

UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA

FACULTAD DE INGENIERÍA

MAESTRÍA Y DOCTORADO EN CIENCIAS E INGENIERÍA



**Aprendizaje automático en los movimientos de los dedos con visualización en
Opensim y diseño inalámbrico para la portabilidad en la adquisición de
señales de electromiografía.**

T E S I S

Que presenta para obtener el grado de DOCTOR EN CIENCIAS

AMEZQUITA GARCIA JOSE ALEJANDRO

**DIRECTOR DE TESIS
ROBERTO LOPEZ AVITIA**

**CODIRECTOR DE TESIS
MIGUEL ENRIQUE BRAVO ZANOQUERA**

MEXICALI, B. C.

JUNIO DE 2022

AGRADECIMIENTOS

A las personas que creyeron en este proyecto, principalmente a mi familia, a mis maestros y a todas aquellas partes que contribuyeron a mi formación y crecimiento.

Resumen

En este trabajo se desarrolla la identificación de 15 gestos de la mano y la visualización de los movimientos en un simulador biomecánico, asimismo se diseña la adquisición de señales EMG a través de un dispositivo portátil, todo esto con el fin de preparar el control de una prótesis multifuncional de mano-dedos con señales adquiridas en antebrazo. Por lo que se conjuntaron varios temas de investigación:

- Desarrollo de un dispositivo portátil para la adquisición de señales EMG con el uso de microcontroladores.
- Transmisión inalámbrica multicanal a la máxima tasa de adquisición de señales EMG.
- Procesamiento de las señales EMG con aprendizaje automático para la identificación de 15 gestos de la mano.
- Visualización del movimiento de los dedos a través de un software de simulación biomecánica como Opensim.

La experiencia con todos estos trabajos nos brinda una buena herramienta para el diseño práctico de prótesis de mano o para el control de una interfaz hombre-maquina a través de gestos de la mano. En especial se obtuvieron los resultados siguientes: una clasificación de 78.36% de reconocimiento con el clasificador cuadrático, para un grupo muestral de 8 sujetos; también se demostró que se puede obtener un reconocimiento de 96.16% cuando se usa un clasificador personalizado con solo 4 electrodos, y usando solo el 18% de las mediciones características de los EMG; de igual forma se estableció una transmisión inalámbrica de datos, con seis microcontroladores trabajando simultáneamente, asegurando una captura de mil muestras por segundo para señales EMG con una resolución de 8 bits por muestra.

Abstract

This work develops the identification of 15 hand gestures and the visualization of the movements in a biomechanical simulator, and also designs the acquisition of EMG signals through a portable device, all this in order to prepare the control of a multifunctional hand-finger prosthesis with signals acquired in the forearm. Therefore, several research topics were combined:

- Development of a portable device for EMG signal acquisition using microcontrollers.
- Multichannel wireless transmission at the maximum EMG signal acquisition rate.
- Processing of EMG signals with machine learning for the identification of 15 hand gestures.
- Visualization of finger movement through biomechanical simulation software such as Opensim.

The experience with all these works gives us a good tool for the practical design of hand prosthesis or for the control of a human-machine interface through hand gestures. The experience with all these works gives us a good tool for the practical design of hand prosthesis or for the control of a human-machine interface through hand gestures. In particular, the following results were obtained: a classification of 78.36% recognition with the quadratic classifier, for a sample group of 8 subjects; it was also demonstrated that a recognition of 96.16% can be obtained when using a customized classifier with only 4 electrodes, and using only 18% of the features EMG measurements; in the same way, a wireless data transmission was established, with six microcontrollers working simultaneously, ensuring a capture of one thousand samples per second for EMG signals with a resolution of 8 bits per sample.

ÍNDICE TEMÁTICO

AGRADECIMIENTOS	ii
Resumen	iii
Abstract.....	iv
ÍNDICE TEMÁTICO	v
ÍNDICE DE FIGURAS.....	xi
ÍNDICE DE TABLAS.....	xiv
Introducción.....	1
Planteamiento del problema	2
Hipótesis	2
Objetivos	2
Metas	3
Justificación.....	3
CAPITULO 1 ESTADO DEL ARTE	4
1.1 Introducción	4
1.2 Análisis y decodificación de la señal EMG	6
1.2.1 Músculos y señales mioeléctricas	6
1.2.1.1 Señal mioeléctrica	6
1.2.2 Factores determinantes en la adquisición electromiográfica.....	7
1.3.1.1 Electroodos	8
1.2.2.2 Región anatómica en la adquisición	9
1.2.2.3 Preprocesamiento y Procesamiento.....	11
1.2.2.4 Eliminación de ruido.....	11
1.2.2.5 Normalización.....	11
1.2.3 Minería de datos y aprendizaje automático con señales EMG	12
1.3 Avance en el prototipado de un microcontrolador capaz de implementar un modelo de clasificación.....	13
1.3.1 Simulación de una clasificación en tiempo real con plataformas como Matlab, Labview y Python	14
1.4 Sistemas de transmisión inalámbrica en línea de datos para señales EMG	15
1.4.1 Investigaciones consideradas que buscan una transmisión inalámbrica eficiente de datos EMG	16
1.4.1.1 Bluetooth	16
1.4.1.2 Nrf24l01	16

1.4.1.2 Wifi.....	17
1.5 Simulación en tiempo real con Opensim.....	17
1.5.1 Investigaciones que trabajan con Opensim como instrumento de simulación.	17
1.5.2 Trabajos fuera de línea.....	17
1.5.3 Trabajos en línea.....	18
CAPITULO 2 MARCO TEORICO	20
2.1 Introducción	20
2.2 Señal EMG	21
2.2.1 Músculos de antebrazo	22
2.3 Adquisición electromiografica y su procesamiento con aprendizaje automático	23
2.3.1 Electrodo en el mercado.....	24
2.3.2 Selección de las Regiones anatómicas.....	24
2.3.3 Sistema de adquisición analógico para EMG	24
2.3.3.1 Circuitos Integrados en el mercado.....	25
2.3.3.1.1 AD8232.....	25
2.3.3.1.1.1 Diseños sugeridos	25
2.3.3.1.1.2 Programa para diseños de circuitos eléctricos con el AD8232: Diptrace.....	25
2.3.4 Preprocesamiento.....	26
2.3.4.1 Base de datos	26
2.3.4.2 Ventaneo de datos	31
2.3.4.3 Características seleccionadas	32
2.3.4.4 Normalización.....	35
2.3.4.5 Método de eliminación de valores atípicos por una distribución chi cuadrada	36
2.3.5 Minería de datos y aprendizaje automático.....	36
2.3.5.1 Algoritmos de clasificación considerados.....	36
2.3.5.2 Optimización en la selección de características SFS	37
2.3.5.3 Transformación a nuevos espacios: Método de matrices dispersas	38
2.3.6 Métodos de validación en aprendizaje automático	38
2.3.6.1 Formulación estadísticas	38
2.3.6.2 Validación cruzada	39
2.3.6.3 ANOVA.....	39
2.3.6.4 Método de la prueba de significancia estadística de Duncan	39
2.4 Microcontroladores y simulación en línea	40

2.4.1	Microcontroladores en el mercado	40
2.4.1.1	Microcontroladores en el mercado capaces de realizar aprendizaje automático: TM4C123G	41
2.4.1.1.1	Plataformas de desarrollo Code Composer Studio y Keil.	41
2.4.2	Plataformas para simulación en línea de un proceso de clasificación.	41
2.4.2.1	Matlab	42
2.4.2.2	Labview	42
2.4.2.3	Python	42
2.5	Sistemas comerciales para la adquisición inalámbrica de señales EMG	43
2.5.1	Myoarmband	43
2.5.2	Delsys trigno avanti Sistema trigno lite.....	43
2.6	Protocolos de comunicación inalámbrica.....	43
2.6.1	Bluetooth.....	44
2.6.1.1	Módulos Bluetooth en el mercado	45
2.6.2	Nordic semiconductor	45
2.6.3	Wifi	45
2.6.4	Dispositivos multiprotocolo.....	46
2.7	Opensim	46
2.7.1	Estructura de Opensim fuera de línea	46
2.7.2	Nuevas herramientas: simulación en línea	47
2.7.2.1	Das3	47
2.7.2.1.1	Función Mex Matlab.....	47
2.7.2.2	Opensense	48
2.7.3	Modelos de miembro superior de antebrazo, mano y dedos	48
Capítulo 3 Experimentación.....		49
3.1	Introducción	49
3.2	Protocolo de adquisición EMG propuesto	49
3.2.1.	Factores determinantes que afectan la señal EMG.....	49
3.2.2	Consideraciones extras para una adquisición adecuada:.....	51
3.3	Sistema de adquisición analógico para EMG.....	51
3.4	Electromiografía y su procesamiento con aprendizaje automático.....	53
3.4.1	Pre procesamiento y procesamiento de la señal EMG.....	53
3.4.1.1	Detección de valores atípicos multivariable utilizando la distancia Mahalanobis.	53

3.4.1.2	Método de selección de atributos por SFS	54
3.4.1.3	Normalización.....	56
3.3.1.4	Conversión al espacio de matrices dispersas	56
3.4.2	Modelos de clasificación	56
3.4.2.1	Funciones discriminantes: Análisis discriminante lineal y cuadrático con un enfoque Bayesiano	57
3.4.3	Rutas de preprocesamiento consideradas.....	58
3.4.4	Métodos de validación estadística de los resultados.....	58
3.4.4.1	Validación cruzada	58
3.4.4.2	Formulaciones estadísticas	60
3.4.4.3	ANOVA	60
3.4.4.4	Método de Duncan	60
3.5	Simulación virtual del movimiento Opensim.....	61
3.5.1	Análisis del movimiento del peor al mejor movimiento clasificado y su simulación en Opensim	62
3.6	Microcontroladores y simulación en línea	64
3.6.1	TM4C123G6HPM.....	64
3.6.2	Pruebas del sistema de adquisición	64
3.6.2.1	Transferencia de datos: Evaluando la adquisición y transferencia de los datos del ADC a los registros que controlan los datos de entrada de este periférico.....	65
3.6.3	Transferencia serial de datos desde el microcontrolador hasta el programa de adquisición en el equipo concentrador	67
3.6.3.1	Configuración del sistema inalámbrico: Modulo Bluetooth HC06 modelo ZS-040 versión linvosVI.8.....	68
3.6.3.1.1	Configuración del modulo	68
3.6.3.1.1.1	Arduino UNO	68
3.6.3.1.1.2	PL2303.....	68
3.6.3.2	Transferencia de datos Bluetooth-Bluetooth	70
3.6.3.2.1	Envío de datos a través del puerto serial UART a un módulo Bluetooth, generación de una conexión serial virtual en la computadora.....	72
3.6.3.2.1.1	Terminales virtuales	72
3.6.3.2.1.2	Programación en Python	73
3.6.4	Transferencia de datos de forma simultanea.....	75
3.6.4.1	Evaluación de la cantidad de datos y velocidad máxima de transmisión que puede realizarse vía Bluetooth	75

3.6.4.2 Envío de datos a modulo maestro PC.....	76
3.7 Simulación en línea	79
3.8 Programación en Labview para crear una clasificación en línea con instrumentos virtuales.	80
Capítulo 4 Resultados y Discusión	81
Resultados.....	81
4.1 Introducción	81
4.2 Sistema de adquisición analógico para EMG.....	81
4.3 Electromiografía y su procesamiento con aprendizaje automático	86
4.3.1 Pre procesamiento y procesamiento de la señal EMG.....	86
4.3.1.1 Detección de valores atípicos multivariantes	87
4.3.1.2 Modelos de clasificación	89
4.3.1.3 Método de selección de atributos por SFS	90
4.3.1.4 Normalización.....	91
4.3.1.5 Normalización y SFS.....	91
4.3.2 Métodos de validación	92
4.3.2.1 ANOVA	92
4.3.2.2 Método de la prueba de significancia de Duncan.....	94
4.3.3 Conversión a un nuevo espacio de matrices dispersas.....	94
4.3.3.1 Matriz de confusión	95
4.3.3.2 Formulaciones estadísticas después del cambio de espacio.....	96
4.3.4 Modelo de clasificación considerado	96
4.4 Simulación del movimiento en Opensim	97
4.4.1 Simulación a partir del modelo de clasificación	99
4.5 Microcontroladores y simulación en línea	102
4.5.1 Pruebas del sistema de adquisición	102
4.5.1.1 Transferencia de datos: Evaluando la adquisición y transferencia de los datos del ADC a los registros que controlan los datos de entrada de este periférico.....	102
4.5.2 Transferencia serial de datos desde el microcontrolador hasta el programa de adquisición en el equipo concentrador	102
4.5.2.1.1 Terminales virtuales	103
4.5.2.1.2 Programación en Python	104
4.5.3 Transferencia de datos de forma simultanea.....	105

4.5.3.1 Evaluación de la cantidad de datos y velocidad máxima de transmisión que puede realizarse vía Bluetooth	105
4.5.3.2 Envío de datos a modulo maestro PC.....	108
4.5.3.3 Resultados puntuales de la prueba de adquisición de datos via ADC.....	109
4.5.4 Prototipado de un microcontrolador.....	110
4.5.4.1 Prototipado Componentes (Prototipo de un microcontrolador virgen).....	110
4.6 Simulación en línea: Herramienta para clasificar movimientos en línea a partir de un clasificador implementado en Labview (Herramienta registrada en Indautor).....	112
Discusión	115
Conclusiones	122
Referencias.....	123

ÍNDICE DE FIGURAS

Figura 2.1. Electromiograma de Mano cerrada y sus 8 electrodos.	22
Figura 2.2. a) Principales músculos involucrados en la región de adquisición del arreglo utilizado A: Brachiocardial (BR), B: Pronator teres (PT), C: Flexor carpi radialis (FCR), D: Palmaris longus (PL), E: Flexor carpi ulnaris (FCU), F: Flexor digitorum superficialis (FDS), G: Flexor pollicis longus (FPL), H: Flexor digitorum profundus (FDP).	23
Figura 2.3. a) Principales músculos que intervienen en la región de adquisición del arreglo de electrodos, cada uno de ellos participa en la adquisición de la señal de mayor o menor intensidad dependiendo que tan cerca o alejado se encuentre del electrodo. A: Braquiocardial (BR), B: Pronator teres (PT), C: Flexor carpi radialis (FCR), D: Palmaris longus (PL), E: Flexor carpi ulnaris (FCU), F: Flexor digitorum superficialis (FDS), G: Flexor pollicis longus (FPL), H: Flexor digitorum profundus (FDP). b) arreglo utilizado en la adquisición de la base de datos, donde se puede observar el arreglo anillo utilizado.	27
Figura 2.5 Selección de la ventana sin traslape.....	35
Figura 2.6. Creación de un modelo de clasificación.....	37
Figura 3.1 Interfaz gráfica de la herramienta AD8232 Filter Design.	52
Figura 3.2 Programa utilizado para seleccionar los valores atípicos de la base de datos.	54
Figura 3.3 Seudocódigo de SFS.....	55
Figura 3.4 Algoritmo utilizado para la selección de características hacia adelante.....	55
Figura 3.5 Método utilizado para cambiar los datos analizados a un nuevo espacio.....	56
Figura 3.6 Función utilizada para el modelo de clasificación a partir de Naive Bayes.	57
Figura 3.7 Función utilizada para crear modelos de clasificación LDA y QDA.....	57
Figura 3.8 Secuencia de experimentación creada para evaluar el desempeño de los clasificadores a partir de una variación en el pre procesamiento.	58
Figura 3.9 Metodología para la validación cruzada.	59
Figura 3.10 Modelo modificado de Opensim para reproducir movimientos evaluados.	61
Figura 3.11 Ciclo desde la clasificación hasta la generación del archivo de movimiento para Opensim...	62
Figura 3.12 Creación del rango de avance para cada movimiento evaluado.	63
Figura 3.13 Placa de desarrollo utilizada para la experimentación.....	64
Figura 3.14 Conexiones realizadas para introducir una señal senoidal al microcontrolador y al mismo tiempo leer los datos a través del puerto UART.	67
Figura 3.15 Velocidades de los 3 elementos que participan en una transferencia de datos en línea.	67
Figura 3.16 Convertidor USB a serial TTL. A) módulo utilizado PL2303, B) conexión realizada para configurar el módulo Bluetooth.	68
Figura 3.17 Código utilizado para configurar el módulo Bluetooth en Arduino.	70
Figura 3.18 Transmisión inalámbrica de datos a partir de un microcontrolador con una señal senoidal almacenada en su memoria flash.	70

Figura 3.19 Datos almacenados en la memoria del microcontrolador que conforman un registro de 120 datos que definen un ciclo de una señal senoidal.....	71
Figura 3.20 Programa para enviar datos con el microcontrolador a un puerto serie.....	71
Figura 3.21 Diagrama de flujo, programa para transferencia de datos en el microcontrolador TM4C123G.	72
Figura 3.22 Terminal Virtual PuTTY.....	73
Figura 3.23 Diagrama de flujo para el programa creado en Python.	74
Figura 3.24 Programa en Python para la adquisición de los datos vía serial.	74
Figura 3.25 Conexión de dos microcontroladores con su respectivo modulo Bluetooth para el envío simultaneo de datos a un concentrador.	76
Figura 3.26 En esta imagen se muestran los parámetros y la experimentación que se realizó a cada uno de los elementos de nuestro sistema de adquisición.	77
Figura 3.27 Número máximo de módulos que puede soportar teóricamente un Bluetooth maestro.	79
Figura 4.1 Señal del módulo comercial para censar biopotencial cardiaco.	82
Figura 4.2 Esquemático del filtro para señales musculares utilizados.....	82
Figura 4.3 Diseño del pcb del prototipo 1 para realizar una adquisición analógica de la señal EMG.....	83
Figura 4.4 Placa pcb desarrollada para la adquisición analógica (Prototipo 1).	83
Figura 4.5 Se describen las medidas y diseño propuesto para el Prototipo 2 del sistema de adquisición analógico.	84
Figura 4.6 Prototipo 2 del sistema de adquisición analógico, descripción de los conectores y de la carcasa propuesta para su protección.....	84
Figura 4.7 Señal EMG adquirida de antebrazo ante un movimiento de mano cerrada.....	85
Figura 4.8 Valores mantenidos y eliminados de la base de datos analizada.....	87
Figura 4.9 Gráfica de ANOVA.....	93
Figura 4.10 Gráfica del porcentaje de clasificación seleccionando un menor número de características después de la conversión a matrices dispersas.....	95
Figura 4.11 Reproducción del movimiento RL durante 5 segundos en Opensim A) Movimiento ideal B) movimiento total de la repetición 1 para el movimiento del sujeto 6 que fue el peor clasificado.	102
Figura 4.12 Interfaz gráfica de Putty donde se configura en el equipo concentrador los parámetros de la conexión serial creada a partir del módulo bluetooth concentrador.....	103
Figura 4.13 Parte de la señal adquirida a partir de una lectura de un puerto serial virtual por parte de Putty.	104
Figura 4.14 a) Datos adquiridos con el programa en Python, b) tiempo de procesamiento.	105
Figura 4.15 Datos enviados por el puerto UART con un baudrate de 1382400.	106
Figura 4.16.....	107
Figura 4.17.....	¡Error! Marcador no definido.
Figura 4.18.....	108

Figura 4.19	¡Error! Marcador no definido.
Figura 4.20 Experimentación realizada por medio de Analog Discovery y su analizador lógico.	109
Figura 4.21	111
Figura 4.22	111
Figura 4.23 Panel frontal del clasificador en Labview.	112
Figura 4.24 Pestaña que nos permite ver los parámetros del modelo creado.....	113
Figura 4.25 La tercera pestaña de nuestra aplicación nos permite ver los coeficientes de cada una de las variables que componen a nuestro modelo de clasificación.	114

ÍNDICE DE TABLAS

Tabla 2.1 Base de datos consideradas para el análisis propuesto.	27
Tabla 2.2 Extensores Extensor carpi radialis (ECR). Extensor carpi ulnaris (ECU). Extensor digitorum communis and Extensor digiti minimi proprius (EDC y EDMP). Extensor Pollicis Longus (EPL). Flexores: Flexor carpi radialis (FCR). Flexor carpi ulnaris (FCU).....	28
Tabla 2.3 Músculos y su relación con los movimientos de la base de datos.	30
Tabla 2.4 Formulaciones estadísticas donde TP=True positive, FP=false positive, TN=True Negative y FN=False Negative.	39
Tabla 4.1 Tabla con promedios de valores obtenidos con varios niveles de voltaje de alimentación ante una misma señal de entrada.	86
Tabla 4.2 Resumen de los valores mantenidos y eliminados por sujeto.	88
Tabla 4.3 Resumen de los valores mantenidos y eliminados por sujeto.	88
Tabla 4.4 Resumen de los valores Mantenidos y eliminados en función de los movimientos.	89
Tabla 4.5 Resultados del modelo de clasificación obtenido a partir del punto número 5 de la figura 3.8. Porcentaje de reconocimiento a través de una validación cruzada 10 x 10 dividiendo los datos en entrenamiento (TRN), validación (VAL) y prueba (TST). La columna de sujetos se divide en dos partes para cada clasificador, 1..8: indica que se realizó un promedio entre los datos de los 8 sujetos, S1:8 indica que el total de los datos fue un grupo muestral de los 8 individuos.....	90
Tabla 4.6 Resultados del modelo de clasificación obtenido a partir del punto número 6 de la figura 3.8. Porcentaje de reconocimiento a través de una validación cruzada 10 x 10 dividiendo los datos en entrenamiento (TRN), validación (VAL) y prueba (TST). La columna de sujetos se divide en dos partes para cada clasificador, 1..8: indica que se realizó un promedio entre los datos de los 8 sujetos, S1:8 indica que el total de los datos fue un grupo muestral de los 8 individuos. La columna N.F corresponde a las características obtenidas a partir de la selección hacia adelante.	90
Tabla 4.7 Resultados del modelo de clasificación obtenido a partir del punto número 7 de la figura 3.8. Porcentaje de reconocimiento a través de una validación cruzada 10 x 10 dividiendo los datos en entrenamiento (TRN), validación (VAL) y prueba (TST). La columna de sujetos se divide en dos partes para cada clasificador, 1..8: indica que se realizó un promedio entre los datos de los 8 sujetos, S1:8 indica que el total de los datos fue un grupo muestral de los 8 individuos.....	91
Tabla 4.8 Resultados del modelo de clasificación obtenido a partir del punto número 8 de la figura 3.8. Porcentaje de reconocimiento a través de una validación cruzada 10 x 10 dividiendo los datos en entrenamiento (TRN), validación (VAL) y prueba (TST). La columna de sujetos se divide en dos partes para cada clasificador, 1..8: indica que se realizó un promedio entre los datos de los 8 sujetos, S1:8 indica que el total de los datos fue un grupo muestral de los 8 individuos. La columna N.F corresponde a las características obtenidas a partir de la selección hacia adelante.	91
Tabla 4.9 Porcentaje de reconocimiento para cada uno de los tratamientos para cada sujeto las columnas con numero representan el preprocesamiento declarado en la Figura 3.8.	93
Tabla 4.10 Análisis ANOVA para determinar si las medias de los tratamientos utilizados tenían diferencias significativas.....	93

Tabla 4.11 Tabla de resultados de la prueba de significación de Duncan, la conclusión es que las medias de los resultados obtenidos de los procedimientos A, B y C son iguales, y D (Sin ningún tratamiento) no es igual. Por lo tanto, se puede elegir cualquiera de los tratamientos "Normalización y SFS", "Sólo Normalización" y "Sólo con SFS"	94
Tabla 4.17 Matriz de confusión.	95
Tabla 4.12 Promedio de los parámetros estadísticos para los 8 sujetos Evaluados.	96
Tabla 4.13 Parámetros estadísticos para un grupo muestral de 8 sujetos con 49 características.	96
Tabla 4.14 Orden de los porcentajes de reconocimiento de menor a mayor de los movimientos por sujeto.	99
Tabla 4.15 Porcentajes de reconocimiento de la Tabla 4.14.	100
Tabla 4.16 Porcentaje de reconocimiento promedio de los 8 sujetos.	100
El siguiente paso fue determinar la velocidad de transferencia con varios módulos transmitiendo simultáneamente la Tabla 4.17 muestra los resultados obtenidos. Se consideraron múltiples baudrates, tiempo de retraso utilizado, número de bytes enviados por muestra y número de módulos activos. ¡Error! Marcador no definido.	
Tabla 4.17 Experimentación para determinar la viabilidad de múltiples módulos bluetooth esclavos transmitiendo simultáneamente a un concentrador en un equipo de cómputo. Se realizó un total de 10 pruebas por cada experimento.....	106

Introducción

Los avances de la ingeniería junto con los desarrollos que conlleva en la tecnología, en las ciencias que forman parte de la vida de los seres humanos se vuelve cada vez más compleja, sin embargo, hemos diseñado sistemas para hacer de la mayoría de nuestras actividades más fáciles. La vida como la conocemos se encuentra apoyada con desarrollos tecnológicos en todos los niveles, si hablamos específicamente de la ingeniería, podemos encontrar avances a nivel médico que ayuda a los médicos a realizar sus labores y a las personas enfermas en mejorar su calidad de vida. En este trabajo nos centramos en varias áreas del conocimiento de la ingeniería buscamos desarrollar una ruta que sea viable para desarrollar un sistema de control para una prótesis multifuncional. Apoyándonos del mundo de la instrumentación para diseñar un dispositivo de adquisición de señales EMG y del aprendizaje automático para evaluar las señales musculares y diseñar modelos de clasificación para varios movimientos de mano; de igual forma nos apoyamos en métodos de simulación para verificar y validar nuestros resultados.

Una parte importante de todo trabajo de la ingeniería es su validación estadística ya que es necesario tener resultados validados robustamente para poder concluir en función de ellos. Nos apoyamos en áreas de la biomecánica, electrónica, en aprendizaje automático y en lenguajes de programación como C, Matlab, Labview y Python.

Uno de los objetivos clave de este trabajo es generar una ruta viable para desarrollar un sistema de adquisición de señales EMG inalámbrico multicanal para la aplicación de un clasificador de señales EMG en tiempo real. Se utilizó diseño electrónico, métodos de aprendizaje automático (Naive Bayes, Análisis discriminante lineal y cuadrático), métodos de validación estadística (validación cruzada, ANOVA y Método de Duncan), de igual forma se utilizó un software de simulación biomecánica Opensim para visualizar lo movimientos generados a partir del clasificador.

Planteamiento del problema

En la última década se han analizado los sistemas de control protésico se reporta que actualmente en el mercado 60% poseen 1 a 4 grados de libertad (GDL), 30% tienen de 5 a 10 GDL y solo el 10% tiene más de 10 GDL. Los sistemas de control protésicos actuales no han permitido el desarrollo de prótesis multifuncionales.

La investigación en estos sistemas se encuentra en progreso abordando problemas definidos, sin embargo, las actuales investigaciones y circuitos integrados nos permiten desarrollar sistemas portables con una buena clasificación de señales, con un porcentaje de reconocimiento aceptable, de esta forma nos es posible generar nuestra propia integración de las ramas y proponer una ruta viable para cumplir nuestro objetivo.

Abordar este problema nos dará las bases para definir los problemas específicos que hay que considerar el momento de crear un sistema integrado. Y poder contar con un sistema de control por medio de aprendizaje automático para una prótesis multifuncional.

Hipótesis

Los métodos de clasificación creados a partir de aprendizaje automático con señales electromiográficas tendrán un alto porcentaje de reconocimiento en la clasificación de los movimientos de la mano que serán reproducidos en un ambiente de simulación biomecánica como Opensim.

Objetivos

Objetivo general

Este proyecto tiene como objetivo establecer los puntos clave y sus desafíos en el desarrollo de un sistema de control protésico de mano a través de una clasificación con aprendizaje automático.

Objetivos específicos

Investigar continuamente la información contextualizada sobre los protocolos de adquisición EMG en antebrazo para su uso en clasificación de movimientos de mano con revisiones de la literatura reciente a lo largo del periodo del proyecto.

Desarrollar y evaluar modelos de clasificación simple para su implementación en un control de un sistema virtual de movimiento biomecánico.

Diseñar un prototipo de adquisición analógica de señales EMG evaluando el AD8232.

Evaluar microcontroladores que encajen con una visión a futuro en su integración a prototipos de adquisición de electromiografía para realizar una adquisición en tiempo real de un sistema inalámbrico.

Evaluar medios de transmisión inalámbrica para transferir datos de múltiples dispositivos a un concentrador.

Utilizar el software Opensim para validar el movimiento generado a partir de los clasificadores.

Metas

Evaluación y desarrollo de modelos de clasificación, definiendo cual es conveniente utilizar en función de simplicidad y eficiencia, considerando los métodos clásicos como Naive Bayes, análisis discriminante lineal y cuadrático.

Propuesta de una tarjeta de adquisición analógica para señales EMG, en las frecuencias establecidas en la evaluación de protocolos de adquisición.

Evaluación y selección de un microcontrolador apegándonos a los objetivos establecidos, una buena velocidad de procesamiento, memoria y protocolos de comunicación.

Selección de un medio de transmisión inalámbrica para generar un arreglo multicanal de señales EMG.

Desarrollo de un método para traducir los movimientos clasificados a partir del modelo de clasificación a su representación en un modelo biomecánico de la mano-dedos y al mismo tiempo contar con la oportunidad de realizar análisis musculares de los movimientos.

Justificación

El estudio y cumplimiento de los objetivos establecidos nos permitirá crear un sistema de control para una prótesis multifuncional con definición en los movimientos de todos los de dos de la mano, el uso de Opensim como medio de validación del movimiento nos permitirá evaluar el porcentaje del movimiento realizado y su aplicabilidad en una prótesis multifuncional.

CAPITULO 1 ESTADO DEL ARTE

1.1 Introducción

En este trabajo buscamos concentrar todos aquellos avances que son de utilidad para el desarrollo de un sistema de adquisición de electromiografía (EMG) en línea que sea portátil. Que sea económico y poderoso, que ayude a enfrentar los nuevos retos que suponen el control de nuevos dispositivos, pudiéndose utilizar en prótesis multifuncionales, en control de aplicaciones de forma inalámbrica con un número suficiente de DOF que requieren las aplicaciones futuras.

Las investigaciones en control protésico están dirigiéndose a tratar por separado puntos de interés específicos, en la mayoría de los casos el desacoplamiento de la señal eléctrica muscular superficial, por lo que la decodificación de la EMG superficial (EMGs) es un área importante en este campo. Dentro de las investigaciones con más relevancia podemos encontrar las que se especializan en buscar lugares idóneos para la colocación de los electrodos (un tema que aún está en discusión), con el objetivo de encontrar la región muscular óptima para la adquisición de las señales, otras se enfocan en la importancia del tipo, tamaño, forma y material del electrodo, siendo este el medio de adquisición de la señal, que posteriormente se analizará. Estos puntos son importantes ya que una señal más definida le da características que son útiles al momento de intentar separarlas. Otra área de interés es la que se enfoca a desarrollar o implementar algoritmos para el procesamiento de las señales con el objetivo de su decodificación. En la última década se ha establecido una corriente donde se busca instalar modelos complejos en procesadores, microprocesadores o en algunos casos microcontroladores para su uso en tiempo real. Con el avance de estas tecnologías esto está siendo posible.

Es un hecho que los dispositivos electrónicos protésicos cada vez poseen más GDL, sin embargo, se reporta que actualmente en el mercado 60% poseen 1 a 4 GDL, 30% tienen de 5 a 10 GDL y solo el 10% tiene más de 10 GDL. En [1] mencionan que los sistemas de control en aplicaciones y dispositivos cada vez son más complejos por lo tanto es una necesidad crear sistemas de control más simples con un mayor número de funcionalidades.

De las investigaciones más trabajadas en la decodificación de la señal EMGs encontramos el uso de Deep learning, la reducción de la dimensionalidad a través de métodos específicos o el uso de características especiales. También encontramos estudios donde utilizan transformaciones a nuevos espacios para extraer la información relevante. Aún con tan diversos puntos de investigación y desarrollo se puede observar que el control de sistemas protésicos de miembro superior requiere de mayor investigación para ser de utilidad práctica.

Aunque tenemos grandes avances aislados en investigaciones y desarrollo de elementos electrónicos aún no es suficiente para generar una prótesis multifuncional que trabaje eficientemente con señales EMG, entre otras cosas por la naturaleza de la señal. La señal EMG podría confundirse o tratarse como una señal estocástica, sin embargo, sabemos que la naturaleza de la señal tiene un orden y un significado que puede descifrarse y utilizarse para la reproducción de movimientos en dispositivos o aplicaciones.

Dentro de esta multifuncionalidad compleja, buscamos definir un método que nos permita incluirnos en el mundo de la portabilidad, creando modelos pequeños, poderosos y fáciles de introducir en microcontroladores. Se busca definir una vía que permita clasificar movimientos en tiempo real. En otras palabras, utilizar métodos para manejar la complejidad y aumentar el reconocimiento en los modelos de clasificación actuales sin perder el objetivo de que dicho modelo debe ser lo suficientemente sencillo para poder introducirlo en un microcontrolador. Con esta simplificación en el procesamiento vía hardware/algoritmo se sientan las bases para desarrollar: controles en tiempo real y portabilidad, aplicaciones en ambientes de simulación musculo esquelético como Opensim y control de prótesis multifuncionales.

En este trabajo revisaremos los avances de cuatro áreas principalmente: Modelos de clasificación obtenidos a partir de aprendizaje automático clásico para señales EMG, avance en el prototipo de un microcontrolador capaz de implementar un modelo de clasificación, desarrollo de un sistema en línea de transmisión simultánea inalámbrica para señales EMG a un concentrador computadora y simulación en tiempo real con Opensim.

1.2 Análisis y decodificación de la señal EMG

1.2.1 Músculos y señales mioeléctricas

La composición de músculo esquelético es generalmente a partir de fibras musculares que a su vez forman el cuerpo muscular, este es el tejido que nos proporciona movimiento, nos permite la interacción con nuestro ambiente permitiéndonos realizar todas las actividades de la vida diaria, por lo tanto, es un tejido realmente importante y complejo. Nuestro cuerpo cuenta con más de 600 músculos, cada uno de estos nos brinda la habilidad de movernos de forma rápida y precisa. Cierta grupo de fibras que compone a un músculo en particular recibe una señal eléctrica del sistema nervioso para activar dicho músculo, el cerebro ya está entrenado para que a partir de un número finito de músculos generen un movimiento específico, por lo tanto, estas señales eléctricas nerviosas van a las fibras musculares para su activación que termina en un movimiento total útil. La EMGs nace a partir de estas señales eléctricas, en realidad el propio músculo genera un desencadenamiento eléctrico entre sus fibras para transferir la señal que recibe del sistema nervioso. La señal que nosotros podemos adquirir desde la piel cercana a los músculos que están siendo evaluados es lo que se le conoce como EMGs.

Desde aquí podemos vislumbrar el problema que nos enfrentamos para la decodificación de la señal EMG. Si cada fibra muscular que está generando movimiento genera una señal eléctrica pequeña y hay miles de señales eléctricas generándose en un instante de tiempo y solo somos capaces de leer la suma de todas estas señales que incluye la activación de múltiples músculos, entonces aquí vemos el problema principal. Es necesario separar la suma de estas señales eléctricas y darles un significado para representar el movimiento que se está realizando. Si eso fuera poco aún tenemos el problema de que no somos activadores de solo movimiento, también podemos diferenciar entre niveles de fuerza y esas señales también son proporcionales al nivel de fuerza que utilizamos en los movimientos. De esta forma aún falta describir el último de los problemas principales al que nos enfrentamos en estas investigaciones, los músculos parecen generar dos tipos de señales musculares una cuando se realiza el movimiento y la otra que es durante una contracción sostenida.

1.2.1.1 Señal mioeléctrica

Como se mencionó el EMGs es un conjunto de señales eléctricas de fibras musculares individuales todas superpuestas en determinado movimiento en un punto definido de la piel que es adquirido

con un electrodo superficial. Además, una señal mioelectrica se ha definido que comprende dos estados: un estado de transición cuando el músculo va del reposo a la activación y un estado estacionario producido durante una contracción constante mantenida en músculo [2].

1.2.2 Factores determinantes en la adquisición electromiografica.

El EMG es utilizado para estimación de la fuerza, análisis de cambio en el funcionamiento muscular causado por anormalidades, control de dispositivos médicos, control de aplicaciones, entre otros usos. Esta señal se ve modificada por la posición donde se encuentra el electrodo [3-6], por el tamaño, material y forma del electrodo [7-11], por el tipo de adquisición, si son electrodos diferenciales, doble diferenciales o puntuales [8,10], por la cantidad de electrodos para una región de adquisición [12,13], por las distancias inter electrodo [9,14-15], por el procesamiento que se les da a los datos adquiridos [16-20], por el tipo de movimiento por la fuerza y la rapidez del movimiento [9].

Por la naturaleza del EMG, es decir, por su apariencia estocástica es difícil trabajar con esta señal, considerando que son tantas las variables que intervienen no solo en la producción de las señales nerviosas que controlan el proceso de contracción muscular, sino en su adquisición y su decodificación, aun así, estudios previos indican que el reconocimiento de múltiples movimientos puede realizarse por medio de los datos EMG multicanal [21].

Su amplitud ronda los micro volts hasta los mili voltios, por su apariencia puede confundirse con ruido y en algunas ocasiones el ruido supera a la señal. Existen muchos factores que distorsionan la señal, como las interfaces electrodo piel, el ruido del ambiente, artefactos de movimiento, el mismo ruido producido por los dispositivos electrónicos utilizados.

Como ya se introdujo anteriormente la posición de los electrodos es un tema de interés en la investigación relacionada al EMG, si se utilizan más de uno, el seleccionar un arreglo adecuado es tema de investigación [3, 4, 22], de esta forma podemos observar cómo cada variante con posibilidad de modificar la señal se convierte en un nuevo tema de interés. Por lo tanto, las investigaciones relacionadas a esta señal son muy específicas o valoran la importancia del electrodo [14, 21] o se enfocan a re descubrir las posiciones específicas para los electrodos con una electromiografía de alta densidad [4,6,9,12,23-25] o establecen una posición arbitraria por ejemplo arreglos en forma de anillo y se dedican a procesar la señal con el presupuesto de que la

información que se necesita está cubierta con el arreglo seleccionado [21, 26-28]. Dentro del procesamiento encontramos muchas otras áreas que de igual forma pueden afectar la interpretación de la señal EMG, por ejemplo, si se establecen nuevos criterios para normalizar las mediciones, si aplican nuevos métodos de aprendizaje automático para clasificar movimientos [21, 23, 29]. Otras investigaciones se están enfocando a eliminar señales que se montan sobre el EMG como lo es la electrocardiografía. Se están desarrollando filtros muy útiles para deshacer esta deformación del EMG [7,18,19,30].

Otro factor de interés también relacionado con el procesamiento y con la colocación de electrodos en regiones es la comunicación cruzada, este es un efecto que modifica la señal, que involucra desde el tamaño del electrodo, el procesamiento y la posición de cada electrodo dentro de un arreglo, este efecto no distorsiona la señal, sino que puede complicar su interpretación.

La energía útil del EMG es de 0-500 Hz y la información utilizable está en el rango de 10-400Hz [7] con esta banda de frecuencias se ha demostrado que los movimientos de la mano pueden ser reconocidos, también se ha establecido que el nivel de la contracción afecta directamente la clasificación, uno de los puntos a considerar es que la fisionomía de las personas afecta directamente a la señal al cambiar un volumen muscular más grande producirá una amplitud mayor [21].

La electromiografía es un registro que está siendo estudiado en muchos aspectos porque es de fácil acceso y representa acciones voluntarias a las que estamos acostumbrados y funcionan como buenos comandos para controlar dispositivos.

1.3.1.1 Electrodo

La mayor parte de las investigaciones citadas utilizan los electrodos Ag/AgCl con variaciones en el diámetro, [3, 7, 9, 18, 19, 26, 30-32]. En [33] determinaron que el tamaño del electrodo y la distancia entre los electrodos son factores que modifican la amplitud de la señal, en dicha investigación demostraron que con un electrodo más pequeño y una distancia inter-electrodo (IED) de 10 mm se amplifica la señal, otra investigación importante es [9], donde se demuestra que los índices espectrales aumentan sistemáticamente como la distancia IED es reducida. También se encontró en [14] que el valor RMS con una contracción máxima voluntaria (MVC) de 50-80 % aumentaba con una distancia IED mayor en este caso 36 mm. Aquí encontramos una relación entre

el contenido espectral de una señal y su contenido de amplitud en el tiempo, con respecto a IED, concluyendo que al reducir la distancia IED aumentarían los índices espectrales de la señal y al mismo tiempo se reducirá su valor RMS.

Un dato importante de [9] es que para todas las IEDs, modalidades de contracción y músculos probados se encontró que las frecuencias espectrales aumentan significativamente con un nivel de fuerza arriba de los 50-60 % del MVC.

Comparando las investigaciones consultadas encontramos que hay un interminable número de combinaciones utilizadas en términos de diámetro, tipo, IED, [19] utilizó un diámetro de 1 cm y un espacio IED fue de 2 cm, [7] utilizó electrodos con un diámetro de 9 mm y 3 cm de IDE, [30] sin mencionar el diámetro de los electrodos utilizados y con una distancia IED de 25 mm, [18] con un diámetro de 2 mm paralelos al eje longitudinal del músculo y con un IED de 15mm, con el electrodo de referencia en la muñeca derecha, [31] utilizó un diámetro de 1 cm y 3 cm de IDE, con un tercer electrodo de referencia sobre la parte lateral del codo.

En [3] y [32] utilizaron electrodos bipolares con una distancia centro a centro de 20 mm y 25 mm respectivamente. En [32] Se utilizó un electrodo con un diámetro de 10 mm, esta investigación se encargó del estudio de los efectos de la variación de la IED sobre la señal EMG, sobre el músculo bíceps con contracciones isométricas variando el porcentaje de MVC con distancias evaluadas en 18 y 36 mm.

