



ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO

FACULTAD DE INFORMÁTICA Y ELECTRÓNICA

ESCUELA DE INGENIERÍA ELECTRÓNICA

“DISEÑO E IMPLEMENTACIÓN DE UN ROBOT
JUGADOR DE FUTBOL CATEGORÍA MIROSOT
CONTROLADO POR RADIO FRECUENCIA”

TESIS DE GRADO
PREVIO A LA OBTENCIÓN DE TÍTULO DE:

INGENIERO EN ELECTRÓNICA Y
COMPUTACIÓN

PRESENTADO POR:

VALENCIA ASTUDILLO CRISTHIAN ALEX

RIOBAMBA – ECUADOR

2008

AGRADECIMIENTO

A Dios por concederme la vida e iluminar el camino a seguir para ser una persona útil, a mis padres y a la sociedad.

A mis padres por sus sabios consejos y apoyo incondicional para concluir con éxitos mis estudios.

A la Universidad por haberme permitido compartir y experimentar procesos pedagógicos científicos que conducen al desarrollo de los pueblos

DEDICATORIA

A mis padres pilares fundamentales para
culminar con éxito mi carrera profesional

NOMBRE	FIRMA	FECHA
Dr. Ms.c Romeo Rodríguez DECANO FACULTAD DE INFORMÁTICA Y ELECTRÓNICA
Ing. Paúl Romero DIRECTOR DE LA ESCUELA DE INGENIERÍA ELECTRÓNICA
Ing. Paul Romero DIRECTOR DE TESIS
Ing. Pedro Infante MIEMBRO DEL TRIBUNAL
Tlgo. Carlos Rodríguez DIRECTOR DPTO. DOCUMENTACION
✓ NOTA DE LA TESIS	

Yo CRISTHIAN ALEX VALENCIA ASTUDILLO, soy el responsable de las ideas, doctrinas y resultados expuestos en esta: Tesis, y el patrimonio intelectual de la misma pertenecen a la Escuela Superior Politécnica de Chimborazo.

Cristhian Valencia

ÍNDICE DE ABREVIATURAS

RX	Receptor de Datos
TX	Transmisor de Datos
RF	Radio Frecuencia
ASK	Modulación por desplazamiento de amplitud
dBm	Relación señal ruido
MHz	Mega Herzio
RS-232	Protocolo de comunicación serial
COM	Puerto de Comunicaciones
Driver	Controlador

INDICE DE FIGURAS

Figura I.1	Imagen de un Partido de la categoría F-2000 de la RoboCup	14
Figura I.2	Robot Mirobot	16
Figura I.3	Configuración de los distintos elementos en un partido de la categoría Mirobot	17
Figura II.4	Arquitectura Harvard	31
Figura II.5	Circuito de Reset	34
Figura II.6	Pulsadores Activado por Nivel Bajo	35
Figura II.7	Pulsadores Activado por Nivel Alto	35
Figura II.8	Interruptores Activado por nivel Bajo	36
Figura II.9	Interruptores Activado por nivel Alto	36
Figura II.10	Circuitos Antirrebotes	37
Figura II.11	Oscilador tipo RC.	38
Figura II.12	Oscilador tipo XT	39
Figura II.13	Oscilador tipo XT Sencillo	39
Figura II.14	PIC 16F628	40
Figura II.15	Pines no Utilizados	43
Figura II.16	Mapa de Registros del Pic 16F628	44
Figura II.17	Regulador Fijo	54
Figura II.18	Fuente de Alimentación Fija Completa	55
Figura II.19	Circuito Puente H	59
Figura II.20	Circuito de Avance	60
Figura II.21	Circuito de Retroceso	60
Figura II.22	Diagrama Interno De 1293	62
Figura II.23	Modulación ASK	66
Figura II.24	Modulo Transmisor	66
Figura II.25	Modulo Receptor	67
Figura III.26	Diagrama de bloques del transmisor	69
Figura III.27	Circuito transmisor	70
Figura III.28	Icono componente MS COMM	71
Figura III.29	Aplicación	72
Figura III.30	Selección de puertos	72
Figura III.31	Selector de robot	72
Figura III.32	Constantes de movimiento	73
Figura III.33	Dirección de movimiento	73
Figura III.34	Diagrama de bloque del receptor	74
Figura III.35	Circuito controlador del robot	74
Figura III.36	Diagrama de flujo de la unidad decodificadora	75
Figura III.37	Diagrama de Flujo del control de velocidad	77
Figura III.38	Modulación PWM	77
Figura III.39	Paredes del robot	79
Figura III.40	Rueda del Robot	79
Figura III.41	Rueda con el engrane	79
Figura III.42	PBC del transmisor	80
Figura III.43	PBC del controlador del robot 1° parte	80
Figura III.44	PBC del controlador del robot 2° parte	81

INDICE DE TABLAS

Tabla I	Registro de configuración del PIC	38
Tabla II	Capacidad de Corriente en los puertos del Pic	44
Tabla III	Registro Status del PIC 16F628	46
Tabla IV	Registro PCON del PIC 16F628	48
Tabla V	Código de Ejemplo	49
Tabla VI	Encabezado del código	50
Tabla VII	Asignación de memoria	50
Tabla VIII	Configuración de Puerto I/O	50
Tabla XI	Reguladores Fijos	53
Tabla X	Tabla de verdad interfaz PUENTE H	58
Tabla XI	Bandas del espectro de radio	63
Tabla XII	Trama enviada por la PC	70
Tabla XIII	Códigos de movimientos	73
Tabla XIV	Asignación de velocidades	76

INDICE GENERAL

INTRODUCCIÓN	11
CAPITULO I	
1 FUTBOL ROBÓTICO	12
1.1 Introducción	12
1.2 Organizaciones De Fútbol Robótico	13
1.3 Categoría Mirosot	15
1.3.1 Reglamento Mirosot	18
Capítulo III	
2. COMPONENTES ELECTRÓNICOS PARA EL DISEÑO DEL ROBOT	29
2.1 Microcontrolador	29
2.1.1 Aplicaciones de los Microcontroladores	30
2.1.2 Características de los Microcontroladores PIC	31
2.1.3 Arquitectura de los Microcontroladores PIC	32
2.1.3.1 Memoria de Programa	33
2.1.3.2 Contador de Programa	33
2.1.3.3 La pila	33
2.1.3.4 Reinicialización (Reset)	34
2.1.3.5 Periféricos Digitales de Entrada	35
2.1.3.6 Tipos Osciladores	37
2.1.4 PIC 16F628	40
2.1.4.1 Características Principales	40
2.1.4.2 Arquitectura Hardware	41
2.1.4.3 Encapsulado DIP (Dual In line Pin) - PIC16F628	42
2.1.4.4 Capacidad de corriente de los puertos	43
2.1.4.5. Mapa de la distribución de memoria de datos	44
2.1.4.6 Configuración de los Puertos del Pic	45
2.1.4.7 Registros importante del PIC 16F628	46
2.1.4.8 Ejemplos de Programación	48
2.2 Reguladores de tensión	51
2.2.1 Historia de los reguladores monolíticos de tensión	51
2.2.2 Reguladores integrados de Voltaje	52
2.2.2.1 Reguladores Fijos	53
2.3 Interfaz Puente H	57
2.3.1 Uso de la Interfaz Puente H	58
2.3.2 Consideraciones de Corriente	58
2.3.3 Funcionamiento	59
2.3.4 L293D	60
2.4 Radio Frecuencia	62
2.4.1 Uso de la radio Frecuencia	62
2.4.2 Modulación ASK	65
2.4.3 Módulos de RF	66
2.4.3.1 Modulo TPL 434 (TX)	66
2.4.3.2 Modulo RPL (RX)	67
CAPITULO III	
3 DISEÑO E IMPLEMENTACIÓN DEL ROBOT	69
3.1 Implementación del Transmisión	69
3.1.1 Implementación de la Aplicación	70

3.1.1.1 Acceso al Puerto Serial a través de V Basic	71
3.1.2 Iniciación de la aplicación	71
3.2 Diseño e Implementación del circuito controlador del robot	74
3.2.1 Implementación del receptor	75
3.2.2 Diseño de la unidad decodificadora	75
3.2.3 Diseño del control de velocidad	76
3.2.3.1 Modulación de amplitud de pulso (PWM)	76
3.2.3.2 Duty cycle	78
3.2.3.3 Parámetros Importantes	78
3.2.3.4 Aplicaciones	78
3.3 Construcción del robot	78
3.4 Implementación de las plaquetas	80
3.5 Pruebas y resultados	81
Conclusiones	82
Recomendaciones	83
Bibliografía	84
Resumen	85
Anexos	87

INTRODUCCIÓN

Si bien no puede negarse que el fútbol juega un rol prominente en la sociedad de hoy, no es común pensar en él como un juego que pueda ser jugado por robots. El fútbol robótico es una actividad que se ha venido realizando desde hace un tiempo atrás en algunos países del mundo.

Los robots para esta actividad son comercializados; en nuestro país no existe una empresa que se encargue de la venta, menos aun de la fabricación de robots de fútbol categoría MIROSOT; por lo cual se decide escoger como tema de tesis de grado “DISEÑO E IMPLEMENTACION DE UN ROBOT JUGADOR DE FUTBOL CATEGORIA MIROSOT CONTROLADO POR RADIO FRECUENCIA”

El presente proyecto muestra la forma de construcción de un robot para esta actividad, utilizando únicamente implementos electrónicos existentes en el país y poder así reducir costos; para que posteriormente puedan ser implementados.

CAPITULO I

1 FÚTBOL ROBÓTICO

1.1 Introducción

El fútbol robótico es un excelente banco de pruebas para la investigación en el campo de los sistemas multiagente, el juego de fútbol con robots se diferencia de los otros sistemas, puesto que los jugadores de un equipo deben cooperar entre sí, considerando además, que su ambiente de trabajo es continuamente alterado tanto por las acciones de los robots de su equipo como por las del equipo oponente. Por estas razones, los algoritmos de control e inteligencia artificial de este tipo de sistemas multi agente deben solucionar problemas que van desde niveles bajos relacionados con las leyes de la cinemática y la dinámica que gobiernan los movimientos de los robots, hasta la implementación de estrategias de alto nivel para competir con otros robots.

Teniendo presente que el sistema multiagente está compuesto por agentes homogéneos reactivos puesto que todos los robots reciben las mismas entradas, y son capaces de realizar las mismas acciones que dependen de las posiciones de los compañeros y de la pelota, no comunicativos debido a que no existe comunicación entre ellos, resulta más conveniente aplicar un método de navegación autónomo reactivo para controlar los movimientos del robot.

FIRA y RoboCup son emprendimientos internacionales dedicados a la organización de campeonatos de fútbol de robots. Ambos tienen como principal objetivo acercar la ciencia y la tecnología de los robots a las nuevas generaciones, a través del fútbol de robots.

1.2 Organizaciones de Fútbol Robótico

La Robot World Cup Initiative y la Federation of International Robot-soccer Association tienen como principal objetivo el promover el desarrollo en temas de robótica a través de un problema estándar en donde una amplia gama de tecnologías pueden ser aplicadas e integradas. Dicho problema estándar es el fútbol, el cual es usado simplemente como excusa en la búsqueda de producir innovaciones que se apliquen en la resolución de importantes problemas sociales e industriales. Un ejemplo de esto es el proyecto RoboCupRescue [RoboCup2004b] de la RoboCup, cuyo propósito es precisamente, la aplicación de tecnologías desarrolladas para búsqueda y rescate en desastres a gran escala. Dentro de dichas tecnologías se incluyen: agentes inteligentes [Weiss2000], colaboración multi-agente, razonamiento en tiempo real, sensado, estrategia y control, etc.

La RoboCup por su parte, es famosa por tener la meta de “para el año 2050 desarrollar un equipo completo de robots humanoides autónomos que puedan

derrotar al reciente equipo humano campeón del mundo bajo las reglas de la F.I.F.A. [FIFA2004]”. Si bien esto suena sumamente ambicioso, podemos establecer una relación con lo que alguna vez John F. Kennedy dijo a principio de la década del 60: “para el final de esta década pondremos un hombre en la luna y lo traeremos sano y salvo de regreso”. En realidad, el verdadero objetivo no era mandar un hombre a la luna y traerlo de vuelta, sino que era el lograr el mayor avance tecnológico posible en el intento, aún cuando aquello tal vez nunca llegase a alcanzarse [Chen2000]. Justamente, es por esto que la RoboCup, al igual que la FIRA, ha introducido varias categorías o ligas cada una de las cuales hace hincapié en diferentes aspectos del problema general. Desde el año 1997, año en que se llevó a cabo la primera copa mundial (ver figura I.1) organizada por la RoboCup, investigadores de todo el mundo han participado de dicha competencia. Actualmente, sus principales ligas son:

- Small Size Robot League (F-180)
- Middle Size Robot League (F-2000)
- Sony Legget Size Robot League
- RobotCup Humanoid League
- Simulation League



Figura I.1 Imagen de un Partido de la categoría F-2000 de la RoboCup

Categorías FIRA

- HuroSot
- KheperaSot
- MiroSot
- NaroSot
- QuadroSot
- RoboSot
- SimuroSot

1.3 Categoría MiroSot

La categoría MiroSot es una categoría en la que compiten robots reales, mientras que la SimuroSot, es una categoría simulada que, supuestamente, está basada en la mencionada categoría real. Así, nosotros estamos interesados en lo que se denomina la Middle League, que tanto en la categoría real como en la simulada, consta de cinco jugadores por equipo. A pesar de que cada una de estas dos ligas (la Middle League MiroSot y la Middle League SimuroSot) posee reglamentos independientes, éstos son similares en lo que al juego en sí se refiere. Por eso, en lo que queda del capítulo no se hará distinción alguna entre las categorías a menos que se indique explícitamente.

Como se expuso anteriormente, cada equipo está formado por cinco jugadores, cuyas dimensiones máximas son de 7,5cm×7,5cm×7,5cm. En la gran mayoría de los casos los robots jugadores son de forma cúbica (7,5cm de lado), a excepción de una pequeña ranura que pueden poseer en su parte frontal para un mejor dominio de la pelota. Así mismo, en la categoría real está permitido dotar a los robots de brazos, piernas, etc., lo que es muy poco común ya que éstos deben estar contenidos dentro de los límites del robot, complicando así su construcción debido al poco espacio disponible.

Finalmente, para el movimiento de los robots, se cuenta con dos ruedas ubicadas en los costados del mismo (ver Figura I.2), las cuales son controladas de manera independiente.

La Figura I.2 imagen de un robot de la categoría MiroSot.

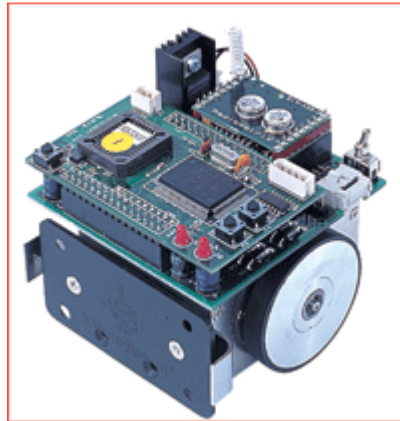


Figura I.2 Robot MiroSot

El campo de juego mide 220cm×180cm con arcos de 40cm de ancho, y está delimitado por paredes verticales de 5cm, las cuales restringen a la pelota para que esta permanezca dentro de la cancha. El juego se desarrolla con una pelota de golf de color naranja de 2,1cm de radio aproximadamente.

En la categoría real, cada equipo controla sus robots mediante una computadora, la cual a través de señales de radiofrecuencia o infrarrojo, le envía comandos a los mismos. En cuanto al censado, está permitida la ubicación de una cámara de video por equipo, colocada más o menos dos metros por encima del terreno de juego (ver Figura I.3). Si bien los robots pueden tener sensores propios, resulta poco práctico debido al pequeño tamaño de los mismos, siendo entonces la cámara de video el único sensor por equipo. En el caso de la categoría simulada, las exactas posiciones y

rotaciones de todos los robots así como también la posición de la pelota, son enviadas directamente como datos a la estrategia

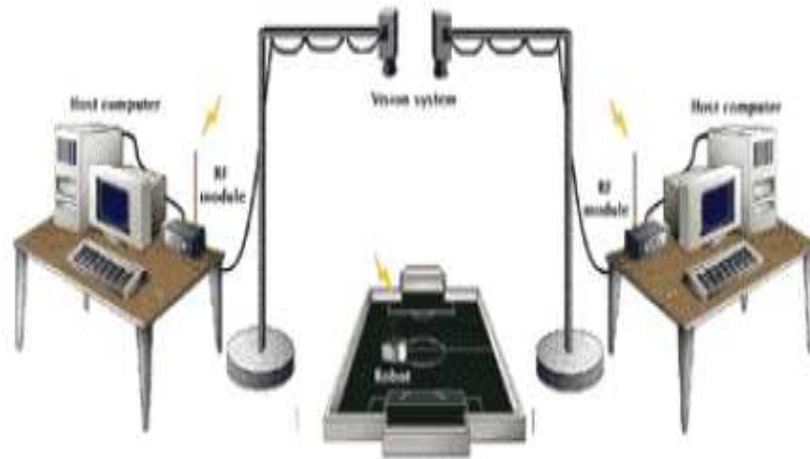


Figura I.3 Configuración de los distintos elementos en un partido de la categoría Mirobot

A los efectos de permitir la identificación de las posiciones y orientaciones de los robots que encuentran en la cancha, cada robot debe poseer en su parte superior una región que incluya parche de $3,5\text{cm} \times 3,5\text{cm}$, del color correspondiente al color del equipo (amarillo o azul). Además, reglamento permite la inclusión de otras regiones en la parte superior del robot a los efectos poderlo identificar individualmente, obviamente esto sujeto a ciertas restricciones. En la categoría simulada, en todo momento se dispone de la identificación individual de cada robot.

