

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DE TELECOMUNICACIÓN  
UNIVERSIDAD POLITÉCNICA DE CARTAGENA



**Proyecto Fin de Carrera**

# **Robot teleoperado de aplicación en entornos no estructurados**



AUTOR: Matías Pérez Ruiz  
DIRECTORES: Javier Toledo Moreo  
Ginés Doménech Asensi

01 / 2007



<b>Autor</b>	Matías Pérez Ruiz
<b>E-mail del Autor</b>	<a href="mailto:matias@telematik.jazztel.es">matias@telematik.jazztel.es</a>
<b>Director(es)</b>	Javier Toledo Moreo, Ginés Doménech Asensi
<b>E-mail del Director</b>	<a href="mailto:javier.toledo@upct.es">javier.toledo@upct.es</a>
<b>Codirector(es)</b>	
<b>Título del PFC</b>	Robot Teleoperado de aplicación en entornos no estructurados
<b>Descriptor(es)</b>	Micro-robot, microcontrolador PIC, tele-operación, cámara video
<p><b>Resumen</b></p> <p>El principal objetivo del proyecto es construir una plataforma robotizada que se desplazará accionada por dos servomotores conectados a las correspondientes ruedas y que dispondrá como sensores principales un sensor CMOS de imágenes, y los sensores necesarios para detectar choques con posibles obstáculos. El robot será teleoperado, tanto en sus desplazamientos como en la función del sensor CMOS. Las imágenes captadas serán transmitidas a un monitor situado en un emplazamiento remoto junto al teleoperador.</p> <p>Entre las disciplinas que abarca el proyecto se encuentran electrónica digital, sensores, fundamentos de computadoras, programación, y fundamentos de transmisión de señales y de telemática.</p>	
<b>Titulación</b>	Ingeniero Técnico Superior de Ingeniería de Telecomunicación, esp. Telemática
<b>Intensificación</b>	
<b>Departamento</b>	Electrónica, Tecnología de Computadoras y Proyectos
<b>Fecha de Presentación</b>	01- 2007

# ÍNDICE

<b>INTRODUCCIÓN</b> .....	11
<b>1 CONTENIDOS</b> .....	11
<b>PLANTEAMIENTO GENERAL</b> .....	13
<b>1 INTRODUCCIÓN</b> .....	13
<b>2 DESCRIPCIÓN FUNCIONAL</b> .....	14
<b>2.1 ESTACIÓN BASE</b> .....	14
<b>2.2 MICRO-ROBOT MÓVIL</b> .....	14
<b>ESTACIÓN BASE</b> .....	17
<b>1 INTRODUCCIÓN</b> .....	17
<b>2 DESCRIPCIÓN DEL SISTEMA</b> .....	17
<b>3 SISTEMA DE COMUNICACIÓN</b> .....	17
<b>4 SISTEMA VISUAL</b> .....	19
<b>5 INTERFAZ GRÁFICA</b> .....	20
<b>MICRO-ROBOT MÓVIL</b> .....	23
<b>1 DESCRIPCIÓN DE LOS COMPONENTES</b> .....	24
<b>1.1 PLACA CONTROLADORA MSX84</b> .....	24
<b>1.1.1 INTRODUCCIÓN</b> .....	24
<b>1.1.2 CARACTERÍSTICAS GENERALES</b> .....	25
<b>1.1.3 LISTADO MATERIALES</b> .....	25
<b>1.1.4 DESCRIPCIÓN FUNCIONAL</b> .....	27
<b>A. LA FUENTE DE ALIMENTACIÓN</b> .....	27
<b>B. EL MICROCONTROLADOR</b> .....	29
<b>1.1.5 ADAPTACIÓN DE PERIFÉRICOS A LA MSX84</b> .....	30
<b>1.2 MÓDULOS BLUETOOTH</b> .....	38
<b>1.2.1 MÓDULO RF AUREL TRANSEPTOR RTF SAW</b> .....	38
<b>1.2.2 BLUETOOTH RS232 INTERFACE PROMI-SD202</b> .....	39
<b>1.3 SISTEMA DE SENSADO</b> .....	41
<b>1.3.1 BUMPERS</b> .....	41

<i>1.3.2</i> <b>SENSORES O BUMPERS ELECTROMECAÑICOS</b> .....	42
<b>1.4 MOTORES</b> .....	44
<b>1.5 MICRO-CÁMARA Y MODULADOR RF CANAL 22UHF</b> .....	47
<i>1.5.1</i> <b>MICRO-CÁMARA</b> .....	47
<i>1.5.2</i> <b>MODULADOR RF CANAL 22 UHF AUREL MODELO MAV- UHF 479</b> .....	47
<b>1.6 DISEÑO ESTRUCTURA MÓVIL</b> .....	48
<b>1.7 SISTEMA DE ALIMENTACIÓN</b> .....	50
<b>1.8 MAX232</b> .....	51

## **SOFTWARE DEL SISTEMA**.....55

### **1 INTRODUCCIÓN**.....55

### **2 FUNCIONES DESEMPEÑADAS POR EL SOFTWARE BASE**.....55

### **3 FUNCIONES DESEMPEÑADAS POR EL MICROCONTROLADOR DEL MICRO-ROBOT**.....59

<b>3.1 ComprobarSensorD()</b> .....	60
<b>3.2 RS232_Inicializa()</b> .....	61
<b>3.3 TesteaTeclado()</b> .....	62
<b>3.4 Subrutina de atención a la interrupción del rb0/Int (TesteaSensores())</b> .....	63

