



Universidad
Zaragoza

Trabajo Fin de Grado

Sistema para el control de una prótesis de mano por medio de señales electromiográficas.

System for the control of a prosthetic hand by electromiographical signals.

Autor

Pilar Rodríguez Barrio

Director

Daniel Mercado Barraqueta

ESCUELA DE INGENIERÍA Y ARQUITECTURA
2020



DECLARACIÓN DE AUTORÍA Y ORIGINALIDAD

(Este documento debe entregarse en la Secretaría de la EINA, dentro del plazo de depósito del TFG/TFM para su evaluación).

D./D^a. Pilar Rodríguez Barrio ,en

aplicación de lo dispuesto en el art. 14 (Derechos de autor) del Acuerdo de 11 de septiembre de 2014, del Consejo de Gobierno, por el que se aprueba el Reglamento de los TFG y TFM de la Universidad de Zaragoza,

Declaro que el presente Trabajo de Fin de (Grado/Máster)

Ing. Tecnologías Industriales

(Título del Trabajo)

Sistema para el control de una prótesis de mano por medio de señales electromiográficas.

es de mi autoría y es original, no habiéndose utilizado fuente sin ser citada debidamente.

Zaragoza, 1 de Septiembre de 2020

Fdo: Pilar Rodríguez Barrio

AGRADECIMIENTOS

Después de todo el trabajo realizado, escribo con gran satisfacción este apartado de agradecimientos a las personas que más cercanas han sido y más importantes para mí durante todo este proceso.

A mis padres Inmaculada y Manuel su paciencia, cariño y apoyo a lo largo de este tiempo, haberme brindado siempre todas las oportunidades que han tenido en sus manos y confiar en mí. A mi tía Carmen, por animarme a seguir estudiando, y ayudarme siempre en todo lo que ha podido y más y ser un gran modelo a seguir. Y a mi pareja Nacho, por estar en los días buenos y en los que no lo han sido tanto, por relajarme y por permitirme ponerle todos los electrodos que han sido necesarios. A mi tutor, Daniel Mercado por brindarme esta oportunidad. Y sin olvidarme de los amigos que me han apoyado y comprendido.

A todos muchas gracias.

Sistema para el control de una prótesis de mano por medio de señales electromiográficas.

RESUMEN

Las principales razones que implican la amputación de un miembro son las lesiones tumorales (cáncer), traumatismos (generalmente producidos en accidentes) o enfermedades del sistema vascular. En estos casos, la amputación se realiza como última medida y para salvar la vida del paciente, pero como consecuencia pierde las habilidades motoras que derivan de dicho miembro.

Con la realización de este trabajo se busca, conseguir que una persona con una amputación de mano pueda manejar una prótesis mediante los músculos del antebrazo. Las señales se obtendrán por medio de electromiografía superficial, el circuito diseñado permitirá su adquisición y filtrado y finalmente, el microprocesador permitirá la clasificación de las mismas.

Índice

1. Electromiografía - EMG.	9
1.1. Electroodos.	9
1.2. Características de la señal.	11
1.3. Posicionamiento y recomendaciones.	11
1.3.1. Preparación de la piel.	11
1.3.2. Colocación de los electrodos.	11
2. Electrónica.	13
2.1. Etapa de preamplificación.	14
2.2. Etapa sumadora.	15
2.3. Etapa de filtrado.	16
2.3.1. Filtro pasa bajo.	16
2.3.2. Filtro pasa alto.	19
2.3.3. Filtro rechaza banda.	20
2.4. Etapa de rectificación	21
2.5. Alimentación y protección del circuito	22
2.6. Simulaciones de los circuitos	23
2.7. Circuito impreso	25
3. Trabajo experimental y resultados	27
3.1. Montaje del circuito	27
3.2. Selección de los movimientos y anatomía del antebrazo.	28
3.3. Microprocesador - Arduino Uno.	29
3.4. Creación de una base de datos	29
3.5. Creación del método de clasificación	31
3.6. Resultados experimentales	32
4. Mejora del sistema a través de Python	35
4.1. Procesamiento de la señal - Transformada de Wavelet	35
4.2. Procesado de las señales	37

4.2.1. Pre-procesamiento de las señales	37
4.2.2. Eliminación del ruido a través de la DWT	37
4.2.3. Clasificación de las señales	39
4.3. Aplicación del clasificador a la base de datos del proyecto	40
5. Conclusiones y líneas futuras.	41
6. Bibliografía	43
Lista de Figuras	45
Lista de Tablas	47

Capítulo 1

Electromiografía - EMG.

La electromiografía, conocida como EMG, permite la medición de potenciales eléctricos en las fibras musculares, generados por la contracción de las células en el movimiento. Comenzó a usarse para la detección de lesiones musculares y como herramienta para generar diagnósticos, pero en la actualidad es uno de los métodos más utilizados para el control de prótesis y brazos robóticos.

En cuanto a la obtención de estas señales se tienen dos métodos, el primero de ellos no invasivo, conocido como método superficial, y el segundo el método intramuscular, en este caso, invasivo. A pesar de que el segundo método obtiene mejores resultados en la captación de señales, para casos como el de este trabajo se utiliza el método no invasivo ya que es más cómodo y seguro para el paciente.

La electromiografía de superficie se usa generalmente en músculos superficiales y se basa en un equilibrio químico creado entre la superficie de los electrodos y la de la piel gracias a una conducción electrolítica, permitiendo el paso de corriente a través del electrodo.

1.1. Electrodos.

Como se ha comentado en la introducción del capítulo, hay dos grandes grupos de electrodos, intramusculares y superficiales. Dentro de estos grupos se tienen nuevas diferenciaciones, como se aprecia en el esquema de la Figura 1.1, el intramuscular puede ser coaxial o monopolar y el superficial activo o pasivo. Dado que se va a usar el método no invasivo, nos centramos en estos. Este tipo de electrodos se puede clasificar también en función del material del que está formado y en función de si son secos o húmedos.

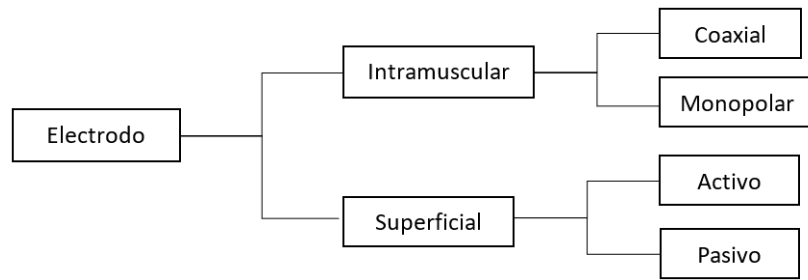


Figura 1.1: Esquema de los electrodos.

La diferencia entre los electrodos activos y pasivos es que los primeros, incluyen amplificadores que mejoran la adquisición de las señales. Es decir, incluyen una etapa de preamplificación a diferencia de los segundos. Los electrodos pasivos están formados por un pequeño disco metálico altamente conductivo y un material adhesivo que permite la fijación a la piel, en la Figura 1.2 se puede apreciar una comparación entre ambos. Cabe destacar que los activos tienen un tamaño y peso superior a los pasivos.



Figura 1.2: Electrodos de superficie pasivos (a) y activos (b)

Por otro lado, los electrodos húmedos son aquellos que necesitan un gel electrolítico como interfase entre la piel y el metal, mientras que con los secos el metal entra en contacto directo con la piel. Estos requerimientos dependen del material del que puede variar entre los siguientes:

- Plata y cloruro de plata: es el tipo de electrodo más usado en EMG debido a su estabilidad y bajo ruido. Son electrodos húmedos, por lo que requieren el uso de un electrolito que suele ser cloruro de sodio o potasio.
- Oro: generalmente utilizados en electroencefalografías, son reutilizables y poseen un coste elevado.
- Polímero conductor: están formados por un polímero adhesivo que permite la conducción de la señal hacia un fondo de aluminio o plata. Como contra, son muy resistivos y no son buenos para señales de bajo potencial.

- Carbón o metal: Son los menos usados debido a su alta resistividad, pero son reutilizables y económicos.

1.2. Características de la señal.

La señal generada por los músculos es de unos $250 \mu\text{V}$ durante la contracción, pudiendo llegar en algún caso a unos 10 mV . Por otro lado, otras características de la señal como la frecuencia o la amplitud dependen de diversos factores como pueden ser:

1. Tiempo e intensidad.
2. Distancia del electrodo y zona de colocación de este.
3. Propiedades de la piel, del electrodo o de la electrónica.
4. Calidad del contacto entre piel y electrodo.

Por otro lado, se sabe que la frecuencia se encuentra en un rango de 20 a 500 Hz con mayor potencia entre 50 y 200 Hz. [1],[2]

1.3. Posicionamiento y recomendaciones.

Para este apartado se toma como punto de referencia el estudio SENIAM, que es una recomendación europea para sensores y su disposición así como procedimientos de obtención y tratamiento de señales EMG. Toda esta información puede consultarse en la web oficial www.seniam.org.

1.3.1. Preparación de la piel.

La piel puede ser considerada como una barrera que separa al electrodo del músculo. Es por ello, que es necesario reducir la impedancia de esta, tanto como sea posible. Para ello, se debe eliminar el pelo de la zona de colocación de los electrodos, por lo cual se realiza un rasurado; también se recomienda eliminar las células muertas mediante un proceso de exfoliación. Por último, es importante eliminar la humedad y o sudor de la piel, para lo que se recomienda la limpieza de la zona con alcohol.

1.3.2. Colocación de los electrodos.

Para la adquisición de una señal de mayor calidad se recomienda la disposición bipolar de electrodos, es decir, el uso de dos electrodos o superficies de detección por músculo. La distancia que deberá existir entre los centros del área de conducción

de cada electrodo debe ser de unos 20mm, o inferior a un cuarto de la longitud del músculo, cuando se trata de músculos pequeños.

Por otro lado, en cuanto a la disposición de los electrodos en el músculo se hablará de localización longitudinal y transversal. Respecto a la primera, se recomienda colocar los sensores en mitad del músculo y la zona del tendón distal. En cuanto a la disposición transversal, se recomienda colocar los electrodos alejados de la zona en la que el músculo delimite con otro. En la Figura 1.3 se puede apreciar la colocación ideal de cada par de electrodos. Estos deben estar situados entre la unidad motora y el tendón de forma que se capte la mayor densidad de fibras posible.

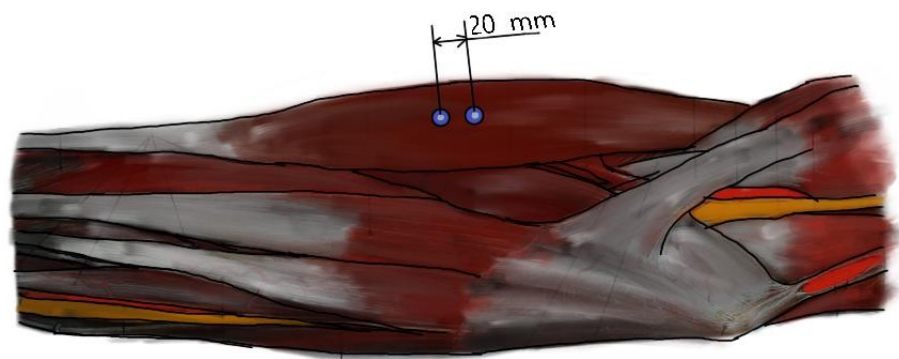


Figura 1.3: Posicionamiento ideal de los electrodos en el músculo.

Se usará también un electrodo de referencia, actuando de tierra para la señal, que deberá colocarse lejos de los electrodos de captación. En concreto, en una zona neutra, por ejemplo, sobre los huesos del codo o del hombro.

Para este trabajo se decide fijar el número de electrodos en 7, es decir, se estudiarán tres músculos, mas las referencia.