Otro factor determinante encontrado en [10] es la configuración de la adquisición de la señal EMG, si se utiliza una técnica single diferencial, double diferencial o branched electrode donde reportan que una 'double diferencial' o 'branched technique' proporciona un 5 % menos de comunicación cruzada que la 'single diferencial' , por lo tanto también modifica la información recabada, por el contrario encontramos investigaciones que no reconocen estos problemas ya que electrodos grandes son utilizados en sus trabajos, dichos electrodos generan un filtrado pasa bajas y un promediado de la señal que da como resultado una limitación en la calidad de la señal.

En [33] se establece que cuando un par de electrodos es utilizado la distancia IED debe ser más pequeña con respecto a la distancia entre la zona de inervación y el tendón.

1.2.2.2 Región anatómica en la adquisición

Gran parte de la investigación electromiográfica se centra en encontrar las posiciones más adecuadas y definir el tipo de arreglo que se debe utilizar para crear una mayor diferenciación entre

las señales de distintos tipos de movimientos, aunque estudios como [16] reportan que la posición de los electrodos no fue crítica en el reconocimiento de patrones para control cuando un número suficiente de electrodos es utilizado, usualmente más de 4, sin embargo, hay una variedad de estudios que comprueban la importancia de la posición de los electrodos. Básicamente son dos las formas de adquirir la información de los músculos:

a) Apoyándose bajo el presupuesto de que el utilizar un anillo de electrodos abarcando toda el área perimetral de antebrazo habrá la información necesaria para hacer la decodificación de la información.

b) Colocar los electrodos sobre los músculos involucrados en los movimientos evaluados y una técnica para resolver este último punto es utilizar un arreglo multicanal de alta densidad EMG para encontrar los puntos que en verdad son significativos al momento de hacer la adquisición.

Como ejemplo del inciso “a” encontramos a [21] donde utilizan un anillo circular para decodificar la información del EMG, de igual forma [27] se utiliza posiciones generalizadas con dos anillos concéntricos distales al pliegue del codo. Contemplando el inciso ‘b’ encontramos [3] donde se plantean el objetivo de evaluar todas las regiones de antebrazo, es decir una electromiografía de alta densidad con 192 electrodos, para definir cuál es la región más adecuada en la adquisición, donde se evaluaron más de 300 combinaciones para 11 diferentes movimientos. En [33] se demostró que la posición del electrodo en función de la región muscular es importante, y que la región entre la zona de inervación y la zona de los tendones es la mejor para hacer la adquisición. En [3] se establecieron las zonas de inervación de la región evaluada, aquí el objetivo de estas investigaciones es procesar esta información para desechar la información redundante por varios tipos de algoritmos.

La investigación [4] evalúa más de 200 diferentes combinaciones de electrodos promediando el reconocimiento de 11 diferentes movimientos contemplando la mano, muñeca y dedos, se evaluaron de 2 hasta 192 electrodos y se analizaron más de 300 combinaciones, también se utilizó una distancia IED de 10 mm en ambos lados 8 filas y 24 columnas anillo circular en antebrazo. La matriz fue aplicada en antebrazo 3 cm distales al pliegue del codo. En esta investigación se establece con 16 electrodos tal presentación es la representada por dos filas de ocho electrodos colocados lo más lejos posible entre sí en el primer tercio proximal del antebrazo. En [26] utilizaron canales virtuales para dos electrodos esto mejoró notablemente la clasificación, se seleccionaron los dos electrodos más distantes uno del otro sobre la masa muscular del radio y

sobre la masa muscular del cubito. Comprobaron que para un mismo movimiento es estadísticamente igual utilizar dos canales que cuatro.

1.2.2.3 Preprocesamiento y Procesamiento

En general el procesamiento de la información va dirigido a hacer decodificable la información de la señal EMG, es decir el definir los movimientos y características de estos que generan las activaciones musculares globales. En este contexto encontramos una variedad de algoritmos desde la normalización de la señal, la eliminación de la comunicación cruzada a través de separación ciega de fuentes y otros algoritmos similares que tratan de eliminar la información redundante del EMG para hacerla decodificable. Uno de los métodos más utilizados es la implementación de algoritmos de aprendizaje automático para redefinir la información y hacerla utilizable en un modelo de clasificación y generar una decodificación de la señal EMG/movimiento.

1.2.2.4 Eliminación de ruido

La señal EMG se encuentra en un rango de 10-2000 Hz y encontramos muchas investigaciones que valoran un rango de frecuencias dentro de esa banda [3,4,22,26,27,29,34,35]. La eliminación del ruido no es solo fuera de las bandas analizadas sino el eliminar ruido dentro de estas bandas es el objetivo de muchos trabajos, actualmente se ha buscado eliminar señales anexas como la del ECG de señales propias del músculo [7,18,19,30]. También es objeto de estudio la eliminación de las señales típicas como es la frecuencia de la línea de potencia 50- 60 Hz, los artefactos de movimiento y el ruido generado por los mismos dispositivos electrónicos utilizados.

Generalmente se utiliza un filtro de 4to orden Butterworth para obtener el ancho de banda requerido [3,36], en [7] se establece que un filtrado con un filtro Butterworth pasa alta con una frecuencia de corte arriba de los 60 Hz altera la señal EMG en tiempo y frecuencia. También proponen un nuevo método usando un filtrado dinámico, el filtro Savitzky-Golay. En este mismo trabajo se realizó un estudio comparativo con dicho filtro y con un filtrado pasa altas, (filtro Butterworth de cuarto orden con una frecuencia de corte de 30 Hz).

1.2.2.5 Normalización

Cuando se habla de normalización se contemplan muchas posibles variantes, desde normalizar las mediciones crudas de EMG, normalizar las características extraídas de esta señal, normalizar las mediciones entre cada uno de los movimientos evaluados, hasta las mediciones de distintos sujetos, todo esto con distintos fines dentro de ellos encontramos mejorar el rendimiento de las

características, el rendimiento del modelo de clasificación, y el poder concluir en función de un grupo de individuos.

En [3] se estandarizó la señal con la siguiente fórmula: $sEMG=X/\max[X]$, donde X representa la señal filtrada. Si bien el normalizar una medición EMG no es difícil el detalle es hacerlo de una forma correcta, ya que se debe evaluar el grado de contracción de cada uno de los registros porque la fuerza en los movimientos evaluados varía. La normalización se calcula como una razón de las características originales a las que son extraídas con una MVC (contracción máxima voluntaria) del músculo, con el fin de mejorar el rendimiento de las características en el dominio del tiempo se normaliza bajo el área promedio de la curva RMS, este enfoque requiere la información del sujeto realizando MVC, bajo este supuesto las señales EMG han sido normalizadas a valores de pico máximo, estos cambios repercuten en la señal original. En este mismo estudio se encontró que los datos normalizados fueron estadísticamente distintos a los originales con una $p < 0.05$. Además, los resultados mostraron que las diferencias entre los sujetos en las características EMG fueron reducidas cuando usaron la normalización propuesta.

1.2.3 Minería de datos y aprendizaje automático con señales EMG

Cuando se habla de aprendizaje automático es necesario hablar de minería de datos. Dentro de la minería de datos tenemos que siempre es necesario seleccionar características que le den a nuestros modelos la capacidad de funcionar adecuadamente. Con aprendizaje automático podemos crear modelos de clasificación que nos permitan seleccionar a través de un conjunto de datos (características) una clase determinada, también podemos generar modelos de regresión que nos permiten calcular valores continuos a partir de un conjunto de variables seleccionadas. En nuestro caso utilizamos aprendizaje automático para seleccionar un movimiento a partir de señales eléctricas musculares. Son muy pocas las investigaciones que tratan de crear un algoritmo de clasificación con la señal cruda. Lo que se hace generalmente es tomar las señales crudas a las cuales se le aplican una serie de operaciones llamadas características que nos dan información del estado de los movimientos generados para poder clasificarlos adecuadamente.

De los artículos resultantes se encontró que el flujo común de dichas investigaciones es la adquisición de la señal, pre procesamiento que involucra extracción de características, el procesamiento que es más que nada la implementación de uno o varios algoritmos de clasificación y la evaluación estadística para validar resultados y asegurar la separación de las clases evaluadas. En [21] establecen que el objetivo de las investigaciones ya no es solamente el crear un modelo de

clasificación para los movimientos si no el tener una mayor aplicabilidad, por lo tanto, es necesario un algoritmo que facilite directamente el uso del sistema sin cualquier tipo de entrenamiento, seleccionaron como características descriptoras longitud de onda (WL), cambio de signo de la pendiente (SSC), cruces por cero (ZC) y valor absoluto medio (MAV), características usadas también en [16] con un discriminante lineal como modelo de clasificación. En [29] se evaluó la raíz cuadrática media (RMS), Densidad del espectro de potencia (PSD), desviación estándar (SD), entre otras como características descriptoras. En [21] se prueban 5 modelos de clasificación Vecinos cercanos (KNN), Análisis discriminante cuadrático (QDA), Naive Bayes (NB), Random Forest y máquinas de soporte vectorial (SVM), dicha investigación utiliza un sistema de adquisición comercial como la Myo armband que da un reconocimiento de 90 %, sin embargo, con cada nuevo individuo se tuvo que sincronizar.

1.3 Avance en el prototipado de un microcontrolador capaz de implementar un modelo de clasificación.

El uso de microcontroladores se ha realizado más que nada para control de actuadores, en [37] se desarrollan varios clasificadores para movimientos de mano, y el resultado se envía a un microcontrolador para el control de los actuadores considerados. Se buscaron investigaciones que utilizaran los microcontroladores para el procesamiento de los datos. En [38], se establece el uso de un microcontrolador como el PIC32MX360F512L el cual establece la fuerza muscular a partir de un método novedoso llamado algoritmo de fusión de modelo lineal optimizado (OLMFA). Actualmente los microcontroladores en su mayoría se están utilizando para adquisición de los datos y transmisión a un concentrador [39] muy pocos realizan el procesamiento ahí mismo es decir la clasificación. Hasta ahorita sabemos que sería una tarea difícil ya que siempre es necesario reunir los datos del arreglo evaluado y procesar los datos a la par, por lo tanto, en este tipo de análisis siempre es necesario un microcontrolador concentrador. En [40] se establece la necesidad que está surgiendo para el monitoreo en tiempo real de variables fisiológicas, satisfaciendo la necesidad de la actual medicina móvil. Establecen que actualmente en el mercado dispositivos como pulseras están siendo un mecanismo para resolver estas circunstancias, en este caso se emplean varios sensores MPU6050 que es una unidad de medición inercial con 6 ejes con un microcontrolador para transferir los datos inerciales apoyándose de un microcontrolador MSP430f149.

1.3.1 Simulación de una clasificación en tiempo real con plataformas como Matlab, Labview y Python

Encontramos investigaciones que realizan el procesamiento de la señal adquirida en línea con programas como Labview [41,42], Matlab [43-47] y Python [46-48], donde se desarrolla un procesamiento en tiempo real. Aquí describimos los resultados más representativos.

Cuando un modelo de clasificación es creado, a veces es necesario probar su eficacia en tiempo real sin necesidad de activar actuadores reales, o sistemas motores completos, es por eso que se utilizan sistemas de procesamiento virtual, en plataformas como Matlab, Labview o Python. Esto permite hacer una pausa y verificar los resultados obtenidos, velocidades de procesamiento, errores de validación y porcentaje real de clasificación. Nos permite comprobar las validaciones fuera de línea y asegurarnos de que el proceso en tiempo real estará bien en la práctica.

El uso de un procesador completo con toda la capacidad que posee una computadora es útil ya que nos permite empezar la evaluación de la robustez de nuestro proceso y de las capacidades que podría necesitar un microprocesador o un microcontrolador.

Actualmente estos programas poseen herramientas que son de gran utilidad para la implementación de aprendizaje automático y desarrollo de clasificadores, muchas de las investigaciones aprovechan estas librerías para el desarrollo de sus trabajos, más adelante veremos cómo utilizamos las herramientas de Matlab y Labview para generar nuestros propios clasificadores y su implementación en tiempo real.

En la mayoría de los trabajos evaluados en esta sección se trabaja con sujetos sanos, introducen una selección de mejores características y mejores clasificadores utilizan Labview y Python para realizar una clasificación en línea. En [45] utilizan una banda comercial diseñada para leer señales EMG llamada Myoarmband. Con este dispositivo realizan la adquisición de las señales EMG, dejando a un lado el objetivo de desarrollar su propio sistema de adquisición. Realizaron un modelo para reconocimiento de gestos de la mano con una red neuronal artificial hacia adelante con un tiempo de respuesta de 227.76 ms. En [44] también utilizan este dispositivo y emplean la típica ruta para procesar señales EMG desde la extracción de características, entrenamiento y clasificación hasta la validación en tiempo real, utilizando Matlab como plataforma para realizar el procesamiento.

Otro trabajo donde también utilizan las herramientas de Matlab es [49] utilizan un clasificador simple LDA, evalúan las posiciones de los electrodos, de igual forma prueban algoritmos como PCA para verificar el desempeño de las características evaluadas. Terminando con la evaluación de su sistema en tiempo real. El último estudio considerado fue [39] donde cumplen con varios objetivos, desde la clasificación en tiempo real, el desarrollo de una pulsera para la adquisición de señales EMG y su transmisión vía inalámbrica a una computadora para su análisis.

1.4 Sistemas de transmisión inalámbrica en línea de datos para señales EMG

Uno de los objetivos de esta investigación es recabar todos aquellos avances en el área de la transmisión multicanal inalámbrica, para ponernos al nivel de los nuevos desarrollos comerciales. Como es parte de los objetivos establecer una ruta viable en el desarrollo de un dispositivo portátil e inalámbrico, con la suficiente robustez para adquirir una señal EMG de calidad. Aquí se establecen todas aquellas investigaciones de interés que manejan este tipo de objetivos. En el mercado se han disparado el número de microcontroladores incluyendo a los microprocesadores con tamaños casi idénticos, es decir, la línea que separa estos dos dispositivos cada vez se hace más delgada. De la misma forma sabemos que los microcontroladores están incluyendo más tecnologías en el área de la comunicación, agregándosele cada vez más protocolos para crear interfaces entre una gran cantidad de dispositivos periféricos que podemos encontrar en el mercado. De esta forma encontramos microcontroladores con Wifi, Bluetooth, Xbee, Zigbee, protocolos de comunicación inalámbricas con antenas de radiofrecuencia como NRF240, de igual forma los desarrolladores de estas comunicaciones como los mencionados, están agregando microcontroladores con convertidores analógico digitales en sus módulos de transmisión. Esto nos da una señal que el área de los dispositivos portátiles inalámbricos está desarrollándose a pasos agigantados. Ahora podemos encontrar integrados con convertidores analógico digitales, con un microcontrolador y un protocolo de transmisión inalámbrico. Sabemos que estas tecnologías aún están en desarrollo y son prácticamente nuevas por lo que sus ambientes de desarrollo aún son de difícil acceso. Lo que complica incluirlos en nuestras investigaciones.

La integración de una red de sensores inalámbricos viene con muchos desafíos, principalmente es la cantidad de información que se transferirá por un medio inalámbrico, es decir, de qué tamaño serán los paquetes de datos enviados y de cuantos dispositivos consta dicha red, también se incluye la velocidad de transferencia y la integridad de los datos enviados y recibidos. Por estas razones

muchas investigaciones optan por evaluar dispositivos comerciales, que les brindan ciertas características como en [26].

1.4.1 Investigaciones consideradas que buscan una transmisión inalámbrica eficiente de datos EMG

Hay pocas investigaciones que encontramos con un desarrollo completo desde la adquisición analógica de las señales, su digitalización, preprocesamiento y el envío de datos. Reducimos más la búsqueda al agregar trabajos que realicen una transmisión inalámbrica.

Actualmente en el mercado encontramos dispositivos portátiles basados en EMG, en ECG y en EEG dichos dispositivos tienen varios objetivos desde dispositivos médicos, industriales, de videojuegos, de aptitud física denominados muchas veces fitness, de entretenimiento [50]. Por lo tanto, es un mercado que se abre camino en la vida de las personas supliendo las necesidades actuales.

Como es de esperarse las señales EMG se adquieren de forma simultánea y necesitan ser procesadas, por lo tanto, fue necesario investigar aquellos trabajos que pudiesen transferir de forma simultánea los datos y al mismo tiempo procesarlos. Se encontraron trabajos que utilizan ciertos protocolos de comunicación inalámbrica como Bluetooth, Wifi, y sistemas de radiofrecuencia como NRF24I01.

1.4.1.1 Bluetooth

Actualmente las investigaciones se encuentran utilizando dispositivos comerciales para realizar la adquisición de los datos, uno de los más usados es Myoarmband pulsera que permite realizar una adquisición de EMG sin preocuparse por la parte del sistema de adquisición. En [51] se evalúa este dispositivo sus capacidades y sus limitaciones, este es un dispositivo con 8 electrodos de EMGs con una unidad inercial de 9 ejes y un módulo de transmisión inalámbrica Bluetooth que te permite muestrear en tiempo real las señales musculares de antebrazo para los movimientos de mano, muñeca y antebrazo. Está siendo una excelente opción para desarrolladores de aplicaciones en el cuidado de la salud como su uso en prótesis de miembro superior.

1.4.1.2 Nrf24I01

Nordic semiconductor es la compañía que se encuentra desarrollando módulos de radiofrecuencia integrados con poderosos microcontroladores y con otros protocolos de comunicación,

actualmente está compitiendo en la comunicación inalámbrica del mercado, siendo sus módulos perfectos para una implementación de conexiones tipo malla, de bajo consumo. Sus módulos poseen un mejor rendimiento energético, mayores buffers de memoria y velocidades de transferencia de hasta 2 Mbps. Los módulos NRF de radiofrecuencia están siendo muy utilizados, ya que te permite trabajar simultáneamente con más dispositivos que Bluetooth, incluso te permite trabajar con Bluetooth simultáneamente, también encontramos que su consumo energético es menor y que están incorporando estos módulos con otros protocolos como Bluetooth, Zigbee. En [52] encontramos el uso de estos módulos para una transferencia de datos de forma inalámbrica específicamente señales EMG.

1.4.1.2 Wifi

Aunque wifi te ofrece una alta gama de funciones, tiene sus desventajas desde su consumo energético hasta el tamaño de los módulos, estas dos limitaciones están siendo un problema al tratar de considerarlos en el mundo de los dispositivos portátiles, su desarrollo está destinado principalmente a dispositivos para el hogar, donde nace el concepto del internet de las cosas.

1.5 Simulación en tiempo real con Opensim

Opensim es un software de simulación músculo esquelético que ha ido mejorando sus capacidades conforme pasa el tiempo, inicialmente era un programa que te permitía realizar análisis biomecánicos de una variedad increíble de modelos, en los últimos años este programa ha mejorado y se le han agregado herramientas para poder analizar los movimientos en tiempo real, en [53,54], trabajan con las herramientas de Opensim apoyándose en herramientas externas tanto Python y Matlab, para generar un análisis en tiempo real.

1.5.1 Investigaciones que trabajan con Opensim como instrumento de simulación.

Opensim provee herramientas de fácil uso, es un programa extensible es decir cada participante puede desarrollar herramientas que se apeguen a su tema de estudio, este programa fue desarrollado bajo la comunidad de SimTK y es básicamente para modelar, simular, controlar y analizar sistemas neuromusculares [55]. Opensim se desarrolla bajo una comunidad internacional SimTK con aportes de todo el mundo, trabajando y mejorando sus modelos de simulación encontramos trabajos que aumentan o validan las herramientas utilizadas por este programa.

1.5.2 Trabajos fuera de línea

La simulación músculo esquelética está teniendo un gran impacto en el ambiente médico, desde la simulación de cortes musculares y reinserciones musculares para aliviar ciertos padecimientos

incluyendo la planeación de cirugías para niños con parálisis cerebral [56], hasta en terapias de rehabilitación.

En el ámbito biomecánico se están realizando proyectos para evaluar la deformación muscular en las contracciones que producen movimientos como es el caso de [57], el objetivo de este último es permitir la exploración del modelado de la fuerza muscular, para mejorar los gráficos músculo esqueléticos y aumentar el control de dispositivos de asistencia.

También encontramos trabajos más puntuales que permiten entender un poco más como la arquitectura muscular y los momentos afectan los movimientos de flexión y extensión de la muñeca [58].

1.5.3 Trabajos en línea

Desde hace un par de años en Opensim se han desarrollado trabajos para incluir una simulación en línea, aunque en un inicio la plataforma de Opensim no te permite trabajar con simulaciones en tiempo real, se han incorporado herramientas con Matlab y Python para hacer uso de un ambiente en tiempo real. Se inició con el proyecto simulador de brazo dinámico, el objetivo fue desarrollar un modelo músculo esquelético para la simulación dinámica en tiempo real, este modelo incluye un modelo completo de tronco hombro, brazo, codo, antebrazo, muñeca y mano. Esta simulación es implementada usando una función MEX y usa Opensim para el preprocesamiento y la visualización [59]. En [53] se establece un trabajo derivado de este proyecto, el objetivo es evaluar la estimulación eléctrica en paciente con pérdida de la función de brazo. Como se sabe el control del brazo paralizado bajo estimulación eléctrica es un problema que requiere sistemas avanzados de control para el usuario. Por lo tanto, se busca un modelado en tiempo real, otro de los objetivos de dicho trabajo es evaluar el rendimiento y comportamiento mecánico de un modelo dinámico en tiempo real del brazo y de la cintura escapular.

En [54] se presenta los conceptos que permitieron desarrollar las funciones para la dinámica músculo-esquelética fuera de Opensim, se establece como este tipo de simulaciones en tiempo real requieren pequeños pasos de tiempo y los problemas de control óptimo son difíciles de calcular. De esta forma se desarrollaron las funciones en Matlab y Python que permiten realizar estos cálculos y hacer uso de los modelos y las herramientas visuales de Opensim.

En el último año Opensim se ha actualizado desde la versión 4.0 hasta la 4.2 donde se agregaron varias herramientas para hacer uso de un sistema de evaluación en tiempo real con señales de sensores de movimiento. Opensense permite a los usuarios calcular el movimiento de segmentos del cuerpo basándose en datos de unidades de medición inercial, esta herramienta es novedosa libre y gratis, en [60] aclaran los objetivos del análisis del movimiento humano desde el diagnóstico de trastornos de movimiento hasta evaluación de la enfermedad de Parkinson, también explican que los sistemas comerciales actuales para este tipo de análisis son costosos y plantean un sistema portátil de bajo costo y de código abierto que es capaz de estimar la cinemática de las extremidades en tiempo real.

CAPITULO 2 MARCO TEORICO

2.1 Introducción

En este capítulo repasaremos todos los conceptos, métodos y algoritmos que se utilizarán para definir el marco de trabajo y experimentación. Basándonos en nuestros objetivos buscando el describir una ruta que nos permita generar modelos de clasificación simples capaces de ser embebidos en microcontroladores. De igual forma poder simular movimientos en un programa de simulación biomecánica como Opensim a partir de la clasificación de señales EMG a través de un modelo generado con aprendizaje automático.

Iniciaremos desde los conceptos generales de la señal EMG, los músculos considerados para una adquisición de señales, los métodos de adquisición y los dispositivos que encontramos disponibles en el mercado.

Se abordarán los sistemas de adquisición analógicos para señales EMG, se considerarán los microcontroladores y sus convertidores analógico digitales (la capacidad de procesamiento de instrucciones y sus puertos de comunicación). Una gran parte de este capítulo está enfocada a métodos de aprendizaje automático y procesamiento de la información.

El mundo de los microcontroladores es cada vez más extenso y la línea que los separa de los microprocesadores cada vez se hace más difusa. Actualmente los microcontroladores poseen procesadores más poderosos, capaces de trabajar con 32, 64 y hasta 128 bits, tiene memorias más grandes y convertidores analógico digitales con mayor resolución, poseen cristales capaces de trabajar a altas velocidades para una transferencia de datos y ejecución de instrucciones con la mayor rapidez posible. Aunque hay una gran variedad pudimos reducir la búsqueda con algunas características que se desean y son necesarias al momento de trabajar con señales musculares como, por ejemplo, que posea un convertidor analógico digital de al menos 12 bits, que posea una ejecución de instrucciones de al menos decenas de microsegundos, que posea un protocolo de comunicación compatible con módulos Bluetooth, y sobre todo una plataforma de programación de libre acceso y con soporte.

Otro de los puntos clave en esta investigación es la transferencia de datos de forma inalámbrica, se evaluaron varios protocolos de comunicación inalámbrica con lo cual encontramos una gran cantidad de módulos para cada tecnología evaluada, nos dimos cuenta que los desarrolladores de estos módulos están integrando microcontroladores robustos para el procesamiento de los datos que son capaces de transmitir, eso conlleva a que los ambientes de programación de dichos módulos sean particulares para cada fabricante. Muchos de estos ambientes son de difícil acceso, es decir, sus librerías se encuentran descuidadas o hay poca información sobre su programación. Por lo tanto, necesitamos averiguar los dispositivos que tenemos disponibles en el mercado, desde los más actuales hasta los clásicos. La selección del módulo se realizará en función de nuestros requerimientos, una alta velocidad de transferencia de datos, un diseño compacto y un bajo consumo energético.

Terminaremos revisando conceptos generales en el uso de Opensim para simulación de movimiento de un modelo músculo esquelético virtual. Revisaremos las bases que encontramos en la plataforma SIMTK para una simulación en tiempo real, con un modelo biomecánico de antebrazo y mano.

2.2 Señal EMG

La señal mioeléctrica denominada también señal electromiográfica es aquella señal que se produce en el músculo al momento de recibir la señal nerviosa para que el músculo se contraiga. Una definición particular de Carlo J. De Luca es: “La electromiografía provee fácil acceso a procesos fisiológicos que causa el músculo para generar fuerza, producir movimiento y generar múltiples funciones que nos permiten interactuar con el mundo que nos rodea” aunque la señal electromiográfica puede obtenerse a partir de varios métodos el más simple y rápido es la electromiografía de superficie, aquí la señal se adquiere a través de electrodos superficiales que se colocan sobre la piel, es un proceso no invasivo, también encontramos electromiografías donde la señal se adquiere a través de agujas, este método es invasivo y debe ser llevado a cabo por un especialista. Un ejemplo de este tipo de señales adquiridas a través de un electrodo de superficie es el mostrado en la Figura 2.1.

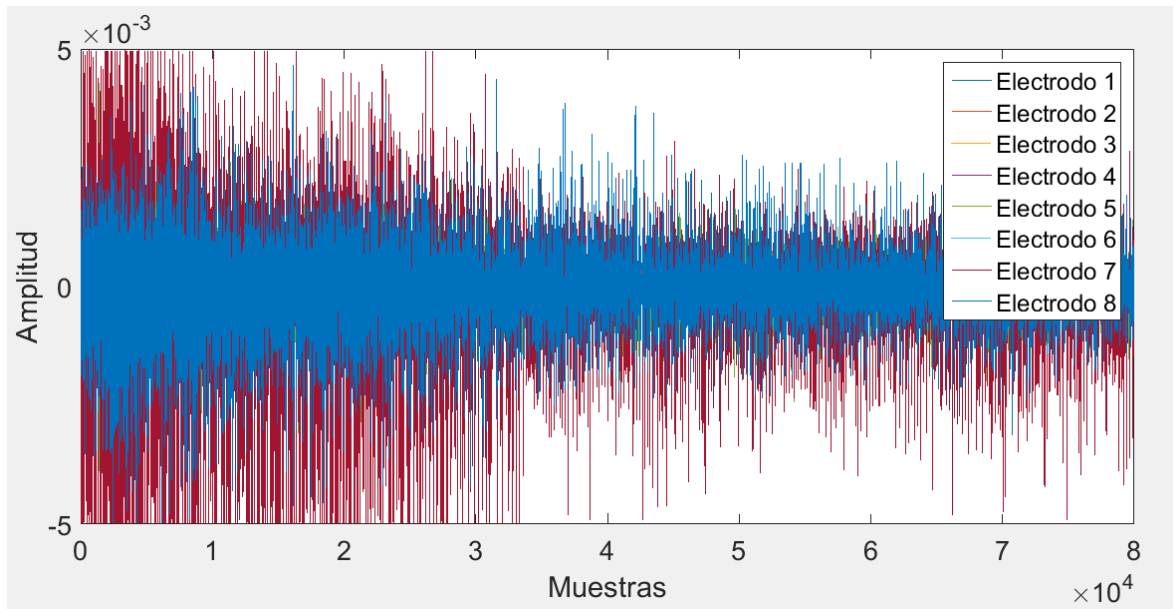


Figura 2.1. Electromiograma de Mano cerrada y sus 8 electrodos.

2.2.1 Músculos de antebrazo

Antebrazo pertenece a la porción libre de miembro superior conformado por el radio y el cubito, seguidos por los huesos de la mano que incluyen los metacarpos, las falanges, falangetas y falanginas. Estas secciones poseen una gran cantidad de músculos junto a tejido circundante como pleuras que separan los músculos, vasos sanguíneos que transportan la sangre oxigenada y la desoxigenada, nervios los impulsos de sistema nervioso central, tejido adiposo que es una reserva de energía, vasos linfáticos, nódulos y piel como barrera ante el medio ambiente.

Encontramos 28 músculos en antebrazo-mano de los cuales 19 se encuentran en antebrazo, técnicamente podemos dividir las secciones musculares de antebrazo en tres partes un plano profundo, uno intermedio y uno superficial, en la Figura 2.2 podemos visualizar estos planos. Todo músculo del cuerpo posee la capacidad de contraerse y dependiendo de la posición anatómica donde se encuentre adquirirá su funcionalidad. Los músculos que se encuentran en la parte posterior de antebrazo generalmente son para movimientos que tienen que ver con extensión de la mano y dedos sobre antebrazo y los músculos que se encuentran en la sección anterior son músculos flexores de la mano y dedos sobre el antebrazo.

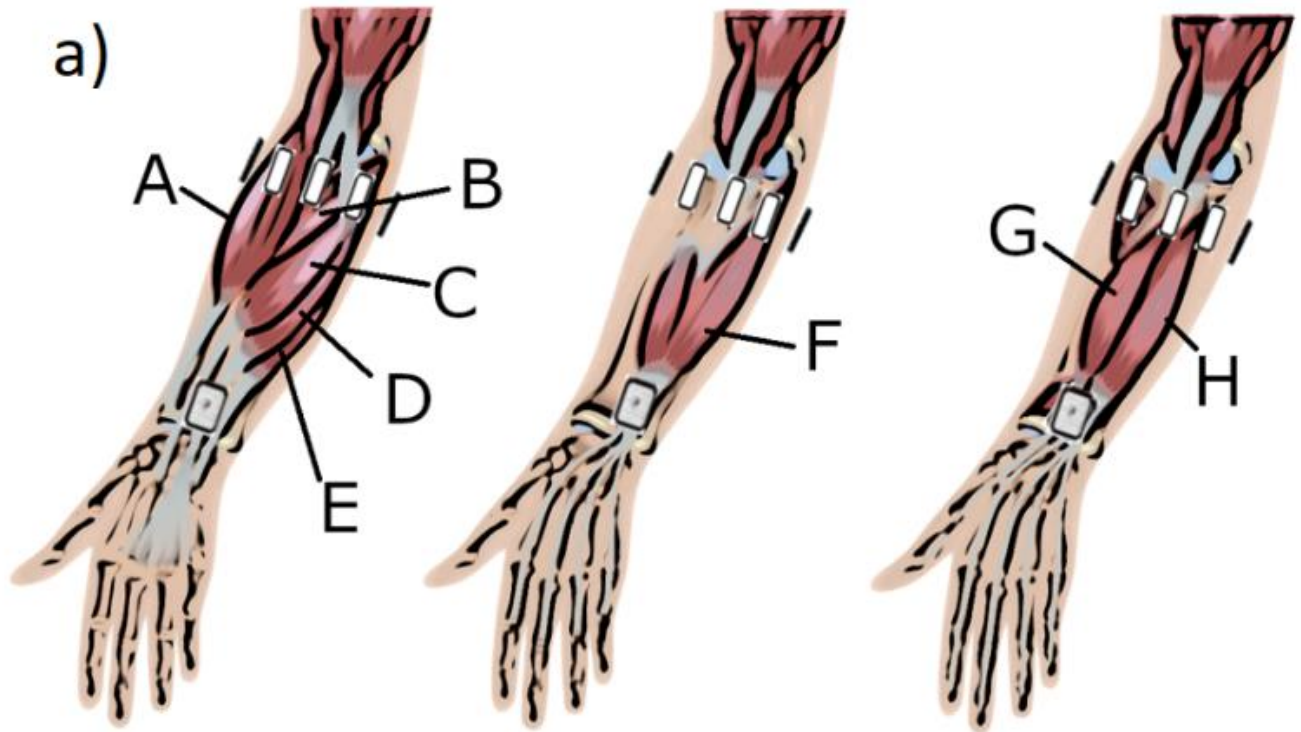


Figura 2.2. a) Principales músculos involucrados en la región de adquisición del arreglo utilizado A: Brachiocondrial (BR), B: Pronator teres (PT), C: Flexor carpi radialis (FCR), D: Palmaris longus (PL), E: Flexor carpi ulnaris (FCU), F: Flexor digitorum superficialis (FDS), G: Flexor pollicis longus (FPL), H: Flexor digitorum profundus (FDP).

2.3 Adquisición electromiografica y su procesamiento con aprendizaje automático

Generalmente se realiza una adquisición de señales EMG con el objetivo de generar procesamiento de los datos con aprendizaje automático y obtener un modelo de clasificación para un procesamiento en tiempo real. Cuando se realiza este tipo de procesamiento hay varios parámetros que se deben cuidar, entre ellos la disponibilidad de electrodos, la región anatómica de adquisición, el sistema de adquisición analógico y su procesamiento digital. Por otro lado, tenemos el procesamiento y el aprendizaje automático utilizados generalmente para crear un modelo de clasificación capaz de tomar los datos EMG y transformarlos en movimientos bien clasificados. Como último paso en estas técnicas encontramos a la validación de los resultados para evitar un sobreajuste del modelo. Se busca la ruta para obtener un modelo que sea capaz de discriminar las clases con nuevos datos de señales EMG. En esta sección revisaremos todos aquellos elementos para lograr este último objetivo

2.3.1 Electrodo en el mercado

En el mercado encontramos una gran cantidad de electrodos disponibles con variaciones en la forma, tamaño, material y tipo, se buscan electrodos superficiales dada la simplicidad de su uso. Dentro de estos electrodos tenemos a los de tipo palanca, pulsera, de chupón, hasta integrados con guantes y los más comunes los adhesivos. Un electrodo es un sensor capaz de registrar señales eléctricas, técnicamente se encuentra definido como el extremo de un conductor que registra una señal eléctrica en contacto con el medio. Ejemplos de estas señales en sistemas vivos tenemos la electrocardiografía, electromiografía y encefalografía.

2.3.2 Selección de las Regiones anatómicas

Hay una gran cantidad de posiciones sugeridas por múltiples investigaciones, desde el lugar óptimo para obtener la señal de un músculo individual, hasta el punto óptimo para obtener las señales EMG de una región como antebrazo. Incluso encontramos libros que nos muestran los puntos sugeridos para evaluar las señales eléctricas para cada músculo de antebrazo, desde como ubicar al músculo hasta donde colocar el electrodo [61].

Cuando evaluamos un conjunto de movimientos en específico debemos tener en cuenta los músculos que participan en dichos movimientos, para así tener información sobre la activación muscular real de los movimientos que se desean evaluar. El lugar óptimo para obtener la señal es el centro del cuerpo muscular donde el trabajo se lleva a cabo.

Aunque en el mercado encontramos equipos que nos facilitan todo lo relacionado a la adquisición de señales como los equipos Delsys y sus sensores EMG De-2.1 nos enfocaremos a sistemas de adquisición desarrollados con las características que buscamos.

2.3.3 Sistema de adquisición analógico para EMG

Los sistemas de adquisición analógicos para EMG están concentrándose en circuitos integrados, por ejemplo, el AD8232 que es un sistema de adquisición de biopotenciales generalmente utilizado como monitor de la frecuencia cardíaca, como este hay muchos otros como por ejemplo el ADS1293 de Texas Instruments, este tipo de dispositivos facilitan la creación de sistemas de instrumentación médica escalables. Estos dispositivos vienen integrados con filtros analógicos configurables, con amplificadores de instrumentación y con tamaños reducidos para poder incorporarlos a sistemas portátiles, son de bajo consumo eléctrico ayudando a su portabilidad.

2.3.3.1 Circuitos Integrados en el mercado

Los circuitos integrados para la adquisición de bioseñales están siendo agrupados en un grupo de circuitos llamados AFE (Analog front-end) siendo un conjunto de circuitos acondicionadores de señales analógicas que usan amplificadores analógicos, amplificadores operacionales de instrumentación y filtros.

Nombrando dos de las compañías que lideran el mercado de la electrónica encontramos a Texas instruments (TI) y Analog Devices (AD), estas poseen un gran número de AFEs. Evaluaremos un solo dispositivo de cada compañía empezando por el AD8232 de AD y el ADS1293 de TI.

2.3.3.1.1 AD8232

El AD8232 es un bloque integrador acondicionador de señales para ECG que puede ser utilizado para otro tipo de bioseñales. Sus aplicaciones van desde monitor de ritmo cardiaco en la actividad o el deporte, ECG portables, monitores remotos de salud y adquisición de biopotenciales. está diseñado para extraer, amplificar y filtrar biopotenciales pequeños en la presencia de ruido, su diseño con un ultra bajo consumo de energía permite incorporarlo a dispositivos portátiles [62]. El voltaje que se le suministra es de 3.6 V, tiene filtros para eliminar los artefactos de movimiento y filtros para seleccionar un ancho de banda específico, a partir de un filtrado pasa bajas y uno pasa altas.

2.3.3.1.1.1 Diseños sugeridos

Aunque los diseños que encontramos en las hojas de datos de este dispositivo son para filtrar señales ECG a una frecuencia definida, encontramos que Analog Devices proporciona un simulador de filtros para cambiar el ancho de banda evaluado, permitiéndonos modificar las frecuencias de adquisición. El programa podemos encontrarlo en [62] en la sección de “Tools & Simulations” y es utilizado para calcular todos los elementos que se necesitan para crear un filtro con frecuencias de corte de nuestra elección.

2.3.3.1.1.2 Programa para diseños de circuitos eléctricos con el AD8232: Diptrace

Es un programa para el diseño electrónico, permite crear diseños de tarjetas de circuitos impresos, generar los diagramas de los esquemáticos, archivos necesarios para crear placas de circuitos impresos, crear diseños en 3D de las placas diseñadas, entre otras aplicaciones.

2.3.4 Preprocesamiento

La señal cruda del EMG es la señal que obtenemos directamente de los electrodos siendo una señal filtrada, en un ancho de frecuencias de 20 a 500Hz, amplificada, dicha señal se encuentra dependiendo de la ganancia que se le dio, valores de voltajes que pueden ir desde los mili-volts hasta los volts. Si se evalúa un arreglo de 8 electrodos se tendrán mediciones de 8 electrodos, como se ha mencionado las señales eléctricas musculares tienen una apariencia estocástica, es decir las señales tienden a parecer ruido aleatorio, sin embargo, sabemos que tienen un patrón en función a la fuerza de contracción, a la velocidad y al tipo de movimiento evaluado. Estas señales pocas veces se utilizan directamente en modelos de clasificación, antes de eso se preprocesan, de cierta forma se pulen los datos para hacerlos decodificables. Entre los procedimientos típicos que encontramos está el ventaneo de los datos, la selección de características que se le aplicaran a cada una de las ventanas, y la normalización de los datos para darles a todas las características evaluadas un mismo peso en los clasificadores generados.

2.3.4.1 Base de datos

Se realizó una búsqueda exhaustiva para encontrar bases de datos publicadas en línea que fueran de libre acceso [63-69], para luego determinar cuál era la más conveniente a utilizar. Algunas características de las bases de datos consultadas se pueden observar en la tabla 1. Se seleccionó la base de datos [67], ya que fue una de las más completas con movimientos que abarcan cada uno de los dedos. La base de datos consiste en 15 movimientos de dedos, tanto movimientos individuales como combinados. Los datos fueron adquiridos con un arreglo en anillo de 8 electrodos EMGs en el antebrazo, a 5 cm distal al pliegue del codo, teniendo en cuenta que los datos recolectados deben tener un estándar de adquisición para tener una referencia. Esta región se encuentra ubicada entre la mayor cantidad de cuerpos musculares y la región de las uniones de la articulación del codo [70]. Ocho sujetos normales fueron evaluados, la ejecución de cada movimiento se repitió 12 veces; más detalles sobre los métodos de adquisición, así como las características de los sujetos pueden ser encontrados en [70]. Los 15 movimientos corresponden a flexiones, incluyendo la flexión de cada uno de los dedos individuales y combinaciones entre estos: pulgar (T), índice (I), medio (M), anular (R), meñique (L), las combinaciones T-I, T-M, T-R, T-L, I-M, M-R, R-L, I-M-R, M-R-L, y finalmente la mano cerrada (HC).

Tabla 2.1 Base de datos consideradas para el análisis propuesto.

Nombre	Bases de datos		
	Número de electrodos	Número de movimientos	Número de sujetos
UCI Machine Learning Repository [63]	2	6	6
IEE EMG Database* [64]	12	6,10	4
SEEDS Dataset [65]	134	13	25
CapgMyo [66]	128	8,12	18
EMG Datasets Repository* [67]	2-10	10-15	8
The Ninapro Project: Dataset 2* [68]	12	50	10-40
PutEMG* [69]	24	8	44

En la Figura 3 se muestra el arreglo de electrodos utilizado en el antebrazo. En este tipo de arreglos de electrodos, la información relacionada con el movimiento generado se encuentra en el tejido circundante y se transmite en todas direcciones, entonces esta información es suficiente para generar una clasificación.