En lo referente a las reglas del juego, los partidos constan de dos tiempos de cinco minutos uno, sin considerar el tiempo en que el juego está detenido. En la práctica, esto hace que partidos lleguen a durar cerca de una hora. Además, existen diversas faltas y reglas que corresponden con aquellas del fútbol tradicional, adaptadas por supuesto a las condiciones fútbol de robots. Entre

ellas, las cuales son cobradas por un árbitro humano, se incluyen piques, tiros libre, penales, saques de arco, y saques desde la mitad de la cancha.

1.3.1 Reglamento Mirosot

Ley 1: El Campo y la Pelota

(a) Dimensiones del campo

Una cancha rectangular de madera de color negro (no reflectivo) con tamaño de 150cm X 130cm, con un borde de madera de 5cm alto y 2.5cm de espesor de color blanco. La parte superior del borde de madera que rodea la cancha será de color negro y las caras laterales de ese borde son de color blanco (vista lateral). Un sólido triángulo isósceles de 7cm x 7cm se ubicaran en las cuatro esquinas de la cancha para evitar que la pelota quede acorralada en las esquinas. La textura de la superficie de la cancha será igual a la de una mesa del ping-pong.

(b) Las señales sobre la cancha

El campo de juego será marcado como muestra en el Apéndice 1. El círculo central tendrá un radio de 20cm.

Un pequeño arco que es parte del área de meta será de 20 cm. a lo largo de la línea de meta y 5 cm. perpendicular a él.

Las principales líneas/arcos/circunferencias (la línea central, los bordes del área de meta y el círculo central) serán de color blanco y de 3mm de grosor. La marca de Free ball (Ley 13) la de posición del robot (círculos) se pintaran de color gris.

(c) La portería

La portería será de 40cm de ancho. Postes ni redes serán usados para la portería.

(d) La línea de la meta y área de la meta

La línea de la meta es la línea delante de la meta que es de 40cm de largo.

El área de meta comprenderá el área contenida por el rectángulo (de tamaño de 70cm X 15cm delante de la meta) y la línea de arco adyacente al rectángulo (20cm paralelo a la línea de meta y 5cm perpendicular a él).

(e) La pelota

Una pelota de golf anaranjada se usará como pelota, con 42.7mm diámetro y 46g peso.

(f) La ubicación de la cancha

La cancha estará en un lugar cerrado.

Ley 2: Los Jugadores

(a) El sistema completo

El juego será jugado por dos equipos, cada uno de tres robots. Uno de los robots puede ser el arquero (Ley 2.b.2). Tres miembros humanos del equipo, un "administrador", un "director técnico" y un "entrenador" estarán permitidos en el escenario. Una sola computadora por equipo, principalmente dedicado al procesamiento del sistema de visión y a la identificación de posiciones.

(b) Los robots

El tamaño de cada robot se limitará a 7.5cm X 7.5cm X 7.5cm en tamaño. La altura de la antena de comunicación de RF no será considerada en el tamaño del robot.

La parte superior de los robots no debe ser de color naranja. Un parche que contenga una región de color azul o amarillo, como lo asignen los

organizadores, identificará los robots en un equipo. Todos los robots deben tener (por lo menos) en el parche una región sólida de 3.5cm X 3.5 como identificador de su equipo, ya sea azul o amarillo, que sea visible en su parte superior. El color que identifica a un equipo cambiará de juego en juego, y el parche de colores usado deberá ser intercambiable. Cuando se asigne uno de los 2 colores al equipo (azul o amarillo), los robots no deben contener en el parche ninguno de los colores usados por el equipo oponente.

Nota: Se recomiendan a los equipos preparar un mínimo de 6 parches de color diferentes, Tanto como para el equipo azul como para el amarillo, para la identificación individual de cada robot.

Para habilitar sensores infrarrojos a los costados de cada robot, los cuales deben tener una luz de color, excepto en las regiones necesariamente usadas para la funcionalidad del robot, como por ejemplo, las ruedas y el mecanismo para recoger la bola. Los robots deben llevar uniformes y el tamaño máximo de los robots será limitado a 8cm X 8cm X 8cm.

Un robot dentro de su propia área de la meta (Ley 1.d.) será considerado como el "arquero". Al robot arquero se le permitirá coger o sostener la bola solo cuando este dentro de su propia área de meta.

Cada robot debe ser totalmente independiente, con mecanismos de fuente de energía y de motores contenidos dentro de este. Sólo comunicación inalámbrica se permitirá para cualquier tipo de interacción entre el robot y el computador.

Se permite a los robots dotarlos con brazos, piernas, etc., pero ellos deben obedecer las restricciones de tamaño (Ley 2.b.1) incluso después de todos los accesorios añadidos. Ninguno de los robots, excepto el único designado como

arquero, se le permitirá coger o sostener la bola de tal forma que cubra el 30% de la bola ya sea visto desde la parte superior o desde los costados.

Mientras un fósforo es en marcha, a cuando quiera el árbitro silba el operador humano debe detener todos los robots que usan la comunicación entre los robots y la computadora central.

(c) Substituciones

Dos suplentes serán permitidos mientras el juego esta en marcha. En el medio tiempo, pueden hacerse ilimitadas substituciones. Cuando una substitución se desea hacer mientras el juego esta en marcha, el administrador del equipo interesado debe pedir 'tiempo-fuera' (time-out) para notificar al árbitro, y el árbitro detendrá el juego en un momento apropiado. Al reiniciar el juego, todos los robots y la pelota estarán ubicados en las mismas posiciones que estaban ocupando en el momento que se interrumpió el juego.

(d) Tiempo - fuera (time-out)

El operador humano puede pedir 'tiempo-fuera' para notificar al árbitro. Cada equipo tendrá derecho a dos tiempos-fuera en un juego y cada uno durará dos minutos.

Ley 3: La Información transmisible

El administrador, director técnico o entrenador pueden transmitir ciertos comandos directamente desde la computadora central remota a sus robots. No se permitirá transmitir comandos con señales de reset para detener uno o todos los robots o comandos de reiniciar, sin el permiso del árbitro. Cualquier otra información, como la estrategia de juego, puede ser comunicada a los robots siempre y cuando el juego no este en marcha. El operador humano no

debe controlar directamente el movimiento de sus robots ya sea con un joystick o por medio de un teclado bajo ninguna circunstancia. Mientras un juego este en marcha la computadora central puede enviar cualquier información autónomamente.

Ley 4: El Sistema de Visión

Para identificar los robots y la pelota en la cancha, se puede usar un sistema de visión. La ubicación de la cámara de cada equipo o sistema de sensores debe restringirse a, encima de y dentro de su propia mitad del campo incluido la línea del centro, a fin de que la cámara no tenga que ser movida después del cambio de lado en el medio tiempo. Si ambos equipos desean mantener sus cámaras encima y sobre el círculo central de la cancha, ellos pondrán estos de lado a lado, equidistante a la línea central y tan cerca la una de la otra como sea posible. La ubicación de una cámara en lo alto o sistema de sensores debe estar a 2m de altura o superior.

Ley 5: Duración del Juego

La duración de un juego será dos períodos iguales de 5 minutos cada uno, con un descanso (medio tiempo) de 10 minutos. El cronómetro oficial hará una pausa del reloj durante las substituciones, mientras transporten al robot herido fuera del campo, durante el tiempo-fuera y durante tales situaciones que merezcan ser consideradas correctas acorde con la prudencia del que cronometra el tiempo.

Si un equipo no está listo para reanudar el juego después del medio tiempo, se le otorgaran 5 minutos adicionales. Incluso si después del tiempo adicional

permitido el equipo no está listo para continuar el juego, el equipo será descalificado del juego.

Ley 6: El Comienzo del juego

Antes de comenzar el juego, el color del equipo (azul/amarillo) o la pelota será decidido por el lanzamiento de una moneda. El equipo que gana el lanzamiento de moneda se le permitirá escoger el color que identificará al equipo (azul / amarillo) o la pelota. El equipo que recibe la pelota se le permitirá también escoger la banda portadora de frecuencia.

Al comienzo del juego, el equipo atacante se le permitirá posicionar sus robots libremente en su propia cancha y dentro del círculo del central. Entonces el equipo que se defiende puede colocar sus robots libremente en su propia cancha excepto dentro del círculo del central.

Al principio del primer y segundo tiempo, y después de que un gol haya sido anotado, la pelota debe ser ubicada dentro del círculo central y la pelota debe ser pateada o debe pasarse hacia la propia cancha del equipo que haga el saque. Con la señal del árbitro, el juego empezará y todos los robots pueden moverse libremente.

Al principio del juego o después de que un gol sea anotado, el juego será iniciado o continuado, con las posiciones de los robots descritas en la **Ley 6.2.**

Después del medio tiempo, los equipos tienen que cambiar de lado.

Ley 7: El método de Anotar

(a) El Ganador

Un gol será anotado cuando toda la pelota pasa por encima de la línea de meta. El ganador de un juego se decidirá en base al número de goles anotados.

(b) El Desempate

En caso de un empate después del segundo tiempo, el ganador se decidirá por el esquema de muerte súbita. El juego continuará después de un descanso de 5 minutos, por un período máximo de tres minutos. El equipo que logre anotar el primer gol será declarado como el ganador. Si el empate persiste después de los 3 minutos extras de juego, el ganador se decidirá a través de los tiros penaltis. Cada equipo tomará tres tiros penaltis que difieren de Ley 11 debido a que sólo el que cobra el penal el arquero serán permitidos en el campo de juego. Los arqueros deben permanecer dentro de su área de la meta y las posiciones del que pateo el penal y de la pelota será igual a lo que dispone la Ley 11. Después del silbato del árbitro, el arquero puede salir del área de meta. En caso de que continúen empates después de los tres tiros penaltis, tiros penaltis adicionales serán cobrados uno-por-uno, hasta que el ganador pueda decidirse. Todos los tiros penaltis serán cobrados por un solo robot y comenzarán con el silbato del árbitro. Un tiro penal será finalizado, cuando cualquier de lo siguiente casos ocurran:

El arquero agarre la pelota con sus accesorios (si tuviera) dentro del área de la meta.

Ley 8: Las faltas

Una falta será sancionada en los siguientes casos.

Colisionar con un robot del equipo opuesto, intencionalmente o de otra manera: el árbitro pitará las faltas que directamente afecten al desarrollo del juego o que parezcan causar un potencial daño al robot oponente. Cuando un robot defensor intencionalmente empuja a un robot oponente, un tiro libre se otorgará al equipo opuesto. Está permitido empujar con la pelota a un jugador

oponente que este de espaldas con tal de que el jugador que esta empujando siempre está en contacto con la pelota.

Esta permitido empujar al robot arquero dentro del área de meta, si la pelota está entre el robot que empuja y el arquero. Sin embargo empujar al arquero dentro de la portería junto con el balón no esta permitido. Si un robot atacante empuja al arquero junto con la pelota dentro de la portería o cuando el robot oponente empuja al arquero directamente entonces el árbitro sancionara saque de arco por caga contra el arquero.

Atacando con más de un robot en el área de meta del equipo oponente será sancionado con un saque de arco que será cobrado por el equipo del arquero. Se considera que un robot está en el área de la meta si está más de 50% dentro, como lo juzgue el árbitro.

Defendiendo con más de un robot en el área de meta será sancionado como por un tiro penal. (Se considera que un robot está en el área de la meta si está más de 50% dentro, como lo juzgue el árbitro.) Una excepción a esto es la situación cuando el robot adicional en el área de meta no esta allí para defender o si no afecta directamente a la jugada. El árbitro juzgará la decisión de tiro penal cuando el robot adicional en el área de la meta no está allí para defender o si no afecta directamente a la jugada. El árbitro juzgará la decisión de cobrar un tiro penal.

Eso se refiere a la manipulación, tal como lo juzgue el árbitro, cuando un robot diferente al arquero coge la pelota. También es considerado como manipulación, si un robot sujeta firmemente la pelota de tal manera que ningún otro robot pueda manipularla.

El robot arquero debe patear la pelota fuera de su área de meta (Ley 1.d.) dentro de 10 segundos. El desacato a esta ley se penalizara como penal a favor del equipo ponente.

Se otorgará saque de meta a favor del arquero del equipo, cuando exista el bloqueo intencional del arquero en su área de la meta.

Sólo el árbitro y uno de los miembros humanos del equipo (administrador, director técnico o entrenador) podrán tocar a los robots. Se concederá tiro penal cuando se toque a los robots sin permiso del árbitro.

Ley 9: Interrupciones del juego

El juego se interrumpirá y la reubicación de los robots será hecha por un operador humano, sólo cuando:

- Un robot tiene que ser cambiado.
- Un robot se ha desplomado de tal manera que bloquee el área de meta.
- Un gol se anota o una falta ocurre.

El arbitro sanciona un saque de arco (Ley 12) o free ball (Ley 13).

Ley 10: Tiro libre

Cuando un robot defensor intencionalmente empuja a un robot oponente, un tiro libre se le otorgará al equipo oponente (Ley 8.1.). La pelota se pondrá a la posición marcada para el cobro de tiro libre pertinente (FK) sobre el campo. El robot encargado de patear el tiro libre se ubicará detrás de la pelota. El equipo atacante puede posicionar sus robots libremente dentro de su propia cancha.

Los robots defensores se ubicaran delante de la línea del área de meta tocando la línea en cualquier costado del arco dibujado. Con el silbato del árbitro todos los robots pueden empezar a moverse libremente.

Ley 11: Tiro Penal

Se sancionará penal bajo las siguientes situaciones:

Defender con más de un robot dentro del área de meta (Ley 8.4.).

El despeje tardío del arquero para sacar la bola fuera de su área de meta dentro de 10 segundos (Ley 8.6.).

Cuando cualquier de los integrantes humanos toca los robots sin el permiso del árbitro, cuando el juego esta en marcha (Ley 8.8.).

Cuando el árbitro sanciona tiro penal, la bola se ubicará en la posición marcada en la cancha para el cobro del tiro penal pertinente (PK) robot encargado de ejecutar el tiro penal se ubicará detrás de la pelota. Mientras que el arquero tendrá uno de sus lados en contacto con la línea de meta. El arquero puede pueden orientarse en cualquier dirección. Los otros robots se ubicarán libremente detrás de la línea central del campo de juego (detrás de la mitad de la cancha), el equipo atacante tendrá preferencia de posicionar sus robots. El juego reiniciará normalmente (todos los robots empezarán a moverse libremente) después del silbato del árbitro. El robot encargado de patear el tiro penal puede patearlo directamente o puede driblar (avanzar con el balón junto a él) la pelota.

Ley 12: Saque de arco

Se sancionara saque de arco en las siguientes situaciones.

Cuando un robot atacante empuja al robot arquero dentro del área de meta, el árbitro sancionará saque de arco que será ejecutado por el arquero (Ley 8.2.).

Atacar con más de un robot dentro del área de meta del equipo oponente se sancionará con saque de arco a favor del equipo defensor (Ley 8.3.).

Cuando el robot oponente bloquea intencionalmente al arquero dentro de su área de meta (Ley 8.7.).

Cuando el arquero coge la pelota con cualquiera de sus accesorios (si tuviese) dentro de su área de meta.

Cuando se paraliza el juego dentro del área de meta durante 10 segundos.

Durante el cobro del saque de arco sólo el arquero se permitirá dentro del área de meta y la pelota puede ubicarse en cualquier parte dentro del área de meta.

Los otros robots del equipo se ubicarán fuera del área de meta durante el saque de arco. El equipo atacante tendrá preferencia para posicionar sus robots dentro de cualquier parte de la cancha, pero debe estar de acuerdo a la Ley 8.3. El equipo defensor puede poner sus robots dentro de su propio lado de la cancha. El juego reiniciará con el silbato del árbitro.

Ley 13: Free ball

El árbitro sancionará free-ball cuando una paralización del juego ocurre durante 10 segundos fuera del área de la meta.

Cuando se sanciona un free ball dentro de cualquier cuarto de la cancha de juego, la pelota se ubicará en la posición marcada de free ball pertinente (FB).

Un robot de cada equipo se lo pondrá a 20cm alejados de la posición de la pelota en dirección longitudinal de la cancha de juego. Los otros robots (de ambos equipos) pueden ponerse libremente fuera del cuarto de cancha donde se ejecutará el free ball, pero con la regla de que, el equipo defensor tendrá preferencia de posicionar sus robots. El juego continuará cuando el árbitro de la señal y todos los robots podrán moverse libremente.

CAPITULO II

2 COMPONENTES ELECTRÓNICOS PARA EL DISEÑO DEL ROBOT

2.1 Microcontrolador.

- Un microcontrolador es un circuito integrado de alta escala de integración que incorpora la mayor parte de los elementos que configuran un controlador.
- Un microcontrolador dispone normalmente de los siguientes componentes:
 - Procesador o UCP (Unidad Central de Proceso).
 - Memoria RAM para Contener los datos.
 - Memoria para el programa tipo ROM/PROM/EPROM.
 - Líneas de E/S para comunicars7e con el exterior.
 - Diversos módulos para el control de periféricos (temporizadores, Puertas Serie y Paralelo, CAD: Conversores Analógico/Digital, CDA: Conversores Digital/Analógico, etc.).

- Generador de impulsos de reloj que sincronizan el funcionamiento de todo el sistema.

Los productos que para su regulación incorporan un microcontrolador disponen de las siguientes ventajas:

- Aumento de prestaciones: un mayor control sobre un determinado elemento representa una mejora considerable en el mismo.
- Aumento de la fiabilidad: al reemplazar el microcontrolador por un elevado número de elementos disminuye el riesgo de averías y se precisan menos ajustes.
- Reducción del tamaño en el producto acabado: La integración del microcontrolador en un chip disminuye el volumen, la mano de obra y los stocks.
- Mayor flexibilidad: las características de control están programadas por lo que su modificación sólo necesita cambios en el programa de instrucciones.

El microcontrolador es en definitiva un circuito integrado que incluye todos los componentes de un computador. Debido a su reducido tamaño es posible montar el controlador en el propio dispositivo al que gobierna. En este caso el controlador recibe el nombre de controlador empotrado (embedded controller).