## **FASES DEL PROYECTO**.....65

### **1 FASE 1**.....65

<b>1.1 RECOPIRAR INFORMACIÓN HARDWARE</b> .....	65
<i>1.1.1</i> <b>SENSORES TELEMÉTRICOS</b> .....	65
<i>1.1.2</i> <b>MOTORES</b> .....	70
<i>1.1.3</i> <b>COMUNICACIÓN INALÁMBRICA</b> .....	70
<i>1.1.4</i> <b>COMUNICACIÓN PUERTO SERIE PC</b> .....	72
<b>1.2 CONCLUSIÓN Y ELECCIÓN DE DISPOSITIVOS</b> .....	75
<b>1.3 DISEÑO FUNCIONAL DEL ROBOT</b> .....	76
<b>1.4 RECOPIRAR INFORMACIÓN SOFTWARE</b> .....	78
<i>1.4.1</i> <b>PROGRAMACIÓN COMUNICACIONES PC-PUERTOS</b> .....	78
<b>1.5 DISEÑO FÍSICO Y CONSTRUCCIÓN DEL ROBOT</b> .....	79
<i>1.5.1</i> <b>UBICACIÓN DE COMPONENTES</b> .....	79
<i>1.5.2</i> <b>OPTIMIZACIÓN DE ESPACIO EN EL ROBOT</b> .....	81
<i>1.5.3</i> <b>OPTIMIZACIÓN DEL CONSUMO</b> .....	82

### **2 FASE 2**.....82

<b>2.1 CONSTRUCCIÓN DEL ROBOT</b> .....	82
<b>2.2 SOFTWARE DEL MICROCONTROLADOR</b> .....	83
<b>2.3 SOFTWARE DEL PC</b> .....	83

### **3 FASE 3**.....83

<b>3.1 PRIMERA ETAPA</b> .....	84
<b>3.2 SEGUNDA ETAPA</b> .....	86
<b>3.3 TERCERA ETAPA</b> .....	87
<b>3.3.1 PUESTA EN MARCHA DEL ROBOT</b> .....	87

<b>PLANOS</b> .....	89
---------------------	----

<b>1 ESQUEMA INSTALACIÓN SISTEMAS DE SENSADO, MOTORIZACIÓN Y COMUNICACIÓN</b> .....	89
---	----

<b>2 ESQUEMA INSTALACIÓN SISTEMA VISIONADO</b> .....	90
--	----

<b>PRESUPUESTO</b> .....	91
--------------------------	----

<b>PLIEGO DE CONDICIONES</b> .....	93
------------------------------------	----

<b>1 INTRODUCCIÓN</b> .....	93
-----------------------------	----

<b>2 CONDICIONES DE LOS COMPONENTES</b> .....	93
---	----

<b>3 CONDICIONES DE REALIZACIÓN DEL ROBOT</b> .....	93
---	----

<b>4 CONDICIONES DE SOFTWARE</b> .....	93
--	----

<b>PRUEBAS, CONCLUSIONES Y TRABAJO FUTURO</b> .....	95
---	----

<b>1 FASE DE PRUEBAS</b> .....	95
--------------------------------	----

<b>2.1 PRUEBAS DE CONTROL REMOTO</b> .....	95
--	----

<b>2 CONCLUSIONES</b> .....	96
-----------------------------	----

<b>BIBLIOGRAFÍA</b> .....	99
---------------------------	----

# ÍNDICE DE FIGURAS

Figura 2.1	Esquema general.....	13
Figura 2.2	Esquema estructura de la Estación Base.....	14
Figura 2.3	Esquema estructura del Micro-Robot.....	15
Figura 3.1	Diagrama de bloques de la Estación base.....	17
Figura 3.2	Sistema de comunicaciones.....	18
Figura 3.3	Conexión RS232-Módulo Bluetooth.....	19
Figura 3.4	Modulador UHF canal 22 UHF.....	19
Figura 3.5	Tarjeta capturadora TV.....	20
Figura 3.6	Selección driver captura.....	21
Figura 3.7	Ventana principal aplicación gráfica.....	21
Figura 3.8	Campo de Botones.....	22
Figura 3.9	Ventana informativa.....	22
Figura 4.1	Esquema general vehículo móvil.....	23
Figura 4.2	Tarjeta MSx84.....	27
Figura 4.3	La fuente de alimentación.....	28
Figura 4.4	El micro-controlador PIC 16X84.....	29
Figura 4.5	Las entradas de la tarjeta MSx84.....	30
Figura 4.6	Descripción de pines del L293B.....	32
Figura 4.7	Esquema por bloques del L293B.....	33
Figura 4.8	El puente en H.....	34
Figura 4.9	Accionamiento de un motor de DC.....	35
Figura 4.10	Conexión del L293B en la tarjeta MSx84.....	36
Figura 4.11	Esquema general de conexión de los módulos RF aurel.....	38
Figura 4.12	Especificaciones mecánicas y técnicas módulos RF Aurel.....	38
Figura 4.13	Placas prueba módulos RF aurel.....	39
Figura 4.14	Módulo Bluetooth Promi-SD202.....	40
Figura 4.15	Símbolos más usuales de los bumpers.....	41
Figura 4.16	Los bumpers.....	42
Figura 4.17	Conexión de mecanismos con MSx84.....	42
Figura 4.18	Conexión de un bumper con MSx84.....	43
Figura 4.19	Conexión conectores J0-J4 a la tarjeta.....	43
Figura 4.20	Imagen real bumper utilizado.....	44
Figura 4.21	Características electro-mecánicas y dimensiones de los servos utilizados..	44
Figura 4.22	Imagen servo desmontado.....	45
Figura 4.23	Detalle del tope que hay que cortar.....	45
Figura 4.24	Posibles métodos de corte.....	46
Figura 4.25	Montaje de los engranajes.....	46
Figura 4.26	Conexión de un nuevo potenciómetro.....	46
Figura 4.27/28	Detalle de modificación del potenciómetro original.....	47
Figura 4.29	Modulador RF canal 22 UHF.....	48
Figura 4.30	Detalle de sistemas de acoplamiento y distribución de elementos en el vehículo.....	49
Figura 4.31	Sistema de transmisión-reducción del vehículo.....	49

