



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

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

TÍTULO:

**DISEÑO E IMPLEMENTACIÓN DE DOS ROBOTS SEGUIDORES DE
LÍNEA MODALIDAD VELOCISTA Y DESTREZA PARA
PARTICIPACIONES EN CONCURSOS DE ROBÓTICA**

AUTORES:

JORGE ANDRÉS VERA ARENAS
EDWARD ANDRÉS ALEJANDRO PROAÑO

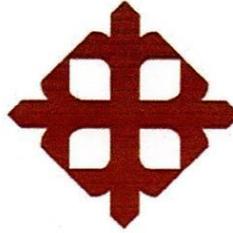
Previa la obtención del Título

INGENIERO EN TELECOMUNICACIONES

TUTOR:

Ing. Celso Bayardo Bohórquez Escobar, MSc.

Guayaquil, Ecuador



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

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

CERTIFICACIÓN

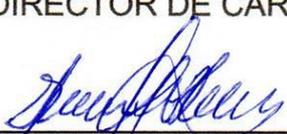
Certificamos que el presente trabajo fue realizado en su totalidad por los
Sres. **Jorge Andrés Vera Arenas** y **Edward Andrés Alejandro Proaño**
como requerimiento parcial para la obtención del título de INGENIERO EN
TELECOMUNICACIONES.

TUTOR



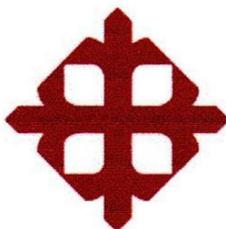
Ing. Celso Bayardo Bohórquez Escobar, MSc.

DIRECTOR DE CARRERA



Ing. Miguel A. Heras Sánchez, MSc.

Guayaquil, a los 14 del mes de Marzo del año 2016



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

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

DECLARACIÓN DE RESPONSABILIDAD

Nosotros, **Jorge Andrés Vera Arenas** y **Edward Andrés Alejandro Proaño**

DECLARAMOS QUE:

El trabajo de titulación "DISEÑO E IMPLEMENTACIÓN DE DOS ROBOTS SEGUIDORES DE LÍNEA MODALIDAD VELOCISTA Y DESTREZA PARA PARTICIPACIONES EN CONCURSOS DE ROBÓTICA" previa a la obtención del Título de Ingeniero en Telecomunicaciones, ha sido desarrollado respetando derechos intelectuales de terceros conforme las citas que constan al pie de las páginas correspondientes, cuyas fuentes se incorporan en la bibliografía. Consecuentemente este trabajo es de nuestra autoría.

En virtud de esta declaración, nos responsabilizamos del contenido, veracidad y alcance científico del Trabajo de Titulación referido.

Guayaquil, a los 14 del mes de Marzo del año 2016

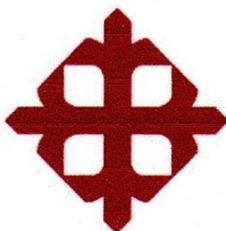
LOS AUTORES



JORGE A. VERA ARENAS



EDWARD A. ALEJANDRO PROAÑO



**UNIVERSIDAD CATÓLICA
DE SANTIAGO DE GUAYAQUIL**

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

AUTORIZACIÓN

Nosotros, Jorge Andrés Vera Arenas y Edward Andrés Alejandro Proaño

Autorizamos a la Universidad Católica de Santiago de Guayaquil, la publicación, en la biblioteca de la institución del Trabajo de Titulación: "DISEÑO E IMPLEMENTACIÓN DE DOS ROBOTS SEGUIDORES DE LÍNEA MODALIDAD VELOCISTA Y DESTREZA PARA PARTICIPACIONES EN CONCURSOS DE ROBÓTICA", cuyo contenido, ideas y criterios es de nuestra exclusiva responsabilidad y autoría.

Guayaquil, a los 14 del mes de Marzo del año 2016

LOS AUTORES



JORGE A. VERA ARENAS



EDWARD A. ALEJANDRO PROAÑO

DEDICATORIA

Dedicamos este trabajo en primer lugar a Dios por habernos dado la vida y permitirnos llegar a este momento tan especial e importante de nuestra formación profesional. A nuestras madres por ser una fuente de cariño y apoyo incondicional brindado en todo el trayecto de nuestra carrera. A nuestros padres por sus constantes consejos que nos ayudaron en cada paso del proceso formativo y constituyeron un pilar fuerte en nuestra formación.

A nuestros amigos que siempre estuvieron pendientes y nos dieron su apoyo cuando lo necesitamos. A nuestros compañeros universitarios con los que compartimos horas de estudio y sacrificio tanto dentro como fuera de las aulas de clases.

A nuestros profesores que marcaron cada etapa de nuestro trayecto universitario con su apoyo y sabiduría, y que nos asesoraron en el desarrollo de nuestro trabajo de titulación.

LOS AUTORES

JORGE ANDRÉS VERA ARENAS
EDWARD ANDRÉS ALEJANDRO PROAÑO

AGRADECIMIENTO

En primer lugar, agradecemos inmensamente a Dios por habernos dado la fuerza para llegar a la culminación de esta etapa de nuestras vidas.

A nuestras madres, que siempre fueron un ejemplo para nosotros y nos enseñaron a no rendirnos nunca, nos brindaron sus consejos y apoyo, y demostraron su infinito amor haciéndonos ver nuestros errores y celebrando nuestros aciertos.

A nuestros padres, cuyo apoyo, comprensión y cariño han sido parte fundamental de nuestras vidas, por compartir nuestras alegrías y tristezas.

A los ingenieros Celso Bayardo, Néstor Zamora, Edwin Palacios, Luis Córdova, Carlos Romero y Manuel Romero por sus enseñanzas y apoyo a lo largo de esta carrera universitaria.

A todas las personas que contribuyeron directa o indirectamente en la realización de este trabajo de titulación.

LOS AUTORES

JORGE ANDRÉS VERA ARENAS
EDWARD ANDRES ALEJANDRO PROAÑO

Índice General

Índice de Figuras	X
Índice de Tablas	XII
Resumen.....	XIII
Abstract	XIV
CAPÍTULO 1: INTRODUCCIÓN.....	15
1.1. Introducción.	15
1.2. Antecedentes.....	16
1.3. Justificación del Problema.	17
1.4. Definición del Problema.....	18
1.5. Objetivos del Problema de Investigación.	18
1.5.1. Objetivo General.....	18
1.5.2. Objetivos Específicos.	19
1.6. Hipótesis.....	19
1.7. Metodología de Investigación.	19
CAPÍTULO 2: FUNDAMENTACIÓN TEÓRICA	22
2.1. Reseña histórica de los microcontroladores.	22
2.2. Microcontroladores.....	26
2.2.1. Características de los microcontroladores.	28
2.3. Microcontrolador y Microprocesador.....	29
2.4. Arquitectura que compone un microcontrolador.....	30
2.4.1. El procesador o CPU	31
2.4.2. Memorias.....	33
2.4.3. Puertas de entrada y salida.	37
2.4.4. Reloj principal.....	37
2.5. Elementos auxiliares.....	38
2.5.1. Temporizador o timers	39
2.5.2. Perro guardián o watchdog.	39
2.5.3. Protección ante fallo de alimentación o “Brownout”	40

2.5.4.	Estado de reposo o de bajo consumo.....	40
2.5.5.	Convertor Analógico – Digital.....	41
2.5.6.	Convertor Digital – Analógico.....	41
2.5.7.	Comparador analógico.....	41
2.5.8.	Fuente de alimentación.....	42
2.5.9.	Reset Input (Reseteo de entrada).....	42
2.5.10.	Interrupciones.....	43
2.6.	Tipo de microcontroladores.....	43
2.6.1.	Microcontroladores de 4 bits.....	44
2.6.2.	Microcontroladores de 8 bits.....	44
2.6.3.	Microcontroladores de 16 bits.....	45
2.6.4.	Microcontroladores de 32 bits.....	45
2.7.	Familia de Microcontroladores.....	46
2.7.1.	Microcontroladores PIC.....	46
2.7.2.	Microcontroladores Atmel AVR.....	48
2.8.	La robótica.....	51
2.8.1.	Reseña histórica.....	51
2.8.2.	¿Qué es un robot?.....	54
2.8.3.	Tipos de robot.....	55
2.8.4.	Impacto de la robótica.....	58
2.9.	Robótica Móvil.....	59
2.9.1.	Vehículos Guiados.....	61
2.9.2.	Sensores aplicables a la robótica.....	61
2.9.3.	Motores CD.....	63
2.9.4.	Motores de Engranaje.....	63
2.10.	Seguidores de línea.....	64
2.10.1.	Modelo de un sensor.....	65
2.10.2.	Modelo de dos sensores.....	65

CAPÍTULO 3: DISEÑO Y CONSTRUCCION DE DOS ROBOTS SEGUIDORES DE LÍNEA.....	67
3.1. Materiales y dispositivos electrónicos para la implementación de los robots móviles.....	67
3.1.1. Microcontrolador ATmega328/P.....	67
3.1.2. Controlador de motores o puente H (TB6612FNG).....	72
3.1.3. Adaptador de USB a serial UART (TTL).....	75
3.1.4. Sensor Óptico Reflexivo.	77
3.1.5. Motores Pololu CD.....	79
3.1.6. Regulador de voltaje.	82
3.1.7. Ruedas pololu.....	83
3.1.8. Ruedas omnidireccionales (Ball casters).....	84
3.2. Controlador PID.....	84
3.3. Diseño electrónico de seguidor de línea velocista y destreza.....	87
3.3.1. Diseño PCB (Printed Circuit Board).....	93
3.4. Construcción de los robots seguidores de línea.	94
3.5. Presupuesto Referencial de los robots seguidores de línea.	99
3.6. Algoritmos de programación.....	101
3.6.1 Programación seguidor de línea velocista.....	101
3.6.2. Programación seguidor de línea destreza.....	105
CAPÍTULO 4: CONCLUSIONES Y RECOMENDACIONES.	112
4.1. Conclusiones.....	112
4.2. Recomendaciones.....	113
REFERENCIAS BIBLIOGRÁFICAS	114
ANEXO	116

Índice de Figuras

Capítulo 2

Figura 2. 1: Microcontrolador PIC16C74.....	28
Figura 2. 2: Estructura de un sistema abierto con microprocesador	29
Figura 2. 3: Estructura de un microcontrolador de sistema cerrado.....	30
Figura 2. 4: Adaptacion de Frankenstein	53
Figura 2. 5: Robots de largometraje original de Star Wars	53
Figura 2. 6: Motor de engranaje CD.....	64
Figura 2. 7: Patrón Z con un sensor.....	65
Figura 2. 8: Patrón de movimiento del modelo de dos sensores	66

Capítulo 3

Figura 3. 1: Chip Atmega328/P.....	68
Figura 3. 2: Diagrama de bloque del núcleo AVR.....	69
Figura 3. 3: Puente H (TB6612FNG).	72
Figura 3. 4: Distribución de pines del Puente H	73
Figura 3. 5: Adaptador USB a Serial UART.....	75
Figura 3. 6: Diagrama de bloques Adaptador FT232RL.	76
Figura 3. 7: Configuración de poder por Bus USB.....	77
Figura 3. 8: Sensor QRD1113/QRD1114.....	78
Figura 3. 9: Motor de engranaje Pololu.....	80
Figura 3. 10: Distribución de pines del regulador de voltaje.	82
Figura 3. 11: Rueda Pololu acoplada a un motor de engranaje.....	83
Figura 3. 12: Pololu ball Casters.	84
Figura 3. 13: Patrón de seguimiento de línea sin controlador PID.....	85
Figura 3. 14: Patrón de seguimiento con controlador PID	86
Figura 3. 15: Entrada de la batería.	87
Figura 3. 16: Sensor QRD1113.....	87
Figura 3. 17: Tarjeta Arduino Nano.....	88
Figura 3. 18: Puente H.....	88
Figura 3. 19: Microcontrolador ATmega 328/P	89
Figura 3. 20: Depurador externo ICSP.....	89
Figura 3. 21: Auto selector VDD.	90

Figura 3. 22: Fuente de alimentación de +5VDC.	90
Figura 3. 23: Entradas digitales y analógicas.....	90
Figura 3. 24: Adaptador USB a Serial UART	91
Figura 3. 25: Controlador de motores.	91
Figura 3. 26: Arreglo de sensores Izquierdo y Derecho.....	92
Figura 3. 27: Sensor adicional y asignación de señales.	93
Figura 3. 28: Diseño PCB (vista superior).....	93
Figura 3. 29: Diseño PCB (vista inferior).....	94
Figura 3. 30: Seguidor de línea (prototipo)	94
Figura 3. 31: Vista superior de la placa impresa	95
Figura 3. 32: Vista inferior previa al montaje de motores.....	95
Figura 3. 33: Seguidor de línea modalidad “Velocista”	96
Figura 3. 34: Seguidor de línea modalidad “Destreza”	96
Figura 3. 35: Prototipo y robot finalizado	97

Índice de Tablas

Capítulo 3

Tabla 3. 1: Distribución de los pines del puente H (TB6612FNG).....	73
Tabla 3. 2: Especificaciones técnicas y eléctricas del controlador.....	74
Tabla 3. 3: Especificaciones Técnicas para un rango de funcionamiento ($T_a = -20 \sim 85^\circ\text{C}$) del controlador.	74
Tabla 3. 4: Especificaciones técnicas y eléctricas del adaptador.....	76
Tabla 3. 5: Índices máximos absolutos del sensor QRD1113/QRD1114.....	78
Tabla 3. 6: Principales características de diferentes micro motores.	79
Tabla 3. 7: Valores mínimos y máximos para las características técnicas y eléctricas del regulador de voltaje.....	83
Tabla 3. 8: Listado de elementos utilizados para la construcción del seguidor de línea.	98
Tabla 3. 9: Costo de materiales del robot prototipo.	99
Tabla 3. 10: Costo de materiales para la construcción final del robot velocista.....	99
Tabla 3. 11: Costo de materiales para la construcción final del robot destreza.	100
Tabla 3. 12: Costo total de la implementación de los robots.....	100

Resumen

El presente proyecto de titulación busca profundizar el conocimiento en la robótica móvil y contemplar el ámbito competitivo durante su desarrollo. Explorar las secciones de diseño, hardware y software para la elaboración de dos robots seguidores de línea modalidad velocista y destreza. Describimos la influencia de la electrónica desde décadas atrás, las leyes fundamentales que rigen a un robot y su papel en la cultura popular.

En la fundamentación teórica tenemos lo concerniente a la electrónica detrás de la robótica, los microcontroladores, sus tipos, familias y arquitectura. Describimos las partes esenciales de los vehículos guiados, y damos un vistazo general a los seguidores de línea en específico y sus modelos de sensores básicos.

Este proyecto detalla cada uno de los elementos utilizados y su funcionamiento para la construcción de los robots seguidores de línea, el diseño electrónico y los algoritmos de programación detallados de ambas modalidades.

La finalidad es analizar el proceso completo en la construcción de un robot y así motivar a los estudiantes a involucrarse en cualquier sección del mismo, mejorar diseños, impulsar la creatividad y lograr importantes resultados en el ámbito competitivo.

Abstract

This project seeks to deepen the knowledge degree in mobile robotics and watch the competitive environment during development. Sections explore design, hardware and software for the development of two line follower robots speed and skill mode. Describe the influence of electronics decades ago, the fundamental laws that rule a robot and its role in popular culture.

In the theoretical foundation, we expand concerning to the electronics behind robotics, microcontrollers, types, families and architecture. We describe the essential parts of guided vehicles, and give an overview of the specific line followers and their basic models of sensors.

This project describes each of the elements used and its function in order to build line followers robots, electronic design and detailed programming algorithms of both.

The purpose is to analyze the entire process in building a robot and thus motivate students to engage in any section involved, improve designs, boost creativity and achieve important results in the competitive arena.

CAPÍTULO 1: INTRODUCCIÓN

1.1. Introducción.

En el transcurso de las décadas, la electrónica ha estado entrando poco a poco en diversos campos o disciplinas llegando a convertir la robótica en uno de gran importancia y de constante desarrollo en la actualidad, ya que incursiona en las etapas de diseño, elaboración y programación de robots de diversos tipos, ya sea para satisfacer necesidades u optimizar procesos en áreas industriales o variar metodologías e incluir la tecnología en áreas educativas. Cualquiera que sea el caso estos dispositivos deben efectuar procesos y responder a acciones específicas en las cuales tenemos la intervención de múltiples elementos electrónicos, entre ellos los microcontroladores que son muy utilizados en el campo de automatización y en la programación de procesos.

Con lo anterior mencionado se plantea a la Universidad Católica de Santiago de Guayaquil el diseño e implementación de dos robots seguidores de línea tipo velocista y destreza. Teniendo la intención que esto a su vez sirva a los actuales y futuros estudiantes de carreras afines a la electrónica para el análisis de tarjetas electrónicas, componentes y la respectiva programación llegando a ser fuente de motivación adicional para que emprendan el diseño propio de circuitos y dispositivos con fines similares, consiguiendo con esto generar una mayor variedad práctica en la formación universitaria de los

estudiantes de la Facultad de Educación Técnica para el Desarrollo y sientan el gratificante mérito de finalizar un proyecto con un igual o superior nivel de complejidad.

1.2. Antecedentes.

Se conoce que hace muchos años atrás brillantes matemáticos, físicos, artistas habían diseñado dispositivos mecánicos pero solo de forma teórica y gráficos no obstante, nunca se vio o se catalogó esos diseños como los inicios de la robótica. No fue hasta 1770 que fabricantes suizos de relojes crean muñecas con una función única ya sea poder escribir, dibujar y tocar música; entonces comenzó a ver se la ciencia y tecnología robótica como algo innovador y con un futuro por delante. Existieron grandes inventores como Nikola Tesla quien construyó y demostró un barco robot controlado a distancia, Alan Turing considerado realmente el padre de la informática construyo un ordenador capaz de descifrar comunicaciones por códigos morse que eran enviadas a una maquina llamada enigma que se utilizó en la segunda guerra mundial dando como resultado en cierta forma el cese de la guerra.

El escritor (Asimov, 1984) quien produce una serie de cuentos sobre robots a quien se le atribuye la popularización del término robótica y que cuya más importante contribución fueron sus tres leyes de la robótica escritas a continuación:

- Ley 1: Un robot no debe dañar a un ser humano o, por su inacción, dejar que un ser humano sufra daño
- Ley 2: Un robot debe obedecer las órdenes que le son dadas por un ser humano, excepto cuando estas órdenes están en oposición con la primera Ley
- Ley 3: Un robot debe proteger su propia existencia, hasta donde esta protección no esté en conflicto con la primera o segunda leyes.

Sin embargo, Asimov agrega una “ley cero”: Un robot no debe dañar a la humanidad o, por inacción, permitir que la humanidad sufra daño.

Agencias espaciales desarrollan robots cuyo propósito va a depender de lo que se quiera hacer como robots exploradores, industriales, androides, móviles entre otros que de cierta forma ayudan u optimizan el trabajo de los humanos ya sea de fuerza física o producción en área industrial entre otros.

1.3. Justificación del Problema.

El presente trabajo de titulación pretende elaborar dos robots cuyas características cumplirán las normativas de los concursos de robótica como lo son RIOTRONIC (Universidad Superior Politécnica de Chimborazo, ESPOCH), UMEBOT (Escuela Politécnica Nacional, EPN) y el CER (Concurso Ecuatoriano de Robótica) en sus respectivas categorías siendo una de estas correspondiente a destreza y modalidad velocista donde se

permitirá observar la creatividad plasmada en los robots a concursar y el ingenio por parte de los estudiantes de las distintas universidades del país.

A la vez se busca proporcionar a la Facultad de Educación Técnica para el Desarrollo de más instrumentos que faciliten el futuro desarrollo de robots móviles por parte de los estudiantes de las carreras técnicas y en un proceso conjunto con los docentes logren participar en eventos de importancia como es el Concurso Ecuatoriano de Robótica donde se vea finalmente reflejado el esfuerzo realizado y el conocimiento adquirido.

1.4. Definición del Problema.

Necesidad de diseñar e implementar robots utilizando tarjetas electrónicas Atmel para los diferentes concursos de robótica y así impulsar la iniciativa del estudio y elaboración de nuevos robots por parte de los estudiantes de la Universidad Católica de Santiago de Guayaquil.

1.5. Objetivos del Problema de Investigación.

1.5.1. Objetivo General.

Diseñar e implementar dos robots seguidores de línea modalidad velocista y destreza para participaciones en concursos de robótica.

1.5.2. Objetivos Específicos.

- Describir la fundamentación teórica de los sistemas microcontrolados y de la robótica móvil.
- Diseñar los modelos del seguidor de línea velocista y destreza cumpliendo las normativas de los concursos de robótica.
- Realizar los algoritmos en lenguaje de alto nivel para el seguidor de línea velocista y destreza.

1.6. Hipótesis.

Si se diseñan robots utilizando tarjetas Atmel se podrá ofrecer el más alto nivel de tecnología posible para competir en concursos a nivel nacional e internacional en el ámbito universitario, el campo de la robótica tendrá un impulso importante en nuestra universidad, incentivando directamente a los actuales y futuros estudiantes de esta especialidad a integrarse a las competencias de robótica nacionales e internacionales.

1.7. Metodología de Investigación.

El método teórico y sistémico fue elegido debido a que es necesario descubrir las relaciones esenciales y cualidades fundamentales en el objeto de investigación para proceder con su implementación. Se define al método

sistémico como aquel dirigido a modelar el objeto mediante la determinación de sus componentes y las relaciones entre ellos, las cuales determinan su estructura y su dinámica.

