

UNIVERSIDAD NACIONAL AUTÓNOMA  
DE MEXICO

---

---

Facultad de Ingeniería



"Implantación de un sistema neuronal  
para seguimiento de rutas"

Tesis

Que para obtener el título de:  
Ingeniero Eléctrico Electrónico  
presenta:

Pineda Villegas Andy Steve

e

Ingeniera en Telecomunicaciones  
presenta:

Ramos Torres Laura Karina

Asesor: Dr. Héctor Benítez Pérez



México, D. F.

2002

TESIS CON  
FALLA DE ORIGEN



Universidad Nacional  
Autónoma de México



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

# PAGINACIÓN DISCONTINUA

## *Agradecimientos*

*Agradecemos a PAPPET IN 106100 por el apoyo recibido para la realización de este trabajo.*

*También agradecemos la dirección del Dr. Héctor Benítez Pérez, sin el cual no hubiera sido posible la elaboración de este proyecto, así como su paciencia e invaluable amistad.*

*Por último, queremos agradecer al Dr. Jesús Savage Carmona por darnos su confianza y brindarnos la oportunidad de trabajar en el laboratorio de Interfaces Inteligentes (Facultad de Ingeniería, UNAM), lo cual fue el inicio de un trabajo que concluyó con el desarrollo de este proyecto.*

## *Dedicatorias*

*Andy:*

*A Pedro y María, mis padres, a Alexander y Johann, mis hermanos, por su apoyo, confianza y sobre todo, por creer en mí.*

*A Laura por haberme tenido confianza y paciencia.*

*Laura:*

*A mis padres y mi hermana por todo su cariño, su apoyo, su confianza y por ser la base de todos mis éxitos.*

*A Andy por ser un compañero de trabajo muy confiable y responsable.*

# Índice

<b>1</b>	<b>Introducción</b>	<b>1</b>
1.1	Introducción general	1
1.2	Objetivos	2
1.3	Descripción de la tesis por capítulo	2
<b>2</b>	<b>Generalidades</b>	<b>4</b>
2.1	Introducción	4
2.2	Comunicación	4
2.2.1	Fundamentos de la comunicación inalámbrica	6
2.2.2	Sistemas basados en radio	6
2.2.3	Propagación de las ondas de radio	6
2.2.4	Características de una onda de radio	7
2.2.5	Caracterización de la radio propagación	7
2.2.6	Panorama de las redes de información inalámbricas	8
2.3	Sistemas neuronales	9
2.3.1	Redes neuronales biológicas	9
2.3.2	Redes neuronales artificiales	11
2.3.3	Aplicaciones de las redes neuronales	14
2.3.4	Topología o arquitectura de las redes neuronales	15
2.3.5	Mecanismos de "aprendizaje"	17
2.4	Descripción de los sistemas móviles	21
2.5	Visión inicial del sistema	23
2.6	Resumen	24
<b>3</b>	<b>Diseño del sistema</b>	<b>25</b>
3.1	Introducción	25
3.2	Generalidades de la red neuronal	25
3.2.1	Fuzzy c-means	26
3.2.2	Red de funciones de base radial	27
3.2.3	Algoritmo de aprendizaje de mínimos cuadrados ortogonales	28
3.3	Comunicación inalámbrica	30
3.3.1	Protocolo de comunicación	31
3.3.2	Detección del número de nodo de comunicación	33
3.3.3	Transmisión de mensajes	34

3.3.4	Recepción de mensajes	35
3.4	Integración mecánica y eléctrica	36
3.4.1	Sección motriz	37
3.4.2	Sección de sensado	40
3.4.3	Parte eléctrica	43
3.4.4	Armazón	50
3.5	Pruebas preliminares	55
3.5.1	Calibración de la llanta delantera del robot	56
3.5.2	Calibración de los sensores fotorreflectivos	58
3.5.3	Calibración de los sensores táctiles	58
3.5.4	Obtención de la ecuación que determina el desplazamiento del robot hacia adelante	58
3.5.5	Obtención de la ecuación que determina el giro del robot sobre su propio eje	61
3.6	Conclusiones	64
<b>4</b>	<b>Evaluación de la red neuronal</b>	<b>65</b>
4.1	Introducción	65
4.2	Revisión del entrenamiento de la red neuronal	65
4.2.1	Obtención y significado de la información	65
4.2.2	Tratamiento de la información	67
4.2.3	Resultados del entrenamiento de la red neuronal	69
4.3	Evaluación del escenario 1	72
4.4	Evaluación del escenario 2	73
4.5	Evaluación del escenario 3	74
4.6	Evaluación de escenarios con fallas	75
4.7	Conclusiones	77
<b>5</b>	<b>Controlador PID</b>	<b>78</b>
5.1	Introducción	78
5.2	Diseño del controlador PID	79
5.3	Selección de las constantes del controlador PID	80
5.4	Evaluación del controlador para los escenarios con fallas	80
5.5	Respuesta del robot sin controlador	82
5.6	Conclusiones	84
<b>6</b>	<b>Conclusiones</b>	<b>85</b>
<b>Apéndice A</b>	<b>Módulo de comunicación</b>	<b>89</b>
A.1	Propósito de la comunicación	89
A.2	Descripción general	89
A.3	Aplicaciones inalámbricas de baja potencia	90
A.4	Características	91
A.5	Protocolo de comunicación	92
A.6	Consideraciones de la antena	92
A.7	Manejo del ruido interno	92
A.8	Tarjeta de protocolo	93

A.9	Teoría de operación	94
A.9.1	Tarjeta de radio para datos	94
A.9.2	Tarjeta de protocolo	97
A.10	Protocolo firmware	99
A.11	Formatos de mensaje del protocolo de la PC	101
<b>Apéndice B</b>	<b>Revisión del sistema mínimo para el microcontrolador MC68HC11F1</b>	<b>103</b>
B.1	Microcontrolador MC68HC11F1 de Motorola	104
B.1.1	Modos de operación	105
B.2	Descripción del sistema mínimo para el microcontrolador MC68HC11F1 de Motorola	107
B.3	Modificaciones al sistema mínimo	109
<b>Apéndice C</b>	<b>Revisión de componentes mecánicos</b>	<b>111</b>
C.1	Sensores	111
C.1.1	Interruptor óptico y sensor fotorreflectivo	111
C.1.2	Microconmutadores	114
C.2	Motores	115
C.2.1	Motores de corriente directa	115
C.2.2	Motor de pasos	119
<b>Apéndice D</b>	<b>Revisión del pseudo código</b>	<b>126</b>
D.1	Recorrido por el robot móvil de la ruta 1	126
D.2	Entrenamiento de la red neuronal	129
<b>Apéndice E</b>	<b>Descripción de la conexión de puertos</b>	<b>131</b>
E.1	Puerto A	131
E.2	Puerto G	135
E.3	Puerto D	137
<b>Bibliografía</b>		<b>138</b>

# Capítulo 1

## Introducción

### 1.1 Introducción general

El uso de robots móviles se ha vuelto común en distintas áreas, tales como la industria, la medicina, la exploración y hasta en el ámbito del entretenimiento, entre otras más. Uno de los problemas más interesantes con el que se ha enfrentado el diseñador de robots móviles es el seguimiento de rutas y el desempeño de tareas por parte de éstos en ambientes con obstáculos, en donde es importante que los robots móviles puedan evadirlos.

En el Laboratorio de Interfaces Inteligentes (Facultad de Ingeniería, UNAM) se han desarrollado principalmente dos tipos de robots móviles, unos se han empleado para el seguimiento de rutas en ambientes sin obstáculos y otros para desplazarse desde un punto inicial hasta un punto final en un ambiente con obstáculos. Para el caso de los robots que siguen rutas, se han utilizado sensores fotorreflejos y para el otro caso, se ha hecho uso de microconmutadores para detectar los obstáculos.

En general, ambos tipos de robots empleaban dos motores de corriente directa en la parte trasera del robot y una llanta libre en la parte delantera para el movimiento del robot. Uno de los problemas que se presentan con este tipo de diseño es la llanta delantera. Ésta provoca un error en el desplazamiento de los robots, por lo que de antemano, se tiene que trabajar en un controlador o con otro tipo de solución para corregir dicho error.

Se han buscado distintas soluciones para corregir el error ocasionado por la llanta delantera; por ejemplo, el uso de un motor de pasos que controle el giro de la llanta delantera, o el uso de una llanta libre con un mecanismo similar al de un bolígrafo.

Por otro lado, para obtener información acerca del movimiento de los robots móviles, se utilizaban codificadores colocados sobre las flechas de los motores de corriente directa, los cuales tienen una resolución baja.

Con base en estos antecedentes, se decidió diseñar un robot móvil que contara tanto con sensores fotorreflejos como con sensores táctiles, además de contar con un motor de pasos para controlar el movimiento de la llanta delantera, y otro tipo de codificadores con mayor resolución, para obtener mayor información acerca del desplazamiento de los motores de corriente directa.

## 1.2 Objetivos

Se han utilizado distintos métodos para resolver los problemas que presentan los robots móviles en el seguimiento de rutas y evasión de obstáculos, tal como la utilización de redes neuronales. El objetivo de este trabajo es desarrollar un sistema neuronal para el seguimiento de rutas.

- Como objetivo inicial se plantea la construcción de un robot móvil que cuente con sensores fotorreflectivos, los cuales le permitan detectar zonas blancas y negras sobre la superficie en que se desplaza; con sensores táctiles, que le permitan encontrar obstáculos durante su recorrido; y con interruptores ópticos, de los cuales se obtiene información acerca del recorrido del robot; además de contar con un módulo de comunicación que le permita comunicarse inalámbricamente con una computadora.
- Como segundo objetivo se desarrollará una red neuronal basada en el algoritmo de entrenamiento de mínimos cuadrados ortogonales.
- Por último, se presentará la integración de la red neuronal aplicada al seguimiento de rutas que realice el robot móvil.

## 1.3 Descripción de la tesis por capítulo

Este trabajo consta de seis capítulos y cinco apéndices. El presente capítulo presenta una introducción general a este trabajo y los objetivos del mismo.

En el segundo capítulo se desarrolla una revisión bibliográfica de los tópicos que se emplean en el desarrollo de este proyecto. Los temas que se cubren son la comunicación, las redes neuronales, los sistemas móviles y, finalmente, se presenta una visión inicial del sistema.

El tercer capítulo detalla el diseño del sistema. Lo primero que se presenta en dicho capítulo es el tipo de red neuronal a usar, así como el tipo de entrenamiento que se emplea; posteriormente se hace una descripción de la comunicación inalámbrica y del protocolo de los módulos de comunicación. Además, se presenta la construcción de las diferentes etapas que integran al sistema móvil. Finalmente, se muestran los resultados de las pruebas de calibración inicial del sistema móvil.

El entrenamiento y la evaluación de la red neuronal para distintos escenarios se presentan en el capítulo cuarto.

El quinto capítulo despliega los resultados de la combinación de la red neuronal con un controlador PID (proporcional, integral y derivativo). Éstos se presentan bajo un ambiente de simulación del sistema.

El sexto y último capítulo presenta las conclusiones finales de este trabajo, así como propuestas para trabajos futuros.

Los apéndices contienen información técnica de los elementos que comprenden al sistema. Así, en el apéndice A se presenta el módulo de comunicación empleado; el apéndice B muestra la revisión del sistema mínimo para el microcontrolador MC68HC11F1; en el apéndice C se hace una revisión general de los componentes mecánicos del robot móvil; en el apéndice D se presenta el pseudo código de algunos programas empleados; y finalmente, en el apéndice E se describe la conexión de los puertos del microcontrolador.

# Capítulo 2

## Generalidades

### 2.1 Introducción

En este capítulo se presenta una revisión bibliográfica de los temas cubiertos en el desarrollo de esta tesis. Principal cuidado se da al estudio de las redes neuronales y los sistemas de comunicación, dado que son éstos los que nos permiten modularizar al sistema móvil. Dicha modularización se basa en la división de tareas según los elementos de procesamiento disponible. Tales elementos son tanto la computadora que se encarga de procesar a la red neuronal, y el sistema móvil, el cual se encarga de definir rutas de aproximación.

Este capítulo consta de las siguientes secciones: esta sección de introducción; la segunda presenta la comunicación; la tercera sección corresponde a los sistemas neuronales; la descripción de los sistemas móviles se presenta en la cuarta sección; en la sección cinco se describe la visión inicial del sistema; y, finalmente, un resumen se presenta en la sección seis.

### 2.2 Comunicación

La comunicación electrónica es la transmisión, recepción y procesamiento de la información usando circuitos electrónicos. La información está definida como conocimiento, experiencia o hechos que pueden presentarse como señales en forma analógica (proporcional o continua), tal como la voz humana, información en video, música, o bien en forma digital (pasos discretos), tal como números en código binario, códigos alfanuméricos, símbolos gráficos o información de bases de datos. Toda la información debe ser convertida en energía electromagnética antes de que pueda ser propagada a través de un sistema electrónico de comunicaciones [Tomasi, 1994].

Un sistema electrónico de comunicaciones está compuesto de tres partes principales: un transmisor, un medio de transmisión y un receptor. El transmisor convierte la información de la fuente original a la forma más adecuada para la transmisión; el medio de transmisión proporciona un medio de conexión del transmisor con el receptor (tal como un conductor metálico, una guía de onda o el espacio libre); y el receptor convierte de nuevo la información recibida a una forma conveniente y la transfiere a su destino. La información

original puede ser originada de diferentes fuentes y puede darse en forma analógica o digital.

La energía electromagnética está distribuida en un rango casi infinito de frecuencias. El espectro de frecuencias electromagnético muestra la localización aproximada de varios servicios dentro de la banda que se muestra en la Fig. 2.1. Se puede ver que el espectro de frecuencia se extiende desde las frecuencias subsónicas (unos cuantos hertz) hasta los rayos cósmicos ( $10^{22}$  hertz). Cada banda de frecuencias tiene una característica única que la hace diferente de las otras bandas.

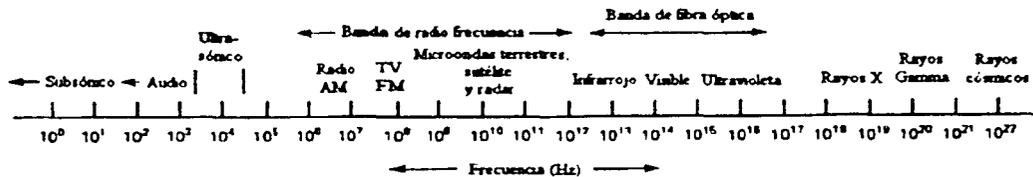


Fig. 2.1 Espectro de frecuencias electromagnético.

En la tabla 2.1 se presentan las designaciones y el rango de frecuencias para cada una de las bandas que conforman el espectro de frecuencias electromagnético.

Número de banda	Rango de frecuencias	Designación
2	30 - 300 Hz	ELF (frecuencias extremadamente bajas)
3	0.3 - 3 kHz	VF (frecuencias de voz)
4	3 - 30 kHz	VLF (frecuencias muy bajas)
5	30 - 300 kHz	LF (frecuencias bajas)
6	0.3 - 3 MHz	MF (frecuencias medias)
7	3 - 30 MHz	HF (frecuencias altas)
8	30 - 300 MHz	VHF (frecuencias muy altas)
9	0.3 - 3 GHz	UHF (frecuencias ultra altas)
10	3 - 30 GHz	SHF (frecuencias super altas)
11	30 - 300 GHz	EHF (frecuencias extremadamente altas)
12	0.3 - 3 THz	Luz infrarroja
13	3 - 30 THz	Luz infrarroja
14	30 - 300 THz	Luz infrarroja
15	0.3 - 3 PHz	Luz visible
16	3 - 30 PHz	Luz ultravioleta
17	30 - 300 PHz	Rayos X
18	0.3 - 3 EHz	Rayos gamma
19	3 - 30 EHz	Rayos cósmicos

Tabla 2.1 Banda de designaciones de la CCIR<sup>1</sup>.

<sup>1</sup> Comité Consultativo Internacional de Radio Frecuencia

### **2.2.1 Fundamentos de la comunicación inalámbrica**

Probablemente una de las más interesantes tecnologías en la industria hoy en día es el mundo inalámbrico. Los sistemas inalámbricos han existido, por décadas, en variaciones de usos y técnicas. Las transmisiones inalámbricas constituyen una potente herramienta de transferencia de información; por ejemplo, se pueden emplear para comunicar estaciones de trabajo a través de ondas de radio permitiendo la movilidad y flexibilidad del sistema en general [Feher, 1995].

### **2.2.2 Sistemas basados en radio**

Así como emergieron los sistemas basados en radio, las comunicaciones inalámbricas comenzaron a ser disponibles y fáciles de usar. Se utilizó un transmisor eléctrico para enviar las ondas de sonido y modular la voz humana en una frecuencia de radio de banda base. La onda de radio portaba la señal transmitida y podía viajar grandes distancias, permitiendo más veracidad y minimizando el proceso de retransmisión. Varias porciones del espectro de frecuencia de radio fueron localizadas (asignadas) a estos sistemas de transmisión. Cada portadora tiene sus propias capacidades y sus limitaciones de distancia asociadas con la banda (rango de frecuencias) usadas para su transmisión [Feher, 1995].

El conocimiento de los principios de radio es esencial para comprender cómo funcionan varias técnicas de comunicación inalámbrica. En sistemas de telecomunicaciones modernos (particularmente en el ámbito de la telefonía) las ondas de radio son una parte integral de la red de desarrollo. Tal como los sistemas más modernos han emergido, ha habido modificaciones y avances que permiten a las redes transportar todo tipo de comunicación, incluyendo voz, datos, imágenes, fax y video.

### **2.2.3 Propagación de las ondas de radio**

Las porciones del espectro electromagnético que constituyen las ondas de radio se extienden desde una frecuencia de 30 kHz (longitud de onda de 10 km) hasta 300 GHz (longitud de onda de 0.1 cm) [Feher, 1995].

La energía electromagnética en forma de ondas de radio se propaga hacia el exterior mediante una antena transmisora. En el espacio libre, las ondas de radio se propagan en línea recta, y es por esto que el transmisor y el receptor deben "verse" el uno al otro, es decir, deben tener una línea de vista.

Las ondas de radio sobre la tierra son afectadas por el terreno, la atmósfera y los objetos naturales y artificiales que se encuentran sobre la tierra.

La onda terrestre es esa porción de la radiación que es directamente afectada por la tierra y por los objetos que se encuentran sobre la tierra. La onda terrestre opera en todas las frecuencias, desde muy bajas hasta frecuencias de microondas.

Debido a las características de propagación de rango corto de VHF y UHF, es posible asignar la misma frecuencia a diferentes usuarios móviles en áreas separadas por distancias de 50 a 100 km, con un alto grado de confianza de que la señal no interferirá con alguna otra, excepto bajo condiciones de propagación anormales, siempre y cuando la potencia de la señal sea baja y la altura de la antena sea restringida.

#### **2.2.4 Características de una onda de radio**

Las ondas de radio se propagan a través del espacio como ondas electromagnéticas. La energía de las señales existe en forma de campos eléctricos y campos magnéticos. Ambos campos siempre existen juntos porque un cambio en el campo eléctrico genera un campo magnético y un cambio en el campo magnético genera un campo eléctrico. Por lo tanto, existe un flujo de energía continuo de un campo hacia el otro [Feher, 1995].

Las ondas de radio llegan a un receptor móvil de diferentes direcciones con diferentes retardos de tiempo. Se combinan mediante un proceso que puede modelarse como una adición vectorial en la antena receptora para obtener una señal resultante con una amplitud grande o pequeña, dependiendo si la onda entrante se combina para reforzar a la otra, o si se cancelan mutuamente. Como resultado, un receptor en una determinada ubicación puede recibir la fuerza de una señal con varias decenas de decibeles diferentes de un receptor similar localizado a una distancia más corta. Como un sistema móvil no permanece en un lugar fijo y se mueve de una ubicación a otra, la relación de fase entre varias ondas entrantes también cambia. Por lo tanto, hay fluctuaciones sustanciales tanto en amplitud como en fase, lo que genera una atenuación.

#### **2.2.5 Caracterización de la radio propagación**

El diseño efectivo, la valoración y la instalación de una red de radio requiere de una caracterización precisa del canal. Las características del canal varían de un ambiente a otro, y las características particulares determinan la factibilidad de usar una técnica de comunicación en un ambiente de operación dado. Los modelos de canal también son empleados para determinar la localización óptima para la instalación de antenas y para analizar la interferencia entre diferentes sistemas [Pahlavan y Levesque, 1994].

Las redes inalámbricas operan desde frecuencias en el rango de unas cuantas centenas de kilohertz hasta unas cuantas de centenas de gigahertz. Sin embargo, los nuevos sistemas y servicios que están emergiendo operan en un rango de alrededor de 900 megahertz a unos cuantos gigahertz.

Las frecuencias en la región de las unidades de gigahertz tienen varias características atractivas para usarse en las redes de información inalámbricas. A estas frecuencias un transmisor con una potencia menor a 1 W puede proveer una cobertura de varios pisos dentro de un edificio, y si es usado en el exterior puede cubrir distancias en el orden de unidades de millas. Además, a estas frecuencias el tamaño de una antena eficiente puede estar en el orden de una pulgada. A bajas frecuencias, el ancho de banda es estrecho y se

requiere de antenas más grandes y de separaciones más grandes entre ellas, además hay altos niveles de interferencia debido a ruido. A altas frecuencias, el ancho de banda es amplio pero sufre de mayor atenuación en la transmisión a través de las paredes. Para frecuencias en la región de decenas de gigahertz, la propagación de la señal es confinada por las paredes de una habitación y esto restringe su aplicación para algunos sistemas. Sin embargo, desde el punto de vista de seguridad, esta característica puede ser atractiva.

La radio propagación en ambientes tanto interiores como exteriores, es complicada por la distancia tan corta que hay entre el transmisor y el receptor, ya que usualmente está bloqueada por paredes, techos y otros objetos en un espacio interior o por construcciones y características del terreno en el exterior. Por lo tanto, la potencia de la señal es típicamente transportada desde el transmisor hasta el receptor mediante una multiplicidad de rutas. Los tiempos de llegada de la señal sobre varias rutas son proporcionales a las longitudes de las rutas, las cuales a su vez son afectadas por el tamaño y la arquitectura del ambiente, así como también por la localización de los objetos alrededor del transmisor y del receptor. La intensidad en cada una de las rutas depende de la atenuación causada por el paso de la señal a través de varios objetos en la ruta (o por reflexión de la señal).

## **2.2.6 Panorama de las redes de información inalámbricas**

En las últimas décadas se ha venido dando una revolución dentro de las comunicaciones; antes el modo dominante de comunicaciones electrónicas era el servicio telefónico estándar y las comunicaciones de datos, y ahora, el ambiente de comunicaciones móviles es una realidad. Las redes de información inalámbrica, que incluyen sistemas inalámbricos y teléfonos celulares, las redes de datos móviles y los sistemas móviles satelitales, han experimentado un enorme crecimiento en la última década [Pahlavan y Levesque, 1994].

La industria de las comunicaciones inalámbricas es una de tantas que continuará con los beneficios de la introducción de una tecnología digital. Mientras la demanda de servicios de comunicaciones aumente, los fabricantes y los proveedores de servicios buscarán implementaciones digitales para ofrecer más y mejores servicios a sus usuarios.

Se han expandido las comunicaciones de datos en el sector de los negocios, en la industria y en el gobierno, así como para uso personal y cultural. Tal como los usuarios se han acostumbrado a los servicios de comunicaciones de voz en forma inalámbrica, también esperarán tener la misma conveniencia con servicios de datos en ambientes móviles e inalámbricos.

En el presente trabajo se utiliza la comunicación inalámbrica para el acoplamiento de un módulo móvil y un sistema central de procesamiento. Esta opción se toma como base de comunicación para dar flexibilidad a los movimientos del sistema autónomo.

## 2.3 Sistemas neuronales

El desarrollo de las redes neuronales artificiales comenzó hace 50 años aproximadamente, motivado por el deseo de entender el cerebro humano como de imitar algunas de sus potencialidades. El reciente interés renovado en las redes neuronales puede ser atribuido a varios factores, por ejemplo, las técnicas de entrenamiento han sido desarrolladas por arquitecturas de red más sofisticadas, y las altas velocidades de las computadoras actuales hacen que la simulación de los procesos neuronales sean más factibles. Sin embargo, al mismo tiempo que los progresos en la computación tradicional han hecho más sencillo el estudio de las redes neuronales, las limitaciones encontradas en la inherente naturaleza secuencial de la computación tradicional han motivado nuevas líneas de investigación de las redes neuronales [Fauset, 1994].

Las redes neuronales artificiales son de interés para investigadores en muchas áreas por diferentes razones. Los ingenieros eléctricos encuentran numerosas aplicaciones en el procesamiento de señales y en la teoría de control. Los ingenieros en computación se ocupan de analizar el potencial de implementar redes neuronales eficientemente en hardware y buscan aplicaciones dentro de la robótica. Los científicos en computación encuentran que las redes neuronales pueden resolver problemas en áreas tales como inteligencia artificial y reconocimiento de patrones. Para los matemáticos prácticos, las redes neuronales son una poderosa herramienta para el modelado de problemas.

Las redes neuronales son, básicamente, modelos matemáticos de procesamiento de información. Como otros métodos matemáticos, las redes neuronales son muy eficientes para la aproximación, especialmente para problemas complejos.

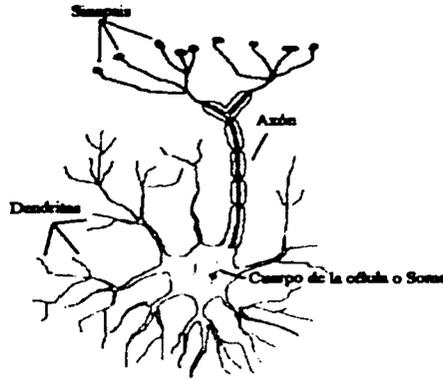
### 2.3.1 Redes neuronales biológicas

La teoría y modelado de redes neuronales está inspirada en la estructura y funcionamiento de los sistemas nerviosos, donde la neurona es el elemento fundamental. Existen neuronas de diferentes formas, tamaños y longitudes [Hilera y Martínez, 1995].

Una neurona es una célula viva y, como tal, contiene los mismos elementos que forman parte de todas las células biológicas. Además, contiene elementos característicos que las diferencian. En general, una neurona consta de un cuerpo celular (soma) más o menos esférico, del que salen una rama principal, el axón, y varias ramas más cortas, llamadas dendritas. A su vez, el axón puede producir ramas en torno a su punto de arranque, y con frecuencia se ramifica extensamente cerca de su extremo. En la Fig. 2.2 se ilustra una neurona con sus elementos principales.

Una de las características que diferencian a las neuronas del resto de las células vivas, es su capacidad de comunicarse. En términos generales, las dendritas y el cuerpo celular reciben señales de entrada; el cuerpo celular las combina e integra y emite señales de salida. El axón transporta esas señales a las terminales axónicas que se encargan de distribuir la información a un nuevo conjunto de neuronas. Por lo general, una neurona recibe

información de miles de neuronas y, a su vez, envía información a miles de neuronas más. Se calcula que en el cerebro humano existen del orden de  $10^{15}$  conexiones.



**Fig. 2.2 Esquema general de una neurona.**

Las señales que se utilizan son de dos tipos: eléctricas y químicas. La señal generada por la neurona y transportada a lo largo del axón es un impulso eléctrico, mientras que la señal que se transmite entre las terminales axónicas de una neurona y las dendritas de las neuronas siguientes es de origen químico; concretamente, se realiza mediante moléculas de sustancias transmisoras (neurotransmisores) que fluyen a través de unos contactos especiales, llamados sinapsis, que tienen la función de receptores, y están localizados entre las terminales axónicas y las dendritas de la neurona siguiente.

La generación de las señales eléctricas está íntimamente relacionada con la composición de la membrana celular. Existen muchos procesos complejos relacionados con la generación de dichas señales; sin embargo, se pueden simplificar del siguiente modo: las neuronas están compuestas de un líquido que difiere de la composición del líquido exterior, que produce una variación de potencial, llamada potencial de reposo. La llegada de señales provenientes de otras neuronas reduce el potencial de reposo hasta llegar a un cierto valor crítico, el cual provoca una inversión del voltaje, conocida como potencial de acción, que se propaga a lo largo del axón y, a su vez, provoca la emisión de los neurotransmisores a las terminales axónicas.

Este proceso puede repetirse, y el resultado de todo esto es la emisión de trenes de pulsos por parte de la neurona.

Existen dos tipos de sinapsis: las sinapsis excitadoras y las sinapsis inhibitoras. Las sinapsis excitadoras son aquellas cuyos transmisores provocan disminuciones de potencial en la membrana de la célula postsináptica, facilitando la generación de impulsos a mayor velocidad. Las sinapsis inhibitoras son aquellas cuyos neurotransmisores tienden a estabilizar el potencial de la membrana, dificultando la emisión de impulsos. Casi todas las neuronas reciben entradas procedentes de sinapsis excitadoras e inhibitoras. En cada

instante, algunas de ellas estarán activas y otras se hallarán en reposo; la suma de los efectos excitadores e inhibidores determina si la célula será o no estimulada, es decir, si emitirá o no un tren de impulsos, y a qué velocidad.

Para establecer una similitud directa entre la actividad sináptica y la analogía con las redes neuronales artificiales, se fijan los siguientes aspectos: las señales que llegan a las sinapsis son las entradas a la neurona; éstas son ponderadas (atenuadas o amplificadas) a través de un parámetro, denominado peso, asociado a la sinapsis correspondiente. Estas señales de entrada pueden excitar a la neurona (sinapsis con peso positivo) o inhibirla (peso negativo). El efecto es la suma de las entradas ponderadas. Si la suma es mayor o igual que el umbral de la neurona, entonces la neurona se activa (da salida). La facilidad de transmisión de señales se altera mediante la actividad del sistema nervioso. Algunos eventos pueden incrementar o disminuir el grado de activación; esta habilidad de ajustar señales es un mecanismo de aprendizaje. Las funciones umbral integran la energía de las señales de entrada en el espacio y en el tiempo.

La forma en que una red neuronal artificial modela un sistema neuronal biológico particular varía. Para algunos investigadores, este modelo es un aspecto medular; para otros, la habilidad de la red para realizar tareas útiles (tales como aproximar o representar una función) es más importante que la verosimilitud biológica de la red.

### 2.3.2 Redes neuronales artificiales

Existen numerosas formas de definir lo que son las redes neuronales, desde las definiciones cortas y genéricas hasta las que intentan explicar más detalladamente lo que significa red neuronal. Algunas definiciones son las siguientes [Hilera y Martínez, 1995]:

- *“Una nueva forma de computación, inspirada en modelos biológicos”.*
- *“Un modelo matemático compuesto por un gran número de elementos de procesamiento organizados en niveles”.*
- *“... un sistema de computación hecho por un gran número de elementos simples, elementos de proceso muy interconectados, los cuales procesan información por medio de su estado dinámico como respuesta a entradas externas” [Heicht-Niesen, 1988].*
- *“Redes neuronales artificiales son redes interconectadas masivamente en paralelo en elementos simples (usualmente adaptativos) y con organización jerárquica, las cuales intentan interactuar con los objetos del mundo real del mismo modo que lo hace el sistema nervioso biológico” [Kohonen, 1988].*

Una red neuronal artificial es un sistema de procesamiento de información que tiene ciertas características de funcionamiento en común con las redes neuronales biológicas. Las redes neuronales artificiales han sido desarrolladas como generalizaciones de modelos matemáticos del conocimiento humano o de la biología neuronal, basado en las siguientes suposiciones [Fauset, 1994]:

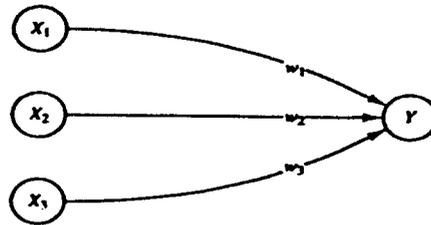
- 1 El procesamiento de la información ocurre en unos elementos simples llamados neuronas.
- 2 Las señales pasan de neurona a neurona a través de líneas (ligas) de conexiones.
- 3 Cada liga de conexión tiene asociado un peso, el cual, en una red neuronal típica, se multiplica con la señal transmitida.
- 4 Cada neurona aplica una función de activación (usualmente no lineal) hacia su red de entrada (la suma de las señales de entrada multiplicadas por los pesos) para determinar su señal de salida.

Una red neuronal está caracterizada por su patrón de conexiones entre las neuronas (llamada arquitectura), su método para determinar los pesos en las conexiones (llamado entrenamiento, aprendizaje o algoritmo) y su función de activación.

Una red neuronal consiste de un gran número de elementos de procesamiento simples llamados neuronas, unidades, células o nodos. Cada neurona está conectada a otras neuronas por medio de ligas de comunicación, cada una con un peso asociado. Los pesos representan la información que es usada por la red para resolver un problema. Las redes neuronales pueden ser aplicadas a una gran variedad de problemas, tales como almacenamiento de patrones, clasificación de patrones, creación de mapeos generales de un patrón de entrada a un patrón de salida, agrupando patrones similares o resolución de problemas de optimización restringidos.

Cada neurona tiene un estado interno, llamado activación o nivel de actividad, el cual es una función de las entradas que ha recibido. Típicamente, una neurona envía su activación como una señal a diferentes neuronas. Es importante notar que una neurona puede enviar solamente una señal en un tiempo, aunque esa señal sea transmitida a varias neuronas.

En la Fig. 2.3 se muestra el esquema de una neurona artificial simple. En dicha figura, las  $X_i$  representan las entradas a la neurona, las  $w_i$  representan los pesos (asociados a las sinapsis correspondientes) y  $Y$  representa la salida de la neurona.



**Fig. 2.3 Esquema de una neurona artificial simple.**

Debido a su constitución y a sus fundamentos, las redes neuronales artificiales presentan un gran número de características semejantes a las del cerebro. Por ejemplo, son capaces de aprender de la experiencia, de generalizar de casos anteriores a nuevos casos, de abstraer características esenciales a partir de entradas que representan información irrelevante. Esto hace que ofrezcan numerosas ventajas y que este tipo de tecnología se esté aplicando en múltiples áreas. Entre otras ventajas, se incluyen las siguientes [Hilera y Martínez, 1995]:

- Aprendizaje adaptativo. Capacidad de aprender a realizar tareas basadas en un entrenamiento o experiencia inicial.
- Auto-organización. Una red neuronal puede crear su propia organización o representación de la información que recibe mediante una etapa de aprendizaje.
- Tolerancia a fallas. La destrucción parcial de una red conduce a una degradación de su estructura; sin embargo, algunas capacidades de la red se pueden retener, incluso sufriendo un gran daño.
- Operación en tiempo real. Los computadores neuronales pueden ser realizados en paralelo, y se diseñan y fabrican máquinas con hardware especial para obtener esta capacidad.

Varias características principales del procesamiento de elementos de las redes neuronales artificiales son sugeridas por las propiedades de las neuronas biológicas, tales como [Fauset, 1994]:

- 1 Los elementos de procesamiento reciben señales.
- 2 Las señales pueden ser modificadas por un peso en la sinapsis receptora.
- 3 El elemento de procesamiento suma las entradas con sus respectivos pesos.
- 4 Bajo condiciones apropiadas, la neurona transmite una salida simple.

- 5 La salida de una neurona en particular puede ir a muchas otras neuronas (las ramificaciones del axón).
- 6 El procesamiento de la información es local (aunque otros medios de transmisión, tales como la acción de las hormonas, sugiere medios de un conjunto de procesos de control).
- 7 La memoria está distribuida en:
  - a. memoria de largo término que reside en las sinapsis o pesos de la neurona;
  - b. memoria de corto término que corresponde a las señales enviadas por las neuronas.
- 8 La intensidad de una sinapsis puede ser modificada por la experiencia.
- 9 Los neurotransmisores para las sinapsis pueden ser excitatorios o inhibidores.

Otra característica que las redes neuronales artificiales comparten con los sistemas neuronales biológicos es la tolerancia a fallas. Los sistemas neuronales biológicos son tolerantes a fallas en dos aspectos: primero, nosotros somos capaces de reconocer señales de entradas que son ligeramente diferentes de otra señal que hemos visto antes. Un ejemplo de esto es nuestra habilidad de reconocer a una persona en una fotografía que no hemos visto antes o reconocer a una persona después de un largo periodo de tiempo. Segundo, nosotros somos capaces de tolerar los daños en nuestro sistema neuronal, porque a pesar de la continua pérdida de neuronas, seguimos aprendiendo.

### 2.3.3 Aplicaciones de las redes neuronales

El estudio de las redes neuronales se encuentra en un campo extremadamente interdisciplinario, tanto en el desarrollo como en su aplicación. A continuación se mencionan solamente algunas de las aplicaciones [Hilera y Martínez, 1995]:

- Procesamiento de señales

Existen varias aplicaciones de las redes neuronales en el área general de procesamiento de señales. Una de las primeras aplicaciones comerciales fue suprimir el ruido a la línea telefónica.

- Control

Dentro del área de control, existen un sinnúmero de aplicaciones. Como ejemplo se tiene el controlar el movimiento de un trailer cargado que se mueve en reversa para poder acomodarse en un lugar dado, tarea que resulta muy complicada para un conductor inexperto.

- Reconocimiento de patrones

Muchos problemas interesantes caen dentro del área de reconocimiento de patrones. Un área específica en la cual muchas aplicaciones de redes neuronales han sido desarrolladas es el reconocimiento automático de los caracteres escritos (dígitos o letras). La gran variación de tamaños, posiciones y estilos hacen de éste, un problema difícil para las técnicas tradicionales.

- Medicina

Una de las aplicaciones dentro de la medicina fue entrenar una red neuronal de memoria autoasociativa, la cual almacenara una gran cantidad de reportes médicos, cada uno de ellos incluye información de síntomas, diagnósticos y tratamientos para un caso en particular. Después del entrenamiento, la red es presentada con entradas consistentes de síntomas, a partir de las que la red encontrará el patrón completo almacenado que representa el “mejor” diagnóstico y tratamiento.

### 2.3.4 Topología o arquitectura de las redes neuronales

Es conveniente visualizar a las neuronas como arreglos en capas. Típicamente, las neuronas en la misma capa tienen el mismo comportamiento. Los factores principales para determinar el comportamiento de una neurona son su función de activación y el patrón de conexiones ponderadas sobre las cuales se envían o reciben señales. En cada capa, las neuronas usualmente tienen la misma función de activación y el mismo patrón de conexiones a otras neuronas. En muchas redes neuronales, las neuronas en una capa están total o parcialmente interconectadas [Hilera y Martínez, 1995].

El arreglo de neuronas en las capas, y los patrones de conexión en y entre las diferentes capas es llamado arquitectura de la red. Muchas redes neuronales tienen una capa de entrada en la cual la activación de cada unidad es igual al reconocimiento de una señal de entrada externa.

Las redes neuronales son frecuentemente clasificadas como monocapa (una capa simple) o multicapa. Para determinar el número de capas, las unidades de entrada no son contadas como una capa, porque no forman parte del cálculo. Equivalentemente, el número de capas en la red puede estar definido por el número de capas ponderadas interconectadas entre los extremos de las neuronas de entrada y salida.

Tanto la red monocapa como la red multicapa son ejemplos de redes hacia adelante (*feedforward*), las cuales son redes en las que las señales fluyen desde las unidades de entrada hasta las unidades de salida en forma progresiva.

## Red monocapa

Una red monocapa tiene una capa de conexión de pesos,  $w_{xy}$ . Generalmente, las unidades pueden ser distinguidas como unidades de entrada,  $X_i$ , las cuales reciben señales desde el mundo exterior, y unidades de salida,  $Y_i$ , en donde puede ser leída la respuesta de la red. La Fig. 2.4 muestra un esquema de una red neuronal monocapa.

En una red monocapa típica, las unidades de entrada son totalmente conectadas a las unidades de salida, pero no están conectadas a otras unidades de entrada, y las unidades de salida no están conectadas a otras unidades de salida.

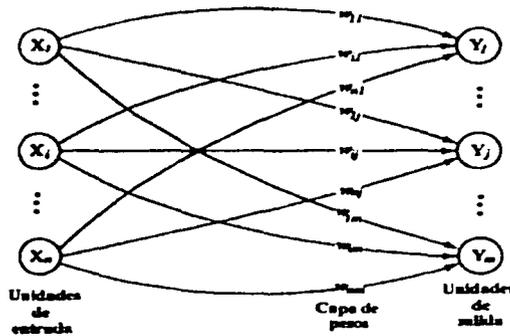


Fig. 2.4 Red neuronal monocapa.

Para la clasificación de patrones, cada unidad de salida corresponde a una categoría particular, en la cual, un vector de entrada puede pertenecer o no a dicha salida. Nótese que para una red monocapa, los pesos para una unidad de salida no interfieren en los pesos de las unidades de salida. Para la asociación de patrones, puede ser usada la misma arquitectura, pero ahora todo el patrón de las señales de salida da la respuesta del patrón asociado con la señal de entrada. Estos dos ejemplos ilustran el hecho de que el mismo tipo de red puede ser usado para diferentes problemas, dependiendo de la interpretación de la respuesta de la red.

Por otra parte, problemas de mapeo más complicados podrían requerir una red multicapa.

## Red multicapa

Una red multicapa es una red con una o más capas (o niveles) de nodos (unidades ocultas),  $Z_i$ , entre las unidades de entrada,  $X_i$ , y las unidades de salida,  $Y_i$ . Típicamente, hay una capa de pesos,  $w_{11}$ , entre dos niveles adyacentes de unidades (de entrada, ocultas o de salida).

Las redes multicapa pueden resolver problemas más complicados que las redes monocapa, pero el entrenamiento puede ser más difícil.

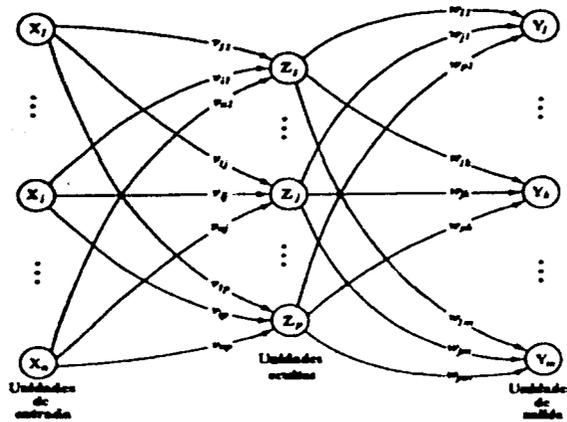


Fig. 2.5 Red neuronal multicapa.

### 2.3.5 Mecanismos de “aprendizaje”

Además de la arquitectura, el método para fijar los valores de los pesos (entrenamiento) es una característica importante para distinguir a las redes neuronales. Por convención, se distinguen dos tipos de entrenamiento para una red neuronal: supervisado y no supervisado; también existen redes en las cuales sus pesos son fijados sin necesidad de un proceso de entrenamiento iterativo.

El aprendizaje es el proceso por el cual una red neuronal modifica sus pesos en respuesta a una información de entrada. Los cambios que se producen durante el proceso de aprendizaje se reducen a la destrucción, modificación y creación de conexiones entre las neuronas. En los modelos de redes neuronales artificiales, la creación de una nueva conexión implica que el peso de la misma pasa a tener un valor distinto de cero. De la misma forma, una conexión se destruye cuando su peso pasa a ser cero [Hilera y Martínez, 1995].

Durante el proceso de aprendizaje, los pesos de las conexiones de la red sufren modificaciones, por tanto se puede afirmar que este proceso ha terminado (la red ha “aprendido”) cuando los valores de los pesos permanecen sin cambio dentro de un intervalo de tiempo correspondiente a la evaluación.