Figura 4.32	Servo Hitec-HS422.....	50
Figura 4.33	Detalle de ubicación de la batería del vehículo.....	51
Figura 4.34	Conexiones interiores max232.....	51
Figura 4.35	Esquema eléctrico max232.....	52
Figura 4.36	Diagrama eléctrico que muestra la conexión PIC-RS232.....	53
Figura 5.1	Flujograma general programa micro-controlador del Micro-robot.....	60
Figura 5.2	Flujograma de ComprobarSensorD( ).....	61
Figura 5.3	Flujograma de RS232_Inicializa( ).....	62
Figura 5.4	Flujograma de TesteaTeclado( ).....	63
Figura 5.5	Flujograma de Subrutina de Interrupción.....	64
Figura 6.1	Esquema básico bumper.....	66
Figura 6.2	Imagen bumpers comerciales.....	66
Figura 6.3	Sensor infrarrojo Sharp GP2D02.....	67
Figura 6.4	Diagrama de conexiones sensor infrarrojo.....	67
Figura 6.5	Imagen real sensor I.R. ....	68
Figura 6.6	Sensor ultrasonidos alta frecuencia.....	68
Figura 6.7	Sensor distancias por ultrasonidos.....	69
Figura 6.8	Diagrama de tiempos SRF04.....	69
Figura 6.9	Tx Aurel mod. TX 4M50SA05IA.....	71
Figura 6.10	Interfaz bluetooth RS232 Promi-SD202.....	71
Figura 6.11	Encapsulación de las tramas.....	72
Figura 6.12	Tx del dato binario 10011010.....	72
Figura 6.13	Conexión de un Micro-Controlador al puerto serie de un pc .....	73
Figura 6.14	Conexión circuito adaptador de niveles.....	73
Figura 6.15	Conectores DB9 Macho y Hembra.....	74
Figura 6.16	Esquema conexión max232.....	74
Figura 6.17	Cable y programador conexión PC-MicroControlador.....	75
Figura 6.18	Robot estructura con ruedas.....	76
Figura 6.19	Robot estructura con múltiples patas.....	77
Figura 6.20	Robot estructura humana bípedo.....	77
Figura 6.21	Fase 1 construcción estructura del robot .....	79
Figura 6.22	Fase 2 construcción estructura del robot .....	79
Figura 6.23	Fase 3 construcción estructura del robot .....	80
Figura 6.24	Fase 4 construcción estructura del robot .....	80
Figura 6.25	Fase 5 construcción estructura del robot .....	81
Figura 6.26	Fase 6 construcción estructura del robot .....	81
Figura 6.27	Descripción etapas desarrollo robot.....	84
Figura 6.28	Adaptador para max232.....	86

## ÍNDICE DE TABLAS

Tabla 4.1	Características físicas Micro-Robot.....	24
Tabla 4.2	Características electro-mecánicas del Micro-Robot.....	24
Tabla 4.3	Alimentación MSx84.....	28
Tabla 4.4	Asociación conectores MSx84-Micro-controlador.....	31
Tabla 4.5	Tabla de verdad L293B.....	33
Tabla 4.6	Parámetros eléctricos driver L293B.....	33
Tabla 4.7	Características eléctricas driver L293B.....	34
Tabla 4.8	Combinaciones Jumper MSx84-L293B.....	36
Tabla 4.9	Características módulo bluetooth.....	40
Tabla 6.1	Información pines DB9.....	74
Tabla 6.2	Software desarrollo proyecto.....	78
Tabla 6.2	Correspondencia Tecla-combinación RA3-RA0.....	84
Tabla 6.4	Correspondencia Tecla pulsada-movimiento de las ruedas.....	85







# INTRODUCCIÓN

---

---

Hoy en día, cada vez es más común el uso de robots, tanto en la industria como en otros ámbitos de la vida cotidiana. La robótica se ha convertido en un elemento de uso esencial para muchas personas y empresas, debido a que el empleo de todo tipo de robots minimiza esfuerzos y facilita la ejecución de multitud de tareas que pudieran resultar peligrosas o repetitivas para el hombre. El auge de la robótica está intrínsecamente relacionado con el desarrollo de la tecnología, con el desmedido abaratamiento de costes y con la reducción del tamaño de los dispositivos micro-electrónicos.

Los robots móviles constituyen un área importante dentro de la robótica, ya que son empleados en multitud de aplicaciones, como pueden ser: investigación, entretenimiento, inspección y mantenimiento, etc. La principal característica que hace que este tipo de robot sea tan empleado es la autonomía, asociándose frecuentemente este término a la idea de inteligencia artificial, ya que un robot autónomo controla su trayectoria basándose en un sistema sensorial que le permita tener una visión del entorno, y sin necesidad de ser manejado por un usuario.

Cabe destacar, que para nosotros la profundización en el estudio de este tipo de robots tiene como base ahondar en el aprendizaje de electrónica, ya que esta área engloba la enseñanza de diferentes partes de esta ciencia, como pueden ser: electrónica digital, electrónica analógica, instrumentación y automatización. En el control de los robots móviles autónomos también intervienen diversas disciplinas tales como comunicaciones, inteligencia artificial, teoría de control, lenguajes de programación, etc.