Previo a la implementación del proyecto se realizó una extensa búsqueda y análisis de información de múltiples fuentes en base a varias temáticas como lo son:

- Funcionamiento de robot seguidor de línea modalidad velocista.
- Funcionamiento de robot seguidor de línea modalidad destreza.
- Lenguajes de programación que utilicen software libre.
- Reglamento y normativas de concursos de robótica a nivel nacional.

Una gran parte de este proceso se centró en búsqueda y consultas como:

- Consulta y análisis de competencias previamente realizadas y las categorías correspondientes.
- Búsqueda de información en páginas web y textos.
- Búsqueda y análisis de distintos proveedores de partes en el mercado electrónico local y extranjero.
- Consultas a los ingenieros docentes de la Facultad Técnica en la Universidad.
- Obtención de las partes necesarias.

Cada uno de los puntos mencionados anteriormente fueron de suma importancia para el desarrollo de nuestro proyecto, el levantamiento de la información y la sección teórica permitieron finalmente la exitosa implementación que se había propuesto.

CAPÍTULO 2: FUNDAMENTACIÓN TEÓRICA

2.1. Reseña histórica de los microcontroladores.

En la década de los 60's (1969) un grupo de ingenieros japoneses que trabajaban en una compañía llamada Busicom impulsan un modelo de trabajo que consiste en poder usar menos circuitos integrados. Este modelo de trabajo tuvo su aceptación gracias a INTEL que fue la primera compañía en desarrollar el primer microprocesador.

En cambio en la década de los 70's (1971) la compañía INTEL decide comprar los derechos del modelo de trabajo a Busicom para de esta manera poder integrar al mercado el primer microprocesador cuyo nombre será 4004, este microprocesador era 4 bits y de 6000 operaciones por segundo.

En la misma década de los 70's tanto la compañía INTEL como Texas Instruments trabajan en conjunto para desarrollar el siguiente microprocesador el mismo que fue llamado 8008 y éste incluía 8 bits, era capaz de almacenar 16 kb de memoria con un total de 45 instrucciones y aproximadamente 300,000 operaciones por segundo.

Después del desarrollo del microprocesador 8008, la compañía INTEL comercializa el único microprocesador llamado 8080 que contenía tecnología NMOS, lo que permitía que el mismo pudiera realizar aproximadamente unas

500,000 operaciones por segundo, de tal forma también se incrementó la capacidad de direccionamiento de memoria a 64 Kbyte.

En la década de los 70's (1975) una compañía llamada MOS Technology llega a diseñar el MOS 6502, un microprocesador de 8 bits con un bus de dirección de 16 bits, en este mismo año surgen los primeros PIC que se los denominó PIC16750 cuyo desarrollo se llevó a cabo por la división de microelectrónica de General Instrument. Este grupo de PICs emplea micro código que fácilmente es guardado en la memoria ROM para poder ejecutar operaciones. Esta fue una de las primeras familias PIC.

Específicamente en el año 1976 se dan a conocer las primeras microcomputadoras de un solo chip, que luego de mucho tiempo se llegarían a denominar como microcontroladores. En este año la compañía llamada Zilog Anuncia el Z80 que llega a excluir vertiginosamente al INTEL 8080 del mercado y se llega a transformar en uno de los procesadores de 8 bits más reconocidos. Las versiones iniciales funcionaban a 2,5 MHz pero su velocidad aumentó hasta los 20 MHz.

No fue hasta la década de los 80's que los fabricantes de circuitos integrados iniciaron la expansión de un nuevo diseño de circuito para control, medición e instrumentación basado en un solo chip el cual fue nombrado microcontrolador. En esta misma década surge el 8051(INTEL), un microcontrolador desarrollado por Intel específicamente para el uso de

productos impregnados. Muestra un diseño no muy usual, sin embargo, es muy poderoso y sencillo de programar. Su arquitectura sigue siendo Harvard modificada con espacio de direcciones separadas para memoria y proceso de datos.

Se comercializan los chips 80186 y 80188 desarrollados por INTEL los cuales eran la versión en microcontrolador de 8086 y del 8088 del famoso pc de IBM. Este Chip posee dos canales de DMA, dos contadores/temporizadores, controlador de interrupción programable y pueden usar herramientas de desarrollo estándar para pc.

En algún momento de la década de los 80's surgen las imitaciones de los PIC. Grandes compañías comienzan a desarrollar copias o imitaciones más baratas y superficialmente modernizadas. Una de estas compañías es Ubicom.

En el año 1985 se introduce la tercera generación de microcontroladores INTEL con el 80c196, un microprocesador de 16 bits. Fue creado con tecnología NMOS, sin embargo fue rediseñado con tecnología CMOS. Contiene un multiplicador y divisor hardware con 6 canales de direccionamiento. Alta Velocidad de E/S y 8 controladores de interrupción programables.

En la década de los 90's se introduce el microcontrolador 80386EX el cual era una versión del 386 diseñado para sistemas embebidos. Este microcontrolador fue muy utilizado debido a que su uso de hoy en día cae en el campo de satélites espaciales. Contiene dos canales DMA, ISO asíncrona, cache de 32 kb y direccionamiento de 26 bits con 64 Mb en RAM.

Se desarrollan los PICs Wireless por el año 2001, un microcontrolador rfPic que incorpora todas las ventajas de los PicMicro para obtener así una comunicación inalámbrica.

DsPic pasan a ser el penúltimo lanzamiento de Microchip en el año 2003 ya que fueron los primeros PICs con un bus de datos inherente de 16 bits. Estos llegan a incorporar todas las características pasadas de los PIC pero adicionalmente se agregan las operaciones DSP en hardware. Mostraba arquitectura Von-Neuman en donde las instrucciones, datos, entradas/salidas y temporizadores abarcaban el mismo espacio en memoria. Más tarde ese mismo año los microcontroladores 68HC05 de Motorola pasan a ser obsoletos en el mercado.

La compañía Microchip en el 2007 lanza los nuevos microcontroladores de 32 bits con una velocidad de procesamiento de 1.5 DMIPS/MHz con capacidad host USB.

2.2. Microcontroladores.

Los microcontroladores son diminutos computadores que abarcan gran variedad de los dispositivos que fabricamos y usamos los seres humanos. En la actualidad estos microcontroladores se encuentran en dispositivos electrónicos que usamos comúnmente en nuestros hogares, como por ejemplo teléfonos celulares, televisores, computadores portátiles o de escritorio, cocinas, en un sinnúmero de aparatos electrónicos.

Para poder comprender mejor que son los microcontroladores debemos tener el conocimiento de que es un controlador.

Se llama controlador ya que permite dirigir el funcionamiento de un dispositivo electrónico mediante la ejecución de uno o varios procesos, aunque la definición de controlador ha permanecido intacta a través de décadas, la única variación que si ha tenido en realidad, es la parte física del mismo.

Hace décadas los controladores se construían exclusivamente con componentes de lógica discreta, después de un tiempo se emplearon los microprocesadores que se rodeaban con chips de memoria y E/S sobre una tarjeta de circuito impreso. (Chamorro Hernandez, Acuña Ortiz, & Acero Patiño, 2009)

Con el pasar de los tiempos todos estos componentes que se encuentran en un controlador se pudieron incluir en un chip cuyo nombre es el microcontrolador.

Se dice microcontrolador al dispositivo electrónico integrado que se distingue por poseer una gran variedad de componentes que forman parte de un controlador.

Para poder programar un microcontrolador se debe tener conocimiento de una gran variedad de lenguajes de programación ya que muchos de estos microcontroladores soportan varios de estos lenguajes de programación como Pascal, Java, Lisp entre otros. De cierta forma estos lenguajes resultan ser más amigables al momento de su uso ya que permiten el desarrollo de programas para el mismo.

Por otro lado, los microcontroladores son fabricados por distintas compañías en la actualidad, es decir que poseen diferentes idiomas de lenguaje de montaje, por ende el usuario final debe tener conocimiento de un nuevo idioma con cada uno de los microcontroladores que vaya a utilizar.

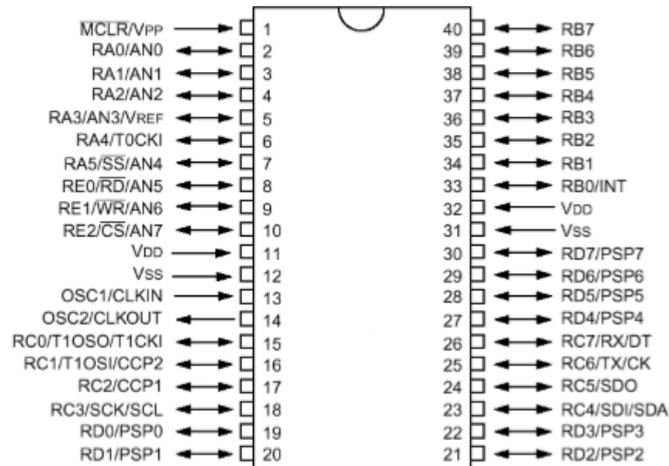


Figura 2. 1: Microcontrolador PIC16C74

Fuente: (Angulo Usategui & Angulo Martínez, 2005)

2.2.1. Características de los microcontroladores.

Al momento de la fabricación de un microcontrolador su memoria interna llamada ROM no posee ningún dato, por lo tanto es indispensable crear o desarrollar una instrucción y luego grabar en la memoria EEPROM dentro del microcontrolador utilizando algún programa ya sea desarrollado en un lenguaje compatible con el mismo o un lenguaje ensamblador.

Para que el programa pueda ser grabado en la memoria del microcontrolador, debe ser codificado en sistema numérico hexadecimal, cuando éste es alimentado con el voltaje adecuado y asociado a dispositivos analógicos y discretos hará que finalmente trabaje el microcontrolador. (Angulo Usategui & Angulo Martínez, 2005), (Piúa Martínez & Castillo Valarezo, 2015)

2.3. Microcontrolador y Microprocesador.

El microprocesador es un circuito integrado que contiene la Unidad Central de Proceso (UCP) llamado procesador, de un computador. La UCP está formada por la unidad de Control, que interpreta las instrucciones y el camino de datos que la ejecuta.

Se debe tomar en cuenta que la configuración de un microprocesador es cambiante de acuerdo a las aplicaciones a las que se destine por eso se lo cataloga como un sistema descubierto.

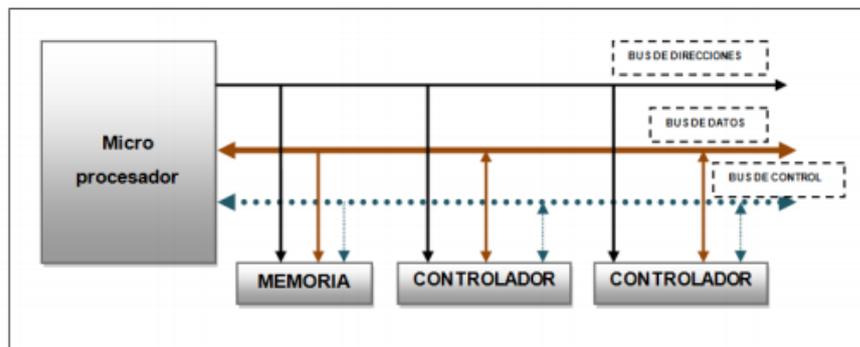


Figura 2. 2: Estructura de un sistema abierto con microprocesador

Fuente: (Angulo Usategui & Angulo Martínez, 2005)

Por otro lado, un microcontrolador posee todos los componentes de un computador pero con unas características fijas que no pueden alterarse, tiene todos los periféricos incluidos, sin embargo existe la posibilidad de elegir tanto la cantidad de espacio de memoria para utilizar, las terminales de entrada y salida y también los componentes secundarios pero permitirán tanto la

potencia como velocidad de procesamiento del microcontrolador. Cabe recalcar que sirve para aplicaciones de control y es de un bajo costo.

Se dice que el microcontrolador es un sistema sellado que contiene internamente elementos de una computadora y la única forma de relacionarse con estos elementos es mediante las terminales del microcontrolador.

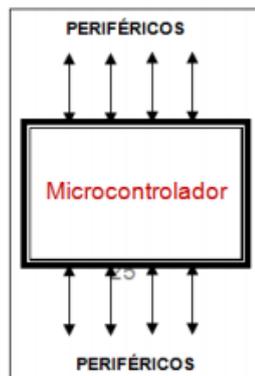


Figura 2. 3: Estructura de un microcontrolador de sistema cerrado

Fuente: (Angulo Usategui & Angulo Martínez, 2005)

2.4. Arquitectura que compone un microcontrolador.

Un microcontrolador contiene todos los componentes que le permiten operar independientemente, y que han sido diseñados, en particular, para el seguimiento y / o tareas de control. Un microcontrolador aparte de disponer de su componente principal el procesador, también contiene memorias que le permiten el almacenamiento de datos, posee varios controladores de interfaz, tiene uno o más temporizadores, dispone también de un controlador de interrupciones, y por último, pero definitivamente importante pines de E/S que

le permiten interactuar directamente con su entorno. Los microcontroladores también incluyen operaciones de bits que le permiten convertir un bit en un byte.

A continuación, se detallan algunos de los elementos que componen un microcontrolador.

2.4.1. El procesador o CPU

Es sin duda el elemento más importante del microcontrolador ya que tiene como encargo el controlar y encaminar las instrucciones de memoria, para poder tener una correcta ejecución de las operaciones de instrucciones internas y un buen acopio de los resultados que se generen.

Por otro lado sabemos que el cerebro del microcontrolador CPU (unidad de control de proceso) es el encargado de realizar todo el procedimiento tanto lógico como aritmético para el correcto funcionamiento del mismo. Esta unidad de control CU tiene como propósito controlar las operaciones internas del microprocesador y enviar las señales a los diferentes destinos del microcontrolador para llevar a cabo los procesos necesitados.

A continuación se detallan tanto el funcionamiento como la arquitectura que existe en la actualidad de tres tipos de procesadores.

➤ CISC.

Computadores de juego de instrucciones complejos: fueron desarrollados con un gran número de procesadores que poseen más de 100 instrucciones algunas de las cuales son muy eficaces y complejas ya que para poder ser ejecutadas requieren de bastante tiempo.

Este tipo de procesadores permiten al usuario almacenar ciertas instrucciones para después colocarlas en ejecución de manera seguida, tanto los datos como instrucciones tiene que ser de 8 bits de ancho.

➤ RISC.

Computadores de juego de instrucciones reducido: las instrucciones que se encuentran en su inventario son muy limitadas y se pueden ejecutar en un ciclo, sus datos son de 8 bits de ancho, aunque las palabras que conforman las instrucciones son de más de 8 bits de ancho.

La simplicidad y velocidad de estas instrucciones permiten optimizar tanto el software como hardware del procesador.

➤ SISC.

Computadores de juego de instrucciones específico: cuando se habla de específico se refiere a que cuando los microcontroladores están designados a aplicaciones muy concretas las instrucciones que se ejecutarán se adaptarán a la necesidad de la aplicación prevista.

2.4.2. Memorias.

Cuando se habla de memoria se debe tener en cuenta que en los microcontroladores, la memoria de instrucciones y datos se encuentra incorporada en el propio chip que se puede catalogar como dos tipos de memoria: una que se llama memoria de programa que va ser la encargada de contener el código de instrucciones que gobierna la aplicación, escrito por el programador y que no es volátil, es decir que los datos no se perderán después que se encuentre apagado, como una memoria tipo ROM y por ultimo tendremos la memoria de datos que tiene como función el contener datos temporales donde serán guardadas variables que serán usados en el programa y que es volátil, es decir que una vez se encuentre apagado los datos se perderán, hablamos de una memoria tipo RAM.

A continuación se detallaran ciertos tipos de memorias que se usan normalmente en los microcontroladores:

➤ Memoria RAM.

Memoria de Acceso Aleatorio como se dijo anteriormente ésta estará destinada a almacenar los datos de uso general usados por el usuario, es una memoria volátil que no puede retener datos cuando no existe energía.

En cuanto al uso en microcontroladores como estos son dispositivos de poca capacidad pues la memoria RAM interna será de unos 256 bytes que es lo común aunque variará de microcontrolador pues esta memoria solo tendrá que abarcar la información que se produzca de los cambios de las variantes en la duración de ejecución del programa.

➤ Memoria ROM.

Memoria de solo Lectura: esta es una memoria estable, capaz de mantener la información almacenada sin que esta se pierda, aunque exista ausencia de energía y cuando regrese esa energía los datos originales estarán aun almacenados.

La memoria contiene el programa o los datos fijos del usuario que serán grabados durante la fabricación del chip y el usuario no podrá cambiar el contenido.

➤ Memoria PROM.

Memoria Programable de solo Lectura: a diferencia de la memoria ROM esta no se programa durante el proceso de fabricación del chip, sino que el usuario es el encargado de efectuar la programación, pero solo lo podrá realizar una vez ya que después de realizarlo no podrá ser borrada la programación o volver almacenar otra información, estas memorias son utilizadas por lo general en aplicaciones de bajo costo que requieren un número limitado de estas memorias.

➤ Memoria EPROM.

Memoria borrable de solo lectura programable, los microcontroladores que disponen de este tipo de memorias pueden borrarse y programarse muchas veces usando un dispositivo de programación adecuado.

Si posteriormente se desea borrar el contenido de la memoria, este contiene una pequeña ventana de cristal en la superficie por lo que puede estar sometida a luz ultravioleta por varios minutos lo cual hará que los datos contenidos se puedan borrar.

Cabe recalcar que antes de que se desee reprogramar la memoria se debe borrar el contenido anterior, este proceso se puede realizar varias veces hasta que el usuario esté satisfecho con el programa.

➤ Memoria EEPROM.

Memoria de solo lectura programable y borrable eléctricamente: este tipo de memoria no es volátil es decir puede ser reprogramada siempre y cuando se realice de manera eléctrica desde un dispositivo de programación adecuado, no poseen una ventana de cristal en la superficie como en las EPROM, se debe tomar en cuenta que el número de reprogramaciones que podría soportar este tipo de memoria es reducido ya que al tener una continua reprogramación podría afectar el buen funcionamiento de la memoria, este modelo de memoria es parcialmente tardío y es más costoso que un chip EPROM, por ende la demanda de este tipo de memorias frente a otras es diminuta.

➤ Memoria FLASH.

Una de las ventajas significativas de este tipo de memoria es que puede ser reprogramada eléctricamente, esta memoria posee grandes características de las memorias vistas con anterioridad y es porque tiene una capacidad muy alta de contener información. En cuanto a costos resulta ser

más barato fabricar este tipo de modelo de memoria ya que pueden ser fabricadas con capacidad igual o semejante a las memorias anteriormente vistas, esta memoria está recomendada frente a las EEPROM cuando se precisa una gran cantidad de memoria de programa no volátil ya que es más rauda y tolera más periodos tanto de borrado como escritura.

2.4.3. Puertas de entrada y salida.

La cápsula que contiene un microcontrolador tiene dos terminales que tiene como finalidad recibir información, dos terminales mas donde estará el cristal de cuarzo que es el encargado de normalizar la frecuencia de operación y por último una terminal más que servirá para el Reset, las demás terminales servirán para la comunicación al procesador con el mundo exterior a través de interfaces o con otros dispositivos, según los controladores de periféricos estas líneas de E/S tienen como beneficio primordial el dar soporte tanto a las señales de control como a las de entradas y salidas.

2.4.4. Reloj principal.

En la fabricación de los microcontroladores se debe tener en cuenta que los mismos deben incorporar un circuito oscilador cuya función será el generar una onda cuadrada de alta frecuencia que a su vez permitirá

desarrollar una estructura de los impulsos de reloj que llegarán a ser utilizados como sincronización en las operaciones que se desarrollaran en el sistema.

Si bien es cierto, este circuito de reloj está incorporado en el microcontrolador lo que no requerirá de muchos componentes exteriores que permitan la selección y estabilización de esa frecuencia entregada.

Los materiales usados en estos componentes suelen ser de cristal de cuarzo en conjunto con elementos pasivos.

Si la frecuencia entregada de reloj aumenta, se necesitará reducir el tiempo de ejecución de las instrucciones ya que aumentaría el consumo de la energía y el calor generado.

2.5. Elementos auxiliares.

Cada modelo de microcontrolador es manufacturado según las aplicaciones a las que orienta el fabricante, estos microcontroladores poseen una variedad de elementos complementarios que fortalecerán la potencia y flexibilidad del dispositivo sin dejar a un lado el poder minimizar el costo, el hardware y software.

A continuación, se detallan estos elementos auxiliares que incorpora un microcontrolador.

2.5.1. Temporizador o timers

El temporizador del microcontrolador no es más que un contador que permite el control de periodos de tiempo además de tener control de los acontecimientos que suceden en el exterior.

Este se acciona ya sea con un pulso de reloj externo u oscilador interno, pueden ser de 8 bits o 16 bits de ancho y los datos pueden ser cargados mediante la utilización de un programa. En cuanto a la configuración para crear esa interrupción cuando alcance un determinado valor o desbordamiento el usuario puede utilizar un software que le permita realizar la interrupción para obtener datos fundamentales en el tiempo en el que está operando el microcontrolador.

2.5.2. Perro guardián o watchdog.

Un temporizador de vigilancia o perro guardián proporciona un medio de recuperación adecuado de un problema del sistema. El problema del sistema podría ser un programa que entra en un bucle sin fin, o un problema de hardware que impide que el programa funcione correctamente. Si el programa no logra restablecer el organismo de control en algún intervalo predeterminado, se iniciará un restablecimiento de hardware. El error todavía puede existir, pero al menos el sistema tiene una forma de recuperar. Esto es especialmente útil para sistemas sin supervisión.

