



# INSTITUTO POLITÉCNICO NACIONAL

CENTRO DE INVESTIGACIÓN EN COMPUTACIÓN

“CONTROL DE NAVEGACIÓN EN SUPERFICIES PLANAS, DE UN  
TRACTOR NO TRIPULADO”.

TESIS

QUE PARA OBTENER EL GRADO DE:

MAESTRO EN CIENCIAS EN INGENIERÍA DE CÓMPUTO

CON OPCIÓN EN SISTEMAS DIGITALES

PRESENTA:

ING. IGNACIO HERNÁNDEZ BAUTISTA

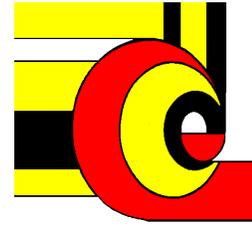
DIRECTOR: DR. OSCAR CAMACHO NIETO.

DIRECTOR: M. EN C. ROMEO URBIETA PARRAZALES.

# Resumen

Los vehículos no tripulados, pueden ser usados para realizar tareas en lugares difícilmente accesibles, con riesgo de accidentes, en condiciones peligrosas para la salud, o trabajos que resultan difíciles por el tamaño de los objetos que es necesario manipular. Por mencionar algunas aplicaciones donde se requiere el uso de vehículos no tripulados, tenemos el caso de la exploración espacial, exploración en el fondo del mar, mantos petroleros, zonas volcánicas, como apoyo al trabajador en la minería, la construcción, así como en la agricultura para fumigación, entre otros.

Como una aportación a esta disciplina, en el presente trabajo se desarrollan y describen los elementos necesarios para el entendimiento, manejo y manipulación de un tractor no tripulado con control a distancia, usando tanto hardware como software para el desarrollo de las rutinas de movimiento, utilizando como kit de desarrollo el Microcontrolador Rabbit 4400W, actuadores, drives, sensores, y servomotores.



## *Agradecimientos*

*Al Instituto Politécnico Nacional. Por la oportunidad y apoyo para finalizar una meta más.*

*Al Centro de Investigación en Computación. Por darme los medios necesarios para la terminación de esta tesis.*

*A mis asesores. Dr. Oscar Camacho Nieto y M. en C. Romeo Urbieta Parrazales. Por su guía, y apoyo para la terminación de esta meta.*

*Al Jurado por sus sugerencias, correcciones y comentarios.*

*A todas aquellas personas que estuvieron conmigo...*



---

# 1. INTRODUCCIÓN.

En los países de primer mundo como Estados Unidos y Japón, entre otros, se ha logrado madurar la tecnología relacionada con el desarrollo de robots; donde Japón se está centrando en crear robots que se muevan como seres humanos, por poner un ejemplo, ya se tiene un robot ASIMOV ("*Advanced Step in Innovative Mobility*", de Honda) que puede subir escalones, correr, bailar, traducir, etc. Mientras que los Estados Unidos se concentran en robots inteligentes y la habilidad para lograr expresiones faciales, aunado al desarrollo de robots, se han logrado grandes avances en el desarrollo de vehículos no tripulados, como el *Spirit* que fue el primer robot móvil no tripulado que se envió para la exploración en el planeta Marte, el segundo fue *Opportunity* que se encuentra explorando el planeta Marte. Los vehículos no tripulados, pueden ser usados para realizar tareas en lugares difícilmente accesibles, con riesgo de accidentes, en condiciones peligrosas para la salud, o trabajos que resultan difíciles por el tamaño de los objetos que es necesario manipular. Por mencionar algunas aplicaciones donde se requiere el uso de vehículos no tripulados, tenemos el caso de la exploración espacial, exploración en el fondo del mar, mantos petroleros, zonas volcánicas, como apoyo al trabajador en la minería, la construcción, manipulación de dispositivos explosivos, así como en la agricultura para fumigación, entre otros. Como una aportación a esta disciplina y con la finalidad de reducir la dependencia de tecnología que se tiene con los países de primer mundo, en relación al diseño y construcción del sistema de control digital de los vehículos no tripulados, en el



---

presente trabajo se desarrollan y describen los elementos necesarios para el entendimiento, manejo y manipulación de un tractor no tripulado con control a distancia, usando tanto hardware como software para el desarrollo de las rutinas de movimiento, utilizando como kit de desarrollo el Microcontrolador Rabbit 4400W, actuadores, drives, sensores, y servomotores.

### **1.1. Antecedentes.**

En los últimos tiempos, el estudio, análisis y desarrollo de los robots, ha sido parte importante en el desarrollo de la tecnología de cada país. En el término robot confluyen las máquinas para la realización de trabajos productivos e imitación de movimientos y comportamientos de seres vivos.

Los robots actuales son obras del desarrollo del estudio, investigación e ingeniería y como tales son diseñadas para producir bienes y servicios, o exportar recursos naturales. Desde esta perspectiva son máquinas que desarrollan una actividad que parte de los propios orígenes de la humanidad, y que desde el comienzo de la edad moderna se fundamenta esencialmente en conocimientos científicos y desarrollo tecnológico.

Desde la antigüedad, el hombre ha externado su fascinación por las máquinas que imitan la figura y los movimientos de los seres vivos. Existe una larga producción de autómatas en el mundo, desde los automotores de los artesanos franceses y suizos del siglo XVIII, que incorporaban dispositivos mecánicos para control automático de movimientos, hasta nuestro siglo, con los actuales robots que cantan, bailan, caminan, corren, y andan en bicicleta.



El esquema básico de un robot, cuenta con un sistema mecánico, eléctrico, electrónico, actuadores, sensores, programación y de control. [1] **¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia.**

Hay diferentes tipos de robots, dependiendo de su concepción o aplicación.

- Robots manipuladores (brazos).
  - Cilíndricos.
  - Esféricos.
  - Cartesianos.
  - Mixtos.
- Robots móviles.
  - Tipo y número de ruedas.
- Robots híbridos (móviles con manipulación).
- Vehículos autónomos:
  - Tele operados.
  - Repetitivos.
  - Autónomos.
- Robots caminantes.
  - 2 patas (*humanoides*).
  - 4/6 patas (*insectos*) .

En la sección del Estado del Arte, se desarrollarán estos temas.



## 1.2 Justificación.

Los robots se consideran como máquinas que integran un conjunto de componentes de diferentes disciplinas como es el caso de la mecánica, eléctrica y electrónica, así como de las comunicaciones; que al integrarse se les incorporan sistemas informáticos que se encargan del control y administración de los recursos para comprensión e interacción con el medio ambiente donde serán usados.

A finales del siglo XX, se desarrollaron máquinas que favorecían las actividades del trabajador en la industria, así como la procuración de su salud e integridad física, es decir, se desarrollaron máquinas que realizan actividades que el ser humano venía realizando pero que ponía en riesgo su vida o integridad física y donde el hombre difícilmente puede llegar, como es caso de la exploración espacial; por mencionar algunas: en lugares difícilmente accesibles, como en la minería y la arquitectura, para evitar que se expongan a sustancias tóxicas, como en la agricultura cuando se desarrolla la actividad de fumigación, en la medicina para manipular con precisión los instrumentos quirúrgicos, entre otras. La característica fundamental de estas máquinas, es que son operadas de forma remota, las cuales, para que formen parte del grupo considerado como robots, es necesario que contengan tal autonomía suficiente, que les permita trabajar de forma independiente.

Por no dejar de lado otras aplicaciones de la robótica, como es el caso de los robots diseñados para dar servicio doméstico, ayuda a discapacitados, entre otros, se menciona que se están realizando avances de gran importancia, tanto que se han logrado desarrollos de androides que simulan hasta los gestos humanos y



robots que muestran la apariencia de algún ser vivo, así como los desarrollos que se emplean para ofrecer servicios de recreación; dado que no es la meta de la presente aportación, se salen de los objetivos de este trabajo. Cabe aclarar que las aportaciones descritas en el cuerpo del presente trabajo, pueden servir como complemento en el desarrollo de cualquiera de los robots mencionados anteriormente.

Durante los últimos años los aspectos relacionados con la robótica denominada móvil han cobrado una gran importancia. Los robots móviles a diferencia de los robots manipuladores convencionales es que no están anclados, sino que por el contrario pueden desplazarse por el terreno, por el agua o incluso volar libremente. Los aspectos más específicos de la robótica móvil son los relacionados con tales desplazamientos autónomos y de navegación del robot.

Es por ello que se hace esta propuesta de tesis, la cual nos permitirá desarrollar una tecnología propia y de bajos costos, además del entendimiento de los diferentes tipos de dispositivos móviles, arquitectura, locomoción y aplicación.

### **1.3 Objetivos.**

Objetivo Principal:

- Control de navegación en superficies planas, evitando obstáculos, de un tractor no tripulado.

Objetivos Particulares:

- Diseño de red inalámbrica para comunicación y control.
- Diseño de drivers para circuito PWM (Pulse Wide Modulate).



- Desarrollo de software para control de los mecanismos para la navegación.
- Desarrollo de las rutinas de activación del sensor detector de obstáculos.

#### **1.4 Organización del Documento.**

En el capítulo 2, **Estado del Arte**, se presenta un panorama del desarrollo de los robots, así como la descripción de los diferentes tipos de robots, arquitecturas y configuraciones. También se describen las redes inalámbricas y sus configuraciones.

En el capítulo 3, **Propuesta del Prototipo**. Se describe el sistema a controlar, las partes que lo componen, así como los lenguajes de programación y las características de cada parte.

En el capítulo 4, **Desarrollo de Rutinas de Trabajo**. Se muestra el sistema global de navegación, se presenta como se fueron generando los módulos que componen el sistema, así como sus diagramas de flujo, programas y diagramas esquemáticos de cada parte del sistema.

En el capítulo 5, **Disquisición Experimental**. Se presenta una descripción detallada de cada parte del sistema y se hace un recuento puntual de cada parte que lo conforma, así como la problemática que se fue presentando para la realización de esta tesis; de igual forma se describe como se fueron resolviendo estas problemáticas del sistema.

En el capítulo 6, **Conclusiones**. Se presentan las conclusiones a las que se llegaron, así como las propuestas de trabajos a futuro.





---

## Resumen

Los vehículos no tripulados, pueden ser usados para realizar tareas en lugares difícilmente accesibles, con riesgo de accidentes, en condiciones peligrosas para la salud, o trabajos que resultan difíciles por el tamaño de los objetos que es necesario manipular. Por mencionar algunas aplicaciones donde se requiere el uso de vehículos no tripulados, tenemos el caso de la exploración espacial, exploración en el fondo del mar, mantos petroleros, zonas volcánicas, como apoyo al trabajador en la minería, la construcción, así como en la agricultura para fumigación, entre otros.

Como una aportación a esta disciplina, en el presente trabajo se desarrollan y describen los elementos necesarios para el entendimiento, manejo y manipulación de un tractor no tripulado con control a distancia, usando tanto hardware como software para el desarrollo de las rutinas de movimiento, utilizando como kit de desarrollo el Microcontrolador Rabbit 4400W, actuadores, drives, sensores, y servomotores.



---

## Abstract

The unmanned vehicles can be used to carry out tasks in hardly accessible places, at the risk of accidents, in dangerous conditions for the health, or works that are difficult by the size of the objects that are necessary to manipulate. To mention some applications where the use of vehicles nonmanned is required, we have the case of the reconnaissance space, oil exploration, mantles at bottom of the sea, volcanic zones, like support to the worker in the mining, the construction, as well as in fumigation applied to agriculture, among others.

Like a contribution to this discipline, in the present work the necessary elements for the understanding, handling and manipulation of a tractor nonmanned with remote control are developed and described, using as much hardware as software for the development of the movement routines, using Rabbit 4400w Microcontroller development kit, actuators, drives, sensors, and actuators.



# 1. INTRODUCCIÓN.

En los países de primer mundo como Estados Unidos y Japón, entre otros, se ha logrado madurar la tecnología relacionada con el desarrollo de robots; donde Japón se está centrando en crear robots que se muevan como seres humanos, por poner un ejemplo, ya se tiene un robot ASIMOV ("*Advanced Step in Innovative Mobility*", de Honda) que puede subir escalones, correr, bailar, traducir, etc. Mientras que los Estados Unidos se concentran en robots inteligentes y la habilidad para lograr expresiones faciales, aunado al desarrollo de robots, se han logrado grandes avances en el desarrollo de vehículos no tripulados, como el *Spirit* que fue el primer robot móvil no tripulado que se envió para la exploración en el planeta Marte, el segundo fue *Opportunity* que se encuentra explorando el planeta Marte. Los vehículos no tripulados, pueden ser usados para realizar tareas en lugares difícilmente accesibles, con riesgo de accidentes, en condiciones peligrosas para la salud, o trabajos que resultan difíciles por el tamaño de los objetos que es necesario manipular. Por mencionar algunas aplicaciones donde se requiere el uso de vehículos no tripulados, tenemos el caso de la exploración espacial, exploración en el fondo del mar, mantos petroleros, zonas volcánicas, como apoyo al trabajador en la minería, la construcción, manipulación de dispositivos explosivos, así como en la agricultura para fumigación, entre otros. Como una aportación a esta disciplina y con la finalidad de reducir la dependencia de tecnología que se tiene con los países de primer mundo, en relación al diseño y construcción del sistema de control digital de los vehículos no tripulados, en el



---

presente trabajo se desarrollan y describen los elementos necesarios para el entendimiento, manejo y manipulación de un tractor no tripulado con control a distancia, usando tanto hardware como software para el desarrollo de las rutinas de movimiento, utilizando como kit de desarrollo el Microcontrolador Rabbit 4400W, actuadores, drives, sensores, y servomotores.

### **1.1. Antecedentes.**

En los últimos tiempos, el estudio, análisis y desarrollo de los robots, ha sido parte importante en el desarrollo de la tecnología de cada país. En el término robot confluyen las máquinas para la realización de trabajos productivos e imitación de movimientos y comportamientos de seres vivos.

Los robots actuales son obras del desarrollo del estudio, investigación e ingeniería y como tales son diseñadas para producir bienes y servicios, o exportar recursos naturales. Desde esta perspectiva son máquinas que desarrollan una actividad que parte de los propios orígenes de la humanidad, y que desde el comienzo de la edad moderna se fundamenta esencialmente en conocimientos científicos y desarrollo tecnológico.

Desde la antigüedad, el hombre ha externado su fascinación por las máquinas que imitan la figura y los movimientos de los seres vivos. Existe una larga producción de autómatas en el mundo, desde los automotores de los artesanos franceses y suizos del siglo XVIII, que incorporaban dispositivos mecánicos para control automático de movimientos, hasta nuestro siglo, con los actuales robots que cantan, bailan, caminan, corren, y andan en bicicleta.



---

El esquema básico de un robot, cuenta con un sistema mecánico, eléctrico, electrónico, actuadores, sensores, programación y de control. [1] **¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia.**

Hay diferentes tipos de robots, dependiendo de su concepción o aplicación.

- Robots manipuladores (brazos).
  - Cilíndricos.
  - Esféricos.
  - Cartesianos.
  - Mixtos.
- Robots móviles.
  - Tipo y número de ruedas.
- Robots híbridos (móviles con manipulación).
- Vehículos autónomos:
  - Tele operados.
  - Repetitivos.
  - Autónomos.
- Robots caminantes.
  - 2 patas (*humanoides*).
  - 4/6 patas (*insectos*) .

En la sección del Estado del Arte, se desarrollarán estos temas.



## 1.2 Justificación.

Los robots se consideran como máquinas que integran un conjunto de componentes de diferentes disciplinas como es el caso de la mecánica, eléctrica y electrónica, así como de las comunicaciones; que al integrarse se les incorporan sistemas informáticos que se encargan del control y administración de los recursos para comprensión e interacción con el medio ambiente donde serán usados.

A finales del siglo XX, se desarrollaron máquinas que favorecían las actividades del trabajador en la industria, así como la procuración de su salud e integridad física, es decir, se desarrollaron máquinas que realizan actividades que el ser humano venía realizando pero que ponía en riesgo su vida o integridad física y donde el hombre difícilmente puede llegar, como es caso de la exploración espacial; por mencionar algunas: en lugares difícilmente accesibles, como en la minería y la arquitectura, para evitar que se expongan a sustancias tóxicas, como en la agricultura cuando se desarrolla la actividad de fumigación, en la medicina para manipular con precisión los instrumentos quirúrgicos, entre otras. La característica fundamental de estas máquinas, es que son operadas de forma remota, las cuales, para que formen parte del grupo considerado como robots, es necesario que contengan tal autonomía suficiente, que les permita trabajar de forma independiente.

Por no dejar de lado otras aplicaciones de la robótica, como es el caso de los robots diseñados para dar servicio doméstico, ayuda a discapacitados, entre otros, se menciona que se están realizando avances de gran importancia, tanto que se han logrado desarrollos de androides que simulan hasta los gestos humanos y



robots que muestran la apariencia de algún ser vivo, así como los desarrollos que se emplean para ofrecer servicios de recreación; dado que no es la meta de la presente aportación, se salen de los objetivos de este trabajo. Cabe aclarar que las aportaciones descritas en el cuerpo del presente trabajo, pueden servir como complemento en el desarrollo de cualquiera de los robots mencionados anteriormente.

Durante los últimos años los aspectos relacionados con la robótica denominada móvil han cobrado una gran importancia. Los robots móviles a diferencia de los robots manipuladores convencionales es que no están anclados, sino que por el contrario pueden desplazarse por el terreno, por el agua o incluso volar libremente. Los aspectos más específicos de la robótica móvil son los relacionados con tales desplazamientos autónomos y de navegación del robot.

Es por ello que se hace esta propuesta de tesis, la cual nos permitirá desarrollar una tecnología propia y de bajos costos, además del entendimiento de los diferentes tipos de dispositivos móviles, arquitectura, locomoción y aplicación.

### **1.3 Objetivos.**

Objetivo Principal:

- Control de navegación en superficies planas, evitando obstáculos, de un tractor no tripulado.

Objetivos Particulares:

- Diseño de red inalámbrica para comunicación y control.
- Diseño de drivers para circuito PWM (Pulse Wide Modulate).



- Desarrollo de software para control de los mecanismos para la navegación.
- Desarrollo de las rutinas de activación del sensor detector de obstáculos.

#### 1.4 Organización del Documento.

En el capítulo 2, **Estado del Arte**, se presenta un panorama del desarrollo de los robots, así como la descripción de los diferentes tipos de robots, arquitecturas y configuraciones. También se describen las redes inalámbricas y sus configuraciones.

En el capítulo 3, **Propuesta del Prototipo**. Se describe el sistema a controlar, las partes que lo componen, así como los lenguajes de programación y las características de cada parte.

En el capítulo 4, **Desarrollo de Rutinas de Trabajo**. Se muestra el sistema global de navegación, se presenta como se fueron generando los módulos que componen el sistema, así como sus diagramas de flujo, programas y diagramas esquemáticos de cada parte del sistema.

En el capítulo 5, **Disquisición Experimental**. Se presenta una descripción detallada de cada parte del sistema y se hace un recuento puntual de cada parte que lo conforma, así como la problemática que se fue presentando para la realización de esta tesis; de igual forma se describe como se fueron resolviendo estas problemáticas del sistema.

En el capítulo 6, **Conclusiones**. Se presentan las conclusiones a las que se llegaron, así como las propuestas de trabajos a futuro.





---

## Resumen

Los vehículos no tripulados, pueden ser usados para realizar tareas en lugares difícilmente accesibles, con riesgo de accidentes, en condiciones peligrosas para la salud, o trabajos que resultan difíciles por el tamaño de los objetos que es necesario manipular. Por mencionar algunas aplicaciones donde se requiere el uso de vehículos no tripulados, tenemos el caso de la exploración espacial, exploración en el fondo del mar, mantos petroleros, zonas volcánicas, como apoyo al trabajador en la minería, la construcción, así como en la agricultura para fumigación, entre otros.

Como una aportación a esta disciplina, en el presente trabajo se desarrollan y describen los elementos necesarios para el entendimiento, manejo y manipulación de un tractor no tripulado con control a distancia, usando tanto hardware como software para el desarrollo de las rutinas de movimiento, utilizando como kit de desarrollo el Microcontrolador Rabbit 4400W, actuadores, drives, sensores, y servomotores.



---

## Abstract

The unmanned vehicles can be used to carry out tasks in hardly accessible places, at the risk of accidents, in dangerous conditions for the health, or works that are difficult by the size of the objects that are necessary to manipulate. To mention some applications where the use of vehicles nonmanned is required, we have the case of the reconnaissance space, oil exploration, mantles at bottom of the sea, volcanic zones, like support to the worker in the mining, the construction, as well as in fumigation applied to agriculture, among others.

Like a contribution to this discipline, in the present work the necessary elements for the understanding, handling and manipulation of a tractor nonmanned with remote control are developed and described, using as much hardware as software for the development of the movement routines, using Rabbit 4400w Microcontroller development kit, actuators, drives, sensors, and actuators.



## 2. ESTADO DEL ARTE

Haciendo una revisión bibliográfica, se encontró que la palabra “Robot” se usó por primera vez en una obra teatral en 1920, la obra tenía por título "Los Robots Universales de Rossum" (Ver Fig. 2.1.) escrita por el checo Karel Capek.



**Fig. 2.1. Robot “RUR” de 1920.**



---

Según La Real Academia Española, el robot se define como: “*Máquina o ingenio electrónico programable, capaz de manipular objetos y realizar operaciones antes reservadas solo a las personas*”. Y la robótica se define como: “*Técnica que aplica la informática al diseño y empleo de aparatos que, en sustitución de personas, realizan operaciones o trabajos, por lo general en instalaciones industriales.*”

**¡Error! No se encuentra el origen de la referencia.** Por lo que, según la bibliografía revisada, un sistema robótico puede describirse, como “*Un sistema que tiene la capacidad de recibir información, procesar dicha información, comprender su entorno a través de sensores, ejecutar y controlar o supervisar su operación*”.

También encontramos que los robots industriales con los que se cuenta actualmente, se componen principalmente de brazos articulados. Según la definición del Robot Institute of America: “*un robot industrial es un manipulador programable multifuncional diseñado para mover materiales, piezas, herramientas o dispositivos especiales, mediante movimientos variados, programados para la ejecución de distintas tareas*”.

La definición de la Organización Internacional de Normas (ISO): “*Un robot industrial es un manipulador automático reprogramable y multifuncional, que posee ejes capaces de agarrar materiales, objetos, herramientas mecanismos especializados a través de operaciones programadas para la ejecución de una variedad de tareas*”.

Según la literatura actual, fue en el año de 1960 cuando surge la Robótica como una nueva tecnología. Aunque ya pasaron más de 48 años de su surgimiento, ésta se puede considerar como una tecnología en proceso de maduración, es decir, aún cuando se han logrado aportaciones relevantes, es posible seguir desarrollando e innovando.



La robótica es una materia multidisciplinaria y se apoya en gran medida en los progresos de la electrónica, computación, control y mecánica, así como en diferentes aspectos del procesamiento computacional, tales como el reconocimiento de patrones, posicionamiento global, inteligencia artificial, radar y transferencia de información.

En las historias de robots de Isaac Asimov, (Fig. 2.2) éste prevé un mundo futuro en que existían reglas de seguridad para que los robots no puedan ser dañinos para los seres humanos. Por tal razón Isaac Asimov dentro de estas historias propuso las siguientes tres leyes para la robótica:

- *Un robot no puede dañar a un ser humano o, a través de la inacción, permitir que se dañe a un ser humano.*
- *Un robot debe obedecer las órdenes dadas por los seres humanos, excepto cuando tales órdenes estén en contra de la primera ley.*
- *Un robot debe proteger su propia existencia siempre y cuando esta protección no entre en conflicto con la primera y segunda ley.*



Fig. 2.2. Isaac Asimov, en 1965.