## 1 Contenidos

La exposición del trabajo la estructuramos en los siguientes capítulos. En primer lugar, en el Capítulo Planteamiento General, se expone el planteamiento general del sistema, en el cual se presentan los diferentes bloques que componen el sistema desarrollado: estación base y micro-robot móvil. A continuación, en el Capítulo Estación Base, se describe la estación base. Se explicará la función de la estación y la de los elementos que la forman. Hablaremos de la interfase usuario-micro-robot, de la electrónica de control y del sistema de comunicaciones. Posteriormente, en el Capítulo Micro-Robot Móvil, pasaremos a describir el micro-robot móvil, centrándonos en el diseño del sistema mecánico del micro-robot y en su parte hardware. Especificaremos y justificaremos el tipo de vehículo utilizado, y hablaremos, entre otras cosas, de la construcción del vehículo, de las características técnicas más relevantes de los elementos que lo constituyen, del tipo de motores empleados, de la arquitectura de la placa micro-controladora que gestiona la velocidad y dirección del vehículo, y por último, analizaremos el sistema de percepción del entorno utilizado en el prototipo. El Capítulo Software del Sistema está dedicado al software. En cuanto al micro-robot, se explican las diferentes tareas desempeñadas por el microcontrolador, así como las subrutinas encargadas de realizarlas. Respecto a la estación base, comentaremos el

desarrollo de la aplicación gráfica desarrollada en Java, que actuará como interfaz usuario-PC. Para la explicación del software desarrollado nos basamos en flujogramas que esquematizan como se procesa la información. El Capítulo Fases del Proyecto estará dedicado a explicar las distintas fases del desarrollo del proyecto. El Capítulo Planos estará destinado al apartado de planos, mientras que en el Capítulo Presupuesto se detallará el presupuesto aproximado. El Capítulo Pliego de Condiciones estará dedicado al pliego de condiciones, y por último, el Capítulo Pruebas, Conclusiones y Trabajo Futuro, está dedicado a las pruebas, conclusiones y especificación de trabajo futuro.

Además en el CD se adjuntan los anexos correspondientes a cada uno de los capítulos en los que se estructura el proyecto.

# PLANTEAMIENTO GENERAL

## 1 Introducción.

El objetivo de esta introducción es proporcionar una visión global de todo el sistema, para lo cual, se procede a hacer un breve análisis de la estructura y funcionamiento de cada elemento que forma parte del prototipo.

Nos hemos planteado desarrollar un micro-robot móvil capaz de trabajar teleoperado por un usuario a través de un sistema de control remoto, situación en la cual el robot es teledirigido obedeciendo las órdenes de una aplicación Java controlada por un usuario. Además será capaz de variar su trayectoria basándose en la información proporcionada por un sistema de sensores electromecánicos. El robot también irá equipado con una micro-cámara que nos enviará información visual del entorno del mismo. Es por ello que la solución se compone de dos subsistemas representados en la figura 2.1. Estos dos subsistemas son:

- Estación base: Constituye la interfase con el usuario, y está implementada mediante un conjunto teclado-ratón, una tarjeta de televisión comercial, marca Pinnacle, modelo PCTV stereo y un enlace bluetooth con el micro-robot.
- Micro-robot: Constituido por el vehículo móvil, una placa microcontroladora, un sistema de sensores electromecánicos y una micro-cámara.

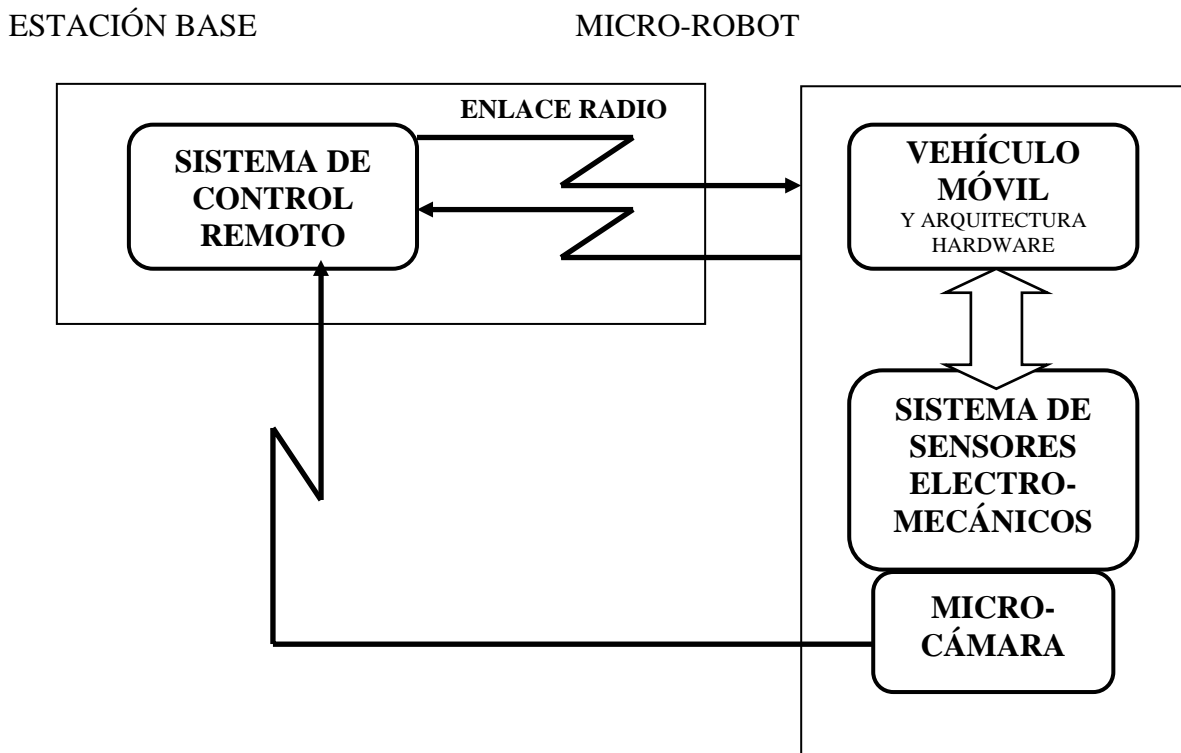


Figura 2.1. Esquema general

## 2 Descripción funcional.

Pasamos ahora a describir a nivel de bloques cada uno de estos subsistemas.