2.5.3. Protección ante fallo de alimentación o “Brownout”.

Protección brownout es por lo general un circuito de protección que restablece el dispositivo cuando el voltaje de operación (V_{cc}) es inferior a la tensión baja de voltaje.

El dispositivo se mantendrá reinicio y permanecerá ahí cuando el V_{cc} se siga manteniendo por debajo de la tensión baja de voltaje. El dispositivo se reanudará o se ejecutará después de que el V_{cc} se haya elevado por encima de la de tensión baja de voltaje.

2.5.4. Estado de reposo o de bajo consumo.

Existen muchas situaciones en la que el microcontrolador se encuentra activo sin nada que hacer, solo esperar a que se produzca algún suceso externo que le permita entrar de nuevo en operación, de manera que para poder ahorrar energía cuando el microcontrolador no esté funcionando existen comandos especiales como (SLEEP en los PIC) que va a permitir entrar en estado de reposo o bajo consumo.

Si llega a haber algún suceso externo el microcontrolador sale del estado de reposo y empieza a trabajar.

2.5.5. Conversor Analógico – Digital.

Existe una gran diversidad de microcontroladores que poseen este conversor que les permite procesar señales analógicas que suelen encontrarse en grandes cantidades en aplicaciones. Muchos de estos microcontroladores disponen de un multiplexor que permite introducir a la entrada de este conversor diferentes señales analógicas.

2.5.6. Conversor Digital – Analógico.

Básicamente realiza la transformación de datos digitales que se obtienen del proceso de un computador a una señal analógica que es enviada al exterior de la cápsula del microcontrolador por medio de sus terminales, existen efectores que pueden laborar con señales analógicas.

2.5.7. Comparador analógico.

Comúnmente en el interior de ciertos microcontroladores existe un amplificador operacional que tiene como una sencilla función comparar una señal que se encuentre fija de referencia con una variable que se introduce por una de las terminales de la cápsula. El resultado de este comparador estará dado por un nivel lógico en este caso 1 o 0 que va a depender mucho de sí la señal sea menor o mayor que la otra.

2.5.8. Fuente de alimentación.

Los microcontroladores tienen la capacidad de entrar en función con una tensión de 5V. Gran variedad de microcontroladores pueden funcionar con una tensión de 2.7V, otros pueden resistir trabajar con una tensión de 6V sin ningún inconveniente.

Pues bien, para saber con cuanto voltaje trabaja el microcontrolador siempre el fabricante dispondrá de una hoja de información donde su contenido permitirá a los usuarios ver cuáles son los límites permitidos de la tensión de alimentación.

2.5.9. Reset Input (Reseteo de entrada).

Básicamente el Reset del microcontrolador se utiliza para poner al mismo en estado conocido. Eso significa que si el microcontrolador empieza a comportarse de manera imprecisa bajo ciertas condiciones indeseables, para poder continuar su buen funcionamiento tiene que ser reseteado, es decir, todos los registros del microcontrolador se colocarían en una posición original o de partida "0" de la memoria del programa.

Este reseteo se logra mediante un interruptor pulsador que es colocado en la entrada de la terminal de reset, cuando se pulsa este interruptor el microcontrolador se resetea.

Restablecer no sólo se utiliza cuando el microcontrolador no se comporta de la manera que queremos, también se puede utilizar cuando se trata de un dispositivo como una interrupción en la ejecución del programa, o para tener un microcontrolador listo antes de cargar algún programa.

2.5.10. Interrupciones.

Tiene como función hacer que el microcontrolador pueda suspender si así se configura, alguna tarea que se esté ejecutando en cualquier momento para así poder ejecutar otra tarea que es asignada por algún periférico.

Cada uno de los periféricos de los microcontroladores tiene su correspondiente interrupción, algunos de ellos pueden tener más de una, estas interrupciones pueden ser habilitadas o deshabilitadas pueden hacerse de manera general o individual, pero dependerá de que el usuario asigne mediante programación la prioridad de cada interrupción con la que se quiera trabajar.

2.6. Tipo de microcontroladores.

Como los microprocesadores, existe una familia de microcontroladores, cada una de estas familias poseen diferentes chips y fuentes de apoyo para desarrollar determinado sistema en el microcontrolador. Para poder elegir el dispositivo apropiado para cumplir con los requisitos del sistema, debemos

entender las diferencias y las diferentes opciones y características de varios microcontroladores

Dependiendo de la potencia y las características que se necesitan, se podría elegir un microcontrolador de 4 bits, 8 bits, 16 bits o 32 bits. Además, ciertas versiones especializadas de estos microcontroladores incluyen características para las comunicaciones, manejo de teclado, procesamiento de señales, procesamiento de vídeo, y otras tareas.

2.6.1. Microcontroladores de 4 bits.

Los microcontroladores de 4 bits fueron los primeros en introducirse en el mercado y todavía se utilizan en aparatos pequeños y juguetes, su memoria RAM va desde los 32 hasta los 128 bytes, así como su memoria ROM va desde los 512 hasta los 2k bytes. Muchos de los fabricantes para agregarle diversidad a sus microcontroladores los desarrollaban con pantalla led, E/S de bits en serie entre otros.

2.6.2. Microcontroladores de 8 bits.

El primer microcontrolador de 8 bits fue desarrollado por Intel, lanzado en la década de los 70 y fue diseñado para el control de tareas generales. Este microcontrolador de 8 bits ha demostrado ser muy útil ya que los datos ASCII se encuentran almacenados en formato de 8 bits. Esto hace que el

microcontrolador de 8 bits sea la selección natural para la comunicación de datos. Muchos de los fabricantes de este tipo de microcontrolador ampliaron los bytes tanto de la memoria RAM como la del ROM además de incluir puerto serial para memoria externa, E/S de bits en serie etc.

2.6.3. Microcontroladores de 16 bits.

Estos tipos de microcontroladores pueden ser usados en una gran variedad de aplicaciones que involucren limitados cálculos y aplicaciones de control relativamente sencillas.

Los microcontroladores de 16 bits están diseñados para aplicaciones de rendimiento de alta velocidad. En general estos microcontroladores proporcionan grandes espacios para la memoria de datos, el costo de este chip de microcontrolador es mucho más barato que el de las generaciones anteriores.

2.6.4. Microcontroladores de 32 bits.

Estos microcontroladores están diseñados para aplicaciones como el control de los robots, instrumentación inteligente, procesamiento de imágenes, telecomunicaciones, coches automáticos y otros sistemas de control de gama alta.

En los microcontroladores de 32 bits hablando del diseño cambia de características en el chip como RAM, ROM, temporizadores, y puertos serie, entre otras características.

2.7. Familia de Microcontroladores.

2.7.1. Microcontroladores PIC.

Estos microcontroladores PIC son una excelente elección para empezar proyectos usando microcontrolador debido a su pequeño tamaño, alto rendimiento y bajo costo en el mercado, estos fueron desarrollados por la empresa Microchip Technology, cuya principal fábrica de producción está localizada en Arizona. Esta compañía se dividió en 4 grandes familias cada una de éstas con su variedad de componentes que proporcionan un diseño de microcontrolador con características especiales:

- **La primera familia, PIC10 (10FXXX) - es llamada de gama baja.**

Las principales características de estos dispositivos es que eran de bajo costo, pero de alto rendimiento, de 8 bits, totalmente estáticos, eran microcontroladores CMOS basados en memoria flash. Emplean una arquitectura RISC anteriormente vista con 33 únicas palabras y un único ciclo de instrucciones.

Las instrucciones de un ancho de 12 bits son altamente simétricas. Es un conjunto de instrucciones fáciles de usar y de recordar, reduce significativamente el tiempo de desarrollo. Estos dispositivos contienen una ALU de 8 bits y un registro de trabajo.

- **La segunda familia, PIC12 (PIC12FXXX) – es llamada de gama media.**

Estos dispositivos de gama media disponen de una arquitectura de programa de 14 bits y están disponibles en paquetes de 8 hasta 64 pines que ofrecen una tensión de funcionamiento con rango de 1.8-5.5V, manejo de interrupciones, múltiples canales A/D, pequeños paquetes de huellas y memoria de datos EEPROM, en general ofrecen una amplia gama de opciones de paquetes y de integración de periféricos, ya sean analógicos o digitales de serie como: convertidores LCD y A/D, SPI (interfaz periférica serial) entre otros.

- **La tercera familia es PIC16 (16FXXX).**

Con seis variantes que van desde 3.5k-14kbytes de memoria flash hasta 256bytes de RAM y una mezcla de periféricos incluidos como EUSART, CCP y comparadores analógicos, estos dispositivos son muy adecuados para los diseñadores con aplicaciones que necesiten más espacio de código, E/S

de 14 pines además de que estén buscando aumentar el rendimiento del sistema y la eficiencia de código mediante el empleo de hardware de control de motor y capacidad de comunicación.

➤ **La cuarta familia es PIC 17/18 (18FXXX).**

Esta familia de PIC utiliza un programa con una arquitectura de 16 bits e incorpora una arquitectura avanzada RISC, multiplicador de hardware de 8x8 y múltiples interrupciones tanto internas como externas, con el más alto rendimiento en el catálogo de 8 bits de microchip esta familia PIC18 ofrece hasta 16 MIPS y memoria lineal. PIC18 es la arquitectura más popular para los nuevos diseños de 8 bits donde los clientes quieren programar en lenguaje C, incorpora unas 77 instrucciones y posee periféricos de comunicación avanzados.

2.7.2. Microcontroladores Atmel AVR.

Un bajo consumo de energía y un alto nivel de integración, los microcontroladores Atmel AVR de 8 bits y 32 bits (MCU) son dispositivos que ofrecen una combinación única de rendimiento, eficiencia energética y flexibilidad de diseño. Se basan en el mayor eficiente código de arquitectura de la industria que es el C y lenguaje ensamblador. Ningún otro microcontrolador ofrece más rendimiento informático con una mejor eficiencia energética.

Estos microcontroladores poseen 140 instrucciones y todas ellas son basadas en 1 ciclo. Por defecto los microcontroladores AVR operan con el ciclo de reloj de 1 MHz. La velocidad del microcontrolador AVR es de 12 millones de instrucciones por segundo. La familia AVR tiene en chip cargador de arranque. Por esto podemos programar nuestro microcontrolador fácilmente sin ningún programador externo.

Los controladores AVR tienen varios puertos, temporizadores / contadores, interrupciones, convertidores de E/S, A/D, USART, interfaces I2C, canales PWM, comparadores analógicos en el chip.

A continuación, se detallarán las características especiales de algunas de las familias de los Atmel AVR:

➤ **Familia AVR UC3 de 32 bit.**

Este microcontrolador toma la eficiencia a un nuevo nivel ya que más allá del alto rendimiento y el bajo consumo de energía tiene un soporte nativo fijo SRAM de doble puerto, múltiples capas de bus de datos, controlador periférico DMA. El controlador periférico DMA y la multi-capa de la arquitectura de bus de alta velocidad hacen que la categoría de microcontroladores UC3 sean ideales para aplicaciones de alto rendimiento. Ciertos dispositivos UC3 incluyen una unidad de punto flotante integrada (FPU), que mejora el rendimiento de la aritmética de los números decimales, con mayor precisión y rango dinámico más amplio.

➤ **Familia AVR XMEGA MCU.**

Estos dispositivos ofrecen una combinación líder de rendimiento en tiempo real, integración y eficiencia energética.

El diseño altamente integrado incluye características como cifrado AES y DES, módulos analógicos de alta velocidad, temporizadores flexibles y contadores, múltiples módulos de comunicación.

El controlador DMA y el innovador sistema de eventos garantizan un rendimiento de tiempo real predecible incluso con alta carga, en la serie de microcontroladores XMEGA existe una personalización lógica que hace que sea posible realizar funciones lógicas básicas.

➤ **Mega AVR MCU.**

Estos microcontroladores son la opción ideal para los diseños en que se necesita un poco de fuerza adicional, ya que son utilizados para aplicaciones que requieren grandes cantidades de código, estos dispositivos ofrecen programas y datos de memoria importantes con un rendimiento de hasta 20 MIPS, mientras tanto la tecnología utilizada en estos dispositivos minimizan el consumo de energía, todos estos dispositivos ofrecen auto programación rápida y segura, actualizaciones rentables en circuito e incluso puede ser actualizada la memoria flash mientras se ejecuta alguna aplicación.

➤ **TinyAVR MCU de 8 bit.**

Todos los dispositivos TinyAVR se basan en una misma arquitectura y son compatibles con otros dispositivos AVR, estos contienen una memoria EEPROM un ADC integrado, un detector de salida que permite crear aplicaciones sin necesidad de añadir componentes externos.

El TinyAVR ofrece memoria flash, actualizaciones rápidas son compatibles con plataformas de desarrollo y permite el desarrollo de código en C o en algún lenguaje ensamblador, proporciona la simulación de un ciclo exacto y se integra perfectamente con kits de iniciación de evaluación y diseños de referencia.

La potencia y el rendimiento analógico hacen que el TinyAVR sea el dispositivo más compacto y rico en funciones en la familia AVR ya que es el único dispositivo capaz de funcionar con solamente 0.7V.

2.8. La robótica.

2.8.1. Reseña histórica.

El licenciado en ciencias de la computación (Zabala, 2007) menciona en su guía teórica y práctica que la robótica ya ha llegado a nuestras casas y trabajos, nos rodea, y es parte de nuestra vida diaria; y ya que nosotros

comprendemos, aceptamos y apoyamos esta realidad está demás decir que es inevitable que esto siga sucediendo y progrese con el transcurso de los años. Desde diminutos juguetes hasta maquinarias enormes, ocupando lugar en instituciones educativas, industrias y nuestros hogares, pasando por nuestra mano, la de nuestros hijos e incluso en adultos mayores, es muy difícil lograr definir cuándo empezó nuestra relación con la robótica teniendo en cuenta la gran cantidad de disciplinas que esta engloba.

Cuando se empieza a hablar de robótica surge inmediatamente la esencial interrogante ¿Qué es un robot? Podemos afirmar que desde los orígenes el hombre visualizó para el futuro la creación de una inteligencia artificial y curiosamente retratada a su imagen y semejanza. Ejemplos muy claros de esto se pueden encontrar remontándonos a largometrajes muy antiguos como Rossum's Universal Robot y a famosas obras literarias como Frankenstein que posteriormente vería adaptaciones en pantalla chica y grande.

Esta visión primitiva de los robots surge de la necesidad del hombre de crear máquinas que cumplan sus actividades más arduas y sean eficientes en ello, pero en la actualidad sabemos que este modelo humanoide no era una característica fundamental en los robots.



Figura 2. 4: Adaptacion de Frankenstein
Fuente: (Whale, 1931)

Décadas más adelante llegaría el auge del cine y con este llegarían a la cultura general robots como R2-D2, un ícono de la saga cinematográfica Star Wars, igual que su compañero humanoide C3PO. Posteriormente se explotaría el aspecto amenazante y peligroso de la inteligencia artificial y es así como surgiría la famosa cinta Terminator y otras con similar argumento permitiéndole a los seres humanos experimentar nuevas realidades fantásticas donde los robots se podían desenvolver, cooperar e incluso evolucionar.



Figura 2. 5: Robots de largometraje original de Star Wars
Fuente: (Lucas, 1977)

La distancia en la carrera entre fantasía y realidad cada vez se acorta más ya que actualmente si se introduce un robot en un largometraje, ya no se busca simular sus habilidades, en su lugar se construye al robot directamente y éste pasa de ser un ser humano dentro de un disfraz o un personaje generado por computadora a un miembro tangible más del elenco.

2.8.2. ¿Qué es un robot?

Una vez que salimos de la ciencia ficción la realidad es que un robot no tiene una estructura, apariencia ni mucho menos una función específica en su definición. Al ser esto así, existen muchas definiciones de la palabra robot, donde se destacan aspectos particulares que son de interés del que lo busca definir.

La Real Academia Española define la palabra robot como “Máquina o ingenio electrónico programable, capaz de manipular objetos y realizar operaciones antes reservadas solo a las personas.” Como podemos apreciar, esta definición es bastante simple y muy limitada, utilizando nuestros conocimientos sabemos que necesitamos una definición más compleja que encierre varios aspectos, que sea más flexible y mantenga su esencia. En vista de esto hemos decidido aceptar la bastante completa definición que proporciona la Asociación Japonesa de Robótica Industrial (JIRA), “*Los robots son dispositivos capaces de moverse de modo flexible análogo al que poseen los organismos vivos, con o sin funciones intelectuales, lo que permite la*

realización de operaciones en respuesta a órdenes recibidas de los humanos”.

En esta definición podemos apreciar que se busca la similitud con los humanos, pero esto no implica aspectos de forma, apariencia ni intelecto; destaca que la función de un robot es en base a las órdenes de un ser humano. Si nos regimos por esta base jamás cruzaremos esa línea de peligro que la ciencia ficción dibujó casi un siglo atrás donde nuestra integridad se ve amenazada por la presencia asfixiante de robots.

2.8.3. Tipos de robot.

El autor (Somolinos S., 2002) en su libro “Avances en robótica y visión por computador” agrupa los robots en 3 categorías basándose en intervalos de años y su tecnología:

- 1ra generación: desde 1960 hasta 1980; comprende los robots industriales que realizan tareas repetitivas. Se desarrollan las funciones de manipulación fundamentalmente.
- 2da generación: entre 1980 y 1985; incluye a los robots dotados de sensores que interactúan con el entorno. El robot puede modificar su accionar en tiempo real en función de los cambios en su entorno. Aquí se desarrollan funciones perceptivas y planificativas.
- 3ra generación: comprende la época desde 1985 hasta la actualidad; construcción de robots móviles lo que deriva en inteligencia artificial

aplicada a la robótica. Constantes mejoras en percepción y planificación.

Así también define 3 partes fundamentales en la constitución de un robot: el sistema mecánico, los accionadores y un sistema de control. En la sección del sistema mecánico se encuentran las partes rígidas del robot ya sean brazos, muñecas, pinzas de fijación, barras, cámara, manguera, ventosa etc.

Los accionadores son todos aquellos dispositivos que le brindan el movimiento a cualquier articulación del robot. Generalmente un elemento motriz y un sistema de transmisión. Dependiendo el tipo de movimiento puede ser neumático, hidráulico o eléctrico, mientras que sus accionamientos eléctricos vienen de distintos tipos de motores.

Las partes pueden variar conforme la tecnología avance ya que en la actualidad los dispositivos sensoriales son un común denominador en los robots, debido a que la habilidad de recibir información del ambiente, procesarla y realizar acciones distintas ubicadas en su programación encuentra cada día más campos de aplicación. Asemejándolo más al modelo de una computadora que consta de entradas, procesos y salidas.

Existen además otros autores que varían su división cronológica e incluyen un mayor número de generaciones basándose en la evolución del movimiento,

tecnología sensorial y el sistema de control, pero no es de importancia subdividir estos cambios generacionales.

Analizando la estructura, la tecnología y su funcionamiento también podemos clasificar a los robots por su trabajo realizado.

- Androides: robots cuyo accionar refleja la actividad humana, se los considera artilugios que no brindan una utilidad real, muy ligado a la ciencia ficción y a la investigación.
- Móviles: robots que cuentan con partes que le permiten desplazarse en un espacio siguiendo su programación definida. Ruedas, patas, sensores y bandas los caracterizan.
- Industriales: utilizados en procesos de fabricación y manipulación a gran escala. Cabe recalcar que todos aquellos robots fuera de esta clasificación también se los conoce de forma generalizada como no industriales.
- Domésticos: los dispositivos caseros como lavadoras, heladeras o aspiradoras que cambian su comportamiento según el ambiente de trabajo.
- Médicos: aquí se incluyen las prótesis para personas con movilidad reducida y a robots que brindan asistencia quirúrgica o intervenciones completas.

- Tele-operadores: como su nombre lo indica son aquellos que son operados remotamente por un humano. Amplia utilidad en situaciones extremadamente peligrosas.
- Poli-articulados: son diseñados con gran movilidad en sus terminales, pero son sedentarios en su totalidad.
- Zoomórficos: con una apariencia muy similar a animales conocidos ya sean caminadores o no caminadores brindan utilidad en labores de exploración y reconocimiento de terreno.

Aquí hemos visto algunas de las clasificaciones más generalizadas de los robots enfocándonos en su generación, utilidad y estructura; pero también existen muchas otras formas de agruparlos basándonos en criterios más específicos como lo son: inteligencia, programación, arquitectura, grado de movilidad, tamaño, geometría, consumo, o el medio en el que desarrollan la actividad.

2.8.4. Impacto de la robótica.

Desde el instante en que los robots incursionaron en el campo industrial se comenzó a dar pasos gigantados en lo que respecta a la automatización; de esta manera las conocidas empresas multinacionales complican la situación de las demás para poder continuar sobreviviendo competitivamente ya que se ven obligadas a adoptar sus métodos. La gran ventaja del avance

en la automatización es que genera un increíble aumento en la productividad y también en la calidad, haciendo así a la empresa más competitiva.

Esta automatización trae consigo un gran impacto socio laboral ya que conforme se invierte más en la automatización industrial se ven drásticamente reducidos los puestos de empleo. Se puede decir que este desempleo causado en el área industrial se compensa con el aumento de puestos en otras áreas como la educativa, mantenimiento, servicios y manufactura de robots.

