    case S1_MOVE:
        // write your action code here
        // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands
        break;
    case S1_TURN:
        // write your action code here
        // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands

```

```

    break;
case S1_RUN:
    // write your action code here
    // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands
    break;
case S1_LOOK:
    // write your action code here
    // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands
    break;
case S1_ATTACK:
    // write your action code here
    // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands
    break;
case S1_STOP:
    // write your action code here
    // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands
    break;
case S1_HELLO:
    // write your action code here
    // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands
    break;
}
break;
case SET_2:
    switch (idx)
    {
    case S2_LEFT:
        // write your action code here
        // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands
        break;
    case S2_RIGHT:
        // write your action code here
        // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands
        break;
    case S2_UP:
        // write your action code here
        // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands
        break;
    case S2_DOWN:
        // write your action code here
        // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands
        break;
    case S2_FORWARD:
        // write your action code here
        // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands

```

```

    break;
case S2_BACKWARD:
    // write your action code here
    // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands
    break;
}
break;
case SET_3:
    switch (idx)
    {
    case S3_ZERO:
        // write your action code here
        // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands
        break;
    case S3_ONE:
        // write your action code here
        // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands
        break;
    case S3_TWO:
        // write your action code here
        // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands
        break;
    case S3_THREE:
        // write your action code here
        // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands
        break;
    case S3_FOUR:
        // write your action code here
        // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands
        break;
    case S3_FIVE:
        // write your action code here
        // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands
        break;
    case S3_SIX:
        // write your action code here
        // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands
        break;
    case S3_SEVEN:
        // write your action code here
        // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands
        break;
    case S3_EIGHT:
        // write your action code here
        // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands

```

```

        break;
    case S3_NINE:
        // write your action code here
        // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands
        break;
    case S3_TEN:
        // write your action code here
        // group = GROUP_X\SET_X; <-- or jump to another group or wordset for composite commands
        break;
    }
    break;
}
}

void movera(int anguloentrada)
{
    tiempoan = (100 * anguloentrada); //Cambio de ángulo a tiempo
    tiempoan = tiempoan / 9;          //Cambio de ángulo a tiempo
    tiempoan = tiempoan + 500;        //Cambio de ángulo a tiempo
    for(int hz = 0; hz < 50; hz++) //Ciclo que dura un segundo
    {
        if(dindice == 1) digitalWrite(indice,HIGH); //Señal de 5V al índice en caso de valer 1
        if(dmedio == 1) digitalWrite(medio,HIGH); //Señal de 5V al medio en caso de valer 1
        if(danular == 1) digitalWrite(anular,HIGH); //Señal de 5V al anular en caso de valer 1
        if(dpulgar == 1) digitalWrite(pulgar,HIGH); //Señal de 5V al pulgar en caso de valer 1
        if(dmenique == 1) digitalWrite(menique,HIGH); //Señal de 5V al meñique en caso de valer 1
        delayMicroseconds(tiempoan); //Temporizador en microsegundos correspondiente al ángulo
        if(dindice == 1) digitalWrite(indice,LOW); //Señal de 0V al índice en caso de valer 1
        if(dmedio == 1) digitalWrite(medio,LOW); //Señal de 0V al medio en caso de valer 1
        if(danular == 1) digitalWrite(anular,LOW); //Señal de 0V al anular en caso de valer 1
        if(dpulgar == 1) digitalWrite(pulgar,LOW); //Señal de 0V al pulgar en caso de valer 1
        if(dmenique == 1) digitalWrite(menique,LOW); //Señal de 0V al meñique en caso de valer 1
        delayMicroseconds(20000 - tiempoan); //Temporizador en microsegundos restantes para 20 milisegundos
    }
    delay(500); //Temporizador de 500 milisegundos
    dindice = 0; //Desactivar movimiento de índice
    dmedio = 0; //Desactivar movimiento de medio
    danular = 0; //Desactivar movimiento de anular
    dmenique = 0; //Desactivar movimiento de meñique
    dpulgar = 0; //Desactivar movimiento de pulgar
}
}

```

Fuente: Creación propia.