



**UNIVERSIDAD ESTATAL  
PENÍNSULA DE SANTA ELENA**

**FACULTAD DE SISTEMAS Y  
TELECOMUNICACIONES**

**CARRERA DE ELECTRÓNICA Y TELECOMUNICACIONES**

**TEMA**

**“IMPLEMENTACIÓN DE ROBOT DE VIGILANCIA Y  
SEGURIDAD CON VISION NOCTURNA DESMONTABLE  
APLICANDO TECNOLOGÍA FPGA COMO HERRAMIENTA  
DE LABORATORIO PARA ADQUISICIÓN DE DATOS.”**

**TÉSIS DE GRADO:**

**Previo a la obtención del Título de**

**INGENIERA EN ELECTRÓNICA Y  
TELECOMUNICACIONES**

**AUTORA:  
FRANCO ARCE MARÍA GABRIELA**

**TUTOR:  
ING. SENDEY VERA GONZÁLEZ**

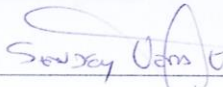
**LIBERTAD – ECUADOR  
2016**

## APROBACIÓN DEL TUTOR

En mi calidad de Tutor del trabajo de titulación denominado: **“IMPLEMENTACIÓN DE ROBOT DE VIGILANCIA Y SEGURIDAD CON VISIÓN NOCTURNA DESMONTABLE APLICANDO TECNOLOGÍA FPGA COMO HERRAMIENTA DE LABORATORIO PARA ADQUISICIÓN DE DATOS”**, elaborado por la egresada Franco Arce María Gabriela, de la Carrera de Electrónica y Telecomunicaciones de la Universidad Estatal Península de Santa Elena, me permito declarar que luego de haber orientado, estudiado y revisado, la apruebo en todas sus partes y autorizo al estudiante para que inicie los trámites legales correspondientes.

La Libertad, 7 Enero del 2016

Atentamente



Ing. Sendey Vera González.

**TUTOR**

## DEDICATORIA

El presente trabajo lo dedico con todo el amor del mundo a mi padre Enrique Franco Rodríguez por su apoyo, consejos, comprensión, afecto, y por ayudarme con los recursos necesarios para estudiar a lo largo de mi carrera.

A mi esposo Joffre Reyes De la Rosa quien me brindo su amor, cariño, estímulo y apoyo constante cuanto más lo necesitaba y a mi hijita bella Brithany Reyes Franco para Uds. quienes son la luz de mi vida y quienes amo con toda mi alma, les dedico mi esfuerzo y mis ganas por seguir siempre adelante.

A mis hermanas y hermanos, que han sido un apoyo emocional por estar conmigo y darme animo de continuar cada día con mi proyecto, los quiero mucho.

Todo este trabajo ha sido posible gracias a ellos.

María Gabriela Franco Arce

## AGRADECIMIENTO

Agradezco primeramente a Dios, por las oportunidades que me ha dado en esta vida por guiarme y bendecirme durante la trayectoria de mi carrera profesional dándome fuerzas para superar obstáculos y dificultades.

A mi padre, que con su demostración ejemplar ha sabido guiarme y me ha enseñado a no rendirme ante nada y siempre perseverar con sus sabios consejos.

A mis suegros y cuñados, por su apoyo incondicional que me han manifestado.


Al tutor Ing. Sendey Vera por su tiempo y asesoramiento a la realización de este proyecto.

A mis amigos y conocidos, que con su amistad y ayuda constante han sido el apoyo que necesite para continuar con constancia para la culminación de este trabajo.

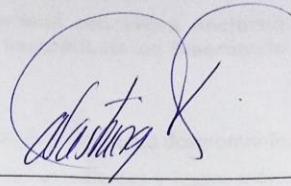
Gracias por todo el apoyo, que sin ustedes no me encontraría en el lugar en que me encuentro ahora.

María Gabriela Franco Arce

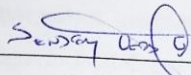
**TRIBUNAL DE GRADO**



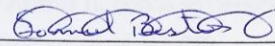
Ing. Walter Orozco Iguasnia, MSc.  
**DECANO DE LA FACULTAD**



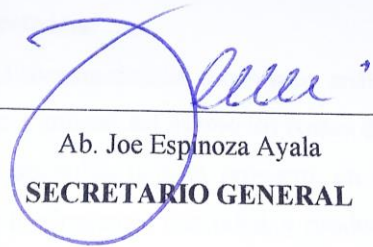
Ing. Washington Torres Guin, MSc  
**DIRECTOR DE CARRERA**



Ing. Sendey Vera González  
**PROFESOR TUTOR**



Ing. Samuel Bustos Gaibor  
**PROFESOR DE ÁREA**



Ab. Joe Espinoza Ayala  
**SECRETARIO GENERAL**

**UNIVERSIDAD ESTATAL PENÍNSULA DE SANTA ELENA  
FACULTAD DE SISTEMAS Y TELECOMUNICACIONES  
CARRERA DE ELECTRÓNICA Y TELECOMUNICACIONES**

**“Implementación de robot de vigilancia y seguridad con vision nocturna desmontable aplicando tecnología FPGA como herramienta de laboratorio para adquisición de datos.”**

**RESUMEN**

El presente trabajo está basado en un robot de vigilancia y seguridad desmontable, diseñado con la finalidad de optimizar el trabajo de los estudiantes y lograr mejor eficiencia en las prácticas de laboratorio de la escuela de Electrónica y Telecomunicaciones.

Se emplearon técnicas de lectura investigativa, métodos lógicos y sistemáticos para crear el prototipo, mediante determinación de componentes. Para su codificación se utilizó software libre arduino que ejecuta las funciones de todos los movimientos del robot móvil, también se manejó el programa de Quartus II, donde se desarrolló el código de VHDL para mostrar adquisición de datos en el display LCD.

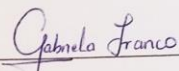
Para el proceso del robot se determinan rutas de trayectoria programadas en superficie plana monitoreando el área determinada, mediante su ejecución tiene dos modos de operación para su selección sea este como modo manual o modo automático obedeciendo órdenes de navegación de forma remota controlando el prototipo desde la computadora.

Se realizó los análisis del sistema técnico, económico, análisis operativo indicando el software y el hardware a utilizar, así mismo los costos de la implementación del robot móvil que fueron accesible para el proyecto, en el análisis operativo se utilizaron técnicas como rendimiento, fiabilidad, y productividad. La elaboración de pruebas y resultados de los sistemas de control de la funcionalidad del prototipo.

Con este trabajo se incentiva a los estudiantes en proyectos de investigación y desarrollo de móviles mediante la elaboración de prototipos de bajo costo, para la realización de prácticas que se dan en laboratorio donde se ponen a prueba las capacidades intelectuales y habilidades de cada uno de ellos.

## DECLARACIÓN

El contenido del presente trabajo de graduación es de mi responsabilidad; el patrimonio intelectual del mismo pertenece a la universidad Estatal Península de Santa Elena.



---

María Gabriela Franco Arce

## TABLA DE CONTENIDOS

ITEM	PÁGINA
APROBACIÓN DE TUTOR	I
DEDICATORIA	II
AGRADECIMIENTO	III
TRIBUNAL DE GRADO	IV
RESUMEN	V
DECLARACIÓN	VI
TABLA DE CONTENIDOS	VII
ÍNDICE DE TABLAS	X
ÍNDICE DE FIGURAS	XI
ÍNDICE DE ANEXOS	XIII
<b>CAPÍTULO I</b>	2
<b>MARCO REFERENCIAL</b>	2
1.1 Identificación del problema.	2
1.2 Situación actual del problema.	2
1.3 Justificación del tema.	3
1.4 Objetivos	3
1.4.1 Objetivo General	3
1.4.2 Objetivos Específicos	3
1.5 Hipótesis	4
1.6 Resultados Esperados	4
<b>CAPÍTULO II</b>	5
<b>MARCO TEÓRICO</b>	5
2.1 Antecedentes Históricos de la robótica	5
2.1.1 Antecedentes Históricos de los robots de vigilancia y seguridad	5
Modelos militares	6
2.2 Bases Teóricas	7
2.2.1 Robótica Educativa.	7
2.2.2 Sistema de locomoción	9



2.2.3	Sensores	14
2.2.4	Tecnología Zigbee	17
2.2.5	Arduino Mega 2560	18
2.2.6	Display LCD 16X2	20
2.2.7	Dispositivo de alimentación Batería de Polímero de Litio (LI-PO)	20
2.2.8	Cámara inalámbrica	21
2.2.9	Herramientas para el desarrollo del prototipo	23
2.3	Variables	23
2.4	Métodos e instrumentos de investigación.	25
2.5	Términos básicos	26
<b>CAPÍTULO III</b>		28
<b>ANÁLISIS</b>		28
3.1	Diagrama del proceso	28
3.1.1	Descripción funcional de los procesos	28
3.2	Identificación de requerimientos	30
3.3	Análisis del sistema	31
3.3.1	Análisis Técnico	31
3.3.2	Análisis económico.	32
3.3.3	Análisis operativo	35
3.4	Métodos e instrumentación de investigación	35
<b>CAPÍTULO IV</b>		37
<b>DISEÑO</b>		37
4.1	Diseño	37
4.2	Arquitectura de la solución	38
4.2.1	Diagrama de algoritmo del funcionamiento del robot	44
4.3	Diseño de interfaz gráfica	46
<b>CAPÍTULO V</b>		48
<b>IMPLEMENTACIÓN</b>		48
5.1	Implementación	48
5.2	Construcción de la carcasa del prototipo	48
5.2.1	Sistema de control	51
5.2.2	Sistema sensorial	52

5.2.3	Sistema de comunicación.-	53
5.2.4	Sistema Actuador	54
5.2.5	Sistema de potencia y alimentación	54
5.3	Pruebas	55
5.3.1	Prueba y resultados	56
5.3.2	Pruebas y resultados del recorrido autónomo libre de obstáculo	57
5.4	Documentación elaboración de manual y guía de usuario	59
5.4.1	Comprobación de hipótesis	59
5.4.2	Manual de usuario y guía manual de prácticas	59
6	<b>CONCLUSIONES</b>	60
7	<b>RECOMENDACIONES</b>	61
8	<b>BIBLIOGRAFÍA</b>	62

## ÍNDICE DE TABLAS

ITEM	DESCRIPCIÓN	PÁGINA
Tabla 1.	Características Generales Del Servomotor MG995 Tower Pro.	12
Tabla 2.	Características generales de la batería LI-PO.	21
Tabla 3.	Especificaciones generales del receptor de cámara inalámbrica de video.	22
Tabla 4.	Variable independiente	24
Tabla 5.	Variable Dependiente	25
Tabla 6.	Hardware para el desarrollo	31
Tabla 7.	Software para el desarrollo	31
Tabla 8.	Requerimiento de hardware para el prototipo	32
Tabla 9.	Costo de software de desarrollo	33
Tabla 10.	Costo de Hardware de desarrollo	33
Tabla 11.	Suministros	33
Tabla 12.	Costo requerimiento de hardware para el prototipo	34
Tabla 13.	Costos totales de desarrollo	34
Tabla 14.	Consumo de energía de los elementos	43
Tabla 15.	Asignación de pines para los componentes electrónicos	52
Tabla 16.	Resultado de prueba del sensor ultrasónico	56
Tabla 17.	Resultado de pruebas del sensor PIR	57
Tabla 18.	Prueba recorrido autónomo del robot modo manual	57

## ÍNDICE DE FIGURAS

ITEM	DESCRIPCIÓN	PÁGINA
Figura 1.	Robot de Seguridad Denning Sentry.	6
Figura 2.	Diagrama de robot móvil.	10
Figura 3.	Servomotor.	12
Figura 4.	Posiciones del Motor por Ancho de pulso.	13
Figura 5.	Esquema básico de un puente H.	14
Figura 6.	Sensor Ultrasónico.	15
Figura 7.	Funcionamiento del sensor ultrasónico.	15
Figura 8.	Sensor PIR.	16
Figura 9.	Conexiones mínimas requeridas por el módulo Xbee s1.	17
Figura 10.	arduino Mega 2560.	18
Figura11.	Tarjeta FPGA.	19
Figura 12.	Tarjeta reguladora de voltaje	20
Figura 13.	LCD 16X2.	20
Figura14.	Cámara inalámbrica.	22
Figura 15.	Diagrama de proceso general del prototipo	28
Figura 16.	Diagrama del prototipo	37
Figura 17.	Diseño de la estructura	38
Figura 18.	Diseño de ubicación de elementos del prototipo	39
Figura 19.	Diseño del sistema de locomoción	41
Figura 20.	Diagrama de actividades de modo manual.	44
Figura 21.	Diagrama de actividades de modo automático	45
Figura 22 .	Construcción de la carcasa del prototipo hecho en material acrílico transparente	49
Figura 23.	Ensamblaje de los cuatro motores DC a las llantas y conexiones al Puente H.	49
Figura 24.	Colocación de los elementos al prototipo móvil con sus conexiones	50
Figura 25.	Soporte para la cámara y servomotor, y colocación del resto de elementos	51

Figura 26. Ubicación de tarjeta FPGA y display LCD 16X2	51
Figura 27. Conexión arduino y sensor HC-SR04	53
Figura 28. Conexión de sensor PIR DYP-ME003 a la tarjeta arduino	53
Figura 29. Conexión arduino - Xbee S1	54
Figura 30. Conexiones arduino y motores	54
Figura 31. Conexión de circuito impreso de tarjeta reguladora de voltaje	55
Figura 32. Prototipo terminado	55
Figura 33. Prueba recorrido autónomo del robot modo manual	58
Figura 34. Prueba recorrido autónomo del robot modo manual	58
Figura 35. Diagrama de cámara inalámbrica en labVIEW “vi”	15
Figura 36. Circuito impreso tarjeta reguladora de voltaje	16
Figura 37. Circuito impreso tarjeta Puente H L298N	17
Figura 38. Función del robot móvil de seguridad y vigilancia de manera general	24
Figura 39. Práctica N° 1 recorrido forma cuadrada	29
Figura 40. El robot móvil comienza a recorrer el área cuadrada de distancia 0.4mx0.4m.	30
Figura 41. El robot móvil comienza a recorrer el área cuadrada de distancia 0.4mx0.4m.	30
Figura 42. El robot móvil comienza a recorrer el área cuadrada de distancia 0.4mx0.4m	31
Figura 43. El robot móvil comienza a recorrer el área cuadrada de distancia 0.4mx0.4m	31
Figura 44. Distancia de detección de obstáculo	34
Figura 45. Distancia de detección de obstáculo	35
Figura 46. Alcance de comunicación entre robot móvil y módulo xbee	40

## ÍNDICE DE ANEXOS

<b>ITEM</b>	<b>DESCRIPCIÓN</b>
ANEXO 1:	Código fuente algoritmo de programación en arduino
ANEXO 2:	Código VHDL
ANEXO 3:	Diagrama de cámara inalámbrica en LaBVIEW
ANEXO 4:	Diagrama de circuitos impresos
ANEXO 5:	Guía manual de utilización de batería LI-PO
ANEXO 6:	Manual de usuario
ANEXO 7:	Guía manual de prácticas para laboratorio de Electrónica y Telecomunicaciones.

## INTRODUCCIÓN

A medida que la robótica avanza, el hombre encuentra cada vez más aplicaciones para los robots, donde se requiere que todo estudiante deba actualizar sus conocimientos para poder hacer frente a los nuevos equipos e instrumentos que se presentarán a lo largo de su carrera profesional, entender los distintos lenguajes de programación y el manejo de diferentes sistemas de control.

Con este diseño de prototipo, se intenta incentivar a los estudiantes de la carrera de Electrónica y Telecomunicaciones a que sus prácticas de laboratorio sean motivadoras y útiles para el desarrollo de la robótica y la electrónica en general. Por lo que este proyecto pretende ser un aporte más en el mejoramiento de las asignaturas en: Fundamentos de Robótica, Electrónica y Sistema Digitales haciendo de este un prototipo pedagógico.

El presente documento contiene cinco capítulos: En el capítulo 1 se especifica la identificación del problema, situación actual del problema, la justificación del tema, los objetivos generales, objetivos específicos, hipótesis y resultados esperados del proyecto.

En el capítulo 2 se detalla los antecedentes históricos de los robots de vigilancia y seguridad, base teórica, términos básicos y variables.

En el Capítulo 3 se detalla los requerimientos del proyecto, el cual indica las especificaciones técnicas, presentando una lista de necesidades y diagramas donde se evidencia el análisis realizado.

En el Capítulo 4 se muestra el diseño de la estructura del robot para la solución que más se adapte a los requerimientos y necesidades del proyecto.

En el Capítulo 5 se presenta la implementación del prototipo mostrando evidencias gráficas, registro fotográfico y explicación descriptiva de la realización del proyecto.

Esta tesis es solo una base que tendrán los estudiantes universitarios de la carrera de Electrónica y telecomunicaciones para el avance y mejora de muchos otros trabajos académicos que se basan en tecnología FPGA y arduino las cuales poseen increíble lenguaje de programación. Finalmente se exponen las conclusiones y recomendaciones para obtener un óptimo funcionamiento del sistema.

# CAPÍTULO I

## MARCO REFERENCIAL

### **1.1 Identificación del problema.**

En el laboratorio de la escuela de Electrónica y Telecomunicaciones, se dictan clases y prácticas de las asignaturas como: Fundamentos de la Robótica, Electrónica y Sistema Digitales, en las mismas se obtienen conocimientos teóricos-prácticos, generalmente las prácticas vinculadas a las materias se ven reducidas a simulaciones por computadora, debido a la carencia de componentes e instrumentos necesarios para la enseñanza debido a su elevado costo.

Por esta razón surge la necesidad de diseñar e implementar un prototipo; una aplicación de la robótica, un robot de vigilancia y seguridad con vision nocturna desmontable que permitirá realizar pruebas y adquisición de datos de los diferentes dispositivos, que será de ayuda para formar profesionales competitivos en el campo de la robótica y desarrollo de la tecnología.

### **1.2 Situación actual del problema.**

La seguridad y la vigilancia es uno de los campos en el cual ha evolucionado la robótica, es por ello que se ha creado diferentes tecnologías que ayudan a cumplir con tareas que son establecidas para un determinado prototipo. En estos robots se ha instalado sistemas de vigilancia y seguridad que permiten observar y controlar un establecimiento de forma autónoma. Además tienen sistemas propios que le permiten reaccionar ante acontecimientos externos, como puede ser la presencia ante un individuo u objeto.

Actualmente el laboratorio de la escuela de Electrónica y Telecomunicaciones, carece de prototipo de robot móvil de vigilancia y seguridad desmontable, por lo tanto optamos por implementar este proyecto que servirá como herramienta de trabajo para las prácticas de laboratorio, motivando a los estudiantes universitarios a que desarrollen sus habilidades haciendo uso de este prototipo.



### **1.3 Justificación del tema.**

Este proyecto plantea diseñar e implementar un prototipo de robot móvil de vigilancia y seguridad con visión nocturna desmontable, con este prototipo se busca incrementar el desarrollo de las prácticas en laboratorio de la escuela de Electrónica y Telecomunicaciones de la Universidad Estatal Península de Santa Elena.

La tecnología robótica ayudará al estudiante universitario a crear líneas de investigación sobre robots de vigilancia y seguridad, destacando algunos beneficios para implementar funcionalidades personalizadas en arduino y FPGA obteniendo ventajas como.

1. Rendimiento
2. Precio
3. Fiabilidad
4. Realizable de programar

Económicamente el proyecto es factible para realizarlo ya que se cuenta con todos los elementos electrónicos tienen bajos costos para adquirirlos por lo tanto es viable.

### **1.4 Objetivos**

#### **1.4.1 Objetivo General**

Implementación de robot móvil de vigilancia y seguridad desmontable aplicando módulos electrónicos y tecnología FPGA para las prácticas de laboratorio de fundamentos de robótica.

#### **1.4.2 Objetivos Específicos**

- Diseño del robot móvil de vigilancia y seguridad para la creación del módulo con fines didácticos.
- Diseñar la circuitería del sistema de control y sus dispositivos del robot móvil de vigilancia y seguridad con visión nocturna para la elaboración de ensamblaje de elementos de conexión desmontable del robot.

- Evaluar los sistemas y elementos utilizados en el robot móvil de vigilancia y seguridad para la toma de adquisición de datos en las prácticas de laboratorio de fundamentos de robótica.
- Elaborar manual técnico y guía de las prácticas con el robot de vigilancia y seguridad como recurso didáctico para el laboratorio de Electrónica y Telecomunicaciones.

### **1.5 Hipótesis**

La implementación de un robot móvil de vigilancia y seguridad con visión nocturna desmontable tendrá una eficiencia del 90% en la detección del objeto, la distancia a la que se encuentra el objeto y comunicarse de forma inalámbrica con el usuario.

### **1.6 Resultados Esperados**

- Se espera que la estructura del robot móvil sea la adecuada para adaptar los elementos utilizados para la creación de modulo desmontable.
- Con la investigación de la tecnología FPGA y arduino se espera demostrar la robótica aplicada, utilizando un robot de seguridad y vigilancia con visión nocturna desmontable como herramienta para las prácticas de laboratorio
- Con los diferentes dispositivos electrónicos utilizados en el prototipo se espera realizar las prácticas para adquisición de datos en el laboratorio de robótica
- Se espera realizar una guía práctica del robot como documento didáctica para las clases impartidas en el laboratorio.

# CAPÍTULO II

## MARCO TEÓRICO

### 2.1 Antecedentes Históricos de la robótica

La robótica es la ciencia y la tecnología de los robots. Se ocupa del diseño, fabricación y aplicaciones de los robots. La robótica combina varias disciplinas como son: la mecánica, la electrónica, la informática, la inteligencia artificial y la ingeniería de control. Otras áreas importantes en robótica son el álgebra, los autómatas programables y las máquinas de estados.

En 1920 Karel Capek utiliza la palabra robot se popularizó con el éxito de la obra RUR “Robots Universales Rossum”. De origen checa robota, que significa trabajos forzados, fue traducida al inglés como robot.

Isaac Asimov definiendo a la ciencia que estudia a los robots, comenzó en 1939 a contribuir con varias relaciones referidas a robots y a él se le atribuye el acuñamiento del término Robótica y con el surgen las denominadas tres Leyes de Robótica que son las siguientes:

- Un robot no puede actuar contra un ser humano o, mediante la falta de movimiento, que un ser humano sufra daños.
- Un robot debe de obedecer las órdenes dadas por los seres humanos, salvo que estén en conflictos con la primera ley.
- Un robot debe proteger su propia existencia, a no ser que esté en conflicto con las dos primeras leyes.

#### 2.1.1 Antecedentes Históricos de los robots de vigilancia y seguridad

Las tareas de un robot de vigilancia son múltiples, desde hacer un recorrido por determinados lugares, comprobar perímetros, seguir y vigilar a determinadas personas, incluso actuar en casos de incendio.

En la década de los 80 se avanza en las técnicas de reconocimiento de voz, detección de objetos móviles y factores de seguridad. También se desarrollaron los primeros robots en el campo de la rehabilitación, la seguridad con fines militares y la realización de tareas peligrosas.

### **Modelos militares**

Un impulsor muy significativo de este tipo de investigaciones es el desarrollo de equipos de espionaje militar. Para proteger aquellas personas que ponen su vida en peligro, los robots de seguridad y defensa aptos para el combate pueden realizar numerosas tareas para ayudar a los profesionales de la seguridad pública y del ejército

### **Denning Mobile Robotics**

Diseñó el Denning Sentry, destinado a ser un robot de seguridad. Mediante sensores de detección de intrusos, puede hacer patrullajes de manera sistemática en una instalación, bodega o similares. Cuando la alimentación se agota, el robot retorna automáticamente a su estación de carga y recarga sus baterías sin asistencia de ningún operador; el equipo incorpora un anillo de sonares para detección de obstáculos, sensores de movimiento de tipo infrarrojo y de microondas. También posee una cámara de TV, micrófono y transmisores inalámbricos para enviar la información a la estación de seguridad.



**Figura 1.** Robot de Seguridad Denning Sentry.  
Fuente: Robot de seguridad.com

## **Technorobot**

VisionBot es un robot de seguridad, diseñado y desarrollado por Technorobot, para intervenciones en áreas con un alto grado de conflicto mediante su uso, donde las condiciones no son propicias para una acción en primera persona, salva la integridad física de los operadores y equipos que lo dirigen. Se opera remotamente, con motorización 6x6 que incorpora 3 cámaras de infrarrojos de alta resolución para obtener un rango de observación de 360° a más de 60 metros, día y noche. Todas estas razones convierten a este robot en la mejor opción para cualquier intervención de riesgo que deba ser llevada a cabo por Unidades de Vigilancia de Prisiones, Equipos de Intervención Policial, y Unidades Militares con necesidades de observación.