El sector educativo y de elaboración de robots ha llegado al ámbito competitivo estableciendo varios concursos de robótica reconocidos a nivel nacional e internacional, siendo éste un entorno de creatividad, ingenio y conocimientos donde mentes jóvenes y adultas ponen cara a cara sus robots diseñados y programados con mucha dedicación haciendo uso de sus conocimientos de electricidad, electrónica e informática de manera innovadora.

2.9. Robótica Móvil

Los robots móviles se han convertido en objeto de gran interés gracias a los desafíos que abarca y al gran rango de posibles aplicaciones. Los robots pueden ser una solución a problemas de transporte ya sea de personas, animales, piezas y demás objetos en general. Otro potencial campo de

aplicación son las labores exploratorias siendo las de reconocimiento de terreno de otros planetas las más conocidas; también se los utiliza para adentrarse en ambientes tóxicos peligrosos para los humanos y secciones inaccesibles dentro de lugares como plantas nucleares. Finalmente, el área de aplicación más desafiante es aquella que involucra la inteligencia artificial, ya que los robots móviles necesitan basar sus movimientos en un sistema de toma de decisiones basándose en los cambios de su entorno y de esta forma dejar atrás el control remoto.

Los robots móviles son equipados con un sistema de automoción que le permite desplazarse en un espacio. Este sistema varía en función del tipo de superficie donde se desplazará; para superficies lisas se utilizan las ruedas, las cuales están disponibles de diferentes tipos dependiendo de la naturaleza del terreno. Para superficies irregulares se utilizan orugas ya que el robot enfrentara gran variedad de obstáculos en su terreno. Finalmente, para superficies extremadamente irregulares se dispone de patas las cuales permiten sobrepasar interrupciones en la vía de desplazamiento del robot evadiendo así obstáculos mucho más pronunciados del terreno

El robot móvil es un robot programable para ejecutar una serie de acciones mediante control automático, partiendo de aquí se hace una marcada distinción entre vehículos guiados y no-guiados.

El vehículo guiado se ve restringido a seguir una serie de rutas o caminos predefinidos en un área o espacio específico; estos caminos pueden ser pistas, líneas magnéticas u ópticas o una secuencia de movimientos almacenados en la memoria.

2.9.1. Vehículos Guiados.

Un aspecto fundamental cuando hablamos de vehículos guiados es el requerimiento sensorial aplicado a la robótica. Entre los que destacan presencia, rango, medición de un eje, ubicación y posición (bidimensional), ubicación y posición (tridimensional), termal y fuerza. Para cada uno existen distintos tipos de dispositivos sensoriales y métodos aplicables a la robótica.

2.9.2. Sensores aplicables a la robótica.

En la categoría de visión tenemos foto-detector, cámara, triangulación laser o tiempo de vuelo.

Foto detector: dispositivo que brinda una respuesta eléctrica en función de la incidencia de radiación óptica en su superficie sensorial.

Cámara: dispositivo que captura imágenes estáticas o en movimiento dentro del espacio tridimensional.

Triangulación laser: método por el cual se determina posición y medidas mediante la utilización de varios captosres y un principio de triangulación óptica sin recurrir al contacto directo.

Tiempo de vuelo: se utiliza un sensor laser para determinar distancias mediante la medición del tiempo de vuelo.

En lo que respecta a la acústica existen detectores y emisores ultrasónicos, y control por voz (micrófono). También están a disposición otros sensores como el infrarrojo o el radar; y métodos como la proximidad magnética y la radiación ionizante.

Sensor infrarrojo: son dispositivos que transforman a una señal eléctrica la radiación proveniente de materiales calientes. Este sensor necesita una comunicación lineal entre emisor y receptor, esto se traduce específicamente a una línea de vista estable.

Radar: dispositivo que utiliza las ondas electromagnéticas para mediciones de distancia, posición y velocidad. Actualmente tienen una muy amplia utilidad en múltiples áreas como la medicina, climatología, geografía, militar, entre otras.

2.9.3. Motores CD.

Los motores CD convencionales se asemejan mucho a los generadores CD en términos de construcción. De hecho, es complicado identificarlos por apariencia solamente. Un motor posee las mismas partes principales que un generador.

El motor CD es un dispositivo que se encarga de la conversión de energía eléctrica a energía mecánica; y esto es gracias a dos partes principales en su constitución física. El estator es la parte estática que contiene los devanados y el rotor es la parte dinámica del motor que se encarga de las rotaciones mecánicas.

Los motores CD se clasifican en función de su voltaje, corriente, velocidad y potencia de salida.

2.9.4. Motores de Engranaje.

Los motores de engranaje son un tipo de motores eléctricos que utilizan el mismo principio de generación de energía con la diferencia que la corriente generada por los campos magnéticos actúa sobre engranajes conectados a otro eje, dando como resultado un enorme aumento en la fuerza de torsión, mientras que simultáneamente el motor produce una reducción en su velocidad de salida.

Finalmente obtenemos un motor que se mueve más lento, con menor necesidad de corriente y con una mayor torsión de salida lo que se resume en una fuente de poder compacta y eficiente.

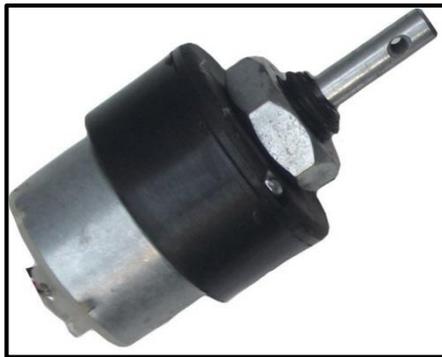


Figura 2. 6: Motor de engranaje CD

Fuente: (V.T, 2015)

2.10. Seguidores de línea.

Un robot seguidor de línea es una máquina móvil que puede detectar y seguir la línea dibujada en el piso. Generalmente el camino es predefinido y puede ser una línea negra visible en una superficie blanca con un alto contraste de colores o también un campo magnético invisible. Este tipo de robots debe detectar la línea con su sensor infrarrojo instalado en su parte inferior. Posteriormente la información es transmitida al procesador, donde se decide los comandos apropiados y los envía al controlador y así el camino será seguido por el robot seguido de línea (Pakdaman & Sanaatiyan, 2009).

Existen varios métodos que pueden ser utilizados para elaborar un robot seguidor de línea. El modelo de un sensor de luz, de dos sensores y de

tres sensores. De estos tres modelos se considera el más versátil, confiable y más práctico de construir, el de dos sensores.

2.10.1. Modelo de un sensor.

El seguidor de línea de un solo sensor utiliza un patrón Z para continuamente salir y entrar de la línea que debe seguir. Este modelo es relativamente simple y consiste en 3 elementos claves: una unidad de control de bucle infinito, un estado de cambio utilizando un sensor para detectar el color negro y un bloque de control de dos motores para girar izquierda o derecha.

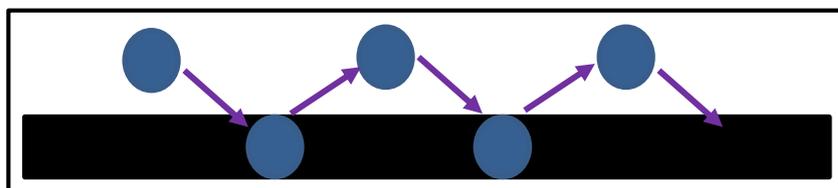


Figura 2. 7: Patrón Z con un sensor.

Fuente: (Eskatul, 2011), Los Autores

2.10.2. Modelo de dos sensores.

En este modelo se instalan 2 sensores para que estén en ambos lados de la línea negra que el robot está destinado a seguir.

Detectando cual sensor cambia de blanco a negro sabemos cómo volver a la línea, este robot es capaz de seguir cualquier línea realizando cualquier tipo de giros.

En este modelo existen cuatro posibles estados que pueden ocurrir en los sensores:

- Izquierda Negro – Derecha Blanco: el robot necesita hacer un giro a la izquierda para posicionarse nuevamente en el centro de la línea.
- Izquierda Blanco – Derecha Negro: el robot necesita hacer un giro a la derecha para posicionarse nuevamente en el centro de la línea.
- Izquierda Blanco – Derecha Blanco: el robot está perfectamente centrado sobre la línea y puede continuar recto hacia adelante sin tomar acción correctiva.
- Izquierda Negro - Derecha Negro: esta situación determina que el robot llegó a una intersección con otra línea negra. Este es un caso especial y puede ser utilizado para ejecutar una acción especial.

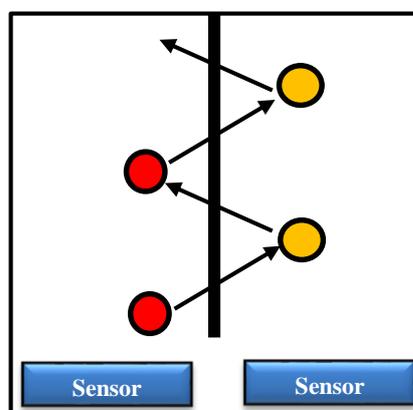


Figura 2. 8: Patrón de movimiento del modelo de dos sensores

Fuente: (Eskatul, 2011), Los Autores

CAPÍTULO 3: DISEÑO Y CONSTRUCCION DE DOS ROBOTS SEGUIDORES DE LÍNEA

3.1. Materiales y dispositivos electrónicos para la implementación de los robots móviles.

3.1.1. Microcontrolador ATmega328/P.

El ATmega328/P es un microcontrolador CMOS de 8 bits de baja potencia basado en AVR, con una mejorada arquitectura RISC ya que mediante la ejecución de instrucciones poderosas en un solo ciclo de reloj, este microcontrolador logra un gran rendimiento que se acercan a 1 MIPS por MegaHertz que proporciona al creador del sistema información tanto de procesamiento del microcontrolador como el consumo de energía para llevar a cabo una mejoría del rendimiento, ATmega328P contiene los siguientes componentes:

- Memoria Flash de 32kb para el almacenamiento de programas.
- Memoria RAM de 2kb.
- Memoria EEPROM de 1kb.
- Temporizadores/Contadores Dos de 8 bit y Uno de 16 bits los mismos que son capaces de contar ciclos de reloj interno o de eventos externos y generar una interrupción cuando se alcanza un valor de conteo específico.

- 6 canales de 10 bit convertidor analógico a digital (CAD).
- Puerto de comunicaciones en serie que puede ser usado para comunicaciones con el puerto COM de un ordenador.

No todas estas funciones están disponibles al mismo tiempo, la mayor parte de las terminales del chip están conectadas a múltiples unidades funcionales y depende del diseñador para decidir lo que hace una terminal en particular.

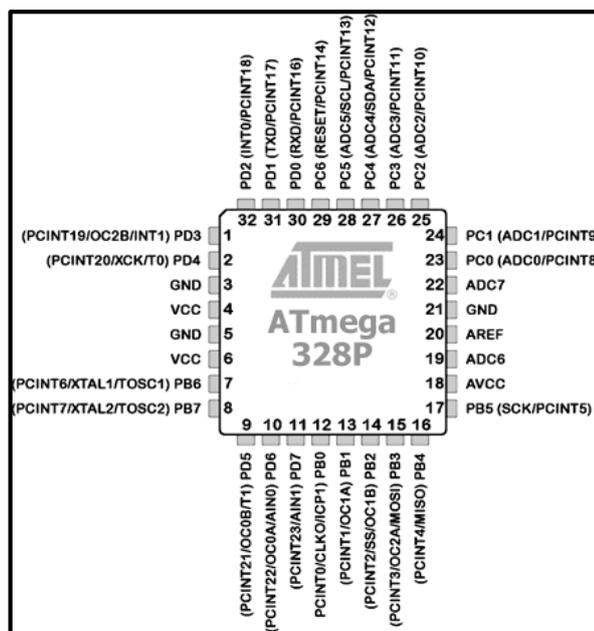


Figura 3. 1: Chip Atmega328/P.

Fuente: (Atmel Corporation, 2016)

Este chip dispone de instrucciones de fácil acceso que contienen registros de trabajo, los mismos que funcionan de manera directa con la unidad aritmética lógica, gracias a ello puede ejecutar de manera independiente ciertos de estos registros en un solo intervalo de tiempo.

Como consecuencia obtenemos una estructura resultante cuyo principal objetivo es el demostrar que el rendimiento del chip es mucho más eficiente que la arquitectura de microcontroladores convencionales.

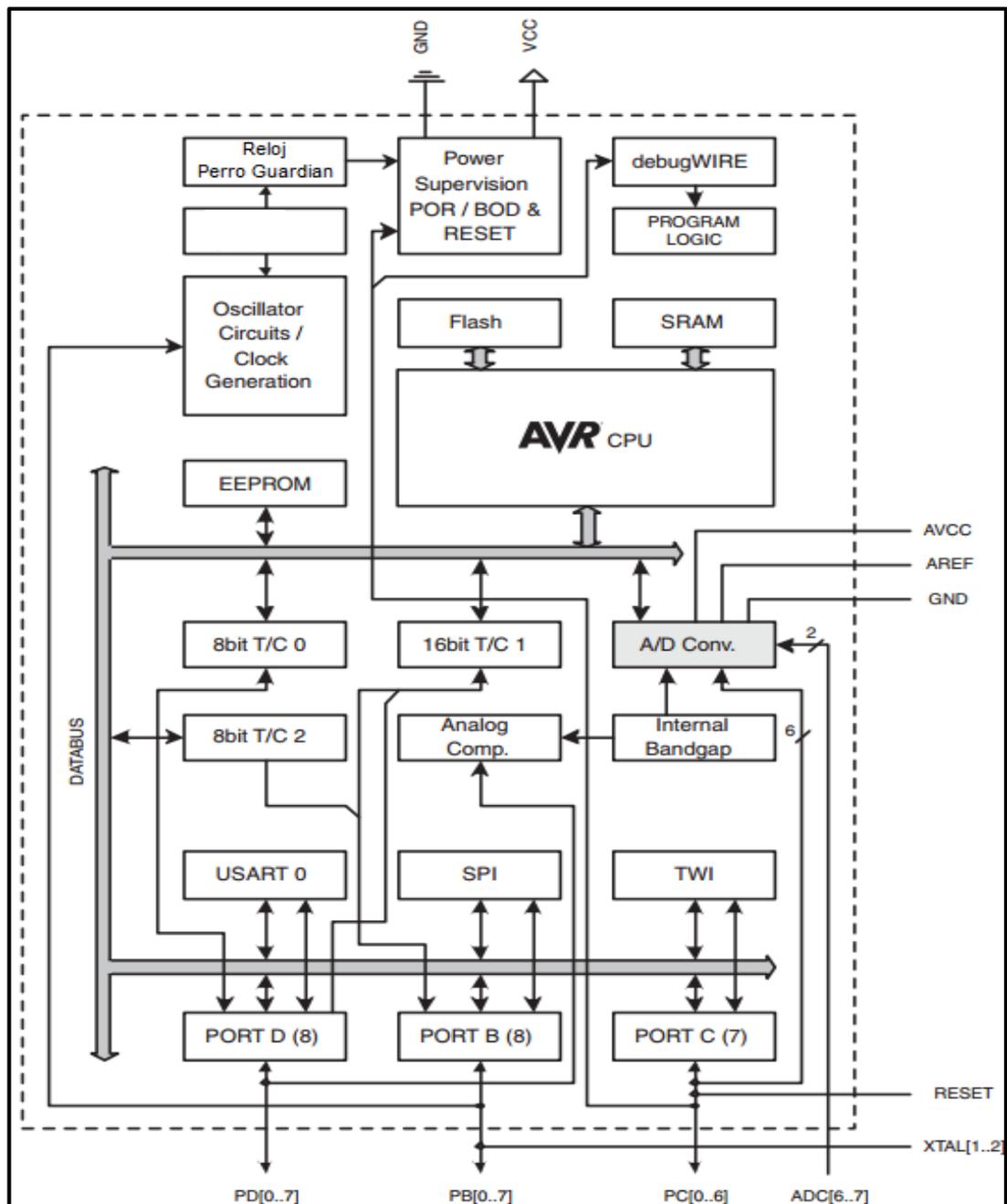


Figura 3. 2: Diagrama de bloque del núcleo AVR.

Fuente: (Atmel Corporation, 2016)

Este microcontrolador ATmega328/P es compatible con una gama completa de programas y sistemas de herramientas de desarrollo que incluye: compiladores C, macro ensambladores, circuitos emuladores y kits de evaluación entre otros.

A continuación se detallan los pines y su funcionamiento del microcontrolador utilizado.

➤ **Puerto B (PB7:0).**

El puerto B es un puerto bidireccional de entrada y salida de 8 bits con resistencias internas elevadoras. Las terminales de este puerto cambian de estado dependiendo si se encuentra activo o no el reseteo, incluso si el reloj no está en funcionamiento. Ciertos pines de este puerto B se pueden utilizar como entrada o salida para el amplificador oscilador inversor y la entrada de circuito operativo de reloj interno.

➤ **Puerto C (PC5:0).**

El puerto C es un puerto bidireccional de entrada y salida de 7 bits con resistencias internas elevadoras. Las terminales de este puerto cambian de estado dependiendo si se encuentra activo o no el reseteo, incluso si el reloj no está en funcionamiento.

➤ **PC6/RESET.**

El PC6 se utiliza como un pin tanto de entrada como de salida, las características eléctricas del PC6 difieren de los demás pines del puerto C ya que se utiliza como entrada de reset. Un nivel bajo de tensión en este pin durante más de la duración mínima de impulso, generará un reinicio incluso si el reloj no se está ejecutando.

➤ **Puerto D (PD7:0).**

El puerto D es un puerto bidireccional de entrada y salida de 8 bits con resistencias internas elevadoras. Las terminales de este puerto cambian de estado dependiendo si se encuentra activo o no el reseteo, incluso si el reloj no está en funcionamiento.

➤ **AVcc.**

El terminal AVcc es un pin de suministro de voltaje de alimentación para el convertidor A/D.

➤ **AREF.**

El terminal AREF es una la referencia analógica del pin para el convertidor A/D.

➤ **ADC7:6.**

Estos terminales sirven como entradas analógicas para el convertidor A/D, estos pines son alimentados desde el suministro analógico y sirven como canales ADC de 10 bit.

3.1.2. Controlador de motores o puente H (TB6612FNG).

Este dispositivo es un gran controlador de motor dual, ya que es capaz de controlar máximo dos motores de corriente continua de 1,2 A (pico 3.2A). Contiene dos señales de entrada que se pueden utilizar para controlar cada uno de los motores en uno de los cuatro estados de funcionamiento CW (sentido de las manecillas del reloj), CCW (contra las manecillas del reloj), short-brake y stop.



Figura 3. 3: Puente H (TB6612FNG).

Fuente: (Toshiba Corporation, 2016)

El suministro de tensión de alimentación (V_{cc}) puede estar en el rango de 2.7-5.5 VDC, mientras que la alimentación del motor (V_M) se limita a un voltaje

máximo de 15VDC. La corriente de salida tiene una clasificación de hasta 1,2 A por canal.

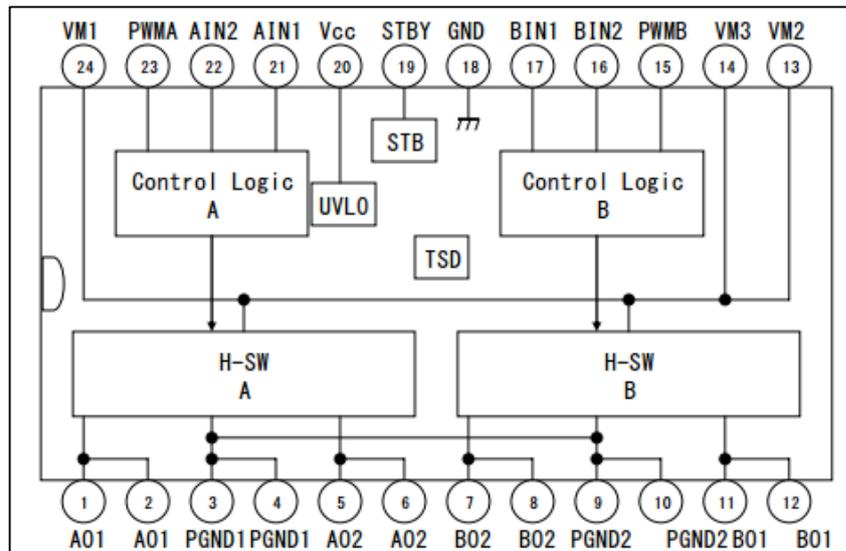


Figura 3. 4: Distribución de pines del Puente H
Fuente: (Toshiba Corporation, 2016)

Tabla 3. 1: Distribución de los pines del puente H (TB6612FNG).

Pin #.	Símbolo	E/S	Indicaciones
1	AO1	S	Canal A Salida 1
2	AO1		
3	PGND1	-----	Energía GND 1
4	PGND1		
5	AO2	S	Canal A Salida 2
6	AO2		
7	BO2	S	Canal B Salida 2
8	BO2		
9	PGND2	-----	Energía GND 2
10	PGND2		
11	BO1	S	Canal B Salida 1
12	BO1		
13	VM2	-----	Alimentador de Motor (2.5V a 13.5V)
14	VM3		
15	PWMB	E	Canal B PWM entrada / 200kΩ
16	BIN2	E	Canal B Entrada 2 / 200kΩ
17	BIN1	E	Canal B Entrada 1 / 200kΩ

18	GND	-----	Pequeña Señal GND
19	STBY	E	Standby / 200k Ω
20	V _{CC}	-----	Pequeña señal de alimentador (2.7V a 5.5V)
21	AIN1	E	Canal A Entrada 1 / 200k Ω
22	AIN2	E	Canal A Entrada 2 / 200k Ω
23	PWMA	E	Canal PWM Entrada / 200k Ω
24	VM1	-----	Alimentador de Motor (2.5V a 13.5V)

Fuente: (Toshiba Corporation, 2016), Los Autores

Tabla 3. 2: Especificaciones técnicas y eléctricas del controlador.

