



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO  
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

TEMA:

**Implementación de algoritmos de programación y de protocolos de  
comunicación inalámbrica para la operatividad de un robot sumo  
autónomo y RC**

AUTOR:

Cordero Pérez, John Paul

Trabajo de Titulación previo a la obtención del título de  
**INGENIERO EN TELECOMUNICACIONES**

TUTOR:

M. Sc. Zamora Cedeño, Néstor Armando

Guayaquil, Ecuador

6 de marzo del 2018



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO  
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

**CERTIFICACIÓN**

Certificamos que el presente trabajo fue realizado en su totalidad por el Sr.  
**Cordero Pérez, John Paul** como requerimiento para la obtención del título  
de **INGENIERO EN TELECOMUNICACIONES**.

TUTOR

---

M. Sc. Zamora Cedeño, Néstor Armando

DIRECTOR DE CARRERA

---

M. Sc. Heras Sánchez, Miguel Armando

Guayaquil, a los 6 días del mes de marzo del año 2018



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO  
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

**DECLARACIÓN DE RESPONSABILIDAD**

Yo, **Cordero Pérez, John Paul**

**DECLARÓ QUE:**

El trabajo de titulación **“Implementación de algoritmos de programación y de protocolos de comunicación inalámbrica para la operatividad de un robot sumo autónomo y RC”** previo a la obtención del Título de **Ingeniero en Telecomunicaciones**, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan en el documento, cuyas fuentes se incorporan en las referencias o bibliografías. Consecuentemente este trabajo es de mi total autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance del Trabajo de Titulación referido.

Guayaquil, a los 6 días del mes de marzo del año 2018

EL AUTOR

---

CORDERO PÉREZ, JOHN PAUL



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO  
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

**AUTORIZACIÓN**

Yo, **Cordero Pérez, John Paul**

Autorizó a la Universidad Católica de Santiago de Guayaquil, la publicación, en la biblioteca de la institución del Trabajo de Titulación: **“Implementación de algoritmos de programación y de protocolos de comunicación inalámbrica para la operatividad de un robot sumo autónomo y RC”**, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y total autoría.

Guayaquil, a los 6 días del mes de marzo del año 2018

EL AUTOR

---

CORDERO PÉREZ, JOHN PAUL

# REPORTE DE URKUND

**URKUND**

<b>Documento</b>	<a href="#">Titulacion-Cordero John.docx</a> (D35850090)
<b>Presentado</b>	2018-02-22 20:36 (-05:00)
<b>Presentado por</b>	johnpaulcp1@hotmail.com
<b>Recibido</b>	edwin.palacios.ucsg@analysis.orkund.com
<b>Mensaje</b>	Tesis Final- John Cordero <a href="#">Mostrar el mensaje completo</a> 2% de estas 46 páginas, se componen de texto presente en 7 fuentes.

Lista de fuentes	Bloques	
+	<b>Categoría</b>	<b>Enlace/nombre de archivo</b>
+		<a href="#">Formato TT.docx</a>
+		<a href="#">TRABAJO FINAL GUALLI-VELIZ 24-...</a>
+		<a href="#">TESIS ACTUALIZADA- GSHL.docx</a>
+		<a href="#">tesis final ROBLES - VACA 2.0.docx</a>
+		<a href="http://docplayer.es/64932556-Uni...">http://docplayer.es/64932556-Uni...</a>

UNIVERSIDAD CATOLICA DE SANTIAGO DE GUAYAQUIL FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

TEMA:

Implementación

de algoritmos de programación y de protocolos de comunicación inalámbrica para la operatividad de un robot sumo autónomo y RC

AUTOR:

Cordero Pérez, John Paul

Trabajo de Titulación previo a la obtención del título de INGENIERO EN TELECOMUNICACIONES

TUTOR:

Zamora Cedeño, Néstor Armando

Reiniciar Exportar Compartir

1 Advertencia

## **DEDICATORIA**

Dedico este trabajo a mis padres, quienes estuvieron a mi lado en todo momento, dando su máximo esfuerzo y consejos para verme realizado como un buen profesional.

A mi Novia Nadia Montero quien me brindo su tiempo y ánimos para ayudarme a terminar mi carrera universitaria.

A mis hermanas María Cordero y Grace Cordero quienes me apoyaron en terminar la tesis.

En especial dedico este trabajo a mis tres hijos Mateo, Aida y Naida quienes fueron inspiración pura para poder terminar este trabajo.

**EL AUTOR**

---

**CORDERO PEREZ, JOHN PAUL**

## **AGRADECIMIENTO**

Agradezco a Dios por haberme dado la oportunidad de llegar en donde me encuentro, agradezco a mi familia por brindarme su amor y apoyo incondicional.

Agradezco a mi tutor el Ing. Néstor Zamora por haber estado pendiente de las correcciones de mi trabajo, y por ayudarme organizar mejor las ideas en el desarrollo del mismo.

Agradezco también al Ing. Edwin Palacios por haberme guiado en la realización de este trabajo de titulación y brindarme su apoyo para conseguir los conocimientos y elementos necesarios.

EL AUTOR

---

CORDERO PEREZ, JOHN PAUL



**UNIVERSIDAD CATÓLICA  
DE SANTIAGO DE GUAYAQUIL**

FACULTAD DE EDUCACIÓN TÉCNICA PARA EL DESARROLLO  
CARRERA DE INGENIERÍA EN TELECOMUNICACIONES

**TRIBUNAL DE SUSTENTACIÓN**

f. \_\_\_\_\_

**M. Sc. ROMERO PAZ, MANUEL DE JESÚS**  
DECANO

f. \_\_\_\_\_

**M. Sc. CORDOVA RIVADENEIRA, LUIS SILVIO**  
COORDINADOR DE ÁREA

f. \_\_\_\_\_

**M. Sc. PALACIOS MELÉNDEZ, EDWIN FERNANDO**  
OPONENTE



## ÍNDICE GENERAL

ÍNDICE DE FIGURAS.....	XV
ÍNDICE DE TABLAS.....	XIX
RESUMEN.....	XX
CAPÍTULO 1: INTRODUCCIÓN .....	2
1.1. Introducción.....	2
1.2. Antecedentes .....	2
1.3. Definición del Problema.....	3
1.4. Justificación del Problema.....	3
1.5. Objetivos del Problema de Investigación.....	4
1.5.1. Objetivo General.....	4
1.5.2. Objetivos Específicos.....	5
1.6. Hipótesis. ....	5
1.7. Metodología de Investigación.....	5
CAPÍTULO 2: FUNDAMENTACIÓN TEÓRICA.....	6
2.1. La robótica .....	6
2.2. Aplicaciones actuales de la robótica.....	7
2.2.1. Robótica Quirúrgica .....	7
2.2.2. Robótica industrial y manufactura .....	7
2.2.3. Robótica Espacial .....	8
2.2.4. Robótica Domestica.....	9
2.2.5. Robótica Educativa.....	9
2.2.6. Robótica Recreativa .....	9
2.2.7. Robot Sumo.....	10
2.2.7.1. Tipos de robots sumo .....	10
2.3. Morfología de los robots.....	10
2.3.1. Robots Industriales .....	11

2.3.2.	Robots Móviles .....	11
2.3.3.	Robots Humanoides .....	12
2.4.	Elementos de un sistema robótico.....	13
2.4.1.	Sensores .....	13
2.4.1.1.	Sensores de posición .....	14
2.4.1.2.	Sensores de efecto hall .....	14
2.4.1.3.	Sensores de luz o brillo.....	14
2.4.1.4.	Sensores de contacto .....	14
2.4.1.5.	Sensores ópticos o de luz.....	15
2.4.1.6.	Sensores ultrasónicos.....	15
2.4.1.7.	Sensores de proximidad .....	16
2.4.1.8.	Sensores de fuerza.....	18
2.4.1.9.	Sensores de aceleración .....	18
2.4.2.	Fuente de alimentación.....	18
2.4.2.1.	Características fundamentales de las baterías .....	19
2.4.2.2.	Tipos de baterías .....	19
2.4.3.	Sistemas de control .....	21
2.4.3.1.	Circuito electrónico programable .....	21
2.4.3.2.	Uso del ordenador .....	21
2.4.4.	Motores .....	21
2.4.5.	Herramientas .....	22
2.5.	Sistemas embebidos .....	23
2.6.	Microprocesadores y Microcontroladores .....	24
2.6.1.	Comparación de capacidad de procesamiento .....	24
2.6.1.1.	Velocidad de procesamiento.....	24
2.6.1.2.	Capacidad de direccionamiento.....	24
2.6.1.3.	Tamaño de los datos .....	24

2.7.	Microcontroladores .....	25
2.7.1.	Componentes de un microcontrolador .....	26
2.7.1.1.	CPU .....	26
2.7.1.2.	Dispositivos de entrada y salida.....	26
2.7.1.3.	Memorias .....	27
2.7.1.4.	Memoria RAM (Random Access Memory).....	27
2.7.1.5.	Memoria ROM (Read Only Memory).....	27
2.7.1.6.	EPROM (Erasable Programmable Read-Only Memory) .....	27
2.7.1.7.	EEPROM (Electrical Erasable Programmable Read Only Memory) .....	27
2.7.1.8.	Flash.....	28
2.7.2.	Arquitecturas de los microcontroladores .....	28
2.7.2.1.	Arquitectura Von Neumann.....	28
2.7.2.2.	Arquitectura Harvard.....	29
2.7.2.3.	Clasificación según su juego de instrucciones .....	29
2.7.3.	Las interrupciones .....	30
2.7.4.	Tipos de microcontroladores.....	30
2.7.5.	Microcontroladores PIC .....	31
2.7.5.1	El núcleo.....	32
2.7.5.2	Los periféricos .....	33
2.7.5.3	Características especiales .....	33
2.7.6.	Microcontroladores AVR .....	34
2.7.6.1.	El Núcleo AVR .....	34
2.8.	Comunicaciones inalámbricas en robots. ....	35
2.8.1.	Comunicaciones por infrarrojos. ....	36
2.8.2.	Radiofrecuencia.....	37
2.8.3.	Bluetooth .....	38

2.9.	Comunicaciones de radiofrecuencia y control remoto .....	38
2.9.1.	Características generales de los controles remotos de radiofrecuencia .....	39
2.9.1.1.	Despertar al receptor .....	40
2.9.1.2.	Sincronización del receptor con el reloj de datos entrantes ..	40
2.9.1.3.	Identificación única y emparejamiento .....	40
2.9.2.	Estándares y protocolos de comunicaciones inalámbricas ...	41
2.9.2.1.	IEEE 802.15.4 .....	41
2.9.2.2.	ZigBee .....	42
2.9.2.3.	Protocolos de radiocontrol .....	44
2.9.2.4.	PWM - Modulación de ancho de pulso.....	44
2.9.2.5.	PPM - Modulación de la posición del pulso.....	46
2.9.2.6.	PCM - Modulación de código de pulso.....	47
2.9.2.7.	Protocolos serie .....	47
2.9.2.8.	Protocolos de Spektrum TX: DSM2 y DSMX .....	48
2.10.	Programación .....	48
2.10.1.	Arduino .....	49
CAPÍTULO 3: DISEÑO Y PROGRAMACIÓN DEL ROBOT MEGA SUMO ..		51
3.1.	Desarrollo del proyecto .....	51
3.2.	Principio de funcionamiento de un robot sumo .....	51
3.3.	Requerimientos del Diseño Mecánico del Robot Mega Sumo .....	52
3.4.	Componentes del robot Mega Sumo .....	53
3.4.1.	Motor Maxon RE Ø40 mm .....	53
3.4.2.	Baterías de Polímero de litio .....	54
3.4.3.	Sensor de reflectancia QTR-1A .....	55
3.4.4.	Sensores E3Z-d62 ONROM .....	55
3.4.5.	Modulo Receptor infrarrojo .....	57

3.4.6.	Radiocontrol .....	58
3.4.6.1	El transmisor FlySky .....	58
3.4.6.2	Modulación por desplazamiento de frecuencia Gausiana GFSK .....	60
3.4.6.3	Módulo de recepción .....	61
3.4.7.	Diseño estructural del Mega Sumo .....	62
3.5.	Sistema de control del robot y funcionamiento .....	63
3.5.1.	Arduino Nano.....	64
3.5.2.	Microcontrolador ATmega328P .....	67
3.5.3.	Placa controladora ArduPRO .....	68
3.5.4.	Sabertooth 2x32 .....	69
3.5.5.	Conexión del ArduPRO y Sabertooth en el robot Mega Sumo. .....	72
3.5.6.	Explicación del funcionamiento actual del Robot Mega Sumo .....	72
3.5.7.	Conexión de RC actual.....	73
3.6.	Diseño propuesto.....	74
3.6.1.	Microcontrolador ATmega32u4.....	75
3.6.2.	Optocoplador .....	75
3.6.3.	Puente H.....	76
3.6.4.	Simulación en Proteus del funcionamiento de un motor dc utilizando ATmega32u4 .....	78
3.6.5.	Diseño de la tarjeta electrónica.....	79
3.6.6.	Placa de circuito impreso y robot ensamblado con la tarjeta diseñada.....	85
3.6.7.	Explicación de la programación y funcionamiento del Robot Mega Sumo con la tarjeta de diseño propio. ....	87
3.6.8.	Modo Radiocontrol en la tarjeta propuesta.....	93

CAPÍTULO 4: CONCLUSIONES Y RECOMENDACIONES.....	95
4.1. Conclusiones.....	95
4.2. Recomendaciones.....	96
REFERENCIAS BIBLIOGRÁFICA .....	97
ANEXOS .....	101

## ÍNDICE DE FIGURAS

### CAPÍTULO 2: FUNDAMENTACIÓN TEÓRICA

Figura 2.1: Robot da Vinci del hospital Carlos Andrade Marín .....	7
Figura 2.2: Robot industrial de la compañía ABB.....	8
Figura 2.3: Sojourner en la superficie de Marte .....	8
Figura 2.4: Cye Robot de casa .....	9
Figura 2.5: Competencia de Sumo de la Universidad Tecnológica Nacional de Bahía Blanca, Argentina. ....	10
Figura 2.6: Robots industriales en procesos automotrices .....	11
Figura 2.7: Robot Humanoide realizado con el lego Nxt .....	12
Figura 2.8: Diferentes sensores de LEGO, sensores de tacto, luz y sonido. ....	13
Figura 2.9: Esquema de sensor de contacto.....	14
Figura 2.10: Esquema básico de un fototransistor. ....	15
Figura 2.11: Estrategia para detectar presencia de objetos con sensores ópticos .....	15
Figura 2.12: Principio de funcionamiento de un sensor de ultrasonido. ....	16
Figura 2.13: Esquema de un condensador eléctrico. ....	16
Figura 2.14: Esquema de un sensor capacitivo de proximidad .....	17
Figura 2.15: Efecto de inducción electromagnética en una bobina eléctrica .....	17
Figura 2.16: Esquema de funcionamiento de un sensor inductivo de proximidad.....	18
Figura 2.17: Baterías de Iones de Litio .....	20
Figura 2.18: Batería de Niquel Cadmio para luces de emergencia .....	20
Figura 2.19: Baterías AA de Niquel Metal de 2500 mAh .....	20
Figura 2.20: Motores de corriente Continua. ....	21
Figura 2. 21: Principio de funcionamiento de un motor de corriente continua. ....	22
Figura 2.22: Elementos de un robot manipulador. ....	23
Figura 2.23: Diagrama general de un sistema basado en un microcontrolador .....	26
Figura 2.24: Memorias EPROM .....	27
Figura 2.25: Arquitectura Von Neumann.....	28
Figura 2.26: Arquitectura Harvard.....	29

Figura 2.27: Microcontrolador PIC16F84, de la firma Microchip.....	30
Figura 2.28: ATmega8 .....	31
Figura 2.29: Microcontrolador PIC16F884887 .....	32
Figura 2.30: Diagrama de bloques del núcleo AVR .....	35
Figura 2.31: Modelo OSI, Zigbee Vs 802.15.4 .....	43
Figura 2.32: Modulación de una señal analógica en PWM .....	45
Figura 2.33: Pulso PWM.....	45
Figura 2.34: Modulación de una señal Analógica en PPM .....	46
Figura 2.35: Pulso PPM.....	47

### **CAPÍTULO 3: DISEÑO Y PROGRAMACIÓN DEL ROBOT MEGA SUMO**

Figura 3.1: Ring de batalla de robot Mega Sumo .....	52
Figura 3.2: Estructura del robot Mega Sumo.....	52
Figura 3.3: Motor Maxon RE 40 .....	53
Figura 3.4: Rango de operación del motor Maxon RE 40 .....	54
Figura 3.5: Sensor QTR-1A.....	55
Figura 3.6: Circuito del sensor QTR-1A.....	55
Figura 3.7: Dimensiones técnicas del sensor E3Z-d62 ONROM .....	56
Figura 3.8: Salida del circuito .....	57
Figura 3.9: Modulo receptor infrarrojo.....	57
Figura 3.10: Transmisor Flysky.....	58
Figura 3.11: Señal emitida con la palanca posición inicial .....	59
Figura 3.12: Señal emitida con la palanca posición superior .....	59
Figura 3.13: Señal emitida con la palanca posición inferior. ....	60
Figura 3.14: Modulación FSK.....	60
Figura 3.15: Principio del filtro gaussiano sobre la banda base. ....	61
Figura 3.16: Receptor Flysky.....	61
Figura 3.17: Dimensiones del robot Mega Sumo.....	62
Figura 3.18: Robot Mega sumo.....	63
Figura 3.19: Componentes del Nano Arduino .....	64
Figura 3.20: Diseño del Nano arduino .....	65
Figura 3.21: Pines de E/S del microcontrolador ATmega 328p.....	68
Figura 3.22: Esquema descriptivo de los elementos del robot conectados al ArduPRO.....	69



Figura 3.23: Conexiones de la placa Sabertooth 2x32 .....	70
Figura 3.24: Modo de operación en R/C .....	71
Figura 3.25: Modo de operación microcontrolador .....	71
Figura 3.26: ArduPRO + Sabertooth .....	72
Figura 3. 27: Algoritmo de programación actual .....	73
Figura 3.28: Esquema de protocolos utilizados en R/C .....	73
Figura 3.29: Conexión Sabertooth y el receptor RC.....	74
Figura 3.30: Disposición de los pines del ATmega 32U4 .....	75
Figura 3.31: Forma básica y circuito de aplicación de un optoacoplador típico .....	76
Figura 3.32: Diagrama de un puente H .....	77
Figura 3. 33: Diagrama de un puente H (Q1 y Q4).....	77
Figura 3. 34: Diagrama de un puente H (Q2 y Q3).....	78
Figura 3.35: Simulación del funcionamiento del motor dc.....	78
Figura 3. 36: Esquema general del diseño de la tarjeta .....	79
Figura 3.37: Diseño de la tarjeta controladora en Altium Designer.....	79
Figura 3.38: Bloque del MIC4684 .....	80
Figura 3. 39: Bloque de regulación en la tarjeta. ....	80
Figura 3.40: Esquema de conexión del ATmega32U4 .....	81
Figura 3.41: Microcontrolador ATmega32U4 en la tarjeta.....	81
Figura 3.42: Bloque de Driver de motor DC.....	82
Figura 3. 43: Bloque de driver en la tarjeta. ....	82
Figura 3.44: Bloque de optocopladores.....	83
Figura 3.45: Ubicación de Optocopladores en la tarjeta y puertos de conexión de sensores.....	83
Figura 3.46: Bloque LSM9DS1 .....	84
Figura 3.47: LSM9DS1 en la tarjeta .....	84
Figura 3.48: Bloque del receptor infrarrojo.....	85
Figura 3.49: Receptor infrarrojo en la tarjeta .....	85
Figura 3.50: Placa de circuito impreso.....	86
Figura 3. 51: Robot ensamblado con la tarjeta diseñada .....	86
Figura 3.52: Pines de control de motores. ....	87
Figura 3.53: Dirección de Motores .....	87
Figura 3.54: Pines de Entrada.....	88

Figura 3.55: Declaración de variables .....	88
Figura 3.56: Configuración de entradas y salidas.....	88
Figura 3.57: Baudrate & Set motors .....	89
Figura 3.58: Selección de estrategia.....	89
Figura 3.59: Espera para ejecutar la estrategia .....	89
Figura 3.60: Movimiento esquivo previo inicio de estrategia. ....	90
Figura 3.61: Búsqueda Aleatoria .....	91
Figura 3.62: Estrategia pájaro carpintero .....	92
Figura 3.63: Búsqueda Aleatoria pausada.....	92
Figura 3.64: Estrategia Tornado .....	93
Figura 3. 65: Modulo de recepción de RC de la tarjeta controladora. ....	94
Figura 3.66: Modo RC .....	94

## ÍNDICE DE TABLAS

### **CAPÍTULO 2: FUNDAMENTACIÓN TEÓRICA**

Tabla 2.1: Propiedades de IEEE 802.15.4 .....	42
---	----

### **CAPÍTULO 3: DISEÑO Y PROGRAMACIÓN DEL ROBOT MEGA SUMO**

Tabla 3.1 Datos técnicos Emisora .....	59
--	----

Tabla 3.2 Datos técnicos del receptor .....	61
---	----

## **RESUMEN**

En el presente trabajo de titulación se implementa un algoritmo de programación completo y preciso mediante una tarjeta de diseño propio, con la finalidad de controlar a un robot sumo autónomo y radio controlado. Para poder desarrollar este proyecto se procedió al estudio y análisis de los componentes de un kit robótico de mega sumo de gama alta, entre sus elementos como los motores y sensores, siendo primordial el estudio del microcontrolador ya que es el cerebro del robot y parte fundamental de su sistema de control. Se analizó también el controlador de motor, el cual se encarga de dar el movimiento al robot direccionando el sentido de rotación de sus motores. Se analizó la comunicación entre el hardware y software del robot en donde se estableció los protocolos que se utilizan, pudiendo así obtener la información necesaria para el diseño de la tarjeta, la cual va a estar programada con el algoritmo propuesto

**Palabras claves:** MICROCONTROLADOR, COMUNICACIONES INALÁMBRICAS, ROBÓTICA, DISEÑO ELECTRÓNICO, ALGORITMO.

# **CAPÍTULO 1: INTRODUCCIÓN**

## **1.1. Introducción.**

El mundo actual pasa por una era tecnológica que avanza a pasos agigantados. Así como la introducción de las computadoras personales cambió nuestras vidas, los robots han comenzado a revolucionar nuestra forma de vida. A medida que los sensores y la inteligencia artificial de los robots mejoran con el paso del tiempo, también lo hacen sus posibles usos.

Desde hace mucho tiempo, las personas han estado fascinadas con los robots, pero solo hasta ahora la tecnología y el conocimiento para construir robots baratos que puedan hacerse cargo de algunas de nuestras tareas diarias. La fascinación por la robótica es especialmente importante en los estudiantes de ingeniería de la facultad técnica del desarrollo en la Universidad Católica de Santiago de Guayaquil (UCSG), que sienten el impulso de ser parte de la revolución de los robots. Sin embargo, muchos de ellos se desaniman de diseñar y programar su propio robot porque creen que implementar incluso un robot simple requiere de mucha habilidad y de una estructura compleja.

En este trabajo de titulación se describe de manera general el desarrollo del proyecto de Implementación de algoritmos de programación y de protocolos de comunicación inalámbrica para la operatividad de un robot sumo autónomo y Radio controlado (RC).

## **1.2. Antecedentes**

La robótica es una disciplina en la cual varias universidades, institutos técnicos y otras instituciones de investigación están trabajando actualmente en proyectos que tienen como objetivo desarrollar un robot o partes de robots, muchos de los investigadores de dichas instituciones tienen experiencia en informática o ingeniería eléctrica siendo esta combinación realmente necesaria a la hora de diseñar e implementar un robot.

Los investigadores en un intento de diseñar y construir un robot, siempre se enfrentan a la necesidad de construir un sistema complejo el cual resulta difícil de controlar o un sistema simplificado el cual no es lo suficientemente bueno como para funcionar ante eventos inesperados del mundo real.

Los robots pueden ser autónomos o radio controlados (RC), los robots RC permiten ser controlados mediante el uso de comunicaciones inalámbricas a través de un dispositivo que transmite señales al sistema receptor del robot el cual se comunica con el controlador en donde se procesa y ejecuta la información recibida (movimientos o trabajo a realizar), mientras que los robots autónomos son dispositivos mecánicos que pueden moverse y tomar su propia decisión por sí mismos. La capacidad autónoma es alcanzada a través de la disponibilidad de sensores exteriores que pueden recibir información del entorno a través de la medición de distancia, imágenes visuales y medición de proximidad. Los sensores más comunes son los sensores de distancia (ultrasónicos, láser, etc.) capaces de medir la distancia al oponente y detectar al oponente. Cuando esta habilidad es utilizada por un robot de sumo inteligente, este debería ser capaz de ganar, empujando al oponente fuera del ring de la competencia de Sumo Robot.

### **1.3. Definición del Problema.**

En los concursos de robótica que se han realizado en el país y los cuales ha participado el club de robótica "ROBOFET" de la Facultad de Educación Técnica del Desarrollo (FETD) en la UCSG, se ha encontrado muchas falencias e inconformidades dentro del desenvolvimiento de los prototipos robóticos sumos de las múltiples instituciones que concursan, esto se debe a la simplicidad de los algoritmos y al sistema de control del robot (tarjeta controladora) que los ejecuta.

### **1.4. Justificación del Problema.**

La robótica ha sido una disciplina de mucho interés para la Facultad Técnica del Desarrollo de la Universidad Católica Santiago de Guayaquil, llegando a conformarse oficialmente el club de robótica "ROBOFET" en el año 2014, cuyo principal objetivo es el de representar de la mejor manera a la

universidad en competencias de robótica a niveles nacionales, como ya lo ha venido haciendo desde hace un par de años en el concurso ecuatoriano de robótica (CER).

Actualmente “ROBOFET”, ha participado en varias competencias nacionales como el IX CER 2013, X CER 2014, en las cuales se ha desempeñado de una manera eficiente alcanzando grandes reconocimientos en distintas categorías de competencia de robots, siendo una de ellas la de SUMOBOT.

El Club de robótica “ROBOFET”, dispone de diferentes tipos de robots siendo el Mega Sumo uno de los más importantes debido a que la competencia de estos robots acude una gran cantidad de espectadores y se demuestra el dominio de las instituciones académicas.

Para poder ganar una competencia internacional, el robot debe tener una buena estrategia sobre cómo vencer al oponente, además de poseer los componentes necesarios para hacerlo, como lo es una gran capacidad de su microcontrolador, una comunicación rápida y eficiente entre el robot y el radiocontrol, una buena programación, técnicas de respuesta, velocidad de detección de los sensores, y la fuerza de sus motores.

Para cumplir con los requisitos de modernización inteligente, estratégica y de diseño que son fundamentales para ganar este tipo de competencias, se desarrolla un algoritmo completo y preciso para un robot Mega Sumo robusto, autónomo y radio controlado mediante el uso de una tarjeta de diseño propio que permita ejecutar el algoritmo antes mencionado.

## **1.5. Objetivos del Problema de Investigación.**

### **1.5.1. Objetivo General.**

Implementar un algoritmo de programación junto con protocolos de comunicación inalámbrica en una tarjeta de diseño propio que permita a un robot sumo su funcionamiento autónomo y control de radio frecuencia.

### **1.5.2. Objetivos Específicos.**

- Definir el estado de arte y los conceptos fundamentales de un robot autónomo y radio controlado.
- Establecer el protocolo de comunicación inalámbrica para la transmisión de datos con el robot sumo.
- Desarrollar un algoritmo completo y preciso que permita controlar de manera autónoma al robot sumo de pelea.
- Diseñar una tarjeta electrónica que permita el control del sistema del robot por medio del algoritmo.

### **1.6. Hipótesis.**

La elaboración e implementación de un algoritmo más completo y preciso dentro de la tarjeta electrónica diseñada, generará un control inteligente del robot sumo, el cual aprovechará de mejor forma las características del mismo, mejorando su sistema y por ende su funcionalidad.

### **1.7. Metodología de Investigación.**

Exploratoria, debido a que se examina la robótica en general y su sistema de control, con el fin de familiarizarnos con dichos temas, se utiliza también una metodología descriptiva ya que se detalla el funcionamiento de los diferentes elementos y procesos que proporcionan la autonomía del robot Sumo para su posterior estudio y análisis con el fin de desarrollar un algoritmo completo y preciso.



## **CAPÍTULO 2: FUNDAMENTACIÓN TEÓRICA**

En este capítulo se describen los conceptos fundamentales de la robótica, los microcontroladores, y sus aplicaciones en la actualidad.

### **2.1. La robótica**

A través de los tiempos la humanidad se ha maravillado por máquinas y dispositivos capaces de realizar tareas y funciones propias de un ser humano, de cierta manera emulando su comportamiento y sus actividades. Estas máquinas autómatas fueron creadas con distintos propósitos como en el área laboral aumentando la productividad de empresas industriales, o en el área de la medicina siendo los cirujanos capaces de utilizar estos equipos para operaciones en donde se necesita seguridad y una precisión milimétrica como lo es el campo de la neurocirugía, también áreas como el entretenimiento en donde se usan robots para realizar actividades de recreación o competencias académicas de demostración y destreza.

La definición de robot va más allá de una simple máquina automática que realiza funciones y operaciones lógicas para facilitarnos o entretenernos la vida, un robot es un sofisticado instrumento de ingeniería que ofrece una gran variedad de aplicaciones en el mundo actual. (Reyes Cortés, 2011)

A diferencia no toda máquina puede ser denominada un robot, un robot se distingue de un sistema mecatrónico debido a su multifuncionalidad, es decir un sistema mecatrónico cumple solamente una única función como lo son las tostadoras, secadoras eléctricas, máquinas dispensadoras, esto se debe a que no ha sido diseñada para cumplir más funciones puesto que su estructura solo le permite realizar una sola función, en cambio un robot puede realizar múltiples funciones simplemente siendo reprogramado y siendo cambiada su herramienta con la cual realizaba tal acción, ya que su estructura mecánica se lo permite.

Hoy en día los robots se los puede encontrar en todas partes, como en nuestras casas realizando tareas domésticas como limpiar el polvo. Según

Reyes Cortés, (2011) La robótica es una disciplina científica en donde se investiga y desarrolla una clase particular de sistemas mecánicos denominados robots manipuladores diseñados para realizar una amplia variedad de aplicaciones industriales, científicas, domésticas y comerciales.

## **2.2. Aplicaciones actuales de la robótica**

### **2.2.1. Robótica Quirúrgica**

El uso de robots en este campo ofrece una gran variedad de ventajas debido a que el cirujano ya no debe ingresar sus manos dentro del cuerpo humano del paciente, gracias a pequeñas incisiones realizadas con gran precisión y el uso de instrumentos especializados que posee el robot lo cual evita traumas o lesiones y permite al médico cirujano operar al paciente con una mayor seguridad y menor riesgo dando como resultado una rápida recuperación después de la cirugía. En la figura 2.1 se observa el robot da Vinci en una operación junto al personal médico encargado de la cirugía.



Figura 2.1: Robot da Vinci del hospital Carlos Andrade Marín  
Fuente: (HCAM-IESS, 2017)

### **2.2.2. Robótica industrial y manufactura**

Esta clase de robots se especializan en la producción, pertenecen a empresas comercializadoras de productos en masa. Según la Organización Internacional de Estándares ISO un robot industrial es un manipulador multipropósitos, reprogramable y controlado automáticamente en tres o más ejes. (Reyes Cortés, 2011)

En la figura 2.2 se observa un robot de tipo industrial. Entre las principales funciones que cumplen estos tipos de robots se encuentran la de

traslado de herramientas, empaquetado, soldadura, y aplicaciones de pintura de carrocería.



Figura 2.2: Robot industrial de la compañía ABB  
Fuente: (Reyes Cortés, 2011)

### 2.2.3. Robótica Espacial

En el espacio el uso de robots disminuye el riesgo de los astronautas a realizar actividades que pongan en riesgo sus vidas, esto se debe a que los robots son construidos para resistir diferentes tipos de dificultades que se dan en el espacio exterior, como lo son las temperaturas extremas, estos robots pueden ser controlados remotamente o pueden ser autónomos. En la figura 2.3 se observa el robot Sojourner diseñado para movilizarse por la superficie de Marte. Entre las funciones de estos robots están las de realizar tareas repetitivas y tareas difíciles de realizar para un ser humano, como por ejemplo tomar muestras de terreno, montar estructuras, y posicionar instrumentos.