## 2.1. Clasificación de Robots.

Un sistema robótico está compuesto de otros sistemas, que al integrarse para conformar el robot por completo, se consideran como subsistemas, los cuales pueden ser:

- Proceso.
- Planeación.
- Control.
- Sensores.
- Eléctrico.
- Mecánico.

A su vez los subsistemas se integran por otros sistemas como es el caso del subsistema de software que se compone de los sistemas (rutinas) de planeación, para interactuar con los sensores y controlar los servomecanismos.

El Subsistema de Proceso, incluye las tareas que lleva a cabo el robot, el medio ambiente en el cual es colocado y la interacción entre ambos. Antes de que un robot pueda realizar una tarea, ésta debe ser buscada dentro de una secuencia de pasos establecidos previamente.

El Subsistema de Planeación, el cual incluye los datos, parámetros e información del entorno, estos parámetros se pueden obtener de una amplia variedad de sensores. Con estos datos y parámetros, se puede planear la adecuada ejecución de una tarea ya establecida.

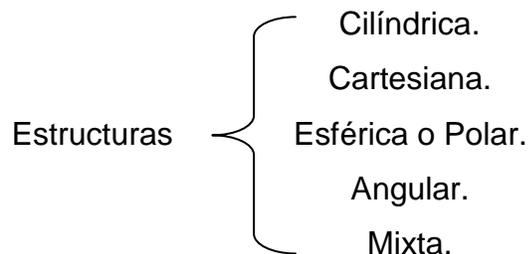


El Subsistema de Control, se usa como una especie de transductor que interpreta las instrucciones para actuar físicamente en los servomecanismos, sensores, entre otros; se usan en lazo cerrado, lo cual permite una retroalimentación de la información emitida por los sensores y con ello actuar en consecuencia, según el fenómeno a controlar.

El Subsistema Mecánico, se encarga de dar fuerza y estabilidad para trabajar en el medio ambiente al que se incorporará el robot, con el fin de realizar una tarea determinada. Comprende las estructuras y partes físicas, que contiene el robot.

### 2.1.1. Robots Manipuladores.

Los robots manipuladores o brazos mecánicos se clasifican dependiendo de su estructura, los tipos de estructuras que se encuentran en la literatura son los siguientes:



A continuación se detallan brevemente cada tipo de estructura. **¡Error! No se encuentra el origen de la referencia. ¡Error! No se encuentra el origen de la referencia.**

### 2.1.1.1. Cilíndrica.

Este tipo de robots manipuladores incorporan generalmente una base giratoria con el primer segmento capaz de plegarse o deslizarse hacia arriba o hacia abajo, como se observa en la figura 2.3, lleva horizontalmente otro segmento que también se puede desplazar hacia dentro o hacia afuera. Son muy simples de representar y bastante intuitivos, son difíciles de manipular con eficiencia porque requieren dos segmentos de movimiento lineal. En un diseño básico, el código de control es bastante simple, es decir, solo se requiere controlar el ángulo de la base, altura del primer segmento, y la extensión del segundo segmento.

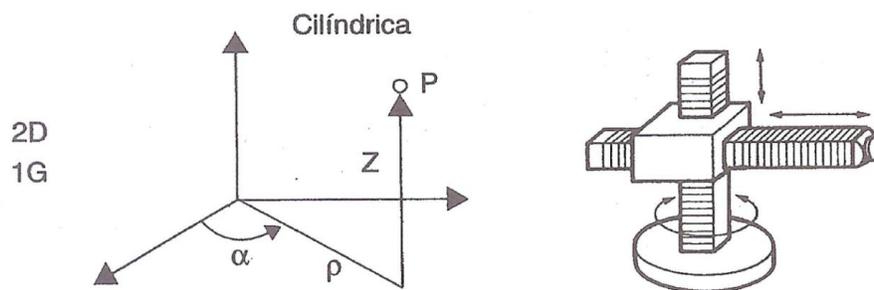


Fig. 2.3. Estructura Cilíndrica.

### 2.1.1.2. Cartesiana.

Esta estructura se basa en un sistema coordinado tridimensional de XYZ. Es muy usada en la industria, puesto que son empleadas para el transporte de cargas voluminosas y pesadas ver figura 2.4. Hay dos presentaciones industriales básicas basadas en cómo se apoyan los segmentos del brazo: pórtico y puente transversal (o cantilever). Tiene la ventaja de requerir un algoritmo de control muy simple, que simplifica la tarea del controlador del robot, que debe generar las ordenes para ejecutar una trayectoria definida mediante una secuencia de puntos expresados en coordenadas cartesianas, es decir, las variables articulares corresponden directamente a las coordenadas que toma el extremo del brazo.

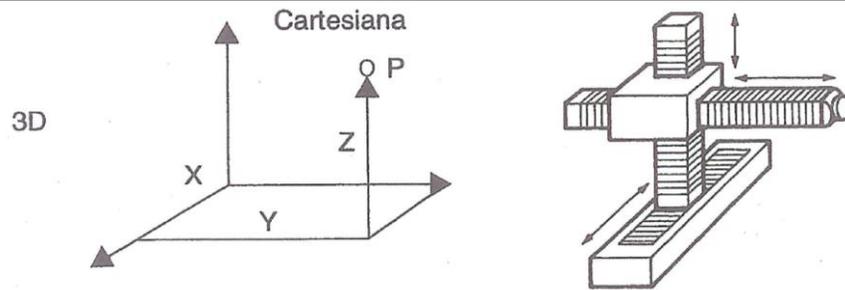


Fig. 2.4. Estructura Cartesiana.

### 2.1.1.3. Esférica o Polar.

Es una de las estructuras más versátiles, ya que permite un buen volumen de trabajo. Hay varias formas de presentación del tipo de brazo de trabajo. El más básico tiene una base giratoria en la cual lleva un segmento del brazo que puede mover hacia arriba y hacia abajo, así como extender hacia fuera o hacia adentro, como se muestra en la figura 2.5.

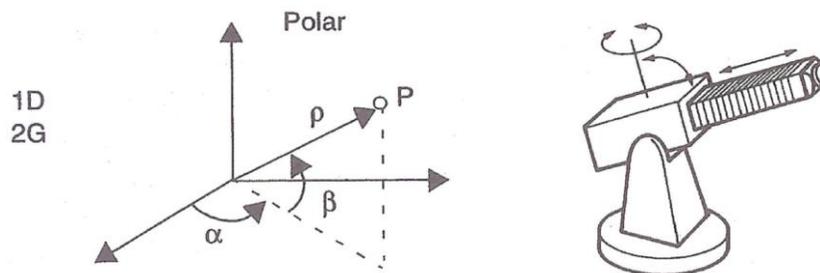


Fig. 2.5. Estructura Polar.

### 2.1.1.4. Angular.

Esta estructura tiene 3 articulaciones de rotación o tres grados de libertad, a diferencia de las otras estructuras, esta presenta mejor acceso a espacios cerrados como se muestra en la figura 2.6, es la configuración mas empleada en robots manipuladores industriales. De hecho, esta estructura es de las más utilizadas en educación y actividades de investigación y desarrollo tecnológico.

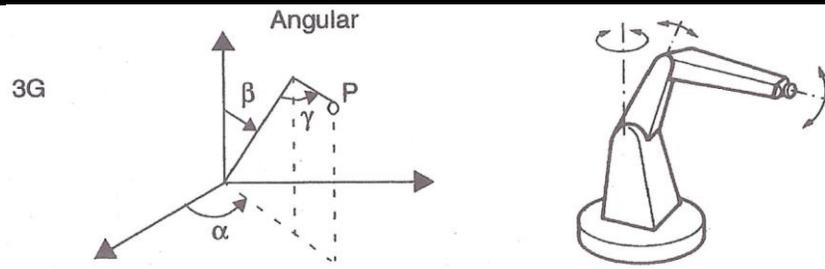


Fig. 2.6. Estructura Angular.

### 2.1.1.5. Mixta.

Poseen varios tipos de articulación, como los SCARA (*Selective Compliant Articulated Robot Arm*). Los robots SCARA son diseñados para realizar tareas en un solo plano, generalmente en el plano horizontal, son usando en la industria para el montaje de piezas.

### 2.1.3. Robots Móviles.

Robots diseñados para explotar el movimiento entre grandes distancias y terrenos sinuosos, se incorporan a una estructura que tiene ruedas o mecanismos que favorecen su desplazamiento, es decir, sistemas de locomoción. La forma en que se conducen, puede ser en forma remota o autónoma vía sistemas de video o por la interpretación del medio ambiente vía sensores.

Más adelante se presenta una clasificación y descripción más detallada.



---

#### **2.1.4. Androides.**

Son robots que intentan reproducir total o parcialmente la forma y el comportamiento cinemático del ser humano. Actualmente los androides son dispositivos con utilidad de demostración, y en los casos más evolucionados, en las tareas de secretariado, recepción y guía de turistas entre otros, y son destinados fundamentalmente al estudio y experimentación.

Uno de los aspectos más complejos de estos robots, es el de la locomoción bípeda, en este caso, el principal problema es la estabilidad y el equilibrio del robot, al momento de caminar, girar, subir o bajar escaleras.

#### **2.1.5. Zoomórficos.**

Este tipo de robots incluye a todos aquellos que integran sistemas de locomoción similares a los aportados por la naturaleza en los seres vivos. Se dividen en dos grupos: los que se desplazan imitando las diversas formas de caminar de los seres vivos como arañas, robots bípedos, por mencionar algunos y los que se desplazan imitando otras formas de desplazamiento como del tipo de reptiles, peces, entre otros.

### **2.2. Tipos de Robots Móviles.**

Un alto porcentaje de éste tipo de robots para moverse de un punto a otro, usan un medio de locomoción que incluye ruedas; lo anterior se justifica por la simplicidad que amerita su control y el beneficio que se obtienen de la estabilidad que presentan, además oponen menor resistencia por unidad de movimiento, lo que permite obtener grandes velocidades, a diferencia de los que usan extremidades. Una característica que los demerita es que este tipo de robots se desenvuelven con dificultad en terrenos sinuosos.



Los robots móviles, utilizan en su mayoría cuatro ruedas posicionadas de forma tal, que al unir los puntos de cada rueda se puede trazar una figura geométrica rectangular. Una posible limitación de este tipo de robots es su maniobrabilidad, ya que ésta se limita a la necesidad de movimiento hacia adelante para poder dar vuelta o en su caso adaptarlos con la velocidad de reversa. Otra limitación surge con la necesidad de contar con un mecanismo de suspensión en cada rueda, de lo contrario no se asegura que se toque el suelo durante su movimiento. Siguiendo con este orden de ideas, se debe considerar en el diseño de la tracción de las ruedas, que al avanzar en línea recta, las cuatro ruedas deben girar con sincronía, es decir a la misma velocidad, cosa que no sucede cuando el robot debe girar para dar vuelta; en este caso, las ruedas de la parte interna giran en una velocidad menor que las ruedas de la parte externa.

Se recomienda para el buen diseño mecánico de un robot móvil, considerar los requerimientos antes mencionados, así como adecuarlos con un sistema de control de velocidad, dirección y suspensión independiente, en cada rueda. Con ello se evitará imprecisión a la hora de exigir al robot una trayectoria definida.

De acuerdo con los grados de autonomía, [1] se sugiere que los robots pueden ser clasificados como sigue:

- Tele Operados.
- Repetitivos.
- Autónomos.



---

### 2.2.1. Tele Operados.

Se le llama tele operación, cuando se controla un dispositivo a distancia, con la ayuda de sensores y actuadores; a dichos dispositivos se les conoce como robots tele operados, ya que con la ayuda de estos sensores y actuadores se incrementa la autonomía del sistema remoto. El sistema remoto requiere de una participación significativa del operador, tanto para su navegación, como para la toma de decisiones y supervisión del sistema. Con esta intervención se cierra el lazo de control.

Dentro de la tele operación existen tres tipos de control, clasificándose por el grado de intervención del operador del sistema.

- Control Supervisado. El operador del sistema decide, planea y ejecuta las tareas y actividades de un sistema semiautónomo de control, incluyendo métodos de simulación, planificación y ayuda a la toma de decisiones del operador empleando sensores, programación y sistemas electrónicos.
  
- Control Compartido. Como su nombre lo indica, es una fusión entre el nivel de autonomía y el grado de tele operación involucrados en una determinada función.
  
- Control Negociado. En este tipo de control se hace una selección de entre los niveles de autonomía y tele operación para realizar el control de una determinada función.

En la actualidad existen sistemas tele operados que cuentan con sensores que retroalimentan a un operador o usuario con información de distancia, temperatura,



---

humedad, entre otros. En el trabajo que se presenta, se diseña el control de navegación para uso en este tipo de robots, con el objetivo de apoyar en las actividades que pongan en riesgo la salud del operario, tal como en actividades de la construcción, minería, agricultura, entre otras.

Las dificultades de estos robots radican en las limitaciones que tiene el operador para el acoplamiento y coordinación con el sistema, además de la precisión de la conducción. En algunas aplicaciones y dependiendo de la tecnología que se ocupa para la transmisión de datos, es importante ver los factores que pueden provocar errores en la transmisión y recepción de datos ya sea por el retraso de transmisión de información, interferencia o pérdida de señal.

### **2.2.2. Repetitivos.**

Son la mayor parte de los robots que se emplean en la producción industrial, trabajan generalmente en tareas con coordenadas ya establecidas, dentro de ambientes de trabajo predecibles e invariantes, y con una limitada o nula percepción del entorno. Son precisos, fuertes y relativamente rápidos; incrementan la productividad ahorrando al hombre trabajos repetitivos, y eventualmente peligrosos.

### **2.2.3. Autónomos.**

Son los robots más evolucionados por el alto procesamiento de la información, y conocimiento de su entorno. Son capaces de percibir, modelar el entorno, planificar y decidir sin la intervención de supervisores. Pueden trabajar en entornos poco estructurados y dinámicos, pueden cambiar de tareas de ejecución dependiendo el cambio del entorno.



En lo que respecta a un robot móvil, en relación con la autonomía, principalmente se enfoca en la autonomía en su navegación; por lo tanto se deben acondicionar con subsistemas de planificación de la ruta a seguir, ya sea la más corta o la más rápida, ya que no siempre la distancia más corta se recorre más rápido, otro subsistema necesario es el de la percepción del entorno, así como el subsistema de control de la navegación. Lo anterior se requiere para que se justifique su calidad de autónomo, es decir, que pueda tomar decisiones sin la intervención de un operario.

Los problemas surgen por la necesidad de una elevada capacidad del procesamiento requerido para el conocimiento del entorno, planeación y ejecución de las tareas, además de la incertidumbre que provoca un entorno no estructurado.

### **2.3. Tipos de Locomoción Móvil.**

La mayoría de los robots móviles usan ya sea ruedas o pistas de deslizamiento para moverse. Las ruedas conforman la base del vehículo, en dicha base se soporta el equipo e instrumentos necesarios para dotarlo de desplazamiento, como por ejemplo sensores, actuadores y equipo de emisión y recepción de información. Un dilema que se genera dentro de los robots móviles, es la necesidad de dotarlos con dispositivos que produzcan y almacenen la energía requerida para su funcionamiento; cuando se tiene una red eléctrica conectada al robot móvil se restringe el movimiento pero se tiene energía ilimitada. A diferencia de los que cuentan con un sistema de respaldo de energía por baterías, estos robots tienen un libre movimiento pero son limitados por la cantidad de energía



---

que puedan almacenar dentro de estas baterías además que requieren de comunicación inalámbrica para su supervisión o control.

La movilidad de los robots con locomoción generalmente se hace mediante ruedas o pistas de deslizamiento. Las ruedas se supone están montadas en una base que permite el movimiento y desplazamiento entre su punto de anclaje y una superficie (el suelo). Para los robots móviles se cuenta con algunas suposiciones:

- El robot está construido con mecanismos rígidos.
- No más de un anclaje por cada rueda.
- Todos los ejes de anclaje son perpendiculares al suelo.
- La superficie de movimiento es un plano.
- No hay deslizamiento entre las ruedas y el suelo.
- La fricción es suficientemente pequeña como para permitir el giro de cualquier rueda.

Todas estas suposiciones se cumplen generalmente para la mayoría de los robots móviles.

Los vehículos móviles con ruedas, son generalmente más eficientes y simples de controlar para superficies planas y sin obstáculos, son usados tanto para interiores como para exteriores. Existen diferentes tipos de locomoción, dependiendo de su concepción original, algunas son modificaciones de vehículos ya establecidos, como automóviles o triciclos, y otros son modificaciones de estos mismos. También tienen características diferentes en cuanto se refiere a la aplicación y maniobrabilidad. A continuación se muestran los diferentes tipos de locomoción más comunes en robots móviles. **¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia.**



### 2.3.1. Ackerman.

Esta es la configuración más utilizada, ya que resulta de la modificación de vehículos de cuatro ruedas como automóviles, tractores, vehículos todo terreno (Ver la figura 2.7.). El mayor problema de la locomoción Ackerman es la limitación en las maniobras ya que se requiere que este en movimiento para poder dar vuelta.

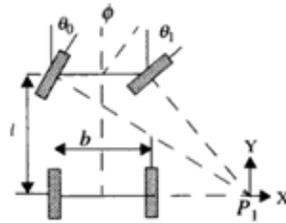


Fig. 2.7. Sistema Ackerman.

Este tipo de locomoción es el que se considero en el diseño del sistema que se presenta en esta tesis, en el capítulo siguiente se verá en forma más precisa, el tipo de tractor que se pretende usar en un trabajo a futuro, para adaptarle el diseño propuesto en este trabajo de tesis, así como sus características técnicas.

### 2.3.2. Triciclo Clásico.

Este sistema de locomoción tiene dos tipos de configuración, una con la rueda delantera con tracción y direccionamiento; y en el otro caso la rueda delantera tan solo es de direccionamiento y las traseras tienen la tracción. La maniobrabilidad es mayor, aunque también se requiere de estar en movimiento para dar vuelta, pero presenta problemas de estabilidad en terrenos difíciles y en vueltas muy cerradas. Es frecuente que se use en vehículos robóticos para interiores y exteriores



pavimentados y bien estructurados. Este sistema se muestra en la siguiente figura 2.8.

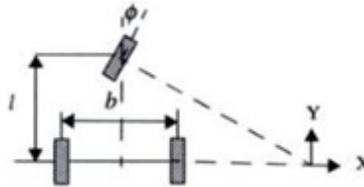


Fig. 2.8. Sistema de Triciclo Clásico.

### 2.3.3. Direccionamiento Diferencial.

El control de la dirección es el resultado de la diferencia de velocidades de las ruedas laterales, y la tracción se obtiene cuando las ruedas laterales giran a la misma velocidad y en el mismo sentido ya que estas ruedas generalmente van montadas sobre motores de corriente continua. Esta configuración es muy usada en robots para ambientes estructurados, y donde la superficie de trabajo sea plana. Para el soporte y estabilidad pueden ocupar una o más ruedas libres para su soporte. En la figura 2.9, se ilustra el sistema de locomoción.

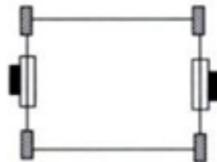


Fig. 2.9. Direccionamiento Diferencial.

### 2.3.4. Skid Steer.

En este sistema, la locomoción se basa en el direccionamiento diferencial, pero con la característica de que se incrementa el número de ruedas en cada lado del vehículo. La dirección al igual que el sistema anterior, se basa en la diferencia de velocidades de cada lado del robot; y la tracción en coordinar estas velocidades.



En la figura 2.10, se muestra el “Terregator”, un vehículo desarrollado por el Robotics Institute de la Carnegie Mellon University.



**Fig. 2.10. Robot Terregator.**

### **2.3.5. Pistas de Deslizamiento.**

Esta locomoción es más conocida como del tipo oruga, tanto la tracción como el direccionamiento se consigue por la combinación de las velocidades de cada oruga o pistas de deslizamiento. Este tipo de locomoción es útil en navegación en terrenos irregulares, en los cuales presenta un buen rendimiento. Este sistema ha sido muy usado en aplicaciones militares, y su uso es más frecuente en tanques de guerra.

### **2.4. Estabilidad.**

Un arreglo de ruedas adecuado proporciona estabilidad; para una estabilidad adecuada existen tres características fundamentales para el diseño de un dispositivo móvil: maniobrabilidad, control y estabilidad.

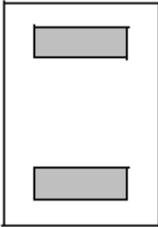
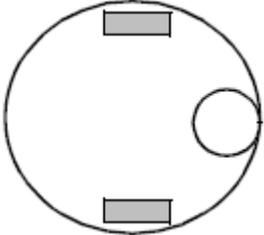
El número mínimo de ruedas requeridas para la estabilidad es de dos. Convencionalmente, la estabilidad estática requiere un mínimo de tres ruedas, con un centro de gravedad que debe de estar dentro del triángulo formado por los puntos de contacto de las ruedas. La estabilidad puede mejorarse agregando más



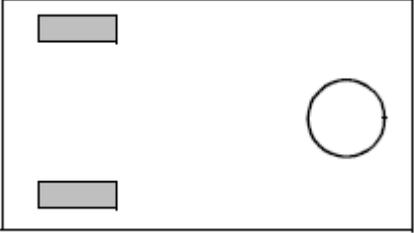
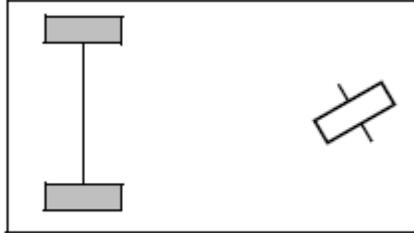
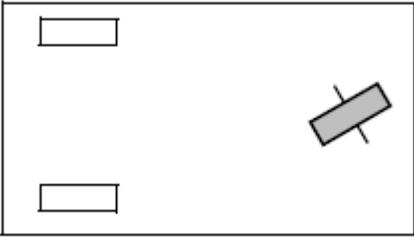
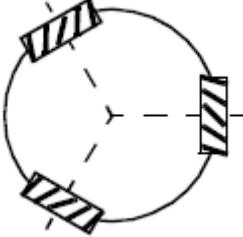
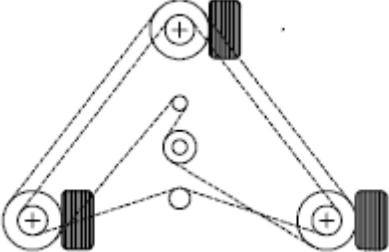
ruedas, aunque una vez que el número de puntos de contacto excede de tres ruedas, requiere de una cierta forma de suspensión flexible para el terreno que se encuentre desigual. ¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia.

La siguiente tabla (Ver tabla 2.1.) muestra los diferentes arreglos y números de llantas que se encuentran para robots móviles.