Características	Símbolo	Relación	Unidad
Voltaje de Alimentación	V _M	15	V
	V _{CC}	6	
Voltaje de Entrada	V _{IN}	-0.2 a 6	V
Voltaje de Salida	V _{OUT}	15	
Corriente de Salida	I _{OUT}	1.2	A
	I _{OUT} (pico)	2	
		3.2	
Disipación de Energía	P _D	0.78	W
		0.89	
		1.36	
Temperatura de Funcionamiento	T _{OPR}	-20 a 85	°C
Temperatura de Almacenamiento	T _{STG}	-55 a 150	°C

Fuente: (Toshiba Corporation, 2016), Los Autores

Tabla 3. 3: Especificaciones Técnicas para un rango de funcionamiento (T_a= -20~85°C) del controlador.

Características	Símbolo	Min.		Max	Unidad
Voltaje de Alimentación	V _{CC}	2.7	3	5.5	V
	V _M	2.5	5	13.5	V
Corriente de Salida	I _{OUT}	-----	-----	1	A
		-----	-----	0.4	
Frecuencia de conmutación	f _{PWM}	-----	-----	100	kHz

Fuente: (Toshiba Corporation, 2016)

Elaborado por Autores

3.1.3. Adaptador de USB a serial UART (TTL).

El adaptador o circuito integrado FT232R es un adaptador de USB a Serial UART que no requiere programación específica de firmware ya que todo el protocolo USB es manejado en el chip. Dispone de generación de reloj integrada, buffer de recepción de 128 bytes y buffer de transmisión de 256 bytes, terminales de entrada y salida configurables.

El soporte de interfaz UART es para 7 o 8 bits de datos y admite varias configuraciones de poder. Además, este dispositivo lleva integrado un convertidor de niveles de 3.3V para las entradas y salidas USB, otro para la interfaz entre 1.8V y 5V; compatibilidad USB 2.0 y un rango de operación extendido de -40°C A 85°C.

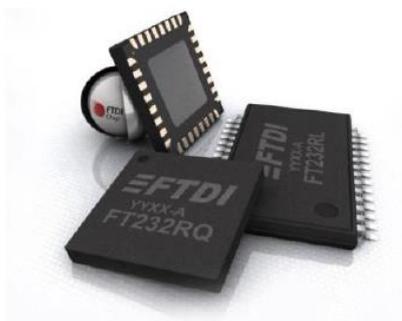


Figura 3. 5: Adaptador USB a Serial UART.

Fuente: (Future Technology Devices Limited, 2015)

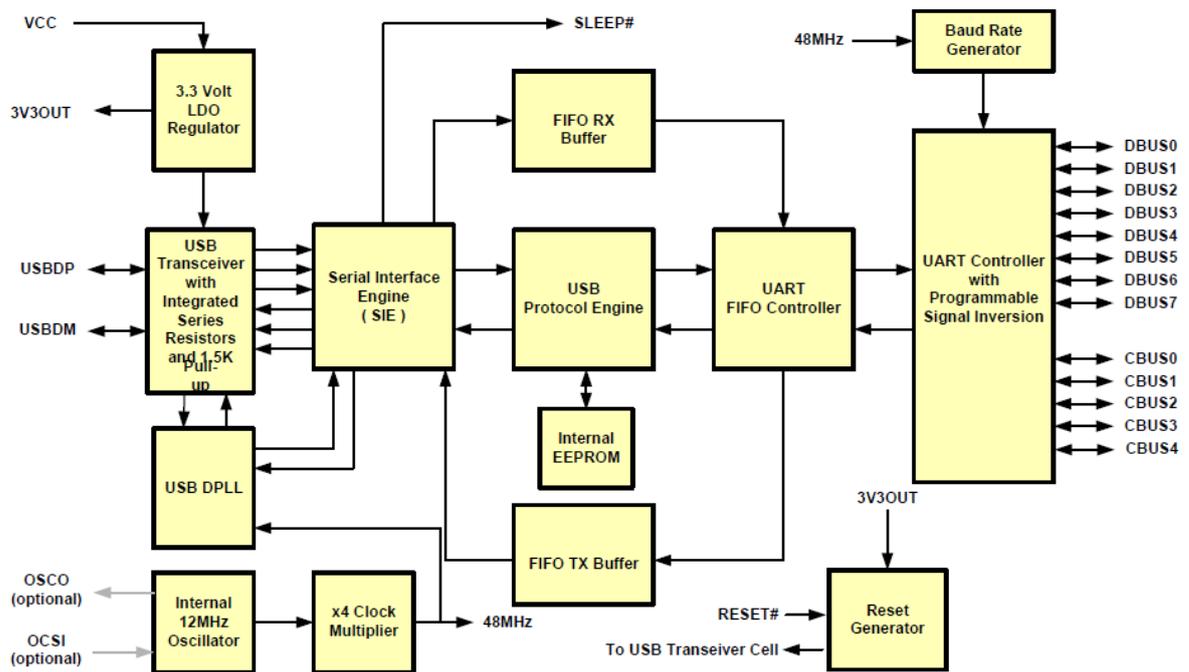


Figura 3. 6: Diagrama de bloques Adaptador FT232RL.

Fuente: (Future Technology Devices Limited, 2015)

Tabla 3. 4: Especificaciones técnicas y eléctricas del adaptador.

Parámetros	Valor	Unidad
Temperatura de almacenamiento	-65 a 150	°C
Temperatura ambiente (poder aplicado)	-40 a 85	°C
Voltaje suministrado VCC	-0.5 a +6.00	V
Voltaje de entrada DC	-0.5 a +3.8	V
Salida de corriente DC	24	Ma
Disipación de poder	500	mW

Fuente: (Future Technology Devices Limited, 2015), Los Autores

Este adaptador nos proporciona en su hoja técnica varias configuraciones posibles de poder USB pero hemos elegido la configuración de poder por bus USB.

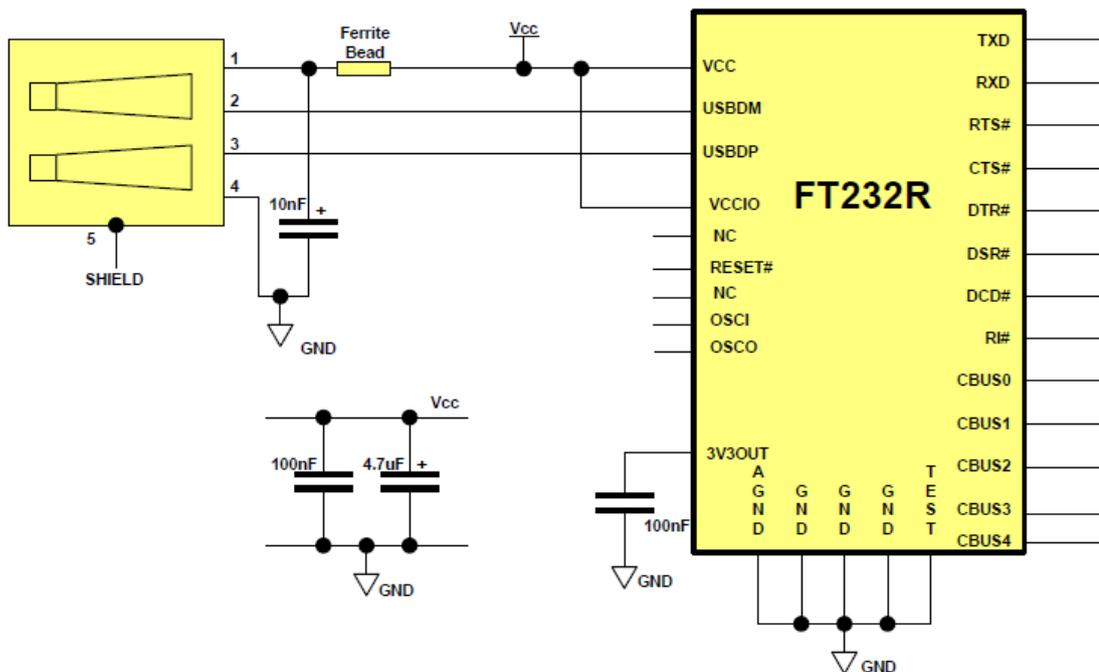


Figura 3. 7: Configuración de poder por Bus USB.
Fuente: (Future Technology Devices Limited, 2015)

En esta configuración típica, el dispositivo es alimentado por el bus USB y no requiere una fuente de poder adicional.

3.1.4. Sensor Óptico Reflexivo.

El sensor QRD1113/QRD1114 es reflectante ya que contiene un diodo emisor de infrarrojos y un fototransistor de silicio NPN montado al lado de otro en un alojamiento plástico negro. La radiación en el eje del emisor y la respuesta en el eje del detector son a la vez perpendicular a la cara del sensor.

El fototransistor responde a la radiación emitida por el diodo solo cuando existe reflejo de un objeto o superficie que se encuentra en el campo de visión del detector.

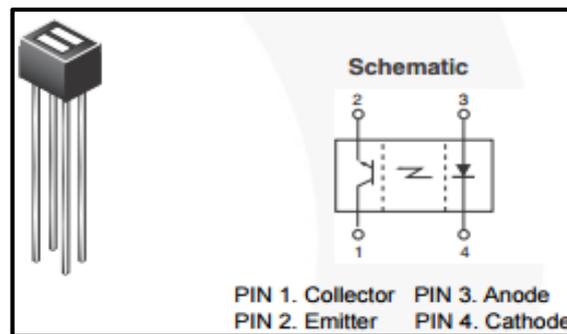


Figura 3. 8: Sensor QRD1113/QRD1114.

Fuente: (Farchild Semiconductor Corporation, 2015)

Existen índices máximos absolutos cuyos valores que superen estos máximos absolutos pueden dañar el dispositivo, la exposición prolongada a las tensiones por encima de las condiciones de funcionamiento recomendadas puede afectar a la fiabilidad del dispositivo.

Tabla 3. 5: Índices máximos absolutos del sensor QRD1113/QRD1114.

Símbolo	Parámetros	Min	Unidad
T_{OPR}	Temperatura de Funcionamiento	-40 a +85	°C
T_{STG}	Temperatura de Almacenamiento	-40 a +100	
Emisor			
I_F	Corriente Directa Continua	50	mA
V_R	Voltaje Inverso	5	V
P_D	Disipación de Energía	100	mW
Sensor			
V_{CEO}	Voltaje Colector-Emisor	30	V
V_{ECO}	Voltaje Emisor-Colector		V
P_D	Disipación de Energía	100	mW

Fuente: (Farchild Semiconductor Corporation, 2015), Los Autores

3.1.5. Motores Pololu CD.

Estos tipos de micro motores de engranaje CD cuyo uso es en especial para la robótica, aparte de ser muy pequeños y ligeros pueden ser muy rápidos, están destinados para trabajar con una alimentación de 6V aunque en general este tipo de motores pueden funcionar a tensiones por encima y por debajo del voltaje nominal, dado por lo que pueden operar cómodamente en un rango de 3-9V (la rotación pueden comenzar con voltajes tan bajos como 0.5V). Las tensiones bajas podrían no ser prácticas y los voltajes muy altos podrían afectar negativamente la vida del motor.

Estos micro motores están disponibles en una amplia gama de engranajes, con valores diferentes de reductora pueden ir desde 5:1 hasta 1000:1 etc.

Tabla 3. 6: Principales características de diferentes micro motores.

Tipo motor	Consumo parado (6V)	Velocidad (6V)	Par motor (6V)	Reductora
10:1 micro HP	1600 mA	3000 rpm	0.3 Kg-cm	10:1
30:1 micro HP	1600 mA	1000 rpm	0.6 Kg-cm	30:1
50:1 micro HP	1600 mA	625 rpm	1.1 Kg-cm	50:1
10:1 micro	360 mA	1300 rpm	0.2 Kg-cm	10:1
50:1 micro	360 mA	250 rpm	0.5 Kg-cm	50:1

Fuente: (Pololu Corporation, 2016), Los Autores

3.1.5.1. Motor pololu 10:1.

Este micromotor CD es de muy alta potencia tiene una sección transversal de 10 x 12 mm, y el eje de salida del reductor en forma de D es de 9 mm de largo y 3 mm de diámetro. Posee una velocidad de 3000 rpm y con un consumo de 120 mA sin carga y un peso de 0.3 Kg-cm.



Figura 3. 9: Motor de engranaje Pololu.

Fuente: (Pololu Corporation, 2016)

Para este tipo de motor existe una fórmula de relación exacta de engranaje, en la ingeniería, una relación de transmisión es una medida directa de la relación de las velocidades de rotación de dos o más engranajes. De forma general, cuando se trata de dos engranajes si el tamaño del engranaje de accionamiento es mayor, esto tendrá como resultado una mayor velocidad de rotación en el engranaje accionado y viceversa.

Entonces podemos expresar este concepto de la siguiente manera:

La relación exacta de engranaje = $\frac{T2}{T1}$

En donde $T1$ es el número de dientes del primer engranaje y $T2$ es el número de dientes del segundo engranaje.

Para el motor utilizado en nuestro robot seguidor de línea velocista el valor exacto dado por la fórmula vista anteriormente es el siguiente:

Relación exacta de engranaje = $\frac{35 \times 37}{13 \times 10} \approx 9.96: 1$

La relación de este motor está diseñada específicamente para un aumento de velocidad y disminución de fuerza (torque).

3.1.5.2. Motor pololu 30:1.

Este micromotor CD es de muy alta potencia tiene una sección transversal de 10 x 12 mm, y el eje de salida del reductor en forma de D es de 9 mm de largo y 3 mm de diámetro. Posee una velocidad de 440 rpm y con un consumo de 40 mA sin carga y un peso de 0.6 Kg-cm.

Para este tipo de motor existe una fórmula de relación exacta de engranaje, para el motor utilizado en nuestro robot seguidor de línea destreza el valor exacto dado por la formula vista anteriormente es el siguiente:

$$\text{Relación exacta de engranaje} = \frac{31 \times 33 \times 35 \times 34}{16 \times 14 \times 13 \times 14} \approx 29.86: 1$$

La relación de este motor está diseñada específicamente para un aumento de fuerza (torque) y reducción de velocidad.

3.1.6. Regulador de voltaje.

Es un regulador de voltaje ajustable, de corriente media y específicamente diseñados para su utilización en aplicaciones de bajo voltaje de entrada. Este dispositivo ofrece una solución económica para la regulación de voltaje precisa mientras mantiene las pérdidas al mínimo.

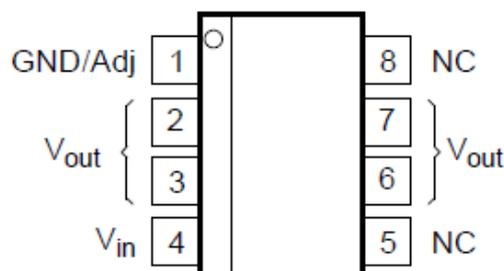


Figura 3. 10: Distribución de pines del regulador de voltaje.

Fuente: (Semiconductor Componets Industries, 2015)

Tabla 3. 7: Valores mínimos y máximos para las características técnicas y eléctricas del regulador de voltaje.

Característica	Valor	Unidad
Voltaje de entrada de fuente de poder	20	V
Temperatura de almacenamiento	-55 a +150	°C
Rango de temperatura ambiente operativa	-40 a +125	°C
Voltaje de Salida	3.465 a 3.535	V

Fuente: (Semiconductor Components Industries, 2015), Los Autores

3.1.7. Ruedas pololu.

Las ruedas plásticas Pololu están diseñadas para acoplarse de manera segura a gran cantidad de motores de engranaje incluyendo a los modelos mini metálicos y mini plásticos. Están disponibles en diferentes dimensiones e incluso múltiples colores para la comodidad del cliente.



Figura 3. 11: Rueda Pololu acoplada a un motor de engranaje.

Fuente: (Pololu Corporation, 2016)

3.1.8. Ruedas omnidireccionales (Ball casters)

Estas ruedas omnidireccionales Pololu son ideales para robots pequeños, ya que le dan la posibilidad de atravesar una mayor gama de obstáculos sin agregarle mucho peso. Están disponibles en múltiples medidas y materiales.



Figura 3. 12: Pololu ball Casters.

Fuente: (Pololu Corporation, 2016)

3.2. Controlador PID.

El controlador PID es un mecanismo de control por retroalimentación en bucle. La función de este controlador es minimizar el error entre la señal esperada y la señal real que nos está brindando el sistema.

En el caso del seguidor de línea, el controlador PID primero calcula la posición actual, luego calcula el error, les ordena a los motores realizar un giro grande si el error es alto o un giro pequeño si es bajo.

La magnitud del giro tomado será proporcional al error. Después de esto, si el error no disminuye o lo hace, pero lentamente, el controlador aumentará la magnitud del giro conforme pase el tiempo hasta que el robot se centre sobre la línea.

En el proceso de centrarse, el robot puede pasar su posición objetivo y posicionarse al otro lado de la línea, en este punto el proceso anterior es efectuado nuevamente. Entonces el robot se mantendrá oscilando a lo largo de la línea con el fin de centrarse en la misma.

Para efecto de comparación tenemos a continuación la forma en que un robot convencional seguiría la línea al no contar con un controlador PID.

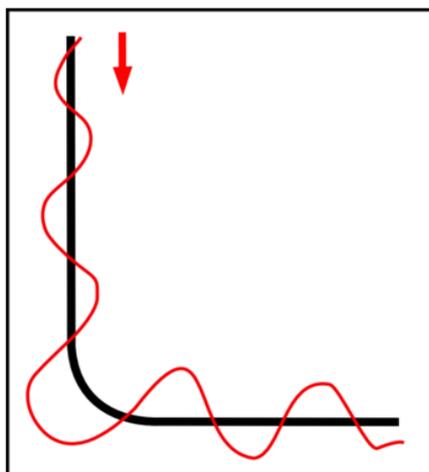


Figura 3. 13: Patrón de seguimiento de línea sin controlador PID.

Fuente: (Palmisano, 2016)

Como se observa en la figura, el robot oscila mucho fuera de la línea gastando tiempo y batería valiosa. En su defecto un robot con controlador PID seguirá la línea como se muestra a continuación.

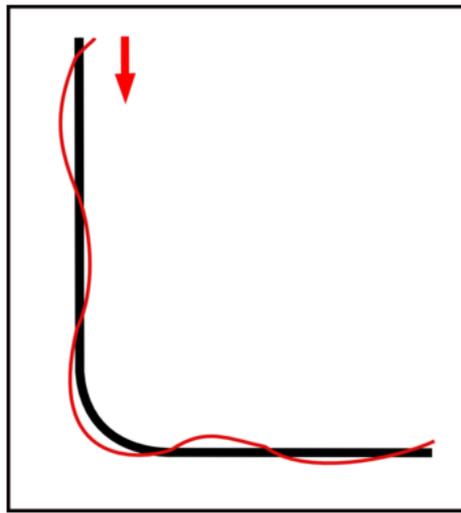


Figura 3. 14: Patrón de seguimiento con controlador PID

Fuente: (Palmisano, 2016)

El robot ahora sigue la línea de manera más fluida manteniendo su centro siempre sobre la línea, teniendo como resultado un seguimiento más rápido y mucho más eficiente.

3.3. Diseño electrónico de seguidor de línea velocista y destreza.

A continuación se presentan los diseños esquemáticos de cada elemento utilizado en la construcción de los robots seguidores de línea. La plataforma utilizada es Altium, un software de ordenador de origen australiano dedicado al diseño electrónico para ingenierías.

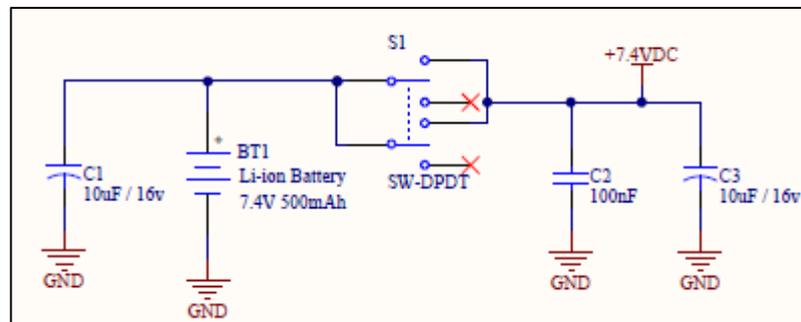


Figura 3. 15: Entrada de la batería.

Fuente: Los Autores.

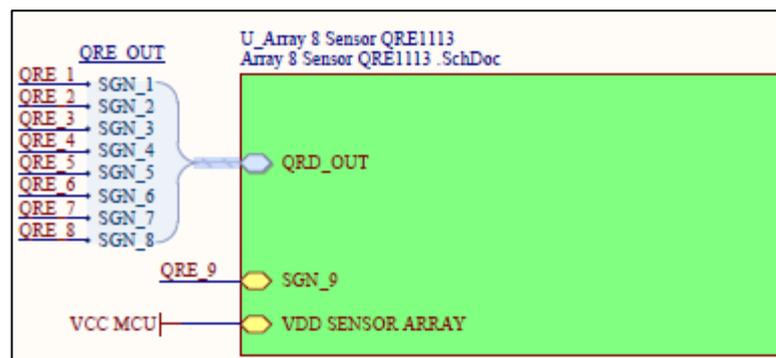


Figura 3. 16: Sensor QRD1113

Fuente: Los Autores.