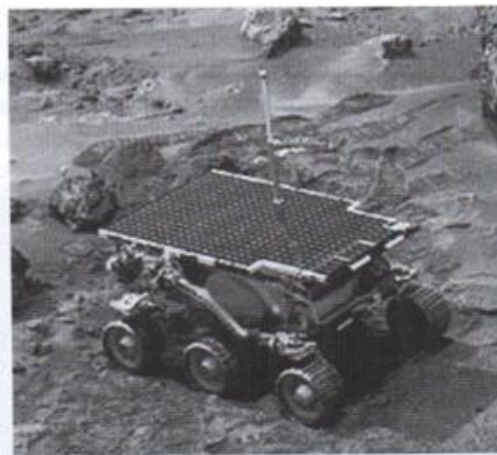


Figura 2.3: Sojourner en la superficie de Marte  
Fuente: (Vazques Fernandez-Pacheco, Ramos de la Flor, Fernández Rodríguez, Payo Gutierrez, & Adán Oliver, 2016)

#### **2.2.4. Robótica Domestica**

En la actualidad existen diversos tipos de robots que sirven para facilitar las tareas del hogar, estos robots cumplen funciones como limpiadores, organizadores, y asistentes personales, como el robot móvil Cyé figura 2.4, un robot de casa que aspira el polvo y hace entregas dentro del hogar



Figura 2.4: Cyé Robot de casa  
Fuente: (Siegwart & Nourbakhsh, 2004)

#### **2.2.5. Robótica Educativa**

El uso de la robótica en el ámbito educativo facilita el aprendizaje ya que se impulsa la creatividad del estudiante mediante el uso de diversas áreas de conocimiento como matemáticas, electrónica, física e informática para la elaboración de prototipos robóticos o dispositivos que solucionen problemas reales del entorno, lo que despierta el interés por parte del estudiante.

#### **2.2.6. Robótica Recreativa**

Otro de los grandes usos de los robots es en el área recreativa, existiendo un amplio espectro de aplicaciones en este campo como lo son las competencias o simplemente robots que entretenga y sirvan para quitar el estrés diario, como lo son los robots de Sumo, estos robots imitan un deporte de Japón denominado Sumo en el cual dos luchadores son ubicados en un espacio circular con límites definidos en donde quien logre sacar a su contrincante fuera de los límites del círculo será el ganador.

En la figura 2.5 se observan diferentes robots de sumo en el área de combate. Este deporte japonés ha sido la inspiración para la creación de este tipo de robots los cuales son usados para competencias entre entidades académicas, existen una gran variedad de reglas para estas competencias,

así como también limitaciones a la hora de crear un específico robot para cada categoría, siendo las categorías más conocidas las de micro, mini y mega sumo.

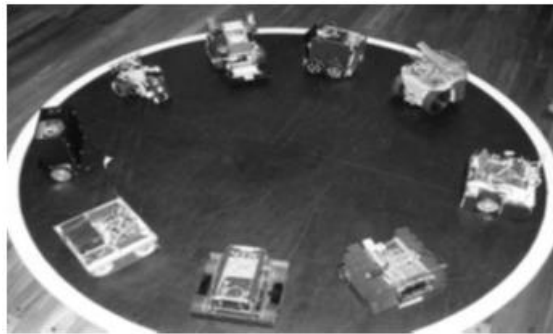


Figura 2.5: Competencia de Sumo de la Universidad Tecnológica Nacional de Bahía Blanca, Argentina.  
Fuente: (Zabala, 2007)

### **2.2.7. Robot Sumo**

Dos robots compiten en un combate mano a mano siguiendo el sistema básico de las competencias de sumo humano tradicional. El único propósito es una competencia de empuje entre los dos robots para forzar al otro a salir de la arena. Se permiten múltiples clases de peso y sistemas de control, los autónomos compiten contra autónomos y RC contra RC

#### **2.2.7.1. Tipos de robots sumo**

Hay dos tipos de robots en Robot Sumo: autónomo y controlado por radio (RC). Las reglas para la determinación del robot ganador son las mismas, pero los métodos de control de los robots son diferentes. Un robot de sumo autónomo es aquel que contiene microcontroladores, sensores y otros dispositivos que permiten que un robot se controle a sí mismo, mientras que un robot de sumo radio controlado utiliza un sistema de control proporcional que permite a las personas controlar el robot de forma remota.

### **2.3. Morfología de los robots**

La gran variedad de robots que existen actualmente crean la necesidad de clasificarlos según su, función, estructura e incluso aplicación generalmente se clasifican de la siguiente manera: Robots Industriales, Robots Móviles, y Humanoides.

### 2.3.1. Robots Industriales

Estos robots cumplen funciones industriales debido a que la gran mayoría fundamentalmente son brazos mecánicos, denominados de tal manera debido a su gran parecido con el brazo humano, estos robots pueden trabajar todo el día, lo cual lo hace adecuado para realizar tareas o procesos repetitivos en grandes empresas que buscan la automatización de sus fábricas como lo es el caso de la industria automotriz en donde son utilizados para el ensamblaje, pintado, soldadura de las piezas del automotor, primordialmente estos robots generan un incremento en la productividad de las empresas. En la figura 2.6 se observan robots industriales cumpliendo funciones en el área automotriz.



Figura 2.6: Robots industriales en procesos automotrices  
Fuente: (Reyes Cortés, 2011)

### 2.3.2. Robots Móviles

Las aplicaciones que se les dan a estos tipos de robots van de la mano con su capacidad de desplazamiento en su tipo de terreno, ya sea este terrestre, aéreo o marítimo. Estos robots por lo general son usados de diferentes maneras y para distintos propósitos ya sean estos para entretenimiento, trabajo o uso militar. Los robots móviles terrestres tienen usos por lo general en la vida diaria para trasladar objetos gracias a su capacidad propia de rastreo y su facilidad de desplazarse mediante ruedas o patas, lo que permite un uso más extendido de sus habilidades en lugares

como hospitales, hogares e incluso en fábricas, dando también uso a estos tipos de robots en competencias académicas o actividades de entretenimiento, como robots mascotas o bien para la demostración de las habilidades que posee un robot en exposiciones al público. Los robots móviles marítimos y aéreos generalmente son usados para actividades de mayor importancia como uso militar o en beneficio de la sociedad como el reconocimiento del terreno, toma de fotografías aéreas o marítimas para rescate, localización y en su gran mayoría son teleoperados. (Reyes Cortés, 2011)

### **2.3.3. Robots Humanoides**

Estos robots se identifican por la gran similitud que tienen al parecerse al ser humano siendo agradables a la vista además de ser capaces de hablar, caminar e imitar movimientos propios de los seres humanos, como se observa en la figura 2.7 un robot humanoide de la firma LEGO, estos tipos de robots poseen una complejidad mucho más alta en lo que es la programación del robot, sus motores y sensores pues realizan diferentes tipos de actividades desde tocar un piano e incluso realizar actividades de rescate, o ser asistentes de personas con discapacidad física, esta versatilidad es lo que los hace tan complejos y sofisticados.



Figura 2.7: Robot Humanoide realizado con el lego Nxt

Fuente: (Zabala, 2007)

## 2.4. Elementos de un sistema robótico

Entre los elementos mecánicos más importantes de un robot se encuentra su estructura ya que esta da soporte a los otros elementos que lo conforman, estos pueden poseer diferentes tipos de estructuras de acuerdo al fin para el que estén programados ya sean para ser usados en fábricas como los robots manipuladores o robots móviles para traslado o seguimiento y control. Todos ellos se componen principalmente por los siguientes elementos: Sensores, fuente de alimentación, motores, sistema de control, herramientas.

### 2.4.1. Sensores

Los sensores son de suma importancia para un robot móvil ya que por medio de estos obtiene información de su entorno o de su propio sistema realizando mediciones ya sean estas de temperatura, proximidad o ubicación, por lo que le permite reaccionar de acuerdo a cada ocasión inesperada de la forma para la cual fue programado. Estos sensores se clasifican de 2 formas, sensores que miden el sistema interno del robot (temperatura interna, velocidad de los motores, nivel de carga) y sensores que obtienen información del exterior es decir del ambiente que los rodea (mediciones de distancia, temperatura ambiental, colores o sonidos).



Figura 2.8: Diferentes sensores de LEGO, sensores de tacto, luz y sonido.  
Fuente: (Zabala, 2007)

Estos tipos de sensores que obtienen información del entorno que los rodea en su gran mayoría son sensores para medir distancias, y se clasifican de acuerdo a las magnitudes físicas que emplean para obtener información. Además de ser incluso capaces de detectar y predecir los movimientos de los objetos, en su mayoría usados por su bajo costo en el mundo de la robótica. En la figura 2.8 se observan algunos de los tipos de sensores de marca LEGO.



#### 2.4.1.1. Sensores de posición

Detectan y determinan la posición del robot, mediante el uso de un giroscopio u otro dispositivo que ubique el espacio en que se encuentre.

#### 2.4.1.2. Sensores de efecto hall

Son sensores que utilizan imanes que cambian su conductividad al encontrarse con un objeto metálico.

#### 2.4.1.3. Sensores de luz o brillo

Detectan la intensidad de luz que llega al sensor, permitiéndole incluso detectar colores.

#### 2.4.1.4. Sensores de contacto

El sensor táctil permite detectar el instante en el cual es activado, permitiendo así programarlo para que ejecute una acción en cualquiera de sus fases: presionado, lanzado o en contacto. Los sensores táctiles se emplean normalmente en robots móviles para proporcionar una indicación de colisión de último momento con alguna obstrucción del entorno, como su nombre mismo lo indica el proceso de detección implica contacto físico entre el sensor y el objeto de interés. (Everett, 1995)

En la figura 2.9 se aprecia el esquema general de los sensores de contacto.

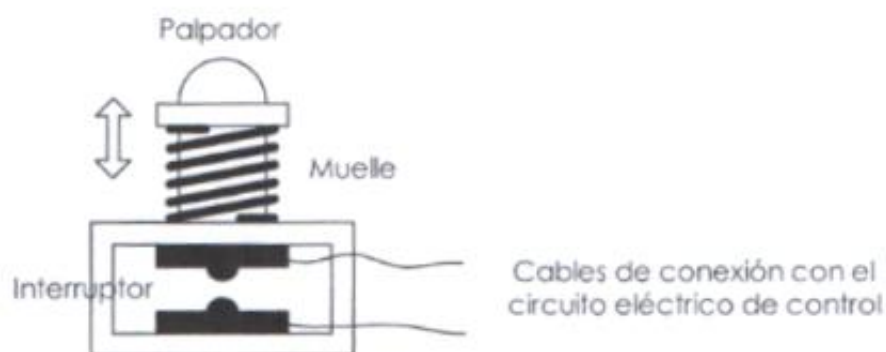


Figura 2.9: Esquema de sensor de contacto  
Fuente: (Vazques Fernandez-Pacheco et al., 2016)

#### 2.4.1.5. Sensores ópticos o de luz

Mide la intensidad luminosa, y está constituido generalmente por fotodiodos, fototransistores y fotorresistencias, ya que dependiendo de la cantidad de luz que detecten generan corrientes eléctricas proporcionales a dicha cantidad, como se observa en la figura 2.10, la corriente de colector de un fototransistor,  $I_c$ , es proporcionada a la intensidad luminosa,  $I_v$ ,

$$I_c = K * I_v$$

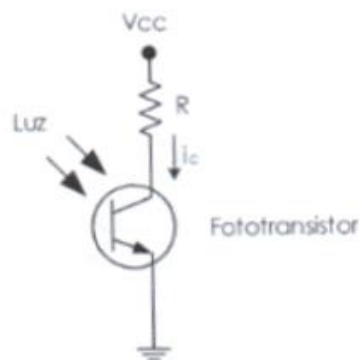


Figura 2.10: Esquema básico de un fototransistor.  
Fuente: (Vazques Fernandez-Pacheco et al., 2016)

Este tipo de sensor consiste en un transmisor que ilumina un objetivo con un haz y un receptor capaz de detectar el componente de la luz que es esencialmente reflejado tal y como se observa en la figura 2.11, estos dispositivos producen un rango estimado en función del tiempo necesario para que la luz alcance el objetivo y regrese. (Siegwart & Nourbakhsh, 2004)

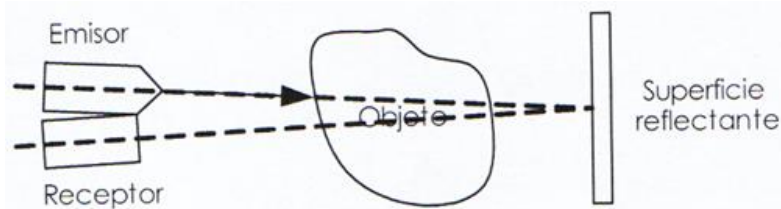


Figura 2.11: Estrategia para detectar presencia de objetos con sensores ópticos  
Fuente: (Vazques Fernandez-Pacheco et al., 2016)

#### 2.4.1.6. Sensores ultrasónicos

Su principio básico es el de usar ondas de sonido (ultrasónicas) para medir el tiempo que toma esa onda en reflejarse con un objeto y regresar al receptor para así calcular la distancia existente entre ambos, se observa este principio de funcionamiento en la figura 2.12. Este principio conocido como

ecolocalización es utilizado en el reino animal por murciélagos, en la robótica por lo general se emite una onda ultrasónica por encima de los 20 KHZ.

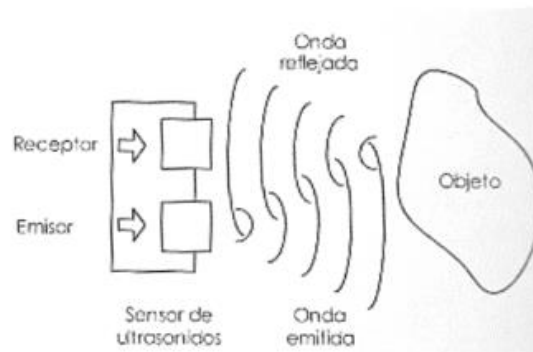


Figura 2.12: Principio de funcionamiento de un sensor de ultrasonido.  
Fuente: (Vazques Fernandez-Pacheco et al., 2016)

#### 2.4.1.7. Sensores de proximidad

Son sensores utilizados para detectar objetos alrededor del entorno del robot sin que exista ningún contacto.

##### a) Sensores capacitivos de proximidad

Su principio de funcionamiento es la de un condensador. Como se observa en la figura 2.13, un condensador se compone por dos placas metálicas, separadas entre sí por un dieléctrico. La carga que puede almacenar el condensador depende del material del dieléctrico de su interior, principio que es utilizado para medir proximidades con objetos. (Vazques Fernandez-Pacheco et al., 2016)

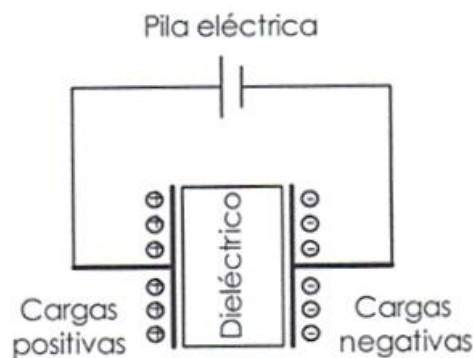


Figura 2.13: Esquema de un condensador eléctrico.  
Fuente: (Vazques Fernandez-Pacheco et al., 2016)

El esquema de funcionamiento se muestra en la figura 2.14, cuando un objeto se acerca, la carga del sensor (condensador) se ve alterada. El

condensador funciona junto a un circuito electrónico que detecta la variación de carga y por lo tanto detecta al objeto próximo al sensor. Entre más conductivo sea el objeto (mayor constante dieléctrica) este se polarizará mejor, por lo que acumulará más carga y será detectado más fácilmente.

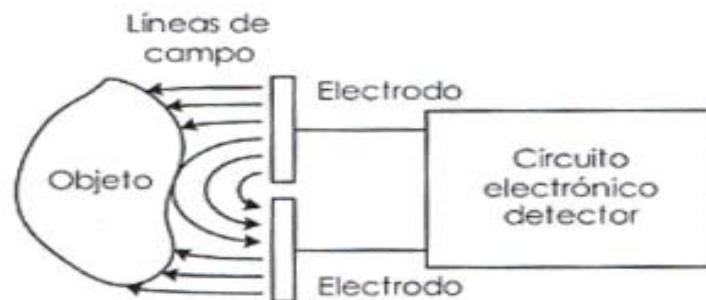


Figura 2.14: Esquema de un sensor capacitivo de proximidad  
Fuente: (Vazques Fernandez-Pacheco et al., 2016)

Estos sensores detectan objetos cercanos de cualquier naturaleza (Conductivo o no conductivo), en distancias pequeñas del orden de milímetros.

#### **b) Sensor inductivo de proximidad**

El principio de funcionamiento de estos sensores es el efecto de inducción electromagnética que ocurre en las bobinas como se ve en la figura 2.15, cuando una bobina se expone a un campo magnético variable, genera una tensión que produce una corriente eléctrica inducida en la bobina.

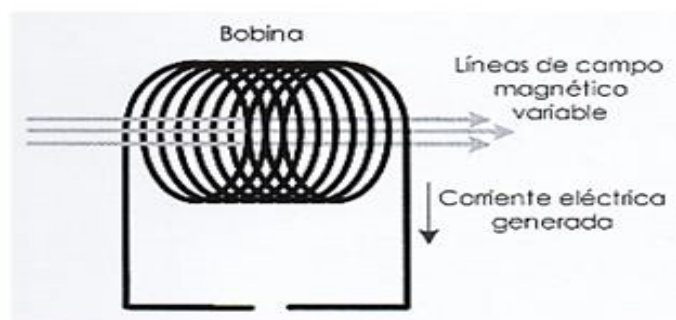


Figura 2.15: Efecto de inducción electromagnética en una bobina eléctrica  
Fuente: (Vazques Fernandez-Pacheco et al., 2016)

En la figura 2.16 se observa que el sensor posee una bobina y dentro de ella se encuentra un imán que genera un campo magnético, en el cual cuando se acerca un objeto magnético, el campo magnético se altera debido al efecto

de inducción, produciendo una corriente eléctrica que circula por la bobina. Esta corriente es detectada por un circuito electrónico el cual nos alerta la proximidad del objeto.

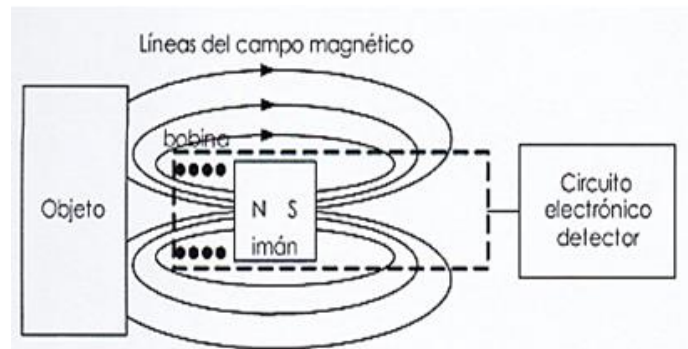


Figura 2.16: Esquema de funcionamiento de un sensor inductivo de proximidad  
Fuente: (Vazques Fernandez-Pacheco et al., 2016)

#### 2.4.1.8. Sensores de fuerza

Estos sensores se utilizan para determinar las fuerzas y momentos con la que el extremo del robot interactúa con un objeto. Su principio de funcionamiento se basa en la deformación que sufre cierto material cuando se le aplican fuerzas o momentos determinados. Esta deformación es medida por sensores piezoresistivos.

#### 2.4.1.9. Sensores de aceleración

Miden la aceleración del cuerpo donde van colocados, sirven para saber cómo se está moviendo el robot o la aceleración de un objeto cuando es movido por un robot. Utilizan materiales piezoeléctricos.

#### 2.4.2. Fuente de alimentación

El suministro de energía eléctrica es de suma importancia para los robots ya que son máquinas electromagnéticas que poseen un sistema de control, además de darle energía a sus motores eléctricos o electromagnéticos según el caso. Por lo general los robots móviles poseen una fuente de alimentación mediante el uso de baterías ya que estar conectados a un cable no es muy beneficioso para este tipo de robots.

#### 2.4.2.1. Características fundamentales de las baterías

- **Voltaje Nominal:** Es el valor de voltaje más estable que tendrá una batería cuando se descargue o cuando se comience a descargar.
- **Capacidad:** El tamaño de la batería y su capacidad son directamente proporcionales.
- **Densidad de energía:** Es la relación de la capacidad por unidad de volumen.
- **Curva de descarga:** La curva muestra el voltaje de la batería en relación con su tiempo de consumo.
- **Resistencia interna:** Es la resistencia al paso de corriente que posee una batería.
- **Capacidad de recarga:** Es la capacidad de la batería de volver a cargarse y también cuantas veces se puede repetir ese proceso.
- **Efecto memoria:** En las baterías de **Níquel cadmio** cuando una batería de este tipo no se descarga en forma completa antes de ser recargada, tiende a recordar este nivel de descarga como el nivel 0. Por lo tanto, la próxima vez considerará que está descargada cuando llegue a este punto. (Zabala, 2007)
- **Costo:** Se ha masificado la producción de baterías debido a la gran cantidad de dispositivos electrónicos que las utilizan, por lo que actualmente se puede encontrar una gran variedad de calidad y precio.

#### 2.4.2.2. Tipos de baterías

##### - **Batería de Iones de Litio (Li-Ion)**

Poseen una elevada densidad de energía y, en general, se presentan en placas rectangulares de menos de 5mm de espesor. Tienen un voltaje de 3,7 V, y carecen de efecto memoria, su tasa de autodescarga es baja. (Zabala, 2007)

Su vida útil es de tres años. Una vez cargada si se la almacena y se guarda en estado de reposo llega a perder sólo un 6% de su carga cada mes. En la figura 2.17 se observan baterías de Iones de Litio.



Figura 2.17: Baterías de Iones de Litio  
Fuente:(Zabala, 2007)

#### - **Baterías de Níquel Cadmio**

Fueron las primeras baterías recargables, su efecto memoria y la contaminación que generaban por el cadmio fueron las principales razones por las que dejaron de ser tan utilizadas y empezaron a ser reemplazadas por las de Níquel Metal, a pesar de que su costo era de tan solo una cuarta parte de lo que vale una de Níquel Metal. En la figura 2.18 se observa una batería de Níquel Cadmio, una ventaja de estas baterías es su cantidad de ciclos de carga, aproximadamente 1.500 ciclos.



Figura 2.18: Batería de Níquel Cadmio para luces de emergencia  
Fuente:(Zabala, 2007)

#### - **Baterías de Níquel Metal**

Son las baterías más utilizadas actualmente, poseen un 40% más de capacidad que las de Ni-Cd. La figura 2.19 se observa una batería de Níquel Metal, estas baterías no se ven afectadas por el efecto memoria, y poseen baja resistencia interna, lo que facilita una carga más rápida y una baja tasa de auto-descarga.



Figura 2.19: Baterías AA de Níquel Metal de 2500 mAh  
Fuente:(Zabala, 2007)

### **2.4.3. Sistemas de control**

Son los encargados de tomar decisiones y ejecutar acciones para que las realicen los motores o actuadores ya que ellos analizan la información que envían los sensores. Estos sistemas se clasifican de 2 maneras:

#### **2.4.3.1. Circuito electrónico programable**

En este sistema de control no se necesita un cable de conexión conectada a un computador, ya que el circuito es programable lo que permite la construcción de robots que desempeñen un trabajo que no requiera permanecer en un lugar fijo.

#### **2.4.3.2. Uso del ordenador**

En este sistema de control se necesita el uso de cables conectados al ordenador por lo que se utilizan en máquinas que no requieran realizar desplazamientos o que permanezcan en un solo punto fijo.

### **2.4.4. Motores**

Son los que proporcionan la movilidad del robot, entre los más conocidos los motores de corriente continua, como se observa en la figura 2.20. En estos motores no se requiere de demasiada exactitud en su giro ya que solo se requiere que generen rotación de una manera continua, otro ejemplo son los servomotores cuya principal característica es que se pueden controlar de una manera precisa su posición, velocidad y movimiento, además de poseer un par de torque mayor que los motores de corriente continua.



Figura 2.20: Motores de corriente Continua.  
Fuente: (Zabala, 2007)

Un motor eléctrico convierte energía eléctrica en mecánica, el motor está conformado por una armadura que gira en un campo magnético, esta armadura posee una bobina de cable arrollada alrededor de un núcleo de



hierro, la fuente de electricidad se conecta a las escobillas que hacen contacto con el conmutador de la armadura. El conmutador es un tipo de interruptor que cambia la dirección del flujo de corriente de la bobina según va girando, la corriente eléctrica fluye de la fuente al motor y otra vez a la fuente en un solo sentido, los cables de la bobina que transportan la corriente, experimentan una fuerza perpendicular en presencia del campo magnético, como se observa en la figura 2.21.

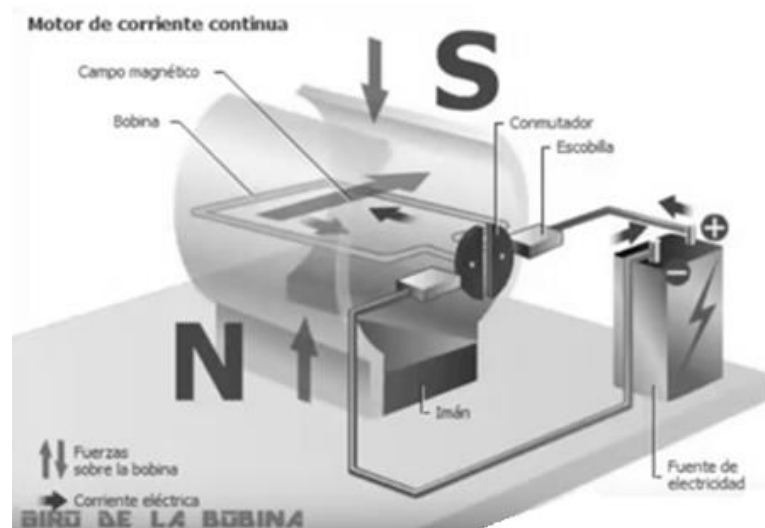


Figura 2. 21: Principio de funcionamiento de un motor de corriente continua.  
Fuente: ("Electricidad Práctica", 2018)

#### 2.4.5. Herramientas

Algunos robots como por ejemplo los robots manipuladores poseen elementos o herramientas necesarias para realizar un determinado trabajo y estos van montados a ellos sin que necesariamente sean parte esencial de su estructura, como por ejemplo las pinzas que se utilizan para manipular y transportar objetos dentro de un proceso de fabricación.

En la figura 2.22 se observa los elementos que constituyen a un robot manipulador.

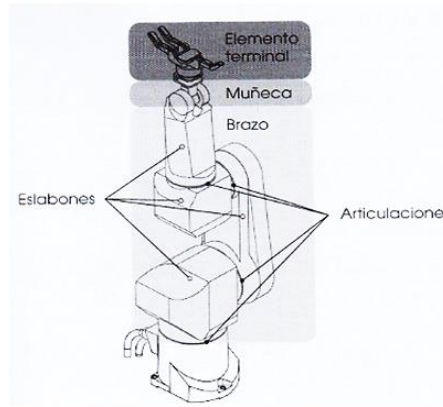


Figura 2.22: Elementos de un robot manipulador.  
Fuente: (Vazques Fernandez-Pacheco et al., 2016)

## 2.5. Sistemas embebidos

Un sistema embebido es básicamente un circuito electrónico digital diseñado para realizar un trabajo específico mediante operaciones de computación en tiempo real para un determinado producto. Los sistemas embebidos no son equivalentes a los sistemas de cómputo usados en ordenadores, ya que los sistemas embebidos tienen recursos limitados y aplicaciones específicas que los hacen eficientes en múltiples ambientes como en teléfonos móviles, alarmas de casas, lavadoras. (Salas Arriarán, 2017)

Un sistema embebido se compone principalmente por su microprocesador el cual ejecuta instrucciones a una velocidad la cual es definida por una señal de reloj. La potencia de procesamiento es medida en MIPS (Millones de instrucciones por segundo) y es definida por la capacidad del microprocesador, sus recursos internos y su máxima frecuencia de operación.

Los lenguajes de programación de un sistema embebido son por lo general C++, Basic, o ensamblador. Una de las ventajas de aplicar los sistemas embebidos en dispositivos electrónicos es su bajo consumo de energía ya que muchos de estos dispositivos utilizan baterías, como por ejemplo los robots móviles, juguetes, etc.

Un microcontrolador es un sistema embebido y actualmente se puede encontrar microcontroladores de un solo chip, los cuales son sistemas computarizados completos ya que contienen en su interior un microprocesador, unidades de memoria (de programas y datos), unidades de entrada y salida. (Salas Arriarán, 2017)

## **2.6. Microprocesadores y Microcontroladores**

Las funciones que cumplen los microprocesadores y microcontroladores son diferentes y esto se debe a sus características y los fines por los cuales fueron diseñados, según Santiago Espinosa, (2012) un microprocesador básicamente contiene una CPU (Unidad Central de procesamiento), mientras que un microcontrolador además de la CPU contiene memoria, temporizadores, interrupciones y otros recursos útiles para el desarrollo de aplicaciones, todos estos elementos en un circuito integrado. La principal diferencia entre un microcontrolador y un microprocesador es que el microcontrolador posee más recursos que un microprocesador, pero la capacidad de procesamiento de su CPU es limitada.

### **2.6.1. Comparación de capacidad de procesamiento**

#### **2.6.1.1. Velocidad de procesamiento.**

La velocidad de procesamiento de un microcontrolador será mucho menor que la de un microprocesador, siendo la frecuencia máxima de un microprocesador en orden de los GHz mientras que el microcontrolador en orden de MHz.

#### **2.6.1.2. Capacidad de direccionamiento**

Un microcontrolador promedio dispone de 8 Kbyte para instrucciones y 1 Kbyte para datos, en cambio los microprocesadores modernos son capaces de direccionar hasta 1 Terabyte de información (Santiago Espinosa, 2012).

#### **2.6.1.3. Tamaño de los datos**

Los microcontroladores más utilizados son los de 8 bits, mientras que los microprocesadores trabajan con datos de 32 o 64 bits. El microprocesador es utilizado como la CPU de un ordenador, un ordenador se puede definir como

un sistema de propósito general, de alto procesamiento que puede ejecutar cualquier tarea que se le solicite por programación, en cambio los microcontroladores son sistemas de propósito específico, es decir diseñados con una funcionalidad única, la cual va a ser la misma durante el resto de su vida útil, como por ejemplo los microondas o equipos de sonido; sin embargo, estas limitaciones.

En un microcontrolador se encuentran los componentes necesarios para ser considerado como una computadora en un circuito integrado, cuya capacidad de procesamiento es limitado, pero extremadamente eficiente en aplicaciones de propósito específico que no requieran un alto nivel de procesamiento, como por ejemplo el procesamiento de video.

## **2.7. Microcontroladores**

Los microcontroladores son la parte fundamental de cualquier sistema robótico, según Castaño Welgos,(2002) los microcontroladores son una pequeña computadora inyectada en un chip, diseñadas primordialmente para el control industrial. Estos dispositivos son usados también en la robótica por su bajo costo, reducción de espacio e inmunidad al ruido.

Un microcontrolador fundamentalmente es un circuito integrado cuya principal característica es su programación, siendo muy útil para aplicaciones de control de un sistema, permitiéndole ejecutar un sinnúmero de tareas y procesos. Un microcontrolador puede definirse entonces como un sistema completo de cómputo, que incluye una CPU, memoria RAM y ROM, un oscilador y puertos I/O, todo dentro de un mismo circuito integrado.(Castaño Welgos, 2002)

En la figura 2.23 se observan los principales dispositivos de entrada y salida que conlleva un microcontrolador. El microcontrolador es un componente electrónico importante debido a su versatilidad, ya que puede manejar señales tanto de entrada como de salida además de poseer la capacidad de procesamiento de datos y toma de decisiones.

