



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE CIENCIAS DE LA ENERGÍA Y
MECÁNICA**

CARRERA DE INGENIERÍA MECATRÓNICA

**TRABAJO DE TITULACIÓN, PREVIO A LA OBTENCIÓN DEL
TÍTULO DE INGENIERO EN MECATRÓNICA**

**TEMA: DISEÑO Y CONSTRUCCIÓN DE UN MÓDULO
MECATRÓNICO ADAPTABLE QUE PERMITA LA MOVILIDAD
DE SILLAS DE RUEDAS, CONTROLADA POR SEÑALES
ELECTROENCEFALOGRÁFICAS, PARA LA EMPRESA
ALGORITMUN CIA. Ltda.**

AUTORES: ALDÁS ARIAS ALEX SEBASTIÁN

VANEGAS JADÁN, JUAN ANDRÉS

DIRECTOR: MSc. SÁENZ ENDERICA, FABIÁN GUSTAVO

SANGOLQUÍ

2016



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE

CARRERA DE INGENIERÍA MECATRÓNICA

CERTIFICACIÓN

Certifico que el trabajo de titulación, “DISEÑO Y CONSTRUCCIÓN DE UN MÓDULO MECATRÓNICO ADAPTABLE QUE PERMITA LA MOVILIDAD DE SILLAS DE RUEDAS, CONTROLADA POR SEÑALES ELECTROENCEFALOGRÁFICAS, PARA LA EMPRESA ALGORITMUN CIA. Ltda.” realizado por: ALEX SEBASTIÁN ALDÁS ARIAS y JUAN ANDRÉS VANEGAS JADÁN ha sido revisado en su totalidad y analizado por el software anti-plagio, el mismo cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de Fuerzas Armadas ESPE, por lo tanto me permito acreditarlo y autorizar a los señores: ALEX SEBASTIÁN ALDÁS ARIAS y JUAN ANDRÉS VANEGAS JADÁN para que lo sustenten públicamente.

Sangolquí, 24 de junio del 2016

Atentamente,

MSc. Fabián Gustavo Sáenz Enderica

DIRECTOR



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

CARRERA DE INGENIERÍA MECATRÓNICA

AUTORÍA DE RESPONSABILIDAD

Nosotros , ALEX SEBASTIÁN ALDÁS ARIAS, con cédula de identidad N° 1804491817 y JUAN ANDRÉS VANEGAS JADÁN , con cédula de identidad N° 0105079669 declaramos que este trabajo de titulación **“DISEÑO Y CONSTRUCCIÓN DE UN MÓDULO MECATRÓNICO ADAPTABLE QUE PERMITA LA MOVILIDAD DE SILLAS DE RUEDAS, CONTROLADA POR SEÑALES ELECTROENCEFALOGRÁFICAS, PARA LA EMPRESA ALGORITMUN CIA. Ltda.”** ha sido desarrollado considerando los métodos de investigación existentes, así como también se ha respetado los derechos intelectuales de terceros considerándose en las citas bibliográficas.

Consecuentemente declaro que este trabajo es de mi autoría, en virtud de ello me declaro responsable del contenido, veracidad y alcance de la investigación mencionada.

Sangolquí, 24 de junio del 2016

Alex Sebastián Aldás Arias

CI: 1804491817

Juan Andrés Vanegas Jadán

CI: 0105079669



ESPE

UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE

CARRERA DE INGENIERÍA MECATRÓNICA

AUTORIZACIÓN

Nosotros, **ALEX SEBASTIÁN ALDÁS ARIAS** y **JUAN ANDRÉS VANEGAS JADÁN**, autorizamos a la Universidad de las Fuerzas Armadas ESPE publicar en la biblioteca Virtual de la institución el presente trabajo de titulación **“DISEÑO Y CONSTRUCCIÓN DE UN MÓDULO MECATRÓNICO ADAPTABLE QUE PERMITA LA MOVILIDAD DE SILLAS DE RUEDAS, CONTROLADA POR SEÑALES ELECTROENCEFALOGRÁFICAS, PARA LA EMPRESA ALGORITHMUN CIA. Ltda.”** cuyo contenido, ideas y criterios son de mi autoría y responsabilidad.

Sangolquí, 24 de junio del 2016

Alex Sebastián Aldás Arias

CI: 1804491817

Juan Andrés Vanegas Jadán

CI: 0105079669

DEDICATORIA

Este trabajo va dedicado a todas las personas que han estado pendiente de mi vida universitaria especialmente a mi familia y a quienes compartieron su conocimiento conmigo.

Alex Sebastián

Para todas las mentes que no paran de aprender y solo aplican sus conocimientos para hacer de este un mundo mejor.

Juan Andrés

AGRADECIMIENTOS

Especialmente agradezco a mi madre Alicia Arias quien se ha esforzado e impulsado mis estudios desde el inicio, a mi padre quien me ha inculcado valores, mis hermanos por darme su apoyo en los momentos más difíciles y a todas las personas que forman parte de mi vida y que están día a día caminando y progresando junto a mí. Un agradecimiento muy especial a la persona que conocí al estar casi finalizando la carrera y quien me ha dado mucha felicidad y apoyo sobre todo a estado pendiente de mí, gracias por compartir su vida conmigo, gracias mi nua Erica.

Alex Sebastián

En primer lugar, a Dios que siempre me llevó en el camino correcto, para plasmar con ingenio ideas formidables. Al pilar de mi vida, por siempre confiar en mí, cuidarme a la distancia, brindarme su inmenso amor, por saber que siempre cuento con ella, porque mis sueños son sus sueños, y juntos hemos luchado por ellos y por todo lo que hace por mí, a mi mamita Nelly. Por sus concejos sabios, amor y apoyo incondicional, por su puesto por el infaltable pancito de dulce, y por todo lo que ha hecho por mí, a mi Bibi. Por formar la personalidad y carácter que me ha hecho llegar muy lejos, a mi papá Juan Manuel. A mi diseñadora favorita, porque que siempre me ha abierto la imaginación hacia nuevas ideas, a mi ñañita Andrea. A la motivación para ser cada día un mejor hombre, a mi hermosa Camila. A mi cuantiosa familia, porque sé que cuento con cada uno de ellos siempre. Y, por último, pero para nada menos importante, a quien me ha acompañado de la mano desde el primer día que llegue a la universidad, a quién me ha curado de mis un poco frecuentes gripes, a quien me ha sacado una sonrisa en los días más estresantes, ha llorado conmigo en los días más frustrantes, a mi Verito Alexandra.

Juan Andrés

INDICE GENERAL

CARÁTULA	i
CERTIFICACIÓN	ii
AUTORÍA DE RESPONSABILIDAD	iii
AUTORIZACIÓN	iv
DEDICATORIA	v
AGRADECIMIENTOS	vi
INDICE GENERAL	vii
INDICE DE TABLAS	x
INDICE DE FIGURAS	xi
RESUMEN	xvi
ABSTRACT	xvii
ASPECTOS GENERALES	1
1.1 Introducción	1
1.2 Antecedentes	1
1.3 Justificación e importancia	5
1.4 Objetivos	6
1.4.1 Objetivo general	6
1.4.2 Objetivos específicos	6
1.5 Alcance	6
1.6 Estructura del documento	6
MARCO TEÓRICO	8
2.1 Señales Bioeléctricas	8
2.1.1 Historia	8
2.1.2 Sistema Nervioso (SN)	9
2.1.3 El encéfalo	9
2.1.4 Medula Espinal	10
2.1.5 Los ganglios basales	12
2.1.6 Nervios	12
2.2 La Neurona	12
2.2.1 La Sinapsis	13
2.2.2 La Células Gliales.	13
2.2.3 La membrana neuronal	14
2.3 Potencial de acción	16
2.3.1 Introducción al potencial de acción.	16
2.3.2 Principio de funcionamiento	16
2.4 El potencial de reposo	19
2.5 Equilibrio electroquímico	20
2.5.1 Ecuación de Nernst	21
2.5.2 Modelamiento núcleo-conductor.	21
2.5.3 Modelamiento eléctrico para una fibra excitable simple	21
2.6 Tipos de Señales Bioeléctricas	23

2.6.1	Electrocardiograma ECG	23
2.6.2	Electromiografía EMG	25
2.6.3	Electroencefalograma EEG	27
2.7	Sistema de posicionamiento de electrodos	28
2.7.1	Nomenclatura del sistema 10-20	29
2.8	Procesamiento Digital de Señales	33
2.8.1	Filtros Digitales	34
2.8.2	Transformada discreta de Fourier	35
2.9	Reconocimiento de patrones.	38
2.9.1	Redes Neuronales	39
2.10	Tipos de discapacidades	42
2.11	Sillas de Ruedas	45
2.12	Tipos de control de movimiento de sillas de ruedas	49
2.12.1	Consideraciones previas	49
2.12.2	Control por Voz	49
2.12.3	Control por movimientos	49
2.12.4	Control con la mano	49
2.12.5	Control con movimientos oculares	50
2.12.6	Control mediante señales Electroencefalográficas (EEG)	50
DISEÑO Y CONSTRUCCIÓN		51
3.1	Introducción	51
3.2	Requerimiento y limitaciones	51
3.3	Diseño mecánico	53
3.3.1	Requerimientos mecánicos	53
3.3.2	Diseño	53
3.4	Diseño Electrónico	87
3.4.1	Requerimientos y selección de componentes.	87
3.5	Diseño del sistema de control	112
3.5.1	Requerimientos	112
3.5.2	Recepción e interpretación de instrucciones.	112
3.5.3	Control del motor de la rueda de tracción	122
3.5.4	Control del motor de dirección	127
3.5.5	Control motor de frenado	128
CONSTRUCCIÓN E IMPLEMENTACIÓN		129
4.1	Construcción e implementación mecánica	129
4.1.1	Maquinado del perfil de soporte	129
4.1.2	Construcción del Trinche	129
4.1.3	Construcción de engranes cónicos	130
4.1.4	Construcción de caja para elementos electrónicos	131
4.1.5	Construcción de canasta para la batería	132
4.1.6	Implementación los componentes del módulo mecatrónico	132

4.2 Implementación de elementos electrónicos	138
4.3 Implementación y pruebas del sistema de control	141
4.3.1 Recepción e interpretación de instrucciones de señales EEG.	141
4.3.2 Implementación del control del motor de la rueda de tracción	166
4.3.3 Implementación del control del motor de dirección	168
4.3.4 Implementación control motor de frenado	168
4.3.5 Funcionamiento en conjunto del Sistema de control	169
ANÁLISIS ECONÓMICO	170
5.1 Análisis de costos	170
CONCLUSIONES Y RECOMENDACIONES	172
6.1 Conclusiones	172
6.2 Recomendaciones	168
REFERENCIAS BIBLIOGRÁFICAS	170
ANEXOS	177

INDICE DE TABLAS

Tabla 1 Parámetros iniciales para los engranes.....	79
Tabla 2: Parámetros de diseño, engrane de 15 dientes.....	79
Tabla 3: Parámetros de diseño, engrane de 45 dientes.....	80
Tabla 4: Fuerzas que actúan en un engrane cónico.....	81
Tabla 5 Especificaciones técnicas del motor DC brushless.....	90
Tabla 6 Especificaciones técnicas generales del driver del motor DC brushless	91
Tabla 7 Especificaciones de la batería	93
Tabla 8 Especificaciones técnicas del motor paso a paso	94
Tabla 9 Características eléctricas del driver M542	96
Tabla 10 Características servomotor HS 805MG	98
Tabla 11 Datos enviados por el sensor Mindwave Neurosky	114
Tabla 12 Código y longitud de los tipos de carga útil de datos.	115

INDICE DE FIGURAS

Figura 1 . Control de sillas de ruedas con ondas electroencefalográficas.....	2
Figura 2 Módulo Lazaro.....	4
Figura 3 Distribución del sistema nervioso.....	9
Figura 4 Partes del encéfalo	10
Figura 5 Partes de la médula espinal.....	11
Figura 6 Membrana Celular	15
Figura 7 Dispersión del potencial de acción a través del axón	17
Figura 8 La permeabilidad del sodio aumente respecto a la del potasio.....	19
Figura 9 Equilibrio electroquímico	20
Figura 10 Modelamiento Núcleo conductor para una fibra simple en un espacio extracelular restringido.	22
Figura 11 Representación de una membrana de fibras cilíndricas de longitud Δx	22
Figura 12 Membrana ante una excitación que ha superado el umbral	23
Figura 13: Representación de latidos cardíacos	24
Figura 14: Composición de ondas del electrocardiograma	24
Figura 15: Unidad motora y sus componentes.....	26
Figura 16: Esquema de una señal Electromiográfica.....	27
Figura 17: Ondas EEG	28
Figura 18: Marcas sobre el cráneo	29
Figura 19: Punto frontal polar y occipital al 10% de distancia.....	29
Figura 20: Puntos Fz (Frontal), Pz (Pariental), Cz (Central o Vertex).	30
Figura 21: Puntos temporales mediales, T3 y T4.....	30
Figura 22: Puntos C3 y C4.....	31
Figura 23: Posición de los electrodos F3 y F4	31
Figura 24: Posición de los electrodos P3 y P4	32
Figura 25: Posición de los electrodos FP1, FP2, O1 Y O1	32
Figura 26: Posición de los electrodos A1 y A2.....	33
Figura 27: Sistema Internacional 10-20 para colocación de electrodos.....	33
Figura 28 Estructura básica de una red neuronal	39
Figura 29 Función sigmoid.	40
Figura 30 Estructura de una red neuronal multicapa y clasificador multiclases.....	40
Figura 31 Discapacidad de acuerdo a las extremidades afectadas.	44
Figura 32 posicionamiento vertical de la rueda	46
Figura 33 : Posicionamiento horizontal de la rueda.....	47
Figura 34 Altura del asiento	48
Figura 35 Altura del reposapiés	49
Figura 36 Sensor Mindwave Neurosky.....	52
Figura 37 Rueda con motor brushless interno.	52
Figura 38: Modelos comunes de sillas de ruedas.....	53
Figura 39: Medidas estándar del ancho del asiento	54
Figura 40: Perfil de soporte.....	54
Figura 41: Ranuras laterales.....	55
Figura 42: Ranuras superiores.....	55

Figura 43: Sujeción vertical	56
Figura 44: Eje roscado para regular la altura	56
Figura 45: Rodamientos XR	57
Figura 46: Eje de giro para la rueda	57
Figura 47: Montaje de rodamientos y perfil.....	58
Figura 48: Unión de tubo con arco.....	58
Figura 49: Sujetador con ángulo de inclinación.....	59
Figura 50 Cálculo de reacciones	59
Figura 51: Área de contacto de la rueda en el peor de los caso	61
Figura 52 Movimiento del Diferencial.....	62
Figura 53 Cálculo del torque del motor	63
Figura 54 Cálculo de reacciones sobre los soportes	64
Figura 55: Factor de seguridad en eje roscado.....	65
Figura 56 Cálculo de fuerzas en los pernos	65
Figura 57: Factor de seguridad en perno M5	66
Figura 58: Sujeción horizontal y sus partes	67
Figura 59: Geometría del soporte.....	67
Figura 60: Factor de seguridad para el soporte con un espesor de 3mm	68
Figura 61: Limite elástico para un espeso de 4mm	69
Figura 62: Factor de seguridad para el soporte con un espesor de 4mm	69
Figura 63: Cargas respecto al momento cortante.....	70
Figura 64: Factor de seguridad y esfuerzo de von mises para un espesor de 3mm ...	70
Figura 65: Factor de seguridad y esfuerzo de von mises para un espesor de 4mm ...	71
Figura 66: Geometría del arco.....	72
Figura 67: Limite elástico sobre el arco.....	73
Figura 68: Factor de seguridad del arco	73
Figura 69: Esfuerzos de Von Mises generados sobre el arco.....	74
Figura 70: Factor de seguridad mínimo que tiene el arco.....	74
Figura 71: Unión del disco con la rueda	75
Figura 72: Montaje de la rueda con el trinche.....	76
Figura 73: Montaje principal sobre el perfil	76
Figura 74: Unión de la mordaza con el trinche	77
Figura 75: Dimensiones de los engranes cónicos	78
Figura 76: Acoplamiento de engranes.....	78
Figura 77: Engranes diseñados de 15 y 45 dientes	83
Figura 78: Acople piñón y motor a pasos	84
Figura 79: Acoplamiento de engranes.....	84
Figura 80: Servomotor y soportes para el accionamiento del freno.....	85
Figura 81: Caja para protección	85
Figura 82: Caja para elementos electrónicos	86
Figura 83 Raspberry Pi 2	88
Figura 84 Características Raspberry Pi modelo 2 y B+	88
Figura 85 Adaptador Bluetooth USB Trendnet Versión 1.0.....	89
Figura 86 Distribución de pines de la Raspberry Pi 2.....	89
Figura 87 Cables de conexión del motor DC brushless	91

Figura 88 Conexiones del driver del motor CD brushless.	92
Figura 89 Batería del sistema.....	93
Figura 90 Motor paso a pasos Nema 23 con caja de reducción de 15:1.....	95
Figura 91 Driver M542 del motor paso a paso.	96
Figura 92 Servomotor HS 805MG.....	99
Figura 93: Diagrama de comunicación entre componentes.....	99
Figura 94: Sección de un motor brushless y posición de los sensores Hall.....	100
Figura 95: Señal de sensores Hall.....	101
Figura 96: Fases del circuito de acondicionamiento.....	101
Figura 97: Configuración Darlington.....	102
Figura 98: Circuito para el opto-transistor.....	103
Figura 99: Circuito para schmitt trigger.....	104
Figura 100: Diodo Zener y carga.....	105
Figura 101: Circuito de acondicionamiento.....	105
Figura 102: Conversión de PWM a voltaje DC.....	106
Figura 103: Filtro RC.....	107
Figura 104: Circuito recomendado para controlar el Driver M542m.....	108
Figura 105: Circuitos para control de PUL, DIR (giro) Y ENA.....	108
Figura 106: Circuito de 3.3V a 5V para señal del servomotor.....	109
Figura 107: Circuito final.....	110
Figura 108: Ruteado PCB.....	111
Figura 109: Vista 3D de la tarjeta de control.....	111
Figura 110 Esquema general del Sistema de control.....	112
Figura 111 Estructura de la red neuronal.....	119
Figura 112 Diagrama de flujo de la interpretación de comandos.....	121
Figura 113 Respuesta del motor brushless ante una entrada escalón.....	122
Figura 114 Salida del modelo simulado, junto con el medido.....	124
Figura 115 Respuesta ante una entra impulso y escalón de un motor DC brushless.....	124
Figura 116Parametros del modelamiento del motor DC brushless.....	125
Figura 117 Respuesta del sistema con un control PID.....	125
Figura 118 Parámetros obtenidos para el control PID.....	125
Figura 119Secuencia de pulsos del control de señales.....	128
Figura 120 Señal de entrada para el control del motor paso a paso.....	128
Figura 121: Perfil mecanizado.....	129
Figura 122: Tubo y arco antes de soldar.....	130
Figura 123: Unión de tubo y arco por soldadura.....	130
Figura 124: Soldadura de sujetadores.....	130
Figura 125: Engranés cónicos de 15 y 45 dientes.....	131
Figura 126: Montaje de engranes.....	131
Figura 127: Caja para elementos electrónicos.....	132
Figura 128: Canasta para la batería.....	132
Figura 129: Rodamientos y trinche.....	133
Figura 130: Trinche con el perfil.....	133
Figura 131: Guía de cable.....	134

Figura 132: Motor a pasos y piñón	134
Figura 133: Montaje de engrane de 45 dientes	135
Figura 134: Acople de piñón y engrane	135
Figura 135: Servomotor para frenado	136
Figura 136: Disco para frenado.....	136
Figura 137: Mordaza y cable	136
Figura 138: Sujeción de ejes roscados	137
Figura 139: Montaje de elementos electrónicos.....	137
Figura 140: Montaje del módulo mecatrónico	138
Figura 141: construcción de placa de control	139
Figura 142: ubicación de los elementos electrónicos.....	139
Figura 143: Distribución del cableado	140
Figura 144: Cableado de motores	140
Figura 145 Software SD Formatter	142
Figura 146 Software Win32 Disk Imager	142
Figura 148 Activación de la Secure Shell de la Raspberry Pi.....	143
Figura 149 Software para control remoto Putty	143
Figura 150 Software para la transferencia de archivos FileZilla	144
Figura 151 Código de conexión del electrodo Mindwave con Matlab	145
Figura 152 Adquisición de datos con Matlab del sensor Mindwave.	146
Figura 153 Comando para la instalación de librerías de conexión Bluetooth	146
Figura 154 Comandos para el emparejamiento del electrodo con la Raspberry Pi .	147
Figura 155 Funciones librería electrodo para muestreo de señales del electrodo Mindwave.....	148
Figura 156 Función recorrido del promedio, para filtrar una señal en Matlab.	149
Figura 157 Filtro recorrido de promedio implementada en C.	149
Figura 158 Onda cruda obtenida por el sensor antes de ser filtrada	150
Figura 159 Onda obtenida luego de aplicar el filtro de respuesta finita al impulso.	150
Figura 160 Eventos de pestañeo en la onda cruda.	151
Figura 161 Eventos de pestañeo agrupados.	152
Figura 162 Onda de eventos de pestañeos, aplicado el filtro de respuesta al impulso finito	152
Figura 163 Onda patrón para la detección de un pestañeo.....	153
Figura 164 Esquema del algoritmo para verificación de pestañeo	153
Figura 165 Gesto con la mejilla izquierda. A la izquierda sin filtrar, a la izquierda filtrada	154
Figura 166 Giro de cabeza a la derecha. A la izquierda sin filtrar, a la izquierda filtrada	154
Figura 167 Doble pestañeo. A la izquierda sin filtrar, a la izquierda filtrada	155
Figura 168 Guiño derecho. A la izquierda sin filtrar, a la izquierda filtrada	155
Figura 169 Primera a la cuarta componente de la transformada de Fourier, categorizadas por tipo de gesto.	156
Figura 170 Quinta a la octava componente de la transformada de Fourier, categorizadas por tipo de gesto.	157

Figura 171 Novena a la doceava componente de la transformada de Fourier, categorizadas por tipo de gesto.	157
Figura 172 Décima tercera a la décima sexta componente de la transformada de Fourier, categorizadas por tipo de gesto.	158
Figura 173 Vigésimo primero al vigésimo cuarto componente de la transformada de Fourier, categorizadas por tipo de gesto.....	158
Figura 174 Valores Máximos y mínimos de la onda	159
Figura 175 Media y desviación estándar de los valores de la onda	160
Figura 176 Función para medir la suavidad de la onda.	160
Figura 177 Suavidad de la onda en total y en los últimos valores.	161
Figura 178 Número de picos en alto y bajo	161
Figura 179 Suma de los valores picos en alto y bajo	162
Figura 180 Media de la localización de los picos en alto y bajo.....	162
Figura 181 Implementación del escalamiento en lenguaje de programación C.....	163
Figura 182 Red neuronal Implementada	164
Figura 183 Función tangente hiperbólica sigmoidea	164
Figura 184 Función softmax.	164
Figura 185 Rendimiento del entrenamiento de la red neuronal.	165
Figura 186 Implementación de la red neuronal en la Raspberry Pi.	166
Figura 187 Diagrama de Control PID para el motor DC Brushless.....	167
Figura 188 Procesos implementados para el control de velocidad del motor brushless	167
Figura 189 Código para el giro de la rueda.....	168
Figura 190 Uso de la clase GPIO.PWM	169
Figura 191 Esquema general, sistema de control.....	169

RESUMEN

El proyecto desarrollado consiste en el diseño y construcción de un módulo mecatrónico adaptable a diferentes modelos y tamaños de sillas de ruedas, el cual es controlado por señales electroencefalográficas (EEG). El módulo construido cuenta con un diseño compacto y de fácil adaptabilidad que posee tres actuadores que se encargan del desplazamiento, dirección y frenado. Fue construido aplicando los conocimientos adquiridos tales como diseño de elementos de máquinas, procesos de manufactura, sistemas CAD/CAM/CAE entre otros a lo largo de la carrera. Mientras que en el área de electrónica se hizo uso de sistemas de control, electrónica de potencia, diseño electrónico, programación y automatización etc. El módulo es controlado mediante una red neuronal la cual recibe, procesa y analiza las señales provenientes de un sensor de señales EEG para luego generar las instrucciones necesarias que llaman a varios procesos simultáneos que comandan la silla de ruedas. El proyecto se realizó con fines de investigación orientado para mejorar la calidad de vida de personas que utilizan sillas de ruedas debido a una discapacidad motriz, también se pretende dejar bases sólidas para que se pueda desarrollar futuros proyectos.

Palabras clave:

- **SEÑALES ELECTROENCEFALOGRÁFICAS**
- **RED NEURONAL**
- **MECATRÓNICA**
- **CAE**
- **CAM**

ABSTRACT

The present project consists in the design and the making of a mechatronic module which can be adapted to different kinds and sizes of wheelchairs. The module is commanded by electroencephalographic (EEG) waves and it has a compact design and an easy adaptation. Three actuators handle the displacement, direction, and braking. It was built by acquired knowledge in the fields of design of machine elements, manufacture process, and CAD/CAM/CAE systems. In the electronic fields, we used control system, power electronic, programming, automation and other fields. The system core is a neural network which acquires, processes and analyzes brain waves from a EEG biosensor. The neural network provides instructions to run simultaneously every process which allows mechatronic module to move wheelchairs. This project was built with research purposes for improve the quality lives of people with mobility impairment and need to use wheelchairs. The main purpose of this project is to establish firm foundations for use in future projects.

Keywords:

- **ELECTROENCEPHALOGRAPHIC (EEG) WAVES**
- **NEURAL NETWORK**
- **MECHATRONIC**
- **CAE**
- **CAM**

CAPÍTULO 1

ASPECTOS GENERALES

1.1 Introducción

El presente capítulo justifica la razón por la cual se diseñará y construirá un módulo mecatrónico controlado por señales electroencefalográficas, se muestra una reseña de proyectos relacionados que se han llevado a cabo en el país y se definen los objetivos que nos ayudarán a establecer el alcance del proyecto.

1.2 Antecedentes

Actualmente se realizan proyectos que permiten el control de sillas de ruedas para personas discapacitadas, mediante diversas formas, por ejemplo: detectar movimientos por visión artificial o sensores (acelerómetros, giroscopio, etc.), ondas electromiográficas, por nombrar algunas. Tenemos el caso de la SENA (Servicio Nacional de Aprendizaje) en Colombia que desarrolla un proyecto para ayudar a personas discapacitadas de escasos recursos brindando sillas eléctricas que son comandadas por el movimiento ocular (RCN, 2015).

Las sillas de ruedas fueron cambiando constantemente. Primero teníamos sillas netamente mecánicas y poco ergonómicas, donde el discapacitado no tenía comodidad, y por ello se dedicó tiempo en mejorar su ergonomía. A pesar de que el discapacitado se sentía más cómodo tenía que realizar mucho esfuerzo para su desplazamiento, por lo que se comenzó a integrar motores eléctricos comandados por botoneras o palancas. Hoy en día se intenta reemplazar la forma de comandar las sillas, porque existe una gran variedad de discapacidades motrices en las personas. Una de las soluciones dirigidas a este tema son las ondas electroencefalográficas, en este campo se pretende con pensamientos comandar las sillas de ruedas.

A nivel mundial las investigaciones en torno a las señales electroencefalográficas tienen extensos estudios. Por ejemplo en la Escuela Politécnica Federal de Lausana un grupo de investigadores liderados por el profesor José del R. Millán en Suiza se encuentran realizando un proyecto para el control de sillas de ruedas con ondas electroencefalográficas (EEG), enteramente con el pensamiento, hasta el momento el movimiento no es tan preciso pero si se tiene un

mejor manejo a comparación de otros proyectos. A más de que se activa con ondas EEG tiene sistemas de seguridad para detección de objetos mediante visión artificial. El proyecto todavía se encuentra en desarrollo pero se ve muy prometedor. (Escuela Politécnica Federal de Lausana, 2015).



Figura 1 . Control de sillas de ruedas con ondas electroencefalográficas.

Fuente: (Escuela Politécnica Federal de Lausana, 2015)

A nivel regional pudimos apreciar que las investigaciones en torno a las EEG se las desarrolla en el ámbito de la Ingeniería Biomédica. Por ejemplo tenemos el artículo científico publicado en la revista "Avances en sistemas e informática" de la Universidad Nacional de Colombia, titulado "Máquinas de aprendizaje para clasificar señales electroencefalográficas" (Guillén, 2009) realizado en la Universidad de los Andes (2009). Este artículo trata sobre la detección de epilepsia en pacientes, utilizando el método de aprendizaje automático SVM (Supported Vector Machine) dicho método ayuda a poder clasificar datos cuando se tienen muchos parámetros a analizar o existen variaciones en los datos difíciles de detectar y está dentro de la rama de Aprendizaje Automático que es una de las ramas que aplicaremos en nuestro proyecto de grado.

A nivel nacional se puede encontrar poca información de investigaciones realizadas en el campo de las ondas electroencefalográficas, la más destacada podría ser la tesis de grado realizada en la ciudad de Cuenca por el Ingeniero Sergio Guevara

(2012) para la Universidad Politécnica Salesiana que se titula "Adquisición de señales electroencefalográficas para el movimiento de un prototipo de silla de ruedas en un sistema bci (Brain Computer interfaz)", a breves rasgos trata sobre la adquisición de ondas cerebrales mediante el dispositivo Emotiv EPOC headset que localiza electrodos en la parte exterior de la cabeza. El SDK (Software Development Kit) que incluye este dispositivo permite mediante una interfaz gráfica la detección de intenciones del usuario como jalar empujar girar, y movimientos como pestañeo giros de cabeza. Tomando las intenciones de usuario que permite detectar el SDK es como se procede a mover la silla de ruedas. El resultado no fue tan bueno debido a que se tenían movimientos pausados de la silla de ruedas y necesitaban de mucha concentración de las personas.

En la actualidad a más de intentar comandar sillas de ruedas de distintas maneras, se intenta adaptar motores eléctricos a las sillas que poseen las personas discapacitadas, debido a que una gran mayoría poseen sillas simplemente mecánicas y el gasto para la compra de una silla eléctrica sería para muchos de ellos imposible. Por ejemplo, la empresa "Lázaro" de Chile, ha desarrollado un kit eléctrico para sillas de ruedas mecánicas, este es un módulo adaptable que mediante un motor acoplado a una llanta brinda la tracción necesaria para movilizar a la silla. Un inconveniente en este kit es que la parte de dirección todavía sigue siendo manual, ya que cuenta con un manubrio para ello. (Barrera, Rios, & Olivera, 2014).



Figura 2. Módulo Lazaro.

Fuente: (Barrera, Rios, & Olivera, 2014)

1.3 Justificación e importancia

Según estima el Informe mundial sobre la discapacidad, publicado por el Banco Mundial y la Organización Mundial de la Salud (2011) más de mil millones de personas viven con algún tipo de discapacidad -alrededor del 15% de la población mundial. Y para nuestro país el porcentaje de personas con alguna discapacidad está alrededor del 12.1% de la población. A pesar de que los últimos años se ha prestado mayor atención a dicha población, se han enfocado en brindar apoyo económico, mas no en brindar soluciones para aumentar su independencia. Tenemos el caso de la creación de la fundación Manuela Espejo que se transformó en la Secretaría Técnica para la Gestión Inclusiva en Discapacitadas (SETEDIS). Se ha entregado según la SETEDIS (2015) hasta el momento 543694 ayudas técnicas como productos y dispositivos de apoyo a personas con discapacidad, entre estos: sillas de ruedas, andadores, bastones, muletas, tiflotecnología y kits visuales. A pesar del apoyo brindado aún las personas discapacitadas requieren de ayuda en su diario vivir. Es por todo ello que esta población requiere de una mayor atención que la que se tiene actualmente.

Hemos visto que parte de la población con discapacidad son personas que han perdido motricidad en sus extremidades inferiores o en todas sus extremidades. Para brindar una ayuda se desarrollará el proyecto dirigido hacia este sector.

En nuestro país las personas con pérdida de motricidad en sus extremidades inferiores son de bajos recursos económicos según la SETEDIS (2015) esta población cuenta con silla de ruedas donadas por fundaciones o adquiridas a través de ayuda económica. Por dicha razón el módulo mecatrónico se diseñará para adaptarse fácilmente a cualquier silla ayudando a que los beneficiarios no tengan que adquirir una nueva y generando un ahorro en su economía.

Los campos de estudio del aprendizaje automático, y ondas electroencefalográficas no han sido muy explorados en nuestro país, no se encuentran muchos proyectos o artículos científicos enfocados a estas áreas. Siendo una de las razones de la investigación y aplicación que se pretende generar.

Debido a que una gran población ecuatoriana con capacidades especiales cuenta con sillas de ruedas simplemente mecánicas, donde es necesario la ayuda de

otra persona para poder moverse. No es común encontrar a una persona discapacitada desplazarse en sillas de ruedas eléctricas, es por eso que vemos la necesidad de brindar una mayor independencia a estas personas a través de este proyecto.

En base a todo lo mencionado la empresa Algoritmun Cia. Ltda. ha visto la necesidad de crear un producto innovador dirigido esta población, para mejorar su estilo de vida, así como su independencia al momento de moverse en su vida cotidiana. Además, con este producto la empresa pretende ser un referente nacional en el desarrollo de tecnología para estas áreas.

1.4 Objetivos

1.4.1 Objetivo general

- Diseñar y construir un módulo mecatrónico que permita la movilidad de sillas de ruedas, controlada por ondas electroencefalográficas.

1.4.2 Objetivos específicos

- Diseñar y construir un sistema mecánico que se acople a sillas de ruedas, y que contenga mecanismos para el desplazamiento, giro y frenado de las sillas.
- Diseñar e implementar un sistema electrónico que optimice el funcionamiento del módulo mecatrónico
- Implementar un control en el mecanismo de desplazamiento.
- Implementar un control en el mecanismo de giro.
- Implementar un control en el mecanismo de frenado.
- Implementar un sistema de control comandado por señales EEG.
- Ensamblar todos los componentes de control dentro del módulo mecatrónico optimizando espacio.

1.5 Alcance

El alcance del proyecto será la creación de un módulo mecatrónico que se adapte y permita dirigir una silla de ruedas mediante ondas electroencefalográficas generadas por el movimiento de la cabeza.

1.6 Estructura del documento

El documento consta de 6 capítulos. El primer capítulo detalla el estado del arte así como se justifica la razón por la cual se realiza el proyecto, se definen los objetivos para establecer el alcance. El segundo capítulo sintetiza las formas de monitorización

y adquisición de la actividad cerebral y se da un mayor enfoque a las señales EEG. Así como también se da una descripción del manejo de dichas señales. También se describe los diferentes tipos de control para el movimiento de sillas de ruedas. El tercer capítulo se enfoca en el proceso de diseño tanto mecánico como electrónico así como el procesamiento de señales neuronales. El cuarto capítulo muestra los resultados obtenidos en el funcionamiento de cada una de sus partes así como en conjunto. El quinto capítulo está orientado al análisis económico. En el capítulo final detalla el cumplimiento de proyecto por medio de conclusiones y recomendaciones.

CAPÍTULO 2

MARCO TEÓRICO

El presente capítulo da una visión de los fundamentos teóricos en que se desenvolverá el proyecto, para dar un mayor entendimiento de los capítulos siguientes. El capítulo se divide en dos grandes partes. La primera parte del capítulo trata de las señales electroencefalográficas y todo lo que conlleva, desde su historia, la actividad cerebral, los métodos adquisición, para así poder llegar al procesamiento de dichas señales y finalmente tener los métodos de clasificación de patrones. La segunda parte del capítulo muestra los problemas que tienen las personas con capacidades especiales, centrándonos en las que usan sillas de ruedas y los tipos de soluciones que se han dado para el control de movimiento de dichas sillas.

2.1 Señales Bioeléctricas

2.1.1 Historia

A lo largo de los tiempos el hombre ha estado interesado en saber cómo una persona o un animal puede controlar su cuerpo, como realmente funciona su sistema. La electricidad es uno de los pilares fundamentales para explorar dicho funcionamiento. Tenemos por ejemplo en 1773a Galvani estudiando la reacción de los músculos ante estímulos eléctricos. Y así otros científicos que intentaban encontrar la relación de la electricidad con el cuerpo humano.

Un paso muy importante dio en 1870 dos neurólogos alemanes Fritsch y Hitzig al demostrar las contracciones musculares podían ser producidas por pulsos eléctricos en la corteza cerebral de un perro. Demostrando que existe una parte del cerebro motora. Tras ellos hubieron varias investigaciones, pero hay que destacar que todo esto se hacía en cerebros descubiertos es decir invasivamente(Patterson & Kesner, 2013)

El primer paso hacía las ondas electroencefalográficas la dio el psiquiatra y neurólogo alemán en 1928, afirmando que existen ondas eléctricas producidas por el cerebro, ideando un método para obtenerlas y dividiendo ya dichas ondas en alfa y beta (Millet, 2001).

Luego de ello se ha seguido avanzando en torno a las ondas electroencefalográficas, es amplia la historia e investigaciones que nos han colocado

en donde el día de hoy nos encontramos. Lo más importante a destacar es que a pesar de los años transcurridos y las importantes investigaciones, todavía existe un gran campo que no se ha explorado.

2.1.2 Sistema Nervioso (SN)

El sistema nervioso, es de los más complejos sistemas del cuerpo humano, y por ello existe un gran desconocimiento de su completo funcionamiento. Está constituido por tejido nervioso, en donde se integran células neuronales y gliales. Su función principal es la de comunicar todas las regiones del organismo.

El sistema nervioso consta de dos partes: el sistema nervioso central (SNC) compuesto por la médula espinal y el encéfalo, y el sistema nervioso periférico constituido por ganglios basales, nervios craneales y espinales (raquídeos).

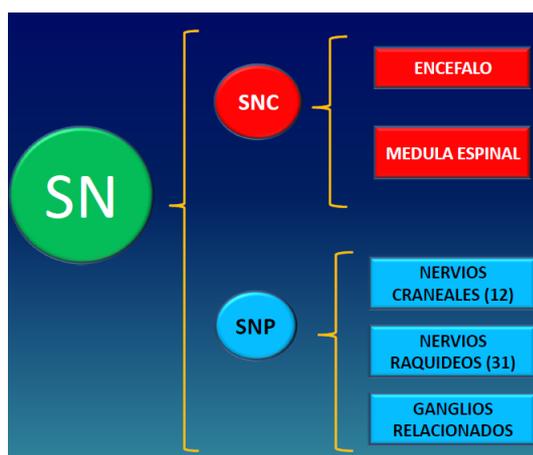


Figura 3. Distribución del sistema nervioso.

Fuente: (Leucoma, 2010)

2.1.3 El encéfalo

El encéfalo recibe todas las señales provenientes del sistema nervioso, realiza un análisis y otorga una respuesta permitiéndole controlar algunas funciones del organismo, para realizar sus funciones el encéfalo está formado por: bulbo raquídeo, tronco cerebral, el cerebro y el cerebelo, todos estos son protegidos por el cráneo. (Morales, 2011).

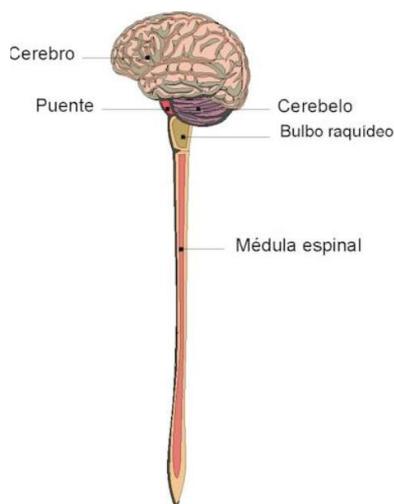


Figura 4. Partes del encéfalo.

Fuente: (Leucoma, 2010)

2.1.4 Médula Espinal

Médula espinal es un cordón nervioso el cual cumple con dos funciones principales en el organismo. Primero produce reflejos simples y compuestos cómo reaccionar ante picaduras o quemaduras. Segundo es el camino por el cual viaja la información entre el cerebro y el cuerpo o viceversa. Está formado por sustancia blanca y gris, la blanca está compuesta por axones mielinizados localizados en la parte externa, y la sustancia gris por compuesta por axones no mielinizados y localizados en la parte interna. (Logroño, 2010).

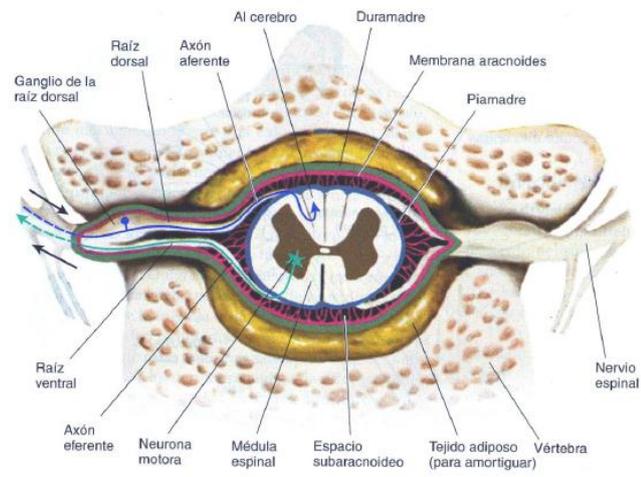


Figura 5. Partes de la médula espinal.

Fuente: (Morales, 2011)

2.1.5 Los ganglios basales

Los ganglios basales son un grupo de neuronas ubicadas cerca la base del cerebro, cumple con funciones motoras y no motoras. Las motoras se encargan de la coordinación e iniciación de los movimientos, mientras que las no motoras regulan el sueño, el estado emocional y la conducta. Los ganglios basales normalmente son afectados por enfermedades como Parkinson y alteraciones motoras las cuales generan movimientos involuntarios en la persona. (Kandel, 2001)

2.1.6 Nervios

Los nervios craneales están constituidos por 12 pares de nervios los cuales están relacionados con la interpretación de los 5 sentidos: gusto, visión, olfato, tacto, audición. También influyen con las funciones motoras y control muscular. (Jesús Borges, 2005)

Los nervios espinales conformados por 31 pares, se forman de dos raíces provenientes de la medula, una raíz sensitiva y una motora. La raíz motora acciona a los músculos con el fin de generar movimientos comandados por el cerebro, en cambio la raíz sensitiva toma y muestrea información del cuerpo para luego enviarla al cerebro. (Bertran, 2009)

2.2 La Neurona

Para poder estudiar el sistema nervioso, es esencial entender el funcionamiento de las neuronas. Un hombre en promedio posee 100,000 millones de neuronas interconectadas entre sí. Cada neurona recibe estímulos eléctricos, llamados también impulsos nerviosos, provenientes de otras neuronas y envía una respuesta hacia otras. Para ello las neuronas están estructuradas en tres partes principales. El cuerpo principal llamado Soma, el cuál para recibir información tiene extensiones celulares llamadas dendritas, y para enviar una respuesta posee una extensión celular llamado Axón. (Dierssen, 2007)

Existen tres tipos de neuronas distribuidas a lo largo del cuerpo, el primer tipo están conectados a receptores para poder detectar y responder ante estímulos exteriores (temperatura, luz, sonidos, etc.) y son las llamadas neuronas sensoriales. El segundo tipo de neuronas son los encargados de controlar la actividad muscular y son las

neuronas motoras. El último tipo son las encargadas de interconectar neuronas pero nunca están en contacto ni con receptores sensoriales ni con fibras musculares y reciben el nombre de neuronas interneuronas. (Morris & Fillenz, 2003)

2.2.1 La Sinapsis

Debido a que cada neurona representa un solo cuerpo, para la conexión entre ellas se da el proceso llamado sinapsis. Cada neurona en promedio tiene 1000 de dichas conexiones. La neurona que envía información recibe el nombre de neurona presináptica y la que recibe la información se le llama postsináptica.

La sinapsis se da de dos tipos una eléctrica y otra química. Para la sinapsis eléctrica entre las membranas de la neurona presináptica y postsináptica se forma una unión comunicante, esta unión es de baja resistencia permitiendo así el paso de iones, ya sea de carga positiva o negativa el flujo de la corriente va desde la neurona presináptica hacia la postsináptica. En la sinapsis química no existe conexión directa, si no que la neurona presináptica libera sustancias químicas, principalmente Calcio, haciendo reaccionar a receptores en la membrana postsináptica. Aunque la liberación de las sustancias se puede dar de forma repentina, en su mayoría se da cuando existe un potencial de acción, el cual abre canales en la membrana de la neurona presináptica permitiendo el paso de Calcio y haciendo reaccionar a los receptores de la siguiente neurona(Merino & Noriega, 2011)

2.2.2 La Células Gliales.

Las células gliales de manera general son las encargadas de adecuar el medioambiente en donde se desenvuelven las neuronas. La cantidad de células gliales es de 10 a 50 veces mayor a la cantidad de neuronas. Existen cuatro tipos de células gliales son: la microglía, oligodendroglía, glía NG2 y la astrogía.

Las células gliales de tipo microglía son las encargadas de hacer que el cerebro conserve su integridad, impidiendo el paso de cualquier célula extraña incluso de los linfocitos. Las de tipo oligodendroglía son las encargadas de mejorar la transmisión de los impulsos eléctricos entre neuronas, produciendo mielina para aumentar la velocidad de transmisión. El tipo de células gliales NG2 representa del 5 al 8 por ciento de las células gliales, y es del cual se tiene menor información, existen estudios contradictorios entre si ayudan a la producción de oligodendrocitos, neuronas, o

astrocitos. El tipo de células astroglia, de igual manera no tan conocido, ayudaría a la homeostasis del cerebro, aunque según estudios recientes también estaría involucrado en la sinapsis. (Reyes-Haro, Bulavina, & Pivneva, 2014)

2.2.3 La membrana neuronal

La membrana que recubre a la neurona, está formada de dos capas lipídicas (conjunto de moléculas orgánicas, principalmente carbono e hidrogeno), estas capas tienen como función aislar la neurona del medio extracelular y regular la entrada y salida de compuestos.

A más de las capas lipídicas la membrana neuronal, la membrana contiene mecanismos que permiten el paso de cierto tipo de moléculas, dependiendo del mecanismo. Estos mecanismos atraviesan las dos capas lipídicas, formando canales, por lo que reciben el nombre de canales iónicos. Existiendo dos tipos de canales, unos que se mantienen abiertos, y otros que pueden cambiar su estado permitiendo o impidiendo el paso de los iones. Los canales que pueden cambiar su estado son sensibles a estímulos, como, por ejemplo: al voltaje transmembranal (Diferencia de voltaje entre la parte externa y la interna), a señales físicas o químicas. La existencia de canales abiertos implica que la membrana tiene una permeabilidad dependiente del número de canales abiertos.(Martínez, 2010)

Otro mecanismo de transporte iónico, son las bombas iónicas. Se diferencian de los canales iónicos en que estas sí necesitan energía metabólica para su funcionamiento. (Roseto, Repetto, & Exeni, 2003)

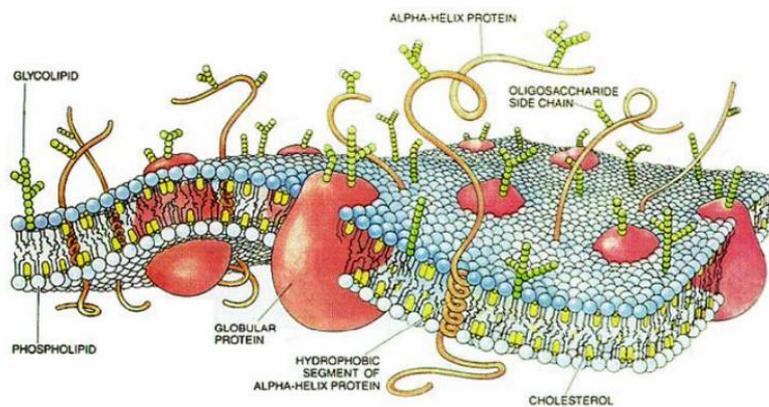


Figura 6. Membrana Celular.