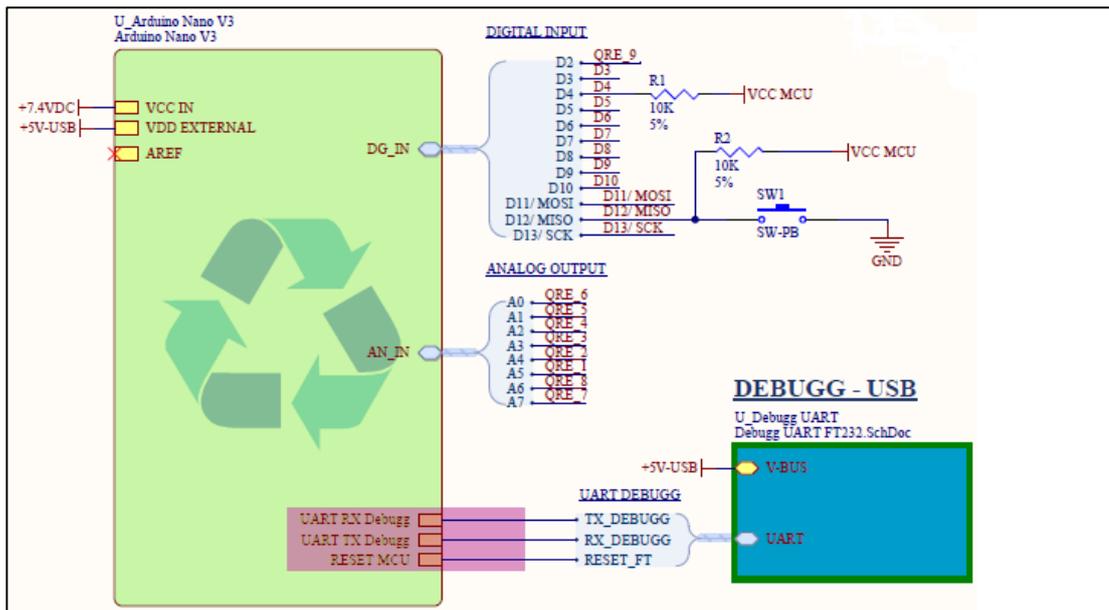


Figura 3. 17: Tarjeta Arduino Nano.

Fuente: Los Autores.

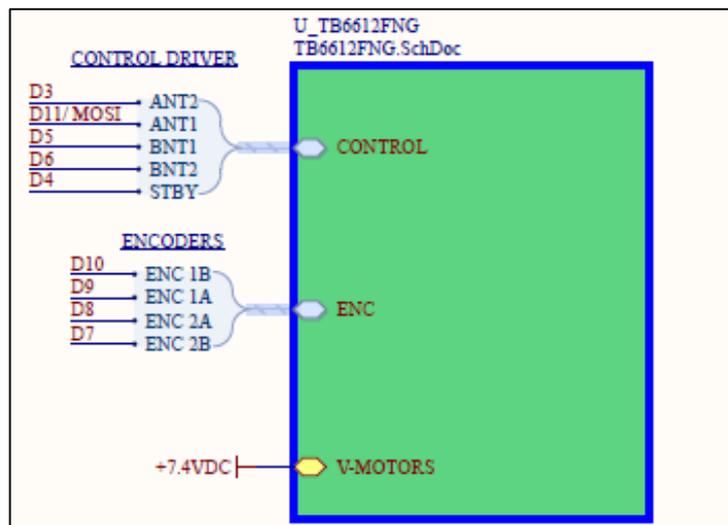


Figura 3. 18: Puente H.

Fuente: Los Autores

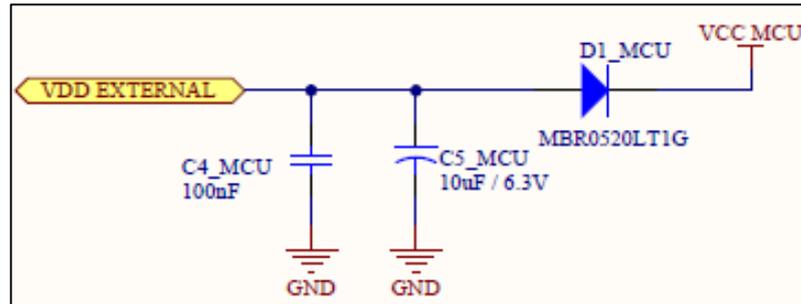


Figura 3. 21: Auto selector VDD.

Fuente: Los Autores.

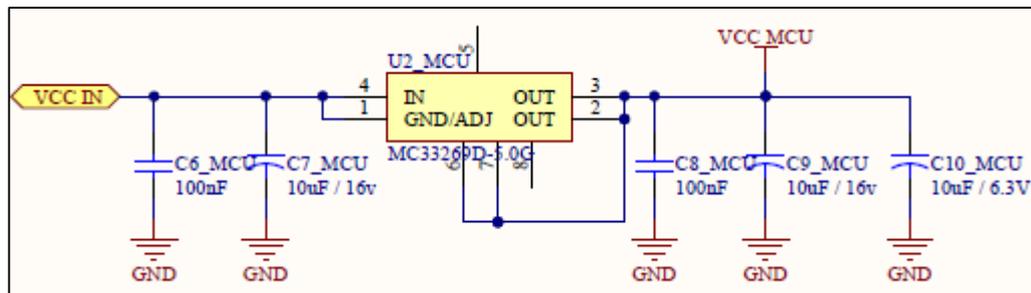


Figura 3. 22: Fuente de alimentación de +5VDC.

Fuente: Los Autores.

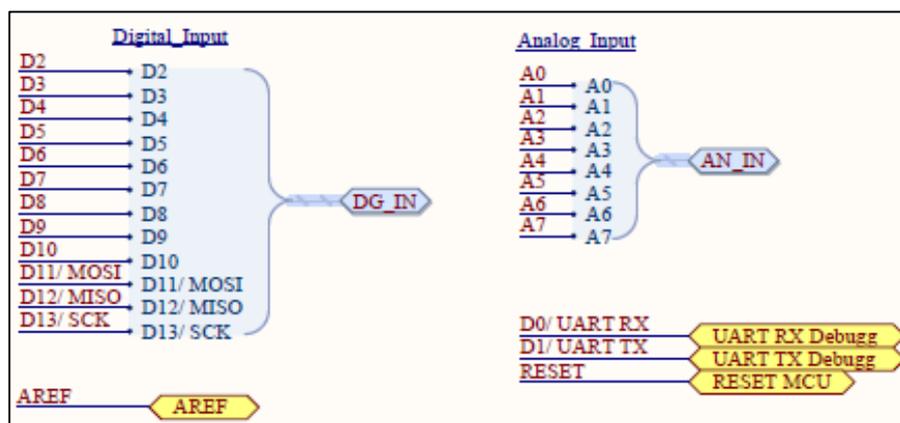


Figura 3. 23: Entradas digitales y analógicas.

Fuente: Los Autores.

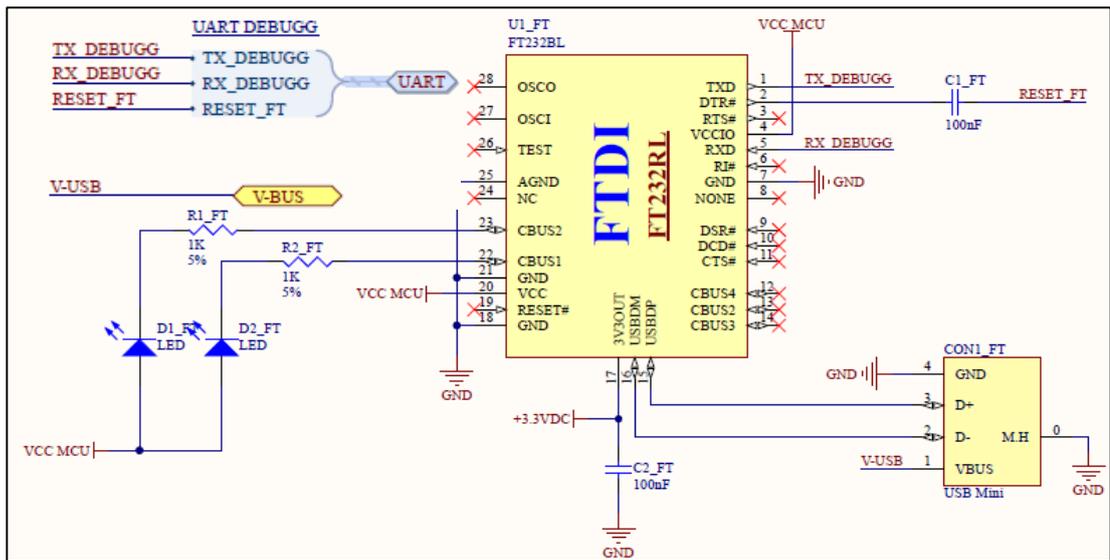


Figura 3. 24: Adaptador USB a Serial UART

Fuente: Los Autores.

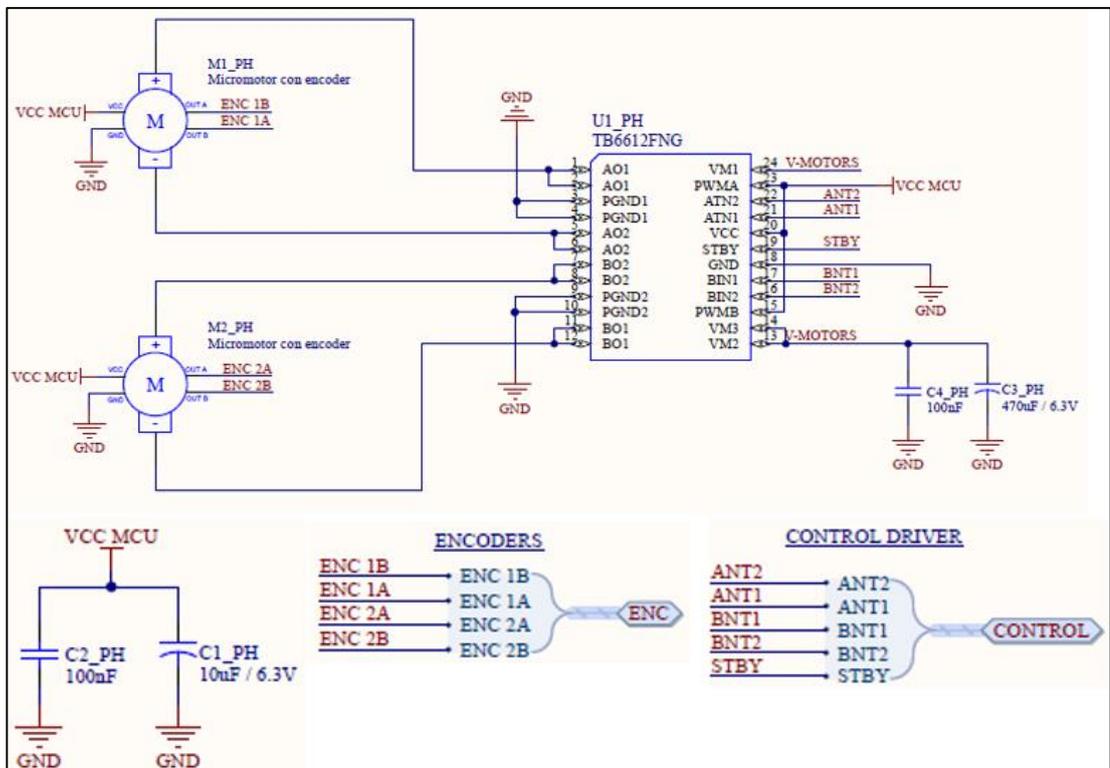


Figura 3. 25: Controlador de motores.

Fuente: Los Autores.

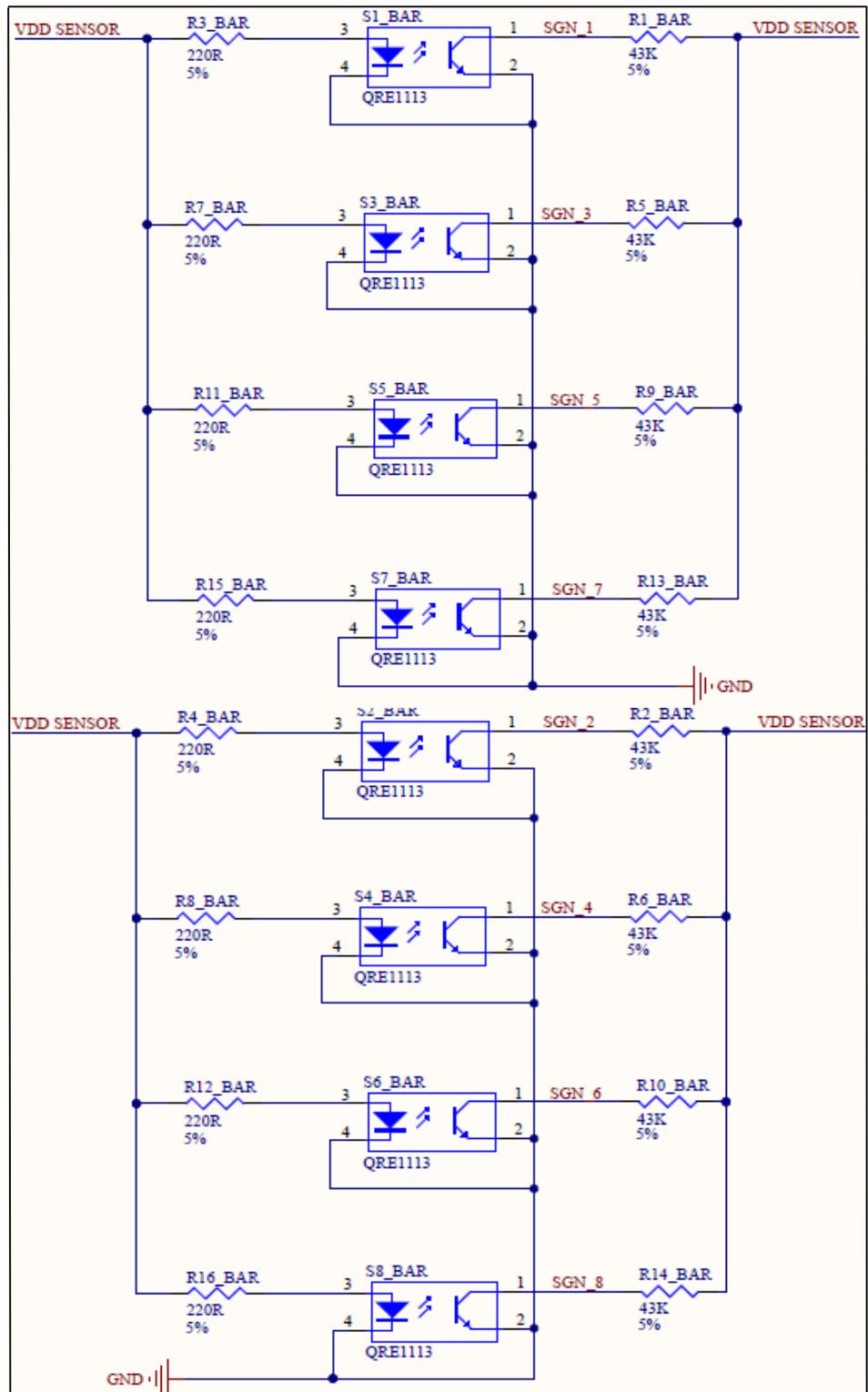


Figura 3. 26: Arreglo de sensores Izquierdo y Derecho.

Fuente: Los Autores.

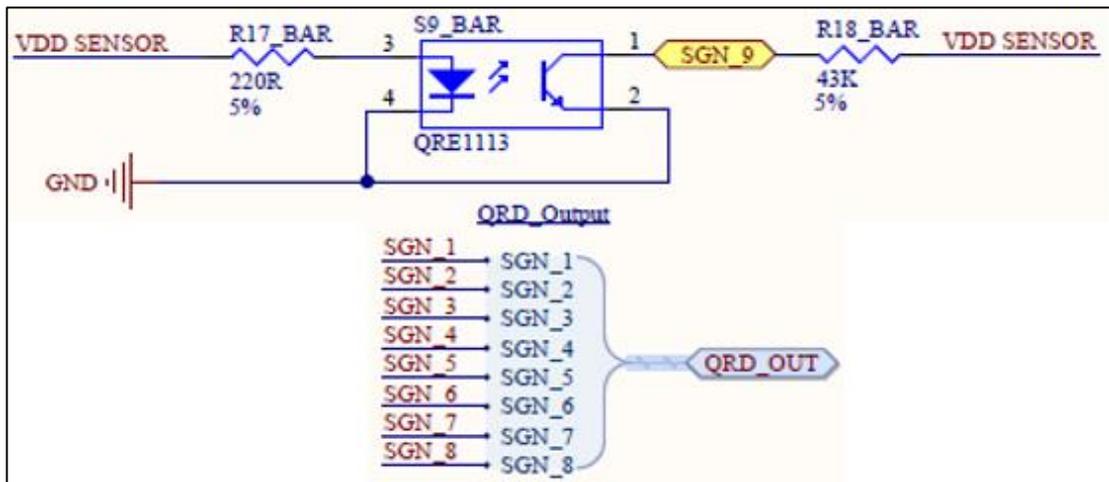


Figura 3. 27: Sensor adicional y asignación de señales.

Fuente: Los Autores.

3.3.1. Diseño PCB (Printed Circuit Board).

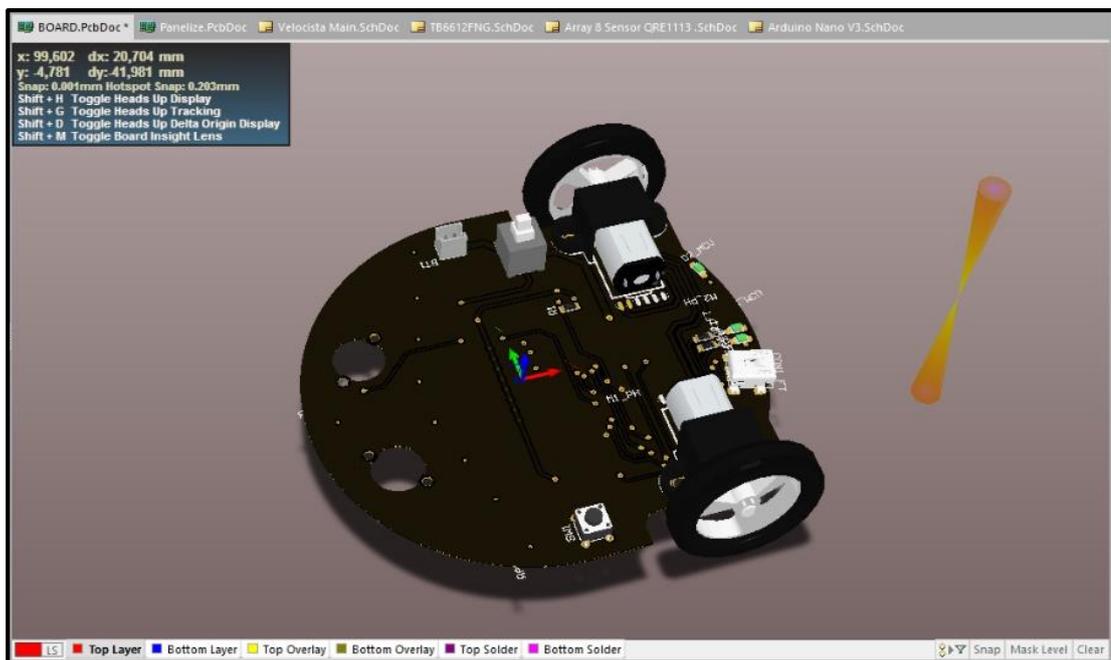


Figura 3. 28: Diseño PCB (vista superior)

Fuente: Los Autores.

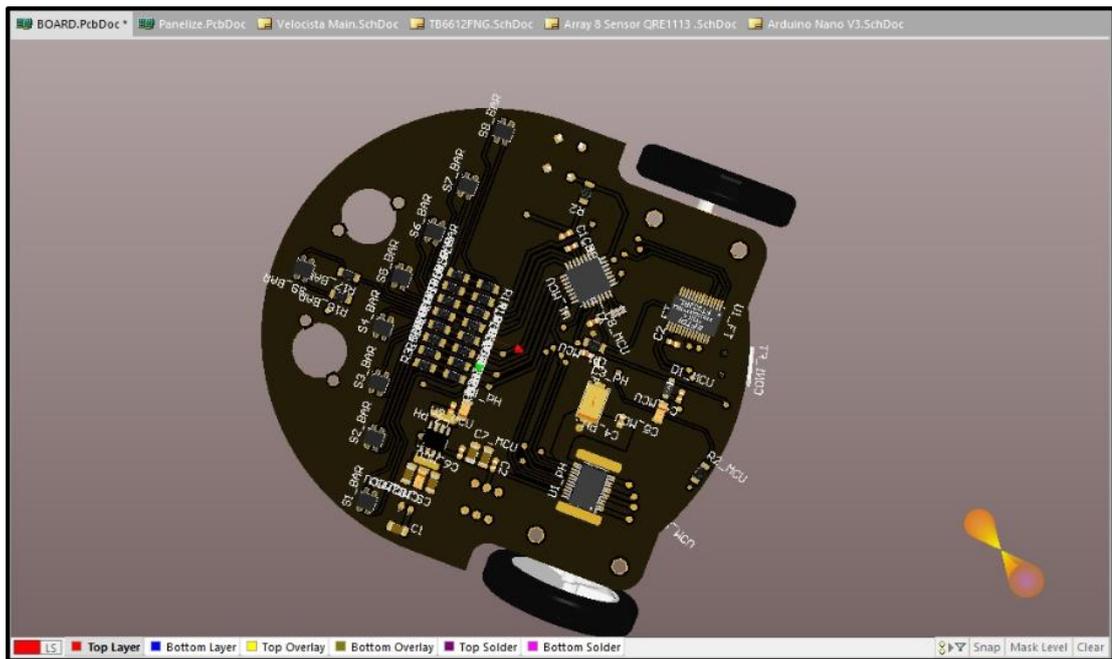


Figura 3. 29: Diseño PCB (vista inferior)

Fuente: los Autores.