2.1.1 Aplicaciones de los microcontroladores.

Cada vez existen más productos que incorporan un microcontrolador con el fin de aumentar sustancialmente sus prestaciones, reducir su tamaño y coste, mejorar su fiabilidad y disminuir el consumo.

Algunos fabricantes de microcontroladores superan el millón de unidades de un modelo determinado producidas en una semana. Este dato puede dar una idea de la masiva utilización de estos componentes.

Los microcontroladores están siendo empleados en multitud de sistemas presentes en nuestra vida diaria, como pueden ser juguetes, horno microondas, frigoríficos, televisores, computadoras, impresoras, módems, el sistema de arranque de nuestro coche, etc. Y otras aplicaciones con las que seguramente no estaremos tan familiarizados como instrumentación electrónica, control de sistemas en una nave espacial, etc. Una aplicación típica podría emplear varios microcontroladores para controlar pequeñas partes del sistema. Estos pequeños controladores podrían comunicarse entre ellos y con un procesador central, probablemente más potente, para compartir la información y coordinar sus acciones, como, de hecho, ocurre ya habitualmente en cualquier PC.

2.1.2 Características de los Microcontroladores PIC.

- **Arquitectura Hardware.**

La CPU se conecta de manera independiente y con diferentes buses, a memoria de instrucciones y datos, por lo tanto la CPU puede acceder simultáneamente a las dos memorias (Ver Figura II.4).

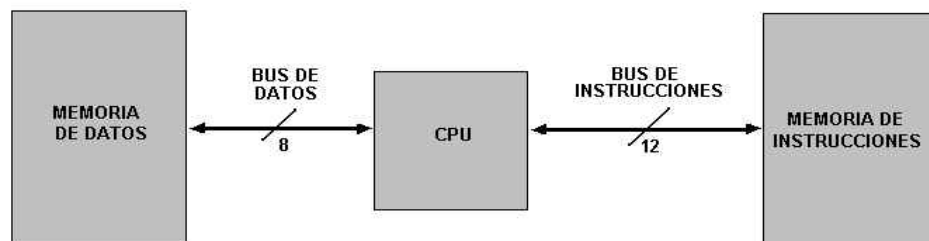


Figura II.4 Arquitectura Harvard

- **Técnica de pipe-line (segmentación).**

La segmentación permite al procesador realizar al mismo tiempo la ejecución de una instrucción y la búsqueda del código de la siguiente. De esta forma se puede ejecutar cada instrucción en un ciclo de instrucción (4 ciclos de reloj). Las instrucciones de salto ocupan dos ciclos de instrucción (8 ciclos de reloj) al no conocer la dirección de la siguiente instrucción hasta que no se haya completado la de bifurcación.

- **Juego de instrucciones RISC.**

Los PICS de la gama baja posee 33 instrucciones, 35 los de la gama media y casi 60 los de la gama alta

- **Juego de instrucciones ortogonal.**

Cualquier instrucción puede manejar cualquier elemento de la arquitectura como fuente o como destino.

- **Arquitectura basada en banco de registros.**

Todos los objetos del sistema se encuentran implementados físicamente como registros.

- **Variedad de modelos de PICS con prestaciones y recursos diferentes.**

La gran variedad de modelos de PICS permite que el usuario pueda seleccionar el más conveniente para su proyecto.

- **Variedad de herramientas de soporte.**

Son muy abundantes los programadores, simuladores, emuladores de tiempo real, ensambladores, compiladores, etc.

2.1.3 Arquitectura de los microcontroladores PIC.

Poseen una arquitectura Harvard con 8 bits para el bus de datos y longitud variable para el bus de instrucciones que dependerá del PIC.

El juego de instrucciones es RISC y varía desde 33 a 58 instrucciones.

Bloques fundamentales de la arquitectura.

2.1.3.1 Memoria de programa.

El PIC 16F628 posee una memoria EEPROM de instrucciones de solo 2K por tanto se ignoran los 2 bits de mayor peso del PIC.

2.1.3.2 El contador de programa.

El rango de direcciones que cubre el contador de programa abarca desde la 000 H hasta la 07FF H (es decir 2048 posiciones). Los dos bits de mayor peso se ignoran, luego las direcciones 0324 H, 0B24 H, 1324 H, 1B24 H son equivalentes. Como es de 13 bits necesita 2 registros para almacenarse en la memoria RAM. Los 8 bits menos significativos se almacenan en el registro PCL, situado en la posición 03 H y los más significativos en los bits de menor peso de PCLATH que se encuentra en 0A H.

En las instrucciones de salto CALL y GOTO los 11 bits de menor peso vienen del código de la instrucción y los 2 de mayor peso de PCLATH.

2.1.3.3 La pila.

La pila tiene 8 niveles de profundidad y funciona como un buffer circular el valor obtenido en el noveno POP es el mismo que en el primero. Las instrucciones CALL y las interrupciones cargan el contenido del PC en la pila.

Las instrucciones RETURN, RETLW, RETFIE saca el nivel superior de la pila al PC sin verse afectado PCLATH.

No existen instrucciones especiales (POP,PUSH) de manejo de pila ,ni tampoco señal de desbordamiento de pila.

2.1.3.4 Reinicialización (RESET).

Cuando se aplica un nivel bajo en la patita MCLR#/Vpp el microcontrolador reinicializa su estado, produciendo 2 acciones importantes:

- El contador de Programa se carga con la dirección 0x00
- La mayoría de los registros de estado y control del procesador toman un estado conocido y determinado

Causas de generación del reset:

- a) Al conectar la alimentación VDD por Power-On-Reset
- b) Al activarse MCLR# durante el estado de reposo
- c) Al activarse MCLR# durante el funcionamiento normal
- d) Al sobrepasar el conteaje el Perro guardián(WDT)
- e) Direcccionamiento de la memoria de datos

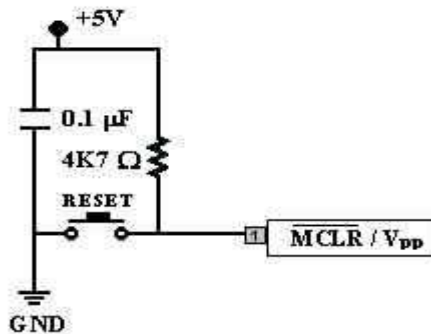


Figura II.5 Circuito de Reset

Los bits del registro de estado RP1 y RP0 determinan el banco a acceder. Se necesita 7 bits para direccionar las 128 posiciones del banco seleccionado. Esto se puede realizar mediante:

- Direcccionamiento directo: Los 7 bits vienen incluidos en el código de operación de la instrucción
- Direcccionamiento indirecto: Se utiliza como operando el registro INDF(0x00), en realidad la dirección se encuentra en los 7 bits menos

significativos del registro (FSR). El bit 8 del registro FSR junto con el bit 8 del registro de estado (IRP) indican el banco a acceder. En el PIC 16C72 $IRP = 0$

2.1.3.5 Periféricos Digitales de Entrada

Pulsadores

Estos dispositivos permiten introducir un nivel lógico en el momento que se les acciona pasando al nivel contrario cuando se deja de hacerlo y vuelven a la posición de reposo (ver figura II.6 y figura II.7)

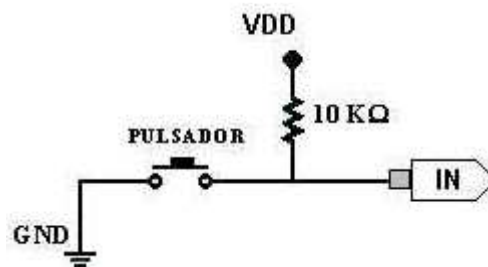


Figura II.6 Pulsadores Activado por Nivel Bajo

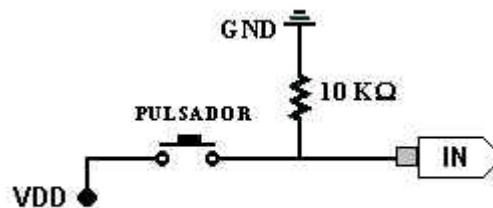


Figura II.7 Pulsadores Activado por Nivel Alto

Interruptores

Los interruptores tienen dos estados estables abierto o cerrado y hay que accionarlos para cambiar de uno a otro (ver figura II.8 y Figura II.9)

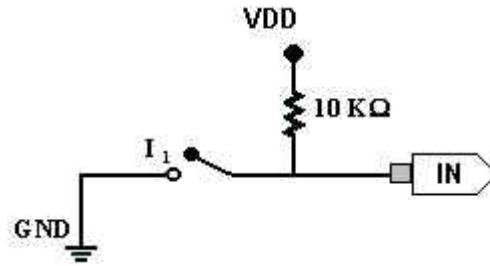


Figura II.8 Interruptores Activado por nivel Bajo

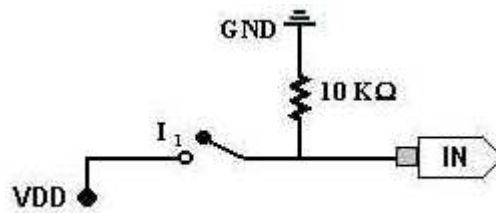


Figura II.9 Interruptores Activado por nivel Alto

Circuito Antirrebotes.

Los rebotes se producen en el instante en que se abren o se cierran las láminas metálicas del interruptor o pulsador. Hay un momento de inestabilidad en el que, aunque solo se haya accionado el dispositivo una vez, las laminas se abren y se cierran múltiples veces en el momento de la transición.

Los rebotes duran varios milisegundos y como el microcontrolador explora el estado de la línea de entrada cada pocos milisegundos, el efecto que produce es como si se hubiese abierto y cerrado el interruptor varias veces. Las consecuencias de esta falsa interpretación pueden provocar resultados erróneos.

Se encarga de eliminarlo por software, que consiste en que al identificar el primer flanco de la señal de entrada, el flujo de control pasa a una subrutina de retardo de varios milisegundos antes de pasar a detectar si se ha producido

el flanco contrario y el otro los elimina con circuitería externa o hardware mediante algunos circuitos (ver figura II.10)

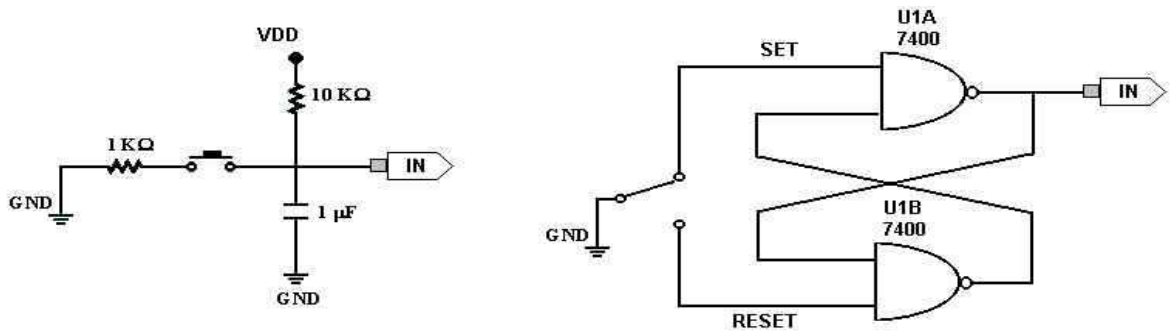


Figura II.10 Circuitos Antirrebotes.

2.1.3.6 Tipos de Osciladores.

Los PICs admiten 4 tipos de osciladores externos para aplicarles la frecuencia de funcionamiento.

- a) RC. Oscilador con resistencia y condensador.
- b) XT. Cristal.
- c) HS. Cristal de alta velocidad.
- d) LP. Cristal para baja frecuencia y bajo consumo de potencia.

El tipo empleado debe especificarse en dos bits (FOSC1 y FOSC0) de la palabra de configuración (Ver tabla II.1).

Registro de configuración

El registro de configuración se puede leer o escribir durante el proceso de grabación (ver tabla I) del dispositivo y responde a una posición reservada de dirección 0xFFFF.

Tabla I Registro de configuración

7	6	5	4	3	2	1	0
0	0	0	0	CP	WDTE	FOSC1	FOSC0

FOSC1:FOSCO Selecciona el tipo de oscilador.

00 = Oscilador LP de bajo consumo.

01 = Oscilador XT estándar.

10 = Oscilador HS de alta velocidad.

11 = Oscilador RC de bajo costo.

Es un circuito externo que le indica al micro la velocidad a la que debe trabajar. Este circuito, que se conoce como oscilador o reloj, es muy simple pero de vital importancia para el buen funcionamiento del sistema. El P1C16C84/F84 puede utilizar cuatro tipos de reloj diferentes.

En el momento de programar o "quemar" el microcontrolador se debe especificar qué tipo de oscilador se usa. Esto se hace a través de unos fusibles llamados "fusibles de configuración" o fuses.

Oscilador tipo RC.

Se trata de un oscilador de bajo coste formado por una simple resistencia y un condensador, cuyo valor depende de los valores de los dos elementos de la red R-C, proporciona una estabilidad de frecuencia mediana, es decir se utiliza cuando no se requiere una gran precisión en la generación de impulsos de reloj (Ver figura II.11).

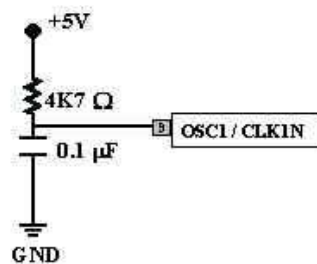


Figura II.11 Oscilador tipo RC.

Oscilador tipo XT.

Es un oscilador de cristal o resonador para frecuencias estándar comprendidas entre 100KHz y 4MHz.

Internamente esta frecuencia es dividida por cuatro, lo que hace que la frecuencia efectiva de trabajo sea de 1 MHz, por lo que cada instrucción se ejecuta en un microsegundo.

Se obtiene una mayor precisión en la generación de impulsos pero son más caros. El cristal de cuarzo o el resonador cerámico se coloca entre las patitas OSC1 y OSC2 (ver figura II.12).

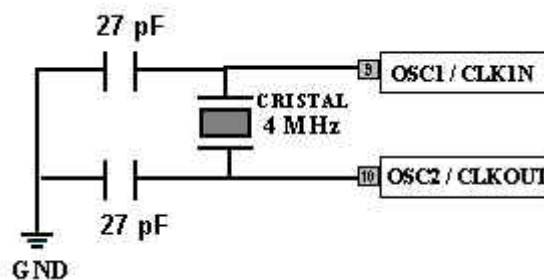


Figura II.12 Oscilador tipo XT

Si no requieres mucha precisión en el oscilador, puedes utilizar una resistencia y un condensador, donde OSC2 queda libre entregando una señal cuya frecuencia es la del OSC/4 (Ver figura II.13).

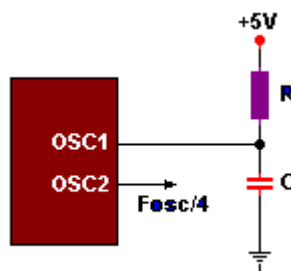


Figura II.13 Oscilador tipo XT Sencillo

Según las recomendaciones de Microchip R puede tomar valores entre 5k y 100k, y C superior a 20pf.

2.1.4 PIC16F628

El PIC 16F628 incorpora tres características importantes Procesador tipo RISC (Procesador con un Conjunto Reducido de Instrucciones) Procesador segmentado.

El PIC16F628 está fabricado en tecnología CMOS, consume baja potencia, y es completamente estático (si el reloj se detiene, los datos de la memoria no se pierden).

El encapsulado más común para estos microcontrolador es el DIP de 18 pines, y utiliza un reloj de 4 MHz (cristal de cuarzo). Sin embargo, hay otros tipos de encapsulado, por ejemplo, el encapsulado tipo surface mount (montaje superficial) es mucho + pequeño (ver figura II.14).

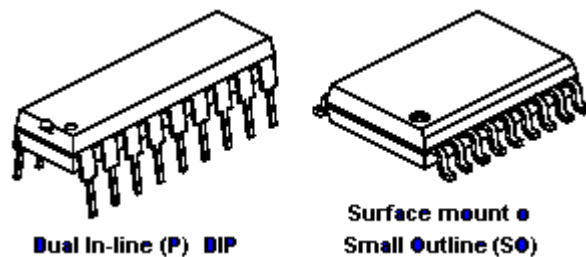


Figura II.14 PIC 16F628

2.1.4.1 Características principales

- Conjunto reducido de instrucciones (RISC).
- Oscilador interno de 4MHz.
- Las instrucciones se ejecutan en un sólo ciclo de máquina excepto los saltos (goto y call), que requieren 2 ciclos. Aquí hay que especificar que un ciclo de máquina se lleva 4 ciclos de reloj, si se utiliza el reloj interno de 4MHz, los ciclos de máquina se realizarán con una frecuencia de 1MHz, es decir que cada instrucción se ejecutará en 1µS (microsegundo).

- Opera con una frecuencia de reloj de hasta 20 MHz (ciclo de máquina de 200 ns).
- Memoria de programa: 2048 locaciones de 14 bits.
- Memoria de datos: Memoria RAM de 224 bytes (8 bits por registro).
- Memoria EEPROM: 128 bytes (8 bits por registro).
- Stack de 8 niveles.
- 16 Terminales de I/O que soportan corrientes de hasta 25 Ma.
- 3 Temporizadores.
- Módulos de comunicación serie, comparadores, PWM.

Otra característica de los PICs es el manejo de los bancos de registros. En línea general, los registros se clasifican como de uso general (GPR) y de uso específico o de funciones especiales (SFR).

Los registros de uso general pueden ser usados directamente por el usuario, sin existir restricciones. Pueden servir para almacenar resultados que se reciben desde el registro W (acumulador), datos que provienen de las puertas de entradas, etc.

Los registros de uso específicos no pueden ser usados directamente por el usuario. Estos registros controlan prácticamente todo el funcionamiento del microcontrolador, pues toda la configuración necesaria para funcionamiento del microcontrolador es hecho a través de algún tipo de SFR.