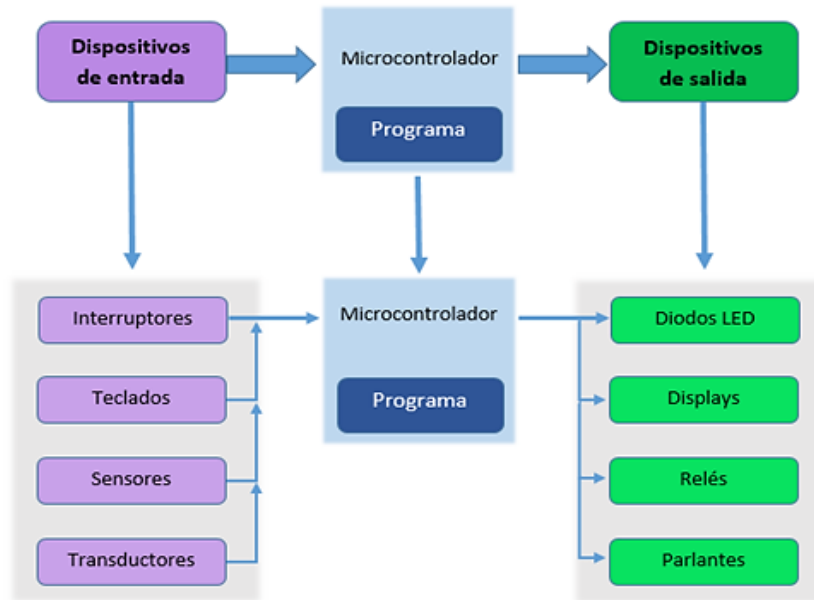


Figura 2.23: Diagrama general de un sistema basado en un microcontrolador  
Elaborado por: Autor

## 2.7.1. Componentes de un microcontrolador

### 2.7.1.1. CPU

Es el encargado de interpretar las instrucciones que llegan en forma de código y efectuar las operaciones correspondientes según el programa

### 2.7.1.2. Dispositivos de entrada y salida

Existen una gran cantidad de dispositivos que pueden ser parte de un sistema conformado por un microcontrolador, estos dispositivos se utilizan para el intercambio de información y el control de medio externo, estos dispositivos pueden ser de entrada o salida, cuando se refieren a dispositivos de entrada son aquellos componentes que al cambiar de estado envían una señal y esta es usada por el microcontrolador para la toma de decisiones, ejemplos simples de estos dispositivos son los sensores e interruptores. Ahora, cuando se refieren a dispositivos de salida estos son todos aquellos que generan una acción ante la decisión del microcontrolador como por ejemplo focos, display, parlantes, zumbadores o la activación de un motor.

### 2.7.1.3. Memorias

Las memorias son dispositivos capaces de almacenar datos informáticos durante un tiempo determinado según el tipo de memoria utilizado, en la figura 2.24 se observa un tipo de memoria conocida como EPROM.



Figura 2.24: Memorias EPROM

Fuente: (Benchimol, 2011)

### 2.7.1.4. Memoria RAM (Random Access Memory).

Memoria de acceso aleatorio, estas memorias de almacenamiento temporal de información, al desconectarse su suministro de energía se pierde la información que contenía su función es la de guardar el resultado de las operaciones y los datos que se generan en el programa del microcontrolador

### 2.7.1.5. Memoria ROM (Read Only Memory).

Memoria de solo lectura, estas memorias, son utilizadas para guardar información o programas de una manera permanente, esto quiere decir que esta información no se va a perder si se desconecta su alimentación, este tipo de memoria normalmente no es mayor a 4 kilobytes. Tipos de ROM:

### 2.7.1.6. EPROM (Erasable Programmable Read-Only Memory)

ROM programable borrable, estas memorias permiten que la información que ya había sido respectivamente grabada se borre y se pueda volver a reescribir nueva información. Esto se logra aplicando una luz ultravioleta sobre una ventana transparente propia de la memoria.

### 2.7.1.7. EEPROM (Electrical Erasable Programmable Read Only Memory)

ROM programable y borrable eléctricamente, este tipo de memoria permite almacenar o borrar un programa mediante el uso de corriente eléctrica, este proceso es mucho más rápido que borrar con luz ultravioleta en las EPROM.

### 2.7.1.8. Flash

Es una memoria que puede soportar un mayor número de borrado y almacenamiento de información, es decir son reprogramables, además de ofrecer una mayor velocidad. Este tipo de principio es muy aplicado en las memorias USB.

## 2.7.2. Arquitecturas de los microcontroladores

### 2.7.2.1. Arquitectura Von Neumann

Esta arquitectura se caracteriza por que la CPU y la memoria de datos y programa se comunican mediante un bus de datos por donde las instrucciones son enviadas, lo que conlleva a una disminución en la velocidad cuando se incrementaba la cantidad de datos a procesar ya que se generaba un cuello de botella.

Como se observa en la figura 2.25, el ancho de bus de datos es de 8 bits, por donde se trasladan datos e instrucciones, la desventaja es que el ancho de los datos limita el ancho de las instrucciones, en donde las instrucciones con más de 8 bits son enviadas en varias partes, por lo que se tenía como consecuencia un sistema lento, además de que se desconocía cuánta memoria de programa usaba, ya que dependía del ancho en byte de las instrucciones, lo cual resultaba muy variable. (Benchimol, 2011)

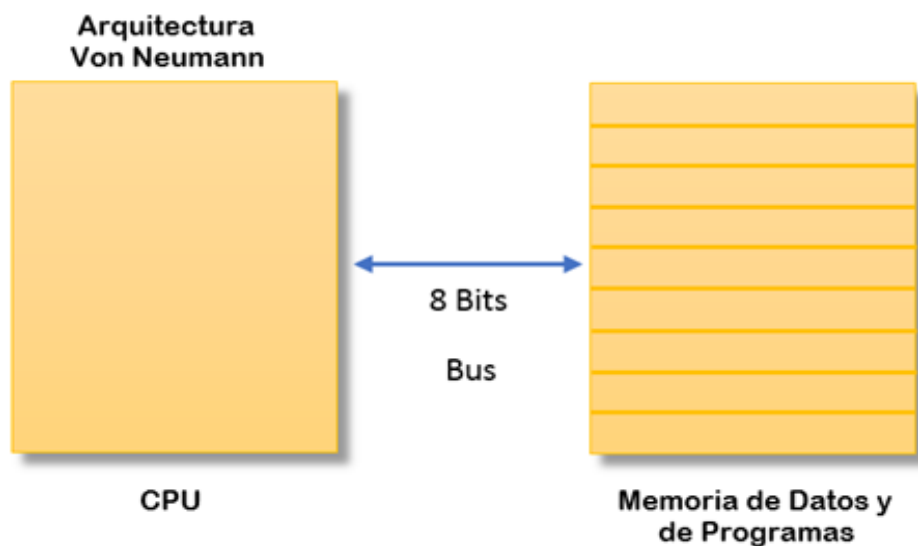


Figura 2.25: Arquitectura Von Neumann  
Elaborado por: Autor

### 2.7.2.2. Arquitectura Harvard

Este tipo de arquitectura reemplazo a la de Von Neumann ya que resolvía los problemas de velocidad de procesamiento, como se observa en la figura 2.26, esta arquitectura conectó la CPU hacia su memoria mediante dos buses distintos, uno de datos y otro de instrucciones, haciendo que se puedan recibir instrucciones por caminos diferentes ya que el bus de instrucciones es independiente del de datos, dando como resultado un sistema mucho más rápido, (Benchimol, 2011)

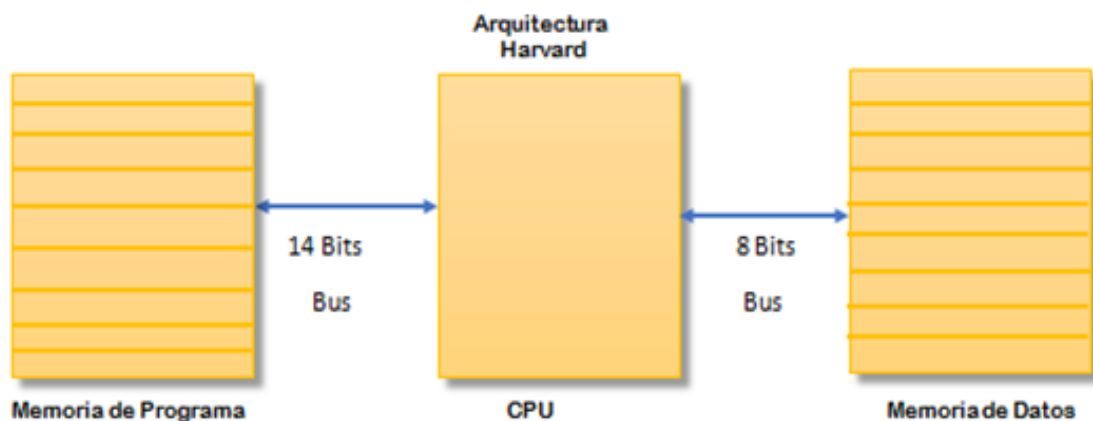


Figura 2.26: Arquitectura Harvard  
Elaborado por: Autor

### 2.7.2.3. Clasificación según su juego de instrucciones

Se entiende por juego de instrucciones al conjunto completo de instrucciones que posee un procesador, al igual que cada idioma tiene su diccionario de palabras con las que construir frases, cada familia de procesadores tiene su juego de instrucciones con la que construir programas. (Vazques Fernandez-Pacheco et al., 2016)

#### - RISC (Reduced Instruction Set Computer)

Conjunto de instrucciones de computo reducido, se utiliza en dispositivos portátiles debido a su eficiencia energética. Por ejemplo, Smartphone, videojuegos portátiles. RISC es un tipo de arquitectura de microprocesador que utiliza un conjunto de instrucciones altamente optimizado, es decir instrucciones simples permitiéndole una alta velocidad de ejecución y un consumo reducido de energía.



El poco espacio que requiere y el poco consumo de energía de este dispositivo lo hace perfecto para los robots móviles ya que el tamaño del robot es muy limitado y se necesita que funcione por un buen periodo de tiempo, por lo que son más utilizados en el mundo de la robótica.

- **CISC (Complex Instruction Set Computer)**

Conjunto de instrucciones de cómputo complejo, contiene un diccionario de instrucciones grande con instrucciones muy variadas o como en algunos casos muchas instrucciones simples, esto hace que el procesador sea complejo y necesite un consumo alto, pero reduce el código del programa. (Vazquez Fernandez-Pacheco et al., 2016)

### **2.7.3. Las interrupciones**

La interrupción es un sistema que provoca un salto en una subrutina, pero disparada por un evento del hardware, cuando la interrupción ocurre, el procesador abandona el programa que está ejecutando y pasa a procesar una rutina que se encuentra a partir de una posición de memoria fija. Durante la interrupción se guarda el contenido interno de los registros más importantes del procesador, de modo que luego, al volver al programa que se estaba ejecutando, se prosiga sin alteraciones. (Benchimol, 2011)

### **2.7.4. Tipos de microcontroladores**

Entre los microcontroladores más utilizados en el mundo de la robótica se encuentran los de la familia PIC de la firma Microchip debido a su bajo costo y eficiencia, en la figura 2.27 se observa un microcontrolador de la firma microchip. El microcontrolador más utilizado para la construcción de robots es el PIC16F84. Posee una memoria de programa Flash de 1 KB con palabras de 14 bits, una memoria RAM de datos de 68 bytes y una EEPROM de 64 bytes, y 13 pines de E/S. (Zabala, 2007)



Figura 2.27: Microcontrolador PIC16F84, de la firma Microchip.  
Fuente: (Zabala, 2007)

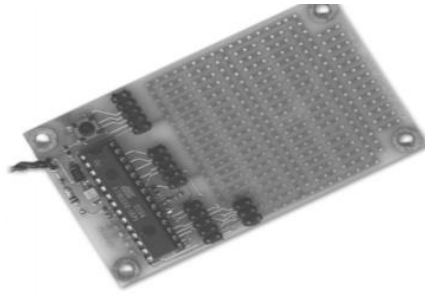


Figura 2.28: ATmega8  
Fuente: (Zabala, 2007)

Existe otro tipo de microcontroladores que también son muy utilizados en la construcción de robots y pertenecen a la empresa Atmel, los denominados AVR de 8 bits. La gran mayoría de microcontroladores AVR poseen las características de bajo poder de consumo, arquitectura RISC y Harvard, 32 registros de 8 bits de propósito general y facilidad de implementación de lenguajes de alto nivel para la programación, siendo el micro ATmega8 ideal, figura 2.28, dado que presenta una memoria de 8 KB de programa, 1 KB de SRAM y 512 bytes de EEPROM, seis canales de PWM, USART (Transmisor-receptor Asíncrono Universal) programable, conversor analógico digital de cuatro canales multiplexados de 10 u 8 bits y dos canales de 8 bits. (Zabala, 2007)

### **2.7.5. Microcontroladores PIC**

Los microcontroladores PIC han tenido una gran participación en la elaboración de robots a nivel mundial gracias a que son capaces de cubrir las exigencias de los fabricantes, ofreciendo nuevas presentaciones y tipos de memoria, además de su sencillo uso por sus instrucciones optimizadas y herramientas de desarrollo, siendo su bajo costo y la facilidad de información que se obtenía de ellos las principales razones para elegirlos.

La firma Microchip diseña microcontroladores de 8bits de propósito general de arquitectura RISC (conjunto reducido de instrucciones), bajo consumo de potencia y una distribución interna en donde la memoria de programa (ROM) y la memoria de datos (RAM) se encuentran en bloques separados, cada una con su propio bus de acceso (estructura Harvard). (Castaño Welgos, 2002).

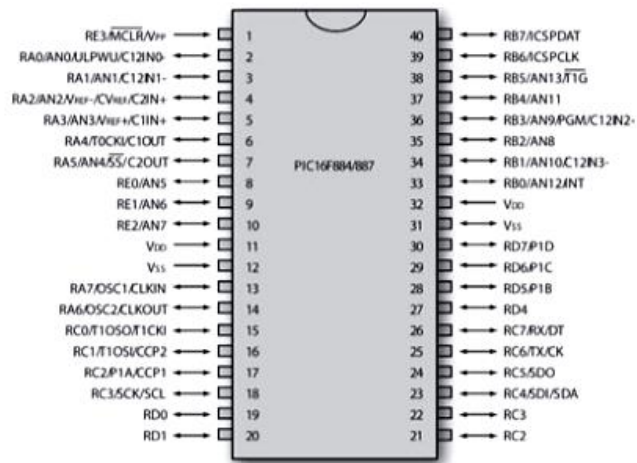


Figura 2.29: Microcontrolador PIC16F884887

Fuente: (Benchimol, 2011)

Son clasificados por la longitud de la palabra de sus instrucciones, siendo 3 en total, los de rango bajo que comprenden palabras de 12 bits de longitud, los de rango medio de 14 bits de longitud y los de rango alto de 16 bits de longitud. En la figura 2.29 se observa los pines de entrada y salida de un microcontrolador PIC, las características generales de un microcontrolador PIC son tres: el núcleo, los periféricos y sus características especiales.

### 2.7.5.1 El núcleo

Contiene las características fundamentales para la operatividad del microcontrolador, formadas por:

- **Unidad central de proceso (CPU):** Recibe las instrucciones de la memoria del programa para el control y operatividad de todo el sistema.
- **Unidad aritmético-lógica (ALU):** Es la que se encarga de las operaciones lógicas y aritméticas del microcontrolador y opera las instrucciones sobre la memoria de datos.
- **Circuito oscilador:** Generar la señal de reloj, que se necesita para que se ejecuten las instrucciones del dispositivo y se activen los periféricos.
- **Sistema para manipular (Reset):** Lleva al sistema electrónico del microcontrolador a su estado inicial con el que siempre comienza y en donde se garantiza su correcto funcionamiento en la ejecución de tareas.
- **Conjunto de instrucciones:** Se identifica la instrucción mediante un código de operación.

### 2.7.5.2 Los periféricos

Facilitan la comunicación con el entorno mediante los puertos de entrada o salida, entradas analógico/digital, manejadores LCD, salidas digitales con PWM (Pulse Wide Modulation) modulación por ancho de pulso.

Los principales periféricos de un microcontrolador PIC son:

- **Líneas de entrada y salida:** Permiten la transmisión de señales desde y hacia el mundo exterior.
- **Los temporizadores:** Son utilizados debido a que establecen bases de tiempo confiables, como por ejemplo medir el tiempo que ha transcurrido entre dos eventos, para realizar tareas en periodos de tiempo regulares, contar eventos, etc.
- **Módulos de captura, comparación y PWM (CCP):** Se utilizan en aplicaciones de control de motores, comparación de señales, la PWM emula un valor analógico es decir que se obtiene a través de la salida digital una onda cuadrada cuyo ancho puede ser variado, ya que los microcontroladores no disponen de salidas analógicas normalmente por su costo y dificultad de implementación. (Vazques Fernandez-Pacheco et al., 2016)
- **Puerto serial síncrono (SSP):** Este módulo es una interfaz serial para la comunicación con otros microcontroladores u otros periféricos tales como memorias seriales, registros de desplazamiento, conversores A/D, etc. (Castaño Welgos, 2002)
- **USART:** Se denomina módulo universal receptor transmisor asíncrono. Es uno de los dos tipos de puertos seriales (el otro es el SSP). Puede configurarse en modo *full dúplex* o como *half dúplex*. (Castaño Welgos, 2002)
- **Conversores:** Convierten una señal análoga en un dato digital de 8 o 10 bits.
- **Manejo de Display:** Permite el control de la información necesaria para manejar un panel LCD y sus respectivos pixeles.

### 2.7.5.3 Características especiales

Incrementa la funcionalidad y flexibilidad del sistema además de que reducir su costo.

- **Bits de configuración del dispositivo:** Facilita la personalización de ciertos aspectos del dispositivo para determinada aplicación.
- **Power-on Reset:** Se lleva el contador de programa a la posición inicial mediante RESET cuando se alimenta el sistema.
- **Temporizador (*Watchdog*):** Es un oscilador tipo RC que sirve para prevenir errores en la operación de un sistema, cuando el dispositivo se bloquea por alguna razón genera un reinicio del sistema.
- **Modo de bajo consumo de potencia (*Sleep*):** Reduce el consumo de potencia desactivando el oscilador del PIC, hasta que se vuelva a requerir una potencia mayor.
- **Reloj principal:** Los microcontroladores disponen de un circuito oscilador que les permite generar una onda cuadrada de alta frecuencia, la cual estructura los impulsos de reloj plasmados en la sincronización de todas las operaciones. Se puede considerar esta señal como el motor del sistema ya que logra que los contadores y el programa puedan avanzar en su ejecución.

### 2.7.6. Microcontroladores AVR

Los microcontroladores AVR se basan en un núcleo cuya arquitectura corresponde de la firma Atmel. El término AVR no tiene un significado implícito, a veces se considera como un acrónimo en el que se involucra a los diseñadores del núcleo, es decir AVR puede corresponder con Alf-Vegard-RISC. (Santiago Espinosa, 2012)

#### 2.7.6.1. El Núcleo AVR

Es la unidad central de procesamiento (CPU), es decir, es el hardware encargado de la captura, decodificación y ejecución de instrucciones, en torno al núcleo se encuentra un bus de 8 bits al cual están conectados los diferentes recursos del microcontrolador, estos recursos pueden diferir entre dispositivos. (Santiago Espinosa, 2012)

Para mejorar el rendimiento el AVR usa una arquitectura Harvard con memorias y buses separados para el programa y los datos, como se observa en la figura 2.30 el programa se encuentra en la memoria flash y los datos

están en 3 espacios diferentes: En el archivo de registros (32 registros de 8 bits), en la SRAM (RAM estática-mantiene datos mientras se encuentre alimentada) y en la EEPROM (Santiago Espinosa, 2012).

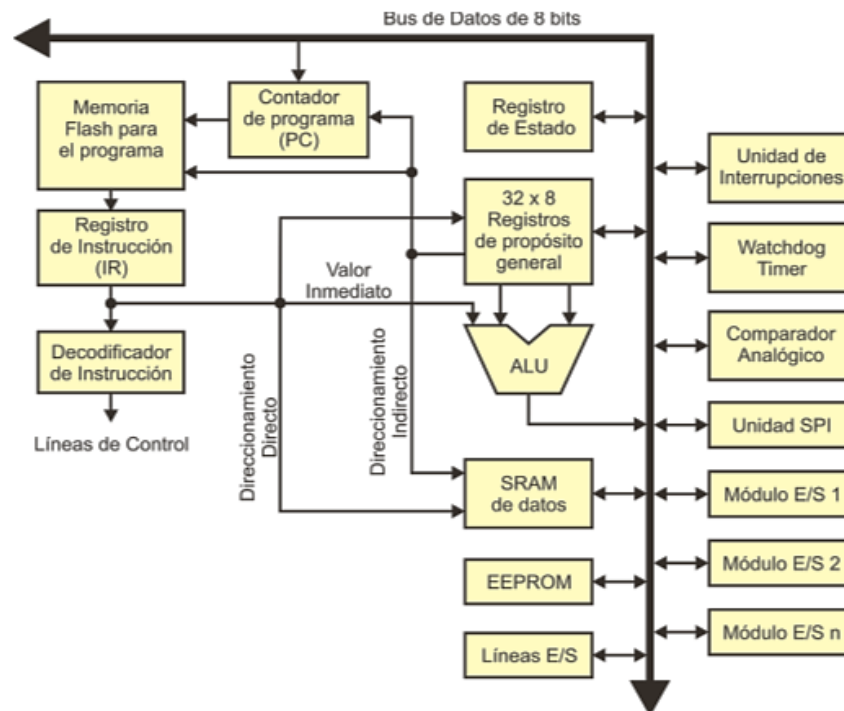


Figura 2.30: Diagrama de bloques del núcleo AVR  
Fuente: (Santiago Espinosa, 2012)

De la memoria flash se obtiene cada instrucción del programa y se coloca en el registro de instrucción (IR) para su decodificación y ejecución. La memoria flash es direccionada por el contador de programa (PC) o bien, por uno de los registros de propósito general. El PC es en sí el registro que indica la ubicación de la instrucción a ejecutar, sin embargo, es posible que un registro de propósito general proporcione esta dirección a modo de que funcione como apuntador y se haga un acceso utilizando direccionamiento indirecto. (Santiago Espinosa, 2012)

## 2.8. Comunicaciones inalámbricas en robots.

Poder monitorear y controlar a un robot de manera remota mediante un enlace inalámbrico es una de las maneras más prácticas a la hora de trabajar en espacios limitados y corta distancia, esto es posible mediante el uso de un controlador remoto que establece comunicaciones inalámbricas por medio de señales infrarrojas, bluetooth, y radiofrecuencia.

El controlador remoto, también llamado transmisor, es un dispositivo móvil que debe proporcionar facilidades de comunicación y control para un dispositivo dado, en donde su única fuente de energía razonable son las baterías, así que el consumo mínimo de energía es uno de los requisitos más importantes.

### **2.8.1. Comunicaciones por infrarrojos.**

Los controles remotos infrarrojos usan pulsos de luz invisibles debajo del espectro de longitud de onda visible (aproximadamente 950nm). En términos de su comportamiento de radiación, es como cualquier otra fuente visible de luz: debe haber una línea de visión entre el transmisor (fuente de luz) y el receptor (detector de luz). Cualquier obstáculo entre el transmisor y el receptor evitará una recepción correcta.

En buenas condiciones, la luz dispersada o la luz reflejada desde las paredes pueden mantener el sistema funcionando. Tener paredes entre el control remoto y el dispositivo receptor definitivamente desactivará el control remoto. Esta desventaja obvia de los controles remotos IR simplifica el protocolo al mismo tiempo. Todos los controles remotos para el mismo modelo de dispositivo pueden usar codificación idéntica. No es necesario un emparejamiento explícito entre el transmisor y el receptor con códigos exclusivos dedicados, ya que la diferenciación está garantizada por la separación óptica de dispositivos idénticos.

La comodidad proporcionada por los controles remotos es esencial en nuestra vida cotidiana. Los controles remotos infrarrojos (IR) han demostrado ser una solución rentable para controlar muchos tipos de dispositivos electrónicos: entretenimiento en el hogar, aire acondicionado, iluminación del hogar, pero existen algunas limitaciones y desventajas al usar luz infrarroja como medio.

Prácticamente no existe una regulación legal de la máxima potencia de luz emitida. Esto está más limitado por el consumo de energía (duración de la batería) y el alcance. Hasta 10 metros es el rango típico de los controles

remotos de infrarrojos asumiendo condiciones perfectas (controlador físicamente orientado hacia el receptor, sin obstáculos y sin perturbación de otras fuentes de luz).

Para que la conexión de infrarrojos sea más robusta frente a cualquier otra fuente de luz infrarroja (por ejemplo, lámparas fluorescentes), la señal de control remoto se activa y desactiva a una frecuencia de aprox. 38 kHz. Uno de los problemas que ocurren normalmente son las interferencias debido a fuentes de luz como el sol o bombillas de luz, ya que también emiten luz infrarroja lo que interfiere con los mandos a distancias como controles remotos, sin embargo, eso se soluciona poniendo un filtro de frecuencia para que solo escuche la frecuencia de 38.5 KHZ que nadie más en la naturaleza es capaz emitir.

### **2.8.2. Radiofrecuencia**

La comunicación por radiofrecuencia, utiliza ondas electromagnéticas como medio de transmisión. Sus propiedades de propagación permiten conexiones de distancias extremadamente largas. Las ondas electromagnéticas pasan a través de materiales no protectores hasta cierto punto (por ejemplo, paredes de hormigón).

Este atractivo exige alguna limitación impuesta por las regulaciones mundiales definidas por los grupos de estandarización como, por ejemplo: European Telecommunications Standards Institute (ETSI), Federal Communications Commission (FCC). Para todas las regiones del mundo existen regulaciones vinculantes que cubren los siguientes aspectos:

- Bandas de frecuencia (espectro)
- Potencia máxima de las ondas de radio emitidas
- Ancho de banda de la señal emitida
- Duración de las emisiones (ciclo de trabajo)
- Propósito de la emisión (por ejemplo, emisión de TV, redes de telefonía móvil, comunicación de la autoridad, propósito general)
- Tarifas de licencia



Para aplicaciones de control remoto hay disponibles algunas bandas sin licencia, las llamadas bandas ISM (bandas industriales, científicas y médicas). Para la Región1 (Europa, África, antigua Unión Soviética), la banda de frecuencias más atractiva se encuentra en 434 MHz, para América del Norte y del Sur en 902 a 928 MHz. Además, en Europa, la banda de SRD (banda de dispositivos de corto alcance) a 868 MHz y en los EE. UU.

La banda de frecuencia a 315 MHz se utiliza comúnmente para los controles remotos. Una alternativa con compatibilidad mundial es la banda ISM a 2,4 GHz. Lamentablemente, 2,4 GHz requiere un mayor consumo de energía a un rango comparable y, por lo tanto, un costo mayor. Prácticamente, los controles remotos de RF usan 315, 434 u 868 MHz como frecuencia portadora de RF. El uso de receptores integrados de bajo costo y tamaño que trabajan en frecuencias entre 300 y 400 MHz permiten la comunicación RF en los robots.

### **2.8.3. Bluetooth**

Bluetooth es un estándar que facilita el intercambio de información entre dispositivos ya sean móviles o estáticos mediante un enlace de radiofrecuencia de 2.4 GHz brindando velocidades de hasta 3Mbps, siendo muy eficientes para comunicaciones punto a punto entre dispositivos.

Una gran cantidad de dispositivos electrónicos poseen bluetooth actualmente como por ejemplo los Smartphone, siendo estos útiles a la hora de controlar y monitorear un robot, la mayoría de los robots tienden a comunicarse vía bluetooth, ya que pueden ser programados desde el mismo móvil o el robot puede utilizar la tecnología de dicho dispositivo, como su conexión a internet, giroscopio, cámara, etc.

## **2.9. Comunicaciones de radiofrecuencia y control remoto**

Se establecen diferentes estándares y protocolos para Radiofrecuencia. Básicamente, los protocolos de control remoto de RF representan información de dirección y comando del dispositivo. El consumo de energía de un

transmisor de RF es bajo y las velocidades de datos máximas de un enlace de RF son altas, lo que da como resultado tramas con una duración más corta.

Es posible utilizar tramas con un mayor número de bits y más contenido de información utilizando RF, al mismo tiempo que se ahorra tiempo de duración de la batería. Este hecho permite algunas características más sofisticadas para los controles remotos de RF. Cada control remoto de RF requiere una identificación única, los controles remotos de RF usan identificaciones únicas reales, lo que significa que cada transmisor en todo el mundo tiene otra ID. Por lo tanto, la longitud de bit reservada para ID de RF es más larga (por ejemplo, 32 bit). (Gotschlich, 2010)

Algunos protocolos de RF incluso implementan un contador que señala la frecuencia con la que se presiona un botón repetidamente. Para una robustez mejorada del enlace de RF, a menudo se generan y transmiten valores de CRC (verificación de redundancia cíclica) como parte de la trama (por ejemplo, 8 bits).

El receptor puede identificar claramente cualquier error de bit recalculando el valor de CRC de la trama de datos recibida y comparándolo con el generado antes de la transmisión. Se puede introducir seguridad adicional mediante códigos sucesivos que cambian algunos bits con cada pulsación de botón (por ejemplo, contador simple de 16 bits) y el cifrado de la carga completa o al menos parte de ella (por ejemplo, AES- Advanced Encryption Standard).

### **2.9.1. Características generales de los controles remotos de radiofrecuencia**

- Consumo de energía al transmitir, entre 5 a 20 mA
- Consumo de energía al recibir, entre 5 mA
- Alcance hasta 50 mts con buena línea de vista.
- Opción bidireccional para mayor robustez del enlace y funciones ampliadas, especialmente para operaciones fuera de la vista)

- Longitud-bit de trama, típicamente, 80 hasta 140 bit es posible. Para características extendidas como seguridad, robustez e información de estado adicional (por ejemplo, estado de voltaje de la batería).
- Velocidad de datos hasta 100 kbit / s

#### **2.9.1.1. Despertar al receptor**

El bajo consumo de energía de reserva del receptor es un requisito esencial. Un receptor RF consume 5 mA o más, mientras que un receptor IR puede permanecer alimentado a un consumo considerablemente bajo (aproximadamente 1 mA). Para superar este problema, se puede aplicar un esquema de sondeo adecuado.(Gotschlich, 2010)

Durante el sondeo, el receptor se enciende durante un corto período de tiempo para verificar si hay una señal válida disponible. De lo contrario, el receptor vuelve al modo de apagado de inmediato. Dependiendo del intervalo de sondeo, debe diseñarse un protocolo de RF adecuado. Después de presionar un botón en el control remoto, se debe enviar una secuencia de ejecución predefinida lo suficientemente larga para que el receptor pueda despertarse una vez finalizado el intervalo de sondeo.

#### **2.9.1.2. Sincronización del receptor con el reloj de datos entrantes**

La comunicación de RF es mucho más sensible a los errores de reloj. Por lo tanto, el receptor tiene que medir el reloj entrante de la trama de datos antes de comenzar la decodificación de la carga útil. El preámbulo suficientemente largo (patrones de inserción y sincronización) con contenido fijo (por ejemplo, 000 ... 0001) en el protocolo permite que el receptor de RF se sincronice con el reloj de la señal entrante.(Gotschlich, 2010)

#### **2.9.1.3. Identificación única y emparejamiento**

El transmisor de RF y el receptor de RF deben emparejarse usando un código único incrustado en la trama de datos. Generalmente, cada transmisor de RF proporciona un identificador único codificado. El receptor debe emparejarse con cada uno de los transmisores asignados durante un

procedimiento de emparejamiento en el que el receptor aprende la ID única de un transmisor determinado. Normalmente, el proceso de emparejamiento se introduce presionando un botón dedicado en la unidad receptora. La próxima identificación recibida se agregará a la lista de dispositivos emparejados.

En funcionamiento normal, el receptor solo aceptará tramas de datos con códigos de ID aprendidos durante el procedimiento de vinculación anterior. El resto de cuadros con códigos que no coinciden simplemente se ignoran. Como consecuencia, el receptor requiere memoria no volátil (por ejemplo, EEPROM, Flash) para mantener la lista de emparejamiento incluso durante el apagado.

## **2.9.2. Estándares y protocolos de comunicaciones inalámbricas**

### **2.9.2.1. IEEE 802.15.4**

Este estándar define las especificaciones de la capa física y control de acceso al medio para conectividad inalámbrica de baja velocidad de datos con dispositivos fijos, portátiles y en movimiento sin batería o requisitos de consumo de batería muy limitados que típicamente operan en el espacio operativo personal de 10 m. La velocidad de datos en bruto es lo suficientemente alta (250 kb/s) para satisfacer un conjunto de aplicaciones. (IEEE Computer Society, LAN/MAN Standards Committee, Institute of Electrical and Electronics Engineers, & IEEE-SA Standards Board, 2011)

Una LR-WPAN (Low-rate wireless personal area network) es una red de comunicación simple y de bajo costo que permite la conectividad inalámbrica en aplicaciones con una potencia limitada y requisitos de rendimiento simples. Los principales objetivos de un LR-WPAN son la facilidad de instalación, la transferencia de datos confiable, un costo extremadamente bajo y una duración de batería razonable de los dispositivos, a la vez que se mantiene un protocolo simple y flexible. (IEEE Computer Society et al., 2011)

Algunas de las capacidades provistas por este estándar son las siguientes:

- Operación estrella o peer-to-peer.