Un aspecto importante respecto al aprendizaje en las redes neuronales es el conocer cómo se modifican los valores de los pesos, es decir, cuáles son los criterios que se siguen para cambiar el valor asignado a las conexiones cuando se pretende que la red “aprenda” una nueva información.

Estos criterios determinan lo que se conoce como la “regla de aprendizaje” de la red. De forma general, se suelen considerar dos tipos de reglas: las que responden a lo que

habitualmente se conoce como “aprendizaje” supervisado, y las correspondientes a un “aprendizaje” no supervisado. Es por ello que una de las clasificaciones que se realizan de las redes neuronales obedece al tipo de “aprendizaje” utilizado por dichas redes. Así se pueden distinguir:

- Redes neuronales con “aprendizaje” supervisado.
- Redes neuronales con “aprendizaje” no supervisado.

La diferencia fundamental entre ambos tipos estriba en la existencia o no de un agente externo (supervisor) que controle el proceso de “aprendizaje” de la red.

Otro criterio que se puede utilizar para diferenciar las reglas de “aprendizaje” se basa en considerar si la red puede “aprender” durante su funcionamiento habitual o si el “aprendizaje” supone la desconexión de la red, es decir, su inhabilitación hasta que el proceso termine. En el primer caso, se trata de un “aprendizaje” en línea (*on line*), mientras que el segundo es lo que se conoce como “aprendizaje” fuera de línea (*off line*).

Cuando el “aprendizaje” es fuera de línea, se distingue entre una fase de “aprendizaje” o entrenamiento y una fase de operación o funcionamiento, existiendo un conjunto de datos de entrenamiento y un conjunto de datos de prueba que serán utilizados en la fase correspondiente. En las redes con “aprendizaje” fuera de línea, los pesos de las conexiones permanecen fijos después que termina la etapa de entrenamiento de la red. Debido precisamente a su carácter estático, estos sistemas no presentan problemas de estabilidad en su funcionamiento, si se trabaja en la etapa de prueba con escenarios similares a los usados en el entrenamiento.

En las redes de “aprendizaje” en línea no se distingue entre fase de entrenamiento y de operación, de tal forma que los pesos varían dinámicamente siempre que se presente una nueva información al sistema. En este tipo de redes, debido a su carácter dinámico, el estudio de la estabilidad suele ser un aspecto fundamental de estudio.

### **Redes con “aprendizaje” (entrenamiento) supervisado**

El “aprendizaje” supervisado se caracteriza porque su proceso se realiza mediante un entrenamiento controlado por un agente externo (supervisor o maestro) que determina la respuesta que debería generar la red a partir de una entrada determinada. El supervisor comprueba la salida de la red y en el caso de que ésta no coincida con la deseada, se procederá a modificar los pesos de las conexiones, con el fin de conseguir que la salida obtenida se aproxime a la deseada.

En este tipo de aprendizaje se suelen considerar, a su vez, tres formas de llevarlo a cabo que dan lugar a los siguientes aprendizajes supervisados:

- “Aprendizaje” por corrección de error.

- “Aprendizaje” por refuerzo.
- “Aprendizaje” estocástico.

En la tabla 2.2 se muestra una lista de los tipos de redes más conocido que emplean aprendizaje supervisado.

Tipo de aprendizaje Supervisado		Modelo de red
Aprendizaje por corrección de error	Fuera de línea	Perceptron
		Adaline / Madaline
		Backpropagation
		Brain-State-in-a-Box
		Counterpropagation
Aprendizaje por refuerzo	En línea	Linear Reward Penalty
		Associative Reward Penalty
		Adaptive Heuristic Critic
Aprendizaje estocástico	Fuera de línea	Boltzmann Machine
		Cauchy Machine

**Tabla 2.2 Tipos de redes con aprendizaje supervisado más conocidos.**

**Redes con aprendizaje (entrenamiento) no supervisado**

Las redes con “aprendizaje” no supervisado (también conocido como autosupervisado) no requieren influencia externa para ajustar los pesos de las conexiones entre sus neuronas. La red no recibe ninguna información por parte del entorno que le indique si la salida generada en respuesta a una determinada entrada es o no correcta; por ello, suele decirse que estas redes son capaces de “autoorganizarse”.

Estas redes deben encontrar las características, regularidades, correlaciones o categorías que se puedan establecer entre los datos que se presenten en su entrada. Puesto que no hay un supervisor que indique a la red la respuesta que debe generar ante una entrada concreta, cabría analizar los resultados que la red produce en estos casos. Existen varias posibilidades en cuanto a la interpretación de la salida de estas redes, que dependen de su estructura y del algoritmo de aprendizaje empleado.

En algunos casos, la salida representa el grado de familiaridad o similitud entre la información que se le está presentando en la entrada y las informaciones que se le han mostrado hasta entonces (en el pasado). En otro caso, podría realizar un agrupamiento o establecimiento de categorías, indicando la red a la salida a qué categoría pertenece la información presentada a la entrada, siendo la propia red quien debe encontrar las categorías apropiadas a partir de correlaciones entre las informaciones presentadas. Una variación de esta categorización es el uso de prototipos. En este caso, la red obtiene ejemplares o prototipos representantes de las clases a las que pertenecen las informaciones de entrada.

También el “aprendizaje” sin supervisión permite realizar una codificación de los datos de entrada, generando a la salida una versión codificada de la entrada, con menos bits, pero manteniendo la información relevante de los datos.

Finalmente, algunas redes con “aprendizaje” no supervisado realizan un mapeo de características, obteniéndose en la neurona de salida una disposición geométrica que representa un mapa topográfico de las características de los datos de entrada, de tal forma que si se presentan a la red informaciones similares, siempre sean afectadas neuronas de salida próximas entre sí, en la misma zona del mapa.

En cuanto a los algoritmos de “aprendizaje” no supervisado, en general, se suelen considerar dos tipos que dan lugar a los siguiente “aprendizajes”:

- “Aprendizaje” hebbiano.
- “Aprendizaje” competitivo y cooperativo.

En el primer caso, normalmente se pretende medir la familiaridad o extraer características de los datos de entrada, mientras que el segundo suele orientarse hacia la agrupación o clasificación de dichos datos.

En la tabla 2.3 se listan las redes con “aprendizaje” no supervisado más conocidas.

Tipo de aprendizaje no supervisado		Modelo de red
Aprendizaje hebbiano	Fuera de línea	Hopfield
		Learning Matrix
		Temporal Associative Memory
		Linear Asociative Memory (LAM)
		Optimal LAM
		Drive - Reinforcement
	En línea	Fuzzy Associative Memory
		Additive Grossberg
		Shunting Grossberg
		Bidirectional Associative Memory (BAM)
Aprendizaje competitivo / cooperativo	Fuera de línea	Adaptive BAM
		Learning Vector Quantizer
		Cognitron / Nocognitron
	Topology Preserving Map	
En línea	Adaptive Resonance Theory	

**Tabla 2.3 Redes con aprendizaje no supervisado más conocidas.**

El uso de las redes neuronales dentro de este proyecto tiene como objetivo presentar una alternativa computacional para el manejo de rutas con cierto contenido de ruido. Así como su corrección de errores.

## 2.4 Descripción de los sistemas móviles

Desde sus inicios, a principios de los años 60's, la robótica se ha enfocado generalmente en el diseño y control de robots manipuladores, lo cual ha estado motivado por las necesidades de la industria. Mucho del trabajo de investigación dentro de la robótica ha tenido como su meta principal la manufactura de robots industriales que sean más rápidos, más precisos y más fáciles de programar [Cox y Gordon, 1990].

La robótica se ha convertido en una herramienta estándar para uso en sistemas de automatización industrial, principalmente en aquellas tareas repetitivas [Congreso China, 1999].

El trabajo de investigación y desarrollo en el área de la robótica en las dos décadas pasadas ha conducido a una variedad de avances en la práctica industrial: trayectorias cartesianas, lenguajes de programación para robots, sistemas de visión para localización e inspección de objetos, son algunos ejemplos. Estos avances tecnológicos han hecho posible la automatización de nuevas aplicaciones. Sin embargo, siempre ha habido una gran diferencia entre las restricciones de los robots construidos para aplicaciones industriales de gran volumen y los construidos para propósitos de investigación [Congreso México, 2001].

Cabe hacer mención que en años recientes, los robots móviles autónomos han tenido una gran aceptación en diversas disciplinas, pues se ha percibido su utilidad en muchas áreas de investigación tales como la neuro-informática, la inteligencia artificial, la neuro-psicología, la bio-cibernética, dentro de la planeación de rutas, navegación en ambientes con obstáculos, entre otras más. Además de su desempeño en el campo de la investigación, los robots móviles autónomos también tienen un gran campo de aplicación dentro de la industria [Congreso Portugal, 2001].

Los robots deben tener su propia interfaz con su ambiente, en donde puede haber elementos para ser manipulados o evitados. Usualmente, un robot formará parte de un sistema más grande, así que será programado para realizar una serie de tareas específicas. Una característica importante de los robots móviles es que permiten modificar su operación (por medio de programación) de acuerdo a la información que se obtenga de su sistema de sensado.

Los robots móviles autónomos son vehículos que son capaces de tener una acción y un movimiento inteligente sin requerir una guía a seguir o un control operado a distancia. Las aplicaciones potenciales de los vehículos autónomos son muchas e incluyen vehículos para el reconocimiento y exploración tanto en el espacio como bajo el mar, en tierra, o bien, en el aire, también para reparación y mantenimiento remoto, así como sillas de ruedas inteligentes para personas discapacitadas.

En contraste con los manipuladores industriales, los robots móviles autónomos generalmente no se emplean en operaciones que requieran una alta precisión, ya que éstos tienen la peculiaridad de operar con cierta incertidumbre en la determinación de su posición y orientación, así como en la posición e identificación de algún objeto. Por esta razón, una

de las actividades más difíciles para un robot móvil es el desplazamiento desde un punto inicial hasta un punto objetivo.

El modelo de un robot móvil debe incluir información sobre la dimensión, forma, posición y orientación de cada objeto presente en el área de trabajo. En la mayoría de los casos, una gran parte del modelo o mapa del “mundo real” generalmente está almacenado en la memoria del robot antes de que empiece a funcionar, lo cual limita su operación a un ambiente que él “conoce”.

Como se mencionó anteriormente, los robots móviles autónomos deben navegar dentro de un ambiente. Para lograr este objetivo, el vehículo debe ser capaz de sensar su ambiente, interpretar la información sensada para refinar su conocimiento de su posición y la estructura del ambiente, además de planear una ruta desde su estado inicial hasta la posición deseada aún en presencia de algún obstáculo.

La meta de la investigación de los robots móviles autónomos es desarrollar sistemas que sean robustos y confiables en la presencia de muchas fuentes de ruido, además del compromiso implícito de que sean baratos.

Los diferentes campos de operación de los robots móviles tienen prioridades en las áreas de sensado y razonamiento, por lo tanto, los robots móviles deben tener las dos cualidades (o capacidades) siguientes [Cox,y Gordon, 1990]:

- Interpretación de sensado: el robot debe ser capaz de determinar su relación con el ambiente sensado. Hay una gran variedad de tecnologías de sensado disponibles: fotorreflectivos, ultrasónicos, rangos de sensado por infrarrojo y láser, táctiles; también han sido exploradas las visiones monoculares, binoculares y trioculares. La dificultad es interpretar esta información, es decir, decidir qué es lo que las señales sensadas nos dicen a cerca del mundo externo.
- Razonamiento: el robot debe ser capaz de decidir qué tipo de acciones se requieren para alcanzar sus metas en un ambiente dado. Esto puede involucrar un amplio rango de decisiones sobre qué ruta tomar o qué sensor usar.

Los robots industriales emplean un mínimo de sensado del ambiente y prácticamente no emplean razonamiento del ambiente porque operan esencialmente en un área limitada y en ambientes estáticos.

## 2.5 Visión inicial del sistema

El ser humano por mucho tiempo ha intentado construir herramientas y máquinas que le faciliten su convivencia con él mismo y con la naturaleza. Entre las máquinas que ha desarrollado se encuentran los robots móviles, los cuales hasta el día de hoy se han empleado en tareas muy específicas y, hasta cierto punto, sencillas. Aunque se ha trabajado en esta área, aún se está lejos de poder tener un robot con características similares al ser humano.

En el campo de los robots móviles, algunos de los problemas que se han intentado resolver son la evasión de obstáculos, el seguimiento de rutas, la capacidad de proponer otras rutas, entre otras más. Se han ocupado distintas herramientas para la resolución de estos problemas, una de las cuales es con la implementación de redes neuronales, técnica muy utilizada en el reconocimiento de patrones, pero que tiene la desventaja de utilizar muchos recursos de computo.

En el presente trabajo se propone construir un robot móvil para desarrollar y probar algoritmos que faciliten la planeación, seguimiento de rutas y evasión de obstáculos.

El robot móvil se diseñará de tal forma que tenga la facilidad de realizar sus desplazamientos en un plano con referencia de coordenadas polares, para que sea compatible con el programa ROC2<sup>2</sup> [Hernández y Velásquez, 2001]. Para lograr esto el robot empleará dos motores de corriente directa y un motor de pasos.

Además, el robot móvil contará con sensores táctiles para que le permitan detectar obstáculos y sensores fotorreflectivos que le permitan distinguir zonas blancas y negras con el fin de seguir rutas. Los sensores táctiles se colocarán en los costados del robot y los sensores fotorreflectivos en la parte inferior del mismo. También se colocarán interruptores ópticos en los motores de corriente directa, con la finalidad de tener una medición de la posición angular de dichos motores.

El robot móvil emplea un microcontrolador, el cual es el encargado de almacenar y ejecutar los programas, además de recolectar la información de los sensores táctiles, fotorreflectivos e interruptores ópticos. El microcontrolador es el encargado de calcular y controlar algunas de las tareas del robot móvil, mientras que otros procesos se realizarán en la PC base.

La comunicación entre la computadora y el robot móvil se lleva a cabo por medio de un módulo de comunicación inalámbrico, esto con la finalidad de que durante el seguimiento de rutas, el cable no sea un obstáculo en su recorrido, y que también el cable no sea un impedimento para desplazarse por lugares alejados de la computadora. Se podrán ejecutar tareas en línea a distancias mayores que las que se pueden alcanzar con un cable.

La herramienta que se emplea para la corrección de errores en el recorrido de ciertas rutas específicas del robot móvil, es una red neuronal de funciones de base radial, la cual se explicará con la sección 3.2.2. Primero el robot seguirá algunas rutas propuestas y con los

---

<sup>2</sup> Robot Command Center

datos obtenidos se calcularán los pesos de la red neuronal, una vez que se tengan los pesos o parámetros de la red, el robot recorrerá de nuevo alguna de las rutas y con la información que vaya recogiendo la red propondrá una nueva salida, en caso de que el robot tuviera alguna desviación de su ruta original se contempla que la red neuronal pueda corregir dicho error.

Debido a que el cálculo de los pesos o parámetros de la red neuronal y la ejecución de la red neuronal requiere de muchos recursos de computo, dichos cálculos se llevan a cabo en la PC base, por lo que la comunicación inalámbrica tendrá un papel muy importante.

En trabajos posteriores se tiene contemplada la construcción de más robots móviles, con el objetivo de formar un sistema mucho más complejo, en el cual se puedan desarrollar algoritmos de sistemas cooperativos, detección de fallas, entre otros.

## **2.6 Resumen**

En este capítulo se presentan una serie de temas que dan sustento básico para el desarrollo de la tesis. Por un lado se revisaron los sistemas de comunicación inalámbrica para establecer la interfaz de datos entre los elementos. Asimismo, los sistemas neuronales son revisados para dar una primera idea de cómo se presentan las distintas opciones de procesamiento de las diferentes rutas del sistema autónoma.

Por otro lado se presenta una revisión inicial de los sistemas móviles desde una perspectiva modular.

# Capítulo 3

## Diseño del sistema

### 3.1 Introducción

En este capítulo se presenta la implantación de los diferentes módulos que integran al sistema móvil, el tipo de red a usar, la comunicación entre el robot móvil y la computadora, la integración mecánica y las pruebas de la calibración inicial del sistema.

En la sección 3.2 se explica el comportamiento de la red de funciones de base radial, en la sección 3.3 se detalla el protocolo que emplea el módulo de comunicación; en la sección 3.4 se describe la integración mecánica y eléctrica del robot móvil, en la sección 3.5 se despliegan los resultados de las pruebas de la calibración del robot y las conclusiones se presentan en la sección 3.6.

### 3.2 Generalidades de la red neuronal

Las redes de funciones de base radial han sido aplicadas en diversos campos dentro del procesamiento de señales. Tales redes ofrecen una alternativa viable a redes neuronales de dos capas para muchas aplicaciones de procesamiento de señales y también para aquellas redes de parámetros altamente no lineales [Nelles, 2001].

El método de funciones de base radial tradicionalmente ha sido ocupado para interpolación en un espacio multidimensional [Powel, 1987]. El método original de funciones de base radial requiere que haya tantos centros de funciones de base radial como datos, lo cual es poco práctico en aplicaciones de procesamiento de señales, ya que el número de datos es generalmente muy grande.

Una red de funciones de base radial es considerada como una red especial de dos capas la cual es de parámetros lineales al momento de fijar todos los centros de las funciones de base radial y parámetros no lineales en la capa oculta. Por lo tanto, una capa oculta realiza una transformación no lineal con parámetros no ajustables y mapea el espacio de entrada sobre un nuevo espacio. La capa de salida realiza una combinación lineal sobre este nuevo espacio y los únicos parámetros ajustables son los pesos de esta combinación lineal. Cada nodo oculto en una red de funciones de base radial tiene una respuesta simétricamente lineales alrededor de un centro y la capa de salida de la red es un conjunto de combinaciones lineales con pesos. Los pesos de la capa de salida pueden ser obtenidos empleando el método de mínimos cuadrados [Chen, *et al.* 1991].

Un algoritmo común de aprendizaje para redes de funciones de base radial se basa en elegir aleatoriamente algunos puntos como centros de las funciones de base radial y, así usar la descomposición singular de un valor para encontrar los pesos de la red. La elección arbitraria de algunos puntos como centros, puede no satisfacer siempre el requerimiento de que los centros pertenezcan al conjunto de los datos de entrada de la red. Además, tal aproximación podría requerir una red de funciones de base radial excesivamente grande para lograr un nivel de rendimiento dado, y esto puede causar problemas de acondicionamiento numérico, por ejemplo, generar matrices singulares. Estos problemas se pueden resolver formulando un modelo de selección de centros que pertenezcan al conjunto de los datos. Para las redes de funciones de base radial con salida escalar, se puede derivar un algoritmo de aprendizaje inteligente basado en el método de mínimos cuadrados ortogonales, el cual construye redes de base radial de una forma racional.

El objetivo de una transformación no lineal en la capa oculta de una red de funciones de base radial, es evitar problemas multicolinealidad en los datos de entrada de la red. Dicha función no lineal generalmente es elegida de unas cuantas funciones típicas no lineales. Un consenso general es que la elección de la no linealidad no es crucial para el desempeño. Sin embargo, el rendimiento de una red de funciones de base radial depende críticamente de los centros elegidos. En la práctica, los centros son un subconjunto de datos. Los centros elegidos deben muestrear apropiadamente el dominio de la entrada [Chen, *et al*, 1991].

Un método adecuado para seleccionar los centros de la red de funciones de base radial es fuzzy c-means.

### 3.2.1 Fuzzy c-means

En este trabajo, el algoritmo que se utilizó para la obtención de los centros de la red de funciones de base radial es fuzzy c-mens.

El resultado del algoritmo de fuzzy c-means reconoce nubes esféricas de puntos en un espacio  $l$ -dimensional. Se asume que los grupos son del mismo tamaño. Cada grupo está representado por su centro. Esta representación de un grupo también se conoce como prototipo, ya que se considera como un dato representativo de todos los datos asignados a un grupo. Para la implementación de esta técnica el número óptimo de centros del grupo debe estar dado. El número de centros óptimo debe ser tal que todos los datos puedan ser agrupados en cada uno de los grupos con un nivel de pertenencia dado [Höppner, *et al*. 1992].

La letra  $c$  en el nombre de este algoritmo fija el número de grupos, por ejemplo, con cuatro grupos, es "fuzzy 4-means".

El algoritmo se aplica cuando el número de los grupos es fijo y está dado, es decir, este método no determina el número de centros de las agrupaciones.

### 3.2.2 Red de funciones de base radial

En la Fig. 3.1 se muestra un esquema de la red de funciones de base radial con  $n$  entradas y una salida de tipo escalar [Chen, *et al*, 1991]. Esta red establece un mapeo  $f_r: R^n \rightarrow R$  de acuerdo a

$$f_r(x) = w_0 + \sum_{i=1}^m w_i \phi(\cdot) \tag{3.1}$$

donde  $x \in R^n$  es el vector de entrada,  $\phi(\cdot)$  es una función no lineal dada, de  $R$  a  $R$ ,  $w_i$  ( $0 \leq i \leq m$ ) son conocidos como los pesos o parámetros de la funciones de base radial y  $m$  es el número de centros. Se considera el caso de la salida escalar por simplicidad en la notación, sin embargo, se extiende el mismo procedimiento para el caso de múltiples salidas. De hecho, una red de funciones de base radial con salidas múltiples se puede separar en grupos de redes de funciones de base radial de una sola salida. En la red de funciones de base radial se asume que la forma funcional  $\phi(\cdot)$  y los centros  $c_i$  son fijos. Proporcionando un conjunto de entradas  $x(t)$  y sus correspondientes salidas deseadas  $d(t)$  para  $t = 1$  hasta  $N$ , los valores de los pesos  $w_i$  pueden ser determinados usando el método lineal de mínimos cuadrados. Sin embargo, las elecciones de  $\phi(\cdot)$  y  $c_i$  deben ser consideradas cuidadosamente para que la red de funciones de base radial sea capaz de aproximarse al funcionamiento de una red neuronal de dos capas.

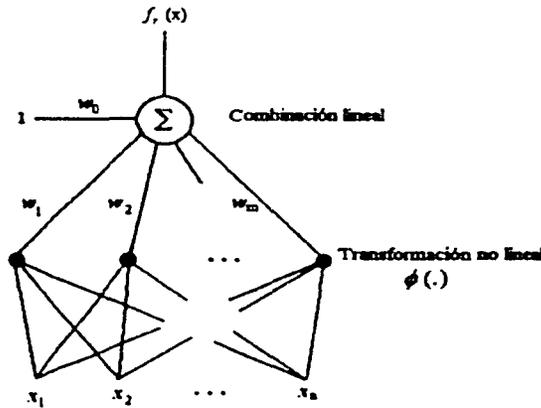


Fig. 3.1 Esquema de una red de funciones de base radial.

Los resultados de investigaciones teóricas y prácticas sugieren que la elección de la no linealidad de  $\phi(\cdot)$  no es crucial para el funcionamiento de la red [Chen, *et al*, 1991]. Por ejemplo, una función no lineal típicamente empleada es la función gaussiana

$$\phi(x) = \exp\left(-\frac{1}{2} * \left(\frac{x-c}{\sigma}\right)^2\right) \quad (3.2)$$

donde  $c$  son los centros de la función gaussiana y  $\sigma$  es una constante real que define la desviación estándar de los datos con respecto al centro de la función gaussiana. Para la ecuación (3.2),  $\phi(x) \rightarrow 0$  cuando  $x \rightarrow \infty$ . Se considera que los centros de la gaussiana están fijos y no cambian.

En la práctica, los centros son elegidos de los datos  $\{x(t)\}_{t=1}^N$ . Un método comúnmente usado para elegir los centros es tomarlos arbitrariamente. Sin embargo, este método no garantiza el funcionamiento adecuado porque no satisface el requerimiento de que los centros representen apropiadamente el dominio de la entrada. Además, para obtener un funcionamiento dado, puede requerirse una red de funciones de base radial innecesariamente grande, lo cual provoca una complejidad computacional, y por lo tanto, un mal acondicionamiento numérico.

Para entender el funcionamiento de mínimos cuadrados ortogonales, es esencial ver a la red de funciones de base radial, ecuación (3.1) como un caso especial del modelo de la regresión lineal:

$$d(t) = \sum_{i=1}^M p_i(t)\theta_i + \varepsilon(t) \quad (3.3)$$

donde  $d(t)$  es la salida buscada,  $\theta$  es la matriz de parámetros y  $p_i(t)$  es conocido como el vector de regresión, el cual es función fija no lineal de  $x(t)$ :

$$p_i(t) = p_i(x(t)) \quad (3.4)$$

Se asume que la señal de error  $\varepsilon(t)$  no está correlacionada con los regresores  $p_i(t)$ . Se puede incluir un término constante en la ecuación (3.3) fijando  $p_i(t) = 1$ . Es aparente que un centro fijo  $c_i$  con una no linealidad  $\phi(\cdot)$  dada, corresponde a un regresor  $p_i(t)$  en la ecuación (3.3). Un procedimiento eficiente de aprendizaje para seleccionar un subconjunto de la ecuación (3.3) se basa en el método de mínimos cuadrados ortogonales [Chen, *et al.*, 1991]. Se considera que los centros la función no lineal,  $\phi(\cdot)$ , están fijos y no cambian.

### 3.2.3 Algoritmo de aprendizaje de mínimos cuadrados ortogonales

La interpretación geométrica del método de mínimos cuadrados se entiende mejor reordenando la ecuación (3.3) para  $t = 1$  hasta  $N$  en forma matricial:

$$d = P\theta + E \quad (3.5)$$

donde

$$d = [d(1) \cdots d(N)]^T \quad (3.6)$$

$$P = [p_1 \cdots p_M], \quad p_i = [p_i(1) \cdots p_i(N)]^T, \quad 1 \leq i \leq M \quad (3.7)$$

$$\theta = [\theta_1 \cdots \theta_M]^T \quad (3.8)$$

$$E = [\varepsilon(1) \cdots \varepsilon(N)]^T \quad (3.9)$$

Los vectores de regresión  $p_i$  de un conjunto de vectores básicos y la solución de mínimos cuadrados  $\hat{\theta}$  satisfacen la condición de que  $P\hat{\theta}$  sea la proyección de  $d$ .

El método de mínimos cuadrados ortogonales involucra la transformación del conjunto de  $p_i$  en el conjunto de vectores básicos ortogonales, y esto hace posible calcular la contribución individual en la energía de salida deseada de cada vector base. La matriz de regresión  $P$  puede ser descompuesta en

$$P = WA \quad (3.10)$$

donde  $A$  es una matriz triangular de  $M \times M$  con unos en la diagonal principal y ceros debajo de la diagonal principal, esto es,

$$A = \begin{bmatrix} 1 & \alpha_{12} & \alpha_{13} & \cdots & \alpha_{1M} \\ 0 & 1 & \alpha_{23} & \cdots & \alpha_{2M} \\ \vdots & 0 & \ddots & \ddots & \vdots \\ & \vdots & \ddots & 1 & \alpha_{M-1M} \\ 0 & 0 & & 0 & 1 \end{bmatrix} \quad (3.11)$$

y  $W$  es una matriz de  $N \times M$  con columnas ortogonales  $w_i$  tal que,

$$W^T W = H \quad (3.12)$$

donde  $H$  es una matriz diagonal con elementos  $h_i$ :

$$h_i = w_i^T w_i = \sum_{t=1}^M w_i(t) w_i(t), \quad 1 \leq i \leq M \quad (3.13)$$

El espacio expandido por el conjunto de vectores básicos ortogonales  $w_i$  es el mismo espacio expandido por el conjunto de  $p_i$ , y la ecuación (3.5) puede ser reescrita como

$$d = Wg + E \quad (3.14)$$

La solución de mínimos cuadrados ortogonales  $\hat{g}$ , está dada por

$$\hat{g} = H^{-1}W^T d \tag{3.15}$$

o

$$\hat{g}_i = w_i^T d / (w_i^T w_i), \quad 1 \leq i \leq M \tag{3.16}$$

Las cantidades  $\hat{g}$  y  $\hat{\theta}$  satisfacen el sistema triangular

$$A\hat{\theta} = \hat{g} \tag{3.17}$$

Los métodos de Gram-Schmidt clásico y Gram-Schmidt modificado pueden ser usados para derivar la ecuación (3.17) y así resolver la  $\hat{\theta}$  estimada [Chen, *et al.*, 1989]. El método Gram-Schmidt calcula una columna de A al mismo tiempo que ortogonaliza P de la siguiente forma: en el  $k$ -ésimo paso se tiene que la  $k$ -ésima columna es ortogonal a cada una de las  $k-1$  columnas ortogonalizadas previamente y se repite la operación para  $k = 2, \dots, M$ . El procedimiento de cálculo puede ser representado como

$$\left. \begin{aligned} w_1 &= p_1 \\ \alpha_{ik} &= w_i^T p_k / (w_i^T w_i), \quad 1 \leq i < k \\ w_k &= p_k - \sum_{i=1}^{k-1} \alpha_{ik} w_i \end{aligned} \right\} \quad k = 2, \dots, M \tag{3.18}$$

El método de mínimos cuadrados ortogonales tiene propiedades numéricas superiores comparadas con el método de mínimos cuadrados ordinario.

Para determinar los centros de las funciones gaussianas se utilizó el método de "fuzzy c-means" y las desviaciones estándar se obtienen de la siguiente forma

$$\sigma_{kj} = \sqrt{\frac{\sum_{i=1}^N (x_{ij} - c_{kj})^2}{N - 1}} \tag{3.21}$$

donde  $\sigma_{kj}$  ( $1 \leq k \leq M$ ,  $1 \leq j \leq$  número de entradas) son las desviaciones estándar,  $x_{ij}$  ( $1 \leq i \leq N$ ,  $1 \leq j \leq$  número de entradas) son los datos de entrada,  $c_{kj}$  son los centros de las funciones gaussianas y  $N$  es el número total de datos en el tiempo.

### 3.3 Comunicación inalámbrica

Uno de los elementos más importantes del sistema es la comunicación entre el robot móvil y la computadora. Dado que el robot no cuenta con suficiente memoria para realizar los cálculos para el entrenamiento de la red neuronal, éste debe enviar la información a la

computadora para que ésta realice dicho entrenamiento y después comunique la información ya procesada al robot móvil. Debido a que el robot móvil debe desplazarse de manera libre en cualquier dirección y en un entorno en el cual puede encontrarse con obstáculos, se optó por la comunicación inalámbrica.

Para poder cumplir con el objetivo de tener una comunicación inalámbrica entre el robot móvil y la computadora, se utilizaron los módulos de comunicación del Kit de Desarrollo DR1007-DK Virtual Wire®<sup>3</sup> [Manual RFM, 1996]. Estos módulos permiten establecer comunicaciones de datos de baja potencia basadas en una transmisión de paquetes de la forma *half-duplex*. Otra característica que hace factible el uso de estos módulos es que únicamente se necesita contar con puertos seriales RS-232 para lograr la comunicación. Entre otras características, estos módulos tienen direcciones programables mediante puentes (hasta 15 direcciones), lo que hace posible tener más robots móviles o más computadoras en el sistema.

En cuanto a la computadora, no hubo ningún problema al conectar el módulo de comunicación y lo único necesario fue ajustar la tasa de transmisión a 19.2 kbps, con ocho bits de datos (byte), un bit de parada y sin bit de paridad dado que el módulo de comunicación trabaja únicamente bajo estas especificaciones. Debido a que el robot móvil cuenta con un microcontrolador MC68HC11F1 de Motorola [Manual Motorola, 1995], el cual posee un puerto serie con protocolo RS-232, no existe ningún problema para conectar el módulo de comunicación; sin embargo, para lograr que el microcontrolador pueda transmitir a una tasa de transmisión de 19.2 kbps (kilobits por segundo) es necesario hacer un escalamiento en frecuencia. El microcontrolador MC68HC11F1 de Motorola cuenta con la característica de poder generar diferentes tasas de transmisión, las cuales se determinan dividiendo entre ciertos factores la frecuencia del cristal del microcontrolador. El sistema mínimo para el microcontrolador MC68HC11F1 de Motorola<sup>4</sup> originalmente contaba con un cristal a una frecuencia de 8 MHz, pero a esta frecuencia no se puede obtener una tasa de transmisión de 19.2 kbps (esto debido a los factores que tiene el microcontrolador para hacer la división de frecuencia), así que se cambió el cristal de 8 MHz por uno de 4.9152 MHz, con el cual se pueden obtener tasas de 19.2 y 9.6 kbps, entre otras.

Una vez resuelto el problema de la tasa de transmisión, se colocó el módulo de comunicación en la parte más alta del robot móvil, con el objeto de tener una mejor línea de vista. Además, se construyeron dos cajas de metal en las cuales se colocaron los módulos de comunicación, esto con el fin de evitar que el ruido producido por los motores del robot móvil, u otras fuentes de ruido, no interfieran en el proceso de comunicación.

### 3.3.1 Protocolo de comunicación

Los módulos de comunicación están formados por dos tarjetas, una de radio y otra de protocolo. En la tarjeta de radio se transmite y se recibe la información, que proviene o es enviada según sea el caso a la tarjeta de protocolo. La tarjeta de protocolo cuenta con las

<sup>3</sup> En el apéndice A se encuentra una descripción detallada sobre este kit de desarrollo.

<sup>4</sup> En el apéndice B se detallan las características de este sistema mínimo.

siguientes características: detección de errores, quince posibles nodos de direcciones (con la posibilidad de enviar el mismo mensaje a todos los nodos o a uno solo), soporte de mensajes en código binario o ASCII de hasta 32 bits por paquete, una retransmisión automática del paquete recibido y una alarma de falla en caso de no completarse la comunicación [Manual RFM, 1996].

La dirección del nodo de cada tarjeta de protocolo se fija mediante puentes que son colocados en los postes que se encuentran en cada una de las tarjetas. Cada tarjeta de protocolo cuenta con un microcontrolador AT89C2051, en el cual se implementa el protocolo.

El protocolo es capaz de transmitir y recibir bytes de datos binarios de cualquier patrón de bits. Los mensajes son enviados y recibidos desde la interfaz RS-232 en un formato estándar asíncrono. El formato general con el que debe contar cada mensaje para ser enviado consta de cuatro campos, como se puede observar en la tabla 3.1. Además del formato general del mensaje, el protocolo cuenta con mensajes especiales, los cuales se presentan en el apéndice A.

Campo de dirección (1 byte)	Campo de número de mensaje (1 byte)	Campo de longitud del mensaje (1 byte)	Campo del mensaje (hasta 32 bytes)
--------------------------------	---	--	---------------------------------------

**Tabla 3.1 Formato general del mensaje.**

En el campo de dirección, se tiene el primer byte del mensaje, el cual debe contener la dirección "hacia"/"de". Los cuatro bits más significativos indican el número del nodo al cual se envía el mensaje y los cuatro bits menos significativos determinan el número del nodo que está enviando el mensaje. El segundo campo se forma por el segundo byte el cual contiene el número de mensaje (de 1 a 7 reciclandose y el 8 usado para paquetes de telemetría). El tercer campo que también consta de un byte es el número de bytes en el mensaje (hasta 32 bytes de datos) y el último campo lo constituye el mensaje, el cual debe tener un máximo de 30 bytes. Dentro del campo mensaje se debe incluir un byte de inicio de mensaje (el número 02<sub>h</sub>) y un byte de fin de mensaje (el número 03<sub>h</sub>).

Cuando se tiene un mensaje de más de 30 bytes éste se debe dividir para su transmisión. En un primer paquete se coloca el byte de inicio y los 31 primeros bytes del mensaje, en un segundo paquete se transmiten los siguientes 32 bytes, ya sin colocar el byte de inicio y así sucesivamente hasta que es enviado todo el mensajes y se coloca el byte de fin de mensaje en el último paquete.

Longitud del mensaje	Formato del mensaje
Mensaje ≤ 30 bytes	Dirección     Número     Longitud     02h mensaje 03h   (1 byte) (1 byte) (1 byte) (hasta 30 bytes)
Mensaje > 30 bytes	Dirección     Número     Longitud     02h mensaje   (1 byte) (1 byte) (1 byte) (31 bytes)
	Dirección     Número     Longitud     mensaje   (1 byte) (1 byte) (1 byte) (32 bytes)
	Dirección     Número     Longitud     mensaje 03h   (1 byte) (1 byte) (1 byte) (hasta 31 bytes)

**Tabla 3.2 Construcción del mensaje.**

### 3.3.2 Detección del número de nodo de comunicación

Al iniciar la comunicación, lo primero que se debe hacer es verificar qué nodo tiene el módulo de comunicación. Para conocer el número del nodo se hace uso del siguiente mensaje especial:

| 00h | | 00h | | 01h | | 31h |

Si el módulo de comunicación tuviera programado el nodo número dos, entonces éste enviaría a la computadora (o al microcontrolador, según sea el caso) el siguiente mensaje:

| 00h | | 22h | | 00h | | 01h | | 35h |

El mensaje que envía el módulo de comunicación se compone de cinco bytes, y es en el segundo byte donde se encuentra el número del nodo. En éste segundo byte se tiene el número del nodo en los cuatro bits más significativos o también se encuentra repetido en los cuatro bits menos significativos.

Una vez que se conoce el número de nodo del módulo de comunicación, esta información se incorpora al formato general del mensaje.

### 3.3.3 Transmisión de mensajes

Debido a que la transmisión de información es de la forma *half-duplex*, es necesario antes de transmitir un mensaje corroborar que el canal de comunicación se encuentra libre (es decir, que en ese momento no se encuentra el módulo de comunicación recibiendo un mensaje). Para esto, primero se envía el byte de dirección al módulo de comunicación. Cabe recordar que el byte de dirección se forma colocando en los cuatro bytes más significativos el número del nodo al cual se quiere enviar la información y en los cuatro bytes menos significativos el número de nodo que tiene el módulo de comunicación que va a enviar la información.

Si el byte de dirección es retransmitido por el módulo de comunicación hacia la computadora (o el microcontrolador) dentro de un lapso de 50 ms, entonces la computadora tiene apartado el canal de comunicación por los siguientes 200 ms (es decir, que la computadora cuenta con 200 ms para poder enviar todo el mensaje).

Una vez que se tiene confirmado el canal de transmisión, se envía el segundo byte, el cual contiene el número de paquete. Recordando que éste debe ser del 1 al 7 y repitiéndose el 1 después del 7 y así sucesivamente. Por lo tanto, si son enviados dos paquetes con el mismo número consecutivamente, el segundo no será transmitido por el módulo de comunicación. Es importante recalcar que una vez que se tiene confirmado el canal de comunicación ya no es necesario enviar el byte de dirección, ya que si se es enviado una vez más con todo el mensaje, éste no será transmitido por el módulo de comunicación.

El tercer byte que se envía es el de la longitud del mensaje, en el cual se debe incluir el byte de inicio de mensaje (el número 02<sub>h</sub>) y el byte de fin de mensaje (el número 03<sub>h</sub>). La longitud del mensaje debe ser de hasta 30 bytes. Y por último se envía el mensaje entre los bytes de inicio de mensaje (02<sub>h</sub>) y de fin de mensaje (03<sub>h</sub>).

El protocolo indica que dentro del campo del mensaje se tiene que colocar un byte de inicio de mensaje (02<sub>h</sub>) y un byte de fin de mensaje (03<sub>h</sub>), dichos bytes se pueden omitir y aún así se puede lograr un proceso de comunicación exitoso.

Una vez terminada la transmisión, la computadora recibe un mensaje de confirmación por parte del módulo de comunicación. Dicho mensaje de confirmación se forma por tres bytes. Estos tres bytes tienen el objetivo de informar a la computadora (o al microcontrolador) si el proceso de comunicación se completó o no. Cuando un módulo de comunicación recibe información, éste envía un mensaje especial al módulo que transmite la información con el fin de informarle que recibió la información. Este proceso se lleva a cabo por el protocolo de los módulos de comunicación, y es totalmente transparente a la computadora (o al microcontrolador).

Cuando un mensaje es transmitido con éxito, el mensaje de confirmación que recibe la computadora (o el microcontrolador) es el siguiente:

| Dirección | | Número | | E1<sub>h</sub> |

El primer byte es el byte de dirección, con la diferencia de que ahora los cuatro bits más significativos indican el número de nodo del módulo que transmite la información y los cuatro bits menos significativos indican el número de nodo del módulo al cual se envió la información (“de”/“hacia”). El segundo byte es el número de paquete que se envió (del 1 al 7). Y el tercer byte es el dato  $E1_h$ , el cual indica que el mensaje fue recibido con éxito por el módulo de comunicación al cual fue enviada la información.

Si no se están modificando adecuadamente los valores del número de paquete del mensaje que se envía, el módulo de comunicación no transmite la información, pero sí envía a la computadora (o al microcontrolador) un mensaje de confirmación de éxito. En otras palabras, si no se está cambiando el número de paquete la información no se transmite, pero la computadora (o el microcontrolador) pueden interpretar lo contrario, dado que el módulo de comunicación le envía un mensaje de confirmación de éxito.

En el caso en el cual el mensaje enviado no es recibido con éxito por parte del módulo de comunicación receptor, el mensaje de confirmación es el siguiente:

| Dirección | | Número | |  $DD_h$  |

El primer byte es de dirección, donde los cuatro bits más significativos indican el número de nodo del módulo de comunicación al cual se envía la información y los cuatro menos significativos el número de nodo del módulo de comunicación que transmite la información (“hacia”/“de”). El segundo byte es el número de paquete que se envió. El tercer byte es el dato  $DD_h$ , el cual indica que la transmisión no se pudo completar.

### 3.3.4 Recepción de mensajes

El módulo de comunicación permanece en espera de un mensaje todo el tiempo. Cuando un mensaje es recibido y validado por el protocolo del módulo de comunicación, es enviado a la computadora (o al microcontrolador) con el siguiente formato:

| Dirección | | Número | | Longitud | |  $02_h$  mensaje  $03_h$  |

El primer byte que se recibe es el de la dirección. Los cuatro bits más significativos corresponden al número de nodo del módulo de comunicación al cual se está enviando la información, y los cuatro menos significativos indican el número de nodo del módulo que está enviando la información (“hacia”/“de”).

El segundo byte es el número de paquete enviado, el cual sólo puede tomar valores del 1 al 7. El número de paquete es importante cuando se recibe un mensaje de más de 30 bytes, o también cuando se reciben paquetes cortos de forma consecutiva. Normalmente se espera que los paquetes recibidos tengan un número consecutivo, pero esto no es necesario, se pueden recibir paquetes con números saltados sin ningún problema. Cabe recordar que se pueden mandar mensajes a diferentes nodos, por lo tanto, un nodo en especial no necesariamente tiene que estar recibiendo paquetes con números consecutivos.

El tercer byte corresponde a la longitud del mensaje, que como se mencionó anteriormente, debe ser de hasta 30 bytes.

Los siguientes bytes forman el mensaje. Dentro de esta cadena el primer byte corresponde al byte de inicio de mensaje (02<sub>h</sub>) y el último al byte de fin de mensaje (03<sub>h</sub>). Entre estos dos bytes se encuentra la información.

### 3.4 Integración mecánica y eléctrica

En esta sección se explica la forma en la cual se construyó el robot móvil. El robot se divide en seis secciones: la primera corresponde a la comunicación inalámbrica, la segunda sección hace referencia al microcontrolador, la tercera explica la parte motriz, la cuarta el sensado, la quinta la electrónica y por último el armazón.

El material que se usó para la construcción del robot es acrílico, debido a que se puede deformar por medio de calor o se puede cortar y unir con la ayuda de cloruro de metileno, además de ser un material resistente y ligero en comparación con otros materiales como lámina metálica o madera.

El robot cuenta con tres motores: dos para el desplazamiento y giro (motores de corriente directa con reductores de velocidad) y uno para la dirección (motor de pasos). Los motores de corriente directa se encuentran ubicados en la parte trasera y el motor de pasos se encuentra ubicado en la parte delantera del robot. Esta ubicación de los motores permite al robot girar sobre su propio eje, dicha característica facilita el desplazamiento del robot dentro de un plano de coordenadas polares.