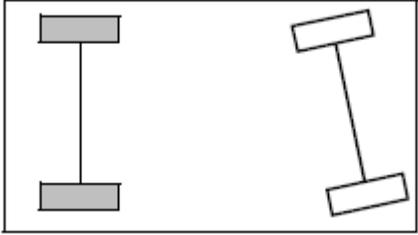
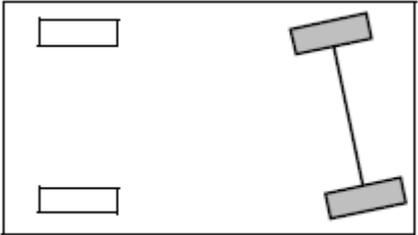
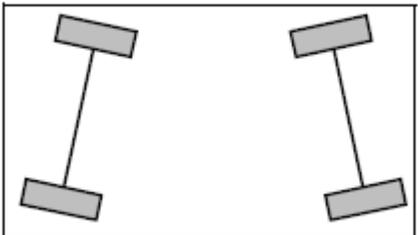
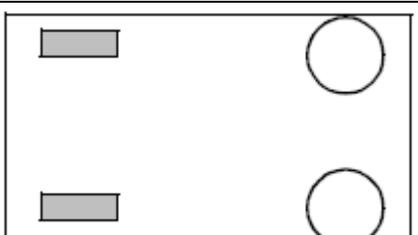
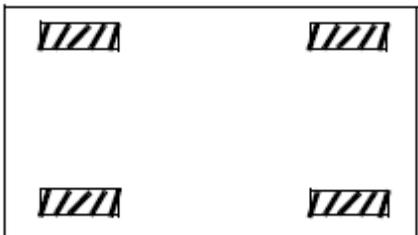
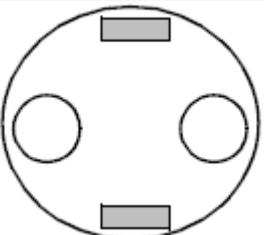
Tabla 2.1. Arreglos de ruedas

No. de Ruedas.	Arreglo.	Descripción.
2		Una rueda de dirección al frente, una rueda de tracción en la parte posterior.
		Impulso diferenciado de las dos ruedas con el centro de masa debajo del eje.
		Dos ruedas centrales de impulso diferencial, con un tercer punto del contacto.



3		Dos ruedas con tracción independiente al frente o atrás y 1 rueda omnidireccional atrás o al frente.
		Dos ruedas conectadas con tracción diferencial en la parte posterior, 1 rueda libre de dirección al frente.
		Dos ruedas libres en la parte posterior, 1 rueda con tracción de dirección al frente.
3		Tres ruedas motorizadas suecas o esféricas dispuestas en un triángulo; con posible movimiento omnidireccional.
		Tres ruedas motorizadas síncronas y ruedas dirigidas; la orientación no es controlable.



4		Dos ruedas motorizadas en la parte posterior, y 2 ruedas de dirección al frente.
		Dos ruedas motorizadas y con dirección al frente y, 2 ruedas libres en la parte posterior.
		Cuatro ruedas con dirección y motorizadas.
		Dos ruedas de tracción (diferencial) en atrás/frente, 2 ruedas omnidireccionales frente/atrás.
4		Cuatro ruedas motorizadas omnidireccionales.
		Tracción diferencial de dos ruedas con 2 puntos adicionales de contacto.



		Cuatro ruedas motorizadas y con dirección.
6		Dos ruedas motorizadas y con dirección alineadas al centro, 1 rueda omnidireccional en cada esquina.
		Dos ruedas de tracción (diferencial) en el centro, 1 rueda omnidireccional en cada esquina.

## 2.5. Generaciones de Control.

La clasificación de los robots basados en sistemas de control se puede dividir en 5 generaciones.

- a) Primera generación: Esta generación es conocida como control de lazo abierto. Los dispositivos son mecánicos.
- b) Segunda generación: También son de lazo abierto, los dispositivos ya tienen una secuencia de control de movimientos los cuales están almacenados en cinta magnética o procesos mecánicos repetitivos.



- 
- c) Tercera generación: Se utilizan computadoras y se tiene conocimiento del entorno a través del uso de sensores, aparecen lenguajes de programación para escribir los programas de control. Aquí inician los robots inteligentes. El control se denomina de lazo cerrado.
  
  - d) Cuarta generación: Los robots cuentan con más inteligencia, resultado de una gran gama de sensores. Estos sensores permiten la supervisión del entorno; y cuenta con un mayor procesamiento computacional.
  
  - e) Quinta generación: Actualmente sigue en desarrollo esta nueva generación de robots, la cual busca que el control venga de modelos de conducta y de cambios del entorno en forma dinámica. [1][2]

Las características con las que se clasifican principalmente son:

- a) Propósito o función.
- b) Sistema de coordenadas empleado.
- c) Número de grados de libertad del efecto formal.
- d) Generación del sistema de control.

La clasificación basada en su propósito o función:

- a) Industriales.
- b) Personales / Educativos.
- c) Militares.



---

## 2.6. Red Inalámbrica.

Una red se puede definir como un conjunto de dispositivos que están conectados y pueden tener comunicación entre sí, ya sea en forma física, o en forma inalámbrica.

Existen tres tipos de redes que se van clasificando por la extensión.

- Redes LAN. Las redes LAN (*Local Area Network*) tienen una extensión máxima de un edificio. Son de las redes más usadas.
- Redes MAN. (*Metropolitan Area Network*), son redes que conectan a dos o más redes LAN, las cuales se encuentran en una distancia no mayor a la misma ciudad.
- Redes WAN (*Wide Area Network*), estas redes se extienden más allá de una ciudad, no tienen un límite de extensión geográfica.

El protocolo TCP/IP está basado en el modelo OSI (*Open System Interconnection*). El protocolo TCP/IP conjunta las capas superiores del modelo OSI (Sesión, Presentación y Aplicación) en una sola capa llamada de Aplicación. Por lo tanto el protocolo TCP/IP se encuentra formado por 5 capas.

1. Física.
2. Enlace.
3. Red (IP).
4. Transporte (TCP).
5. Aplicación.



---

### Capa Física.

Está compuesta por los elementos físicos que comprenden la red, cables, tarjetas de red, entre otros.

### Capa de Enlace.

Se encarga de mantener una buena transmisión y recepción entre los dispositivos de la red, con lo cual añade controles de flujo, y gestión de errores para prevenir colisiones de los paquetes que se envían.

### Capa de Red (IP).

El protocolo IP (*Internet Protocol*) es un protocolo que se encarga de que la información llegue a su destino. Esta información es enviada a través de *switches*, *routers* que conforman una red. Para saber a quién se le entregará esta información, se utiliza un identificador que es llamado dirección IP. Esta IP está formada por cuatro números de 8 bits en decimal y separados por un punto. Con esta dirección IP, se da certeza de saber a quién se le entrega esta información. Existen dos tipos de dirección IP: Estática (fija) y Dinámica; estas IP's se suelen asignar mediante un servidor de DHCP (*Dynamic Host Configuration Protocol*).



Como se muestra en la siguiente tabla, (Ver tabla 2.2.) existen tres clases de direcciones IP.

**Tabla 2.2. Clases de Direcciones IP.**

CLASE	RANGO
A	1.0.0.0—126.0.0.0
B	128.0.0.0—191.0.0.0
C	192.0.0.0—223.0.0.0

Se utiliza una máscara de subred para que los dispositivos de la red sepan a cual clase pertenece la IP, y así también saber si el destino de la información está dentro de la misma red LAN o está fuera de ella. La máscara de subred cuenta con 32 bits y al igual que las direcciones IP, esta dividida por puntos.

Capa de Transporte (TCP).

El protocolo TCP (*Transmission Control Protocol*), es el que se encarga de que la información que se entregará, llegue de forma correcta al destinatario, es capaz de detectar errores en la información que se envía, o la pérdida de estos y en consecuencia puede nuevamente pedir que sean reenviados.

Dentro del TCP existe el uso de puertos, estos puertos nos permiten tener una comunicación concreta con ciertos servicios, ya sea *web*, *telnet*, *ftp*, correo electrónico (*pop*, *smtp*). Este puerto tiene una longitud de 16 bits por lo que puede tener valores de 0 a 65,535, pero dentro del rango de 0 a 1024 son reservados, y son para los servicios ya descritos estos puertos son denominados “puertos bien conocidos”.



---

## Capa de Aplicación.

En esta capa es donde se implementa los servicios al usuario como por ejemplo, la transferencia, administración y control de archivos, el correo electrónico y los servicios de Internet.

Las redes de área local, que se conectan en forma inalámbrica, se le conoce como WLAN, (*Wireless Local Area Network*), existe una asociación que se encarga de definir el estándar *Wi-Fi*, y certificar aquellos dispositivos que cumplen dicho estándar, esta asociación es la *Wi-Fi Alliance*. **¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia.**

Dentro de las ventajas que ofrecen las redes *Wi-Fi* podemos encontrar las siguientes:

- Movilidad.
- Fácil instalación.
- Exteriores.
- Disponibilidad.

Aunque una de las grandes desventajas que se tienen con las redes inalámbricas es la inseguridad. Para tratar de darle seguridad a la información se desarrolló primero una Autenticación, es decir, antes de ser agregado a la red y poder intercambiar información comprueba la identidad del dispositivo y posteriormente la autenticación abierta que no requiere contraseña o cifrado alguno.



Existen tres tipos de cifrado. Se desarrollo un algoritmo de cifrado llamado WEP (*Wired Equivalency Protocol*), este algoritmo ofrece un nivel básico de seguridad, ya que inclusive en la actualidad existen programas para poder encontrar el algoritmo de cifrado y así poder conectarse a la red. También existe otro algoritmo llamado WPA (*Wireless Protected Access*), sus mejoras son la integración de claves tipo TKIP (*Temporal Key Integral Protocol*), y la autenticación de usuarios se hace mediante el protocolo EAP (*Extensive Authentication Protocol*). El tercero es el WPA2 este se basa en algoritmos AES (*Advanced Encryption Standard*), fue definido bajo el estándar 802.11i

El estándar IEEE 802.11 (*Institute of Electrical and Electronics Engineers*), es un estándar que se diseño para especificar la conectividad entre los diferentes dispositivos inalámbricos dentro de una red de área local. El término 802.11 se refiere a una familia de protocolos, incluida la especificación original. En la siguiente tabla 2.3, se muestra el estándar y sus componentes. **¡Error! No se encuentra el origen de la referencia.**

Tabla 2.3. Estándar IEEE 802.11

802.11	Especificación para transmisiones inalámbricas e infrarrojas, en la banda de 2.4GHz.
802.11a	Banda de 5GHz, a 54 Mbps de tasa de transferencia.
802.11b	Banda de 2.4GHz, a 11 Mbps de tasa de transferencia.
802.11c	Nivel de enlace entre redes.
802.11d	Dominios normativos múltiples.
802.11e	Calidad de Servicio ( <i>QoS Quality of Service</i> ).
802.11f	Protocolo de inter-punto de acceso ( <i>IAPP, Inter-Access Point Protocol</i> )
802.11g	2.4GHz 54 Mbps



802.11h	Selección Dinámica de Frecuencia ( <i>DFS, Dynamic Frequency Selection</i> ) y Control de Potencia de Transmisión ( <i>TPC, Transmit Power Control</i> ).
802.11i	Seguridad ( <i>WPA2</i> ).
802.11j	Canales de 5GHz japoneses (4.9 a 5.1 Ghz).
802.11k	Medición.
802.11m	Mantenimiento.
802.11n	Alta velocidad.

Dentro de las topologías de redes inalámbricas, se pueden clasificar básicamente en dos.

- Conjunto de Servicio Básico (BSS).
- Conjunto de Servicio Básico Independiente (IBSS).

Conjunto de Servicio Básico (BSS).

Este modo es conocido como modo infraestructura. De este tipo de conexión los equipos que conforman la red, se conectan a un punto central, conocido como *Access Point*, este se encarga de dar las direcciones IP, autenticar al usuario y dar los servicios que se requieran.

Ventajas.

- Mayor radio de cobertura.
- La red puede contener un mayor número de equipos.
- Autenticación antes de compartir información.



- Se puede compartir una conexión de Internet para todos los equipos que pertenecen a la red.
- Se pueden compartir recursos entre los usuarios de la red (impresoras, almacenamiento, información, entre otros).

### Conjunto de Servicio Básico Independiente (IBSS).

Este conjunto es conocido como *Ad-hoc*, es la configuración más básica y permite que exista comunicación entre dos o más computadoras en forma directa, ya que no usa un *Access Point* para la comunicación. Es usada para compartir información entre dos equipos, es importante dentro de esta configuración tener en cuenta el canal de transmisión.

Esta configuración es usada en la transmisión de datos para el sistema que se propone a controlar. En el siguiente capítulo se darán más detalles de la propuesta del prototipo.

### Ventajas.

- Facilidad de configuración.
- Movilidad.
- Rapidez.
- No necesita de un Access Point. **¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia.**

Hasta aquí hemos visto el desarrollo y clasificación de los diferentes tipos de robots, así como las partes importantes de locomoción, estabilidad y control. Esto nos permite saber qué tipo de locomoción y sistemas usar en la propuesta del prototipo, así como también las propuesta que se harán como uso de



### 3. PROPUESTA DEL PROTOTIPO

En una primera etapa del desarrollo del prototipo, y para justificación del presente trabajo, se diseñaron, simularon e implementaron a nivel de laboratorio, las rutinas, algoritmos y módulos de puesta en marcha, navegación y detección de obstáculos, con supervisión a distancia, es decir, se utilizó una computadora para enviar y recibir datos, una red *Wi-Fi* en modo *Ad-hoc*, para tener comunicación entre el usuario y el tractor, esta red inalámbrica *Wi-Fi (IEEE 802.11b)*, en forma embebida en un tarjeta de desarrollo con un microcontrolador de la marca Rabbit, modelo RCM4400w, que se encarga de hacer, tanto la comunicación como el control de los diferentes dispositivos que contiene el sistema.

El microcontrolador entre otras cosas, tiene el control de los diferentes motores, drivers, sensores, para la navegación, seguridad y comunicación. En una segunda etapa, fuera de los alcances del presente trabajo, se propone usar el control y el método de navegación diseñados en un sistema de control electrónico autónomo, y realizar pruebas en campo.



---

El sistema de navegación está compuesto por: la puesta en marcha del motor de combustión interna (a gasolina), de tracción trasera y un grado de libertad del eje delantero en el plano horizontal de 30 grados, que le permitirá girar hacia la derecha e izquierda, así como en reversa, la palanca de velocidades, la cual nos da la posibilidad de ir hacia el frente o hacia a tras, y el freno, dicho sistema de navegación se diseño y fue validado en el laboratorio con una tarjeta de experimentación, que contiene dispositivos como leds, un servomotor, sensor de distancia entre otros. Para la simulación, puesta en marcha y evaluación del diseño, se desarrollo una pantalla interactiva con el paquete de diseño LabVIEW, más adelante se comentará a detalle el desarrollo de cada una de las partes mencionadas anteriormente.

El sistema de seguridad estará dado por un sensor ultrasónico de la marca DEVANTECH Ltd. Modelo SRF05, este sensor tiene una capacidad de rango de 2 centímetros a 4 metros (el cual se describirá más adelante), que nos indicará si hay un objeto y la distancia en la cual se encuentra, para poder cambiar la trayectoria del tractor y así evadir este objeto, tanto para la seguridad del mismo objeto como del tractor.



En la figura 3.1, se muestra el diagrama a bloques, del sistema que se propone controlar, así como las partes que lo componen.

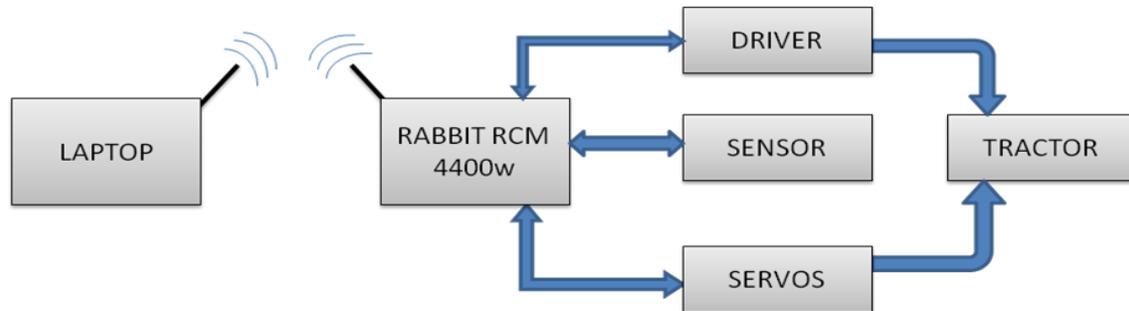


Fig. 3.1. Diagrama del Sistema.

### 3.1. Lenguajes de Programación y Programas Adicionales.

Con el objetivo de contar con un lenguaje de programación robusto, estable y confiable para las aplicaciones que se desarrollaron, se opta por ocupar dos tipos de lenguajes, uno que es *Dynamic C*, el cual es desarrollado por la empresa *Z-World*, para microcontroladores Rabbit, y otro *LabVIEW (Laboratory Virtual Instrument Engineering Workbench)* propiedad de *National Instruments* para desarrollar la interfaz gráfica del usuario. El microcontrolador Rabbit que se utiliza en el presente trabajo tiene como característica, que dentro de los diversos microcontroladores que se encuentran en el mercado, este microcontrolador cuenta con una antena de red inalámbrica en forma embebida tipo *Wi-Fi*, que cumple con las certificaciones de *IEEE 802.11b*.



---

### 3.1.1. Dynamic C.

Dynamic C, es un sistema de desarrollo integral para productos basados en microcontroladores *Rabbit*. Este programa se utilizó para desarrollar la programación tanto de la comunicación como del control. Este programa tiene la capacidad de:

- Editar.
- Compilar.
- Programar.
- Depurar.

Dynamic C, también soporta la programación en lenguaje de ensamblador. No es necesario dejar C o el sistema de desarrollo para escribir código en lenguaje ensamblador. C y el lenguaje ensamblador se pueden mezclar.

El depurador de *Dynamic C* incluye la capacidad de utilizar comandos del *printf*, expresiones *watch* y puntos de depuración. *Dynamic C* introduce características de depuración avanzadas tales como ejecución y pila de trazo. La pila de trazo se puede utilizar para seguir la ejecución de las declaraciones en el depurador, incluyendo la información tal como función/nombre de fichero, línea del código fuente y los números de columna, acción realizada, fecha/hora de la acción realizada y contenido del registro. La pila del trazo demuestra secuencias de la llamada a función y valores de parámetro.

*Dynamic C* proporciona extensiones de lenguaje C (tal como variables, co-declaraciones y co-funciones compartidas y protegidas), así como también soporta trabajos de multitarea. [6]



#### Características del editor:

- Facilidad para construir un texto.
- Sintaxis con código de colores.
- Los programas se pueden ejecutar y depurar ya sea en código fuente, o a nivel ensamblador.
- Menús y atajos de teclado están disponibles para más comandos.
- Soporta lenguaje de programación para sistemas embebidos.

#### Características del Compilador:

- *Dynamic C* puede compilar ya sea directamente a memoria, o puede compilar un archivo \*.bin.
- El programador serial, puede hacer descargas de 460 Kbps (típico de 115.2 kbps).
- Función de encadenamiento de segmentos de código para ser embebidos en múltiples funciones.
- Extensiones de Lenguaje C, que proveen compartir o proteger variables, co-declaraciones y co-funciones.
- Librerías incluidas en código fuente.
- Adaptador Serial a USB.

#### Características del Depurador:

- Expresiones *watch*.
- Puntos de depuración (*breakpoints*).
- Ejecución por pasos, dentro y fuera de las funciones a nivel código fuente o lenguaje ensamblador.



- Ventana de Registros. Despliega todos los registros y banderas del procesador.
- Combina lenguaje C y lenguaje Ensamblador.
- Modifica registros y variables.
- Ventana de Pila. Muestra el contenido de la parte superior de la Pila.
- Memoria Hex. Despliega el contenido de cualquier dirección de memoria.
- Ventana STDIO. La salida *printf* se visualiza en esta ventana, cualquier entrada desde el teclado puede detectarse para propósitos de depuración. La salida *printf* también puede ser enviada a un puerto serial o a un archivo.

### 3.1.2. LabVIEW.

El paquete de programación *LabVIEW (Laboratory Virtual Instrument Engineering Workbench)* propiedad de *National Instruments*, nos permitirá crear una interfaz gráfica, amigable entre la Laptop y el usuario. Con este programa, se estableció la comunicación, entre la laptop y el microcontrolador. Así como el envío y recepción de la información de control.

El software de LabVIEW es una herramienta donde se realiza programación gráfica. Está dividido en dos ventanas principales, una llamada Panel Frontal, la cual es donde una vez que se ejecuta el programa es lo que verá el usuario, aquí es donde se presentan botones, indicadores, gráficas y demás instrumentos, tanto de medición como de control. La otra ventana principal es llamada Diagrama a Bloques, en esta ventana es donde se realiza la programación de los indicadores, botones e instrumentos que se muestran en el Panel Frontal.



---

Esta programación, soporta entre otras cosas, comunicación vía TCP/IP, con lo que nos permite crear un servidor de dicho protocolo, para que usando el microcontrolador como cliente, se comuniquen entre ambos programas. **¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia.**

### **3.2. Sistema a Controlar.**

El sistema o dispositivo a controlar es un tractor tipo podadora, el cual permite tener una estabilidad y maniobrabilidad adecuadas para el propósito de aplicación del desarrollo del presente trabajo. Cabe aclarar que como alcances del presente trabajo solo se diseñaron, simularon y validaron, el control de navegación y el sistema de seguridad, los cuales fueron validados vía experimental en el laboratorio, dejando como trabajo a futuro su implementación en el tractor tipo podadora. El control va a ser tele-operado, es decir va a ser con un mando a distancia, o a control remoto, desde una computadora que tenga una antena *Wi-Fi*, que mande y reciba información a través de una red inalámbrica del tipo *Ad-hoc*, la cual va a estar comunicada con un microcontrolador Rabbit modelo RCM4400w, que estará montado sobre el tractor. Este microcontrolador, como característica principal tiene un sistema embebido de comunicación *Wi-Fi*, mismo que fue utilizado para el desarrollo del sistema de emisión y adquisición de información que controla los sistemas tanto de navegación, como de seguridad.

#### **3.2.1. Tractor.**

El tractor a controlar es del tipo podadora de la marca *Poulan*, modelo PB1638LT. Tiene un arreglo de 4 ruedas, las ruedas delanteras con dirección y las traseras motorizadas. Como se vio en el capítulo anterior, tiene un tipo de locomoción tipo Ackerman, como se muestran en la figura 3.2.

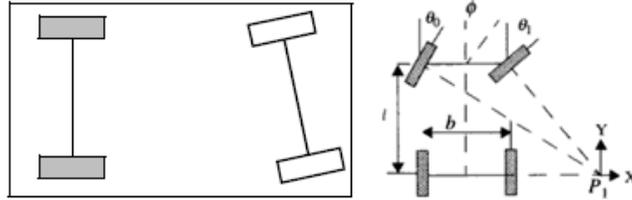


Fig. 3.2. Locomoción Ackerman.

La siguiente imagen (Fig. 3.3.), se muestra físicamente al tractor a controlar.



Fig. 3.3. Tractor a Controlar.



En la siguiente tabla 3.1, se muestran las especificaciones con las que cuenta el tractor. **¡Error! No se encuentra el origen de la referencia.**

Tabla 3.1. Especificaciones del Tractor.

Tanque de gasolina	4.73 l.
Tipo de aceite	SAE 30 (arriba de 32°F). SAE 5W-30 (debajo de 32°F).
Capacidad de aceite	3 l.
Tipo de bujía	Champion RC12YC.
Velocidad (kph)	Frente. 1ra. 1.77 2da. 3.54 3ra. 5.31 4ta. 7.08 5ta. 8.85 Reversa. 2.73
Sistema de carga	3 amp Batería. 5 amp Faros.
Batería	28 amp/hr. 230 min. CCA. Carcasa tipo U1R.
Torque de cuchillas	45-55 FT. LBS.