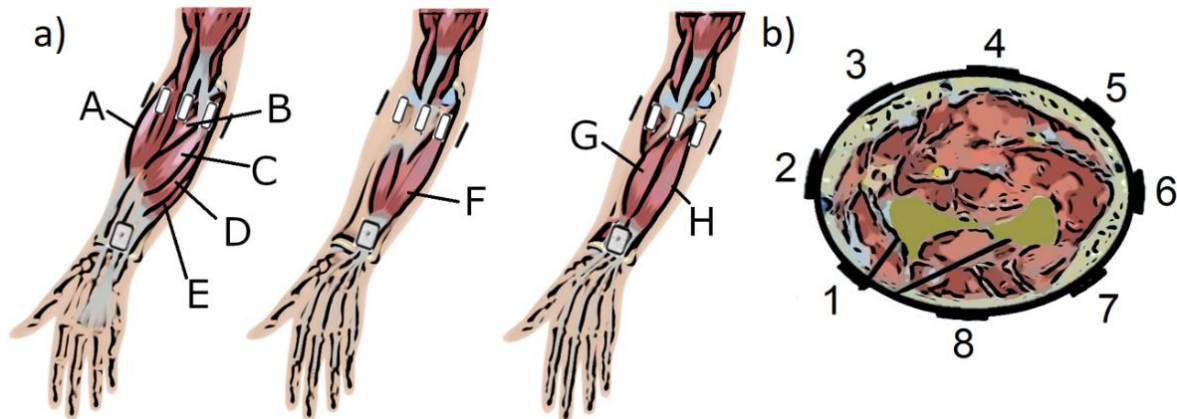


Figura 2.3. a) Principales músculos que intervienen en la región de adquisición del arreglo de electrodos, cada uno de ellos participa en la adquisición de la señal de mayor o menor intensidad dependiendo que tan cerca o alejado se encuentre del electrodo. A: Braquiocardial (BR), B: Pronator teres (PT), C: Flexor carpi radialis (FCR), D: Palmaris longus (PL), E: Flexor carpi ulnaris (FCU), F: Flexor digitorum superficialis (FDS), G: Flexor pollicis longus (FPL), H: Flexor digitorum profundus (FDP). b) arreglo utilizado en la adquisición de la base de datos, donde se puede observar el arreglo anillo utilizado.

Toda la información de movimientos de mano se transmite por esa sección del antebrazo, aunque no es posible indicar un músculo específico para cada electrodo dentro del anillo. La figura 3 muestra que hay músculos más cercanos a cierta región de un electrodo, pero no significa que ese músculo sea el principal o que tenga acción dentro de los movimientos evaluados. La información con la que se cuenta es que hay músculos en una región y electrodos en una posición dada cuando

se capturó la base de datos. La figura 3 muestra músculos flexores, separados en tres capas de profundidad: una superficial “superficial”, una media “middle” y una profunda “deep”. En esta situación todos los electrodos captan señales de todas las capas de distribución muscular. Hay cinco músculos en “superficial”, uno en “middle” y dos en “deep”. Los músculos principales en esta región de adquisición, son: Braquiorradial (BR), Pronator teres (PT), Flexor carpi radialis (FCR), Palmaris longus (PL), Flexor carpi ulnaris (FCU), Flexor digitorum superficialis (FDS), Flexor pollicis longus (FPL), Flexor digitorum profundus (FDP), aunque no todos se activaron en los movimientos de la base de datos evaluados.

La Tabla 2 enlista los músculos principales de extensión y flexión de la región del antebrazo. Hay músculos que sí se activaron durante la adquisición de datos para la creación de la base de datos analizada. Mientras que hay músculos que no participan en los movimientos que fueron considerados en la base de datos, por ejemplo, los movimientos de extensiones de los dedos, movimientos de la muñeca como flexión, extensión, pronación y supinación. La Tabla 3 muestra la relación entre 3 músculos flexores encargados del movimiento de los dedos y como ellos participan activamente en cada uno de los 15 movimientos de la base de datos.

Tabla 2.2 Extensores Extensor carpi radialis (ECR). Extensor carpi ulnaris (ECU). Extensor digitorum communis and Extensor digiti minimi propius (EDC y EDMP). Extensor Pollicis Longus (EPL). Flexores: Flexor carpi radialis (FCR). Flexor carpi ulnaris (FCU).

Movimiento	Extensores				Flexores				
	ECR	ECU	EDCy	EDMP	EPL	FCR	FCU	FDP	FDS
Flexión de los dedos.								x	x
Extensión de los dedos			x		x				
Flexión de la mano						x	x		
Extensión de la mano.	x	x							
Pronación de la muñeca. (músculo cercano a)								x	
Supinación de la muñeca. (músculo cercano a)	x								

La base de datos consta de 8 sujetos y 15 movimientos, en donde cada sujeto realiza 12 repeticiones para cada uno de los movimientos. O sea 1,440 repeticiones de algún tipo de movimiento. Una repetición de un movimiento es equivalente a recolectar, a razón de 4,000 muestras por segundo, las señales de 8 sEMGs durante 5 segundos. Un sEMG se codifica con 12 bits, pero se lee como una unidad de punto flotante. Por lo tanto una repetición de un movimiento por un sujeto equivale a $(4,000\text{m/s}) \times (5\text{s}) = 20,000$ paquetes de datos (muestras), con 8 datos de cada electrodo. La base de datos se descarga de [67], donde se obtiene un archivo zip con 8 carpetas, una carpeta por cada sujeto, y dentro de cada una de esas carpetas se encuentran archivos con formato csv nombrados con el nombre de los movimientos. Los datos se cargan con la función “*csvread*” de Matlab. Por cada sujeto, se crea una matriz para cada movimiento, nombrada con la abreviación que encontramos en la Tabla 3. Esta matriz se compone de 8 columnas sEMG y de 240,000 datos renglones, lo que representa el total de las 12 repeticiones de un movimiento por un sujeto. Además, se le agrega una columna con número de etiqueta que clasifica el tipo de movimiento, según la Tabla 2.3. De tal manera que en todo momento se sabe que muestra le corresponde a cada sujeto, que tipo de movimiento y número de electrodo.

Se crea una carpeta en Matlab para cada sujeto donde se guardan las 15 matrices de los movimientos para cada sujeto. Se concatenan las 15 matrices de un sujeto en una nueva matriz nombrada suji, esta matriz es de 3,600,000 filas por 9 columnas, y se guarda la matriz de cada sujeto.

Después, en una matriz global se guardan los datos de los 8 sujetos formando una matriz de 28,800,000 filas y 9 columnas; todo esto se indica en la Figura 3 con el círculo 1. Cabe aclarar en este punto que los datos se encuentran ordenados del sujeto 1 al sujeto 8, es decir los datos de la matriz va con los movimientos del 1-15 del sujeto 1, después del 1-15 del sujeto 2, y así sucesivamente hasta los movimientos del sujeto 8. Cuando haya un cambio en la columna 9, de tipo de movimiento, que cambie de la etiqueta 15 a la etiqueta 1, indica el borde de datos que existe entre dos sujetos.

Tabla 2.3 Músculos y su relación con los movimientos de la base de datos.

Etiqueta clase	Movimientos evaluados	FDP	FDS	FPL
1	HC	X	X	X
2	I_I	X	X	
3	I_M	X	X	
4	IMR	X	X	
5	L_L	X	X	
6	M_M	X	X	
7	M_R	X	X	
8	MRL	X	X	
9	R_L	X	X	
10	R_R	X	X	
11	T_I	X	X	X
12	T_L	X	X	X
13	T_M	X	X	X
14	T_R	X	X	X
15	T_T	X	X	X

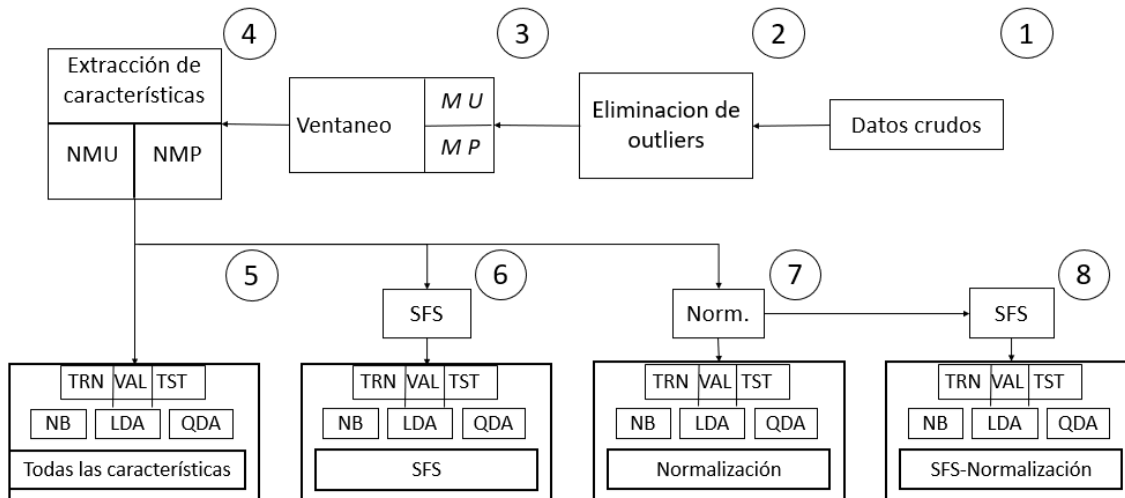


Figura 2.4 Experimentación realizada para determinar cuál es la mejor opción en la creación de un modelo de clasificación considerando porcentaje de reconocimiento, y simplicidad del modelo. MU=Matriz única (los 8 sujetos concatenados), MP=Matriz por sujeto (se forman 8 matrices por cada sujeto). El número de los círculos definen el orden del procesamiento.

Por la gran cantidad de datos se decidió reducirlos con métodos de limpieza y organización estructural antes de llegar a las etapas de creación de clasificadores. Por lo que antes de la extracción de características, se incluyen pasos que van desde la eliminación de valores atípicos, la preparación de los datos en una matriz única o en varias (por sujetos individuales) para su aplicación en los algoritmos de clasificación, y hasta la configuración del tamaño de la ventana de datos para extraer información relevante. Se aplicaron los siguientes métodos.

2.3.4.2 Ventaneo de datos

El ventaneo es una técnica utilizada para extraer la información útil de la señal EMG. Un segmento es una porción de los datos de la señal EMG, las ventanas de datos pueden ir desde una decena de datos EMG hasta cientos de datos, básicamente es una sección de datos en el tiempo, como muchas investigaciones tratan de trabajar en tiempo real se procura que sean pequeñas ventanas de datos para aumentar la velocidad de clasificación, sin embargo, muy pocos datos acarrear problemas ya que se acerca a la aleatoriedad de la propia señal EMG. Se sabe de [71] que la longitud de un segmento debe ser lo suficientemente largo ya que la varianza de las características aumenta cuando la longitud del segmento decrece.

2.3.4.3 Características seleccionadas

Las características evaluadas en este trabajo se encuentran en el dominio del tiempo, aunque podemos encontrar que algunas características en el dominio de la frecuencia son más útiles, aunque trabajar en este dominio es costoso computacionalmente. Si trabajamos en el dominio del tiempo debemos tener en cuenta que la señal posee una propiedad no estacionaria, sin embargo, se tratan a los datos como una señal estacionaria [72]. Las características seleccionadas fueron valor absoluto medio (MAV), integral cuadrada simple (SSI), la varianza (VAR), momentos temporales (TM3, TM4 y TM5), la raíz cuadrada media (RMS), longitud de onda (WL), cruces por cero (ZC), cambio en el promedio de la amplitud (AAC), porcentaje de miopulso (MYOP) y cambio de signo de pendiente (SSC). A continuación, describiremos matemáticamente cada una de estas características.

Valor absoluto medio (MAV)

El valor absoluto medio se puede definir como el promedio del valor absoluto de la amplitud de la señal de un segmento y matemáticamente es descrito como sigue:

$$MAV = 1/N \sum_{i=1}^N |x_i| \quad (1)$$

Integral cuadrada simple (SSI)

Este parámetro es utilizado como un índice de la energía de la señal y se expresa de la siguiente forma:

$$SSI = \sum_{i=1}^N x_i^2 \quad (2)$$

La varianza del EMG

La varianza es una medida de dispersión que representa la variabilidad en este caso de la señal EMG y se representa:

$$VAR = \frac{1}{N-1} \sum_{i=1}^N x_i^2 \quad (3)$$

Momentos temporales (TM3, TM4, TM5)

Los momentos temporales son estimadores estadísticos, así como la media, varianza, sesgo y la curtosis, estos parámetros se han utilizado para extraer información de señal EMG [16 tesis]:

$$TM3 = \left| \frac{1}{N} \sum_{i=1}^N x_i^3 \right|; TM4 = \left| \frac{1}{N} \sum_{i=1}^N x_i^4 \right|; TM5 = \left| \frac{1}{N} \sum_{i=1}^N x_i^5 \right| \quad (4)$$

Raíz cuadrática media (RMS)

La raíz cuadrática media también es una medida estadística que representa la magnitud de una cantidad variable:

$$RMS = \sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2} \quad (5)$$

Longitud de forma de onda (WL)

Longitud de forma de onda (WL) es una medida de la complejidad de la señal EMG [16]. Esta es definida como longitud acumulativa de la forma de onda en un segmento de tiempo. Se calcula como:

$$WL = \sum_{i=1}^{N-1} |x_{i+1} - x_i| \quad (6)$$

Cruces por cero (ZC)

Es una medida como su nombre lo indica de las veces que la señal cruza cierto umbral, se define como una medida de frecuencia en el dominio del tiempo. Es el número de veces que la amplitud pasa por el límite establecido como valor cero. Puede establecerse a través de $x_i > 0$ y $x_{i+1} < 0$, o $x_i < 0$ y $x_{i+1} > 0$ y $|x_i - x_{i+1}| \geq \text{umbral}$. Dicho matemáticamente de otra forma tenemos:

$$ZC = \sum_{i=1}^{N-1} \left[\text{signo}(x_i * x_{i+1}) \cap |x_i - x_{i+1}| \geq \text{umbral} \right] \quad (7)$$

$$\text{signo}(x) = \begin{cases} 1 & \text{if } x \geq \text{umbral} \\ 0 & \text{en otro caso} \end{cases}$$

Cambio en el promedio de la amplitud (AAC)

Cambio en el promedio de la amplitud (AAC) es cercanamente equivalente a WL excepto que la longitud de onda se promedia [16]. Y puede expresarse como:

$$ACC = \frac{1}{N} \sum_{i=1}^{N-1} |x_{i+1} - x_i| \quad (8)$$

Porcentaje de miopulso (MYOP)

Porcentaje de miopulso (MYOP) es un valor promedio de la salida del miopulso el cual es definido como el valor absoluto cuando el valor de la amplitud excede cierto umbral [16] y puede formularse así:

$$MYOP = \frac{1}{N} \sum_{i=1}^N [f(x_i)] \quad f(x) = \begin{cases} 1 & \text{if } x \geq \text{umbral} \\ 0 & \text{en otro caso} \end{cases} \quad (9)$$

Cambio de signo de pendiente (SSC)

Cambio de signo de pendiente (SSC) está relacionada con características como ZC, MYOP [16]. Este es otro método que representa información en frecuencia de la señal EMG:

$$SSC = \sum_{i=1}^{N-1} [f[(x_i - x_{i-1}) * (x_i - x_{i+1})]] \quad (10)$$

$$f(x) = \begin{cases} 1 & \text{if } x \geq \text{umbral} \\ 0 & \text{en otro caso} \end{cases}$$

En donde el umbral usado fue igual a 70 mV en estas características es normal seleccionar un umbral entre 50 μ V y 100 mV [73].

La literatura consultada ha demostrado que las características que son útiles para extraer información de las señales EMG son ampliamente utilizadas en clasificadores. Como parte de los objetivos de este trabajo es generar un procesamiento de lo más sencillo y rápido en términos de velocidad de ejecución, se decidió evaluar solo características en el dominio del tiempo. Como se menciona en [72], una de las mayores desventajas de este tipo de características es que se asume una propiedad de estacionariedad, propiedad que no coincide con la señal EMG, más cuando se usan ventanas de análisis muy pequeñas, sin embargo, da buenos resultados. En muchos estudios se habla sobre el tamaño de estas ventanas teniendo en cuenta el procesamiento que se le dará a los datos, en este caso se busca un procesamiento en tiempo real, por lo tanto, conviene trabajar con ventanas de datos pequeñas. Un conjunto de ventanas sin traslape son mostradas en la Figura 5.

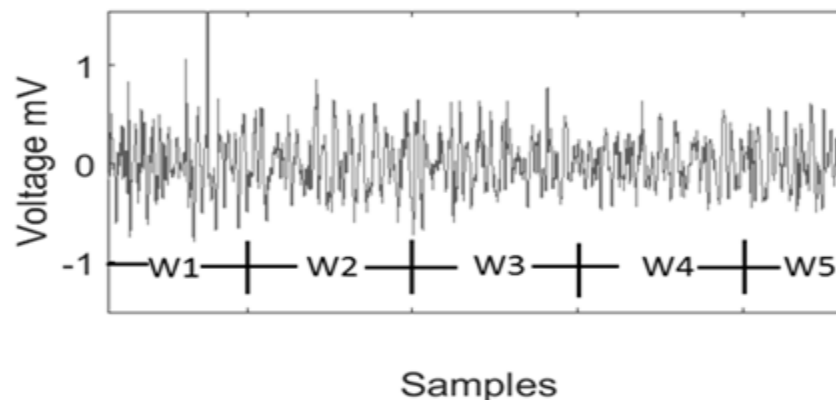


Figura 2.5 Selección de la ventana sin traslape.

Las 12 características seleccionadas fueron concatenadas produciendo un vector de 96 datos (características) por cada ventana por sujeto. Una ventana sin empalme fue utilizada para procesar los datos EMG crudos, con una longitud de 250 muestras por ventana. Esta longitud en tiempo real es equivalente a 62.5 ms en completarse una ventana de datos. Evaluando las 12 características seleccionadas.

2.3.4.4 Normalización

Si se transforma el conjunto de datos evaluado a un conjunto de datos normal es decir que posea una media cero y una desviación estándar 1, se ponen en un mismo rango de valores todas las mediciones de todas las características evaluadas, de esta forma todas tendrán el mismo impacto

en cada una de las operaciones de nuestro clasificador. El procedimiento se realizó de forma general para los tres clasificadores.

2.3.4.5 Método de eliminación de valores atípicos por una distribución chi cuadrada

De acuerdo a la literatura, un valor atípico es considerado una observación que se desvía de las otras observaciones y despierta sospechas de que es generado por otros mecanismos externos a los naturales. Algunos métodos estadísticos para la detección de valores atípicos multivariados utilizan el centro de distribución de los datos y una distancia establecida, en nuestro caso la distancia Mahalanobis para determinar qué datos son atípicos. Se utilizó el método de eliminación de valores atípicos con una distribución Chi cuadrada [74]. Ahora, durante el método de eliminación de valores atípicos, cuando se detecta una lectura irregular de un electrodo, se eliminan todos los 8 canales que corresponden a ese evento atípico y su etiqueta. Se toma la matriz compuesta por los 8 sujetos y se le aplica el algoritmo de valores atípicos definido en [74].

2.3.5 Minería de datos y aprendizaje automático

Cuando hablamos de procesamiento es indispensable empezar a hablar de minería de datos, la minería de datos busca extraer información que se encuentra escondida de un conjunto de datos. Esta ciencia nace a partir de ciencias como la comunicación, la computación y la estadística. El objetivo de usar los conceptos y métodos de la minería de datos es crear rutas que sean capaces de extraer los patrones escondidos en los datos del EMG, apoyándonos en el aprendizaje automático para hacer uso de estos patrones y crear modelos que sean capaces de discriminarlos. De esta forma nuestros modelos tendrán la capacidad de discriminar las clases de movimientos. La minería de datos también nos proporciona muchos métodos para visualizar los datos, separarlos, reagruparlos y encontrar su distribución en algún espacio.

2.3.5.1 Algoritmos de clasificación considerados

Cuando se trabaja con modelos de clasificación obtenidos a través de aprendizaje automático, es deseable tener características que generen una máxima diferenciación entre clases. Así los modelos de clasificación tendrán un mejor rendimiento. Una de las tareas fue buscar las características que mejor separan las clases en uno o más espacios. Los problemas de clasificación son divididos a menudo en dos etapas: La etapa de decisión y la de inferencia, en la figura 6 se visualizan estas dos etapas.

Input data may contain preprocessing

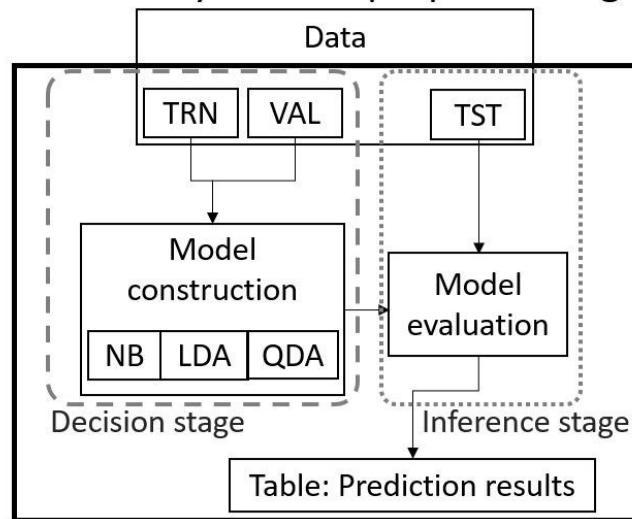


Figura 2.6. Creación de un modelo de clasificación

La etapa de decisión es cuando el modelo está siendo entrenado. La etapa de inferencia es utilizada en clasificación para predecir datos una vez que el modelo ya está entrenado. Estas etapas fueron extensamente utilizadas en este estudio.

Los clasificadores considerados por su simplicidad son Naive Bayes, Análisis discriminante lineal y Análisis discriminante cuadrático, todo ellos con un fundamento bayesiano y teoría probabilística. La importancia de estos clasificadores radica en que son muy eficientes aparte de que son simples, y han sido utilizados para implementar aprendizaje automático en microcontroladores. La función “*fitcdiscr*” de Matlab se usó para crear LDA o QDA, y se utilizó la función “*fitcnb*” para Naive Bayes.

2.3.5.2 Optimización en la selección de características SFS

La función a grandes rasgos de este método es analizar cada una de las características de forma ordenada, es un proceso iterativo una vez que haya analizado cada una de las características individualmente (tomará la que proporcione el mejor porcentaje de reconocimiento) posteriormente pasará a formar grupos de dos, una vez haya conseguido la mejor de las características individuales la unirá con cada una de las restantes, posteriormente se selecciona la que mayor porcentaje brinda al clasificador así sucesivamente hasta que el porcentaje no sea mayor. Es un proceso donde se toman todos los datos disponibles y se crean modelos de

clasificación para evaluar los porcentajes de reconocimiento del modelo, para cada subconjunto de características. El algoritmo 1 muestra el pseudocódigo utilizado para este algoritmo.

Algoritmo 1: Pseudocódigo implementado para SFS.

```
SUBSET= 0; Best_EVAL = 0;
SPACE=1...L // L= number of total features.
EXIT=false
WHILE NOT EXIT
FOR i =1 TO |SPACE|
TEMP(i)=J(SUBSET V SPACE(i)) // TEMP= Temporal variable.
END
BEST_i=ARGMAX(TEMP) // The maximum value of the
// TEMP vector is stored in BEST.
IF TEMP(BEST_i)>BEST_EVAL //Comparison between the best stored
SUBSET = [SUBSET , SPACE(BEST_i)] //evaluation and the best evaluation
BEST_EVAL=TEMP(BEST_i) //of the new subset.
SPACE=SPACE-SPACE(BEST_i)
ELSE
OUTPUT=true
END
END
RETURN SUBSET, BEST_EVAL
```

Estos resultados solo son para seleccionar que conjunto de características hace que el clasificador se desempeñe mejor. Cada clasificador considerado busca separar de la mejor forma las clases, y por lo tanto puede tomar las características que mejor resultados les dé.

2.3.5.3 Transformación a nuevos espacios: Método de matrices dispersas

La conversión de nuestros datos a nuevos espacios, aunque aumenta el procesamiento puede dar buenos resultados en el aumento del reconocimiento del modelo clasificador y/o en la reducción de características. Este algoritmo se utiliza en primera instancia para aumentar las distancias entre las clases evaluadas y reducir las distancias entre los dos datos de las clases en [75] se muestra la descripción de este método.

2.3.6 Métodos de validación en aprendizaje automático

2.3.6.1 Formulaciones estadísticas

La formulación de sensibilidad, especificidad, precisión y Puntaje F1 son parámetros estadísticos que nos permiten evaluar los resultados obtenidos, podemos obtenerlos a través de una matriz de

confusión, donde se compara movimiento con movimiento ubicando específicamente cada uno de los errores que se tuvo al clasificar los datos. Se pueden visualizar en la tabla 3.

Tabla 2.4 Formulaciones estadísticas donde TP=True positive, FP=false positive, TN=True Negative y FN=False Negative.

Formulación	Operación
A=Sensibilidad	$TP/(TP+FN)$
B=Especificidad	$TN/(TN+FP)$
C=Precisión	$TP/(TP+FP)$
D=F1 score	$2TP/(2TP+FP+FN)$

2.3.6.2 Validación cruzada

Uno de los métodos más utilizados en la validación de los modelos de clasificación en aprendizaje automático es la validación cruzada, este método nos permite dentro de un conjunto de datos definido considerar todos los datos como entrenamiento, como validación y como prueba, de esta forma se valida que los resultados son independientes de las particiones de los datos.

Matlab posee una función llamada “*cvpartition*” que separa los datos (particiona) para una validación cruzada.

2.3.6.3 ANOVA

Se utilizó el análisis de varianzas (Anova) para comparar y estudiar el efecto de los métodos aplicados a nuestros datos sobre las medias de los porcentajes de reconocimiento obtenidos a partir de los clasificadores evaluados. Este análisis es un test estadístico a emplear cuando se desean comparar las medias de dos o más grupos. La hipótesis nula de la que parten los diferentes tipos de ANOVA es que las medias de los grupos son estadísticamente iguales, es decir la media es la misma en los diferentes grupos.

2.3.6.4 Método de la prueba de significancia estadística de Duncan

Este método es un test de comparaciones múltiples. Nos permite comparar las medias de los tratamientos (procedimientos que se le aplican a los datos y dan como resultado un porcentaje de reconocimiento) después de haber rechazado la hipótesis nula.

2.4 Microcontroladores y simulación en línea

Un microcontrolador es un circuito integrado, podría compararse con una pequeña computadora que incluye sistemas para controlar elementos de entrada salida. El microcontrolador generalmente se compone de un procesador, su núcleo, y elementos anexos como convertidores analógico-digitales, memorias tanto para guardar el programa y sus variables (memoria flash y RAM).

2.4.1 Microcontroladores en el mercado

Las empresas que dominan el mercado son Texas Instruments, Microchip, Arduino, Atmel, Motorola, Intel, NEC, ST y Philips entre otros. Encontramos microcontroladores clasificados por la longitud de su palabra, con elementos de 4, 8, 16, 32, 64 hasta 128 bits. También se pueden clasificar en función a su conjunto de instrucciones, que pueden ser del tipo CISC y RISC o por la arquitectura de su memoria que puede ser HARVARD o Von Neumann.

Para seleccionar un microcontrolador evaluamos algunas características que debían tener, desde soporte, diversidad de librerías, un ADC con una resolución mayor de 12 bits y una plataforma de desarrollo accesible y no obsoleta para las nuevas herramientas en investigación, que cuente con una velocidad suficiente para alcanzar un muestreo igual o mayor a 1 kHz, a parte que sea de bajo consumo energético, de bajo costo, que sea pequeño para incluirlo en una tarjeta de adquisición portátil y que tenga una placa de desarrollo. Fue necesario incluir esta última característica porque en la mayoría de los microcontroladores no hay una placa de desarrollo y si son nuevos el aprender a usarlos trae nuevos retos desde desarrollar una placa para poder utilizarlos. Como son microcontroladores pequeños muchos de ellos no son fácilmente acoplables a un circuito y generar un prototipo funcional. En este sentido nuestra gama de microcontroladores reduce su número drásticamente.

Uno de los puntos considerados para la selección del microcontrolador fue la resolución de su convertidor analógico-digital (ADC). Un ADC es un periférico que convierte continuamente voltaje analógico a un número discreto digital.

Empezamos por microcontroladores de Texas Instruments, las placas de desarrollo consideradas fueron TM4C123G esta placa cuenta con un ADC de 12 bits de resolución, un reloj interno de 80 MHz de tipo ARM con unidad de punto flotante, 256 KB de memoria flash, 32 KB de memoria SRAM y 2KB de memoria EEPROM, posee aplicaciones de ejemplo gratis junto a un amplio conjunto de librerías, también posee librerías gráficas y una cadena de herramientas de desarrollo

actualizadas con ambientes de desarrollo libre como Code composer studio, aparte también se puede desarrollar código bajo el ambiente Keil e IAR.

Otra placa de desarrollo muy conocida en el mercado es la placa de Arduino Due con un procesador Atmel SAM3X8E, es la primera placa arduino con un microprocesador ARM de 32 bits, con un reloj de 84 MHz, 2 DAC. Tiene 512 KB de memoria flash para almacenar código, posee una SRAM de 96 KB [76]. La placa se puede programar con el software de Arduino IDE [76].

2.4.1.1 Microcontroladores en el mercado capaces de realizar aprendizaje automático:

TM4C123G

Dentro de las librerías con las que cuenta este microcontrolador encontramos unas para realizar aprendizaje automático directamente en el CPU, con librerías para implementar una red neuronal funcional. Con la unidad de punto flotante permite realizar operaciones con una mayor eficiencia, nos brinda la capacidad de implementar modelos lineales simples. Aparte consta de librerías para realizar operaciones con matrices y procesamiento digital de señales.

2.4.1.1.1 Plataformas de desarrollo Code Composer Estudio y Keil.

Code composer studio (CCS) es un entorno de desarrollo integrado desarrollado por Texas Instruments que admite la amplia gama de procesadores integrados y microcontroladores de esta compañía, incluye un compilador optimizado de C/C++, editor de código fuente, entorno de construcción de proyectos y depurador. CCS fue desarrollado bajo el software Eclipse con las capacidades de depuración integradas avanzadas de TI, siendo su distribución gratuita [77].

Keil MVision es también un entorno de desarrollo integrado combina la gestión de proyectos, las funciones de creación, edición de código fuente y depuración de programas. Es un software con soporte, actualizado y con programas de ejemplos. Consta de diversas utilidades que se pueden utilizar para conocer el funcionamiento y la programación de diversos dispositivos. Es un software que necesita licencia para poder utilizarse, no es de libre acceso [78].

2.4.2 Plataformas para simulación en línea de un proceso de clasificación.

Fue necesario incluir algunos ambientes de programación para desarrollar los clasificadores considerados, se evaluaron 3 principalmente Matlab, Labview y Python.

2.4.2.1 Matlab

Matlab pertenece a Mathworks un líder en el desarrollo de software para calculo técnico, es un lenguaje de programación de alto nivel, provee una vasta librería de herramientas, combina un entorno de desarrollo optimizado para procesos de diseño y análisis iterativos con un lenguaje de programación que expresa la matriz y las matemáticas de matriz directamente. Dentro de las capacidades de Matlab está el análisis de datos, el uso de gráficos, desarrollo de algoritmos, creación de aplicaciones web y de escritorio, uso de Matlab con otros idiomas como Python C/C++, Fortran, Java entre otros, es capaz de conectar hardware a Matlab, de realizar computo paralelo incluso puede ejecutarse en entornos de nube. Matlab puede utilizarse para sistemas de control, aprendizaje automático y profundo, procesamiento de imágenes y visión artificial, para mantenimiento predictivo, robótica, procesamiento de señales, y comunicaciones inalámbricas, entre muchas otras aplicaciones [79].

2.4.2.2 Labview

Labview pertenece a National Instruments es un software de ingeniería de sistemas para aplicaciones que requieren pruebas, medidas y control con acceso rápido a hardware e información de datos. Este programa ofrece un enfoque de programación gráfica, permitiendo visualizar de forma global todos los elementos del programa, incluyendo hardware, datos de medidas y depuración. Nos permite medir sistemas físicos con sensores y actuadores, validar y verificar diseños electrónicos, desarrollar sistemas de pruebas de producción, diseñar maquinas inteligentes entre otras aplicaciones [80].

2.4.2.3 Python

Python es un lenguaje de programación ejecutado a través de un intérprete, es interactivo y orientado a objetos, incorpora una amplia gama de módulos, excepciones tipos de datos dinámicos de muy alto nivel. Es un lenguaje de programación de propósito general de alto nivel aplicable a muchas clases diferentes de problemas. Este lenguaje admite múltiples paradigmas de programación más allá de la programación orientada a objetos. Igual que Matlab tiene múltiples librerías para trabajar en muchos contextos, desde procesamiento de datos, aprendizaje automático, procesamiento de imágenes, visión artificial, sistemas de control, aplicaciones web, procesamiento de cadenas (expresiones regulares, Unicode), protocolos de internet (HTTP, SMTP, FTP, XML),

ingeniería de software, procesamiento de lenguaje, aprendizaje profundo entre muchas otras. Se ejecuta en muchas variantes de Unix, incluidos Linux y macOS, aparte de Windows [81].

2.5 Sistemas comerciales para la adquisición inalámbrica de señales EMG

2.5.1 Myoarmband

El brazalete es un arreglo de sensores musculares capaces de reconocer diferentes gestos realizados por la mano, es un sensor de alta sensibilidad, posee sensores de acero inoxidable de grado médico, consta de nueve ejes altamente sensible, tres para el acelerómetro, tres del magnetómetro y tres del giroscopio, consta de un procesador ARM Cortex M4. Se comunica via Bluetooth 4.0 con equipos de cómputo y con dispositivos móviles. Este dispositivo permite controlar la tecnología de forma inalámbrica con gestos de la mano, tiene un ambiente de desarrollo abierto y gratuito.

2.5.2 Delsys trigno avanti Sistema trigno lite

El sistema EMG inalámbrico de la compañía Delsys Inc. es un dispositivo de alto rendimiento. Cada sensor posee un acelerómetro triaxial incorporado, con un rango de transmisión de 40 m con un batería recargable que puede durar hasta 7 horas. El sistema es capaz de transmitir datos al software de adquisición y análisis de EMGworks y de generar 16 canales analógicos de EMG [82]. Algunas de sus características más destacables son: ancho de banda para una señal EMG de 20-450 Hz, una frecuencia de muestreo de hasta 4000 muestras por segundo, 16 bits de resolución, un acelerómetro integrado de 3 ejes. La comunicación inalámbrica ocurre sobre una variedad de frecuencias de 2.4 GHz.

2.6 Protocolos de comunicación inalámbrica

La necesidad de los problemas a los que nos enfrentamos en este momento está cada vez más relacionada al internet de las cosas. Sistemas más compactos y con un menor consumo de energía, con características como sistemas multiprotocolo proporcionan a bluetooth particularidades que satisfacen las necesidades actuales.

Uno de los pilares clave en esta investigación es la comunicación inalámbrica, existen en el mercado varios protocolos de comunicación que utilizan la frecuencia de 2.4 GHz, desde Bluetooth, Wifi hasta módulos de radiofrecuencia como los creados por la compañía NRF Nordic semiconductors.

2.6.1 Bluetooth

. Uno de los intereses por los que se decidió desarrollar este trabajo es una evaluación técnica de la capacidad de los modelos de clasificación convencionales contra los nuevos modelos complejos en su mayoría y sus implicaciones en el mundo de la portabilidad. Por lo tanto, es necesario encontrar módulos de adquisición y transferencia de datos que cumplan con los requerimientos establecidos. El objetivo buscado es la evaluación de la multitransferencia entre módulos esclavos y un maestro de forma simultánea, ya que el sistema de adquisición requerido es una malla de varios módulos de adquisición independientes que envíen la información capturada a un concentrador para que realice algún procesamiento o envíe los datos a un equipo de cómputo en tiempo real para su procesamiento.

Bluetooth Special Interest Group (Bluetooth SIG) es una asociación privada sin ánimo de lucro. Los miembros de SIG dirigen el desarrollo de la tecnología inalámbrica Bluetooth, además de implementar y comercializar la tecnología en sus productos. Bluetooth es un protocolo estándar IEEE 802.15.1, generado para trabajar con redes WPAN. Bluetooth trabaja bajo las frecuencias libres de radio en la banda ISM de 2.4 GHz al igual que otras redes inalámbricas. Utiliza técnica de espectro ensanchado con saltos de frecuencia, con 79 canales de radio-frecuencia con un ancho de banda de 1 MHz cada uno y una tasa máxima de símbolos de hasta 1 Mbps. La variedad de módulos Bluetooth en el mercado es a veces preocupante, el seleccionar uno entre miles de ellos es un tema prioritario si queremos utilizar tecnología Bluetooth. De las 5 versiones de Bluetooth existentes hasta este momento se pueden clasificar en 2 tipos: los módulos clásicos y los de bajo consumo.

Se consideró la velocidad a la que envían datos los módulos esclavos, la velocidad a la que adquiere datos el módulo concentrador y su capacidad para adquirir señales digitales seriales en 7 u 8 puertos virtuales independientes-simultáneamente. Incluyendo aparte, la velocidad de transmisión que juega un papel importante en estos sistemas de adquisición. Actualmente las señales de interés bio-fisiológico se están enfocando a tener datos con más resolución (aumento en la cantidad de datos) y una mayor frecuencia de muestreo para poder analizar dichas señales, lo que conlleva un aumento en las exigencias del hardware.

Además, para poder cumplir nuestro objetivo de utilizar un concentrador en la PC, se evaluó un módulo de Bluetooth v5 tipo dongle usb para que funcione como concentrador ya que los módulos

Bluetooth incorporados en la portátil son módulos multiprotocolo con Bluetooth v4 que aceptan como máximo 7 dispositivos enlazados al mismo tiempo además manejan una recepción de datos de hasta 1 Mbps.

Una de las similitudes entre sistemas es que todos vienen con una programación predeterminada, son fáciles de usar, el probar sus características es decir hasta donde son capaces de funcionar es otro tema.

2.6.1.1 Módulos Bluetooth en el mercado

Bluetooth HC06 model ZS-040, es un módulo bluetooth para establecer comunicación inalámbrica de corto alcance, funciona con el protocolo de comunicación bluetooth 2.0 y solo es capaz de actuar como dispositivo esclavo. El dispositivo funciona en el rango de frecuencias de 2.402 GHz a 2.480 GHz [83]. Es fácil de conectar y comunicar, es una de las opciones más económicas en el mercado, consume muy poca energía y puedo funcionar con batería, su comunicación es asíncrona con puertos UART por lo tanto es capaz de ser conectado con casi todos los controladores o procesadores del mercado.

2.6.2 Nordic semiconductor

Nordic semiconductor es una empresa de semiconductores sin fabrica enfocada en tecnología inalámbrica para el internet de las cosas. La empresa fue pionera en la tecnología inalámbrica y ayudó a desarrollar Bluetooth LE [84].

El módulo evaluado de Nordic es un sistema ultra compacto, multiprotocolo y de bajo consumo, sin embargo, cuando nos introducimos a sistemas de este tipo, con las mejores características del mercado nos topamos con varios retos, sistemas complejos y difíciles de programar, escasos recursos para desarrollar aplicaciones específicas, sistemas caros y pocas aplicaciones compatibles con los nuevos dispositivos, sistemas que son difíciles al momento de incorporar a nuestros diseños de circuitos, sin embargo es una buena forma de iniciar en el mundo de la competitividad.

2.6.3 Wifi

La marca registrada Wifi pertenece a Wifi Alliance, esta empresa impulsa la adopción y evolución global de WiFi a través del liderazgo intelectual, la defensa del espectro y la colaboración de toda la industria. Su trabajo incluye el desarrollo de tecnología, requisitos y programas de prueba. Wifi igual que Bluetooth es una tecnología de red inalámbrica, a diferencia de Bluetooth, Wifi te permite conectarte a una red con otros dispositivos y además pueden interactuar con internet.

2.6.4 Dispositivos multiprotocolo

Como se mencionó Bluetooth es una tecnología y como tecnología los desarrolladores de dispositivos están integrándola con otros protocolos inalámbricos, el siguiente modulo es un ejemplo de esto. EL módulo nRF52840 es un SoC multiprotocolo posee tecnología Bluetooth 5, Thread y Zigbee. Este dispositivo resuelve muchos problemas cuando trabajamos con simultaneidad de protocolos, posee una buena disponibilidad de memoria tanto para flash como para RAM. Como se mencionó tiene soporte para Bluetooth 5, Bluetooth mesh, Thread, Zigbee, 802.15.4, ANT y pilas propietarias de 2.4 GHz. Posee un núcleo basado en ARM Cortex M4 de 32 bits con una unidad punto flotante que funciona a 64 MHz.

2.7 Opensim

Opensim es un sistema programable y extensible por el usuario, es libre y permite a desarrolladores ampliar sus capacidades, desarrollar modelos, estructuras músculo esqueléticas y crear simulaciones dinámicas de movimiento. Es una plataforma que permite modelar el cuerpo humano, animales, robots y ambientes simulando su interacción y movimiento. Opensim tiene una interface gráfica de usuario para visualización de modelos, generación de simulaciones y análisis de sistemas músculo esqueléticos predefinidos. Posee una interface de programación de aplicaciones que es usado generalmente para extender el programa o agregar funcionalidades. Con Opensim se puede: crear y editar modelos, analizar y simular modelos y movimientos, desarrollar flujos de trabajo y extender el programa, usar una plataforma de código abierto para investigación y con fines didácticos para la enseñanza [85]. Encontrando soporte, comunidades y fuentes de investigación.

2.7.1 Estructura de Opensim fuera de línea

Opensim cuenta con 3 herramientas indispensables para el análisis biomecánico de los modelos evaluados, el escalado, la cinemática inversa y la dinámica inversa.

La cinemática inversa calcula los ángulos articulares para un modelo músculo esquelético que reproduce mejor el movimiento de un sujeto. Luego la dinámica inversa usa los ángulos de las articulaciones, velocidades y aceleraciones angulares del modelo para resolver las fuerzas de reacción netas y los momentos netos en cada una de las articulaciones [86].