### 2.1 Estación base.

Es la parte del prototipo encargada de gobernar al micro-robot. La estación base manda consignas de giro al micro-robot, de forma que en éste no se produce cambio alguno de trayectoria si no lo ordena la estación base.

El sistema de control remoto está constituido por una aplicación software programada en lenguaje Java, por un conjunto teclado-ratón comercial de PC, y un emisor de radio bluetooth.

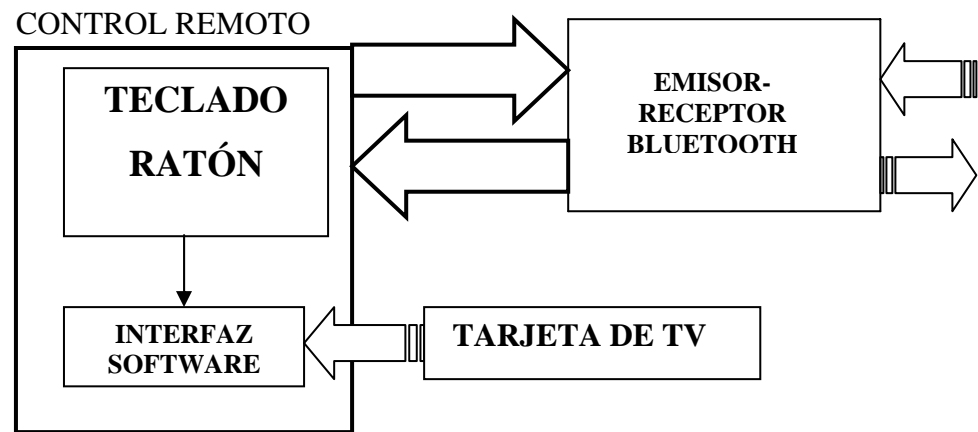


Figura 2.2. Esquema de la estructura de la Estación Base

Como mecanismo de comunicación se ha elegido un enlace en bluetooth, mediante el cual se establece un canal serie asíncrono. Existían otras posibilidades para establecer la comunicación, como la comunicación por infrarrojos, pero elegimos este tipo de enlace porque es menos sensible al entorno, es decir, puede haber objetos entre el emisor y el receptor y la comunicación no desaparece. La banda de frecuencia escogida en la transmisión vía radio ha sido 2.4 GHz, basándose esta elección en dos motivos:

- Se trata de una banda de frecuencia permitida.
- Hay multitud de dispositivos (par emisor-receptor) comerciales que trabajan a esta frecuencia.

### 2.2 Micro-robot Móvil.

Llamamos micro-robot móvil a la plataforma que utilizamos en el prototipo. Esta plataforma está compuesta por varios sistemas:

- Locomoción: Conjunto de elementos mecánicos y motores que conforman el vehículo móvil.
- Guiado: Formado principalmente por una placa microcontroladora MSx84 de Microsystems Engineering que se encargará de controlar el vehículo.
- Alimentación: Constituido por las baterías y un regulador lineal de tensión. Este bloque tiene la función de proporcionar autonomía al prototipo.
- Percepción: Formado por un conjunto de sensores electromecánicos, que obtiene información del entorno y posibilita que el micro-robot pueda realizar una tarea determinada. Además de ello, como hemos dicho anteriormente, el robot también va equipado con una cámara que envía las imágenes capturadas a la estación base.