### 3.2.2. Microcontrolador Rabbit.

Como se mencionó anteriormente, se utilizó un kit de desarrollo marca *Rabbit de Z-World* modelo RCM4400w ya que este sistema tiene en forma embebida una antena *Wi-Fi* 802.11b con la cual, nos permite tener una conexión estable para la comunicación inalámbrica. Soporta modos tipo infraestructura y *Ad-hoc*. Este microcontrolador nos da el recurso de tener ya un sistema con *Wi-Fi* sin la necesidad de instalar una antena externa.

El módulo RCM4400w, incluye 512K de memoria flash, y ejecución en SRAM, 35 líneas de propósito general de entrada y salida, líneas de entrada y salida a 3.3v, tensión de trabajo de 3.3v, decodificador de cuadratura, seis puertos seriales, cuatro PWM, hardware DMA, captura de eventos, *timer*, así como modos de bajo consumo de energía. En la siguiente tabla (Tabla 3.2.) se muestran las características generales de este módulo. [17]

Tabla 3.2. Características de RCM4400w

Característica.	RCM4400w
Microprocesador	Rabbit 4000 a 58.98 MHz.
SRAM.	512K.
Programa de ejecución rápida SRAM.	512K.
Memoria Flash.	512K.
Líneas de propósito general.	35 líneas digitales de I/O configurables con 4 funciones alternativas.
Entradas adicionales.	Modo de arranque, entrada de Reset.



Salidas Adicionales.	Status, Salida de Reset.
Bus auxiliar de I/O.	Puede ser configurada para 8 líneas de datos y 6 de direcciones (compartidas con líneas paralelas de I/O, además de un I/O de escritura/lectura.
Wi-Fi.	802.11b estándar, ISM 2.4 GHz.
Puertos serie.	6 puertos, compatibles con CMOS. <ul style="list-style-type: none"><li>• Los 6 puertos son configurables como asíncronos (con IrDA), 4 como reloj serial (SPI), y 2 como SDLC/HDLC.</li><li>• 1 puerto como reloj asíncrono combinado con puerto de programación.</li><li>• 1 puerto serie combinado con un serial flash.</li></ul>
Tasa Serie.	Máxima tasa de transferencia en modo asíncrono =CLK/8.
Reloj en tiempo Real.	Si.
Timers.	10 timer de 8 bits (6 en cascada desde el primero), 1 timer de 10 bits con 2 registros combinados, y 1 timer de 16 bits con 4 salidas y 8 registros de set/reset.
Watchdog.	Si.
PWM.	4 canales síncronos PWM con un contador de 10 bits.  1 canal síncrono con un contador de 16bits.



Captura de eventos.	2 canales.
Decodificador de cuadratura.	2 canales.
Voltaje.	3.3 V.DC $\pm$ 5% 450 mA @ 3.3 V Transmisión/Recepción. 80 mA @ 3.3 V Standby.
Temperatura de operación.	-30°C a +75°C
Humedad.	5% a 95%
Tamaño de la tarjeta.	1.84" × 2.85" × 0.50" (47 mm × 72 mm × 13 mm)

### 3.2.3. Laptop.

Para tener movilidad al realizar pruebas en campo, se usó un equipo portátil, este equipo como característica y requisito principal, cuenta con una antena inalámbrica, puede ser interna o externa, y cumple los requisitos de una norma IEEE 802.11b/g, con el software de LabVIEW versión 8.0 instalado.



---

#### **3.2.4. Red Inalámbrica**

Parte importante del presente trabajo es la red inalámbrica, se presenta una red inalámbrica en modo *Ad-hoc*, la cual nos permite tener una comunicación punto a punto, entre el microcontrolador y la computadora, esta comunicación tiene como objetivo intercambiar la información y comandos, y así poder tener el control de la navegación del tractor.

Como se vio en el capítulo anterior, una red *Ad-hoc*, no requiere de un Access Point para poder entablar una comunicación entre los dispositivos que cumplen con el estándar de IEEE 802.11. En el caso de la aplicación que se propone, tanto la laptop como el microcontrolador cumplen con dicho estándar.

#### **3.3. Componentes del Sistema.**

Como el objetivo del sistema es el control de la navegación del tractor, se requieren de diversos dispositivos para lograr este objetivo, con el común denominador de que estos dispositivos deben de ser eléctricos y de corriente continua. Ya que, de otro tipo (hidráulico o neumático) modificarían las condiciones de trabajo, movilidad y portabilidad del mismo sistema.



---

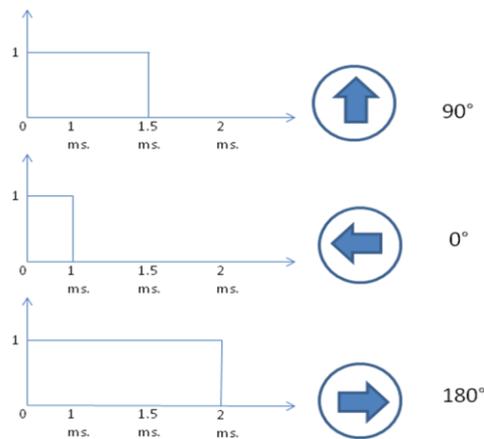
### 3.3.1. Servomotores.

Se utilizaron servomotores tanto para la dirección, como para la palanca de velocidades, ya que a diferencia de los motores de CD, los servos mantienen su posición mientras sigan con energía eléctrica, además, los servos tienen la capacidad de ubicarse en cualquier posición y mantenerse estables en dicha posición, con esto es posible tener absoluto control sobre su eje o flecha de rotación, lo cual nos permite una mayor exactitud en el posicionamiento. Esto nos permitirá tener una dirección del tractor más uniforme ya que al momento de estar circulando la dirección se mantendrá fuerte contra los cambios en el camino, debidos a imperfecciones u objetos diversos que contenga el terreno sobre el que se desplaza. En el ámbito de la palanca de velocidades, permitirá el trabajo del embrague con precisión, ya que como sabemos, si la posición de la palanca no es adecuada, corremos el riesgo de dañar la caja de velocidades.

El motor del servo tiene algunos circuitos de control y un potenciómetro (una resistencia variable) que está conectado al eje central del servo motor. Este potenciómetro, permite a la circuitería de control, supervisar el ángulo actual del servo motor. Si el eje está en el ángulo correcto, entonces el motor está apagado. Si el circuito monitorea que el ángulo no es el correcto, el motor girará en la dirección adecuada hasta llegar al ángulo correcto. El eje del servo es capaz de girar alrededor de los 180 grados. Un servo normal se usa para controlar un movimiento angular de entre 0 y 180 grados. Un servo no es mecánicamente capaz de llegar al ángulo requerido, si hay peso mayor al sugerido en las especificaciones técnicas del fabricante.



El ángulo de giro está determinado por la duración del pulso que se aplica al pin de control. El servo espera un pulso cada 20 milisegundos. La longitud del pulso determinará el giro del motor. Un pulso de 1.5 ms., por ejemplo, hará que el motor se torne a la posición de 90°. Si el pulso es menor de 1.5 ms., entonces el motor se acercará a los 0°. Si el pulso es mayor de 1.5 ms., el eje se acercará a los 180 grados. Un ejemplo se muestra en la figura 3.4.



**Fig. 3.4. Diagramas de Estado.**

### 3.3.2. Actuadores.

Con fines de experimentación, en el sistema desarrollado se realizaron las rutinas de trabajo que permitirán, en la segunda fase del proyecto, usar un actuador para el freno, este actuador, al activarse saldrá un pistón, el cual oprimirá el pedal del freno, este actuador también nos permite mantener el freno activado, (el pedal a fondo) durante el tiempo que sea necesario, ya que mantiene también su posición mientras siga con energía eléctrica. Un ejemplo de estos actuadores se muestra en la siguiente figura 3.5.

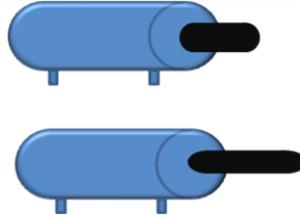


Fig. 3.5. Actuadores.

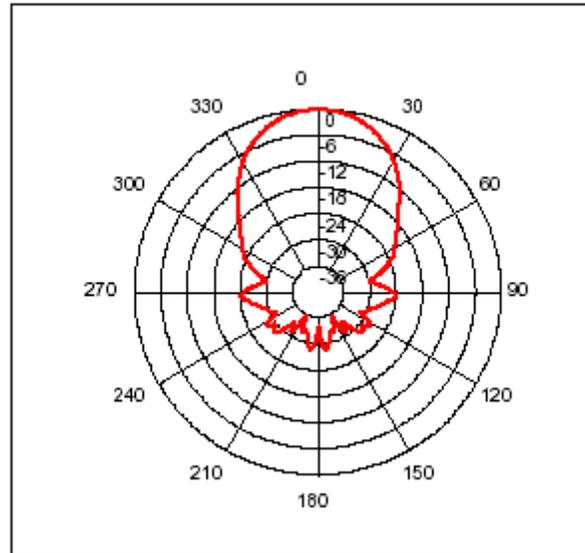
Otros dispositivos que utilizaremos serán SSR (Relevador de Estado Sólido), tanto para el encendido, como el apagado eléctrico del tractor. También los utilizaremos para el actuador del freno. En la siguiente figura 3.6 se muestra una imagen de un SSR.



Fig. 3.6. Relevador de Estado Sólido.

### 3.3.3. Sensor Ultrasónico.

Para mantener una navegación detectando obstáculos, así como para evitar algún accidente, se utilizó un sensor ultrasónico a distancia. El cual al identificar un objeto, envía una señal que es proporcional a la distancia la cual sirve como referencia para modificar la trayectoria de navegación. Una característica importante, considerada al elegir este tipo de sensor, es la de evitar que se pierda la perspectiva del suelo. Este sensor tiene un patrón de radiación que se muestra en la siguiente figura 3.7. **¡Error! No se encuentra el origen de la referencia.**



**Fig. 3.7. Ángulo de Trabajo del Sensor Ultrasónico.**

Por lo tanto, con este sensor podremos percatarnos de algún objeto que este en el camino. Tanto para la seguridad del tractor como del medio en que se desenvuelve.

El sensor que utilizaremos es un ultrasónico de la marca DEVANTECH Ltd, modelo SRF05. El cual nos da un rango de 2 centímetros a 4 metros. (Ver Fig. 3.8)



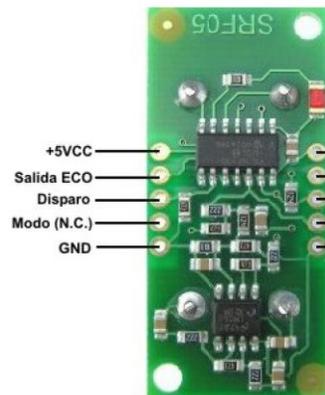
**Fig. 3.8. Sensor Ultrasónico SRF05.**



Las características del sensor se describen a continuación:

- Rango de 2 cm. a 4 m.
- Voltaje de alimentación de 5 v  $\pm$ 10%.
- Interfaz de 4 pines (voltaje, tierra, señal de disparo y eco).
- Indicador LED de medición en progreso.
- Frecuencia de trabajo 40 khz.
- Dimensiones del circuito 43 x 20 x 17.
- Duración mínima del pulso de disparo (nivel TTL) 10  $\mu$ s.
- Duración del pulso eco de salida (nivel TTL) 100-25000  $\mu$ s.
- Tiempo mínimo de espera entre una medida y el inicio de otra 50 ms.

A continuación se muestra la descripción de pines del sensor ultrasónico.



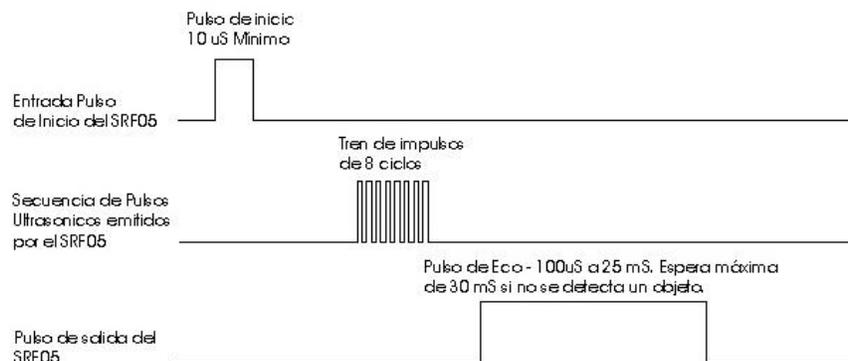
**Fig. 3.9. Descripción de Pines del Sensor SRF05.**

- +5Vcc. Es el voltaje de alimentación.
- ECO. Salida cuyo ancho de pulso es proporcional a la distancia del objeto encontrado.



- Disparo. Entrada de inicio de una nueva medida, cuya duración mínima es de 10  $\mu$ s.
- Modo. Sin conexión en modo 1. Conectado a GND si se requiere el modo 2 de trabajo.
- GND. Tierra de alimentación.

Este sensor tiene dos tipos de modo de funcionamiento. El modo 1, el cual se emplea para la presente tesis y consiste en usar 2 pines del sensor, uno para aplicar el pulso de disparo o *Trigger* y el otro pin para leer el ancho del pulso resultante del ECO emitido. Este modo se selecciona simplemente dejando el pin “Modo” sin conectar. El modo 2 es utilizar el mismo pin para el pulso de disparo como para la lectura del ancho del pulso del ECO.



**Fig. 3.10. Diagrama de Tiempos del Sensor SRF05.**

Como se muestra en el diagrama de tiempos anterior (Fig. 3.10), en el modo 1 de empleo del sensor, se aplica un pulso de disparo o *trigger* de 10  $\mu$ S de duración mínima. El sensor envía un tren de pulsos o “ráfaga” de 8 ciclos a 40KHz. En ese momento la señal de salida ECO pasa a nivel alto, cuando el sensor recibe la señal transmitida como consecuencia de haber rebotado en un objeto (ECO), esta salida pasa de nuevo a nivel bajo. La duración del pulso de esta señal a nivel alto es proporcional a la distancia entre el sensor y un objeto dado.



### 3.3.4. Tarjeta de acondicionamiento de datos.

El microcontrolador maneja solo tensiones de voltaje de 3.3 volts, por lo que debemos de diseñar una tarjeta para adecuar los datos, tanto los que salen como los que entran. Ya que por ejemplo, el sensor maneja voltajes de 5 volts, los servomotores, y actuadores, usados para la experimentación en laboratorio están a 5 volts, los que se integrarán al tractor en la propuesta de trabajo a futuro, serán de 12 volts. Es por ello que se propuso una tarjeta que usa un bus de datos y una etapa de aislamiento que usa SSR, (*State Solid Relay*), además de dispositivos tanto digitales como analógicos para su funcionamiento como un buffer octal 74HCT245, transistores, diodos y resistencias

A continuación se muestra el diagrama esquemático de la tarjeta (ver diagrama 3.1.). Así como el diseño aportado en tarjeta PCB. (Ver Fig. 3.11.)

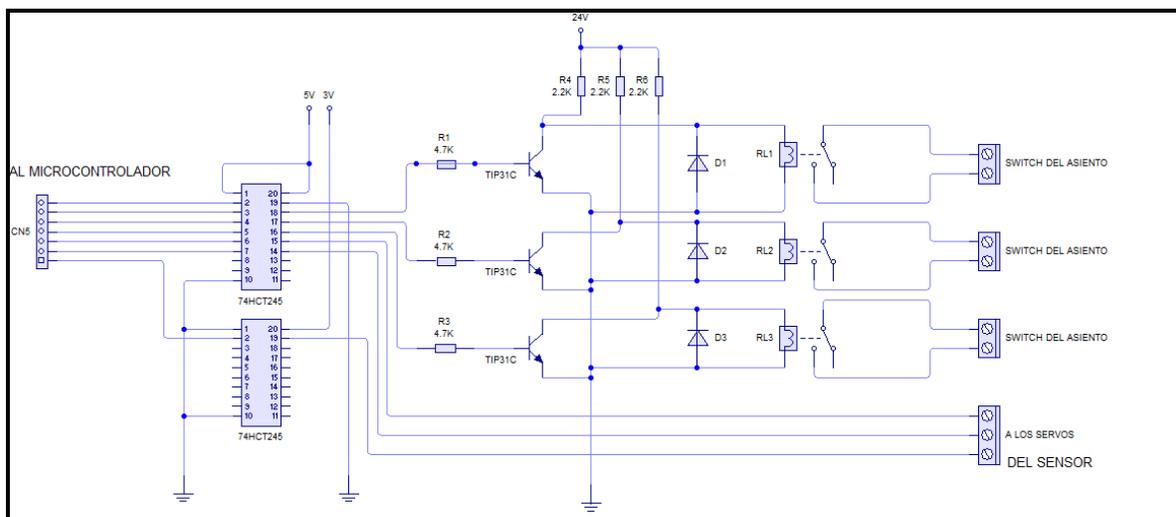
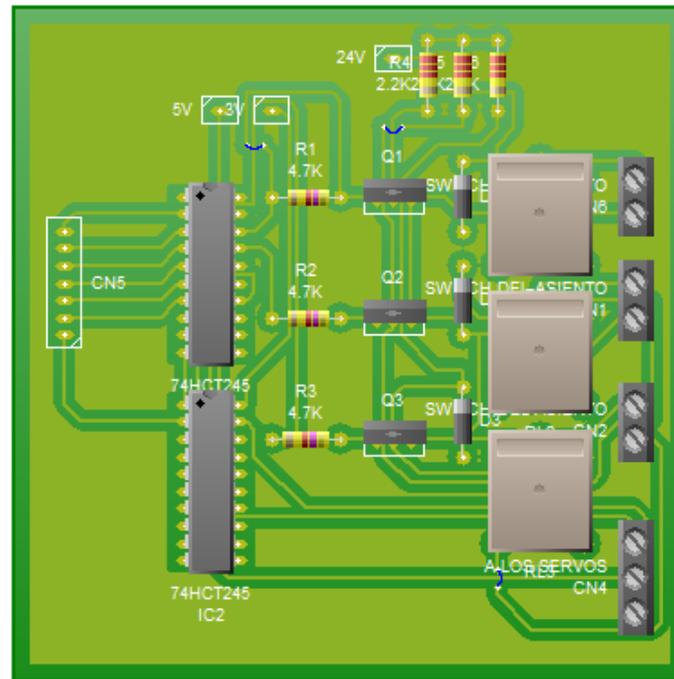


Diagrama 3.1. Diagrama Esquemático de la Tarjeta.



**Fig. 3.11. Tarjeta en PCB**

En este capítulo se describió el sistema del control de navegación y el sistema de seguridad, los dispositivos (actuadores, sensores, servomotor), programas, (Dynamic C, LabVIEW) y herramientas (SSR) con las cuales se va a disponer para llegar al objetivo de la presente tesis. En el siguiente capítulo se presentarán los métodos y programas que se desarrollaron para la comunicación y control de los dispositivos y herramientas.



---

## 4. DESARROLLO DE LAS RUTINAS DE TRABAJO

Para el diseño de los programas y el manejo de los dispositivos periféricos, se fueron atacando diferentes problemáticas que se presentaron para su programación y comunicación. Así como también en el desarrollo de las rutinas de la puesta en marcha del motor de combustión interna y los ciclos de trabajo de la palanca de velocidades.

### **4.1. Metodología de Operación y Control.**

Una herramienta importante para la programación y validación de los diferentes tipos de códigos de programación que se desarrollaron, es hacer diagrama de flujo, para proponer una metodología (algoritmos) de cómo se comportarían esos procesos y como se van realizando. Esto nos va permitiendo tener una idea clara de cómo queremos que se comporte, paso a paso nuestro sistema. A continuación se describe la forma de trabajo de cada uno de los algoritmos propuestos.



## Metodología de forma general.

En esta etapa se describe en forma general cada paso a realizar, se establece como punto principal la conexión en forma remota entre la computadora y el microcontrolador. Se inicializan los mecanismos, y se establece la conexión inalámbrica para recibir el código desde el servidor TCP, este código se va comparando entre los diferentes códigos para realizar tareas específicas. Entre estas diferentes tareas tenemos como encendido o apagado del motor de combustión interna, freno, giro del volante y palanca de velocidades. (Fig. 4.1.)

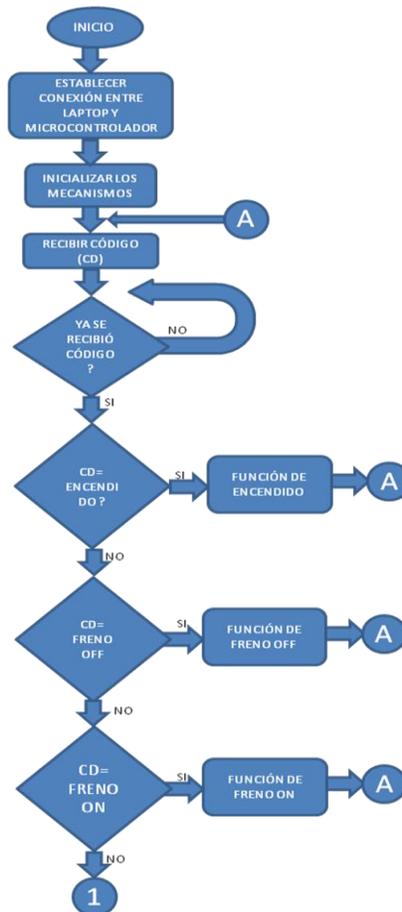


Fig. 4.1. Método General.

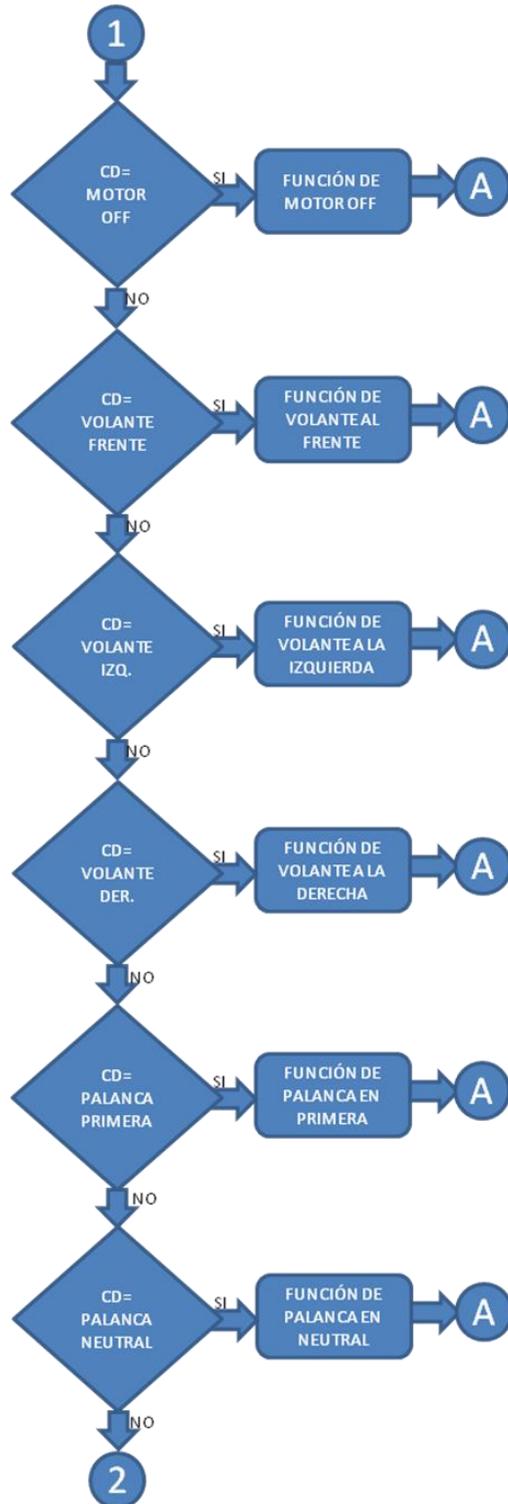


Fig. 4.1. Método General (Continuación).

