



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

---

---

**FACULTAD DE ESTUDIOS SUPERIORES CUAUTITLÁN**

Prótesis humana de brazo robótico utilizando  
señales EMG

## TESIS

QUE PARA OBTENER EL TÍTULO DE:  
INGENIERO EN TELECOMUNICACIONES,  
SISTEMAS Y ELECTRÓNICA

PRESENTA:

ROBERTO HERNÁNDEZ SÁNCHEZ

ASESOR DE TESIS:

Dr. FERNANDO GUDIÑO PEÑALOZA

CUAUTITLÁN IZCALLI, ESTADO DE MÉXICO, 2023



Universidad Nacional  
Autónoma de México



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.



UNIVERSIDAD NACIONAL  
AUTÓNOMA DE  
MÉXICO

**FACULTAD DE ESTUDIOS SUPERIORES CUAUTITLÁN  
SECRETARÍA GENERAL  
DEPARTAMENTO DE TITULACIÓN**

**ASUNTO: VOTO APROBATORIO**



**DR. DAVID QUINTANAR GUERRERO  
DIRECTOR DE LA FES CUAUTITLAN  
PRESENTE**

**ATN: DRA. MARÍA DEL CARMEN VALDERRAMA BRAVO  
DEPARTAMENTO DE TITULACIÓN  
de la FES Cuautitlán.**

Con base en el Reglamento General de Exámenes, y la Dirección de la Facultad, nos permitimos comunicar a usted que revisamos el: Trabajo de Tesis

**Prótesis humana de brazo robótico utilizando señales EMG**

Que presenta el pasante: **Roberto Hernández Sánchez**

Con número de cuenta: 313072725 para obtener el Título de: **Ingeniero en Telecomunicaciones, Sistemas y Electrónica**

Considerando que dicho trabajo reúne los requisitos necesarios para ser discutido en el **EXAMEN PROFESIONAL** correspondiente, otorgamos nuestro **VOTO APROBATORIO**.

**ATENTAMENTE**

**"POR MI RAZA HABLARÁ EL ESPÍRITU"**

Cuautitlán Izcalli, Méx. a 07 de Diciembre de 2022.

**PROFESORES QUE INTEGRAN EL JURADO**

	NOMBRE	FIRMA
<b>PRESIDENTE</b>	Mtro. Jorge Buendía Gómez	
<b>VOCAL</b>	Mtro. Ángel Hilario García Bacho	
<b>SECRETARIO</b>	Dr. Fernando Gudiño Peñaloza	
<b>1er. SUPLENTE</b>	Mtro. Leopoldo Martín del Campo Ramírez	
<b>2do. SUPLENTE</b>	Dr. David Tinoco Varela	

NOTA: los sinodales suplentes están obligados a presentarse el día y hora del Examen Profesional (art. 127).

## **Agradecimientos**

*A mis padres, María del Carmen Sánchez González y Jaime Hernández Sánchez, gracias por acompañarme a lo largo de mi trayectoria de vida, siempre estaré agradecido. También me gustaría agradecer a mis hermanos.*

*Agradezco a la Universidad Nacional Autónoma de México por la gran oportunidad de pertenecer a la máxima casa de estudios.*

*Agradezco al Dr. Fernando Gudiño Peñaloza por el apoyo que recibí de usted a lo largo de la carrera.*

*Agradezco a mis amigos y a cada uno de los profesores por el apoyo que me brindaron a lo largo de mi estancia por la universidad, nunca los olvidaré.*

## Índice

Introducción.....	1
Objetivos .....	2
Capítulo 1. Antecedentes y Marco Teórico.....	3
1.1 Discapacidad y amputaciones en México .....	3
1.2 Descripción de prótesis.....	3
1.3 Prótesis mioeléctricas .....	4
1.3.1 Tipos de prótesis.....	4
1.4 Las señales biológicas .....	10
1.4.1 Las señales bioeléctricas.....	12
1.4.2 Las señales electromiográficas (EMG) .....	12
1.5 Procesamiento de señales.....	14
1.6 Elementos de un sistema de procesamiento de señales EMG .....	16
1.6.1 Electroodos.....	16
1.6.2 Etapa de amplificación.....	17
1.6.3 Componentes de filtrado.....	18
1.7 Herramientas y componentes .....	20
1.7.1 Sensor AD8232.....	20
1.7.2 Software de visualización y manipulación.....	21
1.7.3 La transformada Wavelet.....	22
1.7.4 Controlador ON – OFF con Histéresis. ....	25
Capítulo 2. Metodología .....	27
2.1 Adquisición de señales .....	30
2.2 Configuración de Arduino para adquisición de datos .....	32
2.3 Metodología Arduino-Python.....	34

2.4 Controlador ON – OFF con histéresis.....	38
2.5 Control PC – Prótesis.....	39
2.6 Configuración angular de los servomotores.....	40
2.7 Prótesis de mano.....	42
2.8 Salidas electrónicas del sistema.....	47
Capítulo 3. Experimentación y Resultados.....	48
3.1 Adquisición de las señales utilizando Arduino - Python.....	48
3.2 Clasificación de señales.....	50
3.2.1 Identificación de Wavelets en las señales de muestra.....	55
3.2.2 Interpretación.....	59
3.3 Controlador ON – OFF con histéresis.....	61
3.4 Control PC – Prótesis.....	64
Conclusiones y trabajo a futuro.....	71
Bibliografía.....	74
Anexo A. Códigos del sistema.....	79
Anexo B. Hoja de datos técnica.....	87

# Glosario

*ADC – Convertidor Analógico – Digital.*

*CMRR – Relación de rechazo en modo común.*

*dB – Decibeles.*

*CTW - Transformada de Wavelet Continua.*

*DC – Corriente directa.*

*ECG – Electrocardiograma.*

*EMG – Electromiografía.*

*EMGI – Electromiografía invasiva o intramuscular.*

*EMGS- Electromiografía superficial.*

*GBP – Producto Ganancia de Ancho de Banda.*

*IA – Inteligencia Artificial.*

*IMSS – Instituto Mexicano del Seguro Social.*

*IMU - Unidades de Medición Inercial.*

*Offset – Desplazamiento vertical de DC a través del tiempo.*

*OMS – Organización Mundial de la Salud.*

*PID - Controlador Proporcional, Integrador y Derivador.*

*PLA – Ácido poliláctico.*

*Prótesis Mioeléctrica - Prótesis accionada por señales electromiográficas, señales biológicas que tienen como característica impulso eléctricos, estos pulsos eléctricos son los que controlan la prótesis.*

*SENIAM - Evaluación Muscular de Electromiografía Superficial no Invasiva.*

*SNC – Sistema Nervioso Central.*

*SNP – Sistema Nervioso Periférico.*

*Slew Rate – Velocidad de Respuesta.*

*TFCT -Transformada de Fourier de Corto Tiempo.*

# Introducción

De acuerdo con el *Instituto Mexicano del Seguro Social (IMSS)* en México existen setecientos ochenta mil personas con amputaciones, y de setenta y cinco mil personas amputadas, solo siete mil quinientos tienen una prótesis (10 %) y de ellas, cinco mil doscientos cincuenta (7 %) no saben utilizarla. La principal causa de amputación es el padecimiento de Diabetes Mellitus tipo 2 [1] y accidentes laborales.

En México existen empresas que ofrecen el servicio de venta de prótesis, siendo la empresa “TigreRobotics” quien ofrece una prótesis de antebrazo y mano en el rango de veinte mil pesos mexicanos, de acuerdo con los precios del año 2018 [2], algunas empresas extranjeras ofrecen sus servicios de brazo completo en alrededor de once mil dólares (USD), mano y antebrazo en nueve mil setecientos cincuenta USD y mano completa en seis mil USD, oscilando entre ciento quince mil y doscientos mil pesos mexicanos [3].

De acuerdo con los datos anteriores, el objetivo de este trabajo es demostrar la implementación y desarrollo de un prototipo de prótesis de mano izquierda, siendo por el momento una prótesis no aplicable para uso humano, sin embargo, se trabajará a futuro para dicha aplicación.

El presente trabajo cuenta con 3 secciones donde se explican los temas de investigación de la tesis, estos capítulos son: *Antecedentes y Marco teórico, Metodología, Experimentación y Resultados*.

Dentro de la sección de *Antecedentes y Marco Teórico* se visualiza el tipo de señales bioeléctricas que produce el cuerpo, tipos de prótesis funcionales y el sensor AD8232 que será fundamental para el proyecto. Dentro de la sección de *Metodología* se visualizará la forma en cómo se ataca el problema y el funcionamiento de la prótesis. En la sección de experimentación, se podrán observar las pruebas a realizar, así como los errores y aciertos que se tuvieron en el presente trabajo.



## **Objetivos**

- Se implementará un prototipo de prótesis mioeléctrica de miembro superior, controlada por un sistema *ON – OFF con histéresis*, utilizando el circuito AD8232.
- Se utilizará la forma de onda de la transformada de *Wavelet Mexican Hat* replicando un detector de picos.
- Se implementará diferentes secuencias de movimientos de forma automática para el control y funcionamiento de la prótesis (Control PC – Prótesis).

# **Capítulo 1. Antecedentes y Marco Teórico**

Dentro de este capítulo se establecerán los antecedentes y marco teórico que sustentarán esta tesis, así como demostrar los diferentes tipos de prótesis funcionales junto con la prótesis de interés, como las electromiográficas, adquisición de señales y herramientas para su adquisición.

## *1.1 Discapacidad y amputaciones en México*

De acuerdo con el *Instituto Nacional de Estadística y Geografía (INEGI)* [4], se identifica a las personas con discapacidad como aquellas que tienen la dificultad para llevar a cabo actividades básicas consideradas como: ver, escuchar, caminar, recordar o concentrarse. Según la *Organización Mundial de la Salud (OMS)* [4], hasta el año 2020, se ha contabilizado una cifra de mil millones de personas que han sufrido alguna especie de discapacidad, aproximadamente el 15 % de la población mundial, de este porcentaje, alrededor de 190 millones de personas tienen dificultades en su movilidad y requieren con frecuencias servicios de asistencia. De acuerdo con el Censo de Población y Vivienda 2020 [4], en México hay 6,179,890 personas con algún tipo de discapacidad, lo que representa 4.9 % de la población total del país. De ellas, 53 % son mujeres y 47 % son hombres.

## *1.2 Descripción de prótesis*

De acuerdo con la *Real Academia Española*, una prótesis es una “*pieza o aparato empleados para sustituir un órgano o miembro del cuerpo*” [5]. Definiendo que una prótesis es una extensión artificial de una extremidad amputada, haciendo que una extremidad vuelva a ser funcional, generando mejor calidad de vida y ayuda psicológica dentro del trauma generado. Dentro de estas prótesis existen diferentes modelos.

### 1.3 Prótesis mioeléctricas

De acuerdo con la revista *INGENIUS* [6], unas de las prótesis con mayor estética, precisión y fuerza son las prótesis mioeléctricas, que se basan en el registro de señales musculares mediante electrodos que permitan adquirir las señales eléctricas provenientes del cuerpo, para luego amplificarlas, procesarlas y filtrarlas para el manejo de la prótesis [6]. La ventaja de este tipo de prótesis es que el usuario solo necesita flexionar un músculo para operarla, a diferencia de las prótesis activadas por el cuerpo que requieren un movimiento general del cuerpo.

Recuperando la idea de *Jesús Dorador* [7], las prótesis controladas de forma mioeléctrica eliminan la correa de suspensión de una prótesis tradicional, este arnés funciona como una parte inmóvil que sujeta la prótesis con respecto a la clavícula del cuerpo, para eliminar esto, se utilizan dos técnicas de suspensión: bloqueo de tejidos blandos – esqueleto o succión. Afortunadamente estas prótesis funcionan por baterías, ofreciendo un mayor nivel tecnológico, sin embargo, su precio incrementa en comparación con una prótesis tradicional. Como desventaja, el sistema accionado por baterías requiere de mantenimiento. Debido al peso de la batería y del motor eléctrico, las prótesis que funcionan con batería tienden a ser más pesada que otras opciones [7].

#### 1.3.1 Tipos de prótesis

##### A. Prótesis estéticas

De acuerdo con *Jofre Brito* [8], el objetivo de las prótesis estéticas, es cubrir de forma ornamental el miembro amputado, por lo tanto, no tendrán ningún movimiento. Los materiales que se utilizan para fabricar la prótesis consisten en polímeros, más en específico materiales como látex flexible, silicona o PVC rígido,

esto se debe a que sus materiales deben de ser livianos y de fácil mantenimiento debido a que no cuentan con partes móviles como las prótesis mioeléctricas. En la figura 1.1 se puede visualizar diferentes prototipos de prótesis estéticas.



Fig. 1.1 Diferente tipo de prótesis estéticas de miembro superior; dedos, manos, antebrazos del cuerpo humano.

### *B. Prótesis funcionales*

#### *Prótesis I-Limb*

De acuerdo a la empresa *Össur*, la prótesis biónica *I-Limb* es una de las prótesis que cuenta con mayor presencia en países miembros de la Unión Europea, Estados Unidos, América Latina y algunos países del continente asiático. Como característica principal es que replica la misma cantidad de movimientos que puede tener una mano biológica, ayudando a los pacientes a reincorporarse a la vida laboral y mejorando sus condiciones de vida, esto se puede realizar debido a que la prótesis cuenta con un movimiento independiente para cada dedo, lo que permite que el pulgar pueda rotar en un ángulo de  $90^\circ$ , formar una pinza y en consecuencia poder agarrar algún objeto de forma precisa, replicando un movimiento natural [9].

Existe otro modelo llamado prótesis *I-Limb Quantum* (figura 1.2), siendo el arquetipo con más tecnología y adecuado para un funcionamiento mioeléctrico, este modelo cuenta con una mayor cantidad de articulaciones, los dedos de la mano están

fabricados por titanio, lo que proporcionan un 30 % más en fuerza de agarre y un 30 % más en la velocidad con la que se trata de replicar el movimiento de forma natural, fuerza y funcionalidad de una mano biológica.

Esta prótesis está diseñada en diferentes tamaños: extrapequeño, pequeño, mediano y grande. De acuerdo con los ingenieros de *Össur*, se ha desarrollado una aplicación de control móvil que permite programar 12 diferentes tipos de agarre en la mano que sean de preferencia para el usuario, 24 movimientos automáticos, crear listas de movimientos de rutina, contiene funciones que sirven para comprobar el buen funcionamiento de la prótesis y un entorno visual que permita observar las señales musculares a través de un gráfico en tiempo real [9].



Fig. 1.2 Prótesis I-Limb Quantum. Vista frontal de la mano, prótesis diseñada con 24 movimientos automáticos.

### *Prótesis beBionic*

De acuerdo a la página web de la empresa *Ottobock*, la prótesis de mano *beBionic* tiene como objetivo mejorar la calidad de vida y capacidades de las personas que han sufrido alguna amputación de miembro superior.

Una de las características con mayor relevancia es que la prótesis cuenta con un *control proporcional de velocidad* siendo el usuario quien controla los movimientos, por lo tanto, se puede proporcionar una mayor precisión en la mano, facilitando un agarre natural y coordinado, esto se puede realizar debido a que cada dedo está controlado por un motor individual [10].

La prótesis *beBionic* cuenta con características extras, siendo un producto de múltiples articulaciones, su funcionamiento es por medio de señales mioeléctricas (figura 1.3), lo que permite sujetar, presionar, tirar y mover objetos, replicando la forma anatómica de una mano humana. De acuerdo a los manuales técnicos de la prótesis *beBionic* [11], la prótesis cuenta con pulgares ajustable en dos posiciones (oposición y lateral), lo que permite realizar hasta 14 agarres diferentes, de los cuales 8 se pueden pre ajustar por el técnico ortopédico a través de software. Para proteger la prótesis en contra de un corto circuito los dedos no tendrán la capacidad de accionarse, con lo cual se bloquearán y evitarán su funcionamiento. Como desventaja, las piezas deben de ser sustituidas por un técnico [11].

De acuerdo a los manuales técnicos de la prótesis *beBionic* [11], algunos movimientos que se puede realizar es el agarre con 3 dedos, que son el pulgar, el índice y el dedo medio. El prototipo también puede cerrar todos los dedos, hacer agarres, se puede formar la figura de tiro de canica, el pulgar y el dedo índice se pueden tocar, formando la seña "ok", entre otros movimientos. De acuerdo a los técnicos de *Ottobock* [11], esta prótesis no debe de emplearse para conducir vehículos, maquinaria pesada o maquinaria industrial [11].

En cuanto a innovaciones, la empresa *Ottobock* ha implementado una técnica de reconocimiento de patrones, este sistema se llama *control Myo Plus* y brinda a la mano *beBionic*, una mayor naturalidad en sus movimientos [10, 11]. Con este sistema se pueden ejecutar tareas sencillas, debido a que este prototipo cuenta con 14 tipos agarres y posiciones de la mano, estas posturas pueden ser escogidas por el usuario [10]. Estas prótesis están disponibles en dos tamaños diferentes y con tres estilos de muñecas, por lo que se puede personalizar para satisfacer las necesidades de cada persona [10].



(a)



(b)

Fig. 1.3 Diferentes vistas de la prótesis *beBionic*; (a) Prótesis de perspectiva frontal, (b) Prótesis en perspectiva de vista a la muñeca.

### Prótesis LUKE

La prótesis *LUKE Arm* (figura 1.4) es desarrollada por la empresa *Mobius Bionics*, empresa de dispositivos médicos centrada en llevar tecnologías protésicas avanzadas como el bazo *LUKE* a los amputados de las extremidades superiores. El brazo *LUKE* es una prótesis de brazo modular que se puede configurar para diferentes niveles de amputación, incluida la desarticulación transradial (amputación a nivel general de todo un brazo), transhumeral (amputación a nivel del codo) y Radial (amputación a nivel muñeca) [12].



Fig. 1.4 Diferentes articulaciones de prótesis; amputación a nivel hombro, humeral y radial.

De acuerdo a la página web de *Mobius Bionics* [12], el brazo *LUKE* cuenta con hasta 10 articulaciones motorizadas, lo que permite que un amputado tenga un brazo hasta la altura del hombro con movilidad. La prótesis cuenta con un sistema de

control adaptable que permite al brazo ser manipulado por una variedad de dispositivos de entrada, entre estos, se pueden ocupar electrodos de *electromiografía superficial (EMGS)* e interruptores. Además, el sistema permite añadir un control por medio de *Unidades de Medición Inercial (UMI)* que corresponde a un interruptor, este interruptor se usa en los zapatos del usuario, por lo tanto, al momento de oprimirlo y dependiendo del nivel de inclinación del pie es como si el usuario controlara el movimiento de su brazo, asemejando a un joystick de videojuegos, en el cual controla la orientación de la mano. Algunos puntos importantes sobre la *UMI* se expresan a continuación:

1. Controla las funciones del brazo con los movimientos del pie.
2. Se pueden usar hasta 2 *UMI* a la vez junto con otras opciones de entrada.
3. Las *UMI* están selladas y son impermeables.
4. El dispositivo *UMI* puede ser recargado eléctricamente con una plataforma de carga inalámbrica.
5. La función de detección de caminata deshabilita automáticamente las entradas de *UMI* cuando el usuario está caminando y las vuelve a habilitar tan pronto como el usuario deja de caminar. Las *Unidades de Medida Inercial (IMU)* son el dispositivo de entrada propio de *Mobius Bionics* creado exclusivamente para el brazo *LUKE*.

De acuerdo a la página web de *Mobius Bionics* [12], una característica de importancia se encuentra en el control de velocidad proporcional que permite un movimiento más fluido y permitir que el usuario controle que tan rápido se mueven las articulaciones, ajustando la suavidad o agresividad con las que activan la señal de entrada. La muñeca combina los movimientos de flexión y extensión que consiste en poder mover de derecha a izquierda la muñeca, usando la desviación cubital y radial, lo que permite a los usuarios agarrar objetos por encima de la cabeza o por debajo de la cintura mientras mantiene la mano nivelada. Dentro de las configuraciones a nivel transradial y transhumeral se pueden integrar baterías



internas de manera opcional, con el propósito de extender el tiempo de uso de la batería interna

La mano *LUKE* tiene todos los dedos funcionando de forma independiente, esto se debe a que cada dedo tiene un motor que permiten que la mano se ajuste a diferentes objetos en múltiples patrones de agarre, se puede montar un motor vibratorio entre la extremidad perdida y la prótesis del usuario para proporcionar retroalimentación vibratoria (figura 1.5). Esto se puede activar mediante sensores que están posicionados en el pulgar y/o cambiando los modos y agarres. Los ingenieros de *Mobius Bionics* cuenta con un software de realidad virtual, que es útil para configurar los controles de cada brazo, simulado un brazo que permite a los usuarios probar y practicar sus controles mientras realizan los cambios deseados antes de colocarse la prótesis [12].

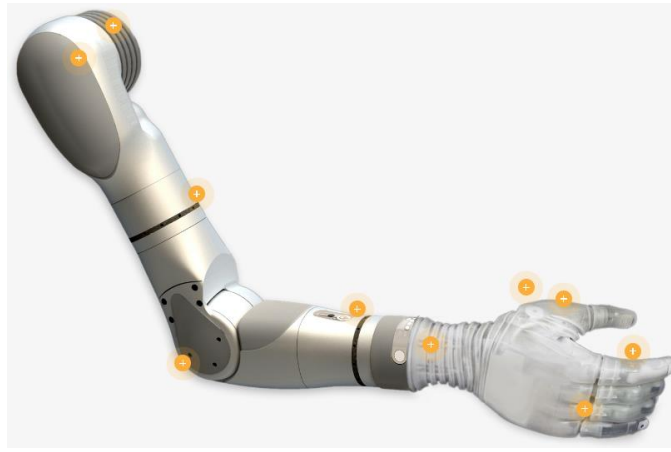


Fig. 1.5 Articulaciones móviles del brazo *LUKE*; los puntos amarillos se muestran las secciones con movimiento o de interés protésico.

#### 1.4 Las señales biológicas

De acuerdo con *Sörnmo y Laguna* [13], el cuerpo humano funciona con señales eléctricas, este fenómeno es causado por reacciones químicas o acústicas, estas señales tienen información dentro su estructura, que no siempre puede ser observada debido a su pequeño voltaje. Por lo tanto, la información debe de interpretarse por medio de algún sistema de procesamiento de señales. Para poder trabajar en este procesamiento, se debe de tener algún conocimiento básico de la

fisiología humana, evitando el riesgo de implementar métodos analíticos que distorsionen o excluyan información importante [14].

Por ende, lo que se busca reducir es el error en la toma de lecturas del procesamiento de señales biomédicas, un factor importante es que, al usar equipos de cómputo, se facilita la observación de las señales, mejorando la precisión y confianza en las mediciones. Además de reducir los errores en la medición, el propio procesamiento de señales se utiliza para implementar métodos que extraen las características de las señales que ayudan a calificar y comprender la información contenida en las señales.

Por ejemplo, de acuerdo con *Sörnmo y Laguna* [13], han demostrado que pequeños cambios en la frecuencia cardíaca no son visibles para el ojo humano, pero contienen información clínica de interés. Para poder observar estas señales, se deben de ocupar las técnicas adecuadas para un buen procesamiento de señales. No hay que olvidar, que, al momento de estudiar las señales biológicas del cuerpo, la mayor cantidad de información que se registra cuenta con varios tipos de ruido e interferencia, este proceso es natural, debido a los propios procesos fisiológicos del cuerpo, así como del ambiente.

En particular, algunos ejemplos de ruido, pueden provenir de los movimientos oculares cuando se trata de estudiar la actividad cerebral, o cuando se usan electrodos de superficie y al momento de tener contacto con la piel, estos tienen deficiencia al mantenerse adheridos al cuerpo. Un ejemplo de ruido ambiental es cuando el generador de funciones o fuentes de voltajes inducen interferencia, debido a la frecuencia de 60 Hz que manejan, interfiriendo con la señal de estudio. Por lo tanto, reducir el ruido es un objetivo fundamental en el procesamiento de señales, con esto se evitará la toma de lecturas erróneas, y, en consecuencia, se puede conocer de mejor forma los procesos fisiológicos del cuerpo [14].

#### 1.4.1 Las señales bioeléctricas

De acuerdo con *Mario López* [15], “la electricidad está presente de muchas formas en la naturaleza y el cuerpo humano no es la excepción”. Por lo tanto, la señal eléctrica que está presente en el cuerpo humano se llama señales bioeléctricas. Anatómicamente, el cuerpo humano es controlado por los impulsos eléctricos que se originan en el cerebro y son generados por las neuronas, creando así la unidad funcional básica del sistema nervioso, por lo tanto, el cuerpo humano funciona por medio de impulsos eléctricos, quien es el encargo de generar las contracciones musculares o en su caso, controlar el ritmo de los latidos del corazón, quien es el encargo de regular la presión arterial.

De acuerdo con el *bachillerato UNAM (B@UNAM)* [16], el sistema nervioso se divide en 2: *Sistema Nervioso Central (SNC)* y *Sistema Nervioso Periférico (SNP)*. El Sistema Nervioso Central (SNC) está formado por el encéfalo, que incluye el cerebro y la médula espinal, mientras que el Sistema Nervioso Periférico (SNP) lo forman la enorme red de neuronas y fibras nerviosas que se extienden desde el SNC hacia todo el organismo, incluyendo la superficie de la piel y las extremidades, desde el cuero cabelludo hasta los dedos de los pies [16].