Fuente: (Martínez, 2010)

2.3 Potencial de acción

2.3.1 Introducción al potencial de acción.

Las bombas iónicas y los canales iónicos generan una concentración de iones diferente de la parte interior con la del exterior de la neurona, esto produciendo el voltaje transmembranal. Se tiene aquí un potencial de reposo que se da cuando el voltaje se estabiliza. Para obtener un impulso nervioso o potencial de acción, se debe estimular a la neurona adecuadamente logrando que se transmita el voltaje de forma ondulatoria.

Al haber un estímulo eléctrico adecuado, los canales sensibles al voltaje se abren, permitiendo el paso de iones en ambos sentidos dependiendo de la carga de los iones. El potencial de acción desplaza iones de adentro hacia afuera, no a lo largo del axón, es por esto que el voltaje se desplaza en forma ondulatoria. Es decir, el potencial de acción de acción genera una reacción en cadena, abriendo los canales iónicos a medida que se desplaza la diferencia de voltaje transmembranal.

En este proceso existen tres características importantes. Primero se tiene un umbral el cual tiene que ser excedido para que se genere la excitación necesaria en la membrana, para abrir los canales iónicos, y activar la bomba sodio potasio. Segunda característica es que la respuesta es de tipo todo o nada, es decir que una vez superado el umbral, la respuesta generada es la misma. Tercera característica es el periodo refractario, es decir que una vez excitada la neurona, pasa un tiempo antes de poder ser excitada nuevamente.

2.3.2 Principio de funcionamiento

Las neuronas sensoriales y motoras realizan actividades de dos formas: química y eléctrica, los axones transmiten y contactan a las dendritas por medio de señales químicas, las dendritas reciben y transforman esta señal de química a eléctrica, esta señal se integra a través del cuerpo celular al resto de señales procedentes de otras sinapsis, esta señal indica a las otras señales de la sinapsis si se propaga o no a la siguiente neurona. Los potenciales eléctricos viajan desde el axón hasta las sinapsis, pasando a las dendritas de la neurona siguiente, luego de esto el proceso se repite.

Las neuronas están constituidas por proteínas que se propagan hacia las dendritas y los axones, las dendritas suelen cambiar de forma para dar lugar a nuevas

conexiones y eliminan otras existentes, mientras que los axones suelen formar nuevos caminos cuando una neurona requiere comunicarse con otras neuronas que la rodean de manera precisa o intensa.

Las neuronas para comunicarse entre ellas requieren que la señal se propague a largo de los axones los cuales transmiten pulsos eléctricos conocidos como *potenciales de acción*, estos pulsos provienen de la energía almacenada en forma de gradientes químicos y físicos los cuales son combinados de forma adecuada.

Cuando una dendrita recibe un potencial de axón se crea corrientes eléctricas pequeñas, las corrientes que se dirigen hacía las células son excitatorias, y las que se dirigen hacia fuera de la célula son inhibitorias. Las corrientes se acumulan en las dendritas y se distribuyen por el cuerpo celular, cuando estas corrientes se suman y superan el umbral de actividad la neurona envía un mensaje a otras neuronas que la rodean, en cambio cuando no se superan el umbral las corrientes mueren y no ocurre nada.

La influencia de los mensajes o señales entre neuronas forman parte de los pensamientos, sensaciones, movimientos, etc. depende de red neuronal en la que se localizan.

Cuando una neurona recibe señales, estas perturban el voltaje transmembranal de la célula que recibe, este voltaje cambia la permeabilidad de la membrana desatando el proceso de excitación. Esta descripción es conocida coma la teoría del circuito global.

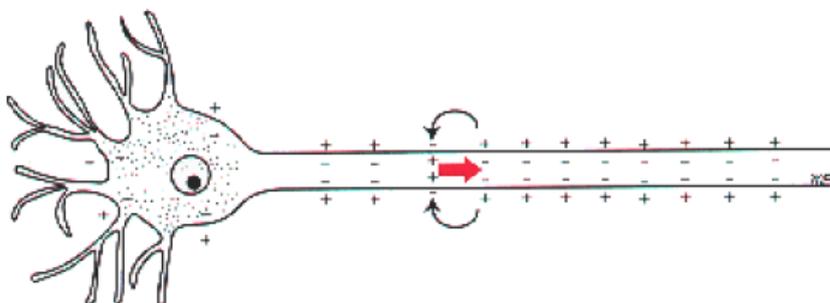


Figura 7. Dispersión del potencial de acción a través del axón.

Fuente: (García, 2005)

2.3.2.1 Canales de potasio

Los canales de potasio son sensibles al voltaje transmembranal, tienen una compuerta la cual esta sincronizada con un sensor que la abre teniendo una permeabilidad en **actividad** cuando se tiene una variación suficientemente grande, esta compuerta se mantiene siempre y cuando el campo eléctrico se mantenga modificado. Una vez que el campo eléctrico regresa a la normalidad es decir al valor que tiene en reposo, la compuerta se cierra provocando que la permeabilidad se reduzca a valores rangos que establecen los canales de potasio que siempre están abiertos y que no tiene compuerta.

2.3.2.2 Canales de Sodio

Existe una baja permeabilidad al sodio cuando la membrana del axón está en reposo, pero también se tiene una gran población de canales de sodio sensibles al voltaje. Los canales de sodio cuentan con dos compuertas: la de **inactivación** y la de **activación**, el canal se mantiene en reposo cuando esta desactivado, es decir se encuentra cerrada la compuerta de activación, en cambio el canal esta desactivado cuando está abierta la compuerta de **inactivación**. El tránsito de sodio ocurre cuando la compuerta de activación se abre y el voltaje ha superado cierto valor umbral, siempre que la variación o cambio de voltaje se mantenga la compuerta permanecerá abierta. Las dos compuertas son independientes, después de que el canal se activa la compuerta de inactivación se cierra en pocos milisegundos después, cortando el flujo de sodio. Los canales de sodio cuentan con 3 estrados: activado, desactivado e inactivado.

2.3.2.3 La bomba de sodio, potasio(Na/K)

La actividad neuroeléctrica depende de las concentraciones de potasio y sodio, las cuales son cambiadas por la proteína transmembranal que utiliza energía del metabolismo para realizar esta función. LA bomba de NA/K provoca que el potencial de acción retorne al reposo, debido a que la bomba devuelve los iones a su localización inicial. (García, 2005)

El peso de sodio se ve reflejado cuando aumenta la permeabilidad del mismo por lo que la concentración de los canales se mueve hacia la del sodio, este fenómeno suceda cuando la permeabilidad del sodio es alta con respecto a la del potasio. ´

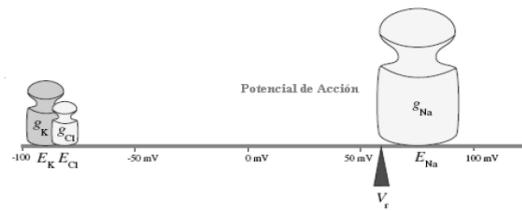


Figura 8. La permeabilidad del sodio aumenta respecto a la del potasio.

Fuente: (García, 2005)

Una onda de excitación es generada cuando se alcanza un valor umbral de potencial en un punto de la membrana y el potencial de acción se desata en esta zona, a lo largo del axón existe un pequeño flujo de corriente pasiva la cual provoca que el potencial en las zonas aledañas tome un valor umbral obligando a que los canales sensibles al voltaje se abran.

2.4 El potencial de reposo

Cuando una neurona se encuentra estable durante un tiempo moderado, podemos decir que la membrana está en reposo. La diferencia de voltaje que existe entre el interior y el exterior de axón se conoce como el potencial de reposo, el cual representa el valor del voltaje transmembranal en equilibrio, todas las membranas celulares cuentan con una permeabilidad la cual es originada por la presencia de canales. El flujo de información en el sistema nervioso es provocado por los cambios bruscos de potenciales de acción, por lo que es complicado conocer porque se originan los potenciales de reposo. (Lamas, 2005)

La ecuación de Nernst no calcula bien el potencial de reposo debido a que solo toma en cuenta solo los iones de potasio, pero en el potencial de reposo influyen tanto los iones de potasio, sodio y cloro, para solucionar este problema se puede calcular el potencial de reposo a través de concentraciones iónicas, para ello se toma en cuenta todos los iones cuando la membrana es permeable en situación de reposo, esto lo hicieron Goldman, Hodgkin y Katz. (L. Hodgkin, 1952)

A continuación, se muestra la ecuación:

$$V_r = \frac{RT}{F} \ln \frac{P_K[K]_e + P_{Na}[Na]_e + P_{Cl}[Cl]_i}{P_K[K]_i + P_{Na}[Na]_i + P_{Cl}[Cl]_e} \quad (2.5.1)$$

Fuente: (L. Hodgkin, 1952)

Donde:

V_r : potencial de potasio el cual se mide en voltios.

P : permeabilidad de cada ion.

$[X]_e$: concentración extracelular para cada ión.

$[X]_i$: concentración intracelular para cada ión.

Como resultado de esta ecuación nos da la media del potencial de reposo para todos los iones que atraviesen las membranas.

2.5 Equilibrio electroquímico

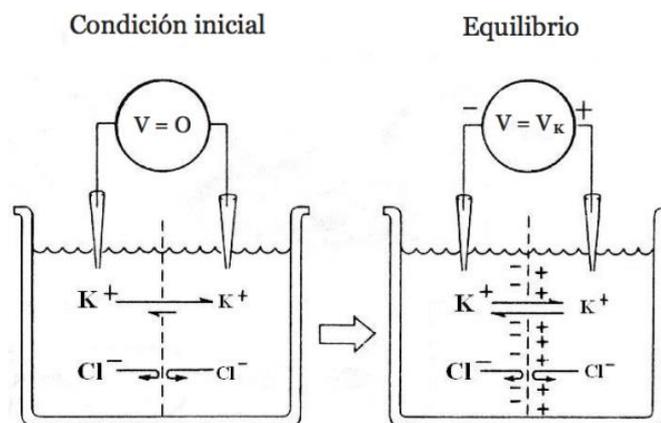


Figura 9. Equilibrio electroquímico.

Fuente: (Huaster, 2002)

Para poder explicar el equilibrio electroquímico, supóngase que se tiene un recipiente con agua dividido por una membrana neuronal con un solo canal de iones de potasio. En la figura 2 se tiene el esquema. Supóngase que al lado izquierdo se introduce una concentración de cloruro de potasio que es el doble que al lado derecho. Al insertar el cloruro de potasio en el agua los iones de cloro y potasio se disocian. Debido a que la concentración es mayor del lado izquierdo los iones de potasio por fuerza de difusión se pasaran al lado derecho a través del canal de potasio, mientras que los iones de cloruro no podrían pasar. Esto genera que el lado izquierdo quede cargado negativamente y el lado derecho positivamente, generando un voltaje entre ambos lados. Existe un punto en donde los iones de potasio ya no pasan por la repulsión

de la carga positiva del lado derecho. Este equilibrio entre la fuerza eléctrica y fuerza de difusión se conoce como equilibrio electroquímico.

2.5.1 Ecuación de Nernst

Cuando se ha alcanzado el equilibrio químico, el voltaje transmembranal al que se llega se le llama potencial de equilibrio. Para poder hallar el potencial de equilibrio se recurre a la ecuación de Nernst, que viene dada de la siguiente manera:

$$E_k = \frac{RT}{ZF} \ln \frac{[K]_e}{[K]_i} \quad (2.6.1)$$

Donde E_k , es el potencial de equilibrio para el potasio medido en voltios, R es la constante de los gases ($8.314 JK^{-1} mol^{-1}$), T es la temperatura absoluta a la que se encuentra (Kelvin), Z es la valencia del ión (para el potasio tiene un valor de 1), F es la constante de Faraday ($9.649 \cdot 10^4 C mol^{-1}$), $[K]_e$ es la concentración extracelular del potasio y $[K]_i$ la concentración intracelular. (Lamas, 2005)

2.5.2 Modelamiento núcleo-conductor.

El término núcleo-conductor se refiere a los conceptos utilizados para el análisis en una dimensión de los problemas de excitación de los nervios, también se refiere a la dirección y localización de los flujos de corriente permitidos. Este modelamiento es muy útil en electrofisiología para entender la propagación de los impulsos nerviosos.

2.5.3 Modelamiento eléctrico para una fibra excitable simple

Este modelamiento se desarrolla basado en conceptos de circuitos eléctricos. Si asumimos que existe una simetría axial es decir que la fibra se va a comportar de la misma manera en todo su alrededor (no a lo largo) entonces se puede hacer el modelamiento en una dimensión. Y dado que una fibra con fluido en sus fronteras, las células extracelulares fluyen longitudinalmente excepto cuando cruzan la membrana, entonces se sigue teniendo corrientes en una sola a lo largo del eje de la fibra. De igual forma para la región intracelular. (Plonsey & Barr, 2007)

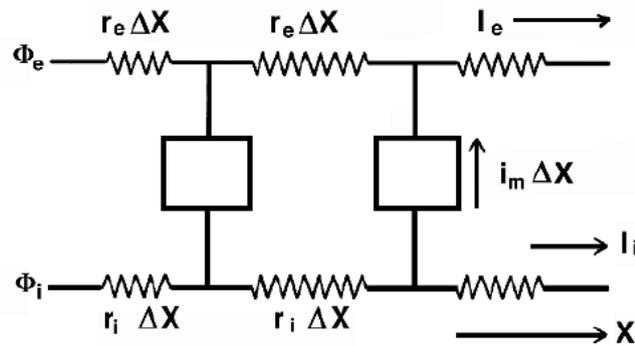


Figura 10. Modelamiento Núcleo conductor para una fibra simple en un espacio extracelular restringido.

Fuente: (Plonsey & Barr, 2007)

La figura 10 refleja lo dicho anteriormente. Teniendo la corriente I_e que circula longitudinalmente en la región extracelular, y la corriente I_i que circula en la región intracelular. El impedimento al flujo se representa con las resistencias. El potencial extracelular e intracelular se los representa como Φ_e y Φ_i respectivamente. Para representar el tamaño de la membrana, se lo hace como un diferencial Δx que tiende a cero. Y este sería el llamado modelamiento lineal núcleo-conductor. (Plonsey & Barr, 2007)

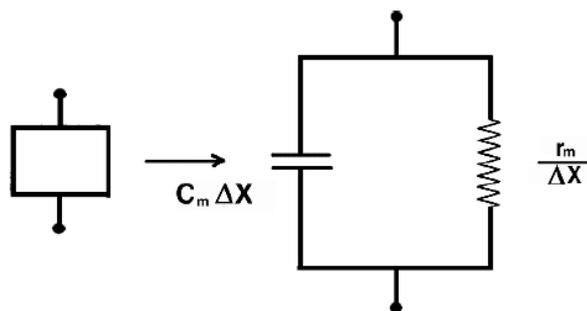


Figura 11. Representación de una membrana de fibras cilíndricas de longitud Δx .

Fuente: (Plonsey & Barr, 2007)

El comportamiento eléctrico transmembranal de una fibra depende de las propiedades de la membrana. Aquí se tiene dos condiciones. La primera es que, bajo una excitación bajo el lumbral, cada membrana se comporta como un circuito RC,

mostrado en la figura 4. La segunda es que bajo una excitación sobre el umbral la membrana tiene un comportamiento no lineal y requiere una descripción como la dada por las ecuaciones de Hodgkin-Huxley, en la figura 5 se ve este caso en donde las conductancias g_k , g_{Na} , g_{cl} son variantes en función del tiempo. Es por eso que en la figura 3 se deja un cuadro blanco porque el comportamiento de la membrana depende de la excitación. (Plonsey & Barr, 2007)

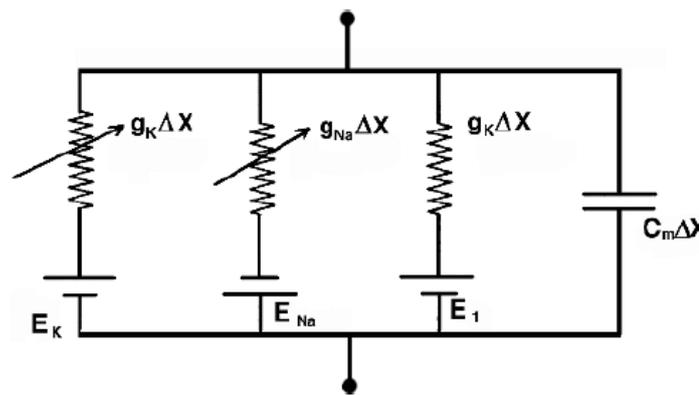


Figura 12. Membrana ante una excitación que ha superado el umbral.

Fuente: (Plonsey & Barr, 2007)

2.6 Tipos de Señales Bioeléctricas

Las señales bioeléctricas provienen del cuerpo humano y son producidas por el desplazamiento de iones especialmente de sodio(Na^+), potasio(K^+) y cloro(Cl^-). El potencial de reposo como el potencial de acción influyen directamente en los parámetros de las señales bioeléctricas debido a que nos proporcionan los datos de carga generada. A continuación se describe los tipos de señales bioeléctricas:

2.6.1 Electrocardiograma ECG

Son generadas por un gran conjunto de células cardiacas a través de las fibras miocárdicas las cuales provocan variaciones de potencial eléctrico que son medidas en la superficie corporal y se registran gráficamente en un tiempo establecido. Las variaciones son captadas por electrodos colocados en la superficie de la piel y por medio de conductores llegan al electrocardiógrafo el cual se encarga de medir y registrar los potenciales de acción del corazón. (Thevenet, 2008).

2.6.1.1 Interpretación de un electrocardiograma

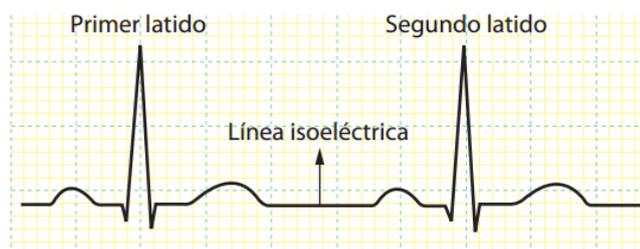


Figura 13. Representación de latidos cardíacos.

Fuente: (Abad, 2008)

En la figura 6 se observa la línea isoelectrónica o basal que es la línea existente entre cada latido, el tiempo transcurrido representa el recorrido en sentido horizontal y la distancia del impulso en sentido vertical representa el voltaje que se está generando. (Abad, 2008)

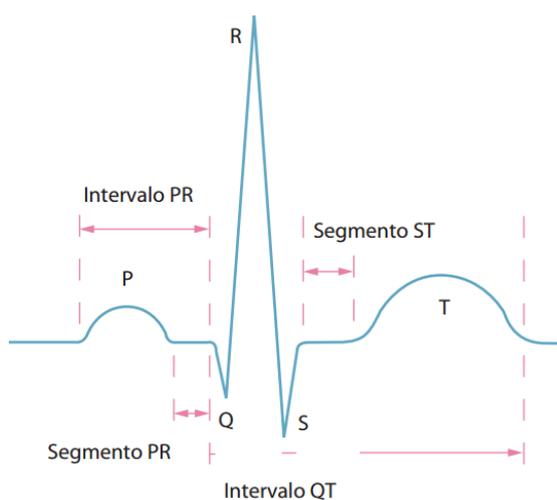


Figura 14. Composición de ondas del electrocardiograma.

Fuente: (Abad, 2008)

2.6.1.2 Onda P

Las aurículas se estimulan en forma eléctrica y se están contrayendo para poder enviar sangre a los ventrículos. Los ventrículos y las aurículas son cavidades con paredes gruesas y delgadas respectivamente que consiguen que el corazón bombee sangre al organismo. (Thevenet, 2008).

2.6.1.3 Segmento P-R

En este segmento las aurículas se vacían y se produce un pequeño descenso de corriente eléctrica por medio del corazón. (Montes, 2007)

2.6.1.4 Complejo QRS

Esta onda cuenta con una sección ascendente y descendente en la cual los ventrículos se estimulan eléctricamente para contraerse y bombear el contenido sanguíneo hacia afuera del corazón. (Thevenet, 2008).

2.6.1.5 Segmento ST

Este segmento nos indica el tiempo que ha transcurrido desde que finaliza la contracción de los ventrículos hasta que se inicia un pequeño periodo de reposo. (Montes, 2007)

2.6.1.6 Onda T

Esta onda indica que el corazón se encuentra en un periodo pequeño de relajación luego haber expulsado sangre que contenía dentro los ventrículos.

2.6.2 Electromiografía EMG

Los músculos del cuerpo están compuestos por unidades motoras(MU) simples que se encuentran conectados a los axones neuronales y estos a su vez están conectados a las fibras encargadas de recibir las órdenes u comandos del sistema nervioso central. La contracción de fibras musculares es provocada por la neurona motora que emite pulsos para dar esta orden, luego de esto le músculo entra en movimiento. (A. Ruiz, 2011)

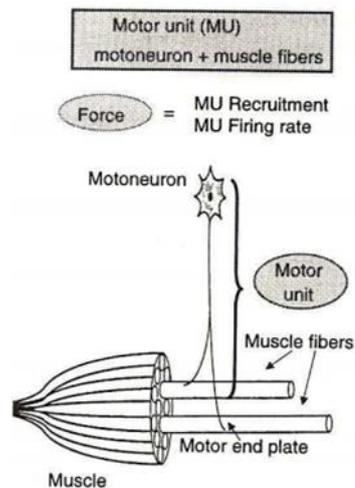


Figura 15. Unidad motora y sus componentes.

Fuente: (Masso, 2010)

Las fibras musculares cuando tienen una excitación o se encuentran en movimiento generan un campo eléctrico, este campo es detectado a través de electrodos de EMG. La generación de una señal EMG se da primeramente con la excitación de las fibras musculares para luego dar paso a la creación del potencial de acción (generación de voltaje), en un cuerpo en movimiento la electromiografía registra la suma de actividad eléctrica generada por todas las fibras musculares que son parte de la unidad motora. (Masso, 2010)

La mayor actividad de señales electromiográficas está alrededor de los 50 Hz y el rango de potencial eléctrico está entre 50 μ V llegando hasta los 5mV requiriendo una amplificación. Hay dos formas de registrar dichas señales; la electromiografía invasiva y la no invasiva o de superficie. (Cifuentes, 2010)

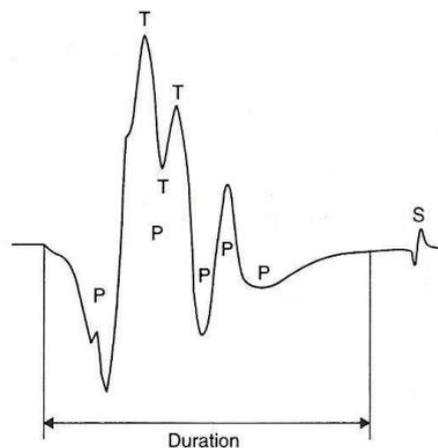


Figura 16. Esquema de una señal Electromiográfica.

Fuente: (Abad, 2008)

En la figura 16 se observa el esquema de una señal EMG donde sus principales características son: los cambios de fase, la duración, la estabilidad y la amplitud del pico principal que se puede medir en un rango muy pequeño desde los μV hasta unos cuantos mV , el espectro de frecuencia esta entre 20 y 500 Hz, mientras que la duración de la señal esta entre los 5 y 15 ms, estos factores pueden ser afectados un acción anormal que ejerce el músculo y por la fatiga muscular. (Abad, 2008) .

2.6.3 Electroencefalograma EEG

Este tipo de señal cuenta con amplitudes extremadamente pequeñas las cuales representan la actividad comprendida en billones de neuronas y que son transmitidas por membranas del cerebro, cuero cabelludo y fluidos. Las EEG están en un rango de 4 a 20Hz las cuales cuentan con una amplitud mínima de 50mV con una fuente cuya impedancia esta entre 10 y 20 $\text{K}\Omega$.

Podemos monitorear las señales eléctricas del cerebro a partir de las siguientes ondas:

Alpha(α): Cuando una persona se encuentra en un estado de relajación y esta despierta, se muestra una actividad eléctrica de 8-12Hz con amplitudes de 20 a 50mv.

Beta(β): Cuando una persona está reaccionando ante un estímulo o se encuentra pensando, se obtiene una señal de 14 a 25 Hz, pero de más baja amplitud.

Theta (θ): cuando la persona se encuentra dormida, se tiene una de 4 a 8Hz y su amplitud puede ser de más de 20mv.

Delta(Δ): cuando una persona está en profundo sueño, la frecuencia esta entre 0.5 a 4 Hz.

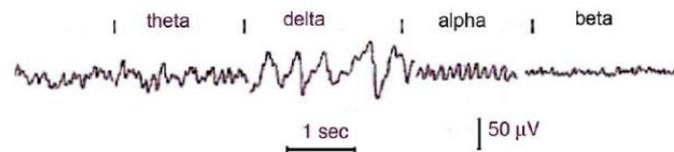


Figura 17. Ondas EEG.

Fuente: (Abad, 2008)

Los amplificadores de ondas EEG deben tener un filtro pasa banda con una respuesta total de (+/- 1dB)

2.7 Sistema de posicionamiento de electrodos

Para el registro de ondas electroencefalográficas existen varios sistemas de posicionamiento de electrodos, como; Marshal, Lennox, Montreal, Illinois, Merlis, Gaustaut, etc. pero el sistema internacional Diez-Veinte (10-20 método Montreal) es el más utilizado actualmente para la detección de estas ondas. El sistema 10-20 coloca y mide las posiciones a distancias iguales sobre la superficie del cuero cabelludo, también toma puntos de referencia en el cráneo utilizando marcas diferentes o identificables. (Navarro, 2010)

En el sistema 10-20 los electrodos son colocados con un espaciado entre ellos del 10 o 20 % de la distancia total que existe entre un par de marcas que están determinadas en el cráneo. Este sistema fue diseñado con el objetivo de dar una adecuada y ergonómica cobertura en la cabeza, con una flexibilidad ideal para colocar electrodos adicionales. (Talamillo, 2011)

2.7.1 Nomenclatura del sistema 10-20

Se utiliza 4 marcas sobre el cráneo: el nasion, el inion, el punto pre auricular izquierdo y derecho como se observa en la siguiente imagen:

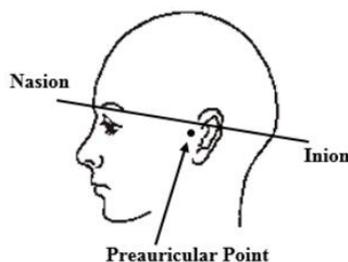


Figura 18. Marcas sobre el cráneo.

Fuente: (Navarro, 2010)

Luego se mide la distancia entre el inion y el nasion pasando por el vertex, el 10 % de esta distancia sobre el nasion señala el punto Frontal Polar (Fp), y el 10 % de la distancia que esta sobre el inion señala el punto Occipital (O). (Navarro, 2010)

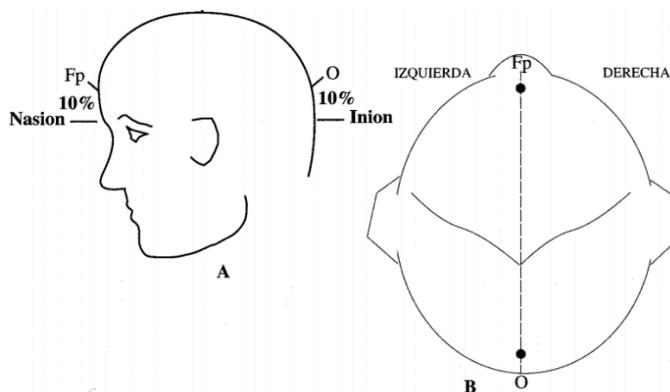


Figura 19. Punto frontal polar y occipital al 10% de distancia.

Fuente: (Navarro, 2010)

Una vez establecido los puntos Fp y O se ponen tres puntos espaciados equidistantemente cada 20% de la distancia comprendida entre el punto frontal y occipital. Estos puntos son: Fz (Frontal cero línea media frontal), Cz (Central cero vértice (Vértex) o coronilla) y Pz (Parietal cero línea media parietal). Estos tres puntos tienen el subíndice z que se utiliza para referenciar el cero de la línea media.

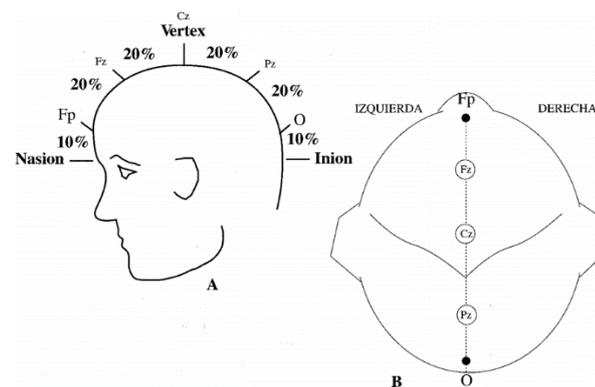


Figura 20. Puntos Fz (Frontal), Pz (Parietal), Cz (Central o Vertex).

Fuente: (Navarro, 2010)

Se mide la distancia desde el punto preauricular izquierdo hasta el preauricular derecho pasando por el punto Cz, luego se toma el 10% de esta distancia la cual representa la posición de los puntos temporales mediales, T3 y T4 que están a la izquierda y derecha respectivamente y forman la línea de electrodos temporales.

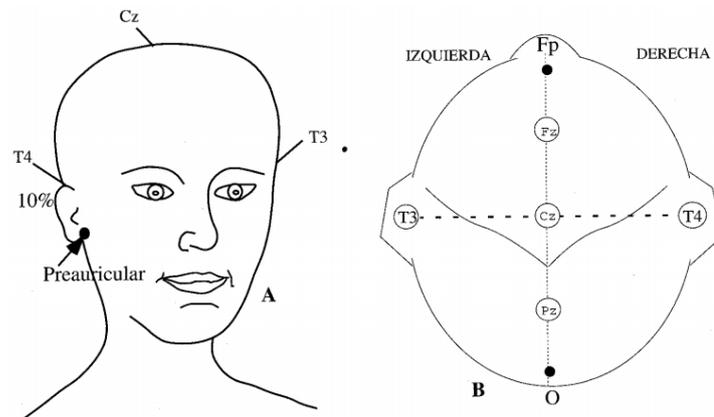


Figura 21. Puntos temporales mediales, T3 y T4.

Fuente: (Navarro, 2010)

Por encima de los puntos temporales mediales se toma un 20% de distancia y ahí se define la posición de los puntos C3 y C4 colocados a la izquierda y derecha.

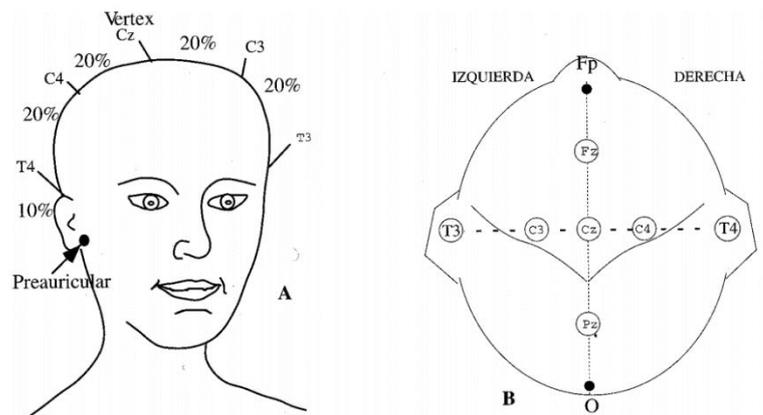


Figura 22. Puntos C3 y C4.

Fuente: (Navarro, 2010)

Los puntos de los electrodos F3 y F4 se colocan en posición equidistante con respecto al punto frontal cero de la línea media y a la línea de electrodos temporales T3 y T4.

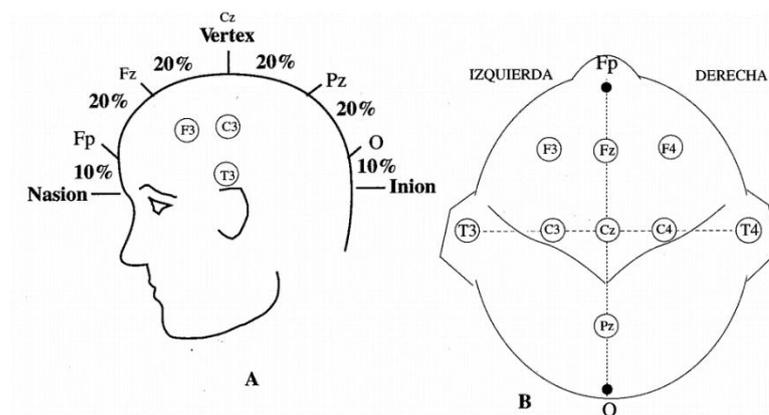


Figura 23. Posición de los electrodos F3 y F4.

Fuente: (Navarro, 2010)

Los electrodos P3 y P4 colocados a la izquierda y derecha respectivamente son equidistantes entre el punto parietal cero (Pz) y la llamada línea de electrodos temporales.

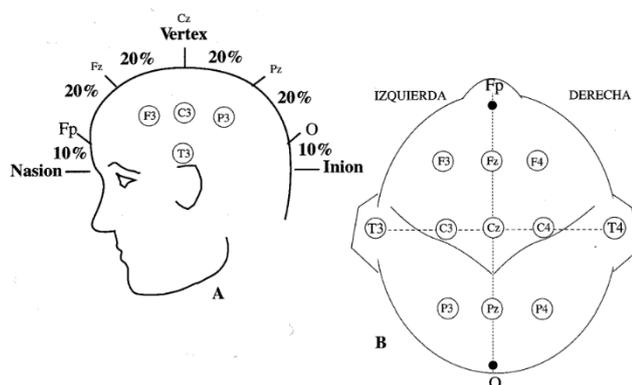


Figura 24. Posición de los electrodos P3 y P4.

Fuente: (Navarro, 2010)

Se toma la medida entre el punto medio de Fp y O por medio de T3, el 10% de esta distancia se a partir de Fp tanto para el lado izquierdo y derecho corresponde a la posición de los electrodos FP1 Y FP2. El 10% de la distancia tomada anteriormente se aplica también al punto O para sacar la posición de los electrodos O1 y O2. (Navarro, 2010)

Los electrodos F7 y F8 son situados en una distancia equidistante entre FP1, FP2 y T3, T4 respectivamente. En cambio, la posición de los electrodos T5 y T6 se colocan en la línea media comprendida entre T3, T4 con los puntos O1, O2 respectivamente. (Cifuentes, 2010)

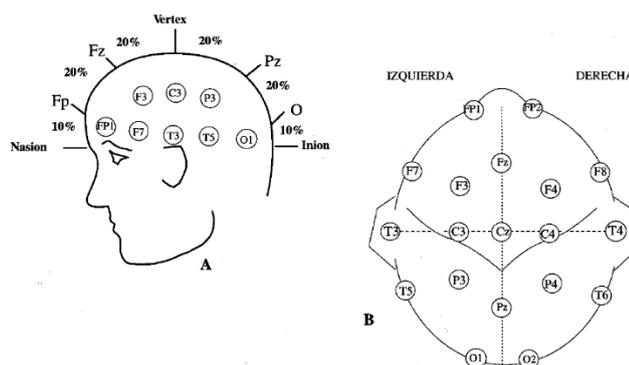


Figura 25. Posición de los electrodos FP1, FP2, O1 Y O1.

Fuente: (Navarro, 2010)

Los llamados electrodos auriculares A1 y A2 son colocados a una distancia correspondiente al 10% de los puntos temporales T3 y T4.

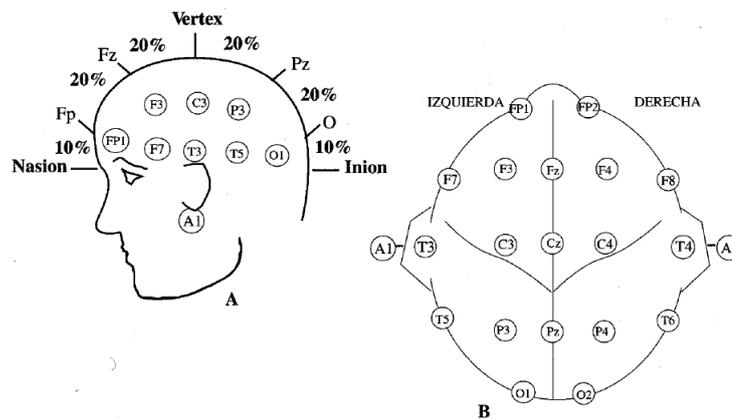


Figura 26. Posición de los electrodos A1 y A2.

Fuente: (Navarro, 2010)

A continuación, se presenta detalladamente los puntos de todos los electrodos establecidos para el sistema internacional 10-20 explicado anteriormente:

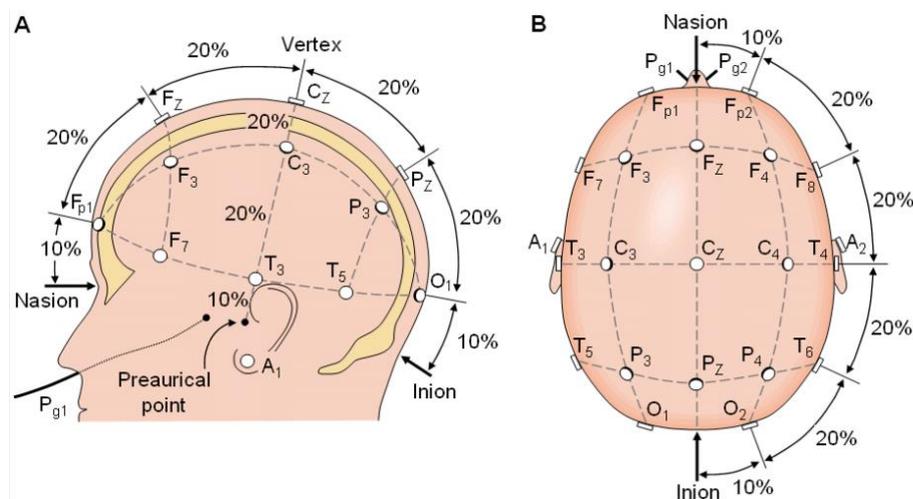


Figura 27. Sistema Internacional 10-20 para colocación de electrodos.

Fuente: (Ortega, 2014)

2.8 Procesamiento Digital de Señales

El procesamiento de señales se basa en representar, transformar, manipular y obtener información útil para propósitos específicos. Cuando se maneja procesamiento digital de señales, la representación se hace en secuencias de números con precisión finita. Dicho procesamiento se lo realiza con procesadores digitales.

En la muchas de las aplicaciones del procesamiento digital de señales, se requiere que dicho procesamiento sea en tiempo real, es decir que con la misma velocidad con la que las señales ingresen al procesamiento se obtengan las señales procesadas. Otro requerimiento muy común es la interpretación de señales, por ejemplo, donde se recuperen patrones para clasificar señales. Para estos casos es común utilizar una etapa de filtrado para luego tener una etapa de reconocimiento de patrones.

2.8.1 Filtros Digitales

El objetivo de los filtros digitales es modificar el espectro de una señal. En breves palabras es utilizar un algoritmo para dada una señal digital de entrada se obtenga una secuencia numérica a su salida. Se caracterizan por ser predecibles, flexibles, de fácil simulación, consistentes y precisos. (Universidad Distrital, 2005)

Muchos filtros son descritos mediante una ecuación de diferencias lineales de coeficientes constantes, relacionando entrada $x(n)$ con salida $y(n)$.

$$y(n) = \sum_{k=0}^N a_k y(n - k) + \sum_{k=0}^M b_k x(n - k) \quad (2.9.1)$$

De acuerdo a su respuesta ante un impulso se clasifican en FIR (Finite Impulse Response) filtros de respuesta infinita e IIR (Infinite Impulse Response) filtros de respuesta infinita al impulso. Se diferencia debido a que ante un impulso la salida tendrá un número finito o infinito de términos no nulos.

2.8.1.1 Filtros FIR

De acuerdo a la ecuación de diferencias (1-1) se modela un filtro de respuesta finita con la ecuación:

$$y(n) = \sum_{k=0}^M b_k x(n - k) \quad (2.9.2)$$

Este sistema solo toma las últimas muestras de la señal de entrada por lo que se le denomina FIR. Como podemos ver de la ecuación (1-2) la señal de salida $y(n)$ no es más que la suma de cada uno de los últimos $M + 1$ datos multiplicados por un factor b_k .

2.8.1.2 Filtros IIR

Con la misma ecuación de diferencias lineal de coeficientes constantes (1-1) se describe un filtro de respuesta infinita como:

$$(n) = \sum_{k=0}^N a_k y(n-k) + \sum_{k=0}^M b_k x(n-k) \quad (2.9.3)$$

Como se puede observar es el mismo modelo que el filtro de respuesta finita, solo que con la excepción que al menos un coeficiente a_k es diferente de cero.

2.8.2 Transformada discreta de Fourier

La transformada discreta de Fourier, es una transformada discreta que nos permite hacer el análisis de Fourier, y pasar del dominio del tiempo al dominio de la frecuencia. Para ello se debe encontrar un grupo de señales oscilatorias de longitud N , que contengan un número entero de periodos sobre ellas. Considérese primero una familia con longitud finita de señales sinusoidales, de la forma:

$$\omega_k[n] = e^{j\omega_k n} \quad n = 0, \dots, N-1 \quad (2.9.4)$$

Donde cada ω_k es una frecuencia diferente. Para determinar estas frecuencias, nótese que para que $\omega_k[n]$ contenga un número entero de periodos sobre N muestras, este debe cumplir

$$\omega_k[N] = \omega_k[0] = 1$$

Que se traduce a:

$$(e^{j\omega_k})^N = 1$$

Que tiene N distintas soluciones, que son las N raíces de $e^{\frac{j2\pi m}{N}}$, $m=0, \dots, N-1$. Si definimos el número complejo $W_N = e^{-j\frac{2\pi}{N}}$, entonces la familia de N señales, puede ser escrita como:

$$\omega_k[n] = W_N^{-nk}, \quad n=0, \dots, N-1$$

Para cada valor de $k=0, \dots, N-1$. Podemos representar esas N señales como un conjunto de vectores $\{\mathbf{w}^{(k)}\}_{k=0, \dots, N-1}$ en \mathbb{C}^N con

$$\omega^{(k)} = \left[1 \ W_N^{-k} \ W_N^{-2k} \ \dots \ W_N^{-(N-1)k} \right]^T \quad (2.9.5)$$

Se puede verificar que $\{\mathbf{w}^{(k)}\}_{k=0, \dots, N-1}$ es un grupo de N vectores ortogonales y por lo tanto una base para \mathbb{C}^N , obtenemos el producto punto, entre dos de estos vectores.

$$\langle \omega^{(m)}, \omega^{(n)} \rangle = \sum_{i=0}^{N-1} W_N^{(m-n)i} = \begin{cases} N & m = n \\ \frac{1 - W_N^{(m-n)N}}{1 - W_N^{m-n}} = 0 & m \neq n \end{cases}$$

2.8.2.1 Forma Matricial

Las fórmulas de análisis y síntesis de la transformada discreta de Fourier, pueden ser fácilmente expresadas en notación matricial, definida una matriz W cuadrada $N \times N$ apilando la conjugada de los vectores base, es decir $W_{nk} = e^{-j(\frac{2\pi}{N})nk} = W_N^{nk}$. Y así se puede tener para todos los vectores $x \in \mathbb{C}$:

$$X = Wx$$

$$x = \frac{1}{N} W^H X \quad (2.9.6)$$

Donde x representa la señal en el dominio del tiempo, y X en el dominio de la frecuencia.

Debido a que la base es ortogonal pero no orto normal, se debe escalar a la señal en $1/\sqrt{N}$. Teniendo por la relación de Parseval, conservando la energía de la señal finita.

$$\|x\|_2 = \frac{1}{\sqrt{N}} \|X\|_2 \quad (2.9.7)$$

2.8.2.2 Forma Explicita

Es común en la literatura escribir el producto punto para las fórmulas de síntesis y análisis, esto con el fin de hacer más eficientes los procedimientos computacionales, teniendo las formulas:

$$X[k] = \sum_{n=0}^{N-1} x[n] W_N^{nk} \quad k = 0, \dots, N-1$$

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] W_N^{-nk}, \quad n = 0, \dots, N-1 \quad (2.9.8)$$

2.8.2.3 Interpretación Física

Para dar la interpretación física de la transformada discreta de Fourier, nosotros solo hemos obtenido la descomposición de una señal de longitud finita en un grupo de N componentes sinusoidales. La magnitud y la fase inicial de cada oscilación están dada por los componentes $X[k]$. Para reforzar el punto nuevamente

- Tómese un arreglo de N generadores complejos sinusoidales
- Determine la frecuencia del generador k -enésimo a $(2\pi/N)k$
- Determine la amplitud del generador k -enésimo a $|X[k]|$, es decir a la magnitud del coeficiente de la transformada k -enésimo.
- Determine la fase del generador k -enésimo a $\angle X[k]$, es decir la fase del coeficiente del coeficiente de la transformada k -enésimo.
- Empezar los generadores al mismo tiempo y sumar sus salidas.

Las primeras N valores de salida de la señal son exactamente $x[n]$.

Observando del lado contrario, cada $X[k]$ muestra cuanto comportamiento oscilatorio a la frecuencia $2\pi/k$ está contenida en la señal. Esto es consistente con el hecho que el producto punto es una medida de similitud. Los coeficientes $X[k]$ están referidos al espectro de señal. La magnitud cuadrática $|X[k]|^2$ es una medida (escalado al factor N) de la energía de la señal a la frecuencia $\left(\frac{2\pi}{N}\right)k$. Por consiguiente los factores $X[k]$ muestran exactamente como la energía global de la señal original es distribuida en el dominio de la frecuencia. La fase de cada coeficiente de Fourier, dado por $\angle X[k]$, especifica la fase inicial de cada oscilación para la fórmula de reconstrucción, es decir la alineación relativa de cada exponencial complejo.

2.9 Reconocimiento de patrones.

Para el reconocimiento de patrones nos enfocaremos en el aprendizaje automático que nos sirve para extraer y clasificar información. Uno de los primeros en incursionar en este campo fue Arthur Samuel que en 1956 definió el aprendizaje automático como: "Field of study that gives computers the ability to learn without being explicitly programmed", de la traducción obtenemos la definición como el campo de estudio que da a los computadores la habilidad de aprender sin una programación explícita.