- Prevención de colisiones CSMA/CA (Acceso múltiple por detección de portadora y prevención de colisiones)
- Protocolo completamente reconocido para la confiabilidad de la transferencia
- Calidad de enlace (IEEE Computer Society et al., 2011).

Tabla 2.1: Propiedades de IEEE 802.15.4

Propiedad	Rango
Rango de transmisión de datos	868 MHz: 20kb/s; 915 MHz: 40kb/s; 2.4 GHz: 250kb/s.
Alcance	10 – 20 m.
Latency	Debajo de los 15 ms.
Canales	868/915 MHz: 11 canales. 2.4 GHz: 16 canales.
Bandas de frecuencia	Dos PHY: 868/915 MHz y 2.4 GHz.
Direccionamiento	Cortos de 8 bits o 64 bits IEEE
Canal de acceso	CSMA-CA y rasurado CSMA-CA
Temperatura	El rango de temperatura industrial: -40° a +85° C

Fuente: (IEEE Computer Society et al., 2011)

### 2.9.2.2. ZigBee

ZigBee es un estándar para redes inalámbricas de área personal (WPAN) de baja potencia, es decir, redes inalámbricas con un rango corto, típicamente 10-100 metros. ZigBee define servicios adicionales (inicia el enrutamiento de topología, el cifrado, los servicios de aplicaciones) sobre 802.15.4. (Gascón, 2009)

ZigBee se usa comúnmente para aplicaciones de monitoreo y control inalámbrico, como redes inalámbricas de sensores (WSN), monitoreo de plantas industriales, control de edificios, hospitales, medición inteligente y domótica. ZigBee opera en las bandas de radio Industrial, Científica y Médica (ISM) y la frecuencia exacta dependerá de dónde se encuentre en el mundo.

Puede usar la banda de 868 MHz en gran parte de Europa, 915 MHz en los EE. UU. Y 2.4 GHz en muchos otros lugares. La banda de 2,4 GHz es muy común ya que muchos de los chipsets (conjunto de circuitos integrados diseñados con base en la arquitectura de un procesador) lo utilizan.

Las velocidades disponibles dependen de la banda que esté utilizando, pero el máximo es de 250 Kbps. Esto es más lento que otras tecnologías inalámbricas populares, como WiFi, pero también es más barato y de menor costo. Más específicamente, ZigBee está construido sobre la especificación 802.15.4 que define la Física (PHY) y los Medios Capas de control de acceso (MAC) para WPAN de baja velocidad (LR-WPAN).

Como se observa en la figura 2.31, ZigBee agrega capas además de esto, agrega más aplicaciones de inteligencia y redes. 802.15.4 es la base para muchos otros protocolos inalámbricos industriales. Una red ZigBee permite que un conjunto de dispositivos se comuniquen de forma inalámbrica a través de una de varias topologías posibles.

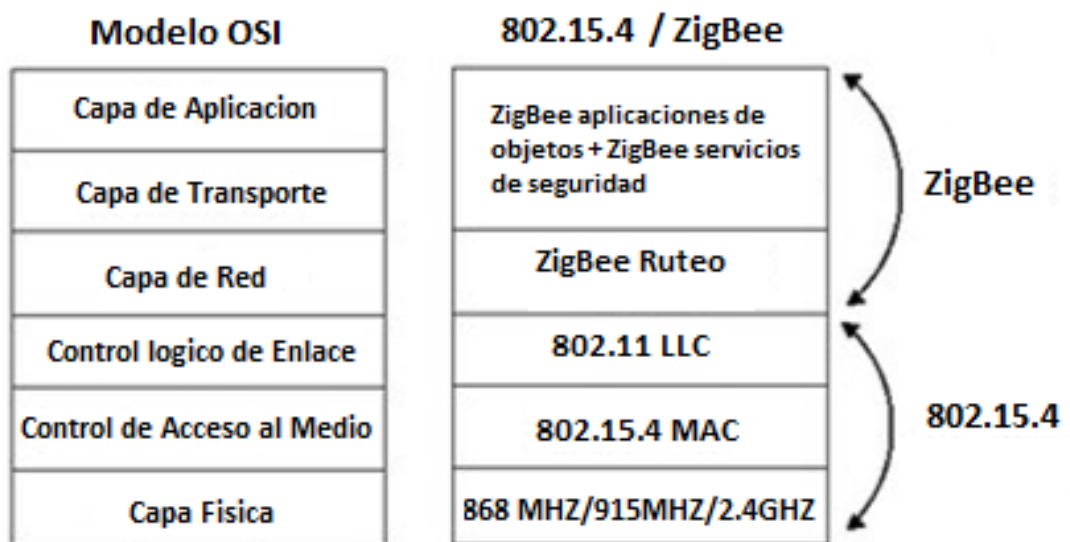


Figura 2.31: Modelo OSI, Zigbee Vs 802.15.4

Fuente: (Gascón, 2009)

Los paquetes de datos pueden enviarse entre nodos y pueden ser enrutados por dispositivos intermediarios a nodos más distantes que de otro

modo estarían fuera del alcance. Cada dispositivo tiene una dirección MAC y una dirección de red ZigBee, y la red como un todo tiene su propia identificación PAN compartida por todos los dispositivos. Los paquetes pueden protegerse mediante cifrado, pero para que esto funcione, todos los nodos necesitarán una clave.

### **2.9.2.3. Protocolos de radiocontrol**

Los protocolos RC son básicamente los lenguajes utilizados por diferentes dispositivos para comunicarse entre sí y se dividen en dos grupos: Los protocolos de transmisión (Entre el transmisor y el receptor) y los protocolos de recepción (Entre el receptor y el controlador o tarjeta controladora).

Algunos de los protocolos de recepción o RX son universales para diferentes marcas de equipos de RF, pero algunos pueden ser exclusivos de ciertas marcas. Algunos de ellos son: PWM (universal), PPM (universal), PCM (universal), SBUS (Futaba, Frsky), IBUS (Flysky), XBUS (JR), MSP (Multiwii), SUMD (Graupner), SUMH (Graupner), CRSF - Crossfire (TBS), FPort (Frsky). (Liang Oscar, 2017)

Mientras que los protocolos de Transmisión (TX) son en su mayoría diferentes entre marcas. Algunas marcas pueden ofrecer múltiples protocolos diferentes dependiendo del receptor de radio de emparejamiento. Algunos de ellos son: D8 (Frsky), D16 (Frsky), LR12 (Frsky), DSM (Spektrum), DSM2 (Spektrum), DSMX (Spektrum) AFHDS (Flysky) AFHDS 2A (Flysky), A-FHSS (Hitec) FASST (Futaba) Hi-Sky (Desviación)

### **2.9.2.4. PWM - Modulación de ancho de pulso**

Este es el protocolo de radiocontrol más simple y antiguo. En la antigüedad, los receptores se usaban para controlar los servos directamente con la señal PWM estándar, un canal para cada servo. Los multirrotores requieren al menos de 4 a 5 canales (a veces incluso más) y se tendrán los mismos números de cables de servo conectados entre el receptor y el controlador de vuelo.

PWM significa modulación de ancho de pulso. Es una señal analógica donde la duración del pulso especifica la salida del servo o la posición del acelerador. La duración del impulso de la señal normalmente varía entre 1000  $\mu\text{s}$  y 2000  $\mu\text{s}$  (microsegundos), siendo 1000 $\mu\text{s}$  el mínimo y 2000 $\mu\text{s}$  el máximo. (Liang Oscar, 2017)

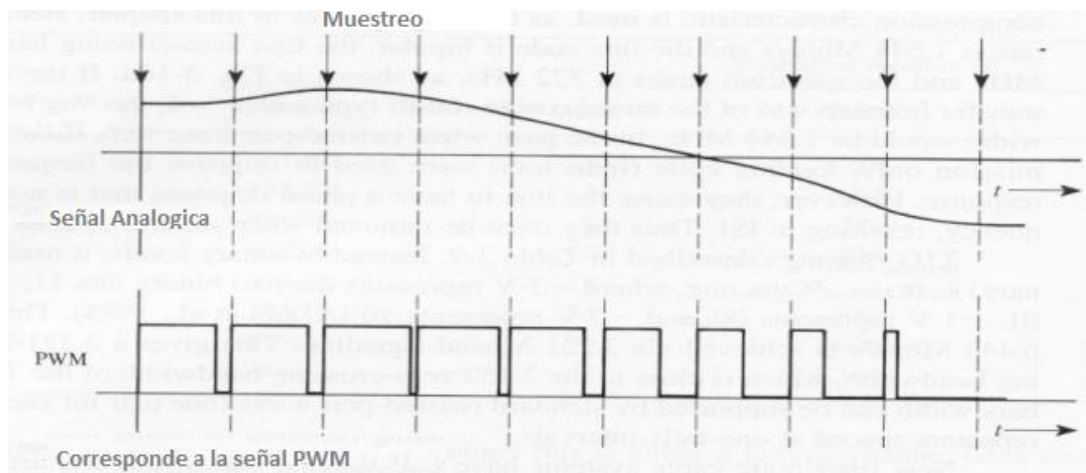


Figura 2.32: Modulación de una señal analógica en PWM  
Fuente: (Oscar Liang, 2013)

El valor de cada canal se representa como una señal "ON" de 1 milisegundo (ms) a 2ms y esta señal se repite (o actualiza) cada 20 milisegundos. Va alto para el 1-2ms, luego va abajo. La duración de tiempo que es alto es el valor para ese canal, como se observa en la figura 2.32.

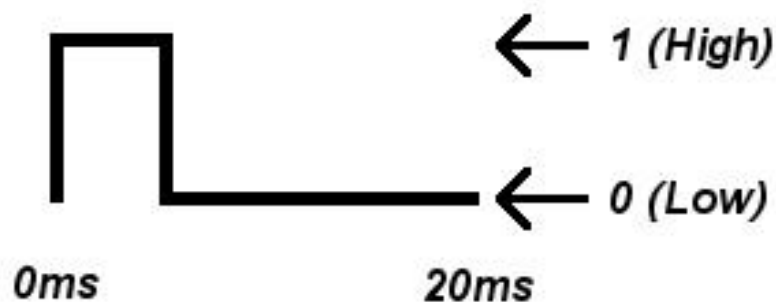


Figura 2.33: Pulso PWM  
Fuente: (Oscar Liang, 2013)

En la figura 2.33, el cuadro completo tiene 20 ms de longitud, la parte importante del pulso es el tiempo en que el pulso está activado; 1-2ms. Aunque el tiempo entre pulsos no es tan importante, juega un papel importante. Por lo general, mantener el tiempo entre pulsos de alrededor de 20 ms es lo mejor. El receptor de radio PWM es la opción más común y



generalmente la más económica. Sin embargo, debido al cableado desordenado, los aficionados ahora prefieren PPM o SBUS sobre PWM.

### 2.9.2.5. PPM - Modulación de la posición del pulso

PPM también se conoce como CPPM. La ventaja de PPM es que solo se necesita un cable de señal para varios canales (normalmente 8 canales como máximo), en lugar de un número de cables individuales. Por lo tanto, solo debe conectar el cable de tierra, alimentación y señal. (Liang Oscar, 2017)

Una señal PPM donde básicamente son una serie de señales PWM enviadas una tras otra en el mismo cable, pero la señal se modula de manera diferente. Aquí los pulsos tienen la misma longitud, pero el tiempo entre ellos varía para representar el valor del canal, como se observa en la figura 2.34.

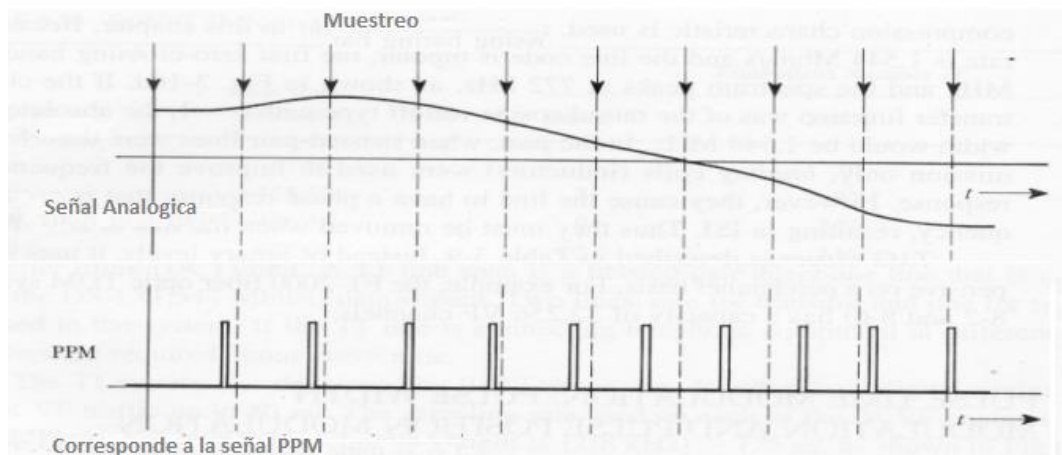


Figura 2.34: Modulación de una señal Analógica en PPM  
Fuente: (Oscar Liang, 2013)

El primer impulso marca el inicio del valor del canal 1, el segundo impulso marca el final del valor del canal 1 y el inicio del valor del canal dos. Al temporizar los espacios entre pulsos, se pueden obtener los datos para múltiples canales. PPM es lo que ellos llaman "señal analógica en el dominio del tiempo", los canales se envían uno tras otro y no al mismo tiempo. Todo este proceso se repite, generalmente cada 20 ms, como se observa en la figura 2.35.

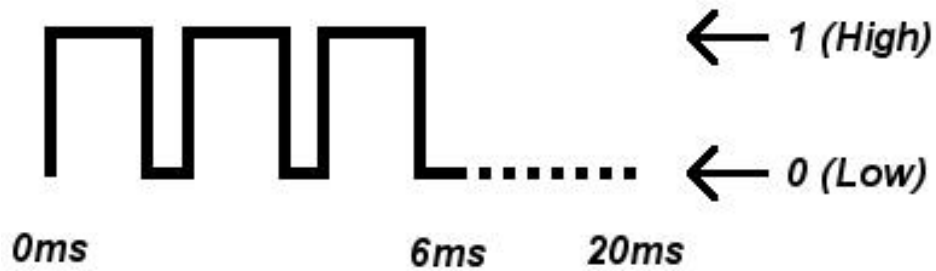


Figura 2.35: Pulso PPM  
Fuente: (Oscar Liang, 2013)

### 2.9.2.6. PCM - Modulación de código de pulso

PCM significa modulación de código de pulso, la señal PCM es una señal digital (utiliza unos y ceros). PCM tiene el potencial de detección de error de señal incluso corrección de errores, pero esto aún depende del producto que se compre. PCM es más confiable y menos susceptible a las interferencias, pero se requiere una conversión adicional por lo que el equipo tiende a ser más costoso.

### 2.9.2.7. Protocolos serie

Receptor serial es un protocolo digital sin pérdida que utiliza solo 3 cables (señal, potencia, tierra) para múltiples canales. Como su nombre indica, este tipo de receptor requiere un puerto serie en el controlador. Esto incluye SBUS, XBUS, MSP, IBUS y SUMD. (Liang Oscar, 2017)

- **SBUS (S.BUS, SERIAL BUS) - Por Frsky, Futaba**

SBUS es un tipo de protocolos de comunicación en serie, utilizado por Futaba y FrSky. Admite hasta 18 canales con un solo cable de señal.

- **CRSF (Crossfire)**

CRSF es un nuevo protocolo desarrollado por TBS. Es similar al SBUS. Las principales ventajas incluyen velocidades de actualización más rápidas y capacidades de comunicación bidireccional, permitiendo que cosas como la telemetría se inyecten en la corriente de comunicación sin necesidad de puertos adicionales. Esto permite la sintonización del controlador a través de su radio, así como la telemetría sin problemas a su control remoto a través del enlace de radio Crossfire, (Liang Oscar, 2017)

- **IBUS - Por Flysky**

IBUS es un protocolo de la marca flysky. Es una comunicación bidireccional que significa que puede enviar y recibir datos. IBUS es un protocolo en serie en el que básicamente se transmiten múltiples canales en serie por un cable. Su velocidad aproximada en baudios es 115200. Un segundo beneficio es que la información es digital, los valores recibidos no se basan en el tiempo de pulsos.

**2.9.2.8. Protocolos de Spektrum TX: DSM2 y DSMX**

"Spektrum" se menciona a menudo en temas relacionados con la radio, que es básicamente una marca de RC especializada en radio. DSM2 / DSMX son protocolos entre TX y RX (protocolo de radio) utilizados por el equipo Spektrum. La señal DSM2 es más resistente al ruido, la interferencia y otros transmisores que transmiten en la misma frecuencia. También encuentra una frecuencia de respaldo al inicio en caso de que la frecuencia primaria falle. Esto reduce la posibilidad de perder señal en gran medida; sin embargo, si ambos canales se vuelven inutilizables, puede perder la conexión. (Liang Oscar, 2017)

DSMX se basó y mejoró a partir de DSM2, que también usa el mismo esquema de codificación. La diferencia es que la señal DSMX puede cambiar a un nuevo canal de frecuencia en caso de corte en un par de milisegundos, por lo que, en teoría, ni siquiera notarías el fallo. DSM2 sigue siendo una tecnología popular, si está lejos de fuentes de interferencias de radio (como WiFi, microondas y cámaras de seguridad inalámbricas), debería funcionar igual de bien que DSMX. Pero DSMX es simplemente más confiable.

**2.10. Programación**

Un programa es un conjunto de instrucciones ordenadas para ejecutarse de forma rápida y precisa en una computadora. La programación es el proceso de escribir un programa, y el conjunto de instrucciones que se utilizan para escribir un programa se lo denomina como lenguaje de programación. (Marcelo Villalobos, 2014)

Existen diferentes tipos de lenguajes de programación, como el lenguaje máquina que conlleva una programación binaria, la cual es difícil de programar debido a que es dependiente de la máquina, y el lenguaje de alto nivel el cual es el más cercano al lenguaje natural, cuyo tiempo de programación es relativamente corto, además de que es independiente de la máquina. Para poder traducir el lenguaje de programación es necesario el uso de un compilador el cual traduce el código fuente (programas escritos en un lenguaje de alto nivel) a programa objeto (ejecutable en código máquina).

Para poder programar es necesario el uso de un algoritmo, que es el método que describe la solución de un problema computacional mediante una serie de pasos precisos, definidos y finitos. Además del uso de un algoritmo es necesario un entorno de programación o entorno de desarrollo interactivo, (IDE), la cual es una aplicación informática que nos permite desarrollar el software de un sistema.

### **2.10.1. Arduino**

Arduino es una plataforma de código abierto utilizada para la construcción de proyectos de electrónica, siendo muy utilizado en la programación de diferentes prototipos robóticos en las distintas categorías de competencia, como seguidores de líneas, robots sumos, robots de pelea y de fútbol y robots de presentación y demostración de habilidades.

Este entorno de programación, consiste en una placa de circuito programable física (a menudo denominada microcontrolador) y una pieza de software, o IDE (Entorno de desarrollo integrado) que se ejecuta en su computadora, que se usa para escribir y cargar códigos de computadora en la placa física.

La plataforma Arduino se ha vuelto bastante popular entre las personas que recién comienzan con productos electrónicos, y por una buena razón. A diferencia de la mayoría de las placas de circuitos programables anteriores, Arduino no necesita una pieza de hardware separada (llamada programador) para cargar código nuevo en la placa; simplemente se usa un cable USB.

Además, Arduino IDE utiliza una versión simplificada de C ++, por lo que es más fácil aprender a programar.

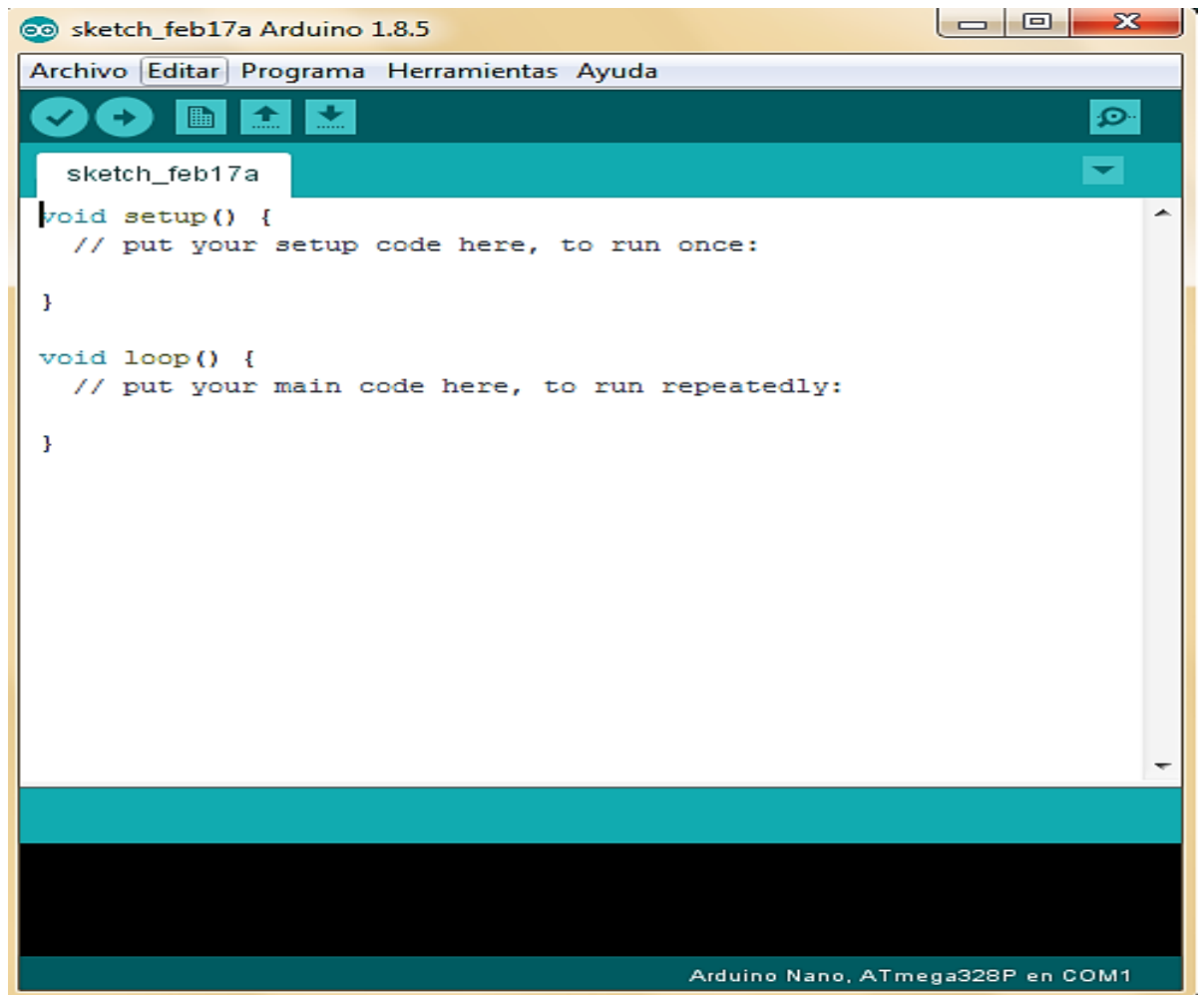


Figura 2.36: Entorno de programación de Arduino.  
Elaborado por: Autor

## **CAPÍTULO 3: DISEÑO Y PROGRAMACIÓN DEL ROBOT MEGA SUMO**

### **3.1. Desarrollo del proyecto**

En el presente capítulo se analiza los componentes electrónicos de un Kit de robot Mega Sumo, entre sus componentes se encuentra su tarjeta controladora, los sensores, los drivers de los motores. Para así entender sus características y poder diseñar una tarjeta electrónica que junto al algoritmo de programación propuesto mejore su sistema y por ende su funcionalidad.

Se procede a detallar la comunicación entre el Hardware y Software del robot la cual permite que funcione de manera autónoma o radio controlada según sea requerido, también se explica el algoritmo de programación propuesto el cual logra la autonomía del robot, y se establecen los argumentos los cuales determinan que el diseño del algoritmo y de la tarjeta propuesta mejoran el sistema del robot y por ende su funcionamiento.

### **3.2. Principio de funcionamiento de un robot sumo**

El robot Mega Sumo tiene como objetivo localizar al oponente durante la batalla mediante el uso de sensores y algoritmos de programación previamente establecidos en el robot, para así poder sacarlo del dojo (área de batalla), en un enfrentamiento de fuerza mediante el uso de sus motores y su diseño estructural, aplicando el empuje en el momento y lugar adecuado. Un robot sumo debe de conocer el entorno en donde se moverá, así como los límites del mismo con el fin de no abandonarlo, esto se logra mediante el uso de sensores que se encuentran generalmente en la parte inferior del robot cerca de los límites de su estructura.

El dojo oficial en donde se enfrentan los Robot Mega Sumo es de forma circular, tiene un diámetro de 154 cm, una altura de elevación de 5cm del piso y es de estructura metálica, posee dos colores en su superficie siendo el blanco y negro, el blanco representa los límites, y el negro el espacio donde el robot puede competir, como se observa en la figura 3.1, el duelo comienza con una distancia de separación entre oponentes que es de 20cm.

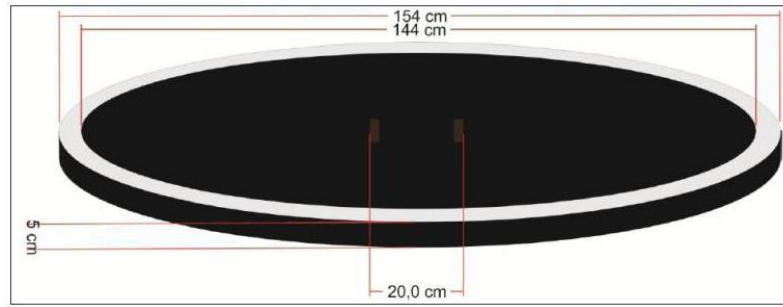


Figura 3.1: Ring de batalla de robot Mega Sumo  
Fuente: (Plúa, A & Castillo, J, 2015)

### 3.3. Requerimientos del Diseño Mecánico del Robot Mega Sumo

Los límites oficiales del diseño de robots Mega Sumo son de 20cm por 20cm de área, sin límite de altura y el peso no puede excederse de los 3000 gramos, incluyendo accesorios, es decir que es un robot robusto y pesado que debe ejercer una gran cantidad de fuerza para moverse y una mayor cantidad para empujar a su contrincante, además cabe mencionar que estos robots se adhieren a la base metálica del dojo mediante el uso de imanes en la parte inferior de su estructura.

Entre los componentes que conforman al robot Mega Sumo se tienen a la fuente de alimentación, sensores, motores, ruedas, driver y la tarjeta de control que viene a ser la inteligencia del robot. En la figura 3.2 se observa los elementos que conforman un robot.

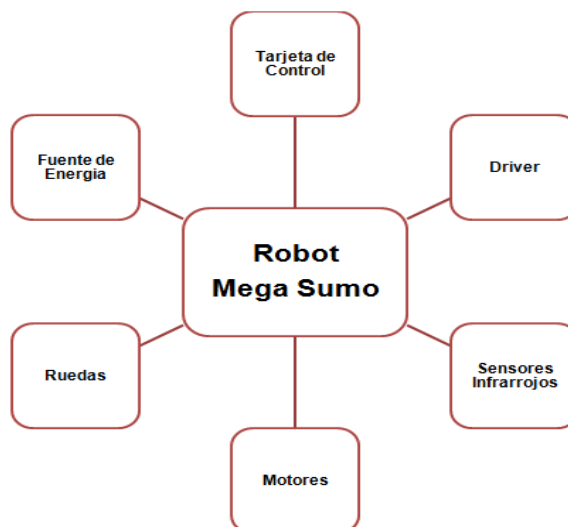


Figura 3.2: Estructura del robot Mega Sumo  
Elaborado por: Autor

### 3.4. Componentes del robot Mega Sumo

El Kit del robot Mega Sumo que se analiza posee 2 motores Maxon RE 40 de 150W que brindan la movilidad al robot, 2 ruedas de caucho de 50 mm, 3 sensores E3Z-D62 que se encargan de localizar al contrincante, 2 sensores QTR-1A que identifican el límite del área por la que se puede mover el robot, 1 módulo de receptor infrarrojo para activar al Mega Sumo manera remota, 10 imanes situados en la base de su estructura para ayudar a fijar al robot en la pista.

La etapa de inteligencia del robot y la de control de sus motores van depender de una tarjeta controladora comercial como lo es ArduPRO y un driver de motor denominado Sabertooth, en la parte de alimentación se utiliza las Lipo baterías. A continuación, se procede a describir los elementos que forman parte del robot.

#### 3.4.1. Motor Maxon RE Ø40 mm

Los motores de corriente continua de Maxon figura 3.3, son motores de alta calidad equipados con potentes imanes permanentes. En el robot mega sumo de este proyecto se utilizan dos motores Maxon RE Ø40 mm de 150 Watt, trabajan con un voltaje nominal de 24 Voltios, su velocidad sin carga llega a las 7580 rpm, su velocidad nominal es de 6940rpm, (pudiendo llegar a 12000 rpm), posee un torque nominal de 177 mNm a una corriente nominal (corriente continua máxima) de 6 A, y con un consumo de corriente de 80.2



Figura 3.3: Motor Maxon RE 40  
Fuente: (Maxon motor, 2018)

A cuando su máximo torque es aplicado el cual es de 2420 mNm, con un peso de 480 gr. En la figura 3.4 se muestra el rango de operación según el amperaje que se le administre en relación a revoluciones por minuto y torque.



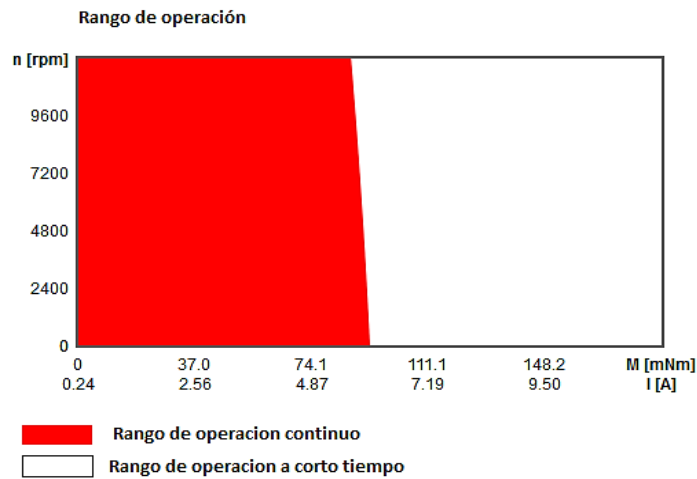


Figura 3.4: Rango de operación del motor Maxon RE 40  
Fuente: (Maxon motor, 2018)

### 3.4.2. Baterías de Polímero de litio

Las baterías de polímero de litio ("baterías LiPo") son un nuevo tipo de batería que actualmente se utilizan en muchos dispositivos electrónicos de consumo. Han ganado popularidad en la industria del radiocontrol y robótica en los últimos años, y ahora son la opción más popular para cualquiera que busque baterías de larga duración y alta potencia.

Entre las ventajas de estas baterías se encuentran que pueden acumular una gran cantidad de energía teniendo en cuenta su volumen y peso, poseen un alto voltaje, proporcionan un voltaje de 3.7 V, (tres veces más que otras baterías similares), vida útil de 600 ciclos de carga, presentan poco efecto de memoria y no se descargan por sí solas, es decir una batería de litio puede almacenar energía por varios días sin descargarse, mientras no se encuentre en uso. (Ardmine, 2017)

Su principal desventaja es que son muy sensibles a altas temperaturas o temperaturas extremas (mayor a 60° C y menor a 0° C). El kit de robot mega sumo contiene dos Lipo de baterías, una de 1600 mA con 6 celdas que proporcionan un voltaje de 22.2 voltios, para la Sabertooth y otra de 500 mA, de 3 celdas que proporciona 11.1 voltios para la placa controladora del Arduino.