Capítulo 2

Electrónica.

La señal obtenida a través de los electrodos tendrá muy baja amplitud además de gran cantidad de ruido. Es por ello, que antes de poder intentar realizar su clasificación se necesita acondicionar las señales. En el esquema de la Figura 2.1 se muestra un diagrama de bloques de sistema de obtención y tratamiento de las mismas. Por tanto, se necesitará un circuito de preamplificación, dado que se usarán electrodos pasivos; filtrado, para eliminar el ruido; amplificación, rectificación para obtener una señal positiva y conversión analógica a digital. Además de estos elementos básicos, se añadirá una etapa sumadora tras la preamplificación, con el objetivo de reducir los elementos electrónicos y facilitar su tratamiento. Por otro lado, se necesitará un microcontrolador, que será programado para clasificar las señales y poder realizar los movimientos deseados.

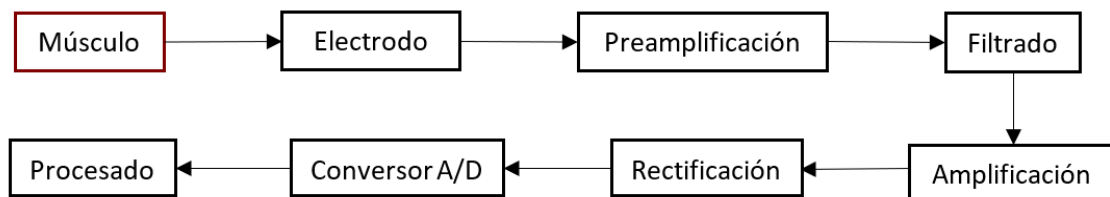


Figura 2.1: Diagrama de bloques del sistema para la obtención de señales.

Para afinar y comprobar el diseño de las diferentes partes del circuito se utilizará el software LTSpice. En el anexo ??, se indica cómo se diseñan los filtros mediante simulaciones y los resultados obtenidos en estas.

Por otro lado, con el objetivo de reducir el ruido de los cables y dada la importancia de este por el bajo voltaje de las señales, los circuitos diseñados a continuación se diseñan también en PCB por medio del programa EAGLE, un software libre de diseño de circuitos que permite crear también el circuito impreso (PCB).

En cuanto a la conversión analógico-digital se puede realizar directamente a través del microcontrolador, sin necesidad de incluir módulos externos.

2.1. Etapa de preamplificación.

Como se ha comentado con anterioridad, la señal captada por los electrodos varía entre los $250 \mu\text{V}$ y 10 mV , por tanto, antes de poder realizar el filtrado necesitaremos amplificar la señal. Para seleccionar un amplificador, se tendrán en cuenta parámetros de la señal como son frecuencia y voltaje y los siguientes parámetros del amplificador.

1. Limitaciones de entrada: Corrientes por las entradas y tensión offset de entrada. Esta última produce un error y debe compensarse si la tensión es continua y del orden de la tensión de offset V_{OI} . Este voltaje aparece cuando la tensión que llega por las entradas del amplificador es nula.
2. Impedancias de entrada y salida: se busca una impedancia de entrada elevada y una de salida baja. De esta forma a la entrada se consigue reducir la corriente introducida acercándolo al funcionamiento ideal.
3. Limitaciones de salida: intensidad máxima de salida, rango de tensiones de salida y Slew-rate (velocidad de respuesta) que cuanto mayor sea su valor, más lenta será la respuesta.
4. CMRR (relación de rechazo al modo común): es la capacidad del amplificador operacional de ignorar los cambios de voltaje que son comunes en ambas entradas. (Un valor típico se encontraría en los 90 dB y disminuye con la frecuencia).
5. Ganancia: factor de amplificación de la señal.

Por tanto, se busca un amplificador que posea un CMRR alto, una impedancia de salida baja, que no incremente el ruido y un nivel bajo de offset. Pueden usarse amplificadores diferenciales y de instrumentación. En ambos casos el sistema no amplifica el ruido gracias a que el voltaje de este es común en las entradas del amplificador. Finalmente, se selecciona uno de instrumentación ya que presenta ventajas frente al diferencial como una ganancia variable y una impedancia de entrada superior. En concreto el amplificador INA 128, cuyo esquema queda representado en la Figura 2.2.

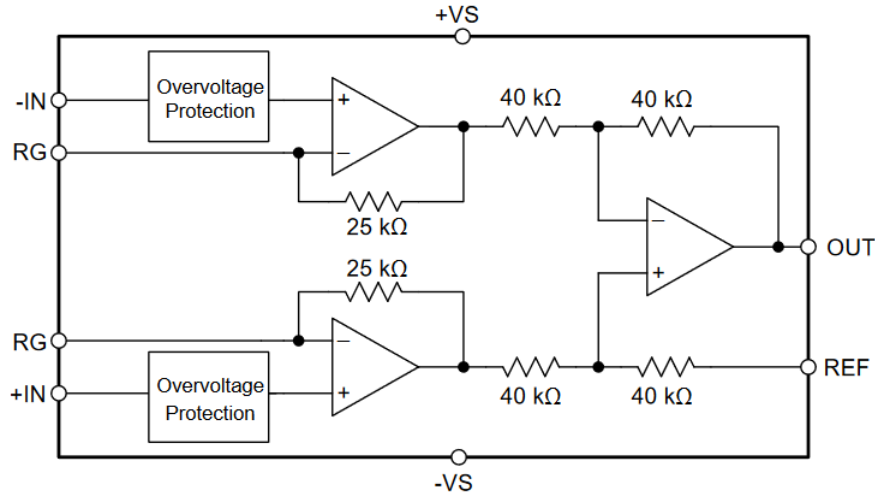


Figura 2.2: Esquema del amplificador INA 128 fuente Texas Instruments.

Se ajusta para tener una ganancia de 200 fijando el valor de la resistencia que se colocaría según la Figura 2.2 entre los puntos llamados R_G . Se utiliza para este cálculo la ecuación 2.1 obtenida de la ficha técnica del amplificador adjunta en el anexo ??.

$$G = 1 + \frac{50k\Omega}{R_G} \quad (2.1)$$

Se decide utilizar una resistencia de 220Ω lo que equivale a una ganancia de 228.

2.2. Etapa sumadora.

Como se ha comentado en el 1.3.2 se utilizan 3 pares de electrodos, más la referencia para la adquisición de señales, a lo que hay que sumar el electrodo de referencia. Por tanto, el circuito descrito en el apartado anterior que hace referencia a la preamplificación debería aplicarse a todos estos pares de electrodos.

Con el principal objetivo de reducir los componentes del circuito, se decide incluir una etapa sumadora, de forma que los circuitos posteriores no tengan que aplicarse de forma individual a cada señal como pasa en el circuito de preamplificación. Esta etapa se representa en la Figura 2.3, donde las entradas V_1 a V_n serán para el caso de estudio las señales electromiográficas. Por otro lado, las resistencias R_1 a R_n poseerán el mismo valor, dado que todas las señales se quieren sumar por igual.

Se designa como R a la resistencia que se encuentra en la entrada de cada señal y R' a la resistencia que se encuentra en la realimentación negativa del amplificador. La ecuación 2.3 permite el cálculo de las resistencias R y R' .

$$V_o = -\left(\frac{R'}{R} \times V_1 + \frac{R'}{R} \times V_2 + \frac{R'}{R} \times V_n\right) \quad (2.2)$$

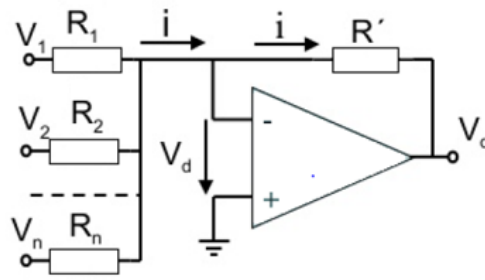


Figura 2.3: Estructura de un inversor sumador de n señales

La suma de las señales, y teniendo en cuenta que se obtendrán tres de estas, V_1 a V_3 será igual a V_o por lo que la ecuación anterior queda resumida en la ecuación 2.2.

$$R = R' \times 3 \quad (2.3)$$

Se decide fijar el valor de la resistencia de realimentación R' a $10 \text{ k}\Omega$, siendo el valor de las resistencias restantes es $30 \text{ k}\Omega$.

Para este caso, el amplificador seleccionado, se trata de un amplificador operacional, en concreto el TL084, que se utilizará además en las etapas posteriores.

2.3. Etapa de filtrado.

En primer lugar, será importante conocer qué parte del filtrado queremos realizar de forma analógica y cuál de forma digital. El primer tipo de corrección tiene como ventajas frente al segundo su sencillez mientras que, por otro lado, los filtros digitales, una vez diseñados permiten realizar cambios a través del código.

2.3.1. Filtro pasa bajo.

El filtro pasa bajo será diseñado de forma que permita el paso de frecuencias por debajo de los 500 Hz . Para la selección de este se deben tener en cuenta los distintos filtros activos existentes. Entre ellos los más comunes son:

- Butterworth: se basa en la máxima uniformidad de la banda pasante. Es decir, optimiza la respuesta plana en la banda de paso y se utiliza cuando se necesita buena precisión en dicha banda.
- Chebyshev: Optimiza la transición, utilizándose cuando el contenido de frecuencias es más importante que la magnitud.

- Bessel: Se tiene una respuesta lineal respecto a la fase lo que implica un retardo en todo el ancho de banda. Optimiza la respuesta en fase.

La comparación de estos filtros ante diferentes magnitudes puede apreciarse en la Figura 2.4 en la que se tiene en color rojo un filtro Bessel de tercer orden, un filtro Butterworth de segundo orden y un filtro Chebyshev de tercer orden. Estas gráficas se han realizado mediante la aplicación de diseño de filtros de Texas Instruments.

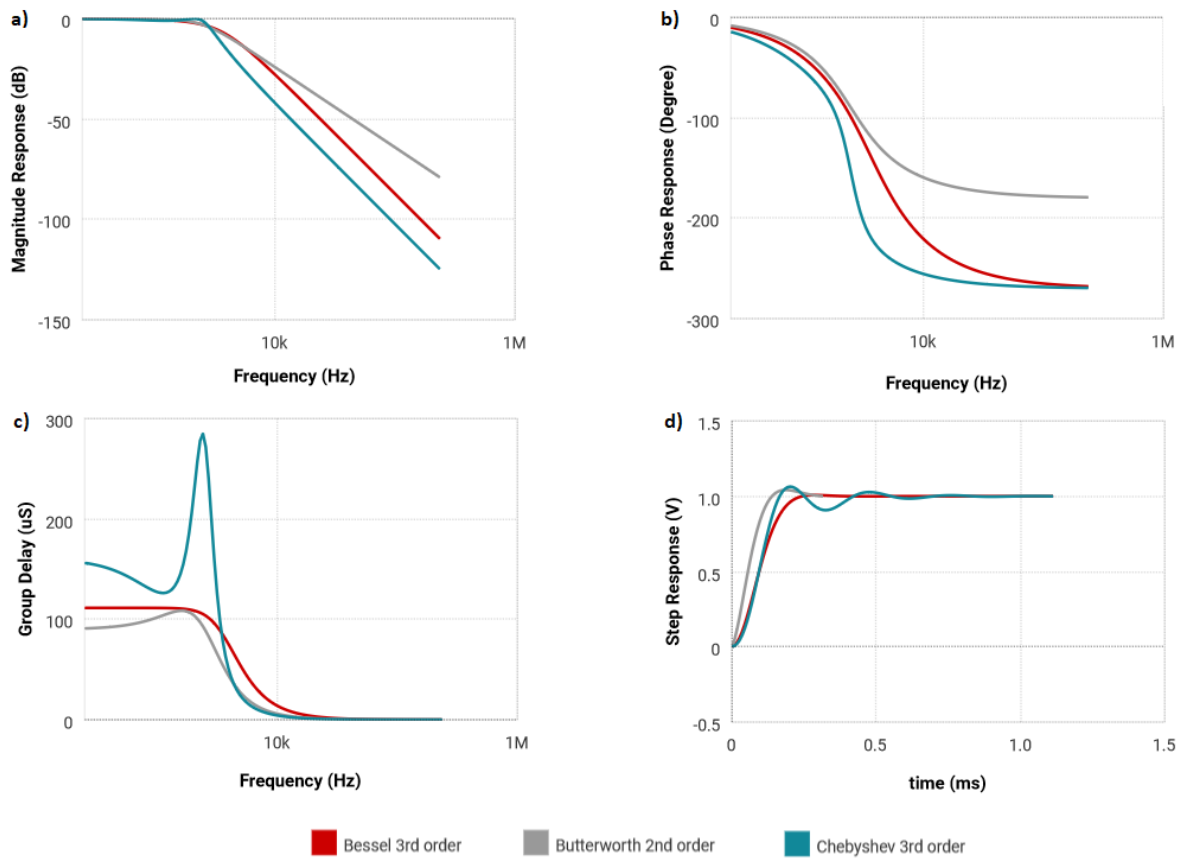


Figura 2.4: Gráfica de comparación de filtros Bessel, Butterworth y Chebyshev, a) comparación en magnitud. b) comparación en fase. c) comparación en retraso. d) comparación de respuesta ante entrada escalón.