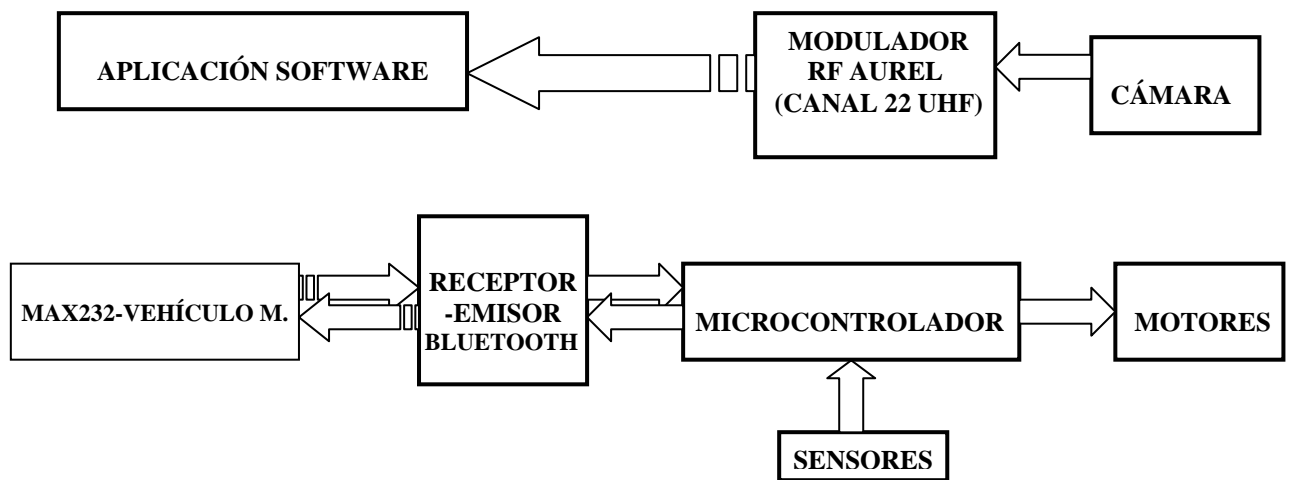


Figura. 2.3. Esquema de la estructura del Micro-robot

La estructura mecánica del vehículo se ha construido a partir de piezas de Mecano. Hemos usado un kit comercial por la facilidad que esto ofrece frente a tener que diseñar todas las piezas del sistema de locomoción (ruedas, engranajes, elementos de sujeción, ejes, acoplamientos, etc.). El apoyarnos en las piezas de Mecano para realizar el prototipo nos ha permitido centrarnos en el desarrollo de los sistemas digitales.

En el caso del micro-robot, una placa micro-controladora montada sobre el vehículo es la encargada de interpretar las señales procedentes del PC y de los sensores, generando las correspondientes señales de control que controlan los motores.

Como sistema de percepción se ha utilizado un conjunto de sensores electromecánicos.





# ESTACIÓN BASE

---

---

## 1 Introducción.

En este capítulo se describe la estación base implementada en el prototipo, la cual permite el control de la trayectoria del micro-robot por parte del operador.

La estación base es la parte fija del prototipo, y se comunica con el micro-robot móvil a través de un enlace bluetooth, también descrito en el presente capítulo. La estación permite al usuario gestionar el movimiento del vehículo.

## 2 Descripción del sistema.

La estación base está estructurada en diversos bloques, los cuales aparecen en la figura 3.1. Básicamente, está compuesta por un teclado y un ratón, que junto una aplicación gráfica programada en JAVA (que explicaremos más adelante en el apartado de software), hacen de interfaz entre el usuario y el PC, y por un emisor de radio, que nos permitirá comunicarnos con el micro-robot móvil.

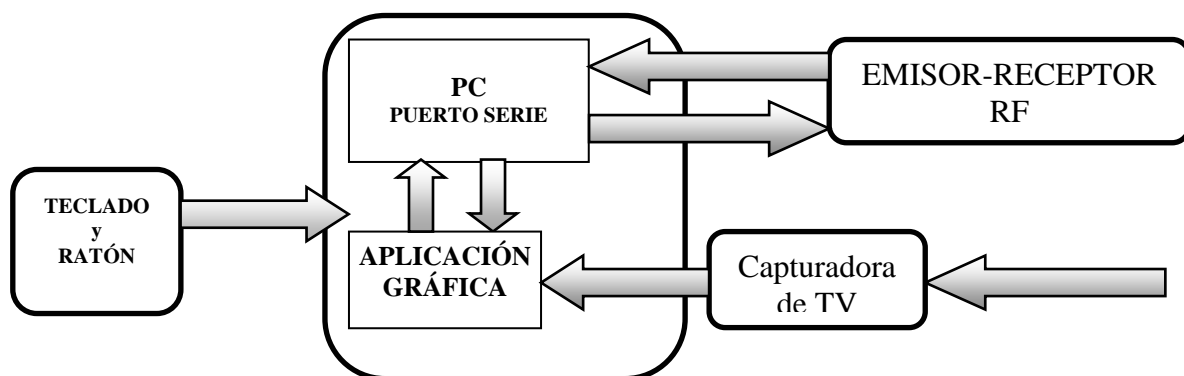


Figura 3.1. Diagrama de bloques de la Estación Base

## 3 Sistema de comunicación.

El objeto de este bloque es enlazar y comunicar la estación base con el micro-robot móvil, de forma que los comandos mandados por la estación base puedan llegar correctamente al micro-robot.

La comunicación que se debe establecer tiene que ser remota y asíncrona. Para satisfacer estas dos condiciones se ha optado por utilizar un enlace bluetooth, mediante el cual establecemos un canal serie asíncrono. Este enlace está compuesto por un par de interfaces bluetooth RS232, que son idénticos tanto para el caso de la estación base, como del robot. Explicaremos detalladamente estos módulos en el apartado del vehículo móvil.

Una comunicación serie se caracteriza porque se envían los datos de un carácter secuencialmente, es decir, un bit detrás de otro. Este tipo de comunicación tiene el inconveniente de ser lento respecto a la comunicación en paralelo, pero en cambio, simplifica bastante el diseño.

Una transmisión asíncrona se caracteriza porque en ella los datos pueden transmitirse en cualquier momento. El tiempo entre bits de un mismo carácter es el mismo, y lo que es asíncrono es el tiempo que separa la transmisión de caracteres. En la siguiente figura, presentamos un esquema del sistema de comunicaciones implementado en el prototipo:

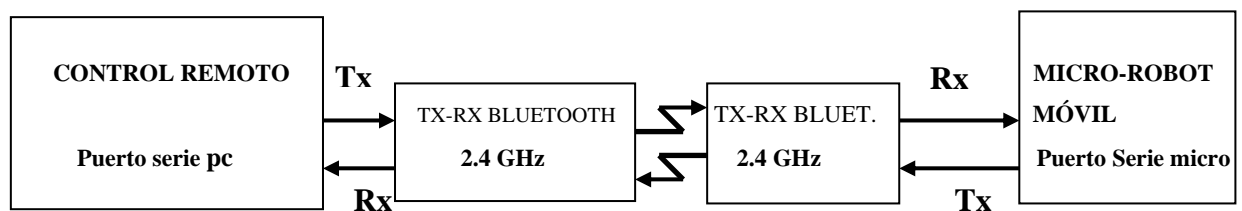
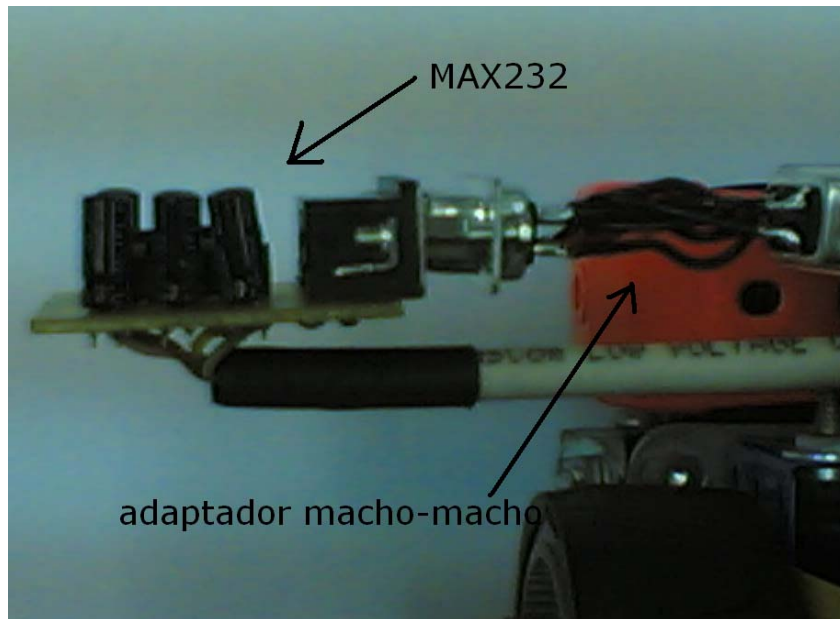


Figura 3.2. Sistema de comunicaciones

La comunicación que se establece es bidireccional (full-duplex), de forma que tanto la estación base, como el micro-robot móvil, reciben y emiten datos simultáneamente. La banda de frecuencia elegida es 2.4 Ghz, principalmente, porque se trata de una banda permitida legalmente. La forma en la que se transmiten los datos es la siguiente. El PC, a través de su puerto serie (Tx), manda los datos al emisor de radio, que modula esta información añadiéndole una portadora de 2.4 GHz. Esta información modulada pasa a la antena de emisión, encargada de convertir la corriente eléctrica que circula a través de ella en campos electromagnéticos. El medio de propagación de una comunicación en bluetooth es aéreo. La antena situada en el micro-robot, obtiene la información mandada por el emisor, y el receptor se encarga de demodular la señal. Esta señal demodulada pasa por el dispositivo MAX-232, que pasa de niveles RS232 (cerca de +12 y -12 V ) a voltajes TTL (0 a +5 V), antes de ser recogida por el microcontrolador a través de su puerto serie (Rx). De la misma manera ocurre la comunicación cuando el microcontrolador del robot-móvil envía datos al PC.

La velocidad del puerto serie de ambos microcontroladores es de 9600 Baudios, y está limitada por la frecuencia a la cual funciona el enlace radio. Se ha comprobado experimentalmente que con el par Tx-Rx que hemos utilizado, esta velocidad podría ser mayor, sin que hubiese pérdida de datos en la comunicación.



**Figura 3.3.** Conexión RS232-Módulo Bluetooth

#### 4 Sistema visual.

En cuanto al sistema visual, el PC donde realizaremos las pruebas estará equipado con una tarjeta capturadora de TV, la cual será la encargada de captar la señal procedente del canal 22 de UHF que emite el modulador incorporado en el micro-robot. Modulador que está formado, fundamentalmente, por tres partes:

- Oscilador: Genera la señal de frecuencia portadora.
- Amplificador de RF: Etapa de ganancia.
- Antena: Elemento de conexión con el espacio.

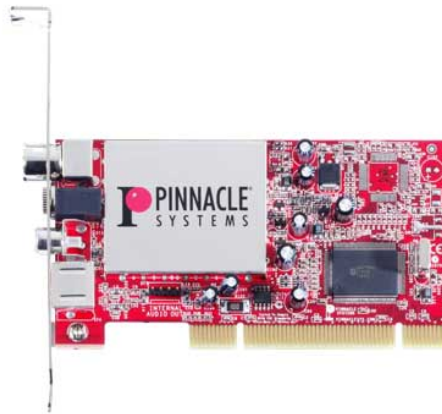
Como modulador se ha utilizado un dispositivo de Aurel, el Saw RF Transmitter Module (TX\_SAW BOOST 650200427). Se ha utilizado este dispositivo por su frecuencia de funcionamiento, 479.5 MHz. Este dispositivo tiene antena externa. Existen dispositivos con antena integrada, pero no era conveniente su utilización en nuestro prototipo, ya que no tenemos restricciones de espacio y se trata de dispositivos más caros y difíciles de diseñar.



**Figura 3.4.** Modulador UHF canal 22

Por otro lado, la función de la capturadora de TV es recibir una señal procedente del modulador, demodularla y proporcionar la información original. La antena receptora recoge las ondas electromagnéticas radiadas por la antena emisora, y las convierte en señales eléctricas, que son tratadas por la tarjeta de TV.

Para completar el par Tx-Rx, hemos utilizado como receptor una capturadora de TV marca Pinnacle, modelo PCTV stereo. Se ha elegido este dispositivo por su bajo coste.