#### 1.4.2 Las señales electromiográficas (EMG)

De acuerdo con el *Instituto de Medicina Johns Hopkins (Johns Hopkins Medicine)* [17], la *electromiografía (EMG)* cuantifica la actividad eléctrica que está presente ante la estimulación muscular. Esta prueba tiene como objetivo identificar anomalías musculares, para realizar este procedimiento se ingresan varios electrodos hacia la zona muscular de interés, esta actividad se observa por medio de un osciloscopio. Al momento de introducir los electrodos al músculo, estos llegan a registrar cierta actividad eléctrica, sin embargo, esta actividad cesa en unos segundos, para después poder registrar los movimientos que sean de interés, como el músculo

estando en reposo, leves contracciones o en su caso la contracción total del músculo, que es donde se genera la mayor cantidad de energía.

De acuerdo a *Mario López* [15], actualmente existen dos técnicas para el registro de señales electromiográficas: *electromiografía superficial (EMGS)* y *electromiografía invasiva o intramuscular (EMGI)* [15], siendo de interés la electromiografía superficial.

Las señales *EMGS* son de interés debido a que registran la actividad muscular sobre la superficie de la piel que cubre el músculo, como característica principal es que se ocupan electrodos que son adheribles a la piel y proporcionan una estimación aproximada de la actividad del musculo que se está estudiando. En cambio, las señales *EMGI* proporcionan una información de mayor presión desde el momento que son insertadas en el musculo, como ventaja, solo se necesita un electrodo en forma de aguja para registrar toda la actividad muscular, en comparación con los electrodos superficiales, los cuales se deben de colocar en mayor cantidad.

Para evaluaciones médicas como el diagnóstico de enfermedades musculares o patológicas, el uso de las señales *EMGI* ofrecen mayor precisión debido a que acceden a las fibras musculares, proporcionando mejores señales para su estudio, sin embargo, es un proceso doloroso para el paciente. Por ese motivo los electrodos de superficie no infligen dolor al paciente, por lo tanto, son más adecuados para capturar la información [15].

De acuerdo con *Mario López* [15], cuando se trata de electrodos de superficie, se diferencian en dos tipos: electrodos húmedos y secos. Los electrodos húmedos se adhieren a la piel usando un gel conductor como una tapa intermedia para asegurar una buena conductividad entre la piel y el electrodo. Los electrodos secos están hechos de metal y aluminio, por lo que proporciona mayor fiabilidad, sin embargo, antes de ser usados, se recomienda limpiar la piel con alcohol para eliminar el polvo y la grasa del cuerpo, al momento de garantizar esto, las lecturas provenientes del

sensor serán de mayor fiabilidad para el análisis de las señales *electromiográficas* [15,18,19].

Como característica fundamental, el voltaje típico de las señales *EMGS* es de 0 a 6 *mV*, el rango de frecuencias oscila entre 0 Hz y 500 Hz, en algunos casos las impedancias de entrada son aproximadamente de 10  $G\Omega$  [18,19,20].

### 1.5 Procesamiento de señales

El acondicionamiento de señales es un procedimiento de adquisición de datos que se realiza por medio de un dispositivo llamado acondicionador de señal. Esta herramienta transforma un tipo de señal analógica en otra, esta puede ser analógica o digital. El propósito de este método consiste en amplificar la señal y convertirla a otro formato fácil de interpretar y compatible con fines de adquisición de datos o de control de una máquina. Un acondicionamiento de señales ayuda a obtener medidas precisas, como condición esencial para la exactitud de la adquisición de datos o del control de máquina. Este tipo de instrumentos son capaces de efectuar otras funciones adicionales [21].

- A. Conversión de señal: El objetivo principal del acondicionamiento de señales se fundamenta en obtener y modificar una señal eléctrica, la mayoría de los sensores tienen un voltaje alrededor de micro volts a mili volts, por lo tanto, se busca transformar este tipo de voltajes a un voltaje típico que pueda soportar cualquier instrumento que se conecte a él. La conversión de señales se suele utilizar en aplicaciones industriales que emplean un amplio espectro de sensores para efectuar mediciones [21].
- B. Linealización: Este proceso determina si ciertas señales pueden llevar a cabo una linealización, como ejemplo, los termopares tienen una lectura de tipo exponencial, por lo tanto, esta información que proviene del sensor no tiene una correspondencia del todo lineal con la magnitud física. En consecuencia, se lleva a cabo un proceso de interpretación de la señal mediante software.

Este método se emplea para obtener una mayor exactitud, porque no todos los sensores son lineales.

- C. Amplificación: La amplificación de señales es el proceso de incrementar el voltaje que proviene de los sensores, esta amplificación servirá para el procesamiento de la señal por medio de microcontroladores para obtener una señal digital. Hay dos formas de amplificar una señal: aumentando el voltaje de salida con respecto al voltaje de entrada. Por ejemplo, incrementando una señal que tiene un voltaje de 1 micro volt hasta conseguir un voltaje de 5 volts. o en su caso, aumentando la relación de señal-ruido. En el acondicionamiento de señales se utilizan diferentes amplificadores para distintos fines; entre ellos se encuentran los amplificadores de instrumentación, que están optimizados para trabajar con señales de corriente continua, y que se caracterizan por una elevada impedancia de entrada, una alta supresión al factor de rechazo al modo común (*CMRR*) y una elevada ganancia.
- D. Filtrado: El filtrado consiste en eliminar cierta información de frecuencias no deseadas, conservando el ancho de banda de interés y bloqueando todo el ruido. Este ruido puede ser causado por campos magnéticos, movimiento, energía, etc. Los filtros pueden dividirse en componentes pasivos, activos o digitales. Un filtro pasivo utiliza resistencias, capacitores e inductancias con una ganancia unitaria, un filtro activo utiliza resistencias, capacitores, inductancias, amplificadores operacionales y transistores, con una ganancia mayor a 1, los filtros digitales se realizan por medio de software, utilizando programas como *Matlab*, *Python*, *Arduino*, etc., sin necesidad de ocupar elementos físicos como los filtros activos y pasivos.
- E. Evaluación: Los acondicionadores de señales modernos tienen funciones especiales para la evaluación de señales y el procesamiento de datos que se miden. Por lo tanto, brindan monitoreo, evaluaciones rápidas, avisos de forma rápida y directa, mediante una salida eléctrica conmutada. Otras funciones inteligentes adicionales, como los canales de cálculo internos, se encargan de realizar operaciones matemáticas, como sumar señales de

sensores o actuar como un *controlador PID (Controlador Proporcional Integral Derivativo)*.

- F. Interfaces: Los convertidores de señal deben de transmitir las señales de los sensores hasta el control de la máquina, utilizando para ello interfaces y protocolos estándares, siendo de interés las interfaces digitales [21].

## 1.6 Elementos de un sistema de procesamiento de señales EMG

### 1.6.1 Electrodo

De acuerdo con *Alvarez Osorio* [22], a partir del manejo de sensores como electrodos, el voltaje iónico que se encuentra en cada célula del cuerpo se puede transformar en voltaje eléctrico, esta información puede ser leída por una electrónica de instrumentación. Uno de los principales problemas que se tienen en la amplificación de las bioseñales es que los electrodos tienen una alta impedancia debido al material con el que están fabricados, así como la piel a la cual se van a adherir, causando distorsión y ruido en la señal, además de adicionar una componente de *DC (Offset)* que interfiere con la señal.

Cabe recordar que, a bajas frecuencias, la impedancia de polarización incrementa, sin embargo, a altas frecuencias, la impedancia decreta, por este motivo es viable utilizar amplificadores de instrumentación con alta impedancia, puesto que no dejan que la señal se distorsione. Otro problema de los electrodos es que disminuyen el *CMRR* del sistema. Para eliminar el *offset* que se genera por los electrodos se diseña un filtro pasa altas con frecuencia de corte de 0.1 Hz a la salida del amplificador de instrumentación [22].

Un factor importante a tratar es la impedancia que existe entre la piel y el electrodo, regularmente esta impedancia está sometida a un valor aproximado a 10  $k\Omega$ . Para reducir esta impedancia, se utiliza un gel conductor que ayuda a reducir la

resistencia entre la piel y el electrodo, ayudando a adherir los electrodos a la posición deseada. En algunos casos, existe un procedimiento en el cual se pela la piel para mejorar el contacto y reducir la resistencia interna, sin embargo, esto conlleva a generar una incomodidad al usuario porque esto puede provocar dolor, irritación e infecciones. La impedancia límite del electrodo es de  $86.5\text{ k}\Omega$ , por lo tanto, este valor no se debe de considerar en la función de transferencia de un amplificador operacional seguidor de voltaje.

Un detalle que se debe de tener en cuenta es que el ruido térmico generado por la impedancia de los electrodos afecta a la medición de las señales *EMG*, por lo tanto, para evitar esta situación, se debe de reducir la cantidad de electrodos y sus posiciones [22].

### 1.6.2 Etapa de amplificación

Las bioseñales tienen bajas amplitudes y bajas frecuencias, aparte de esto son perturbadas por diferentes fuentes de interferencia tanto ambientales como biológicas, algunas de estas interferencias son creadas por conexiones inapropiadas y ruido por movimiento, por esto, para elegir un amplificador apropiado, es necesario tener en cuenta las características necesarias que debe de tener éste para cada aplicación, consideraciones que hay que tener dependiendo de la fuente de donde se extraiga este voltaje, ancho de banda, impedancia de entrada, el ruido, protecciones y estabilidad de temperatura por fluctuaciones de voltaje [23].

El voltaje típico de las señales *EMGS* es de 0 a  $6\text{ mV}$  por lo que se necesita amplificar la señal unas 5,000 veces, esto se logra con un amplificador diferencial de alta ganancia [20]. La tabla 1.1 muestra los diferentes amplificadores operacionales que existen en el mercado y que son de uso para el acondicionamiento de las *EMG* [22].



<i>Tabla 1.1 Circuitos de Instrumentación; dentro de sus aplicaciones sirven como amplificadores de instrumentación de señales biomédicas.</i>			
Referencia	Tipo de Amplificador	Compañía	Especificaciones
INA326	Instrumentación	Texas Instruments	Alimentación simple, 110 dB, CMRR a G=100, 100 $\mu$ V max offset, MSOP
INA321	Instrumentación	Texas Instruments	Alimentación simple, 94 dB, CMRR a G=100, 500 $\mu$ V max offset, MSOP
INA128	Instrumentación	Texas Instruments	Alimentación dual, 120 dB CMRR, 50 $\mu$ V max offset, 5 nA max de corriente de bias, 700 $\mu$ V
INA118	Instrumentación	Texas Instruments	Alimentación dual, 110 dB CMRR, 50 $\mu$ V max offset, 5 nA max de corriente de bias, 350 $\mu$ V
INA121	Instrumentación	Texas Instruments	Alimentación dual, 106 dB CMRR, 200 $\mu$ V max offset, 4 pA max de corriente de bias
AD620	Instrumentación	Analog Devices	Alimentación dual, CMRR=90 dB con G=1 y CMRR=130 dB con G=1000, 15 $\mu$ V max offset, 0.5 nA max de corriente de bias
AD621	Instrumentación	Analog Devices	Alimentación dual, CMRR=90 dB con G=1 y CMRR=130 dB con G=1000, 15 $\mu$ V max offset, 0.5 nA max de corriente de bias, resistencia interna de ganancia

### 1.6.3 Componentes de filtrado

En la etapa de filtrado se pretende atenuar un rango de frecuencias que no pertenecen a las señales *EMGS*, en este caso, algunas interferencias pueden ser causadas por fuentes de voltaje, mala posición de electrodos o alguna interferencia que genere problemas en la señal. El rango de frecuencias de interés para las señales *EMG* va desde los 20 Hz hasta los 500 Hz [20], por lo tanto, se busca atenuar cualquier rango de frecuencias que no pertenezca a nuestro ancho de banda. De acuerdo con *Alzate Arias* [23], uno de los filtros más comunes a encontrar es el filtro pasa banda activo tipo Butterworth (figura 1.6), que funciona para bajas frecuencias y potencia.

Según *Lorena Alvarez* [22], las topologías más prácticas y conocidas para la elaboración de filtros son: Sallen-Key, realimentación múltiple, estado variable y bicuadrática. Para filtros con un orden mayor a dos, se diseña con varias etapas en cascada con respecto al filtro que se esté usando, sin embargo, al momento de aumentar el orden, las impedancias de acoplamiento se vuelven más sensibles, difíciles de conseguir y no presentan buena respuesta. Por lo tanto, el factor de

calidad (Q) de un filtro es de gran importancia debido a que brinda mayor precisión en la frecuencia de corte.

De acuerdo con *Lorena Alvarez* [22], los parámetros básicos requeridos para diseñar específicamente un tipo de filtro son:

1. Frecuencia de corte: frecuencias a la cual el filtro va a presentar el rechazo de las frecuencias indeseadas.
2. Rizado de pasa banda: la respuesta no ideal de los filtros presenta un rizado en su frecuencia de pasa banda.
3. Frecuencia de rechaza banda
4. Frecuencia de pasa banda
5. Orden del filtro.

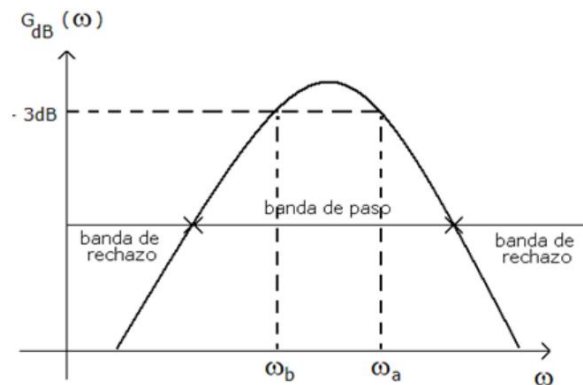


Fig. 1.6 Diagrama de Bode de un filtro Butterworth pasa banda; En el eje horizontal se encuentran las frecuencias de corte que son dadas en Hertz. En el eje vertical se encuentra la ganancia del filtro en  $dB$ .

La mejor opción para diseñar los filtros analógicos es que estén compuestos por el arreglo de amplificadores operacionales, capacitores y resistencias. Dentro de las consideraciones del diseño, se debe de seleccionar un valor de resistencia comercial que se encuentre en el rango de  $1 K\Omega$  hasta  $100 K\Omega$ , estas resistencias deben de ser de precisión para evitar el ruido de Johnson [23]. Para seleccionar los capacitores, se debe escoger un valor comercial que se encuentre en el rango de

1  $nF$  hasta  $\mu F$ ; no es recomendable utilizar capacitores con valores por debajo de 1  $nF$  debido a que pueden inducir capacitancias parasitas; asimismo es preferente que se utilicen capacitores cerámicos o de policarbonato, como recomendación, no se deben usar capacitores electrolíticos ni de tantalio. De acuerdo con Alzate Arias [23], en la elección de los amplificadores operacionales se debe de considerar 3 elementos: la *Ganancia de ancho de banda (GBP)*, *velocidad de respuesta (Slew Rate)* y *bajo ruido (Low Noise)*.

### 1.7 Herramientas y componentes

Dentro de este subtema, se establecerán los programas que se utilizaron para el tratamiento de las señales y la programación necesaria para hacer que funcione la prótesis.

#### 1.7.1 Sensor AD8232

De acuerdo con la hoja técnica del fabricante *Analog Devices* [24], el sensor AD8232 es un módulo que incorpora un bloque de acondicionamiento de señales *ECG (electrocardiograma)*, *EMG (electromiografía)*, entre otras aplicaciones biomédicas. El bloque está diseñado para adquirir, amplificar y filtrar pequeñas señales de voltaje en la presencia de señales con ruido, como el que se genera en el movimiento o en la mala posición de los electrodos. El sensor cuenta con un filtro pasa altas de dos polos, que ayuda a eliminar el ruido de artefactos que integren algún movimiento. Este filtro está integrado a la arquitectura del amplificador de instrumentación, por lo tanto, permite tener una gran ganancia y un filtro pasa altas en una sola etapa, ahorrando así, espacio y costos [24].

Un beneficio del sensor, es que cuenta con un *convertidor analógico - digital (ADC)*, que permite crear una interfaz entre un microcontrolador y la placa. Esta interfaz nos permite manipular por medio de software nuestra señal de interés. Algunas características adicionales se pueden visualizar en la hoja de datos que se encuentra en el anexo, sección B.

### 1.7.2 Software de visualización y manipulación

#### A. Placa y lenguaje de programación Arduino IDE

*Arduino* es un software de distribución libre que añade a una tarjeta de desarrollo un microcontrolador reprogramable y una serie de pines tipo hembra (estos pines están soldados internamente a los pines de las entradas y salidas del microcontrolador), permitiendo conectar sensores y actuadores de forma fácil.

Los diseños que se realizan por medio de *Arduino* pueden funcionar de forma autónoma, esto es posible debido a 2 factores: El primero es porque el microcontrolador almacena en su memoria las instrucciones que se le ordenaron por medio de software. Se debe recordar que un microcontrolador funciona con energía, para que este microcontrolador no dependa de una computadora, la placa de *Arduino* debe de estar funcionando con alguna fuente de voltaje externa. En caso de que la placa de *Arduino* no este conectada a una fuente de voltaje externa, se deberá de conectar de forma alámbrica a una computadora mientras se ejecuta las instrucciones ya programadas. El microcontrolador debe de ser programado por medio de *Arduino*, regularmente el lenguaje de programación que más se utiliza es C++ y C [25].

El microcontrolador que se utilizará en este proyecto será *Arduino UNO* y algunas de sus características de operación se muestran en la tabla 1.2 [26].

Microcontrolador	ATmega328P
Voltaje de operación	5 volts
Pines digitales E/S	14
Pines digitales E/S con PWM	6
Pines de entradas analógicas	6
Corriente continua por pin E/S	20 mA
Frecuencia de Reloj	16 MHz
Memoria Flash	32 KB
SRAM	2 KB
EEPROM	1 KB

## *B. Lenguaje de programación Python*

*Python* es un lenguaje de programación que se utiliza para desarrollar aplicaciones de todo tipo. A diferencia de otros lenguajes de programación como *java* o *.NET*, se trata de un lenguaje interpretado, es decir, que no es necesario compilarlo para ejecutar las aplicaciones escritas en *Python*, sino que se ejecutan directamente por la computadora utilizando un programa denominado interpretador, por lo que es necesario “traducirlo” a lenguaje máquina. *Python* es un lenguaje sencillo de leer y escribir debido a su alta similitud con el lenguaje humano. Además, se trata de un lenguaje multiplataforma de código abierto y, por lo tanto, gratuito, lo que permite desarrollar software sin límites [27].

Con el paso del tiempo, *Python* permite desarrollar aplicaciones en cualquier sistema operativo con una facilidad asombrosa. Una gran cantidad de tecnologías se llevan muy bien con Python debido a su sencillez y a su gran potencia para el tratamiento de datos, algo que sin duda ha hecho resurgir este lenguaje a nivel laboral, donde cada vez son más las empresas que solicitan expertos en *Python* [27]. Su facilidad de escritura y su robustez han convertido a *Python* en el aliado perfecto de la *inteligencia artificial* (IA). Su capacidad de plasmar ideas complejas en pocas líneas, unidas al gran número de frameworks existentes, han hecho que *Python* sea uno de los lenguajes de programación que están impulsando a la IA [27].

### *1.7.3 La transformada Wavelet*

De acuerdo con *Cano, Salcedo y Soto* [28] *la transformada de Wavelet* es una función matemática que permite obtener datos de una señal seccionándolos en pequeños componentes de escala frecuencia – tiempo y poder analizarlos de forma separada.

En comparación con la transformada de Fourier, la wavelet permite trabajar sobre ondas no estacionarias, con discontinuidades o picos, y respecto de otras herramientas que permiten en análisis de este tipo de ondas se distingue por su capacidad de multi resolución.

En ingeniería se representan las señales ya sea en función del tiempo o de la frecuencia. Las técnicas para el análisis de las señales, desde hace tiempo, se basan en los métodos de análisis espectrales en los que se descomponen una onda compleja en una suma lineal de componentes de onda más elementales que en el caso del análisis con Fourier son ondas senos y cosenos. Para muchas señales, el análisis de Fourier es muy útil, ya que el contenido frecuencial de la señal es de gran importancia, sin embargo, se tiene un serio inconveniente. Cuando se produce una transformación al dominio de la frecuencia, se pierde la información del dominio del tiempo. Cuando se observa la transformada de Fourier de una señal, es imposible decir cuando tuvo lugar un determinado evento.

Si las propiedades de la señal no cambian mucho a lo largo del tiempo, es decir, si estamos ante una señal estacionaria, este inconveniente no tiene importancia. Sin embargo, las señales más interesantes contienen características no estacionarias y transitorias, como pueden ser cambios abruptos. Esas características son a menudo las partes más importantes de la señal, y el análisis de Fourier no es adecuado para detectarlos [28].

De acuerdo a Cano, en un esfuerzo para corregir esta deficiencia, *Dennis Gabor* (1946) adoptó la transformada de Fourier para analizar solo una pequeña sección de la señal en un momento dado en una técnica llamada “ventaneo” (“*Windowing*”). La adaptación de Gabor llamada *Transformada de Fourier en Corto Tiempo (TFCT)*, convierte una señal en una función bidimensional de tiempo y frecuencia (Meyer, 1990). Sin embargo, una vez que se elige un determinado tamaño para la ventana, esta será la misma para todas las frecuencias. Ya que muchas señales requieren una aproximación más fiable, es necesaria una técnica en la que se pueda variar el tamaño de la ventana para ganar precisión.

El análisis wavelet constituye el siguiente paso lógico: una técnica de división en ventanas con regiones de tamaño variable. El análisis de wavelet permite el uso de intervalos largos en el tiempo allí donde necesitamos información de baja frecuencia más precisa, y con regiones más pequeñas donde queramos información de alta frecuencia. Se debe tener en cuenta, que el análisis de wavelet no usa regiones de tiempo – frecuencia, sino que usa regiones escala – tiempo. El análisis de wavelet es capaz de revelar aspectos de la señal que otras técnicas de análisis de señal ignoran, aspectos como tendencias, puntos de ruptura y discontinuidades. Además, ya que se logra un punto de vista diferente, el análisis de wavelet puede a menudo comprimir o reducir el ruido de una señal sin una degradación apreciable [28].

De acuerdo con *Orejuela y Ruiz* [29] las clases de la transformada de wavelet son definidas por las propiedades que cada una de ellas poseen, entre las más importantes se tiene: la función de escala, el número de momentos de desvanecimiento, el soporte de contacto, la regularidad y simetría. Dentro de las características de la *Wavelet Mexican Hat* contiene simetría, infinitamente regular, reconstrucción exacta y *TWC (Transformada de Wavelet Continua)*. La wavelet Mexican Hat, se define como la segunda derivada de la función de distribución Gaussiana, puntualizada en la ecuación (1).

$$\psi(t) = \frac{2}{\sqrt{3}\pi^{\frac{1}{4}}} e^{-\frac{t^2}{2}} (1 - t^2) \quad (1)$$

Esta transformada es simétrica y no tiene función de escala asociada. La función de wavelet se ilustra en la figura 1.7 y sus características en la tabla 1.3 [29].

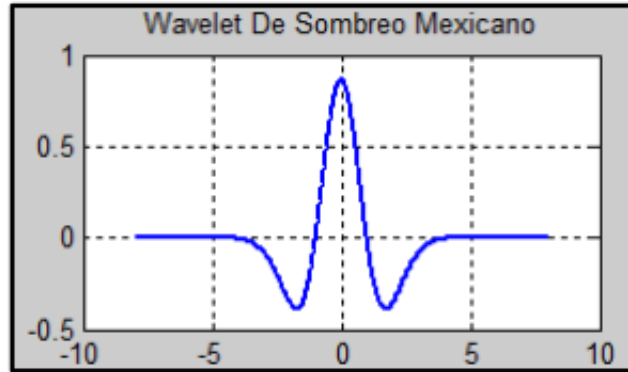


Fig. 1.7 Transformada de wavelet Mexican Hat.

Tabla 1.3 Características propias de la transformada Mexican Hat.	
Nombre corto	mexh
Ortogonal	X
Biortogonal	x
Soporte compacto	X
TWD	X
TWC	Posible
Ancho de ventana	Infinito
Soporte efectivo	[-5, 5]
Simetría	Ok

#### 1.7.4 Controlador ON – OFF con Histéresis.

Un controlador ON – OFF es la forma de control más sencilla, por lo tanto, a la salida del sistema de control, estará en accionamiento o no. Este funcionamiento consiste en que al momento de que la señal de entrada supere el límite de umbral superior o inferior, la salida se accionará solamente cuando se cruce la referencia. Un ejemplo sencillo consiste en el sistema calefacción de una casa, este calentara o enfriara dependiendo de su configuración, en este caso, si la temperatura es menor a 10° centígrados, el sistema empezará a aumentar la temperatura, en caso de que se vea superado por una temperatura mayor a 25° centígrados, el sistema comenzará a enfriar la casa. Por lo tanto, al momento en que la temperatura supera la referencia, la salida cambia de estado, el proceso de temperatura puede ir cíclicamente de abajo hacia arriba y volver hacia abajo [30].