La cinemática inversa es el estudio del movimiento sin considerar las fuerzas y momentos que producen ese movimiento. Su propósito es estimar los ángulos de un sujeto de estudio en particular

a partir de datos experimentales. Recordemos que los datos experimentales recopilados a través de un sistema de adquisición son de un sujeto en particular, dicho sujeto tiene medidas, movimientos particulares, si esos mismos datos los introducimos en un modelo escalado virtual diferirán algunos parámetros, lo que hace la cinemática inversa es tomar estos datos experimentales y hacer que el movimiento del modelo se ajuste en su mejor medida al movimiento real del sujeto. Opensim determina esta mejor coincidencia resolviendo un problema de optimización de mínimos cuadrados ponderados con el objetivo de minimizar el error del marcador. El error del marcador se refiere a la distancia entre el marcador experimental y el marcador del modelo virtual.

La dinámica inversa es el estudio del movimiento y las fuerzas y momentos que producen dicho movimiento. Para realizar este estudio se requieren la estimación de la masa y la inercia. El propósito de la dinámica inversa es estimar las fuerzas y los momentos que causan un movimiento en particular, y sus resultados pueden usarse para inferir la participación muscular en el movimiento evaluado.

2.7.2 Nuevas herramientas: simulación en línea

En un principio Opensim no contaba con herramientas para hacer simulaciones en tiempo real, básicamente su estructura no permitía realizar ese tipo de análisis, hace un par de años se publicó un proyecto donde a partir de la ayuda de programas de cómputo como Matlab o Python implementan una ruta para permitir el uso de análisis en tiempo real, básicamente los cálculos que realiza Opensim en su núcleo fuera de línea, se realizan en Matlab o Python de una forma rápida, siendo estas herramientas capaces de extraer los modelos virtuales de Opensim para su visualización.

2.7.2.1 Das3

Fue el primer proyecto que apareció para tratar de brindarle a Opensim la capacidad de trabajar en tiempo real. Este proyecto tiene como objetivo desarrollar un modelo músculo esquelético de miembro superior para la evaluación de los movimientos de hombro y codo. Se evalúan de igual forma la articulación de la cintura escapular y el contacto escapulo-torácico [59]. Ellos utilizan la función MEX de Matlab y usa Opensim para visualización y procesamiento.

2.7.2.1.1 Función Mex Matlab

Un archivo Mex es como un script o función de Matlab, esta función llama a un programa C/C++ o a una subrutina Fortran. Opensim puede realizar simulaciones dinámicas hacia adelante, sin

embargo, estas simulaciones se ejecutan demasiado lento para tiempo real. Una de las razones es que deben calcularse las trayectorias de los músculos en 3D, estas trayectorias tienen en consideración las figuras geométricas por donde pasan, lo que conlleva a cálculos que consumen mucho tiempo. Por lo tanto, en este proyecto DAS3 utilizan aproximación de funciones para codificar las trayectorias musculares.

2.7.2.2 Opensense

Opensense acaba de salir a la comunidad de SimTK, en noviembre del 2020 fue publicado y es una herramienta que se agregó a la versión 4.1 de Opensim, dicha versión trae incorporados nuevas herramientas capaces de leer sensores de movimiento y enlazarlos virtualmente en tiempo real a un modelo definido.

Este trabajo presume de ser 100% libre, hay incluso documentación para generar un dispositivo que permita generar la interface entre el hardware y el software.

2.7.3 Modelos de miembro superior de antebrazo, mano y dedos

En esencia Opensim tiene alrededor de 5 modelos de brazo con los cuales cada investigador puede modificarlos y crear sus propios modelos de miembro superior, para nuestros propósitos se decidió utilizar el que mejor se adaptaba a estos. Originalmente creado por Delp y Gonzalo para investigar el movimiento de la muñeca y el gasto que se visualizaba en procedimientos quirúrgicos. Este modelo consiste en todos los huesos del brazo y 10 grados de libertad. Pronación, supinación, flexión-extensión desviación ulnar y radial, incluyendo los grados de libertad para el codo, pulgar y dedo índice. Un total de 23 músculos actuadores del control del movimiento [87].

Capítulo 3 Experimentación

3.1 Introducción

En este capítulo revisaremos la experimentación realizada para alcanzar los objetivos establecidos, desde la definición de un buen método de adquisición para señales EMG, a partir de una investigación exhaustiva de referencias, desarrollo de un sistema de adquisición de señales EMG analógico, su participación junto a un microcontrolador para la digitalización de las señales y el envío de esas señales a través de un protocolo de comunicación inalámbrica como Bluetooth hasta mejorar, evaluar y validar las rutas conocidas de aprendizaje automático.

Se desarrollará una clasificación de movimientos en línea a través de un software como Labview, se evaluará un método para clasificar en tiempo real las mediciones de señales EMG en Matlab y se considerarán posibles rutas para su modelado en un programa de simulación músculo esquelética como Opensim.

3.2 Protocolo de adquisición EMG propuesto

La revisión desarrollada nos proporcionó los factores que afectan y son determinantes en la adquisición de la señal EMG, y como estos influyen en la calidad de la señal. Encontramos factores como el tamaño y tipo de material del electrodo, el tipo de adquisición, la posición de los electrodos, la distancia inter electrodo, el filtrado de la señal hasta el rango de frecuencias utilizado. Muchos de estos parámetros son incluidos por la SENIAM “Surface EMG for Non Invasive Assessment of Muscles”, un proyecto europeo como un esfuerzo para la estandarización de las adquisiciones EMG.

3.2.1. Factores determinantes que afectan la señal EMG

1. Tamaño del electrodo: La mayoría de las investigaciones evaluadas [9, 14, 30, 31, 33] junto con la SENIAM recomiendan un diámetro pequeño del electrodo, siendo en su mayoría iguales o menores a 10 mm.
2. Distancia inter-electrodo: Factor donde hay discrepancia entre las investigaciones consultadas, esto debido a su dependencia de otros factores como la región de adquisición, el nivel de

contracción. Se recomienda una propuesta de la SENIAM indicando una distancia de 20 mm, dicha distancia es utilizada por [3] en otras investigaciones mencionan que una distancia menor es mejor, ellos proponen 10 mm, utilizada también en [36].

2. Material del electrodo: La mayoría de las investigaciones involucradas [3,7,9,18,19,26,29-31,35] utilizan electrodos de Ag/AgCl, de igual forma encontramos una recomendación de la SENIAM que indica que un electrodo de este material es bueno para la adquisición de la señal.
3. Tipo de adquisición: Algunas investigaciones utilizan indiscriminadamente el tipo de adquisición, en [10] se estudia la diferencia entre tres tipos de adquisición: electrodo diferencial simple, diferencial doble y ramificado, evaluando la comunicación cruzada de las fuentes y determinan que hay una diferencia significativa entre las mediciones de la forma diferencial simple y de las otras dos. Siendo mejores estas últimas, razón por la cual recomendamos la adquisición diferencial doble.
4. Posición de los electrodos: Existen dos formas de adquisición en función de la posición de los electrodos: una cuando se utiliza un anillo circular en alguna región de antebrazo donde definen que la información completa para decodificar las señales EMG se encuentra ahí y otra donde se buscan los lugares óptimos dependiendo de los músculos que participaran en los movimientos evaluados. En función de la búsqueda de un método eficiente en la adquisición y de las investigaciones que respaldan este método sugerimos por la cantidad de trabajo respaldándolo, que el mejor método para la adquisición es la de los lugares óptimos, ya que el objetivo es utilizar esta información para alimentar un clasificador de movimientos, mientras las características tengan la información más discriminante es mejor. Encontramos el siguiente trabajo [61] que nos proporciona los músculos y la forma de acceder a ellos desde la superficie y donde colocar los electrodos. El seleccionar que músculo será considerado tendrá que ser definido por los movimientos que se consideraran.

La SENIAM establece posiciones de electrodos para algunos músculos, sin embargo, no establece regiones para miembro superior a nivel de antebrazo. Se utilizó la referencia [61] para establecer las posiciones de interés. Considerando solo movimientos de dedos seleccionamos 8 posiciones encontradas en la literatura consultada para 8 puntos de adquisición. De [22] tomamos los sitios 15-16, 32-33 y 54-55. De [3] se selecciona la posición 1 para el musculo flexor y para el extensor, de [32] se seleccionaron las posiciones para el ECU, ED, ECRB y FCR.

5. Filtrado: Se propone un filtro analógico de 20 a 500Hz con un filtro Butterworth de 4to orden.
6. Frecuencia de muestreo: El teorema de Nyquist establece que al menos se debe muestrear al doble de la frecuencia más alta que consideramos presente en la señal de interés, la frecuencia de muestreo debe ser de 1 kHz [34,36].

3.2.2 Consideraciones extras para una adquisición adecuada:

1. Preparación de la piel

Se necesita generar una buena unión electrodo piel por lo tanto se recomienda el uso de gel para disminuir la impedancia de la piel, pasos para la preparación:

1. Rasurado
2. Limpieza con alcohol
3. Aplicación de gel

2. Posición del paciente

Sentado, espalda erguida, con el brazo en diferentes posiciones:

1. A 45 grados con el antebrazo flexionado sobre el brazo.
2. A 90 grados con el brazo en el aire.
3. Con el brazo a 180 grados en reposo colgando del cuerpo.

3.3 Sistema de adquisición analógico para EMG

Para desarrollar un sistema de adquisición analógico se utilizó la herramienta “*AD8232 Filter design tool*”. Esta se utilizó para configurar los filtros analógicos incluidos en el diseño del AD8232, calcular las frecuencias de corte y evaluar las ganancias. Como parte de la evaluación del módulo creado, se procedió a hacer un análisis de ganancias, frecuencias de corte y componentes disponibles en el mercado. La herramienta te permite configurar una serie de filtros activos y pasivos para lograr el filtrado que se necesita para adquirir biopotenciales, en la Figura 3.1 se puede visualizar la interfaz gráfica de la herramienta. En este estudio seleccionamos en la primera etapa un filtro de tercer orden y en la segunda etapa un filtro pasa bajas tipo Sallen-Key.

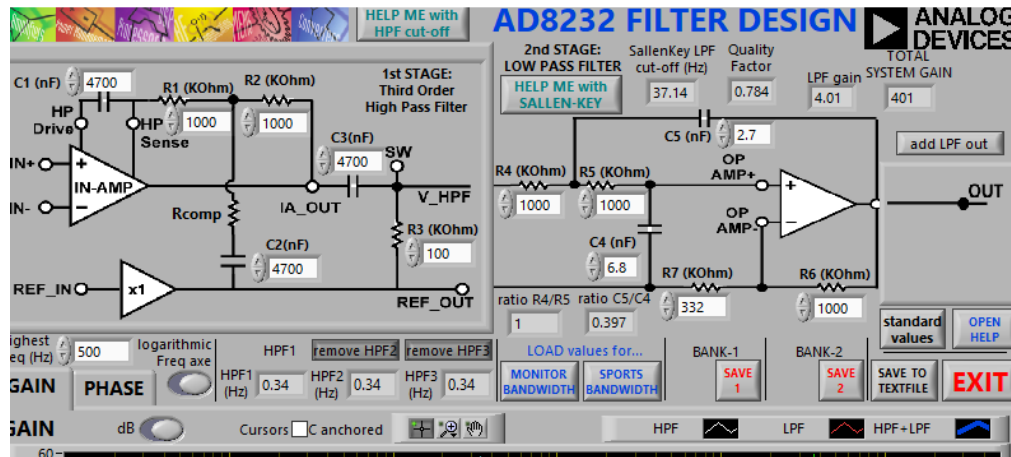


Figura 3.1 Interfaz gráfica de la herramienta AD8232 Filter Design.

Uno de los puntos críticos en esta etapa fue seleccionar la ganancia, ya que trabajamos con filtros activos donde la ganancia está directamente relacionada con las frecuencias de corte. El ancho de banda buscado fue de 20 a 500 Hz, sin embargo, para obtener un filtro pasa-bajas de 500 Hz con una ganancia de 500 y con componentes comerciales tuvimos que redefinir la frecuencia de corte de este filtro a 480 Hz.

Prototipos

Se diseñó el circuito con un software llamado DipTrace. Con el cual se crearon los archivos para fabricar las placas pcb y posteriormente soldar y probar el circuito.

Después de contar con los filtros diseñados en filter design tool, se procedió a generar el esquemático del circuito en Diptrace. Una vez contando con los componentes y el circuito definido en el esquemático se procedió a generar el diseño de la placa pcb. Se buscó contar con un diseño lo más compacto posible.

Una vez teniendo el esquemático y el diseño de la placa se mandaron a maquinar las placas con el diseño propuesto. Para esto se requirió generar los archivos necesarios para su maquinado.

Pruebas de funcionamiento

Una de las pruebas realizadas fue evaluar el funcionamiento del AFE a diferentes valores de voltaje de alimentación, dicha experimentación se realizó para evaluar el funcionamiento del circuito integrado a valores bajos de voltaje en caso de utilizar una batería como fuente de alimentación.

3.4 Electromiografía y su procesamiento con aprendizaje automático

Una vez con el sistema de adquisición analógico se procedió a desarrollar un estudio con aprendizaje automático para crear un clasificador de movimientos a partir de una base de datos ya estudiada en [70,88].

3.4.1 Pre procesamiento y procesamiento de la señal EMG

Como se describió en el capítulo 2 el análisis comienza con una base de datos descrita en la sección 2.2.4.1 los procedimientos utilizados son divididos en un pre procesamiento y un procesamiento de los datos. Dentro del pre procesamiento contamos con métodos como eliminación de valores atípicos, uso de un método para seleccionar características hacia adelante, normalización, uso de matrices de dispersión para transformar los datos a un nuevo espacio. Después de esta etapa se generó el procesamiento que incluye métodos de aprendizaje automático para crear modelos a partir de los datos analizados, las siguientes secciones describen dichos métodos, posteriormente se explican los métodos de validación utilizados.

3.4.1.1 Detección de valores atípicos multivariable utilizando la distancia Mahalanobis.

La idea central es convertir un problema de detección de valores atípicos multivariados en un problema de detección de valores atípicos de una variable.

Para un conjunto de datos sea \bar{o} el vector media. Para un objeto o , en la base de datos, la distancia mahalanobis de o a \bar{o} es:

$$MDist(o, \bar{o}) = (o - \bar{o})^T S^{-1} (o - \bar{o})$$

Donde S es la matriz de covarianza. En este punto $Mdist$ es una variable univariada

Pasos para determinar valores atípicos:

1. Calcular el vector de medias del conjunto multivariable de datos.
2. Para cada objeto o , calcular la distancia Mahalanobis de o a \bar{o} .
3. Se detectan los valores atípicos en este nuevo vector de distancias.
4. Se toma como métrica de una distribución chi cuadrada para encontrar el límite de los datos nombrados como normales.

La siguiente imagen muestra parte del código para calcular estos datos.

```

3  function [outliers,matrizlimpia,dm,elim,mant]=ji2(x)
4  [nmuest,var]= size(x);
5  media=mean(x);           %media de cada variable de los datos
6  s=cov(x);                %sentencia para obtener las covarianzas
7  V12(nmuest,var)=zeros;  %vector para las distancias
8  for i=1:var              %resta de media cada variable a cada dato
9  for j=1:nmuest
10     V12(j,i)=abs(x(j,i)-media(i));
11 end
12 end
13 dl=s\V12';              %covinv=inv(s); dl=covinv*V12'
14                          %covariana inversa por trasnpuesta de
15                          %la matriz sin la media de cada variable.
16 dm(nmuest,1)=zeros;     %vector para distancias Maalanobis
17 for i=1:nmuest          %ciclo para obtener multiplicacion diaonal
18                          %unicamente
19     dm(i,1)=V12(i,:)*dl(:,i);
20 end
21 chi= ncx2inv(0.975,var,0); %valor critico seun la distribucion ji^2
22 elim = find(dm > chi);    %vaores eliminados de mis datos
23 mant = find(dm <= chi);  %valores mantenidos en mis datos
24
25 matrizlimpia=x(mant,:); %matriz oriinal limpia de valores atipicos
26 outliers=x(elim,:);     %matriz con valores atipicos
27                          %para fines raficos
28 ord=sort(dm);           %se ordenan los datos de mayor a menor
29 valelim = find(ord > chi);
30 valmant= find(ord <= chi);
31 lonvalman=size(valmant);
32
33 end

```

Figura 3.2 Programa utilizado para seleccionar los valores atípicos de la base de datos.

3.4.1.2 Método de selección de atributos por SFS

Después de la remoción de los valores atípicos del conjunto total de los datos, se crearon ventanas de 250 datos donde cada ventana posee las 8 mediciones de los 8 electrodos, y a cada señal de cada ventana se les aplicaron los procesamientos vistos en el apartado 2.4.2 posteriormente el algoritmo SFS evaluó el rendimiento de cada uno de las características en cada uno de los electrodos considerados. En la Figura 3.3 y 3.4 podemos visualizar el pseudocódigo y el código utilizado para obtener las características más relevantes en la creación de los clasificadores considerados.

```

SUBCONJUNTO= 0; MEJOR_EVAL = 0;
ESPACIO=1...L
SALIDA= falso
WHILE NOT SALIDA
FOR i=1 A |ESPACIO|
TEMP(i)= J (SUBCONJUNTO v ESPACIO(i))
END
MEJOR_i = ARGMAX(TEMP)
IF TEMP(MEJOR_i)>MEJOR_EVAL
subconjunto = SUBCONJUNTO +
ESPACIO(MEJOR_i)
MEJOR_EVAL= TEMP(MEJOR_i)
ESPACIO = ESPACIO - ESPACIO(MEJOR_i)
ELSE
SALIDA = verdad;
END
END
RETURN SUBCONJUNTO, MEJOR_EVAL

```

Figura 3.3 Seudocódigo de SFS.

```

4  function [Qdam,mejor_eval,subconjunto]=seleccion_atributos_QDA(data)
5  -   class=data(:,97);
6  -   data(:,97)=[];
7  -   [~,var]=size(data);
8  -   mejor_eval=0;
9  -   espacio=(1:var);
10 -   EXIT=false;
11 -   cont=1;
12 -   while ~ EXIT                                % Repetir el ciclo hasta que
13 -       temp=[];                                % Exit sea true
14 -       temp(1:length(espacio))=zeros;
15 -       for i=1 :length(espacio)                % Ciclo para evaluar el error
16 -           if cont==1                          % del clasificador
17 -               [Qdam,QdaCVerrm]=QDA(data(:,i),class);
18 -           else
19 -               [Qdam,QdaCVerrm]=QDA([data(:,subconjunto),data(:,espacio(:,i))],class);
20 -           end
21 -           temp(i)=1-QdaCVerrm;                % Se guarda el porcentaje de rec
22 -       end                                      % de cada una de las características
23 -       [mejor_i,ind]=max(temp);                % evaluadas
24 -
25 -       if mejor_i>=mejor_eval
26 -           subconjunto(cont)=espacio(ind);
27 -           mejor_eval=mejor_i;
28 -
29 -           espacio(ind)=[];
30 -           cont=cont+1;
31 -       else
32 -           EXIT=true;
33 -       end

```

Figura 3.4 Algoritmo utilizado para la selección de características hacia adelante.

3.4.1.3 Normalización.

Si se transforma el conjunto de datos evaluado a un conjunto de datos normal, es decir, que posea una media cero y una desviación estándar 1, se ponen en un mismo rango de valores todas las mediciones de todas las características evaluadas, de esta forma todas tendrán el mismo impacto en cada una de las operaciones de nuestro clasificador. El procedimiento se realizó de forma general para los tres clasificadores.

3.3.1.4 Conversión al espacio de matrices dispersas

La conversión de nuestros datos a nuevos espacios, aunque aumenta el procesamiento puede dar buenos resultados, aumentando el reconocimiento del modelo clasificador y/o en la reducción de características. Este algoritmo se utiliza en primera instancia para aumentar las distancias entre las clases evaluadas y reducir las distancias entre los datos de las mismas clases, en [89] se muestra la descripción de este método, siendo la Figura 3.5 una simplificación de este trabajo. El método se realizó solamente con los datos que mejor resultados dieron es decir solo para el clasificador QDA, se tomó el grupo de características que aumentaban el rendimiento y estas fueron las que se transformaron al nuevo espacio, para posteriormente volver a aplicar un modelo QDA y calcular el porcentaje de reconocimiento.

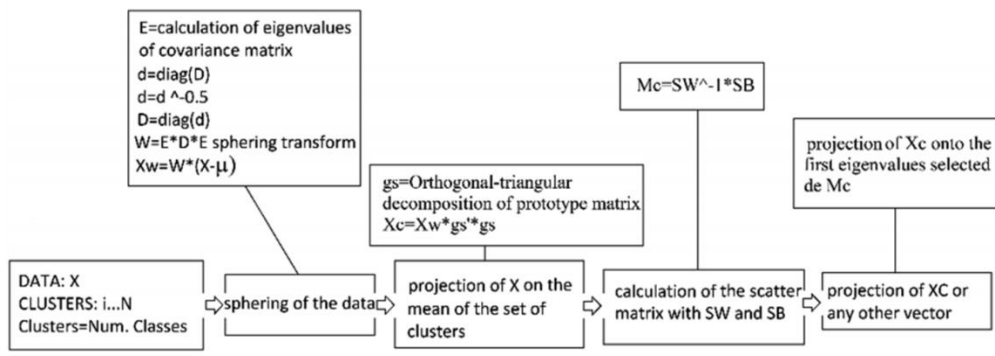


Figura 3.5 Método utilizado para cambiar los datos analizados a un nuevo espacio

3.4.2 Modelos de clasificación

Los modelos de clasificación considerados en este trabajo son Naive Bayes, LDA y QDA, todos con un enfoque Bayesiano. Un clasificador Bayesiano es un clasificador estadístico. Estos predicen la clase de los datos entrantes dada una probabilidad a pertenecer a cierta clase. La clasificación Bayesiana está basada en el teorema de Bayes.

En el teorema de Bayes propondríamos que X es un conjunto de dos datos. En términos Bayesianos X es considerado una evidencia. Sea H alguna hipótesis de que los datos X pertenecen a una clase específica. Para problemas de clasificación, se desea determinar $P(H|X)$, la probabilidad de que los datos X pertenezcan a la hipótesis H , en otras palabras, la probabilidad de que los datos X pertenezcan a la clase C , dado que conocemos la descripción de los datos X . $P(H|X)$ se denomina la probabilidad posterior o probabilidad a posteriori, en contraste, $P(H)$ es la probabilidad priori de H , la cual es independiente de X . Similarmente $P(X|H)$ es la probabilidad a posteriori de X condicionada sobre H . $P(X)$ es la probabilidad priori de X [90]. El teorema de Bayes indica que:

$$P(H|X) = \frac{P(X|H)P(H)}{P(X)}$$

El algoritmo para crear los modelos de clasificación utilizado fue la función “*fitcnb*” para Naive Bayes. La Figura 3.6 muestra un ejemplo del uso de esta función en Matlab.

```
function [Mdl,nbCvErr]=NB(data,class)
Mdl = fitcnb(data,class,...
'ClassNames',{'1','2','3','4','5','6','7','8','9','10','11','12','13','14','15'});
rng(0,'twister'); %crea el clasificador LDA con los datos de entrada
cp = cvpartition(class,'KFold',10); %Se crean las particiones de los datos
cvlda = crossval(Mdl,'CVPartition',cp); %las particiones creadas
nbCvErr = kfoldLoss(cvlda); %Se calcula el error de CV
end
```

Figura 3.6 Función utilizada para el modelo de clasificación a partir de Naive Bayes.

3.4.2.1 Funciones discriminantes: Análisis discriminante lineal y cuadrático con un enfoque Bayesiano

Una función discriminante es una función que toma un vector de entrada y se lo asigna a una determinada clase, denotada como C_k . La función desarrollada para crear un modelo LDA y QDA es la que se muestra en la Figura 3.7. La función “*fitcdiscr*” de Matlab se usó para crear LDA o QDA.

```
function [lda,ldaCvErr]=LDA(data,class)
lda = fitcdiscr(data,class,'DiscrimType','pseudoLinear'); %crea el clasificador LDA
rng(0,'twister'); % con los datos de entrada
cp = cvpartition(class,'KFold',10);%Se crean las particiones de los datos con 10 K folds
cvlda = crossval(lda,'CVPartition',cp); %Se evalua el clasificador lda
ldaCvErr = kfoldLoss(cvlda); %Se calcula el error de CV
end
```

Figura 3.7 Función utilizada para crear modelos de clasificación LDA y QDA

3.4.3 Rutas de preprocesamiento consideradas

La Figura 3.8 muestra la metodología de los experimentos realizados que involucran la secuencia de operaciones aplicadas a la base de datos. Dicha secuencia se realizó para determinar que ruta en el procesamiento de los datos era mejor.

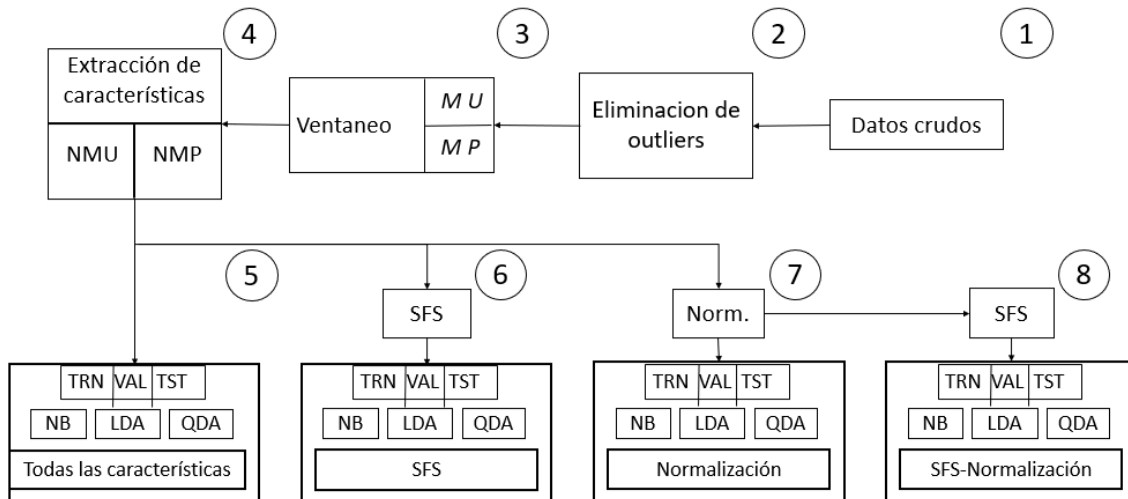


Figura 3.8 Secuencia de experimentación creada para evaluar el desempeño de los clasificadores a partir de una variación en el pre procesamiento.

3.4.4 Métodos de validación estadística de los resultados

3.4.4.1 Validación cruzada

Esta se realiza para cada uno de las matrices de los 8 sujetos y para la matriz con los datos de los 8 sujetos (NMU). Se crearon matrices para entrenamiento, validación y test. Se utilizó una función de Matlab llamada “*cvpartition*” que separa (particiona) los datos para una validación cruzada: Dicha función se utiliza para separar en 10 partes el conjunto original de los datos, a 9 de ellos se les asigna una etiqueta en nuestro caso TRN-VAL y a uno TST, de esta forma se tienen 10 partes de nuestros datos originales, ahora la función crea 10 conjuntos de datos recorriendo la etiqueta de TST a cada una de las partes del conjunto original (que había sido dividido en 10). Ahora Dicha función se utilizó dos veces, una para separar en 10 partes el conjunto original de los datos y creando dos bloques llamados TRN-VAL y TST del conjunto original y otra para separar los datos del bloque TRN-VAL en dos sub-bloques TRN y VAL. Esto se puede visualizar en la Figura 3.9 debajo de cada bloque *cvpartition* encontramos columnas de datos con 10 secciones las cuales se les asigna un nombre. Para el primer bloque tenemos que las columnas forman 2 bloques TRN-VAL (con 9 partes) y TST (con 1 parte), así en cada uno de las 10 columnas, esta es la forma en la que creamos 10 conjuntos de datos distintos con el mismo total de datos. En el segundo bloque

cvpartition encontrarnos otras 10 columnas debajo, las cuales son la forma en la que se dividen uno de los conjuntos de datos anteriores de TRN-VAL del primer bloque. Ahora uno de esos 10 conjuntos se divide en 10 subconjuntos de datos compuestos por 9 partes de TRN y 1 de VAL, así en cada una de las 10 columnas.

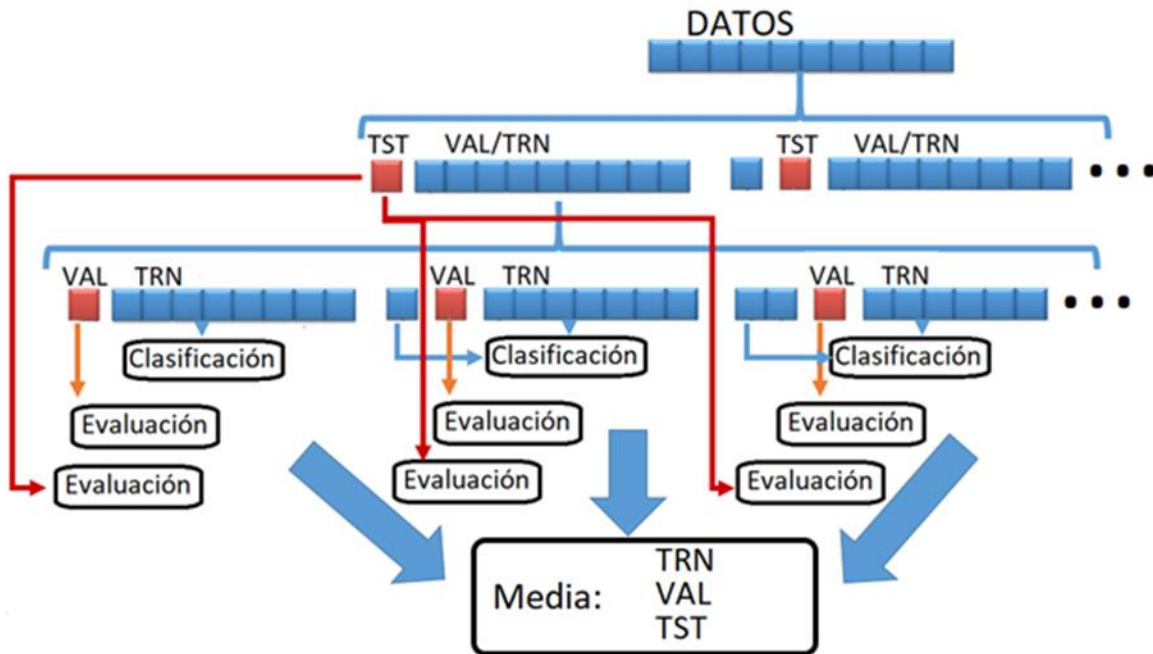


Figura 3.9 Metodología para la validación cruzada.

Hasta aquí tenemos 10 conjuntos de datos divididos en TRN-VAL y TST, de igual forma los 10 conjuntos de TRN-VAL ahora son divididos cada uno en 10 nuevos subconjuntos de datos en TRN y VAL. Es decir, tendremos 100 modelos de TRN validados con 100 conjuntos de VAL, cada uno de estos resultados se promediarán para calcular en porcentaje real de entrenamiento y de validación.

Al finalizar la metodología de la Figura 3.9 obtuvimos 100 modelos de clasificación donde se probó la primera partición del test (10 en total) de cada conjunto de datos en 10 subconjuntos de TRN y VAL. Esos resultados se promediaron para cada clasificador por separado y se obtuvieron los promedios mostrados más adelante.

3.4.4.2 Formulaciones estadísticas

A partir de aquí analizaremos estadísticamente cuál de los 4 procedimientos (tratamientos) se le debe dar al conjunto de datos de los 8 sujetos tanto individualmente como agrupados, uno de nuestros intereses principales es validar los resultados del grupo muestral ya que es en función a ella que se generaran los modelos de clasificación a implementar.

3.4.4.3 ANOVA

Dentro de las pruebas realizadas es necesario definir cuál o cuáles de los tratamientos producen resultados significativos diferentes y cuáles no. Son 8 sujetos de prueba seleccionados de una misma población. Se establecieron 4 tratamientos de los datos: I.-Sin procesamiento, II.-con SFS, III.- Normalización, IV.- Normalización y SFS.

Se utilizó el análisis de varianzas (Anova) para comparar y estudiar el efecto de los métodos aplicados a nuestros datos sobre las medias de los porcentajes de reconocimiento obtenidos a partir del mejor clasificador evaluado. La prueba realizada fue para determinar si había diferencias significativas en los resultados (porcentaje de reconocimiento) de los 8 sujetos a partir de un modelo de clasificación definido como el mejor para cada uno de los tratamientos aplicados.

3.4.4.4 Método de Duncan

Este método es un test de comparaciones múltiples. Nos permite comparar las medias de los tratamientos después de haber rechazado la hipótesis nula. Se aplica el método de Duncan para determinar cuáles de los tratamientos son iguales estadísticamente o cual es diferente.

En este método se hace una comparación entre todas las medias, si las k muestras son de igual tamaño, los k promedios se acomodan en orden ascendente y el error estándar de los promedios se estima con $S_{\bar{y}} = \sqrt{CM_E/n}$. De la tabla de rangos significantes de Duncan se obtienen los valores críticos que son los valores con los que realizaremos las comparaciones. En la tabla de rangos significantes de Duncan se establecen dos parámetros $r_\alpha(p, l)$ donde α es el nivel de significancia prefijado, y l son los grados de libertad para el error.

Con estos k-1 valores se obtienen los rangos de significancia mínima dados por:

$$R_p = r_\alpha(p, l)S_{\bar{y}}; p = 2, 3, \dots, k$$

Se realiza una comparación entre las diferencias de las medias muestrales con los rangos de significancia calculados.

Cuando se establece que la diferencia no es mayor que el rango que le corresponde se concluye que esas medias son iguales cuando la diferencia es mayor se dice que las medias son significativamente distintas.

3.5 Simulación virtual del movimiento Opensim

Se modificó y se compartió un modelo virtual de la muñeca que se encuentra de forma libre en [87]. La modificación del modelo fue reportada en un trabajo anterior [91, 92]. Dicho modelo modificado posee movimiento en todos los dedos de la mano para poder reproducir cualquier movimiento seleccionado. Se puede visualizar este modelo en la Figura 3.10.

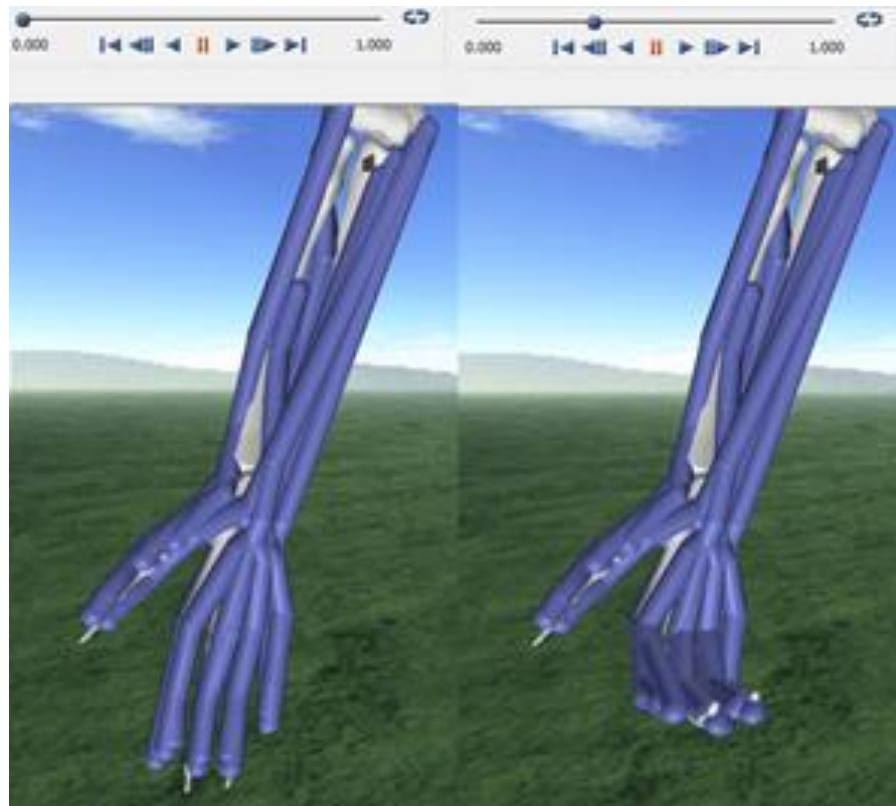


Figura 3.10 Modelo modificado de Opensim para reproducir movimientos evaluados.

Cada 62.5 ms (la duración en tiempo de la ventana seleccionada), se genera una salida en nuestro modelo de clasificación generando equivalente a un cuadro de avance para nuestro modelo virtual en Opensim. Se crearon vectores de movimiento capaces de reproducir cada uno de los movimientos de nuestra base de datos. El vector de avance seleccionado posee cada una de las articulaciones involucradas del movimiento seleccionado y el avance proporcional a 62.5 ms de un movimiento con una duración total de 5 s. Los archivos de movimiento y videos de

movimientos pueden ser encontrados en [87]. En la Figura 3.11 podemos visualizar el ciclo completo desde la selección del movimiento con el clasificador, la selección del vector de movimiento y la suma de dicho vector a la matriz de movimiento, posteriormente ya que el movimiento termina se genera el archivo ‘. mot’ para su posterior simulación en Opensim.

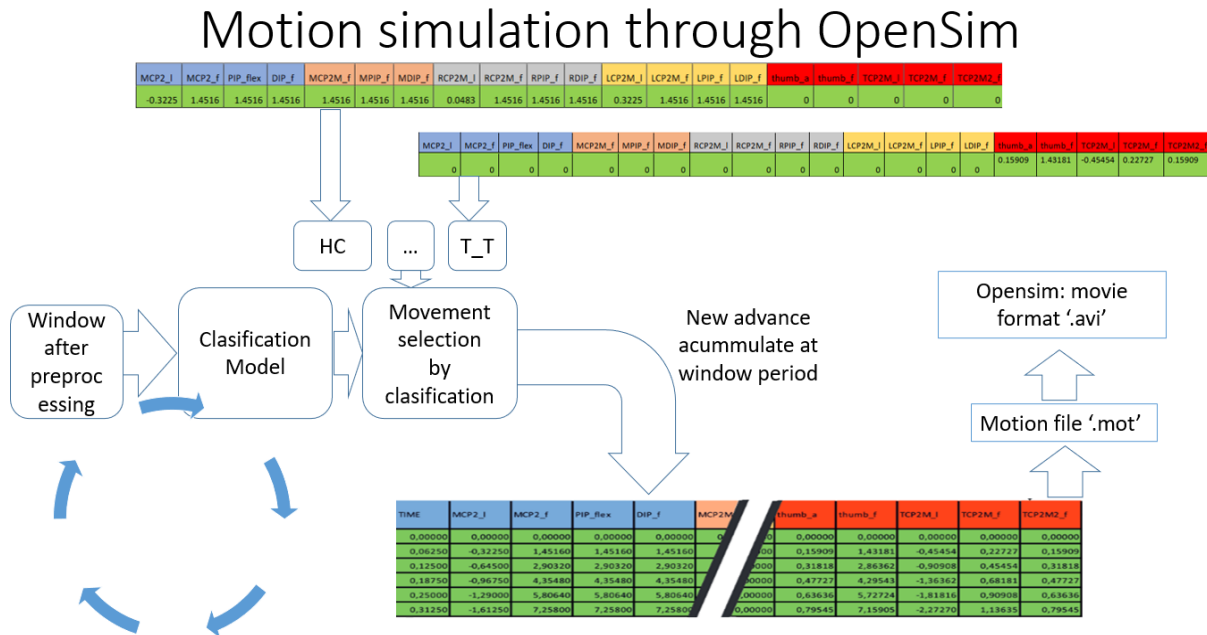


Figura 3.11 Ciclo desde la clasificación hasta la generación del archivo de movimiento para Opensim.

Con los archivos de movimientos generados pudimos realizar análisis biomecánicos de movimiento en Opensim, verificar las activaciones y participaciones musculares dentro de los movimientos.

3.5.1 Análisis del movimiento del peor al mejor movimiento clasificado y su simulación en Opensim

Cuando evaluamos movimientos nos referimos a movimientos estos deben ser estables y suaves en su ejecución. Consideramos que los 15 movimientos de la base de datos son de este tipo, cada uno representa un ciclo completo que fue registrado por medio de señales sEMG y descrito por fotografías.

Después de contar con un clasificador simple, rápido y eficiente como el creado a partir del análisis discriminante cuadrático se procedió a crear la reproducción del movimiento de forma virtual a través de Opensim. El primer trabajo consistió en reproducir los movimientos en Opensim de

forma ideal para eso se tomaron como referencia la posición inicial y final de los movimientos evaluados en la base de datos, que encontramos en [67], la Figura 3.12 detalla la idea general.

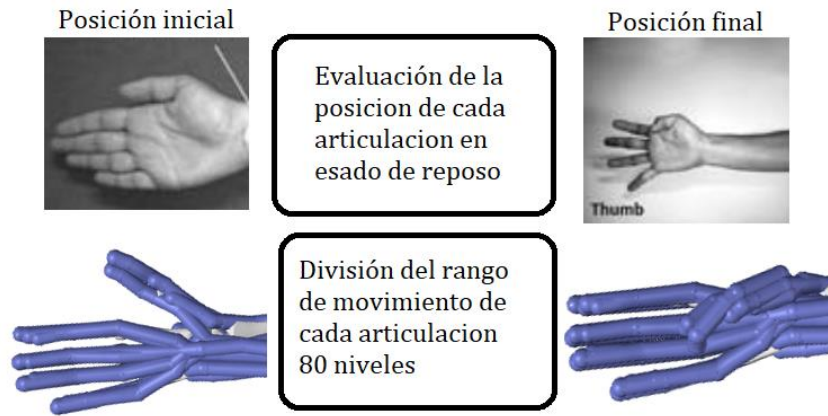


Figura 3.12 Creación del rango de avance para cada movimiento evaluado.

Teniendo en cuenta que el movimiento es realizado durante 5 segundos, que la cantidad de decisiones posibles del clasificador fue de 80 veces dentro de este tiempo y que el rango de movimiento de cada una de las articulaciones involucradas en el desarrollo de cada uno de los movimientos evaluados es fijo. Se procedió a calcular el grado de movimiento de cada articulación ante el movimiento considerado, así se formaron vectores de avance para cada movimiento. De esta forma se crearon 15 archivos de movimiento con extensión mot. Una vez contado con los archivos mot para generar los movimientos ideales con una duración de 5 segundos, se procedió a realizar la simulación ideal de los 15 movimientos.