## **Sistema Lego -Sector Educativo**

El avance de sistema y la investigación aporta a la generación de conocimientos. Específicamente, los sistemas LEGO han desarrollado la robótica en todo el sentido de la palabra; ellos brindan información, elementos y partes que permiten crear cualquier sistema, constituyéndose en un soporte educativo muy fuerte a partir de la utilización potencial del software, piezas mecánicas, sensores, sistemas de desarrollo y demás elementos necesarios para que el estudiante construya sus propios desarrollos asimilando el conocimiento requerido.

### **2.2 Bases Teóricas**

En el presente proyecto de titulación hace énfasis de conceptos en relación a la Robótica Educativa, así como también a los componentes que incluyen en el proyecto; esto involucra al hardware que se hallan en los módulos de locomoción, sensores, control y comunicación mediante la aplicación de software que se utiliza para controlar el prototipo.

#### **2.2.1 Robótica Educativa.**

En el marco de nuestra experiencia creamos la robótica educativa como un contenido de enseñanza y aprendizaje que se apoya en las tecnologías digitales para hacer robótica, incluyendo a quienes participan en el diseño y construcción de prototipos de robots propios.

Estas creaciones pueden tener su origen, en un referente real, por ejemplo: un proceso industrial automatizado, en el que los estudiantes recrean desde la apariencia de las maquinas hasta las formas de movimiento.

El objetivo de la enseñanza de la robótica, es lograr una adaptación de los alumnos a los procesos productivos actuales, en donde la automatización destaca un rol muy importante. No obstante, la robótica se considera un sistema que va más allá de una aplicación laboral.

### **Definición de robot.-**

El robot puede ser un dispositivo electromecánico físico, que funciona en conjunto con un sistema de software que pueden ejecutar distinto tipos de operaciones y movimientos de forma automática. El origen del término robot proviene de una palabra checa robota = trabajo dispositivo mecánico dotado de articulaciones móviles destinado al manejo. Es un manipulador reprogramable y multifuncional diseñado para mover material, partes, herramientas o bien dispositivos especializados para desempeñar una variedad de labores a través de movimientos diversos programados. Un robot está formado por elementos principales como son: transmisores, estructura mecánica y eléctrica, sistema de accionamiento, sistema sensorial, sistema de control y elementos terminales.

En general, un robot, para ser considerado como tal, debería presentar algunas de estas propiedades:

- Puede sentir su entorno.
- No es natural, sino que ha sido creado artificialmente.
- Puede moverse en uno o más ejes de rotación o traslación.
- Puede realizar movimientos coordinados.
- Puede manipular cosas de su entorno.
- Es reprogramable.
- Tiene cierta inteligencia o habilidad para tomar decisiones, basadas en el ambiente o en una secuencia pre programada automática.

### **Clasificación de los robots.-**

Los robots pueden clasificarse de varias formas y maneras. Una de las más conocidas formas de clasificación es la determinada por su arquitectura.

La arquitectura, es definida por el tipo de configuración general del Robot.

Nos enfocamos en la clasificación según su arquitectura, debido a que este proyecto va enfocado a un tipo de robot que está dentro de esta clasificación.

### **Robots móviles.-**

Los robots móviles están provistos de patas, ruedas u orugas los cuales permiten desplazarse de acuerdo su programación. Elaboran la información que reciben a través de sus sistemas de sensores y actuadores, se emplean en determinado tipo de instalaciones industriales, sobre todo para el transporte de mercancías en cadenas de producción y almacenes. También se utilizados para la investigación en lugares de difícil acceso o muy distantes para los seres humanos, como es el caso de la exploración espacial y las investigaciones o rescates submarinos.

#### **2.2.2 Sistema de locomoción.-**

El sistema de locomoción (SL) es el responsable de la traslación del robot en el campo de juego. Existen una gran variedad de maneras de moverse sobre una superficie; entre los robots móviles, ruedas, cadenas y patas.

Los robots con ruedas son la solución más simple y eficiente para lograr la movilidad en terrenos firmes y libres de obstáculos, y son fáciles de controlar.

Emplean diferentes características y propiedades en cuanto a la eficiencia energética, dimensiones, cargas útiles y maniobrabilidad.

**Robot con tracción diferencial.-** Una plataforma móvil de tracción diferencial se caracteriza porque el movimiento se consigue con dos pares de ruedas; dos ruedas de tracción que tienen acoplados cada uno a su propio motor, dos ruedas de equilibrio que mantienen el control del robot.

De esta manera se obtiene que el robot avance en línea recta precisando ambos motores a la misma velocidad, se puede hacer girar en una u otra dirección cuando se aplican velocidades diferentes.

Para hacer más tratable el problema de modelado, en la obtención de los modelos cinemáticos de RMR “Robot móvil de rueda”, es común introducir suposiciones de diseño y de operación prácticas se encuentran las siguientes:

1. El robot móvil de rueda no contienen partes flexibles.
2. Los ejes de dirección son perpendiculares a la superficie.
3. Los Robot móviles de rueda se mueven sobre una superficie plana.
4. La fricción rotatoria en el punto de contacto entre la rueda y la superficie es suficientemente pequeña para que el desplazamiento rotatorio ocurra.

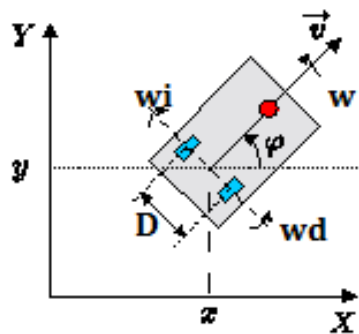


Figura 2. Diagrama de robot móvil.

Fuente: Silvia Ortigoza, Molina Vilchis, M. Marciano y Portilla F (2008).  
Modelado y Control de un robot de rueda tipo diferencial.

### Modelo cinemático del robot móvil.-

Para el modelado del robot móvil se considera que el movimiento del robot se realiza en el plano  $xy$ , con una configuración de tracción diferencial. En la figura 2 se muestra el diagrama del robot móvil para el desarrollo de modelo cinemático del robot móvil diferencial. De acuerdo a la Figura  $(x, y)$  denota la posición del punto medio del eje que une las dos llantas traseras,  $\varphi$  describe el ángulo que forma el eje de simetría del móvil respecto al eje  $X$  positivo,  $w_d$  y  $w_i$  son las velocidades lineales de las llantas derecha e izquierda, respectivamente, y  $D$  es la distancia entre ellas.



$$x = \frac{(w_i - w_d)}{2} \cos(\varphi)$$

$$y = \frac{(w_i - w_d)}{2} \sin(\varphi)$$

$$\varphi = \frac{(w_d - w_i)}{D}$$

De las ecuaciones es claro que para **D** dados, la única forma de imponer que las variables del sistema (x, y,  $\varphi$ ), generen una trayectoria deseada queda determinada por la elección apropiada de  $w_d$  y  $w_i$ , las cuales son conocidas como variables de control del móvil

### **Motores DC.**

Es una máquina que convierte la energía eléctrica en mecánica, lo que hace que se produzca un movimiento rotatorio. Este dispositivo, es un motor de engranaje de gran torque y fácil de usar. Su alimentación es de 6 a 24Vdc, con una tensión nominal de 12V/ 500 mA y una potencia de 1.25W. Su fácil control de posición, par y velocidad lo han convertido en una de las mejores opciones en aplicaciones de control y automatización de procesos. El principio de funcionamiento se basa en la repulsión que ejercen los polos electromagnéticos de un magnetismo que se encuentra en el eje.

### **Servomotor MG995 Tower Pro**

El servomotor es el encargado de realizar el movimiento de la cámara, el modelo utilizado es el MG995 Tower Pro.

El servo MG995 Tower Pro tiene un conector universal tipo “S” que encaja perfectamente en la mayoría de los receptores de radio. Los cables en el conector están distribuidos de la siguiente forma:

Rojo = 5Vcc, Café = GND o tierra, Naranja = Señal PWM.

Este servomotor se puede utilizar sin ningún problema con las tarjetas arduino y compatibles, también se puede usar con microcontroladores PIC, AVR, MSP430, etc. Recomendamos alimentar por separado el microcontrolador y los servos, ya que el ruido eléctrico puede dar lugar a errores en la ejecución del programa.



Figura 3. Servomotor.  
Fuente: <http://servomotor.com>

A continuación se muestra las especificaciones generales del servomotor:

Especificaciones generales del servomotor MG995 Tower Pro	
Dimensiones	1.57" x 0.79" x 1.44" (40 x 20 x 36.5mm)
Peso	1.78oz (48g)
Voltaje recomendada de entrada	3.5 - 8.4 Voltios, nominal 5 voltios.
Velocidad de Operación (4.8V sin carga):	0.17seg / 60 grados
Velocidad de Operación (6V sin carga):	0.13seg / 60 grados
Torque de parada (4.8V):	(13kg/cm)
Torque de Parada (6.0V):	(15kg/cm)
Rango de Temperatura:	-30 to +60 grados C

Tabla 1. Características Generales Del Servomotor MG995 Tower Pro.  
Fuente: [www.geelmed.com](http://www.geelmed.com)

### **Funcionamiento del servomotor - Control PWM**

La modulación por anchura de pulso, PWM es un sistema empleado para el control de servos, consiste en generar una onda cuadrada en la que se varía el tiempo que el pulso está a nivel alto, manteniendo normalmente el mismo período para modificar la posición del servomotor.

Para la generar una onda PWM en un microcontrolador, lo recomendado es usar un timer y un comparador, hará que la señal sea automática y más efectiva. El mecanismo consiste en programar el timer con el ancho del pulso y al comparador con el valor de duración del pulso a nivel alto.

El sistema de control de un servo se confina a indicar en qué posición se debe situar. Los valores más generales se corresponden con pulsos de entre 1 ms y 2 ms de anchura, que dejarían al motor en ambos extremos entre 0° y 180°. La posición del valor 1.5 ms indica la posición central es decir 90°. Cuando sobrepasa los límites de nivel a 1 ms y a 2ms fuera del rango recomendado el servo emite un zumbido, indicando que se debe cambiar la longitud del pulso.

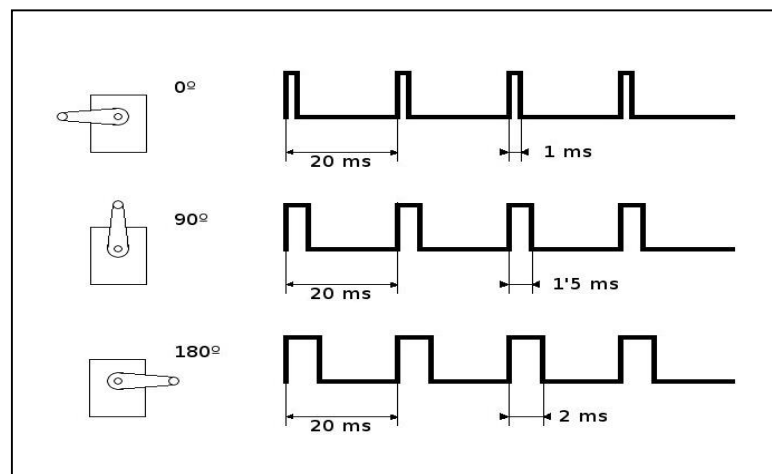


Figura 4. Posiciones del Motor por Ancho de pulso.  
Fuente: Máquinas eléctricas.

**Driver de potencia para control de motores.**-La etapa de potencia es el elemento que hace de intermediario entre las partes mecánicas y electrónicas. Para controlar un motor mediante un microcontrolador se debe tener en cuenta que su conexión no sea de manera directa, ya que el motor tiene un consumo de corriente que el chip no le puede proporcionar, esto provocaría que se quemara la parte electrónica.

Para evitar esto, se coloca una etapa de potencia entre el microcontrolador y los motores, por lo tanto es deseable que esta etapa de potencia nos permita cambiar el sentido de giro de los motores. El método más sencillo para lograr la intensidad requerida es mediante el uso de transistores funcionando como interruptores, en este proyecto se utilizara circuito integrado Puente H L298N.

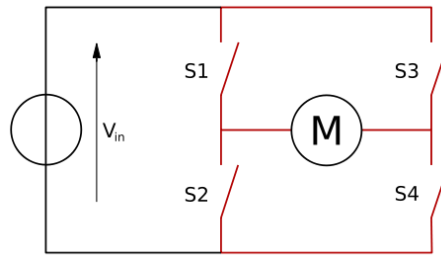


Figura 5. Esquema básico de un puente H.  
Fuente: <http://drbervatov.blogspot.com/>

El uso de este circuito permite invertir la corriente en estos motores, cediendo el cambio del sentido de giro si se mantienen abiertos los interruptores s2 y s3, y cerrados s1 y s4, la parte derecha del motor estará conectada a tierra y la izquierda a alimentación. El motor está girando en un sentido.

Si se cierran los interruptores s2 y s3, y se abren el s1 y el s4, ahora la parte derecha del motor está conectada a alimentación, mientras que la izquierda lo está a tierra, justo al contrario que en el caso anterior. Con esta nueva configuración, el motor gira hacia el sentido contrario que antes.

## 2.2.3 Sensores

### 2.2.3.1 Introducción

Es un proceso que necesita ser controlado, los sensores son los elementos que miden las diferentes variables, indican los errores, recogen los estados y transmiten esta información al sistema de control de proceso.

Para alcanzar que el robot realice una tarea con la adecuada precisión es necesario que tenga conocimiento tanto de su propio estado como del estado de su entorno.

Para este proyecto se hace uso de sensor de ultrasonido, sensor de movimiento, y buzzer activo.

### **Sensor ultrasónico HC-SFR04.-**

Son detectores de proximidad que trabajan libres de roces mecánicas y que detectan objetos a distancias que van desde pocos centímetros hasta varios metros. Son vibraciones del aire, y pueden detectar objetos con diferentes formas, colores, superficies y materiales, que el sonido audible pero de una frecuencia superior a 20kHz, no audible por el oído del ser humano.

Trabajan según el tiempo de transcurso del eco, es decir, se aprecia la distancia temporal entre el impulso de emisión y el impulso del eco, si en un determinado margen de tiempo no se ha recibido eco, se considera que no hay obstáculo.



Figura 6. Sensor Ultrasónico.  
Fuente: <http://articulo.mercadolibre.com.ec/>

Midiendo el tiempo que transcurre entre la emisión del sonido y la percepción del eco se puede establecer la distancia a la que se encuentra el obstáculo que ha producido la reflexión de la onda sonora, utilizando la ecuación donde  $V$  es la velocidad del sonido en el aire y  $t$  es el tiempo transcurrido entre la emisión y recepción del pulso

$$d = \frac{1}{2} V * t$$

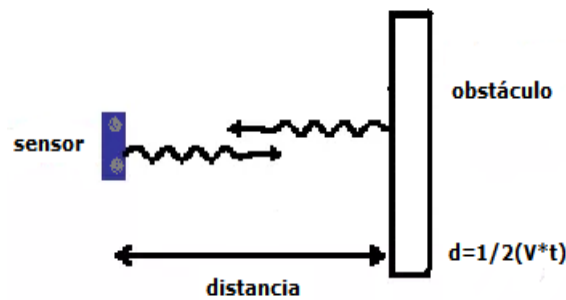


Figura 7. Funcionamiento del sensor ultrasónico.

### **Sensor PIR DYP- Me003.-**

El sensor PIR corresponde a sus siglas Passive Infra red, es un dispositivo piro eléctrico “detector de calor” que mide los cambios en los niveles de radiación infrarroja emitida por los objetos a su alrededor a una distancia que depende del modelo del sensor,

Este sensor mide el cambio de calor, más no la intensidad de calor, contiene dos ranuras hecha de un material especial que es sensible a IR con las que se puede “ver” hacia fuera más allá cierta distancia. Cuando el sensor está inactivo, ambas ranuras detectan la misma cantidad de IR, la cantidad ambiente radiada desde la sala o las paredes o al aire libre. Cuando un cuerpo caliente como un humano pasa por delante, primero intercepta la mitad del sensor PIR, lo que provoca un cambio diferencial positiva entre las dos mitades. Cuando el cuerpo caliente sale de la zona de detección, lo contrario sucede, por lo que el sensor genera un cambio diferencial negativo. Estos pulsos de cambio son lo que se detecta.



Figura 8. Sensor PIR.

Fuente: <http://articulo.mercadolibre.com.ec/>

### **Sensor buzzer.-**

Un timbre o zumbador es un dispositivo de señalización acústica, que puede ser mecánico, electromecánico, o electrónico. Se lo maneja a través del arduino u otros controladores, este módulo será capaz de controlar el sonido de la alarma o la música MID fácilmente. Se extiende con los sensores de placa arduino utilizados en combinación para lograr el control de un sonido interactivo y obras de luz.

Configuración de pines del módulo:

1. Salida,
2. Fuente de alimentación
3. GND.
4. Módulo de conexión de interfaz para PH2.0.
5. Características:
6. Tipo: Digital.
7. Fuente de alimentación: 5VDC

## 2.2.4 Tecnología Zigbee.-

Zigbee es un protocolo de alto nivel de comunicación inalámbrica para su utilización con radiodifusión digital, basada en el estándar IEEE 802.15.4. Zigbee permite que dispositivos electrónicos de bajo consumo puedan realizar sus comunicaciones inalámbricas. Las comunicaciones Zigbee se realizan en la banda libre de 2.4GHz. El alcance depende de la potencia de transmisión del dispositivo así como también del tipo de antenas utilizadas su velocidad es de hasta 256kpps.

**Xbee.-** Los Xbee's son pequeños chips azules capaces de comunicarse de forma inalámbrica unos con otros. Pueden hacer cosas simples, como reemplazar un par de cables en una comunicación serial, lo cual es notable cuando deseas crear, por ejemplo, un robot móvil radio controlado. El módulo requiere una alimentación desde 2.8 a 3.4 V, la conexión a tierra y las líneas de transmisión de datos por medio del UART (TXD y RXD) para comunicarse con un microcontrolador, o directo a un puerto serial utilizando algún convertidor adecuado para los niveles de voltaje. Para este proyecto se utilizara Xbee s1, estos no necesitan ser configurados; por lo tanto fáciles para trabajar e ideales para empezar, y poseen comunicación punto a punto, punto-multipunto.

### Circuito básico para Xbee.-

La figura 9 muestra las conexiones mínimas que necesita el módulo Xbee para poder ser utilizado en el robot móvil de vigilancia y seguridad. Luego de esto se debe configurar según el modo de operación apropiada para la aplicación solicitada por el beneficiario.

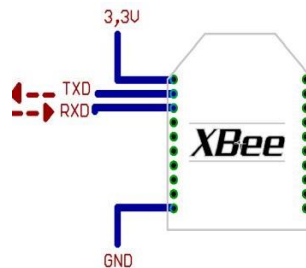


Figura 9. Conexiones mínimas requeridas por el módulo Xbee s1.  
Fuente: Guía del Usuario - Xbee serie 1

### 2.2.5 Arduino Mega 2560.-

Arduino es una plataforma de electrónica abierta para la creación de prototipos basados en software y hardware flexibles y fáciles de usar. Permite conectarse a través del ordenador y por medio de lenguaje de programación, es posible que el usuario logre interactuar con componentes electrónicos y controlarlos por software, los proyectos realizados con arduino son capaces de actuar de manera autónoma sin necesidad de conectarse a un ordenador, si bien tienen la posibilidad de hacerlo y comunicarse con otros tipos de software.

Existen gran variedad de modelos de arduino con diferentes características. Cada modelo posee un nombre, formas, capacidades y funciones distintas. Para este proyecto se hace uso de la tarjeta arduino Mega 2560 como se muestra en la figura 10.

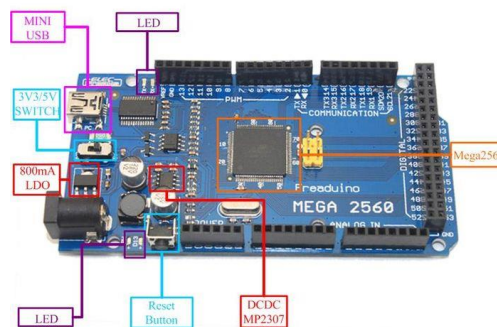


Figura 10. arduino Mega 2560.  
Fuente: <http://arduino-mega-2560>

Una versión ampliada de la tarjeta original de arduino y está basada en el microcontrolador Atmega2560.

Esta tarjeta dispone de 54 entradas /salidas digitales, 14 de las cuales se pueden utilizar como salidas PWM. Además dispone de 16 entradas analógicas, 4 UARTs (puertas series), un oscilador de 16MHz, una conexión USB, un conector de alimentación, un conector ICSP y un pulsador para el reset.

Para comenzar a utilizar la placa arduino sólo es necesario conectarla al ordenador por medio de un cable USB, o alimentarla con un adaptador de corriente AC/DC. También, para empezar, puede alimentarse mediante una batería.



**Tarjeta FPGA.-** Es un dispositivo semiconductor que contiene bloques de lógica cuya interconexión y funcionalidad puede ser configurada mediante un lenguaje de descripción especializado. La lógica programable puede reproducir desde funciones tan sencillas como las llevadas a cabo por una puerta lógica o un sistema combinatorial hasta complejos sistemas en un chip.

Las FPGAs se utilizan en aplicaciones similares a los ASICs sin embargo son más lentas, tienen un mayor consumo de potencia y no pueden abarcar sistemas tan complejos como ellos. A pesar de esto, las FPGAs tienen las ventajas de ser reprogramables, sus costes de desarrollo y adquisición son muchos menores para pequeñas cantidades de dispositivos y el tiempo de desarrollo es también menor.



Figura 11. Tarjeta FPGA.  
Fuente: <http://es.aliexpress.com/>

### **Tarjeta reguladora de voltaje.-**

La tarjeta reguladora de voltaje utiliza dos transistores, el 7805 que trabaja en un rango de 7 – 25 V y el 7809 que trabaja en un rango de 11.5 a 25 V por lo tanto el suministro de voltaje hacia la tarjeta no tiene que ser inferior a 6V ya que si se llegara a presentar este caso la tarjeta presentaría problemas de funcionamiento, adicionalmente si el rango de voltaje sobrepasa los 25V esto llegaría a dañar los transistores, el rango recomendado para que la tarjeta funcione en óptimas condiciones sería de 7 – 25 Voltios.

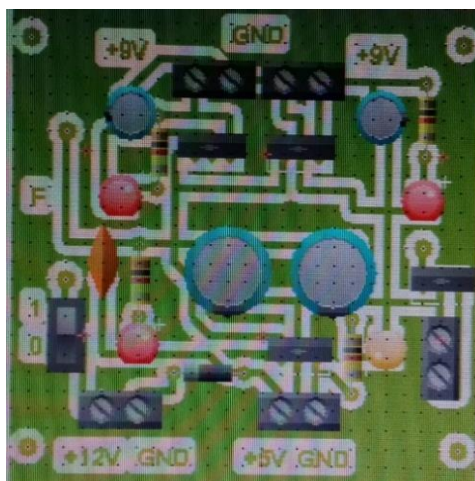


Figura 12. Tarjeta reguladora de voltaje

### 2.2.6 Display LCD 16X2.-

La pantalla de cristal líquido o LCD es un dispositivo de visualización gráfico para la presentación de caracteres, símbolos o incluso dibujos, es este caso dispone de 2 filas de 16 caracteres cada una y cada carácter dispone de una matriz de 5x7 píxeles, aunque los hay de otro número de filas y caracteres. Este dispositivo está gobernado internamente por un microcontrolador y regula todos los parámetros de presentación, este modelo es el más comúnmente usado y esta información se basará en el manejo de este u otro LCD compatible

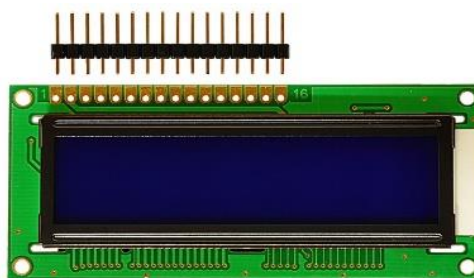


Figura 13. LCD 16X2.

Fuente: <http://www.gravitech.us/>

### 2.2.7 Dispositivo de alimentación Batería de Polímero de Litio (LI-PO).-

El robot móvil es alimentado por una batería de Polímero de Litio tiene de 11.1 V a 2.2 A, cuenta con las siguientes ventajas y desventajas:

- Alta densidad de energía.
- Ocupan poco espacio.
- Escasa resistencia interna, lo cual permite aprovechar casi el 100% de la energía
- No admiten carga rápida.
- Necesita de cargador específico.
- No toleran mucha carga, podrían llegar a dañarse

Especificaciones generales de la batería LI-PO	
-Capacidad Mínima:	2.2Ah
-Configuración:	3S1P / 11.1v / 3celdas
-Constante de descarga:	30C
-Pico de descarga (10sec):	40C
-Peso:	135g
-Dimensiones:	145 x 49 x 27 mm
-Plug de carga:	JST-XH
-Plug de descarga:	5.5mm Bullet-connector.