En ciertas situaciones este ciclo ocurre de forma rápida, y para prevenir daños en actuadores, se añade un control ON- OFF histéresis al proceso de control (figura 1.8) [30]. La histéresis mantiene el valor anterior sin ejercer ninguna acción de control mientras el valor de la variable controlada se encuentra dentro de la zona muerta o de histéresis. Un control ON - OFF es usado donde no es necesario un control preciso de la temperatura, donde los cambios de temperatura son extremadamente lentos [30].

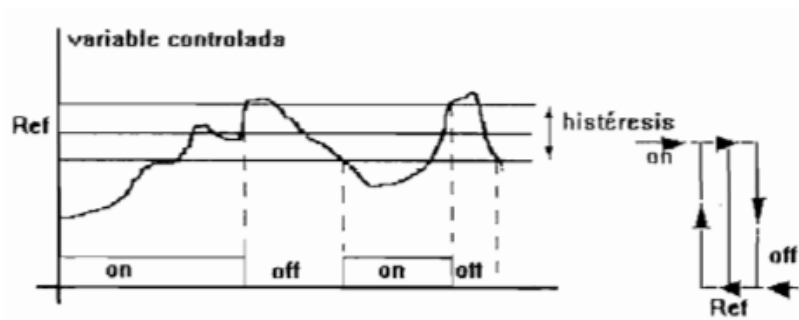


Fig. 1.8 Controlador ON - OFF con histéresis

## **Capítulo 2. Metodología**

En este capítulo se establecerá la metodología de trabajo que consiste en 3 diferentes métodos de trabajo. Cada metodología tendrá su respectivo nombre.

En el primer procedimiento (metodología Arduino - Python) se utilizará el sensor AD8232 para adquirir las señales eléctricas del cuerpo, para después utilizar Arduino y Python para su manipulación de datos. Se optó por utilizar esta interfaz debido a su practicidad. Al momento de que se obtiene la información del sensor, estos datos serán guardados en un formato de archivo txt, donde se crearon 50 archivos y dentro de cada archivo se realizaron 1000 muestras, siendo graficadas en el programa de Excel y observando su comportamiento, sin embargo, ¿Cómo vamos a saber si la información que se contiene es la correcta?, o ¿Cómo podemos comprobar su funcionamiento?

Para esta situación se optó por utilizar una técnica que se llama transformada de Wavelet, más específico se busca encontrar un patrón similar a la técnica “Mexican Hat” para encontrar 5 valores que tengan una simetría entres si, lo que se busca realizar con esto es generar un detector de picos. Se debe de mencionar que este trabajo no involucra el desarrollo de una transformada de wavelet, solo se utiliza como referencia entre 5 puntos para generar un detector de picos.

Para realizar la técnica “Mexican Hat” se siguió utilizando la conexión entre Arduino y Python. Dentro de Python se utilizó un vector de 200 posiciones, estas posiciones guardarán la información proveniente del sensor. Al momento de tener estas posiciones guardadas se buscará el valor máximo, mínimo, promedio y valor al 70 % del valor máximo de la señal, para después encontrar las posiciones de dichos valores. Al momento de encontrar los valores, empezamos a experimentar en qué punto se encuentra la mayor cantidad de información que nos ayudé a encontrar 4 o 5 valores de referencia.

Al momento de obtener estos valores de referencia, estos resultados se seguirán guardando en un archivo txt. Los cuales graficaremos para obtener sus valores promedio, mínimo, máximo, valores al 70 % del valor máximo e histogramas. Añadiendo a esto, se realizaron diferentes pruebas de movimiento, estas secuencias consistían en abrir o cerrar la mano a diferentes velocidades: rápido, lento o un movimiento constante e identificar su comportamiento. Este procedimiento ayudará para encontrar una semejanza entre nuestra señal biológica y la técnica de Mexican Hat.

La figura 2.1 demuestra la metodología Arduino – Python

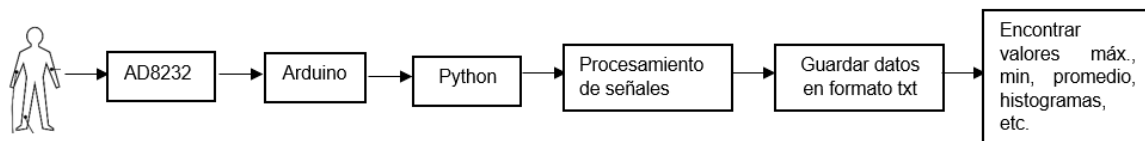


Fig. 2.1 Diagrama experimental. Dentro de este diagrama se visualiza de forma rápida el primer método.

En la segunda metodología (Metodología Controlador ON - OFF) se utilizará el sensor AD8232, Arduino y el prototipo de prótesis. Los electrodos del sensor AD8232 se colocarán en el antebrazo del usuario, estos electrodos funcionarán como transductores de movimiento de apertura y cierre de la mano, el sensor tomará la información y será mandada a Arduino. En Arduino se utilizará la técnica de control ON – OFF con histéresis debido a que como las señales biológicas oscilan en cada momento, esto puede generar cambios rápidos en las lecturas. Estos cambios que se generan hacen que los servomotores de la prótesis se accionen en cada momento, no se queden estáticos o en su caso no siguen la instrucción que se desea.

Por lo tanto, el control ON – OFF con histéresis será de utilidad en este experimento debido a que la prótesis solo se accionará cuando llegue a su umbral superior o umbral inferior y utilizaremos como referencia el valor promedio que proviene de los resultados de la metodología Arduino – Python.

La figura 2.2 muestra el diagrama de bloques experimental del controlador ON – OFF con histéresis. Un punto importante de aclarar es que el movimiento solamente se enfocara en abrir o cerrar los dedos de la mano, debido a que este prototipo de prótesis no cuenta con algún movimiento en el antebrazo o muñeca.

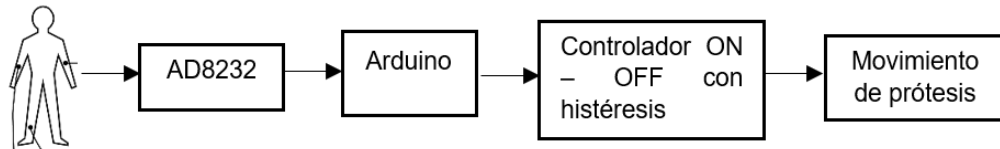


Fig. 2.2 Diagrama de bloques experimental del controlador ON – OFF con histéresis.

En el tercer método (metodología PC – Prótesis) es un complemento de funcionamiento para la segunda metodología. Este proceso consiste en la utilización de una computadora, Arduino y el prototipo de prótesis. Esto se realizó para generar mayor cantidad de movimientos. El movimiento principal que se generará es el abrir y cerrar la mano de forma automática, de forma consecutiva, cada dedo individualmente repetirá la misma secuencia hasta que se genere alguna interrupción desde el Monitor Serie de Arduino. Con lo cual se tendrán 5 movimientos adicionales por cierto tiempo (figura 2.3).

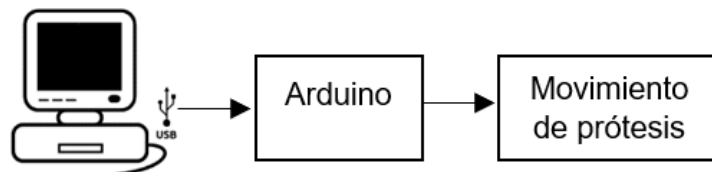


Fig. 2.3 Diagrama de bloques de la metodología auxiliar, dentro de esta metodología la prótesis contara con un movimiento automático repetitivo hasta que se realice una interrupción. Cada interrupción tendrá un movimiento, dando un total de 5 interrupciones.

Cada una de las técnicas redactadas se explicarán de forma adecuada en el capítulo 3, correspondiente a experimentación y resultados.

## 2.1 Adquisición de señales

Para adquirir las señales eléctricas del cuerpo humano se utilizaron electrodos superficiales. A partir de estos electrodos y de acuerdo a *SENIAM* [30] (*Surface ElectroMyoGraphy for Non-Invasive Assessment of Muscle – Evaluación Muscular de Electromiografía Superficial no Invasiva*) existen diferentes formas de conectar los electrodos, a esta configuración se le llamará derivaciones o triángulo de Einthoven. Einthoven determinó que el cuerpo humano es un conductor eléctrico y que el corazón genera corriente, y concluyó que este comportamiento se podía representar mediante un triángulo imaginario con diferencias de potencial entre sus extremos [31].

De acuerdo a *SENIAM*, se recomienda que los electrodos se localicen con una distancia entre el centro de ellos de 20 mm [32]. Cuando el músculo sea pequeño, la inter distancia no debe de exceder  $\frac{1}{4}$  de la longitud del músculo (Figura 2.4). Para los músculos de los bíceps en particular se recomienda:

1. El electrodo de referencia en la muñeca.
2. El brazo en ángulo recto amortiguando el codo en la mano libre.

Dentro de nuestro caso de estudio se determinó posicionar dos electrodos en el antebrazo izquierdo y nuestra referencia a tierra será posicionada en el codo, como lo muestra la figura 2.5.

La mayoría de electrodos que tienen alguna referencia a tierra se colocan en punto donde no exista gran cantidad de movimiento.

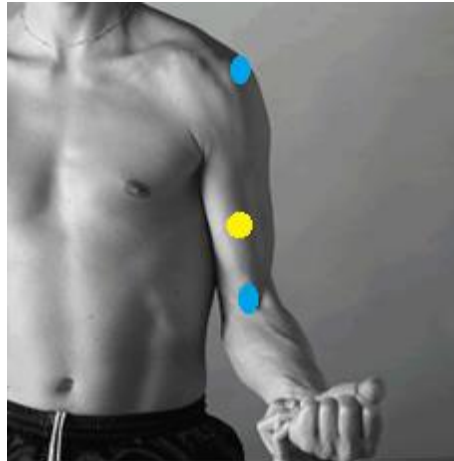


Fig. 2.4 Posición de electrodos en el musculo del bíceps; Un electrodo en el hombro, dos electrodos en el bíceps.



Fig. 2.5 Posición de electrodos; Dos electrodos posicionados en el antebrazo y un electrodo de referencia a tierra en el codo.

El circuito AD8232 es un circuito que se utiliza para realizar pruebas de electrocardiogramas (señales *ECG*), sin embargo, debido a la practicidad de uso que se tiene con este circuito, se optó por utilizarlo en pruebas de electromiografía. Entre las ventajas que contiene, es que este módulo cuenta ya con toda una electrónica de instrumentación interna, ahorrando la etapa de diseño de acondicionamiento de señales [32]. Este diseño se podrá localizar en el anexo B.

Este módulo viene integrado con un cable de tres electrodos que nos ayudará en la toma de lecturas; el circuito tiene una entrada de voltaje de 3.3 volts, voltaje que puede ser suministrado por Arduino Uno, lo cual nos ayuda a tener una interfaz con el microcontrolador. En nuestra experimentación solamente ocuparemos 3 de estos pines, correspondientes a GND, VCC y OUTPUT. Facilitando la toma de lecturas y teniendo un circuito “todo en uno” (figura 2.6).

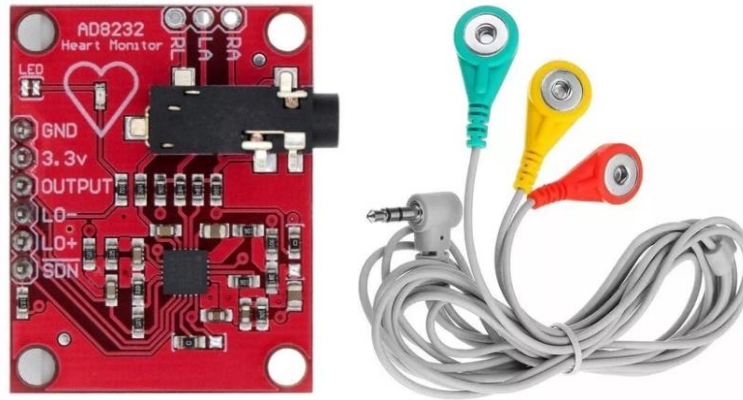


Fig. 2.6 Circuito AD8232.

## 2.2 Configuración de Arduino para adquisición de datos

Para la adquisición de datos de la metodología Arduino – Python se tuvo que realizar la siguiente conexión entre Arduino y el circuito AD8232 [34], esta conexión se puede visualizar en figura 2.7.

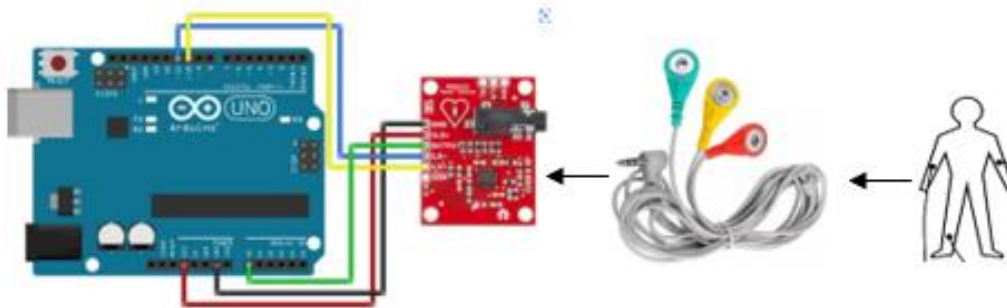


Fig. 2.7 Configuración electrónica entre Arduino, el circuito AD8232, los electrodos y el cuerpo humano; estos elementos son necesarios para su correcto funcionamiento. La alteración del sistema arrojaría una lectura nula.

Para un correcto funcionamiento del sistema de la figura 2.7, se deben de seguir ciertas consideraciones:

1. La piel debe de limpiarse con alcohol.
2. Los electrodos de color rojo y amarillo deben de colocarse como lo muestra la figura 2.5. El electrodo de color verde debe de colocarse en una posición de nulo movimiento, en este caso el codo.
3. Se debe de conectar el Jack de entrada de los electrodos al circuito AD8232 y seguir las conexiones que se observan en la figura 2.7.
4. Se ejecutará el programa A.1 que se ubica en el anexo A de códigos de sistema, este código realizará la lectura de datos provenientes del circuito, lecturas que se observan en la figura 2.8.

En la tabla 2.1 se muestra el pseudocódigo del programa A.1 y su código de funcionamiento se encontrará en el anexo A (códigos del sistema).

Una de las ventajas de utilizar el sensor AD8232 es que está configurado para trabajar únicamente con señales positivas debido a que Arduino está configurado de la misma forma.

<i>Tabla 2.1 Pseudocódigo Interfaz Arduino para Adquisición de Datos.</i>
<i>1. Se configuran los pines 10 y 11 como entradas para la detección de pulsos.</i>
<i>2. Si no se detecta algún valor, imprime un signo de admiración.</i>
<i>3. Pero si hay detección, imprime los valores que se reciben en la entrada A0.</i>

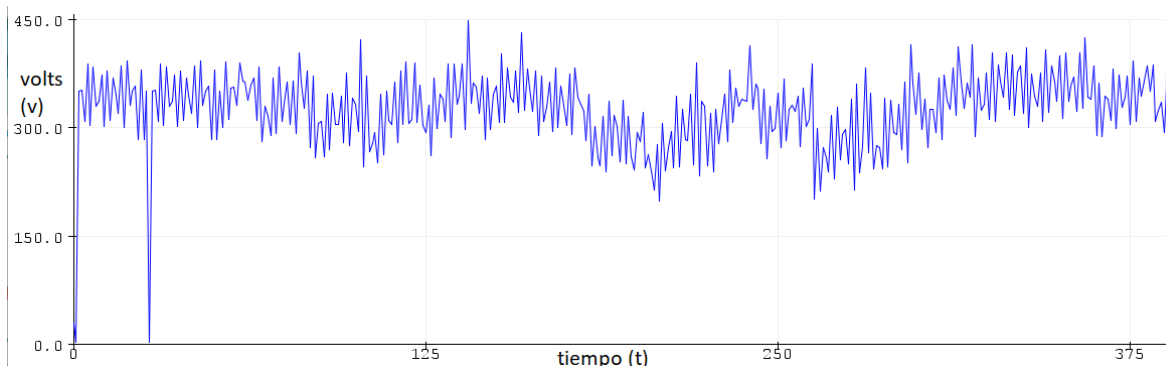


Fig. 2.8 Señal *EMG* proveniente del circuito AD8232; siendo el eje horizontal x, el tiempo de muestreo, con el voltaje de salida en el eje vertical y en mili volts.



### 2.3 Metodología Arduino-Python

La metodología Arduino – Python será de gran utilidad debido a que sentará las bases del trabajo, con esta metodología comprobaremos a través de la forma de onda de la transformada de Wavelet que nuestro sistema replica la forma de onda Mexican Hat y que esta información conllevará a formar un detector de picos. Este detector de picos se auxiliará por la información proveniente Python y por el tratamiento de datos que se realizará por medio de Microsoft Excel.

Para la interfaz entre Arduino y Python se seguirá implementando las conexiones que se muestran en la figura 2.7, así como las indicaciones de la tabla 2.1, hasta el punto número 3. Al momento de tener el circuito ya armado, se comenzará a seguir ciertos pasos para la obtención de los valores provenientes de Arduino, toda esta información se guardará en un archivo txt. Se realizaron 50 archivos txt donde cada archivo cuenta con 1000 datos. El pseudocódigo que explica el proceso se encuentra en la tabla 2.2, y su código de funcionamiento se encontrará en el anexo A (códigos del sistema), figura A.2.

<i>Tabla 2.2. Pseudocódigo de Información proveniente del sensor AD8232</i>
<ol style="list-style-type: none"><li>1. <i>Se importan las clases serial, time y numpy.</i></li><li>2. <i>Se creará una carpeta en nuestra computadora por medio de comandos en donde se guardarán los datos enviados por Arduino.</i></li><li>3. <i>Se establecerá la comunicación con la placa Arduino.</i></li><li>4. <i>Se creará un vector de 1000 posiciones que se adecuarán al tamaño de muestras que se tomen.</i></li><li>5. <i>Se creará un contador de 0 a 1000 para la obtención de muestras.</i></li><li>6. <i>Se transforman los valores a un valor tipo string y se escriben los datos en el archivo txt ya creado.</i></li><li>7. <i>Los archivos en formato txt se graficarán en Excel.</i></li></ol>

Estos pasos se seguirán para obtener las señales proviniendo del circuito AD8232, como se observa en la figura 2.9.

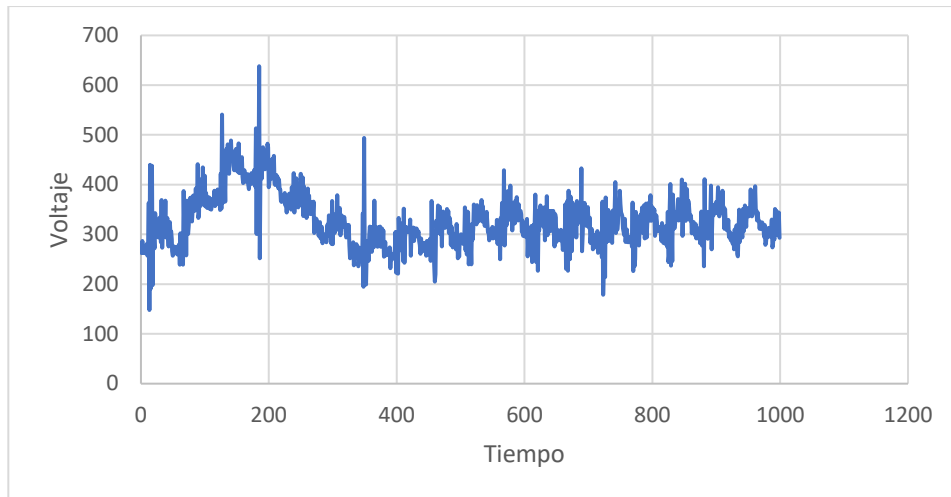


Fig. 2.9 Señal *EMG* proveniente del sensor AD8232.

Al momento de observar la figura 2.8 y 2.9, nuestra señal tiene un comportamiento similar, sin embargo, ¿cómo sabemos que la información que proviene del sensor es la correcta? Para comprobar que la señal *EMG* tiene un comportamiento simétrico como el que se muestra en la figura 2.10, usaremos la forma de onda de la transformada de Wavelet, más específico la técnica “Mexican Hat”. Lo que se trata de buscar es un detector de picos, como se observa en la figura 2.11. Por lo tanto, no nos enfocaremos en desarrollar de forma matemática la transformada de Wavelet.

Para determinar un detector de picos, lo que se buscará es encontrar los puntos máximos, mínimos, valores promedio y un valor al 70 % del valor máximo de la señal. Se escogió el 70 % de la señal debido a que existe una mayor cantidad de datos que se repiten, con esto, estaríamos generando 5 puntos de referencia para encontrar un detector de picos.



Fig. 2.10 Característica de la señal *EMG*. En color naranja se visualiza el valor promedio, en color verde se visualiza el 70 % del valor máximo, En rojo se visualiza el valor máximo y en azul el valor mínimo de un detector de picos.

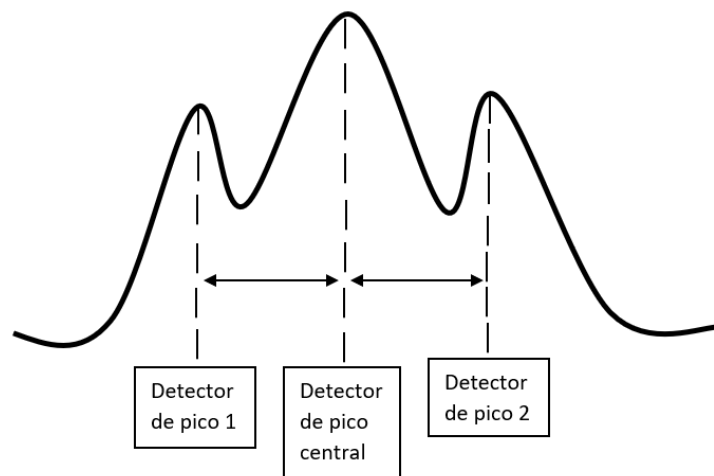


Fig. 2.11 Detector de picos; del lado izquierdo y derecho se encontrarán los valores al 70 % del valor máximo y en el centro se buscará el valor máximo de la señal *EMG*.

En este proceso de experimentación nuestro contador ( $n$ ) se configurará para tomar 200 muestras. La forma en como encontraremos los detectores de pico se resume al pseudocódigo de la tabla 2.3. y su código de funcionamiento se encontrará en el anexo A (códigos del sistema), figura A.2.

*Tabla 2.3 Pseudocódigo detector de picos.*

- 1. Se importan las clases serial, time y numpy.*
- 2. Se establecerá la comunicación con la placa Arduino.*
- 3. Los datos provenientes de Arduino se guardarán en un archivo txt.*
- 4. Se creará un vector de 200 posiciones, que se adecuarán al tamaño de muestras que se tomen.*
- 5. Se creará un contador de 0 a 200 para la obtención de muestras provenientes del sensor AD8232.*
- 6. Se transforman los datos a un valor tipo string y se imprimen los valores del vector.*
- 7. Encontraremos el valor máximo.*
- 8. Buscamos la posición del valor máximo.*
- 9. Encontramos el valor mínimo.*
- 10. Encontramos el valor de la señal al 70 % con respecto al valor máximo.*
- 11. Buscamos el índice del valor máximo.*
- 12. Encontraremos él  $\pm 5\%$  del valor al 70 %.*

Para complementar el experimento, tomamos diferentes lecturas con respecto a la velocidad de movimiento. En este caso, el abrir y cerrar la mano, estos movimientos se representarán de la siguiente manera: Lento – Lento (L–L), Lento – Rápido (L–R), Lento – Normal (L–N), Rápido – Rápido (R–R), Rápido – Normal (R–N), Normal – Normal (N–N).

Al momento de obtener los datos, usaremos nuestra información para obtener gráficas e histogramas en Excel. Dichas Gráficas ayudarán a observar, analizar y encontrar la mayor concentración de valores. Con estos datos se llegaría a una mejor toma de decisiones para gestionar y demostrar que nuestro detector de picos funciona, que la información proveniente del sensor AD8232 y su manipulación de datos por Excel lo respalda. Estos valores serán necesarios para utilizar el controlador ON – OFF con histéresis.

## 2.4 Controlador ON – OFF con histéresis

El controlador ON – OFF con histéresis se ocupará debido a que nos permite que el sistema no tenga conmutaciones en cada momento, esto es debido a que la señal *EMG* oscila como se observa en la figura 2.8, estas oscilaciones provocan que se accionen los servomotores en cada momento y en consecuencia genera inestabilidad. Lo que se busca, es que se accionen los servomotores al momento de que los valores provenientes del sensor se vean superados por los límites superiores e inferiores de histéresis, por lo tanto, la prótesis abrirá o cerrará la mano. La figura 2.12 muestra el diagrama experimental de nuestra metodología.