A continuación, se evaluó el movimiento ante errores del clasificador para eso se decidió utilizar los datos de prueba del modelo seleccionado. Se buscó encontrar los movimientos en los que el clasificador era más propenso a equivocarse para de esta forma tener una visualización del movimiento peor clasificado y su reproducción a través del tiempo.

La matriz de test (datos de prueba) para el clasificador evaluado está compuesta de datos aleatorios de los 8 sujetos evaluados (una fracción de ellos). Nosotros podemos separar los datos de prueba para verificar cuál de los sujetos muestra una peor clasificación. Esto para encontrar el peor movimiento evaluado.

Finalmente se tomaron los datos de una repetición-sujeto completa del peor y mejor movimiento según nuestros datos de clasificación a partir del conjunto de prueba de la base de datos. Se probó el clasificador creado con la repetición seleccionada y se simuló el movimiento.

3.6 Microcontroladores y simulación en línea

La selección del microcontrolador se realizó en función de las características seleccionadas siendo el seleccionado el modelo TM4C123G6HPM. La tarjeta se muestra en la Figura 3.13.

3.6.1 TM4C123G6HPM

En esta tarjeta de desarrollo se consideraron varios elementos, la función de su ADC, su resolución, la velocidad de transferencia de datos de este periférico, la velocidad de transferencia de los datos vía puerto UART, el uso de su memoria para transferir datos a varias velocidades. De igual forma se incluirán los registros necesarios para configurar o si es el caso las funciones en C que nos permiten controlar los elementos a evaluar. De igual forma se consideraron los ambientes de desarrollo que pueden ser utilizados para programar.

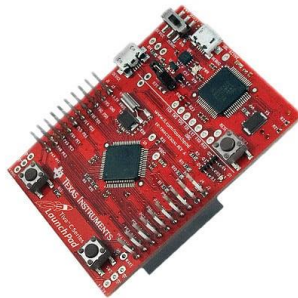


Figura 3.13 Placa de desarrollo utilizada para la experimentación.

3.6.2 Pruebas del sistema de adquisición

Actualmente la mayoría de las placas de desarrollo incluyen ejemplos para realizar una adquisición de datos analógicos utilizando el ADC del microcontrolador. Este microcontrolador puede ser utilizado para realizar adquisición de datos. Teniendo un sistema de adquisición y la forma de guardar los registros de los datos convertidos a digitales se procedió a generar una evaluación de nuestro sistema analógico. Se modificó un programa de las librerías que trae el microcontrolador para adquirir señales analógicas y enviarlas vía puerto serial a un puerto USB en la computadora

que funciona como un puerto virtual serial, que puede ser leído con un terminal serial como Putty. Estos datos se guardaron y posteriormente se graficaron.

Las actividades a realizar son el análisis de la transferencia de los datos, evaluando las instrucciones de lectura y escritura, el almacenamiento de los datos en la memoria flash, lectura de dichos datos y envío de estos al puerto UART seleccionado, para esto es necesario:

- Selección y configuración del puerto UART.
- Conexión física del módulo Bluetooth al puerto UART seleccionado.
- Configuración del módulo Bluetooth.
- Creación de un puerto serial virtual a través de una conexión Bluetooth.
- Uso de una terminal serial.
- Creación de un código para visualizar y almacenar datos adquiridos en tiempo real en Python.

3.6.2.1 Transferencia de datos: Evaluando la adquisición y transferencia de los datos del ADC a los registros que controlan los datos de entrada de este periférico.

Una de las primeras actividades en esta sección fue la evaluación del funcionamiento del ADC del microcontrolador y del puerto UART a utilizar, para una posterior inclusión del módulo Bluetooth a este último.

Las pruebas anteriores son sobre un solo microcontrolador al tener evaluadas todas las características de interés se procede al estudio de la simultaneidad. Como uno de los objetivos es trabajar con un arreglo de electrodos, es necesario trabajar simultáneamente con varios sistemas analógicos junto a microcontroladores enviando señales independientes hacia un módulo Bluetooth maestro. Primero se evaluará un microcontrolador, posteriormente 2, así sucesivamente hasta tener un número máximo de dispositivos cumpliendo con los requerimientos establecidos.

Se plantean varios experimentos para verificar el funcionamiento correcto del microcontrolador desde la función del ADC, la ruta que siguen los datos desde la salida del ADC a través de un puerto serial hacia algún otro periférico o hacia un puerto digital para su salida del microcontrolador. Se enumera la experimentación realizada para la visualización del correcto funcionamiento y procesado de los datos desde el canal analógico hasta la salida del microcontrolador a un puerto digital ya sea a un dispositivo de lectura o hacia un módulo Bluetooth para la importación de los datos a un equipo de cómputo.

1. Se guarda una señal senoidal dentro de la memoria del microcontrolador, dichos datos se envían continuamente al puerto serial UART. Se visualizan con el Analog Discovery 2.

El actual experimento es para analizar el funcionamiento del ADC y la velocidad del puerto serial. Se sabe que el funcionamiento del ADC es más complejo que una simple instrucción del controlador como lo es la lectura de los datos de la memoria. En esta etapa de la experimentación se evaluó únicamente lectura de los datos en la memoria flash y la velocidad del puerto serial.

2. Se adquirieren los datos de una señal senoidal a través de un canal analógico, poniendo en funcionamiento un módulo ADC y se envían estos a un puerto UART, con un equipo externo (Analog Discovery 2) se visualizarán los datos.

Se utiliza el Analog Discovery para visualizar los datos de salida, se puede analizar la cantidad de datos enviados y su relación con el baudrate que es la velocidad de bits por segundo capaz de transferir el puerto serial.

Se decidió evaluar la información que el microcontrolador envía, su velocidad y la cantidad de información enviada por el puerto UART. El procedimiento fue el siguiente:

1. Se evalúa la información que es capaz de enviar el microcontrolador a diferentes velocidades. Se evalúa la velocidad (el baudrate), que era un factor que limita el envío de los datos.
2. Con el analizador lógico del Analog Discovery 2 se observa el formato de la información enviada.
3. Establecimos un baudrate lo suficientemente alto para cubrir nuestras necesidades.

La Figura 3.14 muestra el diagrama general de implementación del arreglo probado.

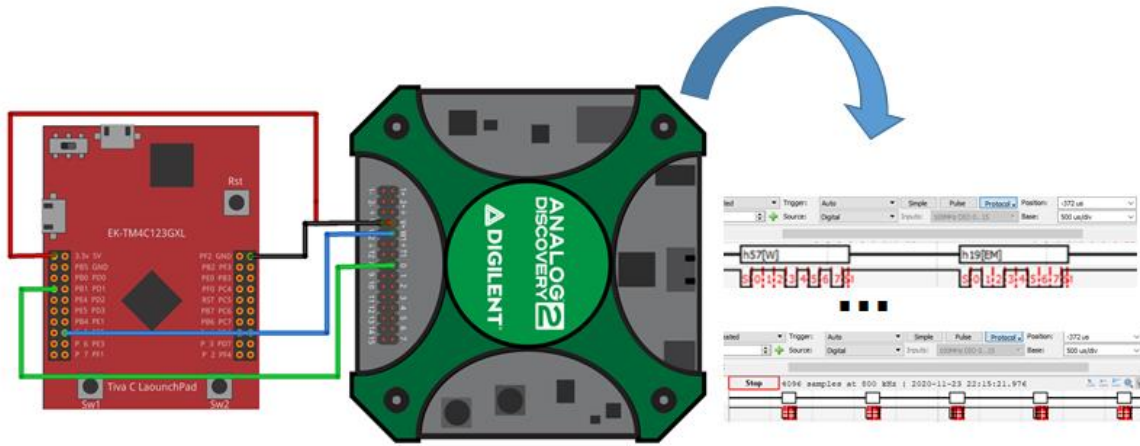


Figura 3.14 Conexiones realizadas para introducir una señal senoidal al microcontrolador y al mismo tiempo leer los datos a través del puerto UART.

3.6.3 Transferencia serial de datos desde el microcontrolador hasta el programa de adquisición en el equipo concentrador

Para establecer una ruta de transferencia de datos desde el ADC hasta el puerto serial de una computadora (que servirá como módulo concentrador para varios dispositivos conectados simultáneamente) se decidió evaluar la transmisión inalámbrica con un módulo Bluetooth HC06/05. El microcontrolador cuenta con varios puertos UART que pueden ser configurados para trabajar a diferentes velocidades. El módulo Bluetooth utilizado también puede ser configurado a trabajar a varias velocidades de transferencia de datos, por lo que fue necesario sincronizar ambas velocidades. En este punto sabemos que el módulo Bluetooth es un limitante de la velocidad de transferencia con una máxima de 1,382,400 baudios, velocidad que también está limitada por el puerto serial del concentrador. Para configurar la velocidad del módulo Bluetooth fue necesario utilizar dispositivos extras, como Arduino UNO y/o un cable serial USB de la computadora al módulo utilizado. El diagrama expuesto en la Figura 3.15 expresa la conexión realizada junto a las velocidades utilizadas en cada parte de la conexión.

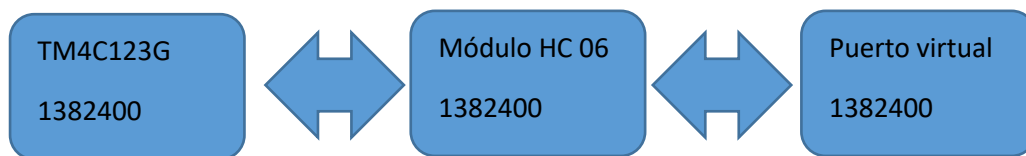


Figura 3.15 Velocidades de los 3 elementos que participan en una transferencia de datos en línea.

3.6.3.1 Configuración del sistema inalámbrico: Módulo Bluetooth HC06 modelo ZS-040 versión linvosVI.8

Los módulos Bluetooth poseen una configuración interna predeterminada, dicha configuración puede modificarse. Muchas de las veces trabajamos con los valores predeterminados, que fueron programados de fábrica, como lo son: parámetros por default, nombre del dispositivo (HC06), velocidad de transferencia (9600 baudios), clave de acceso para poder enlazarse (1234), entre otros.

3.6.3.1.1 Configuración del módulo

La reconfiguración del módulo HC06 se puede realizar (programar) por dos vías: vía serial con un adaptador USB a serial TTL como el PL2303 con cualquier software que cree puertos seriales en la computadora (PuTTY o Tera Term) Figura 3.16 ó con Arduino donde este sirve como comunicador serial entre el USB y el módulo Bluetooth.

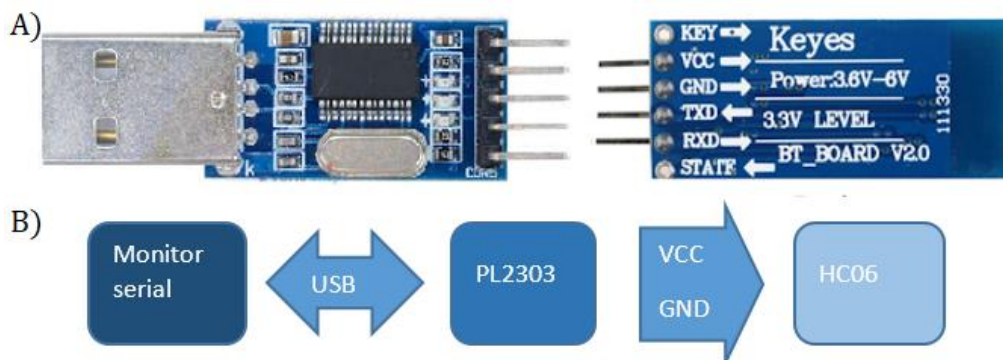


Figura 3.16 Convertidor USB a serial TTL. A) módulo utilizado PL2303, B) conexión realizada para configurar el módulo Bluetooth.

3.6.3.1.1.1 Arduino UNO

Para Arduino UNO la velocidad máxima configurable es de 115200 Baudios. Es posible configurar una velocidad mayor a 115200 pero con el inconveniente de que el monitor serial de Arduino perderá comunicación con el dispositivo.

3.6.3.1.1.2 PL2303

Para aumentar la velocidad de transferencia se utilizó un convertidor USB a serial TTL (PL2303) y se logró recalibrar a cada una de las velocidades permitidas. La Hoja de datos del módulo HC06 indica que puede configurarse hasta 1,382,400 bps. Así se fueron probando cada una de las velocidades hacia adelante hasta llegar a 1,382,400. Comprobando su funcionamiento con el microcontrolador TM4C123G.

El configurar una velocidad de 1,382,400 con un convertidor USB a serial TTL, conlleva una serie de problemas, la velocidad en Baudios es demasiada alta para volver a entrar a la configuración del módulo vía serial, es decir si configuramos dicha velocidad ya no podremos cambiar la velocidad del módulo, ni el nombre del dispositivo, ni la clave de acceso. El módulo es funcional, con el inconveniente de que se deben realizar otros métodos para reconfigurarlo.

La única diferencia es que Arduino solo puede configurar velocidades de hasta 115,200 baudios. Las referencias (Hoja de datos) indican que con un convertidor serial puede configurarse hasta 1,382,400 baudios. La velocidad de transferencia máxima fue comprobada por el método con un convertidor USB a serial (PL2303), comprobando su funcionamiento con el microcontrolador TM4C123G.

Cuando llegamos a la velocidad de 1,382,400 el puerto serie del equipo de Windows utilizado no pudo llevar el ritmo al reloj del Bluetooth, es decir, los comandos enviados para regresar el módulo Bluetooth a una velocidad menor ya no los aceptada, si enviaba y recibía datos a esa velocidad pero no recibía los comandos para reconfigurarse, por lo tanto, se buscó una forma de regresar a una velocidad de 115200 baudios, una velocidad aún modificable para el puerto serie y para el convertidor USB a serial. Se logró cambiar el módulo a 115200 baudios con un método encontrado en [93]. Básicamente el método utiliza el reloj del módulo Arduino UNO y lenguaje ensamblador para enviar específicamente el código necesario para configurar una velocidad de 115200 baudios, utilizando la frecuencia de reloj de 16 MHz bajándolo a la velocidad de 1382400 baudios bits por segundo, a la cual estaba corriendo el módulo. Se calculó cada uno de los bytes necesarios para enviar dicho código a esa velocidad.

El código para establecer la velocidad del módulo Bluetooth en Arduino uno fue el mostrado en la siguiente imagen.

```
#include <SoftwareSerial.h>
SoftwareSerial btSerial(2, 3); // RX, TX
/* Connect pin 2 Arduino to pin TX HC-06
 * Connect pin 3 Arduino to pin RX HC-06
 */
void setup() {
  Serial.begin(115200);
  Serial.println("Enter AT commands:");
  btSerial.begin(115200);
}
void loop()
{
  if (btSerial.available())
  Serial.write(btSerial.read());
  if (Serial.available())
  btSerial.write(Serial.read());
}
```

Figura 3.17 Código utilizado para configurar el módulo Bluetooth en Arduino.

3.6.3.2 Transferencia de datos Bluetooth-Bluetooth

Con el módulo HC06 trabajando a una velocidad de 1,382,400 bps con el microcontrolador TM4C123G y con un terminal serial como PuTTY se realiza una transferencia de datos en tiempo real. Se desarrolló un programa para transferir datos a esa velocidad en tiempo real en Code Composer Studio (CCS). Para poder generar el envío de datos de forma serial se configura la velocidad del puerto UART del microcontrolador, del módulo Bluetooth y de la terminal virtual en 1,382,400 ver Figura 3.18.



Figura 3.18 Transmisión inalámbrica de datos a partir de un microcontrolador con una señal senoidal almacenada en su memoria flash.

Se creó un arreglo con números constantes de 32 bits con 120 datos hexadecimales, datos que forman una señal senoidal como se muestra en la Figura 3.19.

```

#define PERIOD 120
static const uint32_t D_Val[PERIOD] = { 0x7ff, 0x86a, 0x8d5, 0x93f, 0x9a9, 0xa11, 0xa78, 0xadd, 0xb40, 0xba1,
0xbff, 0xc5a, 0xcb2, 0xd08, 0xd59, 0xda7, 0xdf1, 0xe36, 0xe77, 0xeb4,
0xeec, 0xf1f, 0xf4d, 0xf77, 0xf9a, 0xfb9, 0xfd2, 0xfe5, 0xff3, 0xffc,
0xfff, 0xffc, 0xff3, 0xfe5, 0xfd2, 0xfb9, 0xf9a, 0xf77, 0xf4d, 0xf1f,
0xeec, 0xeb4, 0xe77, 0xe36, 0xdf1, 0xda7, 0xd59, 0xd08, 0xcb2, 0xc5a,
0xbff, 0xba1, 0xb40, 0xadd, 0xa78, 0xa11, 0x9a9, 0x93f, 0x8d5, 0x86a,
0x7ff, 0x794, 0x729, 0x6bf, 0x655, 0x5ed, 0x586, 0x521, 0x4be, 0x45d,
0x3ff, 0x3a4, 0x34c, 0x2f6, 0x2a5, 0x257, 0x20d, 0x1c8, 0x187, 0x14a,
0x112, 0xdf, 0xb1, 0x87, 0x64, 0x45, 0x2c, 0x19, 0xb, 0x2,
0x0, 0x2, 0xb, 0x19, 0x2c, 0x45, 0x64, 0x87, 0xb1, 0xdf,
0x112, 0x14a, 0x187, 0x1c8, 0x20d, 0x257, 0x2a5, 0x2f6, 0x34c, 0x3a4,
0x3ff, 0x45d, 0x4be, 0x521, 0x586, 0x5ed, 0x655, 0x6bf, 0x729, 0x794
};

```

Figura 3.19 Datos almacenados en la memoria del microcontrolador que conforman un registro de 120 datos que definen un ciclo de una señal senoidal.

Se crea un ciclo para enviar cada uno de los datos al puerto serial, al mismo tiempo se crea un ciclo para generar un total de 10,000 periodos de la señal a enviar ver Figura 3.20.

```

int
main(void)
{

    InitConsole();
    int i=0;
    int d=0;
    int TIEMPO=100000;

    SysCtlClockSet(SYSCTL_SYSDIV_2_5 | SYSCTL_USE_PLL | SYSCTL_OSC_MAIN |
        SYSCTL_XTAL_16MHZ);

    // Display the setup on the console.
    //
    while(d<TIEMPO)
    {
        while(i<PERIOD)
        {
            UARTprintf("%4d\n", D_Val[i]);
            SysCtlDelay(11500);
            i++;
        }

        i=0;
        d++;
    }
}

```

Figura 3.20 Programa para enviar datos con el microcontrolador a un puerto serie.

El diagrama de flujo de dicho código es el mostrado en la Figura 3.21.

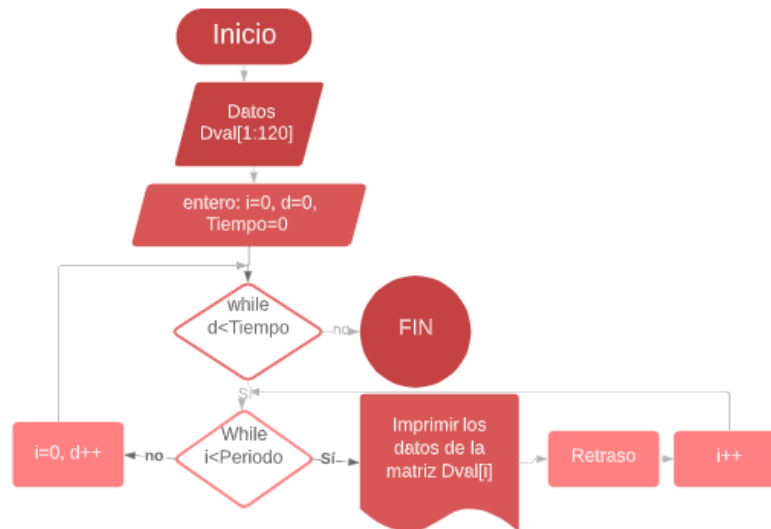


Figura 3.21 Diagrama de flujo, programa para transferencia de datos en el microcontrolador TM4C123G.

De esta forma tenemos un método para enviar datos al módulo Bluetooth. En la siguiente sección definiremos un método para recibirlos en el equipo concentrador, visualizarlos y evaluar el efecto de la velocidad sobre la señal transmitida.

3.6.3.2.1 Envío de datos a través del puerto serial UART a un módulo Bluetooth, generación de una conexión serial virtual en la computadora.

Cuando los datos de un puerto serial llegan al equipo de cómputo a través de un puerto serial virtual estos llegan a una memoria dedicada para almacenar los datos de entrada, posteriormente esos datos son enviados a la memoria del sistema, creando un cuello de botella si muchos datos llegan en un instante de tiempo.

La configuración del módulo Bluetooth maestro es administrada por el sistema operativo, en nuestro caso decidimos utilizar Windows, también se consideró utilizar otros sistemas basados en Unix, lo cual no es reportado.

3.6.3.2.1.1 Terminales virtuales

Una herramienta útil en la visualización de los datos que se obtienen en la adquisición a través de los puertos seriales son las terminales virtuales como PuTTY, esta herramienta nos permite visualizar en pantalla y guardar en archivos de texto todo dato que llega a los puertos USB de la portátil. La terminal que se utilizó fue PuTTY visible en la Figura 3.22.

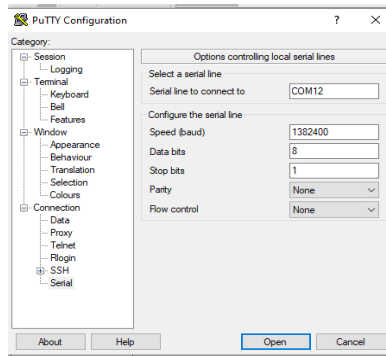


Figura 3.22 Terminal Virtual PuTTY.

Como se mencionó anteriormente el módulo Bluetooth de un equipo de cómputo es capaz de crear puertos seriales virtuales, para establecer rutas de entrada de datos, estos puertos seriales pueden conectarse a un terminal serial.

El microcontrolador utilizado utiliza registros de 32 bits para almacenar los datos, aunque la transferencia se realiza en bytes. Para poder configurar el envío de datos de forma serial se configuró la velocidad del puerto UART del microcontrolador en 1382400, se configuró el módulo Bluetooth con esa velocidad y se configuró la terminal virtual de la PC para trabajar también a esa velocidad.

3.6.3.2.1.2 Programación en Python

En la Figura 3.23 y 3.24 podemos observar el diagrama de flujo y código del programa desarrollado en Python que se utilizó para leer el puerto serial y crear un archivo de texto con los valores leídos, escribe el dato adquirido a un archivo, y repite el ciclo anterior. Una vez que cuenta 20,000 datos imprime el tiempo que se llevó solo en la adquisición y la escritura de los datos.

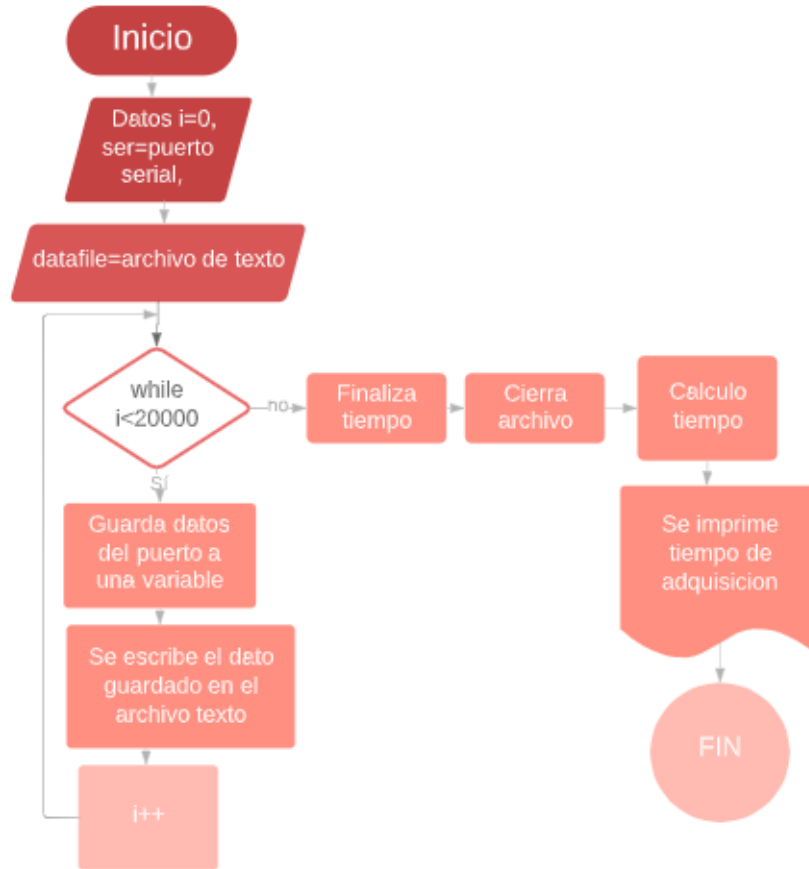


Figura 3.23 Diagrama de flujo para el programa creado en Python.

```

import serial
import time

ser= serial.Serial('COM12', baudrate = 1382400, timeout=1)
time.sleep(3)
dataFile = open('dataFile.txt','w')
i=1
inicia=time.time()
while 1:
    while i<20000:
        TIData = ser.readline().decode('ascii')
        dataFile.write(TIData + ' ')
        #print(TIData)
        i=i+1

    final=time.time()
    dataFile.close()
    tiempo=round(final-inicia,3)
    print("tiempo de: " +str(tiempo) + "s")

    break
  
```

Figura 3.24 Programa en Python para la adquisición de los datos vía serial.

Con el archivo texto creado se procedió a crear un archivo ‘.xls’ (formato de hoja de cálculo) y después se importó al ambiente de programación de Matlab para graficar los datos con la función “plot”.

3.6.4 Transferencia de datos de forma simultanea

Una vez con una transferencia correcta de datos desde el microcontrolador hasta un archivo de texto a una máxima velocidad de transferencia se procedió a aumentar el número de módulos trabajando simultáneamente. Lo que conlleva una serie de retos a resolver, desde contar con una velocidad aceptable para la adquisición, asegurar la integridad de los datos, coordinación entre las lecturas, considerar el exceso de trabajo del módulo concentrador Bluetooth, inicio/fin de la adquisición y una adquisición adecuada para su posterior procesamiento.

1. Se visualizaron los datos en una terminal virtual como putty

Una de las variantes para verificar los datos de entrada fue utilizar una terminal virtual, de esta forma los datos pueden visualizarse de una forma rápida y sin necesidad de entrar a un lenguaje de programación para hacer la lectura de algún puerto serial.

2. Se visualizaron los datos en Python

La visualización es más rápida que el guardado de los datos en Python, por lo tanto, el visualizar los datos nos permitió verificar una transferencia exitosa desde la memoria del microcontrolador/ADC hasta la consola virtual del equipo de cómputo, posteriormente nos permitió crear múltiples puertos seriales y conectarlos simultáneamente a múltiples microcontroladores. Esta etapa fue crítica al momento de realizar una transferencia simultanea de datos a través de múltiples microcontroladores.

3. Se guardaron en un archivo en Python

El almacén de datos era necesario evaluarlo porque en ciertas circunstancias es de vital importancia guardar los registros que llegan a la memoria del equipo de cómputo.

3.6.4.1 Evaluación de la cantidad de datos y velocidad máxima de transmisión que puede realizarse vía Bluetooth

Se realizó la misma configuración para establecer la transmisión simultánea de 2, 3, 4, 5, 6 y 7 módulos.

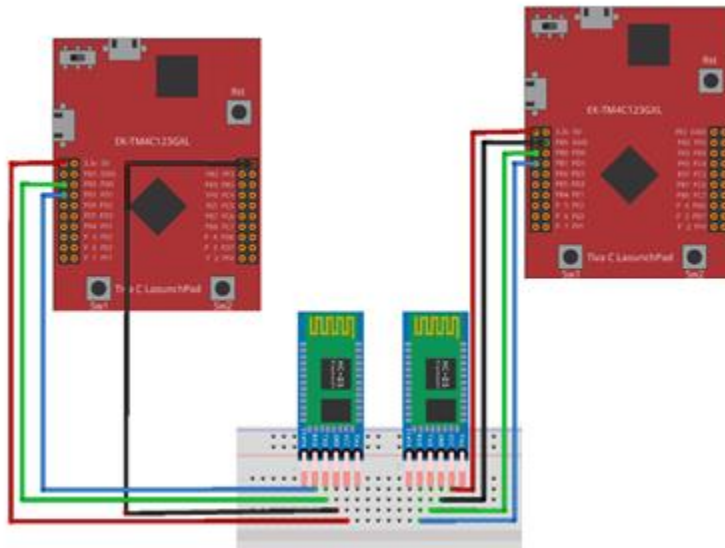


Figura 3.25 Conexión de dos microcontroladores con su respectivo módulo Bluetooth para el envío simultáneo de datos a un concentrador.

Se configuraron los módulos para que transmitieran a la velocidad mínima según nuestros requerimientos, que fue de 1000 Hz. El ADC posee una resolución de 12 bits envía la información al FIFO el cual es un banco de datos de 16 registros de 12 bits, después envía los datos al puerto serial UART configurado.

Formato de envío de datos

Una vez los datos en los registros del microcontrolador pasan a el puerto UART configurado, los datos se seccionan en bytes de información, con la función UARTPrintf se hace el envío al módulo HC06/05 conectado a los pines establecidos. Los datos del ADC pasan en formato de bytes por lo tanto tenemos dos registros uno con 8 bits y otro con 4.

3.6.4.2 Envío de datos a modulo maestro PC.

Los datos son adquiridos con el módulo Bluetooth de la portátil quien se encarga de crear puertos virtuales seriales a donde ingresan los datos de los módulos HC 06. Con un programa para leer los puertos seriales virtuales se hace la adquisición de los datos.

1. Se visualizaron y guardaron en una terminal serial como Putty.

Algunas terminales como Putty permiten guardar los datos que se visualizan, son una opción de rápida implementación. También permite crear varias conexiones al mismo tiempo con puertos

virtuales independientes, lo que nos permitió conectar simultáneamente módulos al concentrador sin la necesidad de programar. En esta etapa se evaluó la adquisición del ADC, se configuró a la velocidad establecida y se controló la velocidad del funcionamiento del ADC con un retraso (delay).

2. Se modifica el programa en Python para conectar los dos puertos seriales creados por Bluetooth para guardar los datos en dos registros independientes.

Se realiza este mismo procedimiento para 3, 4, 5, 6, 7 y 8 microcontroladores y módulos Bluetooth conectados.

Se realizaron varios experimentos para evaluar el funcionamiento del ADC.

1. Evaluando la velocidad de los periféricos ADC.
2. Evaluando la velocidad de transferencia de datos del puerto UART.
3. Evaluando el uso de la memoria para transferir datos a diferentes velocidades.

El flujo de la experimentación fue como se presenta en la Figura 3.26.

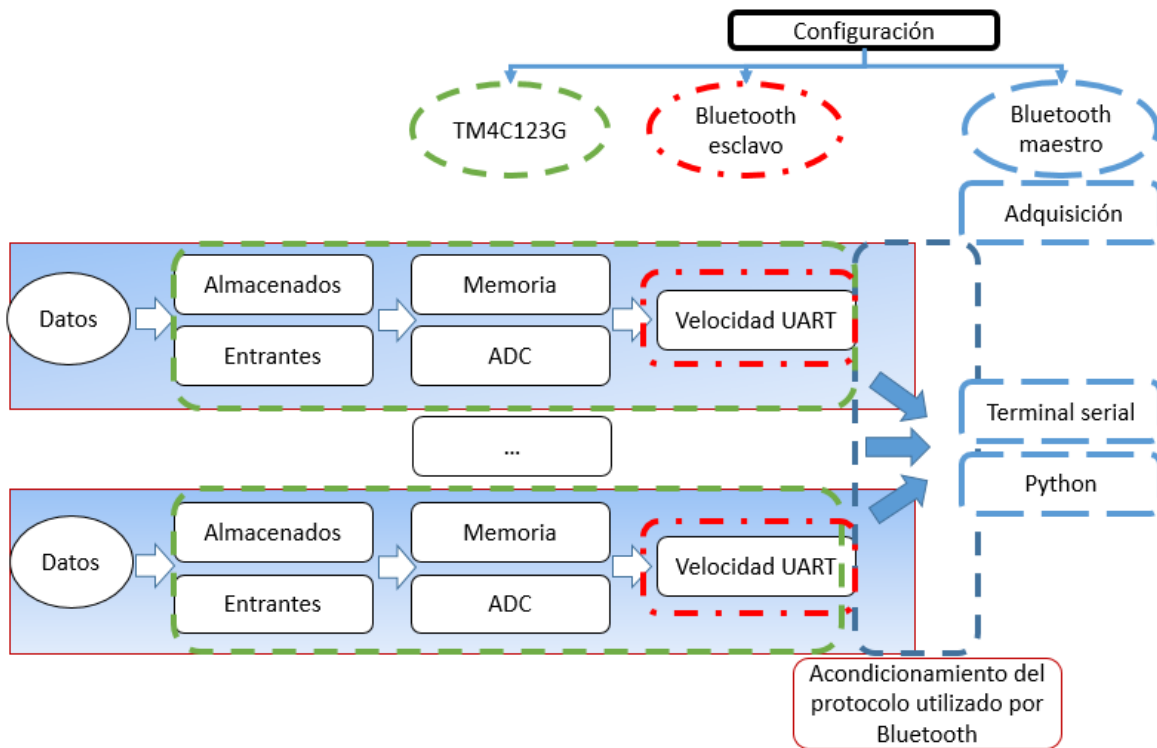


Figura 3.26 En esta imagen se muestran los parámetros y la experimentación que se realizó a cada uno de los elementos de nuestro sistema de adquisición.

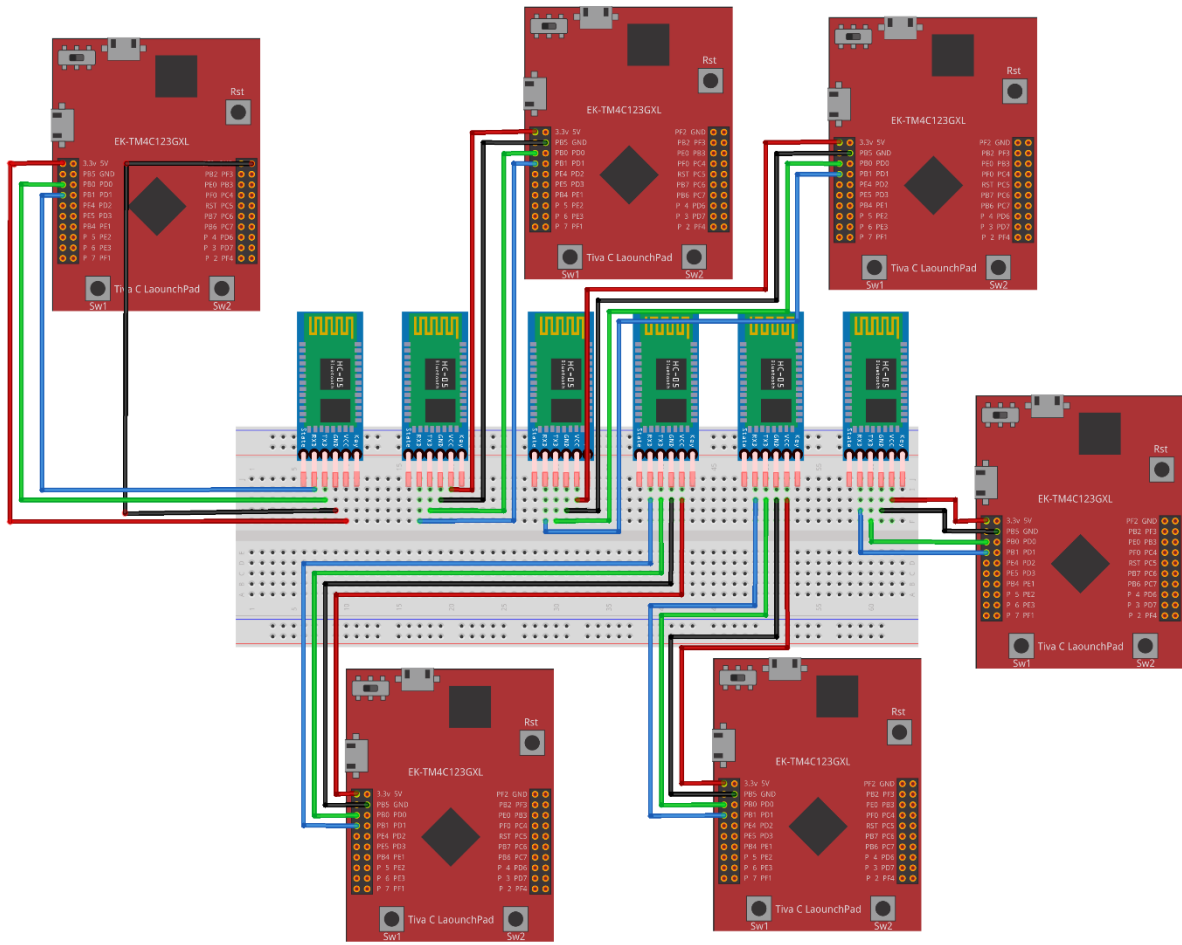
En la imagen anterior se observó que había problemas con la conexión inalámbrica al momento de conectar un número mayor de módulos al concentrador (interrupción de la comunicación), por lo que se evaluaron diferentes opciones:

1. El problema era debido a las velocidades de transmisión de datos.
2. La suspensión de la conexión se debía a la inestabilidad de conectar el máximo de módulos permitidos simultáneamente
3. Una saturación de información en las vías de transmisión.

Se decidió realizar la siguiente experimentación para descubrir el origen de la inestabilidad en la conexión.

1. Se conectó un solo modulo con parámetros establecidos en la velocidad para poder muestrear una señal de 1 KHz.
2. Se generó una señal senoidal con el analog Discovery para enviar los datos directamente al ADC del microcontrolador.
3. Se leyó la información del puerto serial virtual con la terminal. Una vez corroborada la información de la terminal.
4. Se utilizó un programa en Python para almacenar los datos entrantes en archivos y poder analizar los datos transmitidos.
5. Habiendo corroborado la información, se agregó otro modulo.
6. Se realizaron los pasos desde 1 al 6 hasta tener el número máximo de módulos capaces de transmitir datos sin errores.

La Figura 3.27 muestra la configuración de 6 módulos bluetooth con sus microcontroladores



fritzing

Figura 3.27 Número máximo de módulos que puede soportar teóricamente un Bluetooth maestro.

3.7 Simulación en línea

En esta etapa se requería investigar uno de los proyectos encontrados que indicaban que el trabajo/simulación en línea era posible por lo que se estudió el proyecto Das3-sim, utilizando las herramientas en Opensim y algunas librerías de Matlab y Python.

Para evaluar el funcionamiento del simulador se descargó el modelo de la dirección [94], de igual forma se revisó la información complementaria encontrada en [95].

Los pasos para la modificación del modelo fueron los siguientes:

1. Modificar el modelo base agregándose los grados de libertad faltantes.
2. Configurar la interfaz gráfica del usuario (GUI) para crear los controles en tiempo real.

3. Y finalmente evaluar la función MX de Matlab que enlaza todas las herramientas del modelo.

3.8 Programación en Labview para crear una clasificación en línea con instrumentos virtuales.

Aquí la idea era desarrollar un programa en Labview para realizar una clasificación en tiempo real de 6 movimientos de la mano, el clasificador se creó fuera de línea en Matlab. El programa Clasificador_vi_EXE utilizará una base de datos obtenida de [67], la misma base de datos que utilizamos para crear un clasificador de 15 movimientos. De igual forma como se realizó el clasificador en la sección X, se creó este nuevo clasificador, pero con solo 5 movimientos de mano (la flexión de cada dedo individualmente y mano cerrada).

Una vez con el clasificador creado, se procede a sacar las 5 funciones cuadráticas del modelo.

Se reproducen todas las operaciones matemáticas del preprocesamiento y del procesamiento en Labview, para que los datos tengan el mismo tratamiento que el procesamiento fuera de línea con el cual se crea el modelo en Matlab.

Con una base de datos cargada se procede a hacer la clasificación en línea.

Los datos de cada uno de los movimientos se colocan en registros que se seleccionan en el panel frontal del programa.

Dentro de la creación de la interface grafica se planea brindar más información del modelo, de la clasificación y de las matrices generadas.

Capítulo 4 Resultados y Discusión

Resultados

4.1 Introducción

En el presente capítulo se desglosan los resultados que se obtuvieron a partir de la secuencia de experimentación del capítulo anterior. El capítulo inicia con los resultados obtenidos del sistema de adquisición analógico desarrollado, el esquemático de su circuito, los prototipos desarrollados y algunas pruebas de funcionamiento que se le realizaron, posteriormente se detallan las rutas examinadas para tener un preprocesamiento de una base de datos analizada llegando a la ruta propuesta y al modelado de un clasificador de 15 movimientos de mano-dedos, posteriormente se estudia el uso del microcontrolador para realizar una transferencia de datos de forma inalámbrica utilizando bluetooth y el uso de Opensim para la simulación de movimiento fuera de línea, se examina la viabilidad de llevar estos análisis a tiempo real dentro del software. En la última parte del capítulo se encuentran los resultados del programa desarrollado en Labview para crear una clasificación en este ambiente a partir del análisis creado en Matlab para un clasificador de 5 movimientos, terminando el capítulo con el análisis del microcontrolador seleccionado para crear una adquisición de señales EMG.