Tabla 2. Características generales de la batería LI-PO.  
Fuente: <http://mercadolibre.com/especificaciones>

### 2.2.8 Cámara inalámbrica.-

Una cámara oculta también denominada "cámara espía", es una cámara de vídeo usada para captura de imagen de una persona o intruso. La cámara está "oculta" ya que, o bien no es visible para el sujeto que está siendo filmado, o se camuflada con la forma de otro objeto. Las cámaras ocultas se han hecho populares para la vigilancia del hogar, y puede ser incorporada en objetos domésticos comunes, tales como detectores de humo, despertadores, móviles. Las cámaras ocultas también pueden ser utilizadas comercialmente o industrialmente como cámaras de seguridad.



Figura 14. Cámara inalámbrica.  
Fuente: articulooulo.mercadolibre.com

**Especificaciones técnicas:**

- Sistema: NTSC
- Definición: 380 líneas
- Sensor: 1/3 CMOS
- Distancia focal lente: 6mm
- Ángulo de visión: 60°
- Frecuencia de escaneo: 60 Hz
- Iluminación mínima: 2 lux
- Frecuencia transmisión: 2.4 GHz
- Potencia: 100 MW
- Alcance transmisión: 50 a 100 mts
- Alimentación cámara: 9 Volts DC
- Dimensiones: 25 x 35 x 15 mm

**Receptor de cámara inalámbrica de video**

Este se encarga de gestionar la transmisión de video que proviene de la cámara inalámbrica infrarroja.

Especificaciones Generales del Receptor de cámara inalámbrica de video	
Alimentación	9 a 12 Vdc
Frecuencia de trabajo	900 Mhz – 1200 Mhz
Salida	RCA(video)
Dimensiones	20x80x140 mm
Peso	145

Tabla 3. Especificaciones generales del receptor de cámara inalámbrica de video.  
Fuente: <http://mercadolibre.com/especificaciones>

### **2.2.9 Herramientas para el desarrollo del prototipo**

Para la implementación de este prototipo se manejó programas que facilitan el desarrollo del mismo, para ejecutar su funcionamiento las principales herramientas de software utilizadas son:

**Software Arduino.-** El código abierto Arduino Software (IDE) hace que sea fácil de escribir código y cargarlo a la tarjeta E/S. Se ejecuta en Windows, Linux y Mac OS X. El entorno está escrito en Java y basados en el procesamiento y el otro software de código abierto.

**Software Quartus II versión 10.1.-** Quartus II es diseño de la colección de software de Altera para el análisis y la síntesis de diseños realizados en HDL. Permite al desarrollador de cada programa compilar sus diseños, realizar análisis temporales, examinar diagramas RTL y configurar el dispositivo de destino con el programador.

**Software NI LabVIEW 2013.-** Es una plataforma y entorno de desarrollo para diseñar sistemas, con un lenguaje de programación visual gráfico. Estos programas desarrollados con LabVIEW se llaman Instrumentos Virtuales, o VIs, es recomendado para sistemas hardware y software de pruebas, control y diseño, simulado, pues acelera la productividad.

**X-CTU.-** Este es un programa diseñado para configurar y probar módulos Xbee, no solo eso, también contiene un terminal con el cual poder mandar y recibir datos mediante el puerto que está conectado el Xbee.

### **2.3 Variables**

Se detalla las operaciones de las variables independiente y dependiente.

**Variable independiente: Prototipo de robot móvil de vigilancia y seguridad con visión nocturna desmontable aplicando tecnología FPGA como herramienta de laboratorio para adquisición de datos.**

Variable	Definición	Dimensiones	Indicadores	Instrumento
Prototipo de robot móvil de vigilancia y seguridad con visión nocturna desmontable aplicando tecnología FPGA como herramienta de laboratorio para adquisición de datos.	Son robots con capacidad de desplazamiento, basada en plataformas y dotadas de un sistema locomotor de tipo rodante, se desplazan con movilidad basada en programación del micro controlador y la detección de movimiento y objeto por medio de sensores.	<ul style="list-style-type: none"> <li>-Sistema de control</li> <li>sistema de locomoción</li> <li>-sensores de proximidad</li> <li>Trabajo remota</li> <li><b>Guía práctica como herramienta de laboratorio:</b></li> <li>-monitoreo y vigilancia (modo automático)</li> <li>- Adquisición de datos con tarjeta FPGA (modo manual).</li> </ul>	<ul style="list-style-type: none"> <li>Nivel de inteligencia con el que trabaja el robot móvil para su funcionamiento.</li> <li>Modelo cinemático de RMR con tracción diferencial.</li> <li>Uso de sensor ultrasónico “detección de la presencia de objetos u obstáculo” y sensor PIR “detección de movimiento de persona”</li> <li>- medio inalámbrico para la interconexión y comunicación entre dispositivos</li> <li>Robot móvil con visión nocturna usando sensores y cámara inalámbrica</li> <li>Presentar de lectura en LCD utilizando tarjeta FPGA.</li> </ul>	<ul style="list-style-type: none"> <li>programación en tarjeta FPGA y Arduino</li> <li>-Simulación y diagrama en LabVIEW para el manejo de la cámara inalámbrica</li> </ul>

**Tabla 4.** Variable independiente

<b>Variable dependiente : Mejorará la calidad de conocimientos adquiridos en los estudiantes de la carrera de escuela de electrónica y telecomunicaciones en el área de robótica.</b>				
VARIABLE	DEFINICIÓN	DIMENSIONES	INDICADORES	INSTRUMENTO
Permite incrementar el desarrollo de tecnología robótica en la carrera de electrónica y telecomunicaciones	La tecnología arduino aplicada a un robot móvil desmontable que efectúa el recorrido mediante rutas programadas para las prácticas de laboratorio de la escuela de electrónica y telecomunicaciones	Utilización tarjeta FPGA y arduino para la implementación del robot de vigilancia y seguridad.	módulo desmontable pedagógico para las prácticas de laboratorio de la escuela de Electrónica de la carrera de Electrónica y Telecomunicaciones	Técnicas: -Manual técnico y guía de práctica del robot de seguridad y vigilancia desmontable como recurso didáctico.

Tabla 5. Variable Dependiente

#### **2.4 Métodos e instrumentos de investigación.**

Esta sección se divide en etapas en las cuales usamos diferentes métodos de investigación para poder alcanzar el objetivo de cada etapa, debido a que están estructuradas de diferentes formas.

Los métodos empíricos permiten la obtención y elaboración de los resultados adquiridos de las pruebas y sus errores, dicho método revela la relación esencial y las características elementales del tema de estudio, mediante procesos prácticos con el tema y diversos medios de estudio empleado en el método de la primera y segunda etapa.

El método sistémico vendría a ser un orden declarado por reglas, que nos permitiría llegar a tener una comprensión sistémica de una situación dada, este método se aplicará en la tercera y cuarta etapa.

## **Primera Etapa**

**Antecedentes de investigación.-** En esta etapa se investiga la información referente al proyecto tanto teórico como práctico, sobre robots de vigilancia y seguridad y robot desmontables para realizar el desarrollo del tema.

## **Segunda Etapa**

**Definición de insuficiencias del sistema.-** En esta siguiente etapa se prueba describir análisis técnico, análisis económico y operativo de los elementos de software y hardware del prototipo móvil.

## **Tercera Etapa**

**Diseño técnico.-** Se especifica qué tipo de estructura se diseñará para el prototipo y de que material esta hecho, además de su interfaz gráfica y el desarrollo.

## **Cuarta Etapa**

### **Programación y pruebas**

Se proyecta los ensayos y pruebas necesarias para corregir errores y tener resultados registrados para garantizar el funcionamiento adecuado del robot de vigilancia y seguridad con visión nocturna desmontable para uso de las prácticas de laboratorio de la escuela de electrónica y telecomunicaciones.

## **2.5 Términos básicos**

**Locomoción:** Estructura del diseño para realizar movimientos

**Microcontrolador:** Es un elemento electrónico, un circuito de memoria reprogramable.

**Adquisición de datos:** Tomar datos o señales del medio físico para transformarlas en movimientos a través de un robot.

**RMR:** Robot Móvil de Rueda.



**PWM:** Modulación por Ancho de Pulso, cuando se cambia el ciclo de la señal.

**Timer:** Dispositivo con frecuencia programable que permite medir el tiempo

**AC:** Corriente Alterna

**DC:** Corriente directa

**PIR:** Detector de movimiento por infrarrojo.

**LCD:** Es un dispositivo para la presentación de imágenes o caracteres.

**LI-PO:** Polímero de Litio, para alimentar los actuadores de todo el sistema.

**IDE:** Entorno interactivo de desarrollo, que permite programar fácilmente la tarjeta.

# CAPÍTULO III

## ANÁLISIS

### ANÁLISIS

Este capítulo se describe a la etapa de análisis del proyecto, especificaciones técnicas, procedimientos y viabilidad del prototipo.

#### 3.1 Diagrama del proceso

A continuación se muestra el diagrama de bloques del sistema que controlará el prototipo de vigilancia y seguridad desmontable; para esto se consideró dividir el sistema en pequeños bloques que se considerarán como subsistemas para su respectivo funcionamiento, como se aprecia en la siguiente figura 15.

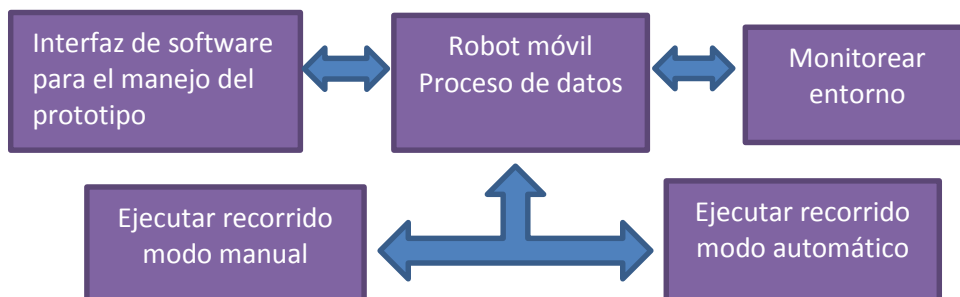


Figura 15. Diagrama de proceso general del prototipo

#### 3.1.1 Descripción funcional de los procesos

En esta sección se describe cada uno de los procesos que forman parte del proyecto.

**Interfaz de software para el manejo del prototipo.-** La interfaz con la que se opera el prototipo es por medio de software arduino ya que maneja un lenguaje de programación sencillo para su ejecución, designando tareas específicas y enviando comandos al prototipo a través del módulo de comunicación haciendo uso de un ordenador.

Estas órdenes enviadas desde la interfaz permiten seleccionar el modo de operación del robot móvil, como también manipular los movimientos del mismo, activar los sensores que detectan movimiento, la presencia de un obstáculo y activación de alarma utilizando un buzzer; encender y visualizar la imagen captada por la cámara haciendo uso del software LaBVIEW

### **Proceso de datos**

En este proceso el usuario define el modo de operación del robot móvil haciendo uso del software arduino, este recibe la información de forma remota, el microcontrolador de la tarjeta arduino procesa el comando y actúa en uno de los modos de uso; sea este como modo manual cumpliendo de manera remota las órdenes del usuario o modo automático ejecutando el recorrido del robot de monitoreo y vigilancia autónomo.

### **Ejecutar recorrido modo manual**

El usuario se encargará de manejar el robot móvil de vigilancia y seguridad de manera remota, haciendo uso del software arduino y el módulo de comunicación xbee, en el monitor serial del software arduino se designa las letras que se asignaron en la codificación para esto se hace uso del teclado del ordenador teniendo como opción avanza (u), gira a la derecha (l), giro a la izquierda (g), detener (p) y hacia atrás(n), movimiento de servomotor (s), este tiene como interfaz el software LaBVIEW para visualizar la imagen, mientras que el sensor ultrasónico está midiendo la distancia, en este modo está disponible la opción automático (a) haciendo que el prototipo se ejecute cumpliendo el monitoreo en el área.

### **Ejecutar recorrido modo automático**

Cuando el usuario selecciona este modo de operación automático, el robot móvil de vigilancia y seguridad cumple con el recorrido de guardianía proyectada en la codificación, ejecutando los tres casos como avanzar, gira a la derecha y nuevamente avanzar, para la detección de obstáculo o movimiento el sensor ultrasónico y el sensor PIR se accionaran, y la alarma del buzzer se activará pasando a encender la cámara para captura de imagen lo que hará que esté en

movimiento el servomotor, estos permiten informar las condiciones del entorno, caso contrario si no detecta obstáculo o movimiento se ejecuta los tres casos mencionados anteriormente cumpliendo su ciclo, luego pasa al modo manual.

### **Monitorear entorno**

Este proceso se realizará cuando el robot móvil de seguridad y vigilancia se encuentre operando el recorrido en modo automático que es asignada en la codificación, cumpliendo la ronda de guardianía en el entorno, empleando los sensores PIR para detectar la presencia de personas en movimiento y el sensor ultrasónico para la presencia de un obstáculo accionándose la alarma y activar la cámara, el prototipo deberá estar detenido para su correcto funcionamiento.

### **3.2 Identificación de requerimientos**

En este capítulo veremos el desarrollo del proyecto basado en los objetivos generales y específicos expuesto en el principio de esta tesis y debe capitularse a los siguientes requerimientos para su total funcionamiento.

Para el control del robot móvil de vigilancia y seguridad su desarrollo está basado en el software Arduino, Quartus II y NI LabVIEW.

El programa basado en hardware y software arduino será capaz de controlar los motores DC del sistema del prototipo creando movimientos como avanzar, ir a la derecha, izquierda, detener e ir hacia atrás.

El robot móvil servirá como instrumento para las prácticas de Laboratorio de la escuela de Electrónica de la carrera de Electrónica y Telecomunicaciones de la Universidad Estatal Península de Santa Elena

Este prototipo deberá recorrer la trayectoria programada, censando su entorno para cumplir el recorrido eficientemente en una superficie plana libre de obstáculo y sin personas en movimiento, excepto en casos de detección, se debe contener un algoritmo que basado en las lecturas de los sensores permita reconocer estas funciones, deberá encenderse la cámara desde el software LabVIEW para que el usuario pueda captura la imagen.

### 3.3 Análisis del sistema

En esta sección explicaremos y mencionaremos la viabilidad del proyecto, los implementos a utilizarse así como el análisis económico.

#### 3.3.1 Análisis Técnico

Luego de un estudio realizado para la implementación del prototipo de robot móvil se estableció que los recursos necesarios para el desarrollo y ejecución son los siguientes:

- Hardware para el desarrollo
- Software arduino, Quartus II, LaBVIEW y X-CTU para el desarrollo de la programación
- Hardware para la implementación del prototipo

RECURSO TECNOLÓGICO	DESCRIPCIÓN	CANTIDAD
HARDWARE	Computadora portátil procesador Intel (R) Core (TM) i7-4700MQ CPU@ 2.40 GHz , 12.GB RAM	1
	Impresora multifunción	1
	Disco externo	1

Tabla 6. Hardware para el desarrollo

RECURSO TECNOLÓGICO	DESCRIPCIÓN	CANTIDAD
SOFTWARE	Sistema operativo Windows 8 Pro.	1
	Microsoft Office 2013 Profesional	1
	Software arduino versión 1.0.5	1
	NI LabVIEW 2010	1
	QUARTUS II Versión 10.1	1
	Software de configuración X-CTU Versión 5.2.8.6	1

Tabla 7. Software para el desarrollo

RECURSOS TECNOLÓGICOS	ELEMENTOS /MATERIALES	CANTIDAD
Estructura de prototipo	Estructura mecánica	1
	Placa acrílico	1
	Aluminio	1
	Abrazaderas y ángulos	10
	Pernos y tornillos	42
	Imanes	12
	Resina (pegamento)	1
	Kit de llantas	1

Componentes electrónicos	Motor DC 12V/ 500 mA	4
	Tarjeta Puente H L298N	1
	Tarjeta arduino Mega 2560	1
	Tarjeta reguladora de voltaje	1
	Sensor ultrasónico SFR04	1
	Sensor PIR DYP-ME003	1
	Kit Cámara inalámbrica 9V	1
	servomotor	1
	Adaptador USB-TTL para Xbee	2
	Módulos de comunicación Xbee S1	2
	Display LCD 16X2	1
	buzzer	1
	Diodo LEDS	8
	swiches pequeño 2 posiciones on/off	4
	Tubo de suelda	1
	Terminales	32
	Borneras plásticas	18
	Cable negro, rojo #18	3 metros
	Cable arduino M-H Y H-M	42
	Capacitor cerámico 103	1
	Capacitor electrolítico 100uf /25V	2
	Capacitor electrolítico 1000uf /25V 1n4007	2
	Resistencias de 1k/220	8
	Fusible de vidrio de 3A	1
	Diodo rectificador 1n4007	1
	Regulador L7805 cV/L7809 cV	4
	baquelita	2
	Broche para batería de 9V	2
	Base para pilas de 1.5V	1
	Pilas "AAA"	3
Terminales de cobre	20	

Tabla 8. Requerimiento de hardware para el prototipo

### 3.3.2 Análisis económico.

De acuerdo al estudio realizado y con previo análisis económico se determinó para este proyecto que era necesario contar con recursos indispensables que deben ser considerados para el desarrollo del prototipo.

Se especificará el costo que llevará el desarrollo del procedimiento además de los suministros, con todo lo necesario a utilizar para desarrollar el programa que servirá en el prototipo de robot móvil de vigilancia y seguridad con visión nocturna desmontable, los cuales se especifican más adelante con su respectiva tabla de descripción de recurso y costo por unidad y total de acuerdo a lo relacionado con el proyecto.

DESCRIPCIÓN	CANTIDAD	VALOR	SUBTOTAL
Sistema operativo Windows 8 Pro.	1	\$ 100.00	\$ 100.00
Microsoft Office 2013 Profesional	1	\$ 52.00	\$ 52.00
Software arduino versión 1.0.5	1	\$ 0	\$ 0
NI LabVIEW 2010	1	\$ 0	\$ 0
QUARTUS II Versión 10.1	1	\$ 0	\$ 0
Software de configuración X-CTU Versión 5.2.8.6	1	\$ 0	\$ 0
Total en gastos de software			\$ 152.00

Tabla 9. Costo de software de desarrollo

DESCRIPCIÓN	CANTIDAD	VALOR	SUBTOTAL
Computadora portátil	1	\$ 650.00	\$ 650.00
Total gasto de hardware			\$ 650.00

Tabla 10. Costo de Hardware de desarrollo

CANTIDAD	DESCRIPCIÓN	VALOR	SUBTOTAL
4	Paquete resma formato A4	\$ 4.50	\$ 18.00
4	Tinta para impresora	\$ 10.00	\$ 40.00
1	Impresora multiuso	\$120.00	\$ 120.00
1	Gastos varios	\$ 100.00	\$ 100.00
Total			\$ 278.00

Tabla 11. Suministros

ELEMENTOS /MATERIALES	CANTIDAD	VALOR UNITARIO	VALOR TOTAL
Estructura mecánica	1	80.00	80.00
Placa acrílico	1	38.00	38.00
Aluminio	1	2.00	2.00
Abrazaderas y ángulos	10	0.26	2.60
Pernos y tornillos	42	0.15	6.30
Iman 30X5 rectangular	12	2.15	25.80
Super bonder	1	2.10	2.10
Resina (pegamento)	1	5.00	5.00
Llantas	4	4.46	17.84
Motor DC 12V/500mA	4	33.04	132.16
Kit Batería LIPO 11.1V/2.2A	1	152.00	152.00

Batería 9V/200mA	1	5.15	5.15
Tarjeta Puente H L298N	1	12.00	12.00
Tarjeta Arduino Mega 2560	1	27.25	27.25
Tarjeta reguladora de voltaje	1	40.00	40.00
Sensor ultrasónico SFR04	1	6.25	6.25
Sensor PIR DYP-ME003	1	6.15	6.15
Kit Cámara inalámbrica 9V	1	64.95	64.95
Servomotor	1	10.00	10.00
Adaptador USB-TTL para Xbee	2	24.00	48.00
Módulos de comunicación Xbee S1	2	35.27	70.54
Display LCD 16X2	1	5.80	5.80
Buzzer activo	1	0.40	0.40
Diodo LEDS	8	0.44	3.52
Tubo de solda	1	1.69	1.69
Terminales	20	0.09	1.80
Borneras plásticas	18	0.50	9.00
Cable negro, rojo #18	3 metros	0.27	0.81
Cable Arduino M-H Y H-M	42	0.15	6.30
Capacitor cerámico 103	1	0.30	0.30
Capacitor electrolítico 100uf /25V	2	0.50	1.00
Capacitor electrolítico 1000uf /25V 1n4007	2	0.50	1.00
Resistencias de 1k/220	8	0.05	0.40
Fusible de vidrio de 3ª	1	0.15	0.15
Switch pequeño on/off	4	0.44	1.76
Diodo rectificador 1n4007	1	0.25	0.25
Regulador L7805 cV/L7809 cV	4	1.20	4.80
Baquelita	2	1.00	2.00
Plug DC-orificio ancho 3.5	2	0.22	0.44
Portapila 2xAAA	2	0.53	1.07
Pilas "AAA" carbón	3	0.44	2.67
Terminales de cobre	32	0.09	2.88
Total			802.13

Tabla 12. Costo requerimiento de hardware para el prototipo

DESCRIPCIÓN	SUBTOTAL
Software	\$ 152.00
Hardware	\$ 650.00
Suministros	\$ 278.00
Costo requerimiento de hardware para el prototipo	\$ 802.13
Total	1882.13

Tabla 13. Costos totales de desarrollo

El costo total de este proyecto es de \$ 1882.13



### **3.3.3 Análisis operativo**

En el análisis operativo se deberá al reflejo de qué manera favorece y el grado de aceptación que tendrá su implementación del robot móvil de vigilancia y seguridad con visión nocturna desmontable. El proyecto es factible operativamente debido a que su proceso de implementación no necesita un alto nivel de conocimientos en base a la tecnología robótica para poder hacer uso del mismo. Pues la facultad de la carrera de Electrónica y Telecomunicaciones será la que se beneficie con el desarrollo del prototipo para las prácticas de laboratorio de electrónica de la Universidad Estatal Península de Santa Elena

#### **• Rendimiento y Fiabilidad**

Su rendimiento y fiabilidad se basará primordialmente en los beneficios que la Escuela de Electrónica de la carrera de Electrónica y Telecomunicaciones ocupará con la implantación del robot móvil de vigilancia y seguridad en base al uso de idea tecnológica.

#### **• En cuanto a su fiabilidad**

Se da en base al análisis profundo y detallado que se ha realizado sobre el proyecto, complementado con una documentación amplia del sistema en capítulos anteriores. Además que el robot móvil será de fácil uso al estudiante y le ayudará a profundizar sus conocimientos básicos mediante la práctica que se lleve a cabo dentro del laboratorio de la escuela de electrónica.

**• Productividad:** La productividad del robot móvil de vigilancia y seguridad con visión nocturna desmontable será aprobada ya que su objetivo es ayudar a los estudiantes de la carrera de Electrónica y Telecomunicaciones involucrado en las tareas de investigación obteniendo resultados productivos mediante las prácticas que maneja el prototipo.

### **3.4 Métodos e instrumentación de investigación**

Se busca estratégicamente por medio de técnicas y procedimientos, realizar el proyecto de investigación de la manera más sencilla y eficaz, teniendo en claro los procedimientos a ejecutar en cada etapa, y así poder cumplir con los objetivos propuestos que rige esta tesis.

## **Metodología de la investigación**

Para la ejecución y desarrollo del prototipo de robot móvil de vigilancia y seguridad con visión nocturna desmontable, para alcanzar resultados propios extraídos de la realidad.

- **Detectar el problema:**

Mediante la observación, se determinó el problema que se presenta a la hora de realizar las prácticas en el laboratorio, ya que la carrera de electrónica y telecomunicaciones no cuenta con robot móvil desmontable en el momento de recibir las clases impartidas por su docente.

- **Proponer hipótesis:**

Para esta tesis se concluyó una hipótesis general en la que se manejará; Con el sistema de robot móvil de vigilancia y seguridad con visión nocturna desmontable aumentará el grado de conocimiento teórico práctico en los estudiantes acerca del uso de dispositivos electrónicos utilizados en el área robótica

- **Sistematizar solución del problema:**

Para el diseño e implementación del prototipo de robot móvil, se analizó la estructura físico y control del mismo. Se dio estabilidad al sistema de locomoción para ejecutar el recorrido en el área determinada.