La diferencia de los órdenes de estos filtros se debe a que se han realizado comparando su factor de calidad Q , que representa la calidad del polo. Siendo para el caso anterior de 0,691 para el filtro Bessel, 0,707 para el filtro Butterworth y 2,018 para el filtro Chebyshev.

Con el fin de seleccionar el filtro se calcula el factor de calidad necesario mediante las ecuaciones 2.4 y 2.5, donde f_m es la frecuencia central, f_H la frecuencia de corte superior y f_L la de corte inferior. Los valores obtenidos son una frecuencia central o

media de 100 Hz y un factor de calidad de 0,208.

$$Q = \frac{f_m}{f_H - f_L} \quad (2.4)$$

$$f_m = \sqrt{f_H \times f_L} \quad (2.5)$$

Dada la importancia de precisión en la banda pasante y la necesidad de un valor de Q bajo se selecciona el filtro Butterworth. En cuanto a la topología dado que se trata de un ancho de banda estrecho se plantean únicamente la topología Sallen-Key y la Multiple FeedBack, representadas en la Figura 2.5. La topología tipo Multiple FeedBack, tiene una mayor precisión ya que no depende de la ganancia del filtro, pero carece de flexibilidad en cuanto a la tolerancia de los valores de los componentes pasivos; se usa para valores de Q altos. Puesto que el filtro Sallen-Key es más simple y que el valor de Q requerido es bajo se selecciona este para el filtro pasa-bajo.

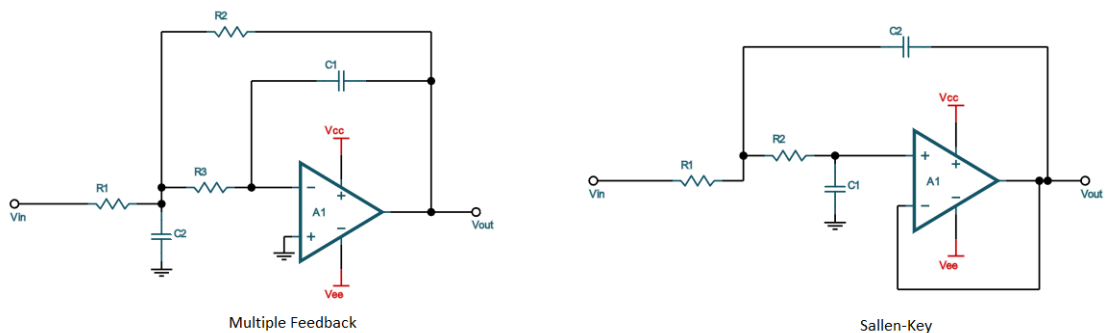


Figura 2.5: Topología Multiple Feedback frente a topología Sallen-Key.

Para el cálculo de los valores de las resistencias y condensadores es necesario conocer los siguientes parámetros:

- Orden del filtro, n: para este caso es 2.
- Número del filtro parcial, i: en el caso de orden 2 sólo puede ser 1.
- Coeficientes del filtro, ai y bi, que se obtienen a partir de la tabla de la figura 2.6, siendo esta una tabla de coeficientes normalizados para una frecuencia de corte de 1. Se entra a esta tabla con el orden n y el número de filtros parciales i.
- Coeficiente entre la frecuencia de corte del filtro parcial y la del filtro total. De nuevo se obtiene de la tabla de la figura 2.6.
- Factor de calidad de cada filtro parcial, Q: este valor aparece en la tabla, y en la ecuación 2.4 se había calculado el valor necesario para este caso.

<i>Butterworth</i>					
n	l	a_l	b_l	$k_l = \frac{f_{Cl}}{f_c}$	Q_l
1	1	1.0000	0.0000	1.000	—
2	1	1.4142	1.0000	1.000	0.71
3	1	1.0000	0.0000	1.000	—
	2	1.0000	1.0000	1.272	1.00
4	1	1.8478	1.0000	0.719	0.54
	2	0.7654	1.0000	1.390	1.31
5	1	1.0000	0.0000	1.000	—
	2	1.6180	1.0000	0.859	0.62
	3	0.6180	1.0000	1.448	1.62
6	1	1.9319	1.0000	0.676	0.52
	2	1.4142	1.0000	1.000	0.71
	3	0.5176	1.0000	1.479	1.93
7	1	1.0000	0.0000	1.000	—
	2	1.8019	1.0000	0.745	0.55
	3	1.2470	1.0000	1.117	0.80
	4	0.4450	1.0000	1.499	2.25

Figura 2.6: Tabla de coeficientes para filtro Butterworth

Se obtiene de esta forma la función de transferencia para el filtro de orden $n = 2$ siendo esta: $(s^2 + 1,4142s + 1)$. Una vez obtenida la función se elige el valor del condensador representado como C_1 en la Figura 2.5, tomándose un valor de 47 nF, tras ello se aplican las ecuaciones 2.6, 2.7 y 2.8 para calcular los valores del condensador C_2 y de las resistencias R_1 y R_2 , donde el valor de f_c es la frecuencia de corte, que como se ha indicado anteriormente es de 500 Hz.

$$C_2 \geq C_1 \times \frac{4 \times b_1}{a_1^2} \quad (2.6)$$

En el cálculo del condensador C_2 , se obtiene que su valor deberá ser superior a los 94 nF, para normalizar este valor, se selecciona un condensador de 150 nF y se procede al resto de cálculos.

$$R_1 = \frac{a_1 \cdot C_2 + \sqrt{a_1^2 \cdot C_2^2 - 4b_1 \cdot C_1 \cdot C_2}}{4 \cdot \pi \cdot f_c \cdot C_1 \cdot C_2} = 6k\Omega \quad (2.7)$$

$$R_2 = \frac{a_1 \cdot C_2 - \sqrt{a_1^2 \cdot C_2^2 - 4b_1 \cdot C_1 \cdot C_2}}{4 \cdot \pi \cdot f_c \cdot C_1 \cdot C_2} = 20k\Omega \quad (2.8)$$

De acuerdo con las ecuaciones 2.7 y 2.8 y los resultados obtenidos en las simulaciones del anexo ??, los valores de las resistencias una vez normalizadas son de 6 y 20 k Ω R_1 para de 6.8 k Ω para R_2 .

2.3.2. Filtro pasa alto.

En el caso del filtro pasa alto, este se diseña para que permita el paso de frecuencias por encima de los 20 Hz. En cuanto a su topología se selecciona de la misma forma

que el filtro pasa bajo. Por tanto, de nuevo se utiliza un filtro Butterworth con dos polos y topología Sallen-Key, como queda representado en la Figura 2.7.

El cálculo de los componentes se realiza partiendo del mismo polinomio de transferencia que en el apartado 2.3.1 de diseño del filtro pasa-bajo. En el caso de este filtro se toma el mismo valor para C_1 y C_2 , en concreto, se escoge un condensador de 100 nF, y se aplican las ecuaciones 2.9 y 2.10 para el cálculo de las resistencias.

$$R_1 = \frac{1}{4 \cdot \pi \cdot f_c \cdot C \cdot a_1} \quad (2.9)$$

$$R_2 = \frac{1}{4 \cdot \pi \cdot f_c \cdot C \cdot b_1} \quad (2.10)$$

Aproximando los resultados obtenidos, se decide utilizar una resistencia R_1 de valor 28 k Ω y una R_2 de 40.2 k Ω

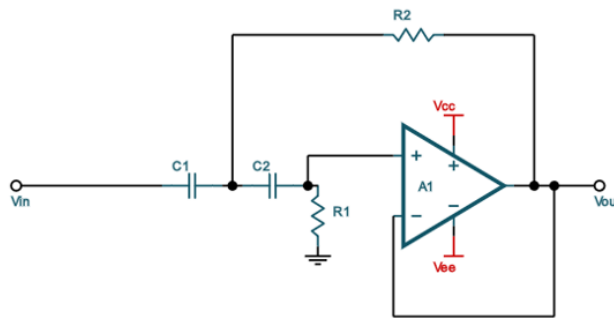


Figura 2.7: Filtro paso alto Butterworth con topología Sallen-Key

2.3.3. Filtro rechaza banda.

Este filtro, permitirá eliminar el ruido introducido a una frecuencia de 50 Hz. El filtro rechaza banda, puede crearse como una combinación de los filtros pasa-bajo y pasa-alto, pero no se consigue una buena selectividad, consiguiéndose como mucho un factor de calidad Q de 0.25. Se utiliza para ello el filtro de la Figura 2.8. Este filtro esta compuesto por un filtro activo en doble T al que se le ha añadido un operacional para mejorar el factor de calidad.

Para calcular los parámetros de este último filtro se aplican las siguientes ecuaciones, en las que fijamos el factor de calidad Q a 10, y f_m a 50 Hz. Por último, se decide fijar el valor del condensador C a 68 nF.

$$Q = \frac{1}{2 \cdot (2 - G)} = 10 \quad (2.11)$$

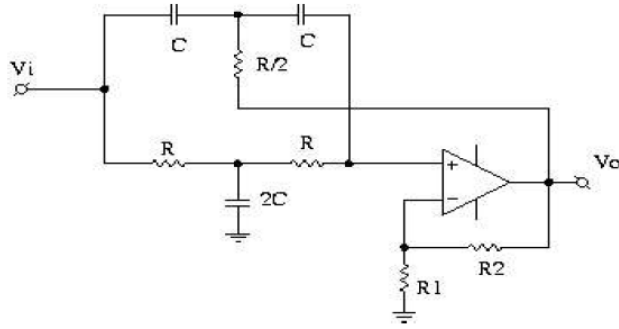


Figura 2.8: Filtro rechaza banda

$$G = 1 + \frac{R_2}{R_1} \quad (2.12)$$

$$f_m = \frac{1}{2 \cdot \pi \cdot R \cdot C} = 50Hz \quad (2.13)$$

De esta forma se obtiene una ganancia de 1.95, una relación de 0.95 entre las resistencias R_1 y R_2 , (lo que implica que $R_2 = 0.95 \cdot R_1$) y finalmente, gracias a la elección del condensador, una resistencia R de 47 k Ω .

2.4. Etapa de rectificación

La etapa de rectificación permitirá obtener una señal positiva. Esto es necesario, ya que el rango de tensiones de los pines de entrada del Arduino se encuentra entre los 0 y 5 V.