El robot tiene cuatro sensores fotorreflectivos, dos interruptores ópticos y cuatro sensores táctiles. Los cuatro sensores fotorreflectivos están ubicados en la parte inferior del robot, uno en cada esquina, formando un cuadrilátero, esto con el fin de detectar zonas blancas y negras en la superficie en la cual se desplaza. Los dos interruptores ópticos se encuentran en cada uno de los motores de corriente directa encargados del desplazamiento y giro. La finalidad de estos interruptores es determinar una señal que proporciona información acerca del desplazamiento de dichos motores. Los sensores táctiles están distribuidos uno a cada lado del robot móvil, y se emplean para detectar obstáculos en su recorrido. Además, el robot cuenta con dos detectores de nivel bajo de voltaje de las baterías para su reemplazo.

El robot móvil se compone de tres "pisos", como se muestra en la Fig. 3.2. En el primero se guardan las baterías y los motores de corriente directa; este piso se encuentra cerrado y cuenta con dos puertas laterales que permiten el reemplazo de baterías. En el segundo piso se encuentra colocada la tarjeta con los circuitos correspondientes a la etapa de alimentación, potencia de los motores, detector de nivel bajo de voltaje y acondicionamiento de las señales sensadas. En el tercer piso o parte superior del robot está colocado el sistema mínimo para el microcontrolador MC68HC11F1 de Motorola<sup>5</sup> y el

---

<sup>5</sup> Motorola, [www.mot.com](http://www.mot.com)

módulo de comunicación Virtual Wire de RFM<sup>6</sup>. Por último, en la parte frontal del robot se encuentra un motor de pasos.

La sección de la comunicación la realiza un módulo de comunicación inalámbrica, el cual se explica en el apéndice A. El sistema mínimo para el microcontrolador MC68HC11F1 de Motorola es explicado en el apéndice B.

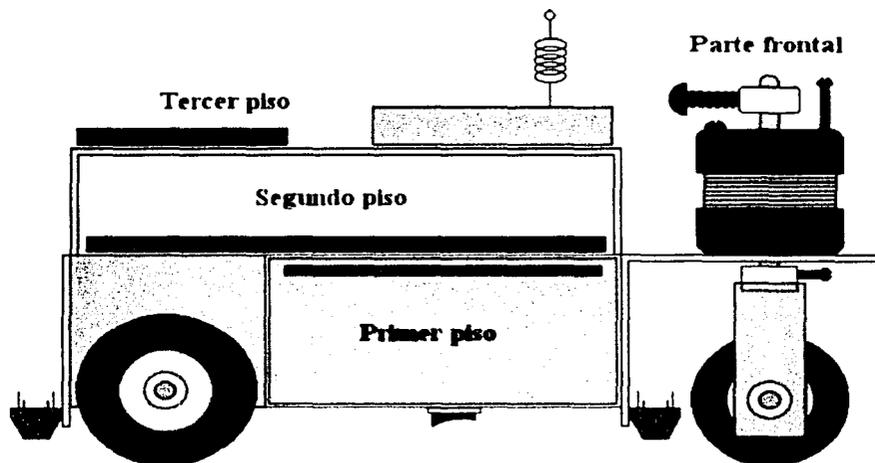


Fig. 3.2 Diagrama del robot móvil.

### 3.4.1 Sección motriz

La parte motriz está constituida por dos secciones, la primera es la que se encarga del desplazamiento y giro, y la segunda de la dirección. Para la primera parte, se utilizaron motores de corriente directa con reductores de velocidad para aumentar el par de los mismos. La ventaja de usar motores de corriente directa es que únicamente se requieren dos líneas de control para manejarlos, una para la habilitación y la otra para el sentido; además, tienen un bajo peso. En la parte de la dirección se empleó un motor de pasos porque se tiene un mayor control y precisión en su movimiento; la desventaja es que se requieren cuatro líneas para controlar su movimiento (debido a que se emplea el método de accionamiento en medio paso<sup>7</sup>).

Los encargados de la tracción del robot son dos motores de corriente directa que están colocados en la parte trasera del robot, en una misma dirección (en el mismo eje), tal como se observa en la Fig. 3.3. Dicha ubicación permite al robot girar sobre su propio eje, además de avanzar o retroceder.

<sup>6</sup> RF Monolithics, [www.rfm.com](http://www.rfm.com)

<sup>7</sup> Este accionamiento se explica en la sección C.2.2.

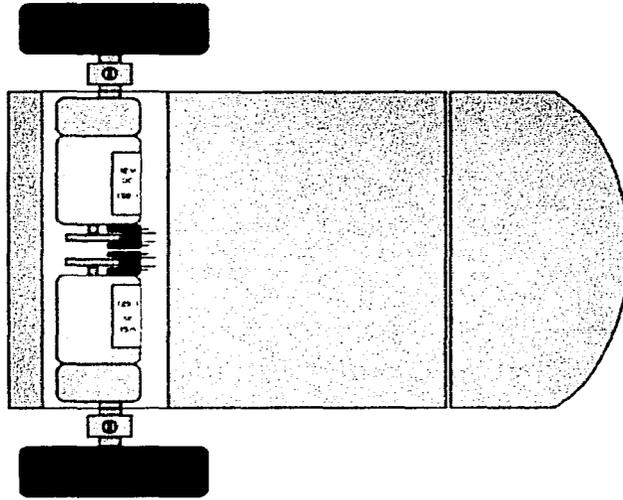


Fig. 3.3 Ubicación de los motores de corriente directa.

Cuando los motores de corriente directa se hacen girar en sentido contrario y la llanta delantera se encuentra paralela a las llantas traseras, el robot puede avanzar hacia adelante o hacia atrás, dependiendo del sentido del giro de los motores: si, por el contrario, los motores de corriente directa se hacen girar en el mismo sentido y la llanta delantera se encuentra perpendicular a las llantas traseras, el robot puede girar hacia su derecha o hacia su izquierda, lo cual depende del sentido de giro de los motores. En la Fig. 3.4 se muestran diagramas que ilustran las configuraciones de movimiento del robot.

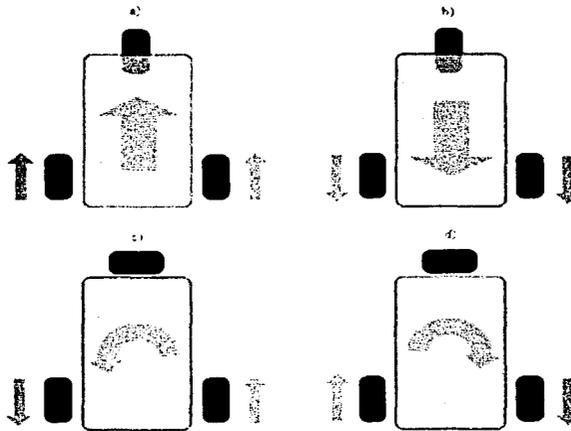
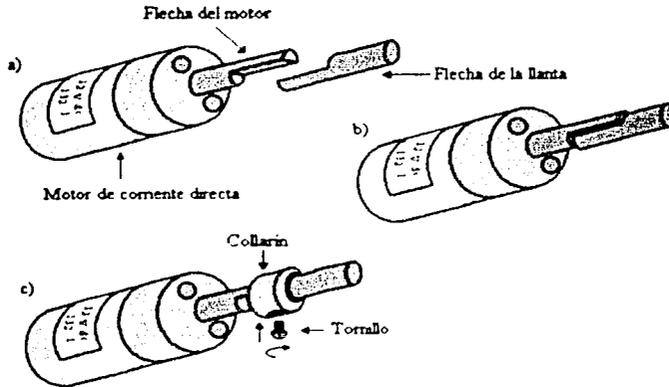


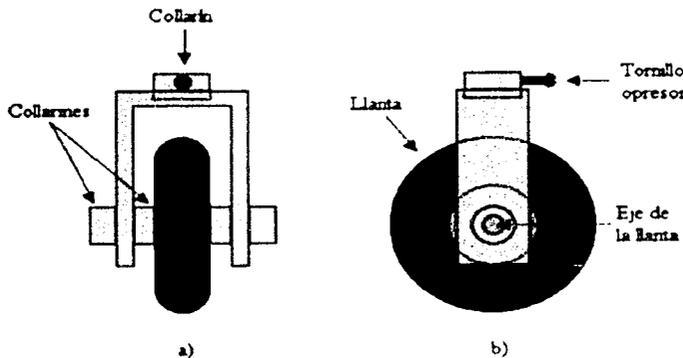
Fig. 3.4 Movimientos permitidos para el robot:  
a) adelante, b) atrás, c) giro hacia la izquierda, d) giro hacia la derecha.

Los motores de corriente directa están sujetos a una base de acrílico por medio de tornillos. Las flechas de los motores presentan un corte longitudinal, lo cual se aprovecha para colocar el eje de las llantas, haciéndoles a dichos ejes un corte similar y en estas uniones se sujetan ambas flechas por medio de collarines. En estas flechas se encuentran montadas las llantas. Ver Fig. 3.5.



**Fig. 3.5 Unión de la flecha del motor de corriente directa a la flecha de la llanta, mediante un collarin.**

La dirección está controlada por un motor de pasos, el cual se encuentra en la parte frontal del robot. La flecha del motor de pasos está colocada en una base de acrílico, la cual tiene arriba un collarín que sirve para sujetar dicha base a la flecha del motor. En la base de acrílico se encuentra un eje transversal a la flecha del motor de pasos, en el cual se monta una llanta sujeta por collarines. Dicha llanta tiene un movimiento libre en su eje. Ver Fig. 3.6.



**Fig. 3.6 Llanta delantera: a) vista frontal, b) vista lateral.**

Para establecer un punto de referencia en cuanto a la posición de la llanta delantera, se colocó un collarín sobre la flecha en la parte superior del motor de pasos. Este collarín en lugar de contar con un opresor para sujetar la flecha, tiene un tornillo, el cual sobresale aproximadamente un centímetro a partir de la superficie del collarín. En la parte superior del motor de pasos se colocó un tornillo (denominado tope) el cual es perpendicular al plano que describe el tornillo del collarín en su movimiento. El tope obstaculiza el movimiento del motor de pasos mediante el tornillo que sujeta el collarín dado que son perpendiculares. En ese punto se define la referencia de la dirección. Conociendo el punto de referencia de la dirección se alinea la llanta delantera, ver Fig. 3.7.

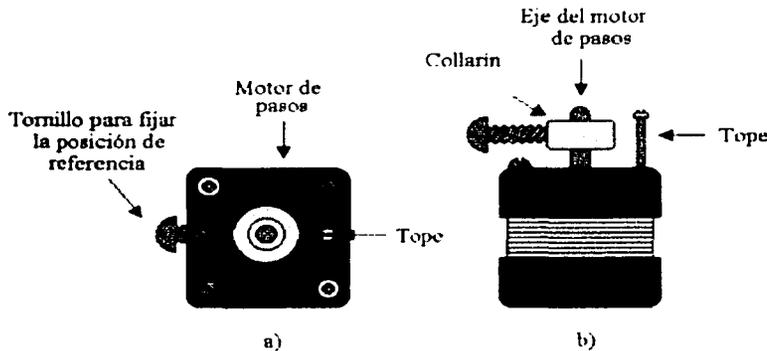


Fig. 3.7 Tope del motor de pasos: a) vista superior, b) vista frontal.

### 3.4.2 Sección de sensado

Los motores de corriente directa poseen un reductor de velocidad con una relación 132:1; así, por cada 132 revoluciones en la flecha del motor de corriente directa se tiene una revolución en la flecha del reductor. Para obtener información del movimiento de los motores de corriente directa, se colocó una hélice sobre cada una de las flechas de dichos motores, tal como se muestra en la Fig. 3.8.

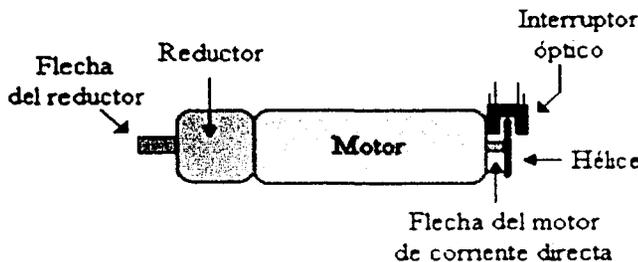


Fig. 3.8 Diagrama de un interruptor óptico colocado sobre el motor de corriente directa.

Para sensar la información descrita por la hélice, se colocó un interruptor óptico sobre la base del motor; de esta forma cada vez que la hélice cruza por el interruptor óptico, éste proporciona una señal correspondiente al movimiento del motor; por lo tanto, se generan 264 señales por una revolución en la flecha del reductor, dado que por cada revolución del motor de corriente directa, la hélice cruza dos veces el interruptor óptico.

Los sensores fotorreflektivos están colocados en la parte inferior del robot, uno en cada extremo del mismo. Éstos se encuentran colocados a 5 mm de la superficie aproximadamente. Cada señal de los sensores fotorreflektivos es comparada con un voltaje de referencia, lo que hace posible modificar el umbral de detección de los sensores fotorreflektivos con solo mover el voltaje de referencia. Esto es importante porque dependiendo de la intensidad de luz en el entorno, puede ser necesario modificar el umbral de detección. En la Fig. 3.9 se muestra la ubicación de los sensores fotorreflektivos.

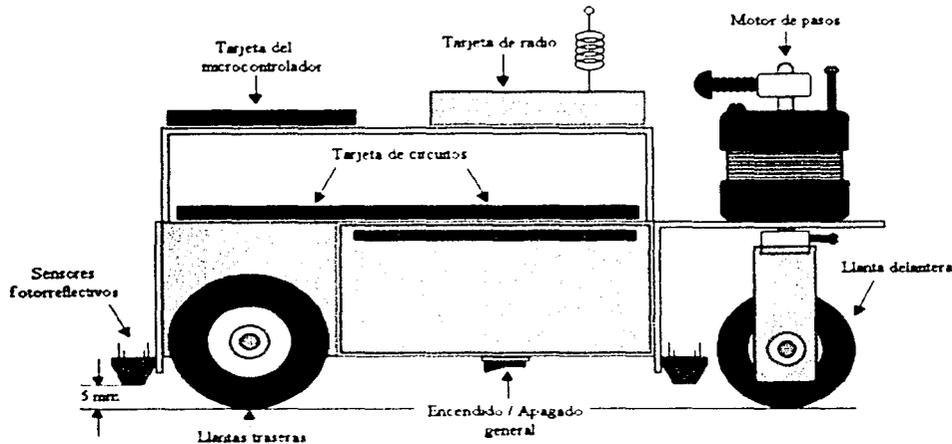


Fig. 3.9 Sensores fotorreflektivos colocados en el robot.

En cada uno de los cuatro lados del robot móvil se encuentran colocadas placas de acrílico, las cuales están sujetas al armazón del robot mediante tornillos, tal como se muestra en la Fig. 3.10. Las formas de dichas placas se explican con detalle en la sección del armazón.

Las placas de acrílico y el armazón se hallan separados por resortes ubicados alrededor de los tornillos. Esto permite que al aplicarse una fuerza perpendicular en alguna de las placas, ésta presione los resortes y, por consiguiente, éstos a su vez se contraigan, permitiendo que las placas de acrílico activen los microconmutadores, los cuales se encuentran ubicados entre las placas y el armazón. Al retirarse la fuerza, los resortes empujan la placa en sentido contrario y la regresan a su posición inicial, lo cual tiene como consecuencia la desactivación de los microconmutadores. Cabe mencionar que tanto en las placas laterales como en la frontal únicamente fue necesario colocar un microconmutador, no así en la placa trasera, en la cual fue necesario colocar dos microconmutadores (esto debido a la forma de las placas). Dichos microconmutadores se encuentran conectados en paralelo, así que en el momento de activarse uno, es suficiente para detectarse la señal.

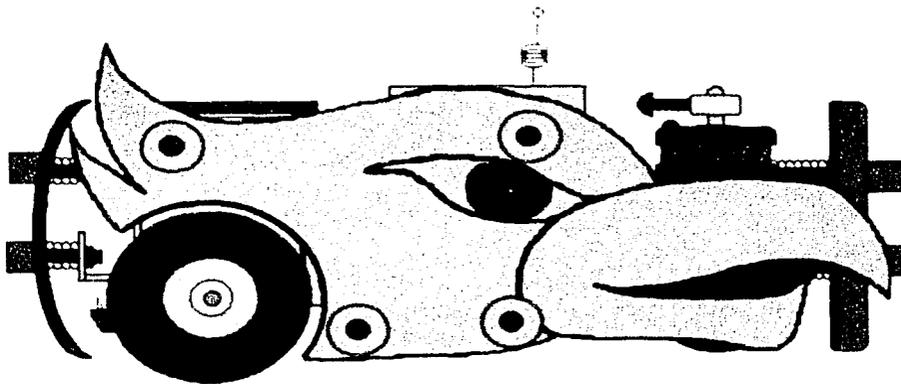
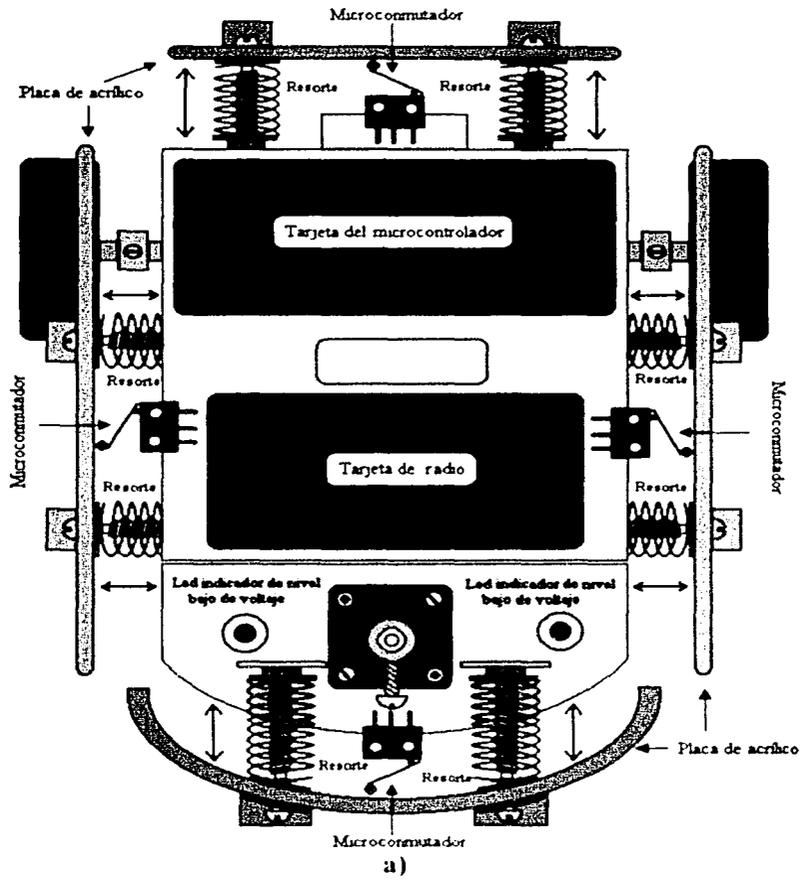


Fig. 3.10 Sensores táctiles del robot: a) vista superior, b) vista lateral.

3.4.3 Parte eléctrica

### 3.4.3 Parte eléctrica

La parte eléctrica está constituida por las siguientes etapas: la que proporciona la alimentación al sistema, la de detección de nivel bajo de voltaje, la de potencia necesaria para el funcionamiento de los motores y la de acondicionamiento de las señales sensadas.

Las etapas que componen la parte eléctrica del robot móvil se ubican en una sola tarjeta, llamada de circuitos. En la Fig. 3.11 se puede observar la ubicación física de las distintas etapas sobre la tarjeta de circuitos.

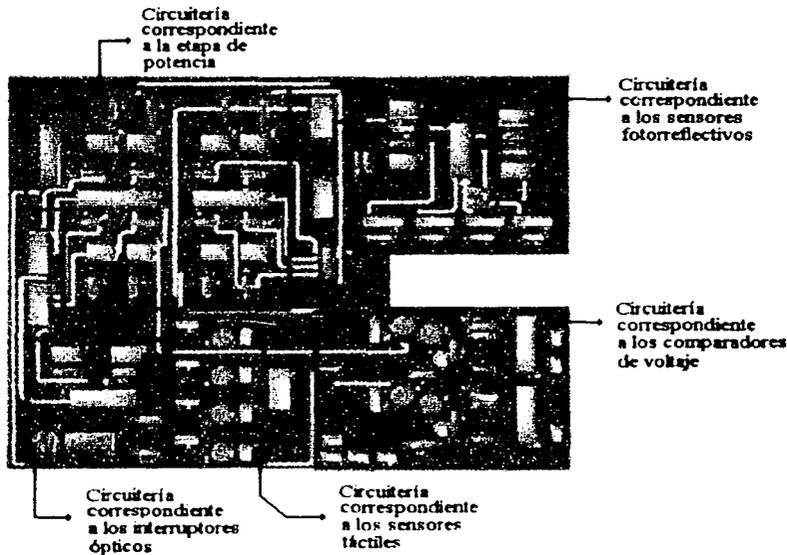


Fig. 3.11 Ubicación de las etapas de la parte eléctrica.

#### Etapa de alimentación

El robot móvil emplea dos baterías para su funcionamiento. Una de las baterías es la encargada de suministrar la potencia a la parte motriz del robot (motores tanto de corriente directa, como de pasos) y la otra es la que proporciona la alimentación tanto a la circuitaría como al sistema mínimo del microcontrolador. Una razón por la cual se emplean dos baterías, es para minimizar el ruido producido por los motores y evitar que éste interfiera con el funcionamiento de la circuitaría adicional y del microcontrolador, y además, de esta forma se aísla la parte de potencia de la parte de control y se previene al microcontrolador y a la circuitaría adicional de un posible corto circuito o de una sobrecorriente.

Las baterías empleadas por el robot móvil son baterías recargables de Níquel-Cadmio de 12 volts (V) a 1.2 amperes/hora (AH). Tienen un peso aproximado de 600 gramos cada una.

Los motores de corriente directa son alimentados con 12 volts (directamente de la batería), y el motor de pasos emplea un voltaje de 8 volts. La circuitería adicional y el microcontrolador utilizan 5 volts para su alimentación. El diagrama de la Fig. 3.12 presenta el circuito eléctrico de la fuente de 8 volts, la cual emplea un regulador de voltaje positivo LM7808. Los capacitores que acompañan al regulador tienen como finalidad reducir los transitorios de voltaje que pudieran presentarse, tanto en la batería (C1) como a la salida del regulador (C2).

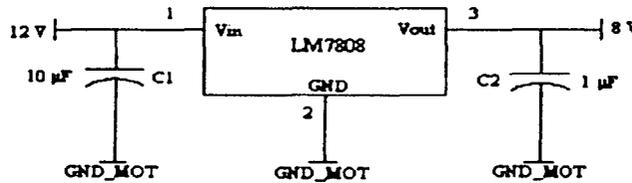


Fig. 3.12 Fuente de voltaje de 8 volts.

En la Fig. 3.13 se muestra el circuito eléctrico de la fuente de alimentación de 5 volts, la cual alimenta a la circuitería y al sistema mínimo del microcontrolador. El circuito es semejante al de la Fig. 3.12, con la diferencia que el regulador utilizado es el LM7805. Este circuito también emplea capacitores para reducir los transitorios de voltaje debidos a la batería o al regulador.

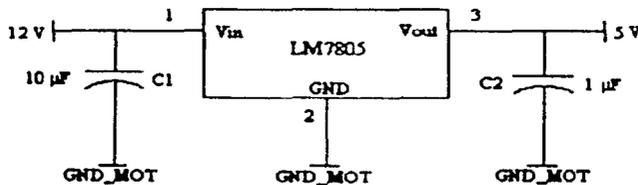
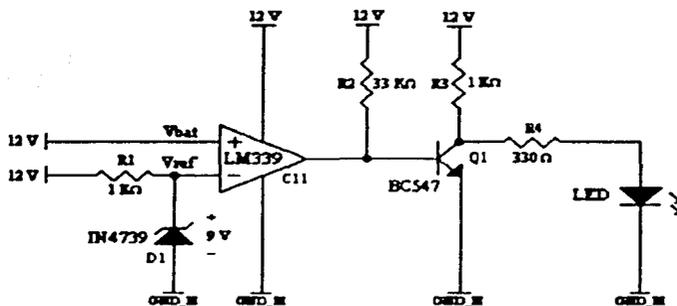


Fig. 3.13 Fuente de voltaje de 5 volts.

### Detección de nivel de voltaje

Con el fin de saber que las baterías proporcionan al robot un nivel de voltaje adecuado, se diseñaron dos detectores de nivel bajo de voltaje, uno para cada batería, esto es para evitar el mal funcionamiento del robot y así, evitar fallas debidas a baterías con un nivel de voltaje bajo. De esta forma se minimiza las fallas electrónicas, para que en caso de existir fallas, éstas sean debidas a la programación y no a la alimentación del robot.

El principio de operación es el mismo en ambos casos; el circuito se basa en un comparador de voltaje no inversor C11, al cual se le aplica un voltaje de referencia ( $V_{ref}$ ) en la entrada inversora y en la entrada no inversora se aplica el voltaje a sensar proveniente de la batería ( $V_{bat}$ ). El voltaje de referencia es proporcionado mediante un diodo zener D1, el cual está conectado al voltaje de la batería mediante una resistencia R1, ver Fig. 3.14.



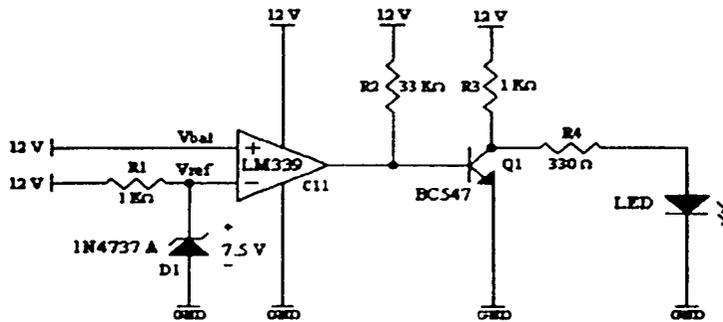
**Fig. 3.14** Circuito del comparador para la batería que suministra a los circuitos de potencia de los motores.

Mientras que el voltaje de la batería sea mayor que el voltaje nominal del diodo zener, el voltaje de referencia será el mismo que el voltaje nominal del diodo zener. En el instante en que el voltaje de la batería sea menor o igual al voltaje nominal del diodo zener, el voltaje de referencia será el mismo que el voltaje de la batería, dado que el diodo zener se comportará como un circuito abierto. Debido a que la impedancia en las entradas del comparador son muy altas, la caída de voltaje en la resistencia será prácticamente cero. Si el voltaje de la batería (entrada no inversora) es mayor que el voltaje de referencia (entrada inversora), la salida del comparador será una señal de 12 volts. Si, por el contrario, el voltaje de la batería es menor o igual que el voltaje de referencia, la salida del comparador será una señal de 0 volts.

En la práctica, los circuitos integrados comparadores de voltaje requieren que el voltaje sea menor y no igual para que ocurra una transición de voltaje en la salida, pero para el caso del circuito integrado LM339 (C11), en su configuración no inversora es suficiente que el voltaje en sus entradas sea el mismo para hacer la transición de voltaje a la salida, cuando se alimenta con +Vcc y tierra. La resistencia R2 es una resistencia de *pull up*. La salida del comparador está conectada a un transistor bipolar de juntura NPN (Q1), el cual está en configuración de emisor común y opera en las regiones de corte y saturación. Cuando en la base se tiene un nivel de voltaje mayor al voltaje de encendido del transistor, éste se pondrá en saturación, y el diodo emisor de luz (LED) no encenderá. Pero cuando en la base se tiene un nivel de voltaje menor al voltaje de encendido del transistor, éste se pondrá en corte y, por consiguiente, el LED se encenderá dado que no circulará corriente por el transistor y ésta pasará a través del diodo.

Para el circuito detector de nivel bajo de voltaje correspondiente a la etapa de potencia de los motores se emplea un diodo zener de 9 volts, por consiguiente, el voltaje de referencia será de 9 volts; se eligió este valor porque a este nivel de voltaje los motores de corriente directa todavía se comportan de una manera estable (el robot todavía se desplaza rápidamente). Para el detector de nivel bajo de voltaje correspondiente al microcontrolador y la circuitería adicional se emplea un diodo zener de 7.5 volts; este voltaje permite al regulador LM7805 mantener un nivel de voltaje de 5 volts a su salida; con un voltaje menor

a 7.5 volts el regulador comienza a presentar una variación de voltaje a la salida, ver Fig. 3.15.



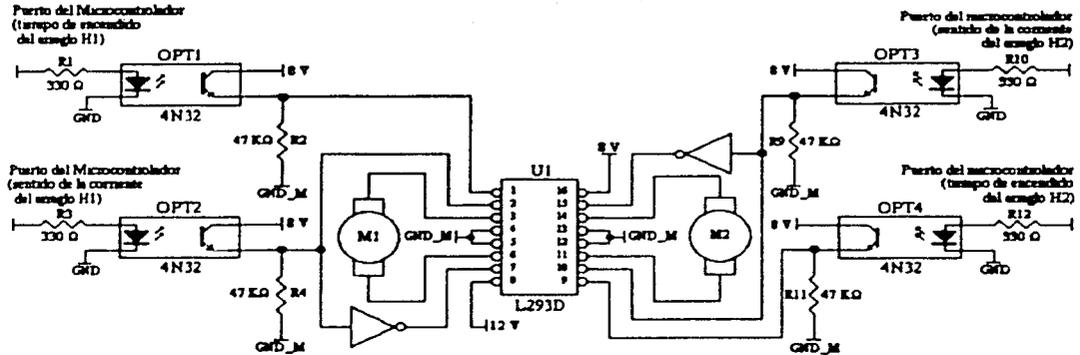
**Fig. 3.15** Circuito del comparador para la batería que suministra a los circuitos de sensado y al microcontrolador.

Los diodos emisores de luz de cada comparador están colocados en la parte frontal del robot; de esta manera es fácil visualizar si las baterías necesitan ser recargadas.

### Etapa de potencia de los motores

La alimentación de los motores se realiza mediante un circuito integrado L293 (U1), que es un manejador (*driver*) adecuado para aplicaciones con motores; internamente cuenta con cuatro arreglos tipo medio H, ver Fig. 3.16. Este tipo de manejador reduce el tamaño físico del circuito porque se pueden alimentar dos motores al mismo tiempo formando dos arreglos tipo H [Anaya, 1998]. Además, si se ocupa este arreglo, solamente se necesita un inversor como componente externo, para el control de los motores. Otra ventaja es que se puede manejar con un voltaje de 5 volts y obtener a la salida voltajes en un rango de 5 a 36 volts, y además cuenta con una señal de habilitación para cada par de arreglos de tipo media H.

Todas las señales de control provenientes del microcontrolador se aíslan de la etapa de potencia de los motores mediante optoacopladores, para prevenir posibles sobrecorrientes que pudieran dañar al microcontrolador, además, evitan que señales de ruido provenientes de los motores puedan interferir en el buen funcionamiento del microcontrolador.



**Fig. 3.16 Diagrama esquemático del circuito eléctrico manejador de los motores de corriente directa.**

El modo de operación es el siguiente, cuando se coloca un motor entre dos arreglos tipo media H, se forma un arreglo H, en el cual el tiempo de encendido del motor se controla mediante la señal de habilitación de dicho par de arreglos media H. La señal para controlar la dirección en la cual circula la corriente en el arreglo H (que determina el sentido de giro del motor) es la entrada a cada arreglo media H, invertida de un arreglo media H con respecto al otro mediante un inversor. Por lo tanto, únicamente se requiere de una señal para el sentido del giro y otra para el tiempo de encendido de cada motor.

Para los motores de corriente directa se requiere de un solo manejador y un inversor. El manejador opera con un voltaje de 5 volts para las señales de control y un voltaje de 12 volts para la alimentación de los motores.

El motor de pasos utiliza sólo un manejador para su operación; cada fase se coloca en un arreglo H y utiliza cuatro líneas de control, las cuales son las entradas de cada arreglo media H. Las líneas de habilitación se conectan a 5 volts y de esta forma el manejador se encuentra activo en todo momento. Las líneas de entrada para el motor de pasos son cuatro en vez de dos como se tiene para los motores de corriente directa (en el caso de los motores de corriente directa, cada señal se invierte para obtener las cuatro líneas de entrada de cada arreglo media H). La razón por la cual se tienen cuatro líneas de control en lugar de dos es para hacer girar al motor de pasos cada medio paso en lugar de cada paso completo, lo cual no se podría lograr si se emplearan solamente dos líneas de control. El voltaje que utiliza el manejador para alimentar al motor de pasos es de 8 volts y el voltaje para las líneas de control es de 5 volts. La Fig. 3.17 muestra el diagrama del circuito eléctrico para la etapa de potencia del motor de pasos.

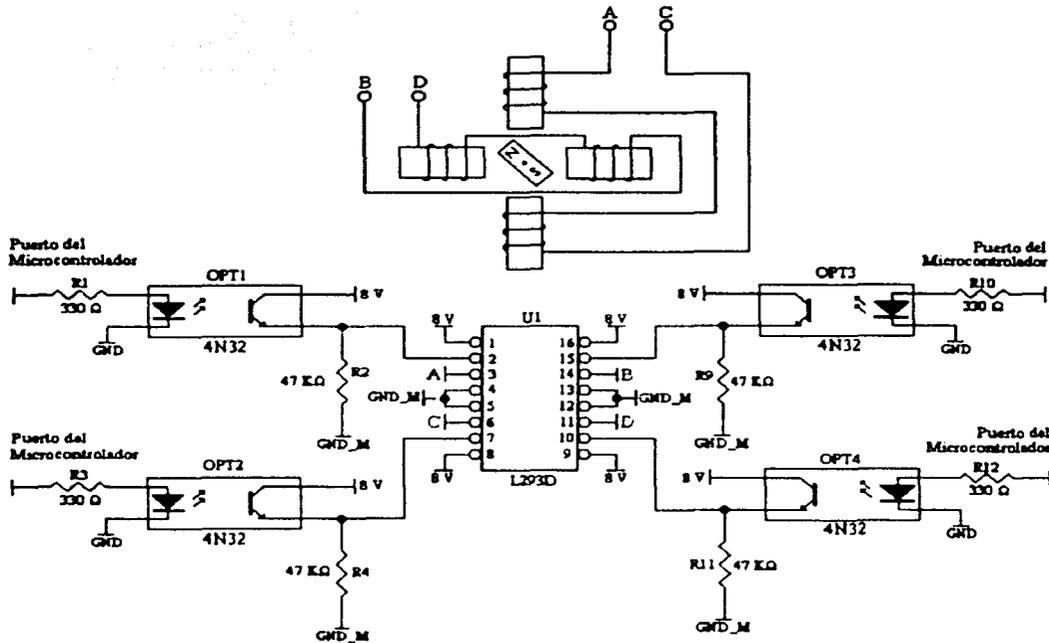


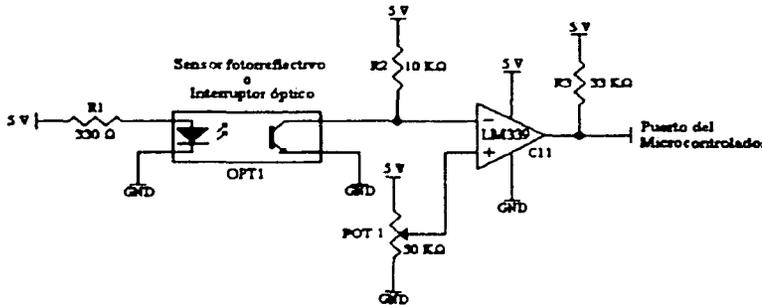
Fig. 3.17 Diagrama del circuito eléctrico de la etapa de potencia del motor de pasos.

### Etapa de acondicionamiento de las señales provenientes de los interruptores ópticos y los sensores fotorreflejos

La etapa de acondicionamiento para las señales provenientes de los interruptores ópticos y los sensores fotorreflejos es la misma, dado que tienen los mismos elementos (fotodiodo y fototransistor); lo único que varía es la posición de éstos (en el caso de los sensores fotorreflejos, el fotodiodo y el fototransistor están colocados paralelamente y, en el caso de los interruptores ópticos, el fotodiodo está colocado enfrente del fototransistor).

La etapa consta de la alimentación del fotodiodo, de la polarización del fototransistor y de la comparación de la señal detectada. La alimentación del fotodiodo se lleva a cabo mediante la resistencia R1, la cual está conectada en serie con el fotodiodo con el único objetivo de limitar la corriente que circula a través de éste. El fototransistor está en configuración emisor común y por medio de la resistencia R2 se polariza. La comparación de la señal detectada se realiza por medio de un comparador inversor, en cuya entrada inversora se conecta la señal proveniente del fototransistor y en la entrada no inversora se tiene un voltaje de referencia que puede ser modificado por medio del potenciómetro POT1. El objetivo de comparar la señal detectada por el sensor fotorreflejo con un voltaje de referencia es para evitar que las variaciones de voltaje producidas por el fototransistor debidas a variaciones de luz en el entorno afecten la lectura de la señal. Así

que, por medio del potenciómetro POT1 se puede ajustar el voltaje de referencia dependiendo de la intensidad de luz que se tenga en el ambiente. Dado que se tiene una configuración emisor común en el fototransistor, en el momento que se enciende éste, la señal a la salida comenzará a bajar su nivel de voltaje y en el momento en que esta señal sea menor que el voltaje de referencia del comparador, éste producirá un “1” lógico, y en el caso contrario, la salida será un “0” lógico. En la Fig. 3.18 se muestra el circuito eléctrico correspondiente a esta etapa [Anaya, 1998].



**Fig. 3.18** Circuito eléctrico correspondiente a la etapa de acondicionamiento de las señales sensadas por los sensores fotorreflectivos.

### Etapa de acondicionamiento de las señales provenientes de los sensores táctiles

Los sensores táctiles utilizan microconmutadores normalmente abiertos. La señal proveniente de cada microconmutador es filtrada antes de ser detectada por el microcontrolador por medio de un filtro paso bajas con la finalidad de evitar transitorios de corriente que pudieran presentarse al accionar el microconmutador. Dicho filtro está implementado por medio de un arreglo RC, cuya frecuencia de corte se determina por la siguiente ecuación.

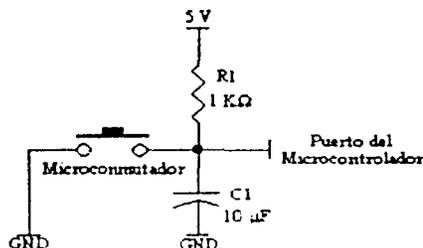
$$f_c = \frac{1}{2\pi RC} \tag{3.28}$$

Se propuso la frecuencia de corte ( $f_c$ ) igual a 15 Hz y se fijó el valor del capacitor ( $C$ ) a 10  $\mu$ F (esto por facilidad, dado que hay una mayor variedad de valores de resistencias que de capacitores). De la ecuación (3.28), al despejar  $R$  se obtiene

$$R = \frac{1}{2\pi f_c C} \tag{3.29}$$

y mediante esta última ecuación se calculó el valor de la resistencia  $R$ , el cual fue 1061.033  $\Omega$ . El valor comercial más cercano es de 1 k $\Omega$ , por esto se eligió  $R$  igual a 1 k $\Omega$ .

Cuando el microconmutador se encuentra abierto la señal detectada es un "1" lógico; en cambio, cuando el microconmutador se cierra la señal detectada es un "0" lógico. En la Fig. 3.19 se muestra el circuito eléctrico de la etapa de acondicionamiento de las señales detectadas por los sensores táctiles.



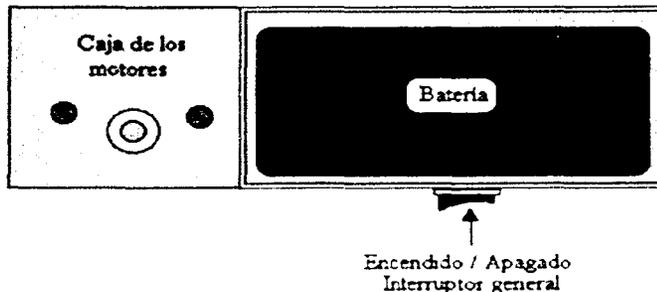
**Fig. 3.19** Circuito eléctrico correspondiente a la etapa de acondicionamiento de las señales sensadas por los sensores táctiles.

### 3.4.4 Armazón

La finalidad del armazón es proporcionar una estructura robusta en la cual se puedan colocar los diferentes elementos que conforman al robot móvil. Como ya se mencionó, el material que se empleó para la construcción del armazón fue acrílico. Todas las piezas de acrílico se unieron mediante cloruro de metileno, una sustancia para pegar acrílico.

Inicialmente, se construyó una caja para albergar los motores de corriente directa, en cuyos extremos se hicieron orificios tanto para las flechas del motor como para los tornillos que sujetan cada motor a dicha caja.

La caja se construyó para evitar que algún objeto pudiera obstaculizar el movimiento de la hélice que se colocó en la flecha de los motores de corriente directa (mostrada en el esquema de la Fig. 3.8); además, esta caja evita la interferencia de la luz en los interruptores ópticos.



**Fig. 3.20** Primer piso del robot.

Posteriormente, se construyó una base en la cual se pudieran colocar tanto la caja de los motores de corriente directa como las baterías. A esta base, en la parte frontal, se le colocó una placa de acrílico de tal forma que la única posibilidad de insertar las baterías fuera por los costados, dado que en la parte posterior se encuentra la caja que contiene los motores de corriente directa. En los costados de dicha base se colocaron dos puertas, una en cada extremo, con el fin de permitir el reemplazo de las baterías. En la Fig. 3.20 se muestra lo detallado anteriormente, a lo que se le denominó el primer piso del robot. También se colocó un interruptor en la parte de abajo de esta base para encender y apagar el robot (interruptor general).

El siguiente paso fue colocar una tapa al primer piso del robot para poder colocar la tarjeta de circuitos. Para pasar los cables de la alimentación provenientes de las baterías y los cables para polarizar y detectar las señales de los interruptores ópticos se hizo un orificio en esta tapa. Para sujetar esta tapa se atornilló a la caja de los motores. Ver Fig. 3.21.

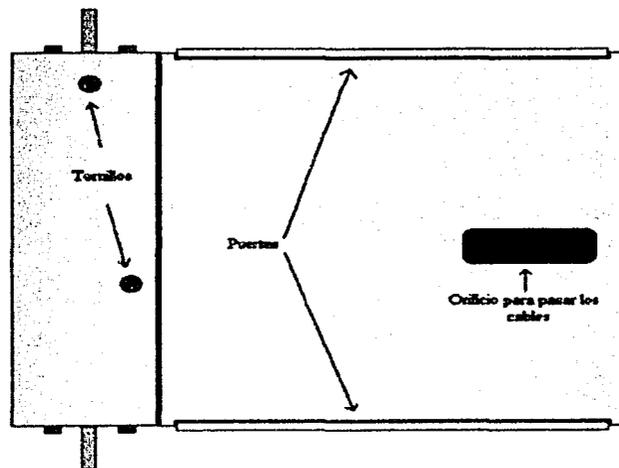


Fig. 3.21 Tapa del primer piso.

El paso posterior en la construcción del armazón fue colocar la base sobre la cual se encuentra colocado el motor de pasos, además de los LED's indicadores de nivel bajo de voltaje. Para esto, se añadió en la parte frontal del armazón una pieza de acrílico a la altura de la tapa del primer piso. Esta pieza de acrílico tiene el mismo ancho que el armazón, pero tiene la característica de que en la parte delantera tiene una forma redondeada (para darle una mejor presentación). Sobre esta base se hicieron los orificios tanto para sujetar al motor de pasos como para su flecha, además de los orificios para los LED's indicadores de nivel bajo de voltaje. Esto se puede observar en la Fig. 3.22.