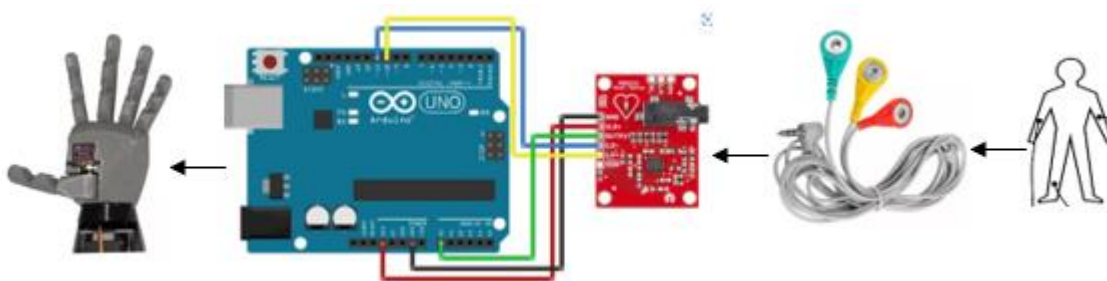


Fig. 2.12 Diagrama experimental del controlador ON – OFF con histéresis.

El pseudocódigo que explica su funcionamiento se observa en la tabla 2.4 y su código de funcionamiento se encontrará en el anexo A (códigos del sistema), figura A.3.

Tabla 2.4 Pseudocódigo del controlador ON – OFF con histéresis.
<ol style="list-style-type: none"><li>1. Se importa la librería servomotor.</li><li>2. Se crea una variable por cada dedo.</li><li>3. Se establecerá un punto de referencia y un valor de histéresis.</li><li>4. Se establecerán los límites superiores e inferiores.</li><li>5. Se gestionan los pines de salida.</li><li>6. Los valores provenientes del sensor se empiezan a registrar.</li><li>7. Si el valor es mayor al límite superior, se cierra la mano.</li><li>8. Si el valor es menor al límite inferior, se abre la mano.</li></ol>

## 2.5 Control PC – Prótesis

Como método auxiliar se implementó la metodología de control PC – Prótesis para incrementar la cantidad de movimientos, la figura 2.13 muestra el diagrama experimental a desarrollar. El método consiste en la apertura y cierra de la mano, para después generar una secuencia de cierre y apertura por cada dedo, hasta reiniciar su ciclo. El ciclo puede ser interrumpido al momento de ingresar un valor por medio de la herramienta “Serial Plotter”, herramienta que se encuentra en el software de Arduino. Esta prótesis cuenta con 5 interrupciones que van de la letra a hasta la e. El pseudocódigo que explica esto se encuentra en la tabla 2.5, y el código se encontrará en el anexo A (códigos del sistema), figura A.4.



Fig. 2.13 Diagrama experimental del control PC – Prótesis.

Tabla 2.5 Pseudocódigo del control entre Laptop – Prótesis por medio de Arduino

1. Se importa la clase servo y se crea la variable tipo char llamada letra.
2. Se crean 5 objetos que corresponderán a cada dedo de la mano y se posicionaran en un ángulo de 0 grados.
3. Se establecen los pines de salida.
4. Si se ingresa la letra “a” por medio de serial plotter, los dedos realizarán la figura del símbolo de “paz.
5. Si se ingresa la letra “b” los dedos realizarán el símbolo de “rock”.
6. Si se ingresa la letra “c” los primeros 3 dedos estarán abiertos y los restantes se cerrarán.
7. Si se ingresa la letra “d” todos los dedos se mantendrán cerrados excepto el dedo anular.

8. Sino se recibe ninguna instrucción la mano tendrá un movimiento de abrir y cerrar los dedos, para después abrir y cerrar cada dedo hasta que se genere una interrupción.

## 2.6 Configuración angular de los servomotores

Para la configuración de los servomotores todos se alinearon en una posición angular de cero grados, como lo muestra la figura 2.14. Cada dedo tendrá cierta extensión y contracción, esto depende de la flexibilidad que tengan los hilos de la prótesis y los resortes ubicados en los nudillos de la mano, como se observa en la figura 2.15. Cada uno puede variar dependiendo de la flexibilidad de los materiales.



Fig. 2.14. Posición inicial de 0 grados. Cada servomotor tendrá esta configuración. El silicón ubicado en la polea sirve para tener un mejor agarre entre el servomotor y la pieza móvil.

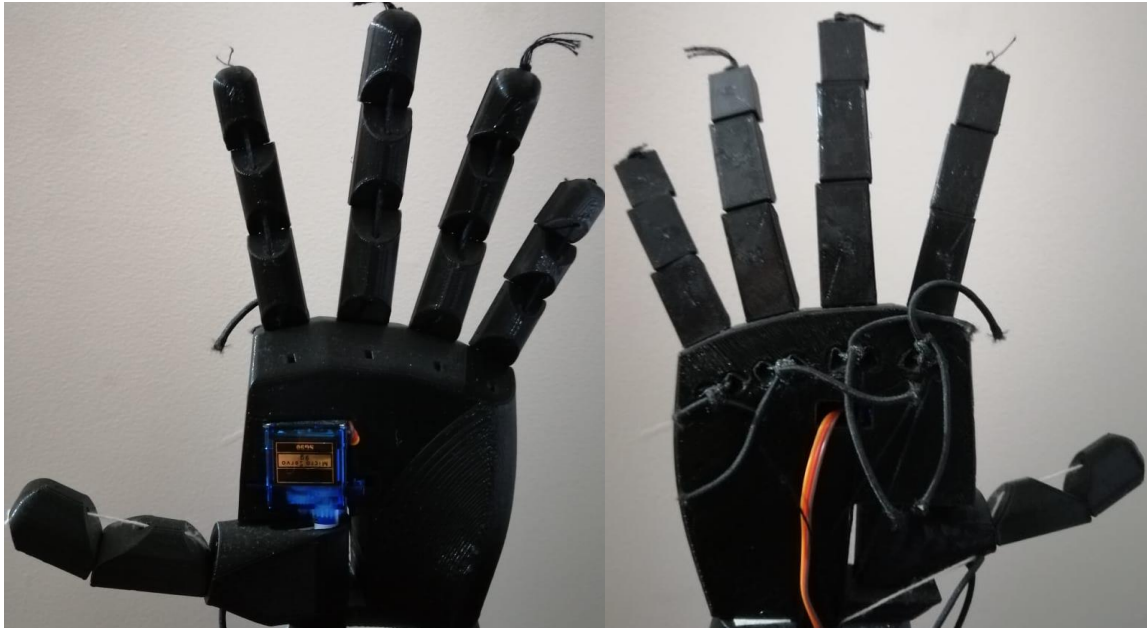


Fig. 2.15 La prótesis cuenta dos materiales para sujetar los dedos: Hilos de costura en la parte interna de las falanges y resorte elástico que sujeta los dedos en los nudillos.

En la tabla 2.6 se visualiza el pseudocódigo de la posición angular de los servomotores, este ángulo gestionará la posición de los dedos de la prótesis.

<i>Tabla 2.6. Pseudocódigo de Posicionamiento de Dedos</i>
<ol style="list-style-type: none"> <li>1. <i>Definimos como salida el PIN 4 y 5 (PIN 4 es un led, PIN 5 es un servomotor).</i></li> <li>2. <i>Gestionamos las posiciones iniciales para ambos en 0.</i></li> <li>3. <i>Ingresamos manualmente la posición del servomotor.</i></li> <li>4. <i>Al momento de encontrar el ángulo adecuado proseguimos a localizar los siguientes ángulos.</i></li> </ol>

En la tabla 2.7 se establece el valor angular de los servomotores, una de las ventajas al haber acomodado los servomotores a una posición inicial de 0 grados, es que ayudarán a tener una simetría en cada posición, excepto en el dedo pulgar, debido a que no tiene la misma posición física que los demás dedos.



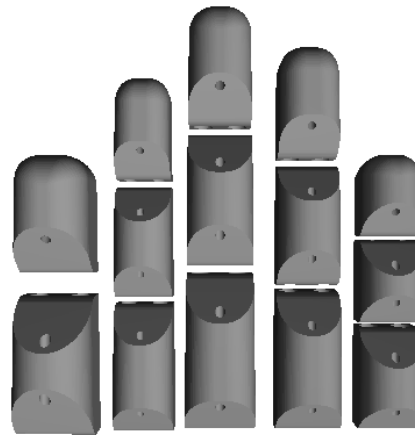
Por lo tanto, su configuración se reflejará en la tabla 2.7 y su código se encontrará en la sección de anexo A (códigos del sistema), figura A.5.

Dedo	Abierto (°)	Cerrado (°)
Pulgar	180	120
Índice	0	150
Medio	0	150
Anular	0	150
Meñique	0	170

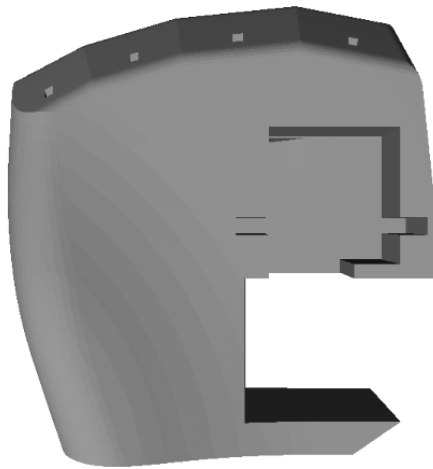
### *2.7 Prótesis de mano*

Para la creación de la prótesis de mano en 3D se prefirió conseguirla con un vendedor de mercado libre, por lo tanto, me siento agradecido con el señor Francisco Pacheco Díaz, esta prótesis no fue diseñada por su servidor, por lo tanto, conseguimos evitar la etapa de diseño en 3D. Esta prótesis de mano izquierda junto con sus piezas se imprimió en un material llamado *PLA (ácido poliláctico)*.

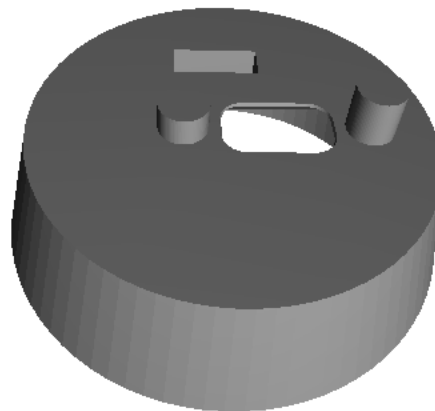
Las piezas fundamentales de la prótesis se visualizarán en las figuras 2.16, 2.17 y 2.18.



(a)



(b)

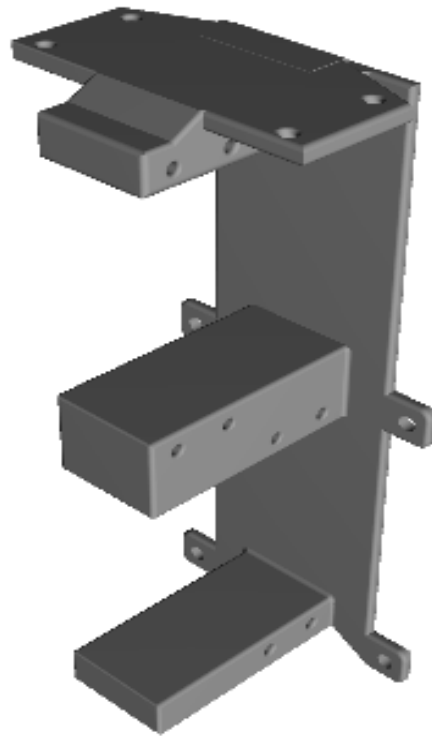


(c)

Fig. 2.16 Materiales de la prótesis de mano (a) dedos, (b) palma, (c) base.



(a)



(b)

Fig. 2.17 Materiales del antebrazo de la prótesis (a) antebrazo, (b) placa de sujeción de los servomotores.

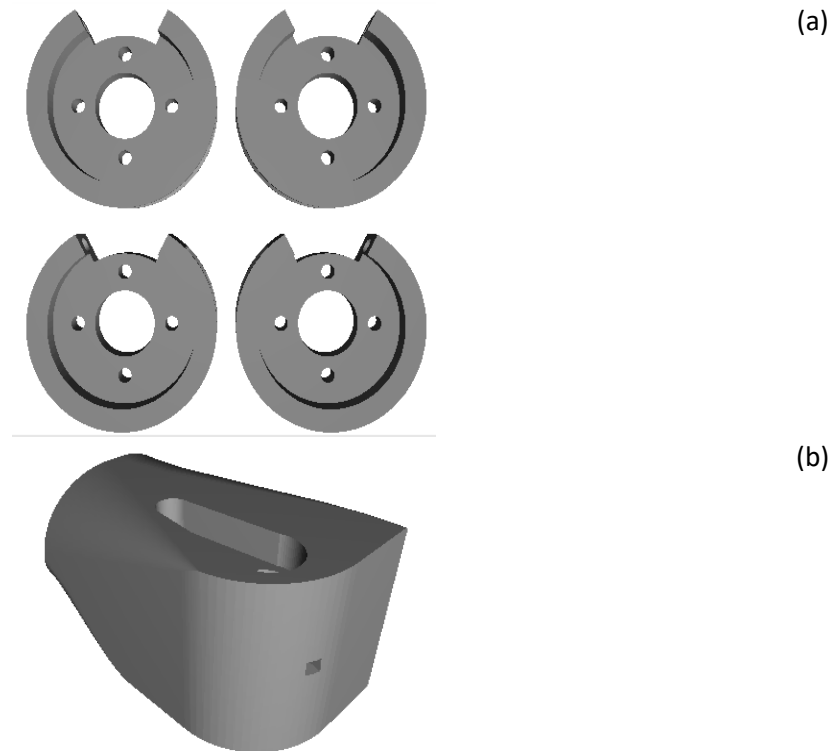


Fig. 2.18 Materiales de prótesis (a) Poleas para servomotores, (b) falange del dedo pulgar.

Para el armado de la prótesis se utilizó hilo de costura, esto se escogió debido a que el diámetro interno de los dedos es milimétrico, por lo tanto, cuentan con un diámetro similar (como se observa en la figura 2.16(a), este hilo partirá desde la punta de los dedos hasta las poleas de los servomotores, como se observa en la figura 2.15. Para sujetar los dedos y que tuvieran firmeza, se ocupó hilo resorte, esto debido a que el resorte es un poco rígido, pero su flexibilidad sirve para cerrar y abrir las falanges de los dedos (figura 2.15).

A continuación, se mostrará el armado total de la prótesis en la figura 2.19.



(a)



(b)

Fig. 2.19 Armado de prótesis, (a) Vista anterior, (b) Vista lateral izquierda de los servomotores.

## 2.8 Salidas electrónicas del sistema

Las salidas electrónicas que tiene nuestro sistema son 5 servomotores que serán utilizados para mover cada dedo independientemente, los materiales a utilizar fueron dos servomotores MG995, dos servomotores MG996R y un servomotor SG90. El servomotor SG90 se utilizará para mover el pulgar, esto debido a su facilidad como pieza clave de unión entre el dedo y la palma. Los servomotores MG995 y MG996R se utilizarán para mover los 4 dedos faltantes debido a su torque y facilidad para poder sujetarlos dentro de la placa. Esta placa funcionará como un hueso dentro del prototipo, como lo muestra la figura 2.20.

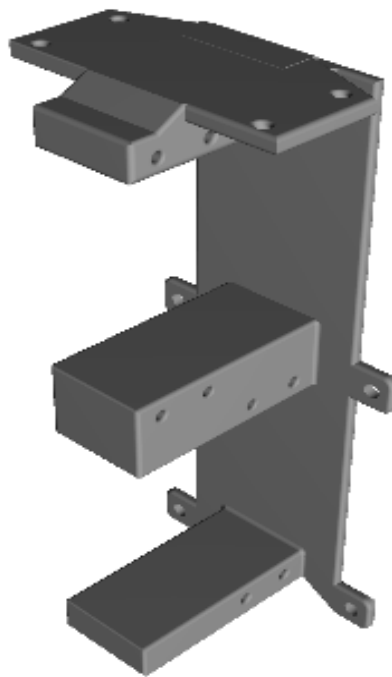


Fig. 2.20 Base de servomotores.

## Capítulo 3. Experimentación y Resultados

En este capítulo se estará redactando acerca de los resultados experimentales, desde la adquisición de las señales, el procesamiento de datos a través de Python – Arduino, y pruebas experimentales.

### 3.1 Adquisición de las señales utilizando Arduino - Python

La adquisición de señales se llevó por medio del circuito AD8232, siguiendo la metodología prevista en el capítulo anterior. Los datos provenientes del sensor AD8232 pasarán por medio de Arduino para después ser tratados en Python, como lo muestra la figura 3.1.

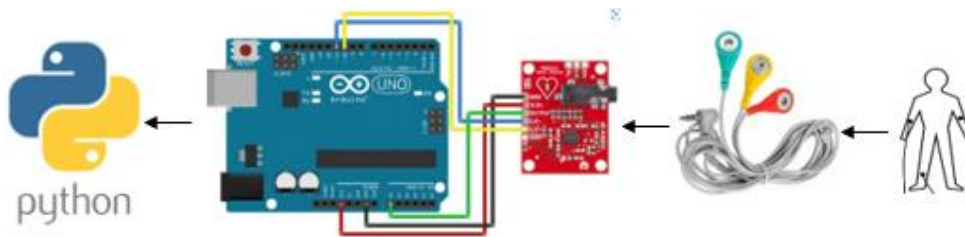


Fig. 3.1 Diagrama experimental adquisición de datos Arduino – Python.

Para comprobar el funcionamiento correcto del circuito AD8232, se utilizó un osciloscopio para comprobar la señal *EMG* proveniente de nuestro cuerpo, como lo muestra la figura 3.2

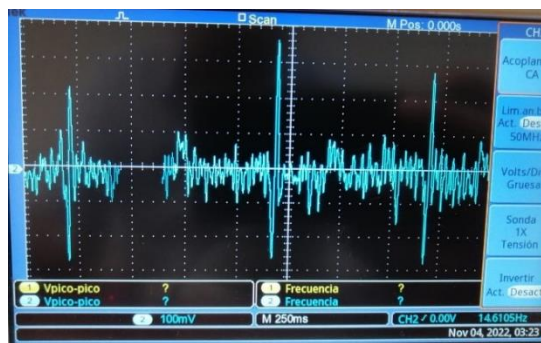
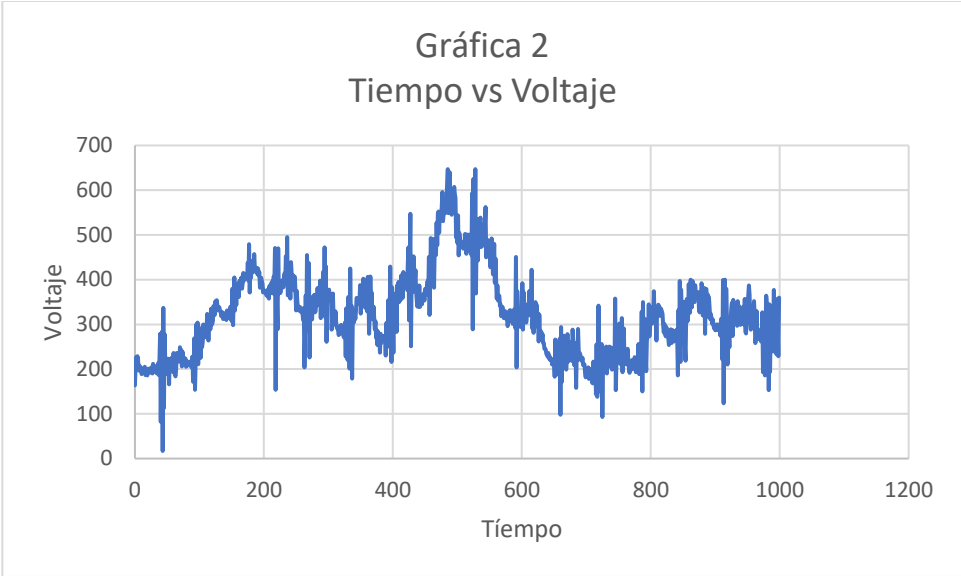
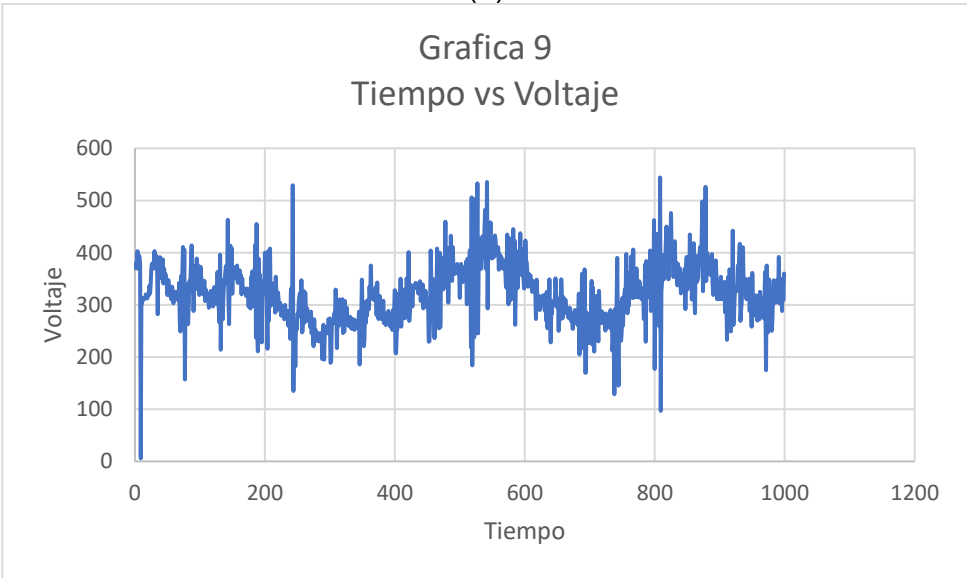


Fig. 3.2 Adquisición de señales *EMG* mediante la placa AD8232, utilizando un osciloscopio como medio de observación.

Nuestros datos se guardarán en un archivo en formato txt, donde se graficarán estas señales bioeléctricas. Se realizaron 50 archivos txt donde cada archivo cuenta con 1000 datos, como se observa en la figura 3.3.

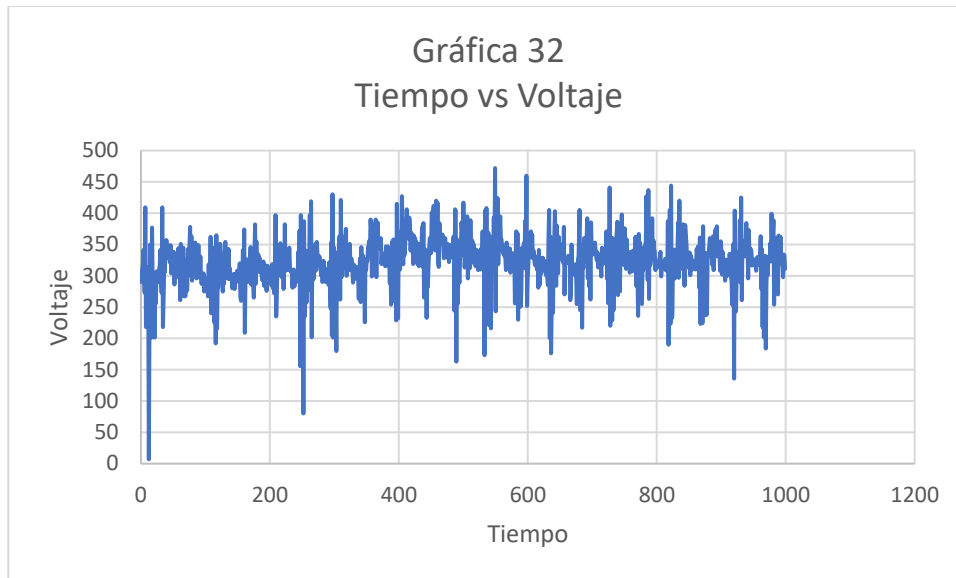


(a)



(b)





(c)

Fig. 3.3 Señal *EMG* proveniente del sensor AD8232. (a) Gráfica 2, (b) Gráfica 9, (c) Gráfica 32.

### 3.2 Clasificación de señales

Para identificar que la información proveniente de Arduino es correcta se realizará el procesamiento de datos para obtener los valores fundamentales de la señal, en este caso el valor promedio, máximo, mínimo, valores al 70 % con respecto al máximo. Esto se realizó para conocer el comportamiento de nuestra señal. Al momento de conocer este comportamiento, sabremos identificar que nuestra señal *EMG* funciona con pulsos positivos (como se observa la figura 2.9, capítulo 2). Al encontrar estos pulsos se identificarán patrones de comportamiento que se asemejan a la forma de onda de una transformada de Wavelet, conocida como “Mexican Hat”. Lo que se trata de buscar es un detector de picos, como se observa en la figura 2.10 (capítulo 2). Por lo tanto, no nos enfocaremos en desarrollar una transformada de Wavelet.

El detector de picos nos ayudará a obtener la información suficiente para desarrollar nuestro controlador ON – OFF con histéresis.

En este proceso de experimentación, se formaron 36 archivos txt, con 200 datos cada uno. Nuestro contador (n) se configurará para tomar 200 muestras. La forma en como encontraremos los detectores de pico se resume al pseudocódigo de la tabla 3.1. y su código de funcionamiento se encontrará en el anexo A (códigos del sistema), figura A.2.