Con el objetivo de conseguir una señal positiva puede utilizarse un diodo colocado en serie, pero solo realizaría un rectificado de media onda, eliminando el contenido negativo de la señal. En concreto, el diodo comenzaría a conducir cuando su tensión de entrada superase los 0.7V. Para obtener una rectificación de onda completa se podría usar un puente de diodos como el que aparece en la Figura 2.9. Como consecuencia del uso de este puente la onda tendría una reducción de amplitud de 1.4V ya que cada vez conducirían dos diodos, reduciendo la amplitud en 0.7 V cada uno.

Para eliminar esta disminución de la amplitud de la onda la mejor opción es usar un rectificador de precisión. Este se construye a base de amplificadores operacionales, diodos y resistencias y al igual que con solo los diodos se tiene la opción de realizar un rectificado de media onda o de onda completa. En la Figura 2.10a se puede apreciar un rectificador de precisión de onda completa, frente al rectificador de media onda

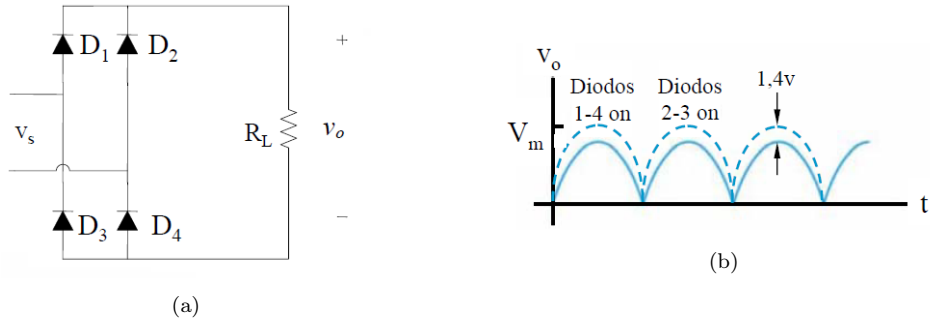


Figura 2.9: a)Puente de diodos. b) Señal rectificada gracias al puente de diodos

de la Figura 2.10b. Cabe destacar que este circuito de rectificación de onda completa doblaría la frecuencia de la señal, frente al de media onda que únicamente eliminaría la parte negativa de esta.

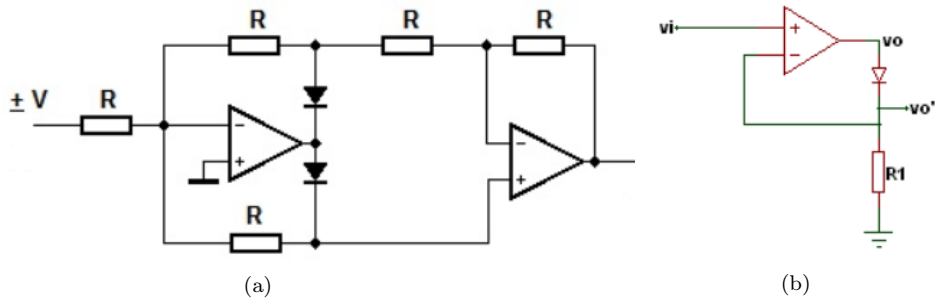


Figura 2.10: Rectificadores de precisión. a)Rectificador de onda completa b) Rectificador de media onda

2.5. Alimentación y protección del circuito

Dado que la tensión máxima de entrada del Arduino es de 5 V es necesario introducir un circuito de protección que permita asegurar que no se supera dicho valor en los pines de entrada. Para conseguirlo, se utiliza un diodo Zéner de 5.1 V y una resistencia de 220 Ω como queda representado en la Figura 2.11.

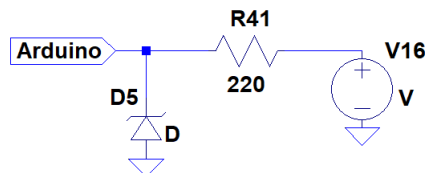


Figura 2.11: Protección de los pines de entrada del Arduino

En cuanto a la alimentación de los operacionales usados, los rangos de valores de los mismos se recogen en sus respectivas hojas de características situadas en los anexos

?? y ???. Se decide usar dos pilas de 9 V, que permitirán suministrar -9 V y 9 V. La tensión positiva, permitirá también la alimentación del microprocesador, que permite voltajes entre 7 y 12V.

2.6. Simulaciones de los circuitos

Para comprobar el correcto diseño de los circuitos, se utiliza la herramienta LTSpice, que es un software gratuito utilizado para la simulación de circuitos.

Se omiten en esta sección las amplificaciones de las señales debido a que únicamente se aumenta su amplitud. Por tanto, se crean en primer lugar las señales que serían percibidas por los electrodos. Dado que se tienen tres pares de electrodos de entrada, se crean tres señales. Estas se crean como la suma de varias señales senoidales con diferentes frecuencias y amplitudes, obteniéndose las señales representadas en la Figura 2.12. La composición de estas, se recoge en la Tabla 2.1, en la que se aprecian la frecuencia y amplitud de las mismas. Estas señales deberían ser las señales obtenidas tras la etapa de pre-amplificación.

	f_1	V_1	f_2	f_2	f_3	V_3	f_4	f_4
señal 1	100	1.5	650	0.5	10	0.3	50	0.7
señal 2	230	0.25	650	0.4	10	0.2	50	0.15
señal 3	115	0.75	630	0.4	5	0.3	50	0.18

Tabla 2.1: Composición de las señales para las simulaciones del circuito en Hz y V.

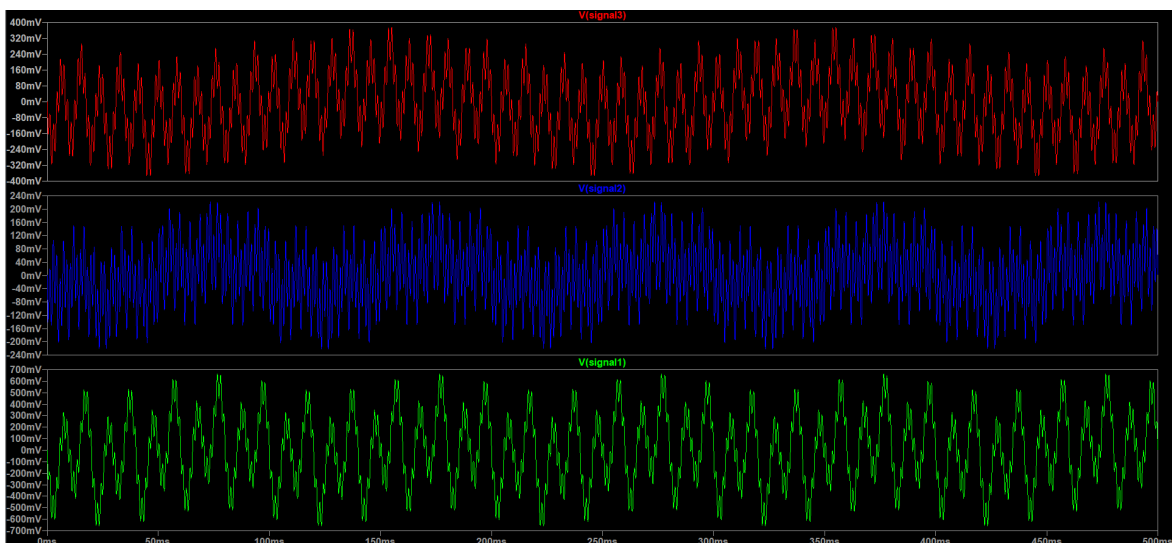


Figura 2.12: Señales creadas por medio de LTSpice para la comprobación de los circuitos

En las siguientes fases las señales pasarán por un sumador, convirtiéndose en una sola y a continuación se somete a los tres filtros para finalmente ser rectificadas. Para comprobar la eliminación del ruido gracias a los filtros, se realiza la transformada de Fourier que permite observar las componentes de la señal en función de la frecuencia. Este diagrama se representa en la Figura 2.13, en la que se aprecia como los filtros van eliminando las componentes no deseadas.

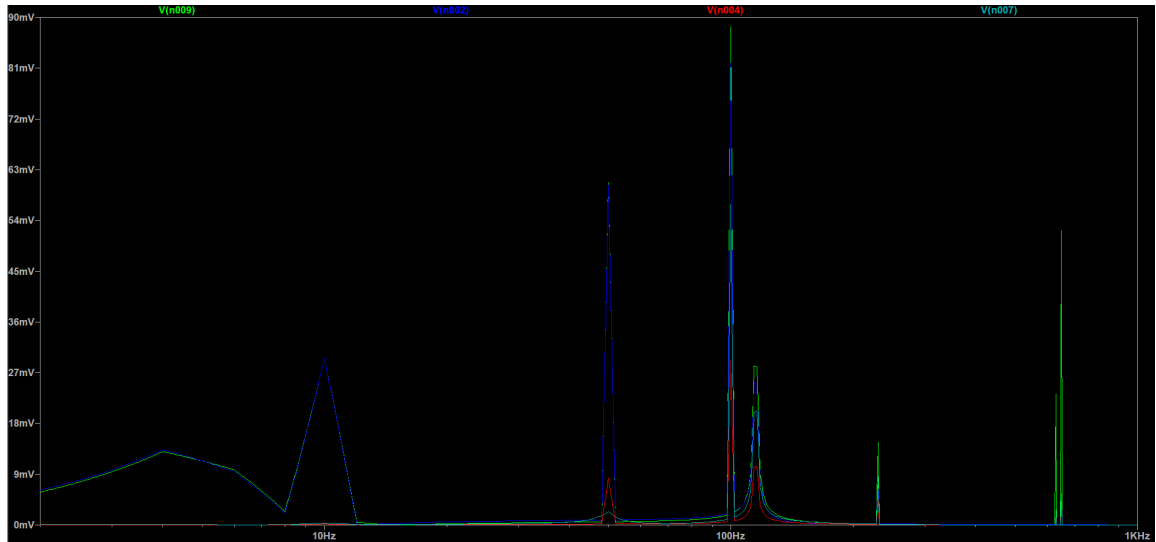


Figura 2.13: FFT de las señales en distintos puntos del circuito. Verde: señal original; azul oscuro: señal tras filtro pasa-baja; rojo: señal tras filtro pasa-alta; azul claro: señal tras filtro rechaza banda.

Finalmente, tras realizar la eliminación de ruido de la señal, se procede a la etapa de rectificación, lo que implica una señal únicamente positiva que sí podrá entrar al Arduino. El ejemplo de esta simulación puede apreciarse en la Figura 2.14, en ella se observa la señal que sería la entrada (roja), que no sólo se rectifica sino que se amplifica mediante el uso de dos resistencias, en este caso, al doble.

Cabe añadir, que el diodo utilizado para la rectificación añadirá un ligero ruido a la señal, pero no será necesario realizar ningún filtrado posterior.

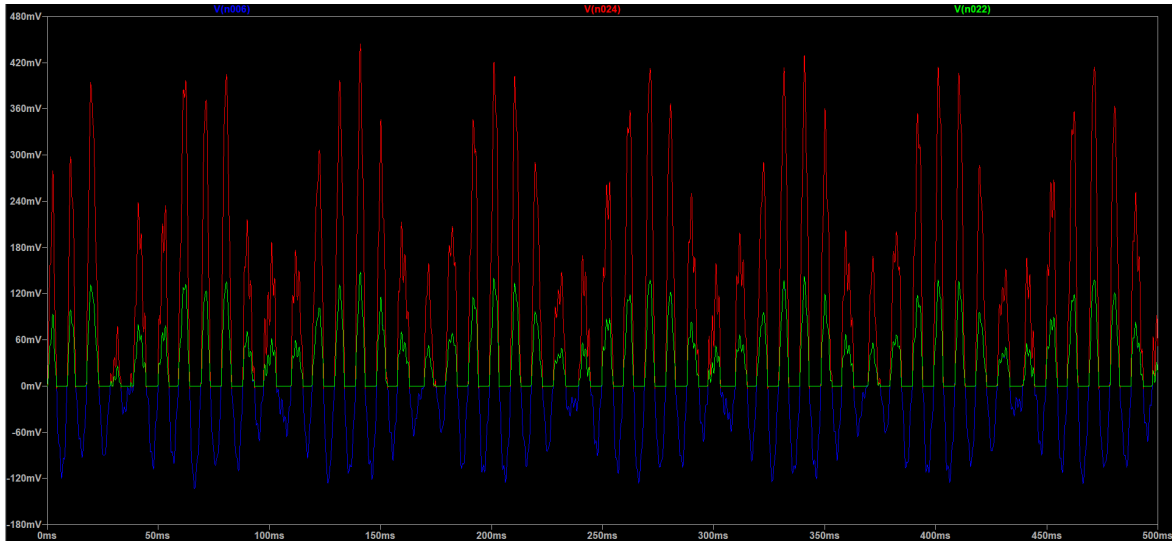


Figura 2.14: Señal de entrada al Arduino. Azul: señal filtrada; verde: señal filtrada y rectificada; rojo: señal filtrada, rectificada y amplificada al doble.