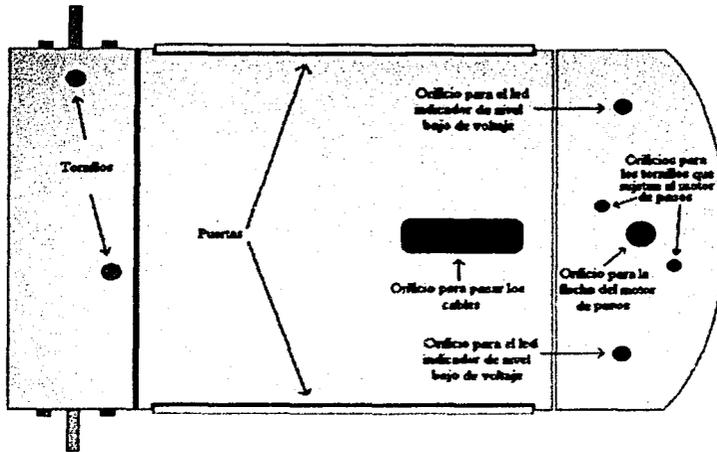


Fig. 3.22 Base para el motor de pasos.

Una vez terminado este primer piso y la base para el motor de pasos, se colocó la tarjeta de circuitos en lo que se define como el segundo piso del robot. Dicha tarjeta se sujetó mediante dos tornillos. En la Fig. 3.23 se puede observar la forma como se coloca la tarjeta a este piso del robot.

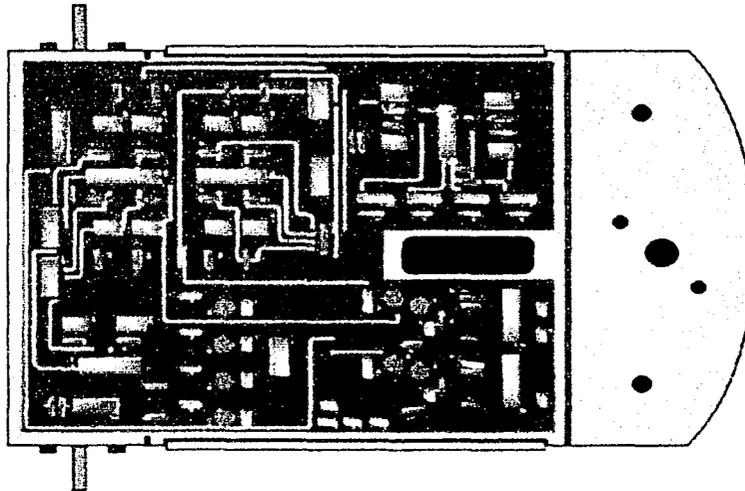


Fig. 3.23 Segundo piso del robot móvil.

Posteriormente se construyó un tercer piso, en el cual se encuentran tanto el sistema mínimo del microcontrolador como el módulo de comunicación. La separación entre el segundo piso y el tercer piso es de 3.5 cm, este espacio es el mínimo requerido para poder manipular los conectores y los elementos que componen a la tarjeta de circuitos. Este tercer

piso al frente termina justo donde comienza la base del motor de pasos, y se sujeta por medio de dos tornillos a la parte trasera del armazón y por un tercer tornillo al segundo piso en la parte frontal. Los cables de la alimentación del sistema mínimo del microcontrolador y los cables correspondientes a los puertos del microcontrolador pasan a través de un orificio en la parte central de dicho piso. También se hizo un orificio para colocar un conector DB-9 hembra para comunicar al microcontrolador con una computadora vía cable serial. Esto se ilustra en la Fig. 3.24.

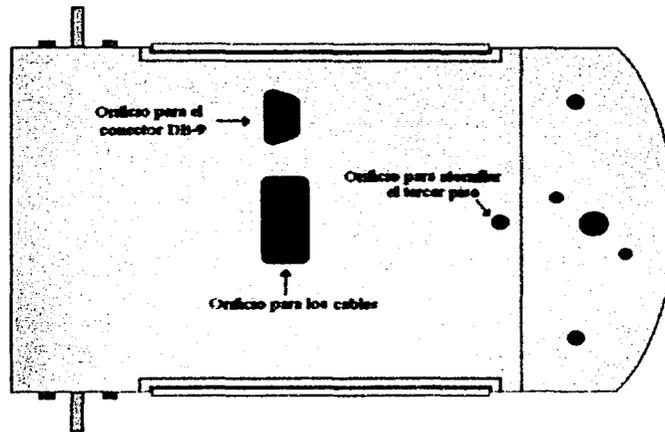


Fig. 3.24 Tercer piso del robot móvil.

En la Fig. 3.25 se muestra una vista lateral del armazón terminado si incluir las tarjetas, ni el motor de pasos, ni las llantas, ni los sensores.

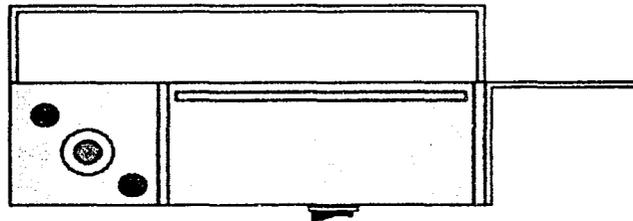


Fig. 3.25 Vista lateral del armazón completo, sin incluir elementos.

Se concluyó la construcción del robot móvil al colocar los sensores fotorreflexivos y los táctiles. Los primeros se fijaron sobre un par de láminas de acrílico, dos sensores en cada lámina. Estas láminas se sujetan al armazón por medio de tornillos, y están colocadas una en la parte frontal y otra en la parte trasera del robot (mostrados en las Figs. 3.26 y 3.27 respectivamente). La distancia entre los sensores fotorreflexivos y el piso para los cuatro casos es de 5 milímetros aproximadamente.

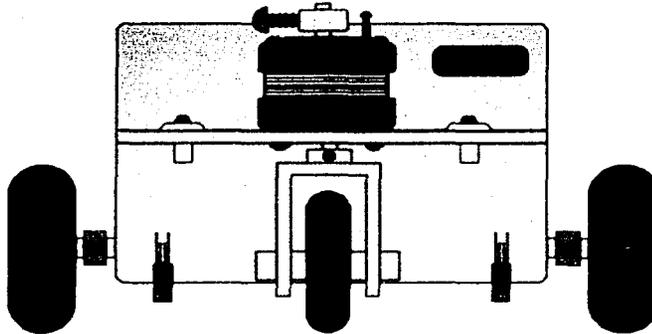


Fig. 3.26 Vista frontal del robot móvil.

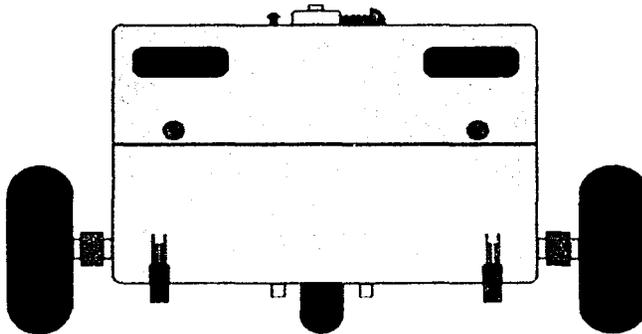


Fig. 3.27 Vista posterior del robot móvil.

Los sensores táctiles se componen de los microconmutadores, las placas de acrílico, los tornillos y los resortes. Primero se cortaron dos láminas de acrílico con forma de cabeza de águila (por razones estéticas), en las cuales se hicieron cuatro perforaciones. Para la parte delantera se cortó y se deformó la placa de acrílico para que tuviera una forma redonda, semejante a la de la base del motor de pasos. En ésta también se hicieron cuatro perforaciones.

Para la parte posterior se cortó una lámina de acrílico y se deformó para obtener una forma redonda; en esta lámina solamente fue necesario hacer tres hoyos. Las perforaciones sirven para colocar los tornillos que sujetan las láminas de acrílico al robot.

En el armazón se colocaron las tuercas de los tornillos que sujetan las placas de acrílico. Dichas tuercas se colocaron en pequeñas placas de acrílico, las cuales se pegaron al armazón para algunos casos (costados y el frente) y en otros se construyeron bases en las cuales se colocaron dichas tuercas (parte posterior y frontal). Entre las placas de acrílico y el armazón se colocaron los microconmutadores. El robot ya terminado se puede observar en la Fig. 3.28.

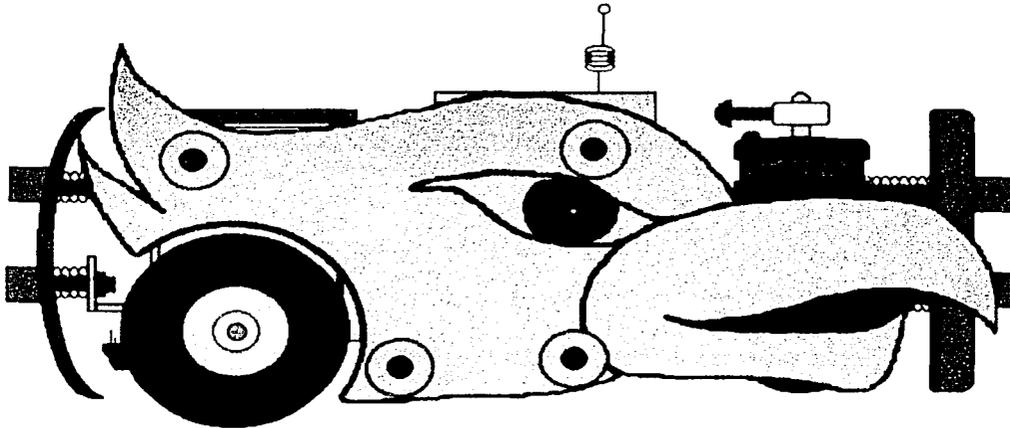


Fig. 3.28 Robot móvil terminado.

Alrededor de cada uno de los orificios sobre los cuales se sujetan los tornillos de las cuatro placas de acrílico, se colocó un tubo de acrílico, con la finalidad de evitar que cuando el robot choque con un objeto, éste no se impacte precisamente en alguno de los tornillos y no sea detectado por los microconmutadores, ya que los tornillos se encuentran fijos. De esta forma se garantiza que el impacto sea sobre los tubos de acrílico, y estos a su vez empujen la lámina de acrílico.

Cabe hacer mención que para poder reemplazar las baterías del robot, es necesario quitar las láminas de acrílico de los costados.

### 3.5 Pruebas preliminares

Una parte fundamental en este trabajo es la calibración de algunos elementos del robot móvil, así como el cálculo de las ecuaciones que modelan el desplazamiento hacia adelante y el giro del robot sobre su propio eje. Una buena calibración y una correcta determinación de los parámetros en las ecuaciones acota el rango de los errores, además de facilitar el manejo y control del robot. La llanta delantera, los sensores fotorreflejos y los sensores táctiles son los elementos del robot móvil que tienen que ser calibrados.

La llanta delantera debe ser calibrada cada vez que se arma el robot móvil, o cada vez que el robot sufre una modificación física.

Los sensores fotorreflejos tienen que ser calibrados para cada lugar y también para cada superficie donde se realicen las pruebas. Esto es porque la cantidad de luz en el ambiente influye en los sensores modificando el umbral de detección; además, cada superficie tiene un índice de reflexión diferente, lo que también modifica dicho umbral. Para cambiar el umbral de detección de cada sensor fotorreflejo en el robot, basta variar el voltaje de referencia con el que se comparan las señales de los sensores fotorreflejos.

Los sensores táctiles deben ser calibrados cada vez que se ensambla el robot móvil, o cada vez que se reemplazan las baterías, debido a que las placas de acrílico que forman parte de estos sensores son la parte más externa del robot.

El ambiente en el cual se desempeña el robot móvil está modelado en coordenadas polares, por lo que resulta necesario que el robot primero gire sobre su propio eje un cierto número de grados y posteriormente se desplace determinada distancia hacia adelante.

Los únicos dispositivos que se tienen para determinar el desplazamiento de los motores de corriente directa, en los cuales están colocadas las llantas traseras, son las señales provenientes de los interruptores ópticos que se encuentran ubicados sobre dichos motores<sup>8</sup>. Por lo tanto, es necesario determinar la relación existente entre el número de veces que se desactiva el interruptor óptico y el desplazamiento del robot móvil hacia adelante o al girar sobre su propio eje. Para obtener dicha relación se aplica un voltaje constante a los motores de corriente directa y se fija el número de veces que se debe desactivar el interruptor óptico, al detectarse dicho número, el robot se detiene y se mide la distancia que el robot se desplazó hacia adelante (la forma en la cual se midió dicha distancia se aborda en la sección 3.5.4), o bien, se mide el ángulo que el robot giró sobre su propio eje (dicha medición se explica en la sección 3.5.5), según sea el caso. Cabe mencionar que, por simplicidad, únicamente se consideró la señal de un interruptor óptico, el del motor 1<sup>9</sup>.

El procedimiento consiste en obtener las distancias recorridas dado el número de veces que se desactivó el interruptor óptico; con esta información se obtiene una ecuación en la cual la variable dependiente sea la distancia recorrida y la variable independiente sea el número de veces que se desactive el interruptor, finalmente, se despeja el número de veces que se desactiva el interruptor óptico, quedando así como variable independiente la distancia por recorrer. Esta última ecuación es la que se emplea para desplazar al robot móvil cierta distancia. Por lo tanto, ahora con una distancia dada el robot calcula el número de veces que debe desactivarse el interruptor óptico antes de detenerse. Este mismo procedimiento se sigue para obtener la ecuación que modela el giro del robot sobre su propio eje.

### 3.5.1 Calibración de la llanta delantera del robot

La llanta delantera está montada sobre una base de acrílico, la cual está sujeta a la flecha que sobresale en la parte inferior del motor de pasos mediante un collarín. En la parte superior de la flecha del motor se encuentra otro collarín, el cual tiene un tornillo en lugar de un opresor, dicho tornillo sirve para detener el giro libre del motor, ya que éste es detenido por otro tornillo al que se denominó tope, el cual está colocado sobre la superficie del motor.

El primer paso para esta calibración es apretar el tornillo correspondiente al collarín localizado en la parte superior de la flecha del motor hasta que quede bien sujeto. Se debe tener cuidado al colocar el tornillo, ya que es necesario que la llanta quede ligeramente

<sup>8</sup> La ubicación de los interruptores ópticos se puede observar en la Fig. 3.7.

<sup>9</sup> En la Fig. E.2 se muestra la ubicación del motor 1.

cargada hacia a la derecha (viendo al robot de frente), tal como se muestra en la Fig. 3.29. El objetivo de colocar la llanta de esta forma es porque es más sencillo alinear la llanta mediante software que atornillarla y lograr que en la posición de referencia la llanta quede perfectamente alineada.

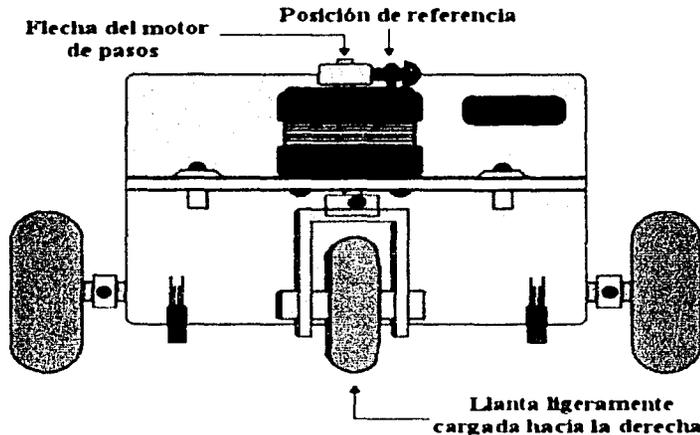


Fig. 3.29 Posición de la llanta delantera.

El siguiente paso de la calibración es alinear la llanta mediante software. Cada vez que se inicializa el robot, éste hace girar al motor de pasos en sentido dextrógiro  $300^\circ$ , y luego gira  $400^\circ$  en sentido levógiro (obviamente no puede recorrer los  $400^\circ$  debido al tope), esto con la finalidad de asegurar que cualesquiera que sea la posición de la llanta, ésta se detendrá justo en el tope. Cabe hacer mención que el tornillo que funge como tope impide que el motor de pasos pueda recorrer los  $360^\circ$ , pues ocupa el espacio que corresponde a  $25.2^\circ$ .

Ya que se encuentra en el punto de referencia (el tornillo tocando justo en el tope), se hace avanzar hacia adelante al robot y se observa que éste se desvía hacia la izquierda (debido a la posición en la cual se sujetó el collarín de la parte superior del motor). Después se mueve unos cuantos grados en sentido dextrógiro al motor y el robot vuelve a avanzar hacia adelante. Este proceso se repite hasta que se observa que el robot avanza derecho. Una vez encontrado el número de grados que se necesita girar el motor para que el robot avance derecho, esta posición es considerada como el "grado cero" para el robot móvil para posteriores aplicaciones.

Este procedimiento se debe llevar a cabo cada vez que se arma el robot móvil o cuando sufre alguna modificación física. Una vez encontrado el grado cero, ésta información se almacena en una variable, la cual es usada en la función de inicialización del robot y de esta forma cada vez que el robot se inicializa, éste se ubica en el grado cero.

Llevando a cabo este procedimiento, se encontró experimentalmente que el grado cero corresponde a un giro del motor de pasos de  $6.3^\circ$  en sentido dextrógiro a partir de la posición de referencia (justo en el tope).

### **3.5.2 Calibración de los sensores fotorreflectivos**

Para cada ambiente diferente en los cuales se van a llevar a cabo las pruebas y para cada superficie distinta es necesario calibrar los sensores fotorreflectivos. La calibración se lleva a cabo poniendo el robot sobre la superficie en la cual se trabajará y se verifica que los sensores detecten las regiones negras y las blancas, si no es así, se modifica el voltaje de referencia con el que se comparan las señales provenientes de dichos sensores.

Para modificar el voltaje de referencia basta con mover los potenciómetros ubicados en la etapa de acondicionamiento de las señales sensadas provenientes de los sensores fotorreflectivos, ver Fig. 3.11.

### **3.5.3 Calibración de los sensores táctiles**

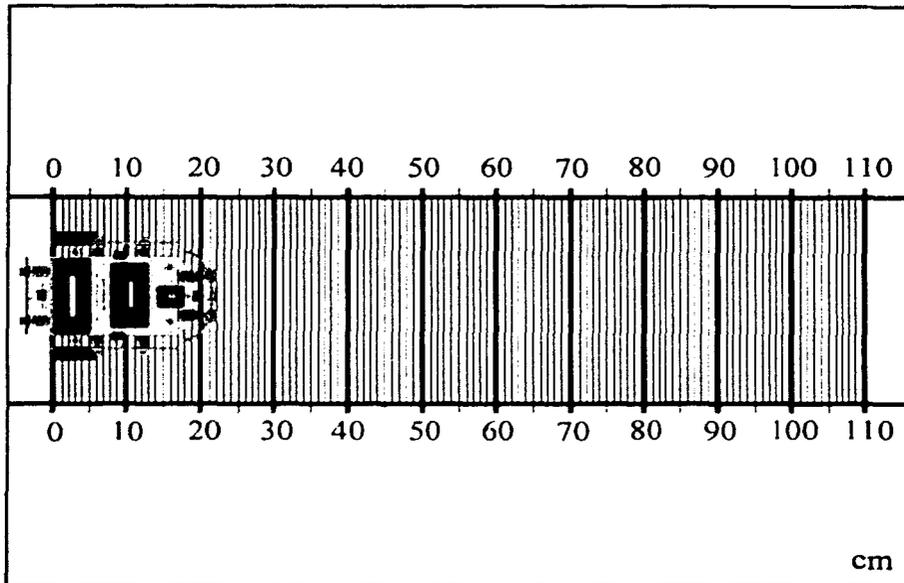
Esta calibración es la más sencilla. Su facilidad radica en que las placas de acrílico que conforman los sensores táctiles se encuentran al exterior del robot móvil.

Para ajustar las placas de acrílico basta atornillar o desatornillar los tornillos que las sujetan: para las placas laterales y para la frontal son cuatro y para la placa trasera únicamente tres. Debido a que los microconmutadores producen un sonido detectable al activarse y desactivarse, es fácil verificar que las placas de acrílico están bien colocadas. Las placas de acrílico deben colocarse de tal forma que hagan contacto con los rodillos de los microconmutadores, pero teniendo precaución de que no queden cerrados o activados. Además, se debe cuidar que las placas de acrílico queden colocadas a una distancia uniforme del robot, para evitar que éstas queden más abiertas de un lado que de otro, lo que produciría que no se activen correctamente los microconmutadores.

### **3.5.4 Obtención de la ecuación que determina el desplazamiento del robot hacia adelante**

Después de haber realizado las calibraciones anteriores, se realiza este proceso. Este procedimiento debe ser repetido cada vez que se realice la calibración de la llanta delantera, o si la superficie de rodamiento cambia (ya que se tiene una fricción diferente), o si las condiciones de limpieza del piso o de las llantas se modifican, o bien, cada vez que el robot sufra alguna modificación física.

Para la obtención de los datos experimentales se colocó al robot móvil sobre un pliego de papel bond graduado previamente, como se muestra en la Fig. 3.30. La escala de la graduación está en centímetros.



**Fig. 3.30** Papel bond graduado y el robot móvil.

Se fija el número de veces que se debe desactivar el interruptor óptico y el robot avanza hacia adelante hasta que se alcance dicho número. Una vez que se detiene el robot, se marca la posición final de éste sobre el papel y con la graduación del papel se miden los centímetros, si el robot no quedó exactamente en una de las rayas marcadas, entonces con un escalímetro se miden los milímetros con el fin de tener una mayor resolución en la medición. Este experimento se repite tres veces para cada ocasión que se fija el número de veces que se desactiva el interruptor óptico, debido a la poca discrepancia que se presentó en los datos. Con estas tres mediciones se saca un promedio. El primer valor del número de veces que se desactiva el interruptor óptico es 100 y se incrementa en 25 para cada experimento, hasta llegar a 575. Se toma como primer valor el 100 porque se encontró experimentalmente que con este valor el robot móvil avanzaba alrededor de 10 cm, lo cual representa una distancia significativa para comenzar a hacer las mediciones, dado que, por el tipo de pruebas que se realizarán posteriormente, no se está interesado en distancias más cortas. Se tomó como valor final 575 porque se observó que el comportamiento del robot es lineal para estas pruebas.

La tabla 3.3 muestra los datos experimentales obtenidos en cada una de las repeticiones y el valor promedio de aquéllos.

No. de veces que se desactiva el interruptor óptico	Primera repetición [cm]	Segunda repetición [cm]	Tercera repetición [cm]	Promedio [cm]
100	8.1	7.9	8.0	8.00
125	10.0	10.1	10.0	10.03
150	12.2	12.0	11.9	12.03
175	14.4	14.1	14.3	14.27
200	16.7	16.2	16.1	16.33
225	18.3	18.7	18.3	18.43
250	20.2	20.1	20.3	20.20
275	22.4	22.2	22.5	22.37
300	24.9	24.0	24.3	24.40
325	27.2	26.6	27.3	27.03
350	29.2	29.0	29.5	29.23
375	30.3	30.5	30.8	30.53
400	32.7	32.1	32.6	32.47
425	34.4	35.0	34.5	34.63
450	36.6	36.6	36.8	36.67
475	38.9	38.6	38.5	38.67
500	40.8	40.5	41.0	40.77
525	42.7	42.4	42.7	42.60
550	45.2	45.4	45.3	45.30
575	47.6	47.1	47.1	47.27

**Tabla 3.3 Datos experimentales de la prueba con el robot avanzando hacia adelante.**

Dado el comportamiento de los datos experimentales, se utilizó el método de mínimos cuadrados para ajustar los datos a una recta. La ecuación obtenida mediante este método es:

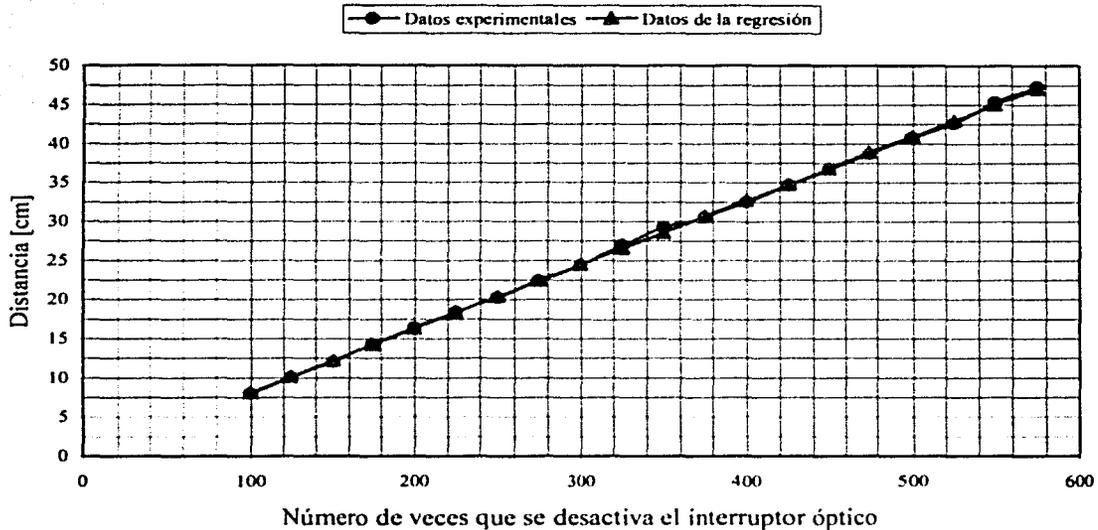
$$\text{distancia} = 0.082013 * (n) - 0.117833 \quad (3.30)$$

Donde n es el número de veces que se desactiva el interruptor óptico.

Y al despejar el número de veces que se activa el interruptor óptico (n), se obtiene:

$$n = 12.193139 * (\text{distancia}) + 1.436758 \quad (3.31)$$

La Fig. 3.31 muestra la gráfica correspondiente al número de veces que se desactiva el interruptor óptico contra la distancia: en color azul la recta obtenida con los datos experimentales, y la recta en color rojo corresponde a la recta obtenida con el método de pares de puntos.



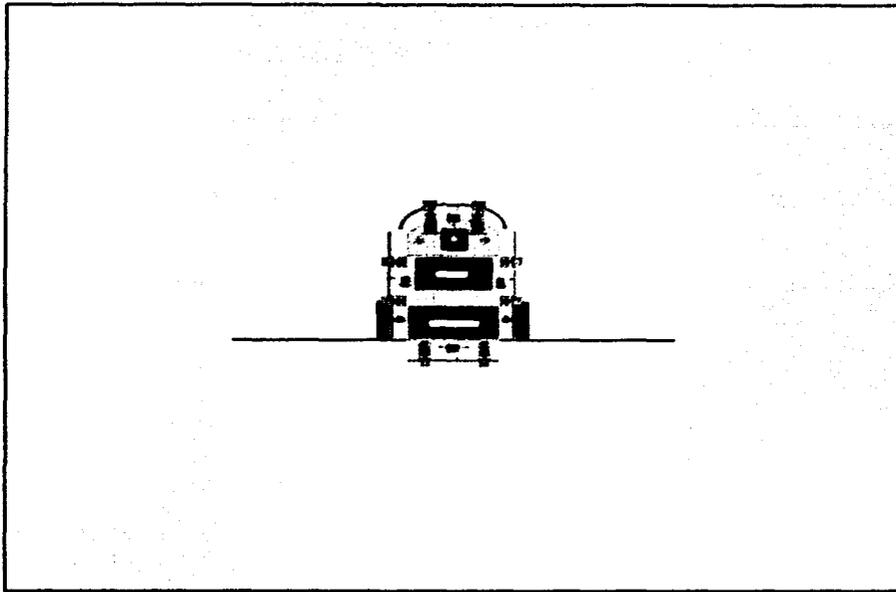
**Fig. 3.31 Gráfica de número de veces que se desactiva el interruptor óptico contra distancia.**

Las pruebas se realizaron en un piso con duela de madera, sobre el cual se colocó el papel bond. El voltaje de la batería que alimenta a los motores, al inicio de la prueba fue de 13.2 volts y al término de ésta fue de 11.9 volts.

### 3.5.5 Obtención de la ecuación que determina el giro del robot sobre su propio eje

Este procedimiento, al igual que en la sección anterior, también debe ser repetido cada vez que se realice alguna modificación física en el robot móvil, o si la superficie de rodamiento cambia, entre otras.

Para la obtención de los datos experimentales se colocó al robot móvil sobre un pliego de papel bond blanco, al cual se le marcó únicamente una línea de referencia. En la Fig. 3.32 se ilustra esto.



**Fig. 3.32 Papel bond blanco con la línea de referencia y el robot móvil.**

Se fija un número de veces que se debe desactivar el interruptor óptico y el robot gira sobre su propio eje en sentido dextrógiro (girando los motores de corriente directa en el mismo sentido y la llanta delantera perpendicular a las llantas traseras), hasta que se alcance dicho número, una vez que se detiene el robot se marca la posición de éste y con la ayuda de escuadras se prolonga la línea de la posición del robot hasta cruzar con la línea de referencia. El ángulo descrito por la línea de la posición y la línea de referencia es medido con un transportador. Como en el caso anterior, este experimento se repite tres veces para cada ocasión que se fija el número de veces que se desactiva el interruptor óptico y con estas tres mediciones se obtiene un promedio. El primer valor del número de veces que se desactiva el interruptor óptico es 40 y se incrementa en 20 para cada experimento, hasta finalizar en 340.

Se consideró como primer valor el 40 porque experimentalmente se encontró que con este valor el robot móvil gira alrededor de  $20^\circ$ , lo cual representa un ángulo significativo para comenzar las mediciones, debido a que, por el tipo de pruebas que se realizarán posteriormente, no se está interesado en ángulos más pequeños. Se escogió como valor final 340 porque el comportamiento del robot es lineal para estas pruebas.

En la tabla 3.4 se presentan los datos experimentales obtenidos en cada una de las repeticiones y el valor promedio de éstos.

No. de veces que se desactiva el interruptor óptico	Primera repetición [°]	Segunda repetición [°]	Tercera repetición [°]	Promedio [°]
40	20	20	20	20.00
60	30	29	30	29.67
80	40	39	41	40.00
100	50	48	49	49.00
120	60	59	59	59.33
140	70	70	69	69.67
160	79	80	78	79.00
180	89	87	88	88.00
200	99	102	98	99.67
220	108	107	107	107.33
240	117	117	115	116.33
260	123	125	125	124.33
280	132	132	135	133.00
300	145	148	145	146.00
320	153	153	153	153.00
340	161	161	161	161.00

**Tabla 3.4 Datos experimentales de la prueba con el robot girando sobre su propio eje.**

Por el comportamiento que presentaron los datos experimentales, se optó también por emplear el método de mínimos cuadrados para ajustar los datos a una recta. La ecuación obtenida para el giro del robot sobre su propio eje mediante este método es:

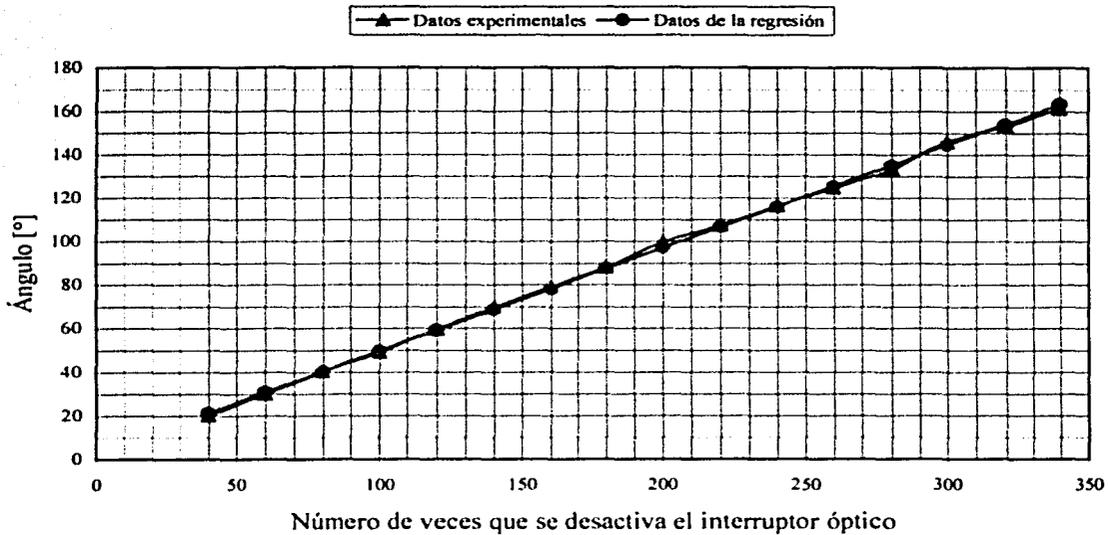
$$\text{ángulo} = 0.473438 * (m) + 2.296875 \quad (3.32)$$

Donde m es el número de veces que se desactiva el interruptor óptico.

Y al despejar el número de veces que se activa el interruptor óptico (m), se obtiene:

$$m = 2.112211 * (\text{ángulo}) - 4.851485 \quad (3.33)$$

La Fig. 3.33 muestra la gráfica correspondiente al número de veces que se desactiva el interruptor óptico contra el ángulo que gira el robot móvil sobre su propio eje: en color azul la recta obtenida con los datos experimentales, y la recta en color rojo corresponde a la recta obtenida con el método de pares de puntos.



**Fig. 3.33** Gráfica de número de veces que se desactiva el interruptor óptico contra ángulo.

Al igual que en la prueba anterior, las pruebas se realizaron sobre papel bond en un piso con duela de madera. El voltaje de la batería que alimenta a los motores, al inicio de la prueba fue de 13.3 volts y al término de ésta fue de 11.6 volts.

### 3.6 Conclusiones

En este capítulo se presentan los distintos módulos implantados para determinar el sistema móvil autónomo.

Se eligió a la red de funciones de base radial debido a que presenta la estabilidad necesaria ante escenarios con fallas.

Asimismo, la red neuronal es entrenada con base en el uso de la técnica de agrupamiento fuzzy c-means y en el uso de mínimos cuadrados ortogonales. Esto hace que la red tenga un mínimo de variables aleatorias en su manejo de datos. Por otro lado, el entrenamiento propuesto para la red utiliza una base de datos obtenida en el seguimiento de rutas sin fallas como se verá en el siguiente capítulo.

# Capítulo 4

## Evaluación de la red neuronal

### 4.1 Introducción

En este capítulo se presenta el entrenamiento y evaluación de la red neuronal con respecto a distintos escenarios definidos como rutas. Cabe señalar que el entrenamiento es llevado a cabo fuera de línea con respecto al recorrido del robot.

Debido a que la comunicación inalámbrica entre el robot y la computadora resultó ineficiente, la evaluación de la red se basó en una simulación realizada en MATLAB para escenarios con y sin fallas. Cabe hacer mención que los datos de los escenarios con y sin fallas se obtuvieron de recorridos que el robot realizó en cada una de las rutas.

### 4.2 Revisión del entrenamiento de la red neuronal

En esta sección se presentan las rutas que el robot deberá recorrer con la finalidad de entrenar la red neuronal. Así mismo, se detalla el tratamiento que se le da a la información y los resultados del entrenamiento de la red neuronal.

#### 4.2.1 Obtención y significado de la información

Con el fin de obtener los datos para el entrenamiento de la red neuronal, el robot móvil recorre tres distintos escenarios o rutas. Para los tres recorridos el robot parte del punto A (punto inicial) y tiene como destino final el punto B (punto final), tal como se muestra en la Fig. 4.1. Durante estos recorridos, el robot no encuentra ningún obstáculo en su desplazamiento.

Como se observa en la Fig. 4.1, la orientación inicial del robot móvil para el recorrido de las tres rutas es hacia el norte (según la referencia mostrada en la parte superior izquierda de dicha figura). Para cubrir la ruta 1, el robot debe avanzar hacia adelante 90 cm, posteriormente hacer un giro de 90 grados en sentido dextrógiro y luego avanzar hacia adelante 190 cm. En la ruta 2, el robot debe iniciar girando 90 grados en sentido dextrógiro, luego avanzar hacia adelante 190 cm, después hacer un giro de 270 grados en sentido dextrógiro y finalmente avanzar hacia adelante 90 cm. Y por último para la ruta 3, que es la diagonal del rectángulo, la recorre de la siguiente forma, estando en el punto inicial, gira 67 grados en sentido dextrógiro y recorre 205 cm. Como se puede notar, el robot sólo gira en

sentido dextrógiro, ya que las pruebas para la obtención de la ecuación que determina tanto el giro del robot sobre su propio eje se hicieron en este sentido.

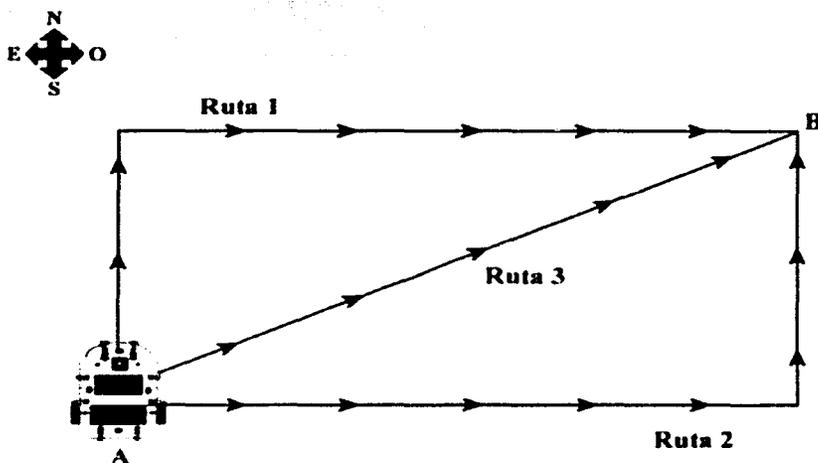


Fig. 4.1 Rutas propuestas para el entrenamiento de la red neuronal.

Durante el recorrido de cada ruta, el robot calcula la distancia que recorre cada uno de los motores de corriente directa cada 0.33 segundos y la almacena en una sección de la memoria RAM del microcontrolador. Para el cálculo de la distancia, el robot almacena el número de veces que se desactiva el interruptor óptico de cada uno de los motores cada 0.33 segundos, con estos datos hace el cálculo de la distancia aproximada recorrida empleando la ecuación (3.30).

Además, cada 0.33 segundos, el robot móvil registra el estado de sus sensores táctiles y fotorreflectivos. Esta información se almacena también en la memoria RAM del microcontrolador del robot móvil.

Terminado cada uno de los recorridos, la información obtenida es enviada a una computadora para su procesamiento. Una vez que se tiene la información de los tres, ésta se utiliza para entrenar la red neuronal.

La pista de entrenamiento está hecha sobre piel sintética de color negro mate, cuyas dimensiones son 247 cm de largo por 142 cm de ancho. Las divisiones están hechas con cinta de aislar plástica de color blanco, la cual tiene 1.9 cm de ancho. Para las pruebas, la pista se extendió sobre un piso de madera. La pista de entrenamiento se muestra en la Fig. 4.2.

Los rectángulos blancos colocados sobre la pista están a 9.5 cm de distancia uno de otro, y las dimensiones de éstos son de 34.5 cm de largo por 5.7 cm de ancho. Los cuadros negros

de los extremos tienen 34.5 cm por lado. El objetivo de tener rectángulos blancos sobre las rutas a seguir por el robot móvil es obtener más información de la ruta recorrida.

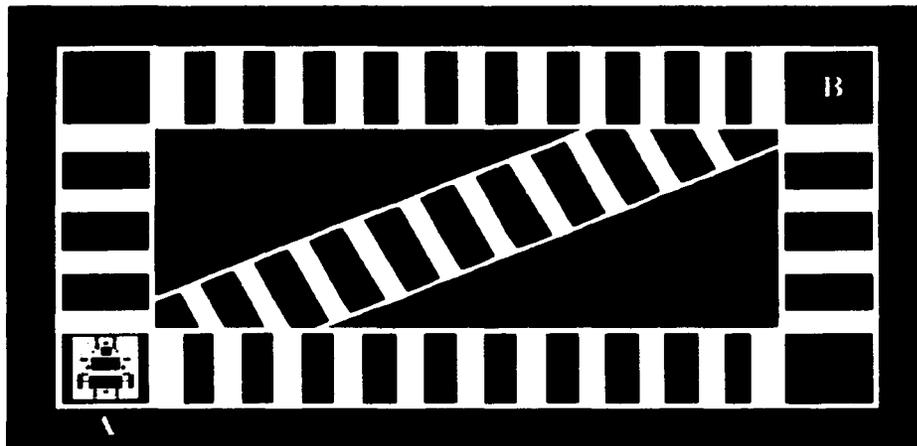


Fig. 4.2 Pista de entrenamiento de la red neuronal.

Cabe mencionar que cuando el robot gira sobre su propio eje para recorrer cierto ángulo el motor 2 gira en sentido opuesto al movimiento que realiza cuando el robot avanza hacia delante y, por lo tanto, el desplazamiento angular que registra el interruptor óptico del motor 2 se multiplica por  $-1$  para identificar que dicho motor cambió su sentido de giro. El motor 1 en ningún momento cambia su sentido de giro para estas pruebas.

Es importante resaltar que la información que se obtiene de cada escenario no tiene el mismo número de datos, debido a que, el tiempo que emplea el robot móvil en cada recorrido es diferente. Por ejemplo, del escenario 2 se obtienen más datos y del escenario 3 menos que en cualquiera de los otros dos escenarios.

#### 4.2.2 Tratamiento de la información

Una vez que se tiene la información de las tres rutas, se procede a ordenarla. Con la información de cada ruta se forma una matriz de datos de entrada de la siguiente forma: en la primera columna se colocan los datos obtenidos del motor 1 (en el caso de su movimiento hacia adelante se trata de distancias y en el caso del giro se trata de desplazamientos angulares), en la segunda columna se tienen los datos del motor 2, y finalmente, en la tercera columna se encuentran los datos registrados por los sensores fotorreflectivos y táctiles.

La información proveniente de los sensores fotorreflectivos y táctiles es recogida por el puerto G del microcontrolador; en la tabla E.4 se puede observar la posición que ocupa cada uno de estos sensores en el puerto.

Cada renglón de la matriz de datos de entrada representa un tiempo diferente en el cual se tomaron las lecturas. La diferencia de tiempo entre un renglón y el siguiente es de 0.33 segundos. Así, la matriz queda de la siguiente forma:

$$\begin{bmatrix} \text{Datos del motor } 1_1 & \text{Datos del motor } 2_1 & \text{Sensores}_1 \\ \text{Datos del motor } 1_2 & \text{Datos del motor } 2_2 & \text{Sensores}_2 \\ \vdots & \vdots & \vdots \\ \text{Datos del motor } 1_n & \text{Datos del motor } 2_n & \text{Sensores}_n \end{bmatrix}$$

donde  $n$  representa el número de datos que se obtiene para cada ruta.

El vector de salida se forma con los datos que definen el comportamiento deseado del motor 2. Cuando se habla del comportamiento deseado se refiere al desempeño que el motor 2 presenta cuando avanza hacia adelante y no sufre ninguna desviación o cuando gira sobre su propio eje de forma correcta.

Se toma únicamente como salida el comportamiento deseado del motor 2 debido a que es el que tiene cambios en su sentido de giro durante el recorrido de las rutas, no así el motor 1, el cual prácticamente tiene un comportamiento constante durante todo el recorrido.