Para el sistema de control se diseñó una tarjeta reguladora de voltaje para suministrar el voltaje para el circuito eléctrico del prototipo.

Se determinó además el software con el que opero el sistema para el desarrollo y su funcionamiento

# CAPÍTULO IV

## DISEÑO

### 4.1 Diseño

En el presente capítulo se establece el diseño estructural del prototipo de robot móvil de vigilancia y seguridad con visión nocturna desmontable tomando en cuenta además de las especificaciones detalladas de los componentes del prototipo. La estructura del robot se ajusta a los requerimientos y objetivos del proyecto, puntualizando las características, interacción y colocación de los elementos que se generaron en el desarrollo del prototipo.

En el diagrama del prototipo como se muestra es la figura 16 , describe el manejo remoto por medio del ordenador; esto se ejecuta en modo de operación manual y modo automático, interactuando con la persona encargada del manejo para las prácticas en laboratorio, de manera que el robot cumple con su respectivo funcionamiento siguiendo su trayectoria programada encargándose de monitorear el área y sea de mucha eficacia para los estudiantes universitarios de la carrera de electrónica y telecomunicaciones ya que es una de la razón elemental para el presente proyecto.

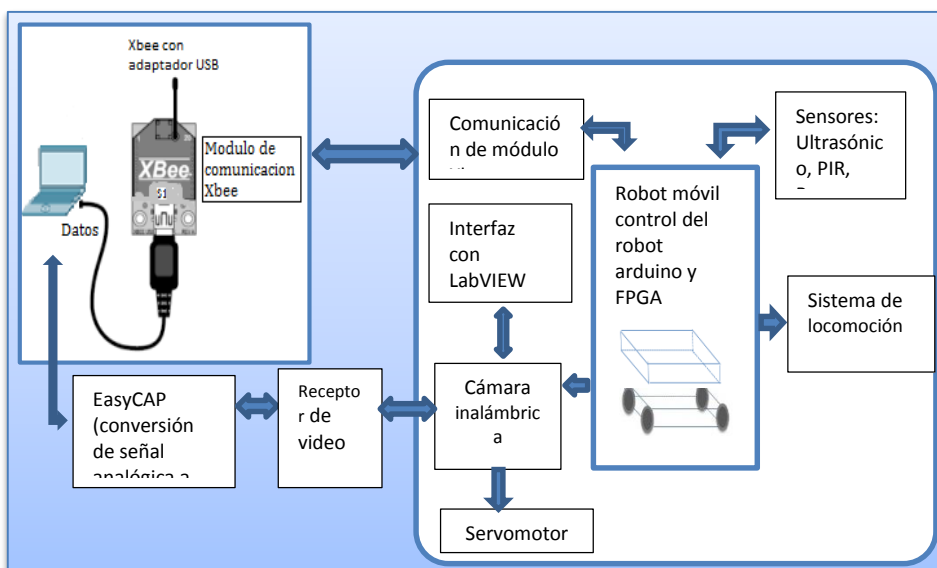


Figura 16. Diagrama del prototipo

## 4.2 Arquitectura de la solución

En esta sección describiremos como fue desarrollado el prototipo para la implementación del robot móvil, estará formado por diferentes componentes, que permitirán obtener los objetivos planteados, a continuación se detallará la estructura, el diseño seleccionado y los componentes del prototipo.

### Estructura del robot.-

Para el proyecto se seleccionó una estructura de material acrílico con dimensiones de 30cm x 21cm y 6.5 cm, donde se acoplaron las cuatro ruedas con sus respectivos motores y el resto de los elementos distribuyendo el peso para el prototipo, en la parte de arriba se hizo una división en dos partes manejables para que se pueda desmontar esto es para tener un robot práctico al momento que se puedan realizar las prácticas que se realizarán en el laboratorio. Se seleccionó este tipo de material para aligerar el peso, y poder cubrir los elementos.

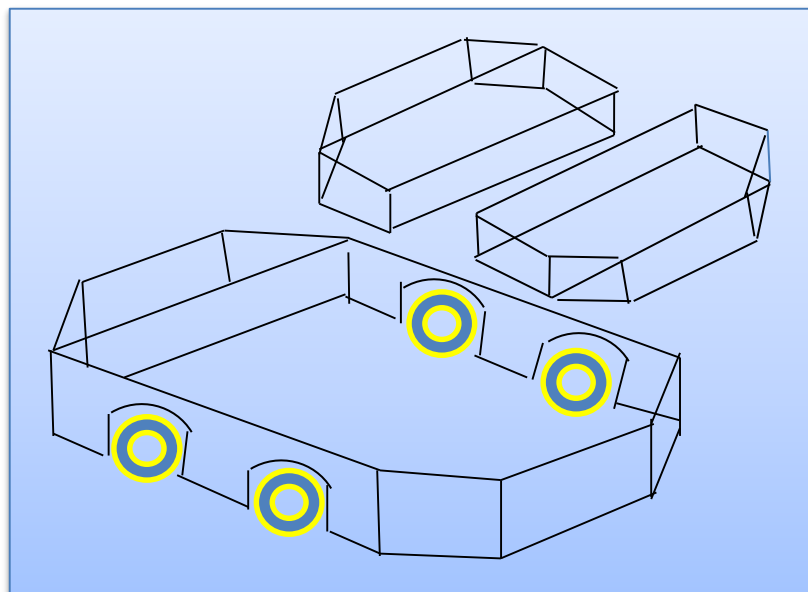
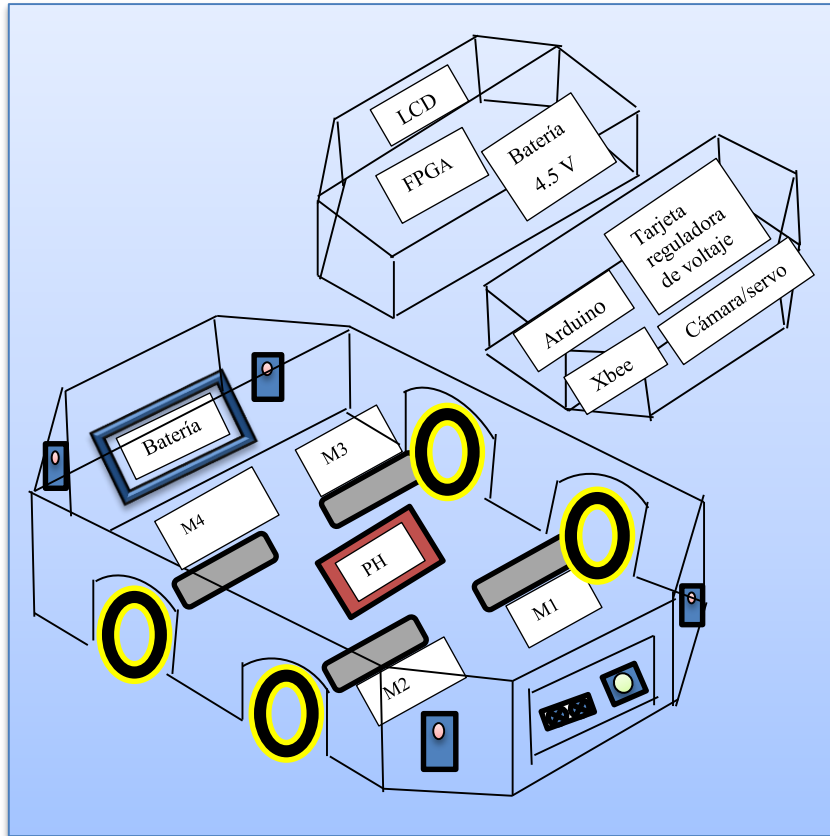


Figura 17. Diseño de la estructura

A continuación en la figura 18, se detalla la ubicación de los elementos electrónicos en el prototipo contenido en la parte de la estructura.



**Figura 18.** Diseño de ubicación de elementos del prototipo

**Módulo de Control del robot.-**

Este es el sistema primordial del robot móvil, ya que contiene la capacidad que va a permitir movilizar al robot; para eso se requiere tener todas las tarjetas y los módulos trabajando correctamente.

Sondea todo lo que obtiene por las entradas de sus sensores y estimula las reacciones necesarias, a través de sus actuadores que le dan movimientos al robot. Para este proyecto se empleó la Tarjeta comercial Mega basada en controladores Atmega2560 de ATMEL, ya que dispone de puertos necesarios para los elementos empleados para el desarrollo del prototipo, La habilidad del código del arduino es de un alto nivel ya que facilita el proceso y control del prototipo.

También se hace uso de la tarjeta FPGA De0-nano establecida en la FPGA de la marca Altera y la familia Cyclone IV, incluyendo en PCB JTAG para programación y depuración, memoria SDRAM, memoria EEPROM, Leds, pulsadores, DipSwitch, y dos puertos de expansión para acceso a pines GPIO de la FGPGA.

## **Sistema de locomoción**

El robot cuenta con un sistema que le permite moverse, de acuerdo a la información que recibe, permitiendo que el robot llegue a cumplir con su funcionamiento para el cuál fue diseñado.

Para implementar un prototipo se debe tener en cuenta las características y especificaciones de cada elemento a utilizar, al momento de elegir el motor se tomó medidas como el precio, torque, velocidad, consumo, tamaño. Es muy importante saber con qué tipo de motor se esté trabajando, porque de ellos depende la electrónica necesaria, la autonomía y las baterías a usar.

El prototipo suministrará para su sistema de locomoción cuatro motores DC con sus respectivas ruedas para el desplazamiento del robot proporcione el torque necesario, para que se movilice sin mucho esfuerzo ya que todo el peso del robot está distribuido sobre las cuatro ruedas; dichos motores tienen un torque de 1.1 Kg/cm, debido al tamaño y peso del prototipo que es de que es 2 Kg (peso repartido para cada motor).

Para el control del sistema del robot móvil de vigilancia y seguridad funcione con exactitud, hemos utilizado cuatro motores DC acoplados en paralelo, por lo que los motores M1P y M1D se controlan como si fuera un solo motor con las entradas del puente H (6 y 7) y los motores M2D y M2P con las entradas de puente H (4 y 5) , para que el robot avance alimentamos los motores M1P y M2P poniendo en alto los Led's LED\_FD y LED\_FI, y estando como bajo los Led's LED\_PD y LED\_PI; para que gire a la derecha deberá estar M1P estar en alto(255) y el resto deberían estar en bajo (0); y el encendido del led debe ser LED\_FD alto los demás debe estar en bajo ; para el giro a la izquierda M2P está en alto(255) el resto en bajo (0), el encendido de los Led's LED\_FI en alto los demás en bajo; para la detención del robot sus motores y led's deben estar en bajo.

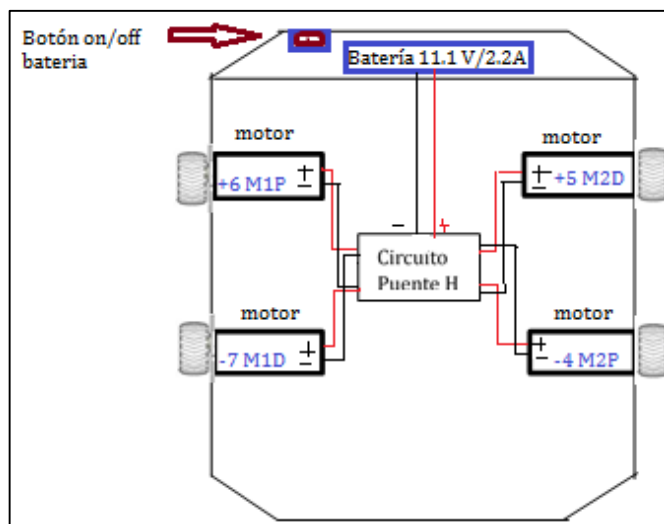


Figura 19. Diseño del sistema de locomoción

**Sistema de video.-** El prototipo de robot móvil de vigilancia y seguridad dispone de una cámara inalámbrica con visión nocturna, para poder cumplir con el monitoreo, esta se ubicó en la parte superior del prototipo, para proveer la visibilidad del entorno al usuario, cuenta con un receptor inalámbrico audio y video de 2.4Ghz con un voltaje de 9V, utilizando un convertidor de señal analógico a digital que se conecta a la PC por medio de una tarjeta reguladora de video: tiene un alcance de transmisión de 30 a 50 metros sus dimensiones son de 25 x 35x 15 mm

**Módulo de comunicación.-** Para la comunicación entre el ordenador y el prototipo se seleccionó el módulo de comunicación Xbee S1 tipo de antena chip, que es un componente que trabaja sin inconvenientes tiene una potencia de transmisión de 1mW su alcance es de 30m para el ajuste a la cobertura de la cámara que también trabaja a 30 metros, con estos parámetros se logra efectuar con la propuesta de monitoreo del área.

Se configura los módulos Xbee desde la PC usando el programa X-CTU, configurar la velocidad y podremos comenzar la comunicación. Importante recordar conectar los RX y TX cruzados con el Arduino

El módulo Xbee que es conectado a la tarjeta Arduino Mega es usado con el adaptador Xbee Explorer Serial (TTL), ya que proporciona la conexión con la tarjeta arduino, mientras que el módulo Xbee para la computadora utiliza la tarjeta

Xbee Explorer (USB). Esta tarjeta aparte de conectar la computadora con el Xbee, también permite configurar simplemente los módulos Xbee.

Los Xbee S1 deberán tener el mismo firmware para que puedan comunicarse, además de la velocidad, canal y red. Configuramos los módulos con el programa XCT-U que lo podemos descargar en la página oficial de los módulos Xbee.

Primero actualizamos la versión de firmware de los módulos desde el X-CTU.

Configuramos los módulos Xbee S1 considerando los siguientes parámetros:

Firmware: 10EC

Velocidad: 9600 baudios

Canal: 0 (el mismo canal)

PAN ID: 888 (la misma red)

Para el emisor. DH: 0, DL: 1, MY: 0

Para el receptor: DH: 0, DL: 0, MY: 1

**Sensores.-** Se seleccionó para el proyecto sensor ultrasónico SFR04, permite detectar obstáculo con un ángulo de 15°; sensor PIR DYP-ME003, permite detectar personas en movimiento en un área de cobertura de 7 metros; sensor buzzer dispositivo de señalización acústica; con estos elementos se podría resolver el monitoreo del área del prototipo.

**Sensor ultrasónico HC-SR04.-** Es un sensor de proximidad que usa un sonar para la detección de obstáculos ya que envía la onda sonora a un ángulo de 15° hacia delante y por el otro lado recibe de nuevo la onda sonora que vuelve tras haber chocado con el obstáculo (pared), tiene un alcance de 2 cm a 400cm en cuanto su funcionamiento no se ve afectado a la luz, pero puede tener inconveniente con algunos materiales corrosivo, para este proyecto detecta obstáculo mayor que 15 cm.

**Sensor PIR DYP-ME003.-** Es un sensor de detección de movimientos, posee tres terminales, el pin Vcc esta alimentado a 5V, pin negativo (GND), pin 3 out (señal) conectadas al arduino para este proyecto detecta personas en movimiento en un alcance de hasta 3metros.



**Buzzer.**-Un dispositivo de señalización que hace un zumbido. Para este proyecto se hace uso del buzzer ya que enciende la alarma cuando el sensor PIR detecta un movimiento en el área determinada.

**Sistema de alimentación.**-Para la elección de la batería se estudió la operación de todo el sistema del prototipo del robot móvil de vigilancia y seguridad con visión nocturna desmontable, se considera como ciclo normal de operación a las acciones como recorrer ruta, sensando entorno y visualización de cámara. Cada acción tiene que efectuarse en un intervalo de tiempo, y dependiendo de la operación a realizar actúan ciertos actuadores con su respectivo consumo de corriente.

ELEMENTOS	CANTIDAD	CONSUMO	RECORRER RUTA	SENSAR ENTORNO	VISUALIZAR ENTORNO
Motores	4	500 mA	2000mA	-----	-----
Puente H	1	50 mA	50mA	-----	-----
Cámara	1	200 mA	-----	-----	200mA
Xbee	2	50 mA	100mA	100mA	100mA
Arduino	1	50 mA	50 mA	50 mA	50 mA
FPGA	1	25 mA	-----	-----	25 mA
LCD	1	25 mA	-----	-----	25mA
Sensor PIR	1	20mA	-----	20mA	----
Ultrasónico	1	15mA	15mA	----	----
servomotor	1	100mA	----	----	100mA
buzzer	1	15mA	15 mA	----	-----
Corriente total			2230 mA	170	500 mA

Tabla 14. Consumo de energía de los elementos

Se optó por utilizar una batería de 11.1 V/2.2 A y una de 9 V/200mA, ya que el requerimiento del sistema así lo amerita. El sistema del circuito de control del robot móvil de vigilancia y seguridad desmontable, funciona con un voltaje de 9V, lo que suministra una batería de 9V, el resto de los elementos son alimentados a la batería de 11.1 V, mediante el regulador de voltaje 7805 se consigue adquirir un Voltaje de 5V utilizado en los sensores, y con el 7809 adquiere un voltaje de 9V para la cámara.

#### 4.2.1 Diagrama de algoritmo del funcionamiento del robot

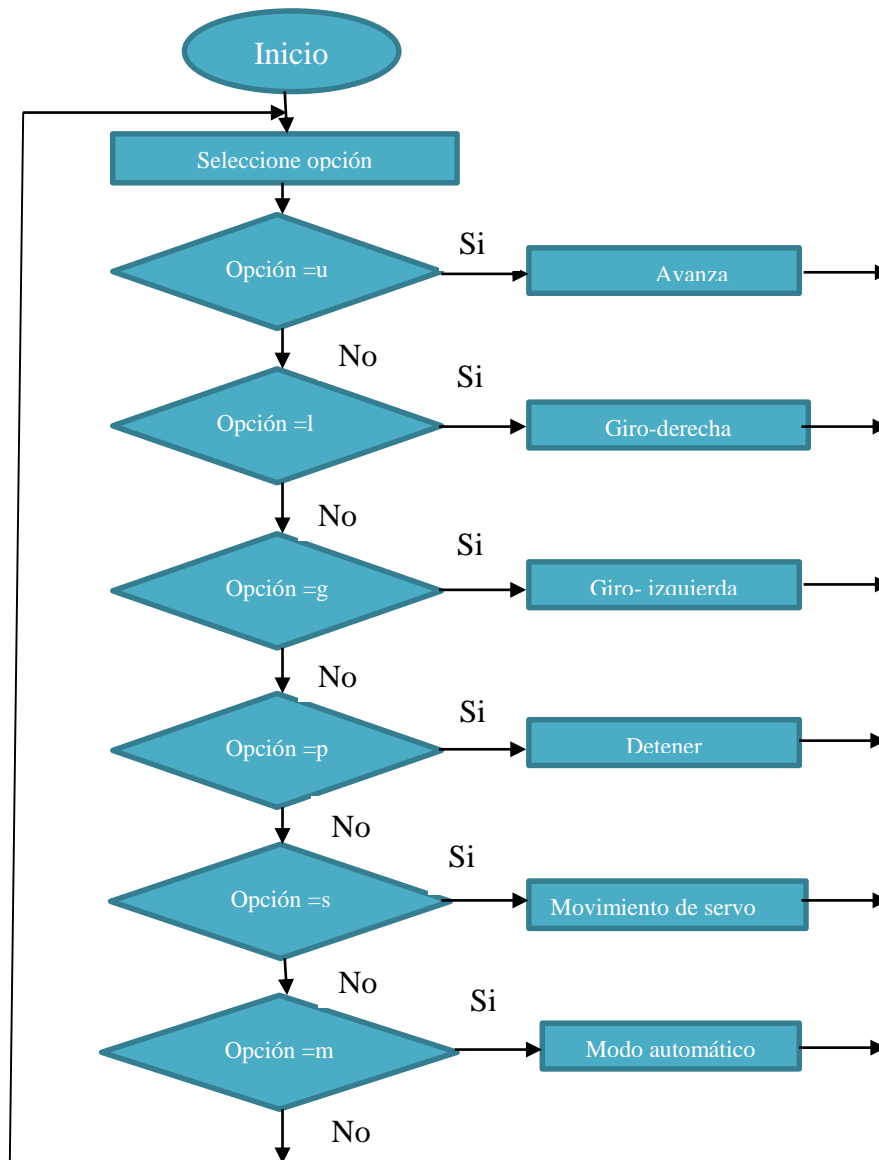
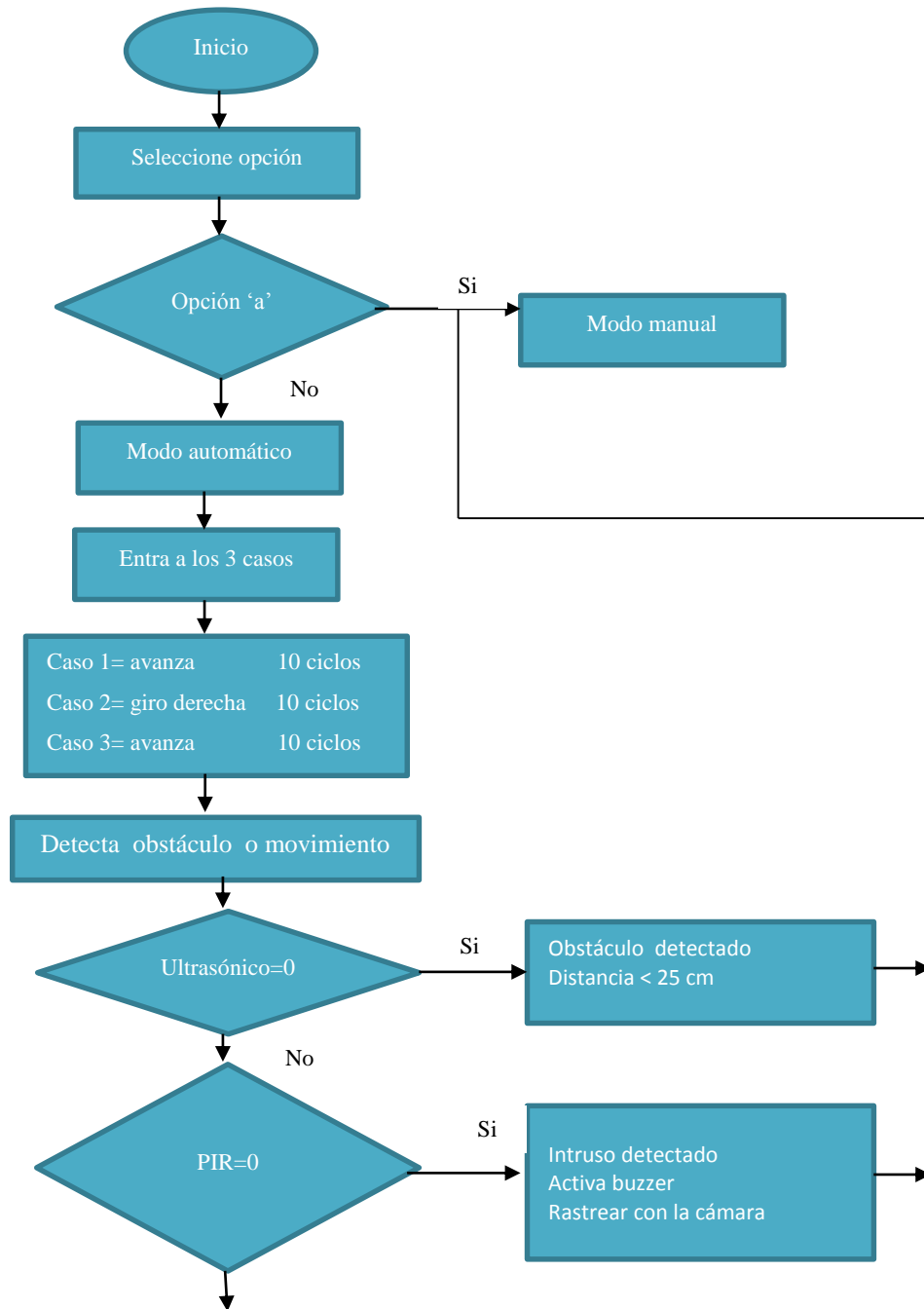


Figura 20. Diagrama de actividades de modo manual.

El diagrama de flujo que explica a continuación:

El recorrido del robot móvil de seguridad y vigilancia en modo de operación manual está trabajando mediante conexión remota con módulo de comunicación xbee entre el ordenador y el prototipo, cuando se elige una opción manda a llamar su función en este caso si elegimos la opción ‘u’ nos indica que el robot avanza, ‘l’ gira a la derecha, ‘g’ gira a la izquierda, ‘n’ hacia atrás, s movimiento de servo, y la opción ‘a’ nos indica el modo automático.