### 3.4.3. Sensor de reflectancia QTR-1A

Es un sensor de reflectancia que puede identificar de cierta manera el color de una superficie, son muy utilizados en los concursos de robótica debido a su amplia variedad de usos en diferentes tipos de robots como seguidores de línea, o robots sumo, está compuesto por un LED infrarrojo y un fototransistor como se observa en la figura 3.5.



Figura 3.5: Sensor QTR-1A  
Elaborado por: Autor

El LED emite una señal infrarroja de corto alcance hacia la superficie de manera continua, el receptor (Fototransistor) es el encargado de recibir esa señal la cual tiene distinta intensidad dependiendo de si la superficie es clara u oscura, es decir que si la superficie es clara enviara una señal pero si es oscura no lo hará debido a que se absorbe la luz, impidiendo que la señal sea reflectada, esta medición de la reflectancia se envía como una tensión analógica, entre (0 y 5 voltios) normalmente, como se observa en la figura 3.6.

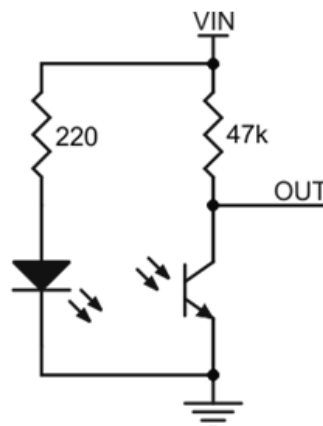


Figura 3.6: Circuito del sensor QTR-1A  
Elaborado por: Autor

### 3.4.4. Sensores E3Z-d62 ONROM

Es un sensor fotoeléctrico de reflexión difusa, funciona con un voltaje entre 12 a 24 Voltios DC, con un consumo de corriente de 30 mA, un rango

de detección de 0 a 1 metro dependiendo del uso que se le quiera dar, un tiempo de respuesta de 1 milisegundo ideal para el funcionamiento del robot a la hora de detectar al oponente rápidamente, en la figura 3.7 se observa las dimensiones del sensor, este sensor posee un peso de 70 gramos.

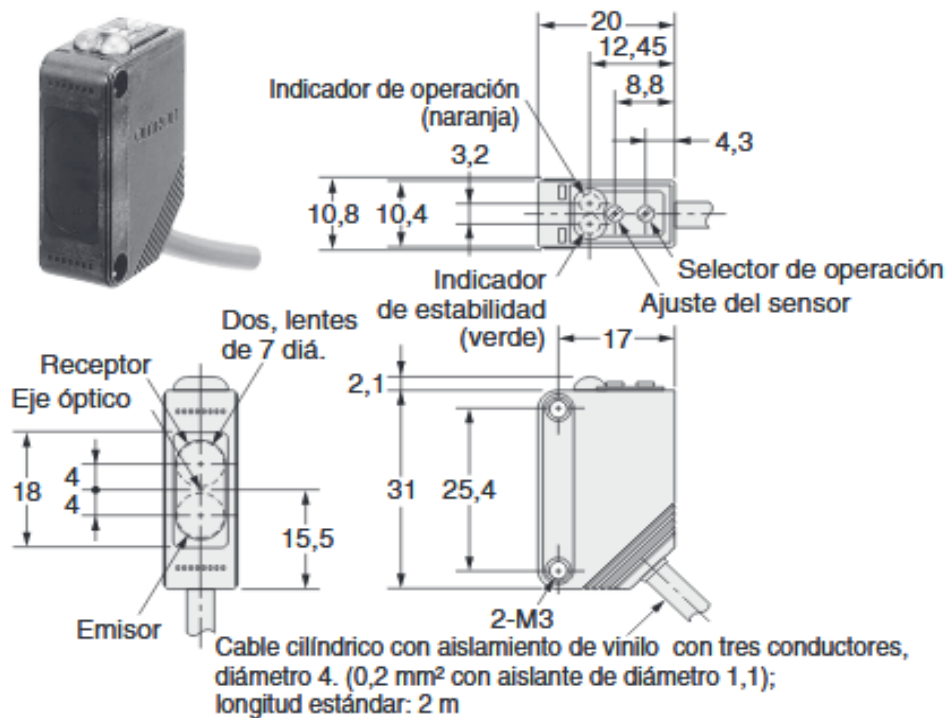


Figura 3.7: Dimensiones técnicas del sensor E3Z-d62 OMRON  
Fuente:("E3Z-D62 2M | OMRON Industrial Automation", 2018)

Su funcionamiento se basa en emitir una luz infrarroja hacia un objeto distante, en donde la superficie de dicho objeto formará un rayo incidente de múltiples ángulos y direcciones, llegando una parte de la luz reflejada al receptor el cual generará un cambio en su señal de salida. En la figura 3.8 se muestra el esquema de la salida del circuito en donde se observan dos LED el cual se si se encuentra en verde indica estabilidad del sensor, mientras en naranja indica que está operando.

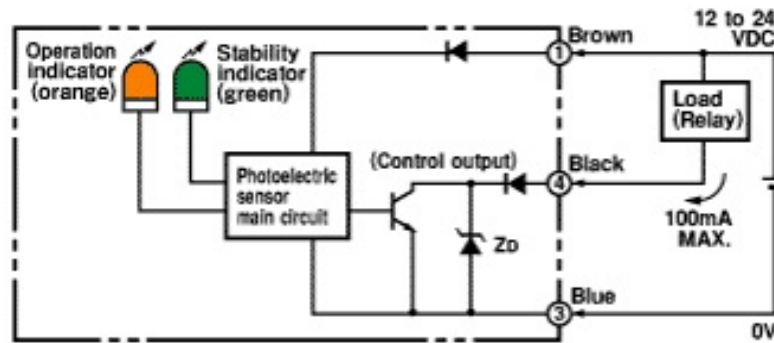


Figura 3.8: Salida del circuito

Fuente: ("E3Z-D62 2M | OMRON Industrial Automation", 2018)

### 3.4.5. Modulo Receptor infrarrojo

Permite iniciar el sistema del robot de manera remota, figura 3.9. Funciona con cualquier control remoto de marca SONY a un voltaje de 5V (adecuado para sistemas basados en Arduino, Pic, Atmel) posee un tamaño pequeño (13 mm x 16 mm x 8 mm), con dos LEDs: LED azul (Inicio), LED rojo (Stop), y pesa solo 2 gramos.



Figura 3.9: Modulo receptor infrarrojo

Fuente: (Jsumo, 2018)

El receptor debe estar conectado a los pines adecuados (Cualquier pin digital), a continuación se procede a registrar las señales de inicio y detención del control remoto (dos botones diferentes, dos señales), primero se presiona el botón blanco del MicroStart, mientras se presiona el botón, pasará al modo de grabación (Ambos LED parpadearán 3 veces), luego se presiona el primer botón del control remoto (será el botón de inicio), se encenderá el primer led (LED azul), se espera entre 1 a 2 segundos. Finalmente se presiona el segundo botón del control remoto (este será el botón detener) se encenderá el segundo led (LED rojo), después de soltarlo, ambos leds parpadearán 5 veces (en 3 segundos), cuando los LED no se encienden, su MicroStart está listo para dar señal.

### 3.4.6. Radiocontrol

El robot mega-sumo va a ser controlado a través de un transmisor y un receptor de radiocontrol de la marca Flysky (FS-T6), el sistema de radiocontrol trabaja en un rango de frecuencia de 2.4 y 2.4835 Ghz, esta banda consta de 160 canales independientes. Cada sistema de radio usa 16 canales distintos y 160 diferentes tipos de algoritmos de “Hopping“(saltos), mediante un Sistema Digital de Salto de Frecuencia Automático (AFHDS), en donde saltan las frecuencias de canal, garantizando una transmisión de radio impecable, de gran alcance y que no consume demasiada batería.

#### 3.4.6.1 El transmisor FlySky

Cada emisora consta de una ID fija. Al emparejar la emisora con un receptor, el receptor memoriza la ID fija de la emisora, por lo cual el receptor solamente puede procesar datos de dicha emisora. Esto evita que otras señales interfieran, de manera que aumenta su seguridad. En este equipo de transmisión, como se observa en la figura 3.10, utiliza componentes de baja energía y un chip en el receptor altamente sensible. En la tabla 3.1 se muestra los datos técnicos del transmisor.



Figura 3.10: Transmisor Flysky  
Elaborado por: Autor

Tabla 3.1 Datos técnicos Emisora

Canales	6
Frecuencia	2,4 GHz
Fuente de alimentación	8 x 1,5V „AA” baterías
Proceso remitente	AFHDS (cambio de frecuencia)
Modulación	GFSK
Transmisión de energía	20dBm (100mW)
Corriente	<250 mA

Elaborado por: Autor

El transmisor en estado de reposo envía una señal de 5V, como se observa en la figura 3.11, esta señal es recibida por el receptor para después ser enviada al Driver.

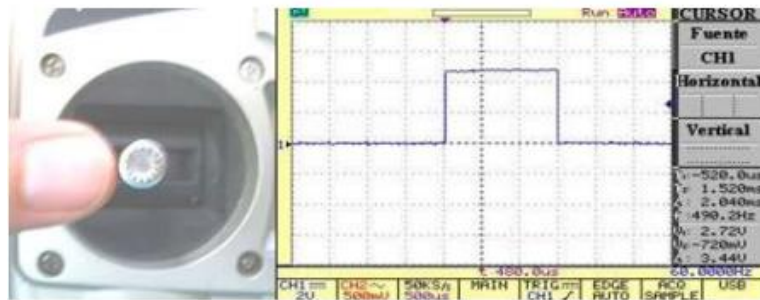


Figura 3.11: Señal emitida con la palanca posición inicial  
Fuente: (Quinde Llerena, J.L. & Ulloa Patiño, L.D., 2012)

Como se observa en la figura 3.12, al mover la palanca hacia arriba aumenta la duración con respecto a su posición inicial en unos cuantos milisegundos.

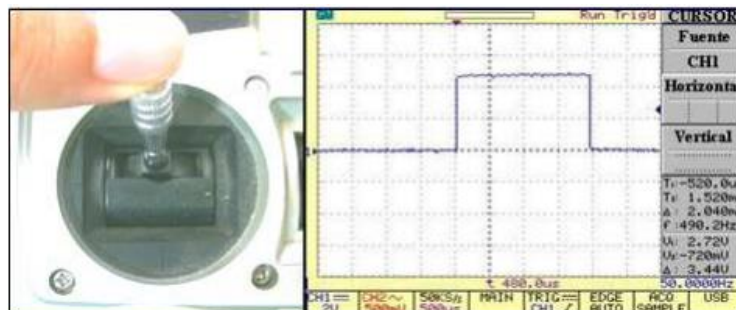


Figura 3.12: Señal emitida con la palanca posición superior  
Fuente: (Quinde Llerena, J.L. & Ulloa Patiño, L.D., 2012)

En la figura 3.13, se observa que al mover la palanca hacia abajo disminuye la duración con respecto a su posición inicial en unos cuantos milisegundos.

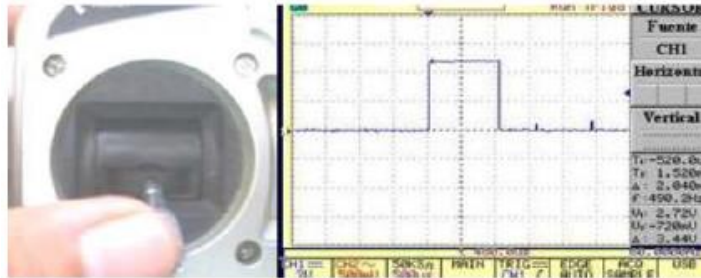


Figura 3.13: Señal emitida con la palanca posición inferior.  
Fuente: (Quinde Llerena, J.L. & Ulloa Patiño, L.D., 2012)

### 3.4.6.2 Modulación por desplazamiento de frecuencia Gaussiana GFSK

La modulación por desplazamiento de frecuencia (FSK) asigna diferentes frecuencias a la portadora dependiendo del bit que se transmite. Por lo tanto, cuando se transmite un bit 0, el operador asume una frecuencia que corresponde a un bit 0 durante la duración de un bit. Cuando se transmite un bit 1, la frecuencia portadora se cambia a un valor correspondiente a un bit 1 y, de manera similar, permanece en esta frecuencia durante la duración de un bit, como se muestra en la figura 3.14.

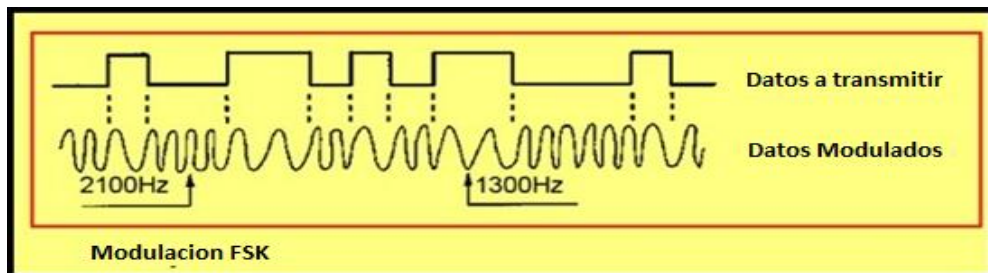


Figura 3 14: Modulación FSK  
Fuente: ("GFSK modulation - Near Communications", 2018)

En la modulación por desplazamiento de frecuencia gaussiana (GFSK) los datos de modulación son codificados en forma de variaciones de frecuencia en un portador de una manera similar a FSK. Por lo tanto, el modulador utilizado puede ser el mismo que se utiliza para la modulación FSK, la diferencia se encuentra en que los impulsos pasan a través de un filtro gaussiano antes de entrar en el modulador de pulso para disminuir el ancho espectral del mismo como se muestra en la figura 3.15. El filtro gaussiano es



un tipo de formateador de pulsos utilizado para suavizar la transición entre los valores de los impulsos. En donde un 1 lógico es representado mediante un incremento de la frecuencia de la onda portadora, y un 0 mediante un decremento de la misma.



Figura 3.15: Principio del filtro gaussiano sobre la banda base.  
Fuente: ("GFSK modulation - Near Communications", 2018)

### 3.4.6.3 Módulo de recepción

El módulo de recepción como se observa en la figura 3.16, tiene un sistema de 2.4GHz que pone la radio fuera del rango de frecuencia de cualquier "ruido" generado por otros componentes electrónicos en el modelo del robot, como el motor, el controlador de velocidad, los servos u otro ruido metal contra metal. Esto elimina interferencias y fallas que pueden afectar a un sistema de radio tradicional. En la tabla 3.2 se observa las características del receptor.



Figura 3.16: Receptor Flysky  
Elaborado por: Autor

Tabla 3.2 Datos técnicos del receptor

Canales	6
Frecuencia	2,4 GHz
Fuente de alimentación	4 x 1,5V „AA” baterías
Proceso remitente	AFHDS (cambio de frecuencia)
Modulación	GFSK
Sensibilidad	-105 dBm
Consumo de energía	<85 Ma (en descanso)

Elaborado por: Autor



### 3.4.7. Diseño estructural del Mega Sumo

La estructura mecánica del robot sumo se encarga de resistir el impacto que se genera a la hora de competir, está compuesta de un material resistente a golpes como lo es la lámina de acero inoxidable de 3mm de grosor, y en lugares de menor impacto utiliza aluminio, posee el espacio necesario para incluir todos los componentes del robot como sus sensores, motores y circuitos de control, en la figura 3.17 se muestra la estructura mecánica del robot y sus medidas elaborado en AutoCAD.

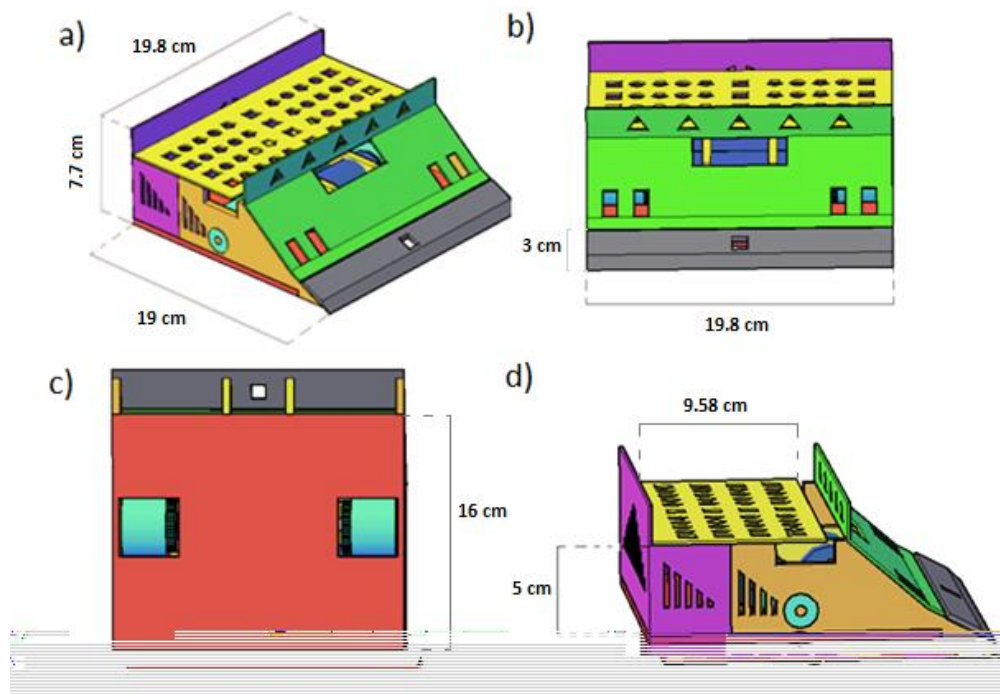


Figura 3.17: Dimensiones del robot Mega Sumo  
Elaborado por: Autor

En la figura 3.17 parte a) se observa que los límites del robot está dentro de las medidas del reglamento oficial para competencias en la categoría Mega Sumo, siendo sus medidas de 19.8 x 19 cm de área, con una altura de 7.7 cm, en la parte b) se observa la medida de la cuchilla rectangular que es de 3 cm y está elaborada con acero quirúrgico la cual se encarga de darle estabilidad al robot al momento del impacto, en la parte c) se observa la base del robot elaborada de láminas de acero inoxidable, cuya medida es de 16 cm sin contar la cuchilla, lugar donde van los imanes, y las ruedas conectadas a sus respectivos motores y en la parte d) se observa la medida donde irá la circuitería del robot que es de 9.58 cm x 19.8 cm, es decir la parte que le

brinda al robot su autonomía. En la figura 3.18 se observa al robot Mega Sumo con todos sus elementos y sus controladores comerciales (ArduPRO y Sabertooth) ensamblado.

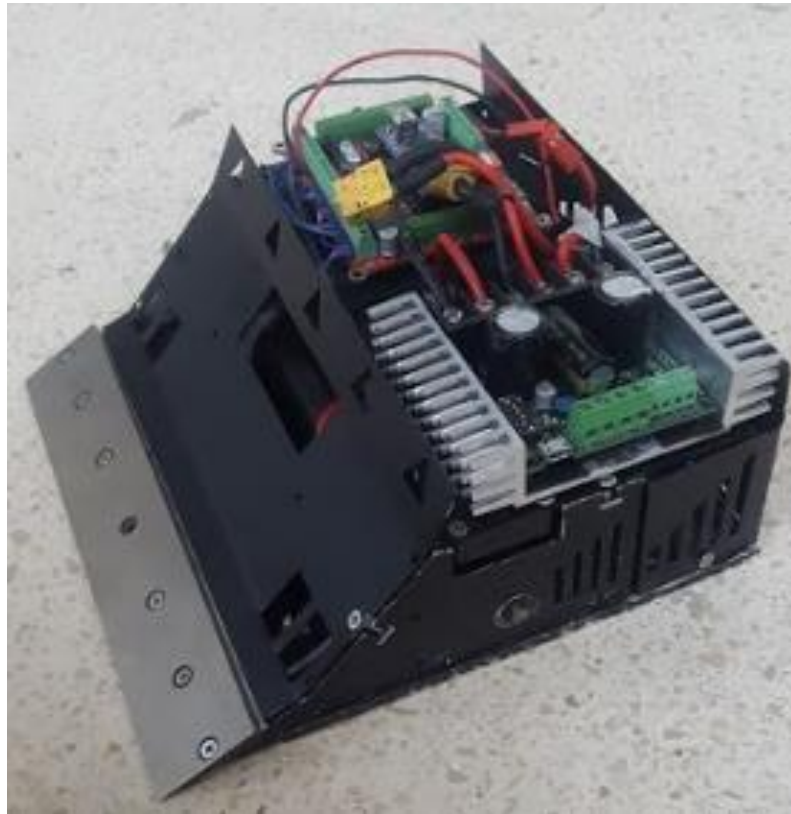


Figura 3.18: Robot Mega sumo  
Elaborado por: Autor

### **3.5. Sistema de control del robot y funcionamiento**

El robot Mega Sumo está conformado por uno de los elementos más utilizados en la etapa de control e inteligencia de robots, el Nano Arduino, esto se debe por su bajo costo, tamaño y por su práctica interfaz de programación. Uno de los principales problemas para poder implementar un Mega Sumo con Nano Arduino es el voltaje con el que trabaja, ya que si ese voltaje es excedido en alguna de sus entradas como resultado el microcontrolador se quemaría. Para evitar eso utiliza una placa controladora comercial denominada ArduPRO que brinda la capacidad de trabajar con una batería de mayor capacidad, la cual permite alimentar al Nano Arduino y los demás componentes del robot como lo son sus sensores, además de poder distribuir los pines de entrada y salida de una manera más eficaz para la conexión de sus componentes.

Para controlar los motores se hace uso de un driver el cual permite el paso de un mayor amperaje y resiste los picos de corriente que se generan con el accionamiento del motor y su consumo máximo de amperaje cuando aplican su máximo torque. La ArduPRO necesita comunicarse con un driver que pueda cumplir con esas características, para la cual se utiliza la Sabertooth 2x32. A continuación, se describe las características principales de los componentes de la etapa de control del robot y de sus motores.

### 3.5.1. Arduino Nano

El Nano Arduino es la versión más pequeña del Arduino uno, según PatagoniaTec, (2014) está basado en el microcontrolador Atmega328P con una arquitectura AVR, pensado para ser utilizado en protoboards debido a que la disposición de sus pines de entrada y salida facilitan la conexión de los componentes sin necesidad de muchos cables. Siendo su gran ventaja el tamaño que posee (18 x 45 mm) y un peso de 7 gramos, en la figura 3.19 se muestran los componentes del Nano Arduino.

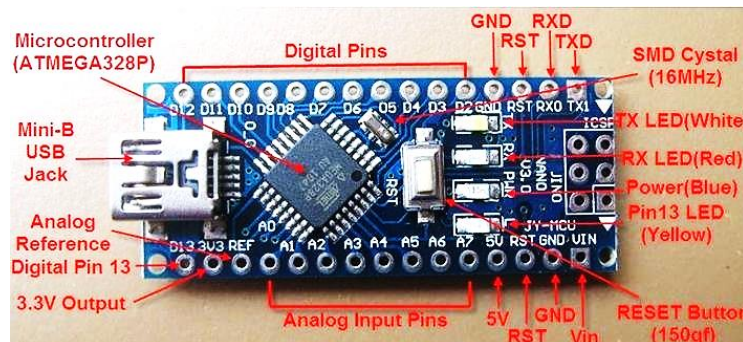


Figura 3.19: Componentes del Nano Arduino  
Fuente: (PatagoniaTec, 2014)

El Arduino Nano puede alimentarse a través de la conexión USB Mini-B, una fuente de alimentación externa no regulada de 6-20V (pin 30) o una fuente de alimentación externa regulada de 5V (pin 27), como se observa en la figura 3.20. La fuente de poder se selecciona automáticamente (la fuente de voltaje más alta). (PatagoniaTec, 2014)

Los 14 pines digitales en el Arduino nano se pueden usar como entrada o salida, operan a 5 V, cada pin puede proporcionar o recibir un máximo de

40 mA y tiene una resistencia interna de pull-up (desconectada por defecto) de 20-50 kOhms. (PatagoniaTec, 2014)

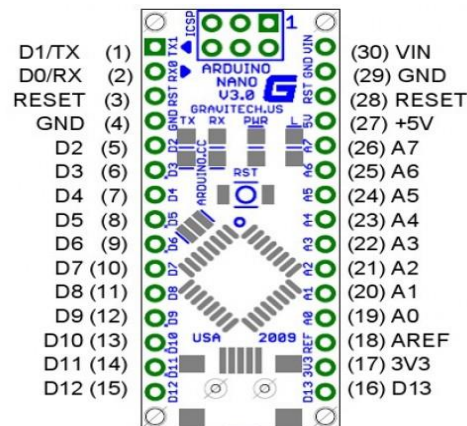


Figura 3.20: Diseño del Nano arduino  
Fuente: (Education, 2014)

Además, algunos pines tienen funciones específicas:

- **Serial:** 0 (RX) y 1 (TX). Se usa para recibir y transmitir datos en serie TTL (lógica-transistor a transistor). Estos pines están conectados a los pines correspondientes del chip serie FTDI USB a TTL. (PatagoniaTec, 2014)
- **Interrupciones Externas:** 2 y 3, pueden ser configurados para activar una interrupción.
- **PWM:** 3, 5, 6, 9, 10, y 11. Brindan una salida PWM de 8-bits.
- **SPI (Serial Peripheral Interface):** Es un protocolo de datos en serie síncrono utilizado por los microcontroladores para comunicarse rápidamente con uno o más dispositivos periféricos en distancias cortas. También se puede usar para la comunicación entre dos microcontroladores. (Arduino, 2018)

Con una conexión SPI, siempre hay un dispositivo maestro (generalmente un microcontrolador) que controla los dispositivos periféricos.

Según Arduino, (2018) Existen tres líneas comunes a todos los dispositivos:

- **MISO (Master In Slave Out)** - La línea esclava para enviar datos al maestro,

- **MOSI (Master Out Slave In)** - La línea Master para enviar datos a los periféricos,
- **SCK (Serial Clock)** - Los pulsos de reloj que sincronizan la transmisión de datos generada por el maestro y una línea específica para cada dispositivo:
- **SS (Slave Select):** el pin en cada dispositivo que el maestro puede usar para habilitar y deshabilitar dispositivos específicos.  
 Cuando el pin de Selección de esclavo de un dispositivo está en bajo, se comunica con el maestro. Cuando está en alto, ignora al maestro.  
 El Nano arduino posee los Pines 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).
- **LED:** Hay un LED integrado conectado al pin digital 13. Cuando el pin tiene un valor ALTO, el LED está encendido, cuando el pin está BAJO, está apagado

El Nano posee 8 entradas analógicas, cada una de las cuales proporciona 10 bits de resolución (es decir, 1024 valores diferentes). Por defecto, miden desde tierra a 5 voltios. (PatagoniaTec, 2014).

Algunos pines tienen una funcionalidad especializada:

- **I2C:** La principal característica de I2C es que utiliza dos líneas para transmitir la información: una para los datos y otra para la señal de reloj. Esto significa que el maestro y el esclavo envían datos por el mismo cable, el cual es controlado por el maestro, que crea la señal de reloj. I2C no utiliza selección de esclavo, sino direccionamiento.

Los pines de datos y reloj 4 (SDA) y 5 (SCL). Admite la comunicación I2C Two Wire Interface (TWI). Otros pines:

- **AREF.** Tensión de referencia por las entradas analógicas.
- **Reset.** Al encontrarse la línea en nivel bajo se resetea el microcontrolador.

El Nano arduino posee la capacidad de comunicarse con un computador, otro Arduino u otro microcontrolador. El microcontrolador ATmega328P

proporciona comunicación serial UART (Transmisor-receptor Asíncrono Universal) TTL (5V), a través de los pines TX 1 y RX 0. La transmisión asíncrona permite transmitir datos sin que el emisor tenga que enviar una señal de reloj al receptor. En cambio, el emisor y el receptor deben acordar previamente bits especiales y se agregan a cada palabra que se usa para sincronizar las unidades de envío y recepción.

En UART para transmisiones asíncronas se utilizan, bits adicionales llamados "Bit de inicio", "Bit de parada", "Bit de paridad", que son agregados al byte que se va a transmitir. El bit de inicio se usa para alertar al receptor de que está a punto de enviarse una palabra de datos. El Nano Arduino posee también un FTDI FT232RL en la placa que canaliza esta comunicación en serie a través de USB y los controladores FTDI proporcionan un puerto virtual para el software en la computadora. (PatagoniaTec, 2014)

### **3.5.2. Microcontrolador ATmega328P**

El microcontrolador de alto rendimiento Microchip picoPower 8 bits basado en AVR RISC combina 32 KB de memoria flash ISP con capacidades de lectura y escritura, EEPROM 1024B, SRAM de 2 KB, 23 líneas de E / S de uso general, 32 registros de trabajo de propósito general, tres temporizadores flexibles contadores con modos de comparación, interrupciones internas y externas, USART serie programable, puerto serie SPI, un convertidor A / D, temporizador de vigilancia programable con oscilador interno, este dispositivo funciona entre 1.8 a 5.5 voltios

Al ejecutar instrucciones potentes en un solo ciclo de reloj, el dispositivo logra rendimientos aproximados a 1 MIPS (Millón de instrucciones por segundo) por MHz, equilibrando el consumo de potencia y la velocidad de procesamiento, en la figura 3.21 se observa la disposición de los pines del microcontrolador ATmega 328p.

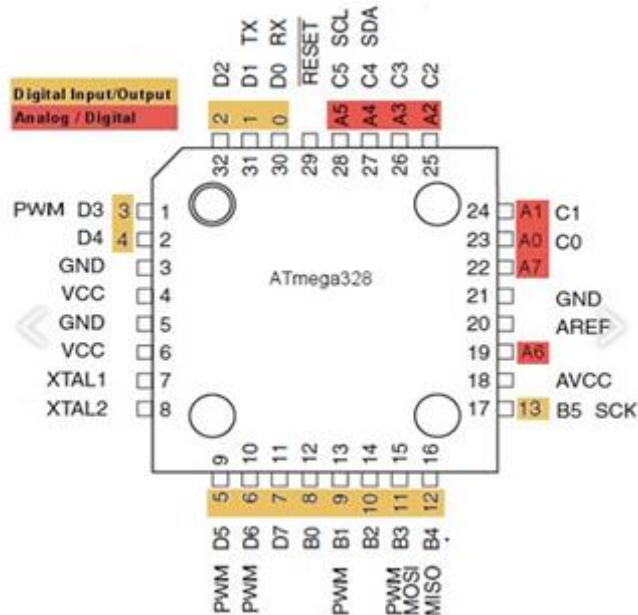


Figura 3.21: Pines de E/S del microcontrolador ATmega 328p  
Fuente:("Dan Chen Lab", 2018)

### 3.5.3. Placa controladora ArduPRO

Es el mejor controlador comercial para proyectos de robots basados en Arduino, en especial robots de Sumo, esto es debido a varias razones algunas de ellas son que posee un modo de conmutación regulador de 5 voltios de alta eficiencia., su voltaje de entrada es de 7V a 32V, el regulador tiene una pequeña protección de corriente y puede dar un total de 500 mA de salida.

Entre las características del ArduPRO se tiene un botón de inicio o (Start Input) para los módulos de inicio conectado al Pin D4, un LED de usuario conectado a pin D13, un Interruptor MOSFET de nivel lógico de canal N conectado al pin A4, que puede ser utilizado para cambiar de mecanismos, o para banderas (permite el paso de corriente), posee dos Trimpot (pequeños potenciómetros que se usan para ajuste y calibración de circuitos), conectados a los pines A6-A7, también posee tres Dipswitch conectados a los pines D0-D1-D2 que se pueden usar para la selección de la táctica (programación) del robot de sumo, tiene 2 puentes de selección Vin y 5V: se utilizan para la selección de voltaje del terminal. (Voltaje de entrada o 5 V regulados).



ArduPRO tiene un total de 13 pines libres que se pueden usar para cualquier propósito. (Arduino Nano tiene un total de 22 E / S, 9 pin se usa dentro del controlador). En la figura 3.22 se muestra el esquema de conexión de los elementos del robot en sus respectivos puertos.