3.4. Construcción de los robots seguidores de línea.

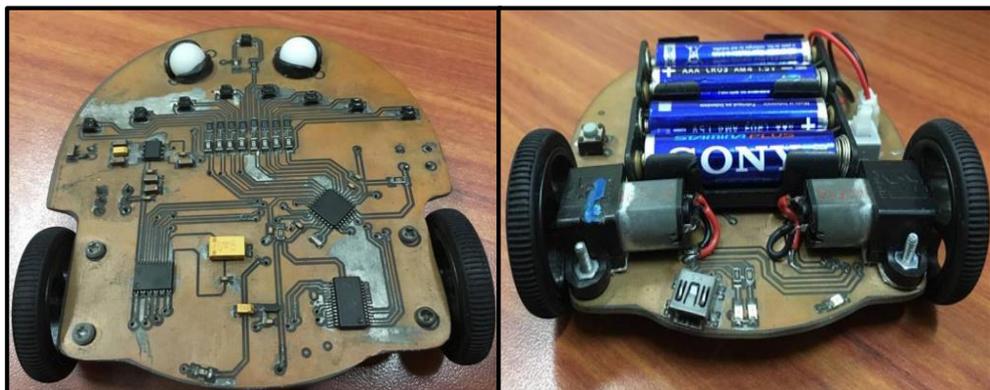


Figura 3. 30: Seguidor de línea (prototipo)

Fuente: Los Autores.

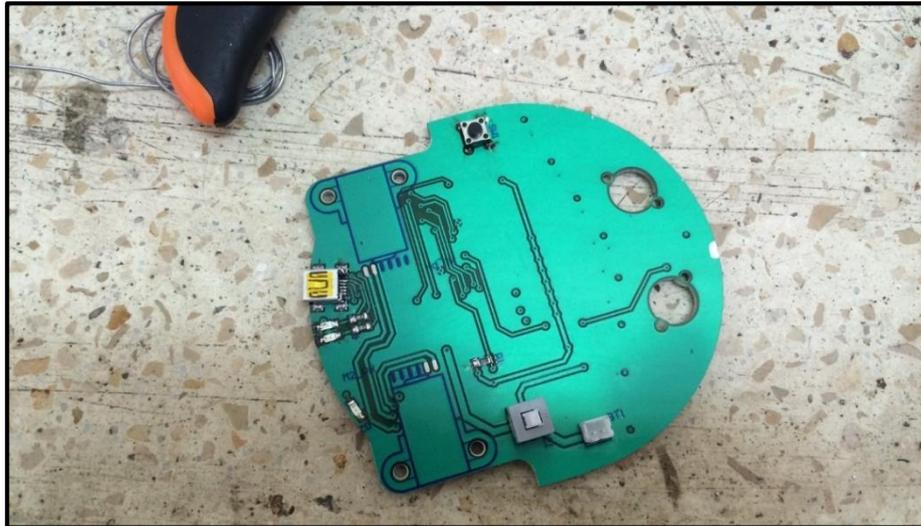


Figura 3. 31: Vista superior de la placa impresa
Fuente: Los Autores



Figura 3. 32: Vista inferior previa al montaje de motores
Fuente: Los Autores



Figura 3. 33: Seguidor de línea modalidad “Velocista”

Fuente: Los Autores.



Figura 3. 34: Seguidor de línea modalidad “Destreza”

Fuente: Los Autores.

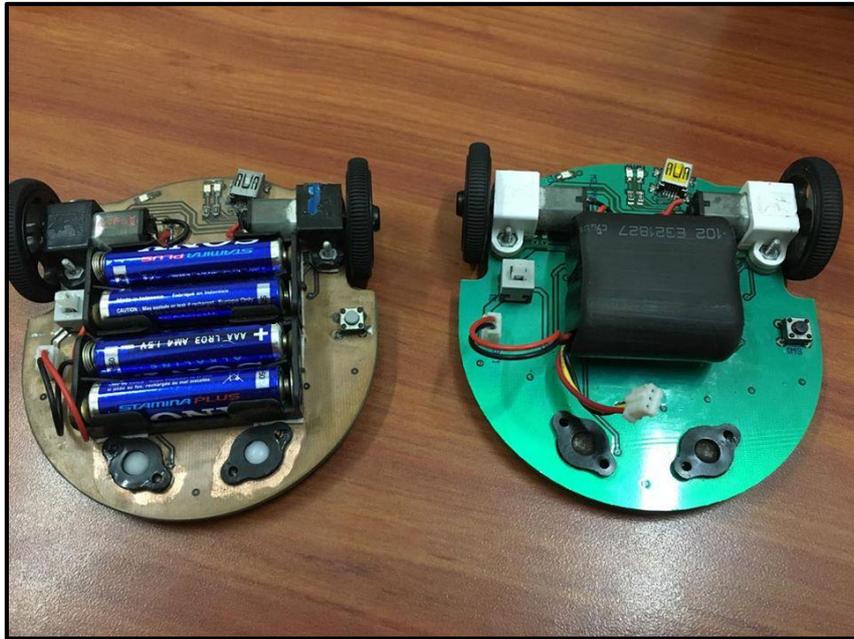


Figura 3. 35: Prototipo y robot finalizado

Fuente: Los autores.

Tabla 3. 8: Listado de elementos utilizados para la construcción del seguidor de línea.

Descripción	Designación	Cantidad
Batería LiPo	BT1	1
Capacitor 10uF/16V	C1, C2_MCU, C3, C7_MCU, C9_MCU	5
Capacitor 100nF	C1_FT, C1_MCU, C2, C2_FT, C2_PH, C3_MCU, C4_MCU, C4_PH, C6_MCU, C8_MCU	10
Capacitor Tan	C1_PH, C5_MCU, C10_MCU	3
Capacitor 470Uf/6.3V	C3_PH	1
USB mini	CON1_FT	1
LED	D1_FT, D2_FT, D2_MCU	3
Rectificador 1A	D1_MCU	1
Micromotor	M1_PH, M2_PH	2
Resistencia 43K	R1_BAR, R2_BAR, R5_BAR, R6_BAR, R9_BAR, R10_BAR, R13_BAR, R14_BAR, R18_BAR	9
Resistencia 1K	R1_FT, R2_FT	2
Adaptador USB UART	U1_FT	1
Sensor QRD1113	S1_BAR, S2_BAR, S3_BAR, S4_BAR, S5_BAR, S6_BAR, S7_BAR, S8_BAR, S9_BAR	9
MC ATmega328P	U1_MCU	1
Controlador de motor	U1_PH	1
Regulador de Voltaje	U2_MCU	1
Ruedas Pololu	-	2
Ball caster	-	2

Fuente: Los Autores

3.5. Presupuesto Referencial de los robots seguidores de línea.

Tabla 3. 9: Costo de materiales del robot prototipo.

Elementos Electrónicos	
Microcontrolador Atmega Sensores QTR TB6612 (puente H) Convertidor Serial Elementos Complementarios	\$110,00
Impresión en placa PCB	\$50,00
Pilas	\$4,00
Elementos Mecánicos	
Soportes Motores Ruedas Ball Casters	\$120,00
Subtotal	\$284,00
Gasto en importaciones	\$132,00
Total	\$416,00

Fuente: Los Autores.

Tabla 3. 10: Costo de materiales para la construcción final del robot velocista.

Elementos Electrónicos	
Microcontrolador Atmega Sensores QTR TB6612 (puente H) Convertidor Serial Elementos Complementarios	\$110,00
Impresión en placa PCB	\$80,00
Baterías	\$25,00
Elementos Mecánicos	
Soportes Motores Ruedas Ball Casters	\$120,00
Subtotal	\$335,00
Gasto en importaciones	\$132,00
Total	\$467,00

Fuente: Los Autores.

Tabla 3. 11: Costo de materiales para la construcción final del robot destreza.

Elementos Electrónicos	
Microcontrolador Atmega Sensores QTR TB6612 (puente H) Convertidor Serial Elementos Complementarios	\$114,00
Impresión en placa PCB	\$80,00
Baterías	\$25,00
Elementos Mecánicos	
Soportes Motores Ruedas Ball Casters	\$120,00
Subtotal	\$339,00
Gasto en importaciones	\$132,00
Total	\$471,00

Fuente: Los Autores.

Tabla 3. 12: Costo total de la implementación de los robots.

Robot	Costo referencial
Prototipo	\$416
Seguidor de línea modalidad velocista	\$467
Seguidor de línea modalidad destreza	\$471
Total	\$1354

Fuente: Los Autores.

3.6. Algoritmos de programación.

3.6.1 Programación seguidor de línea velocista.

Códigos para incluir librerías necesarias para la ejecución de sensores QTR y el Puente H.

```
#include <QTRsensors.h>
#include <OrangutanMotors.h>
```

Código donde se define e inicializan variables con los valores respectivos como el número de sensores que se utilizan, el número de muestras por cada sensor y por último el pin que permite la habilitación de la barra de sensores.

```
#define NUM_SENSORS          8
#define NUM_SAMPLES_PER_SENSOR 4
#define EMITTER_PIN         2
```

Código que permite el envío de datos a la librería QTRsensors

```
QTRsensorsAnalog qtra((unsigned char[]) {6, 7, 0, 1, 2, 3, 4, 5 },
NUM_SENSORS, NUM_SAMPLES_PER_SENSOR, EMITTER_PIN);
unsigned int sensorValues[NUM_SENSORS];
```

Código de estado de los dos motores con un total de cuatro estados, dos para cada uno.

```
OrangutanMotors motors;
int EnableMotors = 4;
```

Código donde se establece la velocidad máxima del robot.

```
int maximum = 140;
```

Código donde se define las constantes del control PID cuya función es el de permitir la estabilidad del robot al momento del funcionamiento.

```
double Kp = 2.1;  
double Kd = 17;  
double Ki = 0.001;
```

Código donde se establece la velocidad de calibración usada para la localización de la línea de la pista.

```
int velcalibrate = 33;  
int velgiro= 75;
```

Código donde se establece los estados de los pines tanto de entrada como de salida, también se definen condiciones y constantes para algunas de las variables.

```
void setup()  
{  
  int inPin = 12;  
  int val = 0;  
  pinMode(13, OUTPUT);  
  pinMode(EnableMotors, OUTPUT);  
  pinMode(inPin, INPUT);  
  digitalWrite(EnableMotors, HIGH);  
  
  val = digitalRead(inPin);  
  while (val == HIGH)  
  {  
    //digitalWrite(9, HIGH);  
    digitalWrite(13, HIGH);  
    val = digitalRead(inPin);  
  };  
  if (val == LOW)  
  {  
    //digitalWrite(9, LOW);  
    digitalWrite(13, LOW);  
  };  
  motors.setSpeeds(0,0);  
}
```

Código donde se establece las instrucciones y condiciones para la ejecución de la calibración de sensores.

```
delay(500);
digitalWrite(13, HIGH);

for (int counter=0; counter<21; counter++)
{
  if (counter < 6 || counter >= 15)
    OrangutanMotors::setSpeeds(-velcalibrate, velcalibrate);
  else
    OrangutanMotors::setSpeeds(velcalibrate, -velcalibrate);
    qtra.calibrate();

  delay(20); // retraso de 20 miliseg
}

digitalWrite(13, LOW);
OrangutanMotors::setSpeeds(0, 0);

delay(200);
digitalWrite(13, HIGH);
delay(200);
digitalWrite(13, LOW);
delay(200);
digitalWrite(13, HIGH);
delay(200);
digitalWrite(13, LOW);
delay(200);
digitalWrite(13, HIGH);
delay(200);
digitalWrite(13, LOW);
delay(200);
```

Código donde se establece el condicionamiento de las variables para terminar la ejecución de la calibración de sensores.

```
pinMode(inPin, INPUT);

val = digitalRead(inPin);
while (val == HIGH)
{
// digitalWrite(9, HIGH);
digitalWrite(13, HIGH);
val = digitalRead(inPin);
};
if (val == LOW)
{
//digitalWrite(9, LOW);
digitalWrite(13, LOW);
delay(500);
};
}

unsigned int last_proportional = 0;
long integral = 0;
```

Código donde se utilizan los valores de los sensores de la etapa de calibración y de esta manera ubicar la línea en el centro. La librería de los sensores proporciona los intervalos múltiplos de mil y una vez obtenido el rango adecuado se procede a asignar el centro. Después se efectúa control PID.

```
void loop()
{
  unsigned int position = qtra.readLine(sensorValues);
  if(position>2000 && position<4000){
    int proportional = (int)position - 3500;
    integral=0;
    integral += proportional;
    int derivative = proportional - last_proportional;
    if (integral>1000) integral=1000;
    if (integral<-1000) integral=-1000;
    int power_difference = Kp*proportional + Ki*integral + Kd*derivative;
    if (power_difference > maximum)
      power_difference = maximum;
    if (power_difference < -maximum)
      power_difference = -maximum;
    if (power_difference < 0)
      OrangutanMotors::setSpeeds(maximum, maximum + power_difference);
    else
      OrangutanMotors::setSpeeds(maximum - power_difference, maximum);
    last_proportional = proportional;
  }
}
```

En el caso que los valores se encuentren por debajo de la frontera inferior o por encima de la frontera superior, la velocidad de los motores será (0,0) y realizara un giro a la izquierda o a la derecha según corresponda.

```
else{
    OrangutanMotors::setSpeeds(0, 0);
    if(position>=4000)
        OrangutanMotors::setSpeeds(-velgiro, velgiro);
    if(position<=2000)
        OrangutanMotors::setSpeeds(velgiro, -velgiro);
    integral=0;
}
}
```

3.6.2. Programación seguidor de línea destreza.

El algoritmo de programación difiere con el mostrado anteriormente debido a que debemos hacer ciertas distinciones ya que para las pistas para la modalidad destreza no presentan curvas tan pronunciadas, la velocidad de giro varia y debemos considerar la presencia de líneas entrecortadas en diferentes patrones alternándose con continuas. Para esto la lectura de los sensores cambia debido a la incorporación de un nuevo sensor entre los arreglos de izquierda y derecha.

En lo que respecta a la configuración de los motores, su velocidad es inferior a la modalidad es inferior a la modalidad velocista para lograr superar con éxito las pista, sin descarrilamiento y con una reducida probabilidad de error.

Códigos para incluir librerías necesarias para la ejecución de sensores QTR y el Puente H.

```
#include <QTRsensors.h>
#include <OrangutanMotors.h>
```

Código donde se define e inicializan variables con los valores respectivos como el número de sensores que se utilizan, el número de muestras por cada sensor, el pin que permite la habilitación de la barra de sensores.

```
#define NUM_SENSORS          4
#define NUM_SAMPLES_PER_SENSOR 4
#define EMITTER_PIN         7
#define S7                   A4
#define S8                   A5
```

Código que permite el envío de datos a la librería QTRsensors.

```
QTRsensorsAnalog qtra((unsigned char[]) { 0, 1, 2, 3 },
NUM_SENSORS, NUM_SAMPLES_PER_SENSOR, EMITTER_PIN);
unsigned int sensorValues[NUM_SENSORS];
```

Código de estado de los dos motores con un total de cuatro estados, dos para cada uno y la velocidad máxima del robot.

```
OrangutanMotors motors;
int EnableMotors = 4;
int maximum = 140;
```

Código donde se define las constantes del control PID cuya función es el de permitir la estabilidad del robot al momento del funcionamiento.

```
double Kp = 1.2;
double Kd = 22.1;
double Ki = 0.003;
```

Código donde se establece la velocidad de calibración usada para la localización de la línea de la pista.

```
int velcalibrate = 25;
int velgiro= 65;
```

Código donde se establece los estados de los pines tanto de entrada como de salida, también se definen condiciones y constantes para algunas de las variables.

```
void setup()
{
  int inPin = 12;
  int val = 0;
  pinMode(13, OUTPUT);
  pinMode(EnableMotors, OUTPUT);
  pinMode(S7, INPUT);
  pinMode(S8, INPUT);
  pinMode(2, INPUT);
  digitalWrite(EnableMotors, HIGH);
  val = digitalRead(inPin);
  while (val == HIGH)
  {
    digitalWrite(13, HIGH);
    val = digitalRead(inPin);
  };
  if (val == LOW)
  {
    digitalWrite(13, LOW);
  };
  motors.setSpeeds(0,0);
}
```

Código donde se establece las instrucciones y condiciones para la ejecución de la calibración de sensores.

```
    delay(500);
    digitalWrite(13, HIGH);
    for (int counter=0; counter<23; counter++)
    {
        if (counter < 5 || counter >= 16){
            OrangutanMotors::setSpeeds(0, 0);
            delay(120);
            OrangutanMotors::setSpeeds(-velcalibrate, velcalibrate);
        }else{
            OrangutanMotors::setSpeeds(0, 0);
            delay(120);
            OrangutanMotors::setSpeeds(velcalibrate, -velcalibrate);
        }
        qtra.calibrate();
    }
    delay(65);
}
digitalWrite(13, LOW);
OrangutanMotors::setSpeeds(0, 0);
delay(200);
digitalWrite(13, HIGH);
delay(200);
digitalWrite(13, LOW);
delay(200);
digitalWrite(13, HIGH);
delay(200);
digitalWrite(13, LOW);
delay(200);
digitalWrite(13, HIGH);
delay(200);
digitalWrite(13, LOW);
delay(200);
```

Código donde se establece el condicionamiento de las variables para terminar la ejecución de la calibración de sensores.

```
pinMode(inPin, INPUT);
val = digitalRead(inPin);
while (val == HIGH)
{
// digitalWrite(9, HIGH);
digitalWrite(13, HIGH);
val = digitalRead(inPin);
};
if (val == LOW)
{
//digitalWrite(9, LOW);
digitalWrite(13, LOW);
delay(500);
};
}
unsigned int last_proportional = 0;
long integral = 0;
int proportional =0;
int derivative = 0;
boolean Forward = false;
unsigned int position;
boolean Entrecortada = true;
```

Código donde se efectúa la lectura de la barra de sensores y se determina la ausencia de línea entrecortada.

```
void loop(){
position = qtra.readLine(sensorValues);
if ((analogRead(A6) < 800) && (sensorValues[3] > 800) && (digitalRead(S7) == HIGH) ){
Entrecortada = false;
while (digitalRead(2) == LOW){
OrangutanMotors::setSpeeds(-velgiro, velgiro);
delay(50);
}
}
}
```

Código para efectuar el control PID.

```
if ( ((analogRead(A6) < 800) && (analogRead(A7) < 800) &&
(digitalRead(S7) == LOW) && (digitalRead(S8) == LOW) )
|| ((Forward == true) && ((position <= 1600) ||(position >= 1400)) ){
    proportional = (int)position - 1500;
    integral=0;
    integral += proportional;
    derivative = proportional - last_proportional;
    if (integral>1000) integral=1000;
    if (integral<-1000) integral=-1000;
    int power_difference = Kp*proportional + Ki*integral + Kd*derivative;
    if (power_difference > maximum)
        power_difference = maximum;
    if (power_difference < -maximum)
        power_difference = -maximum;
    if (power_difference < 0)
        OrangutanMotors::setSpeeds(maximum, maximum + power_difference);
    else
        OrangutanMotors::setSpeeds(maximum - power_difference,maximum);
    last_proportional = proportional;
}
```

Código donde se lleva a cabo la diferenciación entre una línea entrecortada y un fin de línea; dependiendo de esto se efectúa un giro a la derecha.

```
if ( digitalRead(2) == HIGH )
    Forward = true;
else
    Forward = false;
if ( (analogRead(A6) < 800) && (analogRead(A7) < 800) && (digitalRead(S7) == LOW) &&
(digitalRead(S8) == LOW) && (digitalRead(2) == LOW) && ((position == 0) ||(position == 3000)) ){
    while ((Entrecortada == false) && (digitalRead(2) == LOW) ){
        OrangutanMotors::setSpeeds(-velgiro,velgiro);
        delay(50);
    }
}
```

Código donde se verifica el estado e los sensores para determinar la llegada a la meta y realizar el freno de los motores.

```
if ((analogRead(A6) > 800) && (analogRead(A7) > 800) && (digitalRead(S7) == HIGH) &&
(digitalRead(S8) == HIGH) && (digitalRead(S8) == HIGH) &&
((position <= 1550) || (position >= 1450)) )
  while ((analogRead(A6) > 800) && (analogRead(A7) > 800) &&
(digitalRead(S7) == HIGH) && (digitalRead(S8) == HIGH)){
    OrangutanMotors::setSpeeds(0,0); // Freno
    digitalWrite(13, HIGH);
    delay(200);
    digitalWrite(13, LOW);
    delay(200);
  }
}
```

CAPÍTULO 4: CONCLUSIONES Y RECOMENDACIONES.