No  
Figura 21. Diagrama de actividades de modo automático

Diagrama de flujo de modo automático se explica a continuación:

Al seleccionar la opción 'a' en el teclado del ordenador se inicia el modo de operación automático, ejecutando los 3 casos desarrollados en la codificación que son los siguientes, caso 1=avanza, caso 2= giro derecha, caso 3 =avanza

nuevamente, los cuales tienen un ciclo de 10 segundos respectivamente, si dentro del desarrollo de cualquiera de los tres casos el robot detecta un obstáculo a una distancia  $< 25$  cm automáticamente se detendrá, ejemplo, si el robot ha iniciado su operación en modo automático y detecta un obstáculo en los primeros 5 ciclos el robot se detendrá he iniciara el rastreo con la ayuda de la cámara, al terminar el rastreo el robot continuara con los 5 ciclos restante del caso uno y los 10 ciclos del caso 2 y 3 terminados los ciclos el robot pasa a modo manual, adicionalmente si al estar en modo automático el robot no detecta ningún obstáculo o movimiento los ciclos de los casos antes mencionados se repiten constantemente.

Si el robot móvil detecta un movimiento en cualquier línea de los 3 casos, el robot determina la distancia  $< 25$  cm con el sensor ultrasónico y el sensor PIR se active determinando un movimiento encendiéndose la alama esto hará que el robot se detenga y ponga en movimiento el servo para que la cámara capture la imagen, si no detecta ningún movimiento su ciclo terminara y pasara a modo manual.

### **4.3 Diseño de interfaz gráfica**

El diseño de una interfaz es una de las acciones más importantes en el desarrollo de aplicaciones en la actualidad, debido a que representan la primera impresión de una aplicación para los usuarios. Además, una interfaz bien diseñada y estructurada permite mejorar la visualización del contenido o de los servicios que proporciona la aplicación. Para este proyecto se aplicó el lenguaje de la plataforma de arduino, ya que dispone todas las funciones en un solo programa, contiene un conjunto de herramientas para aplicaciones de control, en este caso lo utilizaremos como interfaz gráfica el monitor serial, monitorea las operaciones de modo manual y modo automático entre la comunicación de la plataforma robótica y el lenguaje de programación.

#### **Monitor Serial**

En la parte superior izquierda de la ventana principal de software arduino, se encuentra la barra de menú: archivos, editar, sketch, herramientas y de ayuda; en las barra de herramientas se podrán elegir el tipo de tarjeta a utilizar, como también la asignación del número del puerto COM serial. En el área de mensajes se muestra

cuando la compilación de la codificación ha finalizado, así mismo en el área de la consola se mostrará una línea de mensajes indicando la correcta compilación o pueda presentar los errores. En el área de botones de acceso se encuentra un icono de una lupa que hace referencia al acceso al monitor serial. Para ejecutar el recorrido del robot móvil se seleccionan las siguientes opciones presentando en el monitor serial, las siguientes operaciones que trabaja en su recorrido sea este como modo manual o como modo automático.

Para poder cargar el código de control en la tarjeta arduino se debe compilar con éxito es decir sin ningún error en su programación, luego se suministrará de energía al robot para cargar la programación, se procede abrir la ventana del monitor serial para ejecutar el recorrido y monitoreo del robot móvil. Cuando se abre esta ventana se visualiza el modo de operación manual y luego designamos la opción para el modo automático que hará el recorrido el robot móvil de vigilancia y seguridad comandando seis posiciones para el modo manual, en el de automático el robot tiene como comando una sola letra es este caso la 'a'.

# CAPÍTULO V

## IMPLEMENTACIÓN

### 5.1 Implementación

En este capítulo se describe el proceso que se realiza para la implementación del prototipo móvil de vigilancia y seguridad con visión nocturna desmontable, que consiste en realizar diferentes pruebas para determinar el correcto funcionamiento de cada sistema en conjunto con el robot, que está compuesto por los siguientes:

- Sistema de control.
- Sistema sensorial.
- Sistema de comunicación.
- Sistema actuador.
- Sistema de alimentación.
- Circuito electrónico

Comprobando que las relaciones entre sistema y proceso se efectúen de acuerdo a lo establecido.

### 5.2 Construcción de la carcasa del prototipo

Para esta sección describimos en bloques como se construyó el prototipo del robot móvil de vigilancia y seguridad con visión nocturna desmontable, y cada módulo electrónico del mismo.

En la figura 22 muestra el modelo que se implementó para la parte de la estructura del robot, a continuación se detalla cada uno de los procesos de implementación.

Este diseño se hizo con dos bases en forma octagonal de 30cm largo x 21cm ancho y 6.5 cm altura, el material es de acrílico transparente, práctico de moldear y fácil

de cortar, esta estructura está dividida en tres partes desmontables para la colocación de los elementos al prototipo.



Figura 22 .Construcción de la carcasa del prototipo hecho en material acrílico transparente

En esta etapa se presenta el acople de los cuatro motores DC de 12/500mA y una potencia de 1,25W, encajados a sus respectivas llantas que tienen 6.5 de diámetro y 3 de ancho, brindando la tracción necesaria para ejecutar el recorrido, realizando las conexiones respectivas al puente H L298N como se muestra en la figura 23.

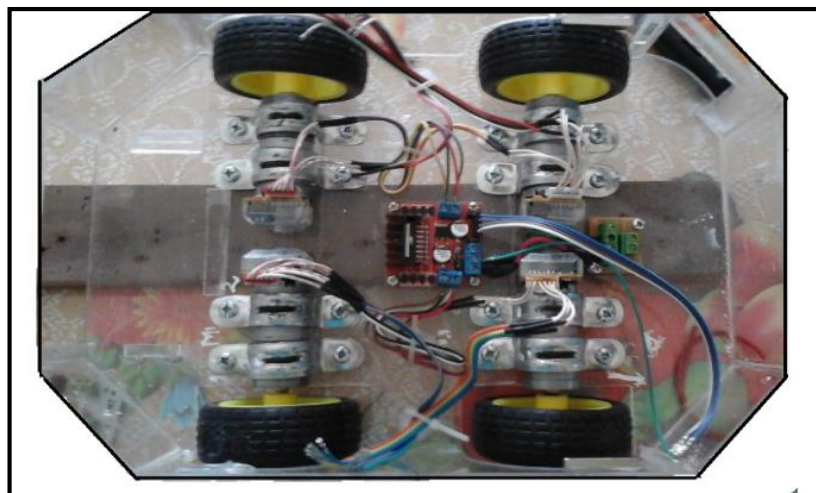


Figura 23. Ensamblaje de los cuatro motores DC a las llantas y conexiones al Puente H.

Se procedió a instalar el resto de los elementos usando sensores ultrasónico y PIR ubicados en la parte baja delantera del prototipo a una distancia de 4.5cm de alto y

1 cm de separación entre cada sensor, en la parte lateral del robot se colocaron los diodos led's tanto en la parte de enfrente como atrás, con una batería LIPO “polímero de litio” recargable de 11.1 V/ 2.2A colocada en la parte trasera con sus respectivas conexiones, para cada componente se encuentra etiquetado como se aprecia en la figura 24.

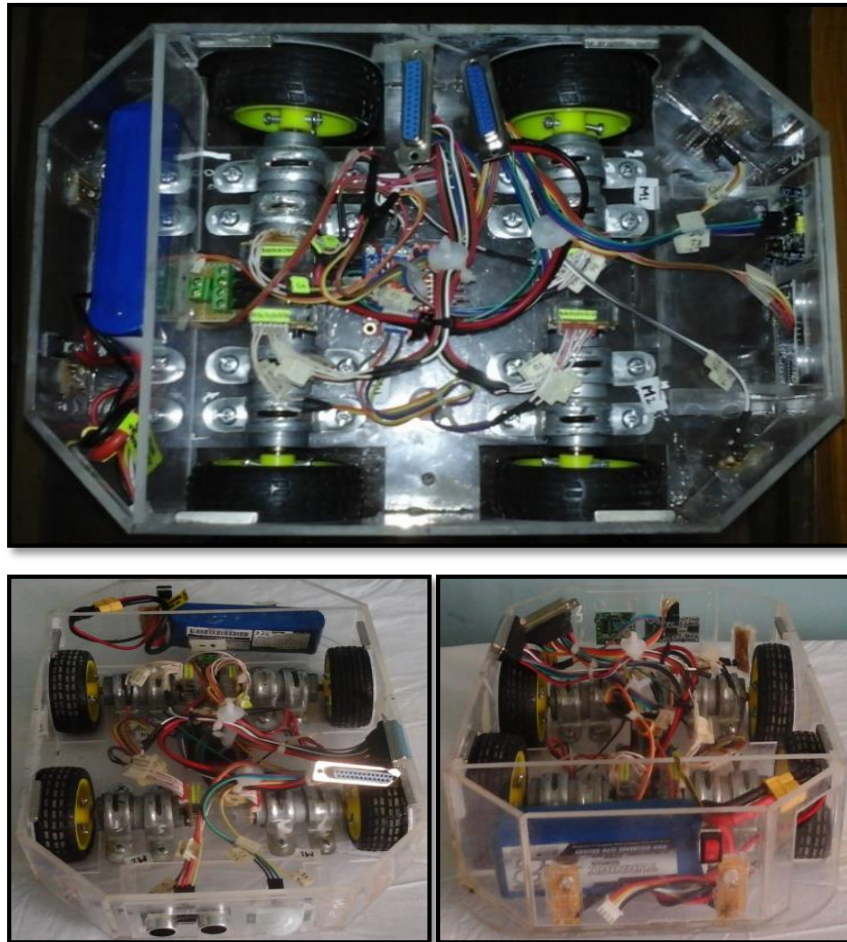


Figura 24. Colocación de los elementos al prototipo móvil con sus conexiones

También se realizó un soporte para la cámara y el servomotor, ubicados en la parte superior del prototipo a una distancia de 10 cm de altura, para que pueda tener un mejor ángulo de captura apropiado, además de los otros elementos que se encuentran en posición como la tarjeta reguladora de voltaje, arduino mega, buzzer, el módulo de comunicación Xbee s1, con sus respectivas conexiones.



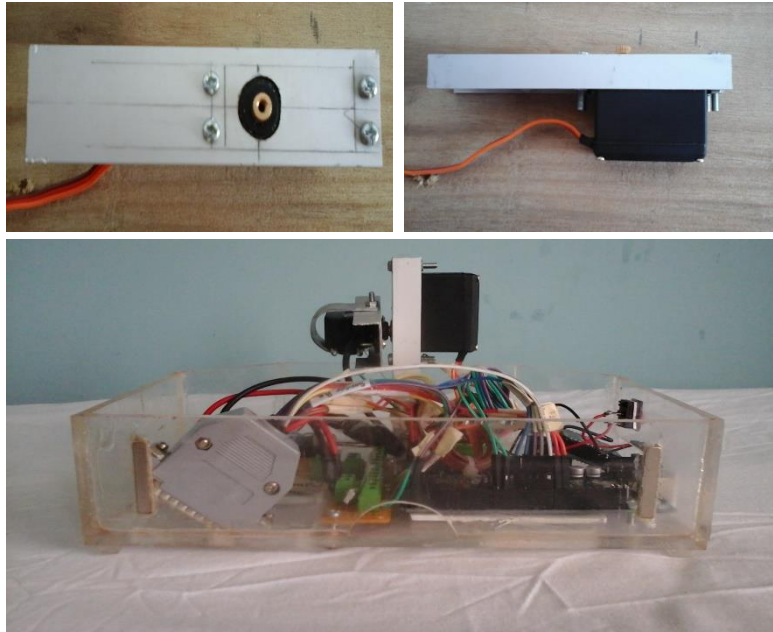


Figura 25. Soporte para la cámara y servomotor, y colocación del resto de elementos

En la otra parte de la estructura dividida se ubicó la tarjeta FPGA De0 nano conectado al display LCD de 16x2, junto a la batería de 4.5 V, dando lugar a cada componente para el ensamblado.



Figura 26. Ubicación de tarjeta FPGA y display LCD 16X2

### 5.2.1 Sistema de control

Para el sistema de control, se usó la tarjeta arduino mega 2560, la cual se encarga de comandar todas las acciones de robot móvil, en la tabla 15 especifica los pines de entrada/salida con la función indicada de cada elemento del prototipo.

Pines del arduino	Funciones
Pin 0	Tx Xbee s1
Pin 1	Rx Xbee s1
Pin 8	Eco (sensor ultrasónico)
Pin 9	Trigger (sensor ultrasónico)
Pin 7	Motor M1D (M3)
Pin 6	Motor M1P (M1)
Pin 5	Motor M2D (M2)
Pin 4	Motor M2P (M4)
Pin 30	Cámara y servo
Pin 3	Sensor PIR
Pin 11	Buzzer
Pin 40	Led posterior derecha
Pin 42	Led posterior izquierda
Pin 44	Led frente derecha
Pin 46	Led frente izquierda
Pin GND	Tierra común
Pin Vcc	+5V

Tabla 15. Asignación de pines para los componentes electrónicos

En el anexo 1 se encuentra el algoritmo de programación realizada en software arduino.

### 5.2.2 Sistema sensorial

Está compuesto por sensor de ultrasonido HC-SRF04, PIR DYP-ME003 y Buzzer activo.

El sensor ultrasónico HC-SRF04 tiene una distancia que va desde los 2 cm hasta los 400 cm, sin embargo no se ve afectado por la luz. Este sensor cuenta con cuatro terminales, para la alimentación Vcc, el de tierra GND, para envío y recepción del pulso ultrasónico (trigger y eco).

En la gráfica siguiente se muestra las conexiones que se hicieron en el sensor ultrasónico a la tarjeta arduino, especificando sus pines designados al mismo.

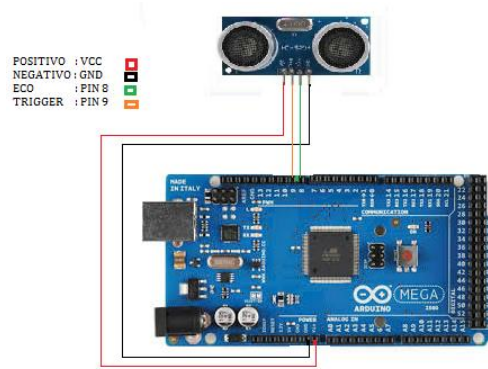


Figura 27. Conexión arduino y sensor HC-SR04

### Sensor PIR DYP-ME003

El sensor piro eléctrico llamado también sensor PIR detecta movimiento de hasta 6m, posee tres terminales que consta de alimentación + 5V, conexión a tierra (GND), y su pin de salida (OUT) como está representada en la figura 28. Para este proyecto se usó el sensor PIR, conectado a la tarjeta arduino a continuación se especifica sus conexiones.

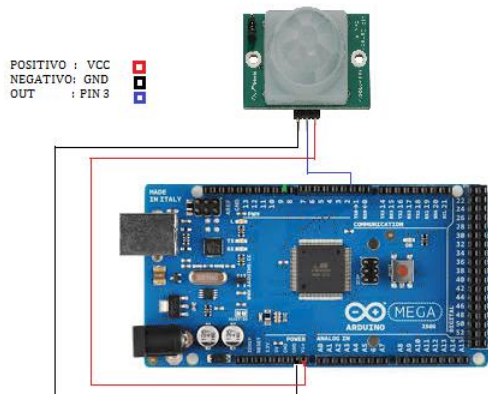


Figura 28. Conexión de sensor PIR DYP-ME003 a la tarjeta arduino

### 5.2.3 Sistema de comunicación.-

En el sistema de comunicación se utilizó el módulo Xbee S1; su potencia de transmisión es de 1 mW, posee una distancia de 30 m para interiores y 100 m en espacios abiertos, suficiente alcance para abarcar el área monitoreada.

Se conectó el RX - TX y TX- RX, del arduino con conexión al adaptador TTL del módulo de comunicación Xbee s1.

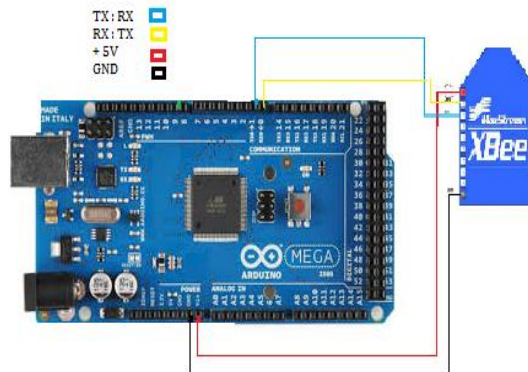


Figura 29. Conexión arduino - Xbee S1

### 5.2.4 Sistema Actuador

Las ruedas del robot son accionadas, cada una, acoplada al eje de un motor DC, suministra un voltaje de 12V/500 mA, los cuatros motores se conectaron a un puente H L298N que permite controlar el sentido de giro de los motores y sus entradas se controlan desde los pines de la tarjeta arduino Mega 2560, en la figura 30 se muestra las respectivas conexiones.

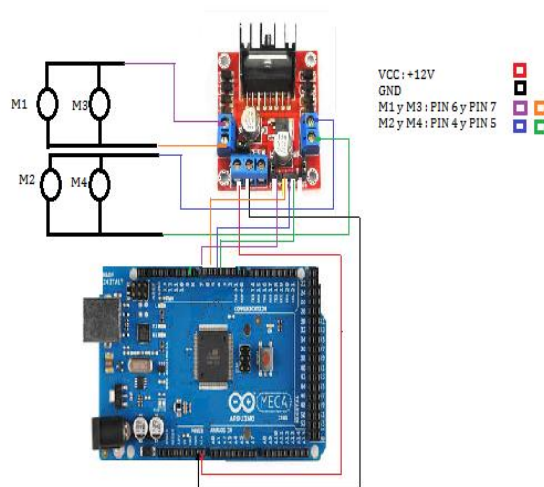


Figura 30. Conexiones arduino y motores

### 5.2.5 Sistema de potencia y alimentación

Para este sistema de potencia y alimentación se requiere que el robot móvil de vigilancia y seguridad desmontable opere sin ninguna dificultad de dos horas para ello se utilizó una batería LIPO (polímero de litio) de 11.1 V/ 2.2A para su conexión y una batería de 9V y 200mA para alimentar el arduino. Se implementó una tarjeta

reguladora de voltaje; utiliza los integrados de regulación de voltaje 7809 para alimentar la cámara y 7805, para alimentar los sensores y tarjeta arduino. Esta tarjeta permite regular una salida de voltaje DC constante respectivamente de 9 y 5 Voltios.



Figura 31. Conexión de circuito impreso de tarjeta reguladora de voltaje

Teniendo listo el prototipo de robot móvil de vigilancia y seguridad con visión nocturna desmontable, se procede a realizar las pruebas para verificar su correcto funcionamiento.

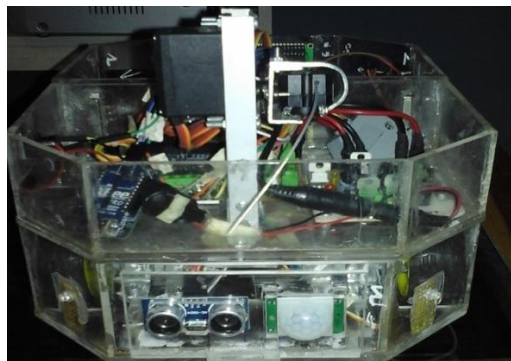


Figura 32. Prototipo terminado

### 5.3 Pruebas

A continuación se presentan los datos obtenidos de las pruebas experimentales de los diferentes procesos realizados con el robot móvil de seguridad y vigilancia y la presentación de resultados como los indicadores de información para la implementación del prototipo. Estas pruebas son necesarias para la verificación del funcionamiento del robot.

### 5.3.1 Prueba y resultados

#### Experimento con sensor ultrasónico

El experimento inicia ubicando el robot móvil en un lugar plano para el recorrido, se utiliza un obstáculo de 30 x 30 cm para que el sensor pueda determinar la distancia de aproximación entre el robot y el obstáculo en un área de 2 x 2 metros, luego se activa el interruptor DC que suministra energía al robot, se carga la información a la tarjeta que es controlado mediante la interfaz de arduino.

En la codificación se especifica que el sensor ultrasónico determina una distancia de  $< 25$  cm mientras que el sensor PIR se encuentre en alto es decir en 1 cuando eso se cumple el robot móvil se detendrá

Finalmente se realizaron 4 pruebas con el robot, operando de modo manual o automático, en modo manual el sensor actuará de modo pasivo es decir el robot realizara el recorrido, sensará la distancia detectará obstáculo pero no se detendrá porque por medio de los comandos se le dará direccionamiento, por otro lado de modo automático el sensor se encuentra activo, calculando la distancia  $< 25$  y  $>15$  cm hará que el robot se detenga.

Como resultado de estas pruebas fueron exitosas ya que el sensor ultrasónico se encuentra trabajando en un 90%.

N° Prueba	Modo de operación del robot	Modo de operación sensor ultrasónico	Distancia en modo de operación	Distancia de aproximación del objeto	Desplazamiento del robot	porcentaje	resultado
1	manual	pasivo	indeterminada	---	avanza	90%	exitoso
2	automático	activo	determinada	$<25$ cm	detiene	90%	exitoso
3	automático	activo	Determinada	$>15$ cm	detiene	90%	exitoso
4	automático	activo	Determinada	$>25$ cm	avanza	90%	exitoso

Tabla 16. Resultado de prueba del sensor ultrasónico

#### Experimento con el sensor PIR

Las pruebas que se realizaron para comprobar el funcionamiento del PIR para detección de personas en movimiento ya que tienen un ángulo de cobertura de  $140^\circ$  estas pruebas fueron realizadas en un área despejada en terreno plano para el

deslizamiento del robot. Se realizaron 5 intentos de cada prueba presentada en la tabla 17.

Para el proyecto este sensor está configurado para un rango que abarca 3m, este está ubicado en la parte de enfrente del prototipo, esto es para que pueda sentir a la persona que se encuentra en total movimiento, cuando se prende hay que esperar 5 segundos para que el sensor se active inmediatamente envía una señal de 5V al arduino.

Prueba	Movimiento de persona	Estado	Sensor PIR
1	Estático	0	Fallido
2	Caminando a paso lento	1	movimiento
3	Caminando a paso normal	1	movimiento
4	Caminando a paso acelerado	1	movimiento
5	Corriendo	1	movimiento

Tabla 17. Resultado de pruebas del sensor PIR

Al realizar las pruebas nos dio a saber que el sensor PIR es capaz de reconocer a la persona si se encuentra en movimiento en un área de cobertura de tres metros para poder ser detectadas al instante.

### 5.3.2 Pruebas y resultados del recorrido autónomo libre de obstáculo

Estas pruebas del robot móvil de vigilancia y seguridad se llevaron a cabo en una superficie plana libre de obstáculo hasta una distancia de 5m. En la siguiente tabla se muestra las pruebas del recorrido del movimiento del robot. Para su resultado de la prueba se consideró 10 intentos

Prueba	Distancia [metros]	Recorrido autónomo	resultados	porcentaje
1	5 metros	Avanza su recorrido	exitoso	90%
2	4.5 metros	Avanza su recorrido	exitoso	90%
3	4 metros	Avanza su recorrido	exitoso	90%
4	3.5 metros	Avanza su recorrido	exitoso	90%
5	3 metros	Avanza su recorrido	exitoso	90%
6	2.5 metros	Avanza su recorrido	exitoso	90%
7	2 metros	Avanza su recorrido	exitoso	90%
8	1.5 metros	Avanza su recorrido	exitoso	90%
9	1 metro	Avanza su recorrido	exitoso	90%
10	< 25 cm	Presenta obstáculo	exitoso	90%

Tabla 18. Prueba recorrido autónomo del robot modo manual

Los resultados de las pruebas del recorrido autónomo libre de obstáculo nos manifestaron que el robot está operando en un 90% capacitado para ejecutar el recorrido de las opciones programadas.

En las siguientes figura se muestra el recorrido del prototipo del robot móvil de seguridad y vigilancia realizando las pruebas del recorrido autónomo libre de obstáculo, como se observa en las imágenes el robot ejecuta su trayectoria de recorrido avanzando desde los 5 metros hasta 1 metros, cuando se presenta un obstáculo <25 entonces el robot se detendrá, esto lo hace en base a su sensor ultrasónico.