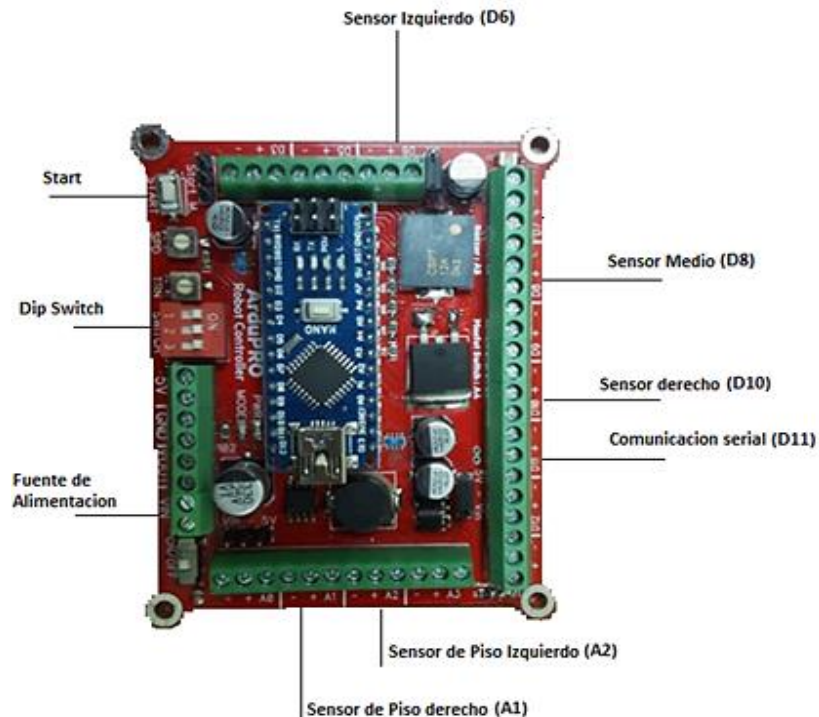


Figura 3.22: Esquema descriptivo de los elementos del robot conectados al ArduPRO

Elaborado por: Autor

### 3.5.4. Sabertooth 2x32

Es un controlador de motor de doble canal muy utilizado en el mundo de la robótica, como en robots de combate de hasta 45 kg, robots móviles y vehículos electrónicos, ya que es capaz de entregar 32 amperios a dos motores, con corrientes máximas de pico de hasta 64 amperios por motor y permite ser operado desde radiocontrol, analógico, TTL serial (lógica transistor a transistor serial) o entradas USB. En la figura 3.23 se muestra un diagrama de la Sabertooth 2x32 y sus puertos de conexión. Entre las características principales de este controlador se encuentran:

- 6-30V nominal
- 33.6V máximo absoluto.
- Accionamiento regenerativo síncrono
- Frecuencia de conmutación de 30 kHz



- Protección térmica y contra sobrecorriente
- Modo de protección de la batería
- Control y monitoreo USB
- Salidas de motor de 12 bits
- Límite de corriente configurable
- 4 modos de operación
- Led de error

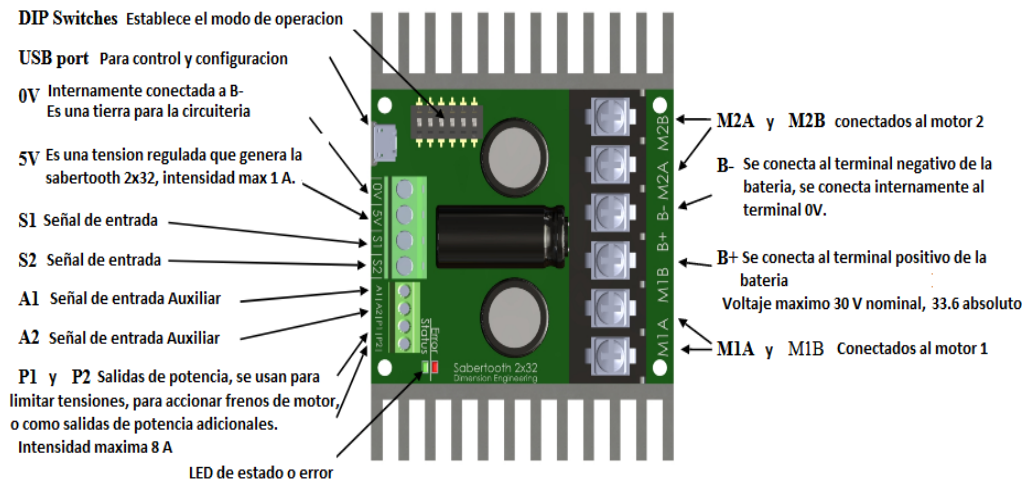


Figura 3.23: Conexiones de la placa Sabertooth 2x32

Fuente: (Dimension Engineering, 2018)

### - Terminales de la batería

B+ y B- Son conectados a los terminales respectivos de la batería.

### - Terminales de motores

Terminales de conexión M1A y M1B se conectan al motor 1 y los terminales de conexión M2A y M2B se conectan al motor 2.

### - Terminal de alimentación

Se encuentran los terminales de tensión de 5 voltios generados por la Sabertooth que proporciona energía para el receptor o hasta 4 servos analógicos estándar, y la tierra (0V), que se conecta a la tierra del dispositivo que envía señales para la correcta recepción de las señales de entrada.

### - Terminal de entrada de señales.

Los terminales S1 y S2, reciben las señales de entrada que controlan la velocidad y dirección de los motores.

La Sabertooth 2x32 es un controlador regenerativo síncrono, es decir que en el momento en que el robot frene o haga un cambio de giro, las baterías se recargarán, a esto se le llama la topología regenerativa. Este controlador de motor altamente flexible y configurable, admite varios modos de funcionamiento. La configuración básica se realiza mediante el uso de los interruptores DIP.

En el modo de operación R/C de la Sabertooth se utilizan pulsos de radiocontrol para enviar comandos a la controladora del motor (estas señales son generadas por el transmisor y recibidas por el receptor de radiocontrol conectado a la sabertooth). En la figura 3.24 se muestra la configuración de los DIP switch para utilizar este modo.

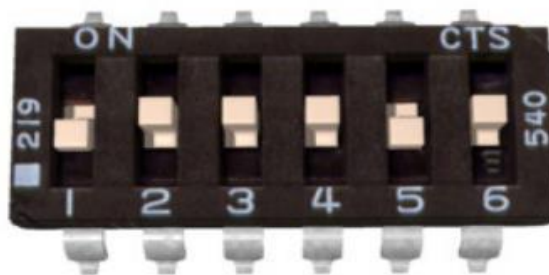


Figura 3.24: Modo de operación en R/C  
Elaborado por: Autor

El modo microcontrolador es necesario el uso de la programación que será cargada en el Nano Arduino. En el modo de microcontrolador, los rangos de pulsos de entrada son fijos. Se detiene cuando se recibe señal de 1500us, a 1000us está da marcha atrás y 2000us va hacia adelante adelante. En la figura 3.25 se muestra la posición de los DIP switch para utilizar este modo.

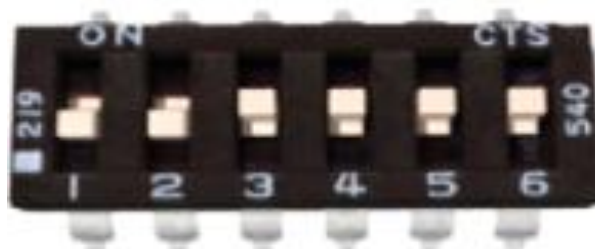


Figura 3.25: Modo de operación microcontrolador  
Elaborado por: Autor

### 3.5.5. Conexión del ArduPRO y Sabertooth en el robot Mega Sumo.

En la figura 3.26 se observa al ArduPRO y la Sabertooth la cual tiene la configuración de los Dip switch para comunicación serial, en donde se va a comunicar a través de sus puertos tierra y S1, siendo el puerto de ArduPRO el pin 11 por donde se envían los respectivos pulsos de control.



**Figura 3.26:** ArduPRO + Sabertooth  
Elaborado por: Autor

### 3.5.6. Explicación del funcionamiento actual del Robot Mega Sumo

El robot Mega-Sumo está diseñado y programado para ejecutar una táctica de pelea según el estado de los Dip-switch's del ArduPRO en que se encuentren, cumpliendo primeramente el tiempo de espera que es de 5 segundos después de presionar el botón de inicio, una vez hecho esto el robot procede a ejecutar las acciones y movimientos pre programados. Es decir que cuando alguno de sus sensores detecte algún cambio o envíen alguna señal a los puertos de entrada del microcontrolador, este se encargara de comunicarse con la Sabertooth mediante pulsos, para mover sus motores de acuerdo con su programación.

El algoritmo de programación del robot resulta ser muy simple como para desempeñar un óptimo funcionamiento de su sistema, siendo este el de atacar directamente de frente sin ningún tipo de técnica llevando a una competencia puramente de fuerza, en donde prácticamente solo se usan sus motores, en donde los sensores son dejados en segundo plano, solo para cuando se

perdida de vista al oponente y direccionarse para ejercer nuevamente fuerza sobre él y sacarlo del ring como se muestra en la figura 3.27.

```

Start:
if (digitalRead(Button) == HIGH) {
  if (analogRead(LQtr) < 150 && analogRead(RQtr) > 150) {
    ST.motor(1, -127);
    ST.motor(2, -127);
    delay(100);
    ST.motor(2, 127);
    ST.motor(1, -127);
    delay(50);
  } else if (analogRead(LQtr) > 150 && analogRead(RQtr) < 150) {
    ST.motor(1, -127);
    ST.motor(2, -127);
    delay(100);
    ST.motor(2, -127);
    ST.motor(1, 127);
    delay(50);
  } else if (digitalRead(LSens) == HIGH && digitalRead(MSens) == LOW && digitalRead(RSens) == HIGH) {
    ST.motor(2, 127);
    ST.motor(1, 127);
  } else if (digitalRead(LSens) == LOW && digitalRead(MSens) == HIGH && digitalRead(RSens) == HIGH) {
    ST.motor(2, 127);
    ST.motor(1, -127);
  } else if (digitalRead(LSens) == HIGH && digitalRead(MSens) == HIGH && digitalRead(RSens) == LOW) {
    ST.motor(2, -127);
    ST.motor(1, 127);
  } else {
    ST.motor(1, 127);
    ST.motor(2, 127);
  }
} else {
  stopp:
  ST.motor(1, 0);
  ST.motor(2, 0);
  .....
}

```

Figura 3. 27: Algoritmo de programación actual  
Elaborado por: Autor

### 3.5.7. Conexión de RC actual

Este radiocontrol se divide en dos partes, el transmisor y el módulo de recepción de las cuales entre el transmisor y el receptor van a tener su protocolo de comunicación que en este caso es el AFHDS y el receptor que se encargara de transformar la información recibida en pulsos, en la figura 3.28 se observa un esquema general de lo antes mencionado.

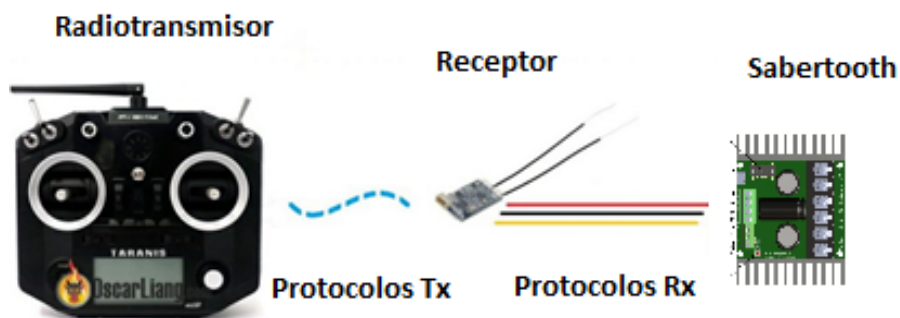


Figura 3.28: Esquema de protocolos utilizados en R/C  
Fuente: (Liang Oscar, 2017)

El módulo de recepción va directamente conectado a la Sabertooth con la cual se comunicará mediante pulsos que se encargan de controlar la rotación de los motores en ambas direcciones (Hacia adelante e inversamente), a continuación, se muestra la figura 3.29, en donde se observan las conexiones de la Sabertooth con el receptor de RC.

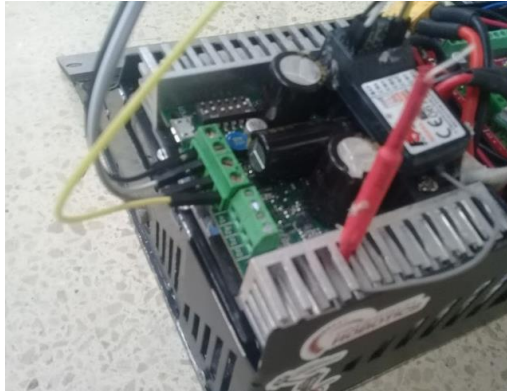


Figura 3.29: Conexión Sabertooth y el receptor RC.  
Elaborado por: Autor

### 3.6. Diseño propuesto.

Para el diseño del circuito electrónico se pensó en integrar ambos sistemas, el de control de motores y control del funcionamiento del Mega-sumo, siendo de gran utilidad el estudio de artículos científicos y proyectos relacionados con la robótica móvil. El diseño de esta tarjeta controladora está basado en el uso del microcontrolador ATmega 32u4 ya que reúne los requisitos necesarios como: Ciclo de trabajo de hasta 16 MIPS, Pines E/S para diferentes propósitos dependiendo del puerto, salidas de PWM para controlar los motores, y la información necesaria para su programación, mediante el uso de manuales de la firma Microchip

Se tomó en cuenta los demás elementos del robot que son fundamentales para un buen desempeño como los sensores que se van a utilizar, los componentes electrónicos necesarios para su correcto funcionamiento, las etapas de regulación de voltaje, pero principalmente la potencia de los motores y los niveles de voltaje con los que trabajan y los picos de corriente que genera su accionamiento ya que este es un factor clave para mejorar su eficacia, además de la programación.

### 3.6.1. Microcontrolador ATmega32u4

El microcontrolador de 8 bits basado en AVR RISC de bajo consumo con memoria de programa de auto-programación de 32KB, SRAM de 2.5KB, EEPROM de 1KB, 2 temporizadores/contadores de 8 Bit y 2 de 16 Bit, los cuales permiten contar ciclos de reloj interno o de eventos externos para generar una interrupción cuando se alcanza un valor de conteo específico. Conectividad mediante I2C, SPI, UART/USART, USB. Posee 12 transductores A/D de 10 Bit. 7 de PWM. El dispositivo alcanza un rendimiento de hasta 16 MIPS a 16 MHz, y funciona con valores de 2.7 a 5.5 voltios.

Al ejecutar instrucciones potentes en un solo ciclo de reloj, el dispositivo logra rendimientos cercanos a 1 MIPS por MHz, lo que le permite optimizar el consumo de energía en comparación con la velocidad de procesamiento. En la figura 3.30 se observa la disposición de los pines del ATmega 32U4.

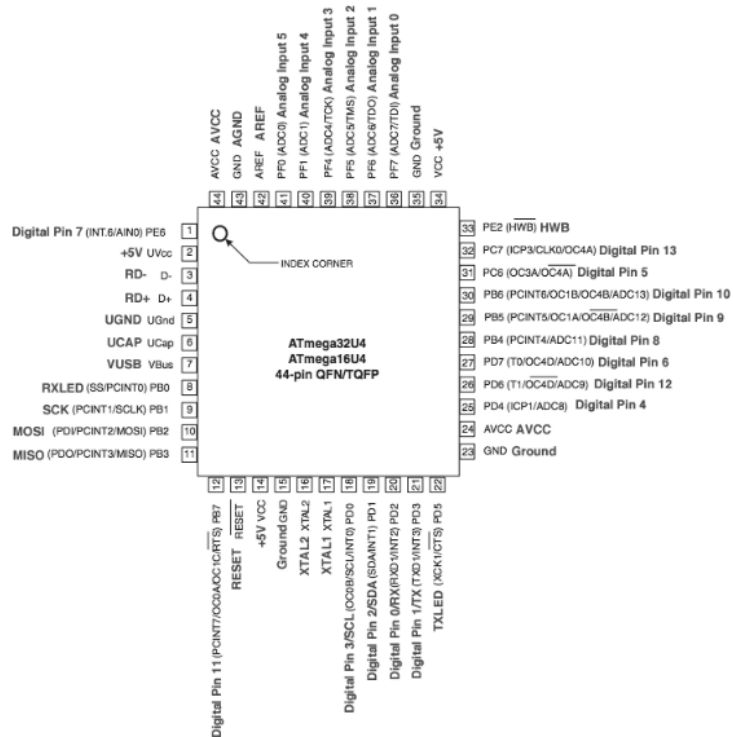


Figura 3.30: Disposición de los pines del ATmega 32U4  
Fuente: (ATmega 32U4-Arduino Pin Mapping, 2018)

### 3.6.2. Optocoplador

Es un dispositivo electrónico que está diseñado para transferir señales eléctricas mediante el uso de ondas de luz para proporcionar acoplamiento con aislamiento eléctrico entre su entrada y salida. El objetivo principal de un

optoacoplador es evitar que los altos voltajes en un lado del circuito distorsionen las transmisiones o dañen los componentes en el otro lado del circuito. En la figura 3.31 se observa el esquema general de un circuito optoacoplador.

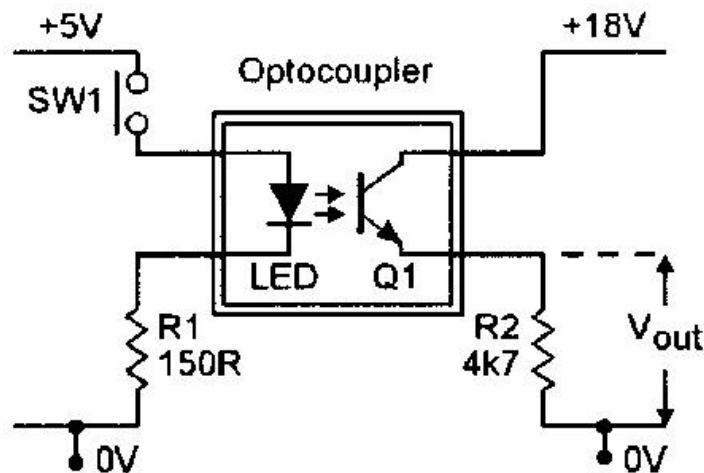


Figura 3.31: Forma básica y circuito de aplicación de un optoacoplador típico  
Fuente: (NUTS & VOLTS MAGAZINE, 2000)

Un optoacoplador contiene una fuente de luz a menudo cerca de un LED que convierte la señal de entrada eléctrica en luz, un canal óptico cerrado y un fotosensor que detecta la luz entrante y modula la corriente eléctrica que fluye de una fuente de alimentación externa o genera energía eléctrica directamente. El sensor puede ser un fotoresistor, un rectificador controlado con silicio, un fotodiodo, un fototransistor.

### 3.6.3. Puente H

Es un circuito electrónico que permite el giro de un motor dc en ambos sentidos (en sentido horario u opuesto). Un puente H es un circuito bastante simple, que contiene cuatro elementos de conmutación, con la carga en el centro, en una configuración tipo H, como se observa en la figura 3.32.

Los elementos de conmutación (Q1.Q4) suelen ser transistores bipolares o FET. Los diodos (D1.D4) se llaman diodos de captura. El extremo superior del puente está conectado a una fuente de alimentación (por ejemplo, la batería) y el extremo inferior está conectado a tierra.

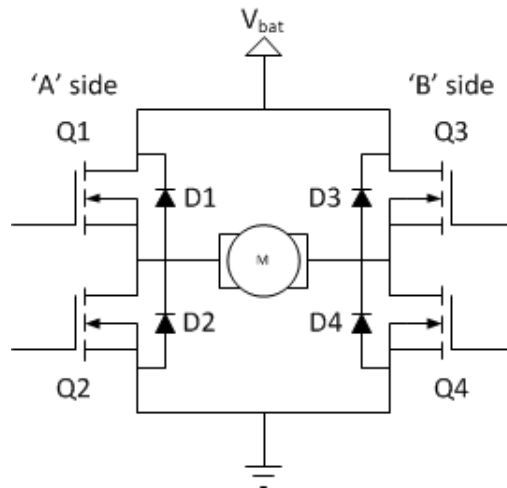


Figura 3.32: Diagrama de un puente H  
 Fuente: (“H-Bridges – the Basics | Modular Circuits”, 2012)

El modo de funcionamiento básico de un puente H es bastante simple: si Q1 y Q4 están encendidos, el cable izquierdo del motor se conectará a la fuente de alimentación, mientras que el cable derecho se conectará a tierra. La corriente comienza a fluir a través del motor que energiza el motor en la dirección de avance y el eje del motor comienza a girar, como se observa en la figura 3.33

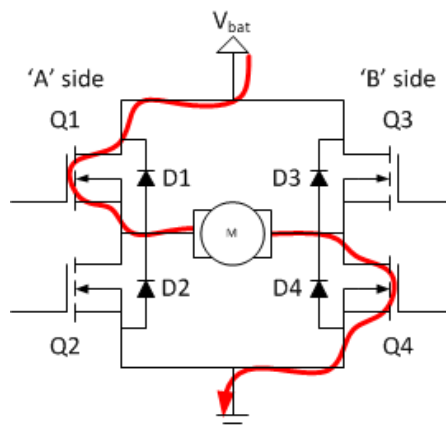


Figura 3. 33: Diagrama de un puente H (Q1 y Q4)  
 Fuente: (“H-Bridges – the Basics | Modular Circuits”, 2012)

Si Q2 y Q3 están encendidos, ocurrirá lo contrario, el motor se energizará en la dirección inversa y el eje comenzará a girar hacia atrás, como se observa en la figura 3.34. Para la etapa de control de motores (Puente H) se utilizará el BTN8982TA que técnicamente es un medio puente integrado de alta corriente para aplicaciones de accionamiento de motor.



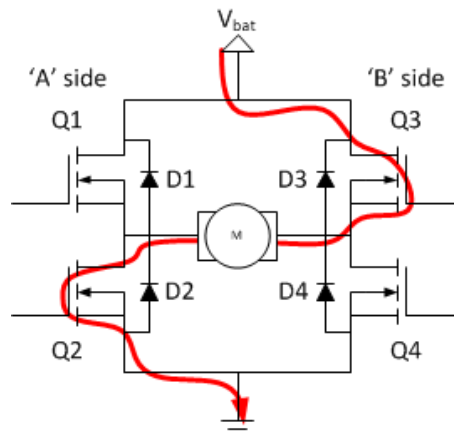


Figura 3. 34: Diagrama de un puente H (Q2 y Q3)  
 Fuente: (“H-Bridges – the Basics | Modular Circuits”, 2012)

### 3.6.4. Simulación en Proteus del funcionamiento de un motor dc utilizando ATmega32u4

Se realizó una simulación del funcionamiento de los motores y el microcontrolador, como se observa en la figura 3.35, para confirmar que las entradas funcionen como se espera, previo a su implementación en el diseño electrónico, al recibir el microcontrolador una señal de entrada, este respondía con el accionamiento de sus motores, hacia adelante o hacia atrás por medio del puente H, según la señal que recibía.

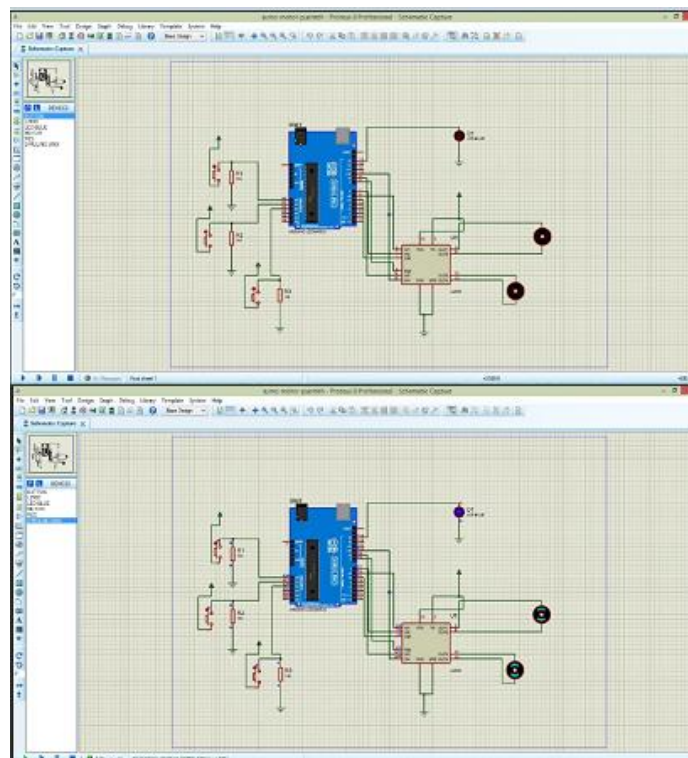


Figura 3.35: Simulación del funcionamiento del motor dc  
 Elaborado por: Autor.

### 3.6.5. Diseño de la tarjeta electrónica

En la figura 3.36, se muestra el esquemático general del diseño de la tarjeta elaborado en el programa Altium Designer ya que es un software de automatización de diseño de PCB y electrónica ideal para la elaboración de tarjetas de circuitos impresos. En la figura 3.37 se muestra su diseño final.

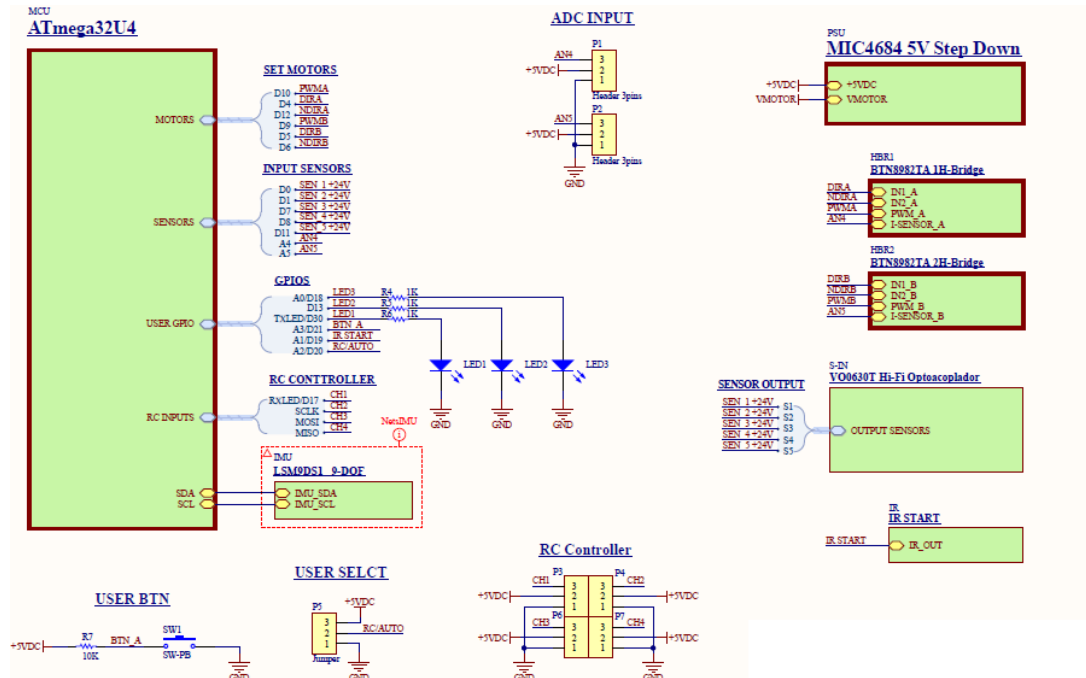


Figura 3. 36: Esquema general del diseño de la tarjeta  
Elaborado por: Autor

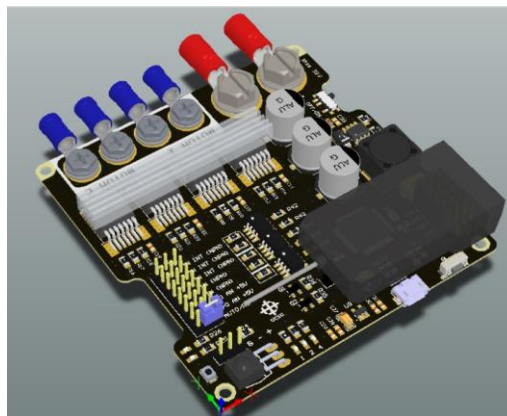


Figura 3.37: Diseño de la tarjeta controladora en Altium Designer  
Elaborado por: Autor

A continuación, se describen los componentes del esquema general del diseño de la tarjeta.

-El MIC4684 es un convertidor DC (DC Buck principal) el cual convertirá el voltaje de entrada que le brinda la fuente de alimentación (batería) en 5V de salida, los cuales alimenta a todos los dispositivos en la tarjeta que lo requieran, Microcontroladores: Atmega32U4 y Pic12F683, Receptor del Radio Control, Optoacopladores, etc. En la figura 3.38 se muestra los componentes de este bloque, y en la figura 3.39 se observa la ubicación del mismo en la tarjeta.

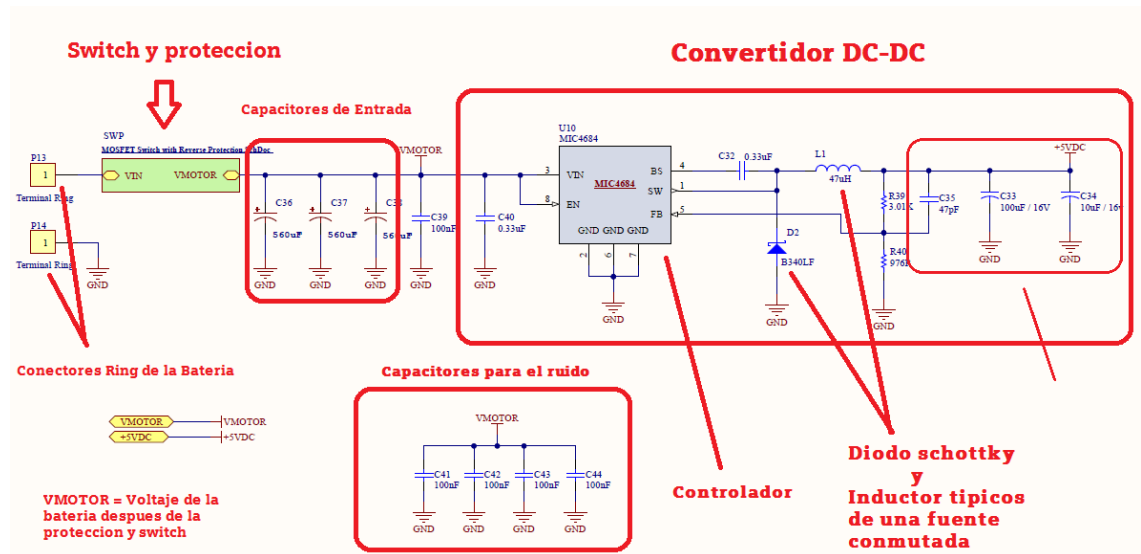


Figura 3.38: Bloque del MIC4684  
Elaborado por: Autor.

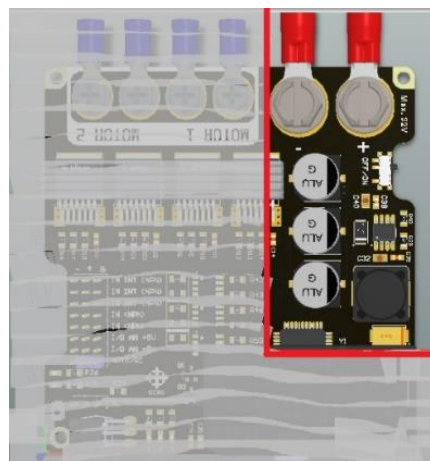


Figura 3. 39: Bloque de regulación en la tarjeta.  
Elaborado por: Autor.

-Atmega32U4 (Basado en el Arduino Leonardo), cerebro principal de la tarjeta, administra y gestiona a cada uno de los periféricos que posee la tarjeta: Leds, Drivers para los Motores, Sensores, Receptor del Radiocontrol,

IMU (Unidad de movimiento Inercial), botonera, USB, etc. En la figura 3.40 se observa los pines a los cuales irán conectados los diferentes elementos del robot y en la figura 3.41 la ubicación del microcontrolador en la tarjeta.