En la siguiente matriz de datos de entrada se despliegan sólo algunos datos correspondientes al escenario 1 sin fallas. En la segunda columna, la cual representa los datos del motor 2, se puede notar que cuando el robot gira sobre su propio eje, la cantidad obtenida del motor aparece como un número negativo. Los datos de la columna 3 están en el rango de 240 a 255 debido a que los sensores táctiles están ubicados en los cuatro bits más significativos del puerto G y dichos sensores arrojan un "1" lógico cuando no están activados.

3.4907	3.5727	252
3.4087	3.4907	240
3.5727	3.4907	240
3.5727	3.4087	240
3.5072	-3.5327	240
3.5327	-3.5327	240
3.5327	-3.5327	240
3.5327	-3.5327	240
3.5327	-3.5327	240
2.4245	2.3425	240
3.4907	3.4087	243
3.5727	3.4907	243
3.4907	3.4907	240
3.5727	3.4907	240
3.4907	3.4907	240

De los tres recorridos se obtienen tres matrices de datos de entrada, una para cada ruta, con sus correspondientes vectores de salida. El siguiente paso consiste en formar una sola matriz de datos de entrada y un solo vector de salida con los datos de las tres rutas, los primeros 80 datos corresponden a la ruta 1, los siguientes 92, a la ruta 2 y los últimos 58 datos pertenecen a la ruta 3. Por lo tanto, se obtiene una matriz de datos de entrada de 230 renglones por 3 columnas y un vector de salida de 230 renglones.

Una vez obtenidas estas dos matrices, se deben normalizar los datos, ya que como se ve en la matriz anterior, los datos de la tercera columna no tienen la misma magnitud que los datos de las dos primeras, por esta razón todos los datos se normalizan entre cero y uno con la finalidad de que todas las cantidades tengan la misma magnitud y sean igualmente significativas para la red neuronal. La normalización se lleva a cabo para cada columna por separado, tanto para la matriz de datos de entrada como para el vector de salida, donde el valor máximo corresponde a uno y el valor mínimo corresponde a un cero.

El siguiente paso consiste en tomar la matriz de datos de entrada y el vector de salida obtenidos para entrenar la red neuronal y así obtener los pesos de dicha red.

#### 4.2.3 Resultados del entrenamiento de la red neuronal

Con base en lo mostrado en los capítulos anteriores se utilizó la red de funciones de base radial, donde la función no lineal que se utilizó en la red neuronal es la función gaussiana. Por lo tanto, los elementos de la matriz  $P$  quedaron definidos como

$$p_{kl} = \prod_{r=1}^m \exp\left(-\frac{1}{2} \left(\frac{x_{kr} - c_{lr}}{\sigma_{lr}}\right)^2\right) \quad (4.1)$$

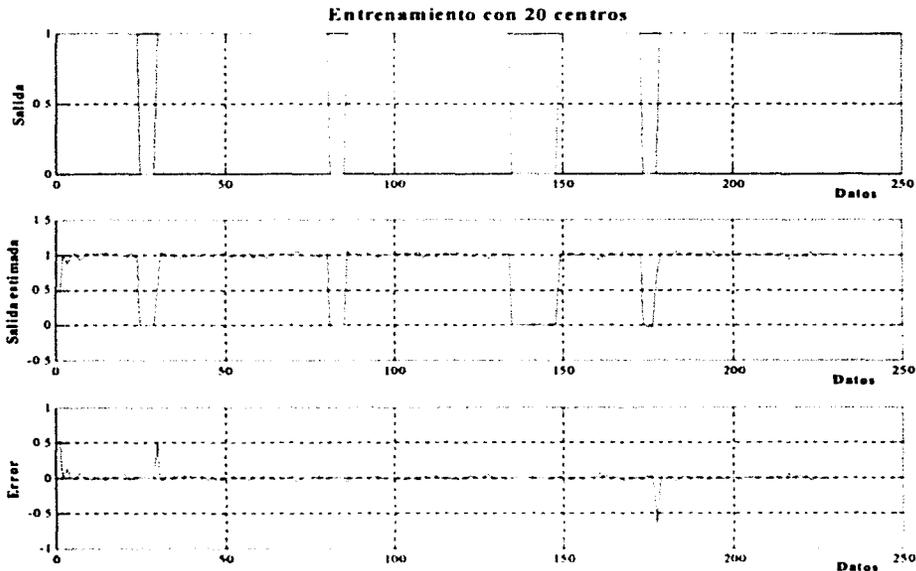
donde  $p_{kl}$  son los elementos que conforman a la matriz  $P$ ,  $x_{kr}$  representa los datos de entrada,  $c_{lr}$  los centros y  $\sigma_{lr}$  las desviaciones estándar.  $m$  es el número de entradas, en este caso  $m = 3$ ,  $k$  es el número de renglones de la matriz de datos de entrada y  $l$  es el número de renglones de la matriz de centros.

Experimentalmente se probó el entrenamiento de la red neuronal para un número distinto de renglones en la matriz de centros y se encontró que la matriz más adecuada para el entrenamiento de la red neuronal es de 80 renglones por 3 columnas, si se considera un error del orden del 0.05 %. Se tienen tres columnas en la matriz de centros debido a que ésta debe tener el mismo número de columnas que la matriz de datos de entrada.

La matriz de desviaciones estándar es del mismo orden que la matriz de centros, por tanto, también tiene 80 renglones por 3 columnas. Finalmente, el vector de pesos tiene 80 elementos, que es el número de renglones de la matriz de centros.

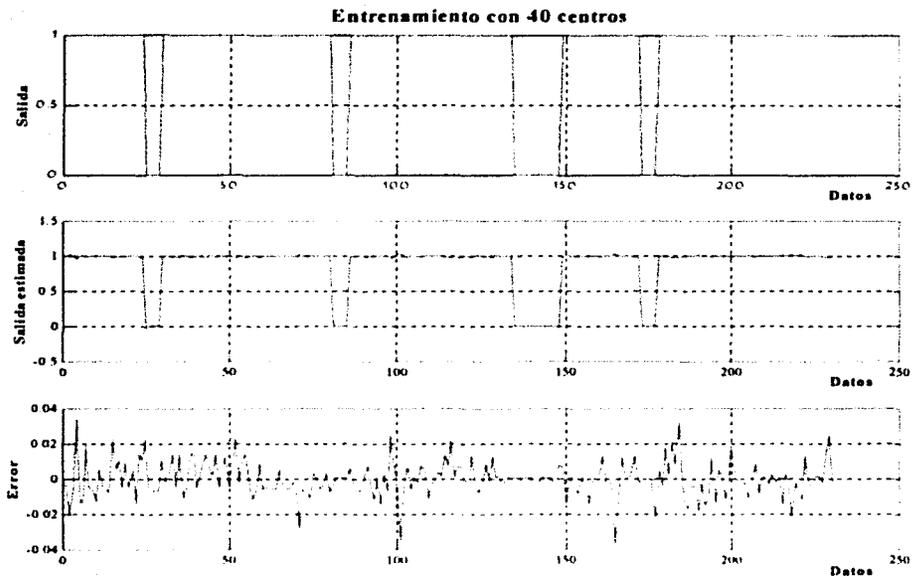
Después de que se han obtenido los centros y las desviaciones estándar, se procede a obtener los pesos de la red. Ya conocidos los pesos, se prueba la red neuronal con la misma matriz de datos de entrada, esto para obtener una salida estimada y calcular un error.

Después de haber hecho el entrenamiento para cada caso se prueba a la red con respecto a los tres escenarios propuestos dando por resultado lo siguiente.

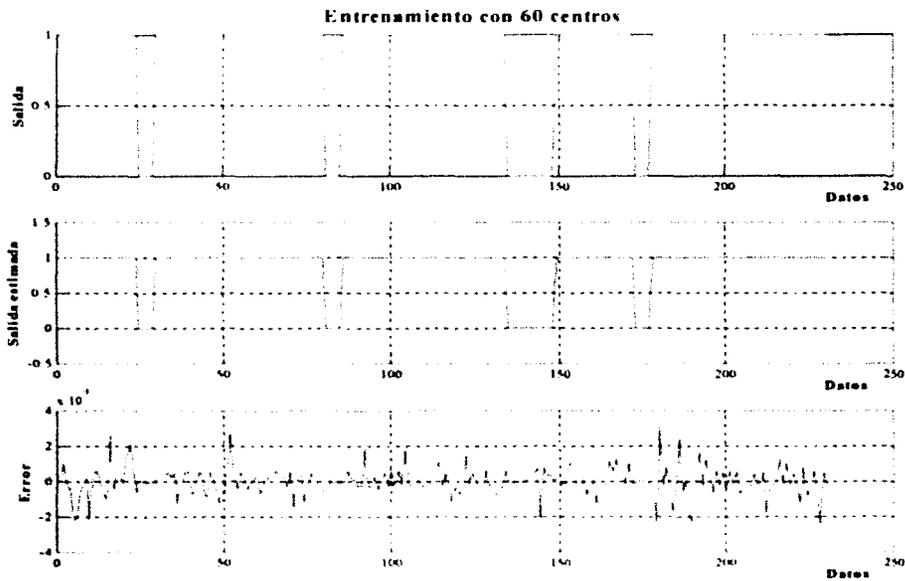


**Fig. 4.3** Gráfica correspondiente al entrenamiento de la red neuronal con 20 renglones en la matriz de centros.

La Fig. 4.3 muestra la respuesta de la red neuronal respecto a los tres escenarios sin fallas con 20 renglones en la matriz de centros. Dichos escenarios están concatenados unos a otros. El primer escenario corresponde a la ruta 1, que va desde el dato 1 hasta el dato 80. El segundo escenario corresponde a la ruta 2, que va desde el dato 81 hasta el dato 172. Por último, el tercer escenario corresponde a la ruta 3, que va desde el dato 173 hasta el dato 230. En esta figura la red neuronal es entrenada para seguir la respuesta del motor 2 que es éste el que presenta los mayores cambios cuando se ejecutan las tres rutas. Como se puede ver en esta figura, el error obtenido es relativamente pequeño aún cuando presenta ciertos disparos de valores bajo ciertos movimientos del motor 2 del robot.



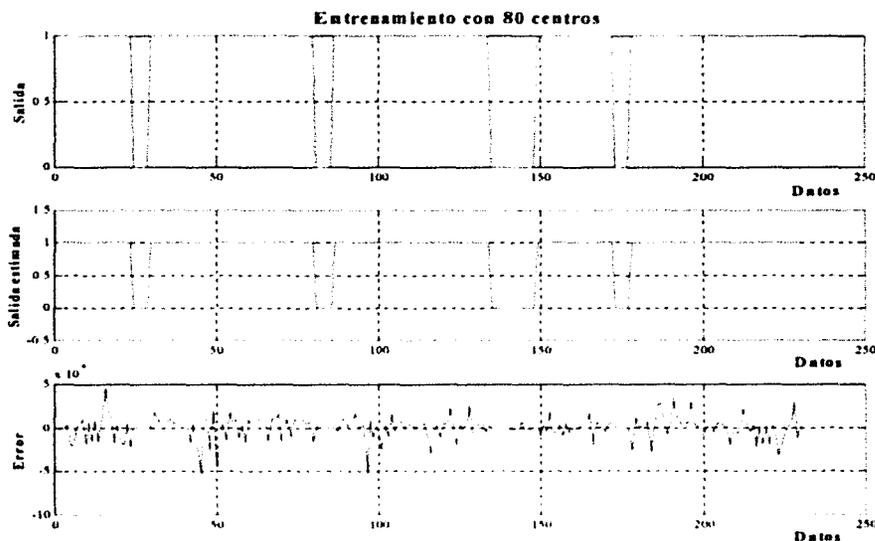
**Fig. 4.4** Gráfica correspondiente al entrenamiento de la red neuronal con 40 renglones en la matriz de centros.



**Fig. 4.5** Gráfica correspondiente al entrenamiento de la red neuronal con 60 renglones en la matriz de centros.

La Fig. 4.4 presenta la respuesta de la red neuronal con un número mayor de centros (40); con este incremento se reduce el error de manera considerable, hasta un 5% del valor de la salida normalizada.

La Fig. 4.5 presenta la misma respuesta, pero para un número mayor de centros (60). El lector puede verificar que la reducción del error es considerable.



**Fig. 4.6 Gráfica correspondiente al entrenamiento de la red neuronal con 80 renglones en la matriz de centros.**

Por último, la Fig. 4.6 muestra la respuesta de la red con 80 centros; en este caso el error es de alrededor de 0.04% sobre el valor normalizado a seguir. El costo que se paga por el uso de un mayor número de centros es el incremento en el número de operaciones.

### 4.3 Evaluación del escenario 1

El escenario 1 está definido como la ruta 1. Para evaluar este escenario se utiliza la matriz de datos que se obtiene del robot al recorrer esta ruta y dichos datos se toman como la entrada a la red neuronal ya entrenada.

Cabe hacer mención que la matriz de centros contempla 80 de éstos y la matriz de pesos tiene la dimensión de 80 renglones por 1 columna. La primera fue obtenida con base en el uso del algoritmo de agrupamiento fuzzy c-means y la segunda con base en la técnica de mínimos cuadrados ortogonales.

La matriz de entrada de datos para este escenario tiene 80 renglones y 3 columnas; esta información corresponde a los primeros 80 renglones de la matriz de entrada empleada en el entrenamiento.

La Fig. 4.7 muestra la salida del escenario 1, así como la salida estimada del mismo y su respectiva gráfica del error.

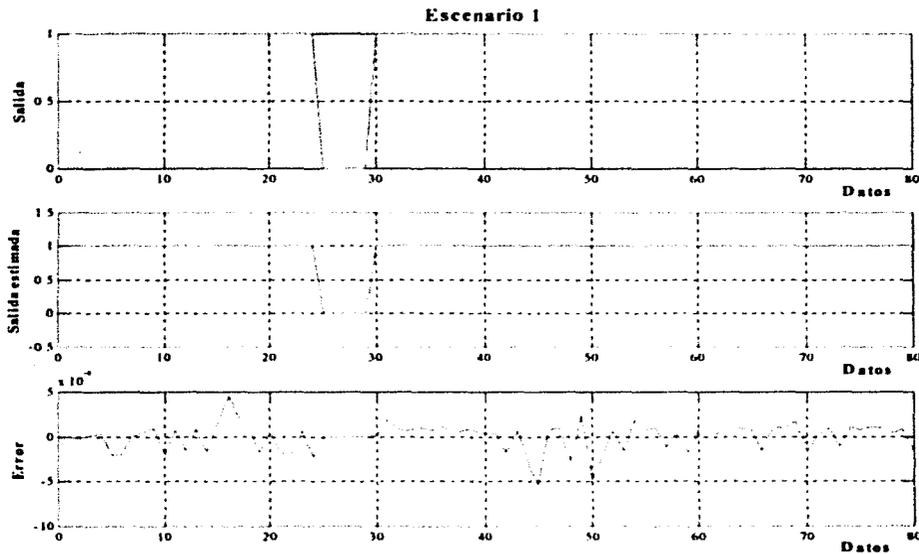


Fig. 4.7 Evaluación del escenario 1 sin fallas.

#### 4.4 Evaluación del escenario 2

Análogo al escenario 1, el escenario 2 está descrito por la ruta 2, y para evaluarlo, se toma la matriz de datos que se obtiene del robot al recorrer esta ruta y dichos datos se toman como la entrada a la red neuronal ya entrenada. Esta matriz contiene 92 renglones y 3 columnas y representa del renglón 81 al 172 de la matriz de datos que se utilizó para el entrenamiento.

En la Fig. 4.8 se puede observar la salida del escenario 2, la salida estimada por la red neuronal y el error.

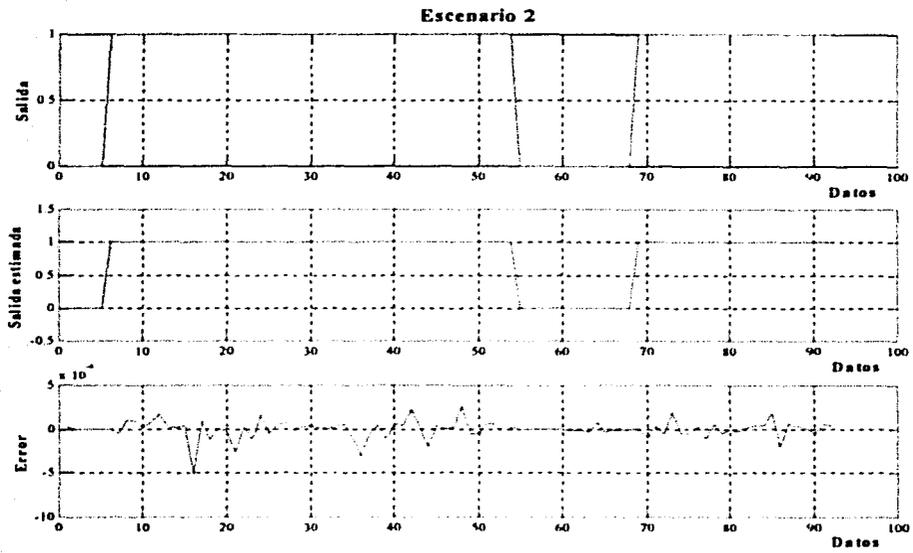


Fig. 4.8 Evaluación del escenario 2 sin fallas.

### 4.5 Evaluación del escenario 3

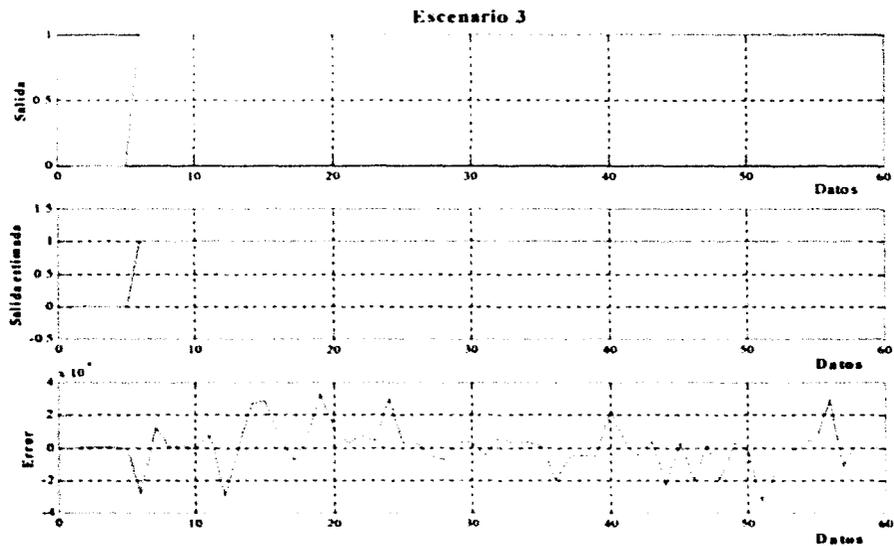


Fig. 4.9 Evaluación del escenario 3 sin fallas.

De la misma forma que se evaluaron los escenarios 1 y 2, de forma similar se evalúa el escenario 3. La matriz de datos de entrada para este escenario es de 58 renglones y 3 columnas y representa del renglón 173 al 230 de la matriz de datos de entrada que se utiliza en el entrenamiento.

La Fig. 4.9 muestra la salida del escenario 3, la salida estimada por la red neuronal y su respectivo error.

### 4.6 Evaluación de escenarios con fallas

Para evaluar el desempeño de la red neuronal bajo escenarios con fallas se toman datos del robot móvil cuando tiene algún error en su recorrido. Los errores más comunes que se presentan son desviaciones cuando avanza hacia adelante o la mala ejecución de un giro sobre su propio eje. Para cada ruta se toma un recorrido con fallas para evaluar a la red neuronal. Las fallas del robot móvil en su recorrido pueden deberse a diversas circunstancias, por ejemplo, una colocación errónea del robot en su punto inicial, una incorrecta alineación de la llanta delantera, entre otras.

Una vez que se obtienen los datos procedentes de cada escenario con fallas, se forman de nuevo matrices de datos de entrada para cada uno los escenarios y se normalizan tales datos antes de evaluar el desempeño de la red neuronal.

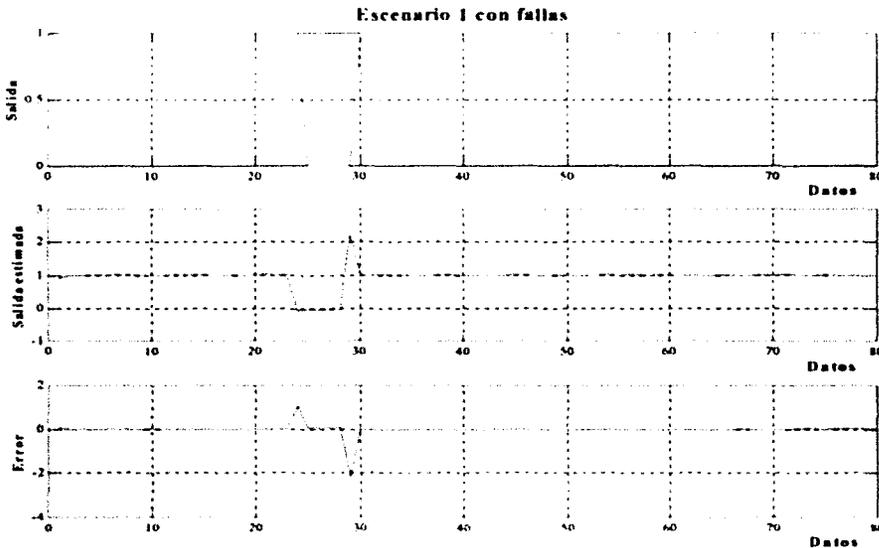


Fig. 4.10 Evaluación del escenario 1 con fallas.

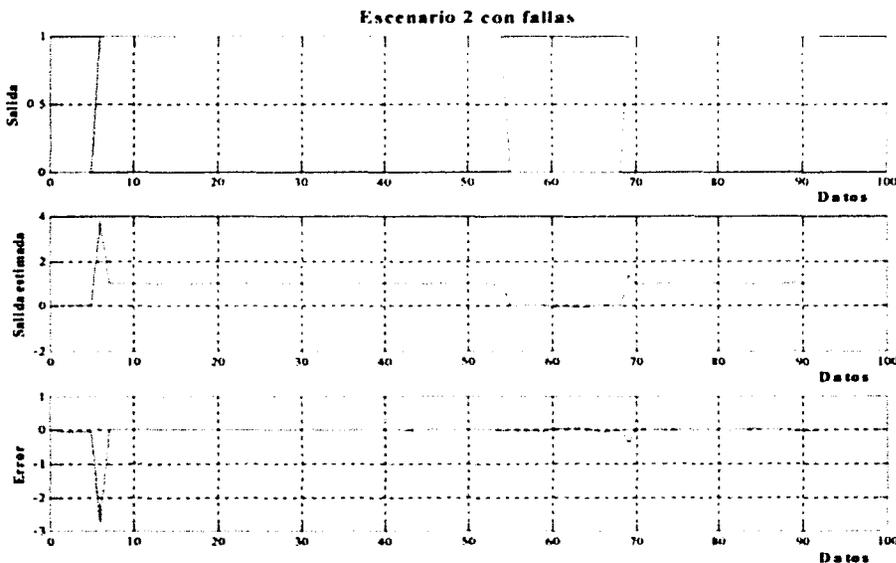
En la Fig. 4.10 se presenta el escenario 1 con fallas; se observa la salida, la salida estimada por la red neuronal y el error. En este ejemplo se presenta la falla durante el recorrido de la

ruta 1, inicialmente en el primer desplazamiento, el robot se desvía de su trayectoria alrededor de  $3^\circ$  hacia la derecha y después en el giro de  $90^\circ$  tuvo un error y giró más de  $90^\circ$ , lo que ocasiono que el robot no llegará al punto B.

La red neuronal sigue el comportamiento del robot aunque presenta un requerimiento mayor con la corrección de la salida, lo que genera un error puntual considerable.

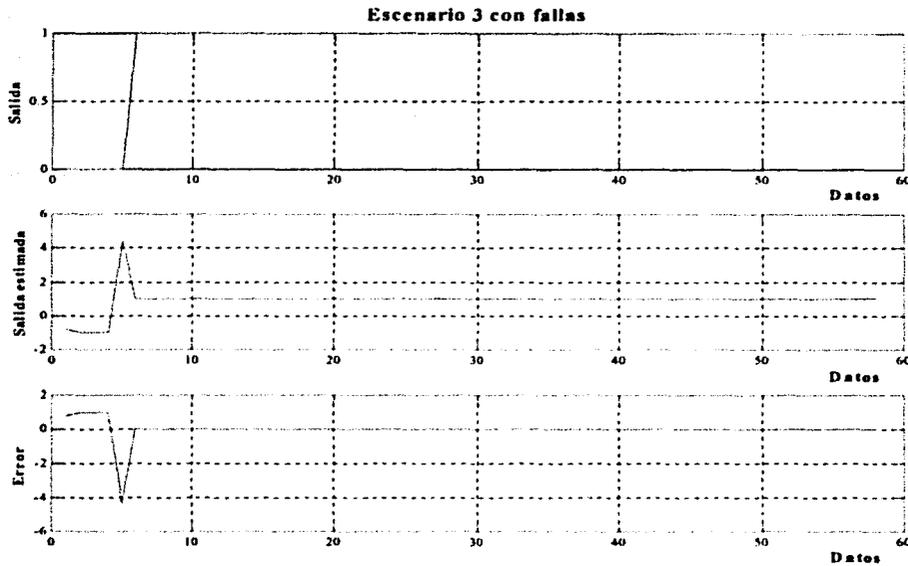
En la Fig. 4.11 se observan los resultados del escenario 2 al probar la red neuronal con datos de un recorrido con errores. Se observa la salida, la salida estimada por la red neuronal y el error. En este experimento se tiene la ruta con el mismo patrón de error que se presentó en el experimento anterior; cabe señalar que existe un giro inicial no presentado en el primer experimento; éste también es motivo para la propagación del error.

Al igual que en el caso anterior, la red sigue la respuesta del sistema sin grandes cambios. Sin embargo, esta presenta un incremento mayor al permitido en el error debido a la falla.



**Fig. 4.11 Evaluación del escenario 2 con fallas.**

En la Fig. 4.12 se muestran los resultados del escenario 3. Primero se presenta la salida, después la salida estimada por la red neuronal y al final el error. En este caso, el giro inicial para tomar la ruta 3 no es alcanzado debido a un error mayor a  $3^\circ$ . Como en los dos casos anteriores, la red determina un error puntual mayor al máximo valor normalizado, lo que sugiere un error debido a que el sistema no puede alcanzar físicamente dicha respuesta.



**Fig. 4.12 Evaluación del escenario 3 con fallas.**

## 4.7 Conclusiones

Al evaluar el entrenamiento de la red neuronal para una matriz de centros de 80 renglones, se observa que el error es menor al 0.1%, lo cual es suficiente para que el robot siga y ejecute una ruta de manera óptima. Cuando se incrementó el número de renglones de la matriz de centros durante el entrenamiento, se encontró que para una matriz de alrededor de 90 renglones, el entrenamiento generaba una matriz  $W$  próxima a ser una matriz singular, esto debido a que las últimas columnas son prácticamente idénticas entre sí, por lo cual, se optó por fijar la matriz de centros con 80 renglones.

Cuando se evaluó la red para los tres escenarios sin fallas, se observó que la red satisface, para cada ruta, que el error que se presenta sigue siendo menor al 0.1%, por lo que se concluye que los centros, las desviaciones estándar y los pesos que se obtuvieron durante el entrenamiento son adecuados para cada uno de los escenarios.

Para la evaluación con fallas se encontró que para los tres escenarios, la salida estimada por la red neuronal es muy similar a la salida, a excepción de los puntos en los que los datos del escenario con fallas se adelantan a los cambios de nivel y provocan un error considerable, el cual es corregido por la red un instante después. El error puntual que se presenta en los tres escenarios no es tan grave si consideramos que el robot no tiene la capacidad para alcanzar dicha respuesta; físicamente lo que sucedería sería es que el robot tomaría el mayor valor posible, lo cual no produciría un error muy grande.

# Capítulo 5

## Controlador PID

### 5.1 Introducción

El concepto de control es extraordinariamente amplio, abarcando desde un interruptor que gobierna el encendido de una lámpara hasta el más complejo ordenador de proceso del piloto automático de un avión.

En este capítulo se presenta el uso inicial de la combinación de la red neuronal con un controlador PID, con el objetivo de disminuir el error a un valor que no represente una saturación en el sistema físico.

Los resultados presentados aquí se basan en los datos obtenidos en los capítulos anteriores. Las correcciones obtenidas se presentan bajo un ambiente de simulación del sistema.

En la sección 5.2 se presenta una descripción de bloques de la aproximación elegida. En la sección 5.3 se determina la selección de las constantes necesarias para el buen funcionamiento del controlador. La sección 5.4 muestra la respuesta a esta aproximación. La sección 5.5 muestra la respuesta del sistema con base en la saturación debida al error sin el uso del controlador. Por último, se presentan una serie de conclusiones parciales a esta aproximación.

## 5.2 Diseño del controlador PID

El controlador PID propuesto se ilustra en la Fig. 5.1.

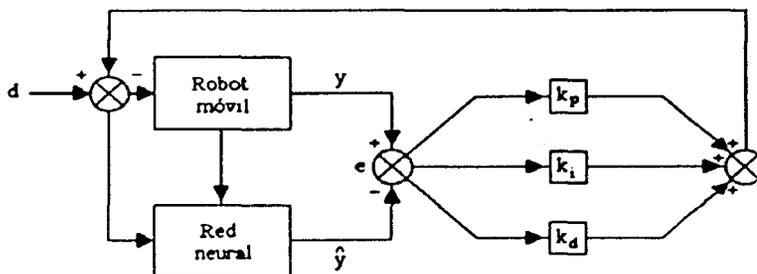


Fig. 5.1 Diagrama de bloques del controlador PID.

donde  $d$  es la señal de entrada al motor,  $y$  es la salida del motor,  $\hat{y}$  es la salida estimada,  $e$  es el error de la estimación,  $k_p$  es la constante de proporcionalidad,  $k_i$  es la constante de la integral del error y  $k_d$  es la constante de la derivada del error del controlador PID.

Para el cálculo de la integral numérica del error se utilizó la integral trapezoidal (interpolación de primer orden), la cual se define como

$$e_i = \frac{h}{2} \left( e_0 + e_n + 2 \sum_{k=1}^{n-1} e_k \right) \quad (5.1)$$

donde  $e_i$  es la integral numérica del error,  $e_0$  es el error inicial en la ventana de datos que se toma para el cálculo,  $e_n$  es el error final en la ventana de datos que se toma para el cálculo,  $e_k$  son los demás errores dentro de la misma ventana y  $h$  es el tiempo de muestreo, que en este caso es 0.33 segundos.

Y para la derivada numérica (interpolación de segundo orden), se empleó la siguiente ecuación

$$e_d = \frac{1}{2h} (e_0 - 4e_1 + 3e_2) \quad (5.2)$$

donde  $e_d$  es la derivada del error,  $e_0$  es el error inicial en la ventana de datos que se toma para el cálculo,  $e_1$  es el primer error en la ventana de datos que se toma para el cálculo y  $e_2$  es el segundo error en la misma ventana y  $h$  es el tiempo de muestreo, que en este caso es 0.33 segundos. Cabe hacer mención que para este cálculo, la ventana de tiempo toma en cuenta 3 datos.

En este caso, la red neuronal entrega la distancia estimada a recorrer la cual es comparada con la distancia recorrida. De esta comparación se obtiene el error. Éste sirve para calcular tanto la integral y la derivada necesarias para corregir el movimiento del robot.

### 5.3 Selección de las constantes del controlador PID

Las constantes se obtuvieron mediante prueba y error. Para cada escenario, las constantes son distintas. La información con la cual se estuvieron probando las constantes, fue la obtenida durante los recorridos de los escenarios con fallas.

Las constantes que se eligieron para el control PID de cada uno de las distintas rutas se enlistan en la tabla 5.1.

La variación de estas constantes es debido a la modificación de las rutas, por lo que la unificación se presenta como un caso de estudio interesante a resolver.

	Constante proporcional ( $k_p$ )	Constante integral ( $k_i$ )	Constante derivativa ( $k_d$ )
Escenario 1	- 0.009	+ 0.09	+ 0.00099
Escenario 2	- 0.020	+ 0.0055	- 0.002
Escenario 3	0	0	0

Tabla 5.1 Constantes del controlador PID.

### 5.4 Evaluación del controlador para los escenarios con fallas

En las Figs. 5.2, 5.3 y 5.4 se presentan las gráficas con las salidas calculadas por el controlador propuesto para cada uno de los distintos escenarios.

La Fig. 5.2 muestra la respuesta del sistema con la realimentación propuesta en la sección anterior; en este caso, se reduce el error en un 60%, con lo que la saturación no presentará una exigencia mayor que la potencia del sistema. Cabe señalar que el error ha sido disminuido, pero aún así presenta el problema de una disfunción en su respuesta real como se verá en la siguiente sección.

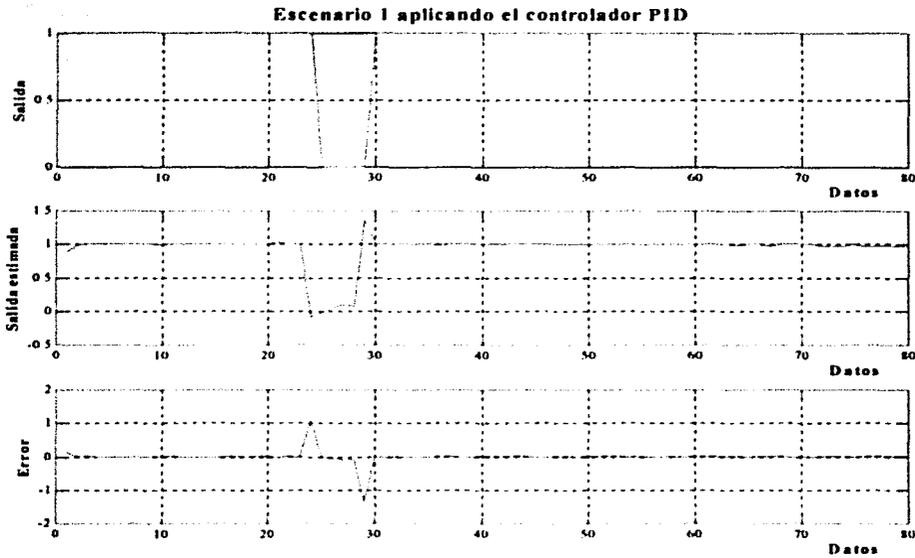


Fig. 5.2 Gráfica del escenario 1 con fallas aplicando el controlador PID.

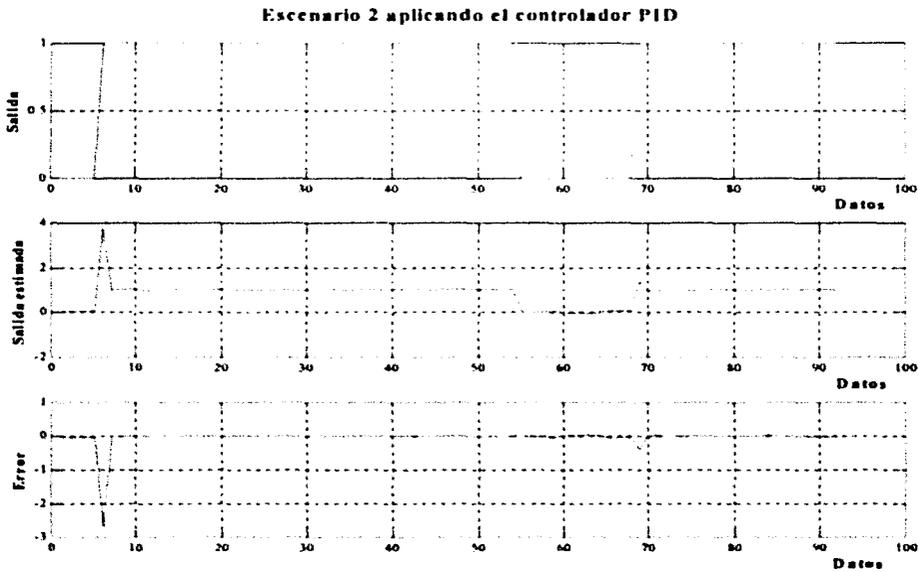
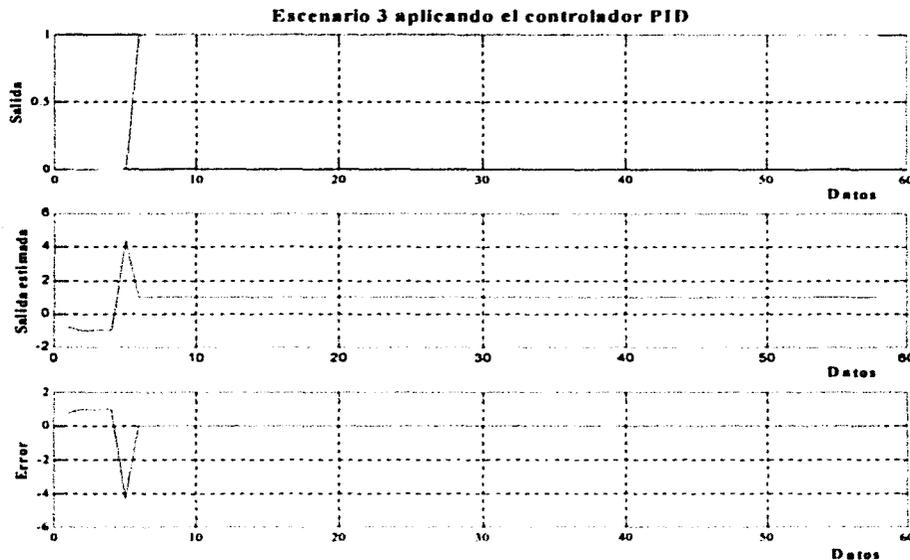


Fig. 5.3 Gráfica del escenario 2 con fallas aplicando el controlador PID.

La Fig. 5.3 muestra la respuesta del sistema con la realimentación propuesta. En este caso no hay un decremento sustancial del error.

Para el caso de la Fig. 5.4, el sistema no usa el controlador porque tiende a ser inestable con cualquier proporción.

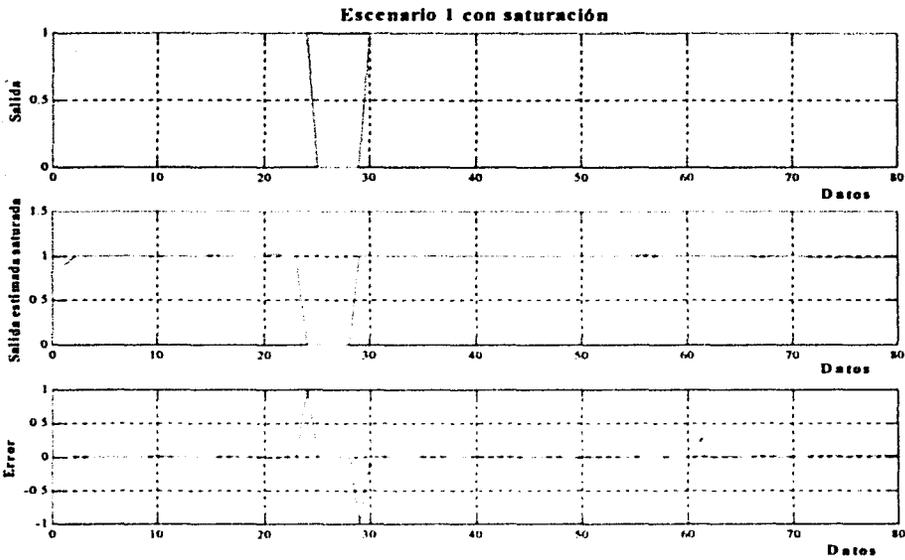


**Fig. 5.4** Gráfica del escenario 3 con fallas aplicando el controlador PID.

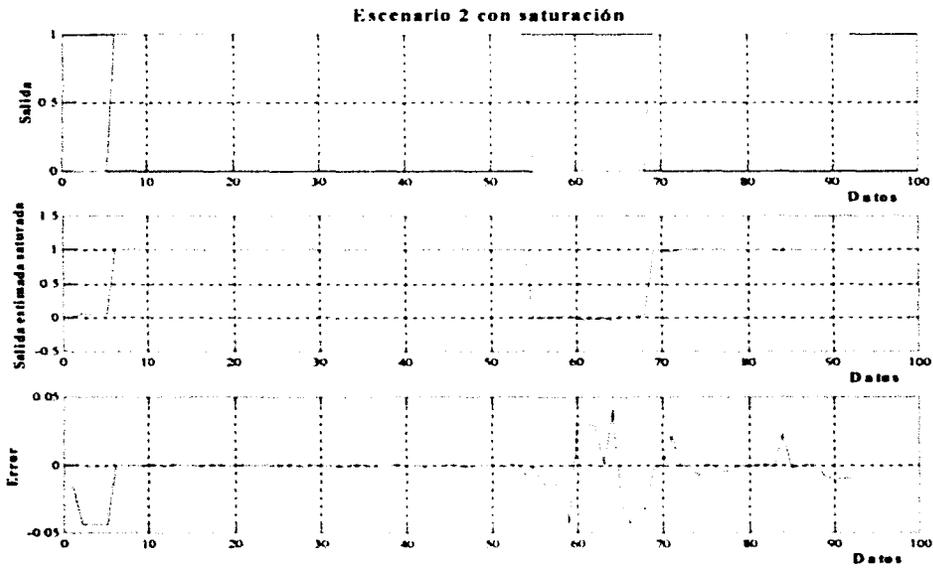
## 5.5 Respuesta del robot sin controlador

Ahora bien, si no se aplicara el controlador, la salida estimada por la red neuronal propone al sistema datos fuera del intervalo  $[0,1]$ , lo cual físicamente representaría que el robot móvil avance una mayor distancia cada 0.33 segundos de lo que en realidad puede avanzar. Así que el robot móvil, al pedirle dicha respuesta, únicamente podría responder con una distancia que al normalizarla sería un valor muy cercano a uno. Por lo tanto, lo que realmente podría hacer el robot con la salida estimada propuesta por la red neuronal sería lo mismo que si se limita la salida estimada por la red neuronal a un valor muy cercano a uno. Algo similar sucedería para valores menores a cero.

En las Figs. 5.5, 5.6 y 5.7 se muestra la salida estimada por la red neuronal limitada en el intervalo  $[-0.05, 1.05]$ . En este caso, todas las respuestas se mantienen en un intervalo menor al 10% del error. Esto es debido a que existe una saturación de intervalos en la amplitud.



**Fig. 5.5 Escenario 1 limitando la salida propuesta por la red neuronal.**



**Fig. 5.6 Escenario 2 limitando la salida propuesta por la red neuronal.**

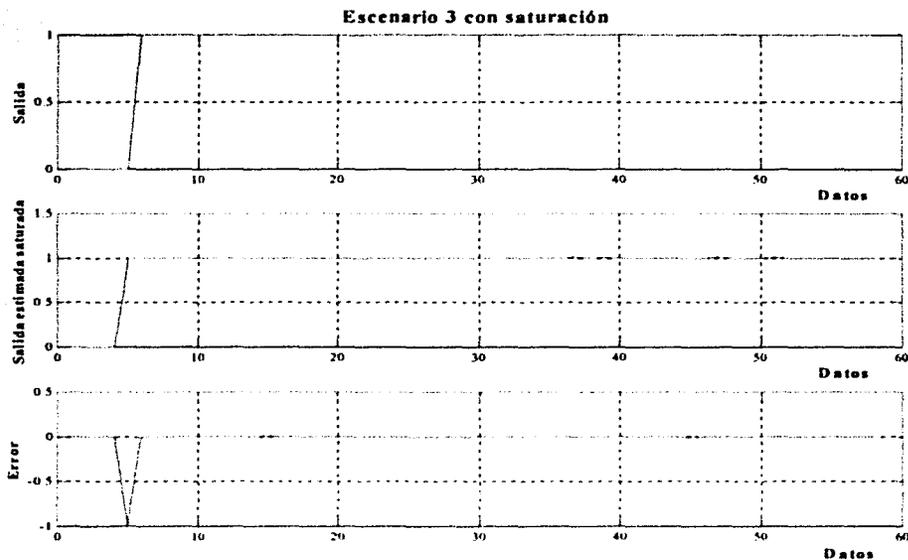


Fig. 5.7 Escenario 3 limitando la salida propuesta por la red neuronal.