2.7. Circuito impreso

Como se ha comentado anteriormente, para reducir el ruido introducido por los cables y el Protoboard, se realiza el diseño del circuito impreso a través de Eagle, un software de diseño de circuitos que permite la realización del mismo en PCB.

El diseño del circuito se ha realizado en dos capas para conseguir un tamaño más reducido, en concreto el uso de forma gratuita implica una limitación de la placa a un tamaño de 100×80 mm. Finalmente la placa realizada que puede contemplarse en la Figura 2.15, contiene desde la preamplificación hasta la entrada al Arduino incluyendo la alimentación ha tenido una dimensión de 80×72 mm, con una anchura de pista de 1.27 mm. Para facilitar la visualización del circuito se incluye una diferenciación de las capas en la Figura 2.16.

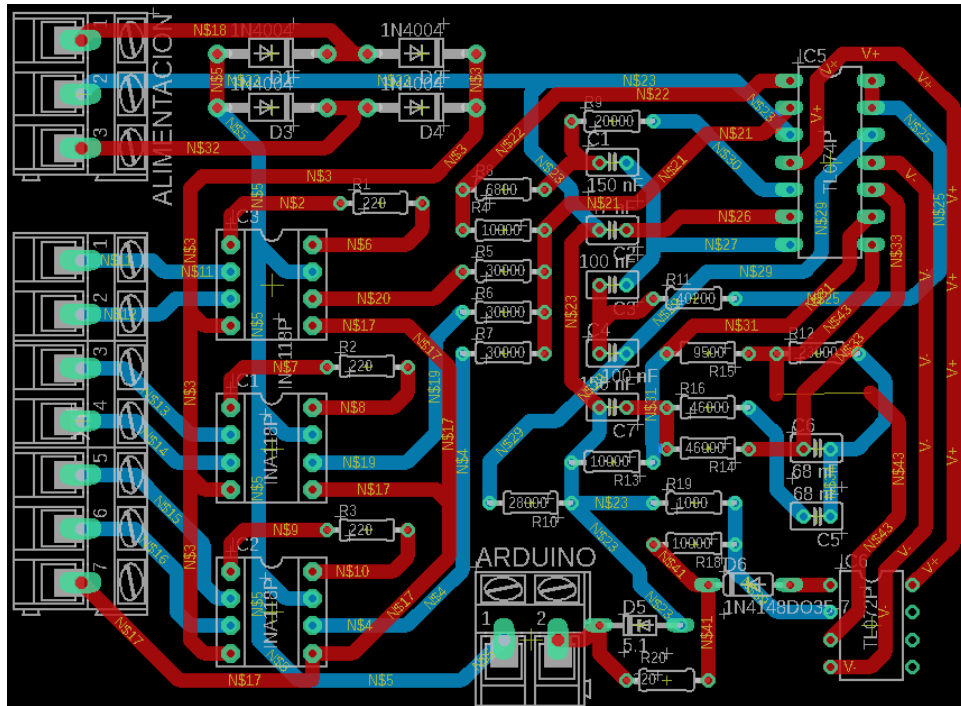
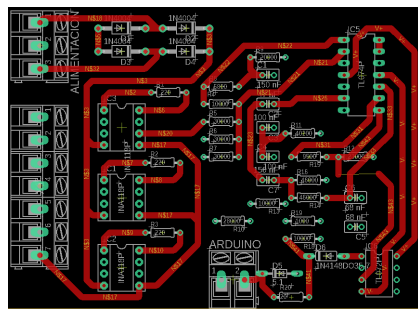
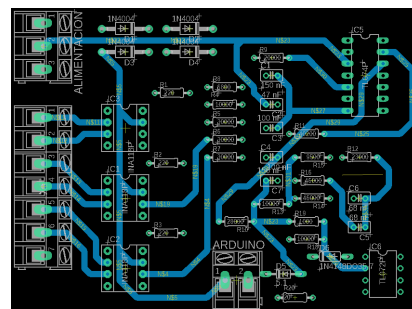


Figura 2.15: Circuito impreso.



(a)



(b)

Figura 2.16: a)Capa superior del circuito impreso b) capa inferior del circuito impreso.

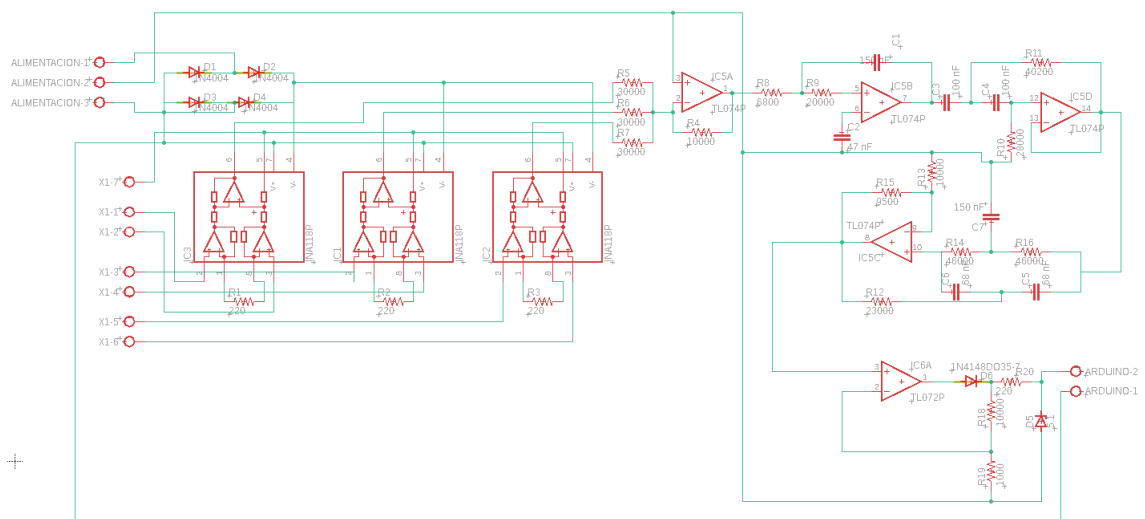


Figura 2.17: Esquema del circuito de adquisición de señales.

Capítulo 3

Trabajo experimental y resultados

3.1. Montaje del circuito

Para poder comprobar todo el trabajo realizado, es necesario, en primer lugar, realizar el montaje del circuito. Para ello, se utiliza una protoboard y los siguientes componentes:

- resistencias: $220\ \Omega$ ($\times 4$), $1\ \text{k}\Omega$ ($\times 2$), $4.7\ \text{k}\Omega$ ($\times 3$), $6\text{k}8\Omega$, $10\ \text{k}\Omega$ ($\times 2$), $20\ \text{k}\Omega$, $22\ \text{k}\Omega$ ($\times 2$), $27\ \text{k}\Omega$, $33\ \text{k}\Omega$ ($\times 3$), $39\text{k}\Omega$.
- condensadores: $47\ \text{nF}$, $68\ \text{nF}$ ($\times 2$), $100\ \text{nF}$ ($\times 2$) y $150\ \text{nF}$ ($\times 2$).
- diodos: Zéner de 5.1V y un diodo.
- amplificadores operacionales: INA 128P ($\times 3$) y TL084 ($\times 2$).
- Alimentación: pila 9V ($\times 2$).

Tras las primeras visualizaciones de la señal, se decidió añadir una etapa extra de amplificación, mostrada en la Figura 3.1 entre filtro de $50\ \text{Hz}$ y el rectificador. Aunque esta no sea necesaria, mejora la visualización, y facilita la clasificación de los datos.

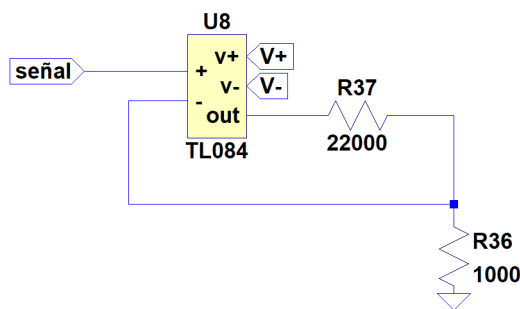


Figura 3.1: Etapa de amplificación.

3.2. Selección de los movimientos y anatomía del antebrazo.

Una de las diferencias anatómicas más importantes del ser humano con el resto de los animales es la existencia del pulgar y el movimiento de oposición que este permite. Este dedo facilita la sujeción de objetos y la pérdida funcional completa o amputación de este implicaría una invalidez del 22 %, frente al 60 % que se correspondería para el caso de la mano completa. Es por ello, por lo que en la selección de movimientos prevalecerán los realizados con el pulgar. Se busca la posibilidad de coger objetos que necesiten el uso de la mano entera como por ejemplo una botella, pero también de agarrar objetos que requieran de más delicadeza, utilizando únicamente el dedo pulgar junto con el índice, como podría realizarse para sostener el tape de un bolígrafo.

Debido a que este trabajo se realiza mediante sensores superficiales, se debe tener en cuenta que para usarse de manera óptima se colocarán en los músculos situados en las capas más externas de la musculatura del antebrazo, siempre que sea posible, reduciendo así al mínimo la interferencia con las señales de otros músculos.

En función de la posición, existen tres grupos musculares en el antebrazo, el anterior o ventral, el posterior o dorsal y el posterior externo [3].

- región anterior: compuesta por cuatro planos, primer plano (pronador redondo, flexor radial del carpo, palmar largo y flexor cubital del carpo); segundo plano (flexor superficial de los dedos); tercer plano (flexor profundo de los dedos, flexor largo del pulgar); cuarto plano (pronador cuadrado).
- región posterior: formado por dos capas, superficial (extensor común de los dedos, extensor propio del meñique y extensor cubital del carpo); profunda (extensor corto del pulgar, abductor largo del pulgar, extensor largo del pulgar y extensor propio del índice).
- región posterior externa: compuesta únicamente por cuatro músculos (supinador largo, extensor radial largo del carpo, extensor radial corto del carpo y supinador corto).

Debido a que en la salida del circuito el ruido queda filtrado por completo, se decide colocar todos los electrodos en músculos flexores, de modo que cuando se abra la mano (músculos extensores), la señal en la salida será nula.



Figura 3.2: Colocación de los electrodos.

3.3. Microprocesador - Arduino Uno.

En este trabajo se utiliza el Arduino Uno para la parte de control. Es una placa electrónica de bajo coste y sencilla programación que utiliza el microcontrolador ATmega328, cuenta con 14 pines digitales, pudiéndose usar 6 para generar señales PWM y 6 pines analógicos; por otro lado, tiene una memoria de 32kb.

Las entradas analógicas permiten una lectura cada $112\mu s$, aunque se puede aumentar la frecuencia de muestreo utilizando los pines digitales. Debido a las limitaciones de memoria, para este trabajo es necesario que los datos sean procesados conforme se reciben. Esto es importante para el periodo de muestreo de la señal, fijándose en 500 Hz.