2.1.4.2 Arquitectura HARDWARE

Con estos recursos el PIC es capaz de ejecutar instrucciones solamente en un ciclo de instrucción. Con la estructura segmentada se pueden realizar simultáneamente las dos fases en que se descompone cada instrucción, ejecución de la instrucción y búsqueda de la siguiente.

La separación de los dos tipos de memoria son los pilares de la arquitectura Harvard, esto permite acceder en forma simultánea e independiente a la memoria de datos y a la de instrucciones. El tener memorias separadas permite que cada una tenga el ancho y tamaño más adecuado. Así en el PIC 16F628 el ancho de los datos es de un byte, mientras que la de las instrucciones es de 14 bits.

2.1.4.3 Encapsulado DIP (Dual In line Pin) - PIC16F628

Patas 1, 2, 3, 17 y 18 (RA0-RA4/TOCKI): Es el PORT A. Corresponden a 5 líneas bidireccionales de E/S (definidas por programación). Es capaz de entregar niveles TTL cuando la alimentación aplicada en VDD es de $5V \pm 5\%$. El pin RA4/TOCKI como entrada puede programarse en funcionamiento normal o como entrada del contador/temporizador TMR0. Cuando este pin se programa como entrada digital, funciona como un disparador de Schmitt (Schmitt trigger), puede reconocer señales un poco distorsionadas y llevarlas a niveles lógicos (cero y cinco voltios). Cuando se usa como salida digital se comporta como colector abierto; por lo tanto se debe poner una resistencia de pull-Up (resistencia externa conectada a un nivel de cinco voltios). Como salida, la lógica es inversa: un "0" escrito al pin del puerto entrega a la salida un "1" lógico. Este pin como salida no puede manejar cargas como fuente, sólo en el modo sumidero.

Pata 4 (MCLR / Vpp): Es una pata de múltiples aplicaciones, es la entrada de Reset (master clear) si está a nivel bajo y también es la habilitación de la tensión de programación cuando se está programando el dispositivo. Cuando su tensión es la de VDD el PIC funciona normalmente.

Patas 5 y 14 (VSS y VDD): Son respectivamente las patas de masa y alimentación. La tensión de alimentación de un PIC está comprendida entre 2V y 6V aunque se recomienda no sobrepasar los 5.5V.

Patas 6, 7, 8, 9, 10, 11, 12, 13 (RB0-RB7): Es el PORT B. Corresponden a ocho líneas bidireccionales de E/S (definidas por programación). Pueden manejar niveles TTL cuando la tensión de alimentación aplicada en VDD es de $5V \pm 5\%$. RB0 puede programarse además como entrada de interrupciones externas INT. Los pines RB4 a RB7 pueden programarse para responder a interrupciones por cambio de estado. Las patas RB6 y RB7 se corresponden con las líneas de entrada de reloj y entrada de datos respectivamente, cuando está en modo programación del integrado.

Patas 15 y 16 (OSC1/CLKIN y OSC2/CLKOUT): Corresponden a los pines de la entrada externa de reloj y salida de oscilador a cristal respectivamente.

Como estos dispositivos son de tecnología CMOS, todos los pines deben estar conectados a alguna parte, nunca dejarlos al aire porque se puede dañar el integrado. Los pines que no se estén usando se deben conectar a la fuente de alimentación de +5V (Ver figura II.15)

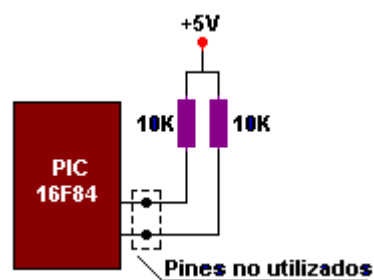


Figura II.15 Pines no Utilizados

2.1.4.4. Capacidad de corriente en los puertos

La máxima capacidad de corriente de cada uno de los pines de los puertos en modo sumidero (sink) es de 25 mA y en modo fuente (source) es de 20 mA. La máxima capacidad de corriente total se muestra en la Tabla II

Tabla II Capacidad de Corriente en los puertos del Pic

	PUERTO A	PUERTO B
Modo Sumidero	80 mA	150 mA
Modo Fuente	50 mA	100 mA

2.1.4.5. Mapa de la distribución de memoria de datos

El PIC 16F628 la distribución de los registros de datos están distribuidos según indica la Figura II.16

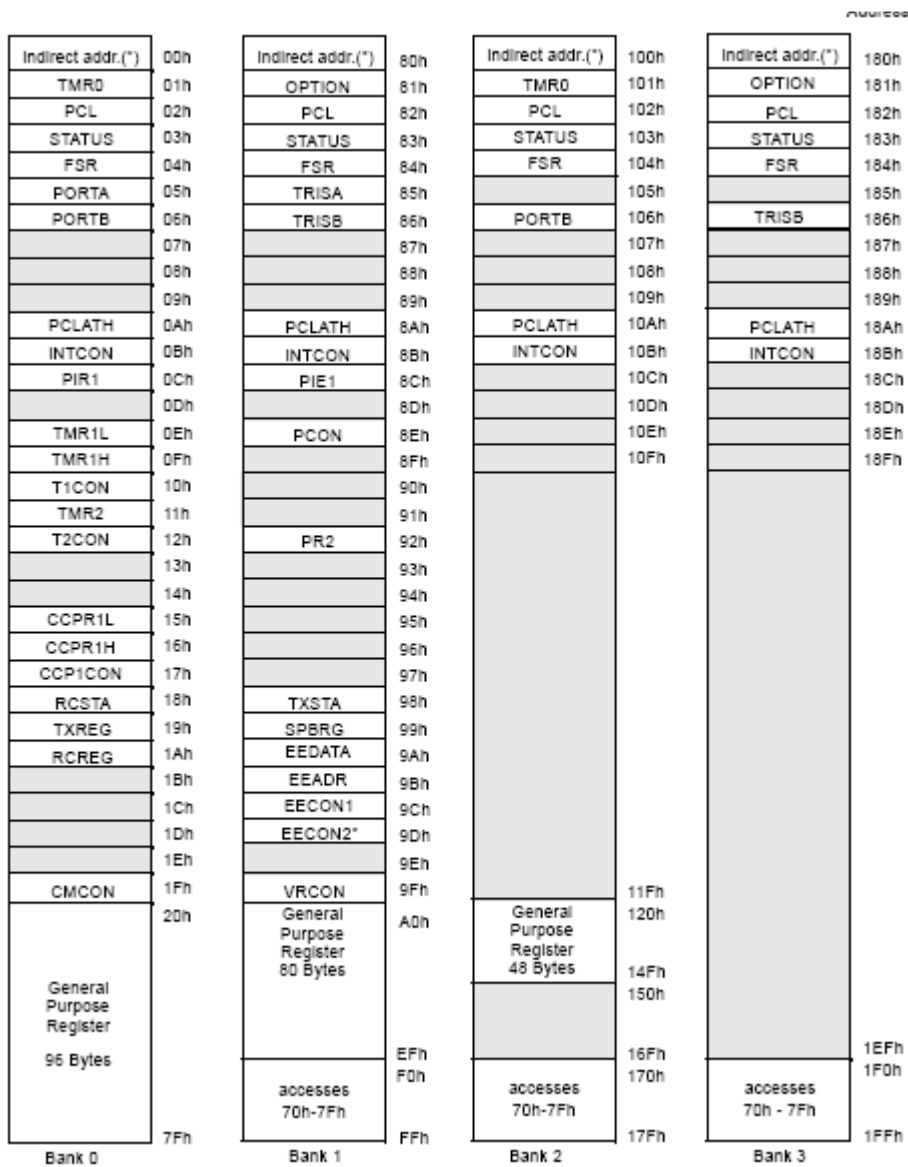


Figura III.16 Mapa de Registros del Pic 16F628

2.1.4.6 Configuración de los puertos del PIC

Llegó el momento de ver como configurar los puertos del PIC. Para poder hacerlo es necesario conocer la tabla de registros de la memoria de datos, la cual como dijimos, está dividida en el BANCO 0 y BANCO 1.

Los registros importantes en la configuración de los puertos son:

STATUS dirección 0x3

PORTA dirección 0x5

PORTB dirección 0x6

TRISA dirección 0x5

TRISB dirección 0x6

Por defecto el PIC tendrá todos los puertos de I/O (es decir los puertos RA y RB), colocados como entrada de datos, y si queremos cambiarlos habrá que configurarlos.

Al configurar los puertos deberás tener en cuenta que:

Si asignas un CERO (0) a un pin, éste quedará como salida y si le asignas un UNO (1), quedará como entrada.

Esta asignación se hace en:

TRISA para los pines del PUERTO A (8 bits)

TRISB para los pines del PUERTO B (8 bits)

Por Ejemplo:

Si TRISA es igual a 11110 todos sus pines serán entradas salvo RA0 que esta como salida.

Si TRISB es igual a 00000001 todos sus pines serán salidas salvo RB0 que esta como entrada.

Cuando el PIC arranca se encuentra en el BANCO 0, como TRISA y TRISB están en el BANCO 1 no queda otra, deberemos cambiar de banco. Esto se logra a través del Registro STATUS.

2.1.4.7 Registros importante del PIC 16F628

STATUS es un Registro de 8 bits u 8 casillas, en el cual la N° 5 (RP0) (ver Tabla III) define la posición del banco en donde nos encontramos.

Si pones un CERO (0) a RP0 estaremos en el BANCO 0.

Si le pones un UNO (1) ya ves, estaremos en el BANCO 1

Tabla III Registro Status del PIC 16F628

REGISTRO STATUS							
7	6	5	4	3	2	1	0
IRP	RP1	RP0	TO	PD	Z	DC	C

IRP: Selecciona de bancos para el direccionamiento indirecto.

Este bit junto con el de más peso del registro FSR sirven para determinar el banco de la memoria de datos seleccionado. En el PIC al disponer de dos bancos no se usa este bit y debe ponerse a 0 (IRP=0).

0 = Banco 0 y 1 (0x00-0xFF)

1 = Bancos 2 y 3 (0x100-0x1FF)

RP1:RP0: Selección de banco para el direccionamiento directo.

Como el PIC16C72 solo tiene dos bancos únicamente emplea el bit RP0 de forma que cuando vale 1 se accede al banco 1 y cuando vale 0 se accede al banco 0. Después de un reset RP0=0

00 = Banco 0 (0x00-0x7F)

01 = Banco 1 (0x80-0xFF)

10 = Banco 2 (0x100-0x17F)

11 = Banco 3 (0x180-0x1FF)

TO#: Timer Out

1 = Se pone a 1 tras conectar la alimentación VDD o ejecutar "CLRWDT" o "SLLEP".

0 = Se pone a 0 cuando se produce el desbordamiento del Perro guardián "WDT".

PD#: Power Down.

1 = Se pone a 1 tras conectar la alimentación VDD o ejecutar "CLRWDT".

0 = Se pone a 0 al ejecutar la instrucción "SLEEP".

Z: Señalizador de cero.

1 = El resultado de una instrucción aritmético-lógica ha sido 0.

0 = El resultado de una instrucción aritmético-lógica no ha sido 0.

DC: Señalizador de acarreo en el 4º bit de menos peso.

1 = Acarreo en la suma o no acarreo en la resta.

0 = No acarreo en la suma o acarreo en la resta.

C: Señalizador de acarreo en el octavo bit (MSB).

1 = Acarreo en la suma o no acarreo en la resta.

0 = No acarreo en la suma o acarreo en la resta.

Algunos registros específicos aparecen por primera vez y se proporcionan unos comentarios aclaratorios de sus funciones.

- PCLATH contiene los 5 bits de más peso del PC.
- INTCON soporta varios señalizadores para el control y activación de todas las posibles funciones de interrupción, excepto la del modulo comparador.
- PIR1 en este registro se ubica el señalizador que indica el estado interno del comparador.

- PIE1 contiene el bit de activación para la interrupción que provoca el comparador.
- PCON soporta los dos señalizadores que diferencian entre el reset provocado por conexión de alimentación (POR: Power-On-Reset) y el que se origina cuando se detecta un fallo en la alimentación (BOD: Brown-Out-Detect).
- CMCON controla las entradas y salidas del modulo Comparador.
- VRCON controla el funcionamiento del modulo de Tensión de referencia.

Registro de OPTION.

Se utiliza para controlar los temporizadores y preescaladores.

Registros INTCON, PIE1, PIR1.

Controla los desbordamientos en los temporizadores y la habilitación de las interrupciones. Se debe hacer un clear antes de las interrupciones.

Registro PCON.

Para diferenciar los Reset, externo Power On Reset (ver tabla IV), o WatchDog Reset. Reset externo: Los PIC 16FXX posee un pin para reinicializar el sistema.

Reset, MCLR#/Vpp

Tabla IV Registro PCON del PIC 16F628

Registro PCON							
7	6	5	4	3	2	1	0
0	0	0	0	0	0	POR#	BOR#

POR#: Indica el estado del Power-On-Reset.

1 = No ha ocurrido el Power-On-Reset.

0 =Ha ocurrido el Power-On-Reset.

BOR#: Indica el estado del Brown-out-Reset.

1 = No ha ocurrido el Brown-out-Reset.

0 = Ha ocurrido el Brown-out-Reset.

2.1.6.8 Ejemplos de Programación

En la tabla V código de ejemplo que configure todos los pines del puerto A como entrada y todos los del puerto B como salida.

Tabla V Código de Ejemplo

```
;-----Encabezado-----
List    p=16f628          ; usaremos el PIC 16f84
radix   hex              ; y la numeración hexadecimal
;-----mapa de memoria-----
estado  equ    0x03      ; Aquí le asignamos nombres a los
trisa   equ    0x05      ; registros indicando la posición
trisb   equ    0x06      ; en la que se encuentran
;-----Configuración de puertos-----
reset   org    0x00      ; origen del programa, aquí comenzaré
                          ; siempre que ocurra un reset
        goto   inicio    ; salto a "inicio"
        org    0x05      ; origen del código de programa
inicio  bsf    estado,5   ; pongo rp0 a 1 y paso al banco 1
        movlw  b'11111'   ; cargo W con 11111
        movwf trisa      ; y paso el valor a trisa
        movlw  b'00000000' ; cargo W con 00000000
        movwf trisb     ; y paso el valor a trisb
        bcf    estado,5   ; pongo rp0 a 0 y regreso al banco 0
;-----
        End            ; se acabó
;-----
```

Descripción del código

Todo lo que escribas luego de un ";" (punto y coma) será ignorado por el ensamblador, estos son los famosos comentarios, y sirve para saber que hace cada línea de código.

Dicho esto no queda más que describir el código, así que vamos por partes.

Tabla VI Encabezado del código

```

;-----Encabezado-----
list    p=16f628A          ; usaremos el PIC 16f628A
radix   hex                ; y la numeración hexadecimal
    
```

En la Tabla VI le indicas al ensamblador para que microcontrolador estas codificando (PIC16F628A). La cuál será el sistema de numeración que utilizarás (hexadecimal).

Nota que hay tres columnas, en este caso la primera está vacía. Respeta las tabulaciones para no confundir al ensamblador.

Tabla VII Asignación de memoria

```

;-----mapa de memoria-----
estadoequ    0x03          ; Aquí le asignamos nombres a los
trisa equ    0x05          ; registros indicando la posición
trisb equ    0x06          ; en la que se encuentran
    
```

Al registro STATUS como la indica en la Tabla VII, que está en la posición 0x03 de la memoria de datos le puse la etiqueta "estado". equ es igual. (Es decir, le estoy asignando el nombre estado al registro que está en la posición 0x03 de la memoria de datos).

Luego lo mismo con trisa y trisb.

Tabla VIII Configuración de Puerto I/O

```

;-----Configuración de puertos-----
reset org    0x00          ; origen del programa, aquí comenzaré
                                ; siempre que ocurra un reset
        goto  inicio          ; salto a "inicio"
        org    0x05          ; origen del código de programa
inicio bsf    estado,5      ; pongo rp0 a 1 y paso al banco1
        movlw b'11111'      ; cargo W con 11111
        movwf trisa          ; y paso el valor a trisa
        movlw b'00000000'   ; cargo W con 00000000
        movwf trisb          ; y paso el valor a trisb
        bcf    estado,5     ; pongo rp0 a 0 y regreso al banco0
    
```

- La directiva **org** (ver Tabla VII) indica el sitio de la memoria en donde se escribe una parte del programa. En este caso el contador de

programa apuntará a la dirección 0x00 (reset) entonces ejecutará la instrucción que sigue a continuación, (saltar a la etiqueta inicio) y nuestro código de programa comienza en la dirección de memoria 0x05 (aquí salto por encima de la interrupción 0x04)

- **BSF**, (SET FILE REGISTER), es la instrucción que pone un uno en el bit del registro especificado, en este caso pone a uno el bit 5 del registro STATUS (el rp0), para pasar al banco 1.
- **movlw**, mueve el siguiente literal al Registro W.
- **W**, es el Registro de Trabajo, y lo usamos para almacenar momentáneamente los datos que queremos mover, una vez hecho esto pasamos el dato a trisa, o a trisb, según el caso.
- **movwf**, mueve el contenido del registro W al registro f, en este caso f sería trisa o trisb.
- **BCF** (BIT CLEAR FILE REGISTER), ésta instrucción limpia el bit del registro especificado, o lo pone a cero, en este caso pone a cero el bit 5 del registro STATUS para regresar al banco 0.

2.2 Reguladores de tensión

2.2.1 Historia de los reguladores monolíticos de tensión

La historia de los reguladores lineales empieza por la necesidad de simplificar el proceso de entrega de un voltaje regulado, con el menor tiempo de diseño posible. En 1968 la empresa Fairchild logró integrar en un solo circuito integrado el uA 723, un amplificador operacional, un diodo zener, unas resistencias y unos transistores, logrando así el primer regulador monolítico que aún se utiliza ampliamente.

Este circuito viene con circuitos internos de protección. Su capacidad de entrega de corriente es limitada por lo que el diseñador debe aumentarle transistores externos para aumentar su capacidad.