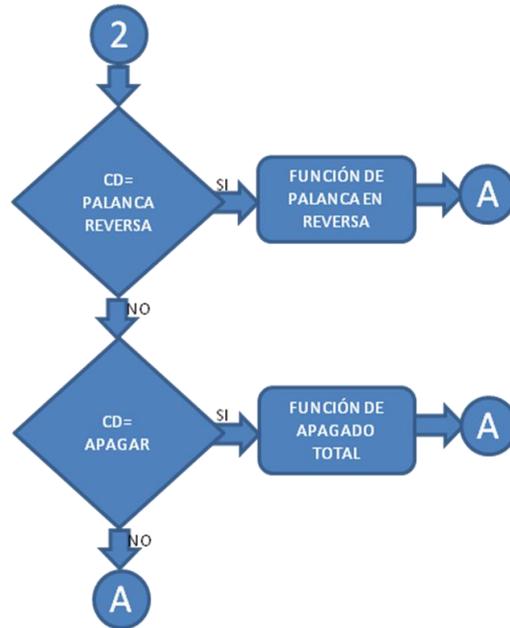


Fig. 4.1. Método General (Continuación).

La función de encendido del motor. En esta función se tiene que tener en cuenta que el tractor tiene un sensor de presencia en el asiento y el freno debe de estar activado, ya con estas dos condiciones se puede encender el *switch* del motor, después del encendido, nos esperamos 2.5 seg. Para el arranque del motor de combustión interna, para después “soltar” el *switch*, ya que como es conocido, si la llave de ignición se deja por un tiempo prologado se puede dañar la marcha del tractor. A continuación se muestra la metodología propuesta. (Fig. 4.2.)



Fig. 4.2. Método de Encendido.

La función del encendido y apagado del freno, nos permite detener o liberar el avance del tractor, esto se hace mediante un actuador que es el que nos proporciona la fuerza necesaria para empujar el pedal del freno a fondo. (Fig. 4.3.)

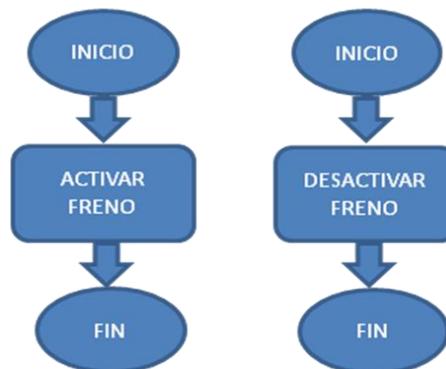


Fig. 4.3. Método de Frenado.



La función de apagado del motor de combustión interna se hace primero parando el movimiento del tractor, accionando el freno y después desactivamos el switch del asiento del mismo tractor, esto hace que la corriente eléctrica que viene de la batería del tractor, deje de circular, provocando el apagado del motor. (Fig. 4.4.)



**Fig. 4.4. Método de Apagado de Motor.**

La función de volante hacia delante en línea recta se hace por medio del ancho del pulso, esto como ya se vio en el capítulo anterior, se realiza ajustando el ancho de pulso a 90 grados, que esto para nosotros será como ajustarlo a 0 grados o en una posición del volante que vaya al frente. (Fig. 4.5.)



**Fig. 4.5. Método de Volante al Frente.**



Función de volante hacia la izquierda se realiza, ajustando el ancho del pulso a 0 grados, o en nuestro tractor sería a -45 grados, esto es porque en forma mecánica, la dirección del tractor no abarca más que 90 grados. (Fig. 4.6.)



**Fig. 4.6. Método de Volante a la Izquierda.**

Función de volante hacia la derecha se realiza llevando el servomotor a una posición de 180 grados, o para nuestro tractor a 45 grados. (Fig. 4.7.)



**Fig. 4.7. Método de Volante a la Derecha.**



En la función de la palanca de velocidades en Neutral, primero detenemos el avance del tractor y luego ajustamos un segundo ancho del pulso a la posición de neutral que ya propiamente tiene la caja de velocidades. (Fig. 2.8.)



**Fig. 4.8. Método de Velocidad en Neutral.**

Función de la palanca de velocidades en primera velocidad. Se requiere saber primero en qué posición está el volante, ya que si no hacemos esta revisión de posición, se puede hacer lento el proceso de avance y a demás podemos dañar la caja de velocidades por un mal manejo del usuario. En este proceso, primero revisamos la posición de la palanca de velocidades, si no está en primera, procedemos a parar el tractor, puesto que no podemos simplemente cambiar la posición del servomotor, ya que se dañaría la caja de velocidades. Después de detener el tractor, cambiamos la posición de la palanca de velocidades a primera usando el ancho de pulso que nos da el microcontrolador. Este proceso se ilustra a continuación. (Fig. 4.9.)

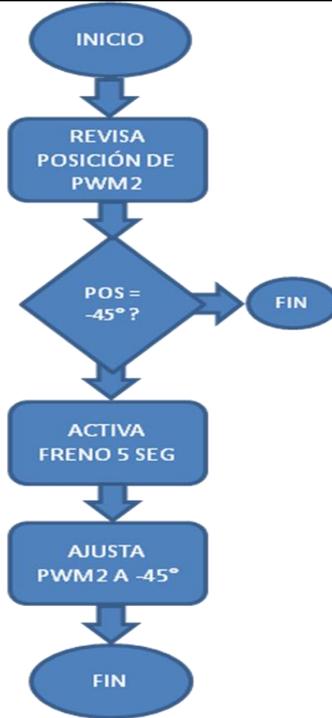


Fig. 4.9. Método de Velocidad en Primera.

Función de la palanca de velocidades en reversa. Es simular el proceso de la palanca de velocidades en primera, tan solo se cambia la posición del servomotor para que se embrague en la posición de reversa. (Fig. 4.10)

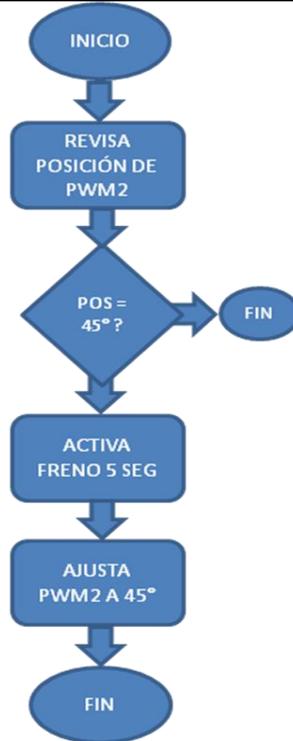


Fig. 4.10. Método de Velocidad en Reversa.

Función de apagado total. En esta función, detenemos tanto al tractor como a la conexión inalámbrica y liberamos el freno. Esto es por si deseamos mover el tractor en forma manual, o para su resguardo. A continuación se detalla esta función. (Fig. 4.11)



Fig. 4.11. Método de Apagado Total.

## 4.2. Transmisión de Datos.

Como se mencionó anteriormente, una de las razones más importantes que justifican el uso del microcontrolador Rabbit RCM 4400w en este proyecto, es que este módulo cuenta ya con una antena *Wi-Fi*, en forma embebida, y con la programación y código necesario para su uso. Esto nos permite bajar los costos, ya que de no tenerlo, hubiéramos tenido que comprar un sistema aparte, ya sea *Wi-Fi*, o por RF (Radio Frecuencia), con las desventaja del costo, programación y configuración. Otra de las ventajas de usar *Wi-Fi*, sobre RF, es el uso de protocolos más fuertes como lo es el TCP/IP, así como ser menos susceptible a interferencias por emisiones de ruido.



Para nuestro sistema hemos optado por una red *Wi-Fi* en modo *Ad-hoc*, entre otras cosas por la portabilidad que nos da, sin la necesidad del uso de un *Access Point*, que necesite energía eléctrica para su funcionamiento. Así, solamente con la energía que nos da la batería de una laptop y la energía que nos proporciona la misma batería del tractor, tenemos cierto rango de independencia.

#### 4.2.1. Red *Ad-hoc*.

A continuación se muestra la forma de crear una red inalámbrica en modo *Ad-hoc* en el sistema operativo de *Windows Vista* propiedad de *Microsoft*. En esta parte también se muestra la forma de darle una IP fija para que sea localizada dentro de la misma red.

Primero hay que abrir el panel de control, posteriormente ubicar el icono de Centro de redes y recursos compartidos, como se puede ver en la siguiente figura. (Fig. 4.12)

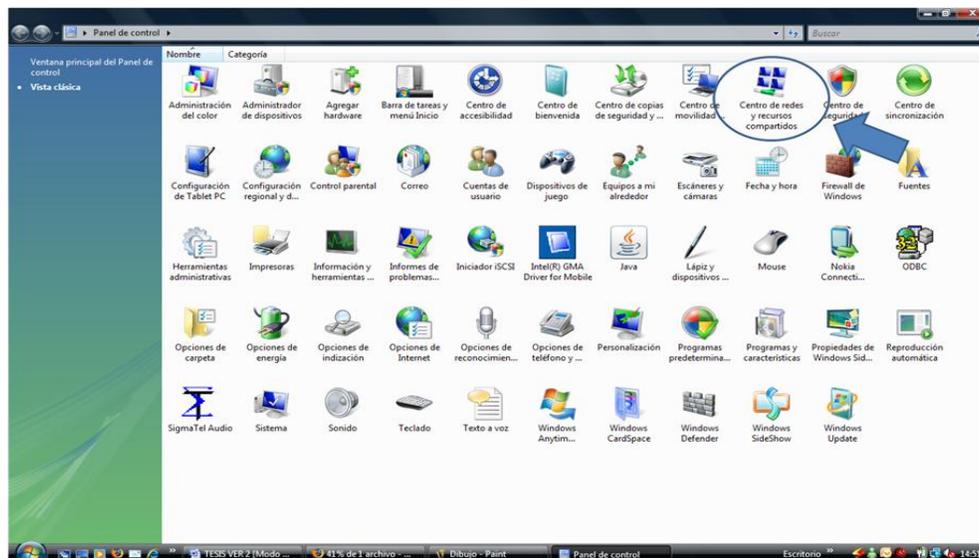


Fig. 4.12 Panel de Control.



Al abrir el Centro de redes y recursos compartidos. Ubicamos la opción de configurar una conexión o red. (Ver Fig. 4.13)

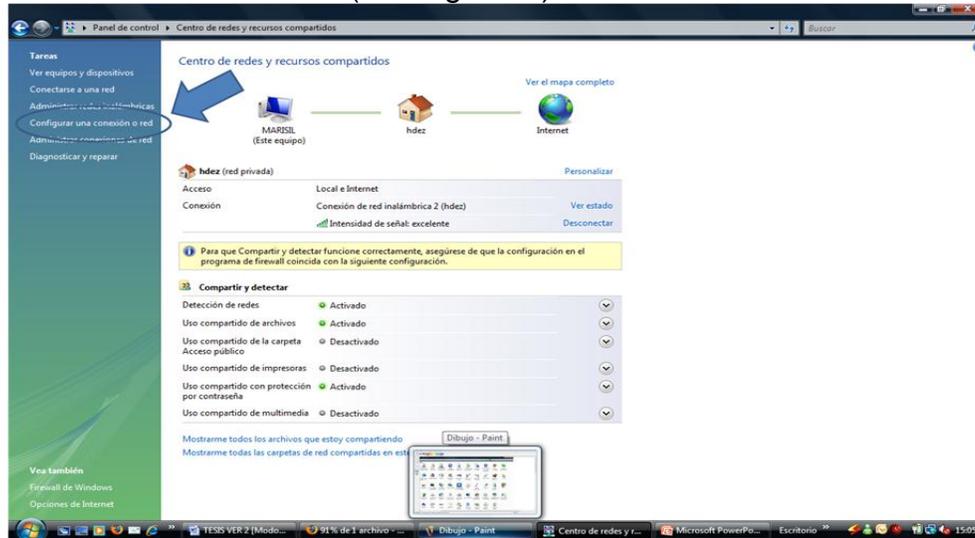


Fig. 4.13. Centro de Redes y Recursos Compartidos.

Abrimos la configuración y se muestra una ventana como la que se presenta en la figura 4.14.

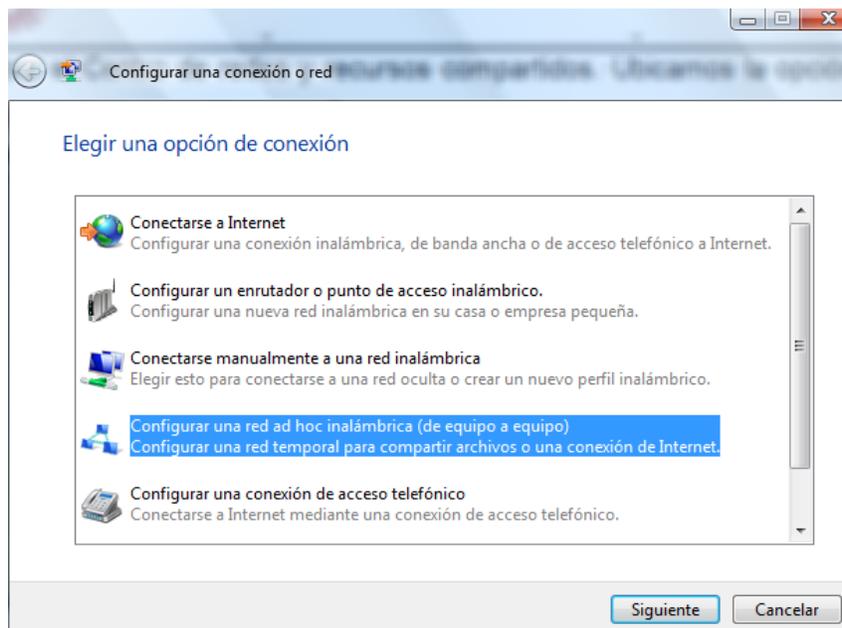
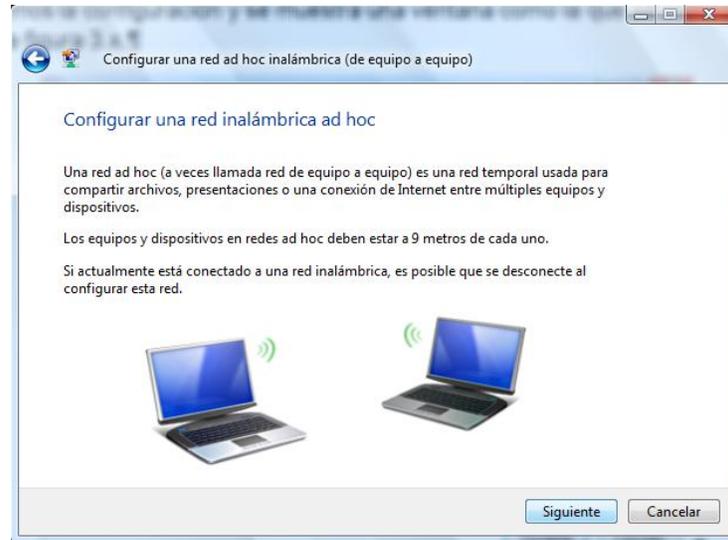


Fig. 4.14. Configuración de Conexión o Red.

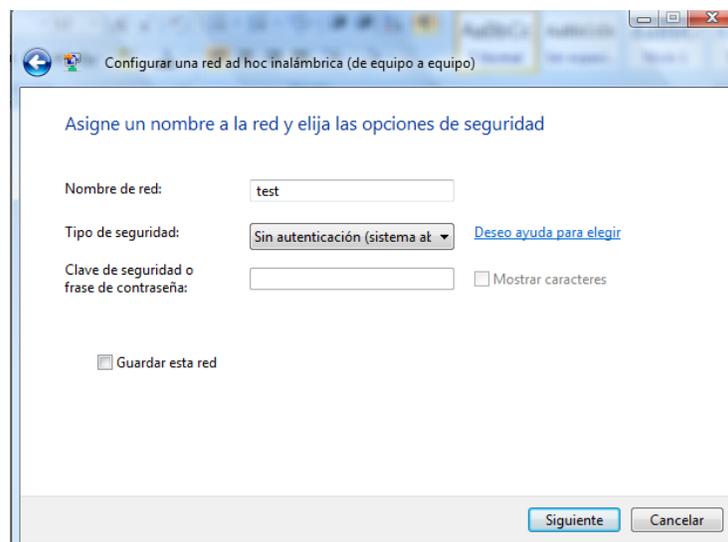


Después de dar clic al botón siguiente, se muestra la pantalla de bienvenida de configuración de una red *Ad-hoc* como se muestra en la figura 4.15.



**Fig. 4.15. Configuración de Red Ad-hoc.**

En esta parte se le da el nombre de la red a la cual se va a conectar, en este caso, le asignamos el nombre de “test”, deshabilitamos la autenticación, y guardamos la red. Como se muestra en la figura 4.16.



**Fig. 4.16. Asignar Nombre y Tipo de Autenticación.**



Después ya tenemos la red *Ad-hoc* lista para poder intercambiar información. (Ver figura 4.17) Aunque en este momento necesitamos configurar la dirección IP del equipo.

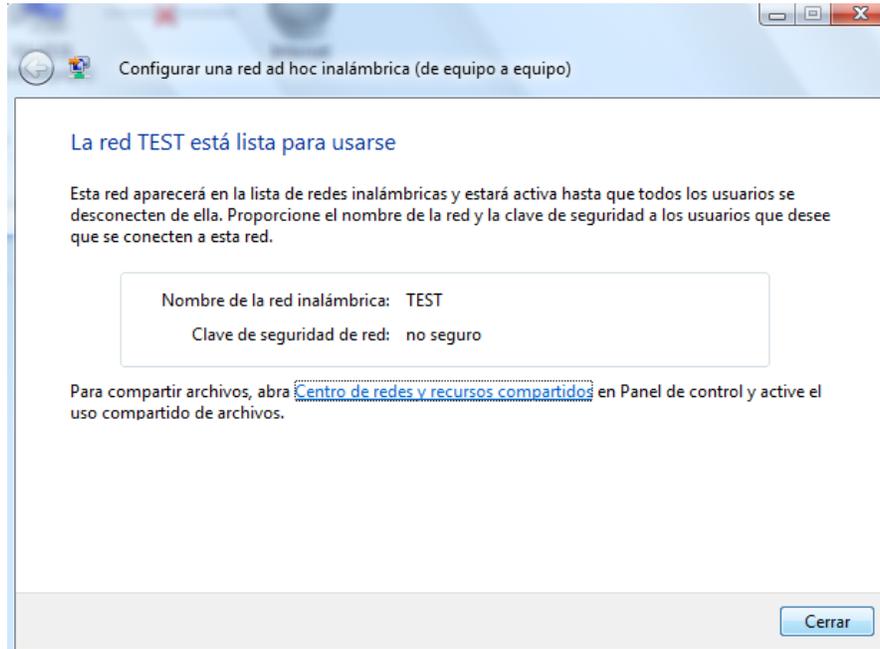


Fig. 4.17. Red *Ad-hoc*.

Para la configuración de la IP en el equipo, se requiere que de nueva cuenta se abra, el panel de control, y después en el Centro de redes y recursos compartidos, pero esta vez nos vamos a la opción de “Administrar conexiones de Red”, como se muestra en la siguiente figura. (fig 4.18)

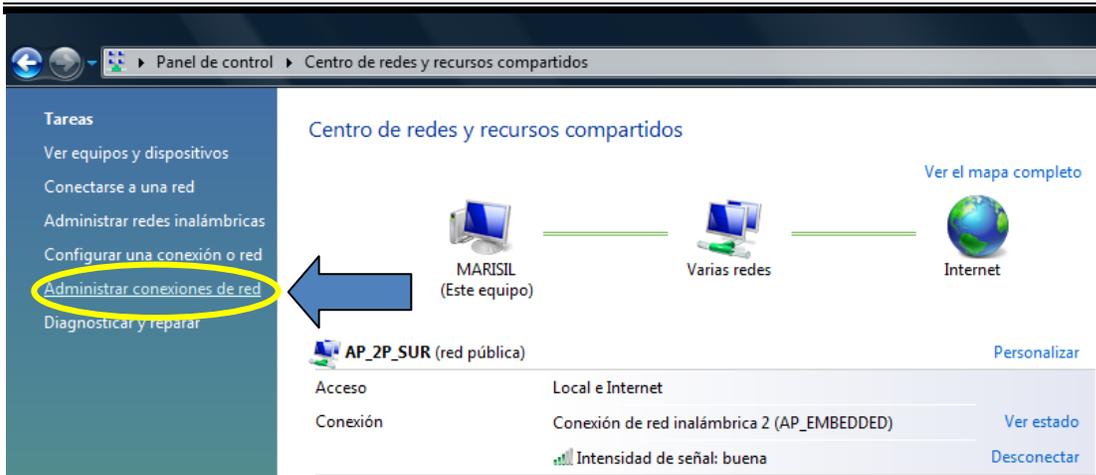


Fig. 4.18. Centro de Redes y Recursos Compartidos.

Al abrir la conexión de redes, se abre la pantalla que se muestra en la figura siguiente. (Ver 4.19)

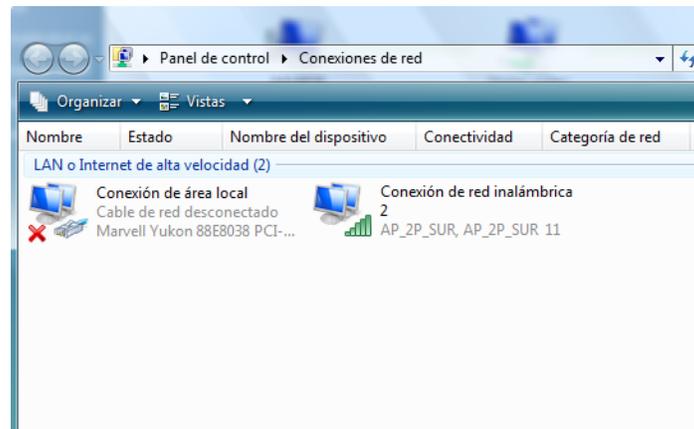


Fig. 4.19. Conexiones de Red.

En esta parte nos vamos a las propiedades de las Red Inalámbrica. Para poder ingresar una IP fija. (Ver 4.20.)