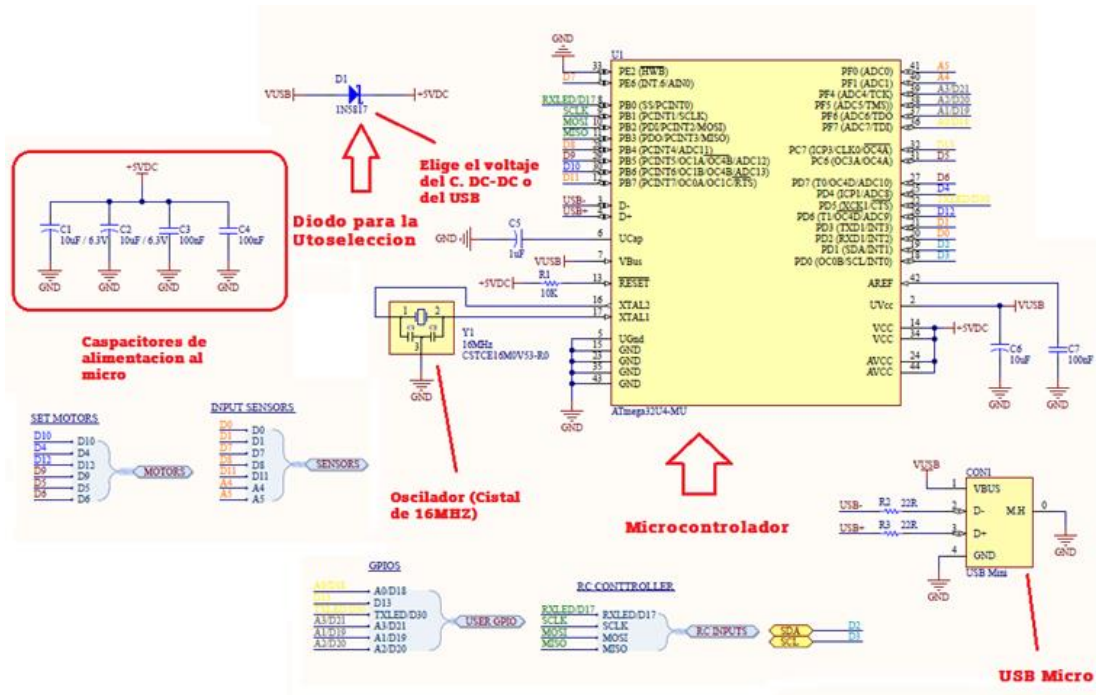


Figura 3.40: Esquema de conexión del ATmega32U4  
Elaborado por: Autor.

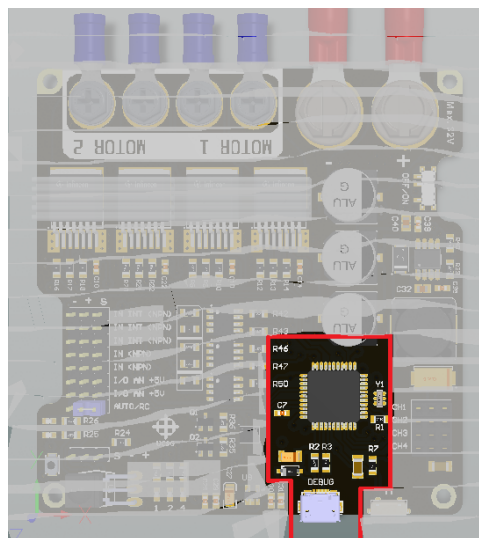


Figura 3.41: Microcontrolador ATmega32U4 en la tarjeta.  
Elaborado por: Autor.

-Drivers de Motores de Corriente Continua, el BTN8982 es un controlador de motores DC de alta corriente pero un solo dispositivo no permitiría la Inversión de giro del motor (High Current PN Half Bridge), es

medio puente H pero si se suma otro en conjunto formarían un puente H completo, pudiendo controlar la Velocidad del motor y sentido de giro. En la figura 3.42 se muestra el esquema del bloque y sus componentes, y la figura 3.43 la ubicación del mismo en la tarjeta.

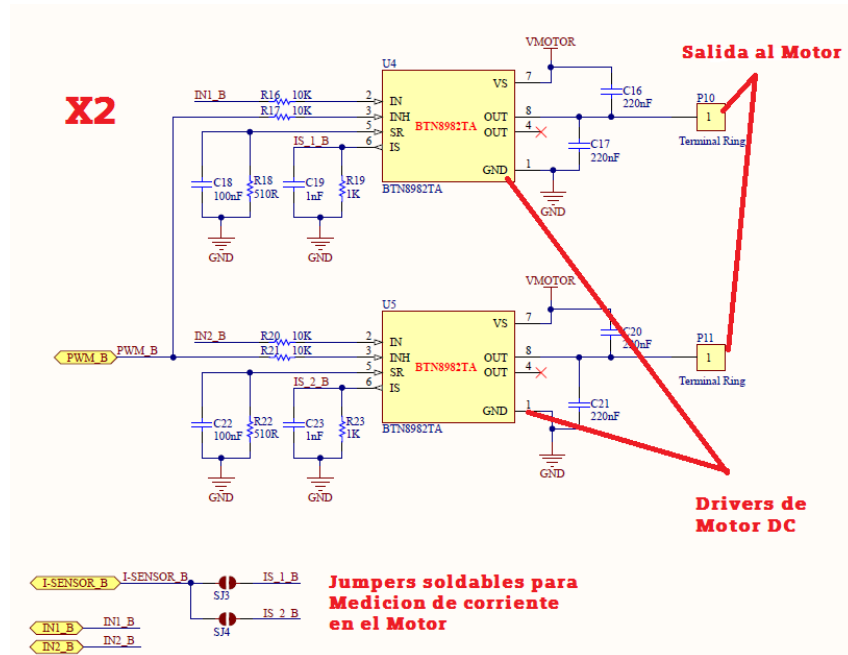


Figura 3.42: Bloque de Driver de motor DC  
Elaborado por: Autor.

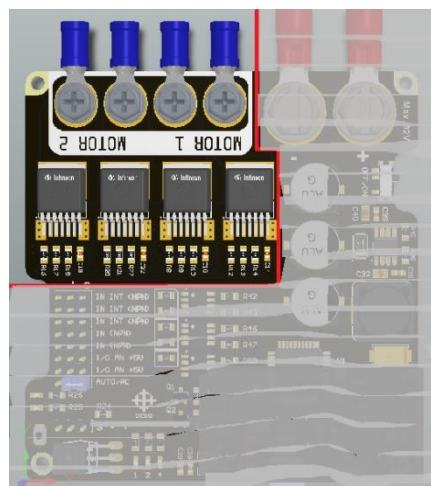


Figura 3. 43: Bloque de driver en la tarjeta.  
Elaborado por: Autor

-Acondicionamiento para sensores de salida NPN de Colector Abierto que funcionen entre 5V y 24V. Se utilizan optoacopladores para acoplar la salida de los sensores a la entrada del microcontrolador Atmetmega32U4. En

la figura 3.44, se muestra el bloque de los optocopladores, y en la figura 3.45, su ubicación en la tarjeta.

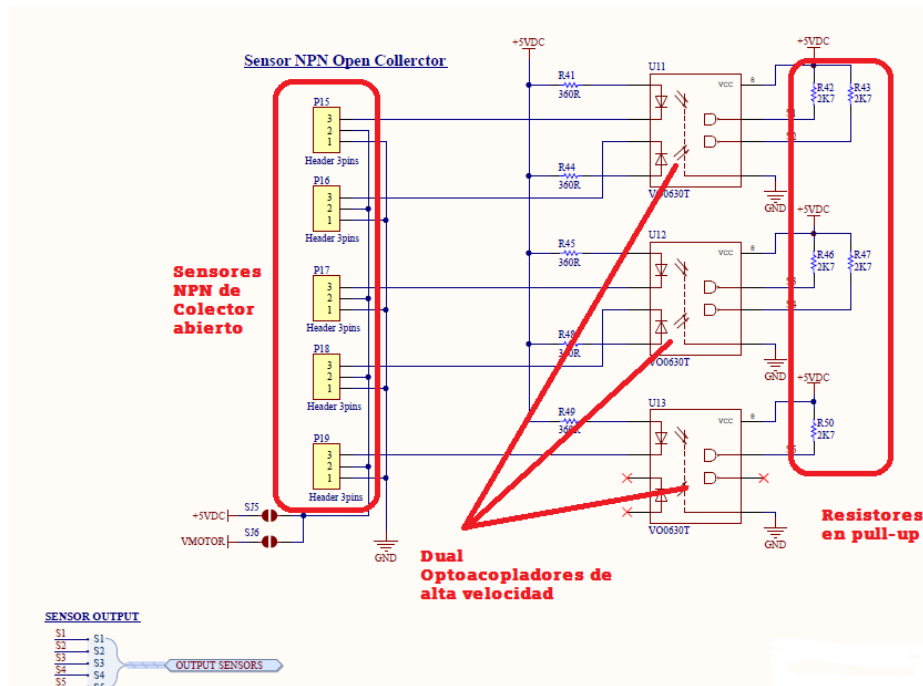


Figura 3.44: Bloque de optocopladores.  
Elaborado por: Autor

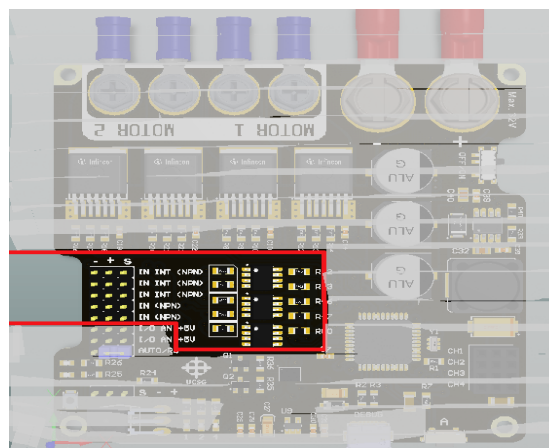


Figura 3.45: Ubicación de Optocopladores en la tarjeta y puertos de conexión de sensores.  
Elaborado por: Autor

-IMU (Unidad de Movimiento Inercial), el LSM9DS1 posee 9 grados de Libertas (9DOF): 3ejes de Acelerómetro, 3ejes de Giroscopio y 3 ejes de Magnetómetro, con este dispositivo se puede posicionar el robot. En la figura 3.46 se muestra el esquema de este bloque, y en la figura 3.47 la ubicación de este bloque en la tarjeta.



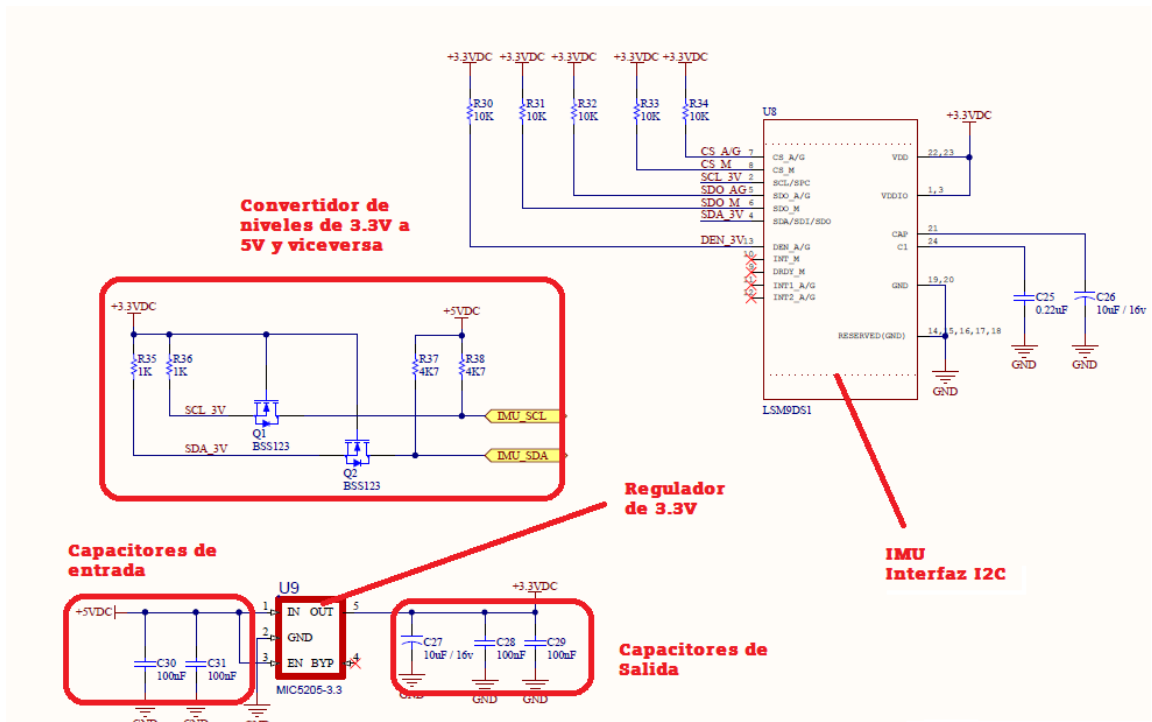


Figura 3.46: Bloque LSM9DS1  
Elaborado por: Autor.

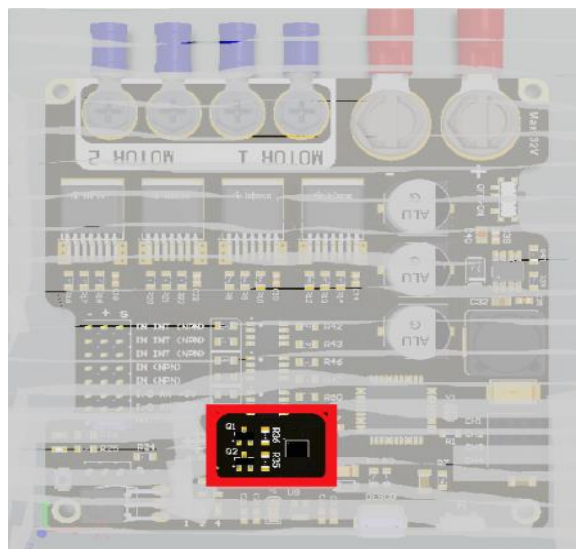


Figura 3.47: LSM9DS1 en la tarjeta  
Elaborado por: Autor.

-Receptor infrarrojo para iniciar y detener el robot, cuenta con un receptor de 40Khz y un PIC12f683 que gestiona y guarda los botones del control remoto luego envía una señal digital al microcontrolador principal el Atmega32U4. En la figura 3.48, se muestra el bloque del receptor infrarrojo, y en la figura 3.49 su ubicación en la tarjeta.

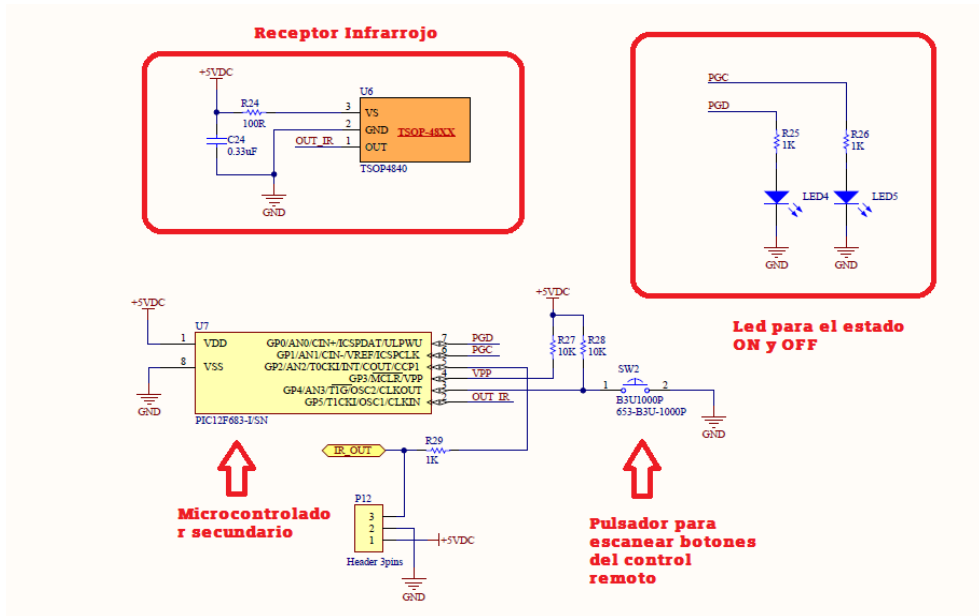


Figura 3.48: Bloque del receptor infrarrojo.  
Elaborado por: Autor

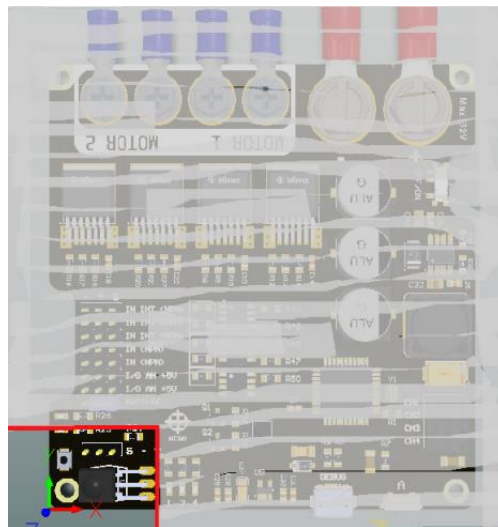


Figura 3.49: Receptor infrarrojo en la tarjeta  
Elaborado por: Autor

### 3.6.6. Placa de circuito impreso y robot ensamblado con la tarjeta diseñada.

En la figura 3.50, se muestra el diseño de la placa de circuito impreso, elaborado en el programa Altium Designer, y en la figura 3.51 el robot ensamblado con la tarjeta diseñada.



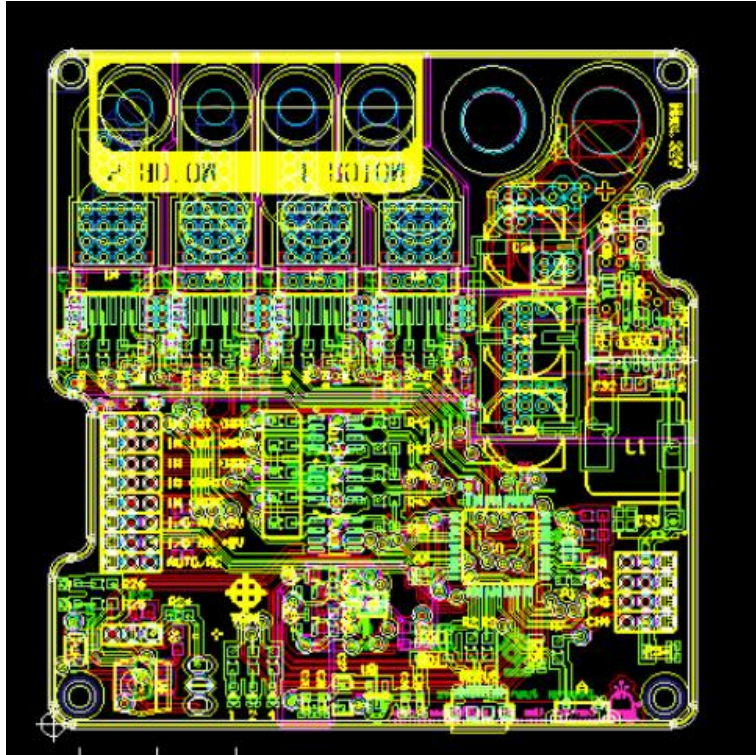


Figura 3.50: Placa de circuito impreso  
Elaborado por: Autor



Figura 3. 51: Robot ensamblado con la tarjeta diseñada  
Elaborado por: Autor

### 3.6.7. Explicación de la programación y funcionamiento del Robot Mega Sumo con la tarjeta de diseño propio.

La plataforma de programación es Arduino IDE, debido a que es un entorno práctico para poder establecer los algoritmos de programación.

A continuación se presenta programación de la tarjeta de diseño propio:

```
MegaSumoCompleto
#define NDIRA 12
#define DIRA 4
#define PWMA 10

#define NDIRB 6
#define DIRB 5
#define PWMB 9
```

Figura 3.52: Pines de control de motores.  
Elaborado por: Autor

En la figura 3.52 se observa los pines de control de motores en donde “DIR y NDIR” son las señales digitales de control de dirección y según sus valores se moverán hacia adelante o en dirección opuesta como se observa en la figura 3.53. PWM Son las Señales de Control de velocidad.

DIR	NDIR	Motor Salida
0	0	Stop
0	1	forward
1	0	Backforward
1	1	Stop Break

Figura 3.53: Dirección de Motores  
Elaborado por: Autor.

En la figura 3.54, el Pin 19 es para la entrada Digital de Inicio del robot, el Pin 21 entrada Botonera selección de estrategia, el Pin 20 entrada seleccionar Modo Radiocontrol o Autónomo, #define NUM\_CHANEL 3 (Número de canales del receptor), numero de bits de los Leds de # de estrategia, Pin 0 entrada Sensor de la Izquierda, Pin 1 entrada Sensor de la Derecha, Pin 8 entrada Sensor de la De Piso Derecho, Pin 11 entrada Sensor de la de Piso Izquierdo. En la figura 3.55 se observa la declaración de variables, el máximo número de estrategias que es 7, y en la figura 3.56 la configuración de entradas y salidas.

```

#define START 19
#define BTN_A 21
#define RC_AUTO 20
#define NUM_CHANEL 3 // Numero de canales del receptor
#define nBits sizeof(ledPins)/sizeof(ledPins[0])
#define SensorIZQ 0
#define SensorDER 1
#define QRT_1 8
#define QRT_2 11

```

Figura 3.54: Pines de Entrada  
Elaborado por: Autor.

```

int ledPins[] = { 30, 13, 18 }; //Pin 30,13 y 18 Leds de # de estrategia 3bits Max 7 Estrategias del 1-7
int count = 1; // Variable que incrementa la Estrategias
int cycle = 0; // Variable de Tiempo (Se utiliza para esperar apxim. 5 segundos que se mantiene presionado el Boton A)
int Estrategia; // Variable de Estrategia
bool SelectMode; // Variable de Estado Modo RC o Autonomo
unsigned int Duty_Cycle; // Variable del FWM Ciclo de Trabajo 0-255 (se mapea valores de los canales del receptor y se guardan Aqui)
unsigned int Chanel[NUM_CHANEL+1]; // Arreglo de los canales del Receptor
unsigned long previousMillis = 0; // Variables del contador Timer para hacer giros sin perder lectura de sensores(Modo Autonomo)
bool MotorState = LOW; // Estado de Motores (Modo Autonomo)

```

Figura 3.55: Declaración de variables  
Elaborado por: Autor

```

void setup() {

    pinMode(NDIRA, OUTPUT);
    pinMode(DIRA, OUTPUT);
    pinMode(FWMA, OUTPUT);
    pinMode(NDIRB, OUTPUT);
    pinMode(DIRB, OUTPUT);
    pinMode(FWMB, OUTPUT);
    pinMode(SensorIZQ, INPUT);
    pinMode(SensorDER, INPUT);
    pinMode(QRT_1, INPUT);
    pinMode(QRT_2, INPUT);
    pinMode(17, INPUT);
    pinMode(16, INPUT);
    pinMode(15, INPUT);
    pinMode(START, INPUT);
    pinMode(BTN_A, INPUT);
    pinMode(RC_AUTO, INPUT);
    for (byte i=0; i<nBits; i++){
        pinMode(ledPins[i], OUTPUT);
    }
}

```

Figura 3.56: Configuración de entradas y salidas.  
Elaborado por: Autor

En la figura 3.57 se establece la Velocidad de Transmisión Serial (Baudrate Serial USB) para Configurar Receptor y ver valores, se llama a función LedsBinary(0) los 3 bits de los leds marcaran Cero (Los 3 Leds Apagados = OFF OFF OFF ), llamada a Función SET\_MOTORS(0, 0); Ambos Motores detenidos, se lee la entrada de modo RC y Autónomo y la guarda en SelectMode, mientras que en la figura 3.58, se observa la selección de

estrategia que va a depender de cuantas veces se presione el BTN\_A. Finalmente en la figura 3.59 se observa la secuencia de espera una vez seleccionada la opción de estrategia.

```
Serial.begin(9600);

LedsBinary(0);
SET_MOTORS(0, 0);
delay(100);
SelectMode = !digitalRead(RC_AUTO);
```

Figura 3.57: Baudrate & Set motors  
Elaborado por: Autor

```
if(SelectMode){ //
while(cycle<=500){ // Seleccion de estrategia
LedsBinary(count);
if(!digitalRead(BTN_A)){
cycle=0;
count++;
if(count>7)
count=1;
while(!digitalRead(BTN_A) && cycle<=500){
delay(10);
cycle++;
}
}
} // Fin de Seleccion de Esrtrategia
```

Figura 3.58: Selección de estrategia  
Elaborado por: Autor.

```

}
Estrategia =count-1;
LedsBinary(Estrategia);
delay(350);
LedsBinary(0);
delay(350);
LedsBinary(Estrategia);
delay(350);
LedsBinary(0);
delay(350);
LedsBinary(Estrategia);
delay(350);
LedsBinary(0);
delay(350);
LedsBinary(Estrategia);
delay(350);
while(!digitalRead(START)){ // A la espera de que presiones e boton de control IR
}
}else{ // De otro modo esta en RC
while(RCcontrollerRead()== 0){
ToggleLed();
}
}
}
}
```

Figura 3.59: Espera para ejecutar la estrategia  
Elaborado por: Autor

El robot está programado para ejecutar movimientos esquivos aleatorios (Giro hacia atrás izquierdo /derecho, zigzaguo hacia adelante izquierdo/derecho, Reversa hacia atrás y empuje frontal) al detectar que tiene un oponente al inicio de la competencia para después utilizar la estrategia. A continuación, se muestran las estrategias propuestas para la mejora de la inteligencia del robot sumo al momento de competir. En la figura 3.60 se muestra una de las estrategias esquivas.

```

if(digitalRead(QRT_1) == digitalRead(QRT_2)){
  if (!digitalRead(SensorIZQ) == !digitalRead(SensorDER))
    SET_MOTORS(80, 80);
  else if(!digitalRead(SensorDER))
    SET_MOTORS(-25, 25);
  else if(!digitalRead(SensorIZQ))
    SET_MOTORS(25, -25);
  else
    SET_MOTORS(-25, 25);
}else if(!digitalRead(QRT_1) == !digitalRead(QRT_2)){
  SET_MOTORS(-25, -25);
  delay(250);
  SET_MOTORS(-25, 25);
  delay(350);
}else if(!digitalRead(QRT_1)){
  SET_MOTORS(-50, 0);
  delay(120);
  SET_MOTORS(-25, -25);
  delay(250);
  SET_MOTORS(-25, 25);
  delay(350);
}else if(!digitalRead(QRT_2)){
  SET_MOTORS(0, -50);
  delay(120);
  SET_MOTORS(-25, -25);
  delay(250);
  SET_MOTORS(-25, 25);
  delay(350);
}
if (!digitalRead(START))
  reset();
}
break;

```

Figura 3.60: Movimiento esquivo previo inicio de estrategia.  
Elaborado por: Autor.

**-Estrategia de búsqueda aleatoria.** - El robot va por el ring a su máxima velocidad buscando al oponente de manera aleatoria, la ventaja de esta técnica es que el robot al ir a su máxima capacidad en el momento de encontrar a su contrincante su fuerza se incrementa logrando sacarlo fuera del ring. En la figura 3.61 se muestra la programación de la estrategia antes mencionada.

```

case 1: // Modo Autonomo
  switch (Estrategia) { // Estrategias
    case 1: // 1) Búsqueda Aleatoria
      while(1){
        if(digitalRead(QRT_1) == digitalRead(QRT_2)){
          if (!digitalRead(SensorIZQ) == !digitalRead(SensorDER))
            SET_MOTORS(80, 80);
          else if(!digitalRead(SensorDER))
            SET_MOTORS(-25, 25);
          else if(!digitalRead(SensorIZQ))
            SET_MOTORS(25, -25);
          else{
            SET_MOTORS(18, 25);
          }
        }else if(!digitalRead(QRT_1) || !digitalRead(QRT_2)){
          SET_MOTORS(-25, -25);
          delay(250);
          SET_MOTORS(-25, 25);
          delay(350);
        }
        if (!digitalRead(START))
          reset();
      }
      break;

```

Figura 3.61: Búsqueda Aleatoria  
Elaborado por: Autor

**-Estrategia pájaro carpintero.** - El robot va a ir avanzando hacia adelante en cortos, rápidos y potentes movimientos de empuje y cuando detecte a su oponente va a ir a su máxima velocidad, esta es una técnica tanto defensiva como ofensiva ya que los pequeños y rápidos empujes impiden que el contrincante lo ataque de algún punto ciego. En la figura 3.62, se muestra la programación de esta estrategia.

**-Estrategia Búsqueda aleatoria pausada.** - En esta estrategia el robot se va a desplazar lenta y cuidadosamente hacia adelante esperando encontrar a su oponente, la ventaja de esta estrategia es que el robot está en movimiento permitiéndole una mayor aceleración que si estuviera en su posición inicial. En la figura 3.63, se muestra la programación de esta estrategia.

**-Estrategia tornado.** - En esta estrategia el robot realiza giros sobre su propio eje a muy alta velocidad, las ventajas de esta estrategia son que, si su oponente viene a atacarlo, este resulta desviado por el rápido gira del robot, para después detectar su posición pudiendo así empujarlo fuera del ring, esta



es una estrategia totalmente defensiva. En la figura 3.64 se muestra la estrategia antes mencionada.

```

case 2:                                     // 2) Pajaro Carpintero
  while(1){
    if(digitalRead(QRT_1) == digitalRead(QRT_2)){
      unsigned long currentMillis = millis();
      if (MotorState == LOW) {
        if(currentMillis - previousMillis >= 100){
          MotorState=HIGH;
          previousMillis += 100;
        }
      }
    }
    else{
      if (currentMillis - previousMillis >= 2000) {
        MotorState=LOW;
        previousMillis += 2000;
      }
    }
    if (!digitalRead(SensorIQ) == !digitalRead(SensorDER))
      SET_MOTORS(100, 100);
    else if(!digitalRead(SensorDER))
      SET_MOTORS(-25, 25);
    else if(!digitalRead(SensorIQ))
      SET_MOTORS(25, -25);
    else if(!MotorState)
      SET_MOTORS(11, 26);
    else
      SET_MOTORS(0, 0);
  }else if(!digitalRead(QRT_1) || !digitalRead(QRT_2)){
    SET_MOTORS(-25, -25);
    delay(250);
    SET_MOTORS(-25, 25);
    delay(350);
  }
  if (!digitalRead(START))
    reset();
}
break:

```

Figura 3.62: Estrategia pájaro carpintero  
Elaborado por: Autor

```

case 3:                                     // 3) Búsqueda aleatoria pausada
  while(1){
    if(digitalRead(QRT_1) == digitalRead(QRT_2)){
      if (!digitalRead(SensorIQ) == !digitalRead(SensorDER))
        SET_MOTORS(80, 80);
      else if(!digitalRead(SensorDER))
        SET_MOTORS(-25, 25);
      else if(!digitalRead(SensorIQ))
        SET_MOTORS(25, -25);
      else{
        SET_MOTORS(8, 18);
      }
    }else if(!digitalRead(QRT_1) || !digitalRead(QRT_2)){
      SET_MOTORS(-25, -25);
      delay(250);
      SET_MOTORS(-25, 25);
      delay(350);
    }
    if (!digitalRead(START))
      reset();
  }
  break:

```

Figura 3.63: Búsqueda Aleatoria pausada.  
Elaborado por: Autor

```

case 4: // 4) Tornado Derecha
  while(1){
    if(digitalRead(QRT_1) == digitalRead(QRT_2)){
      if (!digitalRead(SensorIZQ) == !digitalRead(SensorDER))
        SET_MOTORS(80, 80);
      else if(!digitalRead(SensorDER))
        SET_MOTORS(-25, 25);
      else if(!digitalRead(SensorIZQ))
        SET_MOTORS(25, -25);
      else
        SET_MOTORS(25, -25);
    }else if(!digitalRead(QRT_1) == !digitalRead(QRT_2)){
      SET_MOTORS(-25, -25);
      delay(250);
      SET_MOTORS(-25, 25);
      delay(350);
    }else if(!digitalRead(QRT_1)){
      SET_MOTORS(-50, 0);
      delay(120);
      SET_MOTORS(-25, -25);
      delay(250);
      SET_MOTORS(-25, 25);
      delay(350);
    }else if(!digitalRead(QRT_2)){
      SET_MOTORS(0, -50);
      delay(120);
      SET_MOTORS(-25, -25);
      delay(250);
      SET_MOTORS(-25, 25);
      delay(350);
    }
    if (!digitalRead(START))
      reset();
  }
  break;

```

Figura 3.64: Estrategia Tornado  
Elaborado por: Autor

### 3.6.8. Modo Radiocontrol en la tarjeta propuesta.

El modo de RC en la tarjeta va a depender del receptor, el cual se encargara de enviar los pulsos que controlaran la velocidad y dirección de los motores. En la figura 3.65 se observa la conexión del receptor en la tarjeta diseñada., mientras que en la imagen 3.66, se muestra la configuración RC en la tarjeta.