<i>Tabla 3.1 Pseudocódigo detector de picos.</i>
<ol style="list-style-type: none"> <li>1. <i>Se importan las clases serial, time y numpy.</i></li> <li>2. <i>Se establecerá la comunicación con la placa Arduino.</i></li> <li>3. <i>Los datos provenientes de Arduino se guardarán en un archivo txt.</i></li> <li>4. <i>Se creará un vector de 200 posiciones, que se adecuarán al tamaño de muestras que se tomen.</i></li> <li>5. <i>Se creará un contador de 0 a 200 para la obtención de muestras provenientes del sensor AD8232.</i></li> <li>6. <i>Se transforman los datos a un valor tipo string y se imprimen los valores del vector.</i></li> <li>7. <i>Encontraremos el valor máximo.</i></li> <li>8. <i>Buscamos la posición del valor máximo.</i></li> <li>9. <i>Encontramos el valor mínimo.</i></li> <li>10. <i>Encontramos el valor de la señal al 70 % con respecto al valor máximo.</i></li> <li>11. <i>Buscamos el índice del valor máximo.</i></li> <li>12. <i>Encontraremos él <math>\pm 5\%</math> del valor al 70 %.</i></li> </ol>

Al inicio de la experimentación se buscaron valores entre el 70, 75, 80, 85 y 90 por ciento del valor máximo, para cada valor de porcentaje en específico se realizarán 11 eventos de prueba, en los cuales probaremos su tasa de acierto y fracaso. Estas pruebas se visualizarán en la tabla 3.2. Algunos datos que aparecen en tabla tendrán diferentes anotaciones, esto es debido a que los valores se pueden repetir entre 2 a 6 veces en el vector de posición. Esto hace que haya una referencia cercana a nuestro valor máximo, entre 30 unidades hacia la derecha como a la izquierda, y con eso buscaremos una referencia cercana a la forma de onda

característica de la transformada de wavelet (Mexican Hat) para encontrar un detector de picos. En la figura 3.4 se puede visualizar los valores máximos, mínimos, promedios y porcentajes.

Tabla 3.2 Porcentaje de aciertos y errores para encontrar la mayor cantidad de datos repetidos.

Porcentaje	Eventos											Aciertos	Errores	% de acierto
	1	2	3	4	5	6	7	8	9	10	11			
90%	x	↗	x	x	x	↗	x	x	↗	x	↗	4	7	36.36
85%	↗	↗	↗	x	↗	↗	↗	↗	x	x	↗	8	3	72.73
80%	↗	↗	x	x	x	↗	x	x	x	↗	x	4	7	36.36
75%	↗	x	↗	x	x	x	↗	↗	x	x	↗	4	7	36.36
70%	x	↗	↗	↗	↗	x	↗	↗	↗	x	↗	8	3	72.73

Nota:	↗	Un valor correcto
	↗:	Dos valores repetidos
	↗:	Tres valores repetidos
	↗::	Cuatro valores repetidos
	↗:::	Seis valores repetidos

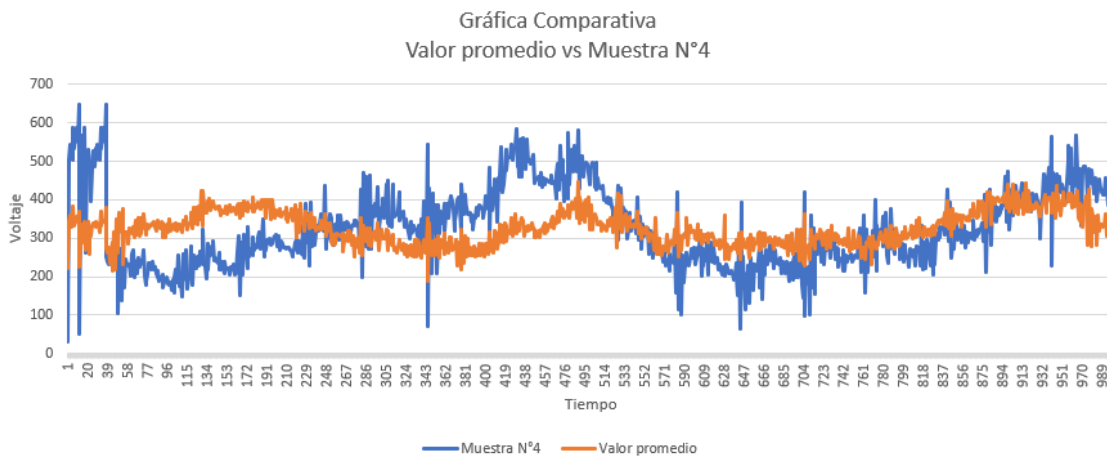
```
C:\Users\Roberto\Documents\Tesis_Phyton_Archivos\Potenci
['293', '350', '321', '311', '359', '314', '317', '337',
393 Valor Máximo
[172] Ubicación del valor máx
['293', '350', '321', '311', '359', '314', '317', '337',
294 Valor al 70%
[8, 92, 168, 197] Ubicación de los valores al 70%
[293, 350, 321, 311, 359, 314, 317, 337, 294, 362, 343,
331 Valor promedio
348.4005 Promedio +-5%
315.2195
282 Valor mínimo
```

Fig. 3.4 Secuencia para obtener los valores máximos, mínimos, promedios, etc.

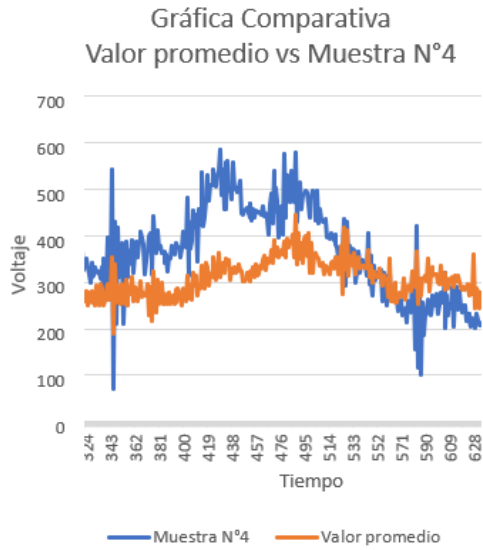
Para nuestra experimentación se prefirió utilizar un porcentaje al 70 % debido a un mayor margen de datos, por lo tanto, nuestros valores no estarían empalmados al valor máximo, pero tampoco se alejan de este.

Los valores promedios que se encontrarán en nuestra señal *EMG* servirán como referencia al momento de graficar, debido a que nos indican en donde se encuentra la mayor cantidad de datos. Tanto el valor promedio como el valor máximo al 70 % nos servirán de referencia. Para demostrar esto, se anexarán gráficas para observar su comportamiento (figura 3.5)

La figura 3.5 (a) demuestra el promedio de 5 gráficas con 5,000 datos promediados comparada con una sola gráfica de 1,000 datos. Como se puede observar, los máximos y mínimos de las señales se reducen en consideración, lo que genera una mayor confiabilidad con la toma de lecturas. En la figura (b) se puede observar un área de interés debido a que ambas gráficas se empalman, generando los mismos pulsos típicos de una señal *EMG*, en consecuencia, se tiene una réplica similar y podemos interpretar que existe información de interés. En la figura (c) y (d) se encuentran de forma individual las gráficas de valor promedio y una gráfica de muestra.

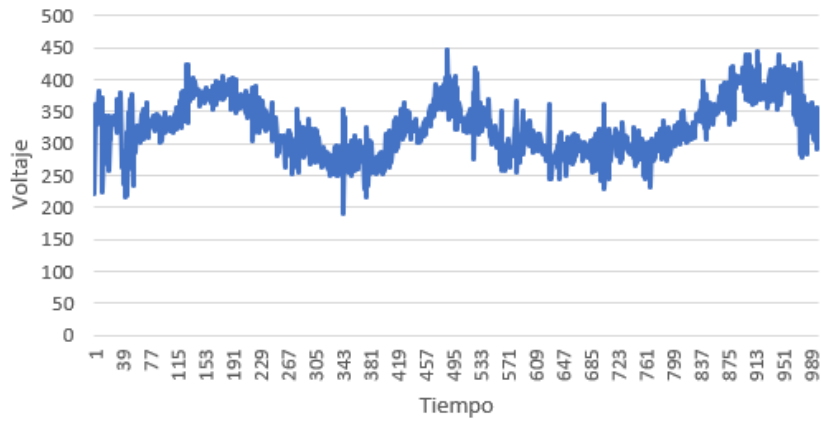


(a)



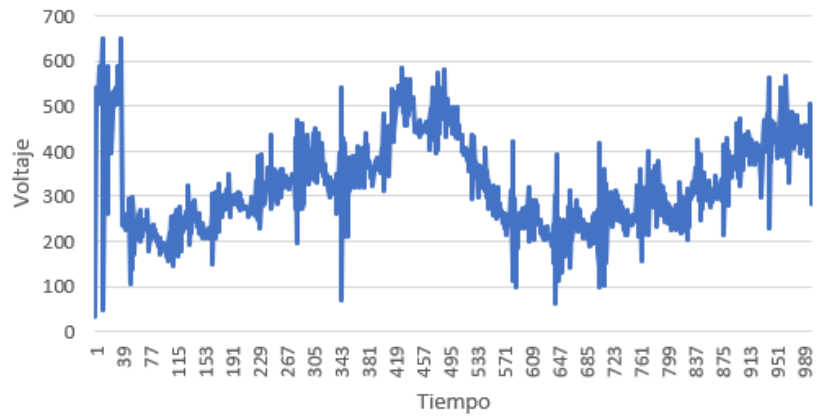
(b)

### Gráfica Promedio



(c)

### Gráfica Muestra 4



(d)

Fig. 3.5 (a) Gráfica comparativa, (b) Área de interés, (c) Gráfica Promedio, (d) Gráfica de Muestra 4.

### 3.2.1 Identificación de Wavelets en las señales de muestra

Para la identificación de nuestra señal de wavelet Mexican Hat se va a seguir como ejemplo la figura 2.11. En la cual se ubicarán 5 puntos de referencia, un valor máximo, un valor mínimo, un valor promedio y el 70 % del valor máximo de la señal. Como se puede observar en la figura 3.4, nuestro valor máximo es de 393 unidades, con una posición en el tiempo de 172 ms, un valor promedio de 331 unidades, un valor mínimo de 282 unidades y el valor al 70 % del valor máximo se encuentra en 294 unidades, con dos posiciones en el tiempo de 168 ms y 197 ms. Por lo tanto, la figura 2.11 sigue una estructura similar a la figura 3.6. Cabe recalcar que los valores han sido cuantizados por el ADC de Arduino y no se ha llevado una conversión analógica (volts).

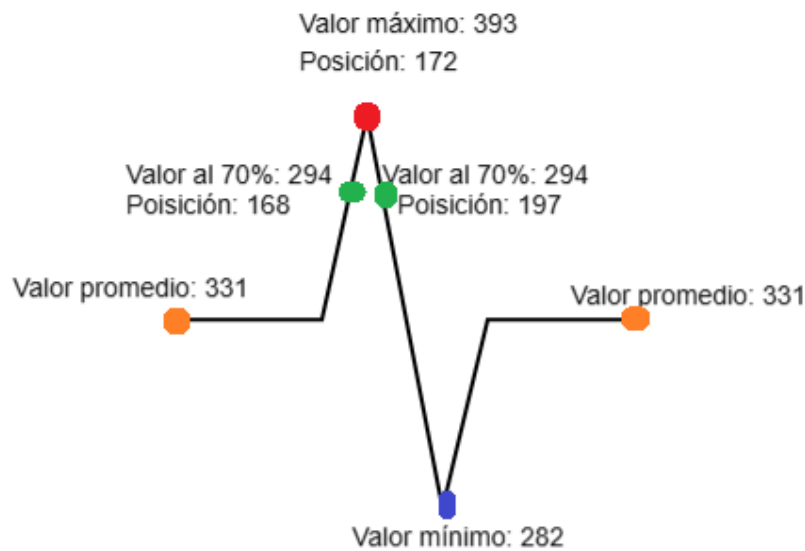


Fig. 3.6 Representación gráfica de la señal *EMG* para formar un detector de picos.

Por lo tanto, para esta experimentación solo ocuparemos la forma de onda de la transformada de wavelet Mexican Hat para encontrar los valores que se asemejen a ella.

Para identificar que la mayoría de datos a tratar se encuentran en un área de interés se trató de integrar diferentes velocidades de movimientos en el cierre y apertura de la mano para determinar el comportamiento de la señal *EMG*. Estos movimientos

tendrán diferentes clasificaciones como: Lento – Rápido (L-R), Lento – Lento (L-L), Lento – Normal (L-N), Rápido – Lento (R-L), Rápido – Rápido (R-R), Rápido – Normal (R-N) y Normal – Normal (N-N), donde L es lento, R es rápido y N es normal. Estas evaluaciones se realizan con un tiempo de 500 ms y se graficaron en histogramas, como lo muestra la figura 3.7, 3.8, 3.9, 3.10, 3.11, 3.12 y 3.13.

Los histogramas nos permitirán encontrar un rango de valores, cada barra tendrá una agrupación de datos, esta información se utilizará para poder encontrar datos significativos e interpretar su comportamiento.

El comportamiento más importante de todas las gráficas es que el valor promedio se ubicara en un rango de 300 a 400 unidades sin importar la velocidad del movimiento. Cabe recalcar que cuanto más lento sea el movimiento, el registro de sus datos se estará graficando hacia el lado derecho. En comparativa con un movimiento rápido, estos movimientos tratarán de ubicarse el lado izquierdo del histograma.

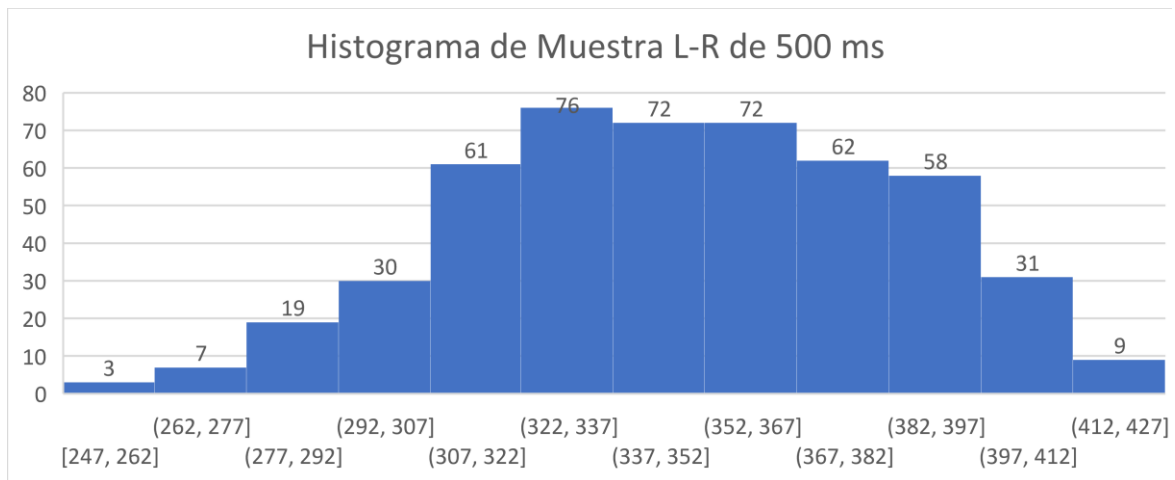


Fig. 3.7 Movimiento Lento – Rápido.

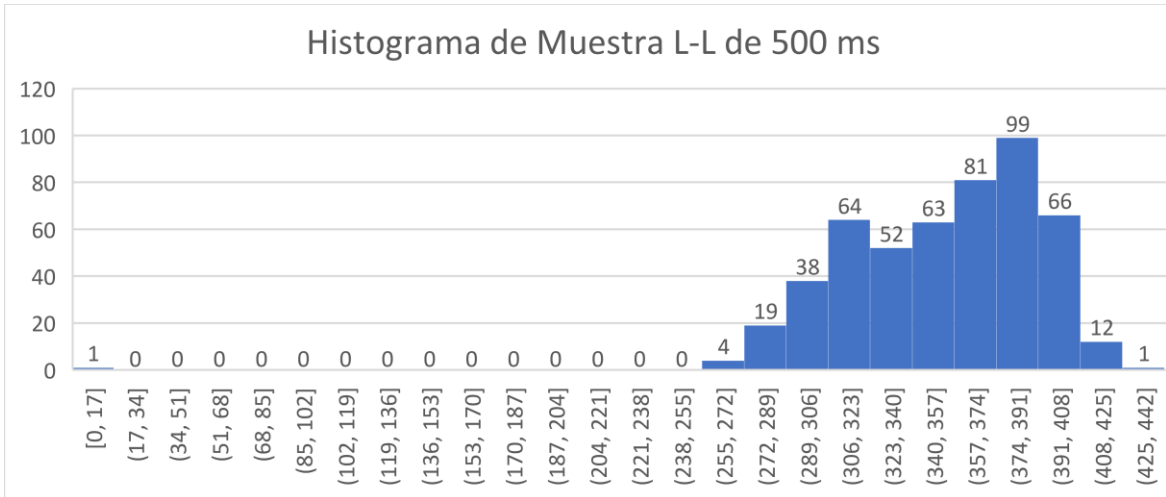


Fig. 3.8 Movimento Lento – Lento.

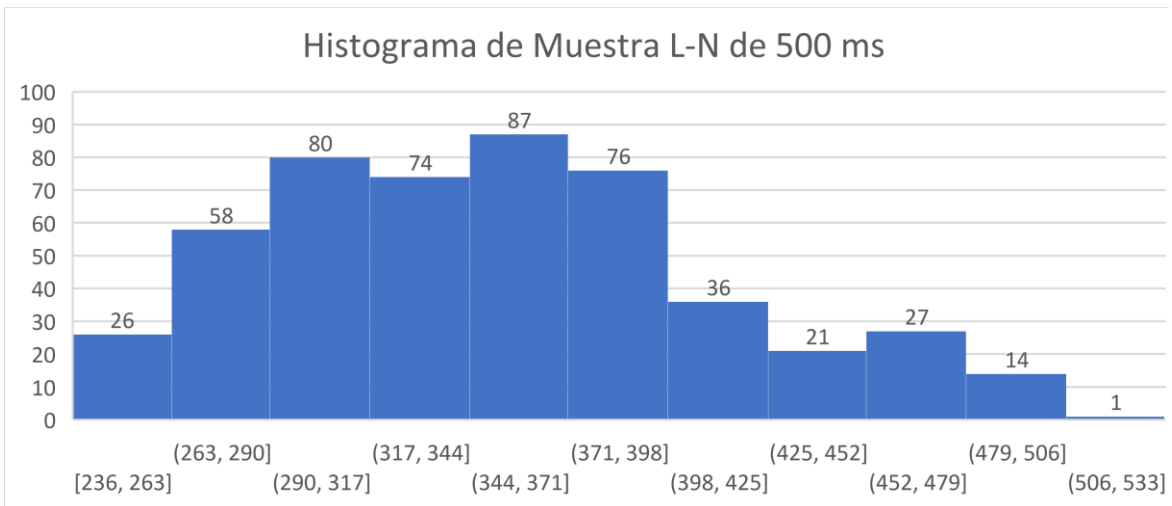


Fig. 3.9 Movimento Lento – Normal.

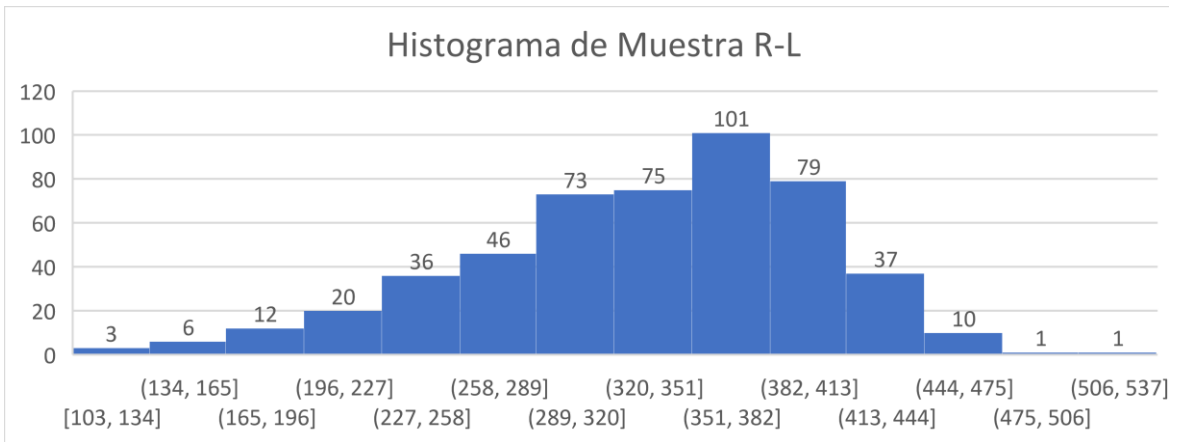
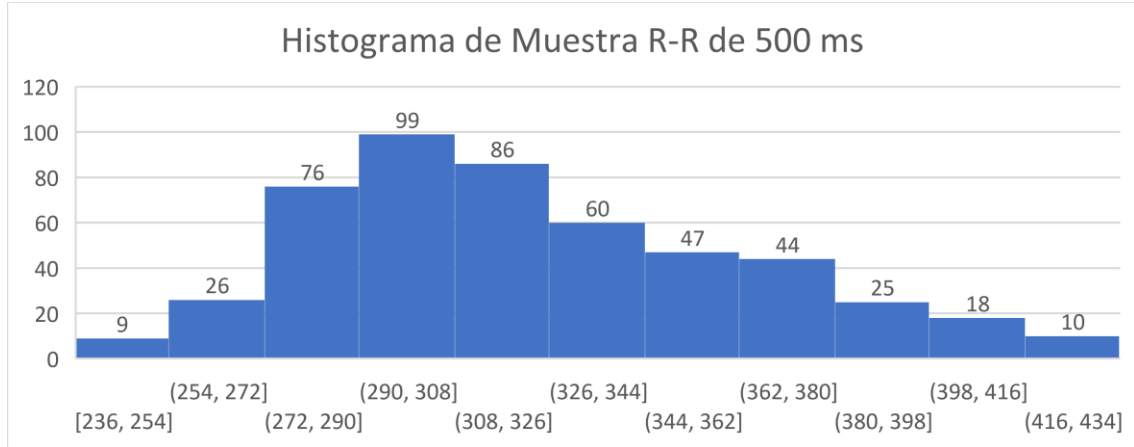
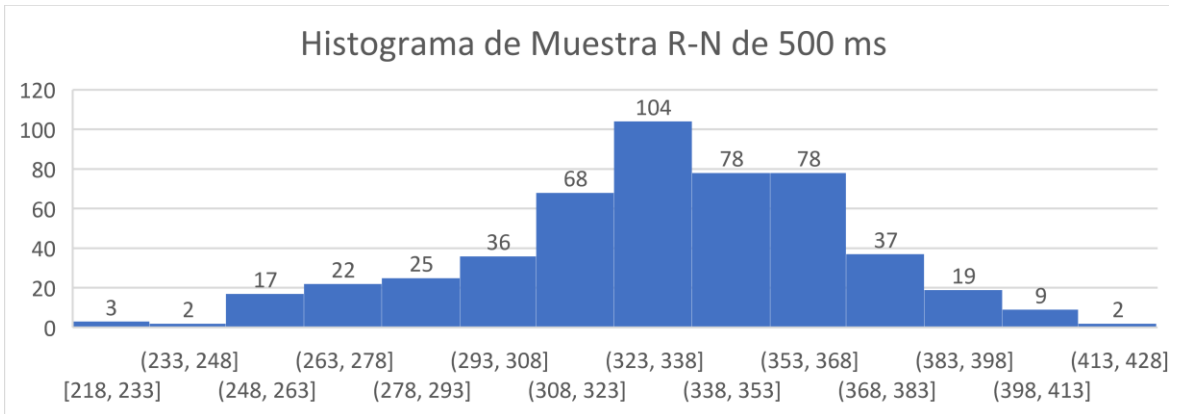


Fig. 3.10 Movimento Rápido – Lento.