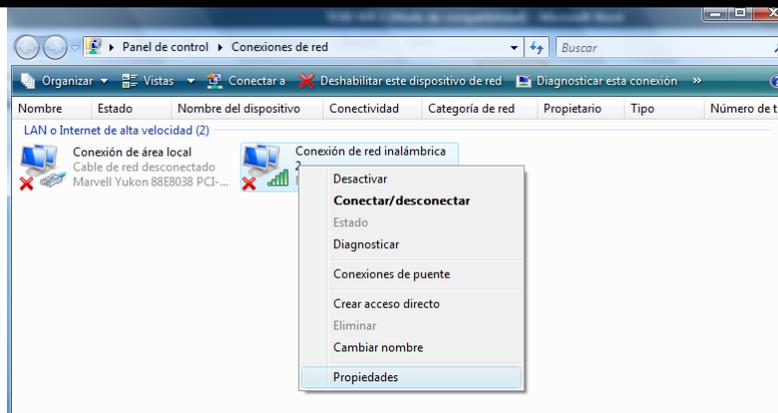


Fig. 4.20. Propiedades de Red Inalámbrica.

Al abrir las propiedades de la conexión inalámbrica, aparecen los diferentes protocolos con que cuenta la conexión. Aquí se ocupa la versión IPv4. (Ver fig. 4.21)

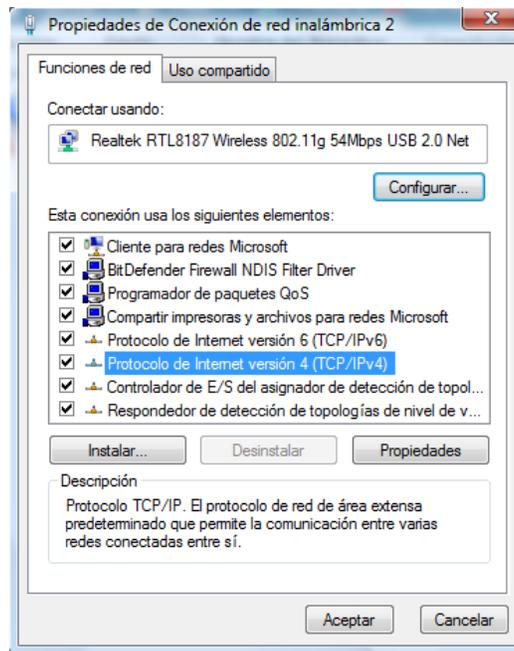
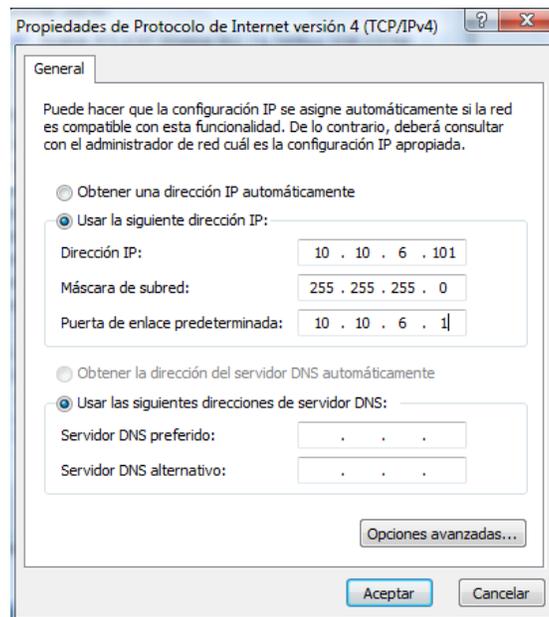


Fig. 4.21. Red Ad-hoc.



Ya que se abren las propiedades de IPv4, aparece una ventana como se muestra a la figura 4.22. En esta parte se selecciona dirección IP en forma manual, y se le introducen tanto la dirección IP, como la máscara de subred, así como la puerta de enlace predeterminada. En el ejemplo se muestra los datos que se dan para la red *Ad-hoc* que se va a ocupar.



**Fig. 4.22. Propiedades de IPv4.**

Con estos pasos, ya se configura tanto la red *Ad-hoc*, como la IP, para el equipo. La computadora se pondrá en espera de que otra computadora o dispositivo inalámbrico entre en la misma red, y así poder empezar con la comunicación entre ambos y el intercambio de la información.

Al estar usando una red *Ad-hoc*, no tenemos un *Access point* que nos de una IP, porque lo que se requiere la asignación de una IP tanto al microcontrolador como a la laptop, es con la que se va a comunicar. A esto se le llama IP estática.



---

#### 4.2.2. Cliente.

Es una parte importante mencionar que la programación y control usando una red inalámbrica *Wi-Fi*, no ha sido muy explotada, dado el costo que se generaba usando un módulo aparte del sistema de control, además de la programación y configuraciones que estos dispositivos requieren.

La comunicación entre los dos dispositivos, se hace mediante uso de cliente servidor. El microcontrolador se puso en modo de cliente, este se comunica con el servidor, el cual está implementado en la computadora. Para esta implementación del servidor se usa el lenguaje de programación LabVIEW. A continuación se muestran las directivas básicas con las cuales se realizaron las pruebas necesarias para la comunicación del cliente utilizando el lenguaje de programación de Dynamic C. [18]**Error! No se encuentra el origen de la referencia.**

```
#define TCPCONFIG      1
#define _PRIMARY_STATIC_IP  "10.10.6.100"
#define _PRIMARY_NETMASK  "255.255.255.0"
#define MY_GATEWAY      "10.10.6.1"
#define MY_NAMESERVER   "10.10.6.1"
#define IFC_WIFI_SSID   "test"
#define IFC_WIFI_ROAM_ENABLE  1
#define IFC_WIFI_ROAM_BEACON_MISS  20
#define IFC_WIFI_CHANNEL  0
#define IFC_WIFI_MODE    IFPARAM_WIFI_ADHOC
#define IFC_WIFI_REGION  IFPARAM_WIFI_REGION_AMERICAS
#define IFC_WIFI_ENCRYPTION IFPARAM_WIFI_ENCR_NONE
```



### 4.2.3. Servidor.

Como ya se ha mencionado, se diseñó una interfaz gráfica, basándose en el programa de LabVIEW, propiedad de *National Instruments*, en dicha interfaz, se puede manejar los controles del tractor, pero la parte fundamental al usar dicho programa, son las herramientas de comunicación en TCP/IP con las que cuenta. Ya que programando en forma gráfica, nos da los resultados al establecer la comunicación. Con algunas herramientas propias del LabVIEW, se programan los puertos tanto para la lectura como escritura de los datos, en la figura 4.23, se muestran dichas herramientas. **¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia.**

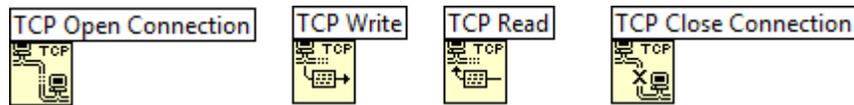


Fig. 4.23. Herramientas de Configuración de TCP/IP

### 4.2.4. Socket.

Un socket es un punto de comunicación en el cual se puede asociar una dirección IP y un número de puerto. Los sockets permiten establecer comunicaciones de dos vías, punto a punto entre dos dispositivos y son un componente básico de comunicación entre dispositivos de una red.

La unión de una dirección IP con un puerto de un dispositivo, forma un socket, el cual es único en toda la Internet. Una aplicación puede tener múltiples direcciones de sockets, con base en los puertos que pueda usar.



---

La comunicación entre los dispositivos de red, se establece cuando dos programas o procesos pueden establecer una conexión entre ellos. Los pasos a seguir para esta conexión son los siguientes.

1. Un puerto local se especifica, dentro de la dirección donde un programa o proceso recibe mensajes.
2. Una dirección host local, que identifica a la computadora que recibirá la información.
3. Un puerto remoto, que identifica el programa o proceso de origen
4. Una dirección host remota, que identifica a la computadora que envía la información.
5. Un protocolo que especifica, como los programas o procesos intercambian datos a través de la red.

Para que exista la comunicación a través de un socket, se necesita un socket en cada extremo de la conexión de la red. Por lo que se requiere las direcciones IP local y remota; así como los puertos de cada dirección IP a donde conectarse. **¡Error! No se encuentra el origen de la referencia.¡Error! No se encuentra el origen de la referencia.**

### **4.3. Funcionamiento del Motor.**

Parte esencial y por demás básica, es el encendido y apagado del motor de combustión interna con que cuenta el tractor, con dicho motor, es la parte que nos va a dar la potencia necesaria para su funcionamiento, tracción, movilidad e independencia eléctrica.



#### **4.3.1. Encendido del Motor de Combustión Interna.**

Se pensó en una metodología para su encendido, el cual cuenta con un switch de seguridad que tiene el propio tractor ya de fábrica, en el asiento del piloto, como tal, este switch hace la función de sensor de presencia, o sensor de presión, ya que como se encuentra en el asiento del piloto, este se cierra cuando una persona se sienta en esa parte del tractor.

Otra parte esencial dentro del encendido es el freno, puesto que este también cierra un interruptor, dentro de la llave de encendido, si el freno no está activo (oprimido), este interruptor no se cierra, y por ende, el tractor no enciende.

Con estas 2 consideraciones (el switch del asiento, y el interruptor del freno) se realizó la metodología antes mostrada. (Véase 4.1.)

Otra parte importante fue el tiempo en que el switch principal del tractor (llave de ignición) está activo, ya que como es de conocimiento general, si le dejamos el switch encendido por más del tiempo requerido por el motor, se puede llegar a dañar el mecanismo de encendido. Al momento de realizar pruebas de encendido, se contabilizó que en promedio el tractor tarda en arrancar 2.5 segundos; por lo que después de este tiempo, se requiere que dicho switch principal se desactive para prevenir posibles daños.



---

#### **4.3.2. Apagado del Motor de Combustión Interna.**

Para el apagado del motor y usando el switch del asiento antes mencionado, se realiza mediante la desactivación de dicho switch, esto provoca que la corriente de la batería se interrumpa y el tractor se apague.

#### **4.3.3. Freno.**

El freno es parte esencial de cualquier dispositivo móvil, tanto para la navegación como para la seguridad, ya sea del medio en el que se desenvuelve, o del mismo dispositivo móvil. En el caso del tractor, este freno realiza tres tareas.

La primera tarea, es cerrar el switch que tiene integrado para el encendido del carro, sin este switch, la corriente de la batería se abre, y el motor de combustión interna no arranca.

La segunda, es propiamente la función de freno. Detener la inercia del tractor.

La tercera tarea, es la función de clutch, para un buen embrague de la palanca de velocidades.

#### **4.4. Navegación.**



---

En esta parte del trabajo que se presenta, se plantea la navegación para conducir el tractor a distancia, generando las rutinas que le permiten avanzar hacia adelante y hacia atrás, dar vuelta hacia la izquierda y derecha, detectar y evitar obstáculos, así como la puesta en marcha y paro de emergencia.

#### **4.4.1. Volante.**

Para el control del volante, se decidió por un servomotor, en vez de algún motor de corriente directa, ya que los servomotores a diferencia de los motores de corriente directa, mantienen su posición y no requieren de algún driver adicional, podemos posicionarlo en cualquier punto dentro de su rango de operación. Esta característica de los servomotores es muy importante dentro de la navegación, ya que si bien, se supone que el terreno es plano, también se piensa que no es al 100% plano, es decir, que pueden existir piedras, baches, o pequeños hoyos, que si, el tractor pasa por ahí, nos pueda modificar nuestra trayectoria. Por razones de diseño de fabricación del propio tractor, tan solo se tiene un rango de operación de aproximadamente 60 grados.

El control del servomotor, se hace mediante PWM (*Pulse Wide Module*), es un ancho de pulso que se va modificando dependiendo de la posición, en grados deseados.

#### **4.4.2. Palanca de Velocidades.**



Al igual que con el volante, aquí también utilizamos un servomotor, pero aquí no es tanto la fuerza en mantener la posición, sino la certeza de posicionar el motor en un punto o grado dado. Ya que, si la palanca de velocidades no embraga bien, la probabilidad de daño hacia la caja de velocidades se incrementa notablemente. Con fines de experimentación, solamente se consideraron 2 velocidades, una hacia adelante (Primera) y otra hacia atrás, (Reversa), esto nos permite, tener una estabilidad en el control de la navegación.

#### **4.5. Detección de Obstáculos.**

La seguridad es esencial en todo dispositivo móvil. Ya que propiamente dentro de la red inalámbrica *Wi-Fi*, se tiene un alcance de aproximadamente 100 metros, esto podría hacer que se pierda cierto rango de visibilidad, o de perspectiva de lo que tiene el tractor en frente, como podría ser alguna piedra, tronco, o algún animal pequeño, como la seguridad es muy importante tanto para el medio ambiente en el que se desenvuelve el tractor; como para el tractor mismo, se pensó en un sensor ultrasónico, ya que una de las ventajas de este tipo de sensor radica en su confiabilidad, bajo nivel de susceptibilidad a ruido y que nos da un rango de trabajo de 2 centímetros, a 4 metros, el sensor no requiere una calibración, ni programación adicional, tan solo tiene 4 pines, voltaje (5 volts), tierra, un pin de disparo y otro de respuesta donde se lee el ancho de pulso resultante, que es proporcional a algún objeto, si éste es detectado, con el ancho de pulso se mide y se obtiene la distancia.

La comunicación con el microcontrolador es a través de un PWM que nos entrega información, dependiendo de la distancia que se encuentre entre el sensor y el obstáculo dentro de la ruta de navegación. Con un ancho de pulso, se procesa e indica al usuario que existe un obstáculo, ya sea para que esté cambie su ruta de navegación, o al menos revise la ruta y sus posibles obstáculos.



## 4.6. Código de Programación.

A continuación se muestran los códigos de programación, usando ambos lenguajes de programación, tanto de Dynamic C, como de LabVIEW.

### 4.6.1. Código en Dynamic C

Como ya se ha mencionado, la programación de este microcontrolador, se hace en Dynamic C, aquí se implementó un sistema de cliente, que esta recibiendo los códigos de navegación y enviando la señal del sensor. A continuación se muestra el código fuente de la programación.

```
/*  
*****  
INSTITUTO POLITÉCNICO NACIONAL.  
  
CENTRO DE INVESTIGACIÓN EN COMPUTACIÓN.  
  
ING. IGNACIO HERNÁNDEZ BAUTISTA  
  
ESTE PROGRAMA ENTABLA LA COMUNICACIÓN ENTRE UNA LAPTOP Y EL  
MICROCONTROLADOR  
  
RABBIT MODELO RCM 4400w, PARA LA NAVEGACIÓN DE UN TRACTOR EN  
FORMA REMOTA USANDO UNA RED WI-FI EN MODO AD-HOC.  
  
JUNIO - 2009  
*****/  
*/
```



```
//INDICA EL MODELO DE LA TARJETA A USAR
#use RCM44xxW.LIB

//DEFINE LA CONFIGURACIÓN DE LA ANTENA WIFI
#define TCPCONFIG 1
#define _PRIMARY_STATIC_IP "10.10.6.100"
#define _PRIMARY_NETMASK "255.255.255.0"
#define MY_GATEWAY "10.10.6.1"
#define MY_NAMESERVER "10.10.6.1"
#define IFC_WIFI_SSID "test"
#define IFC_WIFI_ROAM_ENABLE 1
#define IFC_WIFI_ROAM_BEACON_MISS 20
#define IFC_WIFI_CHANNEL 6
#define IFC_WIFI_MODE IFPARAM_WIFI_ADHOC
#define IFC_WIFI_REGION IFPARAM_WIFI_REGION_AMERICAS
#define IFC_WIFI_ENCRYPTION IFPARAM_WIFI_ENCR_NONE

//INDICA LA IP Y PUERTO A CONECTARSE DE LA LAPTOP
#define DEST_IP inet_addr("10.10.6.101")
#define DEST_PORT 6677
#use "dcrtcp.lib"

// PUERTO A
//-----

// NOMBRE BIT
#define ASIENTO 0
#define FRENO 1
#define ENCEN 2
#define CONEX 3
```



```
#define ADEL      4
#define NEU      5
#define ATRAS    6

#define MAIN_FREQ 58.98

////////////////////////////////////
// ACTIVA FRENO
////////////////////////////////////
void FRENOON()
{
    BitWrPortI(PADR, &PADRShadow, 1, FRENO);
}
////////////////////////////////////
// APAGA FRENO
////////////////////////////////////
void FRENOOOF()
{
    BitWrPortI(PADR, &PADRShadow, 0, FRENO);
}
////////////////////////////////////
// APAGA MOTOR
////////////////////////////////////
void MOTOROFF()
{
    BitWrPortI(PADR, &PADRShadow, 0, ASIENTO);
}
////////////////////////////////////
// RUTINA DE RETARDO DE TIEMPO EN MILISEGUNDOS
```



```
////////////////////////////////////  
nodebug  
void msDelay(unsigned int delay)  
{  
    auto unsigned long done_time;  
done_time = MS_TIMER + delay;  
while( (long) (MS_TIMER - done_time) < 0 );  
}  
main()  
{  
tcp_Socket socket;  
int bytes_written;  
char buffer[8];  
    auto int cuenta;  
    auto double Valor;  
    unsigned long freq;  
    int  pwm_options, err;  
    int  pal, vol;  
    float Resolucion;  
    int  TAT8value, CM;  
    char bus[3];  
    char status;  
    unsigned long pulso;  
    float total;  
//INICIALIZA LA TARJETA DE DESARROLLO  
brdInIt();  
    sock_init_or_exit(1);  
//ABRE EL PUERTO DEL SOCKET UTILIZANDO TCP  
    if (tcp_open(&socket,6677,DEST_IP,DEST_PORT,NULL) == 0)
```



```
{
    printf("No pude abrir el socket");
    exit(1);
}
//INDICA EL MODO A TRABAJAR DEL SOCKET
sock_mode(&socket,TCP_MODE_ASCII);
//ESPERA A ESTABILIZAR EL SOCKET
while(!sock_established(&socket)) && (sock_bytesready(&socket) < 0)
{
    if(tcp_tick(&socket)==0)
    {
        printf("FALLO EN CONEXION \n");           // falla en conexión
    }
    else
    {
        //bytes_written=sock_fastwrite(&socket,"V",1);
        printf("estableciendo socket... \n ");
    }
}
printf("***** YA SE ESTABLECIO CONEXION *****");
    BitWrPortl(PADR, &PADRShadow, 1, CONEX); //ACTIVA LED DE
CONEXION ACTIVA
    bytes_written=sock_fastwrite(&socket,"S",1);
    TAT8value = 255;
    Resolucion = ((TAT8value+1)*2) / MAIN_FREQ; // calcula resolucion
    WrPortl ( TAT8R, NULL, TAT8value );
    WrPortl(PDDDR, &PDDDRShadow, PDDDRShadow & 0xF7); //Port D Pin 3
    BitWrPortl(PDDR, &PDDRShadow, 0, 1);
    WrPortl(ICS1R, NULL, 0x55); //PTO D PIN 3
```



```
WrPortl(ICCR, NULL, 0x00); // normal operation; no interrupts
WrPortl(ICCSR, NULL, 0x0C); // zero out counters
WrPortl(ICT1R, NULL, 0x56);
pulso = 0;
//INICIALIZA LOS PWM A TRABAJAR
freq = pwm_init(400ul);
pwm_options = PWM_USEPORTD;
pal = 0;
vol = 1;
err = pwm_set(0, 0, pwm_options); //puerto D canal 0 palanca
err = pwm_set(1, 0, pwm_options); // puerto D canal 1 volante
Valor = 0.0;
while(1)
{
sock_read(&socket,buffer,1);
cuenta=buffer[0];
if(tcp_tick(&socket)==0) // detecta desconexión
{
    BitWrPortl(PADR, &PADRShadow, 0, CONEX); //DEACTIVA LED DE
CONEXION
    FRENOON(); //ACTIVO EL FRENO PARA Q EL TRACTOR SE
DETENGA
    //msDelay(1000); //ESPERO A QUE EL TRACTOR SE DETENGA
MOTOROFF(); //APAGO EL MOTOR DEL TRACTOR
sock_close(&socket); //cierro el socket
printf(" LA RED SE CAYO \n");
exit(1);
}
//RUTINAS DEL FRENO
```



```
if ((cuenta == 'F') || (cuenta == 'f')){ // ACTIVA FRENO
    FRENOON();
}
if ((cuenta == 'G') || (cuenta == 'g')){ //DESACTIVA FRENO
    FRENOOOF();
}
/*****RUTINAS DEL MOTOR DE COMBUSTION INTERNA *****/
//APAGA MOTOR DE COMBUSTION INTERNA
if ((cuenta == 'T') || (cuenta == 't'))
{
    MOTOROFF();    //APAGO EL MOTOR DE COMBUSTION INTERNA
    FRENOOOF();    //DESACTIVO FRENO
}
//ENCIENDE MOTOR DE COMBUSTION INTERNA
if ((cuenta == 'E') || (cuenta == 'e'))
{
    BitWrPortl(PADR, &PADRShadow, 1, ASIENTO); //ACTIVA SWITCH DEL
ASIENTO
    FRENOON();    //ACTIVA EL PEDAL DEL FRENO
    msDelay(500);    //espera de que el pedal llegue a posicion
    BitWrPortl(PADR, &PADRShadow, 1, ENCEN); //ACTIVA EL SWITCH DE
ENCENDIDO DEL CARRO
    // Valor =512.0; //PONE PALANCA DE VELOCIDADES EN NEUTRAL
    msDelay(1500);    //ESPERA A QUE EL MOTOR DE COMBUSTION
INTERNA ARRANQUE
    BitWrPortl(PADR, &PADRShadow, 0, ENCEN);// "SUELTA" EL SWITCH DE
ENCENDIDO
}
// RUTINAS DE POSICIONAMIENTO DEL VOLANTE
```



```
if ((cuenta == 'A') || (cuenta == 'a')){ // VOLANTE IZQUIERDA
    err = pwm_set(vol,1013, pwm_options);
}
if (cuenta == '2') { // VOLANTE AL FRENTE
    err = pwm_set(vol,765, pwm_options);
}
if ((cuenta == 'W') || (cuenta == 'w')){ // VOLANTE AL FRENTE
    err = pwm_set(vol,512, pwm_options);
}
if (cuenta == '4') { // VOLANTE AL FRENTE
    err = pwm_set(vol,256, pwm_options);
}
if ((cuenta == 'D') || (cuenta == 'd')){ // VOLANTE DERECHA
    err = pwm_set(vol,10, pwm_options);
}
/*****RUTINAS DE POSICIONAMIENTO DE PALANCA DE
VELOCIDADES*****/
// PALANCA PRIMERA
if ((cuenta == 'U') || (cuenta == 'u')){
    if (Valor != 1013){
        FRENOON();
        msDelay(500);
        BitWrPortI(PADR, &PADRShadow, 1, ADEL);
        BitWrPortI(PADR, &PADRShadow, 0, NEU);
        BitWrPortI(PADR, &PADRShadow, 0, ATRAS);
        err = pwm_set(pal,1013, pwm_options);
        Valor = 1013;
        FRENOOOF();
    }
}
```



```
BitWrPortI(PADR, &PADRShadow, 1, ADEL);
BitWrPortI(PADR, &PADRShadow, 0, NEU);
BitWrPortI(PADR, &PADRShadow, 0, ATRAS);
err = pwm_set(pal,1013, pwm_options);
Valor = 1013;
}

// PALANCA NEUTRAL
if ((cuenta == 'J') || (cuenta == 'j'))
{
FRENOON();          //ACTIVO EL FRENO
msDelay(500);       //ESPERO QUE SE DETENGA
BitWrPortI(PADR, &PADRShadow, 0, ADEL);
BitWrPortI(PADR, &PADRShadow, 1, NEU);
BitWrPortI(PADR, &PADRShadow, 0, ATRAS);
err = pwm_set(pal,512, pwm_options);
Valor = 512;        //POSICIONO PALANCA EN NEUTRAL
}

// PALANCA REVERSA
if ((cuenta == 'M') || (cuenta == 'm')){
if (Valor != 10){
FRENOON();
msDelay(500);
BitWrPortI(PADR, &PADRShadow, 0, ADEL);
BitWrPortI(PADR, &PADRShadow, 0, NEU);
BitWrPortI(PADR, &PADRShadow, 1, ATRAS);
err = pwm_set(pal,10, pwm_options);
Valor = 10;
FRENOOOF();
}
}
```



```
BitWrPortl(PADR, &PADRShadow, 0, ADEL);
BitWrPortl(PADR, &PADRShadow, 0, NEU);
BitWrPortl(PADR, &PADRShadow, 1, ATRAS);
err = pwm_set(pal,10, pwm_options);
Valor = 10;
}
if (cuenta == '9')
{
    BitWrPortl(PDDR, &PDDRShadow, 1, 1);
        msDelay(1);
        BitWrPortl(PDDR, &PDDRShadow, 0, 1);
        status = RdPortl(ICCSR);
        if(status & 0x10)
        {
            pulso = RdPortl(ICL1R);
            pulso += (RdPortl(ICM1R)<<8 );// * 256L;
            total = pulso*Resolucion;
            CM = total/58;
            itoa (CM,bus);
            printf ( "CADENA = %s ", bus);
            //bytes_written = sock_fastwrite(&socket,bus,3);
            bytes_written = sock_puts(&socket,bus);
            pulso = 0;
            WrPortl(ICCSR, NULL, 0x04); // zero out counter
        }
    }
} //FIN DE WHILE
} //fin de main
```



#### 4.6.2. Código en LabVIEW

Los códigos de programación en LabVIEW, usando el modo de servidor para la interfaz gráfica, se presenta a continuación el panel frontal en la figura 4.24.