El teorema de Nyquist, postula que la frecuencia de muestreo deberá ser como mínimo el doble de la frecuencia de la señal, lo cual implicaría una frecuencia de 1kHz, pero dado que, la mayor potencia de la señal se encuentra entre los 20 y 200 Hz, la frecuencia fijada será suficiente, cumpliendo dicho teorema para este rango.

3.4. Creación de una base de datos

Para poder establecer el control de la mano es necesario crear una base de datos. Esta permitirá determinar los parámetros de reconocimiento de los movimientos. En

primer lugar, se crea una pequeña muestra para observar las señales (25 señales de 1s de duración y periodo de muestreo de 500Hz) y una segunda muestra más grande en la que sólo se recogen los parámetros de estudio decididos. Se estudia en ambas la realización de cuatro movimientos, agarre cilíndrico, agarre de precisión con tres dedos, señal de ok y mano cerrada salvo el índice.

En cuanto a las características de la señal, se recomienda [1] y [4], el estudio del valor medio absoluto, longitud de forma de onda, cruces por cero, cambios de pendiente y varianza, en cuanto a factores temporales; y la transformada de Fourier o la de Wavelet para obtener características espectrales.

Las características estudiadas tratan sólo su dominio temporal, ya que la realización de la transformadas para trabajar con su dominio espectral resultan costosas en términos de memoria y tiempo para el programa. Finalmente se decide calcular la media, valor máximo, longitud de onda, número de ceros y cruces por cero.

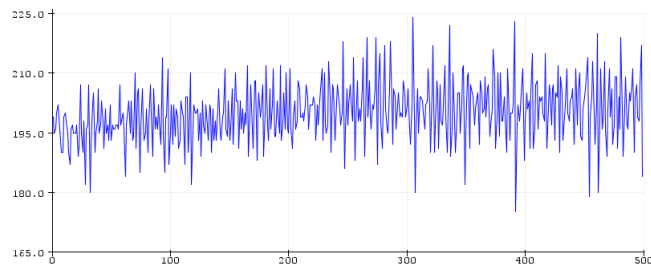


Figura 3.3: Señal de los dedos meñique y anular en el movimiento de agarre cilíndrico.

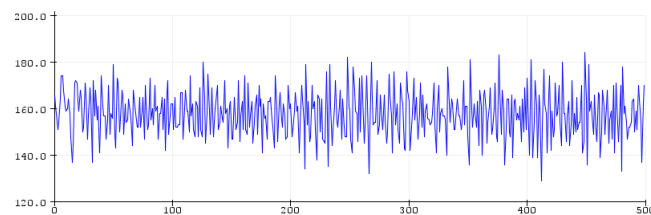


Figura 3.4: Señal de los dedos corazón e índice en el movimiento de agarre cilíndrico.

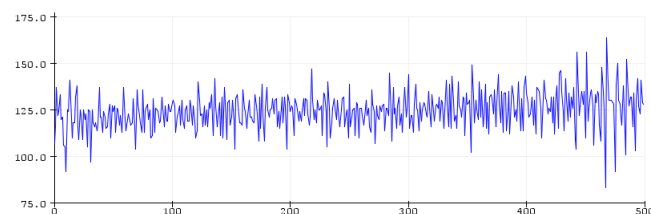


Figura 3.5: Señal del dedo gordo en el movimiento de agarre cilíndrico.

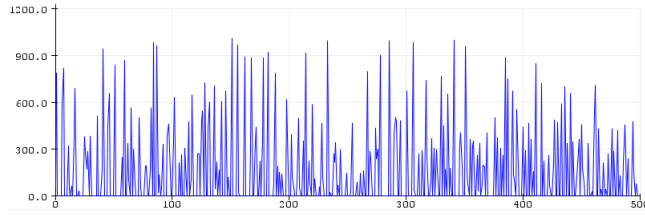


Figura 3.6: Señal obtenida del movimiento de agarre cilíndrico.

3.5. Creación del método de clasificación

Se pretende crear un método de clasificación que funcione de forma independiente, es decir, sin necesidad de estar conectado al ordenador. Softwares como Matlab o Python permiten crear clasificadores de alto rendimiento utilizando técnicas de Machine Learning, pero para implementarlos sería necesaria una comunicación serial continua, haciendo el sistema dependiente de la computadora. Por otro lado, el dispositivo Arduino Nano 33, posee la librería TensorFlow Lite, que permite la creación de pequeñas redes neuronales, pero no es compatible con Arduino Uno. Finalmente, es posible implementar de forma manual redes neuronales muy simples, altamente limitadas por la memoria.

Tras un estudio inicial de las posibilidades, se decide que la mejor opción es un árbol de clasificación, por su sencilla implementación y bajo uso de memoria. Para crearlo, se parte la muestra de forma aleatoria, usando un 70 % de los datos para la creación del modelo y un 30 % para la validación del mismo, esto permite eliminar problemas como que el modelo se adapte en exceso a los datos de entrenamiento, produciendo grandes errores con datos reales (overfitting). [5]

En primer lugar, se estudian el conjunto total de las muestras, eliminando las características de las señales que han tenido un fallo en su grabación, obteniendo finalmente una base de datos con 29 señales de cada movimiento. Por lo que se utilizan 82 para la creación del modelo y 34 para su comprobación. El modelo resultante puede apreciarse en la Figura 3.7, mientras que el código de implementación se recoge en el anexo ???. Los resultados obtenidos fueron de un 100 % con los datos de entrenamiento y de un 85,3 % con los datos de validación.

Los valores que permiten la clasificación de las señales se han obtenido experimentalmente calculando diversos parámetros como máximos, mínimos, medias, varianzas o percentiles. Estos valores dependen completamente de la persona que realice la base de datos y es importante realizar una correcta colocación de los

electrodos, ya que podrían provocar cambios en los parámetros de las señales.

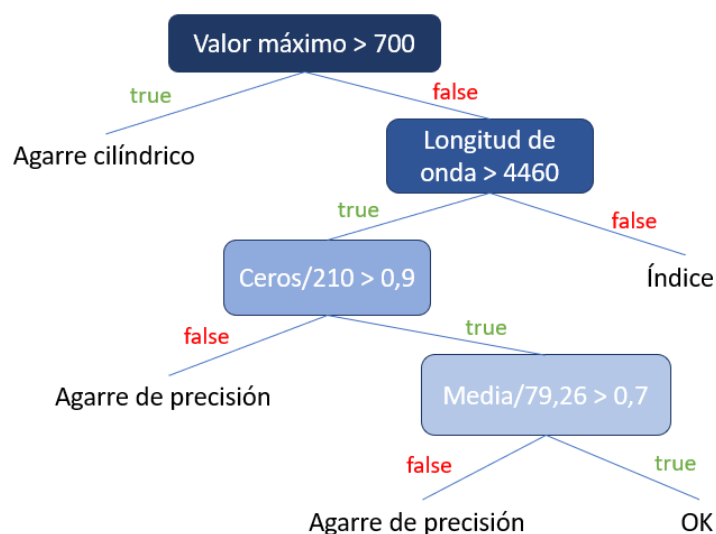


Figura 3.7: Esquema de la clasificación de señales.

3.6. Resultados experimentales

Durante las primeras pruebas tras el montaje inicial, se rompió uno de los amplificadores de instrumentación, por lo que finalmente, debido a los largos plazos de entrega, la base de datos y el clasificador se crearon usando únicamente dos pares de electrodos (uno correspondiente al movimiento del pulgar y otro correspondiente al resto de las falanges), junto con el de referencia. Además fue necesario modificar la etapa sumadora del circuito, cambiando las tres resistencias de $33\text{ k}\Omega$ por dos de $20\text{ k}\Omega$. El montaje de este circuito aparece en la Figura 3.8. Por tanto, todos los resultados obtenidos son para este sistema de 5 electrodos (2 entradas y referencia) y un canal de salida.

Como se ha comentado anteriormente los resultados dependen de la colocación de los electrodos, por lo que los resultados experimentales variaron de un día a otro. El día primer día se consiguió un porcentaje de acierto del 70% teniendo su mayoría de fallos al identificar el movimiento de ok. Se probó dos días consecutivos más en los que el porcentaje descendió permitiendo únicamente la clasificación de tres de los movimientos: agarre cilíndrico (o de fuerza), agarre de precisión e índice. Al eliminar el movimiento ok de las pruebas, se consiguió un porcentaje en ambos días entorno al 85%. Los únicos fallos producidos, se debían a una presión baja en el agarre de fuerza, confundiéndolo con el de precisión.

El código de clasificación de señales se recoge en el anexo ??, en el que a continuación se ha creado una plantilla que permite realizar el control de unos servos en función de los movimientos obtenidos de la clasificación.

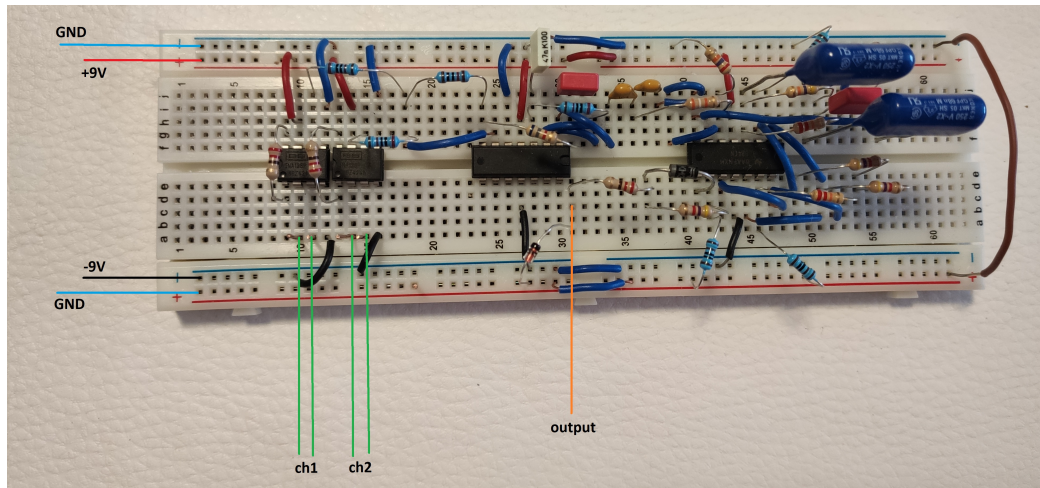


Figura 3.8: Montaje del circuito.

La posición de los electros para el caso final ha sido la que aparece en la Figura 1.3, además de un quinto electrodo en el codo.

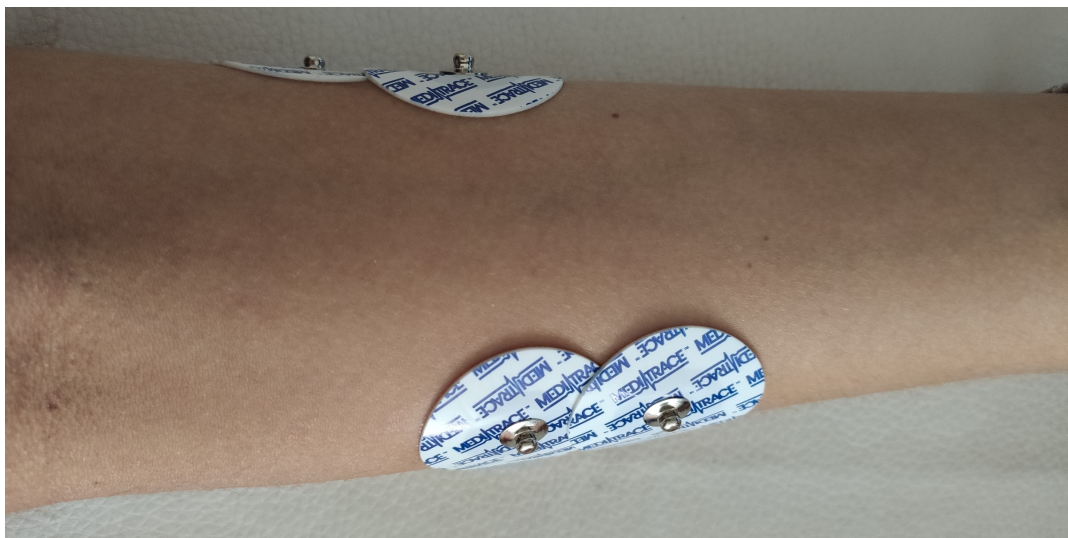


Figura 3.9: Posicionamiento para dos pares de electrodos.