Para lograr que no necesite una programación explícita, se requiere de un aprendizaje previo del algoritmo con diferentes entradas con o sin una salida específica, para poder predecir la salida ante una entrada no conocida. Si es que el aprendizaje se lo realiza con entradas con o sin una salida específica, nos dice si un algoritmo es de aprendizaje supervisado o no supervisado.

El aprendizaje supervisado, se lo puede explicar cuando se trata de predecir el costo de una casa, para este caso se recolectan costos de casas ya vendidas y sus características como numero de cuartos, baños, etc. Para este caso las entradas son las características de la casa y las salidas son los precios de las casas. Con estas entradas y salidas se entrena al algoritmo para predecir el costo de una casa con sus características como entrada. Como se ha dicho para el entrenamiento es necesario tanto entradas como salidas, de casos anteriores, para que sea un aprendizaje supervisado.

El aprendizaje supervisado, es fácil de explicar poniendo el ejemplo de querer agrupar personas por su altura, en bajo, mediano y alto. En este caso se recolecta la altura de un cierto número de personas y se divide en estas tres categorías, para así predecir la categoría de una nueva persona. En este caso las entradas son las alturas de las personas, pero no se tiene una salida, si no que el algoritmo realiza esta categorización para poder predecir una nueva entrada. Entonces para que sea aprendizaje no supervisado se tienen simplemente las entradas de datos recolectados y el algoritmo es el que asigna las salidas.

Existen diferentes métodos para el reconocimiento de patrones, mediante el aprendizaje automático tenemos métodos simples como regresión lineal, hasta más complejos como redes neuronales o como máquinas de vectores de soporte por mencionar. Debido al extenso campo que se tiene, aquí mencionaremos simplemente el uso de las redes neuronales que es el método que se pretende usar.

2.9.1 Redes Neuronales

El uso de redes neuronales es útil cuando se tienen muchas características de entrada al algoritmo. El modelo de las redes neuronales es basado en la funcionalidad del cerebro. Como se puede ver en temas anteriores una neurona tiene dendritas como entrada, un cuerpo celular que responde a las entradas mediante los axones como salida.

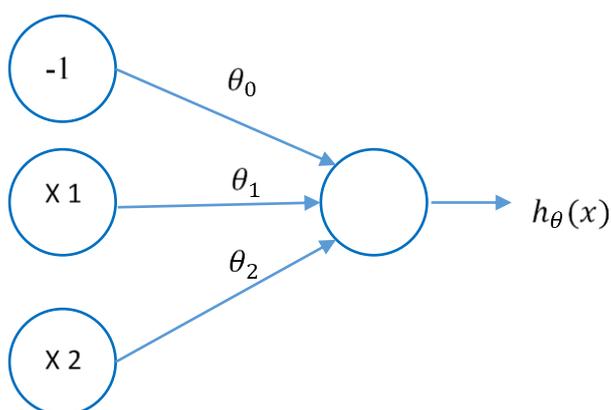


Figura 28 .Estructura básica de una red neuronal.

En la ilustración 1 se puede observar la estructura básica de una red neuronal en donde se tiene un vector de entradas $X = [x_0, x_1, \dots, x_n]'$, por lo general se aumenta una entrada x_0 con un valor de -1 a esta entrada se la llama unidad bias. Cada entrada se multiplica por un específico peso θ_k , y a la sumatoria total es decir a $\theta^T X$ se le aplica una función $h_\theta(x)$.

Una de las funciones más utilizadas es la llamada función sigmoid que viene dada por la ecuación:

$$h_\theta(x) = \frac{1}{1+e^{-\theta^T x}} \quad (2.10.1)$$

Lo que se logra con la función es asignar un valor entre 0 y 1, de acuerdo a la entrada, para valores positivos asigna valores entre 0.5 y 1, y para valores negativos asigna valores entre 0 y 0.5.

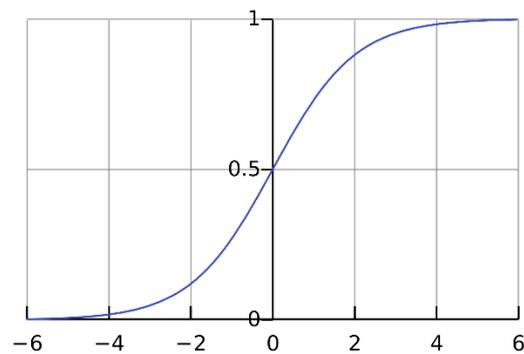


Figura 29. Función sigmoide.

Existen diferentes configuraciones de las redes neuronales, en donde se pueden incluir más capas e incluir más salidas. Añadiendo más salidas se logra tener clasificadores, en donde la salida con el mayor valor obtenido es la clase o categoría a la que pertenece la entrada. En la ilustración 3 se puede apreciar una red neuronal con multicapas y de clasificación multiclases.

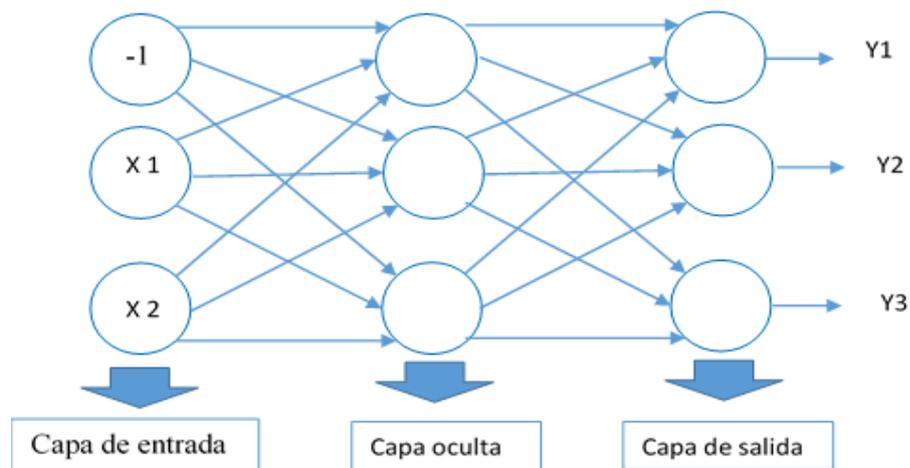


Figura 30. Estructura de una red neuronal multicapa y clasificador multiclases.

Las redes neuronales son uno de los más potentes algoritmos de aprendizaje. Es un algoritmo para el ajuste de parámetros derivados dado un set de entrenamiento. Un set de entrenamiento viene dado por un conjunto $\{(x^1, y^1), (x^2, y^2), (x^3, y^3) \dots (x^n, y^n)\}$, como se puede ver son pares de entradas y salidas. El número de capas existentes en la red viene dado por la letra L . Y a cada una de las capas se las denomina como s_l donde l es el número de la capa que se está analizando.

2.9.1.1 Tipos de clasificadores con redes neuronales

La clasificación se la hace por dos tipos. Clasificación binaria y clasificación multiclase. Si se hace por clasificación binaria es necesaria una sola salida en donde 0 corresponde a una clase y 1 corresponde a la otra clase. Si se necesita una clasificación de 3 o más clases se la deben realizar mediante múltiples salidas con el número de salidas igual al número de clases.

2.9.1.2 Entrenamiento de la Red Neuronal

○ Función de Costo

Para evaluar cuanto error tiene una red neuronal se lo realiza mediante la llamada función de costo. La función lo que hace es a cada uno de los pares entradas y salidas del set de entrenamiento evaluar los pesos escogidos calculando el error de la salida precedida por la salida real. La función de costo viene dada por la siguiente ecuación:

$$J(\theta) = -\frac{1}{m} \left[\sum_{i=1}^m \sum_{k=1}^K y_k^{(i)} \log \left(h_{\theta}(x^{(i)}) \right)_k + (1 - y_k^{(i)}) \log \left(1 - h_{\theta}(x^{(i)}) \right)_k \right] + \frac{\lambda}{2m} \sum_{l=1}^{L-1} \sum_{i=1}^{s_l} \sum_{j=1}^{s_{l+1}} \theta_{ji}^{(l)2} \quad (2.10.2)$$

El término $y_k^{(i)} \log \left(h_{\theta}(x^{(i)}) \right)_k$ nos sirve para asignar un costo, si la salida real es uno. Con la función logarítmica lo que hacemos es que, si la red predice una salida cercana a uno, al aplicar el logaritmo obtenemos un costo cercano a 0, pero si la red predice un valor de cero al aplicar el logaritmo se obtiene un costo muy alto dado que el logaritmo de 0 es infinito. En cambio el término $(1 - y_k^{(i)}) \log \left(1 - h_{\theta}(x^{(i)}) \right)_k$ nos sirve para asignar un costo si la salida real es cero. De la misma forma cuando la red

predice un cero, al tener uno menos cero el costo es cercano a cero, pero cuando predice un número cercano a uno el costo es muy alto.

Podemos concluir que la función de costo, nos sirve para evaluar la asignación de peso y visualizar la respuesta de la red ante cambios en los pesos.

- **Retro propagación o Propagación hacia atrás**

El algoritmo de retro propagación nos sirve para entrenar la red neuronal, como una explicación simple, el algoritmo de retro propagación, lo que hace es calcular la derivada de la función de costo en cada uno de los nodos. Se llama retro propagación debido a que para el entrenamiento se empieza por obtener la derivada de la función de costo con respecto a las salidas, y con estas derivadas se multiplica hacia atrás estas derivadas por los pesos que estamos evaluando y obtenemos en cada nodo una derivada. Esta derivada nos sirva para evaluar en qué sentido debemos variar los pesos, para obtener los mejores pesos existen varios algoritmos avanzados dentro de la rama de la optimización matemática.

La rama de la optimización matemática lo que hace es encontrar los parámetros en una función que haga que la función alcance su mínimo o máximo valor. Existen varios métodos dentro de ellos los computacionales que se los realizan por iteraciones. Algunos de ellos son métodos de programación lineal, de programación no lineal o algoritmos genéticos.

2.10 Tipos de discapacidades

Existen 5 tipos de discapacidades: motora, sensorial auditiva, sensorial visual, visceral y mental. En este caso nos centraremos en la discapacidad motora debido a que se relaciona directamente con nuestro proyecto.

2.10.1 Discapacidad motora (motriz)

Se define como discapacidad motora como la dificultad que presentan algunas personas para participar en actividades propias de la vida cotidiana, que surgen como consecuencia de la dificultad de manipular objetos o acceder a diferentes lugares, espacios y actividades que realizan todas las personas, y las barreras en el que se desenvuelve la persona. Esta discapacidad es causada por un funcionamiento deficiente del sistema nervioso, del sistema muscular o de una interacción de los dos

sistemas, lo cual dificulta o imposibilita la movilidad funcional de algunas de las partes del cuerpo humano.

Desde una perspectiva social y cultural podemos afirmar que la discapacidad motora no depende únicamente de las características físicas o biológicas del niño o niña, por ejemplo, si una persona presenta dificultades para la movilidad de las extremidades inferiores no podrá acceder normalmente a las viviendas, construcciones, escaleras, tampoco podrá usar los servicios higiénicos comunes, otro problema es el uso de transporte público. Las barreras de un entorno hecho por y para personas sin discapacidad limitaran a las personas que padecen de discapacidad motriz.

La discapacidad motora proviene de 3 causas:

Antes del nacimiento

- Deficiente desarrollo de las células cerebrales.
- Trastornos patológicos en la madre.

Después del nacimiento

- Parto prematuro
- Dificultad del parto.

Accidentes o lesiones

- La discapacidad motriz se clasifica en tres grandes grupos:

a.- La parálisis cerebral

Es el trastorno motor más común en los niños de edad escolar, constituye una anomalía de la función motora debida a un defecto, lesión o enfermedad en el sistema nervioso central, contenido en la cavidad craneana y se asocia a diversos trastornos sensoriales y el lenguaje.

Cuando ocurre una lesión cerebral o pérdida de la función nerviosa ocurren los siguientes trastornos:

- Parálisis cerebral espástica: provoca rigidez muscular.
- Parálisis atetósica: la persona realiza movimientos irregulares.

- Parálisis atáxica: afecta de manera especial al equilibrio corporal, también se caracteriza por la dificultad para la realización de movimientos coordinados, provoca un mal control muscular de la cabeza, de la columna y de las extremidades.
- Parálisis mixta: cuando una persona está asociada a los tres tipos de parálisis descritos anteriormente.

b.- La espina bífida

Anormalidad en la espina dorsal debido a la cual el canal vertebral no llega a cerrarse, por lo que generalmente el daño medular es inevitable.

c.- Las miopatías o distrofias musculares progresivas

Trastorno de la musculatura esquelética, de origen genético, que se caracteriza por el debilitamiento y la degeneración progresiva de los músculos voluntarios.

La clasificación de acuerdo al número de extremidades afectadas es la siguiente:

- Monoplejia: un solo miembro del cuerpo.
- Diplejia: afecta las extremidades inferiores.
- Triplejia: afecta un miembro superior y las extremidades inferiores.
- Hemiplejia: afecta el lado derecho o izquierdo del cuerpo.
- Cuadriplejia: afecta las cuatro extremidades del cuerpo.

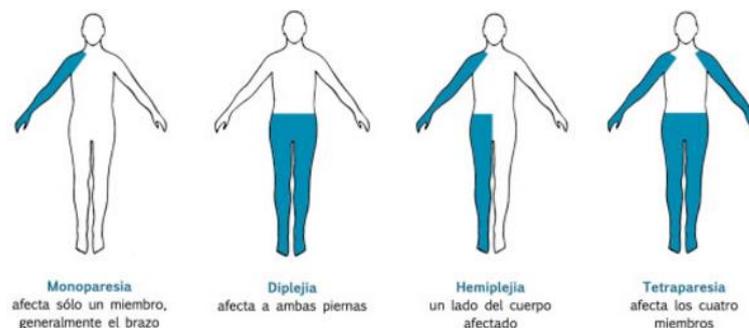


Figura 31. Discapacidad de acuerdo a las extremidades afectadas.

Fuente: (Medical, 2006)

El grado de afectación tiene tres niveles:

- Leve: si realiza sus actividades diarias o básicas por su cuenta
- Moderado: cuando la persona requiere algún tipo de aditamento especial
- Severo: cuando el daño o la lesión requiere del apoyo de una persona para todas las actividades.

2.11 Sillas de Ruedas

Una silla de ruedas debe tener como objetivo permitir al usuario la máxima funcionalidad, comodidad y movilidad. Para cumplir con este objetivo, la silla debe estar pensada para ajustarse a la persona, no es la persona quien debe acomodarse a su silla. Si se escoge una silla de ruedas no apropiada, puede resultar incómoda o por ejemplo tener un asiento en el que el usuario resbale hacia delante o se incline hacia un lado. El resultado será que la energía del usuario se malgastará de manera innecesaria debido al esfuerzo continuado por modificar su postura.

Se considera que lo que más afecta a la maniobrabilidad de la silla son su peso y el material con el que esté hecha su estructura. Sin embargo, hay factores más importantes como el asiento, la postura, la distancia entre ejes de las ruedas, la posición y el tamaño de las ruedas, incluso la forma en que la silla ha sido ajustada o montada, que pueden influir decisivamente en la funcionalidad y movilidad del usuario.

Factores que influyen en la movilidad y ergonomía de una silla de ruedas:

- La movilidad y el rozamiento es uno de los factores importantes que deben ser considerados, debido a que cuanto mayor sea el rozamiento, la resistencia a rodar de la silla será mayor, esto provocará que el usuario requiera mayor esfuerzo para mover la silla.
 - Centro de gravedad de la silla: Al mover el centro de gravedad hacia atrás y hacia arriba se aumenta el peso sobre las ruedas traseras y hace que la silla sea más fácil de manejar, pero más inestable. Si se desplaza el centro de gravedad hacia abajo y hacia delante, la silla gana en estabilidad, pero es más difícil de manejar.
 - Distancia entre ejes de ruedas delanteras y traseras: Una distancia larga entre los ejes delantero y trasero mantiene mejor el rumbo, pero se requiere de mayor esfuerzo para realizar desplazamientos debido a que las ruedas realizan un recorrido

mayores distancias, mientras que una distancia entre ejes corta resulta más suave y fácil de manejar.

- **Tamaño de las ruedas:** las ruedas delanteras van desde un diámetro de 75 a 200 mm, mientras más pequeñas se tendrá facilidad de giro y menor rozamiento, en cambio cuando las ruedas son más grandes resulta más fácil evadir obstáculos y no se quedan enterradas, se recomiendan las ruedas pequeñas para interiores y las grandes para exteriores. (Medical, 2006)

Para las ruedas traseras el tamaño más habitual del diámetro es de 600mm para una persona adulta de estatura normal, para niños van de 500 a 550mm de diámetro, mientras que para personas deportistas o muy altas se utilizan ruedas de 650mm. (Medical, 2006)

- **Altura y posición de las ruedas:** para obtener un impulso eficaz y con mayor propulsión, la posición de las ruedas traseras debe estar ubicadas de tal manera que el usuario al dejar caer el brazo estirado debe topar el eje de las ruedas traseras con la punta de los dedos. Si el eje de las ruedas le queda más alto o más bajo, la propulsión se volverá incómoda e ineficiente porque se deberá flexionar o estirar demasiado los brazos.

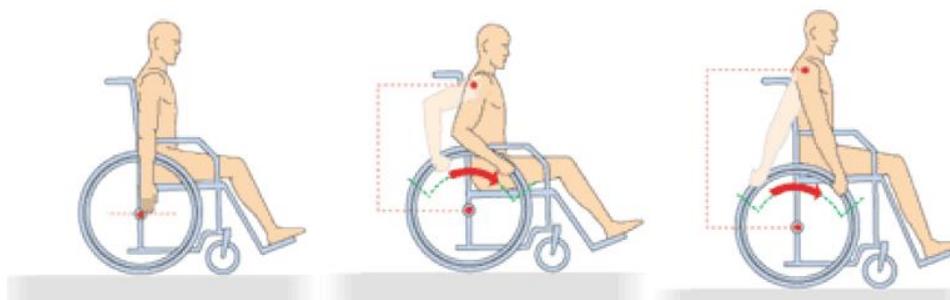


Figura 32. Posicionamiento vertical de la rueda.

Fuente: (Medical, 2006)

Si el eje está adelantado o retrasado con respecto a la punta de los dedos, el impulso que genere el usuario se verá afectado y no se podrá completar todo el recorrido.



Figura 33. Posicionamiento horizontal de la rueda

Fuente: (Medical, 2006)

- **Distribución de peso sobre la silla:** Las sillas de ruedas son diseñadas de tal manera que el peso de una persona se distribuye de la siguiente manera; aproximadamente el 60% del peso está sobre las ruedas traseras y el 40% en las ruedas delanteras, con esta distribución se obtiene un equilibrio, movilidad y desplazamiento muy eficiente con menor esfuerzo de propulsión o impulso. Uno de los factores más importantes que afectan el rendimiento de las sillas de ruedas es la distribución de peso sobre la silla. La distancia horizontal entre el centro de gravedad (CG, de la combinación de masas de la silla y el ocupante) y el eje de las ruedas traseras determina la distribución de masas que soportara cada uno de los ejes. (Brubaker, 2000).

- **Resistencia a la rodadura:** Las sillas de ruedas son diseñadas de tal manera que el peso de una persona se distribuye de la siguiente manera; aproximadamente el 60% del peso está sobre las ruedas traseras y el 40% en las ruedas delanteras, con esta distribución se obtiene un equilibrio, movilidad y desplazamiento muy eficiente con menor esfuerzo de propulsión o impulso. Uno de los factores más importantes que afectan el rendimiento de las sillas de ruedas es la distribución de peso sobre la silla. La distancia horizontal entre el centro de gravedad (CG, de la combinación de masas de la silla y el ocupante) y el eje de las ruedas traseras determina la distribución de masas que soportara cada uno de los ejes.

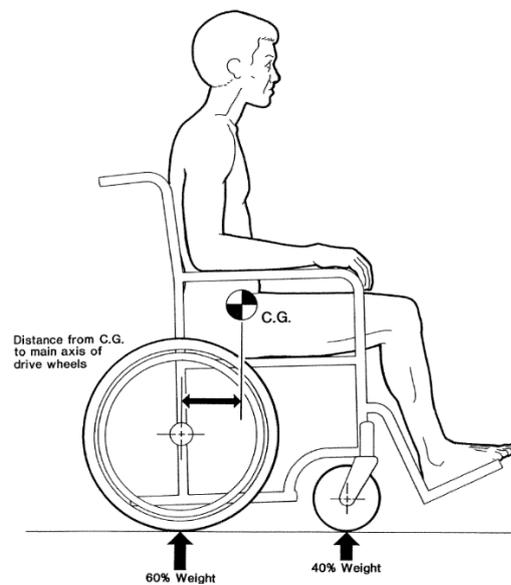


Ilustración 24. Distribución de peso sobre una silla de ruedas.

Fuente: (Brubaker, 2000).

- Alturas del asiento y le reposapiés: La altura adecuada de la silla va de 17 a 21 pulgadas (43 a 53cm) para poder obtener un rendimiento óptimo (Ilustración 24), mientras que el reposapiés tiene una altura respecto al piso de 2 pulgadas (5cm), esta altura es fundamental si el reposapiés está muy cerca la piso esto provocará que el usuario debe deslizarse sobre el asiento, al contrario si está muy arriba, el peso de la persona estaría concentrado sobre el coxis pudiendo provocar lesiones a futuro. (Brubaker, 2000).

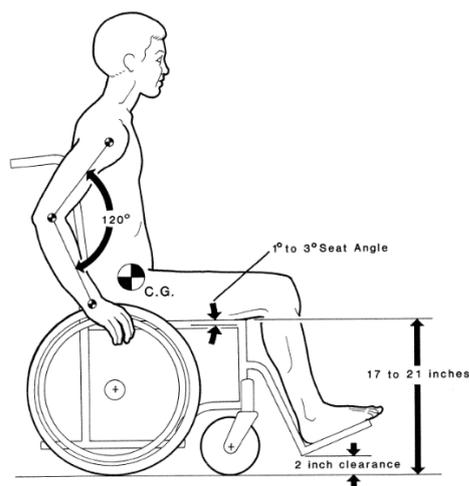


Figura 34. Altura del asiento.

Fuente: (Brubaker, 2000).

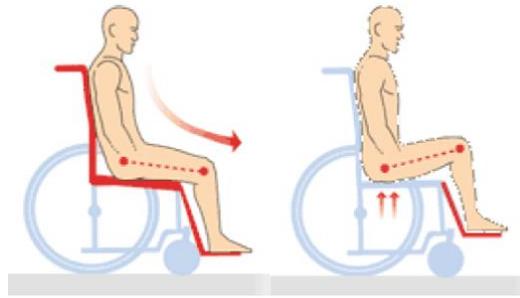


Figura 35. Altura del reposapiés.

Fuente: (Medical, 2006)

2.12 Tipos de control de movimiento de sillas de ruedas

2.12.1 Consideraciones previas

Se debe tener una postura muy estable y cómoda para poder maniobrar una silla, la forma de control y el tipo de mando no debe afectar la comodidad del usuario, al contrario, debe facilitar su movilidad. Si el usuario no puede mantener una postura estable, será muy difícil que pueda realizar gestos o movimientos con la cabeza o extremidades.

2.12.2 Control por Voz

Se utiliza un sistema de reconocimiento de voz donde el usuario por medio de órdenes verbales ejecuta los movimientos que desea realizar sobre la silla, esta tecnología se utiliza en individuos que tienen discapacidades motrices muy severas. (Trina Pérez, 1994).

2.12.3 Control por movimientos

Normalmente este tipo de control se lo realiza con movimientos de la cabeza, para lo cual se utilizan acelerómetros que van ubicados en la parte posterior, los mismos que envían información respecto a cambios inerciales que son provocados por las aceleraciones producto del movimiento, esta información es procesada para luego ejecutar órdenes.

2.12.4 Control con la mano

En este control se utiliza normalmente un joystick el cual cuenta con una palanca y botones, lo que facilita enormemente el modo de operación, la palanca controla todos los movimientos a realizarse, mientras que con los botones se puede dar el encendido y apagado.

2.12.5 Control con movimientos oculares

Para este control se realiza un monitoreo constante de una sección determinada del ojo, por medio de un dispositivo de adquisición y mediante el procesamiento digital de imágenes se extraen y se distinguen los movimientos con los cuales se podrán ejecutar instrucciones sobre la silla de ruedas.

2.12.6 Control mediante señales Electroencefalográficas (EEG)

Las señales EEG representan la actividad eléctrica comprendida en billones de neuronas, esta actividad se puede monitorear mediante electrodos ubicados en lugares específicos de la cabeza, para realizar un control con estas señales se debe realizar un riguroso procesamiento e interpretación para luego proceder a clasificarlas. Una vez clasificadas se le asigna una respectiva acción o mando para poder realizar un movimiento de la silla de ruedas.

CAPITULO 3

DISEÑO Y CONSTRUCCIÓN

3.1 Introducción

En ese capítulo describiremos la funcionalidad de cada una de las partes y elementos tanto en la mecánica, electrónica y programación. La parte mecánica será diseñada con el fin de obtener un módulo sencillo, compacto y principalmente adaptable, pero a la vez cumpla con las normas de diseño, esta etapa se iniciará a partir de una sola rueda y su motor con la que se realizará el recorrido o desplazamiento de la silla y su dirección. Para la parte electrónica se trabajará a partir de un sensor de señales electroencefalográficas que tiene un solo electrodo y con el cual se obtendrá las instrucciones necesarias para realizar el control de todo el módulo adaptable. Finalmente, en la programación se realizará diferentes procesos para poder interpretar las señales provenientes del sensor y para controlar cada uno de los actuadores que están en el módulo.

3.2 Requerimiento y limitaciones

Dado que el presente proyecto se lo realiza con el auspicio de la empresa ALGORITMUN CIA. LTDA. Son varios los requerimientos que la empresa ha solicitado. Los requerimientos se los describe a continuación.

- Sensor Mindwave Neurosky

Dado que la empresa cuenta con el sensor Mindwave ha sido el principal requerimiento hacer uso de dicho sensor. Las especificaciones técnicas se lo describirán más adelante, pero al tener un solo punto de adquisición de señales electroencefalográficas, este requerimiento es considerado también como una limitación.



Figura 36. Sensor Mindwave Neurosky.

Fuente: (Neurosky, 2016)

- Rueda con motor brushless interno

El motor brushless, de igual manera ha sido dotado por la empresa para el proyecto. De igual manera las especificaciones técnicas de las describirá adelante para hacer uso en el diseño. El motor servirá para dar tracción a la silla de ruedas y cuenta con su batería propia y el driver para un manejo más fácil.



Figura 37. Rueda con motor brushless interno.

Fuente: (Motion, 2016)

- Adaptabilidad y modularidad.

El requerimiento en cuanto a la parte mecánica, es que sea modular y adaptable, es decir que sea posibles el montaje y desmontaje del módulo mecatrónico en cualquier tipo de silla de ruedas sin la necesidad de efectuar alguna modificación en la silla de ruedas.

3.3 Diseño mecánico

3.3.1 Requerimientos mecánicos

El diseño del módulo debe estar acorde a los siguientes requerimientos:

- Adaptable: debe funcionar para todo tipo de sillas de ruedas.
- Una sola rueda de tracción: El desplazamiento de ser con un solo motor.
- Dirección (Izquierda, derecha): La misma rueda debe poder girar en cualquier sentido.
- Freno: Para seguridad del usuario.

Diseño

Primeramente, se buscó partes en común que tengan los diseños de sillas de ruedas, después de analizar una gran variedad de sillas se encontró lo siguiente:



Figura 38. Modelos comunes de sillas de ruedas.

Fuente: (Reactiv, 2015)

Como se observa en la figura 39 los modelos más comunes tienen dos tubos que normalmente forman un ángulo de 90 grados, estos tubos sirven de soporte para el reposapiés y también sujetan las llantas delanteras.

Una vez identificado las partes en común, tomaremos estos tubos como soporte para poder sujetar el módulo Mecatrónico.

3.3.1.1 Perfil de soporte

Para dimensionar el perfil se tomó en cuenta las medidas estándar respecto al ancho del asiento (figura 40).

	Ancho de asiento	Largo total
		
1400SR	41	99
	43	
	45	
1401SR	41	105
	43	
	45	

Todas las medidas en centímetros.

Figura 39. Medidas estándar del ancho del asiento.

Fuente: (Ortopedia Directa, 2015)

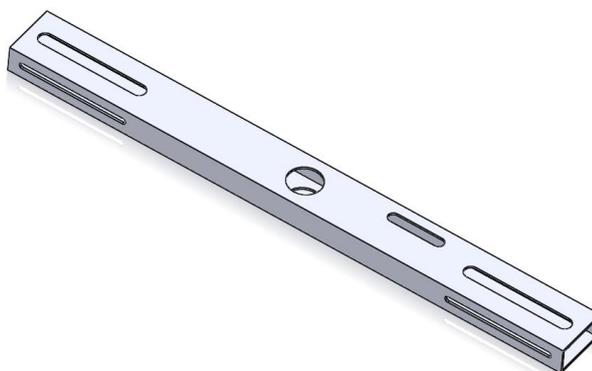


Figura 40. Perfil de soporte.

El largo del perfil se diseñó de 500mm (50cm), sobre este perfil irán varias ranuras las cuales permitirán adaptar el módulo a diferentes anchos de sillas.

Las ranuras laterales nos permitirán realizar esta adaptación, estas ranuras son simétricas en el perfil, tienen un largo de 100mm y un ancho de 6.35mm, las mismas que atraviesan todo el perfil (figura 41).



Figura 41. Ranuras laterales.

Las ranuras superiores también son simétricas respecto al perfil, estas ranuras nos permitirán regular la altura del módulo respecto al piso (Figura 42).

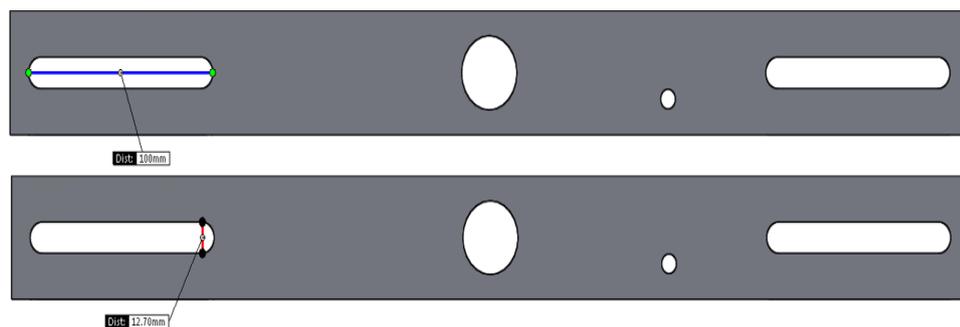


Figura 42. Ranuras superiores.

Sujeción Vertical

El módulo Mecatrónico ira sujeto en de dos maneras; verticalmente y horizontalmente. Para sujetar verticalmente utilizaremos dos abrazaderas tipo U con las cuales sujetaremos el perfil con los tubos verticales de la silla (Figura 44).

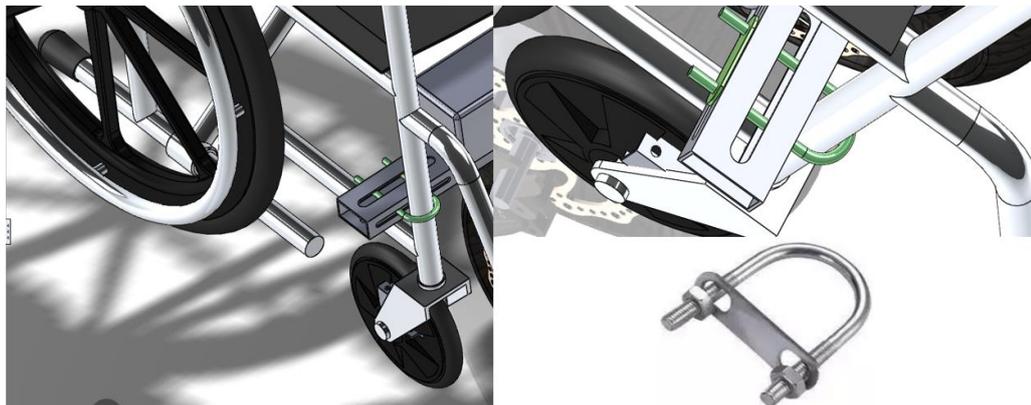


Figura 43. Sujeción vertical.

3.3.1.2 Eje Roscado

Por cada ranura superior pasará un eje roscado el cual nos permitirá regular la altura, el eje ira sujeto al perfil por medio de tuercas.

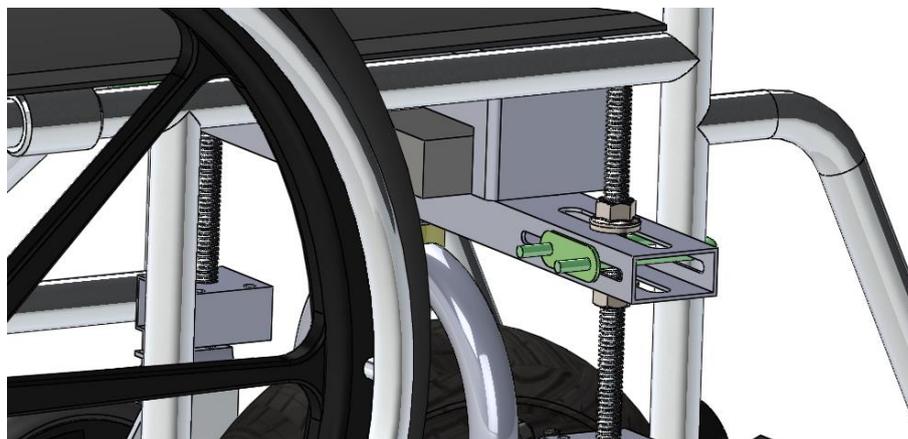


Figura 44. Eje roscado para regular la altura.

3.3.1.3 Fork (trinche)

El trinche debe poder girar en su propio eje y para ello hacemos uso de rodamientos, para la selección de los mismos utilizaremos rodamientos de bicicletas los cuales están diseñados para soportar cargas y esfuerzos generados por el peso de una persona.



Figura 45. Rodamientos XR.

Fuente: (Cixi Xuri Bicycle Industry, 2016)

Una vez adquirido los rodamientos y en función de sus medidas procedemos al diseñar el eje que permitirá el giro de la rueda.

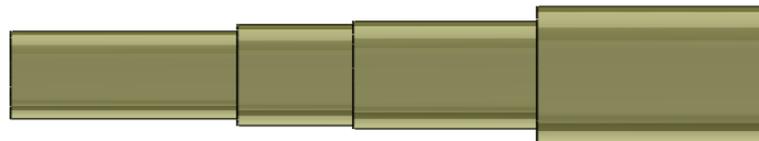


Figura 46. Eje de giro para la rueda.

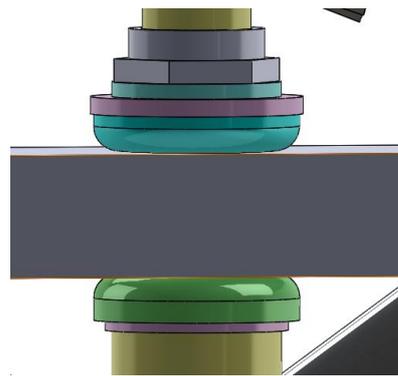


Figura 47. Montaje de rodamientos y perfil.

El eje ira unido a arco de sección tubular, la unión se lo realizara con suelda MIG que es ideal para soldar tubos, luego se ejecutara un análisis de esfuerzos para saber si el trinche soportara las cargas.

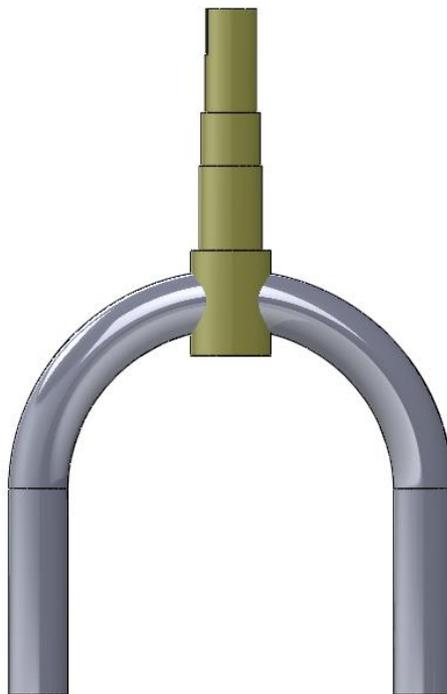


Figura 48. Unión de tubo con arco.

Los sujetadores son diseñados en función de las dimensiones del eje de la rueda. Debe haber un ángulo inclinación en el sujetador, este ángulo esta normado ya que de esto depende la funcionalidad de la silla, para nuestro caso utilizaremos un ángulo de 73 grados el cual nos permite tener una robustez de acuerdo a los estándares normalizados.

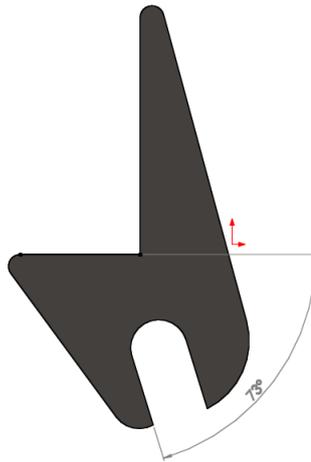


Figura 49. Sujetador con ángulo de inclinación.

3.3.2.5 Cálculos de fuerzas

Cálculo de reacciones

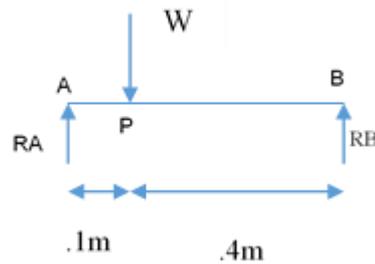


Figura 50 Cálculo de reacciones

$$\sum f_y = 0$$

$$R_A + R_B - W_p = 0$$

Poniendo como estimado que la capacidad del módulo será para una persona de $W_p = 100Kgf$.

$$R_A + R_B = 100Kgf \quad (3.3.1)$$

$$\sum M_C = 0$$

$$-0.1R_A + 0.4R_B = 0$$

$$R_A = 4R_B \quad (3.3.2)$$

(3.3.2) en (3.3.1)

$$4R_B + R_B = 100Kgf$$

$$5R_B = 100Kgf$$

$$R_B = 20Kgf$$

$$R_B = 196.2 N$$

$$R_A = 4R_B = 4 * 196.2$$

$$R_A = 784.8 N$$

Los valores de los componentes del módulo son:

- Peso de la batería

$$W_B = 4kgf = 39.24 N$$

- Peso de la silla de ruedas sobre la llanta delantera

$$W_{S1} = 5kgf = 49.05N$$

- Peso estimado de la estructura del módulo adaptable

$$W_{S2} = 5 kgf = 49.05 N$$

- Peso del motor

$$W_m = 5 kgf = 49.05 N$$

Si se calcula el peso que recae sobre cada extremo del trinche se tendría.

$$W_T = (R_B + W_B + W_{S1} + W_{S2})/2$$

$$W_T = (196.2 + 39.24 + 49.05 + 49.05)/2$$

$$W_T = 166.77 \text{ N}$$

El cálculo de la normal que ejercería sobre la rueda será:

$$N_r = R_B + W_B + W_{S1} + W_{S2} + W_m$$

$$N_r = 196.2 + 39.24 + 49.05 + 49.05 + 49.05$$

$$N_r = 382.59 \text{ N}$$

Para realizar el cálculo del momento requerido para girar el motor. Antes de realizar este cálculo vemos cual es el área de contacto entre la llanta y el piso en el peor de los casos, cuando se encuentra totalmente desinflada:

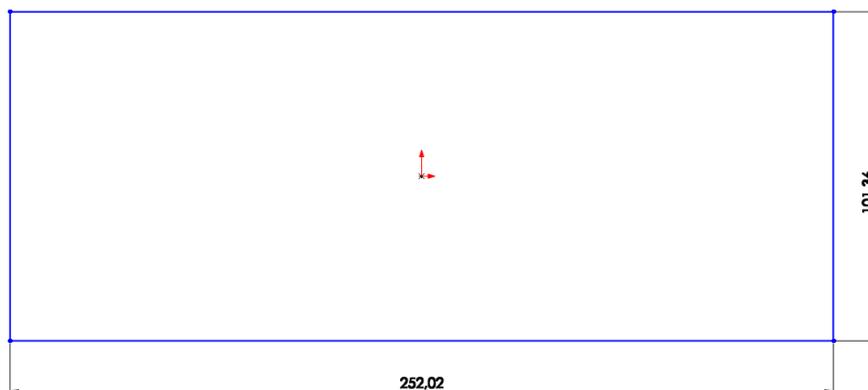


Figura 51: Área de contacto de la rueda en el peor de los caso

$$x = 252.02 \text{ mm} = 0.252 \text{ m}$$

$$y = 101.36 \text{ mm} = 0.101 \text{ m}$$

$$A = x * y$$

$$A = 0.025452 \text{ m}^2$$

La resistencia a la rodadura del neumático desinflado con respecto al piso quedaría definida de la siguiente manera:

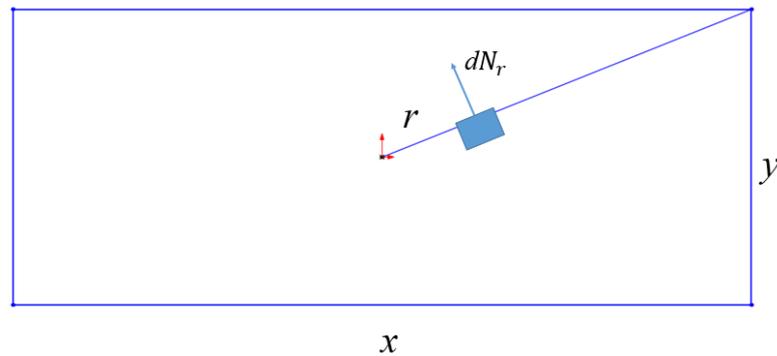


Figura 52 Movimiento del diferencial

$$dM_r = u dN_r$$

$$r = \sqrt{x^2 + y^2}$$

$$dN_r = \frac{N}{A} dx dy$$

$$dN_r = \frac{uN}{A} \sqrt{x^2 + y^2} dx dy$$

$$M_r = \int_{-\frac{x}{2}}^{\frac{x}{2}} \int_{-\frac{y}{2}}^{\frac{y}{2}} \frac{uN_r}{A} \sqrt{x^2 + y^2} dy dx$$

$$M_r = \frac{uN_r}{A} \int_{-\frac{x}{2}}^{\frac{x}{2}} \left(\int_{-\frac{y}{2}}^{\frac{y}{2}} \sqrt{x^2 + y^2} dy \right) dx$$

$$M_r = \frac{uN_r}{A} \int_{-\frac{x}{2}}^{\frac{x}{2}} \left(\int_{-\frac{y}{2}}^{\frac{y}{2}} \sqrt{x^2 + y^2} dy \right) dx$$

Para el cálculo de la integral doble se utilizó la siguiente función en Matlab:

```
x=0.252;
```

```
y=0.101;
```

```
f=@(x,y) sqrt(x.^2+y.^2);
```

Resultado=integral2(f,-x/2,x/2,-y/2,y/2)

Resultado=0.0018

$$M_r = \frac{uN_r}{A} 0.0018$$

Para realizar los cálculos se tomó el rozamiento máximo que puede existir entre el neumático y el piso para ello se tomó el valor de $u=1$.

$$M_r = \frac{1 * N_r}{A} 0.0018$$

$$N_r = 382.59 \text{ N}$$

$$M_r = \frac{1 * 382.59}{0.252 * 0.101} 0.0018$$

$$M_r = 27.0573 \text{ Nm}$$

Cálculo de fuerzas que ejerce el motor sobre los soportes del trinche.

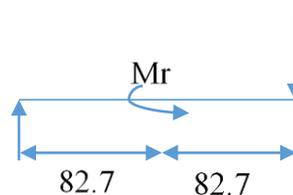


Figura 53 Cálculo del torque del motor

La fuerza que ejerce el torque del motor es de $M_r = 27.0573 \text{ Nm}$, primero se calcula la fuerza que recae en cada sujetador.

$$\sum M = 0$$

$$2F(0.0827) - 27.0573 = 0$$

$$F = 164.78 \text{ N}$$

3.3.1.4 Sujeción Horizontal

Está conformada por dos piezas la primera es un perfil que está unido por medio de tuercas al eje roscado, la segunda pieza es de forma semicircular que se adapta a la forma del tubo de la silla y se une a la primera pieza por medio de pernos.

Debido a que los ejes estarán sometidos a cargas axiales se realizará un análisis de tracción tomando en cuenta que la fuerza total que está ejercida en los ejes roscados es de: $R_B = 196.2 \text{ N}$

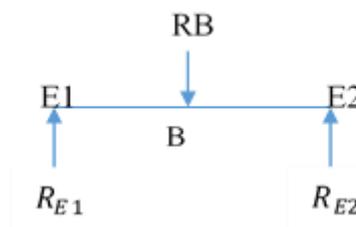


Figura 54 Cálculo de reacciones sobre los soportes

Cálculo de la fuerza en cada eje:

$$\sum F_y = 0$$

$$R_B = R_{E1} + R_{E2}$$

Como la fuerza en cada eje es la misma tenemos:

$$R_{E1} = R_{E2} = R_E$$

$$R_B = 2R_E$$

$$R_E = 196.2/2$$

$$R_E = 98.1 \text{ N}$$

Se utilizará una fuerza de 100N para proceder hacer el análisis de esfuerzos. Al final se obtuvo un factor de seguridad mínimo de 55, esto quiere decir que el eje seleccionado de ½ pulgada (12.7 mm) soportará los esfuerzos a los que estará sometido.

De igual manera se realizará un análisis de tracción para los pernos que sujetaran los ejes roscados.

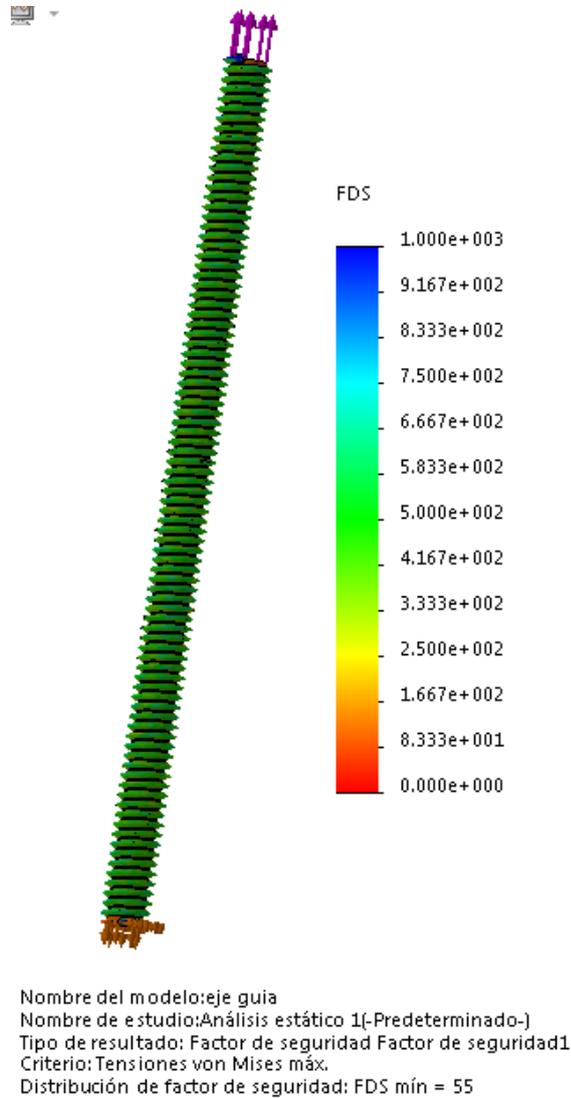


Figura 55. Factor de seguridad en eje roscado.