## 5.6 Conclusiones

Este capítulo presenta la propuesta final de la corrección de errores a partir de la realimentación del error por medio del uso del controlador PID. Una vez definida la respuesta de la red neuronal con base en la presencia de errores se integra la ley de control tratando de eliminar su discrepancia actual.

Aún cuando esta última etapa se realizó con base en una simulación realizada en MATLAB (versión 6.0), los resultados presentados permiten establecer que el robot corregirá su movimiento de manera adecuada sin necesidad de establecer un número mayor de sensores. La propuesta presentada en esta tesis tiene la ventaja de su ligereza en términos de la actividad computacional.

Aún cuando la saturación de intervalos en amplitud presenta una mejor respuesta que la del controlador PID, no se puede asegurar su estabilidad global ni local, debido a que el sistema ahora presenta una respuesta no lineal en la cercanía de los límites de operación de la red. Se puede llevar a cabo el proceso de análisis formal de estabilidad con base en una función no lineal propuesta de Lyapunov, sin embargo, esto se encuentra fuera de los objetivos de esta tesis.

## Capítulo 6

### Conclusiones

Los modelos anteriores de robots móviles con los que se había trabajado en el Laboratorio de Interfaces Inteligentes (Facultad de Ingeniería, UNAM) ocupaban dos motores de corriente directa para la tracción y una rueda libre en la parte delantera. Dadas ciertas deficiencias en el diseño mecánico de los robots construidos bajo este esquema, se decidió desarrollar un modelo basado en el uso de lectores fotorreflectivos para el manejo del movimiento de las llantas traseras, así como la implantación de un motor de pasos de la llanta delantera.

El robot móvil que se construyó para el desarrollo de este proyecto supera una serie de problemas como la falta de precisión en el movimiento del robot. Dicho artefacto cuenta con dos motores de corriente directa para la tracción y con un motor de pasos para controlar la dirección. Con el uso del motor de pasos, la llanta delantera no presenta incertidumbres inherentes al movimiento correspondiente a una llanta delantera libre; además, se tiene un control preciso del giro de la llanta delantera con una resolución de  $0.9^\circ$ . Los motores de corriente directa cuentan con reductores de velocidad con una relación 132:1, por tanto, cada 132 revoluciones en la flecha del motor de corriente directa se tiene una revolución en la flecha del reductor, lo cual se aprovecha para colocar el interruptor óptico del lado de la flecha del motor de corriente directa, y con la hélice que se coloca sobre la flecha del motor se obtienen 264 señales por cada vuelta o revolución de la llanta. Esto es una ventaja considerable respecto a los codificadores que se utilizaban anteriormente, debido a que es muy complicado colocar 264 ranuras en dichos codificadores. Por lo tanto, el robot construido supera el problema de la llanta delantera, y además de contar con un mayor torque en sus motores de corriente directa, ahora se cuenta con más información del movimiento de los mismos.

Si bien es cierto que el motor de pasos para controlar la llanta delantera elimina los problemas de ruido que presentaba esta llanta de manera libre, también es cierto que el contar con un motor adicional hace más complejo el problema de control.

El robot móvil, al igual que los modelos diseñados anteriormente en el Laboratorio de Interfaces Inteligentes, está construido con base en acrílico, con la diferencia de que se utilizó cloruro de metileno para pegar el acrílico, lo cual facilitó tal construcción, y permitió diseñar un robot más robusto y se tuvo un mejor aprovechamiento del espacio.

Aunque los movimientos del robot móvil no se ven afectados por el peso de las baterías, es recomendable en trabajos futuros reemplazar dichas baterías por otras más ligeras.

Los modelos anteriores de robots móviles no contaban con detectores de nivel bajo de en las baterías, lo que ocasionaba que muchas veces se tuvieran errores ocasionados por ello y se confundieran con errores de programación. Aunque el uso de los detectores de voltaje evitó los errores por niveles bajos de voltaje, se debe trabajar aun más para mantener un nivel de voltaje constante en los motores de corriente directa, para reducir aún más los errores ocasionados por caídas en el voltaje.

La calibración de la llanta delantera y el cálculo de las ecuaciones que determinan tanto el giro del robot sobre su propio eje como el desplazamiento del robot hacia adelante resultan ser de gran importancia para que el robot móvil tenga un buen desempeño en su movimiento hacia adelante y en sus giros. Una mala calibración de la llanta delantera o una mala aproximación de las ecuaciones puede generar un error considerable en los desplazamientos del robot móvil; por lo tanto, es importante hacer una correcta calibración y obtener una buena aproximación de las ecuaciones para no tener que invertir tiempo de procesamiento en la corrección de dichos errores mediante la utilización de un controlador.

A pesar de que se obtuvo una buena aproximación de las ecuaciones, lo cual se puede observar en las Figs. 3.30 y 3.32, y una buena calibración en la llanta delantera, se debe trabajar aún más para obtener mejores resultados. Para la llanta delantera, sería adecuado reemplazar el tope que el motor de pasos utiliza para alinear la llanta delantera por un dispositivo electrónico, por ejemplo, un interruptor óptico. Esto eliminaría los errores que se presentan cuando el tornillo del collarín que se encuentra sobre la flecha del motor de pasos rebota al hacer contacto con el tope y no se queda en la posición de referencia. Además, se tendría la posibilidad de que el motor de pasos pudiera recorrer los  $360^\circ$ .

Las ecuaciones que se obtuvieron para aproximar el desplazamiento del robot y el giro de éste sobre su propio eje, se calcularon únicamente con la información del motor 1 por simplicidad. Para determinar la distancia que recorre el motor 2 se utilizaron las ecuaciones del motor uno, debido a que se asumió que ambos motores presentaban el mismo comportamiento, lo que no necesariamente es cierto; por lo tanto, para trabajos posteriores es importante determinar las ecuaciones del motor 2.

Los motores 1 y 2 no presentan el mismo comportamiento a pesar de que se encuentran colocados sobre un mismo eje (ver Fig. E.2). Para lograr que los dos motores desplacen el robot hacia adelante o hacia atrás, el motor 2 debe estar polarizado en sentido contrario al motor 1, y sólo cuando el robot gira sobre su propio eje, los motores se polarizan en el mismo sentido. Aunque los motores de corriente directa se consideren prácticamente idénticos, no presentan el mismo comportamiento en los dos sentidos de polarización; por lo tanto, cuando el robot avanza hacia adelante o hacia atrás tiende a desviarse hacia un costado, esto es sin tomar en cuenta la llanta delantera. Otra causa por la cual se puede desviar el robot móvil es por la forma en que están sujetas las flechas de cada motor de corriente directa con la flecha de las llantas (ver Fig. 3.4); si no se tiene el suficiente cuidado, o si las flechas de las llantas no empalman perfectamente con las flechas de los

motores, esto puede provocar que el movimiento de las llantas describa una forma ovalada o que el movimiento de las llantas no sea perpendicular al eje de los motores.

Una vez establecida la construcción del robot y el control local de los motores por medio de la rectificación de curvas con base en varios escenarios similares, se plantea la necesidad de tener un controlador global para el seguimiento de rutas. Dada la experiencia que se ha obtenido en distintos grupos de investigación con respecto al uso de sistemas inteligentes para dicho controlador, y debido a que el sistema es potencialmente lineal, se optó por utilizar una red basada en funciones de base radial para tal efecto. Aún cuando el sistema presenta una respuesta lineal, las desviaciones inherentes del robot debido a las no linealidades locales presentan la necesidad de recurrir a una estrategia de control capaz de definir una decisión segura para el sistema.

El uso de la red neuronal se hace buscando establecer una respuesta rápida con base en un movimiento no lineal. Con base en este esquema, se observó que la red tiene un error menor al 0.1% usando una matriz de centros de 80 renglones para su entrenamiento; también se observó que con un número mayor de renglones, la matriz  $W$  que se generaba estaba próxima a ser una matriz singular, lo cual ocasionaba no poder determinar un vector de pesos. Mientras más renglones tenga la matriz de centros produce que las últimas columnas de la matriz  $W$  se parezcan más entre sí; ésta es la razón por la que la matriz se aproxime a una matriz singular. Se presenta un resultado satisfactorio para los tres escenarios sin fallas ya que el error obtenido también es menor al 0.1%, concluyendo que los centros, las desviaciones estándar y los pesos obtenidos para cada ruta son adecuados.

Para los escenarios con fallas la red cumple satisfactoriamente, excepto para algunos datos donde los escenarios con fallas se adelantan a los cambios de nivel provocando un error puntual considerable, el cual es corregido por la red un instante después. El error puntual encontrado resulta no ser tan grave si se toma en cuenta que el robot no tiene la capacidad para alcanzar dicha respuesta; lo que sucedería sería que el robot se saturaría produciendo el mayor valor de respuesta posible, lo cual no produciría un error muy grande.

La evaluación de la red neuronal para los escenarios con fallas se basaron en una simulación realizada en MATLAB. La razón por la cual se simuló esta etapa es porque la comunicación entre el robot móvil y la computadora resultó no ser eficiente. El problema radica en el protocolo de los módulos de comunicación, el cual toma un tiempo muy grande para la transmisión y recepción de datos, comparado con el tiempo de muestreo (0.33 segundos), lo que dificulta tener un proceso corriendo en tiempo real debido a que la red neuronal se procesa en la computadora y no en el microcontrolador del robot móvil. La razón por la cual la red neuronal no se encuentra en el microcontrolador del robot móvil es por la complejidad computacional que requiere.

Para corregir los posibles errores puntuales que se presentan, se propuso la utilización de la realimentación del error por medio de un controlador PID. La utilización de esta etapa tiene la ventaja de su ligereza en términos de la actividad computacional, pero si se compara con el uso de la saturación de intervalos en amplitud, los resultados no son tan buenos, tal como muestran los resultados de la sección 5.5. El problema del uso de saturación es que no se puede asegurar la estabilidad cuando ésta se aplica al sistema real, aunque en términos de

simulación presenta una mejor respuesta. Por la misma razón que la evaluación de la red neuronal para los escenarios con fallas, también esta etapa se basó en una simulación realizada en MATLAB.

Para trabajos futuros, se propone mejorar la comunicación inalámbrica. Una posible solución para hacer más eficiente la comunicación es trabajar únicamente con la tarjeta de radio, para lo cual es necesario adaptar algunas señales y la alimentación, así como una base para colocar dicha tarjeta.

También es necesario mejorar o buscar distintas soluciones para los problemas que se presentaron en la alineación de la llanta delantera, del montaje de las llantas traseras y de las baterías del robot móvil.

Asimismo, se debe trabajar en una red mejorada e incluso pensar en una red no supervisada. Además, también debe explorarse el uso de controladores neuronales.

En trabajos posteriores se tiene contemplada la construcción de más robots móviles, con el objetivo de formar un sistema mucho más complejo, en el cual se puedan desarrollar algoritmos de sistemas cooperativos, detección de fallas, entre otras más.

# Apéndice A

## Módulo de comunicación

El módulo de comunicación empleado es un kit de desarrollo de la empresa RF Monolithics y el modelo es el *DR1007-DK Virtual Wire*® [Referencia electrónica RFM, 2001].

### A.1 Propósito del módulo de comunicación

El módulo de comunicación es una herramienta para evaluar la factibilidad o viabilidad de una aplicación de comunicación de datos en forma inalámbrica de baja potencia. El módulo también puede ser usado como una herramienta para facilitar el desarrollo de un sistema real, debido a la facilidad tanto en su manejo como en su instalación [Manual RFM, 1996].

### A.2 Descripción general

El módulo permite al usuario establecer comunicaciones de datos de baja potencia basadas en una transmisión de paquetes de la forma *half-duplex*. Dicho módulo contiene el hardware y el software requeridos para establecer una conexión inalámbrica entre dos computadoras que tengan el sistema operativo MSDOS con puertos seriales RS-232C. El módulo incluye dos nodos de comunicación; cada nodo consiste en una tarjeta de radio de datos y una tarjeta de protocolo. Este módulo opera a 916.5 MHz.

El módulo de comunicación incluye las siguientes características:

- Operación entre computadoras (*PC*'s) que cuenten con sistema operativo MSDOS.
- Radio transceptores (transmisor-receptor) de datos a 3 volts de corriente directa (VCD), a baja corriente y operando en la banda UHF (frecuencias ultra altas) a 916.5 MHz.
- Excelente rechazo a interferencia en el receptor fuera de canal.
- Antenas de referencia.

- Tarjetas de protocolo a 3 VCD (a baja corriente) basadas en el microcontrolador ATMEL<sup>10</sup> AT89C2051.
- Protocolo de paquetes de capa ligada (*link - layer*) con detección de errores ISO 3309 y retransmisión automática de paquetes; mensajes de hasta 32 bytes por paquete de transmisión (código ASCII o binario).
- Código de datos balanceado en corriente directa para un funcionamiento robusto de transmisión en radio frecuencia (RF).
- Diagnóstico mediante diodos emisores de luz (*LED's*) para una evaluación del funcionamiento del sistema.
- Hasta 15 diferentes nodos de direcciones, programables mediante puentes (*jumper's*).
- Operación con 2 baterías tipo AAA.

### A.3 Aplicaciones inalámbricas de baja potencia

Los sistemas inalámbricos de RF de baja potencia transmiten típicamente menos de 1 mW, y operan sobre distancias de 3 a 30 metros. Son diversos los usos que tienen este tipo de sistemas y cada vez emergen nuevas aplicaciones para comunicación inalámbrica de baja potencia.

Los usos clásicos para sistemas inalámbricos de baja potencia son controles remotos y alarmas, incluyendo puertas automáticas de cocheras, transmisores de autos para abrir las puertas sin llave y sistemas de seguridad para el hogar. Recientemente, se han desarrollado aplicaciones de comunicaciones de datos en forma bidireccional. Estos sistemas inalámbricos de baja potencia son utilizados para eliminar la molestia de los cables en todos los productos de tipo digital, tal como los teléfonos inalámbricos. Este módulo está diseñado para adecuarse al diseño de este tipo de aplicaciones. La mayoría de los sistemas inalámbricos de baja potencia operan con pocos problemas de interferencia; sin embargo, estos sistemas operan sobre un canal de radio común, por tanto, la interferencia puede ocurrir en cualquier lugar y a cualquier hora.

Las aplicaciones para comunicaciones de datos inalámbricos de baja potencia están creciendo rápidamente. La siguiente lista de ejemplos de aplicación muestra la diversidad de usos para tecnología inalámbrica de baja potencia:

- Lectores de código de barras inalámbricos y escritores de etiquetas de código de barras.
- Etiquetas identificadoras inteligentes para un seguimiento de inventario e identificación.
- Impresoras receptoras inalámbricas.

<sup>10</sup> ATMEL Corporation, [www.atmel.com](http://www.atmel.com)

- Seguimiento o localización (extensiones telefónicas "*follow-me*").
- Telemetría.
- Sensores para alarmas de seguridad inalámbricas.
- Etiquetas de control de acceso y autenticidad.
- Juegos de galería.

#### A.4 Características

Las tarjetas de radio pueden ser conectadas directamente al puerto serial de la computadora (usando un convertidor de voltaje de RS-232 a 3 volts CMOS); sin embargo, los resultados probablemente no sean tan satisfactorios. Primero, la detección de error está limitada al byte de paridad, el cual puede permitir muchos errores. También, el balance de corriente directa (CD) en los datos puede ser muy pobre, el cual generalmente reduce el rango de alcance de las tarjetas de radio.

Los módulos de comunicación contienen un microcontrolador, que a su vez es el encargado del protocolo. El protocolo es usado extensivamente de forma bidireccional. Este protocolo es útil, ya que provee un conjunto básico de características, incluyendo un medio efectivo para detección de errores en la transmisión y recepción de datos, así como un soporte de direccionamiento ("hacia" y "de") y permite enviar mensajes en forma de paquetes. Con esto se logra una comunicación de datos libre de errores al ser ejecutado en una vía automatizada. Además, el protocolo del microcontrolador usado en los módulos de comunicación provee una retransmisión automática del mensaje y una alarma en caso de que la comunicación no sea exitosa.

El rango de operación de los módulos de comunicación alcanza 60 metros en un lugar al aire libre y sin mucha interferencia eléctrica; sin embargo, el rango de operación en algunas situaciones está en función de los materiales empleados en la construcción del edificio que sea ocupado como área de trabajo y por otros sistemas de radio que operan en la vecindad, y el ruido generado por equipo cercano, tal como equipo de cómputo o teléfonos inalámbricos, entre otros.

En el caso de los módulos de comunicación, cada tarjeta de protocolo puede ser programada para tomar de 1 a 15 direcciones. La dirección "0" está reservada para mensajes de *broadcast* (transmisión a todos los nodos). Por ejemplo, el nodo 1 puede estar transmitiendo la lectura de un código de barras al nodo 2 mientras que el nodo 4 está transmitiendo otro código de barras al nodo 7 en el mismo lugar.

## A.5 Protocolo de comunicación

Casi todas las comunicaciones inalámbricas bidireccionales de datos emplean alguna forma de protocolo de comunicación para asegurar automáticamente que la información sea recibida correctamente en el destino seleccionado. El protocolo seleccionado para los módulos de comunicación es un protocolo de capas ligadas que incluye las siguientes características:

- Cálculo de la detección de errores ISO 3309 de dieciséis bits, para probar la integridad del mensaje.
- Encabezado de dirección "hacia"/"de" (*TO/FROM*) de ocho bits con 15 posibles nodos de dirección.
- Soporte de mensajes en código binario o ASCII de hasta 32 bytes por paquete.
- Retransmisión automática del paquete recibido y una alarma de falla en caso de no completarse la comunicación.

El protocolo de software no requiere de un hardware específico para controlar el flujo de datos; por lo tanto, si se requiere realizar una interfaz diferente, deben hacerse modificaciones en el protocolo de software. En la implementación de esta nueva interfaz debe tomarse en cuenta los paquetes de información transmitidos. Estos deben ser pequeños mensajes de 32 bytes, debido a que en los sistemas de comunicación inalámbrica se utilizan deliberadamente paquetes pequeños de información para mejorar su funcionamiento, evitando la interferencia en los canales de transmisión.

## A.6 Consideraciones de la antena

Para el buen funcionamiento de una aplicación empleando los módulos de comunicación, las antenas juegan un papel importante. En lo posible, la antena debe estar situada fuera del equipo y también, se debe tratar de colocar la antena en la parte más alta del equipo. Tampoco debe ser colocada dentro de una caja de metal, si es que el equipo está blindado, ni debe estar dentro de una caja plástica, debido a que atenúa significativamente las señales de radio.

## A.7 Manejo del ruido interno

Los transeptores son sensibles al ruido en la banda de paso del receptor, porque las señales transmitidas están restringidas a niveles de baja potencia. Comúnmente se encuentran fuentes de ruido en microprocesadores, escobillas de motores y circuitos lógicos de alta velocidad.

Las siguientes recomendaciones deben ser consideradas para aplicaciones que tengan una o más fuentes de ruido incluidas.

- Colocar el transceptor de RF y su antena tan lejos como sea posible de la fuente de ruido.
- Si el transceptor de RF debe estar encerrado con una fuente de ruido, colocar la antena remotamente usando un cable coaxial.
- Si es posible, encerrar las fuentes de ruido en cajas metálicas aterrizadas y usar desacopladores en las líneas de entrada y salida.

### A.8 Tarjeta de protocolo

*Programación del nodo.* La dirección del nodo en cada tarjeta de protocolo puede ser fijada de 1 a 15 mediante la colocación de puentes en los postes que se localizan entre los dos circuitos integrados de la tarjeta de protocolo. Sin puentes, el nodo de dirección es el 1. Colocando un puente entre el par de postes más cercanos al conector RS-232 se programa la dirección 2. La regla es que el número de nodo es el valor binario del puente más 1; con el puente colocado más cerca al conector RS-232 se tiene la posición menos significativa. El nodo 0 está reservado para paquetes de *broadcast* hacia todos los nodos. En la Fig. A.1 se puede observar una tabla con la posición de los puentes para la programación de los nodos de dirección.

Posiciones de los postes para la programación de los nodos  
 No. de nodo = No. binario + 1

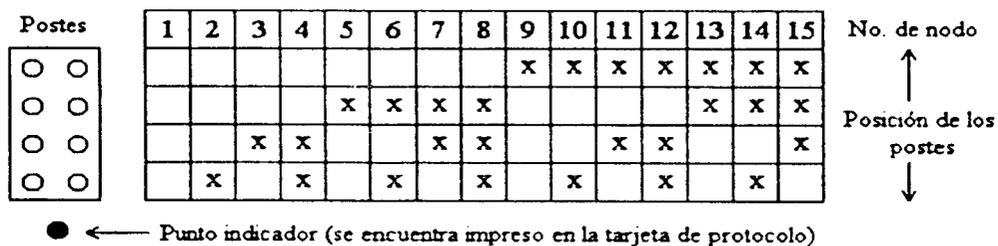


Fig. A.1 Posición de los postes para la programación de los nodos [Manual RFM, 1996].

*Fuente de alimentación.* Cada nodo puede operar con dos baterías tipo AAA de 1.5 volts.

*Interfaz RS-232.* La conversión de niveles de 3 volts CMOS a niveles RS-232 está provista por el circuito integrado MAX218. Es posible remover este circuito integrado y colocar un

puenste de la línea 7 a la 14 y otro de la línea 9 a 12 para una operación directamente con 3 volts CMOS.

*Funciones de los LED's.* Tres LED's indicadores están colocados en la tarjeta de protocolo. El LED con la etiqueta RXI indica que señales de radio frecuencia están siendo recibidas. El LED RF RCV indica que un paquete de radio frecuencia valido está siendo recibido. El LED PC RCV indica que un mensaje proveniente de la PC ha sido recibido.

### A.9 Teoría de operación

En esta sección se presenta el funcionamiento de cada una de las tarjetas que constituyen el módulo de comunicación, así como los elementos que las integran.

#### A.9.1 Tarjeta de radio para datos

*Interfaz de entrada/salida.* Referirse al diagrama esquemático de la tarjeta de radio de datos de la Fig. A.2. El conector P1 es el conector de interfaz hacia la tarjeta de protocolo. La línea 1 es la entrada de transmisión de datos y puede ser manejada directamente por una compuerta CMOS. El transmisor está modulando en *on-off key (OOK)* para una señal en esta línea, cambiando de 0 a 3 volts. Un nivel alto enciende el oscilador del transmisor y un nivel bajo lo apaga. La impedancia de entrada de esta línea es de aproximadamente 18 kΩ. La línea 2 es la línea de alimentación (Vcc) para el transmisor (TX).

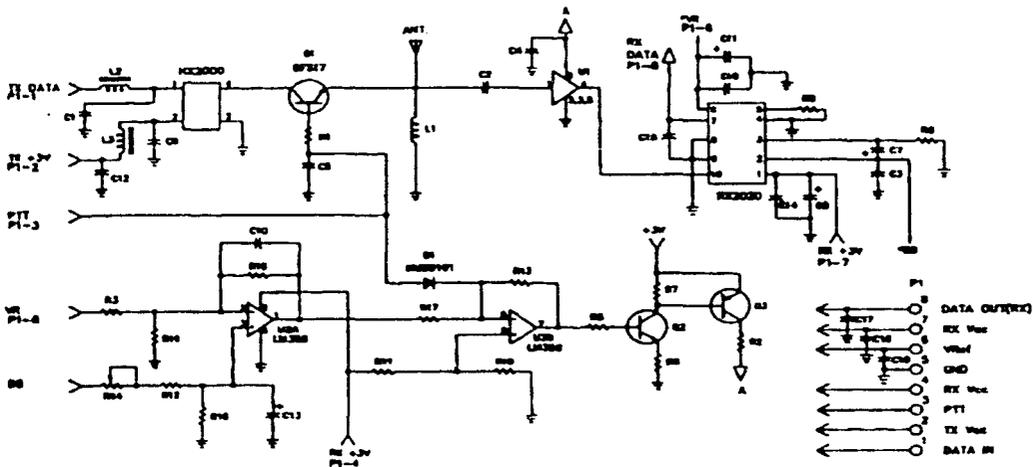


Fig. A.2 Diagrama esquemático de la tarjeta de radio de datos [Manual RFM, 1996].

La línea 3 es la línea llamada PTT que habilita el modo de transmisión. Esta línea habilita el interruptor de radio frecuencia de transmisión-recepción, la cual está en modo de

transmisión cuando el voltaje es alto (2.5 volts mínimo a 2 mA máximo). La línea 4 proporciona la energía (de 2.7 a 3.3 volts) a la circuitería del receptor (RX). La línea 5 es tierra. La línea 6 es la salida de voltaje de referencia ( $V_{ref}$ ) proveniente del receptor híbrido que es usado en la detección de procesos de batería baja en la tarjeta de protocolo. La línea 7 alimenta al receptor híbrido (de 2.7 a 3.3 volts). La línea 8 es la salida de datos del comparador en el receptor híbrido. Esta salida de datos es compatible con niveles CMOS y es capaz de manejar una sola compuerta CMOS o un transistor bipolar con una resistencia de base de 51 k $\Omega$ . La última conexión de la tarjeta de radio de datos es la entrada de la antena de 50  $\Omega$ . Como se mencionó con anterioridad, la antena puede ser conectada directamente a la tarjeta o conectarse remotamente usando un cable coaxial de 50  $\Omega$ .

*Transmisor.* El transmisor de radio frecuencia es un híbrido (HX) con montaje de superficie. La línea 1 del dispositivo HX es la entrada de transmisión de datos previamente discutida. La línea 2 es la conexión del voltaje de transmisión. El dispositivo HX consume un máximo de corriente de 10 mA con un voltaje de alimentación ( $V_{cc}$ ) de 3 volts. La línea 3 del dispositivo HX es tierra y la línea 4 es la salida de radio frecuencia. La potencia de la salida de radio frecuencia del HX es nominalmente de 0 dBm (relación logarítmica entre la potencia estudiada respecto a una potencia arbitraria de 1 mW) con una carga de 50  $\Omega$ . La energía de transmisión es suministrada al puerto de la antena a través del interruptor de transmisión-recepción Q1. Cuando la línea de PTT es puesta en un nivel lógico "1", Q1 es encendido y conecta el transmisor a la antena.

*Receptor.* El receptor es un receptor amplificador secuencial híbrido (ASH) RX2020. El voltaje de alimentación ( $V_{cc}$ ) es aplicado a la línea 1 del receptor proveniente de la línea 7 del conector P1. Un capacitor de 10  $\mu$ F (C8) y un capacitor de paso de radio frecuencia de 27 pF (C14) también están conectados a esta línea. C8 es necesario para mantener al comparador interno del receptor sin ruido debido al encendido y apagado del amplificador de banda base en el receptor.

La línea 2 es la salida de banda base. La señal detectada en la línea 2 es filtrada antes de ser aplicada a la entrada del comparador. Esta misma línea es conectada a la entrada del comparador (línea 3) por medio del capacitor de acoplamiento C7 y también por medio de un capacitor de paso de radio frecuencia C3. El valor del capacitor de acoplamiento C7 es determinado por el ancho del pulso más largo que será encontrado en la cadena de datos. El acoplamiento capacitivo es usado por dos razones, la primera razón es para prevenir voltajes de corriente directa en la salida del amplificador de banda base, provenientes de falsos cambios en el comparador y la segunda es para prevenir la salida del detector (producida por una señal de interferencia de frecuencia modulada) del cambio del comparador, mientras se permiten cambios en el nivel de corriente directa (debidos a señales no deseadas) en la entrada del comparador. La resistencia R8 está conectada entre la entrada del comparador y tierra, y su función será explicada posteriormente.

La línea 4 es la tierra de corriente directa. La línea 5 es el comparador de umbral. Si esta línea está alta, el voltaje de umbral para el comparador es de 25 mV, y si está aterrizada el voltaje de umbral es 0 volts. El nivel de umbral interno de 25 mV es muy deseable para usos en el rango de 300 a 450 MHz debido a la alta sensibilidad del receptor y al alto

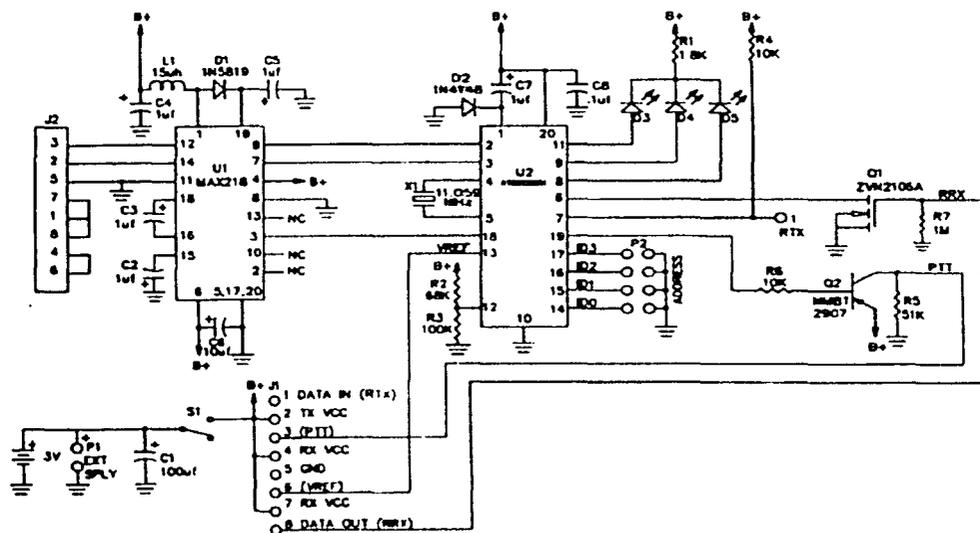
promedio de ruido de radio frecuencia en esta banda; sin embargo, se requiere un voltaje de umbral bajo para obtener un mejor funcionamiento en receptores que operan a 916.5 MHz. Así, la línea 5 está aterrizada para receptores de 916.5 MHz y habilitada para receptores de frecuencias menores. Los espurios de ruido a la salida del comparador de un receptor a 916.5 MHz se minimizan utilizando una resistencia R8 de la línea 3 a tierra. Esta resistencia reduce efectivamente el nivel de voltaje de corriente directa en la entrada del comparador el cual es equivalente a usar un nivel de umbral muy bajo. La línea 6 es la salida de voltaje de referencia de la fuente de alimentación incluida en el circuito integrado usado en el receptor. Esta línea debe ser conectada por un capacitor de 1  $\mu\text{F}$  (C11), el cual elimina el ruido por encendido y apagado en el amplificador de banda base. La línea 6 también está conectada en radio frecuencia por medio del capacitor C16. La línea 7 es la salida del comparador. El comparador es capaz de manejar la entrada de una sola compuerta CMOS. Las líneas 8 y 9 son primordialmente tierras de radio frecuencia. La línea 10 es el puerto de entrada de radio frecuencia del dispositivo RX. Este puerto es manejado por una fuente con impedancia de salida de 50  $\Omega$ .

### Especificaciones de la tarjeta de radio de datos

<b>Frecuencia de operación</b>	916.5 MHz
<b>Modulación</b>	<i>On Off Keyed (OOK)</i>
<b>Antena</b>	50 $\Omega$
<b>Tasa máxima de datos</b>	11.5 kbps (ancho mínimo de pulso de 87 $\mu\text{s}$ para la entrada TX)
<b>Tolerancia de frecuencia TX</b>	Menor que $\pm 200$ kHz
<b>Potencia de salida TX</b>	-5 dBm mínimo
<b>Armónicos TX</b>	Menores que -32 dBc (relación logarítmica entre la potencia estudiada respecto a la potencia de la portadora principal)
<b>Desempeño en la recepción</b>	BER (proporción de error en bits) menor que $10\text{E}-4$ para una entrada de -90 dBm
<b>Distorsión de pulso RX</b>	Menor que $\pm 35\%$ para un pulso TX de 87 $\mu\text{s}$
<b>Rango dinámico RX</b>	De -90 a -10 dBm
<b>Rechazo fuera de canal RX</b>	Mayor que 60 dB
<b>Rechazo en el canal RX</b>	Menor que el 30%
<b>Salida sin señal RX</b>	Menor que una espiga de ruido en un intervalo de ruido de 10 ms bajo condiciones de recepción de ruido blanco
<b>Tiempos de conmutación del tranceptor</b>	
• De RX a TX	1 ms
• De TX a RX	5 ms
<b>Fuente de alimentación de corriente directa</b>	De 2.7 a 3.5 volts de corriente directa, con un rizo de 10 mV pico a pico
<b>Fuente de corriente, modo RX</b>	Menor que 11.5 mA promedio para una fuente de 3 volts de corriente directa
<b>Fuente de corriente, modo TX</b>	Menor que 20 mA pico para una fuente de 3 volts de corriente directa
<b>Interfaz de datos de entrada y salida</b>	Un nivel lógico de 3 volts CMOS para una entrada serial TX; salida serial RX capaz de manejar una compuerta de 3 volts CMOS
<b>Entrada de control TX-RX</b>	Baja para RX, alta para TX (fuente de 2 mA para 2.5 volts mínimo)
<b>Rango de temperatura de operación</b>	De -40° C a 85° C

### A.9.2 Tarjeta de protocolo

**Interfaz de entrada-salida.** El conector J1 es la interfaz de entrada y salida entre la tarjeta de protocolo y la tarjeta de radio de datos, ver Fig. A.3. La línea 1 del conector J1 toma la cadena de datos transmitidos de la línea 7 del circuito U2 hacia la entrada RTX en la tarjeta de radio de datos. La línea 2 de J1 proporciona la alimentación del transmisor híbrido en la tarjeta de radio de datos. La línea 3 de J1 provee la señal de habilitación del transmisor (PTT) proveniente del transistor PNP Q2. La tarjeta de radio de datos requiere 2 mA a 2.5 volts en la entrada de PTT para habilitar el modo de transmisión. La línea 7 de J1 proporciona la energía al receptor híbrido. La línea 5 de J1 es tierra. La línea 6 de J1 es la entrada de voltaje de referencia (Vref) del receptor híbrido hacia la tarjeta de protocolo que es usado en el proceso de detección de batería baja. La línea 4 de J1 alimenta la circuitería de recepción. La línea 8 de J1 toma la señal de salida del receptor (RRX) de la tarjeta de radio de datos. Q1 proporciona un buffer de entrada de alta impedancia entre esta señal y la entrada hacia el circuito integrado U2.



**Fig. A.3 Diagrama esquemático de la tarjeta de protocolo [Manual RFM, 1996].**

**Interfaz RS-232.** El conector J2 es la interfaz RS-232 en la tarjeta de protocolo. Este conector hembra de nueve líneas está configurado para parecer como un equipo de comunicación de datos (DCE). La tarjeta de protocolo corre un software de control, así solamente en la línea 2 y en la línea 3 del conector J2 se tienen señales activas. La línea 2 del conector J2 (RD) envía datos hacia la PC y la línea 3 del conector J2 recibe datos provenientes de la PC (TD). Las líneas 4 y 6 del conector J2 están conectadas entre sí (DTR y DSR), y las líneas 1, 7 y 8 del conector J2 están conectadas entre sí (CD, RQS, CTS) y la línea 5 del conector J2 es tierra.

**Protocolo del microcontrolador.** El protocolo de capa ligada está implementado en el microcontrolador ATMEL AT89C2051, U2. El microcontrolador de 8 bits opera con un cristal de cuarzo de 11.059 MHz. El microcontrolador incluye 2 kbytes de memoria flash EPROM y 128 bytes de memoria RAM. El microcontrolador también incluye dos temporizadores de 16 bits y un puerto serie. Las entradas del microcontrolador incluyen los nodos de programación, líneas de ID0 a ID3 en las líneas 14, 15, 16 y 17, respectivamente; la recepción de datos (RRX) en la línea 6, la entrada de nivel CMOS proveniente de la PC en la línea 2 y la entrada de referencia de voltaje (Vref) en la línea 13. Las salidas del microcontrolador incluyen la transmisión de datos en la línea 7, la salida de datos hacia la PC en la línea 3, la señal de habilitación de transmisión en la línea 19, la señal de control del transceptor RS-232 en la línea 18 y la salida hacia los LED's, en la línea 8 la de RXI, en la línea 9 la de RF RCV y en la 11 la de PC RCV. El diodo D2 y el capacitor C7 forman el circuito de reinicio para el microcontrolador.

**Convertidor de nivel CMOS-RS-232.** La conversión de y hacia el RS-232 y niveles lógicos de 3 volts CMOS es realizada por el circuito MAX218 de la empresa Maxim<sup>11</sup> (U1), que es un transceptor dual RS-232. La operación del MAX218 está controlada por el microcontrolador U2, para minimizar el consumo de corriente promedio. L1, D1 y C5 operan en conjunto con la fuente de alimentación para generar  $\pm 6.5$  volts para las conversiones de transmisión y recepción. La línea 3 del MAX218 controla el modo de conmutación con la fuente mediante la línea 18 del circuito U2. La señal de entrada serial RS-232 de la línea 3 del conector J2 es conectada en la línea 12 del circuito U1 y es convertida en un nivel de 3 volts CMOS y la salida en la línea 9 del circuito U1. La señal de la salida serial CMOS de la línea 2 del circuito U2 es introducida en la línea 9 del circuito U1 y convertida en una salida RS-232 en la línea 12 del circuito U1. Esta señal se encuentra en la línea 3 del conector J2. La conversión RS-232 puede ser evitada para una operación directa con nivel CMOS, quitando el circuito U1 de su base y colocando un puente en la base entre las líneas 7 y 14 y un segundo puente en la base entre las líneas 9 y 12.

### Especificaciones de la tarjeta de protocolo

<b>Interfaz terminal</b>	Equipo de comunicación de datos RS-232 compatible con un conector hembra de 9 líneas, 19.2 kbps, bytes asincrónicos, 1 bit de inicio, 8 bits de datos, sin paridad, 1 bit de parada
<b>Interfaz de radio</b>	Interfaz de radio de datos RFM tipo1, conector SIP de 8 líneas, 11.5 kbps, 12 bits balanceados en corriente directa por byte con control integrado PTT
<b>Fuente de alimentación</b>	Nominalmente 3 volts de corriente directa de dos baterías tipo AAA o de una fuente regulada externa con un rizo menor a 10 mV pico a pico con un pico de corriente de 40 mA, con detección de batería baja
<b>Rango de temperatura de operación</b>	De 0°C a 70°C
<b>Rango de temperatura de almacenamiento</b>	De -40°C a 85°C

<sup>11</sup> Maxim Integrated Products, [www.maxim-ic.com](http://www.maxim-ic.com)

## A.10 Protocolo firmware

*Descripción.* El propósito de este protocolo de liga de datos es proveer una transmisión automática, verificada y libre de errores, de mensajes entre los nodos de los radios vía conexiones seriales RS-232 hacia la PC. La operación tanto en el lado del RS-232 como en el lado del radio es de forma *half-duplex*.

La operación de la conexión serial RS-232 es a 19.2 kbps, con ocho bits de datos (1 byte), un bit de parada y sin bit de paridad. La tasa de transmisión en el lado del radio es aproximadamente 11.5 kbps, usando 12 bits balanceados en corriente directa por cada byte de datos.

Las siguientes líneas de entrada y salida están diseñadas en el protocolo del microcontrolador:

- Línea de recepción del radio (RRX)
- Línea de transmisión del radio (RTX)
- Línea de control de transmisión-recepción del radio, alta en transmisión (PTT)
- Línea de recepción RS-232 (PRX)
- Línea de transmisión RS-232 (PTX)
- Línea de control de encendido-apagado del MAX218
- Línea de entrada de los nodos ID (de ID0 hasta ID3)
- Tres líneas de control de LED's (RXI, RF RCV y PC RCV)

El protocolo de capa ligada es capaz de transmitir y recibir bytes de datos binarios de cualquier patrón de bits. Los mensajes son enviados y recibidos desde la interfaz RS-232 en un formato estándar asíncrono vía PTX y PRX. El primer byte del mensaje proveniente del RS-232 debe contener la dirección "hacia"/"de", con la parte alta del byte (los primeros cuatro bits) el encabezado "hacia" y la parte baja del byte (los últimos cuatro bits) el encabezado "de". El segundo byte es el número de secuencia de mensaje (de 1 a 7 reciclándose y el 8 usado para paquetes de telemetría), el tercer byte es el número de bytes en el mensaje (hasta 20 hexadecimal) seguido por los bytes de datos. Un mensaje simple puede contener hasta 32 bytes de datos con 16 y 24 bytes de datos típicamente.

El software del protocolo continuamente verifica la línea RRX y la línea PRX buscando el bit de inicio. Cuando el bit de inicio es detectado en una de las líneas de entrada, el software intentará recibir un mensaje en la línea de entrada. Si un error es detectado en el mensaje, éste será descartado y el software reanudará la verificación de las líneas de entrada.

Si un mensaje válido es recibido en la línea de entrada PRX, el software construirá un paquete y lo pondrá en la cola de los paquetes para transmisión. El formato del paquete

incluirá un símbolo de inicio (55 hexadecimal), el byte “hacia”/”de”, el número de mensaje, el número de bytes de datos en el mensaje, los bytes de datos y 16 bits de detección de error (“secuencia de verificación de cadena”). La secuencia de verificación de cadena se calcula basándose en todos los bits en el mensaje seguidos por el símbolo de inicio. La secuencia de verificación de cadena empleada está definida por la ISO 3309.

El paquete de la cola es transmitido (la línea RTX con PTT invocada) y entonces el software busca un reconocimiento de un “paquete recibido” (línea RRX). El reconocimiento de un paquete recibido incluye el bit de inicio, el byte “hacia”/”de”, el número de paquete que está siendo reconocido, el dato hexadecimal “en” (donde n es el número de reintentos necesarios para recibir correctamente el mensaje) y la secuencia de verificación de cadena. Cuando un reconocimiento es recibido por el paquete en cola, un mensaje de reconocimiento es enviado por la línea PTX, el paquete es descartado y el software reanuda la verificación de las líneas de entrada. Si el paquete de reconocimiento no es recibido en 120 ms, el paquete es reenviado después de un retardo seleccionado aleatoriamente de 0, 120, 240 o 360 ms (aproximadamente). Si el paquete no es reconocido después de un total de ocho intentos, el software enviará un mensaje de que no fue reconocido en la línea de PTX (dirección “hacia”/”de”, número de paquete y el dato hexadecimal “DD”), el paquete se descarta y se reanuda la verificación de las líneas de entrada.

Cuando un símbolo de inicio es detectado en la línea RRX, el software intentará recibirlo y verificar el mensaje para reconocer la dirección correcta “hacia”/”de”, un número de secuencia de paquete válido, un número de bytes de datos válidos y un cálculo correcto de la secuencia de verificación de cadena. Si el paquete es verificado y el nodo “hacia” es correcto, la dirección “hacia”/”de”, el número de secuencia de paquete, el número de bytes de datos y los bytes de datos del mensaje serán enviados por la línea PTX y un paquete de reconocimiento será transmitido de regreso en la línea RTX. De otra forma, el mensaje es descartado y la verificación de las líneas de entrada se reanuda. El software aceptará paquetes de mensaje y paquetes de reconocimiento en cualquier secuencia. Si un paquete de reconocimiento es recibido por segunda vez (basado en un valor actual del contador de secuencia de mensaje) esto es reconocido por el RTX pero no será retransmitido en PTX.