Capítulo 4

Mejora del sistema a través de Python

Dado que los resultados obtenidos con la clasificación implementada en Arduino se consiguen gracias a la repetición y no dependen de su frecuencia, se plantea una alternativa de un programa realizado en Python que permitiría el análisis de las características espectrales de la señal.

Tras el estudio de varias opciones para implementar una mejora, se decide usar esta dado que Python es un lenguaje de programación gratuito, sencillo y con muchas aplicaciones entre las que se encuentran el análisis de datos y clasificación. Este programa podría ser implementado en una Raspberry, sin depender de un ordenador como se ha trabajado en el capítulo anterior.

4.1. Procesamiento de la señal - Transformada de Wavelet

En el apartado 3.4 se habla de las transformadas de Fourier y Wavelet como métodos para el procesamiento de señales, siendo más conocida la transformada de Fourier. Esta transformada es una herramienta muy potente para el análisis de datos pero no es capaz de representar cambios abruptos de manera eficiente. Podría resumirse como una suma de señales que no están localizadas en tiempo ni espacio y se extienden infinitamente.

Por otro lado, la transformada Wavelet si que está localizada en tiempo y espacio y tiene una duración finita. Esto permite obtener información temporal y frecuencial. Esta transformada, posee diferentes formas y tamaños, gracias a las diferentes familias de wavelets. Dentro de estas, se encuentran dos tipos, la transformada continua y la transformada discreta, que se diferencian por cómo discretizan la escala y los

parámetros de traslación.

- Transformada Wavelet continua, CWT: permite un análisis más fino de la escala, ya que tiene una flexibilidad de análisis superior. Los coeficientes, que dependen de la escala, tiempo y frecuencia. El número de coeficientes será igual al número de puntos de la señal por el número de wavelets escaladas. Se usa para altas frecuencias y filtrado de componentes localizados en el tiempo.
- Transformada Wavelet discreta, DWT: En este caso, el escalado de las señales se realiza únicamente con números enteros a diferencia de la CWT, lo cual permite la eliminación de redundancias en los coeficientes. En este caso, el número de coeficientes es igual al número de puntos de la señal analizada, lo que implica un requerimiento inferior de memoria. La DWT, se implementa como una cascada de filtros pasa-baja y pasa-alta, como puede apreciarse en la Figura 4.1. Se utiliza para la eliminación de ruido y compresión de señales.

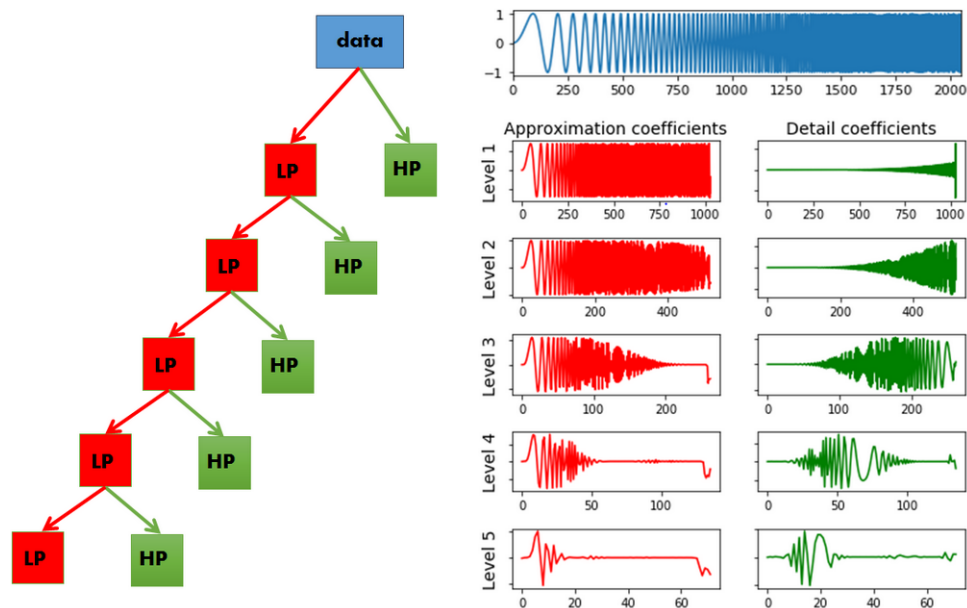


Figura 4.1: Representación de la aplicación de la transformada wavelet discreta, DWT.

En el estudio del artículo *Real-Time Wavelet Analysis of a Vibration Signal Based on Arduino-UNO and LabVIEW* [6], se llega a la conclusión de que se puede realizar una buena eliminación del ruido de la señal en tiempo real a través de la transformada discreta de Wavelet, lo cual se aplicará también en este trabajo, dadas sus aplicaciones y ventajas frente a la transformada continua.

4.2. Procesado de las señales

4.2.1. Pre-procesamiento de las señales

Debido a la disparidad entre los porcentajes de clasificación obtenidos con Arduino, que indicaban un porcentaje del 85.3% en la identificación de 4 movimientos mientras que en las pruebas físicas se tuvo que descartar un movimiento, se decide plantear este apartado con una base de datos externa, cuyas características se recogen en el anexo ??.

La base de datos utilizada, se divide en dos, la primera consta de la repetición de 6 movimientos (30 veces) durante 6s con una frecuencia de muestreo de 500Hz (al igual que en la fase experimental del proyecto como se indica en el capítulo 3) por varias personas; y la segunda, es la repetición de estos mismos movimientos por una sola persona durante 5s y 3 días consecutivos.

Con el objetivo de poder realizar una comparación más fiel, se decide realizar un preprocesado de los datos, para que se asemejen a los trabajados. Se decide utilizar únicamente cuatro de los seis movimientos, transformar los datos para que estos comiencen cuando comienza el movimiento, limitarlos a un tiempo de 0.5s y sumar las entradas, generando un único canal con valores positivos entre 0 y 5V. Un ejemplo de modificación de la señal antes de limitarla a 0.5s, puede observarse en la 4.3, frente a las señales obtenidas por cada canal representadas en la Figura 4.2

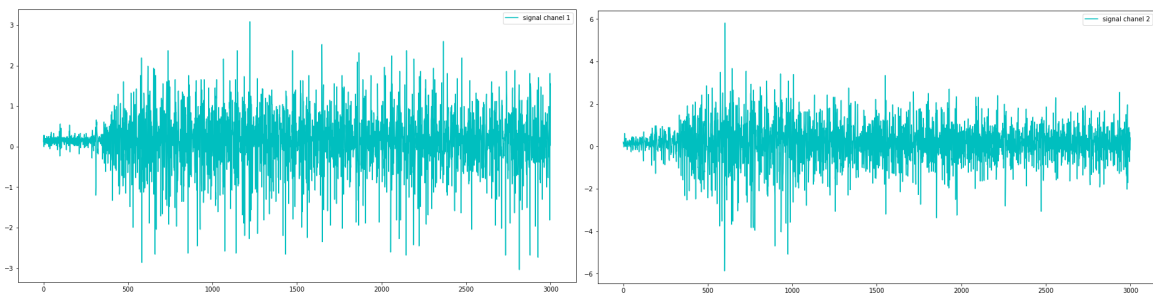


Figura 4.2: Señales de los canales 1 (izq.) y 2 (drch.) para el movimiento de agarre cilíndrico de la base de datos, de la mujer 2, muestra n°5.

4.2.2. Eliminación del ruido a través de la DWT

Una vez se modifican todas las muestras de la base de datos se procede a la eliminación del ruido de la señal. Para ello, se le aplicará la transformada discreta de Wavelet. Como se ha mencionado anteriormente, hay numerosas familias de wavelets, entre ellas, las más recomendadas para el análisis de señales EMG son las Daubechies,

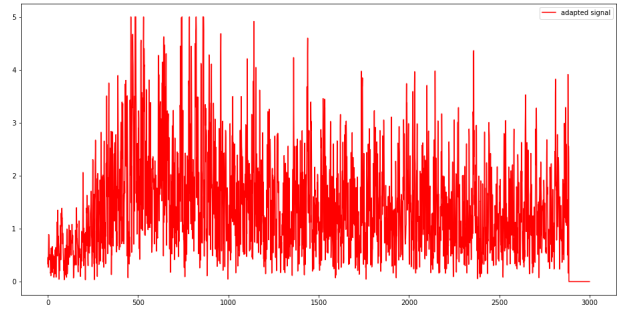


Figura 4.3: Señal adaptada para el movimiento de agarre cilíndrico.

Symlets, Biorthogonal y Reverse biorthogonal [7], pudiendo apreciarse estas en la Figura 4.4.

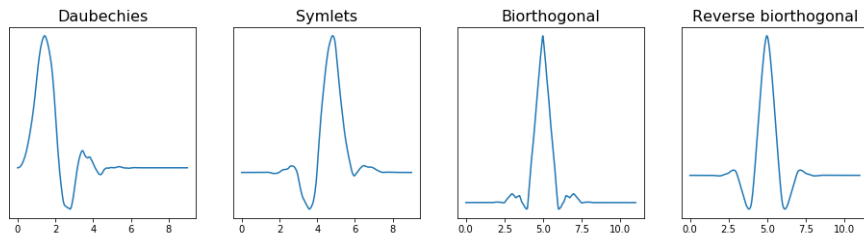


Figura 4.4: db5, sym5, bior5.5, rbio5.5

Este filtrado, se realiza implementando la transformada discreta de wavelet por medio de Python a todas las señales adaptadas. En cuanto a la wavelet aplicada, esta se elige realizando varias pruebas de clasificación en función de la familia usada. Un ejemplo visual de este filtrado se representa en la Figura 4.5, en la que a la señal representada en la Figura 4.3, representada ahora en azul, se le ha aplicado la transformada discreta, en concreto Daubechies 14, obteniéndose la señal representada en negro.

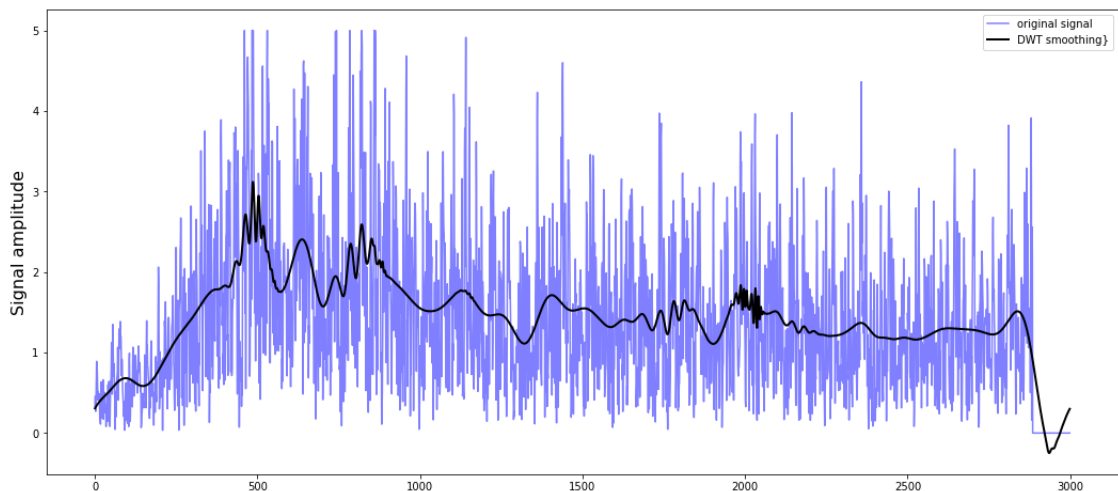


Figura 4.5: Señal original (azul) y señal filtrada con db14(negro).