Calculo de fuerzas en los pernos:

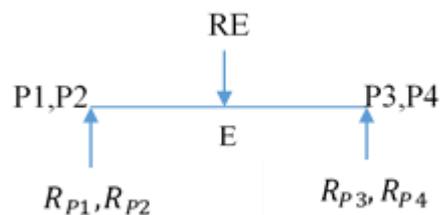


Figura 56 Cálculo de fuerzas en los pernos

La fuerza de tracción de eje se divide por igual en cada uno de los pernos, por lo tanto tenemos: $R_{P1} = R_{P2} = R_{P3} = R_{P4} = R_P$

$$\sum F_y = 0$$

$$R_E = R_{P1} + R_{P2} + R_{P3} + R_{P4}$$

$$R_E = 4R_P$$

$$R_P = 98.1/4$$

$$R_P = 24.53 \text{ N}$$

La simulación se realizará con una fuerza de 30N para tener un factor de seguridad alto.

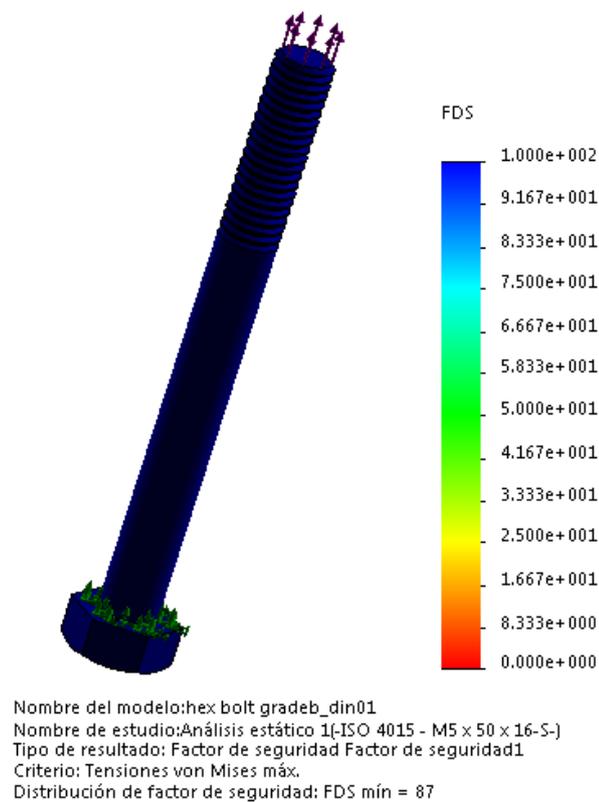


Figura 57. Factor de seguridad en perno M5.

El factor de seguridad es bastante aceptable, los pernos M5 soportaran los esfuerzos generados por el modulo.



Figura 58. Sujeción horizontal y sus partes.

3.3.1.5 en el soporte del trinche.

Una vez definida la geometría

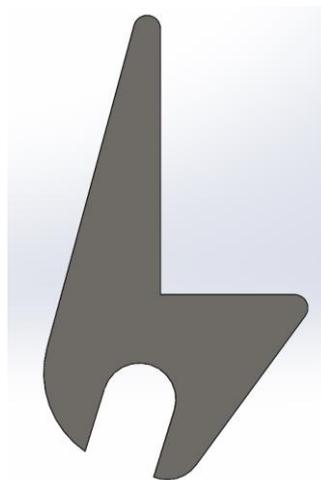


Figura 59. Geometría del soporte.

Se procede a hacer el cálculo de esfuerzos, para definir el espesor de la placa. Dicho cálculo se lo realizo por el método de elementos finitos.

3.3.1.5.1 Carga axial del sujetador

Primero por la carga axial ejercida por el peso de toda la estructura sobre la tenaza; por cálculos anteriores dio un valor de $W_T = 166.77 \text{ N}$. Que sería aplicada sobre las superficies de contacto del trinche con los sujetadores. La parte fija será en donde se sujeta al motor mediante los pernos. Por último el material escogido es el acero A36.

Para un espesor de 3mm.

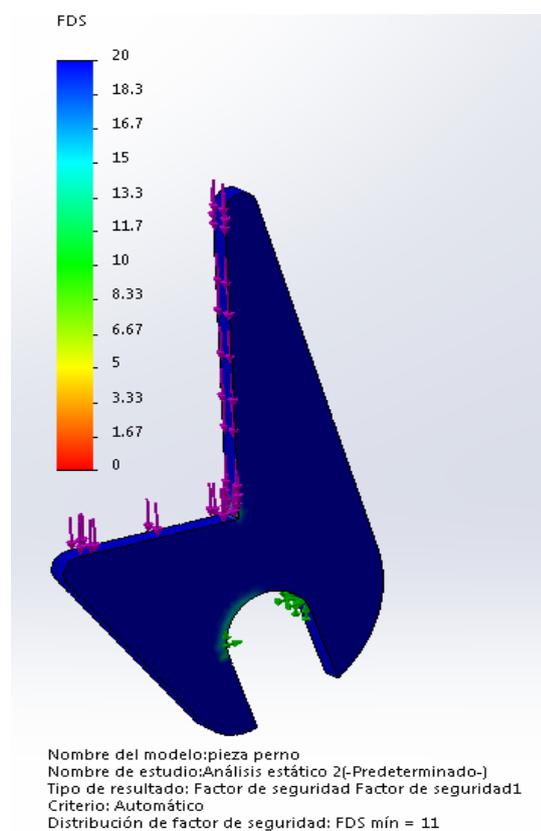


Figura 60. Factor de seguridad para el soporte con un espesor de 3mm.

El factor de seguridad es alto y que la pieza resiste la carga. El mínimo factor de seguridad es de 11.

Para un espesor de 4mm.

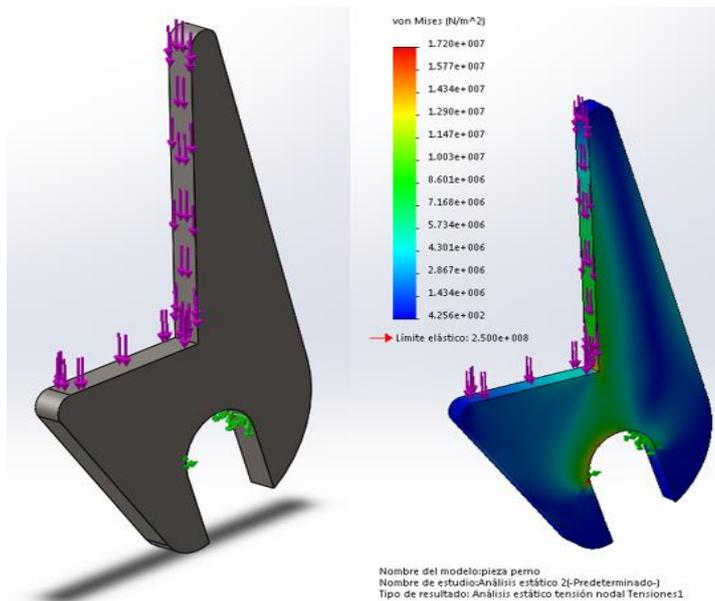


Figura 61. Limite elástico para un espeso de 4mm.

En el gráfico se puede observar que no presenta esfuerzos mayores al límite elástico.

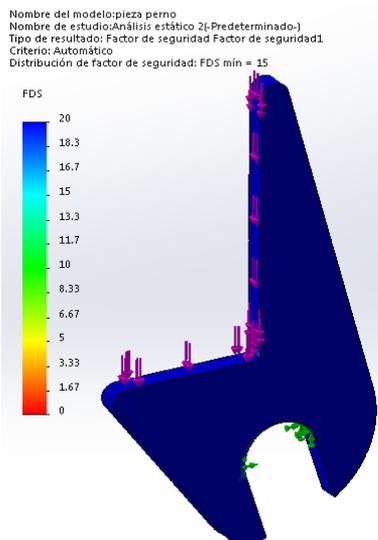


Figura 62. Factor de seguridad para el soporte con un espesor de 4mm.

El factor de seguridad más bajo obtenido en la pieza fue de 15. Por lo que la pieza soporta la carga ejercida.

3.3.1.5.2 Momento cortante torsor del sujetador

Antes de disminuir más el espesor de la placa se hizo el cálculo de los esfuerzos producido por el torque al girar la rueda. Con la fuerza de torsión calculada de $F = 164.78N$

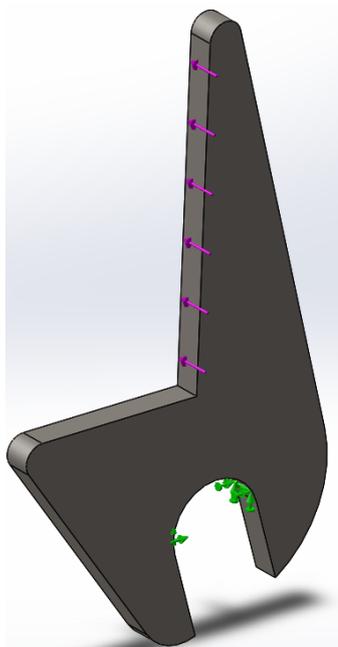


Figura 63. Cargas respecto al momento cortante.

Para un espesor de 3mm

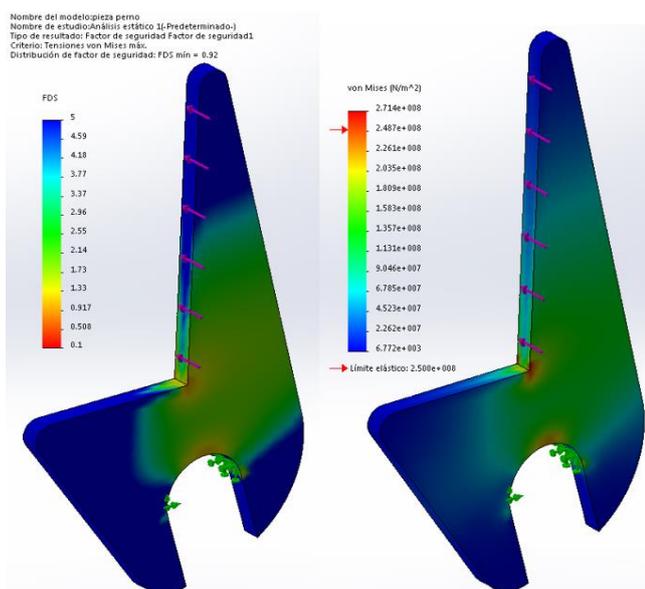


Figura 64. Factor de seguridad y esfuerzo de von mises para un espesor de 3mm.

Se obtuvieron esfuerzos mayores al límite elástico. Con un factor de seguridad mínimo de 0.92.

Para un espesor de 4mm

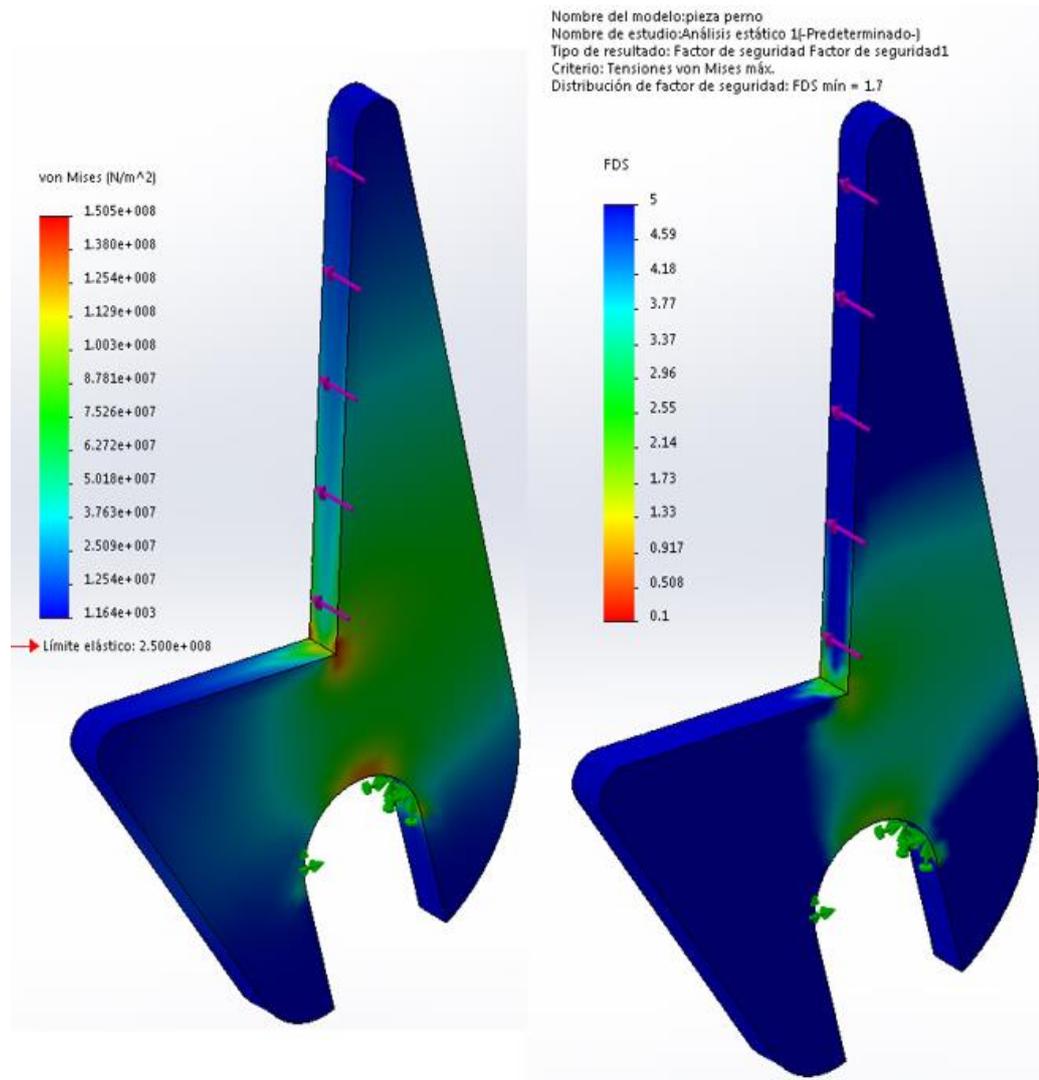


Figura 65. Factor de seguridad y esfuerzo de von mises, espesor de 4mm.

Se obtuvo un factor de seguridad mínimo de 1.7, por el cual el espesor escogido es de 4mm.

3.3.1.5.3 Carga axial del arco

En la siguiente parte del trinche que ya fue definida la geometría

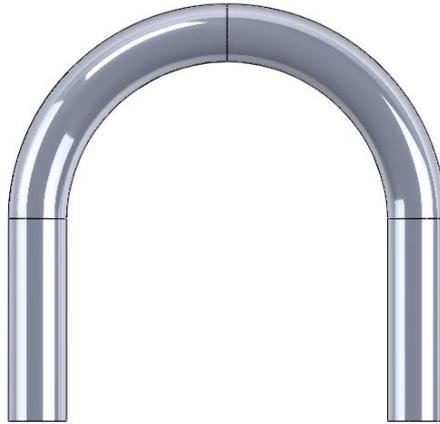


Figura 66. Geometría del arco.

Se procede a realizar el cálculo de esfuerzos, utilizando las reacciones anteriormente calculadas. El cálculo se lo realizo por el método de elementos finitos.

La carga axial ejercida por el peso de toda la estructura sobre el eje dio un valor de $N_r = 382.59 N$. Que sería aplicada en el eje con sus respectivas reacciones. El material escogido es el acero A36. Para realizar la simulación redondearemos el valor de la fuerza normal generada de $382.59 N$ a $390 N$.

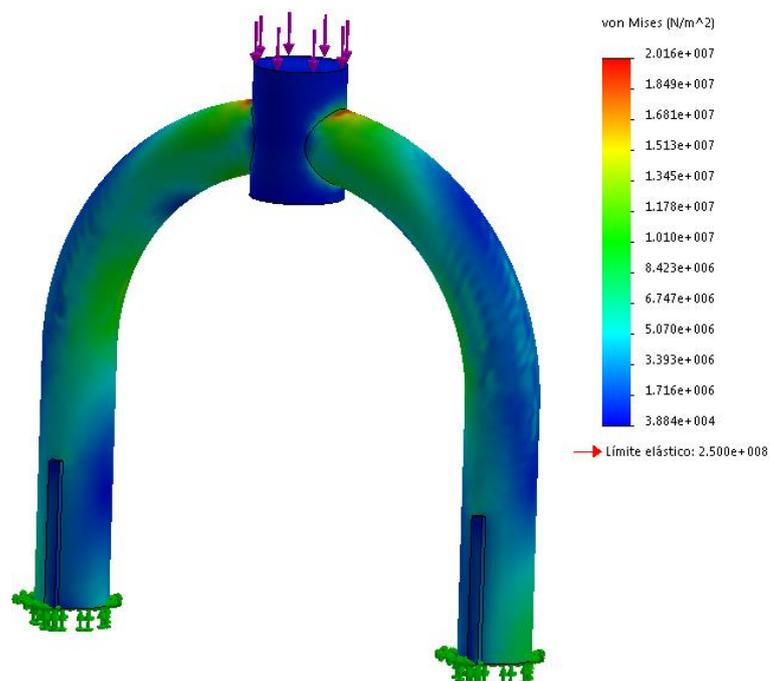


Figura 67. Limite elástico sobre el arco.

El diámetro del tubo es de 1 pulg (25.4 mm) y su espesor de 2.5 mm, como podemos observar en el gráfico no se presentan esfuerzos mayores al límite elástico.

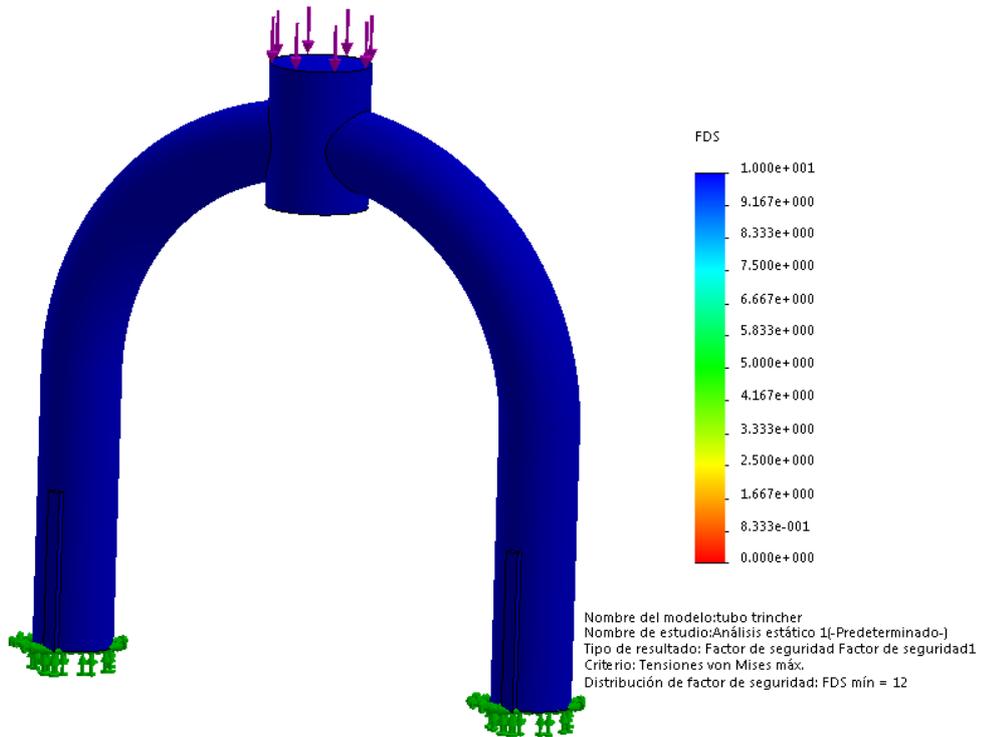


Figura 68. Factor de seguridad del arco.

El factor de seguridad más bajo obtenido en la pieza es de 12. Esta pieza soporta la carga de compresión ejercida por la fuerza normal de 382.59 N.

3.3.1.5.4 Momento torsor aplicado sobre arco

Para realizar el análisis esfuerzos producido por el torque al girar la rueda se utiliza la fuerza de torsión calculada de $F = 164.78N$ que estará aplicada en cada terminal.

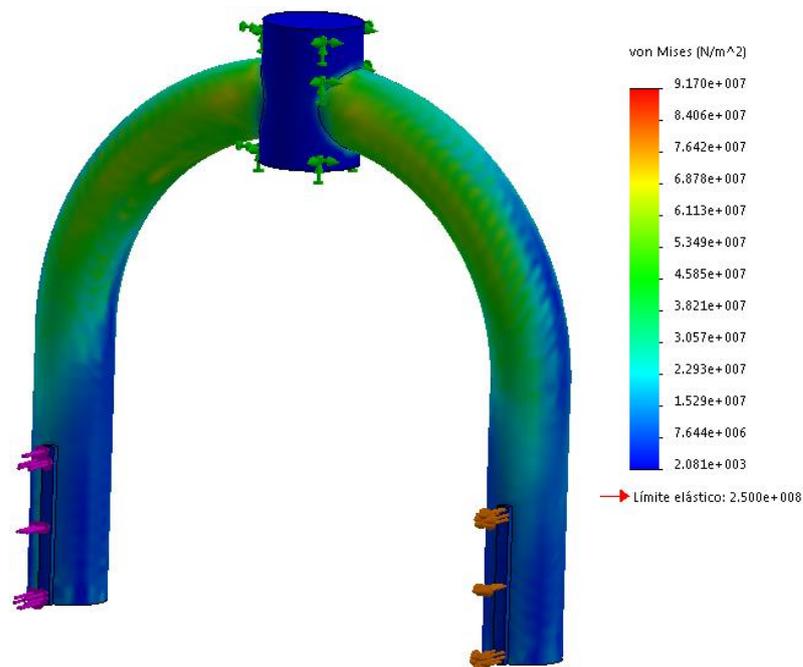


Figura 69. Esfuerzos de Von Mises generados sobre el arco.

Analizando el esfuerzo de Von mises no se presentan esfuerzos mayores al límite elástico.

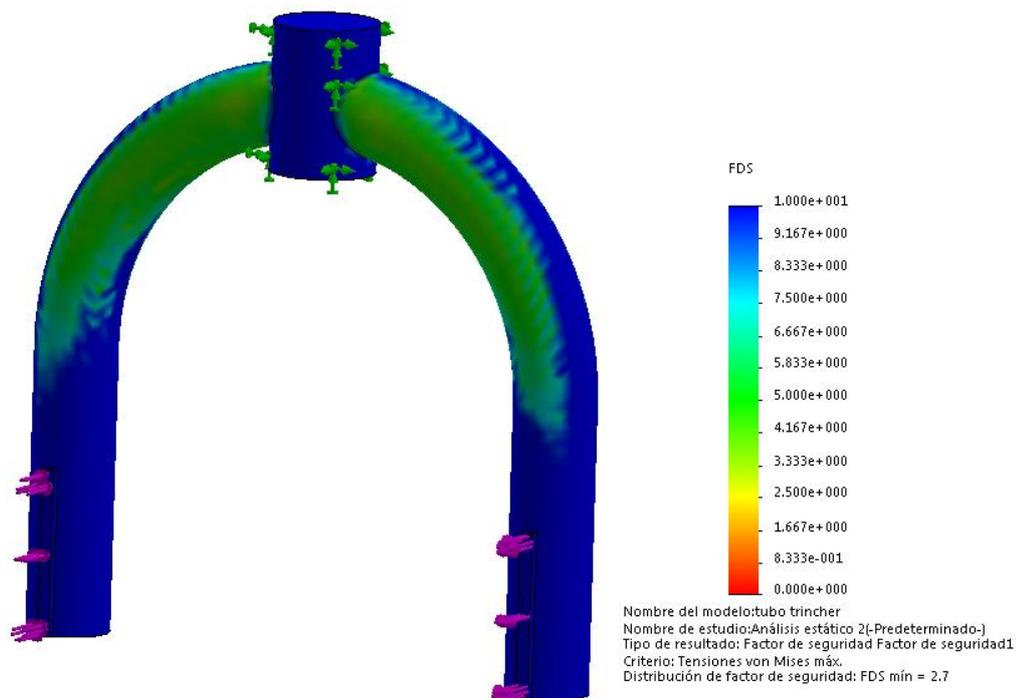


Figura 70. Factor de seguridad mínimo que tiene el arco.

En la figura se observa que el factor de seguridad obtenido es de 2.7 por lo que el diseño del trinche soportara sin ningún problema el esfuerzo de torsión al que está sometido.

3.3.1.6 Disco para el frenado

Para realizar el frenado de la rueda se hará uso de un disco con su respectiva mordaza, las consideraciones a tener sobre el disco son; el espesor y su diámetro, estos dos factores deben estar acorde a las dimensiones del trinche para que le disco pueda girar libremente. Se utilizará un disco de 160mm de diámetro con un espesor de 2mm y con 3 agujeros de sujeción para acoplar a la rueda.

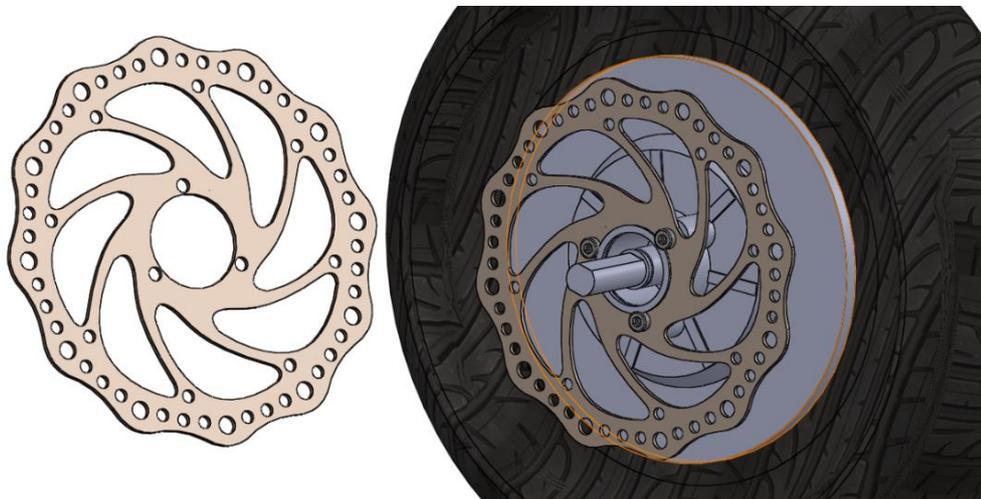


Figura 71. Unión del disco con la rueda.

3.3.1.7 Unión de la rueda con el trinche

La unión se lo realizo por medio de tuercas debido a que el eje de la rueda es roscado, sobre el eje van los sujetadores que se acoplan a través de las ranuras que tienen.

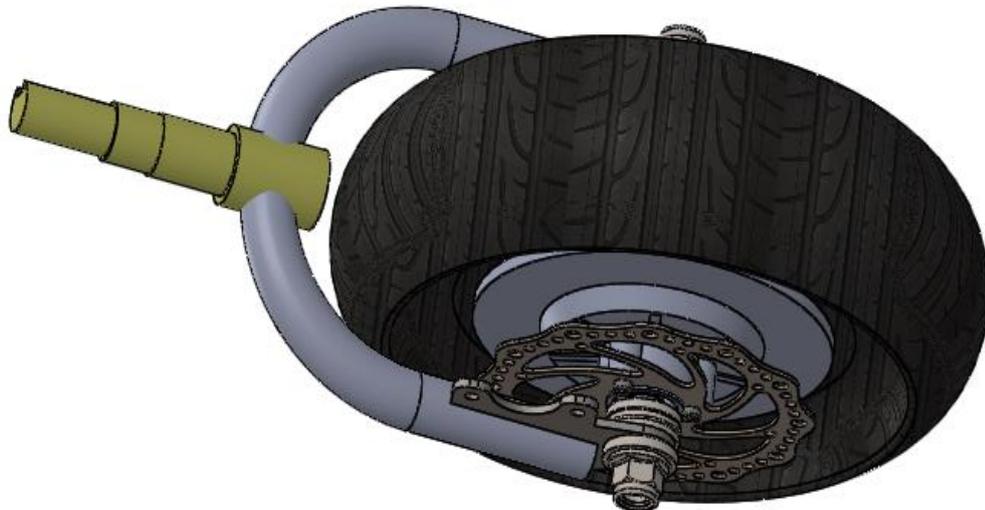


Figura 72. Montaje de la rueda con el trinche.

3.3.1.8 Montaje sobre el perfil

Una vez realizado el diseño de las partes principales del módulo procedemos al montaje con el perfil, con este montaje podremos comprobar que no exista interferencias o cruce entre piezas y ver que haya un libre movimiento del trinche.

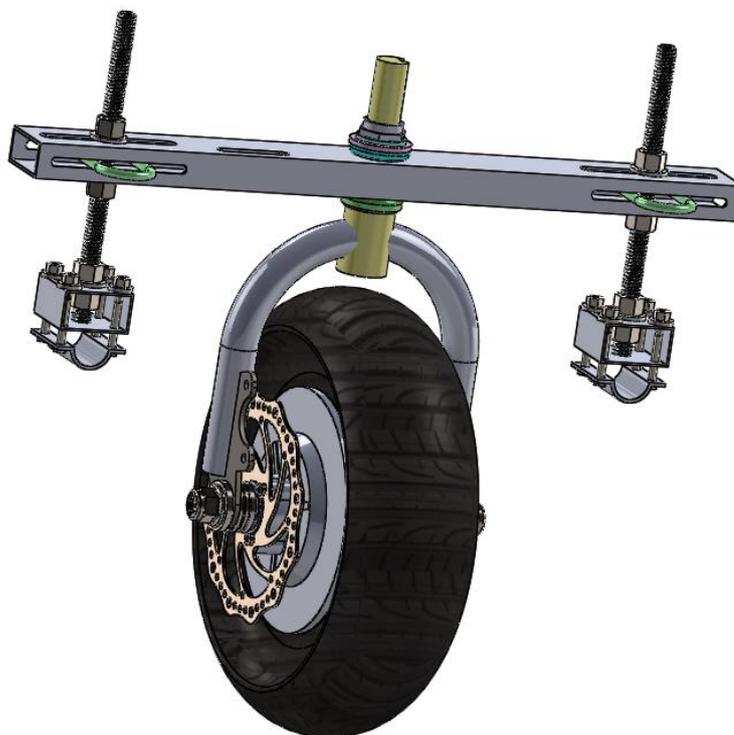


Figura 73. Montaje principal sobre el perfil.

3.3.1.9 Acople de la mordaza

La mordaza ira acoplada al trinche por medio del sujetador que tiene los orificios coincidentes con los agujeros roscados de la mordaza para unir por medio de pernos.



Figura 74. Unión de la mordaza con el trinche.

3.3.1.10 Diseño de engranes cónicos

Los engranes cónicos nos permitirán que la rueda pueda tener dirección, es decir que pueda girar hacia la izquierda o derecha, anteriormente habíamos calculado el torque necesario para girar la rueda en el peor de los casos que es de $M_r = 27.0573 \text{ Nm}$, para poder tener al menos este torque realizaremos un diseño con una relación de 3 a 1, los engranes cónicos serán impulsados por un motor a pasos con una capacidad de 15.5 Nm y en el cual ira montado el piñon por lo que el mismo será diseñado en función de eje del motor a pasos.

Condiciones de diseño:

Torque requerido: 27.0573 Nm

Relación de engranes: 3:1

Torque del motor a pasos: 15.5 Nm

Torque obtenido con la relación de engranes: $15.5 * 3 = 46.5 \text{ Nm}$

Con el torque obtenido de 46.5Nm el factor de seguridad es bastante confiable y podremos que el sistema obtendrá el torque necesario para girar la rueda.

En las siguientes figuras se muestra las dimensiones normalizadas de los engranes cónicos:

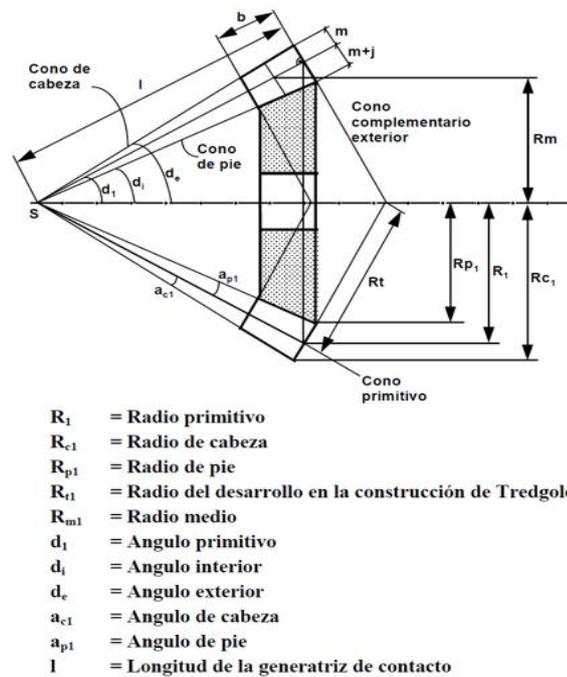


Figura 75. Dimensiones de los engranes cónicos.

Fuente: (Hernández, 2007)

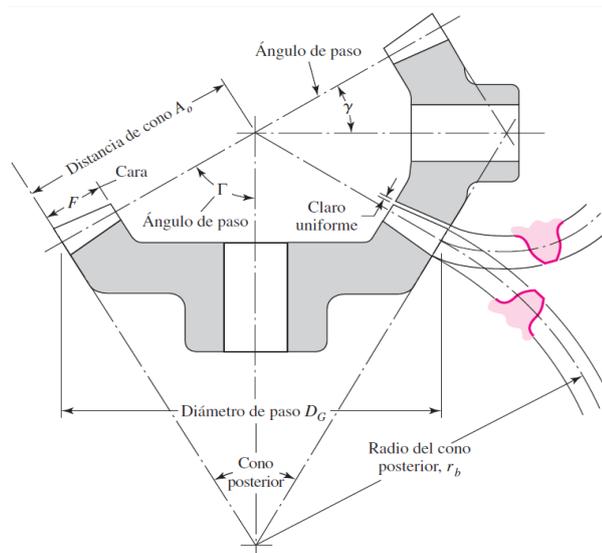


Figura 76. Acoplamiento de engranes.

Fuente: (Hernández, 2007)

Datos iniciales para los engranes.

Tabla 1.

Parámetros iniciales para los engranes.

Datos de entrada	Valor
Ángulo de presión ϕ (PA)	20
Módulo (MDL)	1,5

Calculo de parámetros de diseño para el engrane de 15 dientes:

Tabla 2.

Parámetros de diseño, engrane de 15 dientes

Parámetros	Valor
Diámetro de paso (PD)mm	22,5
Addendum (ADD) mm	1,5
Dedendum (DED) mm	1,7355
Profundidad total (WD) mm	3,2355
Holgura (CL) mm	0,2355
Diámetro externo (OD) mm	25,5
Diámetro de raíz (RD) mm	19,029
Círculo base (BC) mm	21,14308397
Paso circular (CP) mm	4,71238898
Espesor circular diente (T) mm	2,35619449
Espesor de cabeza de diente (ADDc) mm	1,561685028
Profundidad de trabajo (WKG) mm	3
Ángulo del diente (grados)	12,00
Radio R de Grant (mm)	4,23
Radio r de Grant (mm)	2,01
Ángulo cónico de paso (grados)	18,43494882
Radio cónico de paso (mm)	35,57562368
Ancho de cara 2.5 CP (mm)	11,78097245
Ancho de cara 1.5 CP (mm)	7,068583471
Altura de Addendum angular (mm)	1,423024947
Diámetro exterior	25,34604989
Radio exterior	12,67302495
Angulo de Addendum (grados)	2,41437212
Ángulo de Dedendum (grados)	2,792869019

Calculo de parámetros de diseño para el engrane de 45 dientes:

Tabla 3.

Parámetros de diseño, engrane de 45 dientes.

Parámetros	Valor
Diámetro de paso (PD)mm	67,5
Addendum (ADD) mm	1,5
Dedendum (DED) mm	1,7355
Profundidad total (WD) mm	3,2355
Holgura (CL) mm	0,2355
Diámetro externo (OD) mm	70,5
Diámetro de raíz (RD) mm	64,029
Círculo base (BC) mm	63,4292519
Paso circular (CP) mm	4,71238898
Espesor circular diente (T) mm	2,35619449
Espesor de cabeza de diente (ADDc) mm	1,520561676
Profundidad de trabajo (WKG) mm	3
Ángulo del diente (grados)	4,00
Radio R de Grant (mm)	6,945
Radio r de Grant (mm)	6,945
Ángulo cónico de paso (grados)	71,56505118
Radio cónico de paso (mm)	35,57562368
Ancho de cara 2.5 CP (mm)	11,78097245
Ancho de cara 1.5 CP (mm)	7,068583471
Altura de Addendum angular (mm)	0,474341649
Diámetro exterior	68,4486833
Radio exterior	34,22434165
Angulo de Addendum (grados)	2,41437212
Ángulo de Dedendum (grados)	2,792869019

Para determinar los esfuerzos transversales del diente y la durabilidad se hizo uso de las formulas de la AGMA (American Gear Manufacturers Association) que se muestran a continuación:

Carga transmitida W_t (Fuerza tangencial)

$$W_t = \frac{T}{r_{prom}}$$

Donde T es el par de torsión y r_{prom} es el radio de paso

Tabla 4.

Fuerzas que actúan en un engrane cónico.

Para el piñon (15 dientes)	Para la rueda (45 dientes)
Fuerza radial:	Fuerza radial:
$W_r = W_t \tan \phi \cos \gamma$	$W_r = W_t \tan \phi \sin \gamma$
Fuerza axial:	Fuerza axial:
$W_a = W_t \tan \phi \sin \gamma$	$W_r = W_t \tan \phi \cos \gamma$

Donde ϕ es el ángulo de presión y γ es el ángulo de paso que viene dado por la siguiente formula:

$$\tan \gamma = \frac{Np}{NG}$$

Carga transmitida W_t (Fuerza tangencial) para el piñon (15 dientes):

$$W_t = \frac{T}{r_{prom}}$$

$$W_t = \frac{15.5Nm}{\frac{18.75}{2 * 1000}m}$$

$$W_t = 1.65KN$$

$$\tan \gamma = \frac{Np}{NG}$$

$$\tan\gamma = \frac{15}{45}$$

$$\gamma = 18.43^\circ$$

Fuerza radial:

$$W_r = W_t \tan\phi \cos\gamma$$

$$W_r = 1.65KN * \tan 20 * \cos 18.43$$

$$W_r = 0.57KN$$

Fuerza axial:

$$W_a = W_t \tan\phi \sin\gamma$$

$$W_a = 1.65KN * \tan 20 * \sin 18.43$$

$$W_a = 0.18KN$$

Carga transmitida W_t (Fuerza tangencial) para la rueda (45 dientes):

$$W_t = \frac{T}{r_{prom}}$$

$$W_t = \frac{15.5 * 3Nm}{\frac{18.75}{2 * 1000}m}$$

$$W_t = 4.95KN$$

$$\tan\gamma = \frac{Np}{NG}$$

$$\tan\gamma = \frac{45}{15}$$

$$\gamma = 71.56^\circ$$

Fuerza radial:

$$W_r = W_t \tan \phi \operatorname{sen} \gamma$$

$$W_r = 4.95 \text{KN} * \tan 20 * \operatorname{sen} 18.43$$

$$W_r = 1.71 \text{KN}$$

Fuerza axial:

$$W_a = W_t \tan \phi \operatorname{cos} \gamma$$

$$W_a = 1.65 \text{KN} * \tan 20 * \operatorname{cos} 18.43$$

$$W_a = 0.57 \text{KN}$$

Luego de hacer los cálculos necesarios y tomando en cuenta que el diámetro interno del piñón estará acorde al eje del motor a pasos para que se pueda realizar el acople, procedemos a dibujar el engrane:



Figura 77. Engranés diseñados de 15 y 45 dientes.

3.3.1.11 Acople del piñón con motor a pasos

El acople se lo realiza por medio de la chaveta y con prisioneros que van sobre el engrane, el piñón recibirá el torque del motor a pasos para luego transmitirlo a la rueda.

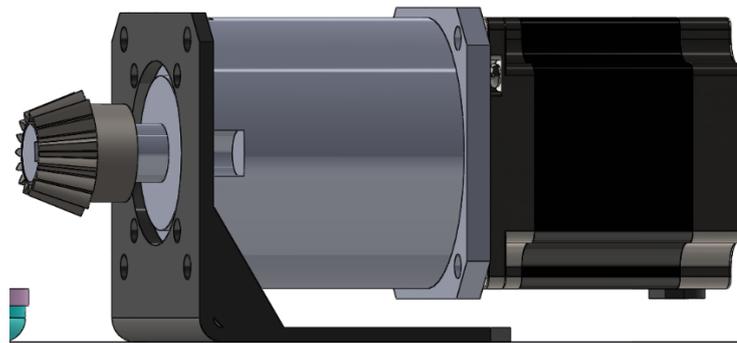


Figura 78. Acople piñón y motor a pasos.

3.3.1.12 Montaje de los engranes

El montaje se lo realizo a través de los ejes perpendiculares tanto de trinche como del motor a pasos, se pudo comprobar en el software que los engranes se acoplan perfectamente sin ninguna interferencia.

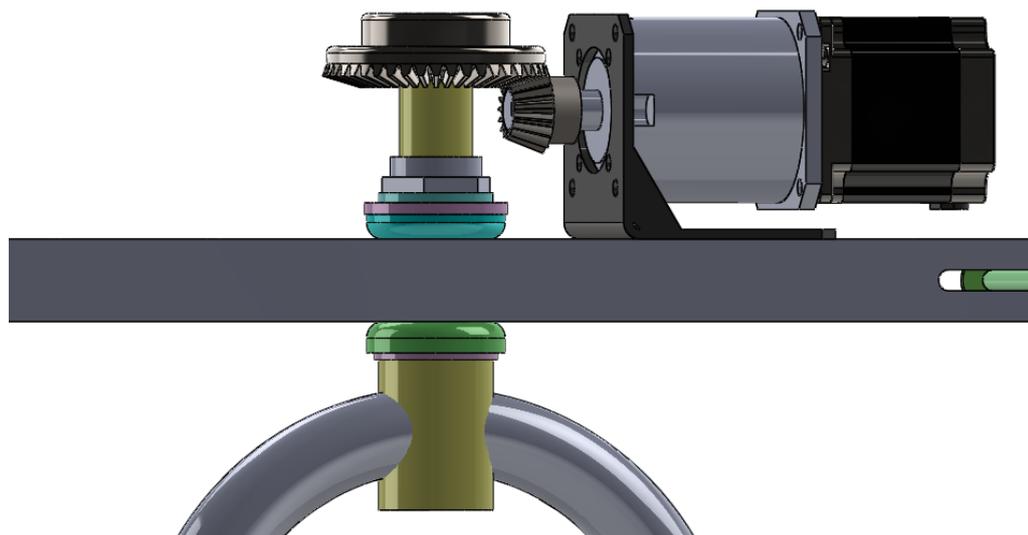


Figura 79. Acoplamiento de engranes.

3.3.1.13 Requerimientos para el frenado

El modulo se detendrá por el accionamiento del freno de disco, debido a que este accionamiento será automático se requiere saber la fuerza necesaria para accionar el freno por medio de la mordaza, para saber la magnitud de esta fuerza se acudió al laboratorio de física y se midió con la ayuda de un Newton metro obteniendo un valor de 1.5Nm. Para obtener un factor de seguridad aceptable utilizaremos un servomotor con un torque de 2Nm.

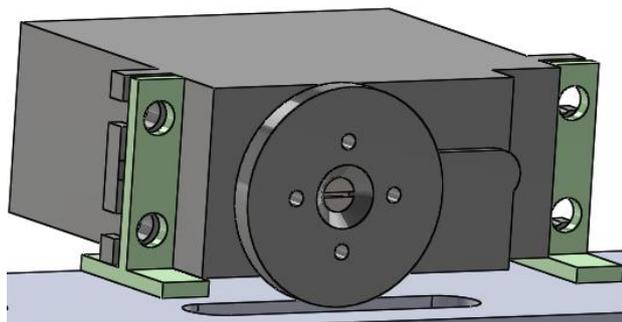


Figura 80. Servomotor y soportes para el accionamiento del freno.

3.3.1.14 Protección para engranes y freno

Para evitar accidentes especialmente con los engranes cónicos y con el frenado, se cubrirán todas estas partes con una caja la cual ayudara a prevenir accidentes.

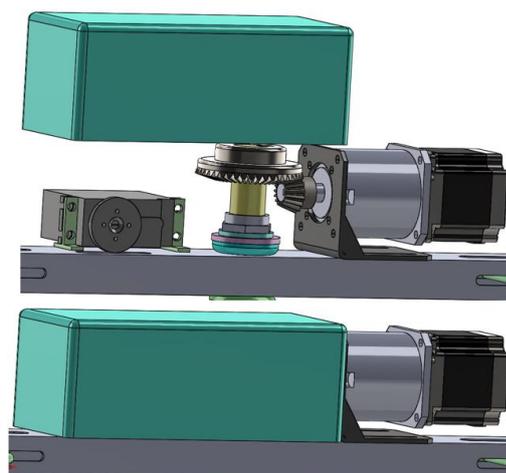


Figura 81. Caja para protección.

3.3.1.15 Caja para drivers y tarjeta de control

Todos los elementos electrónicos irán montados sobre una caja la misma que contará con los agujeros necesarios para poder empernar los drivers y la tarjeta de control, esta caja estará compuesta por dos partes; la base que es fija en la cual irán sujetos los elementos y la tapa que es deslizable para tener un fácil acceso a la electrónica. Las dimensiones de esta caja se definirán en la parte electrónica una vez seleccionado los drives para trabajar.