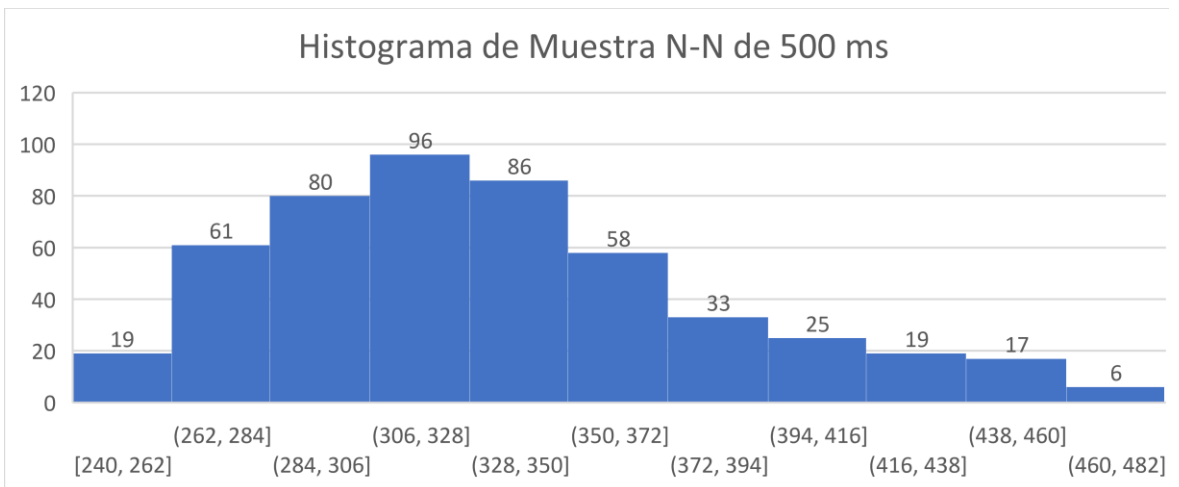




**Fig. 3.11** Movimento Rápido – Rápido.



**Fig. 3.12** Movimento Rápido – Normal.



**Fig. 3.13** Movimento Normal – Normal.

### 3.2.2 Interpretación

El comportamiento típico de una señal *EMG* se representa en la figura 3.14 (a). La información proveniente de Python se tuvo que guardar en archivos txt, con esta información se puede manipular datos y gráficas, como los histogramas que se muestran en la figura 3.14 (b). Los datos de mayor relevancia son los valores máximos, mínimo, promedio, valores al 70 % y sus posiciones, como lo muestra la tabla 3.3. En la figura 3.14 (b) se encuentra un histograma donde se representa la distribución de datos recolectados en 200 ms.

Valor máximo	469	Posición	100
Valor al 70 %	328		47, 120
Valor mínimo	263		54
Valor promedio	365		26, 136, 157

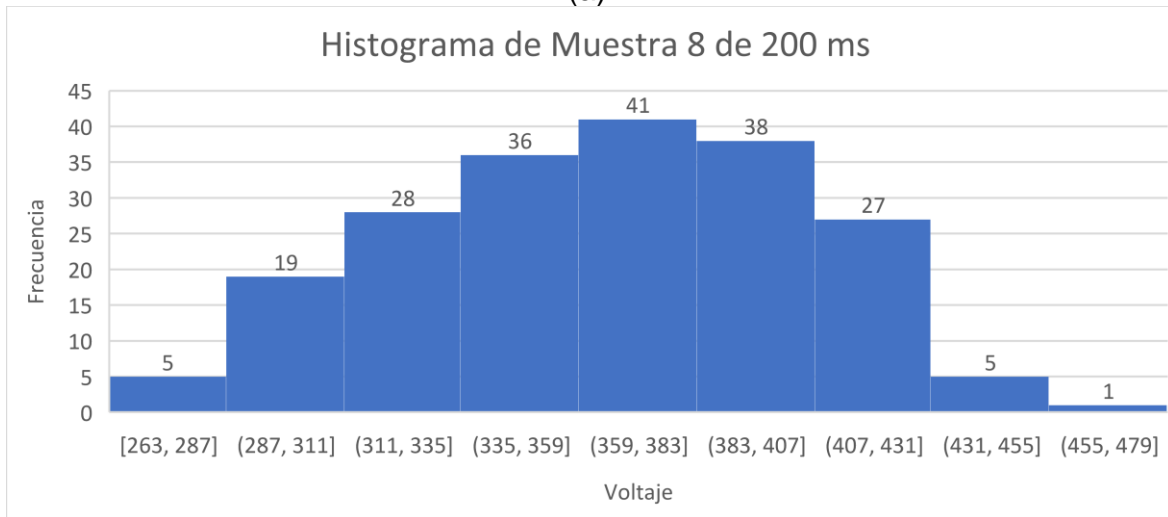
Al analizar la figura 3.14 (b), se observa que la mayor concentración de datos se ubica entre los valores 359 a 383, nuestro valor promedio se ubica en 365, comprobando que existe una relación entre el valor promedio y el histograma.

La figura 3.14 (c) muestra la ubicación de cada dato de interés, con lo cual podemos observar que se asemeja a la figura 3.14 (a) y a la figura 2.11, con lo cual se cumple con el objetivo de replicar la forma de onda de la técnica Mexican Hat (transformada de Wavelet) y comprobar que existe un detector de picos.

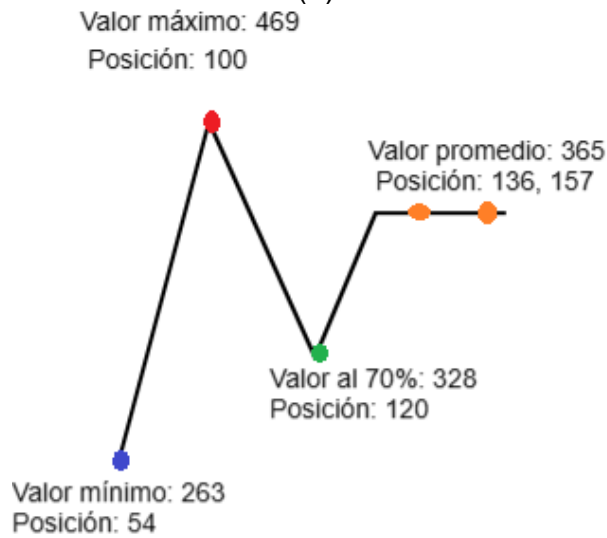
El valor promedio, máximo y mínimo serán de gran utilidad para poder crear las condiciones del controlador ON – OFF, debido a que podemos hacer una interpretación y establecer que nuestro valor promedio oscilaría en 350, y delimitaremos su límite superior e inferior a  $\pm 75$  unidades.



(a)



(b)



(c)

Fig. 3.14 (a) Comportamiento teórico de una señal *EMG* (b) Histograma de señal *EMG*, (c) Comportamiento experimental de un detector de picos, utilizando la técnica de Wavelet Mexican Hat.

### 3.3 Controlador ON – OFF con histéresis

El controlador ON – OFF con histéresis será de utilidad para demostrar el funcionamiento de nuestra prótesis. El objetivo de desarrollar este método es para evitar toda oscilación innecesaria y que esto conlleve al accionamiento de los 5 servomotores en cualquier momento. Con lo cual, se busca que los servomotores accionen al momento de que la señal llegue al umbral superior o inferior. Por lo tanto, las condiciones de funcionamiento para el movimiento de la prótesis se establecen en la tabla 3.4.

Al observar la figura 3.15, se puede comprobar el funcionamiento del controlador ON – OFF. En color anaranjado se visualiza la señal *EMG*. El color verde representa el valor promedio de la señal. En color azul se observa el umbral superior; en este punto, si la señal *EMG* supera el umbral, la prótesis tendrá un movimiento de contracción. En color rojo se puede encontrar el umbral inferior, si este umbral se ve superado, entonces la prótesis tendrá un movimiento de apertura. Las imágenes que demuestran el funcionamiento de la prótesis se visualizan en las figuras 3.16, 3.17 y 3.18.

En la figura 3.16 se puede observar la zona de histéresis, en esta zona se genera gran cantidad de oscilaciones. Si no existiera la configuración del controlador ON – OFF con histéresis, en cada momento los servomotores se estuvieran accionando, este pequeño problema fue el causante por el cual decidí implementar este método. Por lo tanto, este método es efectivo para demostrar un funcionamiento correcto.

Tabla 3.4 Valores y tipos de movimiento para el controlador ON – OFF con histéresis		
Tipo de umbral	Voltaje (ADC)	Tipo de movimiento
Umbral superior	425	Cerrar la mano
Valor promedio	350	Zona de histéresis
Umbral inferior	275	Abrir la mano

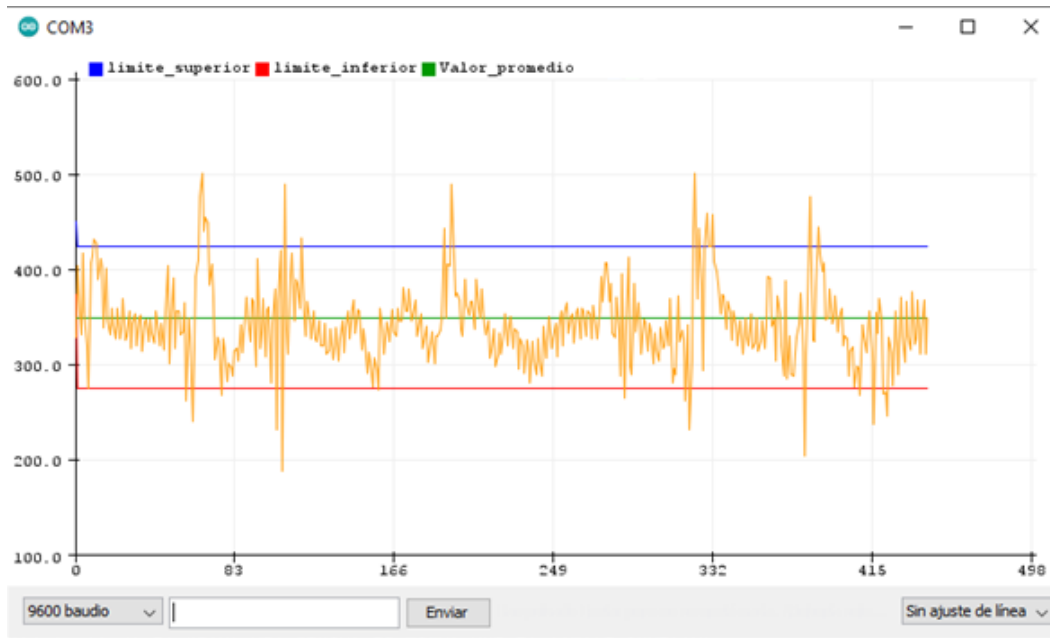


Fig. 3.15 Gráfica del controlador ON – OFF con histéresis.



Fig. 3.16 Vista frontal del movimiento de cierre de mano.

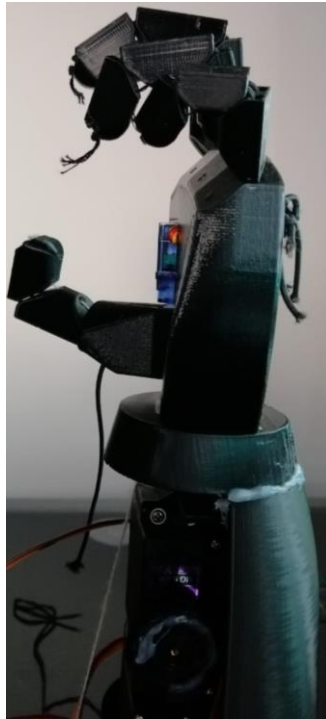


Fig. 3.17 Vista lateral del movimiento de cierre de mano



Fig. 3.18 Vista lateral del movimiento de apertura.

### 3.4 Control PC – Prótesis

El motivo principal por el cual se implementó este método experimental es para complementar una mayor cantidad de movimientos que se puede generar en los dedos. La secuencia principal gestionará de forma automática los movimientos de apertura y cierre de los dedos de forma grupal e individual hasta que ocurra alguna interrupción. Esta interrupción se generará por medio de una herramienta de Arduino conocida como “Monitor Serie”. Para generar la interrupción, se debe de ingresar por medio del teclado la letra “a”, “b”, “c”, “d” o “e”, cada letra tendrá su propio movimiento. La tabla 3.5 muestra estas configuraciones.

En el rango de figuras que van desde la figura 3.19 hasta la figura 3.30 se visualizarán los movimientos que genera la prótesis.

Tabla 3.5 Configuración de movimiento para el control PC – Prótesis	
Movimiento sin interrupción	
Cierre y apertura de los dedos en forma grupal e individual de forma automática	
Movimiento con interrupción	
Letra	Tipo de movimiento
a	Símbolo de paz.
b	Símbolo del cuerno de rock.
c	Únicamente el dedo pulgar, índice y medio se encuentran en posición abierta.
d	Únicamente el dedo meñique e índice se encuentran en posición abierta.
e	Únicamente el dedo pulgar, anular y meñique se encuentran en posición abierta.



Fig. 3.19 Apertura de dedos sin generar una interrupción por medio de “Monitor Serie”.



Fig. 3.20 Cierre de dedos sin generar una interrupción por medio de “Monitor Serie”.





Fig. 3.21 Movimiento individual del dedo pulgar sin generar interrupciones.



Fig. 3.22 Movimiento individual del dedo índice sin generar interrupciones.



Fig. 3.23 Movimiento individual del dedo medio sin generar interrupciones.



Fig. 3.24 Movimiento individual del dedo anular sin generar interrupciones.



Fig. 3.25 Movimiento individual del dedo pulgar sin generar interrupciones.



Fig. 3.26 Interrupción letra "a": símbolo de paz.



Fig. 3.27 Interrupción letra “b”: símbolo de rock.



Fig. 3.28 Interrupción letra “c”: dedo pulgar, anular y medio en posición abierta.



Fig. 3.29 Interrupción letra “d”: dedo pulgar e índice en posición abierta.



Fig. 3.30 Interrupción letra “e”: dedo pulgar, anular y meñique en posición abierta.

## ***Conclusiones y trabajo a futuro***

En el presente trabajo se demostró que a través de la forma de onda de la transformada de Wavelet se puede recrear un detector de picos. Todo este proceso de detección se realizó por medio de Arduino, Python y Excel. Se puede interpretar que las señales eléctricas que provienen del cuerpo son variables, y que en su momento estas señales pueden tener la información necesaria para su manipulación de datos, en algún punto, estas señales también pueden tener algún tipo de ruido, por lo tanto, al usar el detector de picos, nos podemos dar cuenta que la mayoría de sus datos estar de forma consecutiva, como se observa en la figura 3.6 y 3.14 del capítulo 3.

Una de las herramientas importante en este trabajo con los histogramas, debido a que en ellos podemos observar que la mayor cantidad de información promedio se encuentra entre las 300 a 400 unidades. Dentro de la experimentación y en los ejemplos que se visualizó en el capítulo 3, nuestros datos experimentales estarán oscilando entre ese rango de datos, por lo tanto, tanto las gráficas como los histogramas se complementan para comprobar nuestra experimentación.

El beneficio de haber encontrado los valores promedio, máximo, mínimo y el valor al 70 % de la señal máxima es que nos permitió crear las variables para el funcionamiento del controlador ON – OFF con histéresis, por lo tanto, cada uno de estos valores nos permitirá crear los umbrales superiores e inferiores de operación. En la figura 3.15, se puede observar que hubo 4 operaciones de apertura y cierre de la prótesis en un tiempo de 400 ms, por lo tanto, la información que se va registrando por medio del “Monitor Serie” de Arduino y por los movimientos físicos del prototipo, nos demuestra un funcionamiento correcto.

Para implementar una mayor cantidad de movimientos se decidió experimentar con la metodología del control PC – Prótesis, con la cual demostramos que el movimiento de nuestra prótesis funciona y que es capaz de seguir la instrucción que se le ordenaron al momento de ser programada. También se comprobó que, al

momento de ser interrumpida de la secuencia principal, la prótesis tendrá 5 movimientos en particular y se demuestra que su proceso es funcional.

Una herramienta de gran utilidad fue el circuito AD8232, debido a que ya tiene implementada toda una electrónica de instrumentación en un solo chip. Por lo tanto, se evita la construcción de un sistema de adquisición de señales. Dentro de las ventajas de este circuito, es que su uso está recomendado para utilizarse con Arduino.

Como trabajo a futuro me gustaría diseñar una electrónica de instrumentación para el acondicionamiento de señales. Al principio de este proyecto se desarrolló una electrónica de instrumentación, sin embargo, debido a las dificultades que en su momento tuve, no fue viable ni funcional. Por lo tanto, tengo el interés de aprender instrumentación electrónica.

El diseño de una nueva prótesis en 3D sería un tema fundamental, esto es debido a las desventajas de armado, estética y funciones que presenta el modelo actual. Dentro del armado se presenta que sus espacios internos son muy estrechos, con lo cual, hacer cualquier modificación en el antebrazo generará desarmar la placa donde se sujetan los servomotores, por lo tanto, modificar la longitud y tensión del hilo para la apertura y cierre de los dedos generará inconvenientes, retrasos y modificaciones al momento de realizar pruebas para su correcto funcionamiento. Y a la vez una desventaja estética es la cantidad de cables que salen por el antebrazo, por lo tanto, sería bueno tener un compartimiento donde se guarden los cables de los servomotores. Con respecto a las funciones de la prótesis, me gustaría agregar movimiento en la muñeca y antebrazo del prototipo.

A lo largo del proyecto se ha logrado demostrar el funcionamiento del prototipo de prótesis, con lo cual se cumplen los objetivos de este proyecto. También se establecen las bases para un trabajo a futuro, por lo tanto, este trabajo funcionará como base de nuevos proyectos, como:

1. Diseño, desarrollo e implementación de la electrónica de instrumentación para aplicaciones biomédicas, enfocándose en la adquisición de las señales.
2. Implementar un modelo de control que permita replicar el movimiento de un miembro no amputado, con lo cual permitirá realizar actividades con ambas manos, como si fuera un seguidor de movimientos, el cual puede desactivarse por medio de un botón y realizar actividades de forma independiente.
3. Implementar un modelo en el cual, las señales electromiográficas puedan ser clasificadas para ciertas acciones, como la apertura y cierre de los dedos, así como la rotación de las muñecas. Y que el antebrazo pueda realizar una contracción del bíceps.

Al tratar de mejorar estos prototipos, es posible que se pueda replicar el funcionamiento que realiza algún modelo de prótesis que se observe en nuestra metodología. Con lo cual, se podría llegar a comercializar o en su caso llegar a generar un aporte de conocimiento en el área biomédica.



## Bibliografía

1. Social, I. N. D. D. (s. f.). *En Jalisco devuelven autonomía a personas que han sufrido amputaciones*. Gobierno de México. Recuperado 21 de enero de 2023, de <https://www.gob.mx/indesol/prensa/en-jalisco-devuelven-autonomia-a-personas-que-han-sufrido-amputaciones>
2. *Prótesis de bajo costo para miembros superiores, decoradas como brazos de súper héroes, desarrolladas en México*. (s. f.). Código F. Recuperado 21 de enero de 2023, de <https://codigof.mx/protesis-de-bajo-costopara-miembros-superiores-decoradas-como-brazos-de-super-heroes-desarrolladas-en-mexico/>
3. *Lista de precios de prótesis*. (s. f.). DIANCEHT. Recuperado 21 de enero de 2023, de <https://www.manosydedos.com/precios.html>
4. *Población. Discapacidad*. (s. f.). INEGI. Recuperado 21 de enero de 2023, de <https://cuentame.inegi.org.mx/poblacion/discapacidad.aspx>
5. Real Academia Española. (2021). *Prótesis*. Recuperado 1 de julio de 2022, de <https://dle.rae.es/protesis>
6. Brito, J. L., Quinde, M. X., Cuzco, D., & Calle, J. I. (2013). Estudio del estado del arte de las prótesis de mano. *INGENIUS Revista de Ciencia y Tecnología*, 9(1). Recuperado 29 de julio de 2022, de <https://dspace.ups.edu.ec/bitstream/123456789/8447/1/Estudio%20del%20estado%20del%20arte%20de%20las%20pr%C3%B3tesis%20de%20mano.pdf>
7. Dorador González, J. M. (2005, 18 enero). Robótica y prótesis inteligentes. *Revista Digital Universitaria*, 6(1). Recuperado 29 de julio de 2022, de [https://www.ru.tic.unam.mx/bitstream/handle/123456789/840/art01\\_enero.pdf?sequence=1&isAllowed=y](https://www.ru.tic.unam.mx/bitstream/handle/123456789/840/art01_enero.pdf?sequence=1&isAllowed=y)
8. Brito, J., Quinde, M., Cusco, D. & Calle, J. (2013, 30 junio). Estudio del estado del arte de las prótesis de mano. *Ingenius*, 9. Recuperado el 26 de julio de 2022 de <https://doi.org/10.17163/ings.n9.2013.08>

9. *Mano biónica i-Limb® Quantum* | Össur. (s. f.). Recuperado 29 de julio de 2022, de <https://www.ossur.com/es-mx/protesica/miembro-superior/i-limb-quantum#overviewContentAnchor>
10. Mano bebionic EQD | La mano protésica más natural. (s. f.). Recuperado 29 de julio de 2022, de <https://www.ottobock.com/es-mx/product/8E70>
11. Otto Bock Healthcare Products GmbH. (2021, 28 enero). *Instructions for use (user) 8E70, 8E71, 8E72 bebionic hand* (US 9 101 499; US 9 592 134). FDA. <https://cep-spa-prod-commerce-static.azurefd.net/medias/4386521.pdf?context=bWFzdGVyfHJvb3R8MzMzMjMyMjEzOHxhcHBsaWNhdGlvbi9wZGZ8aGM1L2g0Ny85MDcwMTk4MTYxNDM4LnBkZnxiZGVmZDBiNW15Y2YwMTRmNDZkYTlmYzk2NTI1ODU3YTY5ZmE0MjdmYTVIMjkzYWE5ODk4MWZkZDczYWJjYjgz&attachment=true>
12. Mobius Bionics. (s. f.-b). The LUKE Arm. Recuperado 30 de julio de 2022, de <https://www.mobiusbionics.com/>
13. Sôrno, L., & Laguna, P. (2007). *El procesado de señales en medicina* (1.ª ed., Vol. 1) [Online]. Universidad de Castilla - La Mancha. [Procesado de señales biomédicas - Pedro A. Carrión Pérez, Juan Ródenas García, José Joaquín Rieta Ibáñez - Google Libros](#)
14. Pérez, P. A. C., García, J. R., & Ibáñez, J. J. R. (2007). *Procesado de señales biomédicas* (Vol. 53). Univ de Castilla La Mancha. Recuperado 26 de julio de 2022, de [Procesado de señales biomédicas - Pedro A. Carrión Pérez, Juan Ródenas García, José Joaquín Rieta Ibáñez - Google Libros](#)
15. López, J. M., Martí, G., Puente Méndez, S. T., Candelas-Herías, F. A., Úbeda, A., & Torres, F. (2018). Implementación y evaluación de un esquema de control mioeléctrico ON/OFF utilizando hardware de bajo coste. Recuperado el 26 de julio de 2022, de [Implementación y evaluación de un esquema de control mioeléctrico ON/OFF utilizando hardware de bajo coste \(udc.es\)](#)
16. Coordinación de Universidad Abierta, Innovación Educativa y Educación a Distancia de la UNAM. (s. f.). *Sistema nervioso central y sistema nervioso periférico*. B@UNAM. Recuperado 26 de julio de 2022, de [https://uapas1.bunam.unam.mx/ciencias/snc\\_y\\_snp/](https://uapas1.bunam.unam.mx/ciencias/snc_y_snp/)

17. JHONS HOPKINS MEDICINE. (s. f.). *Electromyography (EMG)*. Recuperado 26 de julio de 2022, de <https://www.hopkinsmedicine.org/health/treatment-tests-and-therapies/electromyography-emg>
18. Amon, Clemens. (2013). Design and Evolution of an EMG-based Recording and Detection System. Institute of Electronic Music and Acoustics. Austria. Recuperado el 27 de julio de 2022, de [TI-Project Thesis Clemens Amon 2013 Design and Evaluation of an EMG-based Recording and Detection System.pdf \(kug.ac.at\)](https://www.kug.ac.at/theses/Project_Thesis_Clemens_Amon_2013_Design_and_Evaluation_of_an_EMG-based_Recording_and_Detection_System.pdf)
19. López, D. A. R., López, M. A., Sánchez, J. E. D., & Correa, H. L. (2015). Implementación en FPGA de un clasificador de movimientos de la mano usando señales EMG. *Redes de ingeniería*, 6(1), 85-94.
20. Romo, H. A., Realpe, J. C., & Jojoa, P. E. (2007). Análisis de señales EMG superficiales y su aplicación en control de prótesis de mano. *Revista avances en sistemas e informática*, 4(1), 127-136. Recuperado de <https://www.redalyc.org/pdf/1331/133116856017.pdf>
21. HBK company. (s. f.). *¿Qué es un acondicionador de señal?* Recuperado 27 de julio de 2022, de <https://www.hbm.com/es/7339/que-es-un-acondicionador-de-senal-funciones/#:~:text=El%20acondicionamiento%20de%20se%C3%B1al%20es%20un%20proceso%20de,%28se%C3%B1al%20de%20entrada%29%20en%20otro%20%28se%C3%B1al%20de%20salida%29>.
22. Alvarez Osorio, L. (2007). Acondicionamiento de señales bioeléctricas. Universidad Tecnológica de Pereira. Colombia. Recuperado el 29 de julio de 2020, de <https://repositorio.utp.edu.co/server/api/core/bitstreams/b103d5c7-9c46-4c79-9bef-70ec3e0bbe6d/content>
23. Alzate Arias, N. (2018). Control mioeléctrico de una prótesis de miembro superior-Mano. Universidad Militar Nueva Granada. Colombia. Recuperado el 29 de julio de 2022 de, <https://repository.unimilitar.edu.co/handle/10654/17240>