4.2 Sistema de adquisición analógico para EMG

Se seleccionó el AFE AD8232 como módulo para realizar la adquisición de las señales eléctricas musculares, uno de las primeras pruebas realizadas fue verificar el funcionamiento de uno de los módulos comerciales con este AFE, dicho módulo es utilizado y se encuentra diseñado para realizar una adquisición de señales ECG, con un ancho de banda de 0.5-40 Hz y una ganancia de 1100. Se evaluó con una serie de señales eléctricas la adquisición en términos de ruido y amplificación, verificando el desempeño de estos módulos, con un voltaje de entrada de 3.3V y una entrada senoidal de 1 mV pico, con una frecuencia de 5 Hz se obtuvo la gráfica de la Figura 4.1. En esta grafica podemos visualizar el funcionamiento del dispositivo de acuerdo con las especificaciones técnicas del módulo evaluado.

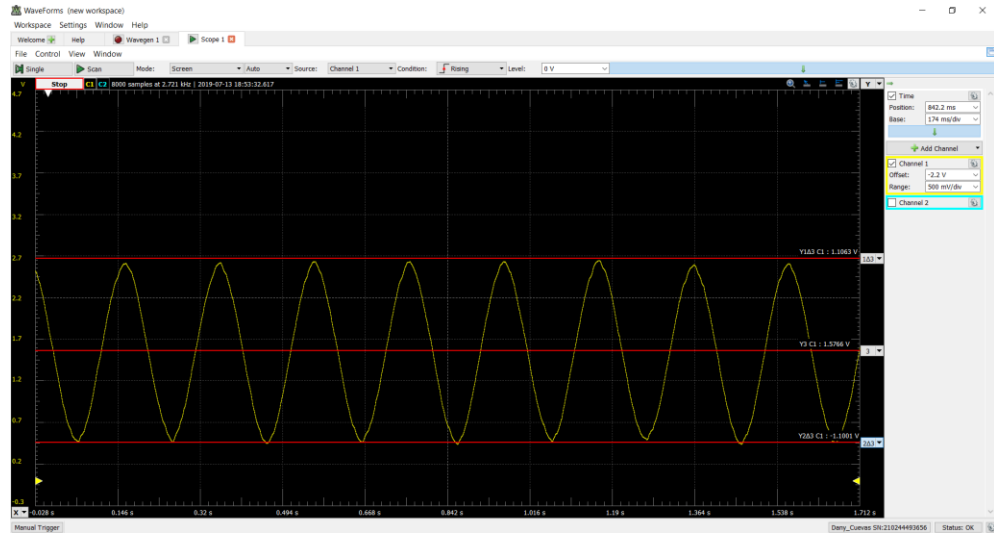


Figura 4.1 Señal del módulo comercial para censar biopotencial cardiaco.

Posterior al análisis de modulo probado se diseñaron los filtros con AD8232 filter design tool. Se obtuvo un filtro con frecuencias de corte en 20 y 480 Hz, con una ganancia de 1000 el diseño desarrollado es el que se muestra en la Figura 4.2, de igual forma se crea la impresión del circuito en la placa pcb con las dimensiones mostradas en la Figura 4.3.

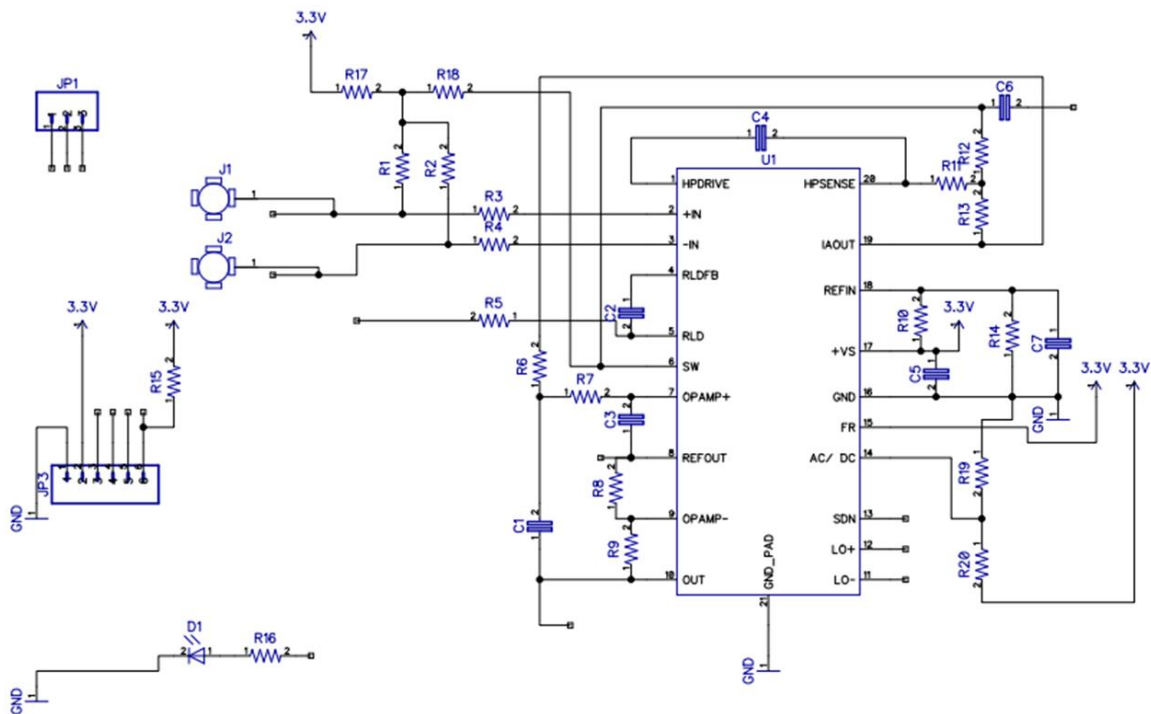


Figura 4.2 Esquemático del filtro para señales musculares utilizados.

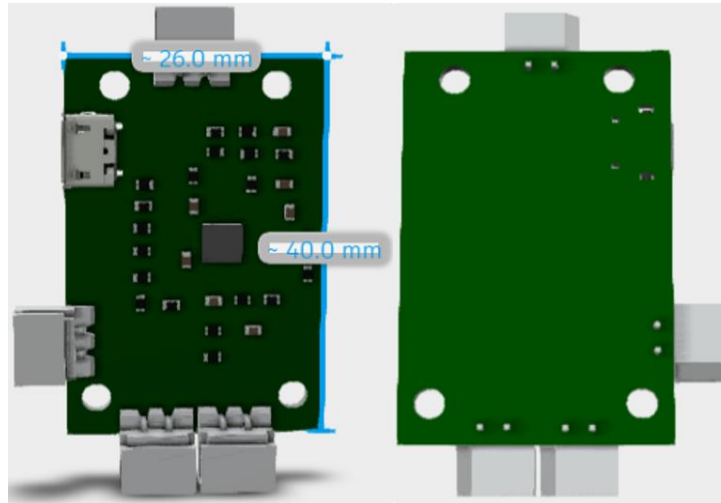


Figura 4.5 Se describen las medidas y diseño propuesto para el Prototipo 2 del sistema de adquisición analógico.

El diseño del prototipo 2, tiene menores dimensiones que su antecesor, en esta versión los electrodos sobre la placa se eliminaron del diseño y se agregó un conector comercial para el uso de electrodos cableados. Como se puede observar cuenta con más conectores, la Figura 4.6 describe la función de cada uno de ellos.

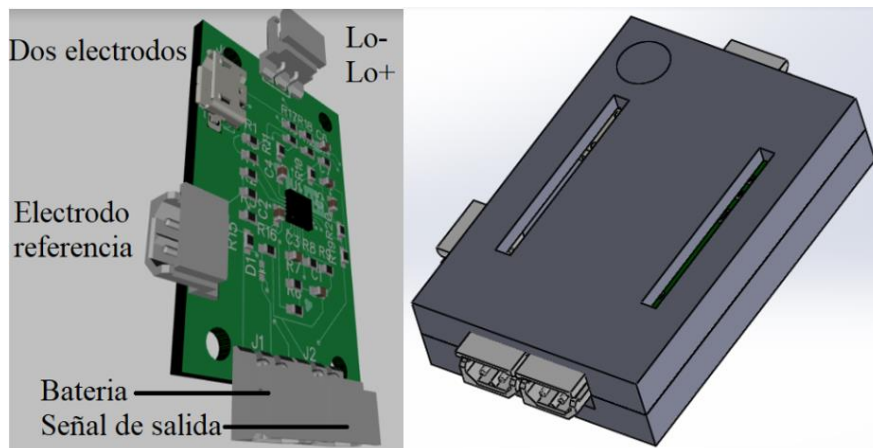


Figura 4.6 Prototipo 2 del sistema de adquisición analógico, descripción de los conectores y de la carcasa propuesta para su protección.

Se tiene un circuito para filtrar la señal EMG de 20 a 480 Hz con una ganancia de 1000. Se tomaron varias señales EMG con movimientos típicos, uno de los mostrados se puede observar en la Figura 4.7 que muestra el movimiento de mano cerrada. Se puede ver un ruido que es la señal de 60 Hz de la línea de corriente.

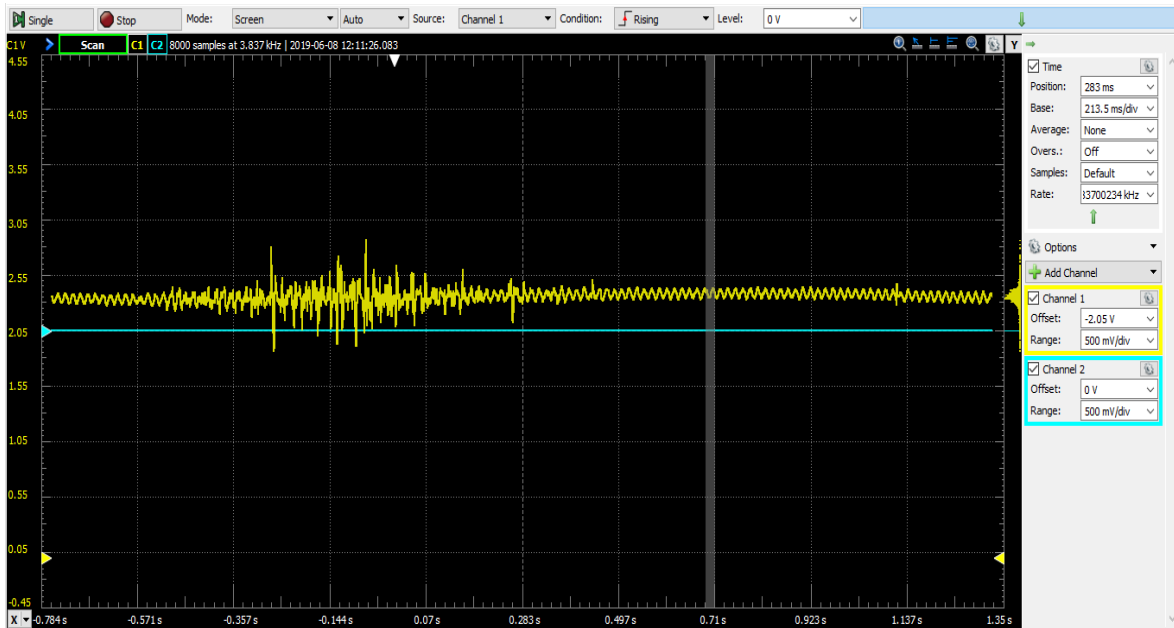


Figura 4.7 Señal EMG adquirida de antebrazo ante un movimiento de mano cerrada.

Una vez contando con el módulo analógico funcionando se validó la funcionalidad del módulo ante posibles variaciones del voltaje de alimentación, como de antemano sabemos que el módulo funcionará con baterías aquí se detallan los resultados. Se procedió a introducir una señal de entrada de 2 mV Vp que son 4 mVpp con una frecuencia de 200 Hz a un pin del electrodo de nuestro circuito. Los otros dos pines de los electrodos se fueron a tierra. Se procedió a variar el voltaje de alimentación del chip AD8232 desde 1.5 hasta 3.7 V, con aumentos de 0.1 V.

Se puede observar en la Tabla 4.1 que el voltaje de entrada está directamente relacionado con el voltaje de referencia del módulo de adquisición, de igual forma la señal de salida variará con este voltaje de entrada sin cambiar las magnitudes con las que se está trabajando. Se puede observar que debajo de 1.9 V el módulo entrega señales incompletas.

Tabla 4.1 Tabla con promedios de valores obtenidos con varios niveles de voltaje de alimentación ante una misma señal de entrada.

Voltaje de alimentación	Señal de entrada	Max V
1.55 V	2 mV 200 Hz	1.55 V Sat
1.66 V	2 mV 200 Hz	1.66 V Sat
1.73 V	2mV 200 Hz	1.73 V Sat
1.80 V	2 mV 200 Hz	1.80 V Sat
1.90 V	2 mV 200 Hz	1.90 V Sat
2.07 V	2 mV 200 Hz	2.08 V
2.11 V	2 mV 200 Hz	2.12V
2.21 V	2 mV 200 Hz	2.16V
2.30 V	2 mV 200 Hz	2.22V
2.42 V	2 mV 200 Hz	2.24 V
2.50 V	2 mV 200 Hz	2.31 V
2.60 V	2 mV 200 Hz	2.35 V
2.71 V	2 mV 200 Hz	2.41 V
2.80 V	2 mV 200 Hz	2.45 V
2.91 V	2 mV 200 Hz	2.54 V
3.00 V	2 mV 200 Hz	2.58 V
3.10 V	2 mV 200 Hz	2.56 V
3.21 V	2 mV 200 Hz	2.64 V
3.31 V	2 mV 200 Hz	2.73 V
3.42 V	2 mV 200 Hz	2.78 V
3.51 V	2 mV 200 Hz	2.80 V
3.60 V	2 mV 200 Hz	2.83 V
3.71 V	2 mV 200 Hz	2.91 V

La Tabla 4.1 muestra una relación entre los valores de voltaje de alimentación aplicados a nuestro circuito analógico y la variación que produce en nuestra salida como señal amplificada. Uno de los parámetros evaluados son los cambios en la señal de salida variando el voltaje de alimentación del circuito. La hoja de datos nos indica que hay un margen, un rango de alimentación donde el voltaje de entrada y salida no variaran, y amplificaran en lo diseñado.

4.3 Electromiografía y su procesamiento con aprendizaje automático

En la presente sección se abordarán los resultados del análisis realizado para la creación de un modelo de clasificación de 15 movimientos de la mano/dedos para determinar cuál de los tratamientos de los datos es el mejor, los preprocesamientos evaluados se definen en la sección 3.4.3, con los presentes resultados se podrá concluir si hay una diferencia significativa, y dependiendo de nuestros requerimientos elegir la ruta que más nos sea de utilidad.

4.3.1 Pre procesamiento y procesamiento de la señal EMG

Básicamente se establecen los resultados de las experimentaciones 5, 6, 7 y 8 descritos en la Figura 3.8.

4.3.1.1 Detección de valores atípicos multivariados

La Figura 4.8 muestra la gráfica con los valores mantenidos y los eliminados. El total de los datos para cada movimiento son 200,000. En esta imagen se puede observar la distribución de los datos por movimiento y la relación entre los datos que permanecieron y los que se eliminaron.

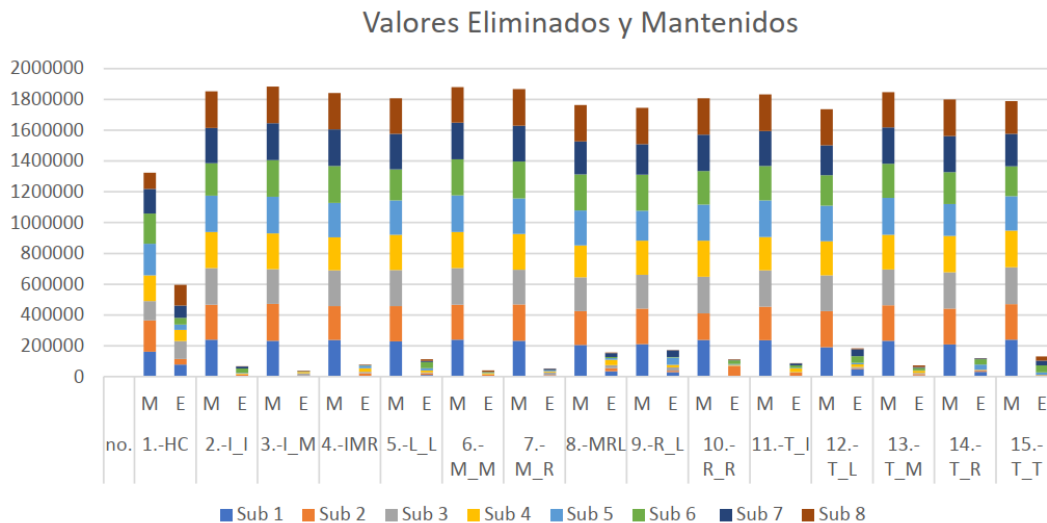


Figura 4.8 Valores mantenidos y eliminados de la base de datos analizada.

En la Tabla 4.2 se observa el número de datos mantenidos por movimiento y por sujeto, información que se encuentra representada en la Figura 4.8. La Tabla 4.3 es un resumen de la Tabla 4.2 donde podemos observar la cantidad en porcentaje de los datos eliminados por sujeto, en la Tabla 4.4 podemos observar el porcentaje de datos eliminados por movimiento de todos los sujetos.

Tabla 4.2 Resumen de los valores mantenidos y eliminados por sujeto.

		Sub 1	Sub 2	Sub 3	Sub 4	Sub 5	Sub 6	Sub 7	Sub 8	Total
1.-HC	M	153104	230698	145943	153825	239078	206588	172271	119322	1420829
2.-I_I	M	239764	231331	239083	217380	239536	227220	227220	237490	1859024
3.-I_M	M	235053	238197	234327	217009	239963	239940	239856	237387	1881732
4.-IMR	M	239167	231389	236495	189392	236372	239882	234516	237117	1844330
5.-L_L	M	216785	229618	236213	213099	231523	228463	228463	227361	1811525
6.-M_M	M	239418	233278	238953	220519	239868	237687	237687	228832	1876242
7.-M_R	M	223237	239034	233936	217670	237277	240000	223161	238723	1853038
8.-MRL	M	193448	229326	230598	195988	238041	239151	199263	237990	1763805
9.-R_L	M	198282	235296	225727	199445	222603	238758	192658	239112	1751881
10.-R_R	M	236843	177135	238620	220815	239346	230990	230990	235562	1810301
11.-T_I	M	238170	218635	236614	208157	238448	225559	225559	237899	1829041
12.-T_L	M	176971	235088	235199	208290	239873	161580	187135	237235	1681371
13.-T_M	M	233692	223312	235284	228009	239995	233660	233660	229941	1857553
14.-T_R	M	212508	233629	234866	226146	221120	223516	223516	239378	1814679
15.-T_T	M	239816	230041	237105	231584	237094	209085	209085	211893	1805703
		3276258	3416007	3438963	3147328	3540137	3382079	3265040	3395242	26861054

Tabla 4.3 Resumen de los valores mantenidos y eliminados por sujeto.

	Valores Mantenidos	Valores Eliminados	Porcentaje Eliminados
MP1	3276258	323742	8.99
MP2	3416007	183993	5.11
MP3	3438963	161037	4.47
MP4	3147328	452672	12.57
MP5	3540137	59863	1.66
MP6	3382079	217921	6.05
MP7	3265040	334960	9.30
MP8	3395242	204758	5.69
Total	26778435	1938946	6.73

Tabla 4.4 Resumen de los valores Mantenidos y eliminados en función de los movimientos.

	Valores Mantenidos	Valores Eliminados	Porcentaje Eliminados
1.-HC	1420829	499171	26.00
2.-I_I	1859024	60976	3.18
3.-I_M	1881732	38268	1.99
4.-IMR	1844330	75670	3.94
5.-L_L	1811525	108475	5.65
6.-M_M	1876242	43758	2.28
7.-M_R	1853038	66962	3.49
8.-MRL	1763805	156195	8.14
9.-R_L	1751881	168119	8.76
10.-R_R	1810301	109699	5.71
11.-T_I	1829041	90959	4.74
12.-T_L	1681371	238629	12.43
13.-T_M	1857553	62447	3.25
14.-T_R	1814679	105321	5.49
15.-T_T	1805703	114297	5.95

4.3.1.2 Modelos de clasificación

La Tabla 4.5, 4.6, 4.7 y 4.8 muestra los porcentajes de reconocimiento de una validación cruzada 10 x 10, de TRN, VAL y TST para cada uno de los clasificadores generados a través de NB, LDA y QDA para las experimentaciones declaradas en la Figura 3.8. En la Tabla 4.5 se observa que con un total de 96 características el mejor clasificador es creado con LDA, tanto para los sujetos individuales como los del grupo muestral considerado.

Tabla 4.5 Resultados del modelo de clasificación obtenido a partir del punto número 5 de la figura 3.8. Porcentaje de reconocimiento a través de una validación cruzada 10 x 10 dividiendo los datos en entrenamiento (TRN), validación (VAL) y prueba (TST). La columna de sujetos se divide en dos partes para cada clasificador, 1..8: indica que se realizó un promedio entre los datos de los 8 sujetos, S1:8 indica que el total de los datos fue un grupo muestral de los 8 individuos.

Experimentación N. 5 sin SFS				
	Sub	TRN	VAL	TST
NB	1...8	89.38	89.38	89.38
	S1:8	29.3	29.3	29.3
LDA	1...8	93.22	93.21	93.23
	S1:8	53	52.99	52.96
QDA	1...8	67.5	68.1	68.01
	S1:8	33.43	29.84	29.79

4.3.1.3 Método de selección de atributos por SFS

La Tabla 4.6 muestra los porcentajes de reconocimiento obtenidos a partir de la creación de los clasificadores con Naive Bayes, LDA y QDA. Una vez corriendo el algoritmo de SFS con el total de los datos se seleccionan las características más importantes las cuales se colocaron en la columna N.F. Se puede observar que el mejor clasificador es el QDA con un menor número de características que las consideradas en la Tabla 4.5.

Tabla 4.6 Resultados del modelo de clasificación obtenido a partir del punto número 6 de la figura 3.8. Porcentaje de reconocimiento a través de una validación cruzada 10 x 10 dividiendo los datos en entrenamiento (TRN), validación (VAL) y prueba (TST). La columna de sujetos se divide en dos partes para cada clasificador, 1..8: indica que se realizó un promedio entre los datos de los 8 sujetos, S1:8 indica que el total de los datos fue un grupo muestral de los 8 individuos. La columna N.F. corresponde a las características obtenidas a partir de la selección hacia adelante.

Experimentación N. 6 con SFS					
	Sub	N.F.	TRN	VAL	TST
NB	1...8	21	90.71	90.72	90.72
	S1:8	27	39.15	39.16	39.18
LDA	1...8	57	93.08	93.10	93.11
	S1:8	80	53.18	53.19	53.15
QDA	1...8	30	94.68	94.72	94.73
	S1:8	24	77.71	77.88	77.86

4.3.1.4 Normalización.

La Tabla 4.7 muestra los resultados de la clasificación con el conjunto de datos sin valores atípicos y posteriormente normalizados utilizando las 96 características. Se observa que el peor clasificador es NB y el mejor clasificador pertenece a QDA.

Tabla 4.7 Resultados del modelo de clasificación obtenido a partir del punto número 7 de la figura 3.8. Porcentaje de reconocimiento a través de una validación cruzada 10 x 10 dividiendo los datos en entrenamiento (TRN), validación (VAL) y prueba (TST). La columna de sujetos se divide en dos partes para cada clasificador, 1..8: indica que se realizó un promedio entre los datos de los 8 sujetos, S1:8 indica que el total de los datos fue un grupo muestral de los 8 individuos.

Experimentación N. 7 sin SFS				
	Sub	TRN	VAL	TST
NB	1...8	89.94	89.97	89.96
	S1:8	29.31	29.3	29.30
LDA	1...8	93.6	93.61	93.60
	S1:8	54.79	54.8	54.78
QDA	1...8	94.06	94.06	94.05
	S1:8	51.15	51.04	51.07

4.3.1.5 Normalización y SFS

Posteriormente en la Tabla 4.8 observamos que el peor sigue siendo NB y el mejor QDA con un número menor de características, aquí, aunque NB posee un menor número de características la diferencia entre los porcentajes de reconocimiento es considerable.

Tabla 4.8 Resultados del modelo de clasificación obtenido a partir del punto número 8 de la figura 3.8. Porcentaje de reconocimiento a través de una validación cruzada 10 x 10 dividiendo los datos en entrenamiento (TRN), validación (VAL) y prueba (TST). La columna de sujetos se divide en dos partes para cada clasificador, 1..8: indica que se realizó un promedio entre los datos de los 8 sujetos, S1:8 indica que el total de los datos fue un grupo muestral de los 8 individuos. La columna N.F corresponde a las características obtenidas a partir de la selección hacia adelante.

Experimentación N. 8 Normalización con SFS					
	Sub	N.F.	TRN	VAL	TST
NB	1...8	21	91.33	91.36	91.34
	S1:8	27	39.15	39.16	39.18
LDA	1...8	40	93.10	93.12	93.12
	S1:8	69	54.73	54.74	54.75
QDA	1...8	25	96.09	96.14	96.16

En la Tabla 4.7 podemos ver que los porcentajes de reconocimiento varían muy poco desde el menor NB seguido de LDA y terminando con mayor porcentaje en QDA, la diferencia con la Tabla 4.8 es que aquí podemos considerar el número de características.

Observando el efecto de la normalización en los datos podemos ver cómo este preprocesamiento afecta directamente en los resultados como se ve en las Tablas 4.5 y 4.7 sin utilizar SFS, es decir, utilizando las 96 características. A simple vista podemos observar que el efecto de normalización tiene un efecto positivo sobre el porcentaje de reconocimiento del QDA.

Entonces se seleccionó QDA por estas dos razones:

- Mejor porcentaje de reconocimiento para sujetos individuales y para el grupo muestral.
- Menor número de característica para sujetos individuales.

4.3.2 Métodos de validación

Uno de los objetivos de esta sección era determinar si había una diferencia significativa entre los resultados de los clasificadores creados a partir de diferentes preprocesamientos, por lo que se decidió promediar los resultados obtenidos para QDA que fue el mejor modelo de clasificación creado. Por lo tanto, se promediaron los resultados de los modelos QDA para todas las rutas (Preprocesamientos) considerados. Estos resultados podemos observarlos en la Tabla 4.9, aquí los resultados están en función de los sujetos individuales. Esta tabla se utilizó como base para generar un análisis ANOVA que encontramos en la Tabla 4.10.

4.3.2.1 ANOVA

Los cuatro tratamientos de datos establecidos fueron los siguientes

A) Normalización y SFS (Tabla 4.8),

B) Sólo Normalización (Tabla 4.7),

C) Sólo con SFS (Tabla 4.6),

D) Sin ningún tratamiento (Tabla 4.5).

La Tabla 4.9 recoge las medias de la validación cruzada de 10x10 del subconjunto de datos de TST de cada tratamiento de cada sujeto. La prueba ANOVA realizada mostró una diferencia

significativa entre las medias de los resultados ($\alpha = 0,05$, $F = 34,37$, valor $p < 0,00001$), rechazando la hipótesis nula

Tabla 4.9 Porcentaje de reconocimiento para cada uno de los tratamientos para cada sujeto las columnas con numero representan el preprocesamiento declarado en la Figura 3.8.

Sujetos	Tratamientos			
	Sin procesamiento	Solo SFS	Solo Normalización	Normalización y SFS
1	65.97	96.53	96.68	98.27
2	88.82	95.38	96.37	97.76
3	59.06	91.80	87.57	91.87
4	71.41	92.26	92.32	93.93
5	73.89	97.06	97.48	98.94
6	72.09	95.47	96.92	97.44
7	65.46	95.35	96.06	97.06
8	47.36	94.01	89.04	93.98
Promedio	68.01	94.73	94.01	96.16

Tabla 4.10 Análisis ANOVA para determinar si las medias de los tratamientos utilizados tenían diferencias significativas.

FUENTE DE VARIACIÓN	GRADOS DE LIBERTAD	SUMA DE CUADRADOS	CUADRADO MEDIO	F	TABLA F 0.99	Aceptación de Ho
Tratamientos	3	4,383.943	1461.314	34.373	2.947	NO
Error	28	1,190.381	42.514			
Total	31	5,574.324				

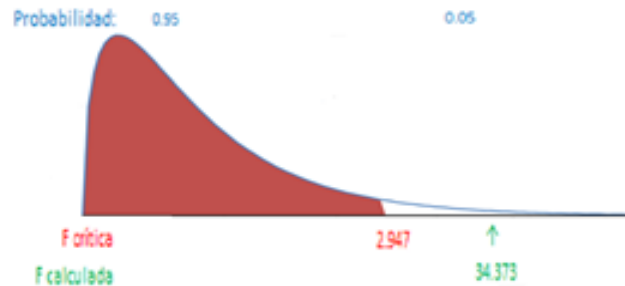


Figura 4.9 Gráfica de ANOVA.

4.3.2.2 Método de la prueba de significancia de Duncan

Se aplica el método de Duncan para determinar cuáles de los tratamientos son iguales estadísticamente o cuáles son diferentes. Como podemos observar en la Tabla 4.11 el método de Duncan nos indica que los tratamientos A, B y C son iguales estadísticamente y que D es diferente. Por lo cual podemos seleccionar cualquiera de los tratamientos 6, 7 y 8, como el 8 nos proporciona un menor número de características y un mayor valor en el porcentaje de reconocimiento se decidió utilizar QDA con normalización y SFS.

Tabla 4.11 Tabla de resultados de la prueba de significación de Duncan, la conclusión es que las medias de los resultados obtenidos de los procedimientos A, B y C son iguales, y D (Sin ningún tratamiento) no es igual. Por lo tanto, se puede elegir cualquiera de los tratamientos "Normalización y SFS", "Sólo Normalización" y "Sólo con SFS".

Diferencia poblacional	Diferencia muestral comparada con el rango R_p		Decisión
$\mu_A - \mu_C$	96.15-94.73=1.42	$< 7.1 = R_4$	No significativa
$\mu_A - \mu_B$	96.15-94.05=2.10	$< 7.0 = R_3$	No significativa
$\mu_A - \mu_D$	96.15-68=28.14	$> 6.6 = R_2$	Significante
$\mu_C - \mu_B$	94.73-94=0.67	$< 7.0 = R_3$	No significativa
$\mu_C - \mu_D$	94.73-68=26.75	$> 6.6 = R_2$	Significante
$\mu_B - \mu_D$	94-68=26.04	$> 6.6 = R_2$	Significante

4.3.3 Conversión a un nuevo espacio de matrices dispersas

Al algoritmo de conversión se le proporcionó el conjunto de características obtenido para la ruta seleccionada, es decir QDA, se trabajó con el total de características (en este punto 49). Como se estableció este método fue utilizado para reducir aún más la cantidad de características utilizadas y de cierto modo para intentar aumentar el porcentaje de reconocimiento del modelo y generar un modelo más simple.

En la Figura 4.10 podemos ver la curva de características principales contra reconocimiento. Se puede observar que mientras más características principales tomemos más crece en porcentaje de reconocimiento.

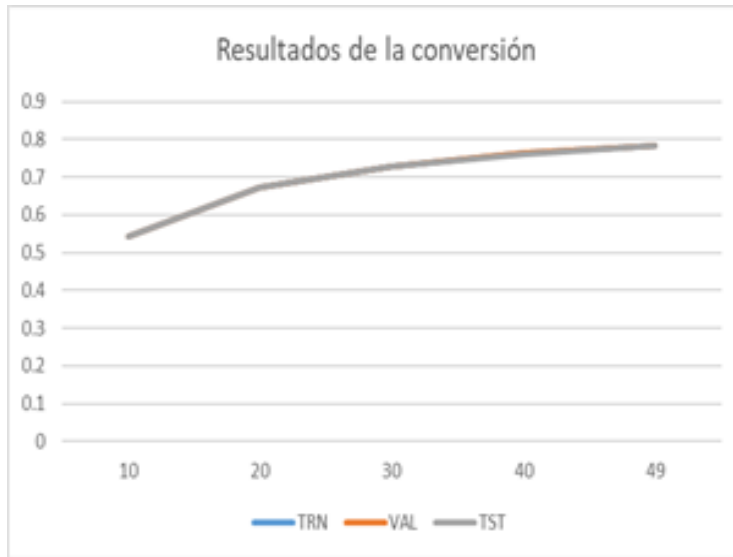


Figura 4.10 Gráfica del porcentaje de clasificación seleccionando un menor número de características después de la conversión a matrices dispersas.

4.3.3.1 Matriz de confusión

Una matriz de confusión nos ayuda a visualizar que movimientos se están confundiendo con los movimientos reales.

Matriz de confusión para los 15 movimientos a partir de la clasificación del set de datos seleccionado.

Tabla 4.17 Matriz de confusión.

	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12	M13	M14	M15
M1	572	0	3	4	0	0	0	89	2	4	0	0	0	12	1
M2	0	568	11	3	5	54	13	1	7	7	24	0	3	1	22
M3	1	11	619	1	9	22	35	0	6	5	23	0	13	5	6
M4	1	0	17	524	1	6	133	6	21	5	11	0	16	0	0
M5	0	30	10	1	510	56	2	0	3	15	20	0	48	7	26
M6	0	21	25	0	10	573	10	0	1	17	2	0	28	2	34
M7	0	0	38	31	1	1	600	4	6	7	3	0	4	1	0
M8	1	1	2	33	0	0	27	560	31	3	1	2	0	11	0
M9	10	3	80	49	3	0	113	18	374	9	2	4	0	46	0
M10	0	12	1	15	12	67	56	1	4	480	2	0	17	9	14
M11	2	21	13	3	6	3	0	0	2	0	621	0	24	6	7
M12	2	4	18	0	24	1	1	1	6	8	33	603	5	40	11
M13	0	1	7	4	6	10	3	0	0	0	3	0	657	5	9
M14	0	2	21	12	8	2	27	6	12	9	1	9	5	619	0
M15	0	47	17	0	5	37	8	0	0	10	22	2	11	0	561

4.3.3.2 Formulaciones estadísticas después del cambio de espacio

La Tabla 4.12 nos presenta los parámetros estadísticos considerados sensibilidad, especificidad, precisión y F1 score para cada una de las 15 clases, un promedio de los 8 sujetos evaluados para la Tabla 4.12 y para un grupo muestral de los ocho sujetos evaluados la Tabla 4.13.

Tabla 4.12 Promedio de los parámetros estadísticos para los 8 sujetos Evaluados.

QDA				
Clases	Sensibilidad	Especificidad	Precisión	F1 score
1	100	100	100	100
2	88.65	99.09	88.65	88.65
3	96.90	99.74	96.90	96.90
4	98.88	99.83	97.80	98.34
5	98.68	100	100	99.33
6	97.32	99.91	99.09	98.19
7	97.80	99.91	98.88	98.34
8	100	100	100	100
9	97.40	99.42	91.46	94.33
10	100	100	100	100
11	92.52	99.74	97.05	94.73
12	100	100	100	100
13	97.75	99.50	93.54	95.60
14	92.39	99.83	97.70	94.97
15	90.78	99.10	86.25	88.46

Tabla 4.13 Parámetros estadísticos para un grupo muestral de 8 sujetos con 49 características.

QDA				
Clases	Sensibilidad	Especificidad	Precisión	F1 score
1	83.26	99.78	97.11	89.65
2	78.99	98.09	78.77	78.88
3	81.87	96.74	70.18	75.57
4	70.71	98.06	77.05	73.75
5	70.05	98.87	85.00	76.80
6	79.25	96.81	68.87	73.69
7	86.20	94.82	58.36	69.60
8	83.33	98.42	81.63	82.47
9	52.60	98.76	78.73	63.06
10	69.56	98.77	82.90	75.65
11	87.71	98.15	80.85	84.14
12	79.65	99.78	97.25	87.58
13	93.19	97.81	79.06	85.54
14	84.44	98.17	81.02	82.69
15	77.91	98.37	81.18	79.51

4.3.4 Modelo de clasificación considerado

El modelo de clasificación seleccionado fue el creado a partir de QDA, este algoritmo nos proporciona 15 funciones discriminantes para la clasificación de 15 clases, la función que tenga el argumento máximo es la función seleccionada es decir la clase seleccionada para un conjunto de datos evaluado.

4.4 Simulación del movimiento en Opensim

Con el modelo funcional de la mano definido en [96] se crearon los siguientes vectores de avance para generar el movimiento de forma progresiva en cada una de las articulaciones involucradas en los movimientos considerados. La Tabla 4.14 muestra cada uno de los vectores creados a partir de cada uno de los movimientos ideales. Los movimientos ideales se crearon a partir de la posición inicial y final de la mano, proporcionado por la base de datos considerada de cada uno de los movimientos. El movimiento se realizó durante 5 segundos con avances de 62.5 ms equivalentes a los 250 datos de la ventana utilizada. Cada vector representa el avance de cada una de las articulaciones involucradas en cada movimiento, se tiene que si seleccionamos la fila 16 de la tabla y la sumamos 80 veces tendremos el movimiento ideal de la flexión del dedo pulgar, así con cada una de las filas de cada movimiento.

Tabla 4.14 Uniones/grados de libertad involucrados en cada uno de los movimientos del modelo modificado de la mano y el avance de cada grado de libertad en grados.

Uniones/GDL																					
	MCP2_lateral	MCP2_flex	PIP_flex	DIP_flex	MCP2M_lateral	MCP2M_flex	MPIP_flex	MDIP_flex	RCP2M_lateral	RCP2M_flex	RPIP_flex	RDIP_flex	LCP2M_lateral	LCP2M_flex	LPIP_flex	LDIP_flex	thumb_abd	thumb_flex	TCP2M_lateral	TCP2M_flex	TCP2M2_flex
HC1	-0.3225	1.4516	1.4516	1.4516	0	1.4516	1.4516	1.4516	0.0483	1.4516	1.4516	1.4516	0.3225	1.4516	1.4516	1.4516	0	0	0	0	0
HC2	-0.3225	1.4516	1.4516	1.4516	0	1.4516	1.4516	1.4516	0.0483	1.4516	1.4516	1.4516	0.3225	1.4516	1.4516	1.4516	0.14	1.26	-0.4	0.2	0.84
I_I	-0.3225	1.4516	1.4516	1.4516	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
I_M	-0.3225	1.4516	1.4516	1.4516	0	1.4516	1.4516	1.4516	0	0	0	0	0	0	0	0	0	0	0	0	0
IMR	-0.3225	1.4516	1.4516	1.4516	0	1.4516	1.4516	1.4516	0.0483	1.4516	1.4516	1.4516	0	0	0	0	0	0	0	0	0
L_L	0	0	0	0	0	0	0	0	0	0	0	0	0.3225	1.4516	1.4516	1.4516	0	0	0	0	0
M_M	0	0	0	0	0	1.4516	1.4516	1.4516	0	0	0	0	0	0	0	0	0	0	0	0	0
M_R	0	0	0	0	0	1.4516	1.4516	1.4516	0.0483	1.4516	1.4516	1.4516	0	0	0	0	0	0	0	0	0
MRL	0	0	0	0	0	1.4516	1.4516	1.4516	0.0483	1.4516	1.4516	1.4516	0.3225	1.4516	1.4516	1.4516	0	0	0	0	0
R_L	0	0	0	0	0	0	0	0	0.0483	1.4516	1.4516	1.4516	0.3225	1.4516	1.4516	1.4516	0	0	0	0	0
R_R	0	0	0	0	0	0	0	0	0.0483	1.4516	1.4516	1.4516	0	0	0	0	0	0	0	0	0
T_I	0	0.3625	0.675	0.75	0	0	0	0	0	0	0	0	0	0	0	0	0	0.7875	-0.25	0.2875	0.1273
T_L	0	0	0	0	0	0	0	0	0	0	0	0	0	0.0585	0.95	0.8812	0.175	1.125	-0.25	0.4125	0.1825
T_M	0	0	0	0	-0.0412	0.28	0.875	0.9875	0	0	0	0	0	0	0	0	0.1875	0.575	-0.25	0.3825	0.325
T_R	0	0	0	0	0	0	0	0	0	0.1691	0.9875	0.875	0	0	0	0	0.1812	0.875	-0.25	0.3678	0.3125
T_T	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.14	1.26	-0.4	0.2	0.84

4.4.1 Simulación a partir del modelo de clasificación

Las siguientes tablas muestran el orden del reconocimiento de cada sujeto y su etiqueta de clase (tipo de movimiento) de los datos de prueba, estas tablas son creadas a partir de los datos del conjunto total de los 8 sujetos, simplemente se separaron los resultados por sujeto y movimiento. Esto se hace para descubrir específicamente que es lo que está sucediendo con el modelo que se obtuvo a partir del grupo muestral. Se muestran dos tablas, la Tabla 4.15 posee el movimiento peor clasificado al mejor, en la Tabla 4.16 encontramos su porcentaje de reconocimiento correspondiente del movimiento de la Tabla 4.15.

Tabla 4.15 Orden de los porcentajes de reconocimiento de menor a mayor de los movimientos por sujeto.


	Suj1	Suj2	Suj3	Suj4	Suj5	Suj6	Suj7	Suj8	
menor reconocimiento	T	T	R	I	HC	RL	R	RL	
	TI	L	L	M	IMR	MR	IM	L	
	IM	RL	RL	IM	TL	R	IMR	R	
	IMR	TR	TL	R	MRL	HC	T	M	
	R	MRL	IMR	RL	RL	M	MR	TR	
	TR	I	MR	MRL	I	IMR	MRL	IMR	
	I	M	I	IMR	L	IM	M	TL	
	RL	HC	IM	T	MR	T	TL	I	
	TM	TI	TI	L	M	TI	TI	IM	
	TL	IM	MRL	TI	TR	I	L	TM	
	M	IMR	TR	TM	TM	MRL	RL	MR	
	MRL	TM	M	TR	IM	L	I	MRL	
	L	R	TM	MR	R	TR	HC	TI	
	mejor reconocimiento	MR	TL	T	TL	TI	TL	TM	T
		HC	MR	HC	HC	T	TM	TR	HC

Tabla 4.16 Porcentajes de reconocimiento de la Tabla 4.15.