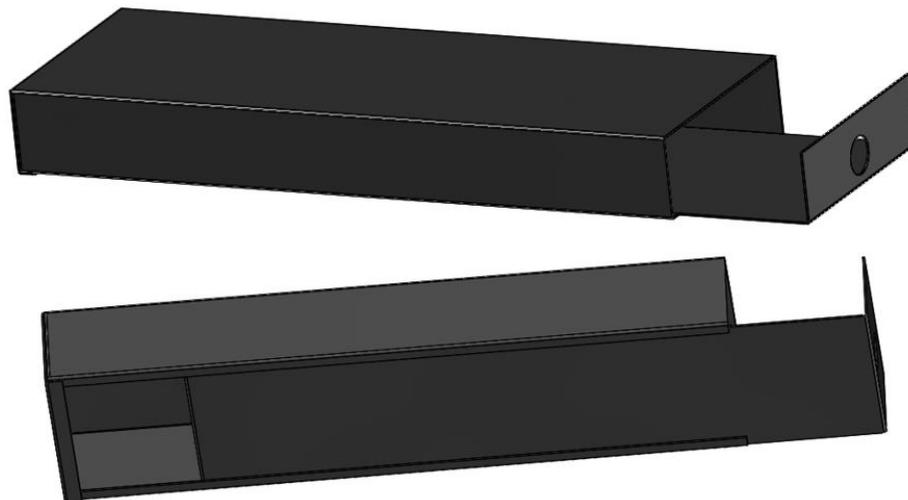


Figura 82. Caja para elementos electrónicos.

3.4 Diseño Electrónico

3.4.1 Requerimientos y selección de componentes.

Para empezar con el diseño electrónico, primero estableceremos los componentes con los que trabajaremos. Para esto se verán los requerimientos en cada una de las funciones que debe ejecutar proyecto, y estas funciones son:

- Receptar y comandar instrucciones
- Avanzar y retrocede
- Girar
- Frenar

3.4.1.1 Receptar y comandar instrucciones.

Para esta función, se deberá incluir una placa con procesador que tengan las siguientes características:

- Facilidad para conexión Bluetooth: esto con el fin de poder comunicarse con el electrodo Mindwave.
- Entradas y salidas digitales: son necesarias para enviar instrucciones.
- Alta velocidad de procesamiento: al momento de procesar las señales EEG es fundamental una velocidad alta de procesamiento.
- Salidas PWM: para el control de servomotor)
- Ejecución en paralelo: dado que tendremos que ejecutar varias funciones al mismo tiempo, es necesario poder ejecutarlas en paralelo.



Figura 83. Raspberry Pi 2.

Fuente: (Raspberry Pi Foundation, 2016)

Con todos los requerimientos anteriores se ha seleccionado la single board computer “Raspberry Pi”. A más de contar con los requerimientos antes dichos, ya se cuenta con previa experiencia utilizándola, que nos facilitará la implementación. En este caso hemos seleccionado el modelo 2, ya que es mucho mejor a comparación del modelo B+. Las características principales se muestran en la figura 14.

	NEW! Raspberry Pi 2	Model B+
Processor Chipset	Broadcom BCM2836 ARMv7 Quad Core Processor powered Single Board Computer running at 900 MHz	Broadcom BCM2835 ARMv6 SoC full HD multimedia applications processor
RAM	1GB SDRAM @ 450 MHz	512 MB SDRAM @ 400 MHz
Storage	MicroSD	MicroSD
USB 2.0	4x USB Ports	4x USB Ports
Power Draw / voltage	1.8A @ 5V	1.8A @ 5V
GPIO	40 pin	40 pin
Ethernet Port	Yes	Yes

Figura 84. Características Raspberry Pi modelo 2 y B+.

Fuente: (Element 14, 2016)

La Raspberry Pi, dado que es una single board computer, se maneja a través de un sistema operativo, y tiene la posibilidad de conectar 4 dispositivos USB. Como se

necesita la comunicación Bluetooth, se lo hará por un adaptador Bluetooth USB Trendnet Versión 1.0, como se aprecia en la figura 16.



Figura 85. Adaptador Bluetooth USB Trendnet Versión 1.0.

Fuente: (Trendnet, 2016)

La Raspberry Pi cuenta con 26 entradas y salidas digitales de propósito general (GPIO, General purpose input output), con 2 salidas de 5v, 2 salidas de 3.3v, y 5 salidas de tierra. En la figura 17 se puede visualizar la localización de cada uno de los pines.

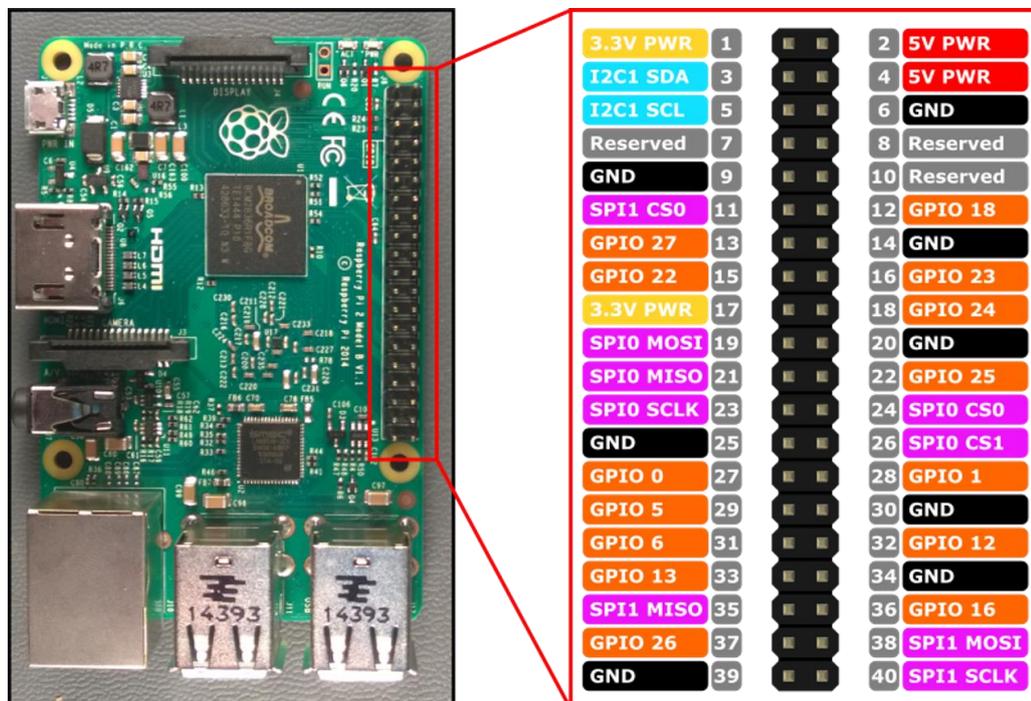


Figura 86. Distribución de pines de la Raspberry Pi 2.

Fuente: (Chia, 2016)

Con respecto a la necesidad de alto procesamiento, la Raspberry Pi modelo 2 cumple con los requerimientos dado que cuenta con un procesador de doble núcleo con una velocidad de 900Mhz, como se puede ver en la figura 15. Ahora con respecto a la posibilidad de salidas PWM, existen diversas librerías que permiten tener salidas PWM en todas las salidas digitales. Y como último requisito teníamos la ejecución de programas en paralelo, que como es una single board computer, el sistema operativo nos va a permitir tener varios programas ejecutándose a la vez. En conclusión, la Raspberry Pi 2 cumple con los requisitos necesarios para la implementación del proyecto.

3.4.1.2 Avanzar y retroceder.

Para que el módulo mecatrónico permita avanzar y retroceder a la silla de ruedas, como se dijo anteriormente se cuenta con una rueda con motor DC brushless interno y su respectivo driver. Las características principales del motor se muestran en la tabla 1. Y el cableado

Tabla 5.

Especificaciones técnicas del motor DC brushless.

Característica	Valor
Potencia	350W
Voltaje	DC 36V/48V
Velocidad sin carga	400-1200rpm
Eficiencia	>85%
Diámetro con llanta	320 mm
Peso	3.5kg (con llanta)



Figura 87. Cables de conexión del motor DC Brushless.

Fuente: (Codd Power, 2016)

De la figura 87, se puede notar que el motor posee sensores de efecto hall, los cuales nos servirán para el control de velocidad del mismo. También podemos ver que tiene tres cables de las fases, dado que es un motor trifásico, aunque para el manejo ya se cuenta con el driver para poder controlarlo.

Tabla 6.

Especificaciones técnicas generales del driver del motor Brushless.

Característica	Valor
Voltaje:	36V/48V
Potencia:	350W
Acelerador (Throttle) :	1-4V
Protección corto circuito :	31.5±0.5V/42±0.5V
Color:	Silver
Dimensiones:	105 x 70 x 35mm/4.13 x 2.76 x 1.38"
Material:	Aluminio

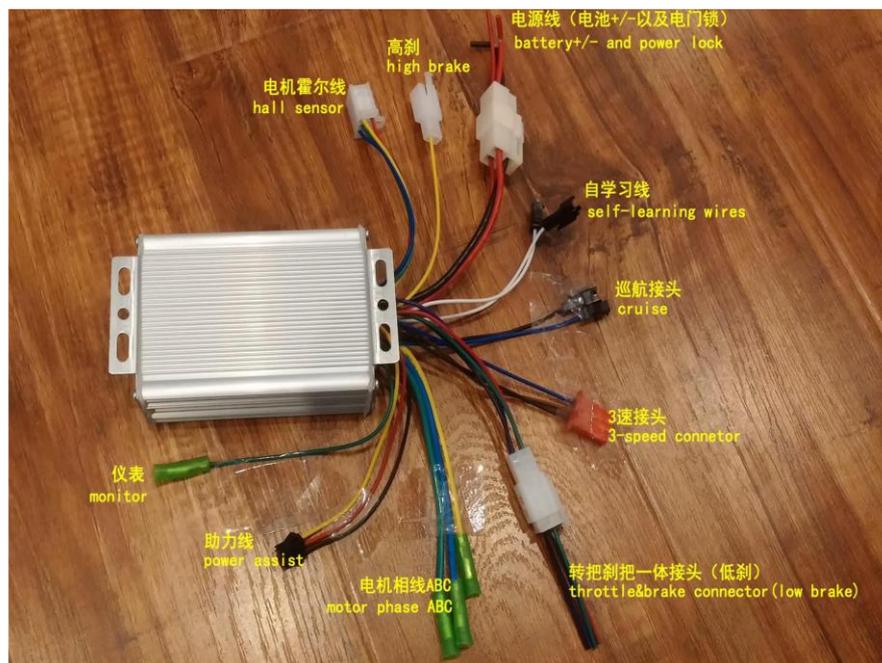


Figura 88 Conexiones del driver del motor CD brushless.

Fuente: (Codd Power, 2016)

De las especificaciones técnicas del driver podemos ver que coincide con la potencia de 350 w, para el control del motor, y que para la aceleración se requiere de un voltaje DC en el rango de uno a cuatro voltios.

En el diagrama de conexiones de la figura 88, podemos ver que el driver cuenta con varias funciones, las cuales en este caso no vamos a utilizarlas, pero vale la pena nombrarlas como son: los cables para autoaprendizaje (self-learning wires) que nos sirven para configurar el sentido de giro de la rueda, freno alto y bajo (high brake, low brake) nos sirven para desenergizar las bobinas del motor, conector de 3 velocidades (3 speed connector) y cable monitor nos sirve para la conexión de un módulo externo para el monitoreo de velocidad y el asistente de poder (power assist) nos ayuda para la conexión de una alarma antirrobo. Las conexiones que si vamos a utilizar son: los sensores de efecto hall nos sirven para medir la velocidad, conexión hacia la batería y los cables de las tres fases.

El ultimo componente electrónico que utilizaremos para la funcione avanzar y retroceder es la batería, aunque esta será la misma para todo el sistema.

Tabla 7.

Especificaciones de la batería.

Característica	Valor
Material de la batería	Litio
Voltaje Nominal (V)	36 V
Capacidad Nominal (Ah)	20 Ah
Max Corriente de Descarga	25 A (BMS SET 500 W)
Tensión máxima De Carga	≤42 V
Descarga de tensión de corte	28 V
Corriente de carga	3 Amperios
peso (NW)	3.5Kg
Ciclo de vida	1200 ~ 1600 veces
Dimensión	370 × 110 × 76.6mm



Figura 89. Batería del sistema.

Fuente: (Codd Power, 2016)

3.4.1.3 Girar

Para la función de giro, como componente electrónico se utilizará el motor paso a paso, dado que se necesita un torque de 27 Nm antes del juego de engranes cónicos y después del juego de engranes de 9Nm con un factor de seguridad de 1.5 se tendría 13.5 Nm, y como medida estándar se buscará un motor de 15 Nm. Como no es necesaria una velocidad alta en el giro se buscará un motor que cuente con una caja de reducción, para así reducir costos y tamaño del motor.

Con las especificaciones requeridas, se seleccionó el motor de la figura 22 y en la tabla 4. Donde se puede apreciar que posee una precisión de 0.12 grados, y cuenta con el torque requerido de 15Nm, a más de eso también se lo puede alimentar con la misma batería del motor DC brushless, dado que necesita de una diferencia de potencial de 24-48v.

Tabla 8.
Especificaciones técnicas del motor pasó a paso.

Detalles	Valor
Manufacturer Part Number	23HS22-2804S-PG15
Motor Type	Bipolar Stepper
Step Angle	0.12°
Holding Torque	15.5Nm
Rated Current/phase	2.8A
Phase Resistance	0.9ohms
Recommended Voltage	24-48V
Inductance	2.5mH±20%(1KHz)

Fuente: (Stepper Online, 2016)



Figura 90. Motor paso a pasos Nema 23 con caja de reducción de 15:1.

Fuente: (Stepper Online, 2016)

Para poder controlar el motor paso a paso por facilidad de manejo se ha visto la necesidad de adquirir el driver apropiado. El modelo escogido es el de la figura 23 con la nomenclatura M542 que cuenta con las características dadas en la tabla 5. Lo más importante a recalcar es que posee protecciones contra cortocircuito.

Tabla 9.

Características eléctricas del driver M542.

Parámetros	M542			Unidad
	Min.	Típico	Max.	
Salida de corriente	1.0	10	4.2(3.0RMS)	A
Voltaje de alimentación	+20	+36	+50	VDC
Corriente de señales lógicas	7	10	16	mA
Frecuencia de entrada de pulsos	0	-	300	kHz
Resistencia de aislamiento	500	-	-	Mohm

Fuente: (Stepper Online, 2016)



Figura 91. Driver M542 del motor pasó a paso.

Fuente: (Stepper Online, 2016)

3.4.1.4 Frenar

La función de freno dado que para poder accionar la mordaza del freno de disco se necesita 1.5 Nm, con un factor de seguridad de 1.5, se dimensiono que se requiere un torque de 2Nm. En este caso se utilizará un servomotor, con lo que el que cumple este requisito es el servomotor HS 805MW. En la tabla 6 se pueden ver las características de este servomotor. Dado que requiere un voltaje de 6 v, se necesitará reducir el voltaje de la batería para alimentarlo. Lo más notable es que cumple con el torque requerido de 2Nm, porque con un voltaje de 6 v alcanza un torque de 24.7Kgcm que equivale a 2.4Nm.

Tabla 10.

Características servomotor HS 805MG.

Detalles	Rangos
Control System:	+Pulse Width Control 1500usec Neutral
Required Pulse:	3-5 Volt Peak to Peak Square Wave
Operating Voltage:	4.8-6.0 Volts
Operating Temperature Range:	-20 to +60 Degree C
Operating Speed (4.8V):	0.19sec/60° at no load
Operating Speed (6.0V):	0.14sec/60° at no load
Stall Torque (4.8V):	274.96 oz/in. (19.8kg.cm)
Stall Torque (6.0V):	343.01 oz/in. (24.7kg.cm)
Operating Angle:	45 Deg. one side pulse traveling 400usec
Continuous Rotation	
Modifiable:	Yes
Direction:	Clockwise/Pulse Traveling 1500 to 1900usec
Current Drain (4.8V):	8mA/idle and 700mA no load operating
Current Drain (6.0V):	8.7mA/idle and 830mA no load operating
Dead Band Width:	8usec
Motor Type:	3 Pole Ferrite
Potentiometer Drive:	Indirect Drive
Bearing Type:	Dual Ball Bearing
Gear Type:	4 Heavy Duty Metal Gears & 1 Nylon Gear
Connector Wire Length:	11.81" (300mm)
Dimensions:	2.59" x 1.18"x 2.26" (66 x 30 x 57.6mm)
Weight:	6.95 oz. (197g)

Fuente: (Servocity, <https://www.servocity.com>, 2016)



Figura 92. Servomotor HS 805MG.

Fuente: (Codd Power, 2016)

Para cualquier especificación extra en la sección de anexos se encuentran todas las características técnicas de todos los componentes utilizados.

3.4.2 Interfaces de comunicación entre componentes

La comunicación entre componentes se realizará mediante una tarjeta que se encargará de enviar y recibir datos hacia los drivers y la Raspberry pi como se muestra en el siguiente diagrama de bloques:

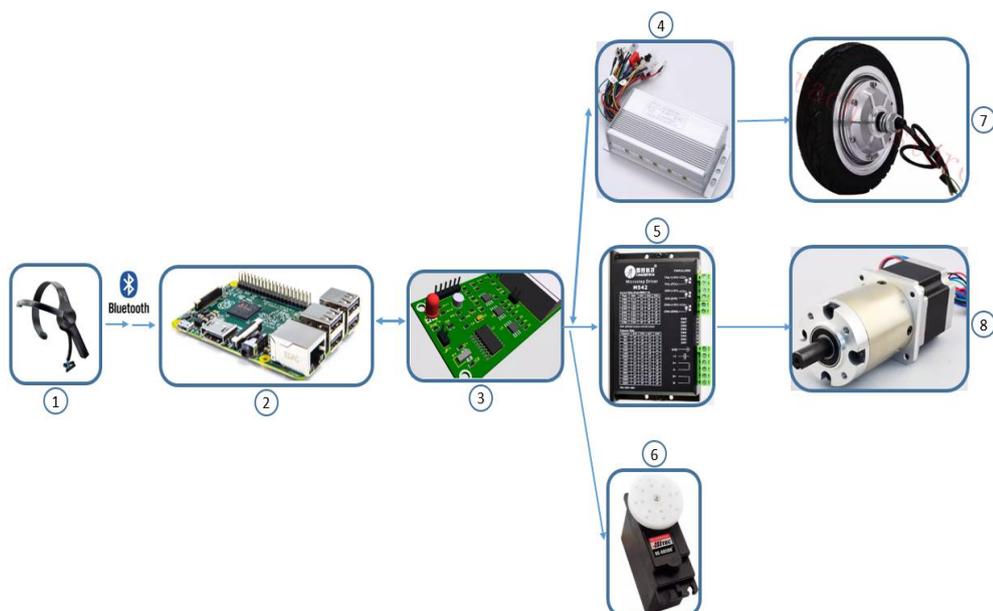


Figura 93. Diagrama de comunicación entre componentes.

Los elementos que conforman la comunicación entre componentes son los siguientes:

1. Mindwave
2. Raspberry pi 2
3. Tarjeta de control
4. Speed control para motor brushless (Rueda)
5. Microstepping Driver
6. Servo Hitec HS-805 Mega Power
7. Motor brushless
8. Motor a pasos

3.4.2.1 Diseño de la placa de control

3.4.2.1.1 Acondicionamiento de señal para el control de velocidad

El motor brushless para poder girar, los devanados del estator deben ser activados mediante una secuencia cuenta. Es importante conocer la posición del rotor con el fin de entender que devanado se activara siguiendo la secuencia de activación, la mayoría de motores brushless tienen 3 sensores de efecto hall incrustados en el estator. Cada vez que los polos magnéticos del estator pasan cerca de los sensores de hall, dan una señal de nivel alto o bajo. Esta señal indica que el polo N o S pasa cerca de los sensores.

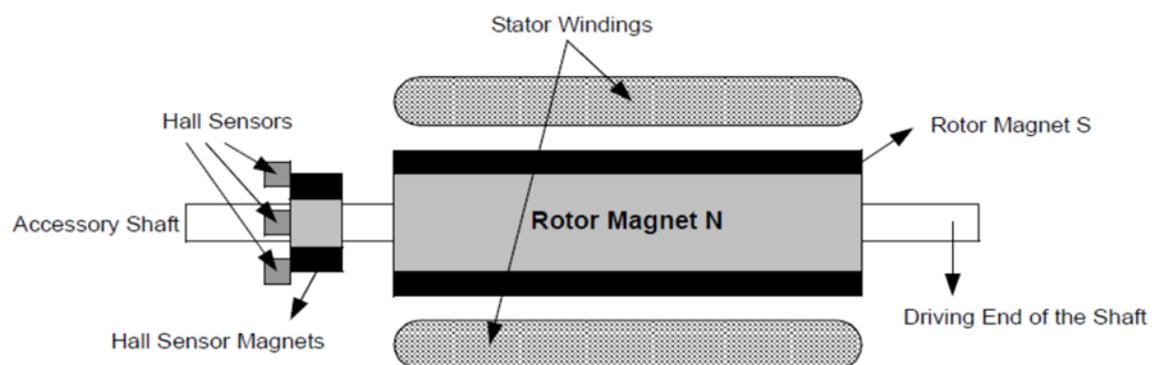


Figura 94. Sección de un motor brushless y posición de los sensores Hall.

Fuente: (Murcia, 2011)

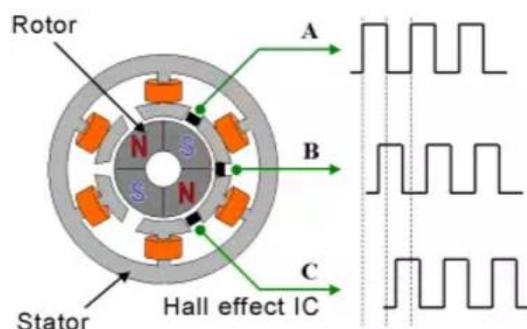


Figura 95. Señal de sensores Hall.

Fuente: (Murcia, 2011)

En el rotor se encuentran imanes permanentes N y S y en el estator los sensores Hall que están colocados con un desfase de 60° o 120° .

La señal tomada y observada de un sensor de efecto hall presenta ruido, este ruido se corregirá a través de un circuito de acondicionamiento que servirá para corregir dicha señal, con esta señal se obtendrá la velocidad, a continuación, se muestra un diagrama de bloques indicando las fases del circuito de acondicionamiento:

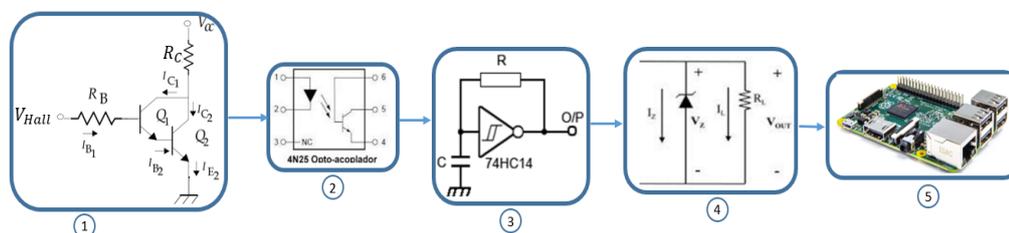


Figura 96. Fases del circuito de acondicionamiento.

El circuito de acondicionamiento consta de 5 fases:

- 1) Configuración Darlington
- 2) Opto-acoplador
- 3) Smith Trigger
- 4) Reducción de voltaje a 3.3V
- 5) Llegada de la señal a la Raspberry pi

Primera fase:

El terminal que nos proporciona esta señal tiene una corriente muy baja por lo que necesitamos realizar una primera fase de ganancia de corriente para ello se hará uso de la configuración Darlington con una ganancia de $\beta_1=\beta_2=\beta=100$ para cada transistores.

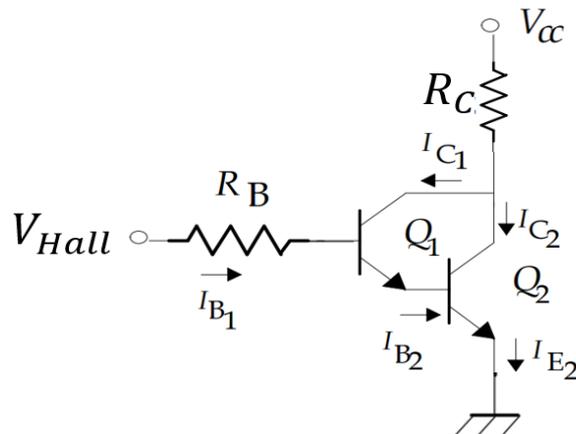


Figura 97. Configuración Darlington.

$$V_{Hall} = I_{B1}R_B + V_{BE1} + V_{BE2}$$

$$5 = I_{B1}(1M\Omega) + 0.7 + 0.7$$

$$I_{B1} = \frac{5 - 1.4}{1M\Omega}$$

$$I_{B1} = 3.6 \mu A$$

$$I_{cc} = \frac{V_{cc}}{R_c}$$

$$I_{cc} = \frac{5}{180}$$

$$I_{cc} = 27.78 \text{ mA}$$

De la 2da le de Kirchoff tenemos:

$$I_{B1} + I_{C1} = I_{B2} = (\beta_1 + 1)I_{B1}, \text{ para el transistor Q1}$$

$$I_{B2} + I_{C2} = (\beta_2 + 1)I_{B2}, \text{ para el transistor Q2}$$

Reemplazando $I_{B2}=(\beta_2 + 1)I_{B1}$ tenemos:

$$(\beta_1 + 1)I_{B1} + I_{C2}=(\beta_2 + 1) (\beta_1 + 1)I_{B1}$$

$$I_{C2}=(\beta_2 + 1) (\beta_1 + 1)I_{B1}-(\beta_1 + 1)I_{B1}$$

$$I_{C2}=(\beta_2 + 1 - 1)(\beta_1 + 1) I_{B1}$$

$$I_{C2}=\beta_2(\beta_1 + 1) I_{B1}$$

$$I_{C2}=100(100 + 1)3.6\mu A$$

$$I_{C2} = 36.36mA$$

Segunda fase:

Como la señal del sensor del efecto Hall ira a la Raspberry pi, esta señal no deberá tener ruido y debe ser aislada por lo que utilizaremos un optotransistor realizando los siguientes cálculos:

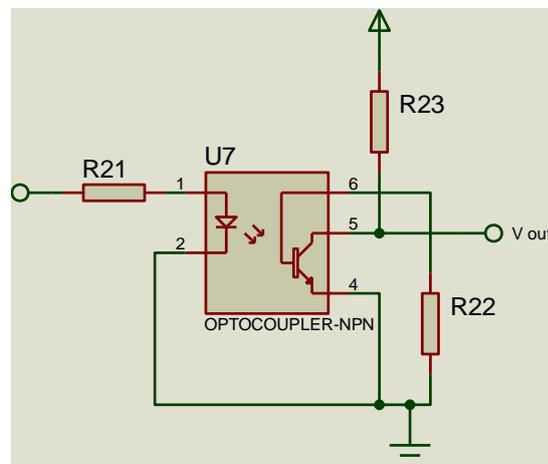


Figura 98. Circuito para el opto-transistor.

$$R_{21} = \frac{V_{in} - V_{diodo}}{I_{C2}}$$

$$R_{21} = \frac{5 - 1.2}{36.36}$$

$$R_{21} = 104.5 \Omega$$

$$R_{23} = \frac{V_{cc} - V_{CE(SAT)}}{I_c}$$

Para una resistencia de $R_{23}=2.2K\Omega$

$$I_c = \frac{5 - 0.3}{2.2K}$$

$$I_c = 2.14mA$$

Tercera fase:

Para ayudar a disminuir el ruido de la señal y poder corregir la misma utilizaremos un schmitt trigger que nos ayudara también a evitar falsos cambios de estado, se midió la frecuencia máxima de la señal generada por el sensor obteniendo un valor de 12Hz. Con el valor de la frecuencia se utilizaron valores recomendados para R y C por la hoja de datos del schmitt trigger 7414.

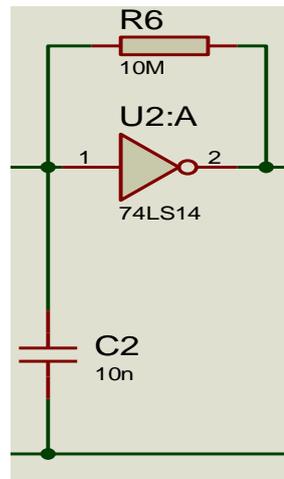


Figura 99. Circuito para schmitt trigger.

Cuarta fase:

La Raspberry pi trabaja en sus entradas y salidas digitales con voltajes de 3.3V, razón por la cual deberemos reducir el voltaje saliente del schmitt trigger, para ello usaremos un diodo zener de 3.3V para regular el voltaje.

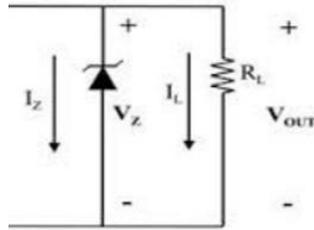


Figura 100. Diodo Zener y carga.

Para una carga de $20k\Omega$

$$I_L = \frac{3.3}{20K}$$

$$I_L = 165\mu A$$

Finalmente, el circuito para corregir la señal nos queda de la siguiente forma:

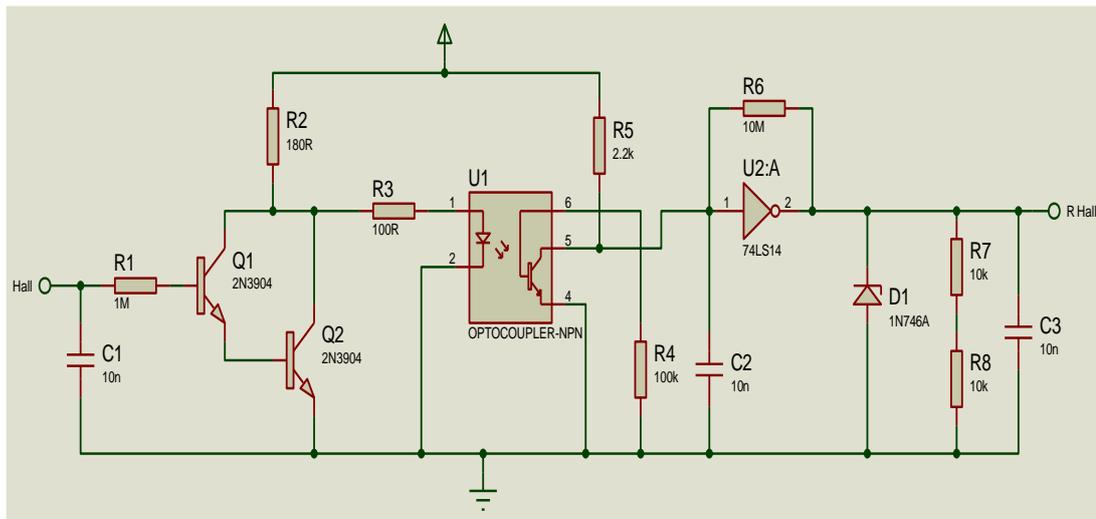


Figura 101. Circuito de acondicionamiento.

3.4.2.1.2 Filtro RC para circuito de control de velocidad

La señal que se debe enviar al driver del motor brushless para controlar la velocidad tiene que ser una señal de voltaje continuo que este en rangos de 1 a 4V debido a la configuración del driver, para generar dicha señal utilizaremos PWM pasado por un filtro RC para convertirlo en voltaje continuo.

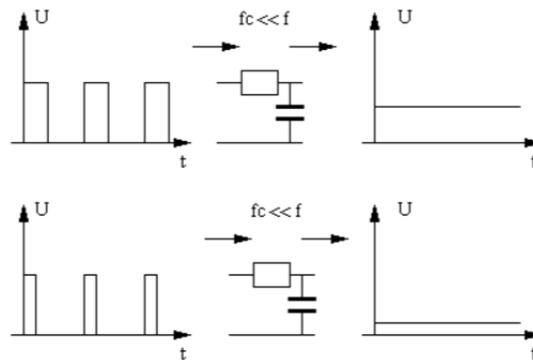


Figura 102. Conversión de PWM a voltaje DC.

En la Raspberry se configura la frecuencia de PWM en $F_{pwm} = 723 \text{ Hz}$. Como realizaremos un filtro pasa bajos tomaremos una frecuencia 1000 veces menos:

$$F_{filtro} = \frac{F_{pwm}}{1000}$$

$$F_{filtro} = \frac{723}{1000}$$

$$F_{filtro} = 0.723 \text{ Hz}$$

Calculo de RC:

$$RC = \frac{1}{2\pi F_{filtro}}$$

$$RC = \frac{1}{2\pi * 0.723}$$

$$RC = \frac{1}{2\pi * 0.723}$$

$$RC = 0.22013$$

Para $C=100\mu\text{F}$ tenemos:

$$R = \frac{0.22013}{100\mu}$$

$$R = 2.2 \text{ K}\Omega$$

El circuito nos quedaría como en la figura 68.

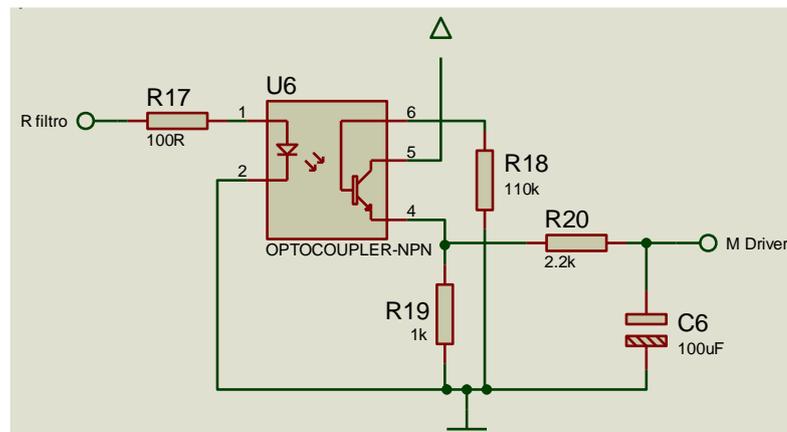


Figura 103. Filtro RC.

De igual manera que el anterior circuito utilizamos un opto-transistor para aislar la señal y evitar que se dañe la Raspberry pi.

Para activar el diodo del opto-transistor se requiere una corriente mínima de 21mA según la hoja de datos, procedemos a realizar el cálculo de la resistencia:

$$R_{17} = \frac{V_{in} - V_{diodo}}{I_{min}}$$

En este caso el $V_{in} = 3.3V$ ya que el opto-transistor recibe la señal de pwm.

$$R_{17} = \frac{3.3 - 1.2}{0.021}$$

$$R_{17} = 100\Omega$$

$$R_{19} = \frac{V_{cc} - V_{CE(SAT)}}{I_c}$$

$$I_c = \frac{3.3 - 0.3}{1K}$$

$$R_{19} = 3mA$$

3.4.2.1.3 Circuito de control para el driver del motor a pasos

Este circuito se diseñó en función de la recomendación indicada en el manual del driver del motor a pasos M542m.

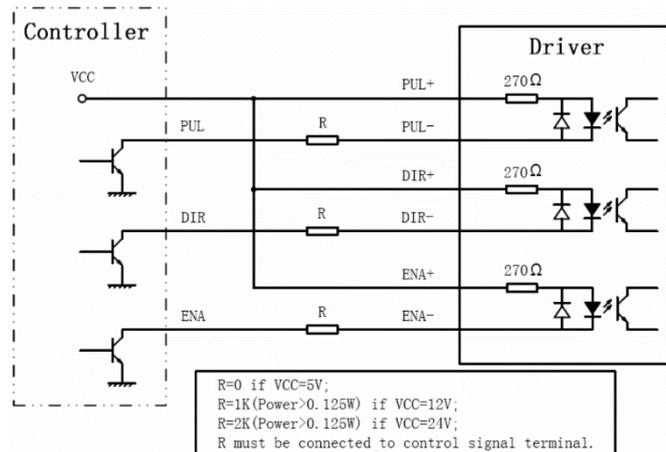


Figura 104. Circuitos para control de PUL, DIR (giro) Y ENA.

Fuente: (M542m, 2008)

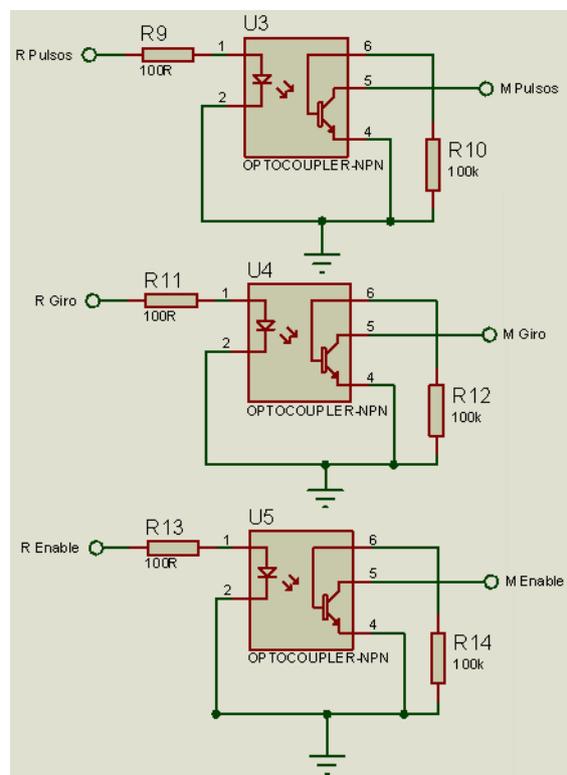


Figura 105. Circuito recomendado para controlar el Driver M542m.

Se tiene 3 parámetros para controlar el motor PUL, DIR (giro) Y ENA, cada uno de ellos se controla con estado de alto y bajo y para cada uno de ellos se realizó un circuito mediante opto-transistores.

De igual manera se realizó los cálculos para las resistencias de los diodos:

$$R = \frac{V_{in} - V_{diodo}}{I_{min}}$$

$$R = \frac{3.3 - 1.2}{0.021}$$

$$R = 100\Omega$$

3.4.2.1.4 Circuito para señal del servomotor

Se realizó este circuito únicamente para transformar la señal de 3.3v proveniente de la Raspberry a 5V para poder controlar el servo motor

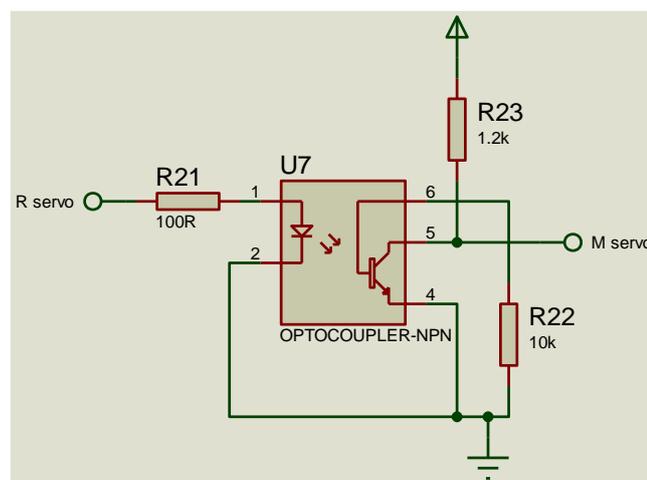


Figura 106. Circuito de 3.3V a 5V para señal del servomotor.

$$R_{21} = \frac{V_{in} - V_{diodo}}{I_{min}}$$

$$R_{21} = \frac{3.3 - 1.2}{0.021}$$

$$R_{21} = 100\Omega$$

$$R_{23} = \frac{V_{CC} - V_{CE(SAT)}}{I_c}$$

$$I_c = \frac{3.3 - 0.3}{1k}$$

$$I_c = 3mA$$

Circuito final y diseño PCB

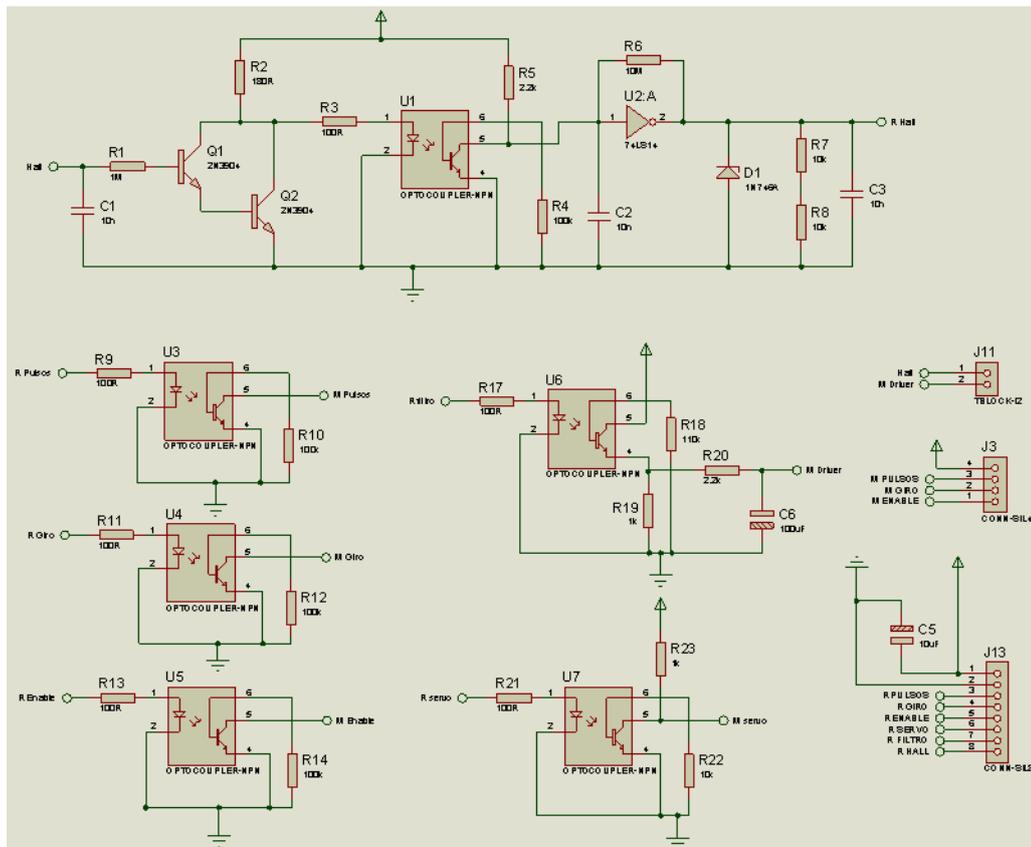


Figura 107. Circuito final.

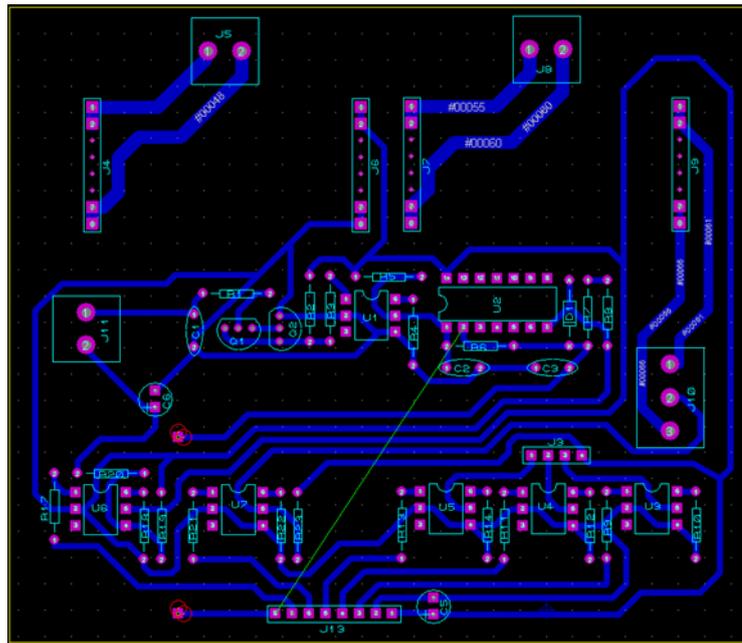


Figura 108. Ruteado PCB.

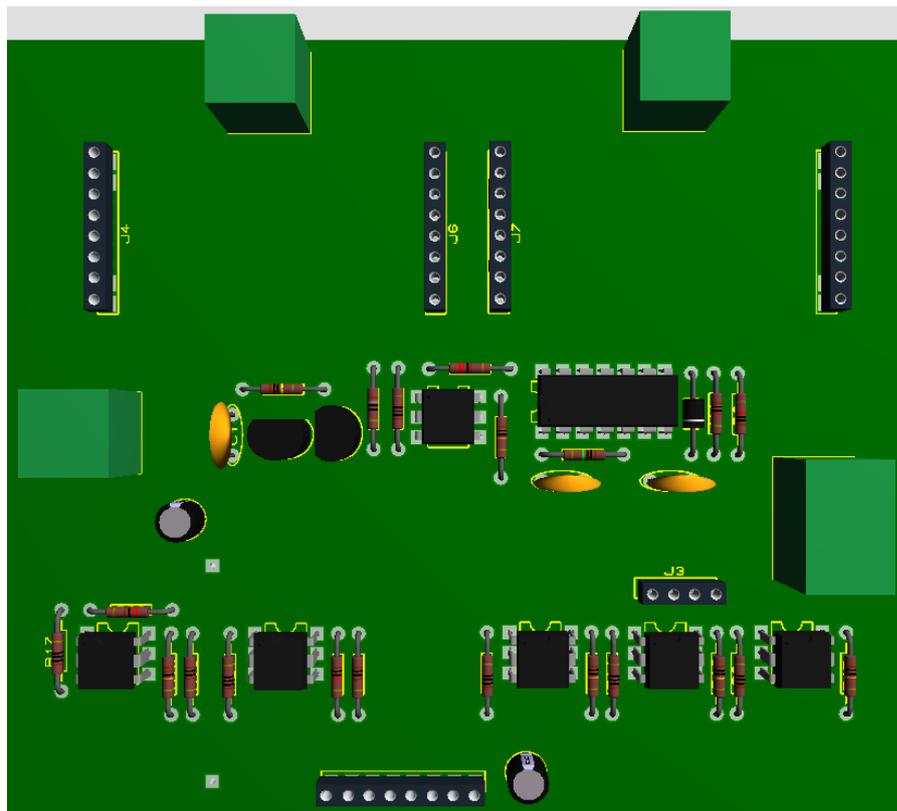


Figura 109. Vista 3D de la tarjeta de control.

3.5 Diseño del sistema de control

3.5.1 Requerimientos

El sistema de control deberá contar con:

- Recepción e interpretación de instrucciones mediante el sensor de señales EEG
- Control del motor de la rueda de tracción
- Control del motor de dirección
- Control motor de frenado

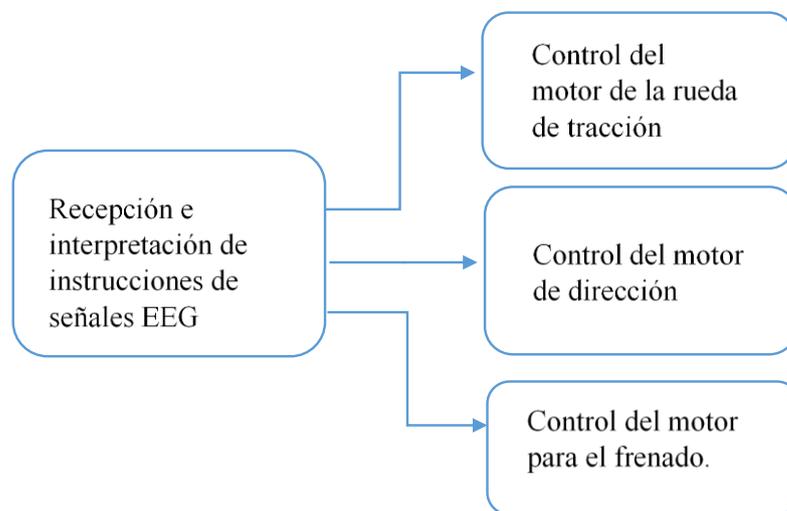


Figura 110 Esquema general del Sistema de control

En la figura 110 se puede apreciar un esquema general del sistema de control en donde tendremos procesos independientes para cada uno de los controles, que será comandado por las instrucciones recibidas por el sensor de señales EEG.