También debe adicionar una resistencia para limitar la corriente a un valor aceptable en caso de que haya un corto circuito en la salida.

Es desde este momento se inicia la carrera para el diseño de un circuito regulador de voltaje fijo de 3 terminales. Esta carrera fue ganada por la National Semiconductor con el LM309.

Poco después la Fairchild sacó la serie 7800 (comenzó con el 7805, que da una salida de 5 voltios). En ambos casos sólo es necesario conectar a la entrada la fuente de tensión no regulada y a la salida la carga. Se pone en la entrada y la salida unos capacitores desacopladores para mejorar el rendimiento.

2.2.2 Reguladores Integrados de Voltaje.

Todo dispositivo electrónico, desde una PC hasta un amplificador, debe necesariamente poseer una fuente de poder para su funcionamiento, que recibe la tensión de 220 volts RMS y 50 Hz. Y la convierte en una señal DC sin ripple. En efecto, diseñada y realizada de mil formas distintas, siempre se encuentra presente en todo equipo electrónico cualquiera sea su uso o complejidad. Es por esto que a la hora de realizar cualquier circuito electrónico práctico, el diseñador debe necesariamente realizar una fuente de alimentación acorde a las necesidades del circuito, en muchas ocasiones diseñar tal fuente resultaba una tarea bastante difícil y tediosa, sobre todo si el diseño incluía transistores, zener, etc. Se debían asumir compromisos que relacionaban la complejidad del diseño con la estabilidad del mismo.

Afortunadamente las empresas diseñadoras de componentes han puesto fin a esta peripecias, presentando, hace ya algunas décadas, los reguladores integrados. Estos dispositivos de gran utilidad aúnan todas las ventajas de una completa fuente de alimentación en un solo encapsulado reduciendo el problema de un buen diseño, a unos pocos componentes. En esta tecnología encontramos reguladores integrados fijos y variables, series y paralelos. Los que a continuación se detallaran corresponden a Reguladores Serie.

2.2.2.1 Reguladores fijos

En la mayoría de las aplicaciones se requiere una tensión fija y estable de un determinado valor. Una de las familias de reguladores ideales para este tipo de necesidades es la 78XX. El fabricante utiliza los siguientes códigos relacionados con este IC (Integrated Circuit), ejemplo el LM7812T.

Las primera letras (LM) corresponden al código que indica el fabricante y los dos número finales (12) corresponden a la denominación que utilizan todos los fabricantes para el mismo componente, mientras que la última letra (T) indica un código adicional que se refiere al tipo de cápsula o case (package de aluminio, epoxy o acero) que tiene relación con la potencia y corrientes del integrado (Ver Tabla IX).

Tabla IX Reguladores Fijos

Número	Tensión de salida
LM7805	5 Voltios
LM7806	6 Voltios
LM7808	8 Voltios
LM7809	9 Voltios
LM7812	12 Voltios
LM7815	15 Voltios
LM7818	18 Voltios
LM7824	24 Voltios
LM7830	30 Voltios

Existen fabricantes que construyen modelos con voltajes intercalados a la tabla anterior.

Cada uno de estos dispositivos posee sólo tres terminales, el pin 1 corresponde a la entrada de tensión no regulada, Pin 3 es la salida regulada y pin 2 la masa común a ambas. En cuanto al encapsulado, conviene aclarar que, si bien están disponibles en varios tipos, generalmente se los suele encontrar en el encapsulado del tipo TO-220, como el usado en clases (Ver figura II.17).

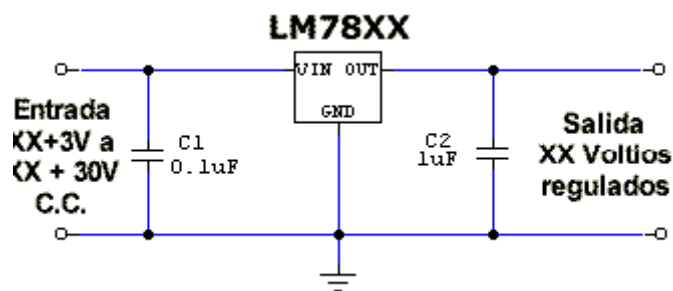


Figura II.17 Regulador Fijo

Como se observa, sólo fueron agregados dos capacitores al circuito integrado, C1, que se halla a la entrada del regulador, filtra la tensión de posibles transitorios, ruido de RF (radiofrecuencia) y picos indeseables, mientras que C2, que se encuentra a la salida, disminuye la posible tensión de rizado que pudiera haber a la salida, a la vez que evita oscilaciones indeseadas o espureas. En cuanto a la tensión de entrada, se puede mencionar que es de un rango muy amplio, como se aprecia. Por ejemplo, si el regulador elegido es uno de nueve volts (LM7809), la tensión de entrada podrá ser de entre 12 y 39 voltios. Para realizar una fuente de alimentación fija completa, observemos la figura II.18.

En este diseño partimos directamente de la tensión alterna de red (220v ó 110v depende del país), para lograr una tensión perfectamente estable.

Primeramente, como es lógico, la tensión es reducida hasta un valor manejable por un transformador. Luego, esta tensión alternada de bajo valor es rectificada por el puente de Graetz D1, obteniéndose así una señal de onda completa. Después la señal se filtra por medio de C1 consiguiéndose de esta forma una tensión continua no estabilizada, es decir con ripple, que es inyectada al circuito anterior para su regulación.

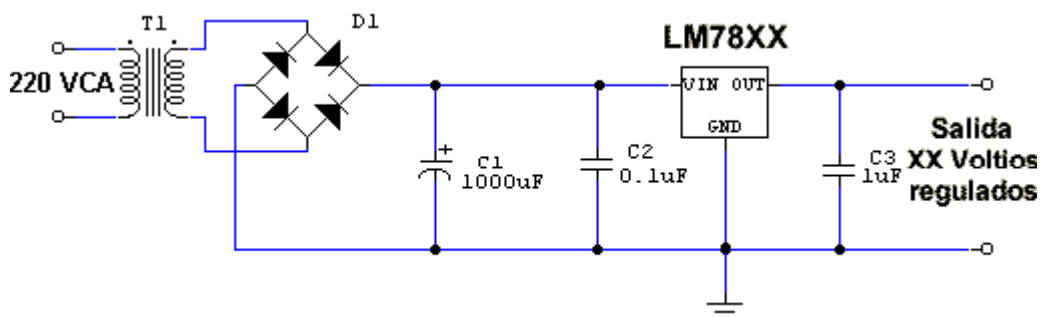


Figura II.18 Fuente de Alimentación Fija Completa

Características de las fuentes con reguladores integrados.

En una fuente de poder existen 3 parámetros que se deben considerar.

- **Regulación de línea:** establece cuánto varía la tensión de salida frente a variaciones en la tensión de entrada. Es posible comprobar que para un cambio de 20 voltios a la entrada se produce una variación de sólo 4 mV a la salida, con lo cual, podemos suponer que es inmune a los cambios de tensión de entrada.
- **Regulación de carga:** indica cuánto varía la tensión de salida cuando la corriente varía de un mínimo al máximo. Nuevamente los resultados obtenidos son excelentes: para una variación de corriente de 1,5 amperios, la tensión de salida solamente se modifica en 10 milésimas de voltio.

- **Ripple Rejection:** este valor indica cuántas veces se reduce la tensión de rizado entre la salida con respecto a la entrada. Con el capacitor de salida se obtienen valores típicos de 75 dB. Esto implica que la tensión de rizado a la salida es 5000 veces menor que a la entrada, lo cual hace que este comportamiento sea muy parecido a una pila o batería. Esta característica posibilita la disminución de la capacidad de C1, con la reducción de costo y tamaño que esto trae aparejado.

Finalmente, la corriente que este tipo de dispositivo es capaz de entregar puede ir en 3 rangos.. y por lo anterior se fabrican en 3 cápsulas o carcasas: 200mA, 1 A y 3 A. Es importante aclarar que estos valores son válidos cuando se utiliza un disipador adecuado y cuando la tensión de entrada no es superior en más de 15 voltios con respecto a la de salida. Es decir que $V_{sal} < 15V$. Igualmente veremos algunos métodos para obtener mayor corriente de salida de estos dispositivos.

Una característica importante de esta línea es la protección térmica y contra corriente excesiva: cuando la corriente que atraviesa al integrado adquiere un valor demasiado elevado o cuando su temperatura es excesiva, el integrado disminuye la tensión de salida en forma automática a cero, lo que significa que el voltaje de salida es cero. Debido a estas últimas características estos dispositivos son casi indestructibles.

En resumen, con unos pocos componentes es posible fabricar, mediante el uso de reguladores de tensión, una fuente de tensión fija con una salida tipo de 1,5A, cuya salida no varía en más de 15mV para cualquier condición. Este

tipo de fuente es más que suficiente para la mayoría de las aplicaciones electrónicas.

De esta forma obtenemos una fuente simétrica con las características de la anterior fuente simple. Es necesario aclarar que, aunque no es conveniente, las tensiones de salida del regulador positivo y negativo no tienen por qué ser las mismas. Sin embargo, es recomendable que no sean muy diferentes una de la otra.

2.3 Puente H

La interfaz de potencia para motores de corriente continua o CD, PUENTE H, es un sistema que permite controlar motores en rangos entre 12 y 30 voltios y con consumos de hasta 2.5 amperios por medio de señales de baja potencia provenientes de un circuito digital.

En general cualquier tipo de sistema digital puede adquirir fácilmente la capacidad para activar motores de CD haciéndolos girar en uno o en otro sentido usando la interfaz Puente H como puente entre el sistema digital y el motor de CD.

Las principales características de la interfaz Puente H se enumeran a continuación:

- Activación de motores en un rango entre 9 y 30 voltios DC.
- Capacidad para entregar hasta 2.5 Amperios a la carga¹.
- Capacidad para activar el giro del motor en cualquiera de los dos sentidos.
- Fácil y rápida conexión gracias a sus borneras de tornillo.
- Reducido tamaño
- Exente calidad.

2.3.1 Uso del la interfaz PUENTE H

La interfaz PUENTE H es básicamente un sistema (ver figura II.19) de conmutación controlado por dos señales digitales de baja potencia. Cuando el sistema detecta un 1 digital en una de sus dos entradas de control y un cero en la otra, este conecta el motor al la fuente de alimentación con determinada polaridad, si la señal de control que estaba en 1 pasa a cero y la de cero a uno el PUENTE H conecta la fuente al motor con la polaridad invertida facilitando así el giro en sentido contrario, en la Tabla X se muestra la tabla de verdad.

Tabla X. Tabla de verdad interfaz PUENTE H

Entradas				Motor	
Alimentación		Control			
GND	+	+	-	+	-
0	V _{IN}	0	0	0	0
0	V _{IN}	0	1	V _{IN} -1.3v	0
0	V _{IN}	1	0	0	V _{IN} -1.3v
0	V _{IN}	1	1	No permitido	

2.3.2 Consideraciones de Corriente

Por su gran flexibilidad, la interfaz PUENTE H puede manejar un gran rango de motores, debido a esta característica es necesario considerar la potencia que se disipara en forma de calor en los transistores de potencia y adaptar a estos un disipador de calor en caso se ser necesario.

Motores de baja potencia que consuman corriente que no exceda 0.5 A, o de hasta 1A que no sean operados continuamente por mas de unos minutos,

pueden ser operados con seguridad sin necesidad de disipadores, motores mas grandes en sitios calurosos pueden requerir el uso de disipadores.

Potencia disipada en forma de calor en los transistores de potencia se puede modelar mediante la siguiente expresión matemática.

$$P=I^2R$$

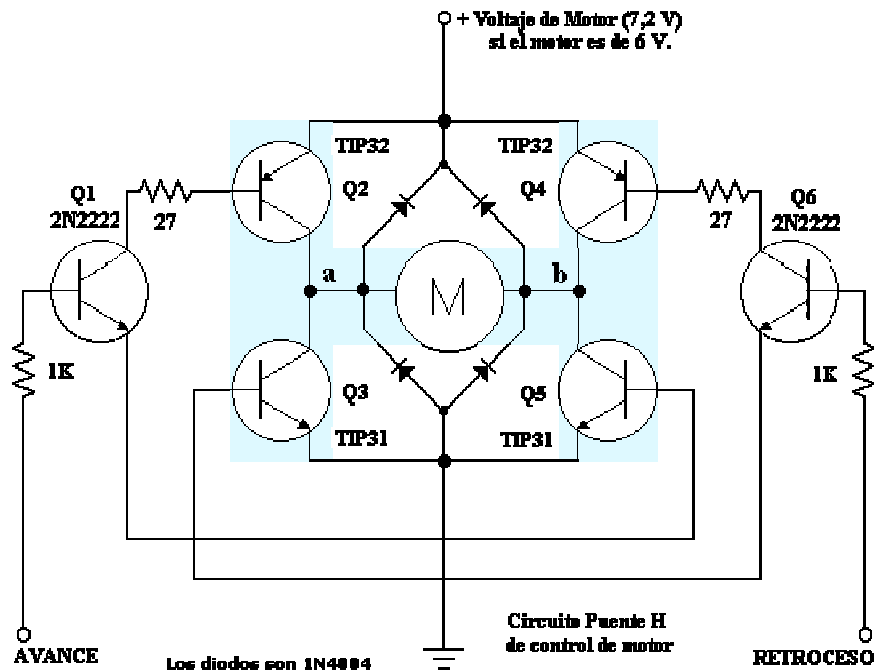


Figura II.19 Circuito Puente H

2.3.3 Funcionamiento

Aplicando una señal positiva en la entrada marcada AVANCE se hace conducir al transistor Q1. La corriente de Q1 circula por las bases, de Q2 y Q5, haciendo que el terminal **a** del motor reciba un positivo y el terminal **b** el negativo (tierra) ver figura II.20.

Si en cambio se aplica señal en la entrada RETROCESO, se hace conducir al transistor Q6, que cierra su corriente por las bases, de Q4 y Q3. En este caso se aplica el positivo al terminal **b** del motor y el negativo (tierra) al terminal **a** del motor (ver Figura II.21).

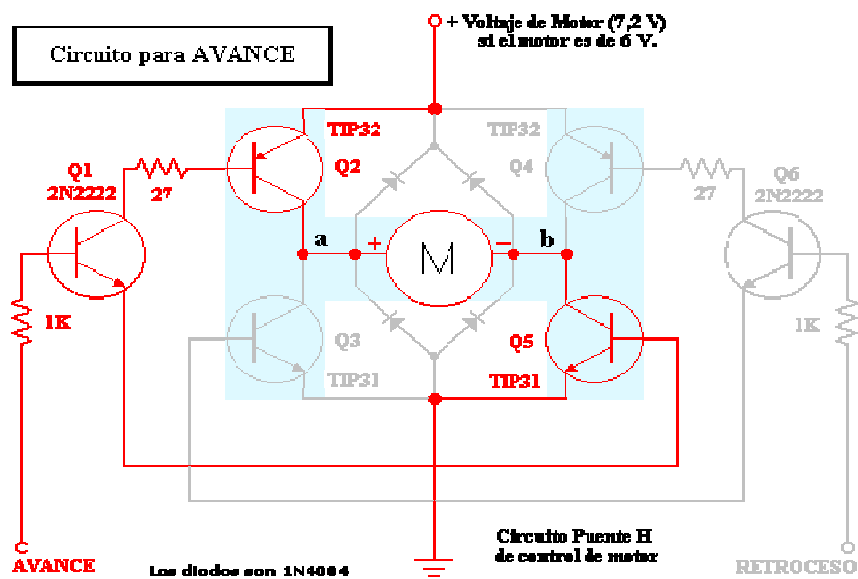


Figura II.20 Circuito de Avance

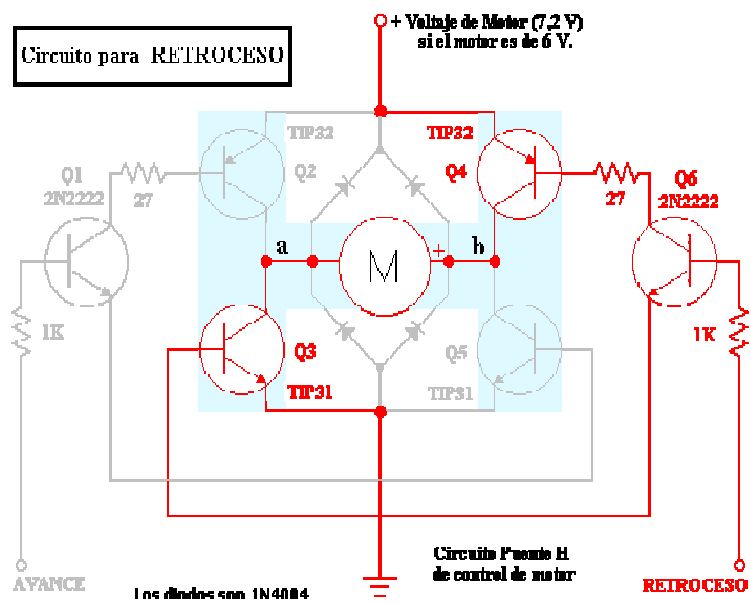


Figura II.21 Circuito de Retroceso

2.3.4 L293D

El dispositivo es un monolítico integrado de alta tensión, alta actual driver de cuatro canales diseñada para DTL estándar aceptar o niveles lógicos TTL y

unidad cargas inductivas (relés, como solenoides, DC y motores paso a paso) y transistores de potencia de conmutación.

Para simplificar su utilización como dos puentes cada par de canales está equipado con un permita la entrada. Otro suministro de insumos está prevista la lógica, lo que permite la operación a menor voltaje y diodos de sujeción interior están incluidos.