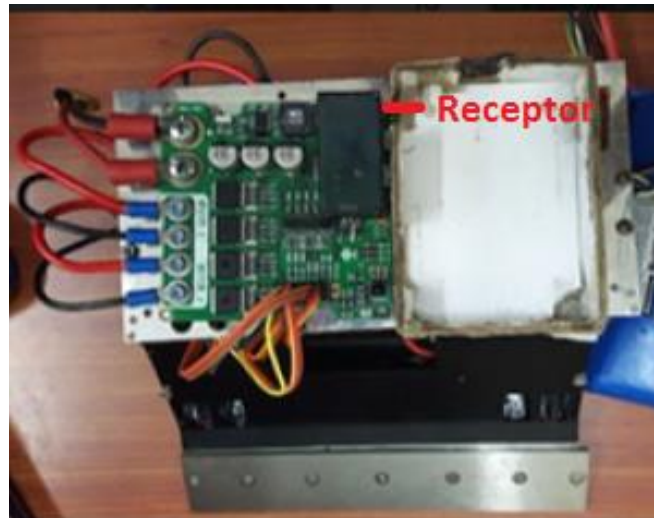


Figura 3. 65: Modulo de recepción de RC de la tarjeta controladora.  
Elaborado por: Autor.

```

void(^ reset) (void) = 0;          // Funcion para el Reset

void loop(){                      // Programa Principal

  switch (SelectMode) {          //SelectMode 0= RC y 1= Autonomo

    case 0:                      // Modo RC
      LedsBinary(4);
      switch (RCcontrollerRead()) { // Lectura del Modulo RC

        case 0:                  // Reset
          setup();
          break;

        case 1:                  // Delante Izquierda
          Duty_Cycle = map(Chanel[1], 1650, 1840, 0, 80);
          if(Duty_Cycle>80)
            Duty_Cycle=80;
          SET_MOTORS(-Duty_Cycle, Duty_Cycle);
          break;

        case 2:                  // Delante Derecha
          Duty_Cycle = 80-map(Chanel[1], 990, 1350, 0, 80);
          if(Duty_Cycle>80)
            Duty_Cycle=80;
          SET_MOTORS(Duty_Cycle, -Duty_Cycle);
          break;

        case 3:                  // Atras Izquierda
          Duty_Cycle = map(Chanel[1], 1650, 1840, 0, 80);
          if(Duty_Cycle>80)
            Duty_Cycle=80;
          SET_MOTORS(-Duty_Cycle, Duty_Cycle);
          break:
      }
    }
  }

```

Figura 3.66: Modo RC  
Elaborado por: Autor

## **CAPÍTULO 4: CONCLUSIONES Y RECOMENDACIONES.**

### **4.1. Conclusiones**

El diseño del algoritmo con sentencias de control por casos, permiten al robot realizar funciones independientes en cada estrategia diseñada, determinados por los estados de los sensores, por ende se tiene un algoritmo completo que direcciona y desplaza al robot de una forma diferente a la que los robots de concursos nacionales están acostumbrados a hacerlo, aprovechando de mejor manera sus características como la fuerza de sus motores y su capacidad de detección, dándole ventaja e incrementando sus posibilidades de ganar.

Con el diseño de la tarjeta electrónica propia se tiene la absorción de dos tecnologías de control que solamente se encuentran de forma individual en el mercado electrónico. La comunicación serial y la comunicación inalámbrica por parte de la radio frecuencia. La necesidad de tener una tarjeta controladora de motores y una tarjeta controladora de sensores es innecesaria cuando se puede tener en un solo diseño estas dos funciones.

Las estrategias de combate desarrolladas en el algoritmo proporcionan al robot sumo una cantidad más variada de técnicas para ser utilizadas según las características del robot competidor, permitiendo elegir la más adecuada con el fin de aumentar sus posibilidades de ganar en la competencia.

La tarjeta controladora posee en su driver una capacidad mayor de amperaje para los motores que la Sabertooth2x32 pudiendo dar hasta 50 A por canal, y 120 A de corriente máxima. (Casi el doble que la controladora previa), los cuales mejoran el funcionamiento del robot ya que el driver no se bloqueara cuando se genere el máximo consumo del motor, lo que ocasiona normalmente que los motores dejen de funcionar y el robot resulte empujado fuera del ring.

Menor vulnerabilidad a los impactos y golpes que se generan en las competencias de sumo debido a que todo su sistema de control se encuentra

soldado en una sola placa y no como en su funcionamiento anterior que tenía muchos cables que se desconectaban durante los golpes lo cual hacía perder al robot Sumo en la competencia.

#### **4.2. Recomendaciones**

Profundizar los conocimientos sobre microcontroladores para el desarrollo de prototipos robóticos móviles.

Realizar una correcta traducción de los Datasheet de los componentes del robot ya que muchos de estos vienen en otros idiomas, para evitar así cualquier confusión a la hora de conectar los elementos del robot a sus respectivos puertos.

Utilizar una pulsera antiestática, al menos durante el ensamblaje del robot para evitar el daño de algún componente del robot.

Al utilizar el sensor infrarrojo de distancia hay que tener en cuenta el voltaje de la señal que envía al microcontrolador para poder regularlo mediante el uso de optoacopladores.

Buscar asesoramiento temprano de los docentes de la universidad para cubrir la falta de conocimiento en cualquier área, ya sea electrónico o de programación para así no tener inconvenientes al momento de la implementación.

## REFERENCIAS BIBLIOGRÁFICA

- Ardmine. (2017). Baterías de polímero de litio | Guías Prácticas.COM. Recuperado el 4 de febrero de 2018, a partir de <http://www.guiaspracticas.com/computadoras-portatiles/baterias-de-polimero-de-litio>
- Arduino. (2018). Arduino - SPI. Recuperado el 5 de febrero de 2018, a partir de <https://www.arduino.cc/en/Reference/SPI>
- ATmega 32U4-Arduino Pin Mapping. (2018). Arduino - PinMapping32u4. Recuperado el 19 de febrero de 2018, a partir de <https://www.arduino.cc/en/Hacking/PinMapping32u4>
- Benchimol, D. (2011). Proyectos con microcontroladores aprenda a desarrollar sus propias aplicaciones.
- Castaño Welgos, J. A. (2002). Curso práctico sobre microcontroladores: teoría, programación, diseño, prácticas y proyectos completos. Pereira, Colombia: CEKIT.
- Dan Chen Lab. (2018). Pin Mapping Atmega 32U4. Recuperado el 19 de febrero de 2018, a partir de <http://danchen.work/output-devices/>
- Dimension Engineering. (2018). Sabertooth 2x32 regenerative dual motor driver. Recuperado el 19 de febrero de 2018, a partir de <https://www.dimensionengineering.com/products/sabertooth2x32>
- OMRON Industrial Automation. (2018). Sensores de reflexión difusa E3Z-D62 2M.| Recuperado el 7 de febrero de 2018, a partir de <https://www.ia.omron.com/product/item/485/>
- Education, U. T. T. (2014). KY-040 Rotary Encoder Module for Arduino | Robotic Class, STEM Project for K-12 students. Recuperado el 5 de

febrero de 2018, a partir de <http://www.utteducation.com/past-projects/ky-040-rotary-encoder-module-for-arduino/>

Electricidad Práctica: Motores Electricos. (2018). Motores Eléctricos. Recuperado el 19 de febrero de 2018, a partir de <http://autodesarrollo-electricidadpractica.blogspot.com/2011/07/motores-electricos.html>

Everett, H. R. (1995). Sensors for mobile robots: theory and application. Wellesley, Mass: A.K. Peters.

Gascón, D. (2009). 802.15.4 vs ZigBee | Libelium. Recuperado el 4 de febrero de 2018, a partir de <http://www.libelium.com/802-15-4-vs-zigbee/>

GFSK modulation - Near Communications. (2018). GFSK modulation. Recuperado el 14 de febrero de 2018, a partir de <https://sites.google.com/site/nearcommunications/gfsk-modulation>

Gotschlich, M. (2010). Remote Controls—Radio Frequency or Infrared. Infineon Technologies AG.

Modular Circuits. (2012). H-Bridges – the Basics. Recuperado el 19 de febrero de 2018, a partir de <http://www.modularcircuits.com/blog/articles/h-bridge-secrets/h-bridges-the-basics/>

HCAM-IESS. (2017). Pacientes urológicos se benefician del Da Vinci. Recuperado el 22 de enero de 2018, a partir de <http://hcam.iess.gob.ec/?p=4662>

IEEE Computer Society, LAN/MAN Standards Committee, Institute of Electrical and Electronics Engineers, & IEEE-SA Standards Board. (2011). IEEE standard for local and metropolitan area networks. Part 15.4, Part 15.4,. New York: Institute of Electrical and Electronics Engineers. Recuperado a partir de <http://ieeexplore.ieee.org/servlet/opac?punumber=6012485>

- Infineon Technologies AG. (2018). BTN8982TA - Infineon Technologies. Recuperado el 19 de febrero de 2018, a partir de <https://www.infineon.com/cms/en/product/power/motor-control-ics/intelligent-motor-control-ics/single-half-bridge-driver/btn8982ta/>
- Jsumo. (2018). MicroStart Sumo & Minisumo Robot Start Module Robot Start Module Jsumo | JSumo.com. Recuperado el 7 de febrero de 2018, a partir de <http://www.jsumo.com/microstart-sumo-minisumo-robot-start-module>
- Liang Oscar. (2017). RC TX RX Protocols Explained: PWM, PPM, SBUS, DSM2, DSMX, SUMD. Recuperado el 28 de enero de 2018, a partir de <https://oscarliang.com/pwm-ppm-sbus-dsm2-dsmx-sumd-difference/>
- Marcelo Villalobos, R. W. (2014). Fundamentos de programación C#: Mas de 100 algoritmos codificados. Perú: Editorial Macro.
- Maxon motor. (2018). maxon motor - Online Shop. Recuperado el 6 de febrero de 2018, a partir de <https://www.maxonmotor.com/maxon/view/product/273752>
- Nuts & volts magazine. (2000). Optocoupler Circuits. Recuperado el 19 de febrero de 2018, a partir de <http://www.nutsvolts.com/magazine/article/optocoupler-circuits>
- Oscar Liang. (2013). PWM and PPM Difference and Conversion. Recuperado el 14 de febrero de 2018, a partir de <https://oscarliang.com/pwm-ppm-difference-conversion/>
- PatagoniaTec. (2014). Arduino Nano |. Recuperado el 5 de febrero de 2018, a partir de <http://saber.patagoniatec.com/arduino-nano-328-arduino-atmega-clon-compatible-arduino-argentina-ptec/>

Plúa, A, & Castillo, J. (2015). Implementación de un Robot Mega Sumo para participaciones en concursos de robótica. Recuperado a partir de <http://repositorio.ucsg.edu.ec/handle/3317/3872>

Quinde Llerena, J.L., & Ulloa Patiño, L.D. (2012). Diseño y construcción de dos Robot tipo Warbot. Recuperado a partir de <https://dspace.ups.edu.ec/bitstream/123456789/6849/1/UPS-GT000656.pdf>

Reyes Cortés, F. (2011). Robótica: control de Robots manipuladores. México: Alfaomega.

Salas Arriarán, S. (2017). Todo sobre sistemas embebidos: Arquitectura, programación y diseño de aplicaciones prácticas con el PIC18F. Recuperado a partir de <https://www.overdrive.com/search?q=96D0DFFB-AAD1-4C56-B81A-841BDC61F7A2>

Santiago Espinosa, F. (2012). Los microcontroladores avr de atmel (Primera). Mexico.

Siegwart, R., & Nourbakhsh, I. R. (2004). Introduction to autonomous mobile robots. Cambridge, Mass: MIT Press.

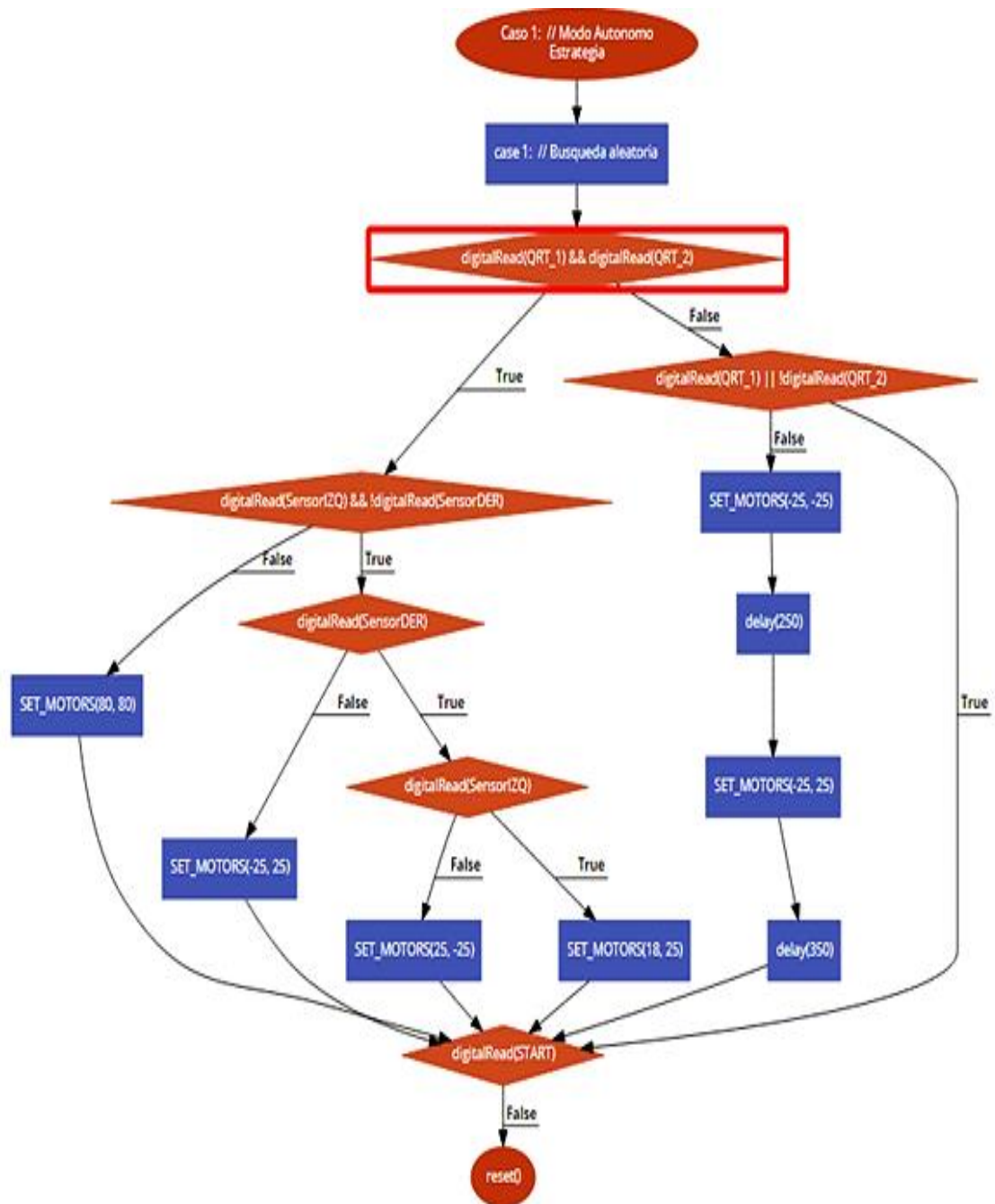
Vazques Fernandez-Pacheco, A. S., Ramos de la Flor, F., Fernández Rodríguez, R., Payo Gutierrez, I., & Adán Oliver, A. (2016). Robotica educativa. Colombia: Ra-ma (España).

Zabala, G. (2007). Robótica. Buenos Aires: Gradi.

## ANEXOS

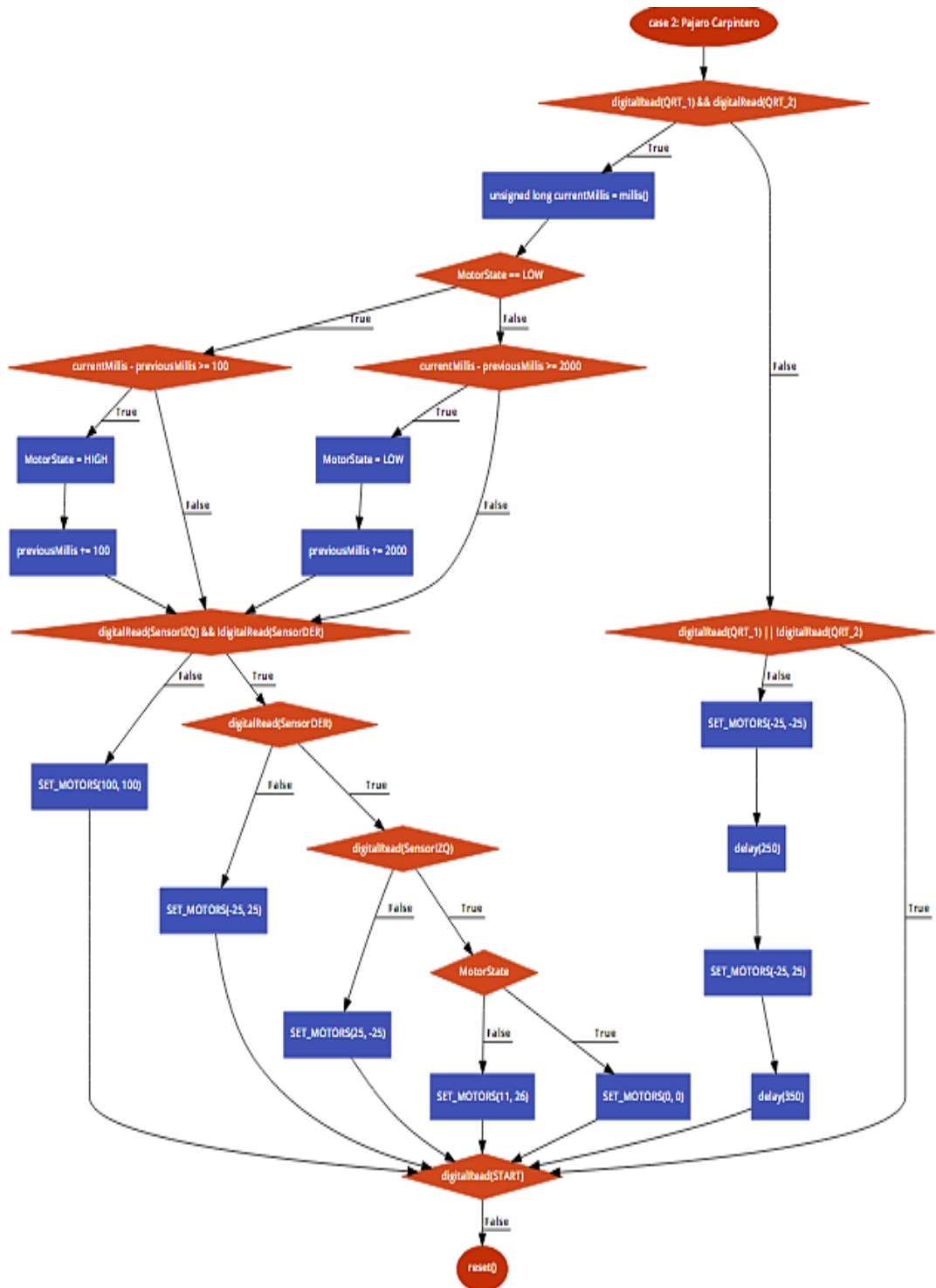
-Diagramas de flujo de estrategias

-Modo Autónomo

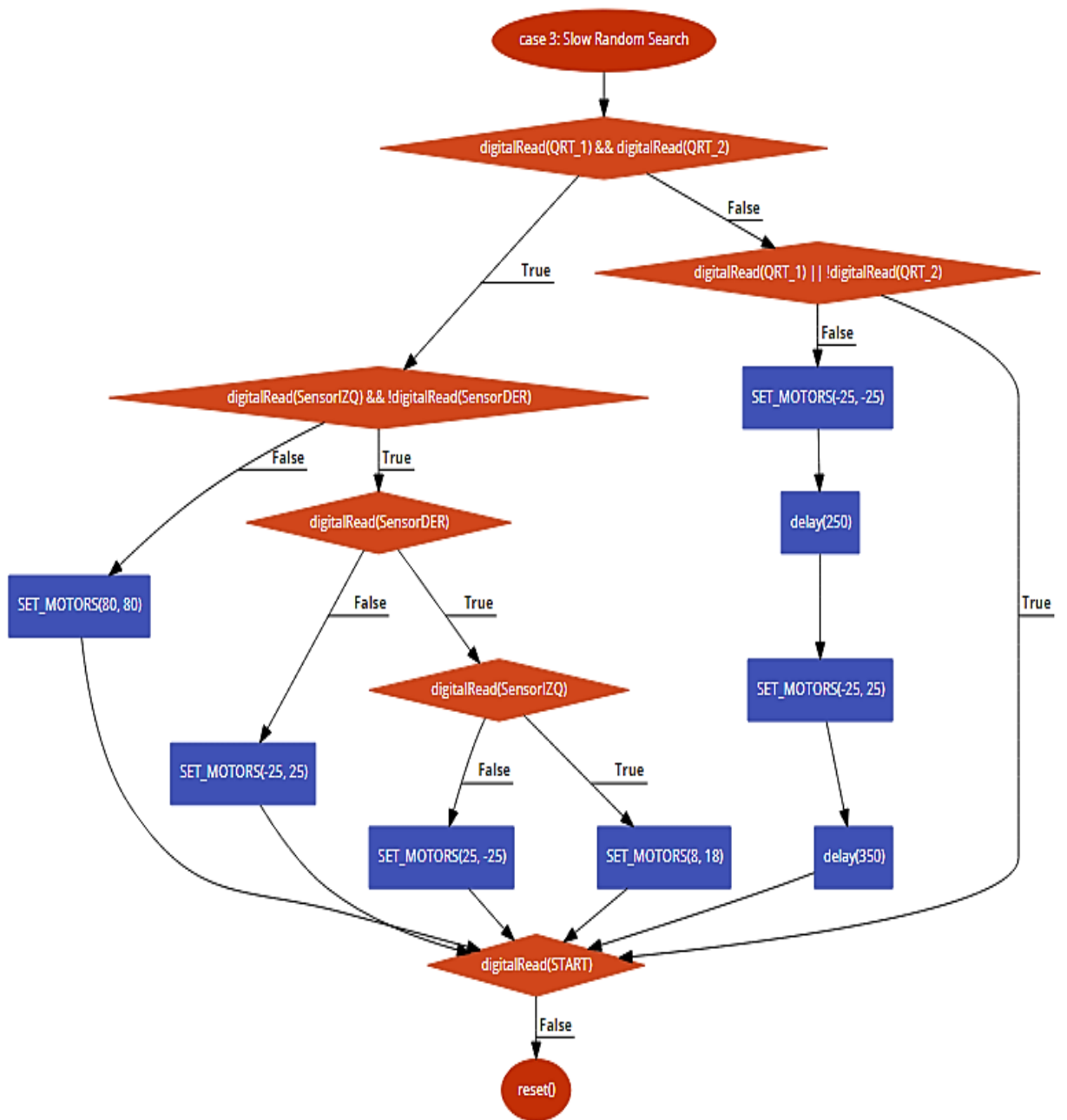




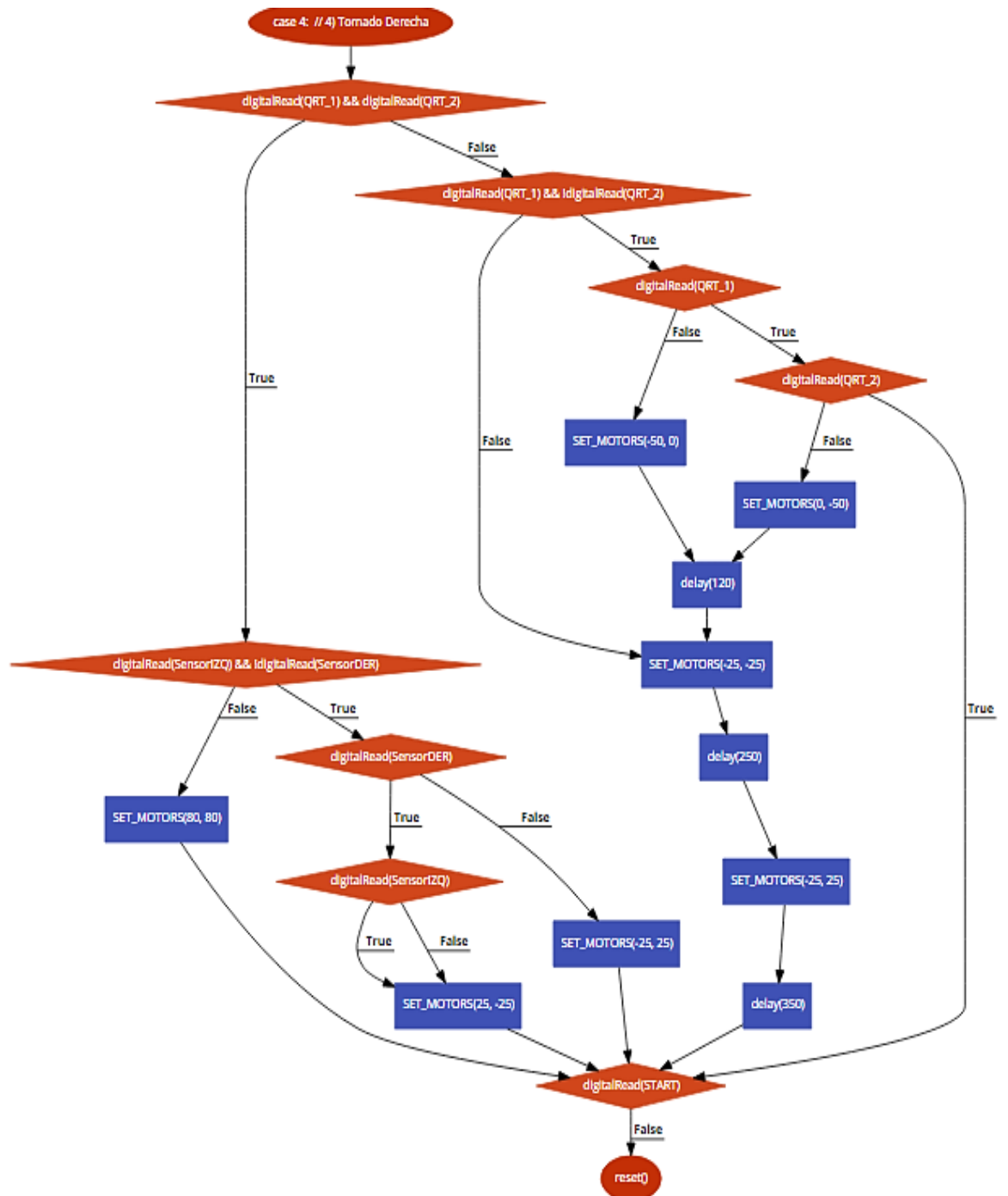
## -Búsqueda aleatoria



## -Búsqueda Pausada



## -Estrategia Tornado





Presidencia  
de la República  
del Ecuador



Plan Nacional  
de Ciencia, Tecnología,  
Innovación y Saberes



SENESCYT  
Secretaría Nacional de Educación Superior,  
Ciencia, Tecnología e Innovación

## DECLARACIÓN Y AUTORIZACIÓN

Yo, **CORDERO PÉREZ, JOHN PAUL** con C.C: # **093028170-4** autor del Trabajo de Titulación: **IMPLEMENTACIÓN DE ALGORITMOS DE PROGRAMACIÓN Y DE PROTOCOLOS DE COMUNICACIÓN INALÁMBRICA PARA LA OPERATIVIDAD DE UN ROBOT SUMO AUTÓNOMO Y RC** previo a la obtención del título de **INGENIERO EN TELECOMUNICACIONES** en la Universidad Católica de Santiago de Guayaquil.

1.- Declaro tener pleno conocimiento de la obligación que tienen las instituciones de educación superior, de conformidad con el Artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de titulación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.

2.- Autorizo a la SENESCYT a tener una copia del referido trabajo de titulación, con el propósito de generar un repositorio que democratice la información, respetando las políticas de propiedad intelectual vigentes.

Guayaquil, 6 de marzo de 2018

f. \_\_\_\_\_

Nombre: Cordero Pérez, John Paul

C.C: 093028170-4



Presidencia  
de la República  
del Ecuador



Plan Nacional  
de Ciencia, Tecnología,  
Innovación y Saberes



SENESCYT  
Secretaría Nacional de Educación Superior,  
Ciencia, Tecnología e Innovación

## **REPOSITORIO NACIONAL EN CIENCIA Y TECNOLOGÍA**

### **FICHA DE REGISTRO DE TESIS/TRABAJO DE TITULACIÓN**

<b>TÍTULO Y SUBTÍTULO:</b>	Implementación de algoritmos de programación y de protocolos de comunicación inalámbrica para la operatividad de un robot sumo autónomo y RC		
<b>AUTOR</b>	Cordero Pérez, John Paul		
<b>TUTOR</b>	Zamora Cedeño, Néstor Armando		
<b>INSTITUCIÓN:</b>	Universidad Católica de Santiago de Guayaquil		
<b>FACULTAD:</b>	Facultad de Educación Técnica para el Desarrollo		
<b>CARRERA:</b>	Ingeniería en Telecomunicaciones		
<b>TÍTULO OBTENIDO:</b>	Ingeniero en Telecomunicaciones		
<b>FECHA DE PUBLICACIÓN:</b>	6 de Marzo de 2018	<b>No. DE PÁGINAS:</b>	123
<b>ÁREAS TEMÁTICAS:</b>	Sistemas Microcontroladores, Comunicaciones Inalámbricas, Robótica.		
<b>PALABRAS CLAVES</b>	Microcontrolador, Comunicaciones Inalámbricas, Robótica, Diseño Electrónico, Algoritmo.		
<b>RESUMEN/ABSTRACT</b> (150-250 palabras):	<p>En el presente trabajo de titulación se implementa un algoritmo de programación completo y preciso mediante una tarjeta de diseño propio, con la finalidad de controlar a un robot sumo autónomo y radio controlado. Para poder desarrollar este proyecto se procedió al estudio y análisis de los componentes de un kit robótico de mega sumo de gama alta, entre sus elementos como los motores y sensores, siendo primordial el estudio del microcontrolador ya que es el cerebro del robot y parte fundamental de su sistema de control. Se analizó también el controlador de motor, el cual se encarga de dar el movimiento al robot direccionando el sentido de rotación de sus motores. Se analizó la comunicación entre el hardware y software del robot en donde se estableció los protocolos que se utilizan, pudiendo así obtener la información necesaria para el diseño de la tarjeta, la cual va a estar programada con el algoritmo propuesto</p>		
<b>ADJUNTO PDF:</b>	<input checked="" type="checkbox"/> SI	<input type="checkbox"/> NO	
<b>CONTACTO CON AUTOR:</b>	<b>Teléfono:</b> +593-9-6-0623575	<b>E-mail:</b> <a href="mailto:johnpaulcp1@hotmail.com">johnpaulcp1@hotmail.com</a>	
<b>CONTACTO CON LA INSTITUCIÓN: COORDINADOR DEL PROCESO DE UTE</b>	<b>Nombre:</b> Edwin Fernando Palacios Meléndez		
	<b>Teléfono:</b> +593-9-6-8366762		
	<b>E-mail:</b> <a href="mailto:edwin.palacios@cu.ucsg.edu.ec">edwin.palacios@cu.ucsg.edu.ec</a>		
<b>SECCIÓN PARA USO DE BIBLIOTECA</b>			
<b>Nº. DE REGISTRO</b> (en base a datos):			
<b>Nº. DE CLASIFICACIÓN:</b>			
<b>DIRECCIÓN URL</b> (tesis en la web):			