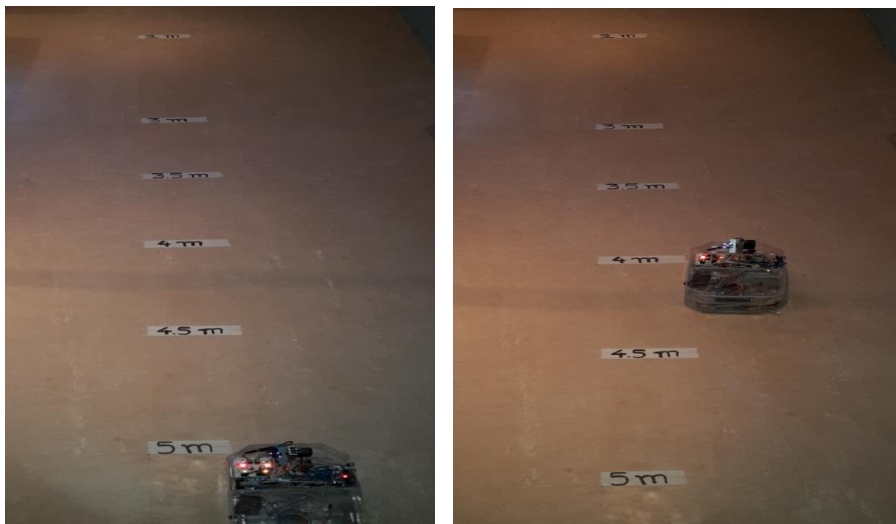


Figura 33. Prueba recorrido autónomo del robot modo manual

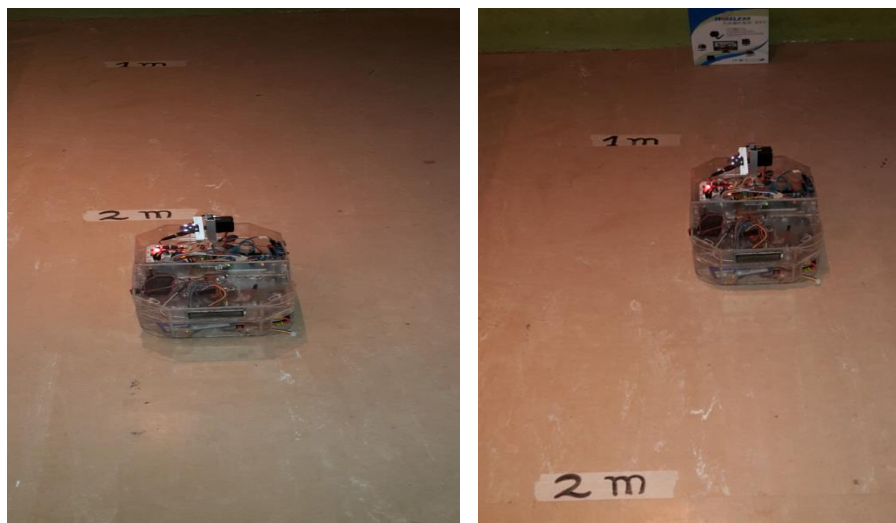


Figura 34. Prueba recorrido autónomo del robot modo manual



## **5.4 Documentación elaboración de manual y guía de usuario**

De acuerdo a todos los requerimientos se realizó el proyecto del robot móvil de vigilancia y seguridad con visión nocturna desmontable, de fácil manejo y operación. Se presenta la demostración de la hipótesis y la guía práctica de usuario para la operación de prácticas en laboratorio de la escuela de electrónica de la carrera de electrónica y telecomunicaciones.

### **5.4.1 Comprobación de hipótesis**

Una vez concluidas las pruebas al prototipo de robot móvil de vigilancia y seguridad con visión nocturna desmontable, se comprobó la hipótesis que se propuso en el proyecto. Por tanto utilizarían este prototipo de robot móvil desmontable; tendrá una eficiencia del 90% tendrá una eficiencia del 90% en la detección del objeto, la distancia a la que se encuentra el objeto y comunicarse de forma inalámbrica con el usuario.

### **5.4.2 Manual de usuario y guía manual de prácticas**

La creación de este manual ayudará a los estudiantes a manipular el prototipo desmontable para el uso de las prácticas en el laboratorio de la escuela de Electrónica y Telecomunicaciones.

Este manual contiene tres prácticas para su desarrollo, para visualizar dirigirse al anexo 6 y 7.

## CONCLUSIONES

- Se construyó el prototipo desmontable con el objetivo que sea didáctico y que al momento de realizar las pruebas sea práctica su manipulación y obtener resultados de los dispositivos que se encuentran montados en él.
- El sistema eléctrico DC se lo elaboró tomando en cuenta los sistemas de control y de fuerza ya que se realizaron pruebas uniendo las conexiones de los dos sistemas y esto ocasionaba errores en el funcionamiento del prototipo.
- En el sistema de locomoción con tracción diferencial, el robot genera desviaciones en el recorrido con un porcentaje de 90%, pero es controlado mediante software cumpliendo la trayectoria programada.
- Se elaboró la guía manual que será la herramienta para el uso de las prácticas en laboratorio de electrónica utilizando el prototipo móvil de vigilancia y seguridad desmontable. En el experimento de la práctica 1 se determina el desplazamiento del robot móvil en un área cuadrada de 40 x 40 cm, se realizó 10 pruebas experimentales obteniendo como promedio de recorrido 39.7 cm con margen de error de 0.1%.  
En el experimento de la práctica 2 de acuerdo a la tabla se determinó que para la detección entre el robot y el obstáculo la distancia es de 30 cm, luego de haber realizado las 10 pruebas el valor promedio de la distancia fue de 30.4 cm sin margen de error.  
En el experimento de la práctica 3 se realizó el recorrido del robot móvil operando en modo manual determinado el alcance de comunicación entre el módulo Xbee y el prototipo ubicado en un área de hasta 40 metros teniendo como resultados de 90% de efectividad.

## RECOMENDACIONES

- Es recomendable que el sensor PIR y el sensor ultrasónico montados en el prototipo trabajen de forma individual para su correcto funcionamiento.
- Verificar que los dispositivos utilizados en el prototipo estén ubicados correctamente, para tener un balance adecuado al construirlo, esto evitará sobretiempos para su ensamblado.
- Es importante verificar que la batería LIPO no esté completamente descargada, observar que su voltaje opere en 11.1V/ 2.2 A, que es el rango de voltaje y corriente con la que trabaja este tipo de batería.
- Es necesario trabajar en un área de 2 x 2 metros para que el prototipo opere en modo automático y cumpla con los tres casos asignados en la programación.
- Cuando se trabaje en modo manual el operador debe tener línea de vista con el robot para que no colisione con un obstáculo.

## BIBLIOGRAFÍA

- [1] N. S. Katty, *Implementación de un prototipo de robot móvil con vision nocturna para vigilancia y seguridad*, Santa Elena: universidad Estatal Peninsula de Santa Elena, 2013.
- [2] G. Bermúdez, *Robots Móviles*, I Semestre 2002.
- [3] Ó. T. Artero, *Arduino Curso práctico de formación*, Libros RC, 2013, pp. 277-349.
- [4] M. M. Mano, *Diseño Digital*, 3era Edición ed., Pearson Educacion, 2003, p. 287.
- [5] S. R. Caprile, *Desarrollo de aplicaciones con comunicacion remota basadas en modulos Zigbee*, 1era Edición ed., Gran Aldea: GAE, 2009, pp. 252-260.
- [6] M. A. P. Garcia, *Instrumentación Electrónica*, España: Paraninfo SA, 2014, pp. 398-399.
- [7] S. G. Cardenas N, *Diseño y construccion de un robot movil didáctico para estudios experimentales de robotica*, Cuenca: Universidad Politecnica Salesiana, 2007.
- [8] P. E. García, M. Hidalgo, J. L. Loza y J. Muñoz, *Prácticas con arduino edubásica*, 4ta Edición ed., 2013, pp. 85-87.
- [9] J. R. L. Vizcaíno y J. P. Sebastia, *Labview Entorno Gráfico de programación*, 2da Edición ed., Barcelona: Marcombo S.A, 2011.
- [10] J. Pandal y P. Peña, *Arduino y Noje.js*, pp. 90-93.

- [11] G. Suárez, *Diseño e implementación de un algoritmo de generación de trayectorias para la evasión de un obstáculo para un robot móvil*, Lima: Universidad Católica del Perú, 2012.
- [12] G. Zabala, *Robotica guía teorica y práctica*, pp. 18-20.
- [13] L. A. Zabala, *Diseño e implementación de prototipo robot publicitario para la escuela de ingeniería Electrónica*, Quito: Escuela Superior técnica de Chimborazo, 2011.

# **ANEXOS**

## ANEXO 1

### "CÓDIGO FUENTE ALGORITMO DE PROGRAMACIÓN EN ARDUINO"

/\*PROGRAMA DE MOVIMIENTOS PARA ROBOT DE VIGILANCIA Y SEGURIDAD CON VISION NOCTURNA

Creado 20 DICIEMBRE DEL 2015  
POR MARIA GABRIELA FRANCO A.  
ELECTRONICA Y TELECOMUNICACIONES

```

                M1      M2
+ 6 M1P #####   ##### +4 M2D
                #####
                #####
                #####
- 7 M1D #####   ##### -5 M2P
                M3      M4

*/
#include <Servo.h>
char val;
int automatico=0;
int EM1 = 0;           //Encoder motor 1 (Izquierda) (Entrada 2)
int EM2 = 1;           //Encoder motor 2 (Derecha) (Entrada 3)
int M2D = 4;           //Motor 2 control de PWM
int M1P = 6;           //Motor 1 control PWM
int M2P = 5;           //Motor 2 control de direccion (derecha)
int M1D = 7;           //Motor 1 control de dirección (Izquierda)
Servo servo1;         //Creamos un objeto Servo de nombre servo1
int pos = 0;           //variable de entrada posicion del servo
long dist = 0;         //variable distancia
byte redireccionado= 0;
int SU = 8 ;           //Sensor de distancia por ultrasonidos eco
int SA = 9;           //Sensor de distancia por ultrasonidos trigger
long duracion = 0;    //Duración del recorrido del ping (microsegundos)
volatile long CRI = 0; //Contador rueda izquierda
volatile long CRD = 0; //Contador rueda derecha
long distizq = 0;     //Distancia del robot (cm) a su izquierda
long distder = 0;    //Distancia del robot (cm) a su derecha
byte giro = 0;        //Variable para saber por donde girar
byte caso = 0;        //Para seleccionar el caso de giro
int y=0;
int contador_detenido=0;
const int tiempo_detenido=10000;
int estado_s=0;
int PIR=3;
```

```

int PR=0;
int buzzer=11;
int camara=30;

int LED_FD=44;           // led frente derecha
int LED_FI=46;           //led frente izquierda
int LED_PD=40;           // led posterior derecha
int LED_PI=42;           // led posterior izquierda

/*****
*****
void setup()              //Se ejecuta cada vez que el Arduino se inicia
{
  Serial.begin(9600);     //abre el puerto serie , establece la velocidad de datos a
9600 bps
  digitalWrite(buzzer,LOW);
  pinMode(PIR,INPUT);    //entrada del pir pin 3
  pinMode(camara,OUTPUT);
  pinMode(buzzer,OUTPUT);
  pinMode (44, OUTPUT);   //salida led frontal
  pinMode (46, OUTPUT);   //salida led frontal
  pinMode (40, OUTPUT);   //salida led posterior
  pinMode (42, OUTPUT);   //salida led posterior
  servo1.attach(30);      //conectamos el servo al pin digital 38
  servo1.write(100);     //Mueve el servo a la posición entrada (excepto si es 0)
}

/*****
*****
void gaby() {

  Serial.println(" manual"); //mostrar por el puerto serial'movimiento servo'
  digitalWrite(camara,HIGH);
  val = Serial.read();
  switch(val)
  {
  case 'u':
    Serial.println("avance"); //mostrar por el puerto serial'avance'
    avanzar();
    break;

  case 'l':
    Serial.println("derecha"); //mostrar por el puerto serial'derecha'
    derecha();
    break;

  case 'g':
    Serial.println("izquierda"); //mostrar por el puerto serial'izquierda'
    izquierda();
    break;

  case 'p':

```



```

Serial.println("detener");          //mostrar por el puerto serial'detener'
detener();
break;

case 'n':
Serial.println("atras");          //mostrar por el puerto serial'atras'
atras();
break;

case 's':
Serial.println("movimiento servo"); //mostrar por el puerto serial'movimiento
dervo'
servo();
break;

case 'a':
Serial.println("automatico");     //mostrar por el puerto serial'movimiento dervo'
automatico=1;
contador_detenido=0;
y=0;
break;

}
}
//*****
*****
*****

void avanzar() {
analogWrite(M1P,255);
analogWrite(M2P,255);
analogWrite(M1D,0);
analogWrite(M2D,0);
digitalWrite(LED_FD,HIGH);
digitalWrite(LED_FI,HIGH);
digitalWrite(LED_PD,LOW);
digitalWrite(LED_PI,LOW);
}

void derecha() {
analogWrite(M1P,255);
analogWrite(M2P,0);
analogWrite(M1D,0);
analogWrite(M2D,0);
digitalWrite(LED_FD,HIGH);
digitalWrite(LED_FI,LOW);
digitalWrite(LED_PD,LOW);
digitalWrite(LED_PI,LOW);
}

void izquierda(){
analogWrite(M1P,0);
analogWrite(M2P,255);
analogWrite(M1D,0);

```

```

    analogWrite(M2D,0);
    digitalWrite(LED_FD,LOW);
    digitalWrite(LED_FI,HIGH);
    digitalWrite(LED_PD,LOW);
    digitalWrite(LED_PI,LOW);

}

void detener(){
    analogWrite(M1P,0);
    analogWrite(M2P,0);
    analogWrite(M1D,0);
    analogWrite(M2D,0);
    digitalWrite(LED_FD,LOW);
    digitalWrite(LED_FI,LOW);
    digitalWrite(LED_PD,LOW);
    digitalWrite(LED_PI,LOW);

}

void atras(){
    analogWrite(M1P,0);
    analogWrite(M2P,0);
    analogWrite(M1D,255);
    analogWrite(M2D,255);
    digitalWrite(LED_FD,LOW);
    digitalWrite(LED_FI,LOW);
    digitalWrite(LED_PD,HIGH);
    digitalWrite(LED_PI,HIGH);
}

void avanzastop(){
    avanzar();
    delay(5000);
    detener();
    delay(1000);
}

//funcion para movimiento de servo de camara
void servo() {
    for(pos = 60; pos <= 120; pos += 1)          //va de 0 grados a 120 grados en paso de un
    grado
    {
        servo1.write(pos);                      //decirle servo para ir a la posición en ' pos '
        variable
        digitalWrite(buzzer,LOW);

        delay(200);
    }
    for(pos = 120; pos >= 60; pos -= 1)          //Va desde 120 grados a 0 grados
    {
        servo1.write(pos);

```

```

    delay(200);                //Espera 15 ms para el servo para alcanzar la
    posición
    }

    servo1.write(100);

}

// funcion para determinar distancia con ultrasonico
int sensorultrasonidos() {    //Función para medir la distancia con el sensor de
ultrasonidos (cm)
    pinMode(SA, OUTPUT);      //Configuramos el sensor de ultrasonidos
    como salida
    digitalWrite(SA, LOW);    //Hacemos ping LOW-HIGH-LOW
    delayMicroseconds(2);
    digitalWrite(SA, HIGH);
    delayMicroseconds(10);
    digitalWrite(SA, LOW);
    delayMicroseconds(20);
    pinMode(SU, INPUT);      //Configuramos el sensor de ultrasonidos
    como entrada
    duracion = pulseIn(SU, HIGH); //Leemos la duración del pulso
    delay(100);
    return duracion / 29 / 2; //Conversión de microsegundos a la distancia
    cm (velocidad del sonido 340m/s o 29 microsegundos por centimetro y son ida y vuelta
    /2)
}

// desicion para los encoder
int comprobarbandas() {
    CRI = 0;
    CRD = 0;
    servo1.write(10);
    delay(500);
    distder = sensorultrasonidos();
    servo1.write(170);
    delay(500);
    distizq = sensorultrasonidos();
    servo1.write(100);
    delay(500);
    redireccionado = 1;
    if (distder > distizq) {
        giro = 1;
    }
    else if(distizq > distder) {
        giro = 2;
    }
    else if(distizq = distder) {
        giro = 3;
    }
    return giro;
}

```

```

}

//*****
*****
*****

void loop() {                                     //Esta funcion se mantiene ejecutando cuando este
energizado el Arduino
    dist = sensorultrasonidos();                 //Llama a la función para saber la distancia
    gaby();                                       // casos de manual
    Serial.print(dist);

    if (automatico==1 && dist>15){               //Si la variable encendido/apagado tiene valor
high y hay distancia suficiente
        Serial.println(dist);
        Serial.println("automatico");
        contador_detenido=0;
        while (contador_detenido<10){
            contador_detenido++;
            dist = sensorultrasonidos();
            PR=digitalRead(PIR);
            digitalWrite(buzzer,LOW);
            Serial.println(PR);
            avanzar();
            Serial.println(dist);
            delay(100);
            //if((dist<12)||(PR==1)){
            if(dist<25&&PR==1){
                contador_detenido=10;
                detener();
                automatico=0;
                digitalWrite(buzzer,HIGH);
                delay(1000);
                digitalWrite(buzzer,LOW);
                digitalWrite(camara,HIGH);
                servo();
                //delay(1000);
                //gaby();
            }
            Serial.println(" automatico_1");
        }

        contador_detenido=0;
        while (contador_detenido<10){
            contador_detenido++;
            dist = sensorultrasonidos();
            PR=digitalRead(PIR);
            digitalWrite(buzzer,LOW);
            Serial.println(PR);
            derecha();

```

```

Serial.println(dist);
delay(100);
if(dist>25&&PR==1){
//if((dist<12)||PR==1){
  contador_detenido=10;
  detener();
  automatico=0;
  digitalWrite(buzzer,HIGH);
  delay(1000);
  digitalWrite(buzzer,LOW);
  servo();
  // gaby();
}
Serial.println(" automatico_2");
}

  contador_detenido=0;
  while (contador_detenido<10){
  contador_detenido++;
  dist = sensorultrasonidos();
  PR=digitalRead(PIR);
  digitalWrite(buzzer,LOW);
  Serial.println(PR);
  avanzar();
  Serial.println(dist);
  delay(100);
  if(dist>25&&PR==1){
  //if((dist<12)||PR==1){
  contador_detenido=10;
  detener();
  automatico=0;
  digitalWrite(buzzer,HIGH);
  delay(1000);
  digitalWrite(buzzer,LOW);
  servo();
  // gaby();
}
  Serial.println(" automatico_3");
}
  detener();
  if(y==1)
  { detener();
  digitalWrite(buzzer,HIGH);
  delay(1000);
  gaby(); //casos manual

  }
}
}
}

```

## ANEXO 2

### CÓDIGO DE VHDL

```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;

entity ADC_LCD is
port(Clk      : IN STD_LOGIC;
     iGO      : IN STD_LOGIC := '0';
     oDIN     : OUT STD_LOGIC;
     oCS_n    : OUT STD_LOGIC;
     oSCLK    : OUT STD_LOGIC;
     iDOUT    : IN STD_LOGIC;
     iCH      : IN STD_LOGIC_VECTOR(2 downto 0);
     LCD_RS   : OUT STD_LOGIC;
     LCD_E    : OUT STD_LOGIC;
     LCD_DataOut : OUT STD_LOGIC_VECTOR(7 downto 0));
end entity;

architecture ADC_LCD_arc of ADC_LCD is

component LCD_Dis is
port( Clk      : IN STD_LOGIC := '0';
     LCD_RS    : OUT STD_LOGIC;
     LCD_E     : OUT STD_LOGIC;
     LCD_DataIn : IN STD_LOGIC_VECTOR(11 downto 0) := "000000000000";
     WaveSel   : IN STD_LOGIC_VECTOR(2 downto 0);
     LCD_DataOut : OUT STD_LOGIC_VECTOR(7 downto 0));
end component;

component ADCModule is
port ( Clk : IN STD_LOGIC;
      iGO : IN STD_LOGIC := '0';
      oDIN : OUT STD_LOGIC;
      oCS_n : OUT STD_LOGIC;
      oSCLK : OUT STD_LOGIC;
      iDOUT : IN STD_LOGIC;
      iCH : IN STD_LOGIC_VECTOR(2 downto 0);
      OutCkt: OUT STD_LOGIC_VECTOR(11 downto 0));
end component;

signal ADC_Data : STD_LOGIC_VECTOR(11 downto 0);

begin

ADC1 : ADCModule port map(Clk, iGO, oDIN, oCS_n, oSCLK, iDOUT, iCH,
ADC_Data);
LCD1 : LCD_Dis port map(Clk, LCD_RS, LCD_E, ADC_Data, iCH, LCD_DataOut);

end architecture ADC_LCD_arc;
```

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;

entity LCD_Dispatch is
port( Clk      : IN STD_LOGIC := '0';
      LCD_RS   : OUT STD_LOGIC;
      LCD_E    : OUT STD_LOGIC;
      LCD_DataIn : IN STD_LOGIC_VECTOR(11 downto 0) := "000000000000";
      WaveSel   : IN STD_LOGIC_VECTOR(2 downto 0);
      LCD_DataOut : OUT STD_LOGIC_VECTOR(7 downto 0));
end entity;

architecture LCD_Dispatch_arc of LCD_Dispatch is

constant N : INTEGER := 21;
Type arrayData is array (0 to N) of STD_LOGIC_VECTOR(7 downto 0);

signal Datas: arrayData;
signal TempVal : INTEGER;
signal TempVal_1 : INTEGER;
signal TempVal_2 : INTEGER;
signal TempVal_3 : INTEGER;
signal TempVal_4 : INTEGER;

begin

--Commands--

Datas(0) <= X"38";
Datas(1) <= X"0c";
Datas(2) <= X"06";
Datas(3) <= X"80";

--Datas--

Datas(4) <= x"44";
Datas(5) <= x"41";
Datas(6) <= x"54";
Datas(7) <= x"41";
Datas(8) <= x"3A";
Datas(9) <= x"20";

TempVal <= (3300 * (to_integer(UNSIGNED(LCD_DataIn)))/4095;
TempVal_1 <= (TempVal) mod 10;
TempVal_2 <= (TempVal/10) mod 10;
TempVal_3 <= (TempVal/100) mod 10;
TempVal_4 <= (TempVal/1000);

with (TempVal_1) select
Datas(14) <= x"30" when 0,
          x"31" when 1,

```

x"32" when 2,  
x"33" when 3,  
x"34" when 4,  
x"35" when 5,  
x"36" when 6,  
x"37" when 7,  
x"38" when 8,  
x"39" when 9,  
x"30" when others;

with (TempVal\_2) select  
Datas(13) <= x"30" when 0,  
x"31" when 1,  
x"32" when 2,  
x"33" when 3,  
x"34" when 4,  
x"35" when 5,  
x"36" when 6,  
x"37" when 7,  
x"38" when 8,  
x"39" when 9,  
x"30" when others;

with (TempVal\_3) select  
Datas(12) <= x"30" when 0,  
x"31" when 1,  
x"32" when 2,  
x"33" when 3,  
x"34" when 4,  
x"35" when 5,  
x"36" when 6,  
x"37" when 7,  
x"38" when 8,  
x"39" when 9,  
x"30" when others;

Datas(11) <= x"2E";

with (TempVal\_4) select  
Datas(10) <= x"30" when 0,  
x"31" when 1,  
x"32" when 2,  
x"33" when 3,  
x"34" when 4,  
x"35" when 5,  
x"36" when 6,  
x"37" when 7,  
x"38" when 8,  
x"39" when 9,  
x"30" when others;

Datas(15) <= x"20";

Datas(16) <= x"56";



```

Datas(17) <= X"C0"; --Move to Line 2--
Datas(18) <= x"41";
Datas(19) <= x"43";
Datas(20) <= x"48";

```

```

with (WaveSel) select
Datas(21) <= x"30" when "000", --0--
    x"31" when "001", --1--
    x"32" when "010", --2--
    x"33" when "011", --3--
    x"34" when "100", --4--
    x"35" when "101", --5--
    x"36" when "110", --6--
    x"37" when "111", --7--
    x"5B" when others;

```

```

LCD_proc: process(Clk)

```

```

    variable i : integer := 0;
    variable j : integer := 0;
    variable k : integer := 0;

```

```

begin
    if (Clk'event and Clk = '1') then
        if(i <= 85000) then i := i + 1; LCD_E <= '1'; LCD_DataOut <= DataS(j)(7 downto
0);
        elsif(i > 85000 and i < 160000) then i := i + 1; lcd_e <= '0';
        elsif(i = 160000) then j := j + 1; i := 0;
        end if;

```

```

    if(j < 4) then LCD_RS <= '0';
    -- Command Signal --
        elsif (j >= 4 and j <= 16) then lcd_rs <= '1';
    -- Data Signal --
        elsif (j = 17) then lcd_rs <= '0';
    -- Command Signal --
        elsif (j > 17 and j < 22) then lcd_rs <= '1';
    -- Data Signal --
        end if;

```