Este dispositivo es apto para su uso en aplicaciones de conmutación a frecuencias de hasta 5 kHz

Características

- 600mA corriente de salida capacidad Por canal
- 1.2A pico corriente de salida (no repetitivos) Por canal
- MECANISMO PARA HABILITAR
- Protección de exceso de temperatura
- LÓGICO "0" voltaje de entrada hasta 1,5 V (HIGH inmunidad al ruido)
- Pinza interior diodos

Este componente consta de 4 drivers (ver figura II.22) con señal de habilitación por parejas, es ideal para la amplificación de señales procedentes de microcontroladores con poca potencia de salida.

Las señales GND y Vcc son las alimentaciones siendo su tensión nominal de 0 y 5V respectivamente, mientras que Vdd es la tensión para las cargas y puede valer desde 5v a 36v. La corriente máxima de salida es de 600mA, mas que suficiente para activar leds, motores pequeños, microrrieles.

Otra importante característica de este chip es que a sus salidas tiene un par

de diodos en push-pull que llevan a masa cualquier sobretensión espurea dañina que pudiesen enviar las cargas.

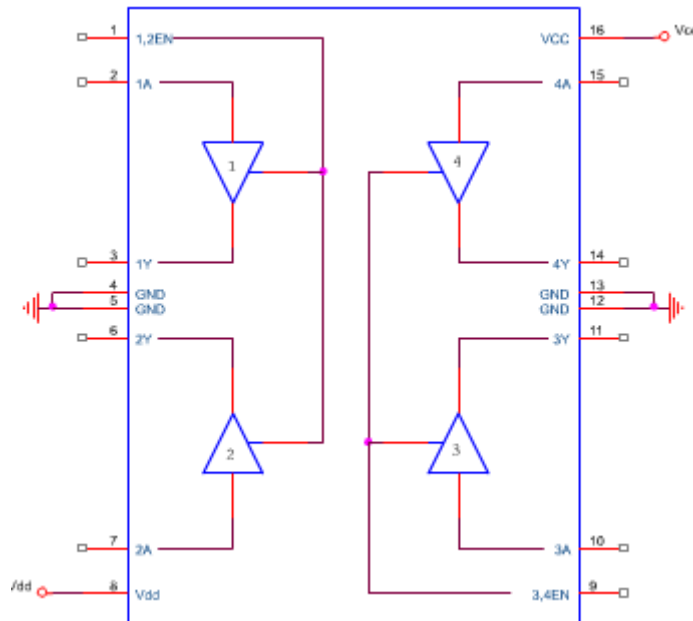


Figura II.22 Diagrama Interno De 1293

2.4 Radio Frecuencia

El término radiofrecuencia, también denominado espectro de radiofrecuencia o RF, se aplica a la porción menos energética del espectro electromagnético, situada entre unos 3 Hz y unos 300 GHz. Las ondas electromagnéticas de esta región del espectro se pueden transmitir aplicando la corriente alterna originada en un generador a una antena. La radiofrecuencia se puede dividir en las siguientes bandas del espectro (Ver tabla XI)

2.4.1 Usos de la radiofrecuencia

Uno de sus primeros usos fue en el ámbito naval, para el envío de mensajes en código Morse entre los buques y tierra o entre buques.

Tabla XI Bandas del espectro de radio

Nombre	Abreviatura inglesa	Banda ITU	Frecuencias	Longitud de onda
			Inferior a 3 Hz	> 100.000 km
Extra baja frecuencia Extremely low frequency	ELF	1	3-30 Hz	100.000 km – 10.000 km
Super baja frecuencia Super low frequency	SLF	2	30-300 Hz	10.000 km – 1000 km
Ultra baja frecuencia Ultra low frequency	ULF	3	300–3000 Hz	1000 km – 100 km
Muy baja frecuencia Very low frequency	VLF	4	3–30 kHz	100 km – 10 km
Baja frecuencia Low frequency	LF	5	30–300 kHz	10 km – 1 km
Media frecuencia Medium frequency	MF	6	300–3000 kHz	1 km – 100 m
Alta frecuencia High frequency	HF	7	3–30 MHz	100 m – 10 m
Muy alta frecuencia Very high frequency	VHF	8	30–300 MHz	10 m – 1 m
Ultra alta frecuencia Ultra high frequency	UHF	9	300–3000 MHz	1 m – 100 mm
Super alta frecuencia Super high frequency	SHF	10	3-30 GHz	100 mm – 10 mm
Extra alta frecuencia Extremely high frequency	EHF	11	30-300 GHz	10 mm – 1 mm
			Por encima de los 300 GHz	< 1 mm

Actualmente, la radio toma muchas otras formas, incluyendo redes inalámbricas, comunicaciones móviles de todo tipo, así como la radiodifusión.

Antes de la llegada de la televisión, la radiodifusión comercial incluía no solo noticias y música, sino dramas, comedias, shows de variedades, concursos y muchas otras formas de entretenimiento, siendo la radio el único medio de representación dramática que solamente utilizaba el sonido.

Otros usos de la radio son:

- Audio

- La forma más antigua de radiodifusión de audio fue la radiotelegrafía marina, ya mínimamente utilizada. Una onda continua (CW), era conmutada *on-off* por un manipulador para crear código Morse, que se oía en el receptor como un tono intermitente.
 - Música y voz mediante radio en modulación de amplitud (AM).
 - Música y voz, con una mayor fidelidad que la AM, mediante radio en modulación de frecuencia (FM).
 - Música, voz y servicios interactivos con el sistema de radio digital DAB empleando multiplexación en frecuencia OFDM para la transmisión física de las señales.
 - Servicios RDS, en sub-banda de FM, de transmisión de datos que permiten transmitir el nombre de la estación y el título de la canción en curso, además de otras informaciones adicionales.
 - Transmisiones de voz para marina y aviación utilizando modulación de amplitud en la banda de VHF.
 - Servicios de voz utilizando FM de banda estrecha en frecuencias especiales para policía, bomberos y otros organismos estatales.
 - Servicios civiles y militares en alta frecuencia (HF) en la banda de Onda Corta, para comunicación con barcos en alta mar y con poblaciones o instalaciones aisladas y a muy largas distancias.
 - Sistemas telefónicos celulares digitales para uso cerrado (policía, defensa, ambulancias, etc). Distinto de los servicios públicos de telefonía móvil.
- Telefonía
 - Vídeo

- Navegación
- Radar
- Servicios de emergencia
- Transmisión de datos por radio digital
- Calentamiento
- Fuerza mecánica
- Otros

2.4.2 Modulación ASK

Es una forma de modulación en la cual se representan los datos digitales como variaciones de amplitud de la onda portadora [figura III.28].

La amplitud de una señal portadora análoga varía conforme a la corriente de bit (modulando la señal), manteniendo la frecuencia y la fase constante. El nivel de amplitud puede ser usado para representar los valores binarios 0s y 1s. Podemos pensar en la señal portadora como un interruptor ON/OFF. En la señal modulada, el valor lógico 0 es representado por la ausencia de una portadora, así que da ON/OFF la operación de pulsación y de ahí el nombre dado.

Como la modulación AM, ASK es también lineal y sensible al ruido atmosférico, distorsiones, condiciones de propagación en rutas diferentes en PSTN, etc. Esto requiere la amplitud de banda excesiva y es por lo tanto una un gasto de energía. Tanto los procesos de modulación ASK como los procesos de demodulación son relativamente baratos. La técnica ASK también es usada comúnmente para transmitir datos digitales sobre la fibra óptica. Para los transmisores LED, el valor binario 1 es representado por un pulso corto de luz y el valor binario 0 por la ausencia de luz. Los transmisores de láser

normalmente tienen una corriente "de tendencia" fija que hace que el dispositivo emita un nivel bajo de luz. Este nivel bajo representa el valor 0, mientras una onda luminosa de amplitud más alta representa el valor binario

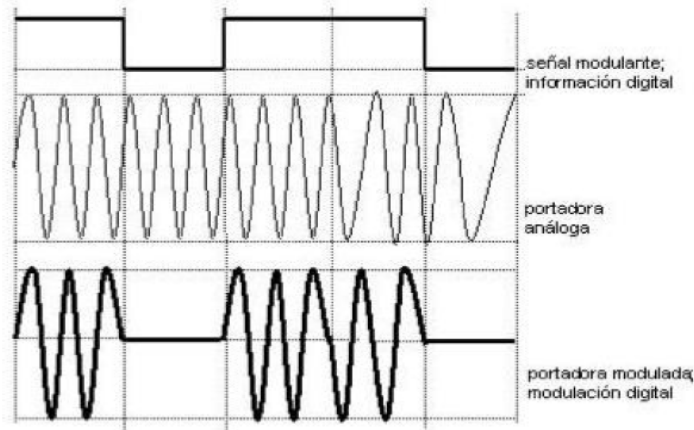


FIGURA III.23 Modulación ASK

2.4.3 Módulos De Radio Frecuencia

2.4.3.1 Modulo TPL434A (TX)



Figura III.24 Modulo Transmisor

Características:

- Rango de Frecuencia: 433,92 MHz
- Modulación: ASK
- Forma Circuito: S / F
- Velocidad máxima de datos: 8Kbps
- Tensión de alimentación: 3 12 V

- Voltaje: 3V; corriente: 4.9mA
- Voltaje: 5V; corriente: 8.4mA
- Suministro de energía y todos los Entrada/Salida Pines: -0,3 a 12,0 V
- Temperatura de funcionamiento de no operacion: -10 a +85 centígrados
- Temperatura de soldadura: 230 centígrados (10 segundos)
- Diseño Pasivo de Alta sensibilidad

Aplicaciones

- El teclado inalámbrico, ratón, juguetes inteligentes
- Sistema de seguridad de carros
- Sensor remoto
- medición industrial
- Control a distancia para el hogar

2.4.3.2 Modulo RPL434 (RX)

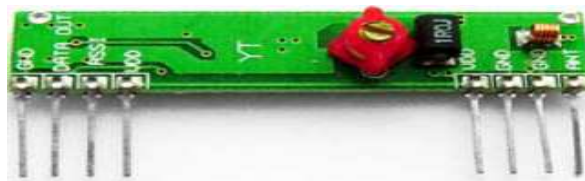


Figura III. 25 Modulo Receptor

Características:

- Rango de Frecuencia 432.92 MHz
- Modulación ASK
- Forma del circuito LC
- Velocidad de Datos 4800 Bps
- Sensibilidad -106 Dbm

- Canal expandido +/- 500khz
- Voltaje de Alimentación 5V
- Diseño Pasivo de Alta sensibilidad

Aplicaciones

- El teclado inalámbrico, ratón, juguetes inteligentes
- Sistema de seguridad de carros
- Sensor remoto
- medición industrial
- Control a distancia para el hogar

CAPITULO III

3 DISEÑO E IMPLEMENTACIÓN DEL ROBOT

3.1 Implementación de Transmisión

El transmisor consta de una aplicación para PC la cual posee una interface con el puerto serial, un convertidor TTL (circuito Max 232), un microcontrolador (Pic 16F628A), y el transmisor de radio frecuencia TPL 434 (ver figura III.26).

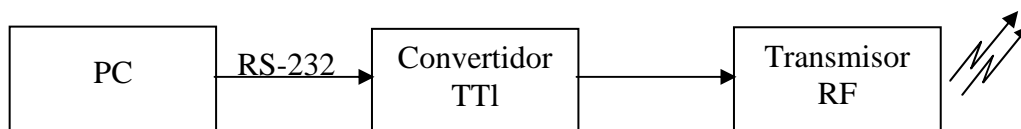


Figura III.26 Diagrama de bloques del transmisor

La PC es la que se encarga de enviar a través de RS-232 (puerto serial) una trama (ver tabla XII); a una velocidad de 2400 bps, la cual contiene, las órdenes del movimiento del robot; estos datos pasan por el convertidor TTL (ver figura III.27), es el que encarga de cambiar la lógica que con al que trabaja el puerto serial; para luego pasar a un microcontrolador el mismo que se encarga de reenviar los datos al modulo transmisor de Radio frecuencia.

El transmisor TPL 434 tiene una frecuencia de 433.92 Mhz y opera con un voltaje de 5-12 v, tiene una potencia de salida de 8mW alcanzando una distancia en espacios abiertos de 50 metros en espacios abiertos y de 10 metros en espacios internos donde se encuentran obstáculos como paredes.

Tabla XII Trama enviada por la PC

FF	FF	AA	Robot	Angulo giro	Dirección	Velocidad	Tiempo
----	----	----	-------	-------------	-----------	-----------	--------

Donde FF son los byte de sincronización y AA es le byte que indica el inicio de los datos.

En la figura III.27 nuestra el circuito transmisor

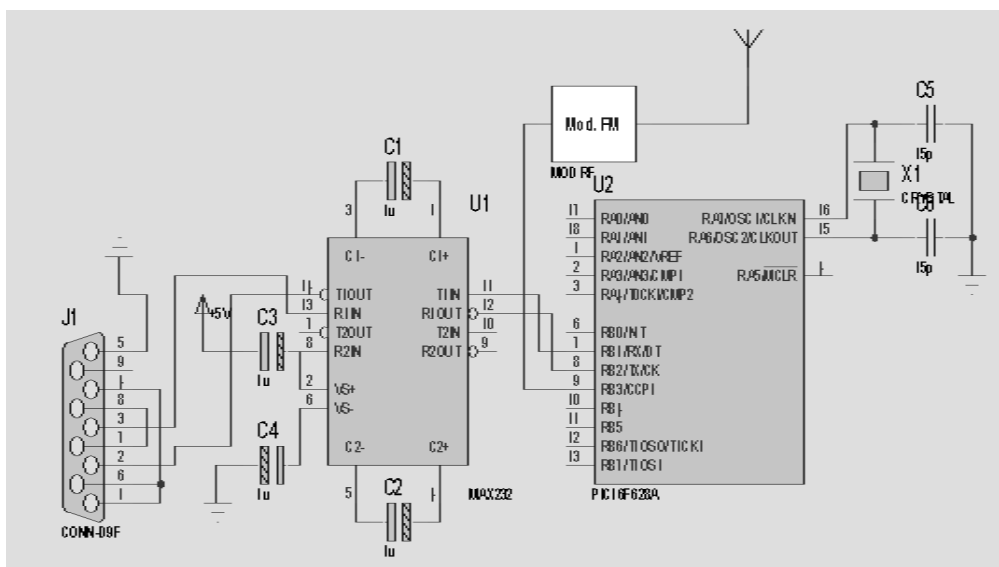


Figura III.27 Circuito transmisor

3.1.1 Implementación de la Aplicación

La implementación de la aplicación de control de los robots se la realizara en Visual Basic 6.0 en la cual se utilizara una interface para el manejo del puerto serial, en el cual transmitiremos de información los cuales serán procesados y presentados a los usuarios en el monitor.

3.1.1.1 Acceso al Puerto Serial a través de V Basic

Para poder acceder al puerto serial y así poder enviar datos utilizando una aplicación creada en Visual Basic, se hace uso del control **MS COMM**, (ver figura III.28) el cual trae incorporadas todas las funciones para configurar el puerto. Es gracias a este control que el manejo del puerto serial se facilita enormemente. Las propiedades más importantes de este control son las siguientes:

- **ComPort:** Activa y regresa el número del puerto serial (Comm1, Comm2)
- **PortOpen:** Activa y regresa el acceso al puerto.
- **Input:** Regresa los caracteres del buffer receptor.
- **Output:** Escribe una cadena sobre el buffer Transmisor.
- **Settings:** Activa y regresa la razón de Baudios, paridad, número de bits, bits de paro.

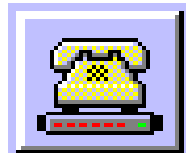


Figura III.28 Icono componente MS COMM

3.1.2 Iniciación de la aplicación

En la figura III.30 muestra la pantalla de la aplicación

Antes de empezar a transmitir debemos seleccionar el puerto COM que en el cual vamos a conectar el circuito; después procedemos a abrir el puerto dando clic en el botón (ver figura III.30).

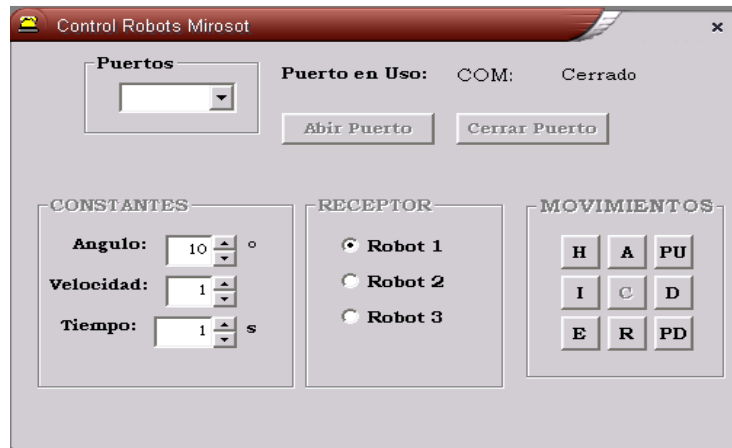


Figura III.29 Aplicación



Figura III.30 Selección de puertos

Luego de seleccionar el puerto debemos definir el robot que se controlara (ver figura III.31).



Figura III.31 Selector de robot

Posteriormente asignaremos las constantes de movimiento (ver figura III.32), ángulo de giro en el caso de existir giro, la velocidad a la cual se moverá, existiendo diez diferentes niveles de velocidad a escoger; y el tiempo en segundos en el cual realizara la acción siendo 10 segundos el tiempo máximo que se moverá.

Ya definidos los parámetros de movimiento procedemos a ejecutar (ver tabla XIII) las acciones pulsando cualquiera de los ocho diferentes botones de movimientos (ver figura III.33)

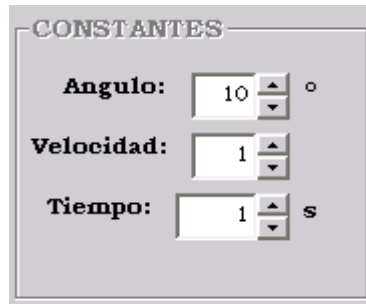


Figura III.32 Constantes de movimiento



Figura III.33 Dirección de movimiento

Tabla XIII Códigos de movimientos

A	Adelante ↑	01100
R	Reversa ↓	01101
D	Derecha ←	01110
I	Izquierda →	01111
PU	Curva delantera – derecha ↗	10000
H	Curva delantera – izquierda ↖	10001
PD	Curva reversa –derecha ↘	10010
E	Curva reversa –Izquierda	10011

3.2 Diseño e Implementación del circuito controlador del robot

El circuito controlador del robot consta del modulo receptor RPL 434 una unidad decodificadora, una unidad de control de velocidad y driver de los motores (ver figura III.34).