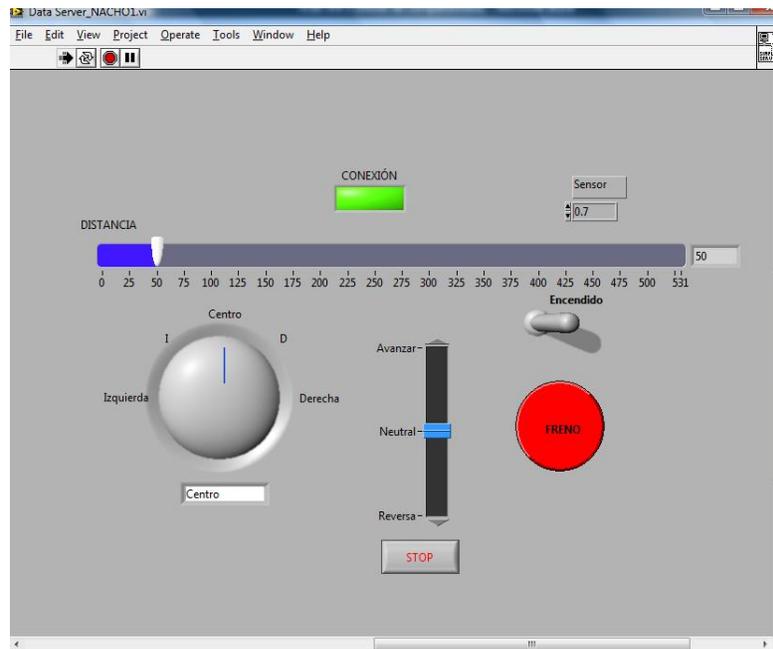


Fig. 4.24. Panel Frontal de la Interfaz Gráfica

Por razones de espacio, se muestra en secciones el diagrama a bloques del servidor el cual contiene la programación de la interfaz gráfica. En esta primera figura (Fig. 4.25), se muestra la configuración del protocolo de TCP, el puerto que se va a abrir, para permitir la comunicación entre éste y el cliente; y muestra tanto la dirección como el puerto remoto de donde se está esperando la conexión, así como el tiempo de espera.

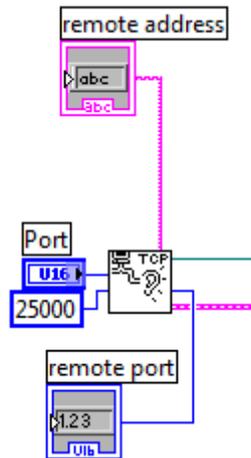


Fig. 4.25. Diagrama de Configuración de TCP

En esta figura 4.26, se muestra la programación de cada dispositivo de navegación, volante, freno, palanca de velocidades, encendido y apagado del motor, y la programación de la lectura del sensor ultrasónico. Así como la forma de escritura y lectura en forma remota usando el protocolo de TCP/IP.

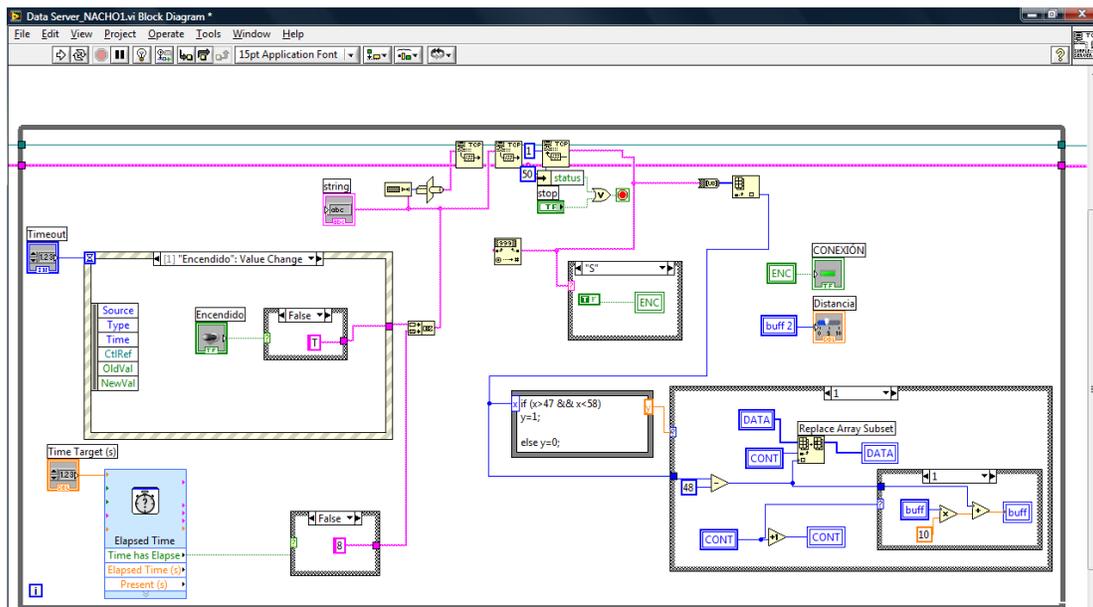


Fig. 4.26. Diagrama a Bloques de la Programación de Dispositivos.

Finalmente se presenta el código que cierra la conexión. Fig. 4.27.

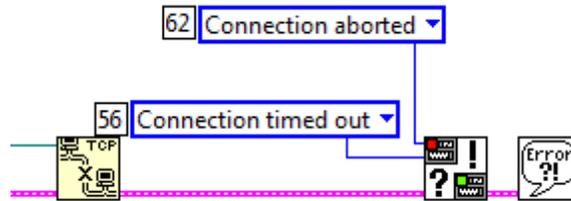


Fig. 4.27. Diagrama a Bloques de Cerrar Conexión TCP.

En este capítulo se ha visto la forma de cómo se fueron diseñando, y creando las herramientas, software y hardware para los diferentes aspectos que contiene la navegación, transmisión de datos, y seguridad. En el siguiente capítulo se detallaran los problemas y necesidades que se fueron dando, así como la forma en que se fueron resolviendo.



---

## 5. DISQUISICIÓN EXPERIMENTAL

Para este capítulo se abordarán algunas problemáticas y detalles que se fueron presentando a lo largo del desarrollo de esta tesis.

### 5.1. Transmisión de Datos.

Durante las pruebas de la transmisión de datos, fundamental en la presente tesis, fue relativamente problemático el entablar la comunicación con el microcontrolador, desde ver el tipo de comunicación y además, como es que se iba a escribir y leer por la red inalámbrica, una parte que ayudó fue el mantener una dirección IP estática, con lo cual desde la computadora se mandaba un “ping” para ver si estaba activa la señal, esto nos da una idea clara de cómo estaba el status de la comunicación. Una primera etapa de pruebas, se realizó prendiendo y apagando un led, tan solo para ver la existencia de comunicación y la transferencia de los datos. Y posteriormente se implementó todo el sistema usando códigos de control, que se propusieron para los diferentes dispositivos y procesos que se requerían.



---

Otro aspecto fue la programación en LabVIEW, desde la implementación del servidor, hasta la programación del mismo. Y su comunicación y transferencia de datos desde la computadora hacia el microcontrolador y en forma inversa. En este programa, en una primera etapa, se realizaron pruebas usando otra interfaz gráfica del mismo LabVIEW, para monitorear, el envío de datos.

Otra parte importante dentro de la transmisión de datos es la seguridad, en este caso significa que debemos de estar seguros que el usuario, realmente tenga comunicación y control del tractor, y que en caso contrario no signifique un peligro para el medio ambiente o el mismo tractor. Esto se fue resolviendo en ambos ambientes de programación, tanto en el Dynamic C, como en LabVIEW, aunque cabe señalar que en un momento dado, es más importante que se tenga en el microcontrolador, ya que como éste es el que tiene el control del tractor, no se quedará esperando a que el tractor recibiera un código, de una red que en algún momento no estuviera activa la conexión inalámbrica y éste siguiera una trayectoria sin control. Esto se realizó provocando que el tractor se detenga en el momento en que el microcontrolador detecta la ausencia de la red, y apague el motor de combustión interna. Aunque también, dentro de la interfaz gráfica se tiene la forma de saber si el programa de LabVIEW tiene la conexión inalámbrica activa.



### 5.1.1. TCP en LabVIEW.

El lenguaje de programación en LabVIEW, tiene ya las funciones de TCP/IP implementadas como se muestra en la figura 5.1. Esto nos permite tan solo hacer la programación de las funciones. A continuación se describen las funciones que utilizamos para diseñar un servidor para la red inalámbrica.

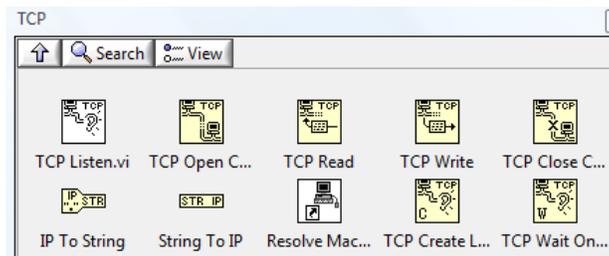


Fig. 5.1. Funciones TCP.

Lo primero que debemos hacer para la implementación de un servidor, es crear un socket, y ponerlo a “escuchar” para esperar a que el cliente se conecte y poder establecer la comunicación. Esto se logra con el TCP Listen, como se puede ver en la figura 5.2, con éste se puede realizar el enlace con el dispositivo remoto, y nos entrega la dirección y puerto en forma remota a la cual se conectó. A este vi (*Virtual Instrument*), le damos el puerto que vamos a compartir.

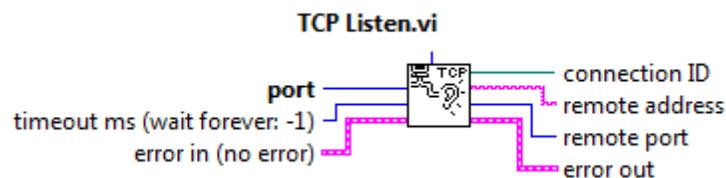


Fig. 5.2. TCP Listen.

Después, ya que el cliente se conectó, se abre el puerto del TCP. Con el vi de *TCP Open* (Ver Fig. 5.3). Con este vi, al momento de conectarlo al *tcp listen*, se puede abrir ya el socket en espera de escribir o leer el socket.



Con este *vi*, parte importante es ver el tiempo de espera para la conexión, ya que puede tener problemas para la conexión. Este *vi* nos entrega una conexión activa y un estatus de error.

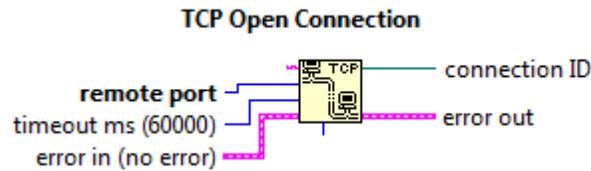


Fig. 5.3. TCP Open.

Cuando queremos escribir algún código al socket, se usa el *vi* de TCP Write, con este *vi*. (Fig. 5.4.) la parte importante son los códigos que deseamos escribir en el socket, para eso usamos el *data in*, que también tienen un tiempo de espera y entrada de error. Este *vi* nos entrega, la conexión activa y los códigos que se escribieron en el socket.

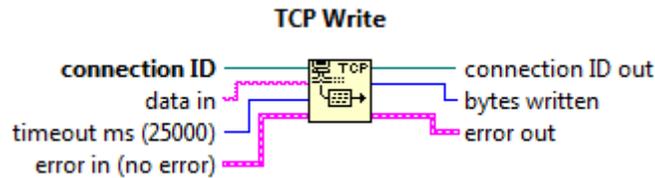


Fig. 5.4. TCP Write.

En sentido inverso, si deseamos leer algún código, se usa el *TCP Read*. (Ver Fig. 5.5.).

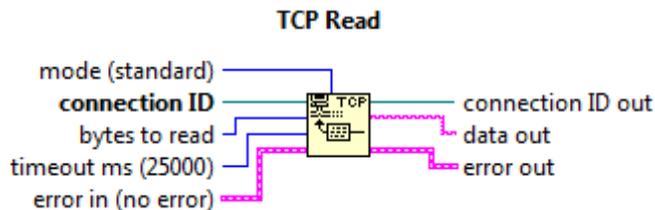


Fig. 5.5. TCP Read.



Para cerrar la conexión del socket, se usa el *vi* de *TCP Close Connection*. (Ver Fig. 5.6.)

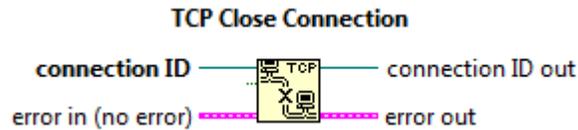


Fig. 5.6. TCP Close Connection.

Con estos *vi*, se crea un socket, se abre la conexión, se realiza la escritura o lectura del socket y se cierra el socket.

### 5.1.2. TCP en Dynamic C.

Para la configuración de TCP sobre Dynamic C, usamos la directiva “#define” con esta directiva y después de compilar el programa y guardarlo dentro de la memoria flash de la tarjeta, ya quedará grabada la información de los parámetros de la tarjeta inalámbrica que tiene de forma embebida. Así mismo es necesario darle los parámetros a utilizar a la tarjeta de red como son:

- La dirección IP.
- La máscara de subred.
- La puerta de enlace.

Otros datos importantes para la red inalámbrica son: el tipo de arquitectura de red; se le indica que va a trabajar en modo *Ad-hoc*, así como el nombre de la red a utilizar, el canal, y el tipo de encriptación o autenticación, a continuación se muestra el código para la configuración de la antena inalámbrica en modo Ad-hoc sin autenticación, y configurando los parámetros de SSID, IP, Gateway y Submascara.



---

```
#define TCPCONFIG      1
#define _PRIMARY_STATIC_IP  "10.10.6.100"
#define _PRIMARY_NETMASK   "255.255.255.0"
#define MY_GATEWAY        "10.10.6.1"
#define MY_NAMESERVER     "10.10.6.1"
#define IFC_WIFI_SSID     "test"
#define IFC_WIFI_ROAM_ENABLE 1
#define IFC_WIFI_ROAM_BEACON_MISS 20
#define IFC_WIFI_CHANNEL  1
#define IFC_WIFI_MODE     IFPARAM_WIFI_ADHOC
#define IFC_WIFI_REGION   IFPARAM_WIFI_REGION_AMERICAS
#define IFC_WIFI_ENCRYPTION IFPARAM_WIFI_ENCR_NONE
```

Como se ha visto, para crear un socket, es necesario tener una dirección IP y un puerto a donde comunicarse, es por eso que usando la misma directiva, se le da dirección y puerto destino, donde se encuentra el servidor con el cual se va a entablar la comunicación, como se puede observar en las siguientes líneas de código.

```
#define DEST_IP  inet_addr("10.10.6.101")
#define DEST_PORT  6677
```

En este momento ya podemos estar en condiciones de crear un socket. Con la siguiente instrucción, creamos un socket con el protocolo TCP y a su vez le asignamos un nombre. Que para fines prácticos, se le dió el nombre de socket.

```
tcp_Socket socket;
```



---

Y para abrir el socket e iniciar con la transmisión de datos es necesaria la siguiente instrucción.

```
sock_init_or_exit();
```

Esta instrucción nos permite establecer la comunicación con el socket o de otra forma salir, lo cual, nos da cierta seguridad con el dispositivo. Ya que así, no podremos intentar nada hasta que el socket no se haya abierto, y establecido correctamente. Esto lo hacemos primero con la siguiente instrucción.

```
tcp_open(&socket,6677,DEST_IP,DEST_PORT,NULL);
```

La tarea de esta instrucción, es la de crear una sesión con otra máquina remota usando el protocolo TCP.

- Parámetro 1: Socket.
- Parámetro 2: Nuestro puerto, que debe de estar dentro del rango de 1025 a 65536.
- Parámetro 3: La dirección IP a conectarse.
- Parámetro 4: El Puerto a conectarse.
- Parámetro 5: Función que puede ser llamada si hay un dato que es recibido. NULL para usar el buffer del socket.

Valor de Retorno: 0 si ha sido abierto, o diferente a 0 de otro modo.

Dentro de Dynamic C, una parte que se debe considerar, es el modo de funcionamiento del socket; existen dos modos de programar el socket, en modo binario y en modo Ascii. Para su programación utilizamos la siguiente instrucción.



---

```
sock_mode(&socket,TCP_MODE_ASCII);
```

```
TCP_MODE_BINARY (default).
```

```
TCP_MODE_ASCII.
```

Parámetro 1: Socket.

Parámetro 2: Modo de funcionamiento.

En este momento, ya se creó un socket, se abrió y se configuró su modo de operación. Parte de la seguridad que se tiene es saber si este socket, funciona de forma adecuada, esto lo hacemos asegurándonos de que el socket sí está conectado y tenemos una comunicación estable con el servidor. Con la siguiente instrucción, podemos tener la seguridad de que esta comunicación está establecida.

```
sock_established(&socket);
```

Parámetro 1: Socket.

Valor de Retorno: 0 no se estableció.

1 si se estableció.

Otra parte fundamental que se debe saber es si esta comunicación está activa, ya que por seguridad se requiere saber si realmente se tiene el control del tractor. Para esta tarea se utiliza la instrucción:

```
tcp_tick();
```



---

Monitorea en todo momento esa comunicación y en dado caso que falle, se establece un código, indicando que la comunicación se perdió, y por lo tanto se toman decisiones, como por ejemplo, detener y apagar al tractor. De esta forma, se da certidumbre al usuario y seguridad tanto para el tractor como para el medio en el que se está desarrollando.

Valor de Retorno: 0 cuando se ha perdido o cerrado la conexión.

En este punto, ya podemos escribir y leer en el socket, para escribir en el socket usamos la siguiente instrucción.

```
sock_fastwrite(&socket,"S",1);
```

Parámetro 1: Socket.

Parámetro 2: El dato a ser escrito en el socket, el buffer contiene el dato a ser escrito.

Parámetro 3: Número máximo de bytes a ser escritos en el socket.

Valor de Retorno: Numero de bytes escritos. O -1 si ocurrió un error.

Cuando se desea leer en el socket usando la siguiente instrucción.

```
sock_read(&socket,buffer,1);
```

Esta instrucción se utiliza de forma similar a la instrucción de escritura.

Parámetro 1: Socket.

Parámetro 2: Buffer donde se va a guardar el dato leído.

Parámetro 3: Máximo número de bytes que se escriben en el buffer.



---

Valor de Retorno: Número de bytes leídos o -1 si ocurrió un error.

Para poder cerrar la comunicación de forma adecuada, se usa la instrucción de cerrar el socket.

```
sock_close(&socket);
```

Si el socket se ha quedado abierto, no puede volver a ser usando, hasta que esté completamente cerrado. Esto también puede ser usando con la instrucción `tcp_tick()`.

Parámetro 1:       Socket.

Con estas instrucciones y haciendo los pasos requeridos se pudo crear, establecer, comunicar y cerrar un socket. Esto nos permitió poder enviar y recibir los códigos de control entre ambos dispositivos, por un lado con la laptop usando el software de LabVIEW, y por el otro, el microcontrolador rabbit usando el software de Dynamic C.

## **5.2. Servomecanismos de Navegación.**

Los servomecanismos de navegación, tanto para el volante como para la palanca de velocidades, se basaron, en las características propias que tienen un servomotor: torque, estabilidad y posicionamiento. Las pruebas que se realizaron



fue el control y manejo de los puertos de PWM con que cuenta el microcontrolador Rabbit, este microcontrolador tiene 4 canales independientes, en 3 diferentes puertos, estos canales son de 10 bits, lo que nos da una resolución de 1024 pulsos.

$$2^n = 2^{(10)} = 1024$$

La parte importante, fueron las pruebas realizadas para el posicionamiento adecuado de los servomotores, es decir, el ángulo de trabajo en donde se van a posicionar y que estos embraguen adecuadamente en la caja de velocidades, así como tener una navegación adecuada del tractor. Como se muestra en la figura 5.7 al enviar la señal al servomotor, éste gira hacia donde se le indica, ya sea a la derecha o hacia la izquierda. La figura 5.7a, muestra la forma en que giraran las llantas del tractor, cuando se le incorporen los desarrollos del presente trabajo de tesis, es decir, para la navegación y hacer las vueltas hacia la izquierda o derecha, se tiene un ángulo aproximado de 30 grados, esto es importante para la navegación, ya que por el tipo de locomoción, es necesario estar en movimiento para poder dar vuelta hacia la izquierda o derecha.



**Fig. 5.7. Grado de Giro del Tractor.**

El aspecto más trascendente en la palanca de velocidades, es el correcto embrague, ya que al momento de hacer la navegación y teniendo como ejemplo,



---

que el tractor tenga una trayectoria hacia adelante y queremos que en algún punto, vaya hacia atrás, no podemos nada más cambiar la posición de la palanca de velocidades, ya que esto dañaríamos a la caja de velocidades, en este caso, se realizó una metodología para su cambio de ángulo o velocidad, es decir, antes de cambiar la posición del servomotor, se requiere revisar la posición de éste, o que se analice el punto donde se encuentra el motor, teniendo en cuenta que el motor guarda su posición mientras siga energizado, lo que se pensó, es monitorear el ancho de pulso que se tiene para el control de dicho servomotor, eso se puede hacer mediante software, lo que entre otros casos, permite disminuir los costos de fabricación, a diferencia de realizarlo por hardware.