```

    if(j = 22) then j := 0;
    -- Repeat Data Display Routine --
        end if;
    end if;
    end process LCD_proc;

```

```

end LCD_Dispatch;

```

```

-- for ADC128S002 Analog to Digital Converter from Analog Devices

```

```

library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.NUMERIC_STD.ALL;

```

```

entity ADCModule is
port ( Clk : IN STD_LOGIC;
      iGO : IN STD_LOGIC := '0';
      oDIN : OUT STD_LOGIC;
      oCS_n : OUT STD_LOGIC;
      oSCLK : OUT STD_LOGIC;
      iDOUT : IN STD_LOGIC;
      iCH : IN STD_LOGIC_VECTOR(2 downto 0);
      OutCkt: OUT STD_LOGIC_VECTOR(11 downto 0));
end entity;

architecture ADCModule_arc of ADCModule is

component SPICLK is
  -- PLL Module : To be defined externally
port( inclk0 : IN STD_LOGIC := '0'; -- Main Clock = 50Mhz --
      c0 : OUT STD_LOGIC ; -- 0 deg. phase w.r.t main clock (2 MHz) --
      c1 : OUT STD_LOGIC ); -- 180 deg. phase w.r.t main clock (2MHz) --
end component;

signal go_en: STD_LOGIC;
signal cont, m_cont: INTEGER;
signal adc_data: STD_LOGIC_VECTOR(11 downto 0);
signal iCLK, iCLK_n: STD_LOGIC;
begin

--CLKPLL: SPICLK port map (Clk, iCLK, iCLK_n); -- Module requires a 2 MHz PLL -
-
oCS_n <= not go_en;

with go_en select
  oSCLK <= iCLK when '1',
    '1' when '0',
    '1' when others;

Start_ADC_proc: process(iGO)
begin
  if (iGO = '1') then go_en <= '1';
  else go_en <= '0';
  end if;
end process Start_ADC_proc;

counter1_proc: process(iCLK, go_en)
begin
  if(go_en = '0') then cont <= 0;
  elsif (rising_edge(iCLK)) then
    if (cont = 15) then cont <= 0;
    else cont <= cont + 1;
    end if;
  end if;
end process counter1_proc;

```

```

counter2_proc: process(iCLK_n)
begin
    if(rising_edge(iCLK_n)) then m_cont <= cont;
    end if;
end process Counter2_proc;

channel_ADC_proc: process(iCLK_n, go_en)
begin
    if(go_en = '0') then oDIN <= '0';
    elsif(rising_edge(iCLK_n)) then
        if (cont = 1) then oDIN <= iCH(2);
        elsif (cont = 2) then oDIN <= iCH(1);
        elsif (cont = 3) then oDIN <= iCH(0);
        else oDIN <= '0';
        end if;
    end if;
end process channel_ADC_proc;

output_ADC_proc: process(iCLK, iCLK_n, go_en)
begin
    if(go_en = '0') then adc_data <= "000000000000";
    elsif(rising_edge(iCLK)) then
        if (m_cont = 3) then adc_data(11) <= iDOUT;
        elsif (m_cont = 4) then adc_data(10) <= iDOUT;
        elsif (m_cont = 5) then adc_data(9) <= iDOUT;
        elsif (m_cont = 6) then adc_data(8) <= iDOUT;
        elsif (m_cont = 7) then adc_data(7) <= iDOUT;
        elsif (m_cont = 8) then adc_data(6) <= iDOUT;
        elsif (m_cont = 9) then adc_data(5) <= iDOUT;
        elsif (m_cont = 10) then adc_data(4) <= iDOUT;
        elsif (m_cont = 11) then adc_data(3) <= iDOUT;
        elsif (m_cont = 12) then adc_data(2) <= iDOUT;
        elsif (m_cont = 13) then adc_data(1) <= iDOUT;
        elsif (m_cont = 14) then adc_data(0) <= iDOUT;
        elsif (m_cont = 1) then OutCkt <= adc_data;
        end if;
    end if;
end process output_ADC_proc;

end ADCModule_arc;

```

### ANEXO 3

## DIAGRAMA DE CÁMARA INALÁMBRICA EN LABVIEW “VI”

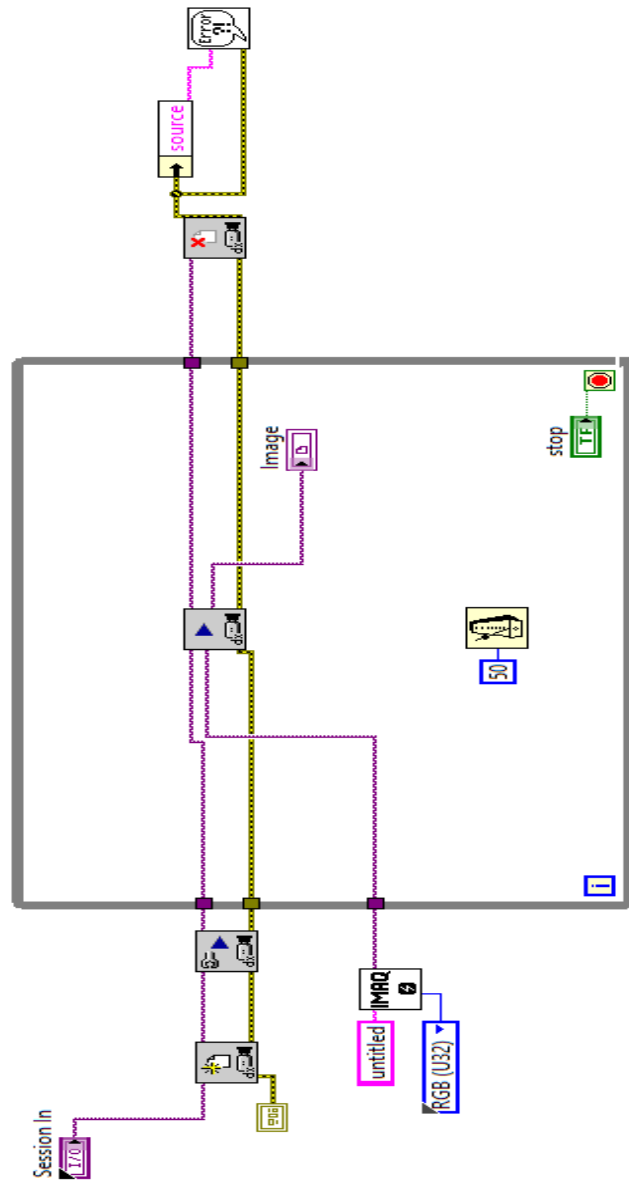


Figura 35. Diagrama de cámara inalámbrica en labVIEW “vi”

## ANEXO 4

### DIAGRAMAS DE CIRCUITO IMPRESOS

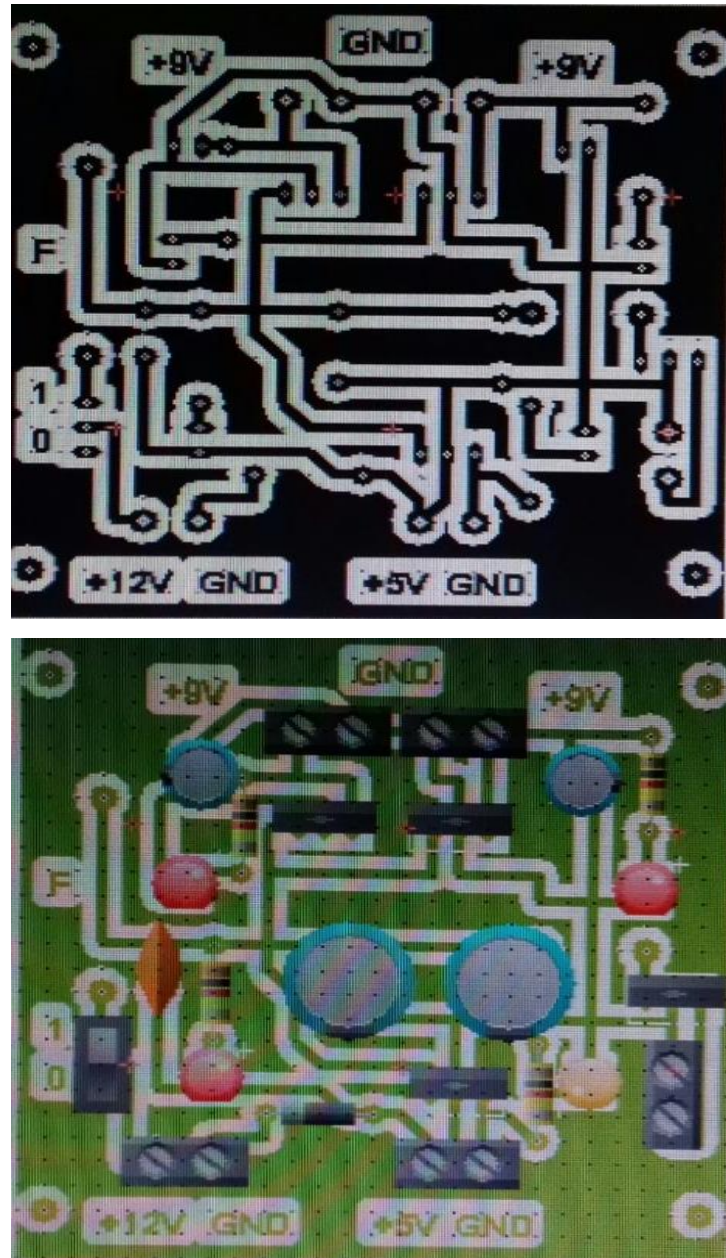


Figura 36. Circuito impreso tarjeta reguladora de voltaje

### Circuito impreso tarjeta Puente H L298N

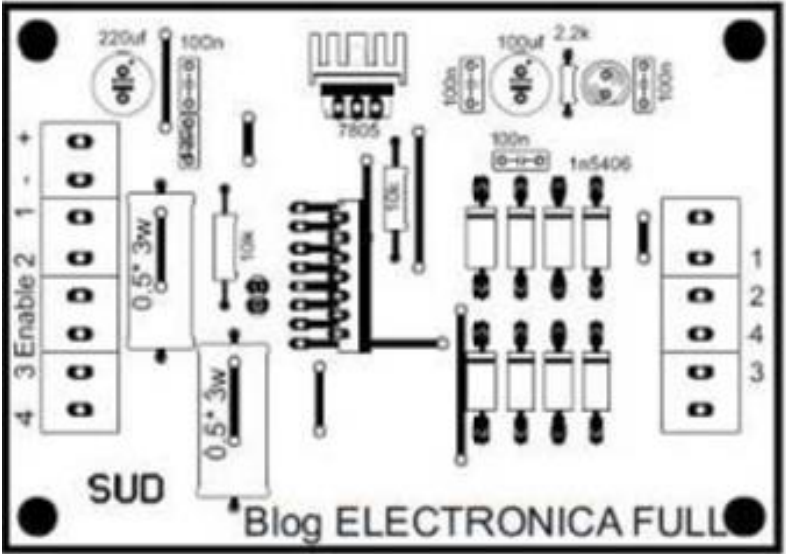
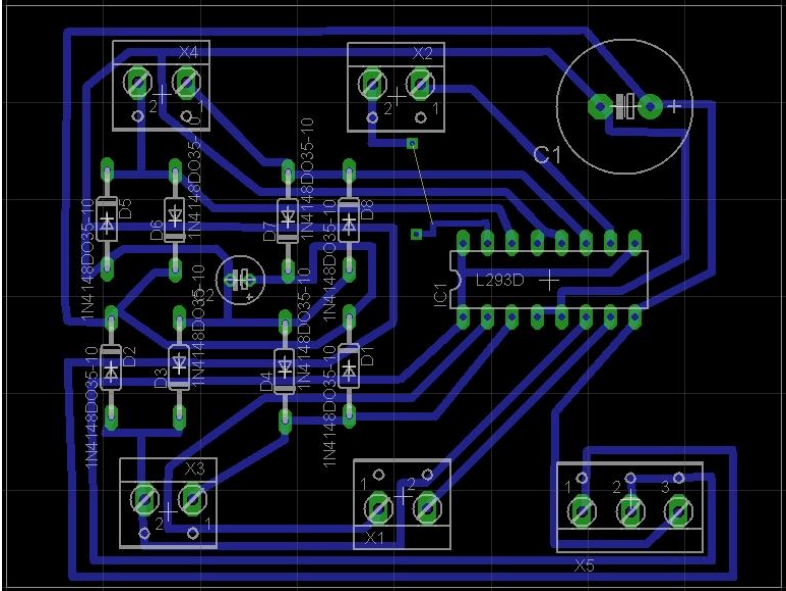


Figura 37. Circuito impreso tarjeta Puente H L298N

## ANEXO 5

### GUÍA MANUAL DE UTILIZACIÓN DE CARGADOR DE BATERÍA LIPO.

- 1.- Se conecta el cable de poder desde la fuente 110V AC al cargador de batería TURNIGY MAX 80W.
- 2.- Con el pulsador “BATT TYPE” se selecciona el tipo de batería, como se va a cargar la batería LIPO de 11.1 V a 2.2 A se selecciona: PROGRAM SELEC “LIPO BATT”.
- 3.- Se pulsa “ENTER”, en la pantalla LCD se visualizará lo siguientes: LIPO CHARGE, VALOR DE AMPERAJE, VALOR DE VOLTAJE Y NUMERO DE CELDAS, al estar en esta opción la batería iniciaría el proceso de recarga, con el pulsador “INC” se selecciona la opción LIPO BALANCE luego se pulsa “ENTER” con esta opción y con los pulsadores “DEC” y “INC” se pueden definir los valores de amperaje y voltaje con lo que están especificadas los diferentes tipos de baterías, al tener definidos los valores de voltaje y amperaje de la batería que va a ser recargada se presiona “ENTER”.
- 4.- Se procede a conectar las clavijas al cargador de batería positivo (+), negativo (-) y los terminales de la batería.
- 5.-Se conecta el cable de anclaje (cable de 4 terminales) para que el proceso de carga de la batería sea balanceado.
- 6.-Se presiona el pulsado “ENTER” por tres segundos se escuchara un pitido y en la pantalla LCD se visualizara las siguientes leyendas: CANCEL (STOP) con esta opción se detendrá el proceso de carga de la batería, si se selecciona CONFIRM (ENTER) se continuara con el proceso de carga, la opción a seleccionar se le hace presionando el pulsador “ENTER”.
- 7.-Durante el proceso de carga, si se desea visualizar la carga por celda se debe presionar el pulsador “INC”, en la pantalla LCD se podrá visualizar los valores de cada celda.
- 8.- Si se desea suspender el proceso de carga de la batería se debe presionar el pulsador “STOP”.

## **ANEXO 6**

### **MANUAL DE USUARIO**

En este manual se describe los objetivos e información clara y concisa de cómo utilizar el robot móvil de seguridad y vigilancia desmontable.

El robot está basado en funciones, fue creado con el objetivo de optimizar el trabajo de los estudiantes y lograr mejor eficiencia en las prácticas de laboratorio de la escuela de Electrónica y Telecomunicaciones.

Es recomendable que se revise este manual antes de que sea utilizado el prototipo, ya que muestra paso a paso en el manejo de las funciones que se le ha asignado para su funcionamiento.

#### **Objetivos**

- Ayudar al usuario a utilizar de forma correcta el robot móvil de seguridad y vigilancia.
- Entender las funcionalidades de robot para el manejo del mismo.

#### **Guía de uso**

El robot de seguridad y vigilancia con visión nocturna desmontable fue diseñado y programado para ejecutar el recorrido de guardiana operando de modo manual y modo automático.

Paso 1: Conectar la fuente de energía del robot, en este caso la batería LIPO DE 11.1 V/ 2.2 A, que está conectada a los motores y puente H L298N respectivamente alimentando de energía a todo el circuito.

Paso 2: Prender el robot con el botón interruptor, que se encuentra en la parte trasera del robot y posteriormente el interruptor que está conectado al arduino ubicado en la parte lateral de la parte superior.

Paso 3: Una vez encendido el robot móvil, conectamos el cable USB del arduino para cargar la codificación, para esto se procede a abrir el archivo donde se encuentra



la programación, hay que tener en cuenta que este designado bien el puerto serial COM para mandar a cargarlo.

Paso 4: Revisar que las conexiones de cada elemento estén bien conectadas

El robot móvil cuenta con los siguientes sistemas:

### **Sistema de locomoción:**

En este sistema el prototipo móvil hace uso de cuatro motores DC con sus respectivas llantas acopladas para su desplazamiento.

Cada motor tiene un torque de 1.1 Kg/cm, las sumatoria de todos los torques de los motores serán distribuidos en el peso del robot para que tenga un balance apropiado y el robot pueda desplazarse sin ningún problema ya que el peso total del prototipo es de 2 Kg

### **Sistema Actuador**

El puente H L298N para control de motores, dispositivo que proporciona corriente necesaria para poder controlar el sentido de giro de los cuatro motores DC.

### **Sistema de control**

Para el control del robot se debe conocer los elementos que actúan en este sistema:

#### **Tarjeta arduino Mega 2560.-**

Acoplada en la estructura del prototipo móvil, contiene un total de 54 pines de los cuales 15 son utilizados, es encargada para ejecutar los datos para que el robot móvil realice las funciones determinadas.

#### **Distribución de los pines de los elementos con la tarjeta arduino.**

La distribución de los pines servirá como guía al momento de programar en el software arduino, indicando cada una de sus funciones conectadas a la tarjeta mega arduino mostrados en la siguiente tabla:

#	Pines del arduino	Función
1	Pin 0	Tx Xbee s1
2	Pin 1	Rx Xbee s1
3	Pin 8	Eco (sensor ultrasónico)
4	Pin 9	Trigger (sensor ultrasónico)
5	Pin 7	Motor M1D (M3)
6	Pin 6	Motor M1P (M1)
7	Pin 5	Motor M2D (M2)
8	Pin 4	Motor M2P (M4)
9	Pin 30	Cámara y servo
10	Pin 3	Sensor PIR
11	Pin 11	Buzzer
12	Pin 40	Led posterior derecha
13	Pin 42	Led posterior izquierda
14	Pin 44	Led frente derecha
15	Pin 46	Led frente izquierda
16	Pin GND	Tierra común
17	Pin Vcc	+5V

### **Tarjeta FPGA NANO.-**

Esta tarjeta está ubicada arriba en la parte trasera del prototipo conectado al display LCD que servirá para la obtención de adquirir datos que se recibe en su programación asignando cada pin para su ejecución, como se muestra en la siguiente tabla.

Nombre de nodo	Localización de pines
LCD_DATAOUT [7]	PIN_D5
LCD_DATAOUT [6]	PIN_A6
LCD_DATAOUT [5]	PIN_D6
LCD_DATAOUT [4]	PIN_C6
LCD_DATAOUT [3]	PIN_E6
LCD_DATAOUT [2]	PIN_D8
LCD_DATAOUT [1]	PIN_F8
LCD_DATAOUT [0]	PIN_E9
LCD_E	PIN_D9
LCD_RS	PIN_B11

### **Tarjeta reguladora de voltaje.-**

Esta tarjeta ubicada en la parte de control del prototipo es encargada de regular el voltaje utilizando transistores de 7805 y 7809 con salidas de voltaje de 5V y 9 V para suministro a tarjeta arduino y cámara inalámbrica.

Para la parte de detección los elementos que intervienen son:

- **Sensor Ultrasónico HC-SR04.-** Es el encargado de indicar la distancia en la que se encuentra algún objeto a detectar.
- **Sensor de movimiento PIR.-** Es el encargado de detectar movimiento de un intruso.
- **Buzzer Activo.-** Es el encargado de activarse la alarma si el sensor PIR detecta un movimiento.
- **Micro Servo MG90S de 2kg.-** Provocara el movimiento de la cámara inalámbrica.
- **Cámara inalámbrica.-** Es la encargada de capturar la imagen.

### **Sistema de comunicación**

La interfaz del usuario inicia con una comunicación inalámbrica donde se utilizará el módulo de comunicación TX y RX del Xbee, seleccionando el puerto COM que este ocupando el Xbee s1, esto se puede verificar ingresando en la ventana del software arduino seleccionando el icono de herramienta ir a puerto serial. Al seleccionarlo nos ubicamos en la parte superior derecha donde nos aparece el icono monitor serial inmediatamente nos abre la ventana seleccionada para la comunicación del Xbee y vemos si el robot está operando en forma manual o automática.

### **Requerimientos de software**

**Software arduino 1.0.5.-** Utilizado para la programación de control del robot móvil de vigilancia y seguridad para ejecutar su desarrollo.

**Quartus II 10.1.-** Utilizado para código VHDL que se mostrará como adquisición de datos presentados en el display LCD.

**LabVIEW 2013.-** Utilizado para mostrar la gráfica de la simulación de captura de imagen con la cámara inalámbrica.

**X-CTU.-** Utilizado para configuración de los Xbee s1.

### **Función del robot móvil de seguridad y vigilancia de manera general**

La función que cumple el robot móvil de seguridad y vigilancia se da mediante su codificación que opera de modo manual y modo automático.

#### **Modo manual**

Para este modo de operación el usuario se encarga de comandar las opciones para que el robot cumpla con el recorrido de guardiana en este caso se designa lo siguiente:

Estado del robot	Opción a ejecutar
avanza	'u'
Giro derecha	'l'
Giro izquierda	'g'
Hacia atrás	'n'
detener	'p'
Movimiento de servo	's'
Automático	'a'

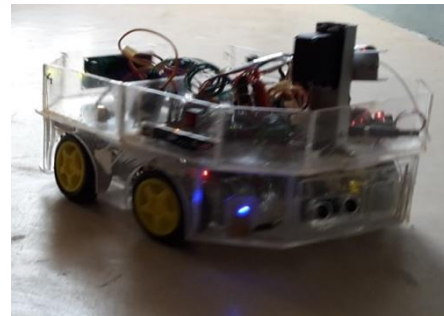
Estas funciones se cumplen a continuación mostrando en cada gráfica la asignación que se ejecutó para el funcionamiento del robot móvil de seguridad y vigilancia.

- En la figura 38 (a) muestra el avance del prototipo móvil para que el robot cumpla esta función se ejecuta la letra 'u', se visualiza que los leds de en frente del prototipo están encendidos, por esta razón se aprecia que el robot está en funcionamiento.
- En la segunda figura 38 (b), se visualiza que robot ejecuta el giro hacia la derecha para que se cumpla esta función se comanda mediante la letra 'l' del ordenador.

- La tercera figura 38 (c), el comando que se ejecuta para la función es la letra 'g' indicando que el robot móvil gire hacia la izquierda.
- En la cuarta figura 38 (d), indica que el robot móvil está en función de ejecución de ir hacia atrás, para esto se le da el comando de la letra 'n'
- La quinta figura 38 (e), el comando a ejecutar para la función es la letra 'p' indicando que el robot se detenga.



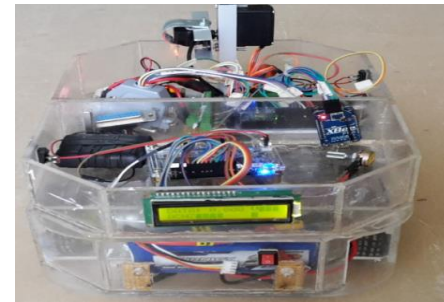
a



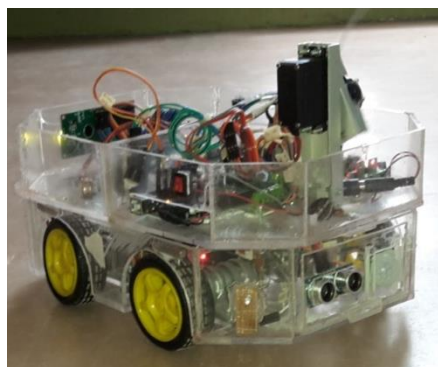
b



c



d



e

Figura 38. Función del robot móvil de seguridad y vigilancia de manera general

### **Modo automático**

Cuando el robot se encuentra en modo automático 'a' comienza a recorrer en un área plana cumpliendo los tres casos caso1: avanza, caso2: gira derecha, caso3: avanza, cuando el robot detecte obstáculo el sensor determinara la distancia  $<25$  el robot se detendrá y si se da un movimiento el sensor PIR y la alarma se activará accionándose el servo para que la cámara capture la imagen del intruso.

## ANEXO 7

### GUÍA MANUAL DE PRÁCTICAS PARA LABORATORIO DE ELECTRÓNICA Y TELECOMUNICACIONES.

#### PRÁCTICA N°1

Esta práctica consiste en realizar el recorrido en forma de un cuadrado, mediante la codificación que se realizó, utilizando los siguientes dispositivos

- Motores
- Arduino mega
- Puente H
- Batería LI-PO de 11.1 V/ 2.2 A

Para este experimento se realizó la codificación en el software arduino, con el objetivo de que el robot cumpla con el recorrido en un área plana formando un cuadrado.

#### CÓDIGO EN SOFTWARE ARDUINO