24. Analog Devices. (s. f.). AD8232. All data sheet. Recuperado 29 de julio de 2022, de <https://www.alldatasheet.com/datasheet-pdf/pdf/527942/AD/AD8232.html>
25. Torrente Artero, O. (2013). Arduino curso práctico de formación. RC Libros. [https://books.google.com.mx/books?hl=es&lr=&id=6cZhDmf7suQC&oi=fnd&pg=PR15&dq=que+es+arduino&ots=A-eAmXGxyH&sig=IFoZ3Lj\\_fx8nscZHFfei5eGUDDE#v=onepage&q=que%20es%20arduino&f=false](https://books.google.com.mx/books?hl=es&lr=&id=6cZhDmf7suQC&oi=fnd&pg=PR15&dq=que+es+arduino&ots=A-eAmXGxyH&sig=IFoZ3Lj_fx8nscZHFfei5eGUDDE#v=onepage&q=que%20es%20arduino&f=false)
26. *Características técnicas del ARDUINO UNO*. (2018, 18 octubre). Recuperado 15 de febrero de 2023, de <https://www.bolanosdj.com.ar/MOVIL/ARDUINO2/CaracteristicasArduino.pdf>
27. Santander Universidades. (2022, 16 septiembre). *¿Qué es Python? | Blog. Becas Santander*. Recuperado 30 de julio de 2022, de <https://www.becas-santander.com/es/blog/python-que-es.html>
28. Cano, E., Salcedo, R., & Soto, G. (2010). Análisis de principios y aplicaciones de la transformada Wavelet. *Universidad Nacional de Catamarca*. [cano-jct-corregido-libre.pdf \(d1wqtxts1xzle7.cloudfront.net\)](#)
29. Ruíz Salazar, J., & Orejuela Caicedo, D. A. (2016). *Implementación de la transformada wavelet sobre un sistema embebido para el preprocesamiento de señales unidireccionales no estacionarias* [Tesis de licenciatura]. UNIVERSIDAD DE SAN BUENAVENTURA CALI. [content \(usb.edu.co\)](#)
30. Dutan Rodríguez, P. M. (2000). *Diseño y reconstrucción del control de temperatura de un horno con calentamiento eléctrico* [Tesis de licenciatura]. Universidad Politécnica Nacional. [T1682.pdf \(epn.edu.ec\)](#)
31. SENIAM. (s. f.). *SENIAM*. Recuperado 10 de agosto de 2022, de <http://seniam.org/>
32. Toinga Villafuerte, Samantha Elizabeth. (2017). Diseño e implementación de un módulo didáctico para la enseñanza interactiva del concepto de

Einthoven. Tesis de licenciatura. Escuela Politécnica Nacional. Ecuador. Recuperado el 10 de agosto de 2022 de, <https://bibdigital.epn.edu.ec/bitstream/15000/18874/1/CD-8266.pdf>

33. SENIAM. (s. f.). *Inter Electrode Distance*. SENIAM. Recuperado 10 de agosto de 18d. C., de <http://seniam.org/>
34. *AD8232 Heart Rate Monitor Hookup Guide - SparkFun Learn*. (s. f.). SparkFun Start Something. Recuperado 3 de febrero de 2023, de <https://learn.sparkfun.com/tutorials/ad8232-heart-rate-monitor-hookup-guide>
35. *¿Cómo conectar y programar el módulo AD8232 ECG con Arduino Uno?*. (s. f.). Blog By Unit Electronics. Recuperado 10 de agosto de 2022, de <https://blog.uelectronics.com/tarjetas-desarrollo/arduino/como-realizar-la-programacion-conexion-del-modulo-ad8232-ecg-con-arduino-uno/>

## ***Anexo A. Códigos del sistema***

Fig. A.1 Configuración entre placa Arduino y el circuito AD8232

```
const int pinLED=13;
void setup(){
  // Inicializar la comunicación en serie:
  Serial.begin(9600);
  pinMode(10, INPUT); // Configuración para la detección LO +
  pinMode(11, INPUT); // Configuración para la detección LO -
  pinMode(pinLED, OUTPUT);
}
void loop() {
  if((digitalRead(10) == 1)|| (digitalRead(11) == 1)){ //Si no existen datos que el sensor reconozca
    Serial.println('!'); //Imprimira ! por falta de datos
  }
  else{ //si no es así
    // Imprimir la lectura del puerto A0
    Serial.println(analogRead(A0));
  }
  //Espere un poco para evitar que los datos en serie se saturen
  delay(10);
  // fin lectura
}
```

Fig. A.2 Interfaz Arduino – Python.

```
import serial
import time
import numpy as np
import numpy

archivo = open('C:\\Users\\Roberto\\Documents\\Tesis_Phyton_Archivos\\Archivos_txt_python\\txt\\R-N.txt', 'w')
serialArduino = serial.Serial("COM3", 9600) #Comunicaremos lo datos de arduino
serialArduino.flushInput() ##Limpia los datos del buffer
serialArduino.setDTR()
time.sleep(1) ##Delay de 1 seg
lista=[] ## Creación de un Vector
lista1=[] ##Creación de un vector

for i in range(0,500,1): #contador de 0 hasta 200 para la generación de muestras

    cad = serialArduino.readline().decode('ascii') ##Transforma los datos en valores string
    archivo.write(cad) ##Escribe los datos
    cad2=cad.rstrip() ##Elimina caracteres de la cadena
    lista.append(cad2) ##Agrega los elementos a la lista
    lista1.append(cad2) ##
print(lista) ##Imprime los valores del vector
maxvalor=max(lista) ##Buscar el valor max del vector
maxvalor2=int(maxvalor) ##Convertir en valor entero el vector
print(maxvalor) ##Imprime el maxvalor
Lista = np.array(lista)
resultado = np.where(Lista == maxvalor) ##codigo de indice
print(resultado[0])
```

```

##### Codigo nuevo para encontrar el 70%
lista1 = lista.copy()          #copiar la lista
print(lista1)

#####Transofrmación del 70% en datos flotante

a=float(70/100)                #Se convierte el 75% en decimal y uso float
b=float(maxvalor2)            #Se convierte el valor maximo en decimal
valor=(a)*(b)                  #Se multiplica a y b en flotante
valor1=int(valor)              #Se convierte en entero
print(valor1)                  #se imprime el valor entero
valor2=str(valor1)             #el numero int lo convertimos a str

#####Indice para encontrar el numero del 75%

pos = []
x=valor2
for i in range(len(lista1)):   #buscamos el indice del valor 2 en
    if lista1[i] == x:         #toda nuestro bloque de numeros y nos
        pos.append(i)         #retorna el lugar donde se encuentre
print(pos)

##### Promedio
ints=[]                         #Vector donde guardaremos los datos de la lista en tipo entero
for element in lista:
    ints.append(int(element))   #ciclo for para su conversión en valores enteros
print(ints)

mean = numpy.mean(ints)         #Comando para buscar el valor proemdio de la señal
#print(mean)                    #valores tipo flotante
conv=int(mean)                  #conversión a tipo entero
print(conv)

#####Promedio +-5% de la señal
c=float(5/100)                  #porcentaje de 5 %
d=float(mean)                   #conversion de tipo int a float
e=(d)+(d)*(c)                   #valor +5% al valor promedio
print(e)
d=(d)-(d)*(c)                   #Valor -5% al valor promerido
print(d)

###Encontrar el valor minimo de la señal
minvalor=min(ints)              #Busca el valor minimo
print(minvalor)

```



*Fig. A.3 Controlador ON – OFF con histéresis.*

```
#include <Servo.h>           //Codigo ON OFF

Servo pulgar; // Servomotor 1
Servo indice; // Servomotor 2
Servo medio; // Servomotor 3
Servo anular; //Servomotor 4
Servo menique; //Servomotor 5

int setpoint = 350; // Punto de consigna
int hysteresis = 75; // Histéresis
int upper_limit = setpoint + hysteresis; // Límite superior
int lower_limit = setpoint - hysteresis; // Límite inferior

void setup() {
  pulgar.attach(2); //Pin2 dedo pulgar
  indice.attach(3); //Pin3 dedo indice
  medio.attach(4); //Pin4 dedo medio
  anular.attach(5); // Pin4 dedo anular
  menique.attach(6); // Pin5 dedo meñique
}

void loop() {
  int value = analogRead(A0); // Lee el valor del sensor conectado al pin A0
  Serial.println(A0);

  if (value > upper_limit) { // Si el valor del sensor es mayor que el límite superior
    pulgar.write(120); // Gira el servo 1 a 90 grados
    indice.write(170); // Gira el servo 2 a 180 grados //Cierra la mano
    medio.write(150); // Gira el servo 3 a 150 grados
    anular.write(150); // Gira el servo 4 a 150 grados
    menique.write(150); // Gira el servo 5 a 150 grados
  }
  else if (value < lower_limit) { // Si el valor del sensor es menor que el límite inferior

    pulgar.write(180); // Gira el servo 1 a 180 grados
    indice.write(0); // Gira el servo 2 a 0 grados //abre la mano
    medio.write(0); // Gira el servo 3 a 0 grados
    anular.write(0); // Gira el servo 4 a 0 grados
    menique.write(0); // Gira el servo 5 a 0 grados
  }

  delay(100); // Espera 100 ms antes de repetir el ciclo
}
```

Fig. A.4 Control PC - Prótesis

```
#include <Servo.h> //codigo de control PC - Prótesis

char letra;

Servo Pulgar; //Crear el objeto del dedo Pulgar
int pos=180; //Poner a 0 la posicion inicial del Pulgar
Servo Indice; //Crear el objeto del dedo Indice 120
int pos2=0; //Poner a 0 la posicion inicial del Indice
Servo Medio; //Crear el objeto del dedo Medio
int pos3=0; //Poner a 0 la posicion inicial del Medio
Servo Anular; //Crear el objeto del dedo Anular
int pos4=0; //Poner a 0 la posicion inicial del Anular
Servo Menique; //Crear el objeto del dedo Meñique
int pos5=0; //Poner a 0 la posicion inicial del Meñique

void setup() {

Serial.begin(9600); //Comunicacion a 9600 con la placa Arduino
Pulgar.attach(2); //Pin2 digital variable Pulgar
Indice.attach(3); //Pin3 digital variable Indice
Medio.attach(4); //Pin4 digital variable Medio
Anular.attach(5); //Pin5 digital variable Anular
Menique.attach(6); //Pin6 digital variable Meñique

Pulgar.write(180); //Pulgar abierto
Indice.write(0); //Indice abierto
Medio.write(0); //Medio abierto
Medio.write(0); //Medio abierto
Anular.write(0); //Anular abierto
Menique.write(0); //Meñique abierto
delay(2000);
}
```

```

void loop() {

if (Serial.available ()>0)
{
  letra=Serial.read();
}
  switch (letra)
  {
    case 'a':          //movimiento de forma de dedos de PAZ

Pulgar.write(120); //Movimiento dedo pulgar cerrado
Indice.write(0); //Movimiento dedo indice abierto
Medio.write(0); //Movimiento dedo medio abierto
Anular.write(150); //Movimiento dedo anular cerrado
Menique.write(170); //Movimiento dedo meñique cerrado
delay(3000);

break;

    case 'b':          //Movimiento en forma de rock

Pulgar.write(180); //Movimiento dedo pulgar abierto
Indice.write(0); //Movimiento dedo indice abierto
Medio.write(150); //Movimiento dedo medio abierto
Anular.write(150); //Movimiento dedo anular cerrado
Menique.write(0); //Movimiento dedo meñique abierto
delay(3000);
break;

    case 'c':          //Movimiento 3 dedos arriba

Pulgar.write(180); //Movimiento dedo pulgar abierto
Indice.write(0); //Movimiento dedo indice abierto
Medio.write(0); //Movimiento dedo medio abierto
Anular.write(150); //Movimiento dedo anular cerrado
Menique.write(170); //Movimiento dedo meñique cerrado
delay(2000);

break;

    case 'd':          //Movimiento dedo anular abierto

Pulgar.write(180); //Movimiento dedo pulgar cerrado
Indice.write(0); //Movimiento dedo indice abierto
Medio.write(150); //Movimiento dedo medio cerrado
Anular.write(150); //Movimiento dedo anular cerrado
Menique.write(170); //Movimiento dedo meñique cerrado
delay(3000);

break;

```

```

default:          //Codigo abrir y cerrar la mano

    Pulgar.write(180); //Movimiento dedo pulgar abierto
    Indice.write(0); //Movimiento dedo indice abierto
    Medio.write(0); //Movimiento dedo medio abierto
    Anular.write(0); //Movimiento dedo anular abierto
    Menique.write(0); //Movimiento dedo menique abierto
    delay(2000);

    Pulgar.write(120); //Movimiento dedo pulgar cerrado
    Indice.write(170); //Movimiento dedo indice cerrado
    Medio.write(150); //Movimiento dedo medio cerrado
    Anular.write(150); //Movimiento dedo anular cerrado
    Menique.write(170); //Movimiento dedo menique cerrado
    delay(2000);

    Pulgar.write(180); //Movimiento dedo pulgar abierto
    Indice.write(0); //Movimiento dedo indice abierto
    Medio.write(0); //Movimiento dedo medio abierto
    Anular.write(0); //Movimiento dedo anular abierto
    Menique.write(0); //Movimiento dedo anular abierto
    delay(2000);

    Pulgar.write(120); //Movimiento dedo pulgar cerrado
    delay(3000);
    Pulgar.write(180); //Movimiento dedo pulgar abierto
    delay(3000);

    Indice.write(150); //Movimiento dedo indice cerrado
    delay(3000);
    Indice.write(0); //Movimiento dedo indice abierto
    delay(3000);

    Medio.write(150); //Movimiento dedo medio cerrado
    delay(3000);
    Medio.write(0); //Movimiento dedo medio abierto
    delay(3000);

    Anular.write(150); //Movimiento dedo anular cerrado
    delay(3000);
    Anular.write(0); //Movimiento dedo anular abierto
    delay(3000);

    Menique.write(170); //Movimiento dedo meñique cerrado
    delay(3000);
    Menique.write(0); //Movimiento dedo meñique abierto
    delay(3000);
}
}

```

*Fig. A.5 Gestión para la posición de los dedos*

```
#include <Servo.h>           //Incluir libreria servo
#define PIN_LED 4           //Definir pin 4 de led
#define PIN_MOT 5          //Definir pin 5 para servomotor

Servo mot;                  //crear el objeto motor

void setup() {
  Serial.begin(9600);      //Comunicación serial a 9600 baudios
  delay(100);              //Retardo de 100 ms
  pinMode(PIN_LED,OUTPUT); //Salida del led en el puerto 4
  digitalWrite(PIN_LED,0); //Pon el led a 0
  mot.attach(PIN_MOT);    //Configura al servo en el pin 5
  mot.write(0);           //Pon en 0 al servomotor
}

int vmot=0, vled=0, pos;   //Valores enteros del motor, led, y posicion
String cad,cad1,cad2;     //Valores tipo string para cad, cad1, cad2
void loop() {

  if (Serial.available()){ //si hay algo disponible
    cad = Serial.readString(); //lee los caracteres de la cadena
    pos = cad.indexOf(',');    //Busca una coma en la cadena
    cad1= cad.substring(0, pos); //Nos regresa la posición donde se encuentre la coma
    cad2= cad.substring(pos+1); //Iniciamos en la posición + 1, despues de la coma

    vled = cad1.toInt();      //Valor de led convertido a entero
    vmot = cad2.toInt();      //Valor del Motor a entero
    Serial.println(cad1);     //Imprime el valor del led 1
    Serial.println(cad2);     //Imprime el valor del motor
  }
  digitalWrite(PIN_LED,vled); // Pon el pin 4 en alto o bajo
  mot.write(vmot);           // Pon el servomotor en la posición deseada
}
```

## ***Anexo B. Hoja de datos técnica***



## FEATURES

- Fully integrated single-lead ECG front end
- Low supply current: 170  $\mu$ A (typical)
- Common-mode rejection ratio: 80 dB (dc to 60 Hz)
- Two or three electrode configurations
- High signal gain ( $G = 100$ ) with dc blocking capabilities
- 2-pole adjustable high-pass filter
- Accepts up to  $\pm 300$  mV of half cell potential
- Fast restore feature improves filter settling
- Uncommitted op amp
- 3-pole adjustable low-pass filter with adjustable gain
- Leads off detection: ac or dc options
- Integrated right leg drive (RLD) amplifier
- Single-supply operation: 2.0 V to 3.5 V
- Integrated reference buffer generates virtual ground
- Rail-to-rail output
- Internal RFI filter
- 8 kV HBM ESD rating
- Shutdown pin
- 20-lead 4 mm  $\times$  4 mm LFCSP package

## APPLICATIONS

- Fitness and activity heart rate monitors
- Portable ECG
- Remote health monitors
- Gaming peripherals
- Biopotential signal acquisition

## GENERAL DESCRIPTION

The **AD8232** is an integrated signal conditioning block for ECG and other biopotential measurement applications. It is designed to extract, amplify, and filter small biopotential signals in the presence of noisy conditions, such as those created by motion or remote electrode placement. This design allows for an ultralow power analog-to-digital converter (ADC) or an embedded microcontroller to acquire the output signal easily.

The **AD8232** can implement a two-pole high-pass filter for eliminating motion artifacts and the electrode half-cell potential. This filter is tightly coupled with the instrumentation architecture of the amplifier to allow both large gain and high-pass filtering in a single stage, thereby saving space and cost.

An uncommitted operational amplifier enables the **AD8232** to create a three-pole low-pass filter to remove additional noise. The user can select the frequency cutoff of all filters to suit different types of applications.

## FUNCTIONAL BLOCK DIAGRAM

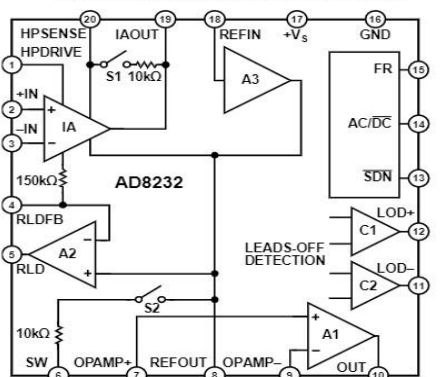


Figure 1.

To improve common-mode rejection of the line frequencies in the system and other undesired interferences, the **AD8232** includes an amplifier for driven lead applications, such as right leg drive (RLD).

The **AD8232** includes a fast restore function that reduces the duration of otherwise long settling tails of the high-pass filters. After an abrupt signal change that rails the amplifier (such as a leads off condition), the **AD8232** automatically adjusts to a higher filter cutoff. This feature allows the **AD8232** to recover quickly, and therefore, to take valid measurements soon after connecting the electrodes to the subject.

The **AD8232** is available in a 4 mm  $\times$  4 mm, 20-lead LFCSP package. Performance is specified from 0°C to 70°C and is operational from  $-40^{\circ}\text{C}$  to  $+85^{\circ}\text{C}$ .

Rev. A

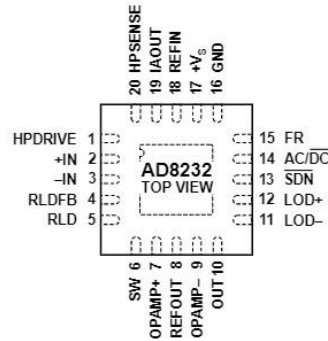
[Document Feedback](#)

Information furnished by Analog Devices is believed to be accurate and reliable. However, no responsibility is assumed by Analog Devices for its use, nor for any infringements of patents or other rights of third parties that may result from its use. Specifications subject to change without notice. No license is granted by implication or otherwise under any patent or patent rights of Analog Devices. Trademarks and registered trademarks are the property of their respective owners.

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.  
 Tel: 781.329.4700 ©2012–2013 Analog Devices, Inc. All rights reserved.  
[Technical Support](#) [www.analog.com](http://www.analog.com)



## PIN CONFIGURATION AND FUNCTION DESCRIPTIONS



NOTES  
1. CONNECT THE EXPOSED PAD TO GND OR LEAVE UNCONNECTED.

108561-002

Figure 2. Pin Configuration

Table 3. Pin Function Descriptions

Pin No.	Mnemonic	Description
1	HPDRIVE	High-Pass Driver Output. Connect HPDRIVE to the capacitor in the first high-pass filter. The AD8232 drives this pin to keep HPSENSE at the same level as the reference voltage.
2	+IN	Instrumentation Amplifier Positive Input. +IN is typically connected to the left arm (LA) electrode.
3	-IN	Instrumentation Amplifier Negative Input. -IN is typically connected to the right arm (RA) electrode.
4	RLDFB	Right Leg Drive Feedback Input. RLDFB is the feedback terminal for the right leg drive circuit.
5	RLD	Right Leg Drive Output. Connect the driven electrode (typically, right leg) to the RLD pin.
6	SW	Fast Restore Switch Terminal. Connect this terminal to the output of the second high-pass filter.
7	OPAMP+	Operational Amplifier Noninverting Input.
8	REFOUT	Reference Buffer Output. The instrumentation amplifier output is referenced to this potential. Use REFOUT as a virtual ground for any point in the circuit that needs a signal reference.
9	OPAMP-	Operational Amplifier Inverting Input.
10	OUT	Operational Amplifier Output. The fully conditioned heart rate signal is present at this output. OUT can be connected to the input of an ADC.
11	LOD-	Leads Off Comparator Output. In dc leads off detection mode, LOD- is high when the electrode to -IN is disconnected, and it is low when connected. In ac leads off detection mode, LOD- is always low.
12	LOD+	Leads Off Comparator Output. In dc leads off detection mode, LOD+ is high when the +IN electrode is disconnected, and it is low when connected. In ac leads off detection mode, LOD+ is high when either the -IN or +IN electrode is disconnected, and it is low when both electrodes are connected.
13	$\overline{\text{SDN}}$	Shutdown Control Input. Drive $\overline{\text{SDN}}$ low to enter the low power shutdown mode.
14	$\overline{\text{AC/DC}}$	Leads Off Mode Control Input. Drive the $\overline{\text{AC/DC}}$ pin low for dc leads off mode. Drive the $\overline{\text{AC/DC}}$ pin high for ac leads off mode.
15	FR	Fast Restore Control Input. Drive FR high to enable fast recovery mode; otherwise, drive it low.
16	GND	Power Supply Ground.
17	+Vs	Power Supply Terminal.
18	REFIN	Reference Buffer Input. Use REFIN, a high impedance input terminal, to set the level of the reference buffer.
19	IAOUT	Instrumentation Amplifier Output Terminal.
20	HPSENSE	High-Pass Sense Input for Instrumentation Amplifier. Connect HPSENSE to the junction of R and C that sets the corner frequency of the dc blocking circuit.
	EP	Exposed Pad. Connect the exposed pad to GND or leave it unconnected.

## LOW-PASS FILTERING AND GAIN

The AD8232 includes an uncommitted op amp that can be used for extra gain and filtering. For applications that do not require a high-order filter, a simple RC low-pass filter should suffice, and the op amp can buffer or further amplify the signal.

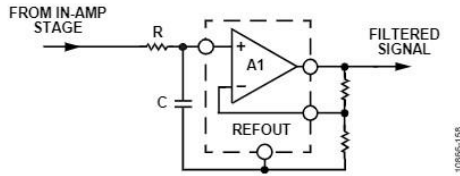


Figure 59. Schematic for a Single-Pole Low-Pass Filter and Additional Gain

Applications that require a steeper roll off or a sharper cut off, a Sallen-Key filter topology can be implemented, as shown in Figure 60.

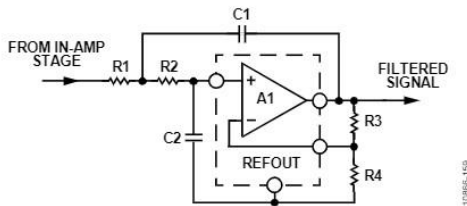


Figure 60. Schematic for a Two-Pole Low-Pass Filter

The following equations describe the low-pass cut off frequency, gain, and Q:

$$f_c = 1/(2\pi\sqrt{R1 C1 R2 C2})$$

$$\text{Gain} = 1 + R3/R4$$

$$Q = \frac{\sqrt{R1 \times C1 \times R2 \times C2}}{R1 \times C2 + R2 \times C2 + R1 \times C1(1 - \text{Gain})}$$

Note that changing the gain has an effect on Q and vice versa. Common values for Q are 0.5 to avoid peaking or 0.7 for maximum flatness and sharp cut off. A high value of Q can be used in narrow-band applications to increase peaking and the selectivity of the band-pass filter.

A common design procedure is to set  $R1 = R2 = R$  and  $C1 = C2 = C$ , which simplifies the expressions for cutoff frequency and Q to

$$f_c = 1/(2\pi RC)$$

$$Q = \frac{1}{3 - \text{Gain}}$$

Note that Q can be controlled by setting the gain with R3 and R4; however, this limits the gain to be less than 3. For gain values equal to or greater than 3, the circuit becomes unstable. A simple modification that allows higher gains is to make the value of C2 at least four times larger than C1.

It is important to note that these design equations only hold true in the case that the output impedance of the previous stage is much lower than the input impedance of the Sallen-Key filter. This is not the case when using an ac coupling network between

the instrumentation amplifier output and the input of the low-pass filter without a buffer.