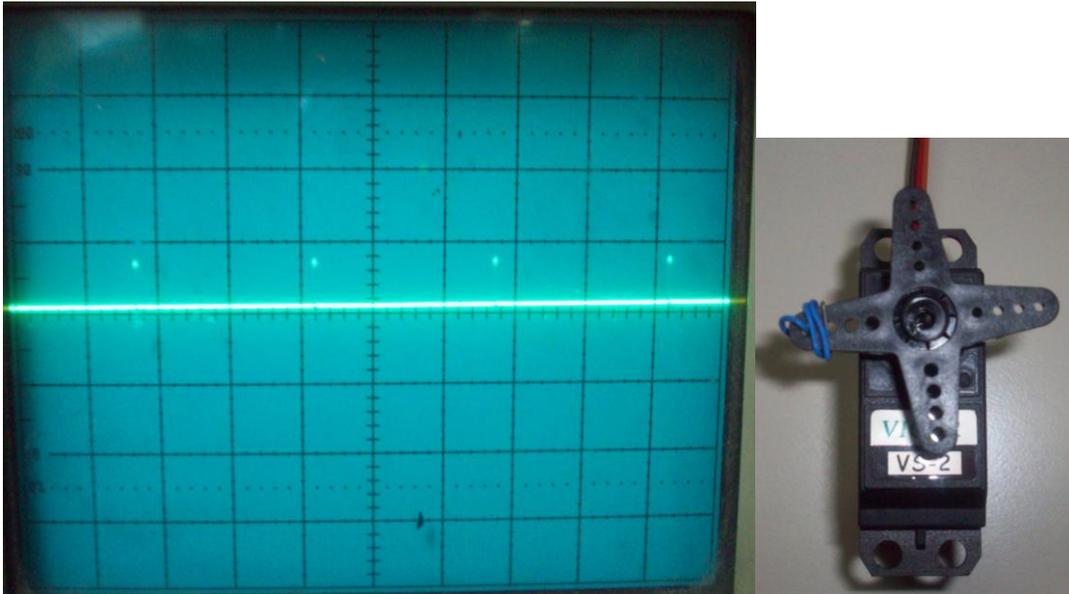
Con la referencia del ancho de pulso, la metodología que se siguió tanto para meter primera, como para la reversa; es revisar el valor del ancho de pulso de control, dependiendo de su valor, se toma la siguiente decisión, si ya está en dicha velocidad, no hacer nada, y en el caso de que no estuviera, (es decir, si estuviera en neutral, en la dirección contraria o a hacia donde queremos la trayectoria), lo primero que se debe de hacer es un frenado del tractor, y usando la configuración del fabricante del propio tractor que nos dice, que el freno también es clutch, hacer el cambio del estado de la palanca hacia la posición deseada.

Lo anterior no importando en qué posición este, ya que al ser servomotor, y teniendo un rango de trabajo definido, pasaría por un punto neutral y después a la posición requerida.

Con estas bases, a continuación se muestran las gráficas que se obtuvieron, de la salida del PWM, usando un osciloscopio digital marca Tektronix, modelo 2236A. Como se mencionó, el ancho del pulso tiene una resolución de 1024. En la siguiente imagen 5.8, se puede observar la salida del ancho de pulso con un ciclo

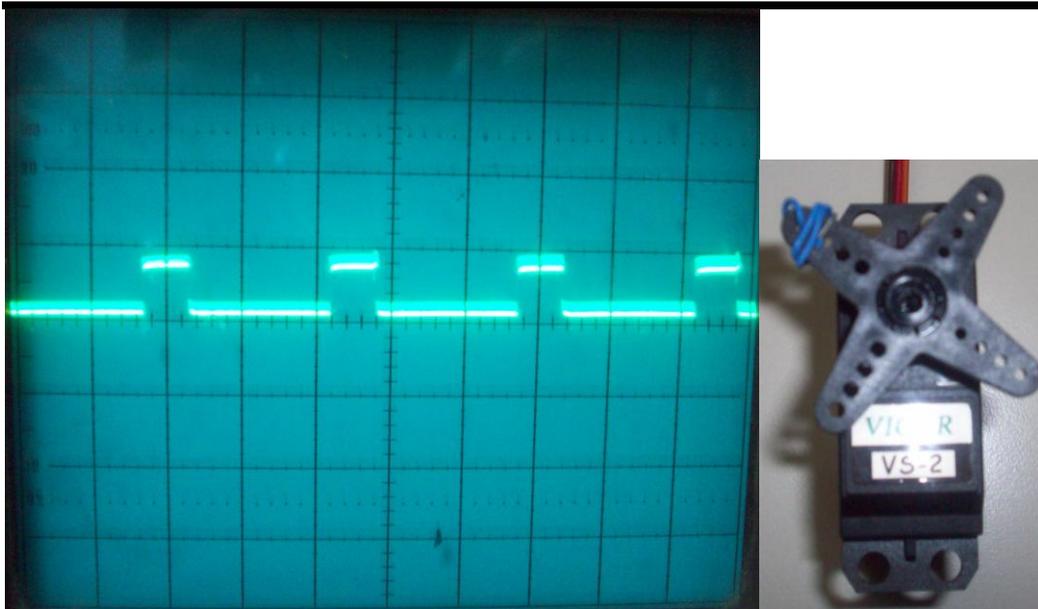


de trabajo del 10%, así como la posición que se establecería en el servomotor.. Lo anterior con la finalidad de poder observar el inicio del periodo.



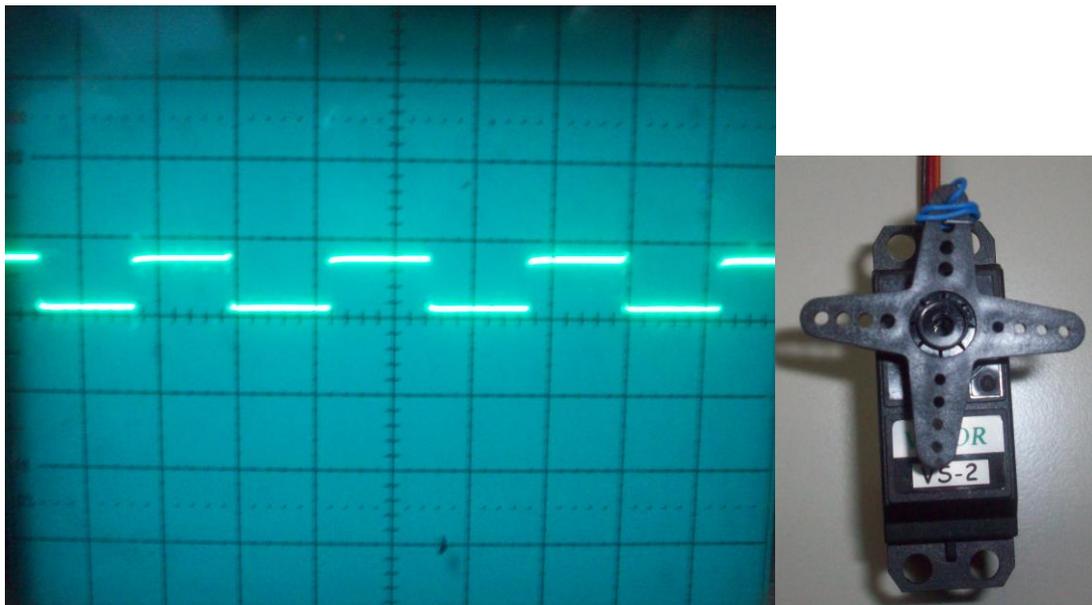
**Fig. 5.8. PWM con un Ciclo de Trabajo del 10%.**

Para la navegación y como parte de las pruebas que se realizaron, se propusieron 5 pasos, en la imagen siguiente se puede observar el segundo paso, este ancho de pulso está a 25% del ciclo de trabajo.



**Fig. 5.9. PWM con un Ciclo de Trabajo del 25%.**

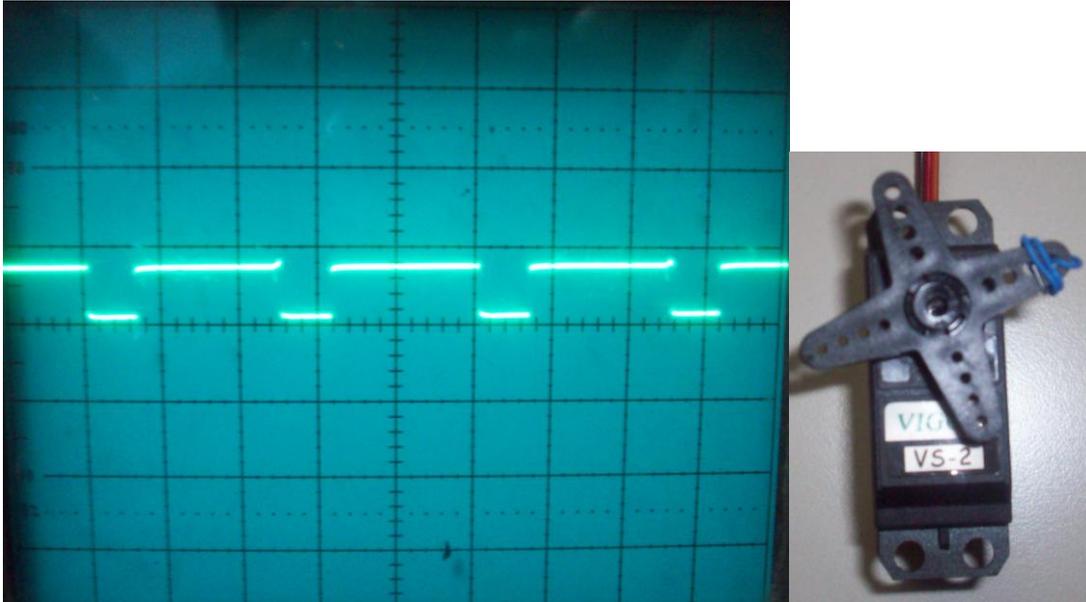
Para el tercer paso, se propuso tener el 50% del ciclo de trabajo, en la salida del ancho de pulso. Esto para poder abarcar toda la resolución que nos da el canal de salida del ancho de pulso.



**Fig. 5.10. PWM con un Ciclo de Trabajo del 50%.**



Para después incrementar el ancho del pulso, al 75% del ciclo de trabajo, lo que nos da una respuesta como se muestra en la figura 5.11.



**Fig. 5.11. PWM con un Ciclo de Trabajo del 75%.**

Ya para finalizar se tiene una imagen del 90% del ciclo de trabajo, se puso ese valor para poder ver su período y tener una idea más clara de cómo se comportan los canales del PWM, y como se comporta el servomotor a los diferentes ciclos de trabajo.

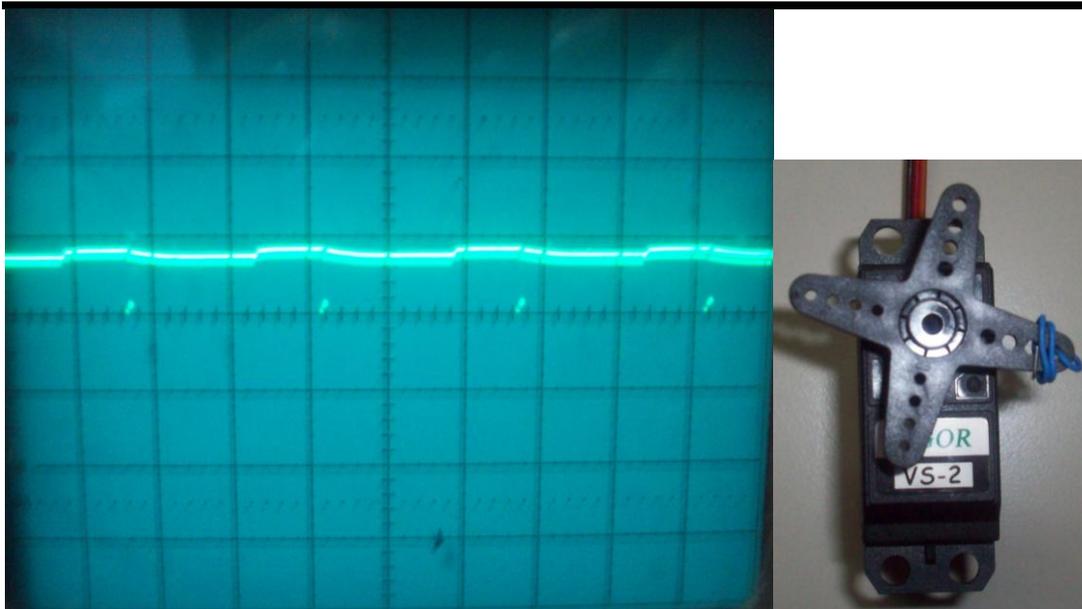


Fig. 5.12. PWM con un Ciclo de Trabajo del 90%.

Con estas imágenes, se puede comprender mejor el funcionamiento del PWM.

### 5.3. Motor de Combustión Interna.

Parte esencial de todo este proyecto fue el correcto encendido y apagado del motor de combustión interna. Para esta actividad, nos basamos en el diseño eléctrico del tractor dado por el fabricante, en el cual podemos ver los diferentes arreglos e interruptores de seguridad que tiene dispuestos. A continuación, en el diagrama 5.1, se muestra el diagrama proporcionado por el fabricante.

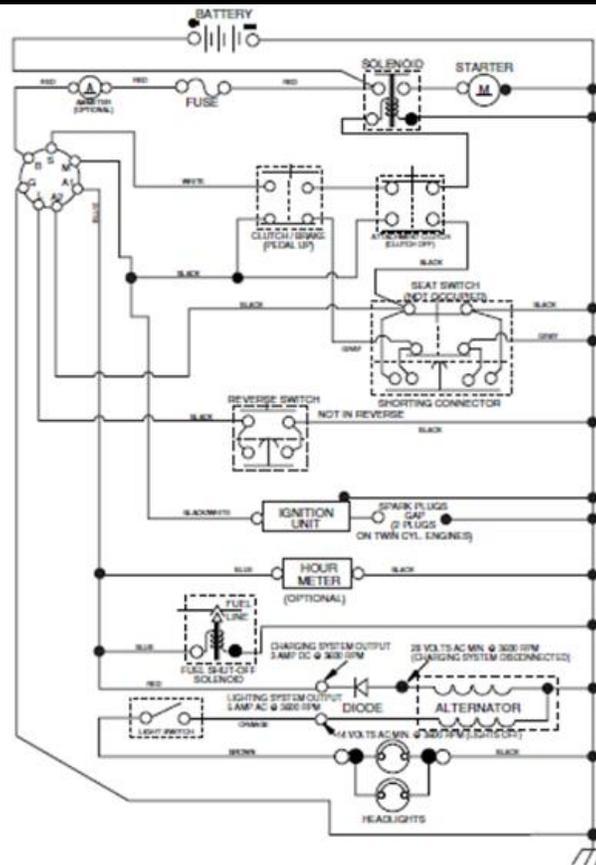


Diagrama 5.1. Diagrama Eléctrico del Tractor.

Con este diagrama, se pudo comprobar el funcionamiento de los interruptores, y su modificación para nuestros propósitos, en forma particular, el asiento del conductor, ya que este interruptor abre todo el circuito de energía desde la batería, apagando el motor.

Otro aspecto es el correcto encendido, también basándose en el diagrama anterior, se logró proponer una metodología para su encendido, que siguiera pasos específicos que cumplieran con el diseño idóneo, para un correcto encendido, navegación y seguridad. Estos pasos, incluyen el activar el actuador de freno, lo que cierra en forma interna otro switch, cerrar el switch del asiento y con estas 2 condiciones, ya poder estar en posición para activar la llave de ignición. Se determinó que el tiempo que le lleva al motor de combustión interna



---

su arranque, es en promedio de 2.5 seg. Por lo que en la programación del algoritmo (módulo) de arranque, se dejó ese valor para su encendido.

Pasado este tiempo, se desactiva la llave de ignición, para así poder evitar posibles daños al sistema de arranque del propio motor de combustión interna.

#### **5.4. Detección de Obstáculos.**

Para las pruebas de detección de obstáculos, revisando las características del sensor propuesto y viendo el tipo de respuesta que nos entrega, las pruebas que se realizaron fueron usando los mismos anchos de pulso que se generaban para el control de los servomotores, pero ahora como simulación de salida del sensor, y así analizar y depurar el funcionamiento del código fuente.

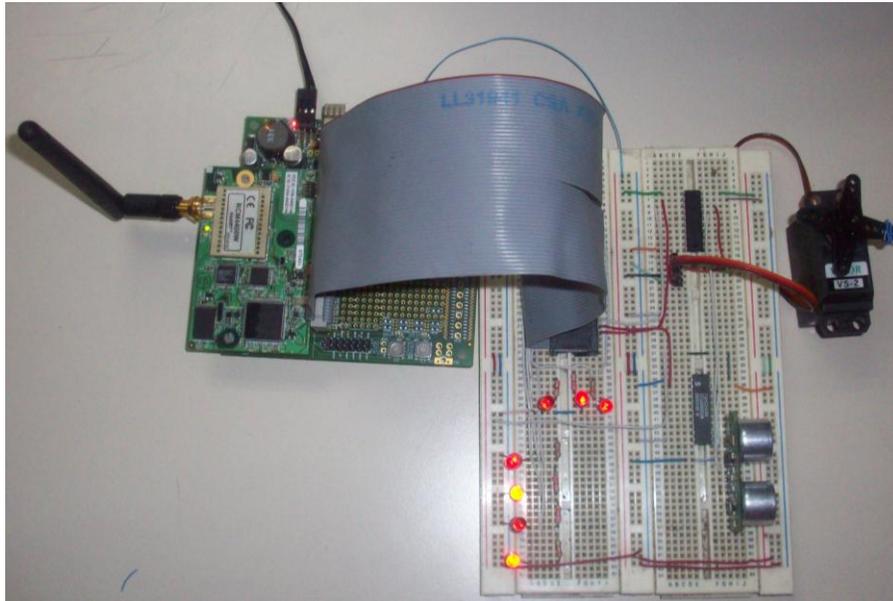
Esto nos permitió tener un claro funcionamiento del código fuente y dejar así el programa propuesto. La seguridad ha sido pieza importante en el diseño y metodología que se ha propuesto para este desarrollo; es por eso que el buen funcionamiento de este sensor, es de suma importancia.

El sensor al tener 4 metros de rango de operación, permite tener una mejor libertad de decisión y por lo tanto de operación, siendo una de las razones por las cuales se decidió por un sensor ultrasónico; ya que a diferencia de sensores infrarrojos, los sensores ultrasónicos tienen un mayor alcance, aunque son un poco más susceptibles a interferencias por emisiones de ruido externo.

Teniendo el respaldo de este sensor, y con los candados dentro el protocolo TCP/IP para la supervisión de que la red se encuentre activa, se da certidumbre al usuario para el empleo, navegación y supervisión que tendría del tractor.



A continuación se presenta el diseño que se elaboro en laboratorio para la validación y pruebas. En el cual se puede observar al microcontrolador rabbit, al sensor ultrasónico, al servomotor y a los diferentes dispositivos para evaluar las rutinas de conexión, freno, encendido del motor de combustión interna.



**Fig. 5.13. Diseño de Laboratorio.**

## **5.5. Clasificación del Robot.**

Como se ha visto, existen diferentes tipos de robots de control, para nuestro caso, el robot de control que se tiene en el desarrollo que se presenta, es del tipo supervisorio, ya que se requiere una constante participación del usuario, para la



---

navegación y toma de decisiones del conjunto de actividades que se deben realizar para la puesta en marcha, navegación, evasión de obstáculos y paro. Así también dentro de las generación que se vieron en el capítulo dos, la propuesta de la presente tesis, recae en la tercera generación. Ya que si bien, no es un sistema autónomo, si utilizamos una computadora, sensor y lenguajes de programación para el control y navegación del mismo.



---

## 6. CONCLUSIONES Y PROPUESTAS DE TRABAJOS FUTUROS

### 6.1. CONCLUSIONES.

Dentro de los logros y conclusiones que se obtuvieron al término de la realización de la presente obra. Se tiene lo siguiente:

- Se asentaron las bases para la programación y mando a distancia (telemando) de dispositivos móviles, usando red inalámbrica *Wi-Fi*.
- Se concluyó con el diseño de los módulos y algoritmos para el control, navegación y operación de dispositivos móviles.
- Se diseñó la tarjeta de adecuación de datos, así como la etapa de adecuación de la potencia de los servomecanismos.
- Se simuló, validó y evaluó el sistema de control de navegación en forma remota, de dispositivos móviles, con la implementación de un sensor detector de obstáculos.



## 6.2. TRABAJO A FUTURO.

El presente trabajo establece las bases necesarias para lograr el control a distancia de la navegación de un tractor no tripulado, con evasión de obstáculos, como trabajo a futuro se sugieren las siguientes acciones:

- Instalación de servomotores y sensores del tractor.
- Respetando las clasificaciones de los dispositivos móviles, sería hacerlo repetitivo y después autónomo, incorporándole:
  - Reconocimiento de patrones.
  - Reconocimiento de voz.
  - Posicionamiento GPS.
  - Control por medio de la Web.



---

## 7.- BIBLIOGRAFÍA

- [1] [OLLERO 2007] OLLERO BATURONE ANIBAL.  
“Robótica Manipuladores y Robots Móviles”  
Editorial. Alfaomega. 2007.
- [2] [DOMINGO 2001] DOMINGO ESTEVE JUAN.  
“Apuntes de Robótica”  
Valencia. 2001.
- [3] [www.rae.es](http://www.rae.es) Página WEB de la Real Academia Española de la Lengua.
- [4] [SANDIN 2003] PAUL E. SANDIN  
“Robot mechanisms and mechanical devices illustrated”.  
Editorial McGraw-Hill. 2003
- [5] [SIEGWART 2004] ROLAND SIEGWART, ILLAH R. NOURBAKHSH.  
“Introduction to Autonomous Mobile Robots”  
Editorial Massachusetts Institute of Technology. 2004
- [6] Manual de Usuario Dynamic C. 2008.
- [7] Manual de Operación. [www.poulan.com](http://www.poulan.com)



- 
- [8] Manual de técnico SRF05. DEVANTECH Ltd
- [9] [LARAJA 2007] LARAJA VIZCAINO JOSÉ RAFAEL, PELEGRÍ SEBASTIÁ JOSÉ.  
“LabVIEW Entorno Gráfico de programación”.  
Editorial. Alfaomega. 2007.
- [10] [www.ni.com](http://www.ni.com)
- [11] [www.Wi-Fi.org](http://www.Wi-Fi.org)
- [12] [www.ieee802.org/11](http://www.ieee802.org/11)
- [13] [GARCIA 2008] GARCIA SERRANO ALBERTO.  
“Redes *Wi-Fi*”  
Editorial. Anaya Multimedia. 2008.
- [14] [ROBLEDO 2002] ROBLEDO SOSA CORNELIO.  
“Redes de Computadoras”.  
Editorial. Instituto Politécnico Nacional. 2002.
- [15] [BEKEY 2005] A. BEKEY GEORGE.  
“Autonomous Robots”  
Editorial Massachusetts Institute of Technology. 2005
- [16] [HYDER 2005] HYDER KAMAL, PERRIN BOB.  
“Embedded Systems Design using the Rabbit 3000 Microprocessor”  
Editorial Elsevier, 2005



---

[17] Manual de Usuario RabbitCore RCM4400w, C PROGRAMABLE Wi-Fi Core Module. 2008.

[18] Manual de Usuario TCP/IP Volumen 1 Dynamic C. 2008.



---

comunicación vía inalámbrica utilizando redes TCP/IP, uso de locomoción Ackerman, usando como impulso un motor de combustión interna. En el siguiente capítulo se presentaran las partes que componen la propuesta del prototipo, el sistema electromecánico, software y hardware que se utiliza para la comunicación, control y acondicionamiento de las señales.