3.5.2 Recepción e interpretación de instrucciones mediante el sensor de señales EEG.

Para la ejecución de la recepción e interpretación de los comandos, dividiremos en las siguientes etapas de este proceso:

- Muestreo del sensor de señales EEG
- Pre procesamiento de la señal EEG
- Detección de pestañeo
- Extracción de características de la señal EEG

- Escalamiento de los valores de las características de la señal EEG
- Red Neuronal
- Interpretación de los comandos.

3.5.2.1 Muestreo del sensor de señales EEG

El sensor Mindwave Neurosky, posee una interfaz de comunicación Bluetooth con los requerimientos de comunicación Serial Port Profile (SPP, Perfil de puerto serial), a una tasa de comunicación de 57600 baudios o bits por segundo.

Los datos que envía el sensor se los puede apreciar en la siguiente tabla:

Tabla 11.

Datos enviados por el sensor Mindwave Neurosky.

Dato enviado	Descripción	Tasa de transferencia	Tamaño
Calidad de señal	Este dato nos informa que tan buen contacto existe entre el sensor y la piel.	1 dato por segundo	1 byte
Nivel de atención	Este parámetro nos entrega la concentración que tiene el usuario	1 dato por segundo	1 byte
Nivel de meditación	Este parámetro nos da el nivel de relajación del usuario.	1 dato por segundo	1 byte
Fuerza de pestañeo	Cuando ocurre un pestañeo nos nada la fuerza con la que pestañea	1 dato cuando ocurre un pestañeo	1 byte
Valor de la onda cruda	Este valor nos da el dato adquirido sin ningún tipo de modificación.	512 datos por segundo	2 bytes
Potencias de la onda EEG dada por el ASIC (Application specific integrated circuit)	Con este dato recibimos las siguientes valores de las potencias EEG: delta, theta, alfa baja, alfa alta, beta baja, beta alta, gama baja y gama media.	1 dato por segundo	24 bytes

Como se puede apreciar en la tabla 11, el sensor nos envía datos muy importantes, pero con el inconveniente en que casi todo los envía solamente una vez por segundo. Es por eso que el procesamiento se realizara con el valor de la onda EEG cruda, dado que tenemos una transferencia de 512 valores por segundo

Protocolo de comunicación.

Teniendo en cuenta que el muestreo se lo realizara en la Raspberry, es necesario implementar el protocolo de comunicación, con el sensor. El envío de datos se los realiza por paquetes, en donde cada paquete consta de las siguientes partes:

- Encabezado
- Carga útil
- Suma de verificación de la carga útil

El encabezado consiste de 3 bytes, 2 de sincronización (SYNC, 0xAA 0xAA) para indicar que empieza el envío de un paquete, seguido por la longitud de la carga útil (PLENGTH).

La carga útil de datos es una serie de bytes, con el número de bytes dado por el encabezado. Para indicar que tipo de dato es el que llega primero se envía un byte de código (CODE), para saber qué tipo de dato llevo. En la siguiente tabla se puede apreciar que valor corresponde a cada tipo de dato.

Tabla 12.
Código y longitud de los tipos de carga útil de datos.

Dato	CODE	PLENGTH
Calidad de señal	0x02	1
Nivel de atención	0x04	1
Nivel de meditación	0x05	1
Fuerza de pestañeo	0x16	1
Valor de la onda cruda	0x80	2
Potencias de la onda EEG	0x83	24

La suma de verificación de la carga útil de datos, viene dada por una formula, en donde se debe seguir los siguientes pasos para obtenerla:

- Sumar todos los bytes de la carga útil del paquete.
- Tomar los 8 bits menos significativos de la suma
- Con estos 8 bits desarrollar la inversa de cada uno.

El resultado obtenido, debe coincidir con el byte de la suma de verificación de carga útil (CHKSUM). Si es que coincide quiere decir que hemos obtenido correctamente el paquete.

3.5.2.2 Pre procesamiento de la señal EEG

En esta etapa se hará la implementación de un filtro suavizador de onda, estos pueden ser filtros FIR o IIR. Esto se hará con el fin de eliminar ruido innecesario en la señal EEG.

3.5.2.3 Detección de pestañeo.

La detección del pestañeo es fundamental dentro del proyecto dado que este será el habilitador para recibir una instrucción y poder clasificarlas en la red neuronal. Dicha detección se la probará de dos maneras, y se verá la respuesta. Las formas son las siguientes:

- Adquisición del pestañeo con el dato obtenido por el electrodo.
- Implementación de un algoritmo para la detección del pestañeo con los datos de la onda EEG cruda.

La prueba se basará en la velocidad de respuesta, dado que el electrodo ya contiene la función en este solo se realizará la prueba. Ahora para el algoritmo, se realizarán las siguientes etapas:

- Muestreo de eventos de pestañeos del usuario
- Aplicación de filtros suavizadores de onda.
- Extracción de una onda promedio.
- Implementación del algoritmo que compare la onda de entra con la onda promedio de pestañeo obtenida.
- Verificación de pestañeo.

Se tomarán las muestras necesarias donde se puedan distinguir los eventos de pestañeo dentro de la onda, al aplicar el filtro se podrán visualizar de mejor manera dichos eventos. Para la extracción de la onda promedio en primera instancia se deberá aplicar simplemente una simple suma de las ondas en los puntos que correspondan y dividir para el número de muestras obtenidas, de ser necesario se podrán aplicar otros métodos para obtener una onda que nos permita verificar si la persona ha pestañado.

Para la verificación de si la persona ha pestañado se comparará la onda de entrada con la onda promedio obtenida, esta comparación se probará obteniendo el error medio cuadrático entre estas ondas. También en la implementación se deberá verificar si es que se las puede hacer con la misma velocidad con la que se muestrea el sensor. Es una parte fundamental para evitar pérdida de datos que ocasionaría fallos en el sistema.

3.5.2.4 Extracción de características de la señal EEG

Con el fin de dar información a la red neuronal, poder extraer los patrones y poder clasificar correctamente las instrucciones dadas, se tienen que extraer características que diferencien los diferentes comandos, para cada una de las características se las deberá implementar tanto en la PC como en la single board computer (Raspberry) algoritmos que respondan exactamente igual para que los pesos que se obtengan en la PC funcionen correctamente en la Raspberry. Esto se profundizará de mejor manera en la explicación de la red neuronal.

Las características de la onda EEG que se probarán serán:

- Potencias de la transformada Discreta de Fourier.
- Valores Máximos
- Valores Mínimos
- Promedio
- Valores iniciales
- Valores Finales
- Picos altos con su localización
- Picos bajos con su localización
- Suavidad de la onda.

3.5.2.5 Escalamiento de los valores de las características de la señal EEG

Uno de los principales requerimientos para que una red neuronal funcione de óptima manera es el escalamiento de los valores de entrada. Dado que se van a tener diferentes características, cada una de estas va a estar en su propia escala por ejemplo el número de picos no va a estar más allá de un máximo de 100 picos, pero el valor del valor máximo va a estar en el rango de 50000, si es que no se realiza un escalamiento el valor del número de picos va a ser insignificante.

Entonces el escalamiento deberá hacerse teniendo los valores mínimos y máximos de cada una de las características, y se los llevará a un mínimo de -1 y un máximo de 1. Con esto logramos que todas las características tengan la misma influencia dentro de la red neuronal.

3.5.2.6 Red Neuronal

La red neuronal constara de dos partes, la primera es el entrenamiento y la segunda es la verificación e implementación. EL entrenamiento se lo realizara con software en una PC, y la verificación e implementación en software dentro de la single board computer Raspberry.

Primero que todo, se debe definir la estructura de la red neuronal en nuestro caso, nos hemos impuesto una red neuronal con una capa oculta, para agregar mayor exactitud, teniendo presente que luego tenemos que tener cuidado con sobre ajustar la red neuronal. Los comandos requeridos serán 4 por lo que a la salida tendremos 4 salidas, que nos enviarán los comando Adelante, Atrás, Izquierda y derecha. Al momento de la implementación se buscarán gestos y movimientos que se puedan distinguir claramente para obtener el menor error dentro del funcionamiento del módulo mecatrónico adaptable.

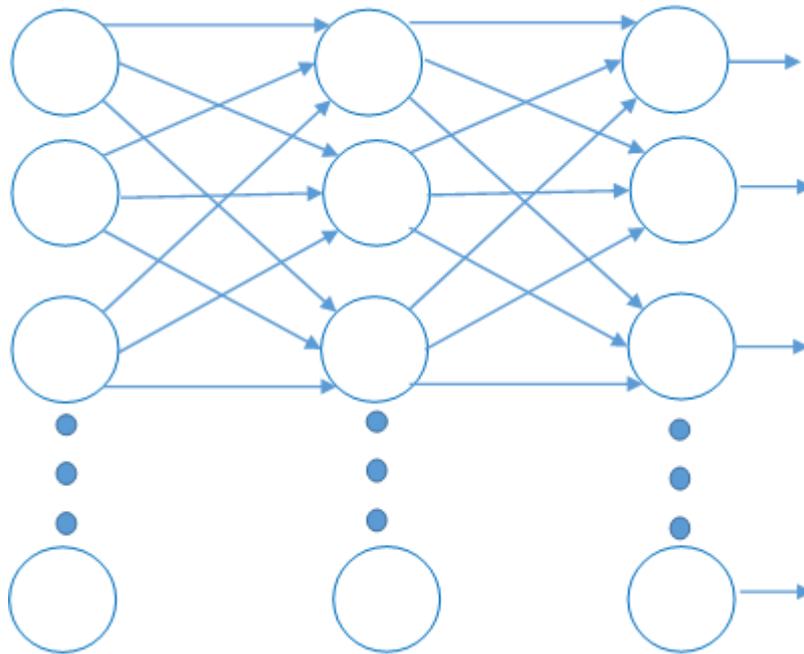


Figura 111. Estructura de la red neuronal

En la figura 111 se puede apreciar la estructura que nos impusieron para la red neuronal. Como se puede observar tenemos una capa oculta de 100 nodos y cuatro salidas. Lo que se definirá en la implementación son las características de entrada.

Es importante mencionar que la estructura y la estructura tiene que ser exactamente igual con las mismas características de entrada, tanto en el entrenamiento como en la implementación, dado que con cualquier cambio los pesos que se obtengan en el entrenamiento variarían el comportamiento en la implementación.

- Entrenamiento de la red neuronal.

Por cuestión de las facilidades en cuanto a software que obtenemos en una PC, el entrenamiento se lo realizará en una PC. Lo que se deberá hacer es muestrear la onda generada por los gestos que se prueben. Para esto se deberá ya tener muy bien escogido e implementado el método de detección de pestañeo, dado que se pedirá al usuario que pestañee y seguidamente emita o ejerza el gesto que se esté probando. De esta manera se podrá localizar con mayor facilidad la onda que genera cada gesto y con esto extraer características que los diferencien. Algunos de los gestos que se probarán serán:

- Pestañeos
- Guiños

- Movimientos de cabeza
- Movimientos faciales

Una vez obtenidos los cuatro comandos que se diferencian de la mejor manera, se procederá a probar que funcionen correctamente dentro de la Raspberry.

- Verificación del funcionamiento

Se verificará el funcionamiento de la red neuronal, primero con entradas específicas que se den a la red neuronal en la PC, se procederá a probar que se obtenga exactamente las mismas salidas en la Raspberry. Una vez probado el funcionamiento en la Raspberry, se deberá probar que estos gestos se los pueda realizar con naturalidad al manejar la Raspberry, de no ser así se repetirá este proceso.

3.5.2.7 Interpretación de los comandos.

El siguiente diagrama de flujo muestra la interpretación propuesta de los comandos para el manejo de la silla de ruedas. En la figura se puede visualizar como va a ser el funcionamiento de la interpretación de los comandos. Primero no se procesará todos los datos dentro de la red neuronal, solo cuando ocurra un doble pestañeo dentro en menos de un segundo, se muestra la señal para ingresar dentro de la red neuronal. Una vez muestreada la señal se obtienen las características y se las mapea. Dependiendo de cada instrucción que obtengamos de la red neuronal, el sistema se comportará de diferente manera. Se puede observar que cuando esta girada la dirección, se deberá primero girar la dirección hacia adelante para poder frenar o moverse hacia adelante. De igual manera si la dirección esta girada y se quiere volver a girar hacia la izquierda o derecha, lo que se hace es girar hacia adelante y frenar, para que con otra instrucción poder girar hacia el lado que se desee. Es decir, solo si esta la llanta girada hacia adelante se podrá girar hacia la izquierda o derecha.

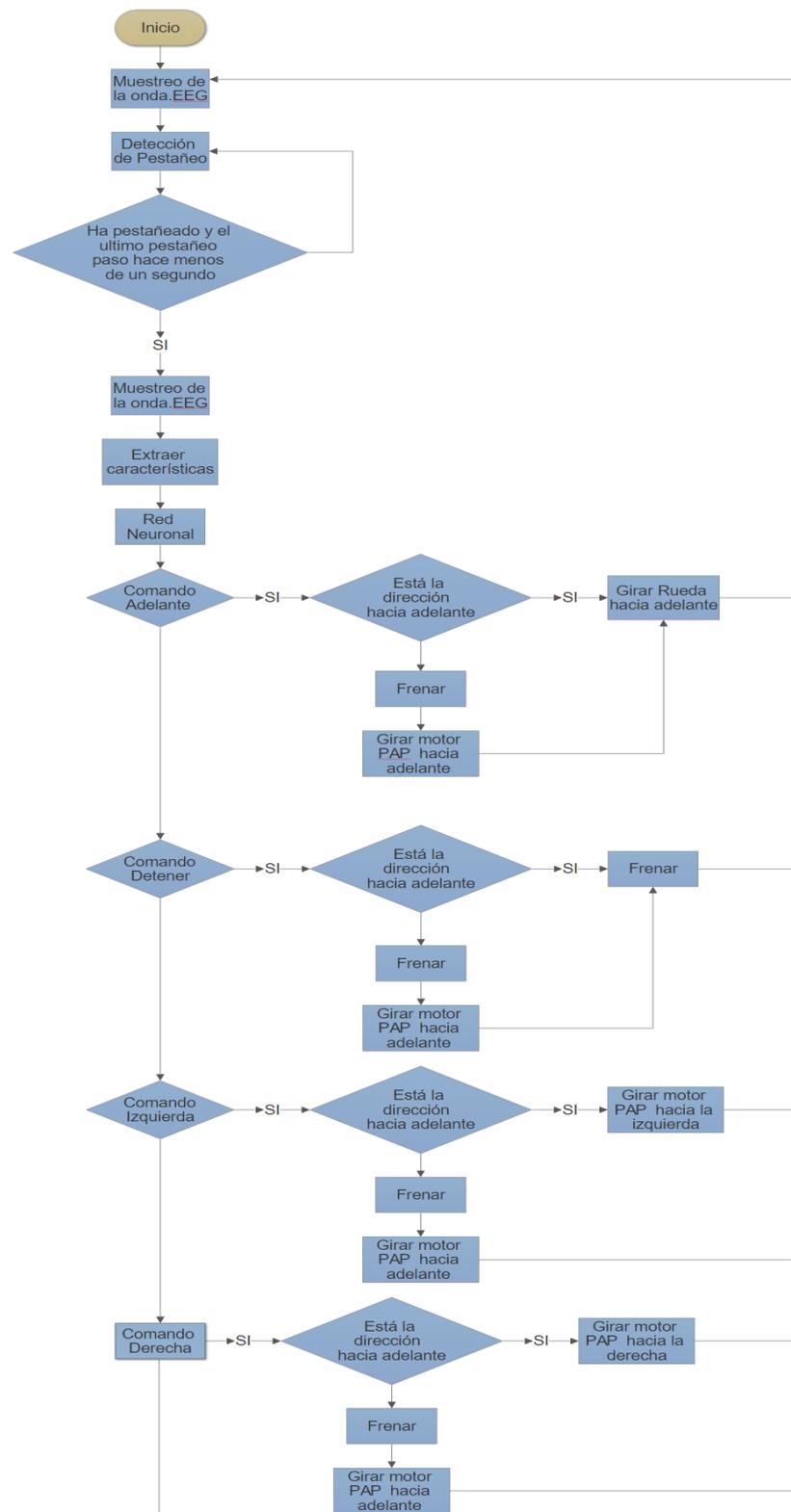


Figura 112. Diagrama de flujo de la interpretación de comandos

3.5.3 Control del motor de la rueda de tracción

Con propósitos de seguridad se realizará el control de velocidad de la rueda de tracción. En este caso contamos con el driver del motor, el cual, para activar el motor, requiere de un voltaje continuo que se debe encontrar en el rango de 0.8 [v] y 4 [v], como se observó en el diseño de la placa de control.

El control de velocidad se lo realizara con un control PID (Proporcional Integral Derivativo), para cuyo diseño se necesita la ecuación de la planta del motor. Para ello utilizaremos la herramienta de identificación de sistemas de Ident de Matlab. Como primer paso debemos tomar los datos de velocidad, ante una entrada escalón. El escalón de entrada será de 2.8 [v], que es equivalente a 800 rpm. La respuesta del motor ante esta entrada se lo puede visualizar en la figura 6, donde podemos observar a $u1$ como la entrada escalón, y a $y1$ como la salida en rpm.

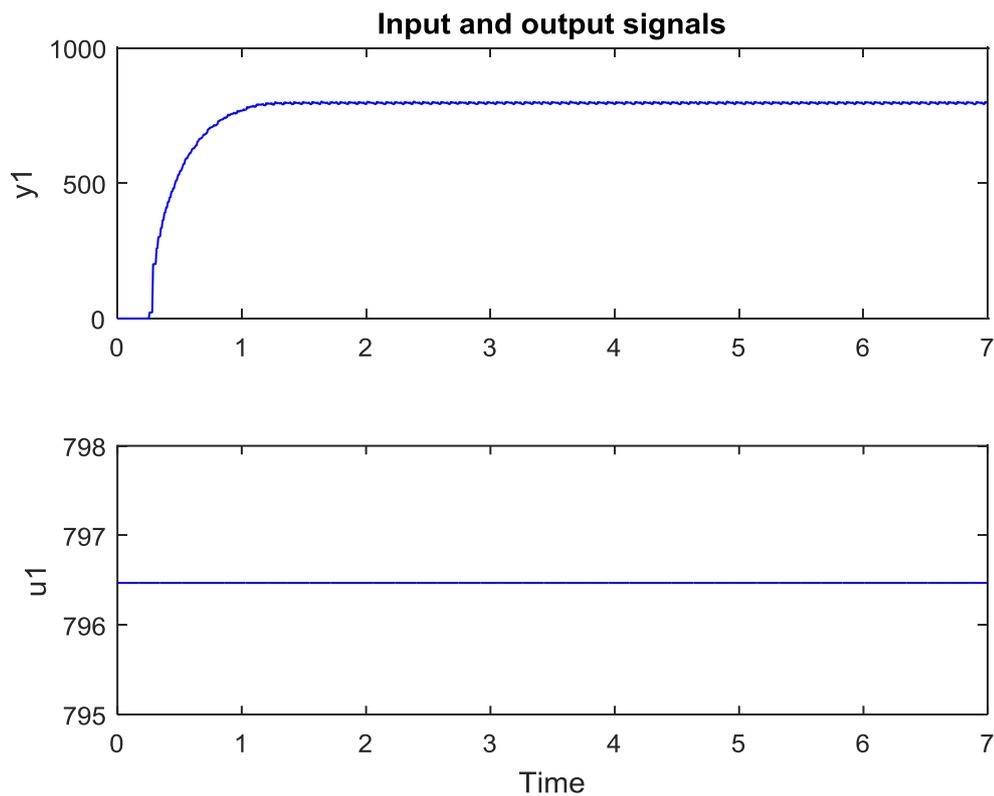


Figura 113. Respuesta del motor brushless ante una entrada escalón.

Antes de introducir los datos obtenidos al software identificador de sistemas Ident de Matlab, debemos saber qué grado tiene la planta. La función de transferencia de un motor brushless viene dado por:

$$G(s) = \frac{\frac{1}{K_e}}{\tau_m \tau_e s^2 + \tau_m s + 1}$$

Teniendo en cuenta que

$$\tau_m = \frac{J3R}{K_e K_t}$$

$$\tau_e = \frac{L}{3R}$$

Donde J es el momento de inercia del motor, R la resistencia de los devanados, K_e la constante de Back EMF y K_t la constante de torque, τ_e la constante eléctrica, τ_m la constante mecánica. (Delgado & Bolaños, 2013).

Por lo que, con el identificador de sistemas, deberá devolvernos una función de transferencia ningún cero, y con dos polos. Analizando en la gráfica dada en la figura 6, existe un retardo en la respuesta del motor, esto también será incluido en el software. Teniendo como resultado el sistema obtenido en la figura 7. El sistema obtenido tiene 95.29% de ajuste. Que es menor al 5 % de error permitido, como se observa el seguimiento que se tiene es casi exacto. Los parámetros obtenidos se los puede visualizar en la figura 114.

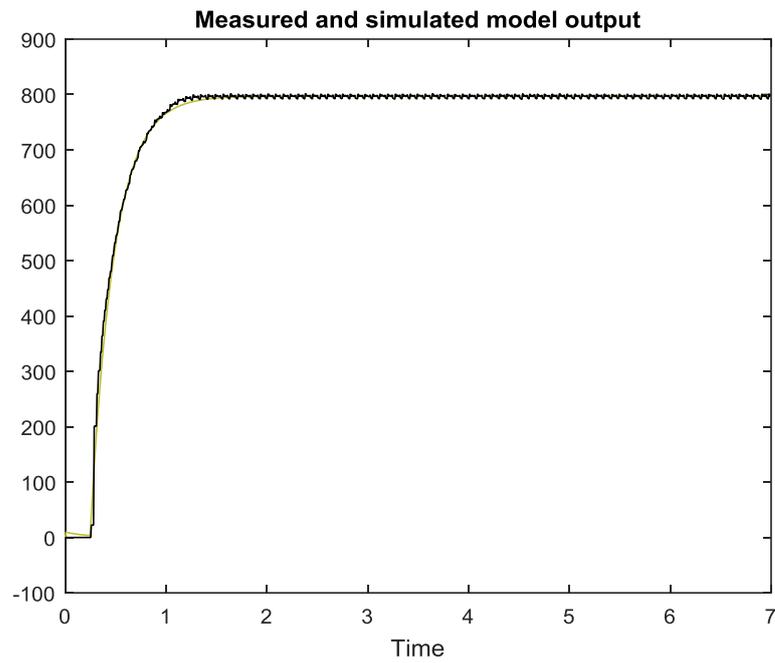


Figura 114. Salida del modelo simulado, junto con el medido.

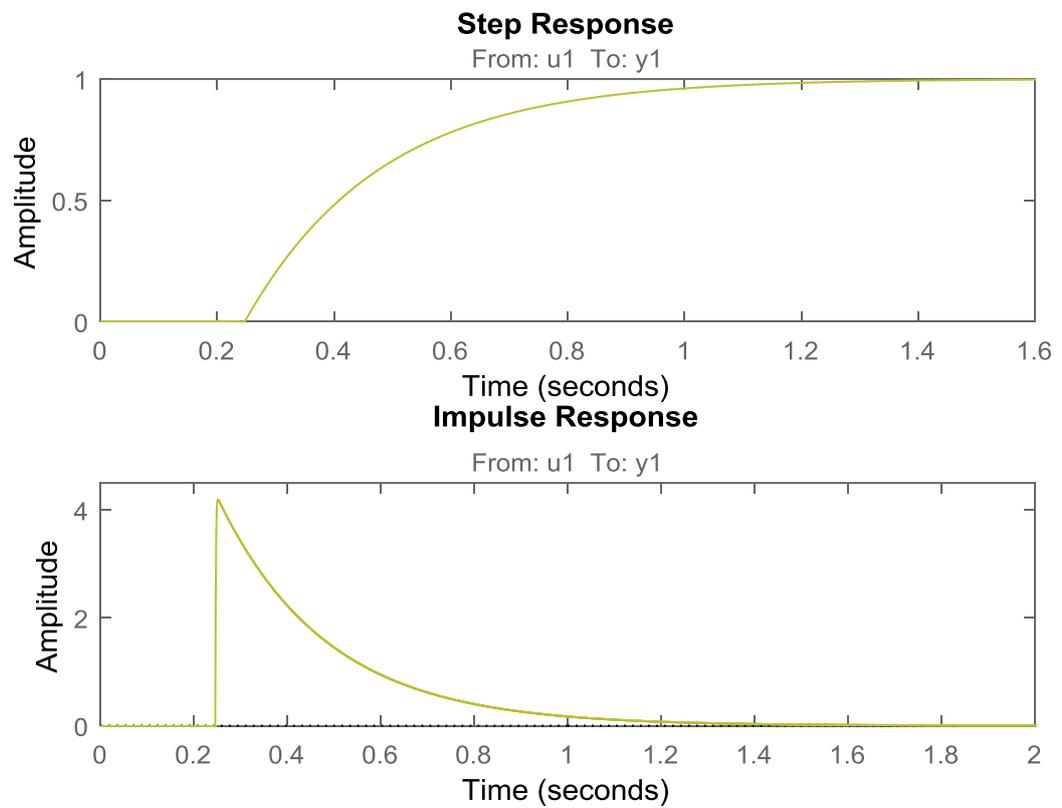


Figura 115. Respuesta ante una entra impulso y escalón de un motor DC Brushless.

Process model with transfer function:

$$G(s) = \frac{K_p}{1+2*\text{Zeta}*T_w*s+(T_w*s)^2} * \exp(-T_d*s)$$

$K_p = 1.0009$
 $T_w = 0.01582$
 $\text{Zeta} = 7.4041$
 $T_d = 0.24596$

Figura 116 Parámetros del modelamiento del motor DC brushless

Con los parámetros obtenidos podemos ingresar en el sintonizador de PID de Matlab, obteniendo la respuesta mostrada en la figura 10, y las constantes del PID visualizadas en la figura 117.

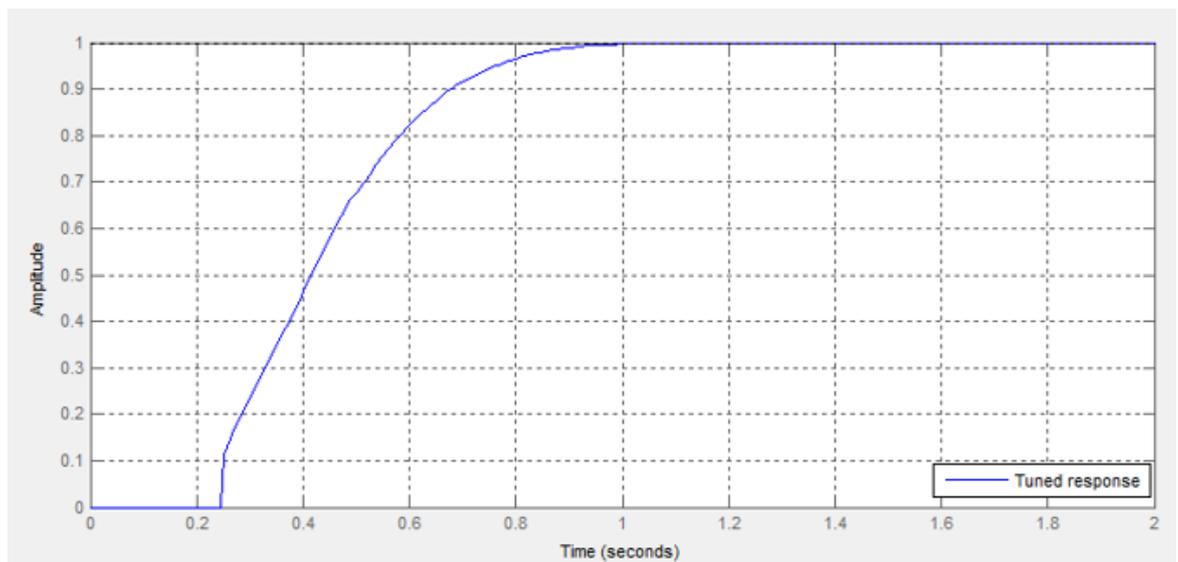


Figura 117. Respuesta del sistema con un control PID.

Controller parameters	
	Tuned
Kp	0.65362
Ki	2.2364
Kd	0.024958
Tf	

Figura 118. Parámetros obtenidos para el control PID.

El tiempo de muestreo se debe calcular con la frecuencia natural de la planta. Que está dada por la ecuación 3.5.1

$$\omega_n = \frac{1}{T_\omega} \quad (3.5.1)$$

Donde $T_\omega = 0.01582$ de la figura 9, teniendo

$$\omega_n = 63.21 \text{ rad/s}$$

Ahora el periodo de muestreo mínimo deberá ser de:

$$T = \frac{\pi}{\omega_n} \quad (3.5.2)$$

Obteniendo:

$$T = 50ms$$

3.5.4 Control del motor de dirección

Como se vio en el diseño electrónico el control de la dirección se realizará con el motor paso a paso Nema 23 con el driver M542. El programa deberá activar tres salidas para el control las salidas serán:

- Pulsos de control (PULSE)
- Sentido de giro (DIRECCIÓN)
- Activador/ desactivador (ENABLE)

La entrada de pulsos al driver del motor de pasos, por la velocidad del motor a pasos tendrá una distancia mínima entre pulsos, si es que es menor no se alcanzara recibir por lo que se pueden perder pulsos, esto se lo verificará al momento de la implementación dado que se deberá calibrar una velocidad adecuada en el driver. Al momento de activar el driver, se debe esperar por lo menos 5 micro segundos antes de enviar pulsos. Y el ancho de pulso mínimo que debe tener es de 1.5 micro segundos. Estos valores se pueden visualizar de mejor manera en la figura 12. La entrada de activación simplemente debe estar en alto para que función y la entrada de dirección en alto gira horariamente y en bajo en sentido anti horario.

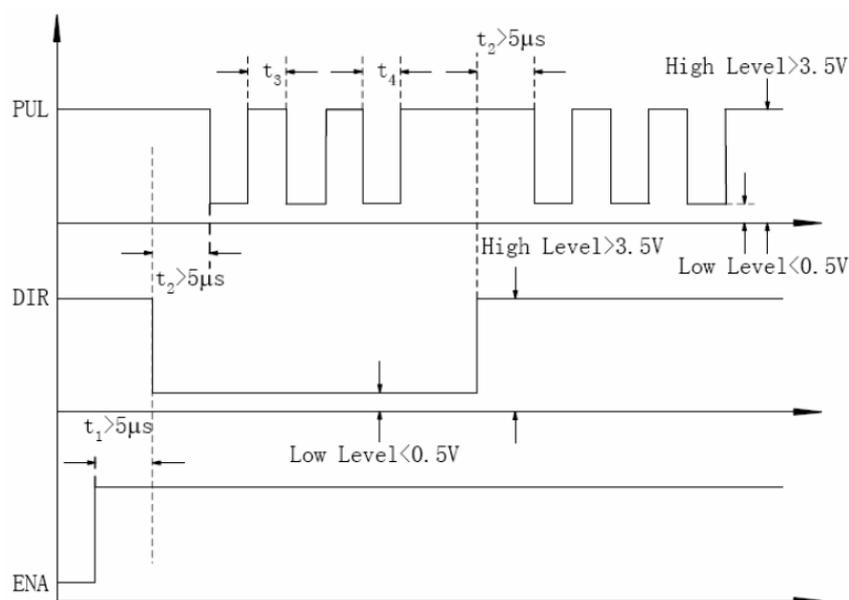


Figura 119. Secuencia de pulsos del control de señales.

Este control se lo realizará en lazo abierto, dada la fuerza que tiene el motor, con el peso máximo para el que está diseñado no se verán afectados los pasos y se podrá saber en qué ubicación se encuentra con un simple contador. Ahora bien, en el caso de que se sobre esfuerce el motor se podrá fácilmente reiniciar al sistema y colocar a la dirección en la posición inicial.

3.5.5 Control motor de frenado

Para el frenado se utilizará un servomotor HS-805MG, como se vio en el diseño electrónico. Para su correcto control se utilizará una frecuencia de 50 Hz. Y el PWM deberá mantener las características de la onda dada en la figura 13.

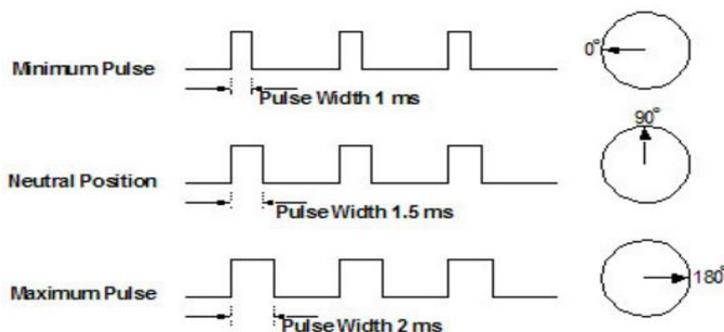


Figura 120. Señal de entrada para el control del motor paso a paso.

Fuente: (Servocity, 2016)

CAPITULO 4

CONSTRUCCIÓN E IMPLEMENTACIÓN

El siguiente capítulo muestra la implementación del módulo mecatrónico y se ha dividido al capítulo en la implementación del sistema mecánico, electrónico y de control.

4.1 Construcción e implementación mecánica

La implementación del módulo mecatrónico se basa en una serie de secuencias que siguen pasos posteriores al diseño, los cuales permitieron que la construcción de la estructura física. Estos pasos se irán detallando a continuación.

4.1.1 Maquinado del perfil de soporte

El perfil de soporte fue mecanizado, haciendo uso de una fresadora y un taladro de pedestal se realizaron las ranuras en los cuatro lados. Sobre el perfil irán montados todos los elementos que componen el modulo.



Figura 121. Perfil mecanizado.

4.1.2 Construcción del Trinche

El trinche está constituido por tres elementos; eje, arco y soportes. Para la construcción del eje se utilizó un torno con el cual se obtuvo los diámetros y el roscado acorde al diseño. El arco fue realizado en una dobladora hidráulica que nos permitió mantener la sección del tubo uniforme. La construcción de los soportes se hizo por medio de corte por electro hiló debido a que la geometría de los soportes es compleja y su construcción sería complicado y costoso si se lo realizaría por métodos tradicionales. La unión de los elementos se hizo con suelda MIG debido a que el arco está firmado por un tubo que se unirá al eje y los soportes.



Figura 122. Tubo y arco antes de soldar.



Figura 123. Unión de tubo y arco por soldadura.



Figura 124. Soldadura de sujetadores.

4.1.3 Construcción de engranes cónicos

La construcción de los engranes cónicos de 15 y 45 dientes se realizó en la fresadora, tomando en cuenta los parámetros necesarios como el módulo de la fresa y

los ángulos de inclinación se procedió al mecanizado, los engranes fueron construidos en acero 1018.

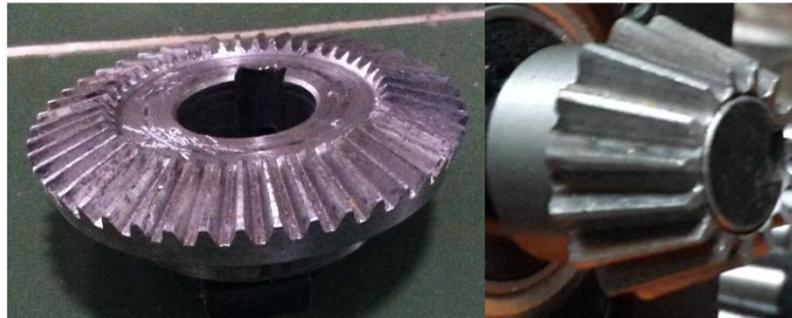


Figura 125. Engranes cónicos de 15 y 45 dientes.

Luego de haber construido se realizó un montaje de prueba para verificar que los engranes encajen correctamente y no exista interferencia entre ellos.



Figura 126. Montaje de engranes.

4.1.4 Construcción de caja para elementos electrónicos

La caja se construyó mediante planchas de tol las mismas que fueron cortadas y dobladas, esta caja consta de dos partes; la base y la tapa, sobre la base que será la parte fija irán sujetos todos los elementos electrónicos, mientras que la tapa será deslizable sobre la base.



Figura 127. Caja para elementos electrónicos.

4.1.5 Construcción de canasta para la batería

Para sujetar la batería se pondrá dentro una canasta, esta canasta ser construida con varillas planas que serán dobladas y soldadas. La caja de los elementos electrónicos ira sujeto a esta canasta.



Figura 128. Canasta para la batería.

4.1.6 Implementación de los componentes del módulo mecatrónico

Luego de haber terminado la construcción de todos los elementos y de haberlos pintado precedemos al montaje del módulo como se mostrará detalladamente a continuación.

Montaje de los rodamientos sobre el trinche:



Figura 129. Rodamientos y trinche.

Montaje del trinche con el perfil y los rodamientos:



Figura 130. Trinche con el perfil.

Guía para cable de frenado:

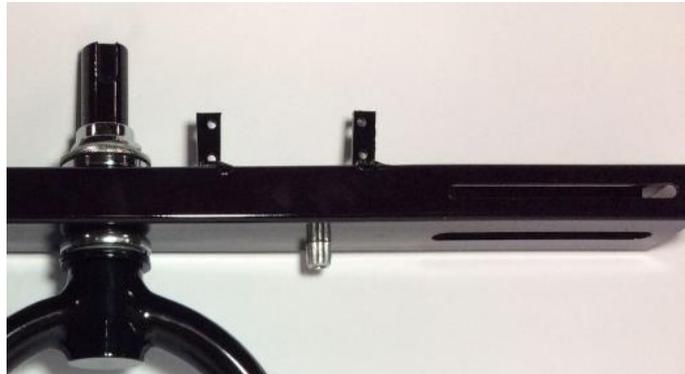


Figura 131. Guía de cable.

Montaje del piñón de 15 dientes sobre el motor a pasos a través de prisioneros con y sujeción del soporte con el motor con pernos:



Figura 132. Motor a pasos y piñón.

El montaje del engrane de 45 dientes se realizó la con la ayuda de los prisioneros y la chaveta:



Figura 133. Montaje de engrane de 45 dientes.

Se fue deslizando poco a poco un engrane con respecto al otro hasta tener la posición correcta de acople para luego fijar cada uno de los engranes:



Figura 134. Acople de piñón y engrane.

Ensamble del servomotor para frenado:



Figura 135. Servomotor para frenado.

Montaje del disco sobre la rueda:



Figura 136. Disco para frenado.

Implementación de la mordaza con el cable de accionamiento:

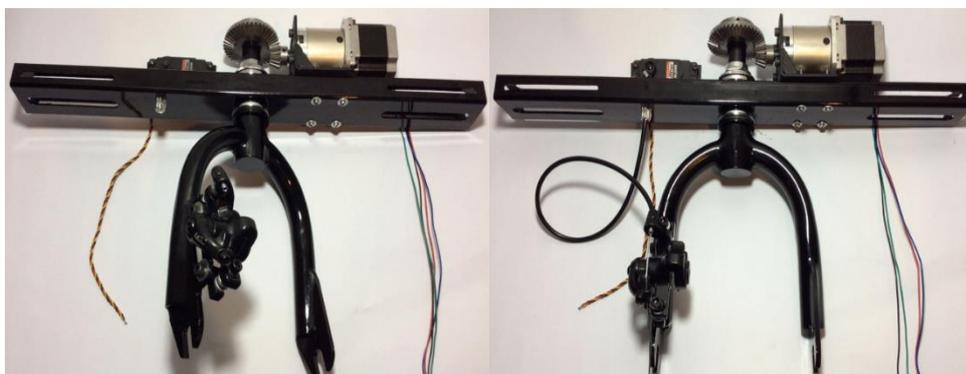


Figura 137. Mordaza y cable.

Acoples para sujeción de ejes roscados:

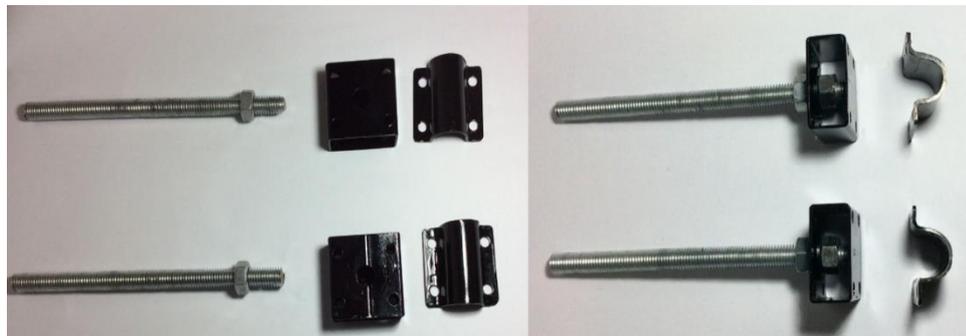


Figura 138: Sujeción de ejes roscados.

Montaje de la batería y los elementos electrónicos sobre el perfil y la canasta:



Figura 139. Montaje de elementos electrónicos.

Para realizar el montaje final del módulo mecatrónico sobre la silla de ruedas se tomó en cuenta la elevación de la silla con respecto a la rueda y que la misma quede centrada como se muestra a continuación:



Figura 140. Montaje del módulo mecatrónico.

4.2 Implementación de elementos electrónicos

En esta fase se construyó la tarjeta de control que comandara los drivers de cada motor, pero también enviara la señal del sensor Hall a la Raspberry y esta a su vez enviara las señales de control para cada uno de los actuadores.

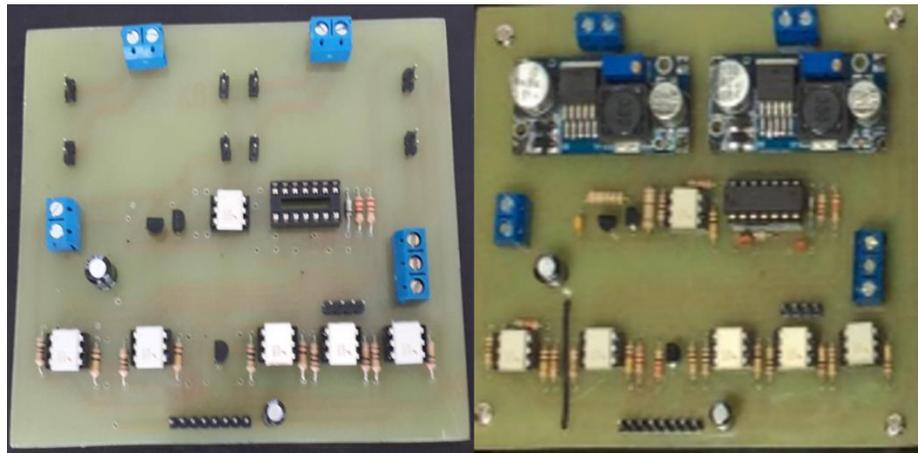


Figura 141. construcción de placa de control.

Se ubicó específicamente los elementos electrónicos, separando físicamente la parte de potencia con la de control, primero se puso el driver del motor brushless, luego el driver del motor a pasos y finalmente se ubicó la tarjeta de control seguidamente de la Raspberry pi como se muestra a continuación:



Figura 142. ubicación de los elementos electrónicos.

Una vez ubicado los elementos electrónicos se procedió al cableado que está distribuido de forma ordenada en caso de que se requiere hacer algún cambio.

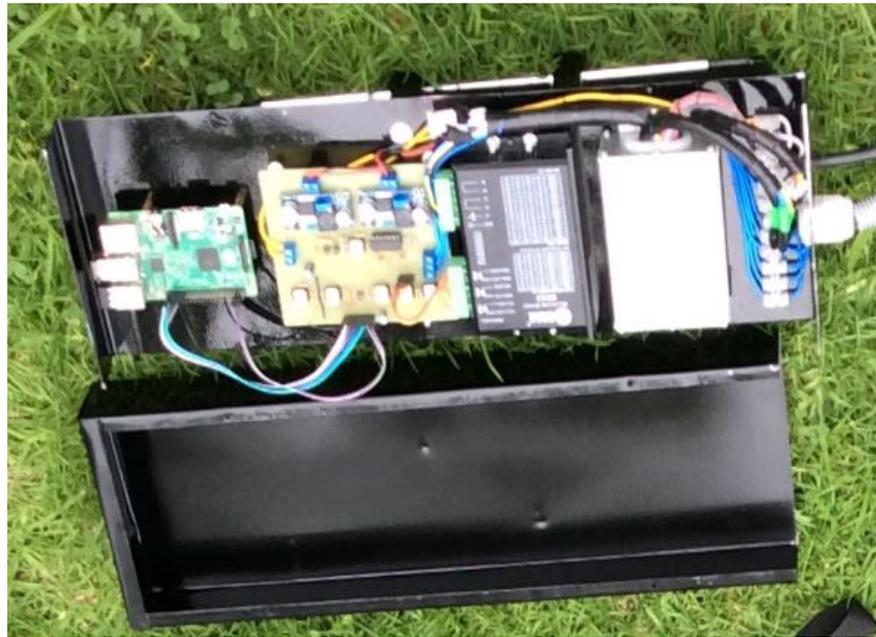


Figura 143. Distribución del cableado.

Los cables de los tres motores; Brushless, a pasos y servo se conectan a los respectivos drivers con cables de aproximadamente 1 m de largo y los cuales van por dentro de una manguera corrugada.

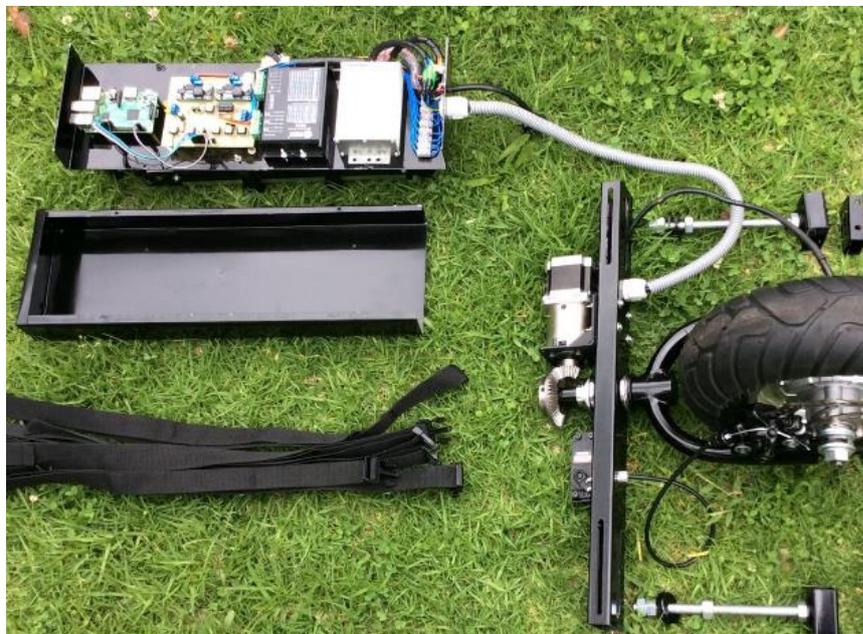


Figura 144. Cableado de motores.