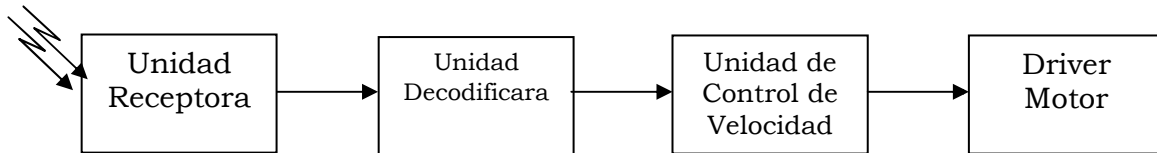


Figura III.34 Diagrama de bloque del receptor

La figura IV.35 muestra el diagrama del circuito controlador del robot

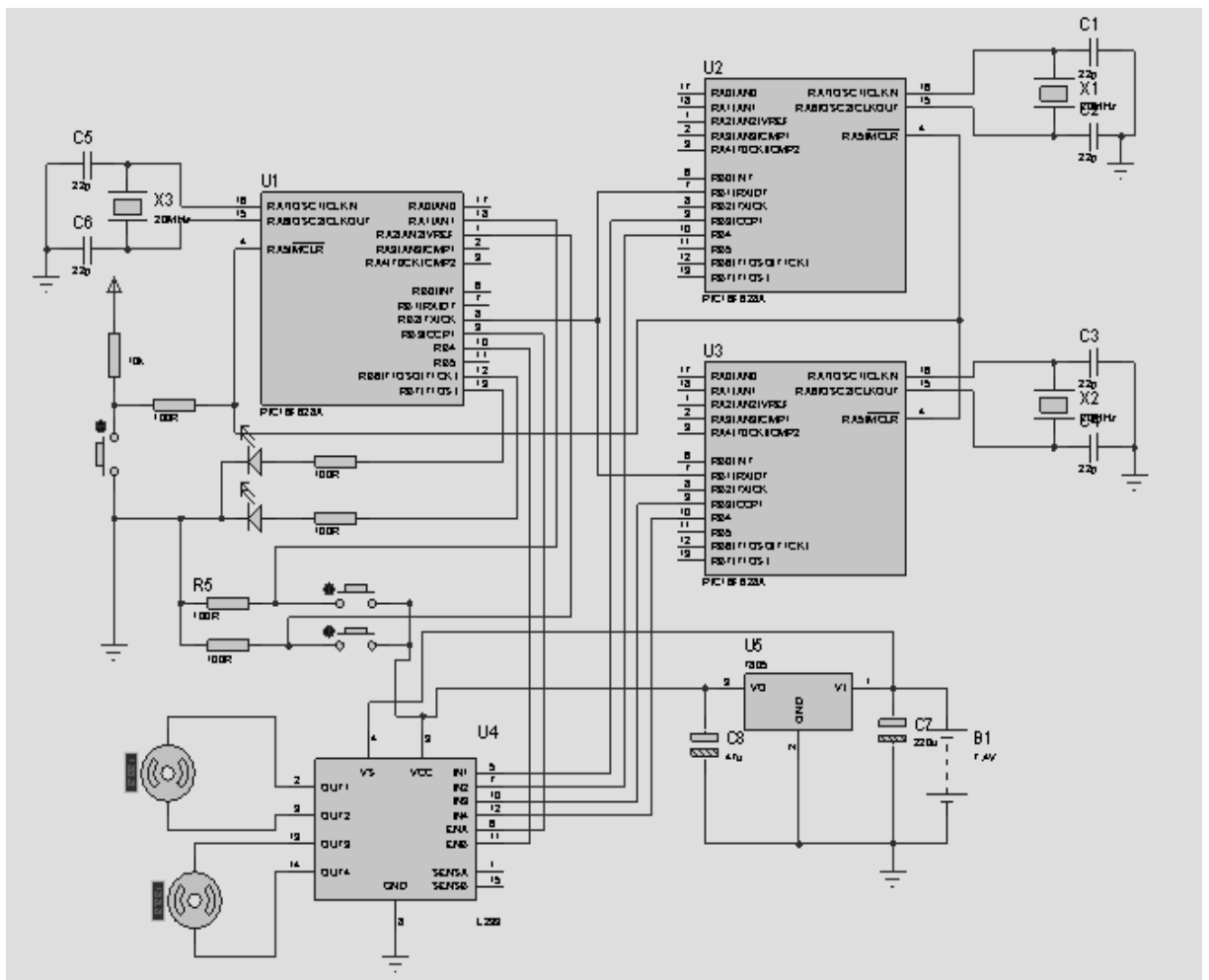


Figura III.35 Circuito controlador del robot

3.2.1 Implementación del Receptor

El modulo receptor RPL 434 opera a 433.92 MHz, y tiene una sensibilidad de 3uV, opera con una voltaje de alimentación de 4.5V y 5.5V de corriente continua, posee una salida lineal y una digital, además contiene un capacitor variable para el ajusta de la frecuencia de recepción utilizando un destornillador plástico.

3.2.2 Diseño de Unidad decodificadora

La unidad decodificadora es un microcontrolador PIC 16F628A el cual de encarga de recibir los datos procedentes del modulo de RF y procesar la información, para comparar si el código del robot enviado corresponde con el seleccionando de la EEprom del PIC y así decidir si continua con la ejecución de las acciones. También se de accionar los motores.

La figura III.36 muestra la estructura del programa del microcontrolador en cargado de recoger y procesar los datos recibidos desde la PC.

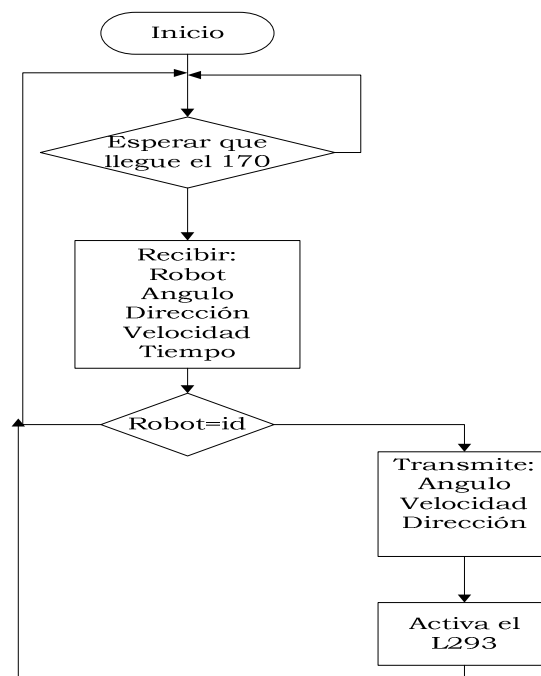


Figura III.36 Diagrama de flujo de la unidad decodificadora

3.2.3 Diseño del control de velocidad

El control de velocidad es un microcontrolador 16F628A, que es el encargado de ajustar la velocidad (ver tabla XIV.10) de funcionamiento del robot mediante el uso de PWM

La figura III.37 muestra la estructura del programa del microcontrolador encarga del control del velocidad de los motores para el funcionamiento del robot

Tabla XIV.10 Asignación de velocidades

Velocidad	Ciclo de servicio
1	111
2	127
3	143
4	159
5	175
6	191
7	207
8	223
9	239
10	255

3.2.3.1 Modulación de amplitud de pulso (PWM)

La *modulación por anchura de pulsos* (ó PWM, del ingles *pulse-width modulation*) es una técnica de modulación en la que se modifica el ciclo de trabajo de una señal periódica para, entre otras cosas, variar la velocidad de un motor.

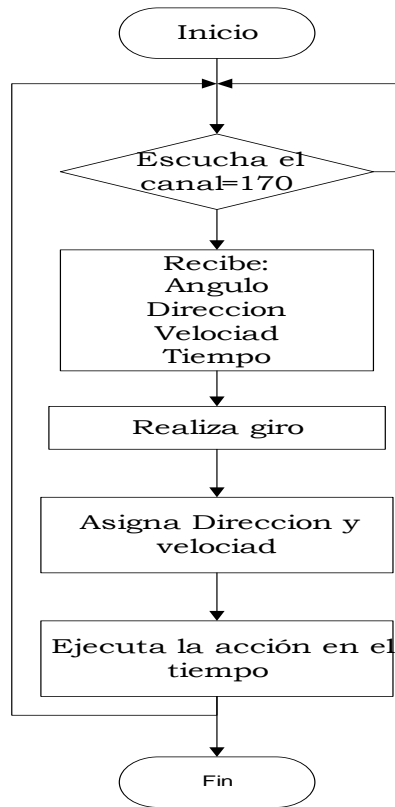


Figura III.37 Diagrama de Flujo del control de velocidad

El ciclo de trabajo de una señal periódica es el ancho relativo de su parte positiva en relación al período [Figura III.38]. Cuando más tiempo pase la señal en estado alto, mayor será la velocidad del motor.

Este tren de pulsos, en realidad, hace que el motor marche alimentado por la tensión máxima de la señal durante el tiempo en que esta se encuentra en estado alto, y que pare en los tiempos en que la señal esta en estado bajo.

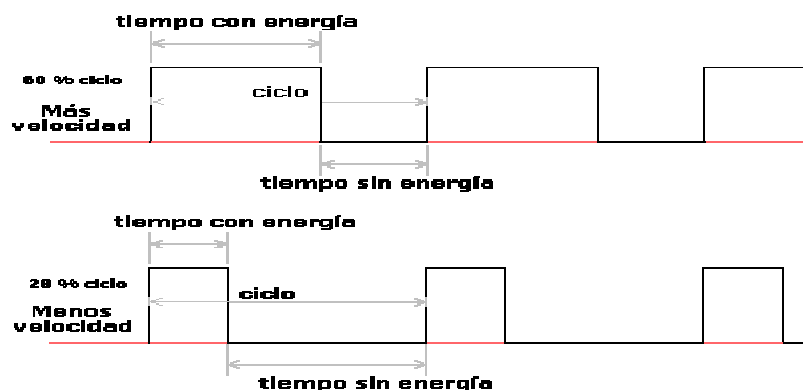


Figura III.38 Modulación PWM

3.2.3.2 Duty cycle

Recibe este nombre la relación de tiempos entre el estado alto y bajo de la señal utilizada. Se expresa como un porcentaje entre el periodo y el ancho del pulso. Cuando el Duty Cycle es cercano al 100%, el motor girara a una velocidad cercana a la máxima, ya que la tensión promedio aplicada en sus bornes será casi igual a V. Si el Duty Cycle se aproxima a 0%, el motor girara muy despacio, ya que la tensión promedio será casi cero.

3.2.3.3 Parámetros Importantes

Algunos parámetros importantes de un PWM son:

- ✓ La relación de amplitudes entre la señal portadora y la moduladora, es recomendable que la señal moduladora no supere el valor pico de la portadora y que esté centrada en el valor medio de ésta.
- ✓ La relación de frecuencias, donde en general se recomienda que la relación entre la frecuencia de la portadora y la de señal sea de 10 a 1

3.2.2.2 Aplicaciones

PWM, los circuitos funcionales pueden controlar fuentes conmutadas, controles de motores, controles de elementos termoelectricos, choppers para sensores en ambientes ruidosos y algunas otras aplicaciones.

3.3 Construcción del robot

El cuerpo del robot fue construido de aluminio, con unas dimensiones de 4 x 6 Cm, dos de 7mm de espesor para poder soportar las plaquetas, y dos de 2mm de espesor para el montaje de los motores (ver figura III.39)



Figura III.39 Paredes del robot

Las ruedas (ver figura III.40) tienen una dimensión de 4,5 cm de diámetro, con un engrane de 1,7 cm. de diámetro.



Figura III. 40 Rueda del Robot

Los motores DC están sujetos por medio de tornillos en las láminas de aluminio de 2 mm de espesor.

La velocidad de los motores tiene una reducción de 5,2:1 (ver figura III.41)



Figura III.41 Rueda con el engrane

3.4 Implementación de las plaquetas

La implementación y el diseño de las plaquetas tanto para el transmisor de la PC como para el robot receptor se encuentran diseñados en el Express PBC.

En la siguiente figura muestra el impreso del transmisor

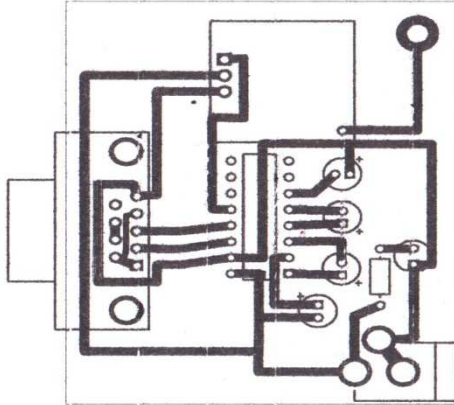


Figura III.42 PBC del transmisor

Por motivo del tamaño de los robots categoría Mirobot el circuito se lo ha dividido en dos plaquetas las cuales se muestran en la figura III.43 y III.44

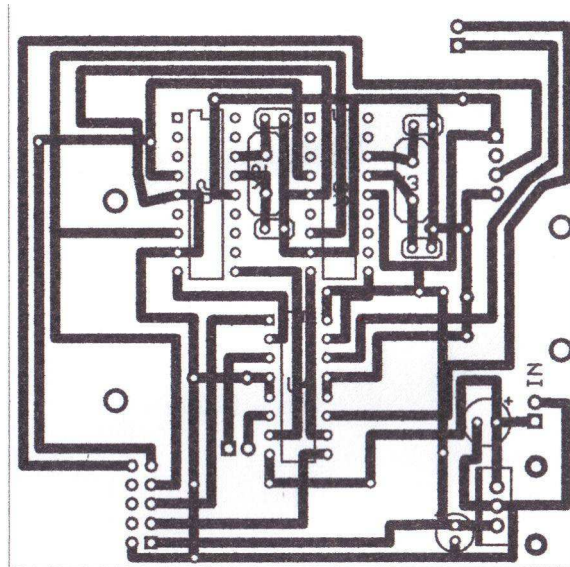


Figura III.43 PBC controlador del robot 1° parte

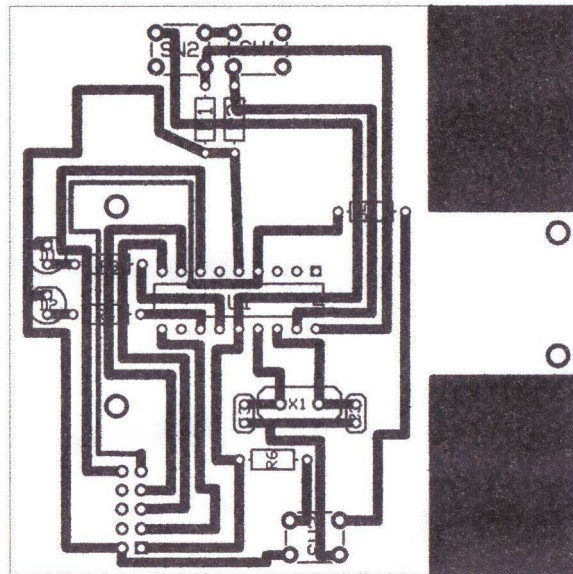


Figura III.44 PBC controlador del Robot 2° parte

3.5 Pruebas y Resultados

El transmisor tiene un alcance de por lo menos 5 metros probados a una frecuencia de 433MhZ; no existe el problema de funcionar con otro transmisor que no se el diseñado en este proyecto, debido a que los datos se encuentran debidamente codificados. La transmisión es unidireccional

Como se dijo anteriormente la transmisión es por el puerto serial de la PC.

Si el código del robot enviado no es el mismo con el código que esta seleccionado en el robot el dispositivo regresa a esperar una nueva transmisión

El robot Mirobot comercial es mayor precisión y menor velocidad, que le robot diseñado

CONCLUSIONES

1. En la implementación del transmisor Radio Frecuencia (RF) se debe tener en consideración, los datos cantidad de datos a enviar, la distancia máxima a transmitir y la tasa de bit a la que se va transmitir.
2. El receptor debe estar en perfecta sincronización con el transmisor para que pueda existir una buena comunicación.
3. En el cuerpo del robot se debe realizar con una gran precisión para que pueda existir un perfecto acoplo entre el motor y los engranes.
4. Para el control de velocidad el microcontrolador nos ayuda a ejecutar el PWM y de esta manera poder aumentar o disminuir la velocidad de los motores
5. Los movimientos del robot se debe tener en cuenta la velocidad del a la que se va a mover y el tiempo que va realizar la acción .

RECOMENDACIONES

- ✓ Se recomienda utilizar microcontroladores en el proyecto, ya que hace más fácil el incremento de nuevos módulos actuadores ya que por su fácil remoción y programación agilizará el entendimiento e implementación de todo el sistema domótico.
- ✓ Para el uso de la radio frecuencia se recomienda el conocimiento de las frecuencias para uso aficionada, y así evitar interferencias de otras frecuencias
- ✓ El uso del microcontrolador para el control del velocidad de los motores DC mediante el uso de PWM

RESUMEN

Se ha diseñado e implementado de un robot categoría Mirosot controlado por radio frecuencia, con la finalidad de diversión, adaptando la tecnología existente en nuestro país, y así poder reducir costos.

Para desarrollar la investigación se utilizó equipo de computación, el software Microcode Studio para programar los PIC; el software Visual Basic para realizar la aplicación del transmisor, el circuito MAX232 (conversor TTL); para el diseño del robot se utilizó microcontroladores PIC 16F628A y el L293D.