```
int EM1 = 0;           //Encoder motor 1 (Izquierda) (Entrada 2)
int EM2 = 1;           //Encoder motor 2 (Derecha) (Entrada 3)
int M2D = 4;           //Motor 2 control de dirección (Derecha)
int M1P = 6;           //Motor 1 control PWM
int M2P = 5;           //Motor 2 control PWM
int M1D = 7;           //Motor 1 control de dirección (Izquierda)

int LED_FD=44;         // led frente derecha
int LED_FI=46;         //led frente izquierda
int LED_PD=40;         // led postero derecha
int LED_PI=42;

void setup()           //Se ejecuta cada vez que el Arduino se inicia
{
  pinMode (44, OUTPUT); //salida led frontal
```

```

pinMode (46, OUTPUT);           //salida led frontal
pinMode (40, OUTPUT);           //salida led posterior
pinMode (42, OUTPUT);           //salida led posterior
                                //Mueve el servo a la posición entrada (excepto si es 0)
}

void avanzar() {
  analogWrite(M1P,255);
  analogWrite(M2P,255);
  analogWrite(M1D,0);
  analogWrite(M2D,0);
  digitalWrite(LED_FD,HIGH);
  digitalWrite(LED_FI,HIGH);
  digitalWrite(LED_PD,LOW);
  digitalWrite(LED_PI,LOW);
}

void derecha() {
  analogWrite(M1P,255);
  analogWrite(M2P,0);
  analogWrite(M1D,0);
  analogWrite(M2D,0);
  digitalWrite(LED_FD,HIGH);
  digitalWrite(LED_FI,LOW);
  digitalWrite(LED_PD,LOW);
  digitalWrite(LED_PI,LOW);
}

void izquierda(){
  analogWrite(M1P,0);
  analogWrite(M2P,255);
  analogWrite(M1D,0);
  analogWrite(M2D,0);
  digitalWrite(LED_FD,LOW);
  digitalWrite(LED_FI,HIGH);
  digitalWrite(LED_PD,LOW);
  digitalWrite(LED_PI,LOW);
}

void detener(){
  analogWrite(M1P,0);
  analogWrite(M2P,0);
  analogWrite(M1D,0);
  analogWrite(M2D,0);
  digitalWrite(LED_FD,LOW);
  digitalWrite(LED_FI,LOW);
  digitalWrite(LED_PD,LOW);
  digitalWrite(LED_PI,LOW);
}
}

```



```
void atras(){
  analogWrite(M1P,0);
  analogWrite(M2P,0);
  analogWrite(M1D,255);
  analogWrite(M2D,255);
  digitalWrite(LED_FD,LOW);
  digitalWrite(LED_FI,LOW);
  digitalWrite(LED_PD,HIGH);
  digitalWrite(LED_PI,HIGH);
}
```

```
void loop() {

  detener();
  delay(2000);
  avanzar();
  delay(2000);
  derecha();
  delay(1400);
  avanzar();
  delay(1500);
  derecha();
  delay(1600);
  avanzar();
  delay(1900);
  derecha();
  delay(1600);
  avanzar();
  delay(2000);
  detener();
  delay(2000);}
}
```

### **Procedimiento Experimental**

Para esta prueba se hace uso del robot móvil de seguridad y vigilancia en la que se va a utilizar los cuatro motores, puente H, tarjeta arduino y la batería para suministrar el robot de energía.

Se prepara el lugar donde se va hacer el recorrido donde se pueda verificar el control de los motores para recorrer distancia en línea recta, realizar giros de 90° y poder cuantificar el error de desviación, para realizar dicha prueba se programó la tarjeta arduino para controlar el robot móvil para un recorrido de distancia de 40x40 cm los escenarios propuestos para estas pruebas de distancia se muestran en las Figura 39, se realizaron secciones de 10 pruebas realizadas para cada trayectoria.

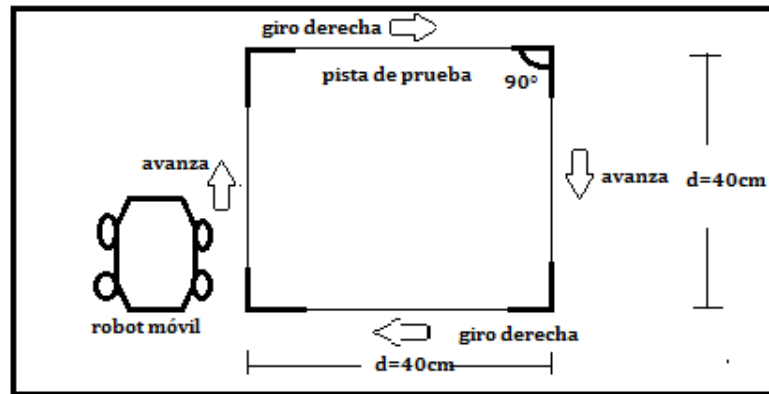


Figura 39. Práctica N° 1 recorrido forma cuadrada

### Resultados y pruebas del recorrido forma cuadrada

Para esta prueba se realizó la codificación en arduino para que el robot móvil realice una trayectoria cuadrada, recorriendo una distancia de 40x40 cm, con esto se pretende determinar el error de desviación cuando el robot gira en un ángulo recto de 90°.

N°Prueba	Distancia recorrida[cm]	Angulo de desviación[grados]	Angulo de giro[grados]	Resultado
1	40	0	89	Exitoso
2	38.8	10	90	Exitoso
3	40.1	5	89	Exitoso
4	42	0	91	Exitoso
5	37.3	10	90	Exitoso
6	39.3	10	88	Exitoso
7	40	0	90	Exitoso
8	40	10	89	Exitoso
9	40	5	90	Exitoso
10	40	5	90	Exitoso
promedio	39,75	5,5	89,7	--

Tabla 1. Datos y pruebas para distancia y giro del robot recorrido en forma cuadrada

Se determinó que para la trayectoria de forma cuadrada cuya distancia es de 40x40 cm, luego de haber realizado las 10 pruebas el valor promedio de dicha distancia fue de 39,75 cm con un error de 0.1% pues su margen de error es relativamente bajo.

## Conclusiones

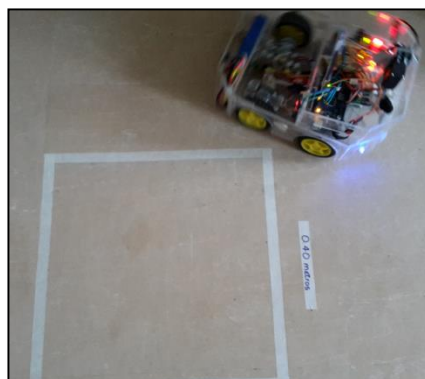
Del experimento realizado se concluye que la distancia recorrida por el robot móvil tiene un margen de error pequeño del 0.1% con un desplazamiento de trayectoria de forma cuadrada de un punto de partida conocido y llegar a un punto de llegada. Se mostró el desarrollo de la codificación en arduino para la ejecución de recorrido del robot y se determinó que los resultados fueron satisfactorios.

En esta figura 40 se visualiza el robot con recorrido en forma cuadrada en un área de 40x40cm, es decir el cuadrado tiene distancia de 0.4x0.4 m, estando el prototipo en un punto inicial.



Figura 40. El robot móvil comienza a recorrer el área cuadrada de distancia 0.4mx0.4m.

Luego del avance del robot da un giro hacia la derecha para luego continuar avanzando con el recorrido de forma cuadrada.



Giro derecha del robot

Figura 41. El robot móvil comienza a recorrer el área cuadrada de distancia 0.4mx0.4m.

El robot avanza

Gira derecha luego avanza

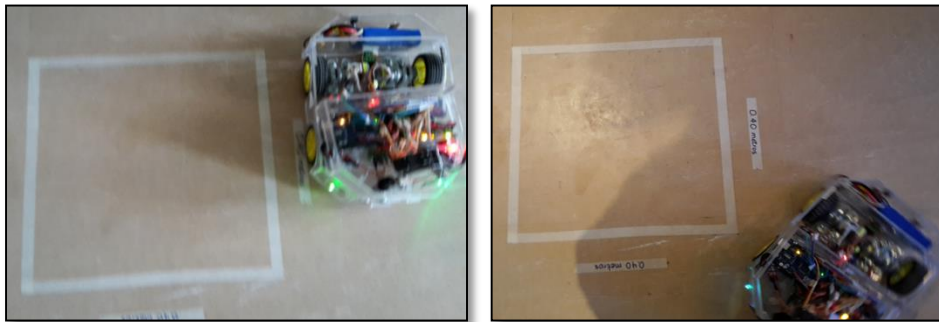


Figura 42. El robot móvil comienza a recorrer el área cuadrada de distancia 0.4mx0.4m

Luego gira y nuevamente avanza hasta llegar al punto inicial, con esto se concluye que el robot móvil ejecuta su recorrido en forma cuadrada



Figura 43. El robot móvil comienza a recorrer el área cuadrada de distancia 0.4mx0.4m

## PRÁCTICA N°2

### Distancia de detección de obstáculo

Esta práctica consiste en realizar el recorrido del robot móvil haciendo uso del sensor ultrasónico, determinado la distancia que se encuentra el obstáculo este se detendrá caso contrario avanzara con el recorrido.

Componentes a utilizar:

- Motores
- Arduino mega
- Puente H
- Sensor ultrasónico
- Batería LI-PO de 11.1 V/ 2.2 A

## CÓDIGO EN SOFTWARE ARDUINO

```
int EM1 = 0;           //Encoder motor 1 (Izquierda) (Entrada 2)
int EM2 = 1;           //Encoder motor 2 (Derecha) (Entrada 3)
int M2D = 4;           //Motor 2 control de dirección (Derecha)
int M1P = 6;           //Motor 1 control PWM
int M2P = 5;           //Motor 2 control PWM
int M1D = 7;           //Motor 1 control de dirección (Izquierda)

int LED_FD=44;         // led frente derecha
int LED_FI=46;         //led frente izquierda
int LED_PD=40;         // led posterior derecha
int LED_PI=42;
int SU = 8 ;           //Sensor de distancia por ultrasonidos eco
int SA = 9;

long tiempo = 0;
long distancia = 0;

void setup()           //Se ejecuta cada vez que el Arduino se inicia
{
  // funcion para determinar distancia con ultrasonico
  pinMode (44, OUTPUT); //salida led frontal
  pinMode (46, OUTPUT); //salida led frontal
  pinMode (40, OUTPUT); //salida led posterior
  pinMode (42, OUTPUT); //salida led posterior
  //Mueve el servo a la posición entrada (excepto si es 0)
  Serial.begin(9600);
  pinMode (9, OUTPUT);
  pinMode (8, INPUT);
}
void avanzar() {
  analogWrite(M1P,255);
  analogWrite(M2P,255);
  analogWrite(M1D,0);
  analogWrite(M2D,0);
  digitalWrite(LED_FD,HIGH);
  digitalWrite(LED_FI,HIGH);
  digitalWrite(LED_PD,LOW);
  digitalWrite(LED_PI,LOW);
}
void derecha() {
  analogWrite(M1P,255);
  analogWrite(M2P,0);
  analogWrite(M1D,0);
  analogWrite(M2D,0);
  digitalWrite(LED_FD,HIGH);
  digitalWrite(LED_FI,LOW);
  digitalWrite(LED_PD,LOW);
  digitalWrite(LED_PI,LOW);
}
```

```

void izquierda(){
  analogWrite(M1P,0);
  analogWrite(M2P,255);
  analogWrite(M1D,0);
  analogWrite(M2D,0);
  digitalWrite(LED_FD,LOW);
  digitalWrite(LED_FI,HIGH);
  digitalWrite(LED_PD,LOW);
  digitalWrite(LED_PI,LOW);
}
void detener(){
  analogWrite(M1P,0);
  analogWrite(M2P,0);
  analogWrite(M1D,0);
  analogWrite(M2D,0);
  digitalWrite(LED_FD,LOW);
  digitalWrite(LED_FI,LOW);
  digitalWrite(LED_PD,LOW);
  digitalWrite(LED_PI,LOW);
}
void atras(){
  analogWrite(M1P,0);
  analogWrite(M2P,0);
  analogWrite(M1D,255);
  analogWrite(M2D,255);
  digitalWrite(LED_FD,LOW);
  digitalWrite(LED_FI,LOW);
  digitalWrite(LED_PD,HIGH);
  digitalWrite(LED_PI,HIGH);
}
void loop() {
  digitalWrite(9,LOW);
  delayMicroseconds(2);
  digitalWrite(9,HIGH);
  delayMicroseconds(10);
  digitalWrite(9,LOW);
  tiempo=pulseIn(8,HIGH);
  distancia= int(0.017*tiempo);
  Serial.println(distancia);
  delay(10);
  avanzar();
  if //(distancia<30 && distancia>10){
    (distancia<30){
      detener();
    }
  }
}

```

## PROCEDIMIENTO EXPERIMENTAL

En esta prueba se configuraron diferentes distancias de detección de obstáculo con la finalidad de medir la distancia que se determinó en su codificación, es decir si el robot se detiene a una distancia  $<30$  determinara que se presentó un obstáculo y se detendrá caso contrario avanzara con su trayecto de recorrido, se realizaron secciones de 10 pruebas.

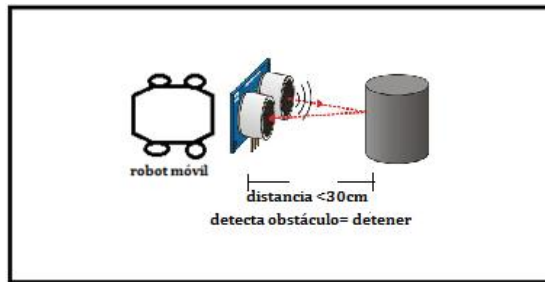


Figura 44. Distancia de detección de obstáculo

## RESULTADOS DE LAS PRUEBAS EN PRACTICA 2

Para esta prueba se realizó la codificación en arduino para que el robot móvil realice el recorrido se espera que el robot pueda determinar la distancia del obstáculo de  $<30$  cm para que se detenga si se retira el objeto el robot seguirá con el recorrido.

N° Prueba	Distancia entre el robot y el objeto [cm]	Presencia obstáculo	Modo de ejecución	Resultado
1	37	No	avanza	Exitoso
2	36	No	avanza	exitoso
3	36	No	avanza	Exitoso
4	35	No	avanza	Exitoso
5	33	No	avanza	Exitoso
6	32	No	avanza	Exitoso
7	31	No	avanza	Exitoso
8	31	Si	avanza	Exitoso
9	30	Si	Se detiene	Exitoso
10	3	Si	Se detiene	Exitoso
promedio	30.4	--	--	--

Tabla 1 resultados de prueba distancia de detección de obstáculo

De acuerdo a la tabla se determinó que para la detección entre el robot y el obstáculo la distancia es de 30 cm luego de haber realizado las 10 pruebas el valor promedio de dicha distancia fue de 30.4 cm relativamente sin margen de error

## Conclusiones

El desarrollo de este experimento se detectó buenos resultados en el control del recorrido del robot determinando la distancia entre el robot y obstáculo a una distancia  $<30$  cm

Con la codificación en arduino para el recorrido del robot móvil se determinó que los resultados fueron satisfactorios.

En esta imagen se visualiza la ejecución del robot móvil presenciando un obstáculo cartón impidiendo el paso a seguir del robot encontrándose a una distancia de 3cm entre robot y obstáculo.

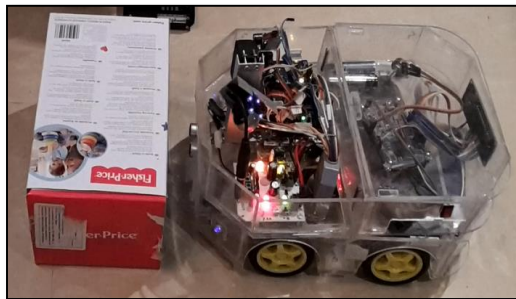


Figura 45. Distancia de detección de obstáculo

## PRÁCTICA N°3

Esta práctica consiste en realizar el recorrido en un área plana operando en modo manual ejecutando comando que están asignados en la codificación, utilizando los siguientes dispositivos

- Motores
- Arduino mega
- Puente H
- Módulo de comunicación xbee
- Batería LI-PO de 11.1 V/ 2.2 A

Para este experimento se realizó la codificación en el software arduino, con el objetivo de que el robot cumpla con el recorrido en un área plana operando de modo manual ejecutando comandos como avanzar, giro derecha, giro izquierda, hacia



atrás y detener, haciendo uso del módulo de comunicación xbee entre el robot y el ordenador asignado el puerto serial COM respectivamente.

## CÓDIGO EN SOFTWARE ARDUINO

```
char val;
int EM1 = 0;           //Encoder motor 1 (Izquierda) (Entrada 2)
int EM2 = 1;           //Encoder motor 2 (Derecha) (Entrada 3)
int M2D = 4;           //Motor 2 control de dirección (Derecha)
int M1P = 6;           //Motor 1 control PWM
int M2P = 5;           //Motor 2 control PWM
int M1D = 7;           //Motor 1 control de dirección (Izquierda)
//Variable para saber por donde girar
byte caso = 0;        //Para seleccionar el caso de giro

int LED_FD=44;        // led frente derecha
int LED_FI=46;        //led frente izquierda
int LED_PD=40;        // led posterior derecha
int LED_PI=42;

//*****
//*****

void setup()          //Se ejecuta cada vez que el Arduino se inicia
{
  Serial.begin(9600); //abre el puerto serie , establece la velocidad de
datos a 9600 bps
  pinMode (44, OUTPUT); //salida led frontal
  pinMode (46, OUTPUT); //salida led frontal
  pinMode (40, OUTPUT); //salida led posterior
  pinMode (42, OUTPUT); //salida led posterior
}

//*****
//*****

void gaby() {

  Serial.println(" manual"); //mostrar por el puerto serial'movimiento servo'

  val = Serial.read();
  switch(val)
  {
  case 'u':
    Serial.println("avance"); //mostrar por el puerto serial'avance'
    avanzar();
    break;

  case 'l':
    Serial.println("derecha"); //mostrar por el puerto serial'derecha'
    derecha();
    break;
  }
```

```

case 'g':
  Serial.println("izquierda");          //mostrar por el puerto serial'izquierda'
  izquierda();
  break;

case 'p':
  Serial.println("detener");           //mostrar por el puerto serial'detener'
  detener();
  break;

case 'n':
  Serial.println("atras");             //mostrar por el puerto serial'atras'
  atras();
  break;
}
}
//*****
*****
*****

void avanzar() {
  analogWrite(M1P,255);
  analogWrite(M2P,255);
  analogWrite(M1D,0);
  analogWrite(M2D,0);
  digitalWrite(LED_FD,HIGH);
  digitalWrite(LED_FI,HIGH);
  digitalWrite(LED_PD,LOW);
  digitalWrite(LED_PI,LOW);
}

void derecha() {
  analogWrite(M1P,255);
  analogWrite(M2P,0);
  analogWrite(M1D,0);
  analogWrite(M2D,0);
  digitalWrite(LED_FD,HIGH);
  digitalWrite(LED_FI,LOW);
  digitalWrite(LED_PD,LOW);
  digitalWrite(LED_PI,LOW);
}

void izquierda(){
  analogWrite(M1P,0);
  analogWrite(M2P,255);
  analogWrite(M1D,0);
  analogWrite(M2D,0);
  digitalWrite(LED_FD,LOW);
  digitalWrite(LED_FI,HIGH);
  digitalWrite(LED_PD,LOW);
  digitalWrite(LED_PI,LOW);
}

void detener(){

```

```

analogWrite(M1P,0);
analogWrite(M2P,0);
analogWrite(M1D,0);
analogWrite(M2D,0);
digitalWrite(LED_FD,LOW);
digitalWrite(LED_FI,LOW);
digitalWrite(LED_PD,LOW);
digitalWrite(LED_PI,LOW);

}

void atras(){
  analogWrite(M1P,0);
  analogWrite(M2P,0);
  analogWrite(M1D,255);
  analogWrite(M2D,255);
  digitalWrite(LED_FD,LOW);
  digitalWrite(LED_FI,LOW);
  digitalWrite(LED_PD,HIGH);
  digitalWrite(LED_PI,HIGH);
}
//*****
*****
*****

void loop() {
//Esta funcion se mantiene ejecutando cuando
este energizado el Arduino
  gaby(); // casos de manual
}

```

## Procedimiento Experimental

Para esta prueba se hace uso del robot móvil de seguridad y vigilancia en la que se va a utilizar los cuatro motores, puente H, tarjeta arduino, módulo de comunicación xbee y la batería para suministrar el robot de energía.

Se prepara el lugar donde se va hacer el recorrido donde se pueda verificar el control de los motores operando de modo manual ejecutando las opciones ‘u’ avanzar, ‘l’ giro derecha, ‘g’ giro izquierda, ‘n’ hacia atrás y ‘p’ detener.

## Resultados y pruebas del recorrido modo manual

Para esta prueba se realizó la codificación en arduino para que el robot móvil realice una trayectoria del recorrido ejecutando el modo de operación manual en un área libre con esto se pretende determinar el alcance de la comunicación del xbee cuando el robot se encuentre ejecutando las funciones de control.

N°Prueba	Alcance de comunicación Xbee y robot móvil con diferentes distancias	Opción de mando	Respuesta de mando	Resultado	Porcentaje
1	6 metros	Avanza 'u', detener 'p' Giro derecha 'l' Giro izquierda 'g'	Inmediato	Exitoso	90%
2	9 metros	Avanza 'u', detener 'p' Giro derecha 'l' Giro izquierda 'g'	Inmediato	Exitoso	90%
3	12 metros	Avanza 'u', detener 'p' Giro derecha 'l' Giro izquierda 'g'	Inmediato	Exitoso	90%
4	15 metros	Avanza 'u', detener 'p' Giro derecha 'l' Giro izquierda 'g'	Inmediato	Exitoso	90%
5	18 metros	Avanza 'u', detener 'p' Giro derecha 'l' Giro izquierda 'g'	Retardo de 2 segundos	No exitoso	80%
6	21 metros	Avanza 'u', detener 'p' Giro derecha 'l' Giro izquierda 'g'	Retardo de 2 segundos	No exitoso	80%
7	24 metros	Avanza 'u', detener 'p' Giro derecha 'l' Giro izquierda 'g'	Retardo de 2 segundos	No exitoso	80%
8	27 metros	Avanza 'u', detener 'p' Giro derecha 'l' Giro izquierda 'g'	Retardo de 3 segundos	No exitoso	70%
9	30 metros	Avanza 'u', detener 'p' Giro derecha 'l' Giro izquierda 'g'	Retardo de 3 segundos	No exitoso	70%
10	40 metros	Avanza 'u', detener 'p' Giro derecha 'l' Giro izquierda 'g'	Retardo de 5 segundos	No exitoso	50%

Tabla 1.Datos y pruebas para alcance de comunicación

Esta tabla especifica datos reales tomados en un área libre de hasta 40 metros para el recorrido del robot móvil mediante comunicación Xbee determinando el alcance operando en modo manual ejecutando opciones de mando y verificar la respuesta de su funcionamiento del robot.

En la imagen siguiente se observa la realización de las pruebas que fueron tomadas en un area libre, efectuando mediciones con el robot movil y el alcance del módulo de comunicación del xbee, para eso se determino una distancia de 6 hasta 40 metros.



Figura 46. Alcance de comunicación entre robot móvil y módulo xbee

## **Conclusiones**

Del experimento realizado se concluye que el alcance del recorrido del robot móvil mediante la comunicación Xbee es de 40 metros en un área libre, se tomaron datos reales para sus resultados operando de modo manual.

Se verifico el desarrollo de la codificación en arduino para la ejecución de recorrido del robot operando en modo manual haciendo uso del módulo de comunicación xbee para determinar su alcance de funcionamiento del recorrido de robot móvil, se determinó que los resultados es exitoso cuando su alcance llega hasta los 15 metros más allá de su distancia su comunicación tiene retardos de hasta 5 segundos.

## **GLOSARIO**

**RMR.-** Robot móvil con ruedas.

**SL.-** Sistema de locomoción

**PH.-** Puente H.

**DH Y DL.-** La dirección de destino para las comunicaciones inalámbricas.

**PAN ID.-** Identificación Red de Área

**MY:** La dirección de 16- bits del módulo Xbee

**LI-PO:** Polímero de Litio