4.3 Implementación y pruebas del sistema de control

Tal como se desarrolló el diseño del sistema de control, se hará la implementación y pruebas del mismo. El sistema de control consta de:

- Recepción e interpretación de instrucciones mediante el sensor de señales EEG
- Control del motor de la rueda de tracción
- Control del motor de dirección
- Control motor de frenado

4.3.1 Recepción e interpretación de instrucciones mediante el sensor de señales EEG.

Para la implementación y pruebas de la recepción e interpretación de los comandos, de igual manera dividiremos en las siguientes etapas de este proceso:

- Configuración de la Raspberry Pi
- Muestreo del sensor de señales EEG
- Pre procesamiento de la señal EEG
- Detección de pestañeo
- Extracción de características de la señal EEG
- Escalamiento de los valores de las características de la señal EEG
- Red Neuronal
- Interpretación de los comandos.

4.3.1.1 Configuración de la Raspberry Pi

Previo a la implementación del sistema de control, se requiere la configuración del ambiente de trabajo con la single board computer.

Como primer paso se deberá descargar el sistema operativo, en este caso se usó “Raspbian Jessie”, con la versión de octubre 2015. Una vez descargado, se procedió con el formateo de la tarjeta Micro SD, esto se lo debe hacer con el software SDFormatter o cualquier otro software que permita dar formato a tarjetas SD a bajo nivel.

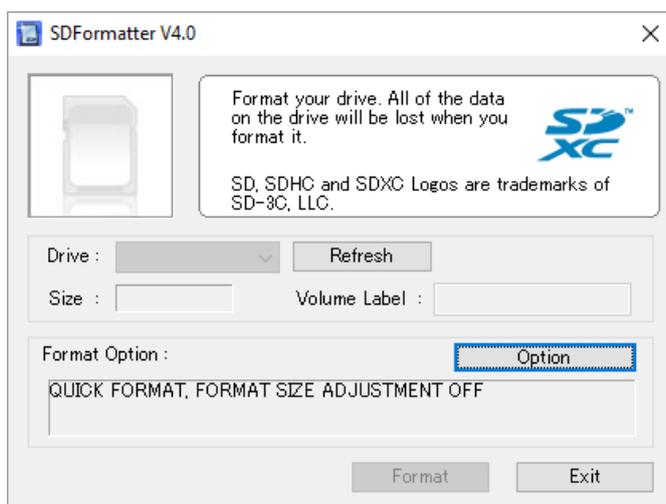


Figura 145. Software SD Formatter.

Con la tarjeta Micro SD dada formato, se procese a cargar la imagen “.iso” descargada del Sistema operativo, para ello se usó el software “Win 32 Disk Imager”.

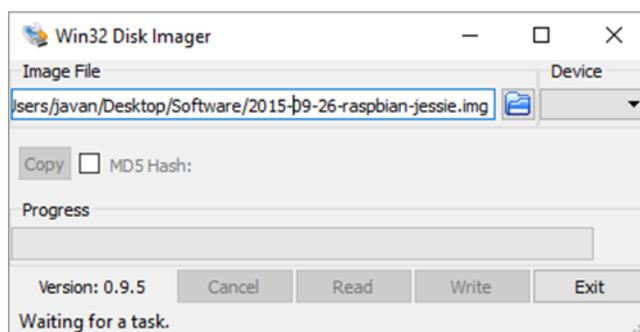


Figura 146. Software Win32 Disk Imager.

Ya con la tarjeta lista, se procedió a encender la Raspberry Pi, con lo que la primera vez se debe habilitar la SSH (Secure Shell), como se ve en la figura 3. Esto nos sirve para el control remoto de la Raspberry, solo se debe tener su IP y se puede acceder remotamente. Para el manejo remoto de la Raspberry se usó el software llamado Putty, y para la transferencia de archivos se utilizó el software llamado FileZilla.

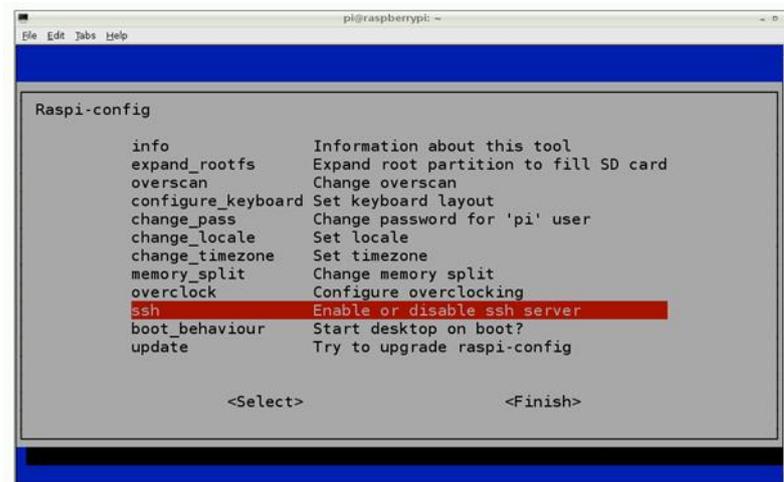


Figura 147. Activación de la Secure Shell de la Raspberry Pi.

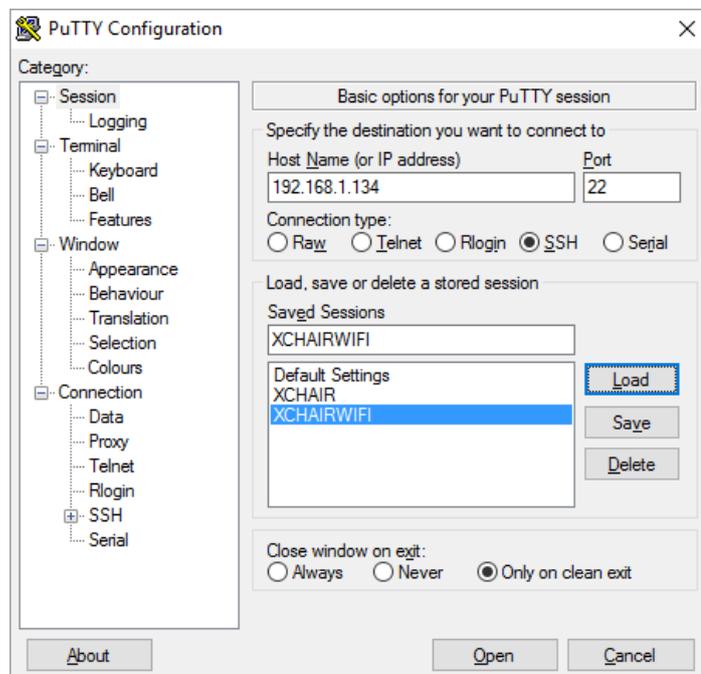


Figura 148. Software para control remoto Putty.

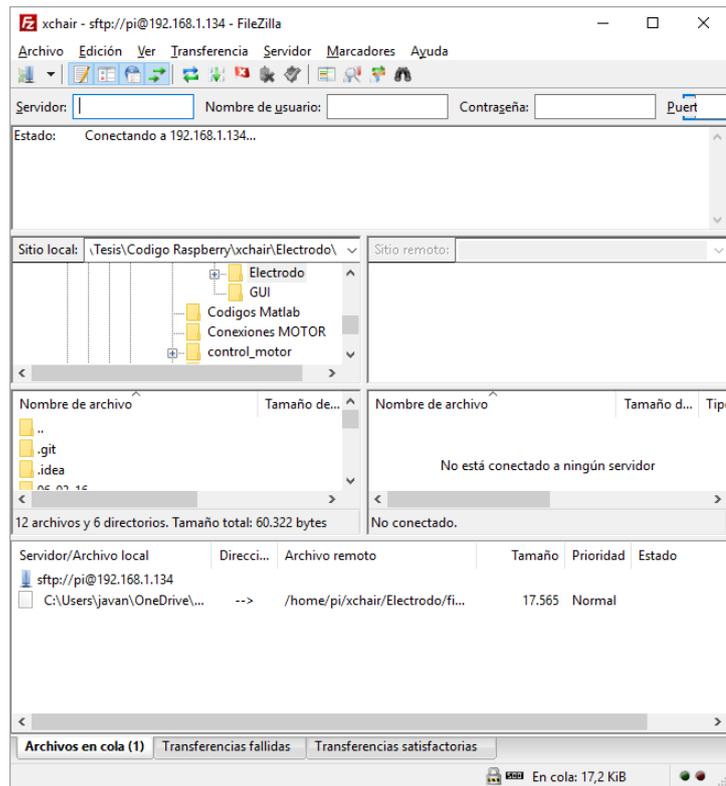


Figura 149 .Software para la transferencia de archivos FileZilla.

Ese es todo el proceso para la configuración de la Raspberry.

4.3.1.2 Muestreo del sensor de señales EEG

Para la recepción de datos del sensor, tendremos que hacerlo tanto en Matlab dentro de la PC, como en Python en la Raspberry.

4.3.1.2.1 Muestro de datos del sensor a través de la PC

En cuanto refiere a la conexión del electrodo con Matlab, utilizaremos una librería liberada por Neurosky, en la figura 6 podemos observar el código. El código lo que hace es cheque la conexión del electrodo hacia algún puerto de la PC, es por ello que siempre primero se tiene que realizar el emparejamiento de los mismos. También en el código se configura la conexión en cuanto a velocidad y los tamaños de paquetes, y se carga la librería de extensión “dll”, si es que no existe ningún error compatibilidad se conecta el electrodo.

```

function readRAW
%run this function to connect and plot raw EEG data
%make sure to change portnum1 to the appropriate COM port
close all
data = zeros(1,256);    %preallocate buffer
portnum1 = 1;    %COM Port #
comPortName1 = sprintf('\\\\.\\COM%d', portnum1);
% Baud rate for use with TG_Connect() and TG_SetBaudrate().
TG_BAUD_57600 = 57600;
% Data format for use with TG_Connect() and TG_SetDataFormat().
TG_STREAM_PACKETS = 0;
% Data type that can be requested from TG_GetValue().
TG_DATA_RAW = 4;
%load thinkgear dll
loadlibrary('Thinkgear.dll','thinkgear.h');
fprintf('Thinkgear.dll loaded\n');
%get dll version
dllVersion = calllib('Thinkgear', 'TG_GetDriverVersion');
fprintf('ThinkGear DLL version: %d\n', dllVersion );
%%
% Get a connection ID handle to ThinkGear
connectionId1 = calllib('Thinkgear', 'TG_GetNewConnectionId');
if ( connectionId1 < 0 )
    error( sprintf( 'ERROR: TG_GetNewConnectionId() returned %d.\n',
connectionId1 ) );
end;
% Set/open stream (raw bytes) log file for connection
errCode = calllib('Thinkgear', 'TG_SetStreamLog', connectionId1,
'streamLog.txt' );
if( errCode < 0 )
    error( sprintf( 'ERROR: TG_SetStreamLog() returned %d.\n', errCode )
);
end;
% Set/open data (ThinkGear values) log file for connection
errCode = calllib('Thinkgear', 'TG_SetDataLog', connectionId1,
'dataLog.txt' );
if( errCode < 0 )
    error( sprintf( 'ERROR: TG_SetDataLog() returned %d.\n', errCode )
);
end;
% Attempt to connect the connection ID handle to serial port "COM3"
errCode = calllib('Thinkgear', 'TG_Connect',
connectionId1,comPortName1,TG_BAUD_57600,TG_STREAM_PACKETS );
if ( errCode < 0 )
    error( sprintf( 'ERROR: TG_Connect() returned %d.\n', errCode ) );
end
fprintf( 'Connected. Reading Packets...\n' );

```

Figura 150. Código de conexión del electrodo Mindwave con Matlab.

Una vez iniciada la conexión se puede empezar a adquirir los datos, en la figura 7 podemos apreciar cómo se adquieren los paquetes de la onda cruda. Primero se verifica si ha llegado un paquete, luego se verifica si es el tipo de paquete que deseamos y de ahí si se adquiere los valores de la carga útil. Como podemos ver aquí ya viene implementado el protocolo de comunicación t solo se llama a las funciones de la librería de Neurosky para adquirir los datos.

```

%%
%record data
j = 0;
i = 0;
while (i < 10240) %loop for 20 seconds
    if (calllib('Thinkgear','TG_ReadPackets',connectionId1,1) == 1)
%if a packet was read...
        if
            (calllib('Thinkgear','TG_GetValueStatus',connectionId1,TG_DATA_RAW) ~= 0)
%if RAW has been updated
                j = j + 1;
                i = i + 1;
                data(j) =
calllib('Thinkgear','TG_GetValue',connectionId1,TG_DATA_RAW);
            end
        end
        if (j == 256)
            plotRAW(data); %plot the data, update every .5
seconds (256 points)
            j = 0;
        end
    end
end
%disconnect
calllib('Thinkgear','TG_FreeConnection', connectionId1 );

```

Figura 151. Adquisición de datos con Matlab del sensor Mindwave.

4.3.1.2.2 Muestro de datos del sensor a través de la Raspberry Pi

Ahora en cuanto refiere a la adquisición de datos con la Raspberry, no existe una librería para este propósito, por lo que hubo que implementarlo. Antes de poder comunicarnos con el sensor, primero se tiene que emparejar, este proceso no es tan simple como en una PC, en Linux tenemos que instalar una herramienta llamada Bluez, por lo que se tiene que ejecutar el comando mostrado en la figura 8, como se puede ver a más de la instalación de la librería Bluez también se hizo la instalación de la librería PyGObject y Bluez para Python, que se utilizaron para la comunicación del puerto serial conectado a la interfaz Bluetooth en Python.

```
sudo apt-get install bluez python-gobject python-bluez
```

Figura 152 Comando para la instalación de librerías de conexión Bluetooth

Una vez instaladas las librerías, se pudo hacer el emparejamiento de los dispositivos, mediante el comando bluetoothctl, el proceso de configuración se puede ver en la figura 9. Lo importante en este punto es encontrar la dirección MAC del electrodo, porque de igual manera esta nos servirá para comunicarnos mediante Python.

```

# bluetoothctl
[NEW] Controller 00:10:20:30:40:50 pi [default]
[bluetooth]# scan on
Discovery started
[CHG] Controller 00:10:20:30:40:50 Discovering: yes
[NEW] Device 00:12:34:56:78:90 myLino
[CHG] Device 00:12:34:56:78:90 LegacyPairing: yes
[bluetooth]# pair 00:12:34:56:78:90
Attempting to pair with 00:12:34:56:78:90
[CHG] Device 00:12:34:56:78:90 Connected: yes
[CHG] Device 00:12:34:56:78:90 Connected: no
[CHG] Device 00:12:34:56:78:90 Connected: yes
Request PIN code
[agent] Enter PIN code: 1234
[CHG] Device 00:12:34:56:78:90 Paired: yes
Pairing successful
[CHG] Device 00:12:34:56:78:90 Connected: no

```

Figura 153. Comandos para el emparejamiento del electrodo con la Raspberry Pi.

Fuente: (ArchLinux, 2016)

Una vez emparejados los dispositivos, se procedió a la implementación del protocolo de comunicación en Python en donde se creó una clase para un fácil uso de las funciones. En la figura 10 se pueden ver las funciones disponibles al crear un objeto de esta clase. Describiendo cada una tenemos:

- `_conectar`: nos sirve para iniciar la comunicación con el electrodo
- `_leerDatosElectrodo`: simplemente retorna una lectura de la cantidad de bytes especificados.
- `_leerUnPaquete`: retorna todos los bytes de un paquete desde el encabezado hasta el pie, aquí es donde se realiza la verificación del protocolo.
- `_readJustRawData`: retorna el valor del siguiente paquete que contenga un valor de la onda cruda.
- `_parseoParcialPaquete`: convierte un paquete de bytes a paquete de enteros.
- `clearPackage`: borra todos los datos en el buffer.
- `parseoPaquetes`: retorna el valor de carga útil
- `parseRawData`: retorna el valor de la onda cruda de un paquete leído con la función `_readJustRawDara`.
- `parseRawData`: convierte un paquete de onda cruda de bytes a enteros.
- `restart`: reinicia la Raspberry.

```

class electrodo():
    def __init__(self):

    def _conectar(self):

    def _leerDatosElectrodo(self, amountOfBytes):

    def _leerUnPaquete(self):

    def _readJustRawData(self):

    def _parseoParcialPaquete(self, paquete):

    def clearPackage(self):

    def parseoPaquetes(self, datosParciales):

    def parseRawData(self, datosParciales):

    def parseRawDataOrd(self, datosParciales):

    def restart(self):

```

Figura 154. Funciones librería electrodo para muestreo de señales del electrodo Mindwave.

4.3.1.3 Prueba del pre procesamiento de la señal EEG

Debido a la variabilidad que se tiene en la señal, se realizó un filtro de respuesta finita al impulso (FIR), específicamente un recorrido del promedio en la figura 11 se puede apreciar el código implementado para filtrar una señal adquirida almacenada en un vector. No se han utilizado funciones de Matlab, debido a que luego se tienen que implementar todas las funciones en la Raspberry. Las características del filtro es que hace la suma de los 100 últimos predecesores y se obtiene el promedio, de esta forma se logra suavizar la onda. En Matlab todo se almacenara en archivo .mat, y por matrices para almacenar varias muestras, es por eso que esta función filtra todos los vectores del nombre de la variable específica del archivo específico.

```

function averageMoving(matFileIn)
%% Average Moving (FIR Filter)
N=100; %Number of predecessors
load(matFileIn);
[mSize,nSize] = size(dataPatterns);
auxVariable=zeros(mSize,nSize);
figure
hold on
for i=1:mSize
    for j=1:nSize
        for k=1:N
            if(j-k)>0
                auxVariable(i,j) = auxVariable(i,j)+dataPatterns(i,j-k);
            end
        end
        auxVariable(i,j)=auxVariable(i,j)/N;
    end
    plot(auxVariable(i,:));
    xlabel(i);
    drawnow
end
save(strcat(matFileIn,'FIR'),'auxVariable');

```

Figura 155. Función recorrido del promedio, para filtrar una señal en Matlab.

Ahora con el fin de aumentar la eficiencia dentro de la Raspberry, los códigos de procesamiento se realizarán en lenguaje C, para luego ejecutarlos a través de Python mediante envolturas o wrappers (Librerías para la utilización de código diferente al que se está ejecutando). En la figura 12 se puede apreciar el código de la misma función implementada en C.

```

double *averageM(double *arrayIn,int N){
    int i;
    int j;
    static double arrayOut[128];
    for(i=0;i<128;i++){
        arrayOut[i] = 0;
    }
    for(i=0;i<128;i++){
        for(j=0;j<N;j++){
            {
                if((i-j)>=0){
                    arrayOut[i] = arrayOut[i] + arrayIn[i-j];
                }
                else{
                    break;
                }
            }
        }
        for(i=0;i<128;i++){
            arrayOut[i] = arrayOut[i]/N;
        }
        return arrayOut;
    }
}

```

Figura 156. Filtro recorrido de promedio implementada en C.

Los resultados obtenidos se los puede apreciar en la figura 13 donde no se ha aplicado un filtro y en la figura 14 donde ya se aplica el filtro.

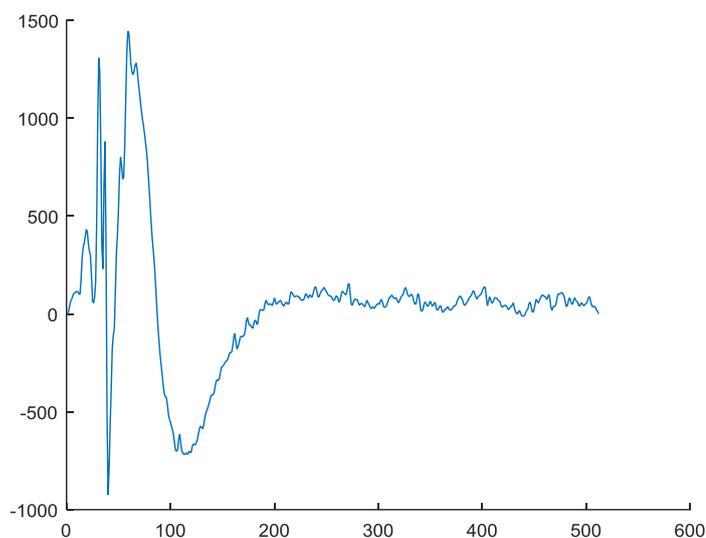


Figura 157. Onda cruda obtenida por el sensor antes de ser filtrada.

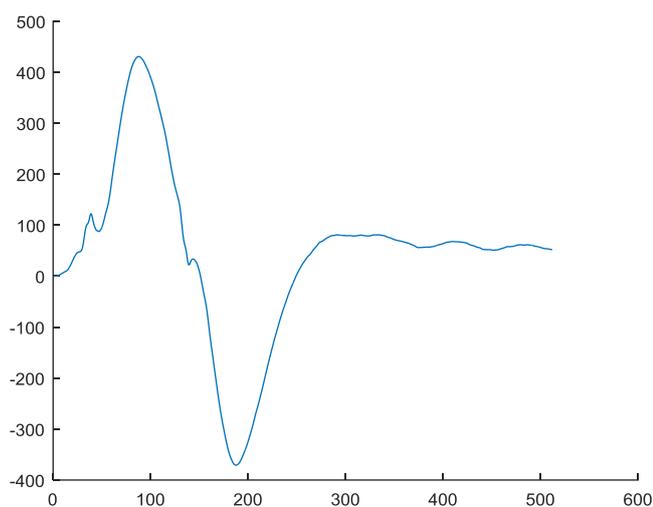


Figura 158. Onda obtenida luego de aplicar el filtro de respuesta finita al impulso.

4.3.1.4 Prueba de la detección de pestañeo.

La detección de pestañeo se probó por dos métodos, mediante el paquete que envía el sensor cuando ocurre un pestañeo y una implementación propia.

- Adquisición del pestañeo con el dato obtenido por el electrodo.

Al momento de hacer la adquisición del pestañeo en el sensor, existe un retardo en la respuesta que va hasta un segundo, esto es un gran inconveniente dado que queremos una respuesta rápida. Es por eso que este método queda descargado de aquí en adelante.

- Implementación de un algoritmo para la detección del pestañeo con los datos de la onda EEG cruda.

Las etapas definidas para la implementación de nuestro método fueron:

- Muestreo de eventos de pestañeos del usuario

Se recolectaron 100 eventos de pestañeos en varias pruebas de muestreo, en la figura 15, se pueden visualizar los picos cuando ocurre un pestañeo.

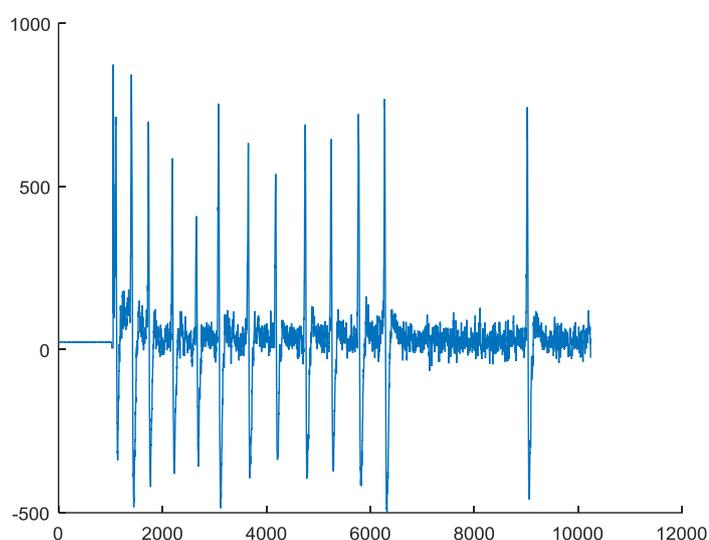


Figura 159. Eventos de pestañeo en la onda cruda.

Luego de eso manualmente se recortó cada uno de estos eventos y se almacenó en una sola matriz que contiene en cada fila un evento. Se notó que todo el proceso de pestañeo dura aproximadamente un cuarto de segundo, por lo que son vectores de 128 datos, en la figura 16 se puede ver que hasta el dato 128 las ondas coinciden, y a partir de ahí los valores dejan de coincidir.

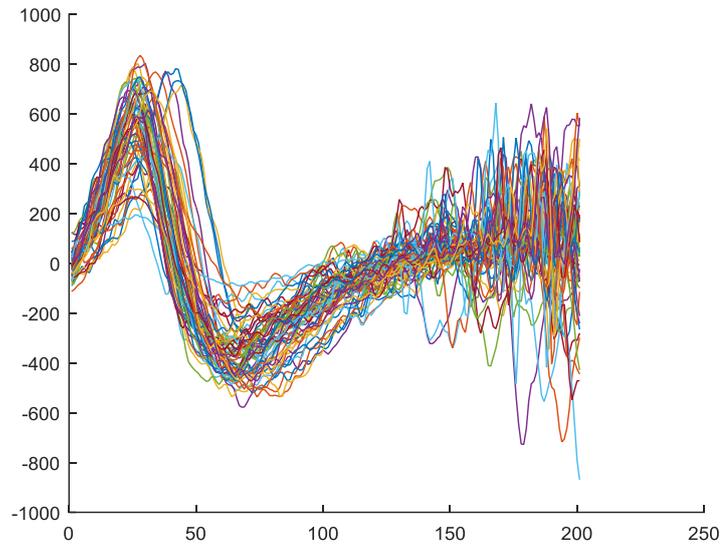


Figura 160. Eventos de pestañeo agrupados.

- Aplicación de filtros suavizadores de onda y extracción de una onda promedio.

Luego se aplicó el filtro a cada vector de 128 valores, esto se puede apreciar en la figura 17. Para tener la onda patrón, se sacó la onda promedio, para luego obtener el vector unitario, esto con el fin de que el pestañeo puede ser fuerte o suave, de igual manera tiene que detectarlo, como se la puede visualizar en la figura 18.

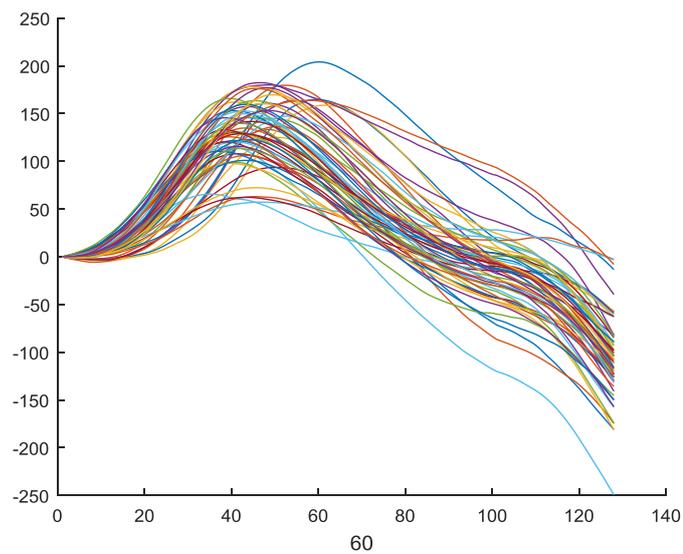


Figura 161. Onda de eventos de pestañeos, aplicado el filtro de respuesta al impulso finito.

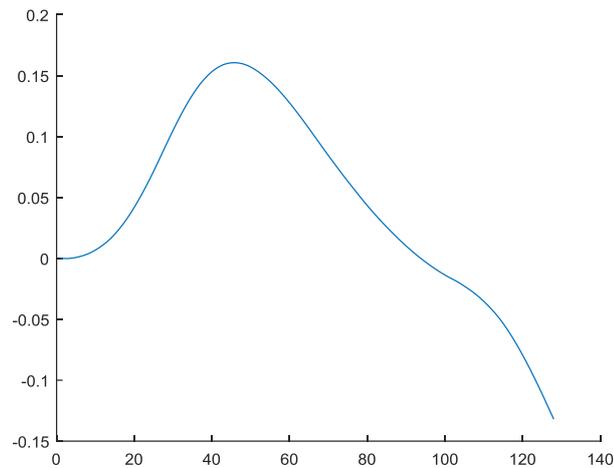


Figura 162. Onda patrón para la detección de un pestañeo.

- Implementación del algoritmo que compare la onda de entrada con la onda promedio de pestañeo obtenida, para verificar un evento de pestañeo

Antes de probar la detección, se tiene que tomar 128 valores e irlos desplazando a medida que llegan nuevos valores, este será nuestro vector de entrada. Al vector de entrada se tiene que pasar por el filtro visto en el anterior punto, y a la salida del filtro se tiene que sacar el vector unitario, esta salida la restamos de la onda patrón y si no existe un error menor de 0.04 en ningún punto, entonces es un evento de pestañeo. En la figura 19 se puede ver este proceso. Como resultado de cien pestañeos que se realizaron detecto los 100 pestañeos.

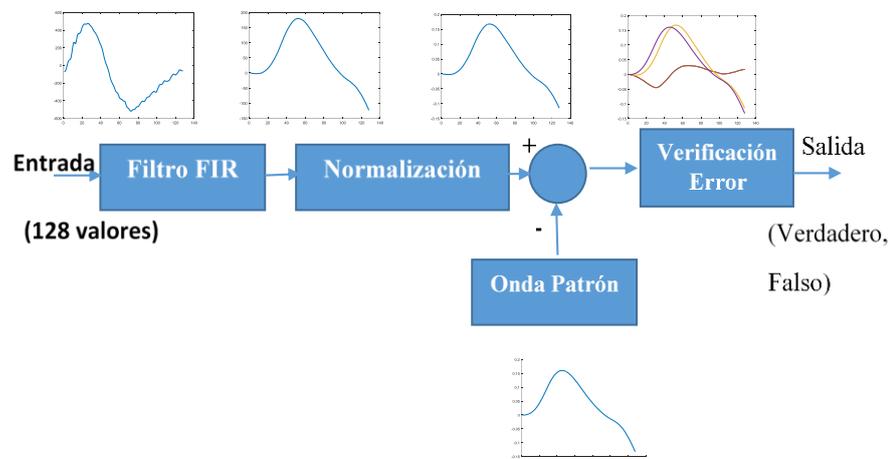


Figura 163. Esquema del algoritmo para verificación de pestañeo.

4.3.1.5 Extracción de características de la señal EEG

En esta etapa para la selección de características cabe recalcar que aquí se plasma el resultado de un proceso iterativo, para la selección de características, así como la selección de los gestos y movimientos que mejor se diferencien para tener el menor error al momento de entrenar la red neuronal. Los comandos que guiarán la silla de ruedas por mejor responder con sus características en la red neuronal son:

- Gesto con la mejilla izquierda (Adelantar)

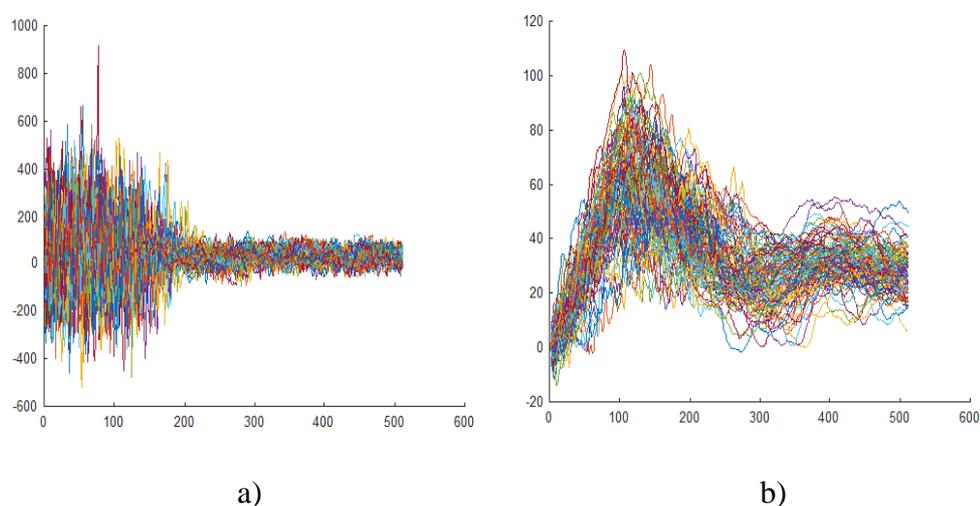


Figura 164. a) Gesto con la mejilla izquierda sin filtrar; b) gesto filtrado.

- Giro de la cabeza hacia la derecha (Detener)

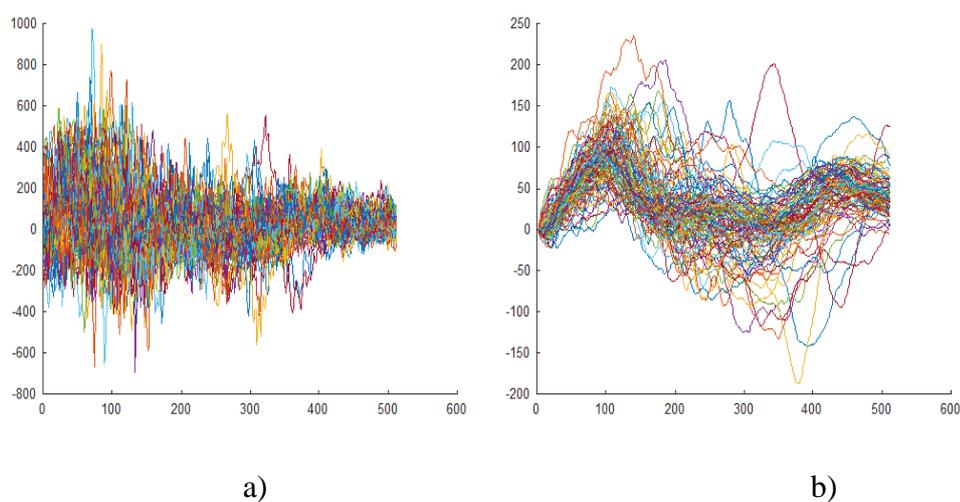


Figura 165. a) Giro de cabeza a la derecha sin filtrar; b) giro filtrado.

- Doble pestañeo (Girar Izquierda)

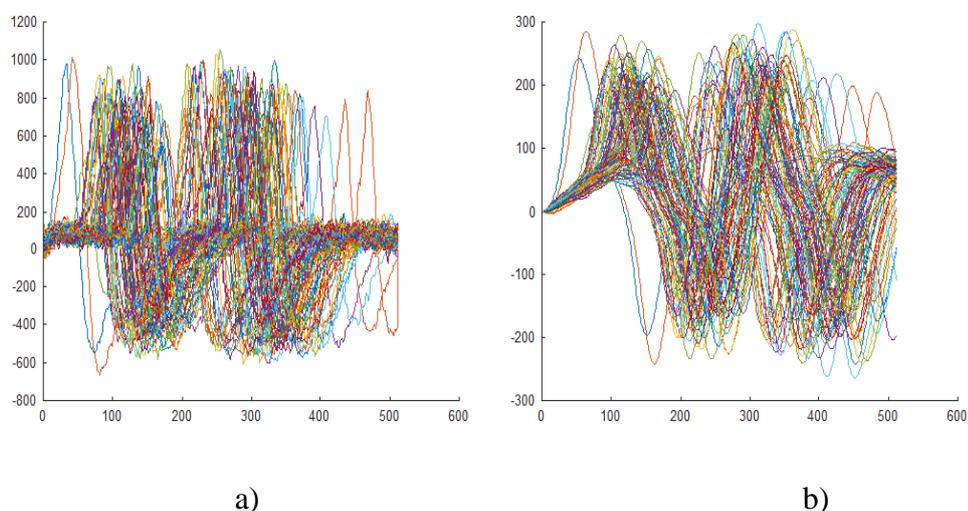


Figura 166. a) Doble pestañeo sin filtrar; b) doble pestañeo filtrado.

- Guiño derecho (Girar Derecha)

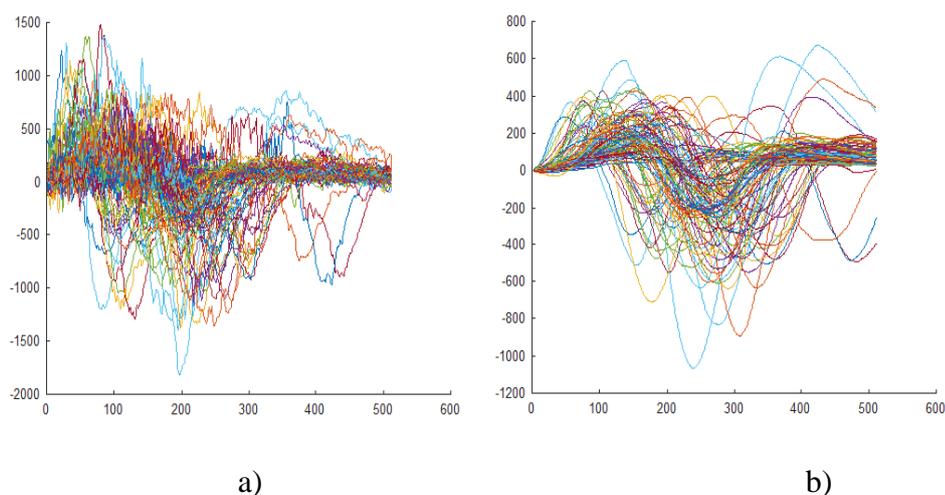


Figura 167. a) Guiño derecho sin filtrar; b) guiño filtrado.

Las características de la onda EEG que se probaron son:

- Potencias de la transformada Discreta de Fourier.

De la figura 24 a la 27 se puede observar como las potencias de las componentes frecuenciales de los diferentes gestos tienen diferentes rangos y, aunque las medias de cada uno están cerca, se puede diferenciar su valor. Pero a partir de la décima séptima, ya casi no hay diferencia entre estos valores, como se puede apreciar en la figura 28, por lo que se tomó como característica solo las dieciséis potencias

primeras de las componentes frecuenciales. La implementación tanto en Matlab como en Python se los realizo con librerías de la transformada rápida de Fourier.

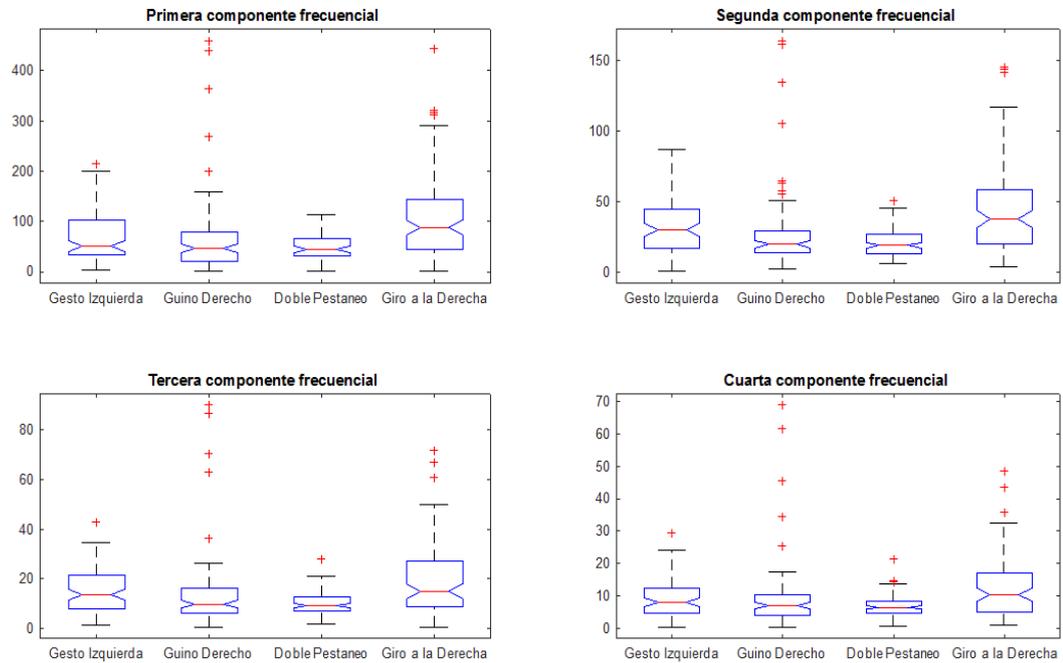


Figura 168. Primera a la cuarta componente de la transformada de Fourier, categorizadas por tipo de gesto.

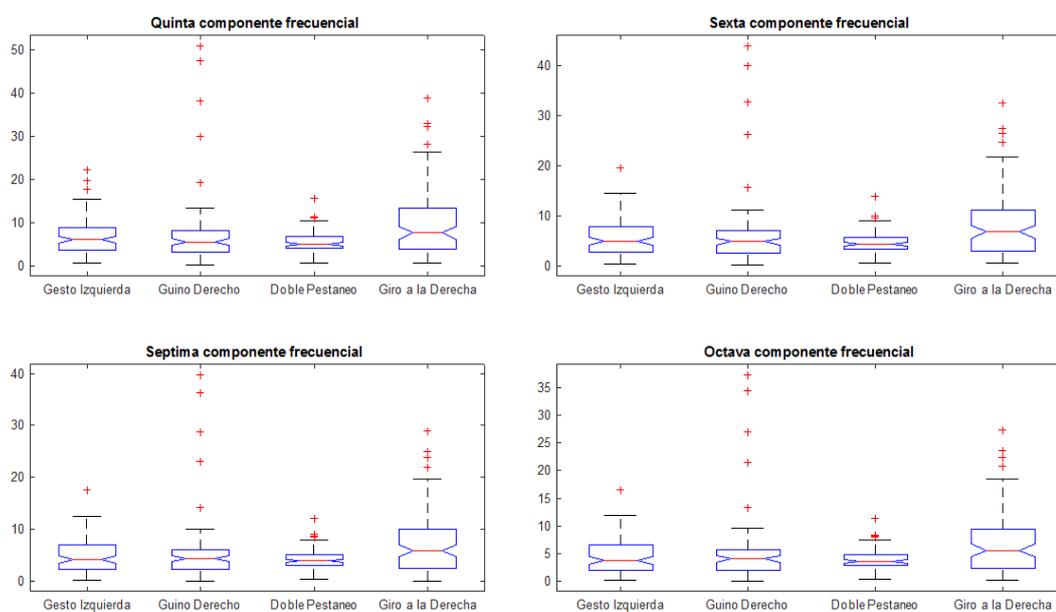


Figura 169. Quinta a la octava componente de la transformada de Fourier, categorizadas por tipo de gesto.

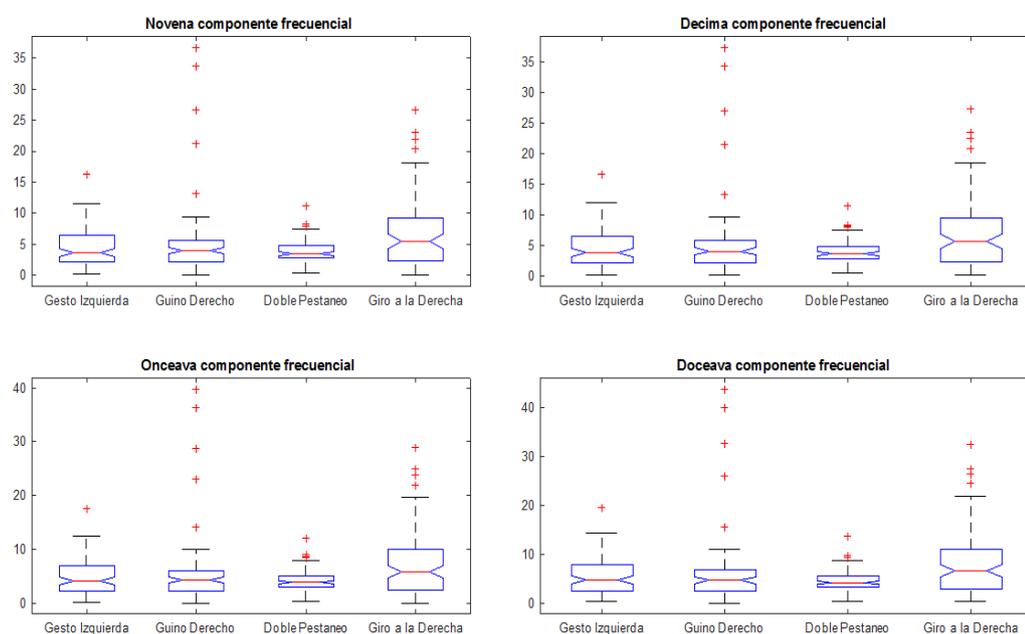


Figura 170. Novena a la doceava componente de la transformada de Fourier, categorizadas por tipo de gesto.

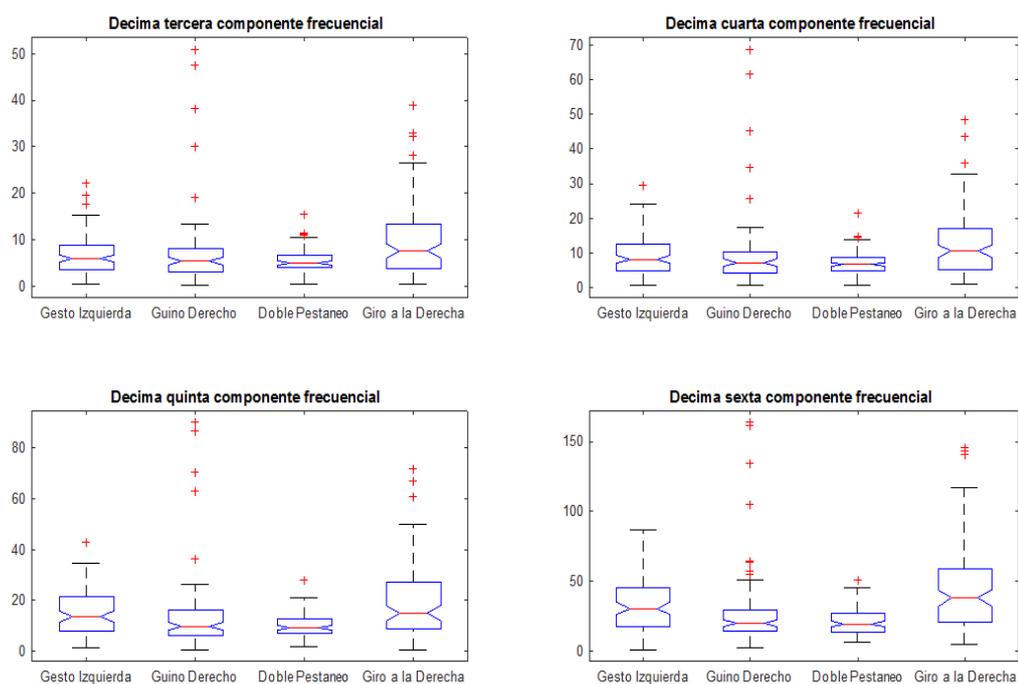


Figura 171. Décima tercera a la décima sexta componente de la transformada de Fourier, categorizadas por tipo de gesto.