El diseño de robot consta de un módulo de transmisión de datos y uno de recepción. El módulo de transmisión consta de una aplicación en la PC que dispone de una interface través del puerto serial, el conversor TTL y el transmisor de radio frecuencia. El módulo receptor consta de el receptor de radio frecuencia, una unidad decodificadora (un microcontrolador PIC 16F628A) que es la encargada de la recolección de datos enviados desde receptor de radio frecuencia para luego procesarlos; también consta de una unidad de control de velocidad que es la encargada de disminuir o aumentar la velocidad de movimiento del robot, y el manejador de los motores el cual se encarga de accionarlos. En base a las pruebas realizadas se observó que el módulo transmisor y el receptor tiene un alcance aproximado de 5 m, logrando su normal funcionamiento.

El robot diseñado alcanzó un 85% aproximadamente de rendimiento con respecto al robot Mirosot comercial, disminuyendo el costo.

SUMMARY

It is designed or implemented a robot Mirobot category; it is controlled by a radio frequency, with the objective of entertainment adapting the technology available in our country, in order to reduce costs.

In order to develop the investigation it used computing equipment, the Microcode Studio software to program the PIC, the Visual Basic software to make the application of the transmitter, the MAX232 circuit (TTL convert); for the design of robot was a used microcontroller: PIC 16F628A and the L293D.

The design of the robot was done by a data's transmitter module and one of reception.

The transmitter module was done with one application in the PC which has an interface through the serial port, the converter TTL and the transmitter the radio frequency. The receiver module consists of radio frequency, a decoder which is responsible for the collection of data sent from the receiver for radio frequency, and then prosecutes them; also consists of a speed control unit that is responsible to reduce or increase the robot movement, and the driver of the engines which is responsible for handles. Based on test carried out we showed that the module transmitter and receiver have a range of approximately 5 m achieving its normal operation.

The robot was designed around 85% performance with regard to robot Mirobot business, lowering the cost.

BIBLIOGRAFIA

- ✓ Angulo, Joan y otros. Microcontroladores PIC, la solución en un chip, Barcelona: Parinfo, 1997. pp. 204.
- ✓ Corrales, Santiago, Electrónica Práctica con Microcontroladores Pic, s.d.edit, 2006. pp. 99, 129
- ✓ Horenstein, Mark. Circuitos y Dispositivos Microelectrónicos. 2a ed. México: Prentice Hall Hispanoamérica, 1997. pp. 1127.
- ✓ Manual De Programación de Microcontroladores (Hojas Fotocopiadas).
- ✓ National Instrument , Datasheet L293D, USA Febrero 2003 (Catalogo)

BIBLIOGRAFÍA DE INTERNET

- ✓ Comunicación Serial
http://robots-argentina.com.ar/Comunicacion_max232.htm
(2008 - 03 13)
- ✓ L293D
http://www.datasheetcatalog.net/es/datasheets_pdf/L/2/9/3/L293D.shtml
(2008 - 04 - 02)
- ✓ Medios de transmisión
[http://medios-de-transmision--componentes-55.asp_files/estilo.css"](http://medios-de-transmision--componentes-55.asp_files/estilo.css)
(2008- 05 - 21)
- ✓ Rpl434
<http://www.laipac.com/Downloads/Easy/rlp434.pdf>
(2008-15-09)
- ✓ Tipos De Modulaciones Digitales
<http://www.ubicom.tudelft.nl/MMS/Docs/introOFDM.pdf>
(2008 - 05 - 10)
- ✓ TPL434A
<http://www.laipac.com/Downloads/Easy/tlp434a.pdf>
(2008-15-09)

ANEXOS

ANEXO 1

CODIGO FUENTE DE LA PC

Esta aplicación se encuentra desarrollada en Visual Basic utilizando el componente **MScomm** que es el objeto que permite enviar y recibir del puerto serial de la PC

```
Private Sub Combo1_Click()  
puerto = Combo1.ListIndex  
puerto = puerto + 1  
Command1.Enabled = True  
End Sub
```

```
Private Sub Command1_Click()  
MSComm1.CommPort = puerto  
MSComm1.PortOpen = True  
Etiq_Puerto.Caption = "COM" + Str$(puerto) + ":"  
Etiq_Estado.ForeColor = &HFF& 'Color verde  
Etiq_Estado.Caption = "Abierto"  
Command1.Enabled = False  
Command13.Enabled = True  
Frame1.Enabled = True  
Frame2.Enabled = True  
Frame4.Enabled = True  
Combo1.Enabled = False  
End Sub
```

```
Private Sub Command10_Click()  
direccion = 19  
MSComm1.Output = Chr(170)  
MSComm1.Output = Chr(Robots)  
MSComm1.Output = Chr(angulo.Text)  
MSComm1.Output = Chr(direccion)  
MSComm1.Output = Chr(velocidad.Text)  
MSComm1.Output = Chr(tiempo.Text)  
  
End Sub
```

```
Private Sub Command11_Click()  
direccion = 13  
MSComm1.Output = Chr(170)  
MSComm1.Output = Chr(Robots)  
MSComm1.Output = Chr(angulo.Text)  
MSComm1.Output = Chr(direccion)  
MSComm1.Output = Chr(velocidad.Text)  
MSComm1.Output = Chr(tiempo.Text)  
End Sub
```

```
Private Sub Command12_Click()
direccion = 18
MSComm1.Output = Chr(170)
MSComm1.Output = Chr(Robots)
MSComm1.Output = Chr(angulo.Text)
MSComm1.Output = Chr(direccion)
MSComm1.Output = Chr(velocidad.Text)
MSComm1.Output = Chr(tiempo.Text)
End Sub
```

```
Private Sub Command13_Click()
MSComm1.PortOpen = False
Etiq_Puerto.Caption = "COM" + ":"
Etiq_Estado.ForeColor = &H80000012           'Color verde
Etiq_Estado.Caption = "Cerrado"
Command1.Enabled = True
Command13.Enabled = False
Frame1.Enabled = False
Frame2.Enabled = False
Frame4.Enabled = False
Combo1.Enabled = True
```

```
End Sub
```

```
Private Sub Command14_Click()
angulo.Text = angulo.Text + 1
End Sub
```

```
Private Sub Command2_Click()
Text1.Text = Robots
End Sub
```

```
Private Sub Command3_Click()
MSComm1.Output = Chr(Text1.Text)
End Sub
```

```
Private Sub Command4_Click()
direccion = 17
MSComm1.Output = Chr(170)
MSComm1.Output = Chr(Robots)
MSComm1.Output = Chr(angulo.Text)
MSComm1.Output = Chr(direccion)
MSComm1.Output = Chr(velocidad.Text)
MSComm1.Output = Chr(tiempo.Text)
End Sub
```

```
Private Sub Command5_Click()
direccion = 12
MSComm1.Output = Chr(170)
MSComm1.Output = Chr(Robots)
MSComm1.Output = Chr(angulo.Text)
```

```
MSComm1.Output = Chr(direccion)
MSComm1.Output = Chr(velocidad.Text)
MSComm1.Output = Chr(tiempo.Text)
End Sub
```

```
Private Sub Command6_Click()
direccion = 16
MSComm1.Output = Chr(170)
MSComm1.Output = Chr(Robots)
MSComm1.Output = Chr(angulo.Text)
MSComm1.Output = Chr(direccion)
MSComm1.Output = Chr(velocidad.Text)
MSComm1.Output = Chr(tiempo.Text)
End Sub
```

```
Private Sub Command7_Click()
direccion = 15
MSComm1.Output = Chr(170)
MSComm1.Output = Chr(Robots)
MSComm1.Output = Chr(90)
MSComm1.Output = Chr(direccion)
MSComm1.Output = Chr(velocidad.Text)
MSComm1.Output = Chr(tiempo.Text)
End Sub
```

```
Private Sub Command8_Click()
Text1.Text = puerto
End Sub
```

```
Private Sub Command9_Click()
direccion = 14
MSComm1.Output = Chr(170)
MSComm1.Output = Chr(Robots)
MSComm1.Output = Chr(90)
MSComm1.Output = Chr(direccion)
MSComm1.Output = Chr(velocidad.Text)
MSComm1.Output = Chr(tiempo.Text)
```

```
End Sub
```

```
Private Sub Form_Load()
Robots = 1
End Sub
```

```
Private Sub Option1_Click()
Robots = 1
```

```
End Sub
```

```
Private Sub Option2_Click()
Robots = 2
```

End Sub

```
Private Sub Option3_Click()  
Robots = 3
```

End Sub

```
Private Sub SpinButton1_SpinUp()  
angulo.Text = angulo.Text + 4  
If angulo.Text > 88 Then  
    angulo.Text = 4  
End If  
End Sub
```

```
Private Sub SpinButton1_SpinDown()  
angulo.Text = angulo.Text - 4  
If angulo.Text < 4 Then  
    angulo.Text = 88  
End If  
End Sub
```

```
Private Sub SpinButton2_SpinDown()  
tiempo.Text = tiempo.Text - 1  
If tiempo.Text < 1 Then  
    tiempo.Text = 10  
End If
```

End Sub

```
Private Sub SpinButton2_SpinUp()  
tiempo.Text = tiempo.Text + 1  
If tiempo.Text > 10 Then  
    tiempo.Text = 1  
End If
```

End Sub

```
Private Sub SpinButton3_SpinDown()  
velocidad.Text = velocidad.Text - 1  
If velocidad.Text < 1 Then  
    velocidad.Text = 10  
End If  
End Sub
```

```
Private Sub SpinButton3_SpinUp()  
velocidad.Text = velocidad.Text + 1  
If velocidad.Text > 10 Then  
    velocidad.Text = 1  
End If  
End Sub
```

ANEXO 2

CODIGO FUENTE DE LA UNIDAD DECODIFICADOR

Se muestra el código fuente con el cual se encuentran programados los microcontroladores PIC 16F628A en Micro Code Studio.

Las líneas de código presentes a continuación, describen el funcionamiento del microcontrolador encargado de recibir los bits transmitidos desde la PC al PIC 16F628A en forma serial, el cual transmitirá al aire.

```
@DEVICE
PIC16F628A,HS_OSC,WDT_OFF,PWRT_OFF,MCLR_ON,BOD_OFF,CPD_OFF,P
ROTECT_OFF
include "modedefs.bas"
define OSC 20
SYNCH con 170
start var byte
dat var byte
Angulo var byte
Direccion var byte
Velocidad var byte
Tiempo var byte
tiempo1 var word
id var byte
robot var byte
CMCON=7 ; Digitaliza el puerto A
trisb=%00000011
trisa=%11111111
read 0,id
read id,robot
portb.6=robot.0
portb.7=robot.1
inicio:

if (porta.0==1)and(porta.1==0)then
  if id<3 then
    id=id+1
    write 00,id
    read id,robot
    portb.6=robot.0
    portb.7=robot.1
    pause 300
  endif
endif
if (porta.1==1)and(porta.0==0)then
```

```

if id>1 then
id=id-1
write 00,id
read id,robot
portb.6=robot.0
portb.7=robot.1
pause 300
endif
endif
SERIN2 portb.0,396,[start]
if start <> 170 then
goto inicio
endif
SERIN2 portb.0,396,[dat]
SERIN2 portb.0,396,[Angulo]
SERIN2 portb.0,396,[Direccion]
SERIN2 portb.0,396,[Velocidad]
SERIN2 portb.0,396,[tiempo]

if dat<>robot then
pause 300
goto inicio
endif
if (Angulo > 90) then
pause 300
goto inicio
endif
if (direccion < 12)and(Direccion > 19) then
pause 300
goto inicio
endif
if (Velocidad>10) then
pause 300
goto inicio
endif
if (tiempo > 10) then
pause 300
goto inicio
endif
SEROUT PortB.2,T2400,[255,255]
SEROUT PortB.2,T2400,[170,Angulo]
SEROUT PortB.2,T2400,[direccion]
SEROUT PortB.2,T2400,[Velocidad]
SEROUT PortB.2,T2400,[tiempo]

if (direccion=12) or (direccion=13) then
tiempo1=tiempo*1000
else
tiempo1=tiempo*1000+angulo*5
endif
portb=portb+24

```

```
pause tiempo1  
portb=portb-24
```

```
goto inicio  
end
```

Anexo 3

CODIGO FUENTE DE LA UNIDAD DE CONTROL DE VELOCIDAD

```
@DEVICE
PIC16F628A,HS_OSC,WDT_OFF,PWRT_OFF,MCLR_ON,BOD_OFF,CPD_OFF,P
ROTECT_OFF
include "modedefs.bas"
define OSC 20
dato var byte
Angulo    var byte
Direccion  var byte
Velocidad var byte
Tiempo    var byte
V_real    var byte
i         var word
time     var byte
giro     var word
trisb=%0000010
inicio:
SERIN PortB.1,T2400,[170],angulo
SERIN PortB.1,T2400,direccion
SERIN PortB.1,T2400,velocidad
serin PortB.1,T2400,tiempo

select case velocidad
  case 1
    v_real= 111
  case 2
    v_real= 127
  case 3
    v_real= 143
  case 4
    v_real= 159
  case 5
    v_real= 175
  case 6
    v_real= 191
  case 7
    v_real= 207
  case 8
    v_real= 223
  case 9
    v_real= 239
  case 10
    v_real= 255
end select
'ejecuta giro

if direccion =14 then
  pause 400
```



```

endif
if direccion=15 then
    portb.4=0
    for I=1 to 400
        pwm portb.3,191,1
    next i
endif
if direccion =16 then
    giro=angulo*4
    pause giro
endif
if direccion=17 then
    giro=angulo*4
    portb.4=0
    for I=1 to giro
        pwm portb.3,191,1
    next i
endif
if direccion=18 then
    giro=angulo*4
    portb.3=0
    for I=1 to giro
        pwm portb.4,191,1
    next i
endif
if direccion=19 then
    giro=angulo*4
    pause giro
endif
'definicion del movimiento
time=tiempo*10
select case direccion
case 12
    portb.4=0
    for i=1 to time
        pwm portb.3,v_real,100
    next i
case 13
    portb.3=0
    for i=1 to time
        pwm portb.4,v_real,100
    next i
case 14
    portb.4=0
    'portb.3=0
    for i=1 to time
        pwm portb.3,v_real,100
    next i
case 15
    portb.4=0
    for i=1 to time
        pwm portb.3,v_real,100

```

```
    next i
case 16
  portb.4=0
  'portb.3=0
  for i=1 to time
    pwm portb.3,v_real,100
  next i
case 17
  portb.4=0
  for i=1 to time
    pwm portb.3,v_real,100
  next i
case 18
  portb.3=0
  for i=1 to time
    pwm portb.4,v_real,100
  next i
case 19
  portb.3=0
  for i=1 to time
    pwm portb.4,v_real,100
  next i
end select
goto inicio
end
```

Anexo 4

Hoja de Datos del TPL 434A



TPL 434A - Ultra Small Transmitter

Frequency : 433.92MHz

Spec:

- Frequency Range: 433.92 MHz
- Modulate Mode: ASK
- Circuit Shape: S/F
- Data Rate: 8Kbps
- Supply Voltage: 3~ 12 V
- Voltage:3V; Current:4.9mA
- Voltage:5V; Current:8.4mA
- Power Supply and All Input / Output Pins: -0.3 to 12.0 V
- Non-Operating Case Temperature: -10 to +85 centigrade
- Soldering Temperature: 230 centigrade (10 Seconds)
- Simple To Apply with Low External Count
- High Sensitivity Passive Design.

Applications:

- Wireless security systems
- Car alarms
- Remote gate controls
- Remote sensing
- Date communication
- Remote water/electric switch
- Pager system
- PDA data communicated

Absolute Maximum Ratings:

Rating	Value	Unit
Power Supply and All Input/Output Pins	-0.3 to +12.0	V
Non-Operating Case Temperature	-20 to +85	centigrade
Soldering Temperature (10 seconds)	230	centigrade

Electrical Characteristics:(T=25centigrade,Vcc=3.6V, Freq=318MHz)

Parameter	Sym.	Min.	Typ.	Max.	Unit
Operating Frequency (200KHz)	Vcc		433.92		MHz
Data Rate	ASK			8K	Kbps
Transmitter Performance(OOK@2.4kbps)					
Peak Input Current,12 Vdc Supply	I _{TP}			45	mA
Peak Output Power	P _O		10		mW
Turn On/ Turn Off Time	T _{ON} /T _{OFF}			1	US
Power Supply Voltage Range	Vcc	3		12	VDC
Operating Ambient Temperature	T _A	-20		+85	centigrade
Tx Antenna Out (3V) +2.4dB	Vcc				mA

Anexo 5

Hoja de Datos del RPL 434



RLP 434 - SAW Based Receiver

Frequency : 433.92MHz

Spec:

- Frequency Range: 433.92MHz
- Modulate Mode: ASK
- Circuit Shape: LC
- Data Rate: 4800bps
- Sensitivity: -106 dBm
- Channel Spacing: +/-500KHz
- Supply Voltage: 5V
- High Sensitivity Passive Design
- Simple To Apply with Low External Count

Applications:

- Wireless security systems
- Car alarms
- Remote gate controls
- Remote sensing
- Data communication
- Remote water/electric switch
- Pager system
- PDA data communicated

DC Characteristics :

Symbol	Parameter	Conditions	Min.	Typ.	Max.	Unit
Vcc	Operating Supply Voltage		4.9	5	5.1	
I Tot	Operating Supply Voltage			4.5		
V Data	Data Out	1 Data = +200 uA (High)	Vcc -0.5	Vcc		V
		1 Data = -10 uA (Low)			0.3	V

Electrical Characteristics:

Parameter	Sym.	Min.	Typ.	Max.	Unit
Operating Radio Frequency	FC		433.92		MHz
Sensitivity	Pref.		-108		dBm
Channel Width		-500		+500	KHz
Noise Equivalent BW	NEB		5	4	KHz
Baseboard Data Rate			3	KB/S	
Receiver Turn On Time				3	ms