	Suj1	Suj2	Suj3	Suj4	Suj5	Suj6	Suj7	Suj8
menor reconocimiento	41.76	42.11	34.74	45.12	13.25	0.00	72.00	2.06
	72.73	42.39	36.84	51.14	31.63	66.67	86.27	27.27
	74.07	44.68	46.51	60.00	36.56	73.03	88.00	36.00
	79.31	66.02	50.55	61.45	72.73	77.32	88.78	39.58
	80.85	76.92	56.44	66.67	76.25	85.92	90.00	60.78
	81.94	77.78	67.07	67.00	82.76	89.32	91.03	61.90
	87.04	77.78	71.60	69.66	89.80	90.43	92.31	65.88
	91.40	79.00	76.77	72.00	93.51	90.91	93.41	70.24
	91.49	88.54	79.45	74.76	97.65	92.31	96.51	74.19
	95.65	92.08	82.95	79.78	97.73	93.55	96.67	82.67
	96.97	92.41	89.69	80.00	98.90	93.90	96.70	83.52
	97.70	94.19	91.92	85.06	100	94.32	96.81	86.59
	98.78	94.25	95.88	88.61	100	98.97	100	91.43
	98.89	99.13	96.92	90.57	100	100	100	92.31
mejor reconocimiento	100	100	100	100	100	100	100	100

La Tabla 4.17 es un resumen de las tablas anteriores donde se promedian los porcentajes de reconocimiento de cada uno de los 15 movimientos evaluados.

Tabla 4.17 Porcentaje de reconocimiento promedio de los 8 sujetos.

De menor a mayor	Porc. Rec.
9.-R_L	53.03
10.-R	69.04
5.-L	70.10
4.-IMR	71.08
15.-T	78.10
2.-I	78.11
12.-T_L	78.97
6.-M	79.16
3.-I_M	81.73
8.-MRL	83.60
1.-HC	83.70
14.-T_R	85.02
7.-M_R	86.03
11.-T_I	87.59
13.-T_M	92.89

Las tablas anteriores muestran como el clasificador para el conjunto muestral de los 8 sujetos se desempeñó. Estos resultados se obtuvieron a partir de una matriz donde estaban los datos de los 8 sujetos, clasificando los datos podemos observar que hay dos movimientos que se empatan el movimiento 15 y el 9 como los peores, sin embargo, también nos proporciona el porcentaje de reconocimiento de cada uno de estos. Se observa como el clasificador se desempeña de mejor o peor forma para los datos de distintos sujetos. Concluyendo que el peor movimiento fue el 9 (RL).

Donde el sujeto 6 tiene un 0% de clasificación y el 8 tiene un 2% comparado con el movimiento 15 con un 41.76% y un 42.11% del sujeto 1 y 2 respectivamente.

En la Tabla 4.17 podemos observar el porcentaje de reconocimiento de cada movimiento promediado de los 8 sujetos para un mismo clasificador. Si promediamos los porcentajes separados de los movimientos podemos observar que el peor clasificado sigue siendo RL con un 53.03% seguido de RR con 69.04%, después tenemos a los mejores clasificados con TM con 92.89 seguido por TI con 87.59%.

La Figura 4.11 muestra una repetición completa (se tomó la repetición 1) del movimiento 9 (RL) del sujeto 6 que fue el que tuvo un 0% de reconocimiento: debiendo ser RL, de 76 clasificaciones 11 fueron para el movimiento RM (14.5%) y las demás fueron el movimiento IM (85.5%). Se observa una predominancia del dedo Medio en la clasificación errónea del movimiento; en la Fig 4.11B se ve como el dedo medio progresa mayormente antes que cualquier otro. Si el movimiento fuera para asir (agarrar) un palo (stick) se cumpliría el objetivo (task), pero si el movimiento fuera de tocar un instrumento musical (play music) sería un completo error. Que tan grave puede ser la clasificación errónea en una prótesis dependerá de la tarea a realizar.

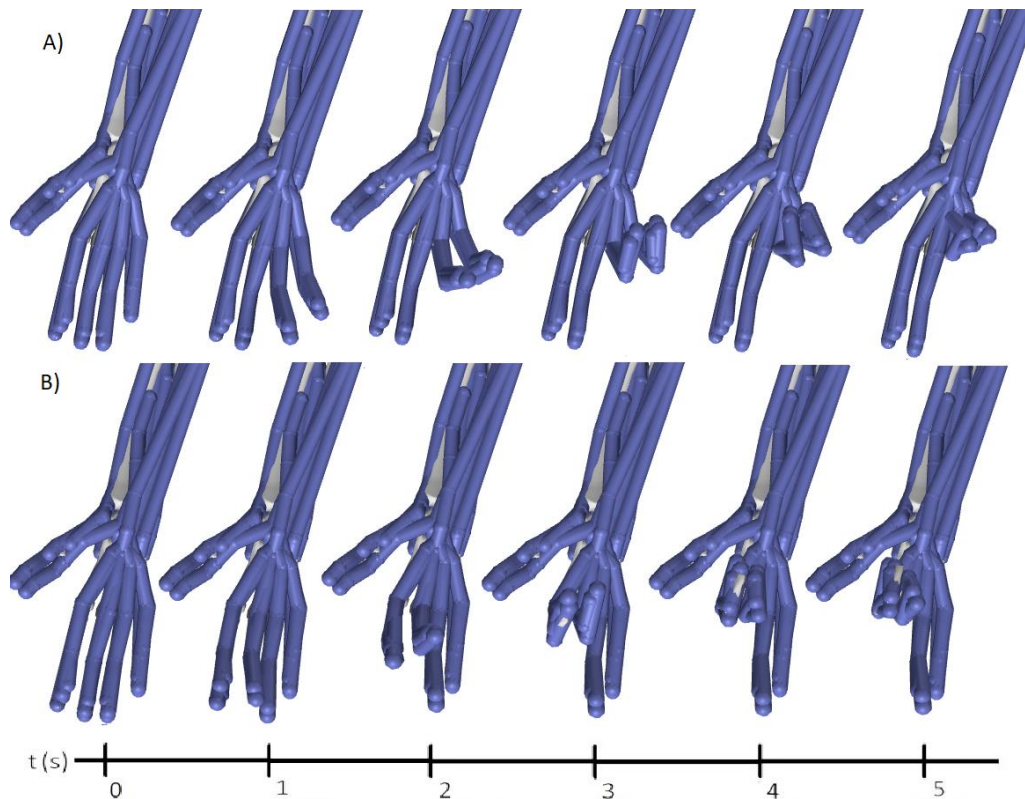


Figura 4.11 Reproducción del movimiento RL durante 5 segundos en Opensim A) Movimiento ideal B) movimiento total de la repetición 1 para el movimiento del sujeto 6 que fue el peor clasificado.

4.5 Microcontroladores y simulación en línea

4.5.1 Pruebas del sistema de adquisición

Esta primera etapa se utilizó para hacer una adquisición en línea de datos de forma inalámbrica. Como se definió en la experimentación fueron necesarias hacer varias pruebas desde la evaluación de la transferencia de datos del canal analógico a los registros del micro, de estos registros al puerto serial (UART), de este puerto a un módulo bluetooth, para posteriormente generar una transmisión y adquisición de datos en un equipo concentrador a través de bluetooth para visualizar y almacenar los datos. A continuación, se describen dichos resultados.

4.5.1.1 Transferencia de datos: Evaluando la adquisición y transferencia de los datos del ADC a los registros que controlan los datos de entrada de este periférico.

El microcontrolador fue configurado para trabajar a diferentes velocidades estableciéndose una velocidad óptima para asegurar un muestreo de la señal EMG con un ancho de banda de 20 a 480 Hz, por lo tanto, configuramos nuestro dispositivo para trabajar con una frecuencia de muestreo de 1000 muestras por segundo, siguiendo el criterio de muestreo de Nyquist.

4.5.2 Transferencia serial de datos desde el microcontrolador hasta el programa de adquisición en el equipo concentrador

La transferencia en línea se llevó de forma satisfactoria, al alcanzar una transferencia sin errores con un Baudrate de 1382400 bps. En la Figura 4.12 podemos ver uno de los experimentos realizados desde la memoria del microcontrolador hasta el puerto UART donde son leídos los datos con el Analog Discovery 2.

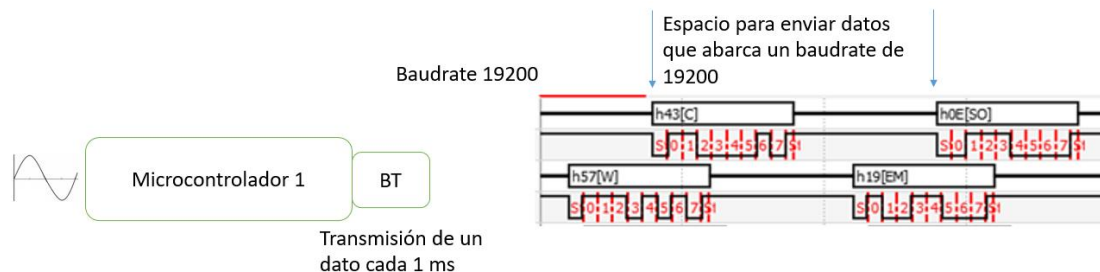


Figura 4.12 Esquemático representando una de las transmisiones de datos evaluadas desde la memoria del microcontrolador hasta el puerto UART donde son leídas a partir del Analog Discovery 2.

4.5.2.1.1 Terminales virtuales

La configuración que se utiliza en la interfaz gráfica de Putty es la que se muestra en la Figura 4.13. Esta misma interfaz se configura para guardar los datos de entrada en un archivo de texto.

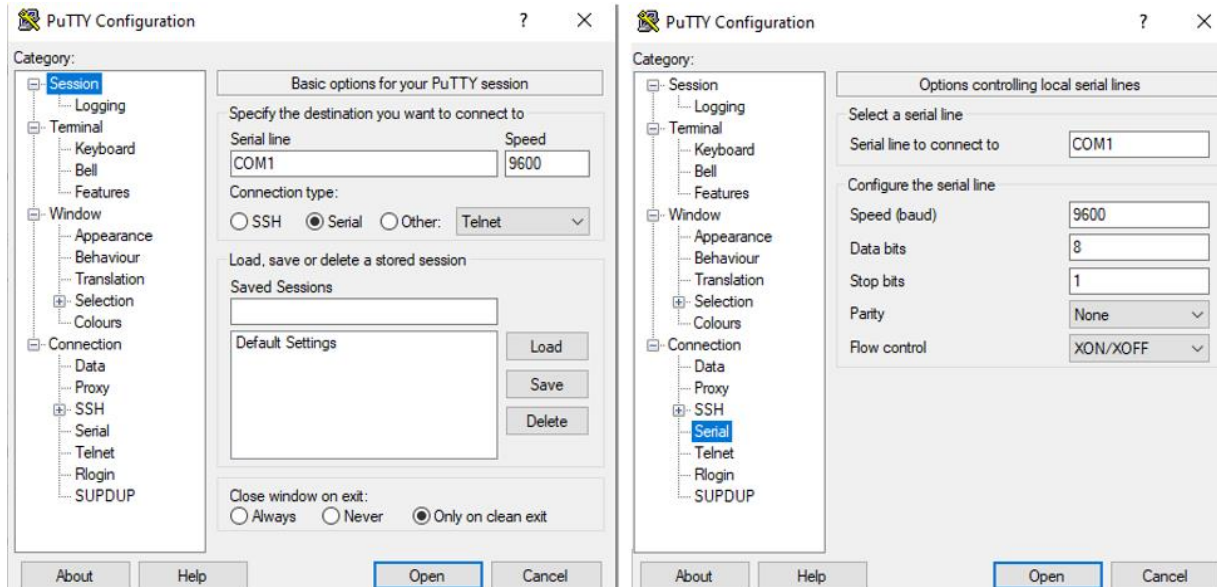


Figura 4.13 Interfaz gráfica de Putty donde se configura en el equipo concentrador los parámetros de la conexión serial creada a partir del módulo bluetooth concentrador.

Los datos que se observan en la Figura 4.14 pertenecen a un registro almacenado en la memoria del microcontrolador, contaba con 120 datos que forman un ciclo de una señal senoidal. Estos datos se grabaron con la terminal PuTTY. Se guardaron alrededor de 40,000 datos (con una velocidad de 1,382,400) sin retrasos en la adquisición, se puede observar una parte de los datos adquiridos alrededor de 5000 datos, la señal tenía una resolución de 12 bits con 4096 valores posibles.

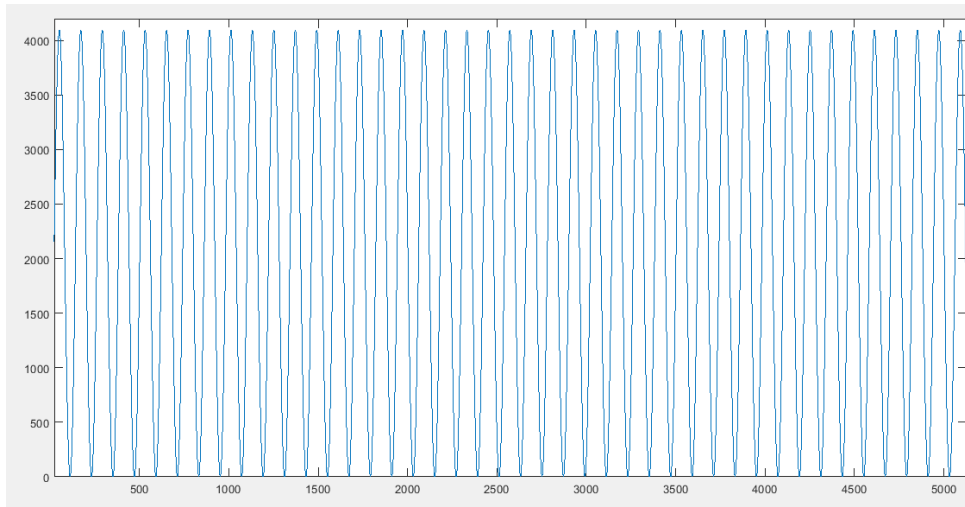


Figura 4.14 Parte de la señal adquirida a partir de una lectura de un puerto serial virtual por parte de Putty.

No se encontró ningún error en la transferencia de la señal. Fue imposible determinar el tiempo de la transferencia porque la terminal no nos permite realizar ese cálculo. En este punto se realizaron 40 adquisiciones a la misma velocidad y los datos no presentaron alteración alguna.

4.5.2.1.2 Programación en Python

Como se describió en el capítulo anterior después de realizar la adquisición exitosa a partir de las terminales PuTTY, era necesario manipular los datos de entrada, calcular el tiempo de procesamiento por lo que se realizó la programación mostrada en Python, la Figura 4.15 nos permite visualizar los datos almacenados en el equipo concentrador y nos brinda el tiempo desde que se adquirieron los datos (20,000) hasta que se guardaron en la memoria del sistema operativo.

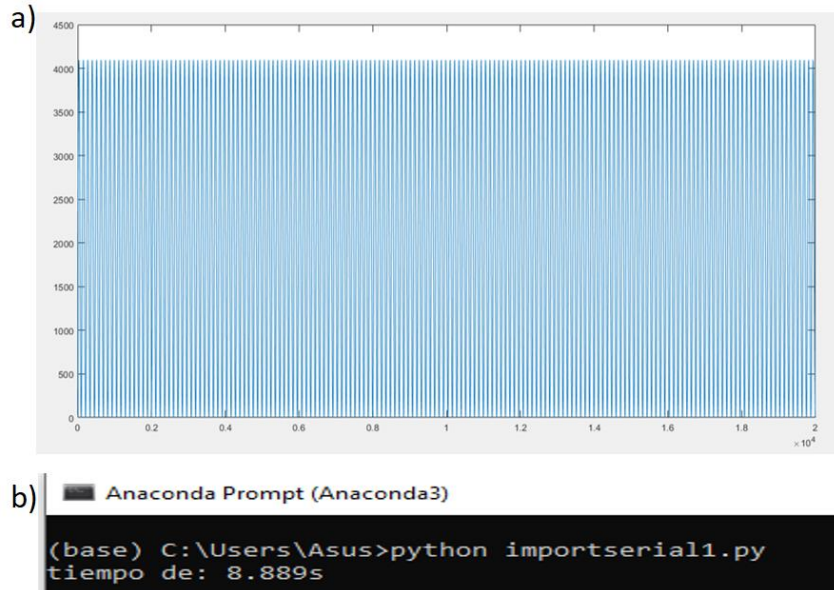


Figura 4.15 a) Datos adquiridos con el programa en Python, b) tiempo de procesamiento.

Cabe señalar que hasta este punto ya tenemos los datos para ejecutar un procesamiento en línea de ellos.

4.5.3 Transferencia de datos de forma simultanea

Aunque se realizaron una gran cantidad de pruebas para encontrar los parámetros idóneos para nuestro análisis, se describen los resultados de la experimentación detallada anteriormente. El incluir módulos extras en la transmisión inalámbrica nos limita en el ancho de datos que es capaz de fluir a través del canal creado por Bluetooth, por lo tanto, fue necesario evaluar la cantidad de módulos, la velocidad de transferencia de cada módulo y la cantidad de información de cada muestra enviada.

4.5.3.1 Evaluación de la cantidad de datos y velocidad máxima de transmisión que puede realizarse vía Bluetooth

Los siguientes resultados fueron para determinar la cantidad de datos de cada muestra que debíamos enviar, la velocidad de transferencia, a través del canal de bluetooth configurado.

La Figura 4.16 nos permite visualizar los datos enviados a través del puerto serial UART del microcontrolador. Se puede observar que se están enviando secuencias de 8 bits, los datos tienen una resolución de 12 bits, el baudrate configurado en el microcontrolador fue de 1382400. Aquí se puede notar que podemos enviar más datos a esa velocidad. Se estableció un retraso de 57600 pulsos de reloj, que es el espacio entre cada muestra.

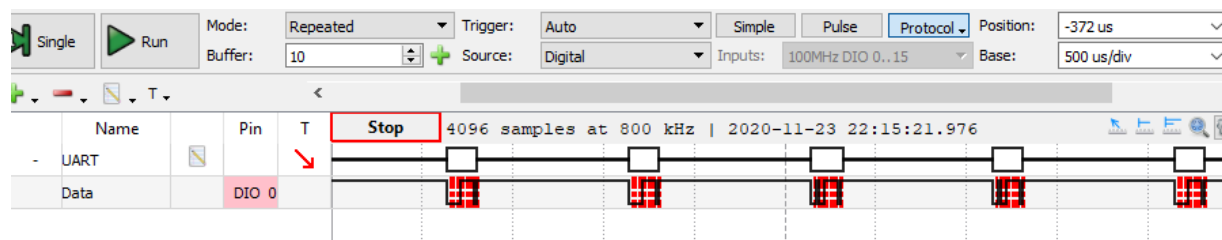


Figura 4.16 Datos enviados por el puerto UART con un baudrate de 1382400.

Hasta este momento podemos observar las formas en la que es modificable la transmisión de los datos, la Figura 4.16 es la interfaz gráfica de Analog Discovery 2 y el dispositivo se encuentra leyendo los datos de los canales digitales del microcontrolador. Entonces el Baudrate, el tiempo que tarda una muestra después de la otra, y la cantidad de bits enviados por muestra, son los parámetros que podemos modificar para aumentar la velocidad en el microcontrolador.

El siguiente paso fue determinar la velocidad de transferencia con varios módulos transmitiendo simultáneamente la Tabla 4.18 muestra los resultados obtenidos. Se consideraron múltiples baudrates, tiempo de retraso utilizado, número de bytes enviados por muestra y número de módulos activos.

Tabla 4.18 Experimentación para determinar la viabilidad de múltiples módulos bluetooth esclavos transmitiendo simultáneamente a un concentrador en un equipo de cómputo. Se realizó un total de 10 pruebas por cada experimento.

Baudrate	Retraso (Tiempo byte a byte)	Numero de bytes enviados	Numero de módulos conectados	Numero de datos transmitidos	¿Funcional?
1382400	11500 (-)	2	2	20,000	si
1382400	11500 (-)	2	3	ninguno	No*
1382400	7500 (-)	2	2	20000	si
1382400	5500 (-)	2	2	20000	si
1382400	4500 (-)	2	2	20000	si
115200	11500 (-)	2	3	20000	si
115200	24000	1	4	5000	si
115200	24000 1 ms	1	5	Max de 300	si
115200	15000 0.53ms	1	4	5000	si
115200	20000	1	5	5000	no
115200	24000 1ms	1	5	5000	no
115200	24000 1ms	1	5	30	si
57800	1 ms	1	1	20000	si
19200	1 ms	1	5	Max de 1000	si
19200	1 ms	1	5	100	si
9600	1 ms	1	Canal sat.	-	No capturado

*Problemas de conexión con el total de módulos considerados, (-) ciclos de reloj.

La Figura 4.17 y 4.18 muestran los datos obtenidos a partir del programa realizado en Python para la adquisición de 5 módulos transmitiendo simultáneamente.

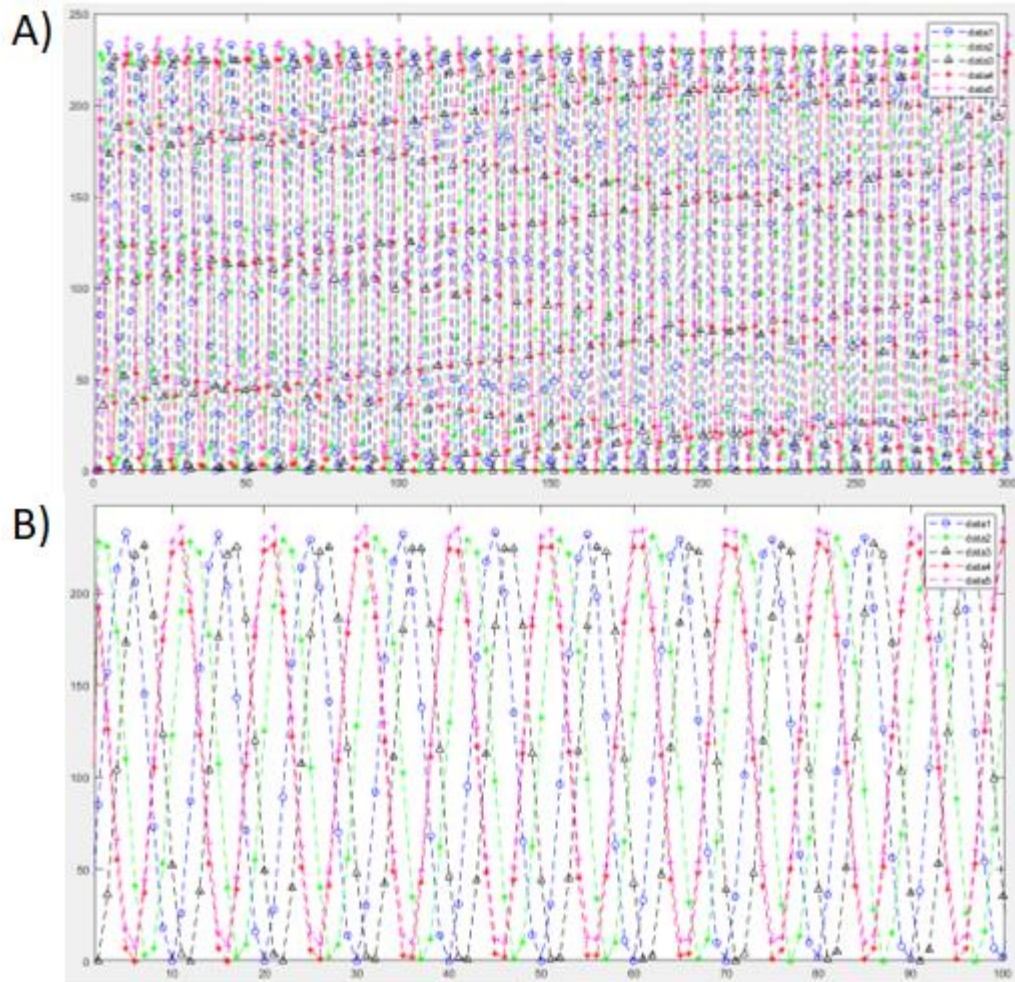


Figura 4.17 Una de las adquisiciones realizadas con 5 microcontroladores transmitiendo simultáneamente 300 datos. Sin pérdida de información y sin interrupciones.

Los siguientes son 1000 datos capturados y su zoom. Se observa una diferencia entre los niveles de referencia entre los voltajes de los diferentes canales capturados, sin pérdida de información.

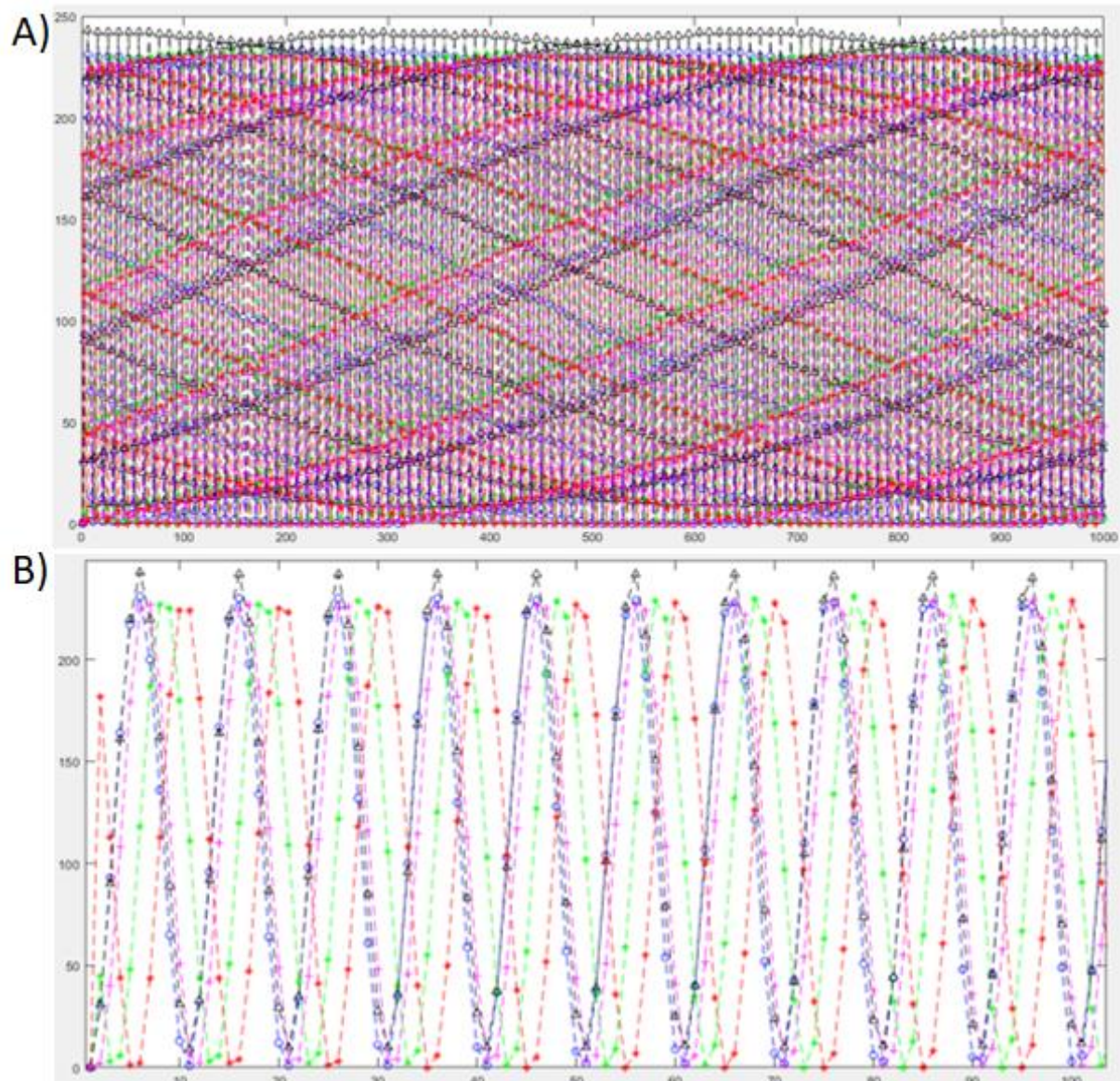


Figura 4.18 Datos capturados de una de las pruebas realizadas transmitiendo simultáneamente con 5 microcontroladores, sin interrupciones y sin pérdida de datos.

La velocidad de muestreo es de 1000 muestras por segundo, con una definición de 8 bits, con una resolución de 256 niveles en la señal adquirida.

4.5.3.2 Envío de datos a modulo maestro PC.

La Figura 4.19 muestra la configuración utilizada a partir de la Tabla 4.18, se muestra la generación de una señal senoidal que es enviada a cada uno de los microcontroladores para ser digitalizada y

enviada a los módulos Bluetooth para su envío al concentrador, el Baudrate seleccionado fue de 19200 con una resolución de 8 bits, con seis módulos transmitiendo simultáneamente.

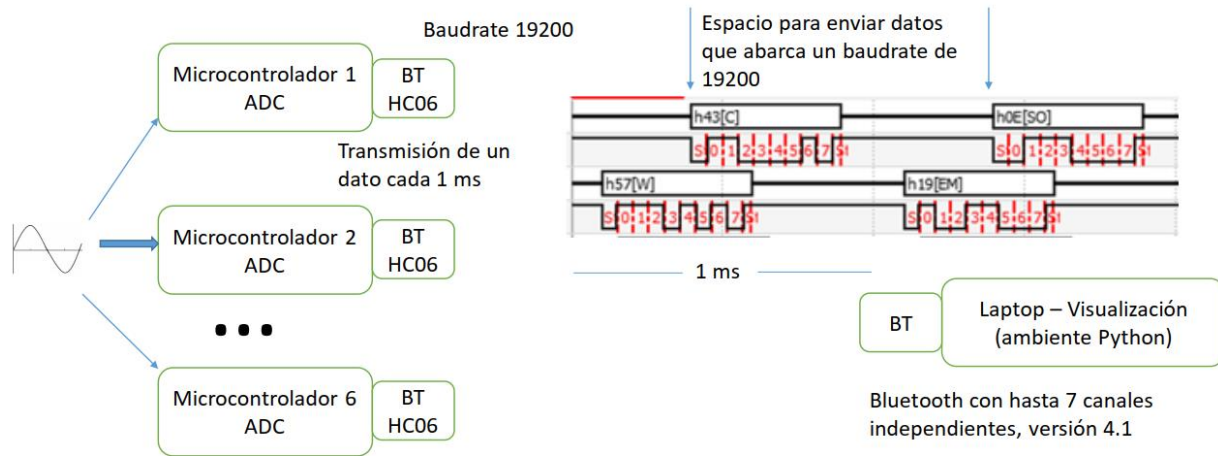


Figura 4.19 Experimentación realizada por medio de Analog Discovery y su analizador lógico.

4.5.3.3 Resultados puntuales de la prueba de adquisición de datos via ADC

Se creó una señal senoidal de 100 Hz con una amplitud de 0 a 3 V con el generador de funciones del Analog Discovery 2.

Aunque se introducía la misma señal los ADC de los diferentes micros iniciaban a diferente tiempo en este caso los datos son enviados inmediatamente. El ambiente de adquisición python almacena los datos en una memoria y de ahí los toma.

La incertidumbre de +- 3 muestras es aceptable ya que un cálculo de sEMG se hace con 250 puntos (una ventana).

Pruebas conjuntas entre la velocidad serial y la velocidad de adquisición.

Se realizaron una serie de pruebas para encontrar el baudrate al que debíamos transmitir la velocidad del ADC para no superar la velocidad del puerto serial, la cantidad de bytes enviados, el número de módulos que correrían a través del módulo concentrador Bluetooth y mantener el rendimiento de datos transmitidos.

- Estos parámetros evaluados están directamente relacionados a la transferencia de la información a través del protocolo utilizado por Bluetooth que supone maneja hasta 7 canales de forma independiente.

Resultados en la transferencia de datos:

Baudrate, Bluetooth y ambiente Python

- Un baudrate de 1,382,400 permite velocidades de muestreo mayores a 1 KHz, pero solo logran conectarse 3 módulos transmitiendo los 2 bytes del ADC.
- Bajando el baudrate a 115,200 se mantienen velocidades de muestreo mayores a 1 kHz y hasta 5 módulos conectados transmitiendo un byte, pero con los 5 módulos la transmisión no se sostiene permanentemente.
- Bajando el Baudrate a 19,200 y dejando la velocidad de muestreo a 1 kHz se logra solo la conexión de 5 módulos transmitiendo 1 byte permanentemente (es confiable en ambiente Python). Al sexto modulo se logra transmisión, pero no se sostiene permanentemente.

4.5.4 Prototipado de un microcontrolador.

4.5.4.1 Prototipado Componentes (Prototipo de un microcontrolador virgen)

Después de contar con los launchpads se consideró trabajar con una versión reducida de estos.

- Se evaluó un prototipo desarrollado por Mikro-electrónica.

En la Figura 4.20 y 4.21 se muestra el prototipo que desarrollamos con los componentes esenciales para poder reproducir la transmisión y adquisición de datos.

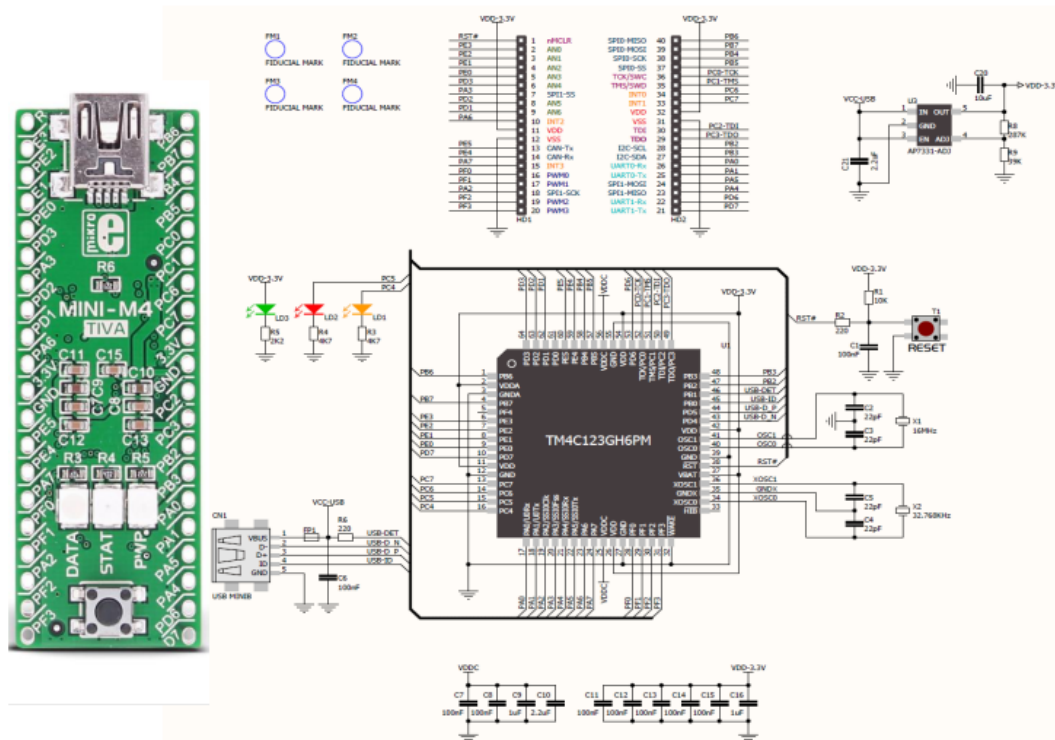


Figura 4.20 versión compacta del microcontrolador adquirido para realizar la transmisión de los datos.

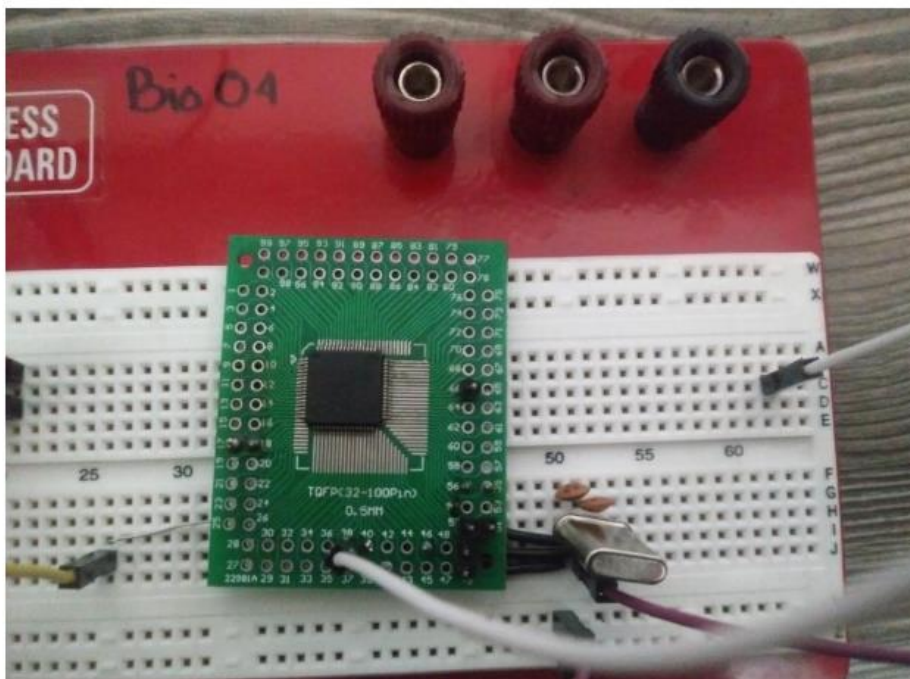


Figura 4.21 Prototipo del microcontrolador fabricado y programado.

4.6 Simulación en línea: Herramienta para clasificar movimientos en línea a partir de un clasificador implementado en Labview (Herramienta registrada en Indautor).

Aquí se describe el uso de un modelo de clasificación para 5 movimientos en un proceso en línea en un ambiente de programación como Labview, se generó una interface y una aplicación ejecutable para crear esta clasificación a partir de registro de datos almacenados, simulando la entrada de nuevos datos y su clasificación en línea.

Se desarrolla un algoritmo de clasificación en Matlab y se crea un modelo de clasificación, un discriminante lineal. Ese modelo de clasificación se implementa en Labview, y se realiza la clasificación en línea del modelo con lectura de datos crudos de las señales EMG cargadas.

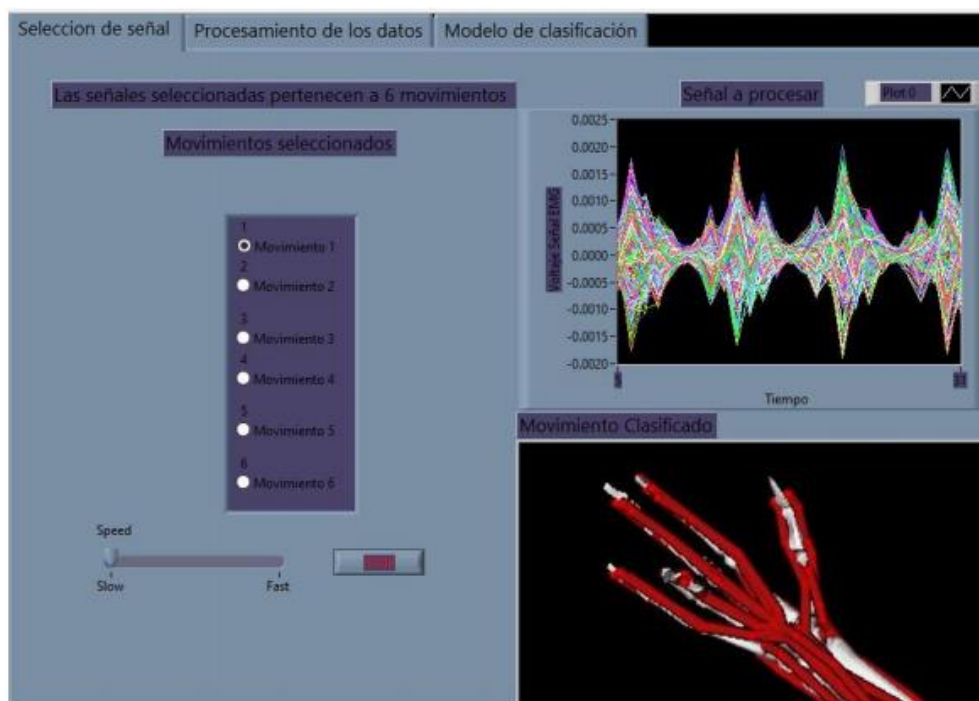


Figura 4.22 Panel frontal del clasificador en Labview.

En la pestaña de Procesamiento de los datos Figura 4.23 podemos apreciar un conteo de las ventanas que se van creando a partir del registro completo de las señales EMG sin valores atípicos, también se observa una sección de la ventana de datos seleccionada (sección 1). Aquí cada fila representa las mediciones de un par de electrodos, es decir 8 canales con una longitud de 250 columnas (sección 1) y los cálculos realizados (sección 2) para formar el nuevo vector de características por cada ventana (sección 3). Se transforma el nuevo vector de características multiplicándolo por una matriz de transformación obtenida en Matlab (sección 4). Concluyendo

con la creación de un nuevo vector para introducirlo al modelo discriminante creado en Matlab (sección 5).