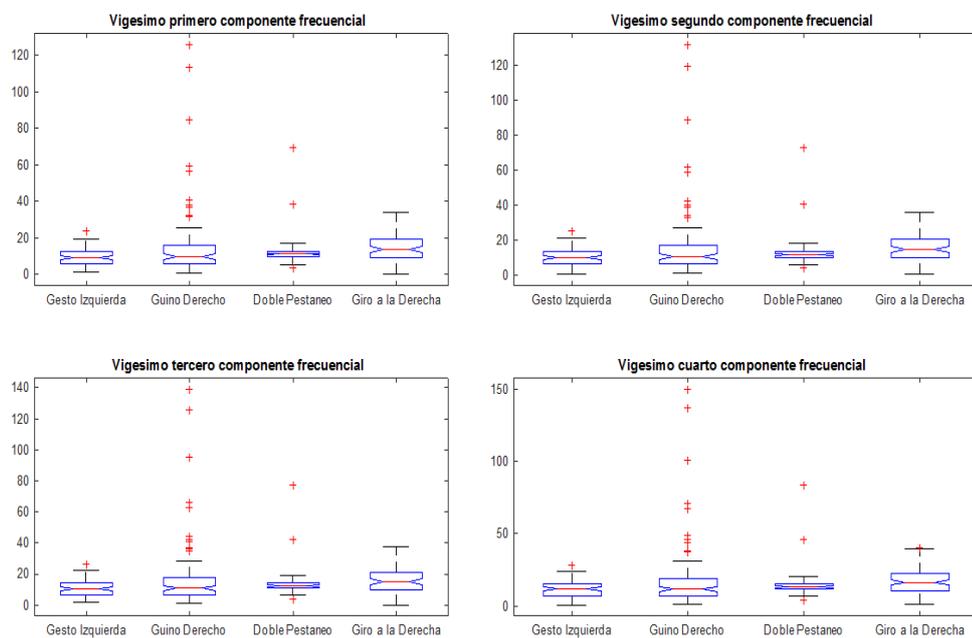


Figura 172. Vigésimo primero al vigésimo cuarto componente de la transformada de Fourier, categorizadas por tipo de gesto.

- Valores Máximos y mínimos

Una característica que les diferencia a los diferentes comandos es obtener los valores de máximo y mínimo de la onda. Como se puede ver en la figura 29, existe una gran diferencia entre los valores mínimos y máximos de los diferentes gestos, por lo que se han incluido estas dos características

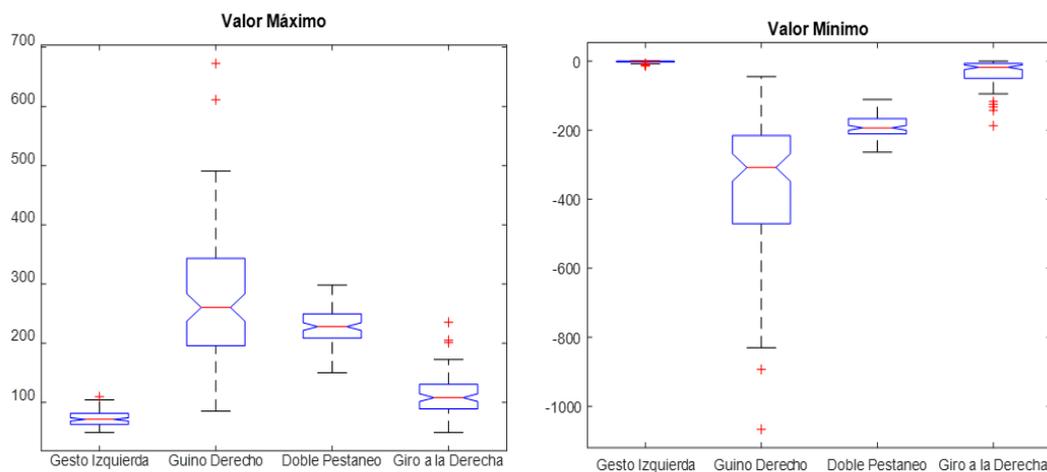


Figura 173. Valores Máximos y mínimos de la onda.

- Promedio y desviación estándar

Una característica importante es la media de los valores, en este caso el tamaño de onda es de 512 datos, el promedio de estos datos es otra característica, del gráfico de la media de la figura 30, se puede ver que la concentración de valores es diferente en los distintos comandos, por lo que se incluyó la desviación estándar que nos dice que tan cerca de la media están dispersados los valores.

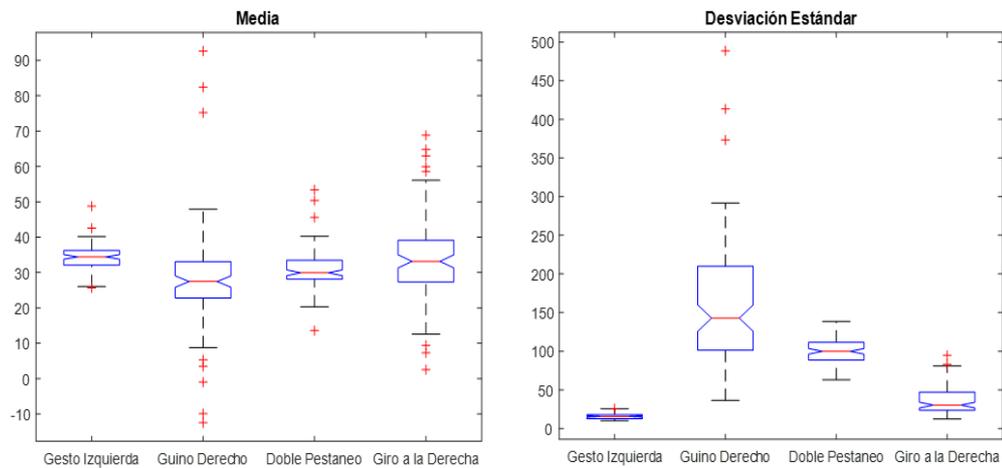


Figura 174. Media y desviación estándar de los valores de la onda.

- Suavidad total y en los últimos valores

La suavidad de la onda nos dice que tan variables son los valores, En este caso hemos implementado una propia función para medir la suavidad. Esta función lo que hace es sumar los valores absolutos de las restas entre los valores en el vector y los anteriores valores. En la figura 32 se puede ver el código para implementar esta función en Matlab, como se puede apreciar lo único que se hace es al vector de entrada se hace un recorrido sumando los valores absolutos de la resta del valor de la siguiente posición del vector con la actual.

```
function st = smoothness(vectorIn)
    [m, n] = size(vectorIn);
    st = 0;
    for i = 1:n-1
        st = st + abs(vectorIn(i+1)-vectorIn(i));
    end
end
```

Figura 175. Función para medir la suavidad de la onda.

De la misma forma se hace este cálculo para los últimos veinte valores. Con esto logramos de alguna manera incluir a la red el tiempo que dura cada comando, dado que los comando que más tiempo duran en ejecutarse, más suavidad tiene en sus últimos valores y viceversa los que duran menos tienen menos suavidad.

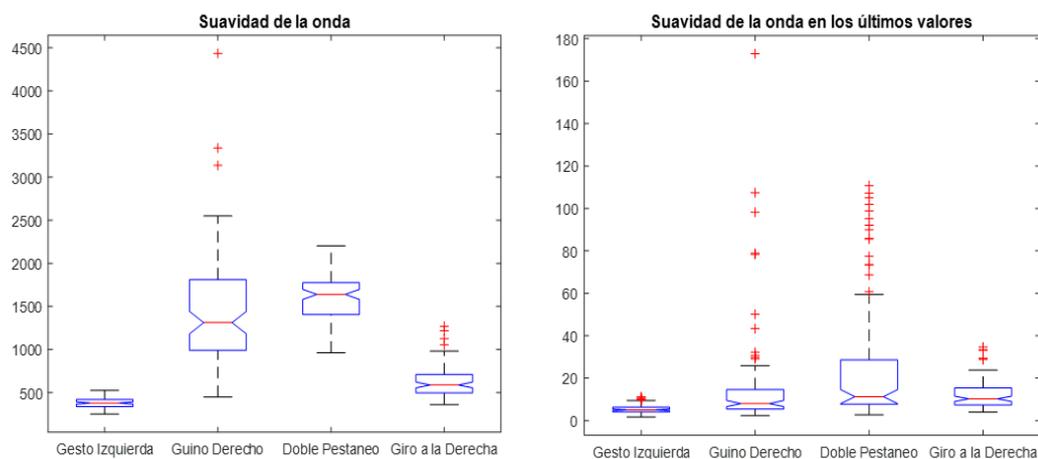


Figura 176. Suavidad de la onda en total y en los últimos valores.

- Número de picos en alto y bajo

El número de picos de la onda es otra forma de medir la suavidad de la onda, esto también se incluyó como características en la red neuronal. En la figura 33 se puede apreciar como varían en los diferentes comandos.

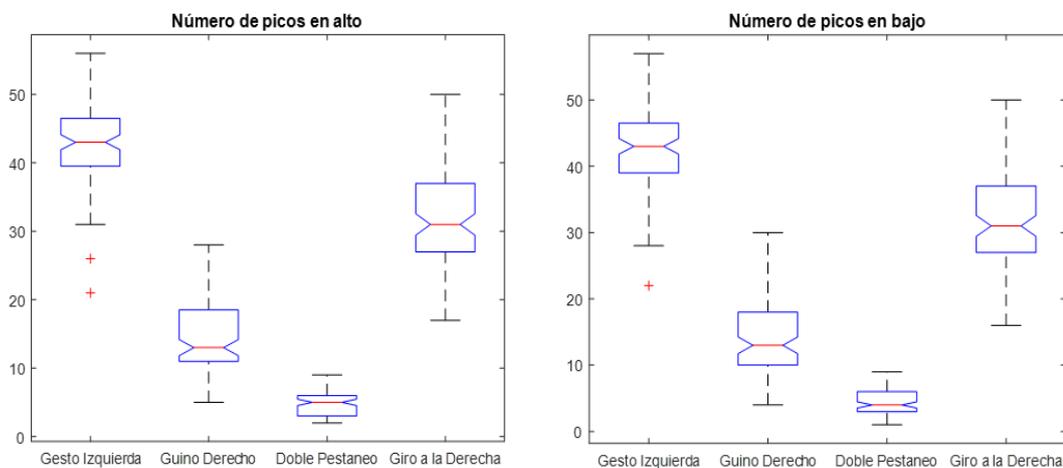


Figura 177. Número de picos en alto y bajo.

- Suma de los valores de los picos en alto y bajo

De las gráficas de las figuras 20 a la 23 se puede apreciar que están en diferentes rangos los valores de cada comando por lo que con la suma de los valores de los picos, nos servirán para incluir esta característica en la red neuronal. En la figura 34, se aprecia esta característica como distingue a los diferentes comandos.

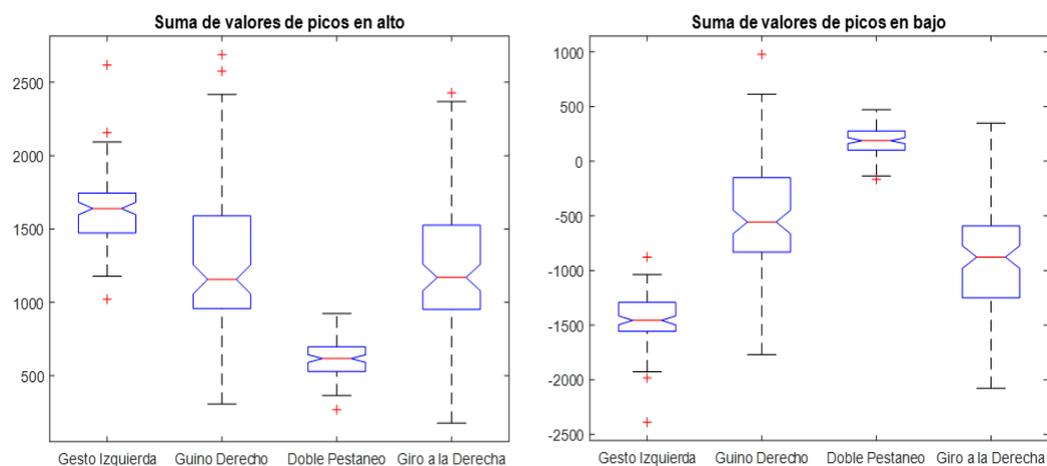


Figura 178. Suma de los valores picos en alto y bajo.

- Media de los valores de localización de los picos en alto y bajo

Para incluir la característica de la localización de los picos, se ha calculado la media de esta. Con lo que obtenemos las gráficas de la figura 35, que nos ayudan a estimar que comando se está realizando.

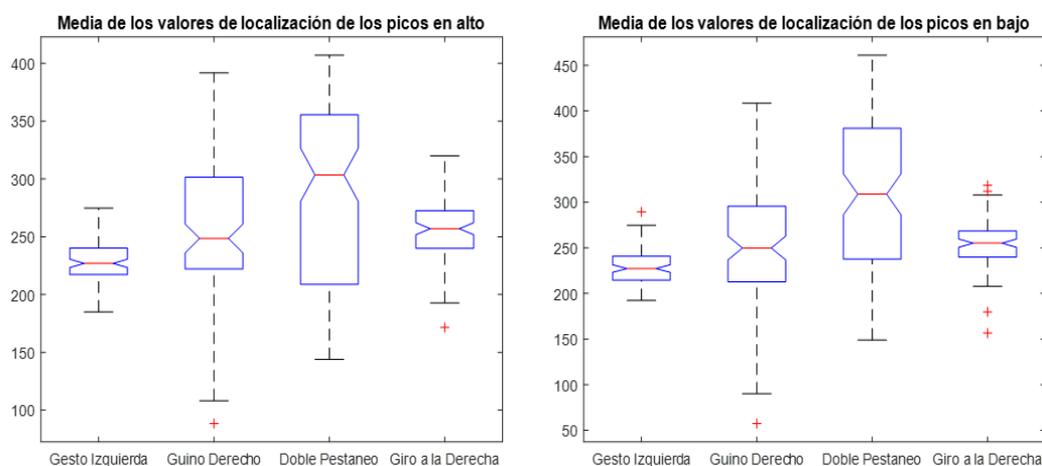


Figura 179. Media de la localización de los picos en alto y bajo.

Es importante recalcar que todas estas funciones, han sido implementadas tanto en Matlab y en las envolturas en C para Python, y se ha verificado que den exactamente el mismo valor y con la misma precisión.

4.3.1.6 Escalamiento de los valores de las características de la señal EEG

Como se vio en el diseño, el escalamiento de los valores es necesario para un buen funcionamiento de la red neuronal. En nuestro caso para el entrenamiento de la

red utilizamos la herramienta de Matlab para Redes Neuronales, la cual nos entrega los valores necesarios para realizar un escalamiento correcto. La fórmula que aplica Matlab para esto es la siguiente:

$$x_o = \frac{2(x_{in} - x_{min})}{x_{range}} + 1 \quad (4.4.1)$$

Donde:

x_o = Es el valor escalado

x_{in} = Es el valor a escalar

x_{min}

= es el valor mínimo obtenido en los valores obtenidos en el entrenamiento

x_{range} = es la resta entre el valor máximo y mínimo

Los valores x_{min} y x_{range} , nos entrega la red neuronal al analizar los valores de las características de entrada. La ecuación (4.4.1) lo que hace es escalar los valores de entrada en el rango de -1 y 1. En la figura 36, se puede ver su implementación, para un vector de entrada.

```
double *mapInput(double *inputV, double *xmin, double
*xrange)
{
    static double vectorOut[inputSize];
    int i;
    for(i=0;i<inputSize;i++)
    {
        vectorOut[i]=2*(inputV[i]-xmin[i])/xrange[i]-1;
    }
    return vectorOut;
}
```

Figura 180. Implementación del escalamiento en lenguaje de programación C.

4.3.1.7 Red Neuronal

La estructura utilizada en la implementación de la red neuronal se la puede ver en la figura 37. Como se puede ver se tienen 29 entradas con todas las características usadas y cuatro salidas con los comandos. Para la capa oculta la función implementada en cada uno de los 100 nodos es llamada “tansig”.

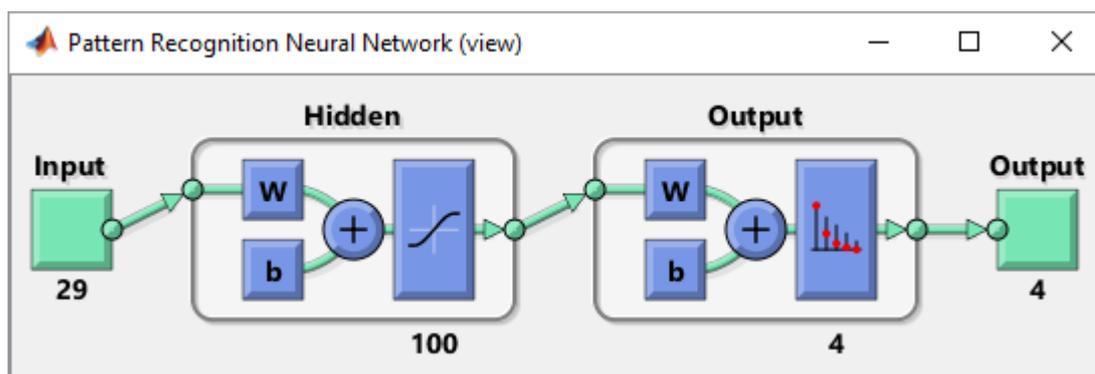


Figura 181. Red neuronal Implementada.

La función tansig es igual que la función sigmoid pero esta asigna valores entre -1 y 1, como se puede apreciar en la figura 38. En la capa de salida ha escogido la función softmax, que pone los valores entre 0 y 1

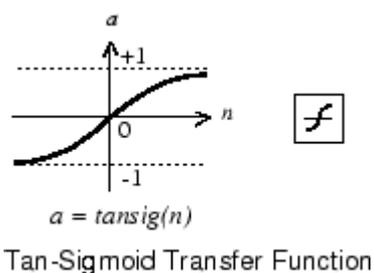


Figura 182. Función tangente hiperbólica sigmoidea.

Fuente: (Mathworks, 2016)

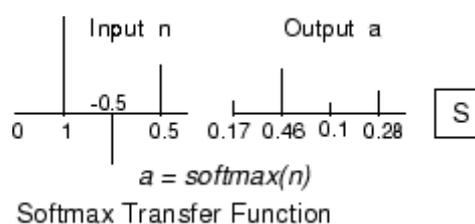


Figura 183. Función softmax.

Fuente: (Mathworks, 2016)

Como se dijo el entrenamiento de la red neuronal se realizó en Matlab, para ello se incluyeron 100 muestras de cada uno de los cuatro comandos. El 70% se utilizó para el entrenamiento, 15% para la validación de parámetro y 15% para la prueba de desempeño. El rendimiento de este entrenamiento se puede visualizar en la figura 40.

Como se puede ver se obtuvo el mejor rendimiento en la iteración 163, con una entropía cruzada de 0.011825

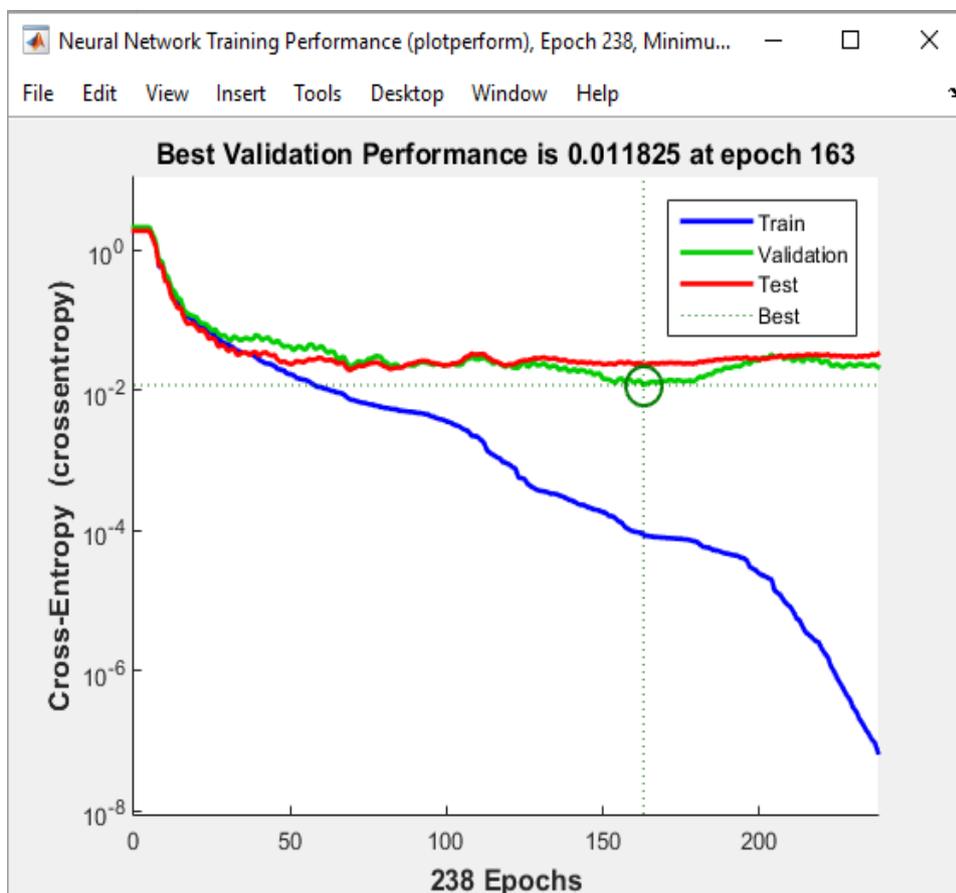


Figura 184. Rendimiento del entrenamiento de la red neuronal.

Para la implementación de la prueba se tiene que pasar los valores de cada uno de los pesos de la red, se ha programado para obtener seis archivos de extensión CSV.

- Pesos de la capa oculta
- Pesos de la capa de salida
- Pesos del bias de la capa oculta
- Pesos del bias de la capa de salida
- Valores mínimos de cada una de las entradas
- Rango de cada una de las entras

En la implementación en Python, lo que se ha hecho es muestrear el sensor en Python, y una vez obtenida la onda se ejecuta envolturas para ejecutar código C, que es en donde se tienen todas las funciones del procesamiento de la red. En el gráfico de la figura 41 se muestra este proceso. Para poder aumentar la eficiencia se han separado

los programas en procesos diferentes, con el fin de no compartir memoria, el programa de muestreo del sensor siempre se ejecuta al igual que el programa principal con el fin de no perder datos. EL programa principal va a ser el que interactúe con los diferentes procesos. Primero solicita la onda muestreada de 512 datos, con esto envía mediante envolturas a ejecutar un programa en código C para obtener las características de la onda y por ultimo ejecutar la red Neuronal que le entrega que comando ha sido realizado, la red neuronal de igual manera esta implementada en lenguaje C.

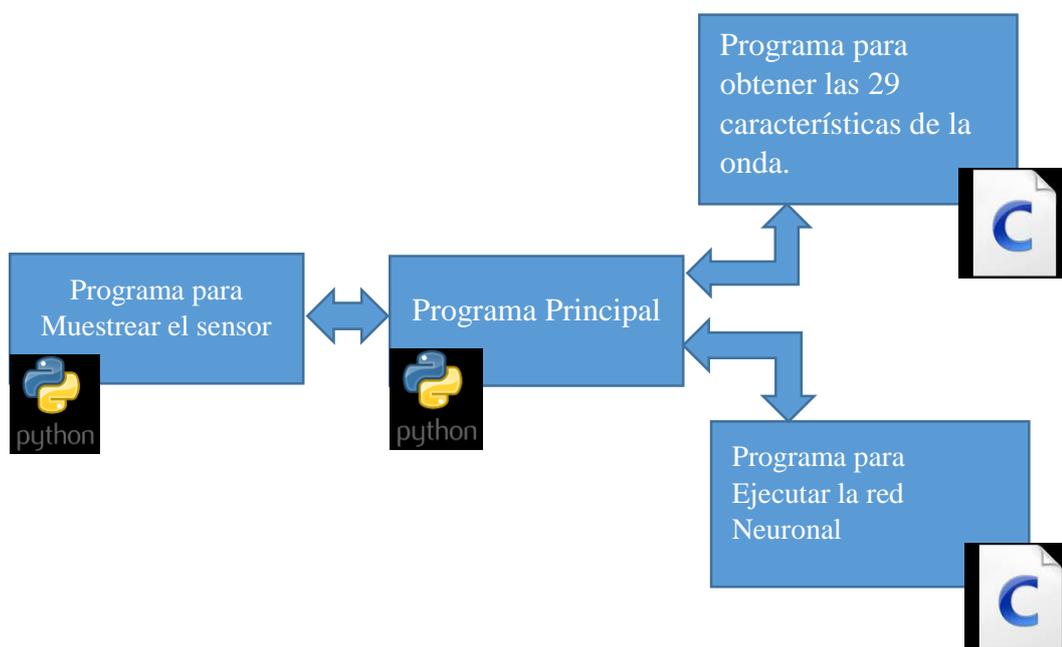


Figura 185. Implementación de la red neuronal en la Raspberry Pi.

Como resultado al implementar la red neuronal distingue correctamente todos los comandos enviados, pero se tiene que tener cuidado de realizar los gestos preciso sin hacer otros movimientos fuertes porque esto ocasiona que la onda varíe y que se prediga otro comando. En los anexos se tiene todo el código para la implementación de la red.

4.3.2 Implementación del control del motor de la rueda de tracción

La implementación del control de velocidad, se realizó en dos procesos independientes, en donde el muestreo de los datos del sensor está siempre ejecutándose como muestra el diagrama de la figura 43. EL código implementado se encuentra en Anexos. Como se puede ver en la figura 24, todo el control PID se realizó en la

Raspberry, se comunica con el driver y recibe datos del sensor mediante la placa de control.

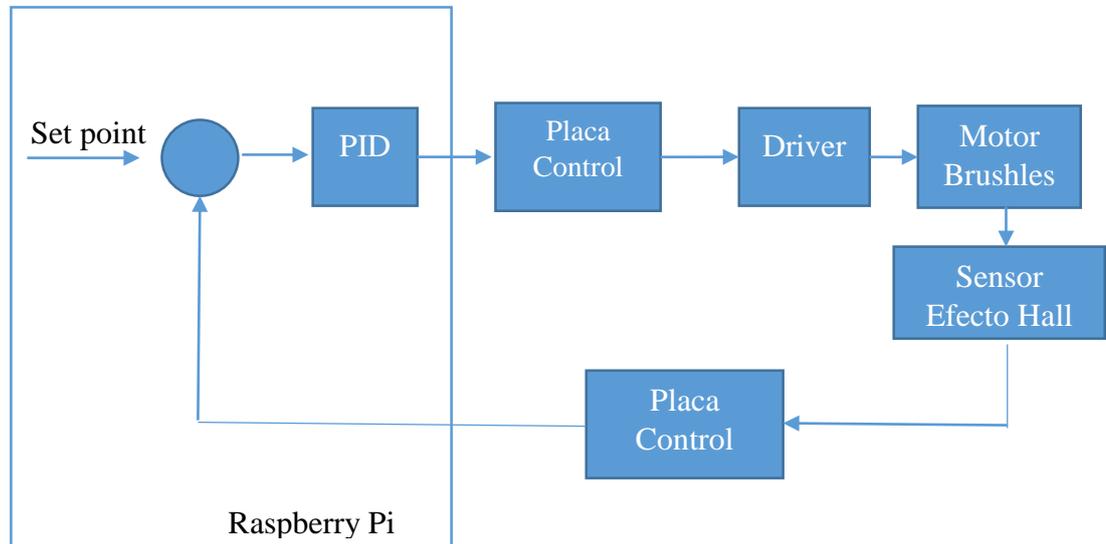


Figura 186 Diagrama de Control PID para el motor DC Brushless

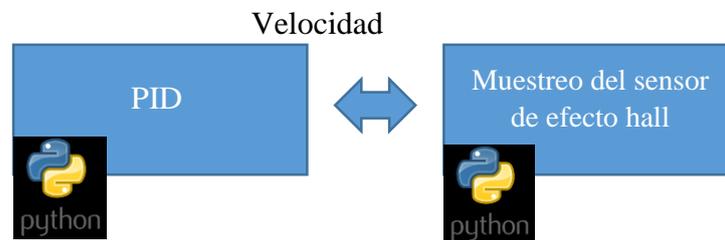


Figura 187. Procesos implementados en la Raspberry para el control de velocidad del motor brushless.

El resultado de la implementación es que el control PID funciona mejor a altas velocidades, dado que para romper la inercia el motor necesita mayor esfuerzo por lo que a velocidades bajas el tiempo de estabilización es mayor.

4.3.3 Implementación del control del motor de dirección

Para el control del motor a pasos se implementó una función, para ejecutar el proceso de girar la rueda, enviando un tren de pulsos a una frecuencia específica. No se utilizó ninguna librería debido a que se necesitaba velocidad y precisión para no perder ningún pulso. EL código implementado se lo puede ver en la figura 44, el código lo que hace es crear un proceso exterior para no compartir memoria con el programa principal y ejecutar con precisión el programa, el programa principal se detiene hasta que termine de ejecutarse el código.

```
def turnStepperMotor(steps, frequency):
    period = 1.0/(2 * frequency)
    for step in range(steps):
        GPIO.output(signalStepMotorPin, True)
        sleep(period)
        GPIO.output(signalStepMotorPin, False)
        sleep(period)

def turn():
    # StepMotor Process
    turnStepMotorProcess = mp.Process(target=turnStepperMotor,
    args=(1000, 500))
    turnStepMotorProcess.start()
    turnStepMotorProcess.join()
```

Figura 188. Código para el giro de la rueda.

El resultado que se obtuvo es que la rueda gira con un mejor rendimiento a una frecuencia de 500Hz, dado que se tiene la mayor velocidad, pero sin que el mecanismo genere un ruido alto, y sin perder ningún pulso.

4.3.4 Implementación control motor de frenado

Para el servomotor, se hizo uso de la librería GPIO de la Raspberry Pi, con la función la clase PWM, en donde se crea un objeto y se configura la frecuencia, y se puede cambiar el ciclo de trabajo de la onda en la figura 45 se puede ver la implementación de un objeto de la clase PWM. Para nuestro caso la frecuencia a la que mejor desempeño tiene el servo es a 50Hz, dado que genera la menor cantidad de ruido y la mayor fuerza. El porcentaje de ciclo de trabajo a la que frena el motor es de 94%, y mediante esto se frenará la silla en el programa principal cuando sea necesario. EN el programa principal el frenado se realizó dentro del control PID, y se frenara cuando el setpoint sea 0.

```

servoFrequency = 50
servoPWM = GPIO.PWM(servoControlPin, servoFrequency)
servoDutyCycle = 0
servoPWM.start(servoDutyCycle)

```

Figura 189. Uso de la clase GPIO.PWM.

4.3.5 Funcionamiento en conjunto del Sistema de control

En la figura 46 se puede observar el conjunto del sistema de control. Como se puede ver el programa principal es el que interactúa con todos los procesos. EL muestreo de señales EEG, el muestreo de señales del sensor de efecto hall y el PID, son procesos independientes que siempre está ejecutándose. En cambio, el control de freno y de dirección son procesos dependientes que solo se ejecutan cuando el programa del control PID o programa principal les invoca. También se tienen las funciones para el procesamiento que se ejecutan en lenguaje C, mediante los antes nombrados Wrappers o envolturas, que son funciones para ejecutar código en otro lenguaje de programación del que se está ejecutando. Todo esto sigue el diagrama de flujo de la figura XX en el capítulo 3.

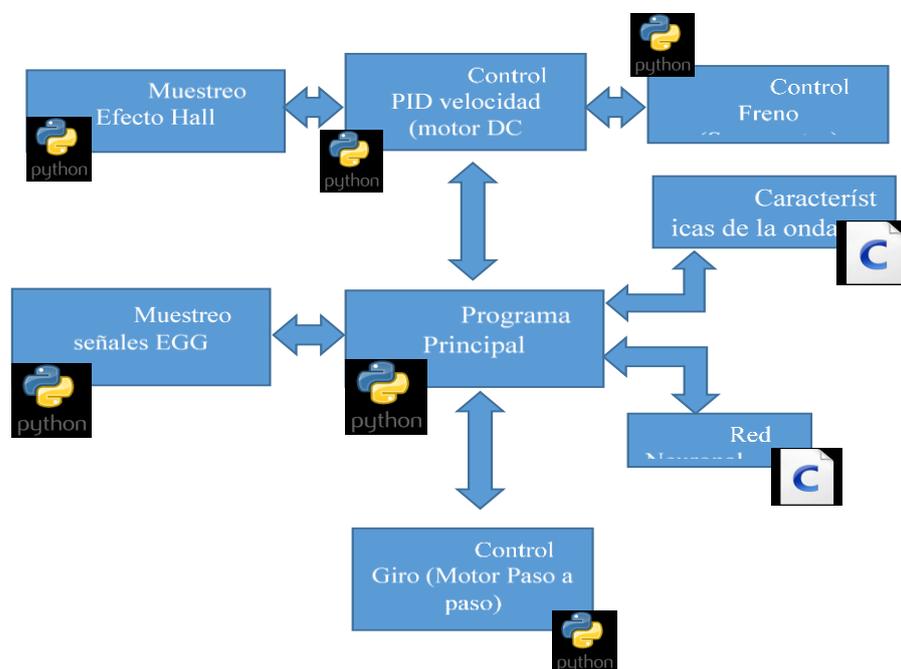


Figura 190. Esquema general, sistema de control

Como resultado, al implementar todo el sistema de control, el sistema funciona correctamente con todas las funciones implementadas.

CAPÍTULO 5

ANÁLISIS ECONÓMICO

El presente capítulo permite conocer los costos incurridos en la parte mecánica y electrónica para la producción de un solo módulo mecatrónico, para trabajos futuros este módulo se orientaría como un producto comercial el cual tendría una disminución significativa de costos si se lo produce en serie.

5.1 Análisis de costos

Se hizo una evaluación de todos los costos de elementos electrónicos, manufactura e implementación que fueron generados durante la en la fase de construcción del módulo, se minimizo los gastos utilizando materiales adecuados, una manufactura dimensionada y sobre todo optimizando el tiempo de construcción.

Tabla 13.

Valores generados.

Elemento	Material	Importación y envío	Manufactura	Total
Kit motor brushless (Rueda con motor DC, driver, disco de freno, mordaza de freno)	227,5	95		322,5
Kit Motor Paso a paso (motor, driver)	135,77	77		212,77
Trinche (Eje, tubo, sujetadores)	15		45	60
Perfil	4		35	39
Sujetadores	5		15	20
Engranajes	10		90	100
Caja de drivers	5		5	10
Cableado	5			5
Raspberry Pi 2	35	40		75
Electrodo Mindwave	115	40		155
Caja protección motores	0	25		25
Servo Motor	60			60
Placa de potencia	10			10
Pintura	8		80	88
			Total	1182,27

En el mercado existen sillas de ruedas eléctricas cuyos costos están entre 1700 a 2000 dólares, mientras que el módulo mecatrónico tiene un costo total de 1182.27 dólares el cual disminuiría sustancialmente en una producción en serie. La gran diferencia del módulo es que se puede adaptar a cualquier silla, dando así una gran ventaja para las personas que ya cuentan con una, caso contrario el costo de una silla común esta alrededor de los 200 dólares. Para un usuario existe un ahorro considerable en su economía, aunque aparte del módulo adquiriera una silla normal.

CAPITULO 6

CONCLUSIONES Y RECOMENDACIONES

En el presente capítulo se presentan las conclusiones a las que se han llegado después del desarrollo del presente proyecto. A su vez, se presentan las recomendaciones que se obtuvieron para la elaboración de mejoras en el módulo mecatrónico, o para proyectos relacionados con las líneas de investigación que se han abordado.

6.1 Conclusiones

- Se diseñó y construyó un sistema mecánico, el cual se adapta a diferentes modelos y tamaños de sillas de ruedas, mediante la regulación de la posición de sus acoples. A su vez, contiene mecanismos que permiten el desplazamiento, giro y frenado de sillas de ruedas, para usuarios con un peso de hasta 100 Kg. Una característica destacable es que el sistema es muy compacto permitiendo un fácil traslado cuando el módulo esta desmontado. El diseño del sistema permite hacer una distribución óptima en cuanto refiere al peso de los componentes, con lo que dota de estabilidad y de una tracción óptima a la rueda del módulo.
- Se diseñó e implemento un sistema electrónico, que permite la manipulación de todos los actuadores. Dentro de esto se incorporó un circuito de potencia que permite la interacción del sistema de control con los motores. Dicho circuito contiene protecciones para prevenir sobrecargas o cortocircuito, protegiendo así los componentes de la etapa de control, además contiene reguladores para proveer a cada componente un voltaje adecuado.
- Se implementó un control PID en la velocidad del motor brushless que permite el desplazamiento de las sillas de ruedas. Para ello el cálculo de la velocidad se realiza a través del sensor de efecto hall que contiene la rueda. El control permite desplazarse a una velocidad segura de cincuenta metros por minuto, dando al usuario suficiente tiempo para enviar las instrucciones.
- Se implementó un control en lazo abierto en el mecanismo para el giro. El control se realiza en el motor de pasos, girando en sentido horario o anti horario a partir

de la posición de encendido del módulo mecatrónico. La velocidad en la que gira optimiza la fuerza del mecanismo y no produce un ruido alto.

- Se implementó un control en el mecanismo para el frenado. Para ello se controla la posición del servomotor para cerrar o abrir la mordaza en el freno de disco. Con el control logramos un frenado, a la velocidad de 50m/min, en un segundo sin ocasionar movimientos bruscos.
- Se implementó un sistema de control comandado por señales EEG. Antes de utilizar este sistema de control se debe generar una base de datos del usuario, para el entrenamiento de la red neuronal. El entrenamiento de la red se lo realiza en un PC y luego se cargan los archivos obtenidos a la Raspberry. La red neuronal se diseñó, entreno e implemento obteniendo un alto rendimiento (entropía cruzada baja). El sistema funciona correctamente, cuando el usuario pestaña dos veces seguidas se anticipa una instrucción y mediante la caracterización de la onda EEG, la red neuronal predice la instrucción enviada. Todo este sistema de control está implementado en la Raspberry Pi 2.
- Se ensamblaron todos los componentes de control dentro una caja, que se adapta mediante correas regulables en cualquier espacio de las sillas de ruedas, optimizando espacio.
- El sistema electrónico, mecánico y de control interactúan perfectamente entre ellos dotando al usuario de comodidad y seguridad al momento de manipular el módulo mecatrónico.

6.2 Recomendaciones

- Se recomienda que para una segunda versión del módulo mecatrónico se sustituya el sensor de señales EEG, con uno que contenga más terminaciones de electrodos, para que sea posible incluir más gestos o movimientos como comandos para el control de las sillas de ruedas.
- Con el fin de aumentar la robustez del módulo, se recomienda incluir, en el sistema mecánico, un mecanismo de suspensión con lo que se lograría desplazar las sillas de ruedas sobre terrenos irregulares.
- Al implementar el proyecto se vio necesario hacer uso de un monitor en la Raspberry Pi, dado que se requería una monitorización de los procesos y una

mayor interacción con el usuario. Recomendamos se incluya este monitor en futuras versiones y se cree una interfaz gráfica.

- Por el costo de implementación del mecanismo de frenado, se recomienda sustituir el servomotor por un motor DC de bajas revoluciones, cuando se construyan versiones comerciales.
- Para mejor el procesamiento y la capacidad de respuesta del sistema de control, se recomienda utilizar la última versión de Raspberry Pi, la cual cuenta con una velocidad del procesador superior y posee un módulo Bluetooth incorporado.
- Dadas las facilidades que ofrece una Single Board Computer como es la Raspberry Pi, se recomienda hacer uso del sistema de control comandado por ondas EEG, para implementar módulos exteriores que permitan el manejo de dispositivos remotamente, como por ejemplo un módulo que se comunique inalámbricamente y permita encender televisores, abrir puertas, encender luminarias, abrir cerrar cortinas, etc.

REFERENCIAS BIBLIOGRÁFICAS

- Morris, R., & Fillenz, M. (2003). *LA CIENCIA DEL CEREBRO*. Liverpool: La Asociación Británica de Neurociencias .
- A. Ruiz, F. B. (2011). *Adquisición y procesado de información EMG*. Madrid .
- Abad, D. (2008). *Monitor EMG* . Bellaterra.
- Andes, U. d. (27 de 02 de 2015). *Universidad de los Andes*. Obtenido de <https://ingenieria.uniandes.edu.co/lab-ingenieria-de-sistemas-y-computacion>
- Barrera, P., Rios, M., & Olivera, P. (2014). *Equipo Lázaro*. Obtenido de Lázaro Recuperando Autonomía: <http://www.productoslazaro.cl/>
- Barriga, A. (2011). *Funadamentos de robótica*. Mexico.
- Berger, N. (2013). *Boston Dynamics*. Holland.
- Bertran, A. A. (2009). Cerebro y sitema nerviso. *Enciclopedia Salud*.
- Brasil, U. d. (27 de 02 de 2015). *Grupo de Mecatrónica*. Obtenido de <http://www.mecatronica.eesc.usp.br/mecatronica/index.php/pt/laboratorios/laboratorio-de-manipulacao-robotica>
- Brubaker, C. (2000). *Ergonomic considerations* . Whasington: Bethesda.
- Cifuentes, I. (2010). *Sistema para la detección de señales electromiográficas* . Yucatán.
- Clausing. (2003). *Quality Funtion Deployment (QFD)*.
- Dierssen, M. (2007). *Viaje al universo neuronal*. España: Fundación española para la ciencia y tecnología.
- Escuela Politécnica Federal de Lausana. (2015). *Delftech Chair Brain Machine Interface*. Obtenido de <http://cnbi.epfl.ch/page-34059.html>
- Fowler, B. (2009). *Dinámica problemas de ingeniería*.
- García, F. C. (2005). *Fundamentos Biológicos del Aprendizaje y la memoria*. Universidad de Huelva.
- Guillén, P. (2009). Máquinas de aprendizaje para clasificar señales. *Avances en sistemas e informática*.
- Huaster. (2002). *House of Quality*.
- Jesús Borges, J. C. (2005). *Nervios Craneales*. UPR-Aguadilla.
- Kandel, E. (2001). *Neurociencia y Conducta*. España: Prentice Hall.
- L. Hodgkin, A. H. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve.
- Lamas, J. (2005). Evolución del concepto de potencial de reposo neuronal. *Revisión en neurociencia*, 549-540.
- Lamas, J. (2005). Evolución del concepto de potencial de reposo neuronal. *REVISIÓN EN NEUROCIENCIA*, 538-549.
- Leucoma, M. (2010). *Biología Celular o Histología Médica*. México.
- Logroño, M. (2010). Sistema Nervioso . *Biología y Geología*, 5.
- Manosalvas, F. (27 de 01 de 2015). *AMBIENTE*. Obtenido de <http://www.ambiente.gob.ec/wp-content/uploads/downloads/2013/03/PART3.pdf>
- Manoslavas, F. (28 de 01 de 2015). *AMBIENTE*. Obtenido de <http://www.ambiente.gob.ec/wp-content/uploads/downloads/2013/03/PART11.pdf>
- Martínez, N. (2010). *Modelación de la actividad neuroeléctrica*. Mexico: UNAM.
- Masso, N. (2010). Aplicaciones de la electromiografía . *Apunts*, 129-131.

- McGRAW-HILL. (1997). *Fundamentos de Robotica*. España: Concepcion Fernandez Madrid.
- Medical, S. (2006). *Consideraciones Biomédicas*. Santiago de Chile: Omega .
- Merino, J., & Noriega, M. (2011). COMUNICACIÓN NEURONAL: SINAPSIS. *Open Course Ware*.
- MINEM. (27 de 01 de 2015). Obtenido de <http://www.minem.gob.pe/minem/archivos/file/institucional/regionales/Publicaciones/GUIA%20HIDROCARBUROS%20III.pdf>
- Ministerio de industrias y de productividad. (2011). *Boletín mensual de analisis sectorial MIPYMES*. Recuperado el 1 de Junio de 2014, de Ropa de Vestir de cuero para exportación: <https://www.flacso.edu.ec/portal/pnTemp/PageMaster/dvwqb97pv9quz2d9pwwqomo3wnqad7.pdf>
- MIT. (27 de 02 de 2015). *MERS*. Obtenido de <http://groups.csail.mit.edu/mers/>
- MIT. (27 de 02 de 2015). *Touch Lab*. Obtenido de <http://touchlab.mit.edu/>
- Montes, F. (2007). *Procesos estocásticos en Biomédica*. Valencia.
- Morales, J. E. (2011). *La estructura del sistema nervioso*. Oaxaca: Conductitlan.
- Navarro, R. B. (2010). *Electroencefalografía*. Alcalá.
- Nerg, H. (2015). *The Space Robotics Lab*. Obtenido de <http://www.robotics.com>
- Ortega, M. (21 de 2014). Interpretación del EEG. *Sapiens Medicus*, 14-22. Obtenido de <http://sapiensmedicus.org>.
- Plonsey, R., & Barr, R. (2007). *Bioelectricity A Quantitative Approach* (Third ed.). New York: Springer.
- Rallete, C. (29 de 06 de 2015). *RIA, Robotic Industry Association*. Obtenido de www.RoboticIndustryAssociation.com
- RCN. (13 de 02 de 2015). Dispositivo para mover sillas de ruedas con los ojos. *RCN La Radio*.
- Reyes Cortés, F. (2011). *Robótica-Control de robots manipuladores*.
- Reyes-Haro, D., Bulavina, L., & Pivneva, T. (2014). La glía, el pegamento de la ideas. *Revista Ciencia*, 12-18.
- Robo-control*. (2015). Obtenido de <http://www.sensorsrobotics.com>
- Rodriguez, J. (18 de 11 de 2012). *Manipulador robótico*. Monte Blanco, Mexico.
- Roseto, A., Repetto, H., & Exeni, R. (2003). *Las membranas celulares. Los canales iónicos y nefrologia celular*. Buenos Aires: Sociedad Argentina de Pediatría.
- Shigley. (2008). *Diseño de ingeniería mecánica*.
- System, R. O. (2015). *ROS*. Obtenido de <http://www.ros.org/>
- Talamillo, T. (2011). *Instrumentación Biomédica*. Santiago de Chile.
- TENARIS. (28 de 01 de 2015). *TENARIS S.A*. Obtenido de <http://www.tenaris.com/es-ES/QHSE/EnvironmentAndEnergyManagement.aspx>
- Thevenet, D. (2008). Electricidad, Electrónica e Instrumentación Biomédica. 5-10.
- Tohoku, U. (27 de 02 de 2015). *The Space Ronotics Lab*. Obtenido de <http://www.astro.mech.tohoku.ac.jp/e/>
- Trossen robotics*. (2015). Obtenido de <http://www.trossenrobotics.com>
- Ulrich, K. T. (2013). *Diseño y desarrollo de productos*. Lisboa, Portugal.

ANEXOS