Para crear y probar el clasificador se usan las señales de la primera base de datos, en concreto las señales mostradas en las figuras superiores son de los correspondientes a los ensayos de la “mujer 2”.

Para el caso de db14, se obtiene un porcentaje de acierto del 94.4% mientras que para el uso de bior5.5 y db5 se obtuvo un porcentaje del 88.8%, en cuanto a sym5 resultó un porcentaje del 94% y para rbior2.2 del 61.1%.

Es importante mencionar, que los porcentajes de fallo se deben a la asociación de la señal a un movimiento incorrecto, es decir, siempre asocian las señal a uno de los posibles movimientos.

4.2.3. Clasificación de las señales

Para clasificar las señales se ha decidido utilizar únicamente los coeficientes obtenidos por medio de la transformada. Para ello, se ha decidido además aplicar un Gradient Boost Classifier. Este tipo de clasificador, utiliza el Machine Learning para realizar la clasificación; se crea añadiendo predictores a un ensamblador, de forma que cada uno corrige a su predecesor. Este método utiliza los errores residuales del predictor anterior para crear la siguiente iteración. [5].

El Machine Learning es una ciencia de programación que permite a la computadora aprender de los datos que se le introducen. El tipo de clasificador utilizado, aplica aprendizaje supervisado, esto quiere decir que para crear el clasificador, se introduce una base de datos (en este caso señales), que contienen una identificación o etiqueta (para este trabajo, el tipo de movimiento).

```
female_250 = pd.ExcelFile(dir_250)
data = pd.read_excel(female_250)
X_250 = np.array(data.drop(['mov'],1))
y_250 = np.array(data['mov'])
validation_size = 0.30
seed = 30
X_train_250, X_validation_250, Y_train_250, Y_validation_250 = model_selection.train_test_split(X_250, y_250, test_size=validation_size, random_state=seed)
cls_250 = GradientBoostingClassifier(n_estimators=1000)
cls_250.fit(X_train_250, Y_train_250)
train_score_250 = cls_250.score(X_train_250, Y_train_250)
test_score_250 = cls_250.score(X_validation_250, Y_validation_250)
print("Train Score for the EMG dataset 250 and GBC: ", format(train_score_250))
print("Test Score for the EMG dataset 250 and GBC: ", format(test_score_250))
```

Figura 4.6: código para crear el clasificador

El código de la Figura 4.6 muestra como se crea un clasificador Gradient Boost, para ello, se usa además la librería scikit.learn de Python. Las dos primeras líneas de esta imagen, se utilizan para leer la base de datos (en este caso guardada en formato

Excel). A continuación se seleccionan los datos relevantes y la etiqueta que permite la identificación de los elementos a clasificar.

El procedimiento para crear el clasificador es similar al seguido en el punto 3.6, el 70 % de los datos se utilizan para crear el modelo y el 30 % restante para comprobar su eficacia.

4.3. Aplicación del clasificador a la base de datos del proyecto

El método de clasificación descrito, se amplía ahora a la base de datos generada en el proyecto y descrita en el apartado 3.4. En este caso, el único pre-procesado necesario es la aplicación de la transformada Wavelet, que puede ser distinta a la aplicada para la otra base de datos. Con el objetivo de obtener el mejor resultado posible de clasificación se crean distintos modelos para las transformadas recomendadas anteriormente.

Los distintos porcentajes validación en la clasificación varían en función de la transformada siendo el mejor resultado el obtenido para el caso de rbio2.2 con un 73.33 %. Para db14, cuyos resultados eran los mejores en las pruebas con la otra base de datos, resultan en un 53 % para esta, por otro lado con sym5 se obtiene un 50 % y con symy2 un 60 %. En la Figura 4.7 puede apreciarse la aplicación de la transformada que ofrece una mayor puntuación para el movimiento de agarre cilíndrico o de fuerza.

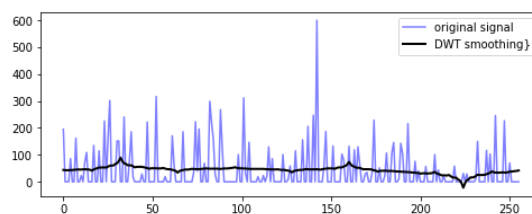


Figura 4.7: Aplicación de la transformada Wavelet rbio2.2 al movimiento de agarre de fuerza.

A pesar de que el resultado pueda parecer peor que el obtenido en el capítulo 3, es importante tener en cuenta que en este caso la clasificación se realiza teniendo en cuenta el dominio espectral de la señal, lo que implica una menor dependencia de la fuerza ejercida en cada movimiento por su identificación. Por otro lado, al igual que ocurría en el capítulo anterior, si se elimina el movimiento ok, el porcentaje de identificación aumenta hasta un 87 %.

Capítulo 5

Conclusiones y líneas futuras.

A pesar de haber tenido problemas con los componentes, se ha conseguido obtener un sistema de adquisición de señales que permite limpiar todo el ruido y que facilita el trabajo posterior de programación.

Los resultados conseguidos para la clasificación de tres movimientos han sido de un 85 %. Aunque parece una buena cifra, es importante mencionar, que sólo se han podido analizar características temporales, por lo que para una correcta clasificación es importante realizar, en la medida de lo posible, la misma fuerza, ya que si por ejemplo, en el movimiento de agarre cilíndrico se coge una botella con suavidad, el sistema probablemente lo identifique como un agarre de precisión.

Como se ha mencionado, se plantean varias alternativas que podrían mejorar los resultados y en concreto, se estudia una de ellas, la creación de un clasificador con Python, aplicando además la transformada de Wavelet. Con este método se prueban de forma teórica dos bases de datos, la creada en el proyecto y una externa. En ambos casos consiguiendo unos muy buenos resultados de clasificación, que permiten analizar el dominio espectral de la señal.

Los resultados obtenidos con el último punto son únicamente teóricos y debería comprobarse su validez de forma experimental. El microprocesador a utilizar para este ensayo, permitiría además mejorar los resultados obtenidos ya sea ampliando el número de canales o la frecuencia de lectura, ya que es un dispositivo ideado para realizar proyectos más complejos que Arduino.

Capítulo 6

Bibliografía

- [1] Judy C. Realpe Ing. Pablo E. Jojoa PhD. Harold A. Romo, Esp. Análisis de señales emg superficiales y su aplicación en control de prótesis de mano. *Revista Avances en Sistemas e Informática, Vol. 136 4 No. 1*, 2007.
- [2] J.G. Núñez Zavala C.X. Guaño Álvarez5 A.C. Zabala Aguiar L.A., Vera Luzuriaga. Análisis e identificación de los movimientos de un brazo humano utilizando un sistema de adquisición y procesamiento de señales electromiográficas. *Espirales revista multidisciplinaria de investigación, ISSN: 2550-6862*, 2018.
- [3] Anne M. Gilroy ; Markus Voll ; Karl Wesker. *Prometheus. Anatomía : manual para el estudiante*. Editorial Médica Panamericana S.A., 2015.
- [4] Villarejo J. J. Costa R. M. Bastos T. Sarmiento, J. F. Protocolo de captura de señales semg para el reconocimiento de gestos de la mano. *Congreso Iberoamericano de Tecnologías En Apoyo a La Discapacidad, IBERDISCAP*, 2013.
- [5] Aurélien Géron. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. "O'Reilly Media, Inc.", ISBN 149203259X, 9781492032595, 2019.
- [6] Robert Bicker Alaa A. Jaber. Real-time wavelet analysis of a vibration signal based on arduino-uno and labview. *International Journal of Materials Science and Engineering Vol. 3, No. 1*, 2015.
- [7] Pornchai Phukpattaranont Angkoon Phinyomark and Chusak Limsakul. *The Usefulness of Wavelet Transform to Reduce Noise in the SEMG Signal*. EMG Methods for Evaluating Muscle and Nerve Function, Mr. Mark Schwartz (Ed.), ISBN: 978-953-307-793-2, 2012.

- [8] Morales B. Diego Andrés Grisales P. Víctor Hugo Pinzón Morales, Rubén D. Caracterización de señales electromiográficas para la discriminación de seis movimientos de la mano. *Scientia Et Technica Año XV, No 42,,* 2009.

Lista de Figuras

1.1. Esquema de los electrodos.	10
1.2. Electrodo de superficie pasivos (a) y activos (b)	10
1.3. Posicionamiento ideal de los electrodos en el músculo.	12
2.1. Diagrama de bloques del sistema para la obtención de señales.	13
2.2. Esquema del amplificador INA 128 fuente Texas Instruments.	15
2.3. Estructura de un inversor sumador de n señales	16
2.4. Gráfica de comparación de filtros Bessel, Butterworth y Chebyshev, a) comparación en magnitud. b) comparación en fase. c) comparación en retraso. d) comparación de respuesta ante entrada escalón.	17
2.5. Topología Multiple Feedback frente a topología Sallen-Key.	18
2.6. Tabla de coeficientes para filtro Butterworth	19
2.7. Filtro paso alto Butterworth con topología Sallen-Key	20
2.8. Filtro rechaza banda	21
2.9. a)Puente de diodos. b) Señal rectificada gracias al puente de diodos . .	22
2.10. Rectificadores de precisión. a)Rectificador de onda completa b) Rectificador de media onda	22
2.11. Protección de los pines de entrada del Arduino	22
2.12. Señales creadas por medio de LTSpice para la comprobación de los circuitos	23
2.13. FFT de las señales en distintos puntos del circuito. Verde: señal original; azul oscuro: señal tras filtro pasa-baja; rojo: señal tras filtro pasa-alta; azul claro: señal tras filtro rechaza banda.	24
2.14. Señal de entrada al Arduino. Azul: señal filtrada; verde: señal filtrada y rectificada; rojo: señal filtrada, rectificada y amplificada al doble.	25
2.15. Circuito impreso.	26
2.16. a)Capa superior del circuito impreso b) capa inferior del circuito impreso.	26
2.17. Esquema del circuito de adquisición de señales.	26
3.1. Etapa de amplificación.	27
3.2. Colocación de los electrodos.	29

3.3.	Señal de los dedos meñique y anular en el movimiento de agarre cilíndrico.	30
3.4.	Señal de los dedos corazón e índice en el movimiento de agarre cilíndrico.	30
3.5.	Señal del dedo gordo en el movimiento de agarre cilíndrico.	30
3.6.	Señal obtenida del movimiento de agarre cilíndrico.	31
3.7.	Esquema de la clasificación de señales.	32
3.8.	Montaje del circuito.	33
3.9.	Posicionamiento para dos pares de electrodos.	33
4.1.	Representación de la aplicación de la transformada wavelet discreta, DWT.	36
4.2.	Señales de los canales 1 (izq.) y 2 (drch.) para el movimiento de agarre cilíndrico de la base de datos, de la mujer 2, muestra n°5.	37
4.3.	Señal adaptada para el movimiento de agarre cilíndrico.	38
4.4.	db5, sym5, bior5.5, rbio5.5	38
4.5.	Señal original (azul) y señal filtrada con db14(negro).	38
4.6.	código para crear el clasificador	39
4.7.	Aplicación de la transformada Wavelet rbio2.2 al movimiento de agarre de fuerza.	40

Lista de Tablas

2.1. Composición de las señales para las simulaciones del circuito en Hz y V. 23

ANEXOS