La dirección “hacia”/”de” 00<sub>h</sub> será interpretada como un mensaje de *broadcast*. En este caso, el paquete recibido es enviado por la línea PTX si el número de bytes de datos está en un rango válido y el cálculo de la secuencia de verificación de cadena se iguala. En el modo *broadcast* el paquete es transmitido ocho veces para aumentar la probabilidad en la recepción. Un paquete de *broadcast* no es reconocido por el (los) nodo(s) de recepción.

Un paquete con número de paquete 8 es tratado como un paquete de telemetría. Si la dirección “hacia” se iguala con el número de nodo local y el número de bytes y la secuencia de verificación de cadena es válida, se colocará en la línea PTX.

Si el bit de inicio es detectado en RRX o en PRX, el software recibe y actúa sobre la información de esa línea de entrada y no toma en cuenta la otra línea de entrada hasta que se recibe y se actúa sobre el dato proveniente de la primera línea de entrada; el software de la PC debe generar un mensaje de transmisión sencillo para adecuar esta característica.

Cuando el software de la PC está listo para enviar un mensaje hacia el software del protocolo, éste verifica la disponibilidad de la interrupción PRX enviando solamente el byte de la dirección "hacia"/"de" hacia la tarjeta del protocolo. Si la dirección "hacia"/"de" es regresada en un intervalo de 50 ms, el software de la PC tiene el control del proceso de interrupción PRX y puede enviar el resto del mensaje en los siguientes 200 ms. Si el byte que llega tiene los cuatro bits más significativos iguales a la dirección del nodo local (o el byte es igual a 00<sub>h</sub>, para un paquete de *broadcast*) dentro de la ventana de tiempo de 50 ms, éste podría ser un mensaje válido para el nodo local y el resto del mensaje debe ser recibido y verificado para saber si es un mensaje válido. Si ningún byte es regresado en 50 ms, seguido de la transmisión del byte "hacia"/"de", la tarjeta del protocolo puede asumirse como ocupada y el protocolo de la PC deberá intentar enviar el primer byte del mensaje otra vez.

Un paquete entrante puede ocurrir en cualquier momento, por lo tanto, cualquier byte con los cuatro bits más significativos iguales a la dirección del nodo local o cualquier byte 00<sub>h</sub> debe ser procesado y verificado para un mensaje válido.

### A.11 Formatos de mensaje del protocolo de la PC

En esta sección se presentan el formato general de mensaje y los mensajes especiales del protocolo de los módulos de comunicación.

#### Formato general de mensaje

Byte "hacia"/"de"	No. de paquete, de 01 <sub>h</sub> a 08 <sub>h</sub>	Byte de tamaño/estado	Mensaje (hasta 32 bytes)
-------------------	--	-----------------------	--------------------------

El byte de tamaño/estado indica el número de bytes en el mensaje (hasta 20<sub>h</sub>), o el estado del mensaje (por ejemplo, reconocimiento = E<sub>h</sub>, sin reconocimiento = DD<sub>h</sub>, etc.). El paquete número 08<sub>h</sub> indica un paquete de telemetría. Una dirección "hacia" de 00<sub>h</sub> indica un paquete de *broadcast*.

#### Mensajes de liga de estado

Reconocimiento	"hacia"/"de"	No. de paquete	E <sub>h</sub>	(n de 1 a 8)
No reconocimiento	"hacia"/"de"	No. de paquete	DD <sub>h</sub>	

#### Mensaje especial de la PC a la tarjeta de protocolo

Reinicio	"de"/"de"	00 <sub>h</sub>	01 <sub>h</sub>	30 <sub>h</sub>
Envío de la dirección del nodo	00 <sub>h</sub>	00 <sub>h</sub>	01 <sub>h</sub>	31 <sub>h</sub>
Fijación de la dirección del nodo	00 <sub>h</sub>	00 <sub>h</sub>	01 <sub>h</sub>	34 <sub>h</sub>
Envío de estado de batería	"de"/"de"	00 <sub>h</sub>	01 <sub>h</sub>	32 <sub>h</sub>
Para correr autoprueba	"de"/"de"	00 <sub>h</sub>	01 <sub>h</sub>	33 <sub>h</sub>

**Mensajes especiales de la tarjeta de protocolo hacia la PC**

<b>Mensaje de la PC</b>	<b>"de"/"de"</b>	<b>00<sub>b</sub></b>	<b>01<sub>b</sub></b>	<b>30<sub>b</sub></b>
<b>Batería baja</b>	<b>"de"/"de"</b>	<b>00<sub>b</sub></b>	<b>01<sub>b</sub></b>	<b>31<sub>b</sub></b>
<b>Batería en buen estado</b>	<b>"de"/"de"</b>	<b>00<sub>b</sub></b>	<b>01<sub>b</sub></b>	<b>32<sub>b</sub></b>
<b>Falla de autoprueba</b>	<b>"de"/"de"</b>	<b>00<sub>b</sub></b>	<b>01<sub>b</sub></b>	<b>33<sub>b</sub></b>
<b>Éxito en la autoprueba</b>	<b>"de"/"de"</b>	<b>00<sub>b</sub></b>	<b>01<sub>b</sub></b>	<b>34<sub>b</sub></b>
<b>Dirección de nodo local</b>	<b>"de"/"de"</b>	<b>00<sub>b</sub></b>	<b>01<sub>b</sub></b>	<b>35<sub>b</sub></b>

"de"/"de" es la dirección local.

## *Apéndice B*

# *Revisión del sistema mínimo para el microcontrolador MC68HC11F1*

El sistema mínimo para el microcontrolador 68HC11F1 de Motorola es una herramienta desarrollada con fines educativos. Este sistema está encaminado a facilitar el aprendizaje y manejo del microcontrolador MC68HC11F1 [Referencia electrónica Motorola, 2001]. Aunque este sistema está diseñado con fines didácticos, dadas sus características también puede ser empleado para aplicaciones de robótica, supervisión de sensores, control de motores, manejo de pantallas de cristal líquido, sistemas de comunicación y procesamiento de señales, entre otras.

Las características de este sistema mínimo son las siguientes: 32 kbytes de memoria RAM, 32 kbytes de memoria EPROM, un cristal de 8 MHz (con la opción de cambiar a un cristal cuya frecuencia se encuentre entre 4 y 8 MHz), sistema de reinicialización, conector para comunicación serial RS-232, selector de modos de operación, selector de memoria y voltaje, conectores para los puertos A, D, E y G del microcontrolador MC68HC11, opción para ser alimentado por una fuente de 5 volts o bien por una de más de 7.5 volts y un led indicador de encendido y apagado.

Dado que el sistema mínimo cuenta con 32 kbytes de memoria RAM y 32 kbytes de memoria EPROM, casi para todas las aplicaciones, el tamaño de los programas no es una limitante para su almacenamiento y ejecución. Otro aspecto importante a considerar del sistema mínimo son sus medidas, las cuales son 127.66 mm de largo por 75.3 mm de ancho, lo cual hace de él un sistema de desarrollo práctico para aplicaciones en robots móviles.

## B.1 Microcontrolador MC68HC11F1 de Motorola

La unidad de microcontrolador de alto desempeño MC68HC11F1 es un perfeccionamiento derivado de la familia de microcontroladores M68HC11 e incluye muchas características avanzadas. Este microcontrolador se caracteriza por su alta velocidad y su bajo consumo de energía [Manual Motorola, 1995].

Dentro de las características fundamentales de este microcontrolador se encuentran:

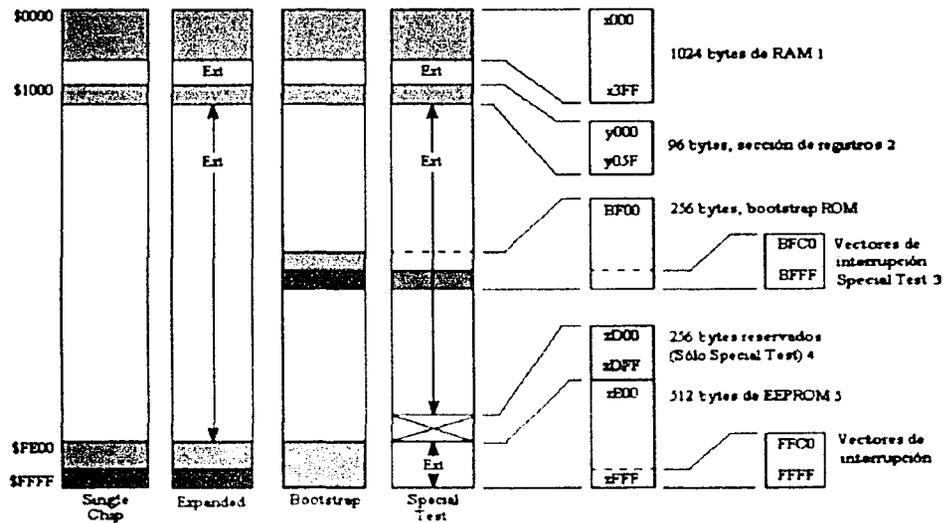
- Unidad Central de Procesamiento (CPU).
- Modos STOP y WAIT ahorradores de energía.
- 512 bytes de memoria EEPROM.
- 1024 bytes de memoria RAM. Retención de datos durante espera.
- Puertos de datos y direcciones no multiplexadas.
- Temporizador interno de 16 bits.
- Tres canales de captura de pulsos (IC).
- Cuatro canales de comparación de salida (OC).
- Un canal adicional, el cual puede usarse como el cuarto canal de captura de pulsos o el quinto canal de comparación de salida.
- Acumulador de pulsos de 8 bits.
- Circuito de interrupción en tiempo real.
- Calculo de operación correcta (COP) "watchdog".
- Interfaz de comunicación serial asíncrona (SCI).
- Interfaz de comunicación serial síncrona (SPI).
- 8 canales de convertidor analógico digital de 8 bits.

En la Fig. B.1 se observa que el MC68HC11F1 posee siete puertos: A, B, C, D, E, F y G. Los puertos A, B, C, E, F y G cuentan con 8 bits, mientras que el puerto D solamente con 6 bits. Los puertos A, C, D y G son completamente bidireccionales, mientras que los puertos B y F son puertos de salida únicamente y el puerto E es sólo de entrada. Dependiendo del modo de operación, estos puertos pueden ser configurados como direcciones de entrada o salida de datos para dispositivos externos.



direccionamientos usados en el modo Single Chip, además de una memoria externa o algún otro dispositivo. La expansión se debe realizar por medio de los puertos B, C y F, así como de la señal de *R/W*.

- *Special Test*. Este modo es una variación del modo Expanded. Básicamente es empleado por Motorola en su fabricación y control de calidad.
- *Bootstrap*. Es una variación especial del modo Single Chip. Este modo permite almacenar programas de propósito especial dentro de la memoria RAM interna del microcontrolador. Cuando se reinicia el microcontrolador, los vectores de interrupción y de reinicialización se alojan en una memoria ROM llamada "bootstrap ROM" y se ejecuta un programa inicializador. La "bootstrap ROM" contiene un pequeño programa que inicializa la comunicación serial asíncrona y permite al usuario cargar un programa de hasta 1024 bytes dentro de la memoria RAM interna.



- 1 La RAM puede ser ubicada cada 4 kbytes
- 2 La sección de registros puede ser ubicada cada 4 kbytes
- 3 En el modo Special Test los vectores de interrupción son direccionados externamente
- 4 En el modo Special Test las localidades \$xD00 - \$xDFF no son direccionables externamente
- 5 La EEPROM puede ser ubicada cada 4 kbytes

Fig. B.2 Mapa de memoria del microcontrolador MC68HC11F1 [Manual Motorola, 1995].

En la Fig. B.2 se presenta el mapa de memoria para los modos de funcionamiento del MC68HC11F1.

## B.2 Descripción del sistema mínimo para el microcontrolador MC68HC11F1 de Motorola

El sistema mínimo esta construido sobre una tarjeta impresa con etiquetas, en la cual son fácilmente identificados sus elementos. Además, los elementos están óptimamente distribuidos. En la Fig. B.3 se puede observar una imagen del sistema mínimo.

Debido a que el sistema cuenta con 32 kbytes de memoria RAM y 32 kbytes de memoria EPROM externas al microcontrolador, éste no tiene disponibles los puertos B, C y F; ésta es la razón por la cual no se tienen conectores para poder hacer uso de estos puertos.

El sistema tiene un cristal a una frecuencia de 8 MHz, aunque se tiene la posibilidad de cambiarse por otro cristal cuya frecuencia se encuentre entre los 4 y los 8 MHz.

También el sistema cuenta con un sistema de reinicialización, el cual inicializa al microcontrolador. Este sistema es activado por un interruptor normalmente abierto, que se cierra manualmente.

El conector para comunicación serial RS-232, cuenta con cuatro líneas, tierra (GND), recepción de datos (RxD), transmisión de datos (TxD) y 5 volts (Vcc).

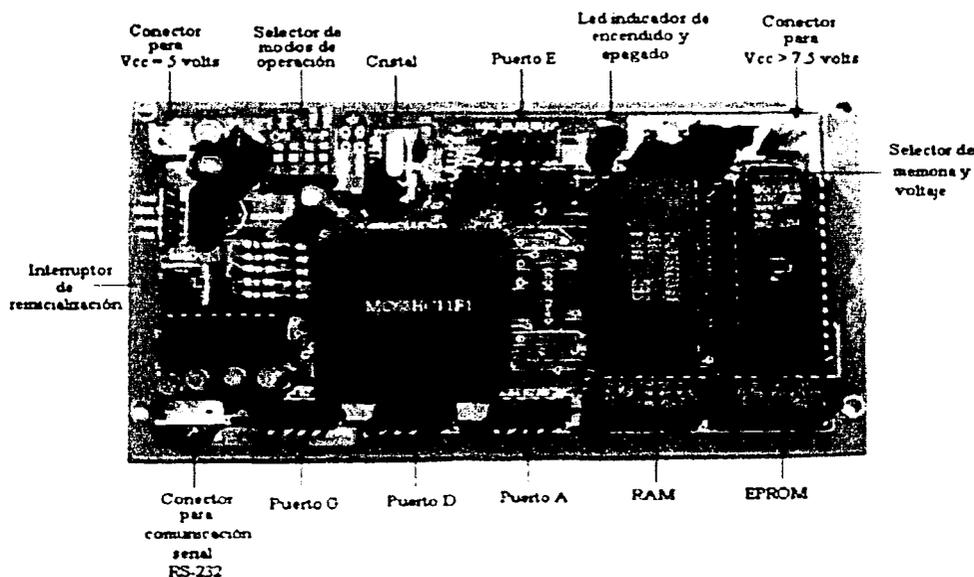


Fig. B.3 Sistema mínimo.

El selector de modos de operación cuenta con cuatro interruptores, de los cuales los dos primeros son las líneas RxD (PD0) y TxD (PD1) del microcontrolador. Cuando estos interruptores están cerrados ("on") permiten utilizar la entrada 0 y la 1 del puerto D como interfaz de comunicación serial asíncrona, pero cuando están abiertos ("off") las entradas 0 y 1 pueden ser utilizadas como líneas de entrada / salida de propósito general. El tercer y cuarto interruptor son las entradas MODA Y MODB del microcontrolador respectivamente. En la tabla B.1 se muestra la posición de estos dos interruptores para elegir el modo de operación del microcontrolador.

Modo de operación	Interruptor no. 3	Interruptor no. 4
Bootstrap	Cerrado	Cerrado
Single Chip	Cerrado	Abierto
Special Test	Abierto	Cerrado
Expanded	Abierto	Abierto

**Tabla B.1 Posición de los interruptores para seleccionar los modos de operación.**

El selector de memoria y voltaje se compone por ocho pares de postes, los cuales son programados mediante la colocación de puentes. Los primeros siete pares de postes configuran los bloques de memoria, mientras que el octavo par se encarga de seleccionar el voltaje de alimentación. En la tabla B.2, se muestra la función de cada uno de estos pares de postes.

Pares de postes	Función	Comentario
Primero	Deshabilita la memoria RAM	
Segundo	Habilita la memoria RAM en la parte alta del mapa de memoria	Dirección \$8000 a la \$FFFF
Tercero	Memoria RAM siempre habilitada	Dirección \$0000 a la \$FFFF
Cuarto	Habilita la memoria RAM en la parte baja del mapa de memoria	Dirección \$0000 a la \$7FFF
Quinto	Deshabilita la memoria EPROM	
Sexto	Habilita la memoria EPROM en la parte alta del mapa de memoria	Dirección \$8000 a la \$FFFF
Séptimo	Habilita la memoria EPROM en la parte baja del mapa de memoria	Dirección \$0000 a la \$7FFF
Octavo	Selecciona el voltaje de alimentación	Sin puente el voltaje de alimentación es de 5 volts

**Tabla B.2 Función de los postes del selector de memoria y voltaje.**

Es necesario tener mucho cuidado al programar el área de memoria que ocupará tanto la memoria RAM como la memoria EPROM, dado que si se colocan en la misma sección del bloque de memoria el sistema no operará adecuadamente. Otro aspecto importante a considerar es que las direcciones reservadas para uso del microcontrolador (sección de registros, memoria RAM interna, memoria EEPROM interna, vectores de interrupción, memoria ROM interna) no pueden ser usadas por la memoria RAM o por la memoria EPROM.

Si se coloca un puente en el octavo par de postes, el sistema se puede alimentar con un voltaje de corriente directa de entre 7.5 y 35 volts. Esto se debe a que el sistema cuenta con un regulador de voltaje positivo LM7805. En caso de no colocar el poste, el sistema deberá ser alimentado con un voltaje de corriente directa de 5 volts.

En la Fig. B.3 se observa que los conectores de los puertos A, D, E y G tienen diez líneas. En el caso de los puertos A, E y G que cuentan con 8 líneas de datos, se colocaron tanto una línea de Vcc como una de GND. En el caso del puerto D, que únicamente tiene seis líneas, se colocaron también una línea de Vcc y otra de GND, y para usar el mismo tipo de conector que para los otros puertos, se colocaron dos líneas más, las cuales no están conectadas a ningún sitio.

### B.3 Modificaciones al sistema mínimo

Para poder hacer uso del sistema mínimo en nuestro robot móvil, fue necesario cambiar el cristal de 8 MHz por uno de 4.9152 MHz. La razón tiene que ver con la tasa ("baud rate") de transmisión y recepción de la interfaz de comunicación serial asíncrona (SPI). El microcontrolador tiene la posibilidad de dividir la frecuencia de su reloj interno para producir diferentes tasas de transmisión y recepción. Esto se logra mediante la modificación de dos registros internos del microcontrolador. En las tablas B.3 y B.4 se observa qué tasas de transmisión y recepción se pueden alcanzar, dependiendo de la frecuencia del cristal del microcontrolador.

Prescalador		Reloj interno dividido por	Frecuencia del cristal (MHz)		
SCP1	SCP0		4.0	4.9152	8
0	0	1	62500	76800	125000
0	1	3	20833	25600	41667
1	0	4	15625	19200	31250
1	1	13	4800	5907	9600

Tabla B.3 Prescalador de la tasa de transmisión y recepción de la SPI.

SCR [2:0]	Dividir el prescalador por	Tasa más alta (salida del prescalador de la tabla 3)			
		4800	9600	19200	76800
000	1	4800	9600	19200	76800
001	2	2400	4800	9600	38400
010	4	1200	2400	4800	19200
011	8	600	1200	2400	9600
100	16	300	600	1200	4800
101	32	150	300	600	2400
110	64	75	150	300	1200
111	128	--	75	150	600

Tabla B.4 Selección de la tasa de transmisión y recepción de la SPI.

De las tablas B.3 y B.4 se puede observar que tanto con un cristal de 8 MHz como con uno de 4.9152 MHz se puede alcanzar una tasa de 9600 bauds que es normalmente la tasa más comúnmente usada para comunicar al microcontrolador con una PC. El problema es que con un cristal de 8 MHz no se puede alcanzar una tasa de 19200 bauds, que es la tasa necesaria para comunicarse con el módulo de comunicación. Por esta razón se optó por usar un cristal de 4.9152 MHz, el cual tiene la posibilidad de alcanzar tanto 19200 como 9600 bauds.

# Apéndice C

## Revisión de componentes mecánicos

### C.1 Sensores

Los sensores son tan diversos como los principios físicos en los que se basan. En la actualidad para medir cualquier variable física tenemos diversos tipos de sensores, entre los cuales se destacan los sensores electrónicos. La propiedad fundamental de un sensor electrónico es medir algunas características del ambiente, tales como luz, sonido o presión, y convertir la medición en una cantidad eléctrica representativa. Por esta razón, los sensores electrónicos son frecuentemente utilizados en los robots.

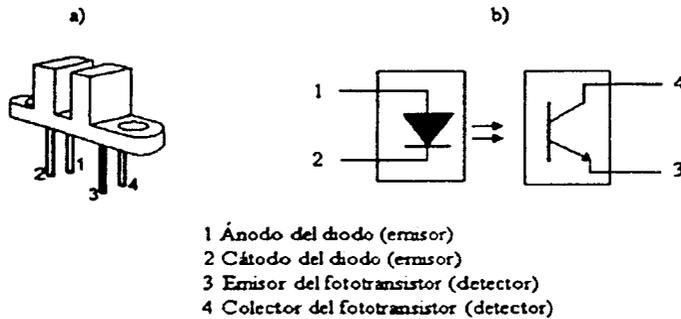
El robot móvil hace uso de sensores fotorreflexivos, táctiles e interruptores ópticos. En la sección 3.4.2 se detalla el objetivo de cada uno de estos sensores, así como su ubicación física y su respectiva etapa de acondicionamiento de las señales sensadas.

#### C.1.1 Interruptor óptico y sensor fotorreflexivo

Con el fin de detectar la proximidad o movimiento, se pueden usar tanto interruptores ópticos como sensores fotorreflexivos, los cuales únicamente usan la tecnología led-fototransistor.

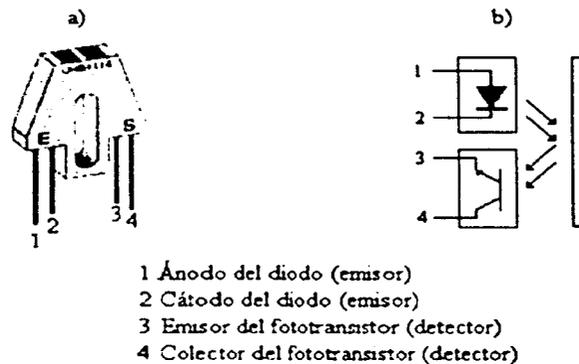
Un interruptor óptico consiste en un led (diodo emisor de luz) acoplado a un fototransistor (es decir, uno enfrente del otro) a través de una ranura de 1/8 de pulgada (3.175 mm) [Horowitz y Hill, 2001]. Éste puede sensar, por ejemplo, la presencia de una lámina opaca o la rotación de un disco ranurado. En la Fig. C.1 se muestra un diagrama físico y uno eléctrico de un interruptor óptico.

Los interruptores ópticos son ocupados en manejadores de discos e impresoras, para sensar el inicio o fin de una tarea. También se pueden utilizar en codificadores (*encoders*) rotatorios.



**Fig. C.1 Interruptor óptico: a) Diagrama físico. b) Diagrama eléctrico.**

Una forma alternativa es colocar el led y el fototransistor apuntando en la misma dirección (es decir, el led y el fototransistor están colocados paralelamente) y de esta forma sensor la presencia de un objeto cercano por reflexión. Este dispositivo es conocido como sensor fotorreflexivo. En la Fig. C.2 se ilustran dos tipos de sensores fotorreflexivos, así como su diagrama eléctrico.



**Fig. C.2 Sensor fotorreflexivo.**

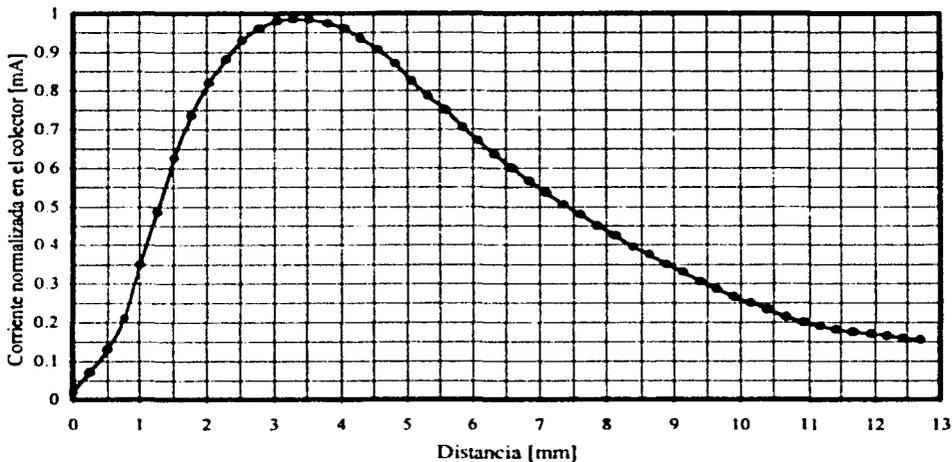
**Características de los sensores fotorreflexivos utilizados en el robot móvil**

Los sensores fotorreflexivos usados en el robot móvil son de la empresa Fairchild Semiconductor®<sup>12</sup> y tienen las siguientes características [Referencia electrónica Fairchild, 2002]:

- Sensado de superficie sin hacer contacto con ella.

<sup>12</sup> Fairchild Semiconductor®, www.fairchildsemi.com.

- Salida fototransistor.
- Filtro de luz natural en el fototransistor
- Protector de polvo.
- Temperatura máxima de operación:  $-40^{\circ}\text{C}$  a  $85^{\circ}\text{C}$ .
- Corriente máxima continua en el emisor: 50 mA.
- Voltaje máximo en inversa en el emisor: 5 V.
- Potencia máxima de disipación en el emisor: 100 mV.
- Voltaje máximo de colector-emisor: 30 V.
- Voltaje máximo de emisor-colector: 4.5 V.
- Potencia máxima de disipación del sensor: 100 mW.



**Fig. C.3 Corriente normalizada vs distancia.**

En la Fig. C.3 se presenta la gráfica correspondiente a la corriente normalizada en el colector a diferentes distancias, en la cual es claro observar que la corriente máxima la alcanza entre 3 y 3.5 mm aproximadamente. La distancia que hay entre el piso y los sensores fotorreflectivos en el robot móvil es de aproximadamente 5 mm.

#### **Características de los interruptores ópticos utilizados en el robot móvil**

- Corriente máxima en el diodo: 50 mA.

- Voltaje en inversa máximo en el diodo: 6 VCD.
- Potencia de disipación máxima en el diodo: 100 mW.
- Voltaje de colector-emisor máximo en el fototransistor: 30 VCD.
- Voltaje de emisor-colector máximo en el fototransistor: 4.5 VCD.
- Corriente máxima en el colector del fototransistor: 20 mA.
- Potencia de disipación máxima en el fototransistor: 150 mW.

### C.1.2 Microconmutadores

Los microconmutadores son dispositivos que se emplean con frecuencia en la implementación de sensores táctiles, a pesar de ser dispositivos cuya operación es primitiva.

El sensor táctil es simplemente un interruptor de contacto momentáneo montado sobre un robot, así que cuando el robot choca con algo, el interruptor es accionado. De esta forma, el robot puede detectar que ha hecho contacto con algún objeto.

Los microconmutadores son interruptores de contacto que cuentan con tres líneas: una línea común, otra normalmente abierta y la tercera normalmente cerrada. En la Fig. C.4 se ilustra el diagrama de un microconmutador.

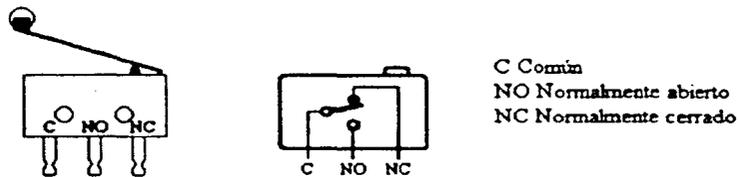


Fig. C.4 Diagrama esquemático de un microconmutador.

**Características de los microconmutadores utilizados en el robot móvil [Referencia electrónica Omron, 2002]:**

- Fuerza de activación: 50 gramos fuerza.
- Tiempo de vida: 30 millones de operaciones mecánicas mínimo.
- Tiempo de vida: 200,000 operaciones eléctricas mínimo.
- Frecuencia de operación mecánica: 400 operaciones por minuto.

- Frecuencia de operación eléctrica: 60 operaciones por minuto.
- Resistencia de aislamiento: 100 M $\Omega$  a 500 VCD.
- Velocidad de operación: 0.1 mm/s a 1 m/s.
- Fácil manejo y fijación.
- Económicos.

## C.2 Motores

Los motores son dispositivos que convierten energía eléctrica en energía mecánica; esta es la razón por la cual se encuentran presentes en la mayoría de los robots móviles. Los motores que se emplean en la construcción de robots móviles pueden ser motores de corriente directa o motores de pasos. Cada uno de estos tipos de motores presenta ventajas y desventajas; por ejemplo, los motores de corriente directa no son tan precisos como los motores de pasos; sin embargo, poseen un mayor torque. Los motores de pasos normalmente son más pesados que los motores de corriente directa [Silva, 1989].

Para la construcción del robot móvil se emplearon ambos tipos de motores; dos de corriente directa para el movimiento de rotación y traslación y uno de pasos para controlar la dirección.

### C.2.1 Motores de corriente directa

Los motores de corriente directa (CD), propiamente dicho de corriente continua, convierten la energía eléctrica de corriente directa en energía mecánica rotacional. El motor de corriente directa es probablemente la forma más antigua de un motor eléctrico. Debido a sus características tales como torque alto, control en la velocidad sobre un amplio rango, portabilidad, características de buen comportamiento al combinar velocidad y torque, y adaptabilidad a varios tipos de métodos de control, los motores de corriente continua siguen siendo ampliamente usados en numerosas aplicaciones de control incluyendo robots manipuladores, mecanismos de transporte de cinta, controladores o manejadores de discos, entre otros [Silva, 1989].

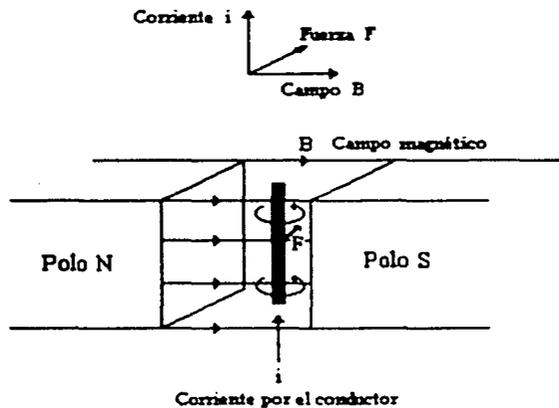
#### Principio de operación

El principio de operación de un motor de corriente directa se ilustra en la Fig. C.5. Considerando un conductor eléctrico colocado en un campo magnético uniforme con ángulo recto en la dirección del campo. La densidad de flujo  $B$  se asume como constante. Si la corriente de directa es pasada a través del conductor, el flujo magnético debido a la corriente hará un lazo que rodea al conductor, como se muestra en la Fig. C.5. Considerando un plano a través del conductor, paralelo a la dirección del flujo magnético, sobre un lugar de este plano, el flujo de corriente y el flujo de campo son aditivos; sobre el

lado opuesto del plano, los dos flujos magnéticos son opuestos entre sí. Como resultado, se genera una fuerza magnética  $F$  no balanceada sobre el conductor, normal al plano. Esta fuerza está dada por

$$F = Bi\ell \quad (C.1)$$

donde  $B$  = flujo de densidad del campo original  
 $i$  = corriente a través del conductor  
 $\ell$  = longitud del conductor.



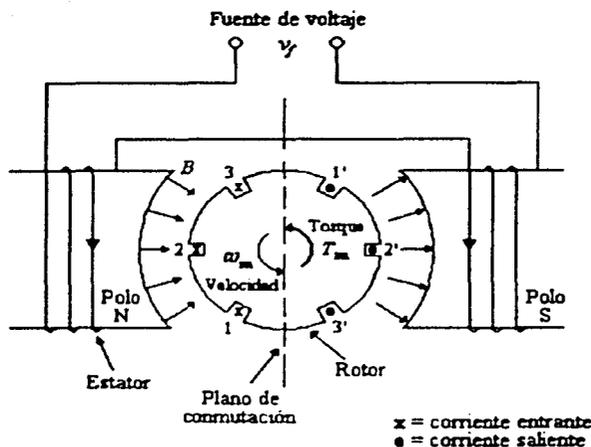
**Fig. C.5 Principio de operación de un motor de corriente directa [Silva, 1989].**

Las componentes  $i$ ,  $B$  y  $F$  son perpendiculares mutuamente y forman las tres componentes de la regla de la mano derecha, como se presenta en la Fig. C.5. Alternativamente, en la representación vectorial de estas tres cantidades, el vector  $F$  puede ser interpretado como el producto cruz de los vectores  $i$  y  $B$ .

Si el conductor es libre para moverse, la fuerza lo moverá con una velocidad  $v$  en la dirección de la fuerza. Como resultado de este movimiento en el campo magnético  $B$ , un voltaje es inducido en el conductor. Esto es conocido como la fuerza electromotriz de regreso (*back e.m.f.*) y está dada por

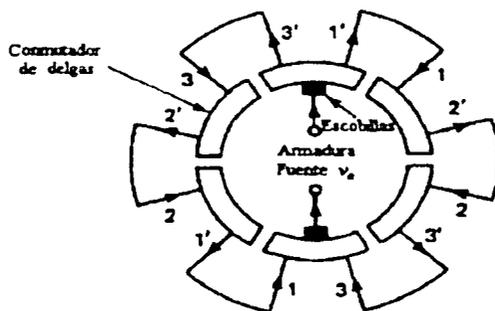
$$v_b = B\ell v \quad (C.2)$$

Por la ley de Lenz, se obtiene que el flujo debido a la fuerza electromotriz de regreso,  $v_b$ , será opuesta al flujo debido a la corriente original a través del conductor, y de esta forma intenta detener el movimiento. La ecuación (C.1) determina el torque de la armadura (torque del motor), y la ecuación (C.2) determina la velocidad del motor.



**Fig. C.6 Diagrama de un motor de corriente directa [Silva, 1989].**

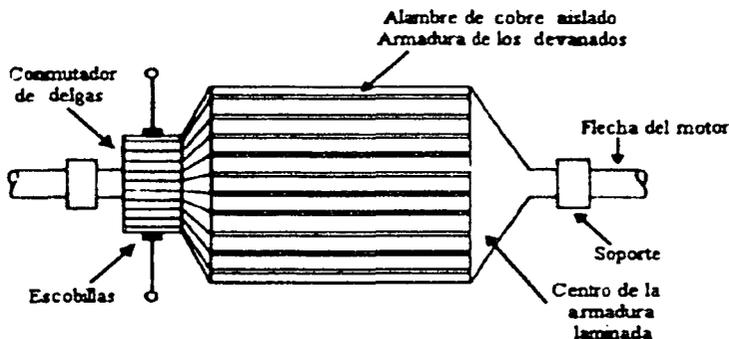
En los motores de corriente directa industriales, el flujo de campo usualmente es generado no por un imán permanente sino eléctricamente en los devanados del estator- mediante un electromagneto, como se muestra en la Fig C.6. Los polos del estator están contruidos por laminillas ferromagnéticas. El rotor es típicamente un cilindro laminado construido de un material ferromagnético. Un centro ferromagnético ayuda a concentrar el flujo magnético hacia el rotor. La laminación reduce el problema de histéresis magnética y limita la generación de corrientes parásitas y la disipación asociada con el material ferromagnético. El rotor tiene muchas ranuras cercanamente espaciadas sobre su periferia, estas ranuras sostienen los devanados del rotor. Los devanados del estator son alimentados por la fuente de voltaje  $v_f$ , y los devanados del rotor (devanados de la armadura) son alimentados por la fuente de voltaje  $v_a$ .



**Fig. C.7 Diagrama del conmutador [Silva, 1989].**

## Conmutación

Para mantener la dirección del torque en cada grupo conductor (numerados 1, 2, 3, 1', 2' y 3' en la Fig. C.6), la dirección de la corriente en un conductor debe cambiar mientras el conductor cruza el plano de conmutación [Silva, 1989]. Físicamente, esto puede llevarse a cabo usando un anillo con hendiduras (delgas) y un conmutador de escobillas, mostrado en la Fig. C.7. El voltaje de la armadura es aplicado a los devanados del rotor a través de un par de escobillas de carbón que mantiene el contacto con las delgas. Las delgas, que son iguales en número a las ranuras del conductor en el rotor, son eléctricamente aisladas una de otra, pero las delgas opuestas están conectadas a través de los devanados de la armadura. Para la posición del rotor mostrado en las Figs. C.6 y C.7, se puede notar que cuando las delgas rotan  $30^\circ$  en la dirección de las manecillas del reloj, la trayectoria de la corriente en los conductores 1 y 1' es inverso pero las trayectorias de las corrientes restantes no son cambiadas; de esta manera se logra la conmutación requerida. Mecánicamente, esto es posible porque las delgas están rígidamente montadas sobre la flecha del rotor, como se muestra en la Fig. C.8.



**Fig. C.8 Configuración física del rotor de un motor de corriente directa [Silva, 1989].**

## Características de los motores de corriente directa utilizados en el robot móvil

Los motores de corriente directa empleados en la construcción del robot móvil son de la empresa HS I ANG NENG<sup>®</sup> y sus características son las siguientes [Catálogo Jameco, 2000]:

- Voltaje típico de operación: 12 VCD.
- Rango de voltaje de operación: 4.5 a 12 VCD.
- Corriente sin carga: 150 mA.
- Velocidad sin carga: 42 rpm.

- Torque de inicio: 8 kg/cm.
- Razón de reducción: 132:1.
- Peso: 0.3 lb (136.08 g).
- Tamaño del motor: 1.2 in (3.048 cm) de diámetro por 0.6 in (1.524 cm) de largo.
- Tamaño del reductor: 1.2 in (3.048 cm) de diámetro por 0.9 in (2.286 cm) de largo.
- Tamaño de la flecha: 0.16 in (0.406 cm) de diámetro por 0.6 in (1.524 cm) de largo.

### C.2.2 Motor de pasos

Los motores de pasos son motores que funcionan girando a “saltos” (pequeños incrementos en la posición del rotor), donde todos los saltos son constantes, es decir, que tienen el mismo ángulo. Los motores de pasos son empleados debido al posicionamiento tan preciso que permiten [Acarney, 1984].

Los términos motor a pasos, motor de pasos, motor paso a paso y motor de pulsos son sinónimos y generalmente son usados de manera intercambiable.

Hay un amplio número de diseños de motores de pasos, pero la mayoría de éstos pueden ser identificados como variaciones de dos tipos básicos: los motores de pasos de reluctancia<sup>13</sup> variable y los motores de pasos de magneto permanente [Bohlke, D., 2001]. Aunque la geometría interna del motor parezca muy diferente, siempre es posible modelar su comportamiento en función de estos dos tipos básicos de motores. Los motores de magneto permanente están divididos en dos tipos: unipolares y bipolares. Los motores híbridos poseen características tanto de los motores de pasos de reluctancia variable, como de los motores de pasos de magneto permanente [Silva, 1989].

La propiedad principal de los motores de pasos es su habilidad de transformar los cambios de excitación conmutados en incrementos bien definidos de la posición del rotor (“pasos”). El posicionamiento preciso del rotor es generalmente producido por el alineamiento magnético de los dientes de hierro entre la parte fija o estacionaria y las partes rotatorias del motor. Para el motor de magneto permanente, la fuente principal de flujo magnético es un magneto permanente y las corrientes fluyen en uno o más devanados. Existen dos configuraciones para los motores de pasos de reluctancia variable, pero en ambos casos el campo magnético es producido solamente por las corrientes en los devanados [Silva, 1989].

Los motores de reluctancia variable generalmente tienen tres (algunas veces cuatro) devanados, con una tierra común, mientras que los motores de magneto permanente usualmente tienen dos devanados independientes, con o sin tap central. Los devanados con tap central son usados en los motores de magneto permanente unipolares.

---

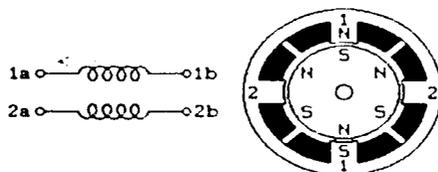
<sup>13</sup> La reluctancia mide la resistencia magnética en un sentido de flujo.

### Principio de operación del motor de pasos de magneto permanente

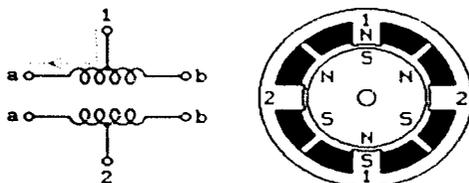
Una característica común en cualquier motor de pasos es que el estator del motor contiene varios pares de devanados magnetizados que pueden ser conmutados para producir pares de polos electromagnéticos (Norte o Sur). Las polaridades pueden ser invertidas en dos formas:

- 1 Invertiendo la dirección de la corriente en el devanado.
- 2 Usando dos pares de devanados (devanados bipolares) por cada par de polos; un par proporciona un conjunto de polos cuando es energizado y el otro par proporciona las polaridades opuestas.

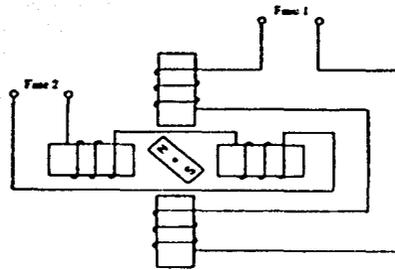
En el caso de los devanados bipolares, es adecuado un mecanismo de conmutación relativamente simple para invertir las polaridades; por esta razón, es empleada una circuitería de control simplificada. En la Fig. C.9 se muestra un diagrama de un motor de pasos de magneto permanente bipolar. La circuitería de control para devanados unipolares es más compleja porque se requiere de más circuitería para la inversión de la corriente; además, se necesitaría el doble del número de devanados que en un motor de devanados bipolares, de los cuales, la mayoría están inactivos en un cierto tiempo. Esto incrementa el tamaño del motor para un mismo torque. En la Fig. C.10 se presenta un diagrama de un motor de pasos de magneto permanente unipolar. Una ventaja adicional del motor de pasos bipolar es su nivel más bajo de voltajes inducidos por auto inducción e inducción mutua, porque no hay inversiones de corriente. Por esta razón, el torque (dinámico) efectivo a una velocidad dada es usualmente más grande para un motor de pasos bipolar que para su equivalente unipolar, particularmente a altas velocidades [Referencia electrónica Jones, D., 2001].