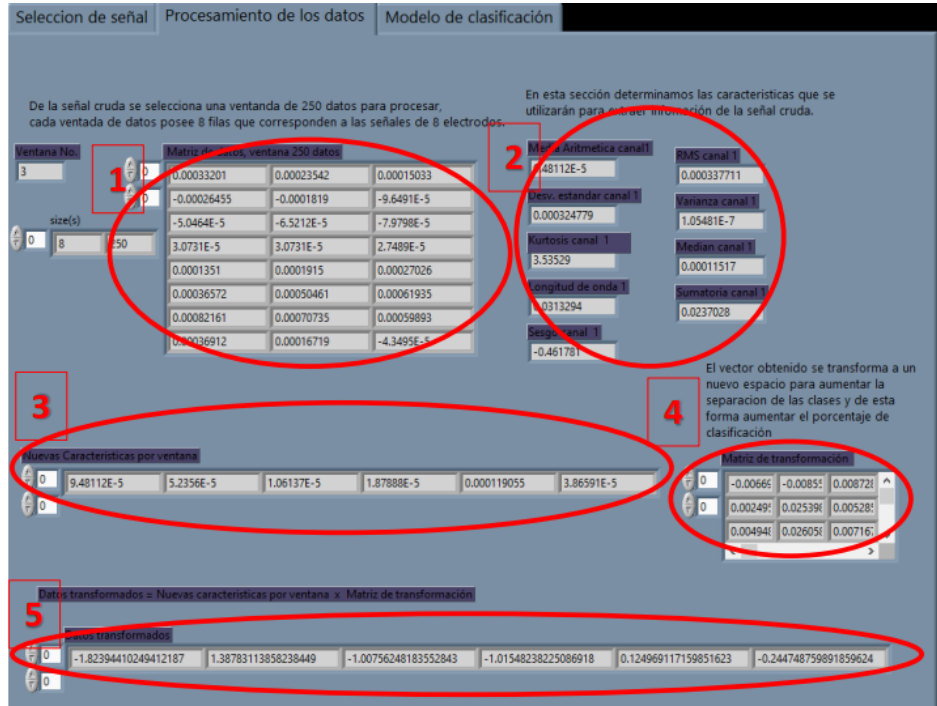


Figura 4.23 Pestaña que nos permite ver los parámetros del modelo creado.

En la pestaña número 3 se muestran las operaciones que utiliza el modelo de clasificación creado, son 6 ecuaciones de funciones discriminantes cada una enlazada a un movimiento específico, la sección 2 muestra los resultados de cada una de las ecuaciones y el valor más alto es la clase a la que se pertenecen los datos evaluados. Se realiza un procesamiento en línea, es decir simulando una clasificación en tiempo real. Se predice con técnicas de aprendizaje automático la clase de movimiento que realiza una persona con una muestra de 250 datos de señales crudas de EMG.

Selección de señal | Procesamiento de los datos | Modelo de clasificación

Un modelo de Clasificación basado en un análisis discriminante lineal es implementado.
 Un análisis discriminante es un método de clasificación, con la suposición de que diferentes clases generan datos basados en diferentes distribuciones gaussianas.
 Se generan 6 funciones discriminantes: cada una enlazada a una clase, la ecuación con mayor resultado definirá la clase a la que pertenecen las mediciones.

1 Operaciones de la primer ecuación discriminante

$A \times B_1$	$A \times B_2$	$A \times C_1$
0 -209.854160	0 233.028	0 -577.9
< >	< >	< >

2 Resultados discriminantes

-554.776
-208.68
-303.198
-211.957
-190.525
-205.749
Valor máximo
-190.525

Operaciones de la segunda ecuación discriminante

$A \times B_2$	$A \times B_3$	$A \times C_2$
0 -209.854	0 106.794	0 -105.62
< >	< >	< >

Operaciones de la tercer ecuación discriminante

$A \times B_3$	$A \times B_4$	$A \times C_3$
0 -209.854	0 121.416	0 -214.76
< >	< >	< >

Operaciones de la cuarta ecuación discriminante

$A \times B_4$	$A \times B_5$	$A \times C_4$
0 -209.854	0 113.492	0 -115.595
< >	< >	< >

Operaciones de la quinta ecuación discriminante

$A \times B_5$	$A \times B_6$	$A \times C_5$
0 -209.854	0 127.579	0 -108.25
< >	< >	< >

Operaciones de la sexta ecuación discriminante

$A \times B_6$	$A \times B_7$	$A \times C_6$
0 -209.854	0 118.1	0 -111.955
< >	< >	< >

Figura 4.24 La tercera pestaña de nuestra aplicación nos permite ver los coeficientes de cada una de las variables que componen a nuestro modelo de clasificación.

Discusión

- Sistema de adquisición analógico para EMG

En el capítulo 3 sección 3.2, se plantea un diseño para la adquisición de señales EMG, considerando las recomendaciones que realizan algunas investigaciones para realizar una adecuada adquisición. El diseño propuesto fue pensado en la portabilidad de los tableros de circuito impreso para que el dispositivo permita libertad de movimiento del usuario. La energía de funcionamiento, las dimensiones y el diseño de los filtros y amplificadores están redirigidos para complementar un diseño integral del dispositivo. Con el diseño de la tarjeta de adquisición de señales para electromiografía tenemos un sistema analógico capaz de realizar una adquisición de señales EMG en un rango de frecuencias de 10 a 480 Hz con una ganancia de 1000. La tarjeta quedó con unas dimensiones de 2.6 cm por 4 cm; dichas dimensiones principalmente se deben a los conectores utilizados, conectores que pueden ser eliminados al considerar la integración del sistema con su parte digital. Por lo tanto, el sistema será nuevamente reconfigurado con mejores productos de la electrónica que tienden a ser más pequeños, integrados multifuncionales, más rápidos y más baratos, pudiendo quedar con factores de forma más compactos. Las pruebas de adquisición de datos de este primer dispositivo fueron limitadas, se realizaron sólo en un ambiente de laboratorio controlado.

- Preprocesamiento y procesamiento

Dentro de los objetivos considerados en este trabajo estaba la evaluación de una base de datos de electromiografía que constaba de 15 movimientos de los dedos de la mano, el objetivo era crear un modelo de clasificación mejorado con un preprocesamiento definido. Se realizaron varias secuencias de preprocesamiento para evaluar el desempeño de estos métodos sobre las señales y su efecto sobre el porcentaje de reconocimiento. De igual forma se buscaba visualizar la participación muscular de los movimientos predefinidos y su reproducción en un ambiente biomecánico virtual, con el fin de generar nuevo conocimiento para la aplicación protésica. Los resultados nos muestran una utilidad en el entorno de simulación musculoesquelética. Se observaron virtualmente la evolución del movimiento junto a la participación muscular. Finalmente se compararon los datos de un movimiento ideal contra el movimiento clasificado.

El trabajo con múltiples movimientos específicos a partir de una base de datos proporcionó un sistema complejo multifuncional capaz de ser utilizado fuera de ambientes de laboratorio, con el cual podemos investigar de forma práctica el control de los sistemas protésicos de miembros superiores. Se creó una ruta viable estándar para el desarrollo de un clasificador con buenos parámetros de respuesta: porcentaje de reconocimiento, exactitud, especificidad, sensibilidad y f1 score. Se utilizó una ventana de datos pequeña para que el procesamiento de los datos tuviese lugar en un sistema embebido dentro de un microcontrolador en el futuro. A continuación, se presentan resultados específicos y se discuten las limitaciones del estudio.

El primer método de preprocesamiento considerado fue la eliminación de valores atípicos. Los resultados indican que el movimiento de la mano cerrada fue el más afectado, situación que puede llegar a ser lógica, ya que el movimiento de la mano cerrada fue el más complejo porque implicaba la flexión de todos los dedos; lo que provocaría el aumento de los valores atípicos en este conjunto de datos. Podríamos entonces cuestionarnos si es o no conveniente mantener o eliminar este movimiento de la base de datos considerada, pero sabemos que la mano cerrada es uno de los movimientos más comunes en las actividades de la vida diaria y al mismo tiempo fue uno de los movimientos mejor clasificados.

Después de la eliminación de valores atípicos, la creación de ventanas sobre los registros EMG crudos y el cálculo de las 96 características en cada una de las ventanas, se crearon 4 grupos de clasificadores como se presentó en la Figura 2.4. En la Tabla 4.5 (experimentación marcada con un círculo 5 en la Figura 2.4) se enumeran los resultados de los clasificadores cuando se utilizan la totalidad de las características (96) sin ningún tipo de preprocesamiento. Se observa que el mejor de los modelos de clasificación resulta ser LDA con una amplia ventaja sobre QDA. La Tabla 4.6 presenta los resultados cuando se realiza una selección de características antes de crear los modelos de clasificación, aquí el método que es utilizado es SFS y en función al porcentaje de reconocimiento se realiza la selección de características. Podemos notar que los porcentajes de reconocimiento aumentan, siendo el cambio más notable en QDA, alcanzando un 77.86% de reconocimiento de la matriz global de los sujetos y hasta un 94.73% con las matrices individuales de los sujetos. Las Tablas 4.7 y 4.8 muestran los efectos de la normalización sin y con selección de características. Es evidente que hay una mejoría cuando se le agrega a la selección de características a la normalización, sin embargo, para poder determinar si el aumento es

significativo se realizaron métodos de validación, hasta este momento podemos decir que la normalización trae una mejoría al modelo de QDA muy parecida a los resultados de SFS, y combinar tanto SFS como normalización aumentó solo unas décimas en el porcentaje. En la tabla 4.8, el mejor rendimiento para las 15 clases es el modelo creado con QDA, con un porcentaje de reconocimiento del 96.16% para sujetos individuales y un 78.36% para el grupo muestral, procesando las 96 características, incluso con las características con SFS y normalizadas se ve un aumento en la clasificación.

De un total de 96 características medibles, con el método de SFS se reducen las mediciones necesarias a 25 características-electrodos para los sujetos individuales y 49 para el conjunto total de sujetos. Un hecho interesante en los electrodos evaluados es que los que mayormente se vieron involucrados o permanecieron fueron los de la sección anterior de antebrazo eliminando los electrodos de la región posterior. Es decir, los electrodos 3, 4, 5 y 6 fueron los que mejor se comportaron, y corresponde a que están mejor posicionados para captar la señal de los músculos flexores activos. Este punto demostraría que si los movimientos que se evalúan son solo flexiones el contar con un anillo de electrodos equidistantes en antebrazo estaría de más.

Los resultados de cada procedimiento están respaldados por varios métodos de validación, desde el asegurarnos de que los resultados no estén comprometidos por la partición de los datos, se utilizó una validación cruzada. Para determinar si había diferencia entre las medias de los resultados y si esa diferencia era significativa se utilizó una prueba ANOVA junto a una prueba de comparaciones Múltiples de Duncan. Los tres últimos procedimientos experimentales (círculos 6, 7 y 8 en la Figura 2.4, o las tablas 4.6, 4.7 y 4.8) fueron estadísticamente no significativos, se determinó que el primer procedimiento según los resultados si muestra una diferencia significativa con los otros 3. Por lo tanto, podríamos utilizar cualquiera de estos tres últimos procedimientos, pero se decidió utilizar el método marcado por el círculo 8 en la Figura 2.4. que corresponde a la selección de características normalizadas.

Las Tablas 4.12 y 4.13 nos presentan los parámetros estadísticos calculados (sensibilidad, especificidad, precisión y score f1) para cada una de las clases dentro de la clasificación para los sujetos individuales y para el grupo muestral. Las formulaciones estadísticas de los ocho sujetos evaluados por separado (tabla 4.12) revelan que los movimientos HC (1), MRL (8), R (10) y TL (12) tienen todos unos porcentajes de reconocimiento del 100%. Este resultado indica que el

clasificador QDA entrenado para cualquier sujeto pudo distinguir estos cuatro gestos sin ningún error. Sin embargo, el reconocimiento perfecto de estas clases no se reproduce cuando los datos de los 8 sujetos son agrupados como un grupo muestral (tabla 4.13). En este grupo se mantiene una gran especificidad para todas las clases, pero los otros parámetros varían dependiendo la clase evaluada, el score f1 presenta un buen rendimiento para las clases HC (1) y T_L (12), confirmándolas como clases bien diferenciadas del resto. La clase T_M podría parecer tener una buena diferenciación ya que tiene la sensibilidad más alta, sin embargo, tiene la precisión como una de las más bajas.

En cuanto a las peores clases evaluadas para los sujetos evaluados individualmente (tabla 4.12) la flexión del dedo índice I_I (2) tuvo la menor sensibilidad y especificidad, lo que coincide con los resultados reportados en otra investigación donde también evaluaban el movimiento del índice frente a otros movimientos. Esto podría corresponder a los altos niveles de destreza motora fina que se desarrollan entre el pulgar y el índice como los dedos que mayormente utilizamos.

En cuanto al clasificador de todo el grupo muestral. la Tabla 4.13 presenta un F1 score que es superior al 63% con la mayoría de las clases equilibradas, pero la clase R_L (9) con la sensibilidad disminuida y la clase MR con la peor precisión. En estos dos movimientos que implican la flexión del dedo anular y la del dedo medio, se observó una clasificación errónea, técnicamente se confundían con un mismo movimiento.

También se analizó la conveniencia de convertir las características a un nuevo espacio, a través del método de matrices dispersas, con la idea de aumentar el reconocimiento de modelo clasificador y/o reducir las características consideradas. En la gráfica presentada en la Figura 4.10 podemos visualizar que este método permite enumerar las características principales desde la más importante hasta la menos significativa. No se observó ninguna mejora al utilizarlo por lo que no se justifica el aumento de la carga en el preprocesamiento con su uso en un microcontrolador.

- Simulación biomecánica del movimiento

Con cada predicción de movimiento por ventana, los modelos de clasificación van indicando los movimientos identificados, entonces fue cuestión de crear los vectores de avance para describir el movimiento indicado por el clasificador; lo que traduce cada movimiento identificado al modelo biomecánico Opensim de la mano. La clase del movimiento detectado proporciona a cada

articulación del modelo biomecánico de la mano un posible cambio en los grados de rotación. Al final de la lectura del movimiento de 5 segundos, se crea una matriz de movimiento virtual representativa de la clasificación. De esta manera fue posible reproducir los movimientos clasificados de cualquiera de los sujetos evaluados. Con las grabaciones de los movimientos ideales y de los resultados de las clasificaciones se pudo comparar las mejores clasificaciones y las peores, dándonos una idea de la eficiencia de los movimientos de la mano de forma real. Además, nos fue posible verificar la participación muscular en los movimientos clasificados y compararlos con la participación muscular ideal involucrada en los movimientos evaluados.

En los videos de los movimientos se observó que la clasificación errónea de un evento no afecta necesariamente al movimiento total generado. Por ejemplo, un solo error de un periodo de una ventana traducido en movimiento puede expresarse de la misma manera que un movimiento ejecutado al 98%, y quizás más si se consideran los movimientos correctos de algunas articulaciones dentro de los errores. Por ejemplo, si se produce una clasificación de la flexión del dedo índice medio y el movimiento real es solo la flexión del dedo índice, la clasificación está mal pero el movimiento generado incluirá el movimiento del dedo índice. Incluso en una mano normal cuando tramos de ejecutar movimientos específicos definidos como es el del dedo anular, nuestro sistema de control nervioso genera unos pequeños movimientos involuntarios en los otros dedos. Es por eso que creemos que algunos errores de movimiento pueden ser insignificantes al movimiento general real.

Hay algunas limitaciones que debemos tener en cuenta en este estudio. Aunque no hubo diferencias significativas en los resultados de las pruebas de SFS y SFS normalizado, se requiere un estudio más profundo a cerca del sistema donde será embebido, ya que con las características del microcontrolador seleccionado se definirán las características de las operaciones y del proceso general de procesamiento. De si se requiere trabajar con el mínimo número de electrodos o si es necesario agregar más movimientos a la base de datos seleccionada, para contemplar los movimientos de la mano en las actividades de la vida diaria. Una advertencia con las discusiones planteadas sobre la reducción de electrodos y la selección de características antes de su aplicación, es que siempre que se tenga un nuevo número de movimientos y numero de electrodos será necesario realizar un nuevo análisis para crear un nuevo clasificador. Por lo tanto, creemos que sería importante seguir experimentando con todo tipo de movimientos de los dedos para optimizar

estos clasificadores. Un argumento importante en los trabajos que involucran señales EMG es la repetibilidad de la adquisición de estas mediciones, ya que la minería de datos y el aprendizaje automático analizan la información a medida que se adquiere, por lo que un punto clave para obtener modelos de clasificación más fiables y robustos es asegurar las mismas condiciones del protocolo experimental durante cada medición además de poseer posiciones más definidas en antebrazo. Otra limitación del estudio fue que la base de datos no tenía sensores de movimiento ni grabaciones de video que nos permitieran tener una relación directa entre las señales EMG y la posición de las articulaciones involucradas en el movimiento.

Es bien conocido que los algoritmos de aprendizaje automático trabajarán con la información que les proporcionemos, está bien demostrado con los resultados de nuestra investigación cuando nos preocupamos por analizar el preprocesamiento de la información de las señales crudas del EMG y vimos que mientras la señal sea de mejor calidad aumenta la eficiencia de nuestros clasificadores, por lo tanto si tenemos calidad en las señales desde que las adquirimos veremos una mejoría en nuestros porcentajes de reconocimiento.

- Microcontroladores y el Sistema de adquisición Python en línea

El dispositivo seleccionado TM4C123GXL es una tarjeta de prueba del microcontrolador TM4C123GH6PM dicho microcontrolador tiene la capacidad de transferir datos desde sus puertos analógicos hasta sus puertos seriales con un tiempo de 1 microsegundo, dicha velocidad permite manipular los datos de entrada tanto para procesamiento como adquisición, las pruebas realizadas nos aseguran una velocidad de muestreo de 1000 muestras/segundo con una resolución de 8 bits por muestra, suficientes para censar las señales de electromiografía. Con las características del sistema de procesamiento específicamente con el tamaño de la ventana con una longitud de 62.5 ms nos permite realizar una transferencia y un procesamiento aún no evaluado. Aunque esta velocidad puede aumentar y se realizaron pruebas para aumentarla, se decidió trabajar con dicha frecuencia de muestreo de 1000 muestras por segundo realizando una simple transferencia de datos.

Como se observó se tiene un sistema de transferencia de datos simultáneos de 6 microcontroladores, a una velocidad de 1,000 muestras por segundo, velocidad que es posible manipular, agregando o reduciendo los otros parámetros evaluados. Se observó que el cuello de botella fue el sistema de conexión inalámbrica, donde los datos saturan el puerto serial de

adquisición, como no solo es una canal de transferencia sino 6 llegando a un módulo bluetooth concentrador.

Si bien hay posibles rutas alternas para solucionar este problema una es que los microcontroladores generen el procesamiento de las características al momento de estar adquiriendo las señales EMG crudas, que son simples operaciones matemáticas sobre la señal, eso implicaría una reducción de la información que está siendo transferida permitiéndonos aumentar las velocidades de adquisición y reduciendo la carga del módulo concentrador. Otra posible ruta es adaptar uno de los microcontroladores como concentrador y que se transmitan los datos sobre un mismo puerto serial de forma inalámbrica, ya que los problemas obtenidos no fueron por la carga de los datos sino por la cantidad de dispositivos conectados simultáneamente sobre el mismo modulo bluetooth concentrador, que reducía la cantidad de datos que podía manejar cada puerto serial independiente.

En un principio se tenía la intención de crear un microcontrolador concentrador, sin embargo, se propuso utilizar un equipo de cómputo para la adquisición con el objetivo de tener una mayor capacidad de procesamiento al momento de hacer la adquisición y de implementar los modelos de clasificación. Sin embargo, nos encontramos con estos inconvenientes.

El código desarrollado nos permite tener esas 6 conexiones seriales transmitiendo datos a un módulo concentrador simultáneamente.

Como trabajo futuro está el incorporar este sistema de adquisición a una tarjeta de adquisición digital con el microcontrolador TM4C123GH6PM. Se propone empezar a realizar operaciones en el microcontrolador (conversión a características) y verificar la velocidad de transferencia que nos permite obtener este dispositivo desde la adquisición hasta la digitalización en nuestro sistema concentrador y realizar una adquisición de las señales evaluando los 15 movimientos considerados en este estudio.

Conclusiones

Dentro de los objetivos de este trabajo era integrar los temas de un sistema de adquisición, un procesamiento con aprendizaje automático y una simulación de los movimientos en línea, plantear una ruta viable para lograr una integración de los tres temas evaluados. Aunque aún faltan algunos elementos para realizar una integración completa de las tres ramas se sentaron las bases para lograrlo.

En el desarrollo de un sistema de adquisición de señales EMG es necesario considerar los protocolos de adquisición, las normas y estándares para una adecuada representación de las señales EMG en el mundo digital.

Es conveniente realizar un preprocesamiento de los datos para generar una clasificación más eficiente.

El uso de electrodos en las posiciones de los músculos que están involucrados en el movimiento, permite que el modelo de clasificación tenga más oportunidades de realizar una mejor separación de las clases y por lo tanto una mejor clasificación de los movimientos.

El uso de base de datos para este tipo de investigaciones es útil ya que nos permite dividir las áreas de trabajo y nos permite trabajar paralelamente en el desarrollo de conocimiento.

El desarrollo presentado combina la simulación biomecánica con la clasificación automática de 15 movimientos de dedos. Dicha conexión se realizó a través de un sistema de vectores de movimiento para cada movimiento considerado el cual logra reproducir idealmente bajo el concepto de que todos los movimientos se realizaron a la misma velocidad en el tiempo definido. Lo que resulta ser una buena herramienta para el diseño práctico de prótesis de mano o para el control de la interfaz hombre-ordenador a través de los gestos de la mano.

Los algoritmos creados y su preprocesamiento de datos de entrada proporcionaron buenos resultados en la clasificación de los movimientos de los dedos

Los experimentos sugieren que es posible hacer un clasificador específico para una persona, utilizando solo 17 características con cuatro electrodos EMG.

Un sistema de simulación musculoesquelética que reproduzca los movimientos evaluados en una clasificación permite realizar validaciones musculares del movimiento considerado.

Referencias

1. GAMA MELO, Erika Nathalia; AVILES SANCHEZ, Oscar Fernando and AMAYA HURTADO, Darlo. Anthropomorphic robotic hands: a review. *Ing. Desarro.* [online]. 2014, vol.32, n.2 [cited 2021-12-06], pp.279-313. Available from: <http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S0122-34612014000200007&lng=en&nrm=iso>. ISSN 0122-3461.
2. Murray, Buchanan, Delp. "The isometric functional capacity of elbow muscles." *Journal of Biomechanics*, 2000.
3. Hossein Ghapanchizadeh, Siti A. Ahmad, and Asnor Juraiza Ishak. Recommended surface EMG electrode position for wrist extension and flexion. *ISSBES 2015 - IEEE Student Symposium in Biomedical Engineering and Sciences: By the Student for the Student*, pages 108112, 2016.
4. Marco Paleari, Michela Di Girolamo, Nicolò Celadon, Alain Favetto, and Paolo Ariano. On optimal electrode conguration to estimate hand movements from forearm surface electromyography. *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS, 2015-Novem:60866089, 2015.*
5. Ken Nishihara, Hisashi Kawai, Toshiaki Gomi, Miho Terajima, and Yu Chiba. Investigation of optimum electrode locations by using an automatized surface electromyography analysis technique. *IEEE Transactions on Biomedical Engineering*, 55(2):636642, 2008.
6. Yuto Konishi, Yusuke Mizobata, and Masaki Yoshida. Development of a system for finding best electrode position for myoelectric hand control for derating of upper limb amputee. *Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings*, 7:56995702, 2007.
7. Ivaylo Christov, Rositsa Raikova, and Silvija Angelova. Separation of electrocardiographic from electromyographic signals using dynamic ltration. *Medical Engineering and Physics*, 57:110, 2018.
8. Lara A. Green, Jessica Mcguire, and David A. Gabriel. Flexor carpi radialis surface electromyography electrode placement for evoked and voluntary measures. *Muscle and Nerve*, 52(5):818825, 2015.
9. Javier Rodriguez-Falces, Daria Neyroud, and Nicolas Place. Influence of inter-electrode distance, contraction type, and muscle on the relationship between the sEMG power spectrum and contraction force. *European Journal of Applied Physiology*, 115(3):627638, 2014.
10. Timothy J. Koh and Mark D. Grabiner. Evaluation of methods to minimize cross talk in surface electromyography. *Journal of Biomechanics*, 26(SUPPL. 1):151157, 1993.
11. David Hary, George A. Bekey, and Daniel J. Antonelli. Circuit Models and Simulation Analysis of Electromyographic Signal Sources???: The Impedance of EMG Electrodes. *IEEE Transactions on Biomedical Engineering*, BME-34(2):9197, 1987.
12. K. Nagata, K. Adno, K. Magatani, and M. Yamada. A Classication Method of Hand Movements Using Multi Channel Electrode. *2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, pages 23752378, 2005.
13. Matteo Beretta Piccoli, Alberto Rainoldi, Carolin Heitz, Marianne Wüthrich, Gennaro Boccia, Enrico Tomasoni, Carlo Spirolazzi, Michele Eglo, and Marco Barbero. Innervation zone locations in 43 supercial muscles: Toward a standardization of electrode positioning. *Muscle and Nerve*, 49(3):413421, 2014.
14. D K Kumar and A Melaku. Electrode Distance and magnitude of SEMG. *Engineering in Medicine and Biology, 2002. 24th Annual Conference and the Annual Fall Meeting of the Biomedical Engineering Society EMBS/BMES Conference, 2002. Proceedings of the Second Joint*, 3:2477 2480, 2002.
15. Aya Tomita, Ryosuke Ando, Akira Saito, Kohei Watanabe, and Hiroshi Akima. Eect of interelectrode distance on surface electromyographic signals of vastus intermedius muscle in women and men. *Journal of Electromyography and Kinesiology*, 25(6):841846, 2015.
16. Yong Ku Kong, M. Susan Hallbeck, and Myung Chul Jung. Crosstalk eect on surface electromyogram of the forearm exors during a static grip task. *Journal of Electromyography and Kinesiology*, 20(6):12231229, 2010.
17. M. Léoure, F. Quaine, and C. Servièrre. Testing of instantaneity hypothesis for blind source separation of extensor indicis and extensor digiti minimi surface electromyograms. *Journal of Electromyography and Kinesiology*, 23(4):908915, 2013.
18. C. Marque, C. Bisch, R. Dantas, S. Elayoubi, V. Brosse, and C. Pérot. Adaptive filtering for ECG rejection from surface EMG recordings. *Journal of Electromyography and Kinesiology*, 15(3):310315, 2005.
19. Yaohang Hu, Lesheng He, Jihong Shi, and Weilian Wang. A methodological approach to remove the ECG noise from the EMG signals. *Proceedings - IEEE 2011 10th International Conference on Electronic Measurement and Instruments, ICEMI 2011*, 3:228231, 2011.

20. Marko I. Vuskovic and X. Li. Blind separation of surface EMG signals. Proceedings of 18th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 4:14781480, 1996.
21. Md Ferdous Wahid, Reza Tafreshi, Mubarak Al-Sowaidi, and Reza Langari. Subject-independent hand gesture recognition using normalization and machine learning algorithms. *Journal of Computational Science*, 27:6976, 2018.
22. Scott H. Walbran, Emilio P. Calius, G. Reg Dunlop, and Iain A. Anderson. A technique for optimizing electrode placement for electromyographic control of prostheses. Proceedings of the 31st Annual International Conference of the IEEE Engineering in Medicine and Biology Society: Engineering the Future of Biomedicine, EMBC 2009, pages 13311334, 2009.
23. Minjae Kim and Wan Kyun Chung. Spatial sEMG Pattern-Based Finger Motion Estimation in a Small Area using a Microneedle-Based HighDensity Interface. *IEEE Robotics and Automation Letters*, 3(1):234241, 2017.
24. Heather Daley, Kevin Englehart, Levi Hargrove, and Usha Kuruganti. High density electromyography data of normally limbed and transradial amputee subjects for multifunction prosthetic control. *Journal of Electromyography and Kinesiology*, 22(3):478484, 2012.
25. Travis W. Beck, Terry J. Housh, Joel T. Cramer, and Joseph P. Weir. The effects of electrode placement and innervation zone location on the electromyographic amplitude and mean power frequency versus isometric torque relationships for the vastus lateralis muscle. *Journal of Electromyography and Kinesiology*, 18(2):317328, 2008.
26. Jiayuan He, Xinjun Sheng, Xiangyang Zhu, Chaozhe Jiang, and Ning Jiang. Spatial Information Enhances Myoelectric Control Performance with Only Two Channels. *IEEE Transactions on Industrial Informatics*, PP(c):1, 2018.
27. Ryan J. Smith, Francesco Tenore, David Huberdeau, Ralph Etienne Cummings, and Nitish V. Thakor. Continuous decoding of finger position from surface EMG signals for the control of powered prostheses. 2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, pages 197200, 2008.
28. D. Farina and C. Cescon. Concentric-ring electrode systems for noninvasive detection of single motor unit activity. *IEEE Transactions on Biomedical Engineering*, 48(11):13261334, 2001.
29. Karan Veer. A flexible approach for segregating physiological signals. *Measurement: Journal of the International Measurement Confederation*, 87:21 26, 2016.
30. Nienke W. Willigenburg, Andreas Daertshofer, Idsart Kingma, and Jaap H. van Dieën. Removing ECG contamination from EMG recordings: A comparison of ICA-based and other filtering procedures. *Journal of Electromyography and Kinesiology*, 22(3):485493, 2012.
31. R. V. Baratta, M. Solomonow, B. H. Zhou, and M. Zhu. Methods to reduce the variability of EMG power spectrum estimates. *Journal of Electromyography and Kinesiology*, 8(5):279285, 1998.
32. Zhongliang Yang and Yumiao Chen. Surface EMG-based Sketching Recognition Using Two Analysis Windows and Gene Expression Programming. 10(October):114, 2016.
33. L. Mesin, R. Merletti, and A. Rainoldi. Surface EMG: The issue of electrode location. *Journal of Electromyography and Kinesiology*, 19(5):719726, 2009.
34. Sang Wan Lee, Taeyoub Yi, Jin Woo Jung, and Zeungnam Bien. Design of a Gait Phase Recognition System That Can Cope With EMG Electrode Location Variation. *IEEE Transactions on Automation Science and Engineering*, 14(3):14291439, 2017.
35. Luciano L. Menegaldo. Real-time muscle state estimation from EMG signals during isometric contractions using Kalman filters. *Biological Cybernetics*, 111(5-6):335346, 2017.
36. Kohei Watanabe, Motoki Kouzaki, and Toshio Moritani. Effect of electrode location on task-dependent electromyography responses within the human biceps femoris muscle. *Journal of Applied Biomechanics*, 32(1):97100, 2016.
37. N A Chaya, B R Bhavana, S B Anooona, M Hiranmai, Niranjana Krupa B, Real-Time Replication of Arm Movements Using Surface EMG Signals, *Procedia Computer Science*, Volume 154, 2019, Pages 186-193, ISSN 1877-0509, <https://doi.org/10.1016/j.procs.2019.06.028>.
38. Chandrasekhar Potluri, Madhavi Anugolu, D. Subbaram Naidu, Marco P. Schoen, Steve C. Chiu, Real-time embedded frame work for sEMG skeletal muscle force estimation and LQG control algorithms for smart upper extremity prostheses, *Engineering Applications of Artificial Intelligence*, Volume 46, Part A, 2015, Pages 67-81, ISSN 0952-1976, <https://doi.org/10.1016/j.engappai.2015.08.007>.
39. S. Tam, G. Bilodeau, J. Brown, G. Gagnon-Turcotte, A. Campeau-Lecours and B. Gosselin, "A Wearable Wireless Armband Sensor for High-Density Surface Electromyography Recording," 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 2019, pp. 6040-6044, doi: 10.1109/EMBC.2019.8857750.

40. Xiupeng Gao, Yiwei Yin, Design on a wearable armband device for assessing the motion function of upper limbs, *Computer Communications*, Volume 153, 2020, Pages 135-144, ISSN 0140-3664, <https://doi.org/10.1016/j.comcom.2020.01.074>.
41. Khokhar, Z.O., Xiao, Z.G. & Menon, C. Surface EMG pattern recognition for real-time control of a wrist exoskeleton. *BioMed Eng OnLine* 9, 41 (2010). <https://doi.org/10.1186/1475-925X-9-41>
42. A. K. Das, V. Laxmi and S. Kumar, "Hand Gesture Recognition and Classification Technique in Real-Time," 2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN), 2019, pp. 1-5, doi: 10.1109/ViTECoN.2019.8899619.
43. Abu, Mohd Azlan & Rosleesham, Syazwani & Suboh, Mohd Zubir & Syazwan, Mohd & Yid, Md & Kornain, Zainudin & Jamaluddin, Nurul Fauzani. (2020). Classification of EMG signal for multiple hand gestures based on neural network. 17. 256-263. 10.11591/ijeecs.v17.i1.pp256-263.
44. Konstantin Akhmadeev, Elena Rampone, Tianyi Yu, Yannick Aoustin, Éric Le Carpentier. A realtime gesture classification using surface EMG to control a robotics hand. ENOC 2017, Jun 2017, Budapest, Hungary. fhal-01707679
45. Zhang, Z., Yang, K., Qian, J., & Zhang, L. (2019). Real-Time Surface EMG Pattern Recognition for Hand Gestures Based on an Artificial Neural Network. *Sensors (Basel, Switzerland)*, 19(14), 3170. <https://doi.org/10.3390/s19143170>.
46. Lashgari, E. & Maoz, U. Electromyography Classification during Reach-to-Grasp Motion using Manifold Learning, *bioRxiv*, Cold Spring Harbor Laboratory, 2020
47. Shim, H.-m.; An, H.; Lee, S.; Lee, E.H.; Min, H.-k.; Lee, S. EMG Pattern Classification by Split and Merge Deep Belief Network. *Symmetry* 2016, 8, 148. <https://doi.org/10.3390/sym8120148>.
48. U. Côté-Allard et al., "Deep Learning for Electromyographic Hand Gesture Signal Classification Using Transfer Learning," in *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 4, pp. 760-771, April 2019, doi: 10.1109/TNSRE.2019.2896269.
49. Jaramillo-Yáñez, A.; Benalcázar, M.E.; Mena-Maldonado, E. Real-Time Hand Gesture Recognition Using Surface Electromyography and Machine Learning: A Systematic Literature Review. *Sensors* 2020, 20, 2467. <https://doi.org/10.3390/s20092467>.
50. Yashodhan Athavale, Sridhar Krishnan, *Biosignal monitoring using wearables: Observations and opportunities*, *Biomedical Signal Processing and Control*, Volume 38, 2017, Pages 22-33, ISSN 1746-8094, <https://doi.org/10.1016/j.bspc.2017.03.011>
51. Visconti, Paolo & Gaetani, Federico & Zappatore, Giovanni & Primiceri, Patrizio. (2018). Technical Features and Functionalities of Myo Armband: An Overview on Related Literature and Advanced Applications of Myoelectric Armbands Mainly Focused on Arm Prostheses. *International Journal on Smart Sensing and Intelligent Systems*. 11. 1-25. 10.21307/ijssis-2018-005.
52. Liu, Xilin & Zhang, Milin & Richardson, Andrew & Lucas, Timothy & Van der Spiegel, Jan. (2016). The Virtual Trackpad: An Electromyography-Based, Wireless, Real-Time, Low-Power, Embedded Hand-Gesture-Recognition System Using an Event-Driven Artificial Neural Network. *IEEE Transactions on Circuits and Systems II: Express Briefs*. PP. 1-1. 10.1109/TCSII.2016.2635674.
53. Edward K. Chadwick, Dimitra Blana, Robert F. Kirsch, Member, IEEE, and Antonie J. van den Bogert. Real-Time Simulation of Three-Dimensional Shoulder Girdle and Arm Dynamics.
54. Antonie J. van den Bogert, Dimitra Blana, Dieter Heinrich, Implicit methods for efficient musculoskeletal simulation and optimal control, *Procedia IUTAM*, Volume 2, 2011, Pages 297-316, ISSN 2210-9838, <https://doi.org/10.1016/j.piutam.2011.04.027>.
55. SimTK Opensim. Available online: URL (<https://simtk.org/projects/opensim>)
56. SimTK Opensim: Proyect Crouchgait. Available online: URL (<https://simtk.org/projects/crouchgait>)
57. Laura Hallock, Akash Velu, Amanda Schwartz, and Ruzena Bajcsy. "Muscle deformation correlates with output force during isometric contraction." In *IEEE RAS/EMBS International Conference on Biomedical Robotics & Biomechatronics (BioRob)*. IEEE, 2020.
58. Gonzalez, R. V., Buchanan, T. S., Delp, S. L. How muscle architecture and moment arms affect wrist flexion-extension moments, *Journal of Biomechanics* Vol. 30, pp. 705-712, 1997.
59. SimTK Opensim, Proyect: DAS. Available online: URL (<https://simtk.org/projects/das>)
60. Slade, P.; Habib, A.; Hicks, J. L. & Delp, S. L., An open-source and wearable system for measuring 3D human motion in real-time, *bioRxiv*, 2021.r55.

61. Buschbacher, Ralph M. MD Anatomical Guide for the Electromyographer: The Limbs and Trunk, American Journal of Physical Medicine & Rehabilitation: March 2007 - Volume 86 - Issue 3 - p 204 doi: 10.1097/PHM.0b013e31803217b1
62. Analog devices AD8232. Available online: URL <https://www.analog.com/en/products/ad8232.html#product-overview>
63. UCI Machine Learning Repository. Available online: URL(<http://archive.ics.uci.edu/ml/datasets/sEMG+for+Basic+Hand+movements>).
64. IEE lab. Available online: URL (https://www.ufrgs.br/ieelab/IEE_sEMG_db.php)
65. OSFHOM SEEDS dataset. Available online: URL (<https://osf.io/wa3qk/>).
66. Computer Animation & Perception Group. Available online: URL (<http://zju-capg.org/research-en.html>).
67. EMG DATASETS REPOSITORY. Available online: URL (<https://www.rami-khushaba.com/electromyogram-emg-repository.html>)
68. The Ninapro project. Available online: URL (<http://ninapro.hevs.ch/>)
69. putEMG: sEMG Gesture and Force Recognition Datasets. Available online: URL (<https://biolab.put.poznan.pl/putemg-dataset/>)
70. Khushaba, R.N., Kodagoda, S.: Electromyogram (EMG) feature reduction using mutual components analysis for multifunction prosthetic fingers control, pp. 1534–1539. IEEE, December 2012 <https://doi.org/10.1109/icarcv.2012.6485374>.
71. Oskoei, M. A., & Hu, H. (October 2007). Myoelectric control systems—A survey. Biomedical Signal Processing and Control, 2 (4), 275-294.
72. Lei, M., Wang, Z., & Feng, Z. (2001). Detecting nonlinearity of action surface EMG signal. Physics Letters A, 290(5–6), 297–303.
73. Angkoon Phinyomark, Pornchai Phukpattaranont, Chusak Limsakul. (15 June 2012). Feature reduction and selection for EMG signal classification. Expert Systems with Applications, 39 (8), 7420-7431.
74. J. Han, M. Kamber and J. Pei, Data Mining - Concepts and Techniques, 3rd ed., Amsterdam: Morgan Kaufmann Publishers, 2012.
75. Lisboa, P.J.G., Ellis, I.O.: Cluster-based visualisation with scatter matrices. Pattern Recogn. Lett. 29, 1814–1823 (2008)
76. Arduino DUE. Available online: URL <https://store-usa.arduino.cc/products/arduino-due?selectedStore=us>.
77. Code Composer Studio IDE. Available online: URL <https://www.ti.com/tool/CCSTUDIO>.
78. Keil uvision IDE. Available online: URL <https://www2.keil.com/mdk5/uvision/>.
79. Matlab. Available online: URL <https://www.mathworks.com/products/matlab.html>.
80. National instruments. Labview. Available online: URL <https://www.ni.com/es-mx/shop/labview.html>
81. Python. Available online: URL <https://www.python.org/>
82. Delsys trigno avanti Sistema trigno lite. Available online: URL <http://www.biomedicale.parisdescartes.fr/pf-sensorimotricite/wp-content/uploads/2014/04/Trigno-Users-Guide.pdf>
83. Bluetooth. Available online: URL <https://components101.com/wireless/hc-06-bluetooth-module-pinout-datasheet>.
84. Nordic Semiconductor. Available online: URL <https://www.nordicsemi.com/About-us>
85. SimTK Opensim. Available online: URL (<https://simtk.org/projects/opensim>)
86. <https://simtk-confluence.stanford.edu/display/OpenSim/Tutorial+3+-+Scaling%2C+Inverse+Kinematics%2C+and+Inverse+Dynamics>.
87. SimTK Opensim, Wrist model. Available online: URL (<https://simtk.org/projects/wrist-model>)
88. Amezcua-García J.A., Bravo-Zanoguera M.E., González-Navarro F.F., López-Avitia R. (2020) Hand Movement Detection from Surface Electromyography Signals by Machine Learning Techniques. In: González Díaz C. et al. (eds) VIII Latin American Conference on Biomedical Engineering and XLII National Conference on Biomedical Engineering. CLAIB 2019. IFMBE Proceedings, vol 75. Springer, Cham. https://doi.org/10.1007/978-3-030-30648-9_29
89. Lisboa, P.J.G., Ellis, I.O.: Cluster-based visualisation with scatter matrices. Pattern Recogn. Lett. 29, 1814–1823 (2008)

90. Jiawei Han, Micheline Kamber, Jian Pei, 8 - Classification: Basic Concepts, Editor(s): Jiawei Han, Micheline Kamber, Jian Pei, In The Morgan Kaufmann Series in Data Management Systems, Data Mining (Third Edition), Morgan Kaufmann, 2012, Pages 327-391, ISBN 9780123814791, <https://doi.org/10.1016/B978-0-12-381479-1.00008-3>.
91. Jose Alejandro Amezcuita Garcia, Valoración de registros de señales electromiográficas de antebrazo para el control de un sistema antebrazo/mano virtual, tesis, acceso en línea: <https://drive.google.com/file/d/1UCbynjxSXE4fj26Edc-Y7Gf4NYgJGH L/view>
92. Amezcuita García, J.A.; Bravo Zanoguera, M.E.; Lopez Avitia, R.; Reyna Carranza, M.A.; Cuevas-Gonzales, D. Opensim Visualization of the classification of finger movements based on electromyography signal as the single input variable during predefined movements., in Proceedings of the 8th International Electronic Conference on Sensors and Applications, 1–15 November 2021, MDPI: Basel, Switzerland, doi:10.3390/ecsa-8-11262.
93. https://www.instructables.com/Recover-Misconfigured-HC-06-Bluetooth-Module/?fbclid=IwAR2Iy3kXyx31W9y_Xs2mhVuZPvEWsxXPnlgJLSS-sgK9r0qXu4fjG0J3M70
94. <https://simtk.org/projects/das>
95. https://github.com/eschearer/DAS_Schearer/tree/master/2011%20Version
96. SimTK Opensim. Available online: URL (<https://simtk.org/projects/moving-fingers>).