4.1. Conclusiones.

Con todo el trabajo investigativo realizado en la fundamentación teórica nos fue posible realizar una selección más consciente y favorable del microcontrolador a utilizar para el diseño de los robots; con base en las especificaciones técnicas se optó finalmente por el Microcontrolador ATmega 328/P de Atmel.

Un factor importante fue la incorporación de 8 sensores en el diseño e implementación del seguidor de línea velocista y 9 en el de destreza, con esto se obtuvo un mejor resultado en la etapa de calibración y en general un mejor desempeño del robot en la pista al momento de las pruebas.

En el transcurso del desarrollo del proyecto se indagó en el reglamento para participación en Concursos de robótica a nivel nacional y se determinaron características físicas que cumplan las normativas y se tomaron decisiones con este contexto en mente.

La totalidad del proyecto queda destinado para el Club de Robótica (ROBOFET) de la Universidad Católica de Santiago de Guayaquil con la expectativa que los estudiantes de la Facultad Técnica de Educación para el Desarrollo continúen realizando pruebas y efectúen mejoras para lograr muy buenos resultados en competencias nacionales e internacionales.

4.2. Recomendaciones.

Al instante de trabajar con microcontroladores se debe tener un especial cuidado en la asignación de valores en sus pines para garantizar un correcto funcionamiento según lo deseado. Se recomienda verificar la distribución de pines de las respectivas fichas técnicas para evitar cualquier error futuro.

Es recomendable lograr un acople adecuado entre diseño, hardware y software para evitar un mal funcionamiento en etapas avanzadas del proyecto. Se recomienda buscar asesoramiento temprano en cualquiera de los tres sectores para cubrir la falta de conocimiento y tener resultados satisfactorios del conjunto.

En el caso de realizar una mejora al diseño original, se recomienda analizar posibles variantes para el tipo de baterías utilizado dependiendo de las necesidades que se quieran cubrir con el resultado final. Existen varios tipos de ruedas con mayor desempeño y un mayor costo que ayudara al robot a alcanzar mejores logros en concursos y garantizaran una mayor durabilidad de las mismas.

La Facultad Técnica a través de sus estudiantes impulsará la investigación de la robótica e innovaciones tecnológicas para de esta manera incentivar a futuros alumnos que vayan a cursar las carreras afines a la electrónica a crear nuevos proyectos y mejorar los ya existentes.

REFERENCIAS BIBLIOGRÁFICAS

Angulo Usategui, J., & Angulo Martínez, I. (2005). Microcontroladores PIC:

Diseño práctico de aplicaciones. In J. Angulo Usategui, & I. Angulo Martínez, *Microcontroladores PIC: Diseño práctico de aplicaciones* (pp. 3-4; 19). Madrid; España: McGraw-Hill.

Asimov, I. (1984). *Yo, robot*. Verdaguier, 1 Capellades (Barcelona) - España: Edhasa.

Atmel Corporation. (2016, Enero 2). *Atmel*. Retrieved from www.atmel.com

Chamorro Hernandez, D., Acuña Ortiz, J., & Acero Patiño, D. (2009). *CONTROL DE PARQUEO INTELIGENTE*. Bogotá, Colombia.

Eskatul, M. (2011). *Design and Fabrication of a Line Follower Robot*. Sarrebruck: Lambert Academic Publishing.

Fairchild Semiconductor Corporation. (2015, Diciembre 27). *Fairchild*. Retrieved from www.fairchildsemi.com

Future Technology Devices Limited. (2015, Diciembre 28). *FTDI Chip*. Retrieved from www.ftdichip.com

Herman, S. (2013). *Electric Motor Control*. delmar: Cengage Learning.

Lucas, G. (Director). (1977). *Star Wars* [Motion Picture].

Office, S. E. (2015, Diciembre). sensors . Basilea, Suiza. Retrieved from <http://www.mdpi.com>

Pakdaman, M., & Sanaatiyan, M. (2009). Design and Implementation of Line Follower Robot. *International Conference on Computer and Electrical Engineering*, 585-590.

- Palmisano, J. S. (2016, Febrero 5). *Society of Robots*. Retrieved from <http://societyofrobots.com/>
- Plúa Martínez, A., & Castillo Valarezo, J. (2015). *IMPLEMENTACION DE UN ROBOT MEGA SUMO PARA PARTICIPACIONES EN CONCURSO DE ROBOTICA*. Guayaquil, Ecuador.
- Pololu Corporation. (2016, Enero 20). *Pololu*. Retrieved from <https://www.pololu.com/>
- Riduttori, B. (1995). *Gear Motor Handbook*. Bologna: Springer.
- Semiconductor Components Industries. (2015, Diciembre 28). *On Semiconductor*. Retrieved from www.onsemi.com
- Somolinos S., J. (2002). *Avances en robótica y visión por computador*. Real, España: Ediciones de la Universidad de Castilla - La Mancha.
- Toshiba Corporation. (2016, Enero 3). *Toshiba*. Retrieved from <http://us.toshiba.com/>
- V.T, J. (2015, Diciembre 29). *Tomson Electronics*. Retrieved from Tomson Electronics: www.tomsonelectronics.com
- Whale, J. (Director). (1931). *Frankenstein* [Motion Picture].
- Zabala, G. (2007). *Robótica*. Banfield- Lomas de Zamora: Gradi S.A. Retrieved from usershop.

ANEXO

Reglamentos de concursos para categorías seguidores de línea a nivel nacional.

Reglamento del Concurso UMEBOT (Escuela Politécnica Nacional).

REGLAS PARA EL ROBOT SEGUIDOR DE LINEA

1. EL RETO

El objetivo consiste en realizar el diseño y construcción de un robot móvil autónomo con la capacidad de seguir una trayectoria marcada por una línea de color negro en un fondo blanco

Esta trayectoria estará formada por planos, curvas con peralte, rampas de subida y de bajada, túneles, puentes, muros falsos e irregularidades en el suelo.

El ancho del camino (distancia entre orilla y orilla), será de 21 cm.

La línea de color negro, cuyo ancho será de aproximadamente de 2.0 cm y pasará por el centro de la pista.

Las rampas tanto de subida como de bajada tendrá un ángulo de inclinación respecto a la horizontal de 20°, el túnel contará con una altura superior 25 cm.

Por lo tanto el móvil ***no*** deberá tener una altura mayor a los 20 cm.

LOS EQUIPOS

Los equipos podrán estar integrados por uno o hasta tres estudiantes universitarios, de institutos técnicos, de colegios o aficionados (tomando en cuenta al asesor).

Todos los participantes deberán registrarse.

2. EL ROBOT

- Deberá ser totalmente autónomo (ningún tipo de control remoto).
- Queda a criterio del constructor tanto las dimensiones, la estructura, como el conjunto cerebro-sensores del robot.

3. EL CONCURSO

El concurso se dividirá en dos etapas:

a. **RONDAS ELIMINATORIAS**

- Exposición oral por parte de los equipos e inspección de los robots por parte de los jueces.
- Carrera de 2.5 m en trayectoria.

b. **RONDAS FINALES**

Desempeño del móvil a través de la trayectoria, mencionada en “EL RETO” máximo de tiempo 20 min.

Reglamento del concurso interno de la Universidad Politécnica salesiana (UPS).

OBJETIVO:

La competencia Seguidores de Líneas, consiste en la creación de un robot autónomo, capaz de seguir una línea negra en un fondo blanco en el menor tiempo posible. Estos robots se caracterizan por el alto grado de algoritmos capaces de poder predecir y pasar las pruebas más inesperadas, como curvas, ángulos rectos, pistas discontinuadas, túneles, pequeñas pendientes, etc.

REGLAS GENERALES:

No está permitida la utilización de robots comerciales. Si algún equipo desea utilizar algún tipo de estos productos considerando que aportará mejoras muy significativas, debe consultarlo con la organización del concurso.

El robot deberá ser completamente autónomo y auto contenido, es decir que no deberá ser dirigido por ninguna especie de control remoto ni podrá tener ningún tipo de contacto con su exterior como ordenadores o fuentes de alimentación al momento de la competición.

Se comprobará antes del concurso que un robot cumple las especificaciones para competir en cada prueba.

Quedan eliminados los equipos que no se presentan en el momento que se hace la llamada a la competencia bajo las siguientes consideraciones. En caso de no presentarse inmediatamente, se continuara con los siguientes participantes, dándole un lapso de 10 minutos desde su primera llamada para que el participante ausente se presente y realice su presentación. Después de este tiempo, será eliminado sin oportunidad de reclamo alguno.

Los organizadores, auspiciantes y otros vinculados al desarrollo de este evento NO se responsabilizarán en ninguna medida por daños, pérdidas u otros perjuicios a los materiales, herramientas y Robot, que en forma accidental se pueda provocar en el evento.

Los daños que pudieran sufrir los robots participantes propios de la competencia, se entenderán como propios de la naturaleza del evento, u no suponen recuperación por parte del comité organizador ni de los ganadores del evento, mucho menos de la universidad.

RECOMENDACIONES GENERALES:

- Diseñar los sensores del robot, de manera que puedan ser fácilmente ajustables durante el desarrollo de la competición, ya que las condiciones externas de iluminación pueden cambiar, así como otros factores externos que puedan influir sobre los sensores. La organización no se hace responsable de los mismos.
- En el diseño del robot, buscar siempre la máxima fiabilidad, dotando al robot de la mayor robustez posible, ya que durante la competición el tiempo para reparaciones de último minuto será mínimo.
- Cada equipo debe encargarse de traer a la competición las herramientas necesarias para utilizar y/o reparar el robot en caso de avería.

EL ÁREA DE COMPETENCIA

El área de competencia se entiende como las pistas asignadas por el comité organizador las cuales serán usadas por el robot seguidor de línea en las distintas etapas de la competencia. Las Pistas consistirán de una superficie de fondo color blanco con una línea negra, la cual tendrá las siguientes características:

- **Tamaño de línea:** Ancho de una cinta aislante estándar.
- **Bifurcaciones:** Las pistas tendrán en su recorrido líneas rectas, líneas cruzadas, líneas entrecortadas de máxima de 5 cm, desvíos, arcos circulares y otras curvas, los cuales el robot tendrá que decidir el camino correcto.
- **Ángulos:** Las pistas podrán tener ángulos cerrados no menor de 30°.
- **Iluminación:** El sitio de competencia estará iluminado de forma artificial o natural y las condiciones de luz pueden ser cambiantes, por lo cual no se garantiza que no existirán sombras o fuerte iluminación solar. Durante la competencia existirán equipos de filmación y cámaras fotográficas que podrán generar haces infrarrojos por sus sistemas de autofocus, por lo tanto los robots deben estar preparados para
 - que estas radiaciones y condiciones de luz cambiante no los afecten.
 - El orden de participación en este evento clasificatorio se define por sorteo, justo antes del evento.

NORMATIVA EN PISTA

- La Salida y la Meta de las Pistas estarán claramente identificadas, y señalizadas.
- Una vez dadas las indicaciones de los jueces, el responsable de equipo se acercará al área de competencia para situar su robot. El robot se situará inmediatamente detrás de la Salida.
- Una vez situado el robot, el responsable del equipo se preparará para activarlo cuando el juez de pista así lo indique. Una vez activado el robot se iniciará el recorrido de la pista y el cronometraje del tiempo.
- El Robot debe seguir en todo momento la trayectoria establecida en cada pista.
- En caso de perder la línea, el Robot dispondrá de 15 segundos para retornar al punto desde donde perdió la línea con un margen de +/- 20cm. no siendo posible la intervención externa en ningún caso. Si pasado este tiempo máximo el Robot no ha retornado al camino en las condiciones anteriormente citadas, se dará por terminada la presentación.
- Cuando los jueces den por finalizado el tiempo de competencia, los responsables de equipo procederán a retirar los robots de la pista.
- Cada robot cuenta con 5 minutos como máximo para realizar todo el recorrido.
- No existe límite en cuanto al número de robots o equipos concursantes.

PUNTAJE DE CALIFICACIÓN

Los robots serán evaluados por el jurado según el tiempo recorrido y distancia

VIOLACIONES

Será considerada una violación por parte de un equipo los siguientes supuestos:

- a) Una parada de la competencia que no se considere justificada por parte de los jueces.
- b) Activación del robot antes de que el juez de pista lo indique.
- c) Hacer alguna cosa que remita contra la integridad de la organización así como a la de sus participantes.

ACCIDENTES DURANTE LA COMPETENCIA (Penalizaciones)

Será considerado como penalización y, por lo tanto, darán por terminada la presentación por parte del robot causante de la penalización los siguientes supuestos:

- a) Provocar desperfectos al área de juego.
- b) Tocar el robot antes de que los participantes/jurados den por terminada la prueba.
- c) El robot es incapaz de terminar el recorrido.
- d) El Robot se salte el recorrido, ya que siempre debe seguir el camino a lo largo de la línea que define su trazado sin posibilidad de poder evitar o saltar una parte del mismo.
- e) El robot se sale del área de pista.
- f) Se rompen o separan partes del robot que impidan su tarea.
- g) El Robot se detiene más de 1 minuto.
- h) El Robot entra en un loop por más de 1 minuto.
- i) Insultar o agredir a miembros de la organización, así como al resto de competidores.
- j) Manipular el robot de forma externa por cualquier medio una vez que ha empezado la participación.

Los jueces y la organización se reservan el derecho de expulsión de la competición de un equipo si así se cree oportuno, comunicando los motivos de la expulsión a las partes afectadas y su decisión será irrevocable.

Reglamento del Concurso Ecuatoriano de Robótica (CER).

Objetivo

La competencia Seguidores de Líneas, consiste en la creación de un robot autónomo, capaz de seguir una línea negra en un fondo blanco en el menor tiempo posible. Estos robots se caracterizan por el alto grado de algoritmos capaces de poder predecir y pasar las pruebas más inesperadas, como curvas, ángulos rectos, pistas discontinuadas, túneles, pequeñas pendientes, etc.

Reglas Generales:

No está permitido robot comerciales en la categoría solo se podrá permitir Hardware de desarrollo

El robot deberá ser completamente autónomo y auto contenido, es decir que no deberá ser dirigido por ninguna especie de control remoto ni podrá tener ningún tipo de contacto con su exterior como ordenadores o fuentes de alimentación al momento de la competición.

Se comprobará antes del concurso que un robot cumple las especificaciones para competir en cada prueba.

Quedan eliminados los equipos que no se presentan en el momento que se hace la llamada a la competencia bajo las siguientes consideraciones. En caso de no presentarse inmediatamente, se continuara con los siguientes participantes, dándole un lapso de 10 minutos desde su primera llamada para que el participante ausente se presente y realice su presentación. Después de este tiempo, será eliminado sin oportunidad de reclamo alguno.

Los daños que pudieran sufrir los robots participantes propios de la competencia, se entenderán como propios de la naturaleza del evento, u no suponen recuperación por parte del comité organizador ni de los ganadores del evento, mucho menos de la universidad.

1. EL CONCURSO Y JUECES:

La categoría de "SEGUIDORES DE LINEA" consiste en la creación de un Robot que cumple una única misión: seguir una línea negra en un fondo blanco en el menor tiempo posible.

EL ÁREA DE COMPETENCIA

El área de competencia se entiende como las pistas asignadas por el comité organizador las cuales serán usadas por el robot seguidor de línea en las distintas etapas de la competencia. Las Pistas consistirán de una superficie de fondo color blanco con una línea negra, la cual tendrá las siguientes características:

Tamaño de línea:

Ancho de una cinta aislante estándar.

Bifurcaciones:

Las pistas tendrán en su recorrido líneas rectas, líneas cruzadas, líneas entrecortadas de máxima de 5 cm, desvíos, arcos circulares y otras curvas, los cuales el robot tendrá que decidir el camino correcto.

Ángulos:

Las pistas podrán tener ángulos cerrados no menor de 30°.

Iluminación:

El sitio de competencia estará iluminado de forma artificial o natural y las condiciones de luz pueden ser cambiantes, por lo cual no se garantiza que no existirán sombras o fuerte iluminación solar. Durante la competencia existirán equipos de filmación y cámaras fotográficas que podrán generar haces infrarrojos por sus sistemas de autofocus, por lo tanto los robots deben estar preparados para que estas radiaciones y condiciones de luz cambiante no los afecten.

El orden de participación en este evento clasificatorio se define por sorteo, justo antes del evento.

NORMATIVA EN PISTA

- * La Salida y la Meta de las Pistas estarán claramente identificadas, y señalizadas.
- * Una vez dadas las indicaciones de los jueces, el responsable de equipo se acercará al área de competencia para situar su robot. El robot se situará inmediatamente detrás de la Salida.
- * Una vez situado el robot, el responsable del equipo se preparará para activarlo cuando el juez de pista así lo indique. Una vez activado el robot se iniciará el recorrido de la pista y el cronometraje del tiempo.
- * El Robot debe seguir en todo momento la trayectoria establecida en cada pista.
- * Cuando los jueces den por finalizado el tiempo de competencia, los responsables de equipo procederán a retirar los robots de la pista.
- * Cada robot cuenta con 5 minutos como máximo para realizar todo el recorrido.

PUNTAJE DE CALIFICACIÓN

Los robots serán evaluados por el jurado según el tiempo recorrido y distancia

VIOLACIONES

Será considerada una violación por parte de un equipo los siguientes supuestos: y se castigará con 5 segundos más al mejor tiempo que realice

- a) Una parada de la competencia que no se considere justificada por parte de los jueces.
- b) Activación del robot antes de que el juez de pista lo indique.
- c) Hacer alguna cosa que remita contra la integridad de la organización así como a la de sus participantes.



DECLARACIÓN Y AUTORIZACIÓN

Yo, Jorge Andrés Vera Arenas, con C.C: 093005005-9 autor/a del trabajo de titulación: "Diseño e Implementación de dos robots seguidores de línea modalidad velocista y destreza para participaciones en concursos de robótica", 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, 14 de marzo de 2016

f.

Nombre: Jorge Andrés Vera Arenas
C.C: 093005005-9



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, Edward Andrés Alejandro Proaño, con C.C: 091873721-4 autor/a del trabajo de titulación: "Diseño e Implementación de dos robots seguidores de línea modalidad velocista y destreza para participaciones en concursos de robótica", 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, 14 de Marzo de 2016

f. 
Nombre: Edward Andrés Alejandro Proaño
C.C: 091873721-4



REPOSITORIO NACIONAL EN CIENCIA Y TECNOLOGÍA

FICHA DE REGISTRO DE TESIS/TRABAJO DE TITULACIÓN

TÍTULO Y SUBTÍTULO:	Diseño e Implementación de dos robots seguidores de línea modalidad velocista y destreza para participaciones en concursos de robótica.		
AUTOR(ES) (apellidos/nombres):	Vera Arenas Jorge Andrés Alejandro Proaño Edward Andrés		
REVISOR(ES)/TUTOR(ES) (apellidos/nombres):	Bohórquez Escobar Celso Bayardo		
INSTITUCIÓN:	Universidad Católica de Santiago de Guayaquil		
FACULTAD:	Facultad de Educación Técnica para el Desarrollo		
CARRERA:	Ingeniería en Telecomunicaciones		
TITULO OBTENIDO:	Ingeniero en Telecomunicaciones		
FECHA DE PUBLICACIÓN:	14 de marzo de 2016	No. DE PÁGINAS:	121
ÁREAS TEMÁTICAS:	Sistemas de Información, Desarrollo de Sistemas		
PALABRAS CLAVES/ KEYWORDS:	MICROCONTROLADOR, VELOCISTA, SENSORES, ROBÓTICA DESTREZA, SEGUIDOR		
RESUMEN/ABSTRACT (150-250 palabras):	<p>El presente proyecto de titulación busca profundizar el conocimiento en la robótica móvil y contemplar el ámbito competitivo durante su desarrollo. Explorar las secciones de diseño, hardware y software para la elaboración de dos robots seguidores de línea modalidad velocista y destreza. Describimos la influencia de la electrónica desde décadas atrás, las leyes fundamentales que rigen a un robot y su papel en la cultura popular. En la fundamentación teórica tenemos lo concerniente a la electrónica detrás de la robótica, los microcontroladores, sus tipos, familias y arquitectura. Este proyecto detalla cada uno de los elementos utilizados y su funcionamiento para la construcción de los robots seguidores de línea, el diseño electrónico y los algoritmos de programación detallados de ambas modalidades. La finalidad es analizar el proceso completo en la construcción de un robot y así motivar a los estudiantes a involucrarse en cualquier sección del mismo, mejorar diseños, impulsar la creatividad y lograr importantes resultados en el ámbito competitivo.</p>		
ADJUNTO PDF:	<input checked="" type="checkbox"/> SI	<input type="checkbox"/> NO	
CONTACTO CON AUTOR/ES:	Teléfono: 0978642217; 0997590036	E-mail: java2693@hotmail.com ; aleiohalo2@hotmail.com	
CONTACTO CON LA INSTITUCIÓN: COORDINADOR DEL PROCESO DE UTE	Nombre: Palacios Meléndez Edwin Fernando		
	Teléfono: 0968366762		
	E-mail: edwin.palacios@cu.ucsg.edu.ec		

SECCIÓN PARA USO DE BIBLIOTECA

Nº. DE REGISTRO (en base a datos):	
Nº. DE CLASIFICACIÓN:	
DIRECCIÓN URL (tesis en la web):	