**Fig. C.9 Diagrama de un motor de pasos de magneto permanente bipolar [Referencia electrónica Jones, D., 2001].**



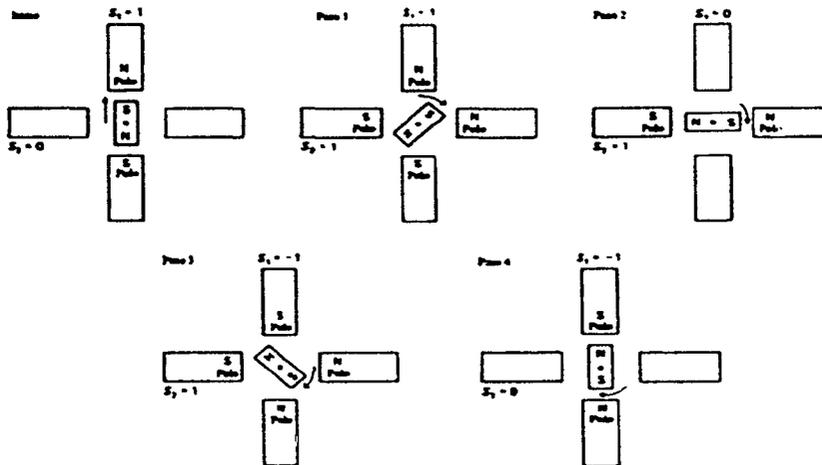
**Fig. C.10 Diagrama de un motor de pasos de magneto permanente unipolar [Referencia electrónica Jones, D., 2001].**



**Fig. C.11 Diagrama de un motor de pasos de dos fases (bipolar) de magneto permanente [Silva, 1989].**

El principio de operación de un motor de pasos de magneto permanente se explica mediante un diagrama sencillo mostrado en la Fig. C.11. El estator tiene dos conjuntos de devanados (bipolar), colocados a  $90^\circ$ . Este arreglo tiene cuatro polos salientes en el estator, cada polo está separado geométricamente con un ángulo de  $90^\circ$  del polo adyacente. El rotor es un magneto permanente de dos polos. Cada fase puede tomar uno de tres estados 1, 0 y  $-1$ , que se definen a continuación [Silva, 1989]:

- Estado 1: corriente en una dirección específica.
- Estado  $-1$ : corriente en la dirección opuesta.
- Estado 0: sin corriente.

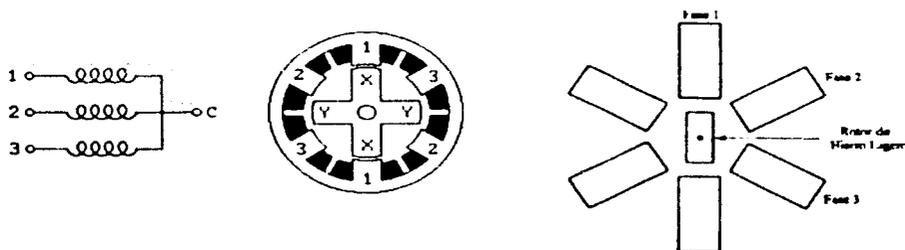


**Fig. C.12 Secuencia de paso (medio paso) para un motor de dos fases de magneto permanente para una rotación en sentido de las manecillas del reloj [Silva, 1989].**

Conmutando las corrientes en las dos fases en una secuencia apropiada, se puede producir una rotación en sentido dextrógiro (en sentido de las manecillas del reloj) o una rotación en sentido levógiro (en sentido contrario a las manecillas del reloj). La rotación en sentido levógiro se muestra en la Fig. C.12. Cabe hacer mención que  $S_i$  denota el estado de la  $i$ -ésima fase. El ángulo del paso para este motor es  $45^\circ$ . Al final de cada paso, el rotor asume la posición de reluctancia mínima que corresponde a un patrón de polaridad magnética particular en el estator. Esta es una configuración de equilibrio estable y es conocida como la posición de parada (*detent position*) para ese paso. Cuando las corrientes del estator son conmutadas para el siguiente paso, la posición de la reluctancia mínima cambia y el rotor asume la correspondiente posición de equilibrio estable: el rotor gira un paso simple ( $45^\circ$  para este ejemplo).

### Principio de operación del motor de pasos de reluctancia variable

El motor de pasos de reluctancia variable se muestra en la Fig. C.13. El rotor es una barra de hierro ligero sin magnetizar. Si son usadas solamente dos fases en el estator, habrá una ambigüedad en cuanto a la dirección de rotación, así que se requerirían al menos tres fases para esta geometría del rotor de dos polos, como se muestra en la Fig. C.13 [Referencia electrónica Jones, D., 2001].



**Fig. C.13 Diagrama del motor de pasos de reluctancia variable de tres fases.**

La secuencia de paso completo para una rotación en sentido dextrógiro se muestra en la Fig. C.14. El ángulo del paso es de  $60^\circ$ . Se energiza solamente una fase en un tiempo para ejecutar la secuencia de un paso completo y únicamente son usados los estados 1 y 0 para cada fase. La secuencia de medio paso para una rotación en sentido dextrógiro se muestra en la Fig. C.15. En este caso, dos fases deben ser energizadas simultáneamente durante algunos pasos, además, se necesitan inversores de corriente, lo cual implica una circuitería más elaborada de conmutación. Sin embargo, la ventaja es que el ángulo del paso ha sido dividido a la mitad, es decir a  $30^\circ$ , de esta manera se proporciona una mejoría en la resolución del movimiento. Cuando dos fases están activadas simultáneamente, la posición de la reluctancia mínima está en medio de los pares de los polos correspondientes, los cuales habilitan el medio paso. Así, dependiendo de la secuencia de energización de las fases, sería posible tanto movimientos de paso completo (ángulo de paso de  $60^\circ$  en este

ejemplo) como movimientos de medio paso (ángulo de paso de  $30^\circ$  en este ejemplo) [Silva, 1989].

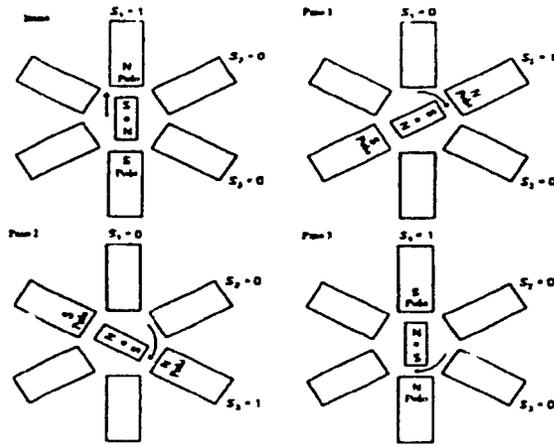


Fig. C.14 Secuencia de paso completo para un motor de pasos de tres fases de reluctancia variable (ángulo de paso =  $60^\circ$ ) [Silva, 1989].

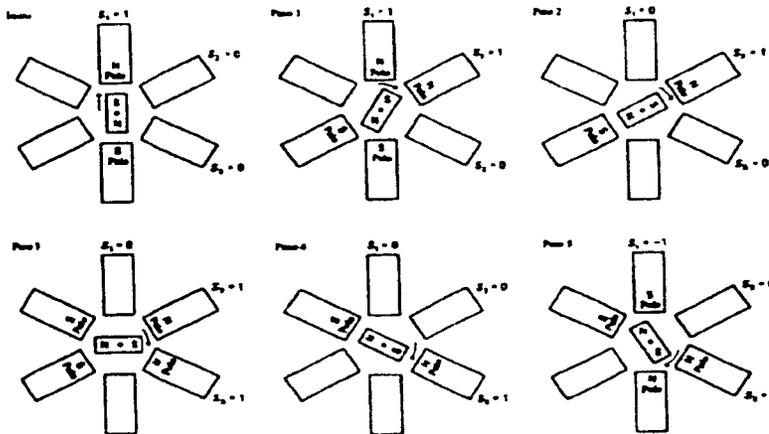


Fig. C.15 Secuencia de medio paso para un motor de pasos de tres fases de reluctancia variable (ángulo de paso =  $30^\circ$ ) [Silva, 1989].

### Aplicaciones del motor de pasos

Los motores de pasos son idealmente apropiados para aplicaciones de control digital. Existen numerosas aplicaciones de motores de pasos. Por ejemplo, un motor de pasos es

adecuado en aplicaciones de impresión (incluyendo impresoras, máquinas para dibujar planos, y máquinas de escribir electrónicas) porque los caracteres de impresión son cambiados en pasos y las líneas impresas (o alimentador de papel) también avanza en pasos. Los motores de pasos son usados extensivamente en manejadores de cinta magnética y manejadores de disco de computadoras digitales, particularmente para posicionar la cabeza o encabezado de lectura. En aplicaciones de manufacturación automatizada, los motores de pasos se encuentran como actuadores<sup>14</sup> de robots manipuladores [Silva, 1989].

Otras aplicaciones de los motores de pasos incluyen fuentes y objetos de posicionamiento en la medicina y radiografía metalúrgica, manejo en cámaras de enfoque automático, movimiento de cámaras en sistemas de visión computarizados, y mecanismos de alimentación de papel en máquinas de fotocopiado.

### Ventajas y desventajas de los motores de pasos

Las ventajas de los motores de pasos son las siguientes:

- 1 El error de posición no es acumulativo. Es posible una alta precisión de movimiento.
- 2 Es posible economizar en sensores (medición del sistema).
- 3 Los motores de pasos son fácilmente adaptables a aplicaciones de control digital.
- 4 No existen problemas serios de estabilidad.
- 5 La capacidad del torque y los requerimientos de potencia pueden ser optimizados y la respuesta puede ser controlada por conmutación electrónica.

Las desventajas de los motores de pasos son:

- 1 Tienen una capacidad de torque bajo comparado con el torque de los motores de corriente directa.
- 2 Tienen una velocidad limitada (limitada por la capacidad del torque y por los problemas de la falta de pulso debido a sistemas de conmutación y circuitos manejadores).
- 3 Tienen altos niveles de vibración debido al movimiento del paso.
- 4 Tienen un mayor volumen y peso que los motores de corriente directa.

---

<sup>14</sup> Un actuador es el dispositivo que maneja mecánicamente un sistema de control.

**Características del motor de pasos utilizado en el robot móvil**

El motor de pasos empleado en la construcción del robot móvil es de la empresa NIPPO® y sus características son las siguientes [Catálogo Jameco, 2000]:

- Ángulo de paso: 0.9°.
- Número de fases: 2.
- Tipo bipolar.
- Voltaje de operación: 8.4 VCD.
- Resistencia de fase: 30  $\Omega$ .
- Corriente de operación: 280 mA.
- Inductancia de fase: 25 mH.
- Torque de inicio: 36 g/cm.
- Torque de operación: 791 g/cm.
- Tamaño del motor: 1.64 in (4.161 cm) de diámetro por 1.2 in (3.048 cm) de largo.
- Tamaño de la flecha: 0.155 in (0.394 cm) de diámetro por 0.29 in (0.737 cm) de largo.
- Peso: 0.53 lb (240.4 g).

## Apéndice D

### Revisión del pseudo código

Los programas que ejecuta el robot móvil están hechos en lenguaje C. Con ayuda del compilador cruzado ICC11 para Windows de la empresa ImageCraft<sup>15</sup> [Manual ICC11, 1997] se genera el código ejecutable (extensión S19) para el microcontrolador MC68HC11F1 de Motorola.

Los programas se transfieren al microcontrolador usando el programa monitor PCBug11 de Motorola, el cual permite interactuar con el microcontrolador bajo condiciones específicas [Manual Pcbug11, 1992]. No se utilizó el compilador cruzado ICC11 para Windows para transferir los programas al microcontrolador, porque dicho compilador cruzado necesita que en el microcontrolador esté instalado el programa monitor BUFFALO<sup>16</sup> [Manual EVBU, 1997].

Normalmente los programas se colocan en la memoria RAM del microcontrolador, y una vez que han sido probados, se pueden colocar en la memoria EPROM si así se desea. Cuando se carga un programa en el microcontrolador se hace vía cable y para ejecutarlo se puede hacer vía cable o vía inalámbrica.

El entrenamiento de la red neuronal, así como la aplicación de la misma, están programados en MATLAB (versión 6). Por lo tanto, una vez que la computadora recibe la información de uno de los recorridos realizados por el robot móvil, se genera un archivo de texto con la matriz de datos correspondientes. Después que se tiene este archivo de texto, el programa en MATLAB toma dicha información y entrena o prueba la red, según sea el caso.

#### D.1 Recorrido por el robot móvil de la ruta 1

Una vez que el robot móvil tiene el programa guardado en la memoria RAM del microcontrolador, el robot debe ser colocado en la posición inicial de la pista de entrenamiento, la cual se puede observar en la Fig. 4.2 y el recorrido de la ruta 1 se puede observar en la Fig. 4.1.

<sup>15</sup> <http://www.imagecraft.com>.

<sup>16</sup> Bit User Fast Friendly Aid to Logical Operations.

El programa para que el robot ejecute el recorrido de la ruta 1, lleva por nombre "ruta\_1.c" y a continuación se presenta su pseudo código.

```

main(){
    inicio();           /* esta función inicializa algunos registros del microcontrolador y además alinea la
                        llanta delantera */
    mov_polar(90,0);   /* esta función hace que el robot gire 0° sobre su propio eje y avance 90 cm hacia
                        adelante */
    mov_polar(190,90); /* esta función hace que el robot gire 90° sobre su propio eje y avance 190 cm
                        hacia adelante */
    inicio_comunicación(); /* esta función inicializa algunos registros del microcontrolador para enviar la
                            información obtenida del recorrido a la computadora */
    offset=0x5000;     /* la variable offset se inicia con el valor 5000h, porque la información obtenida se
                        encuentra colocada a partir de la dirección 5000h en el microcontrolador */
    while(offset<=0x51ff) /* mientras que el valor de la variable offset sea menor o igual a 51ffh, se realizará
                            el ciclo */
    {
        for(s=1;s<=16,s++) /* el ciclo se realizará 16 veces, para formar arreglos de 16 datos */
        {
            a[s]="(unsigned char*)offset; /* el dato número s del arreglo a toma la información del registro
                                            cuya dirección esta dada por la variable offset */
            offset++; /* se incrementa el valor de la variable offset */
        }
        paquete_btx(0x32,16,a); /* la función transmite la información contenida en el arreglo a. El primer
                                campo indica el nodo al cual se transmite la información (nodo 3) y el nodo
                                del cual proviene la información (nodo 2). El segundo campo indica el tamaño
                                del arreglo. Finalmente, el tercer campo indica el nombre del
                                arreglo */
    }
}

inicio(){
    /* esta función inicializa algunos registros del microcontrolador y además alinea la
    llanta delantera */
    DDRA=0xff; /* habilita el puerto A como puerto de salida */
    TCTL1=0x44; /* habilita el registro OC2 y el registro OC4 para que puedan funcionar como línea
                 de salida, la cual pueda tomar valores "0" o "1" lógicos */
    OC1M=0x50; /* habilita el control de la salida de comparación del temporizador sobre las líneas
                 PA6 y PA4 del puerto A */
    OC1D=0x50; /* este registro permite que cuando ocurra una comparación exitosa o que el
                 resultado se coloque en la línea PA6 o PA4 del puerto A, según sea el caso */
    TOC2=0x0000; /* el temporizador para el registro OC2 comienza comparando con el dato 0000 */
    TOC4=0x0000; /* el temporizador para el registro OC4 comienza comparando con el dato 0000 */
    alinea(); /* esta función se encarga de alinear la llanta delantera del robot móvil, primero
               hace girar el motor de pasos 300° en sentido dextrógiro y posteriormente se hace
               girar 400° en sentido levógiro, teniendo el cuidado de terminar dicho
               desplazamiento en la posición denominada "A". La posición "A" se determinó
               experimentalmente y es la posición del motor de pasos donde el tope y el tornillo
               del collarín que se encuentra sobre la flecha del motor de pasos hacen contacto.
               Una vez que el motor de pasos se encuentra en la posición "A", éste se hace girar
               5.4° en sentido dextrógiro y esta posición se denomina como 0°. Los 5.4° se
               determinaron también en forma experimental, debido a que en esta posición el
               robot avanza en línea recta sin desviarse hacia sus costados */
}

mov_polar(distancia, float grados){
    /* esta función hace girar al robot móvil sobre su propio eje el
    número de grados que aparece en el segundo campo de esta
    función y luego avanza hacia adelante la distancia del primer
    campo */
    grados_sensor= 2.112211 * (grados) - 4.851485; /* esta es la ecuación (3.33) y determina el número de
                                                    veces que se debe desactivar el interruptor óptico para que el
                                                    robot gire cierto número de grados */
}

```

```

distancia_sensor = 12.193139 * (distancia) + 1.436758 /* esta es la ecuación (3.31) y determina el
número de veces que se debe desactivar el interruptor óptico para
que el robot avance hacia adelante una cierta distancia */
if(grados_sensor<=0){ /* si la variable grados_sensor es menor o igual a cero el robot no gira */
:
}
else{
/* en esta sección del programa, el robot calcula las distancias que recorren los motores 1 y 2 y
verifica la información de los sensores táctiles y fotorreflektivos cada 0.33 segundos al girar el robot
sobre su propio eje y dicha información la coloca en la memoria RAM del microcontrolador, hasta
completar el número de grados requerido. Esta información se guarda a partir de la dirección que
indica una variable externa llamada offset (en un inicio dicha variable tiene el valor de 5000h) de la
siguiente forma: suponiendo que la variable offset tiene el valor de 5000h, la distancia que recorre el
motor 1 se coloca en el registro con dirección 5000h y 5001h, ya que cada dato ocupa dos registros, la
distancia que recorre el motor 2 se coloca en los registros 5002h y 5003h, y finalmente el dato
proveniente de los sensores táctiles y fotorreflektivos se coloca en el registro 5005h y en el registro
5004h se coloca el dato fh. Después de 0.33 segundos, la nueva distancia que recorrió el motor 1 se
coloca en los registros 5006h y 5007h, y así sucesivamente para la distancia que recorrió el motor 2 y
la información de los sensores táctiles y fotorreflektivos. Cabe mencionar que a pesar de que la
información de los sensores táctiles y fotorreflektivos ocupa únicamente un registro (1 byte) se toman
dos, esto para tener un salto constante en la dirección de los registros, por lo tanto, cada seis
registros se tiene almacenada la información del motor 1, 2 y de los sensores en un determinado
instante */
}
stop(); /* esta función detiene los motores 1 y 2 del robot móvil sin afectar ninguna
otra tarea que esté realizando */
if(distancia_sensor<=0){ /* si la variable distancia_sensor es menor o igual a cero el robot no
avanza */
:
}
else{
/* en esta sección del programa, el robot calcula las distancias que recorren los motores 1 y 2 y
verifica la información de los sensores táctiles y fotorreflektivos cada 0.33 segundos al avanzar hacia
adelante y dicha información la coloca en la memoria RAM del microcontrolador hasta completar la
distancia necesaria. Esta información se guarda después de la información correspondiente al giro
sobre su propio eje con el mismo formato */
}
stop(); /* esta función detiene los motores 1 y 2 del robot móvil sin afectar ninguna
otra tarea que esté realizando */
}
}
inicio_comunicación(){ /* esta función inicializa algunos registros del microcontrolador para enviar
la información obtenida del recorrido a la computadora */
SPCR |= 0x20; /* habilita el puerto D como salida open-drain, si no se habilita de esta forma, se
comportaría como una salida OR alambrada */
SCCR1 = 0x00; /* selecciona recepción y transmisión de 8 bits, un bit de inicio y un bit de parada */
BAUD = 0xA0; /* selecciona la tasa de transmisión a 19200 bauds */
SCCR2 = 0x0C; /* habilita la transmisión y la recepción */
}
}
paquete_txt(unsigned char to_from, int i, unsigned char a[ ]){
/* esta función ordena la información contenida en el arreglo a[ ] (tercer campo de la función) para ser
transmitida por el módulo de comunicación de acuerdo a su protocolo. Para esto, requiere la dirección del
nodo transmisor y la del nodo receptor, dicha información se encuentra contenida en la variable to_from,
también requiere la longitud del mensaje, la cual se encuentra en la variable i (dicha longitud se fijó a 16
bytes) y la información contenida en el arreglo a[ ] */
}

```

## D.2 Entrenamiento de la red neuronal

Una vez que se tiene la información de los tres recorridos del robot móvil, dicha información se ordena en una matriz de datos denominada X, la cual tiene tres columnas y 230 renglones. La primera columna representa la información del motor 1, la segunda columna, la del motor 2 y la tercera columna la información de los sensores táctiles y fotorreflectivos.

También se forma un vector llamado d con la información que se desea que el robot recorra con el motor 2.

A continuación se presenta el listado del programa que determina los pesos de la red neuronal, los centros y las desviaciones estándar de las funciones gaussianas. Dicho programa se realizó en MATLAB.

```

RR=[X d];      /* se forma una sola matriz RR con X y d */
T=size(RR);
n=T(1,2);      /* n es el número de columnas de la matriz RR */
for l=1:n
    RR(:,l)=(RR(:,l)-min(RR(:,l)))/(max(RR(:,l))-min(RR(:,l))); /* los datos de la matriz RR se normalizan entre
                                                                0 y 1. Cada columna se toma por separado,
                                                                donde el cero representa el valor mínimo en
                                                                dicha columna y uno el valor máximo */
end
X(:,1:(n-1))=RRn(:,1:(n-1)); /* se guardan en la matriz X los datos de dicha matriz normalizados entre cero
                               y uno */
d=RRn(:,n); /* se guardan en el vector d los datos de dicho vector normalizados entre cero y uno */
centros=80; /* la variable centros toma el valor de 80, debido a que se eligieron 80 centros para el
             entrenamiento */
[C,U,obj]=fcm(X,centros); /* se obtienen los centros mediante fuzzy c-means y dichos centros se
                           guardan en la variable C */
valores= T(1,1); /* la variable valores toma el número de renglones de la matriz RR, 230 */
tmp=zeros(1,valores+1); /* se forma un vector de ceros llamado tmp de 231 renglones */

for i=1:centros /* en el siguiente ciclo se calculan las desviaciones estándar17 y se colocan en
                la variable sigmas */
    for j=1:3
        for k=1:valores
            tmp(k+1)=(X(k,j)-C(i,j))^2+tmp(k);
        end
        sigmas(i,j)=sqrt(tmp(valores+1)/(valores-1));
    end
end

v_tmp_tot=ones(1,centros); /* se forma un vector de unos llamado v_tmp_tot de 80 renglones */

for k=1:valores /* en este ciclo se calculan las funciones gaussianas18 y el resultado se coloca
                 en P */
    ent_act(1,:)=X(k,:);
    for i=1:centros
        for j=1:3
            v_tmp(j)=exp(-0.5 * (( ent_act(1,j) - C(i,j) ) / sigmas(i,j) )^2 );
        end
    end
end

```

<sup>17</sup> El cálculo de las desviaciones estándar se realiza con la ecuación (3.21)

<sup>18</sup> El cálculo de las funciones gaussianas se hace con la ecuación (3.2)

```

    end
    v_tmp_tot(i)=v_tmp(1)*v_tmp(2)*v_tmp(3);
    end
    P(k,:)=v_tmp_tot;
end

T=size(P);
N=T(1,1);          /* N es igual al número de renglones de P */
M=T(1,2);          /* M igual al número de columnas de P */

A=zeros(M,M);     /* A es una matriz de ceros, con M renglones y M columnas */
for l = 1:M,       /* coloca unos en la diagonal principal de la matriz A. La matriz A resultante es
    A(l,l) = 1;    /* una matriz triangular superior */
end

W(:,1)=P(:,1);    /* para el cálculo de la matriz W, el primer paso es iniciar con la primera
                    /* columna de la matriz W igual a la primera columna de la matriz P */
tempsum=zeros(N,1); /* se crea una vector de ceros llamado tempsum de N renglones */
for K = 2:M,       /* en este ciclo se calcula la matriz W y los elementos del triángulo superior de
                    /* la matriz A */
    for l = 1:(K-1),
        A(l,K)=( W(:,l)' * P(:,K) ) / ( W(:,l)' * W(:,l) );
    end
    for l=1:(K-1),
        tempsum=tempsum+A(l,K)*W(:,l);
    end
    W(:,K)=P(:,K)-tempsum;
    tempsum=zeros(N,1);
end

g=inv( W' * W ) * W' * d; /* se determina el vector g */
teta= inv( A ) * g;      /* se obtiene la matriz de pesos teta */
error=d- P * teta;      /* se calcula el error */

```

## Apéndice E

### Descripción de la conexión de puertos

Como se mencionó en el apéndice B, los puertos disponibles en el sistema mínimo para el microcontrolador son cuatro: A, D, E y G. De estos cuatro puertos, el robot móvil hace uso de tres de ellos; uno para el control de los motores, y los otros dos para la lectura de las señales sensadas. El puerto E (el cual es solamente un puerto de entrada), se dejó libre para aplicaciones posteriores.

#### E.1 Puerto A

El puerto A es un puerto de ocho líneas bidireccionales que comparte funciones con el sistema temporizador. Esta es la razón por la cual se utiliza este puerto para el control de los motores.

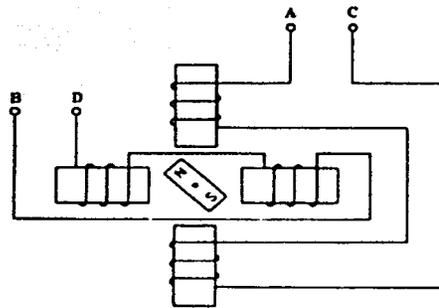
Aunque el puerto A es bidireccional, sólo se utiliza como puerto de salida de datos, donde las primeras cuatro líneas del puerto (A0, A1, A2, y A3) se utilizan para controlar al motor de pasos y las cuatro últimas (A4, A5, A6 y A7) se ocupan para el control de los motores de corriente directa. La razón para utilizar cuatro líneas en vez de dos para controlar el motor de pasos, es que teniendo cuatro líneas se tiene la posibilidad de hacer girar al motor de pasos cada medio paso<sup>19</sup>.

En la Fig. E.1 se presenta un diagrama del motor de pasos<sup>20</sup>, en donde se asignan las letras A, B, C y D a cada una de las entradas del motor de pasos, dicha asignación se conserva durante todo el trabajo.

---

<sup>19</sup> En la sección C.2.2 se da una explicación más detallada del giro del motor cada medio paso y cada paso completo.

<sup>20</sup> El motor de pasos usado en la construcción del robot móvil es un motor bipolar.



**Fig. E.1 Diagrama del motor de pasos [Silva, 1989].**

En la tabla E.1 se muestra la secuencia que se debe cumplir para hacer girar al motor de pasos tanto en sentido dextrógiro como en sentido levógiro. Y en la tabla E.2 se puede observar la forma en la cual están conectadas cada una de las entradas del motor de pasos a las líneas del puerto A.

Sentido dextrógiro				Sentido levógiro			
A	B	C	D	A	B	C	D
1	0	0	0	1	0	0	0
1	1	0	0	1	0	0	1
0	1	0	0	0	0	0	1
0	1	1	0	0	0	1	1
0	0	1	0	0	0	1	0
0	0	1	1	0	1	1	0
0	0	0	1	0	1	0	0
1	0	0	1	1	1	0	0

**Tabla E.1 Secuencia para hacer girar al motor de pasos en sentido dextrógiro o levógiro.**

Línea del puerto	Entrada del motor de pasos
A0	C
A1	A
A2	D
A3	B

**Tabla E.2 Forma en la cual se conectaron las líneas del puerto A al motor de pasos.**

En la Fig. E.2 se muestra la ubicación de los motores de corriente directa y del motor de pasos en el robot móvil. Además, se muestran también los interruptores ópticos, que se encargan de sensar el desplazamiento de los motores de corriente directa.

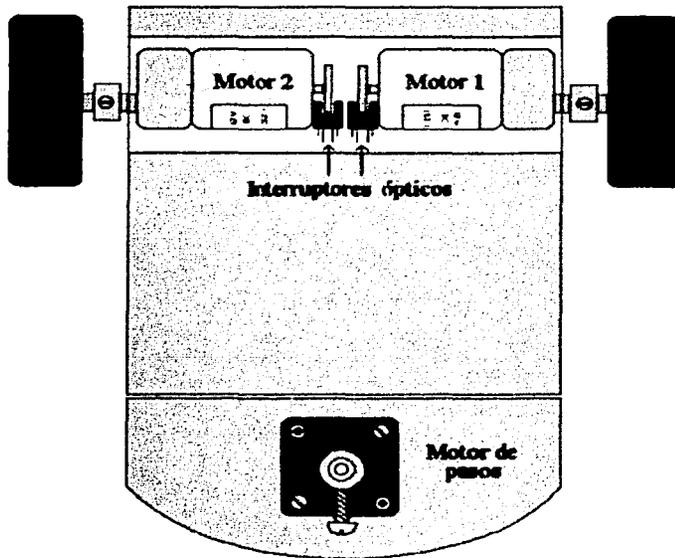


Fig. E.2 Ubicación de los motores en el robot móvil.

Para controlar a cada motor de corriente directa se necesitan dos líneas de datos, una para controlar el sentido de giro del motor y otra para controlar la velocidad con la que se mueve el motor. En realidad lo que se hace con la línea que controla la velocidad del motor es variar el tiempo de encendido del manejador del motor<sup>21</sup>.

Al variar el tiempo que permanece encendido el manejador del motor lo que produce en el motor de corriente directa es una variación en su velocidad angular. Esta es la razón por la cual se ocupó una modulación de ancho de pulso para controlar el tiempo de encendido y apagado del manejador. La modulación de ancho de pulso se generó haciendo uso del sistema temporizador del microcontrolador.

El puerto A cuenta con funciones lógicas para el temporizador y para propósito general de entrada y salida. Las líneas A3, A2, A1 y A0 en el sistema temporizador pueden funcionar como entradas de captura, las cuales pueden detectar el tiempo en el que ocurre un evento externo. La línea A7 puede ser usada como entrada del acumulador de pulsos; además, las líneas A3, A4, A5, A6 y A7 en el sistema temporizador pueden funcionar también como comparadores de salida. Sin embargo, hay que recalcar que las ocho líneas del puerto A pueden ser utilizadas como líneas de propósito general de entrada y salida, y cuando algunas de ellas estén siendo utilizadas por el sistema temporizador las otras siguen siendo de propósito general.

El sistema temporizador del microcontrolador cuenta con un contador de corrimiento libre de 16 bits, el cual se inicia cuando el microcontrolador es reinicializado y comienza a

<sup>21</sup> En la sección 3.4 se presenta una explicación sobre el funcionamiento de este manejador.

incrementarse desde 0000<sub>h</sub> hasta su máximo valor de conteo FFFF<sub>h</sub><sup>22</sup>. Cuando el contador alcanza su valor máximo, éste vuelve a empezar en 0000<sub>h</sub>, y además activa una bandera de sobre flujo. Las funciones de comparación de salida se usan para programar una acción que ocurre en un tiempo específico, es decir, cuando el contador de corrimiento libre de 16 bits alcanza un valor especificado. Cada una de las cinco líneas del subsistema de comparación de salida cuenta con un registro comparador de 16 bits. El valor en el registro de comparación es comparado con el valor del contador de corrimiento libre en cada ciclo de reloj, cuando éstos son iguales se activa una bandera de estado de comparación de salida.

Para producir un pulso de una duración específica, se escribe un valor en el registro de comparación de salida que represente el tiempo en el cual el pulso cambia de estado. El circuito de comparación de salida puede ser configurado para seleccionar la salida apropiada tanto a un nivel alto ("1" lógico) como a un nivel bajo ("0" lógico), dependiendo de la polaridad en que el pulso haya sido producido. Cuando la bandera de sobre flujo se activa (es decir, cuando el contador de corrimiento libre pasa de FFFF<sub>h</sub> a 0000<sub>h</sub>) la línea de salida regresa a su nivel inactivo [Manual Motorola, 1995].

La duración de cada ciclo de la señal que se genera mediante el subsistema de comparación de salida se calcula mediante la siguiente fórmula:

$$\text{Duración del ciclo} = (65535) \times \left( \frac{f_{\text{xtal}}}{4} \right)^{-1} \quad [\text{s}] \quad (\text{E.1})$$

donde  $f_{\text{xtal}}$  es la frecuencia del cristal del microcontrolador y 65535 corresponde al valor máximo del contador de corrimiento libre. Para un cristal de 4.9152 MHz, la duración de cada ciclo es 53.33 ms. Para calcular la duración del nivel alto del ciclo (asumiendo que está polarizado para iniciar en un nivel alto y que al activarse la bandera de estado de comparación de salida, se cambia a un nivel bajo), se emplea la ecuación :

$$\text{Duración del nivel alto del ciclo} = (D_{\text{comparador}}) \times \left( \frac{f_{\text{xtal}}}{4} \right)^{-1} \quad [\text{s}] \quad (\text{E.2})$$

donde  $D_{\text{comparador}}$  es el valor en el registro de comparación de salida. Por ejemplo, si en este registro se coloca el dato 8000<sub>h</sub> y la frecuencia del cristal igual a 4.9152 MHz, la duración del nivel alto del ciclo es de 26.67 ms y la duración del nivel bajo del ciclo es de 26.67; por lo tanto, se tiene un ciclo de trabajo al 50%. De esta forma, al modificar el valor en el registro de comparación de salida lo que varía es el ciclo de trabajo de la señal.

Las líneas del puerto que se ocupan bajo el subsistema del comparador de salida son la A4 y la A6, las cuales controlan la velocidad angular de los motores de corriente directa. Las líneas A5 y A7 se ocupan como líneas de salida, las cuales controlan el sentido de giro de los motores de corriente directa.

<sup>22</sup> Este número equivale a 65535 en decimal.

En la tabla E.3 se muestra la asignación de las líneas del puerto A correspondientes al control de los motores de corriente directa.

Línea del puerto	Descripción
A4	Control velocidad motor 2
A5	Control del sentido de giro del motor 2
A6	Control velocidad motor 1
A7	Control del sentido de giro del motor 1

**Tabla E.3 Líneas de control para los motores de corriente directa.**

## E.2 Puerto G

El puerto G es un puerto bidireccional, en el cual, cada vez que el microcontrolador se reinicia, sus líneas son configuradas como entradas de alta impedancia, excepto en el modo *Expanded*, donde una reinicialización del microcontrolador provoca que la línea G7 se comporte como salida de la señal *CSPROG* [Manual Motorola, 1995]. Este puerto también tiene funciones alternativas, por ejemplo las líneas G4, G5, G6 y G7 pueden ocuparse como las salidas de selección de chip.

El puerto G en el robot móvil se ocupa como un puerto de entrada de datos, por el cual ingresa la información proveniente de los sensores fotorreflejos y de los sensores táctiles. Las primeras cuatro líneas menos significativas del puerto (G0, G1, G2 y G3), se ocupan para las señales provenientes de los sensores fotorreflejos, y las cuatro líneas más significativas (G4, G5, G6 y G7), se utilizan para las señales provenientes de los sensores táctiles. En la tabla E.4 se presenta la asignación de cada línea del puerto G. Cabe recordar que las señales provenientes de los sensores antes de ingresar al microcontrolador pasan por un proceso de acondicionamiento<sup>23</sup>.

Línea del puerto	Sensor
G0	Sensor fotorreflejo 1
G1	Sensor fotorreflejo 2
G2	Sensor fotorreflejo 3
G3	Sensor fotorreflejo 4
G4	Sensor táctil 1
G5	Sensor táctil 2
G6	Sensor táctil 3
G7	Sensor táctil 4

**Tabla E.4 Asignación de las líneas del puerto G para la etapa de sensado.**

En la Fig. E.3 se puede observar la ubicación de cada uno de los sensores fotorreflejos en el robot móvil. Como se puede observar, los sensores fotorreflejos 1 y 2 se encuentran en la parte frontal del robot y los sensores fotorreflejos 3 y 4 en la parte posterior.

<sup>23</sup> Para más información sobre este acondicionamiento, revisar la sección 3.4.3.

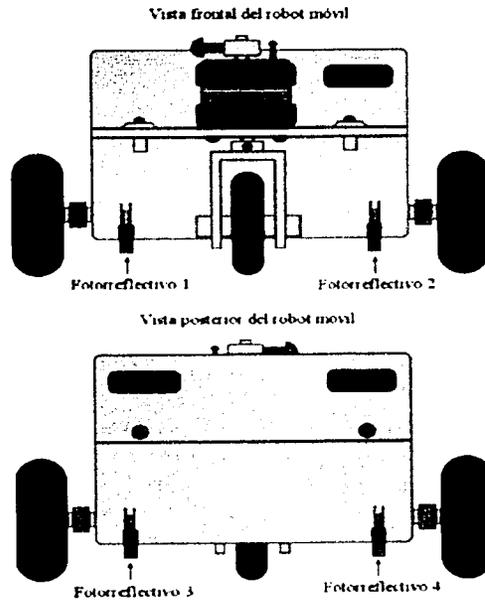


Fig. E.3 Ubicación de los sensores fotorreflexivos.

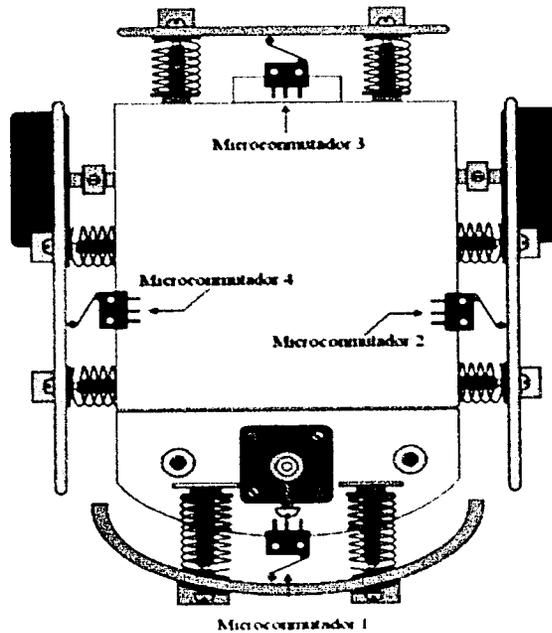


Fig. E.4 Ubicación de los sensores táctiles.

En la Fig. E.4 se presentan los microconmutadores que forman parte de los sensores táctiles, en esta figura se puede observar el número asignado a cada microconmutador, el cual también corresponde con el número de sensor táctil.

### E.3 Puerto D

El puerto D es un puerto que en cualquier modo de operación puede usar sus seis líneas para propósito general de entrada y salida, o bien, con los subsistemas SCI (interfaz de comunicación serial síncrona) o SPI (interfaz de comunicación serial asíncrona) [Manual Motorola, 1995]. Cada vez que el microcontrolador, se reinicia las seis líneas de este puerto son configuradas como entradas de alta impedancia.

Las líneas D0 y D1 son ocupadas para la recepción y transmisión de información bajo el subsistema de interfaz de comunicación serial asíncrona. Cuando se hace uso de las líneas D0 y D1 bajo el subsistema SCI, las cuatro líneas restantes del puerto pueden ser utilizadas como líneas de propósito general.

Las líneas D4 y D5 son utilizadas en el robot móvil como líneas de entrada de información de los interruptores ópticos, las cuales se encuentran en los motores de corriente directa con la finalidad de sensar el desplazamiento de éstos<sup>24</sup>. En la tabla E.5 se muestran las líneas del puerto D que son ocupadas en el robot móvil, así como su asignación.

Línea del puerto	Descripción
D0	Recepción serial asíncrona
D1	Transmisión serial asíncrona
D4	Sensor del motor 1
D5	Sensor del motor 2

**Tabla E.5 Asignación de las líneas del puerto D.**

<sup>24</sup> Para mayor información sobre la forma en la cual están ubicados estos sensores referirse al capítulo 3.4.

# Bibliografía

## Artículo y textos

- Acarnley, P.; "Stepping Motors: A guide to modern theory and practice"; P. peregrinus on behalf of the institution of electrical engineers, Inglaterra, 1984.
- Anaya R.; "Diseño y construcción de robots móviles utilizando microcontroladores"; UNAM, 1998.
- Chen, S., Billings, S. A., Luo, W.; "Orthogonal least squares and their application to non-linear system identification"; Int. J. Control, Vol. 50, No. 5, pp. 1873-1896, 1989.
- Chen S., Cowan C. F. N., and Grand P. M.; "Orthogonal Least Squares Learning Algorithm for Radial Basis Function Networks"; IEEE Transactions on Neural Networks, Vol. 2, No. 2, pp. 302-309, 1991.
- Cox, I., Gordon, W.; "Autonomous Robot Vehicles"; Springer-Verlag, Estados Unidos, 1990.
- Fauset, L.; "Fundamentals of Neural Networks. Architectures, algorithms and applications"; Prentice-Hall, Inc., 1994.
- Feher, K.; "Wireless Digital Communications. Modulation & Spread Spectrum Applications"; Prentice-Hall, Estados Unidos, 1995.
- Heicht-Niesen, R. "Neurocomputing: Picking the Human Brain". IEEE Spectrum, 25, pp. 36-41, Marzo, 1988. Reimpreso en el texto "Artificial Neural Networks: Theoretical Concepts"; V. Vemuri; IEEE Computer Society Series, 1988.
- Hernández, E., Vázquez, G.; "Interfaz 3D para controlar y simular robots móviles usando realidad virtual"; UNAM, 2002.
- Hilera, J., Martínez, V.; "Redes Neuronales Artificiales. Fundamentos, modelos y aplicaciones"; Addison Wesley Iberoamericana, España, 1995.
- Höppner, F., *et al.*; "Fuzzy Cluster Analysis. Methods for classification, data analysis and image recognition"; John Wiley & Sons, Estados Unidos, 1992.
- Horowitz, P., Hild, W.; "The Art of Electronics"; Cambridge University Press, Segunda Edición, Estados Unidos, 2001.

Kohonen, K.; "An introduction to Neural Computing". Neural Networks. Vol. 1, 1988.

Pahlavan, K., Levesque, A.; "Wireless Information Networks" (Wiley Series in Telecommunications and Signal Processing); John Wiley & Sons, Inc., Estados Unidos, 1994.

Powel, M.; "Radial Basis Functions for Multivariable Interpolation: A review"; Algorithms for approximation, pp. 143-167, 1987.

Silva, C. W. de; "Control Sensors and Actuators"; Prentice Hall, Estados Unidos, 1989.

Tomasi, W.; "Electronic Communications Systems. Fundamentals Through Advanced"; Prentice-Hall, Estados Unidos, 1994.

## Catálogos

Jameco Electronics; Estados Unidos, Noviembre 2000 - Enero 2001.

## Congresos

Mobile Robots. Conference on Control Applications & International Symposium on Intelligent Control; México, 2001.

Robotics and Mechatronics, Automotive Vehicles. European Control Conference; Portugal, 2001.

Volume B: Robotics Automation. International Federation of Automatic Control World Congress; Beijing, P. R. China, 1999.

## Manuales

EVBU; "M68HC11EVBU. Universal Evaluation Board. User's manual"; Motorola, 1997.

ICC11; "ImageCraft 68HC11 C compiler and development environment. User's manual"; ImageCraft, 1997.

Motorola; "IIC11 MC68HC11E1, Technical Data"; Motorola, Inc., 1995.

Pcbug11; "MC68HC11 Pcbug11. User's manual"; Motorola, 1992.

RFM; "Virtual Wire Development Kit Manual"; DR 1007-DK, RFM, 1996.

## Referencias electrónicas

- Bohlke, Dennis L.; "Super Tech and Associates";  
<http://www.super-tech.com/root/motors.htm>; 2001.
- Fairchild; "H21A1";  
<http://www.fairchildsemi.com/ds/H2/H21A1.pdf>, 2002.
- Jones, Douglas W.; "Basic Stepping Motor Control Circuits";  
<http://www.cs.uiowa.edu/~jones/step/circuits.html>, 2001.
- Jones, Douglas W.; "Basic Stepping Motor Control Circuits";  
<http://www.cs.uiowa.edu/~jones/step/types.html#unipolar>, 2001.
- Motorola; "68HC11F1 : Microcontroller";  
[http://c-www.motorola.com/webapp/sps/site/prod\\_summary.jsp?code=68HC11F1&nodeId=03M0ym4t3ZGM98635](http://c-www.motorola.com/webapp/sps/site/prod_summary.jsp?code=68HC11F1&nodeId=03M0ym4t3ZGM98635), 2001.
- Omron; "Subminiature Basic Switch";  
[http://www.omron.fr/europe/www/cncweb.nsf/d9d48916a4b384d8c12565050025e194/8b4ce11c090750e5412567070057984c/\\$FILE/Ss.pdf](http://www.omron.fr/europe/www/cncweb.nsf/d9d48916a4b384d8c12565050025e194/8b4ce11c090750e5412567070057984c/$FILE/Ss.pdf), 2002.
- RFM; "Development Kit";  
<http://www.rfm.com/products/vwire.htm#dkits>, 2001.