To connect these two filtering stages properly without a buffer, make the value of R1 at least ten times larger than the resistor of the ac coupling network (labeled as R2 in Figure 55).

## DRIVING ANALOG-TO-DIGITAL CONVERTERS

The ability of AD8232 to drive capacitive loads makes it ideal to drive an ADC without the need for an additional buffer. However, depending on the input architecture of the ADC, a simple low-pass RC network may be required to decouple the transients from the switched-capacitor input typical of modern ADCs. This RC network also acts as an additional filter that can help reduce noise and aliasing. Follow the recommended guidelines from the ADC data sheet for the selection of proper R and C values.

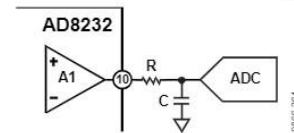


Figure 61. Driving an ADC

## DRIVEN ELECTRODE

A driven lead (or reference electrode) is often used to minimize the effects of common-mode voltages induced by the power line and other interfering sources. The AD8232 extracts the common-mode voltage from the instrumentation amplifier inputs and makes it available through the RLD amplifier to drive an opposing signal into the patient. This functionality maintains the voltage between the patient and the AD8232 at a near constant, greatly improving the common-mode rejection ratio.

As a safety measure, place a resistor between the RLD pin and the electrode connected to the subject to ensure that current flow never exceeds 10  $\mu\text{A}$ . Calculate the value of this resistor to be equal to the supply voltage across the AD8232 divided by 10  $\mu\text{A}$ .

The AD8232 implements an integrator formed by an internal 150 k $\Omega$  resistor and an external capacitor to drive this electrode. Choice of the integrator capacitor is a tradeoff between line rejection capability and stability. The capacitor should be small to maintain as much loop gain as possible, around 50 Hz and 60 Hz, which are typical line frequencies. For stability, the gain of the integrator should be less than unity at the frequency of any other poles in the loop, such as those formed by the patient's capacitance and the safety resistors. The suggested application circuits use a 1 nF capacitor, which results in a loop gain of about 20 at line frequencies, with a crossover frequency of about 1 kHz.

In a two-lead configuration, the RLD amplifier can be used to drive the bias current resistors on the inputs. Although not as effective as a true driven electrode, this configuration can provide some common-mode rejection improvement if the sense electrode impedance is small and well matched.

APPLICATION CIRCUITS

HEART RATE MEASUREMENT NEXT TO THE HEART

For wearable exercise devices, the AD8232 is typically placed in a pod near the heart. The two sense electrodes are placed underneath the pectoral muscles; no driven electrode is used. Because the distance from the heart to the AD8232 is small, the heart signal is strong and there is less muscle artifact interference. In this configuration, space is at a premium. By using as few external components as possible, the circuit in Figure 62 is optimized for size.

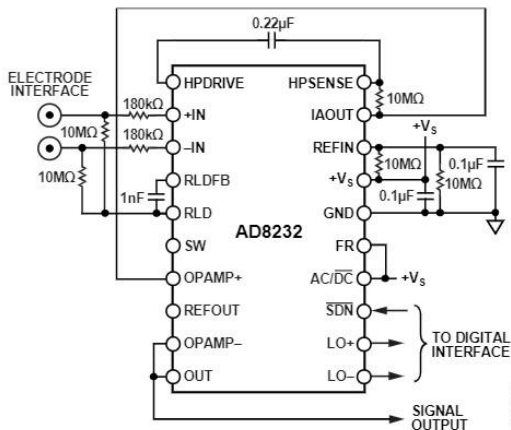


Figure 62. Circuit for Heart Rate Measurement Next to Heart

A shorter distance from the AD8232 to the heart makes this application less vulnerable to common-mode interference. However, since RLD is not used to drive an electrode, it can be used to improve the common-mode rejection by maintaining the midscale voltage through the 10 MΩ bias resistors.

A single-pole high-pass filter is set at 7 Hz, and there is no low-pass filter. No gain is used on the output op amp thereby reducing the number of resistors for a total system gain of 100.

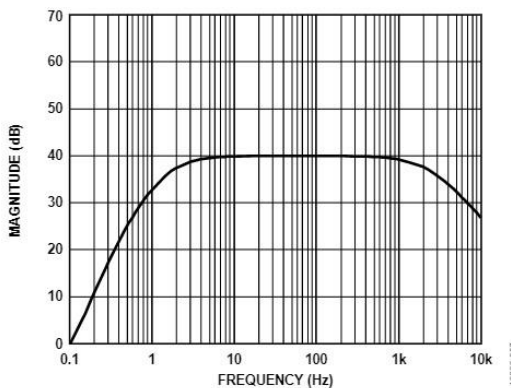


Figure 63. Frequency Response for HRM Next to Heart Circuit

The input terminals in this configuration use two 180 kΩ resistors, to protect the user from fault conditions. Two 10 MΩ resistors provide input bias. Use higher values for electrodes with high output impedance, such as chest electrodes.

The schematic also shows two 10 MΩ resistors to set the midscale reference voltage. If there is already a reference voltage available, it can be driven into the REFIN input to eliminate these two 10 MΩ resistors.

EXERCISE APPLICATION: HEART RATE MEASURED AT THE HANDS

In this application, the heart rate signal is measured at the hands with stainless steel electrodes. The user's arm and upper body movement create large motion artifacts and the long lead length makes the system susceptible to common-mode interference. A very narrow band-pass characteristic is required to separate the heart signal from the interferers.

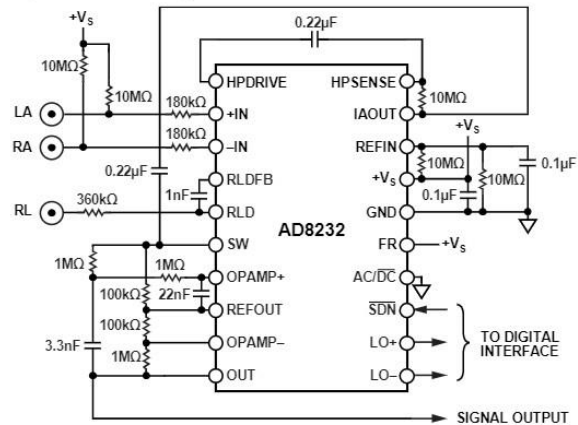


Figure 64. Circuit for Heart Rate Measurement at Hands

The circuit in Figure 64 uses a two-pole high-pass filter set at 7 Hz. A two-pole low-pass filter at 24 Hz follows the high-pass filters to eliminate any other artifacts and line noise.

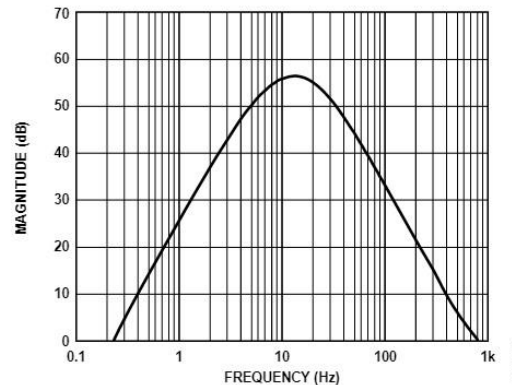


Figure 65. Frequency Response for HRM Circuit Taken at the Hands

The overall narrow-band nature of this filter combination distorts the ECG waveform significantly. Therefore, it is only suitable to determine the heart rate, and not to analyze the ECG signal characteristics.

The low-pass filter stage also includes a gain of 11, to bring the total system gain close to 1100 (note that the filter roll off prevents the maximum gain from reaching this value). Because the ECG signal is measured at the hands, it is weaker than when measured closer to the heart.

The RLD circuit drives to the third electrode, which can also be located at the hands, to cancel common-mode interference.

**CARDIAC MONITOR CONFIGURATION**

This configuration is designed for monitoring the shape of the ECG waveform. It assumes that the patient remains relatively still during the measurement, and therefore, motion artifacts are less of an issue.

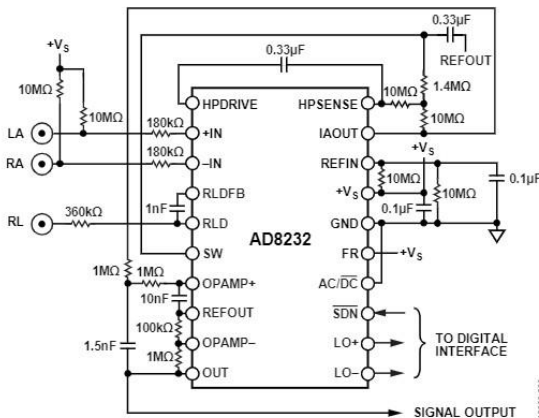


Figure 66. Circuit for ECG Waveform Monitoring

To obtain an ECG waveform with minimal distortion, the AD8232 is configured with a 0.5 Hz two-pole high-pass filter followed by a two-pole, 40 Hz, low-pass filter. A third electrode is driven for optimum common-mode rejection.

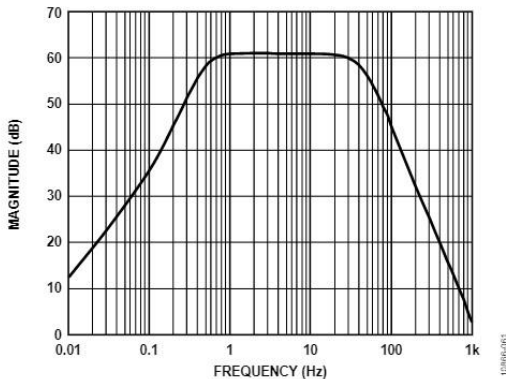


Figure 67. Frequency Response of Cardiac Monitor Circuit

In addition to 40 Hz filtering, the op amp stage is configured for a gain of 11, resulting in a total system gain of 1100. To optimize the dynamic range of the system, the gain level is adjustable, depending on the input signal amplitude (which may vary with electrode placement) and ADC input range.

**PORTABLE CARDIAC MONITOR WITH ELIMINATION OF MOTION ARTIFACTS**

The circuit in Figure 68 shows an implementation of a battery-powered embedded system for monitoring heart rate in applications where the patient engages in moderate activity, such as with a Holter monitor. The AD8232 uses a three-electrode patient interface and implements a two-pole high-pass filter with a cutoff at 0.3 Hz, and a two-pole low-pass filter with a cutoff frequency of 37 Hz. The total signal gain in the pass band is 400. The fully conditioned signal is sampled by the sigma-delta ADC integrated on the low power microcontroller, ADuCM360. The wide dynamic range of this ADC provides flexibility to reduce the signal gain to avoid saturation, depending on electrode placement.

Because the pass band is relatively wide for ambulatory applications, the ADXL346 accelerometer signal can be used to further minimize the noise introduced by the motion of the patient. Moreover, the microcontroller can use the motion information to monitor inactivity and to issue a system shutdown to save battery power.

The low dropout regulator ensures that the maximum of 3 V is not exceeded, especially during charge cycles of the battery, which can be a lithium-ion cell.

In this application, the ADuCM360 uses its Port 0 to perform DMA transfers to the host communication interface or to an on-board memory, if recording the waveform for later transfer. However, in any particular application, this port should be used for the busiest interface to minimize CPU cycles and maintain low power operation.

Note that this circuit is shown to demonstrate the capabilities of AD8232 and other system components. It is not a complete system design and additional effort must be made to ensure compliance with medical safety guidelines from regulatory agencies.

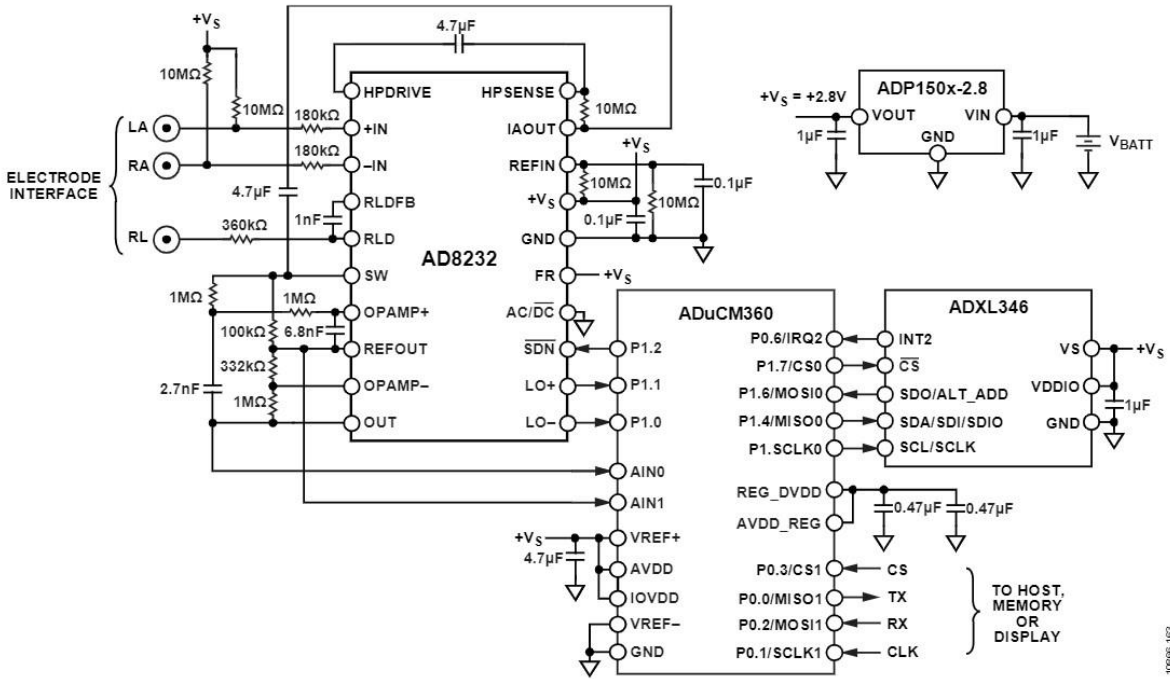


Figure 68. Low Power Portable Cardiac Monitor

10096-163

# 31150-MP

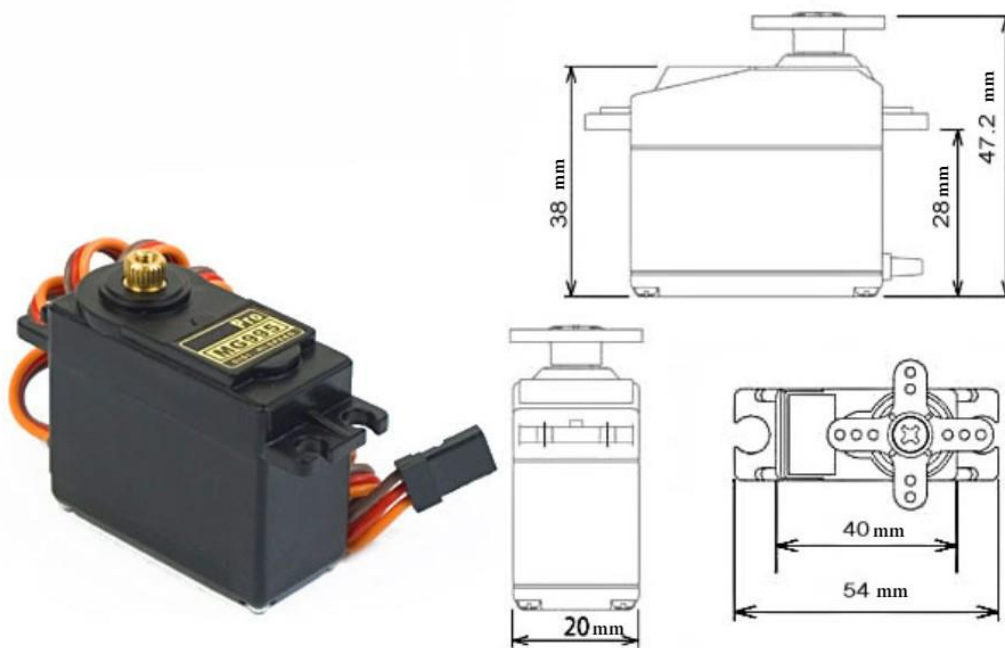
## MG995 High Speed Servo Actuator

The unit comes complete with color coded 30cm wire leads with a 3 X 1 pin 0.1" Pitch type female header connector that matches most receivers, including Futaba, JR, GWS, Cirrus, Blue Bird, Blue Arrow, Corona, Berg, Spektrum and Hitec.

This high-speed servo actuator is not code dependant; You can use any servo code, hardware or library to control them. The MG995 Actuator includes arms and hardware to get started.

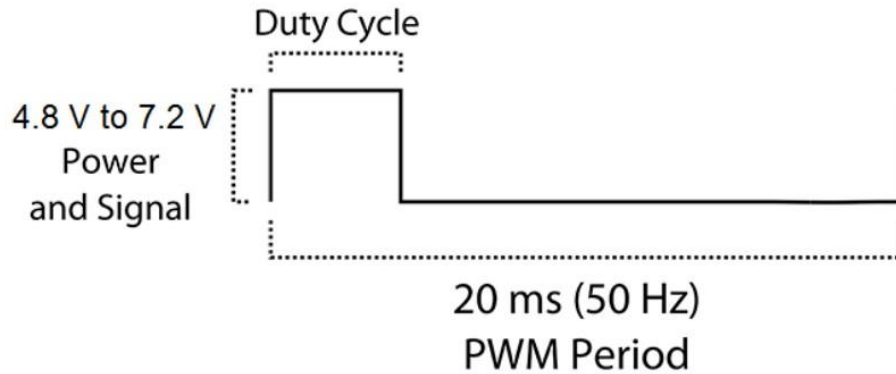
### Specifications

- Weight: 55 g
- Dimension: 40.7 x 19.7 x 42.9 mm approx.
- Stall torque: 8.5 kgf·cm (4.8 V ), 10 kgf·cm (6 V)
- Rotation Angle: 120deg. (+- 60 from center)
- Operating speed: 0.2 s/60° (4.8 V), 0.16 s/60° (6 V)
- Operating voltage: 4.8 V to 7.2 V
- Dead band width: 5  $\mu$ s
- Stable and shock proof double ball bearing design
- Metal Gears for longer life
- Temperature range: 0 °C – 55 °C



# 31150-MP

## MG995 High Speed Servo Actuator

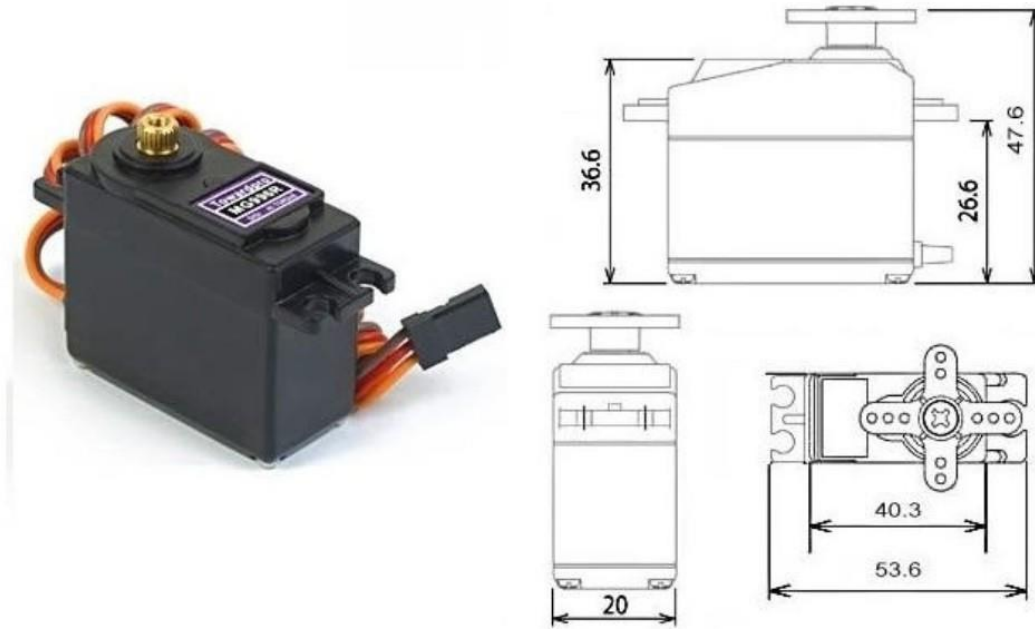


Information obtained from or supplied by mpja.com or Marlin P. Jones and Associates inc. is supplied as a service to our customers and accuracy is not guaranteed nor is it definitive of any particular part or manufacturer. Use of information and suitability for any application is at users own discretion and user assumes all risk.



**MARLIN P. JONES & ASSOC., INC.**  
P.O. Box 530400 Lake Park, FL 33403  
800-652-6733 FAX 561-844-8764  
WWW.MPJA.COM

# MG996R High Torque Metal Gear Dual Ball Bearing Servo



This High-Torque MG996R Digital Servo features metal gearing resulting in extra high 10kg stalling torque in a tiny package. The MG996R is essentially an upgraded version of the famous MG995 servo, and features upgraded shock-proofing and a redesigned PCB and IC control system that make it much more accurate than its predecessor. The gearing and motor have also been upgraded to improve dead bandwidth and centering. The unit comes complete with 30cm wire and 3 pin 'S' type female header connector that fits most receivers, including Futaba, JR, GWS, Cirrus, Blue Bird, Blue Arrow, Corona, Berg, Spektrum and Hitec.

This high-torque standard servo can rotate approximately 120 degrees (60 in each direction). You can use any servo code, hardware or library to control these servos, so it's great for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. The MG996R Metal Gear Servo also comes with a selection of arms and hardware to get you set up nice and fast!

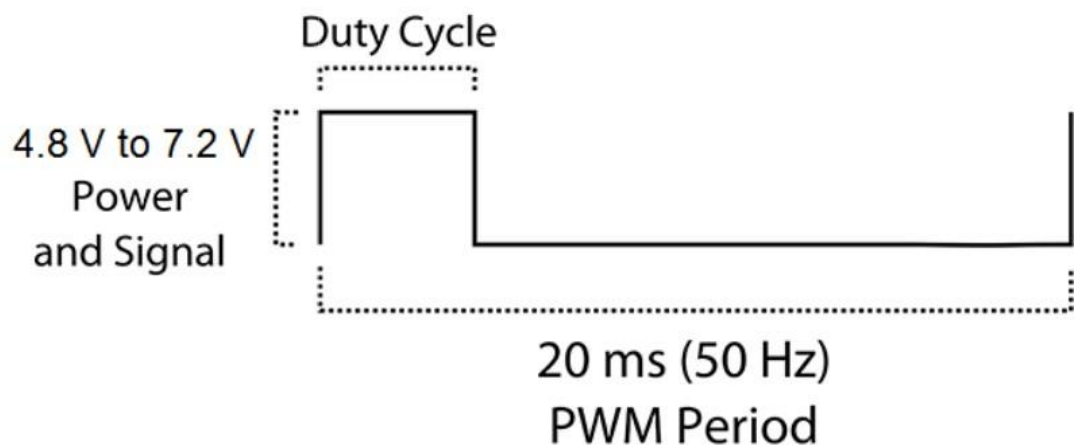
## Specifications

- Weight: 55 g
- Dimension: 40.7 x 19.7 x 42.9 mm approx.
- Stall torque: 9.4 kgf·cm (4.8 V), 11 kgf·cm (6 V)
- Operating speed: 0.17 s/60° (4.8 V), 0.14 s/60° (6 V)



- Operating voltage: 4.8 V a 7.2 V
- Running Current 500 mA – 900 mA (6V)
- Stall Current 2.5 A (6V)
- Dead band width: 5  $\mu$ s
- Stable and shock proof double ball bearing design
- Temperature range: 0  $^{\circ}$ C – 55  $^{\circ}$ C

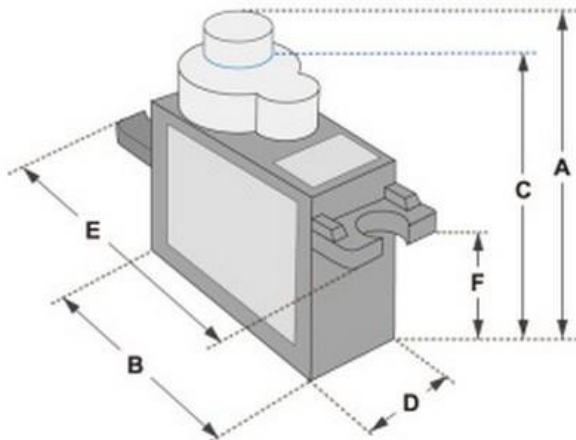
PWM=Orange ( $\square$ )  
 Vcc = Red (+)  
 Ground=Brown (-)

# SERVO MOTOR SG90



Tiny and lightweight with high output power. Servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but smaller. You can use any servo code, hardware or library to control these servos. Good for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. It comes with a 3 horns (arms) and hardware.



Dimensions & Specifications
A (mm) : 32
B (mm) : 23
C (mm) : 28.5
D (mm) : 12
E (mm) : 32
F (mm) : 19.5
Speed (sec) : 0.1
Torque (kg-cm) : 2.5
Weight (g) : 14.7
Voltage : 4.8 - 6

Position "0" (1.5 ms pulse) is middle, "90" (~2ms pulse) is middle, is all the way to the right, "-90" (~1ms pulse) is all the way to the left.