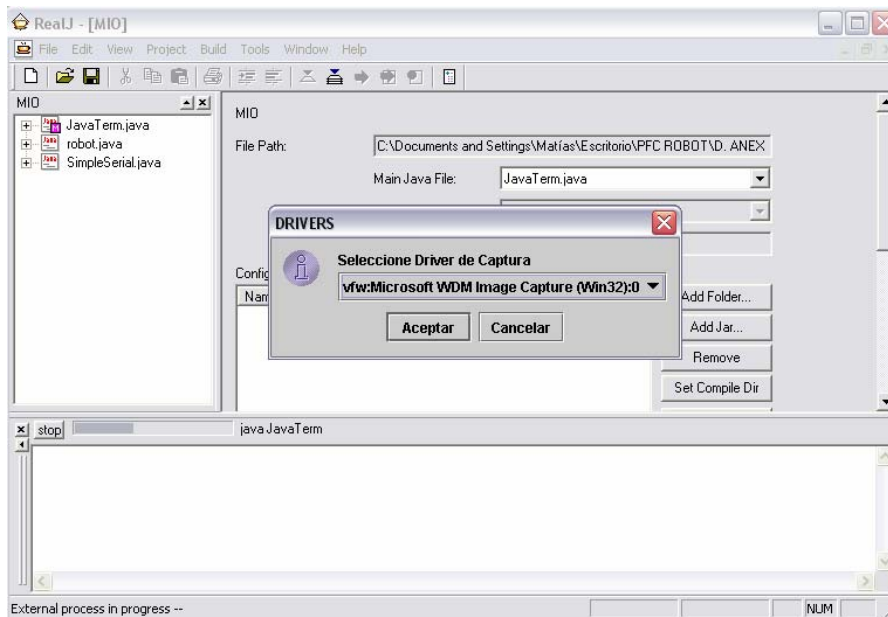
**Figura 3.5** Tarjeta Capturadora TV

Este dispositivo, se encontrará ubicado en una de las ranuras PCI de nuestro PC del laboratorio.

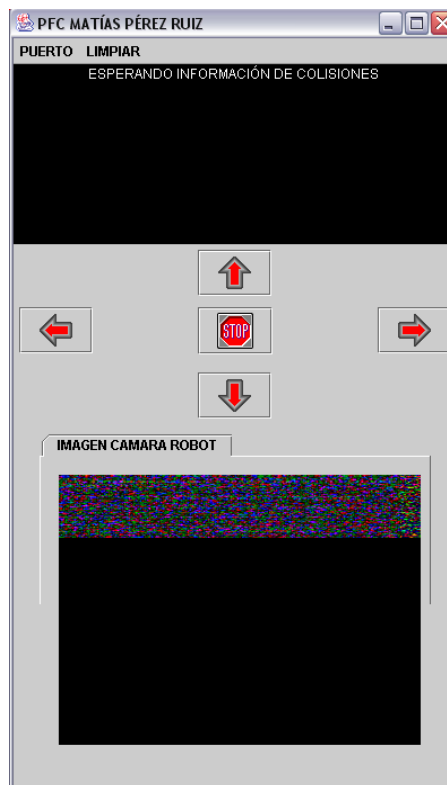
## 5 Interfaz Gráfica

A continuación explicaremos las partes que componen nuestra aplicación gráfica. La parte relativa al código en JAVA, la explicaremos en la sección correspondiente (Sección de Software).

1. La aplicación pide que el usuario seleccione el driver de captura (para el caso en el que tengas más de un dispositivo de captura de video). Lo podemos ver en la figura 3.6.
2. Una vez seleccionado el driver de captura, un mensaje en la ventana del programa RealJ avisará si la carga del driver ha sido exitosa o por el contrario fallida. En el caso de ser correcta, lo siguiente que aparece es una pantalla como la de la figura 3.7, donde



**Figura 3.6.** Selección driver de captura



**Figura 3.7.** Ventana principal aplicación gráfica

En la figura 3.7:

**PUERTO:** Nos proporciona la posibilidad de elegir distintos puertos por donde enviar/recibir la información que se intercambia con el microcontrolador del vehículo móvil.

**LIMPIAR:** Limpia el buffer donde nos llega la información de las posibles colisiones del robot.

**AREA DE TEXTO “ESPERANDO INFORMACIÓN DE COLISIONES”:**  
Informa de las colisiones del vehículo móvil.

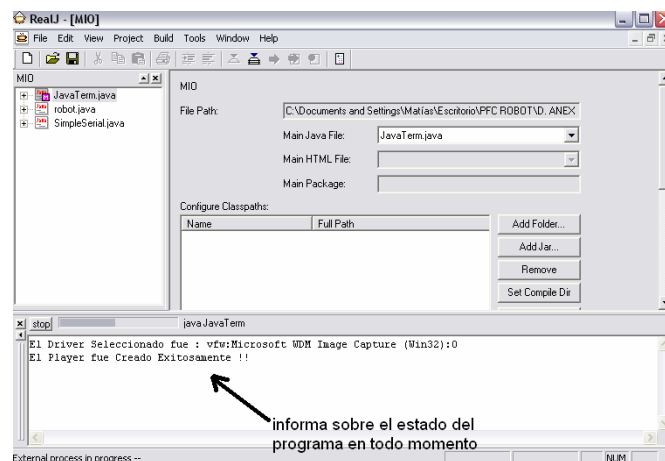
**CAMPO DE BOTONES:** envía la orden de movimiento al robot. En la siguiente figura indicamos las órdenes más detalladamente:



**Figura 3.8.** Campo de botones

**IMAGEN CÁMARA ROBOT:** donde aparece la imagen capturada por la cámara del robot.

**VENTANA DEL PROGRAMA REALJ:** en esta ventana aparecerán todo tipo de mensajes informativos, sobre cualquier incidencia o incluso si la orden que se ha enviado al robot se ha ejecutado correctamente. La ventana aparece en la imagen de más abajo.

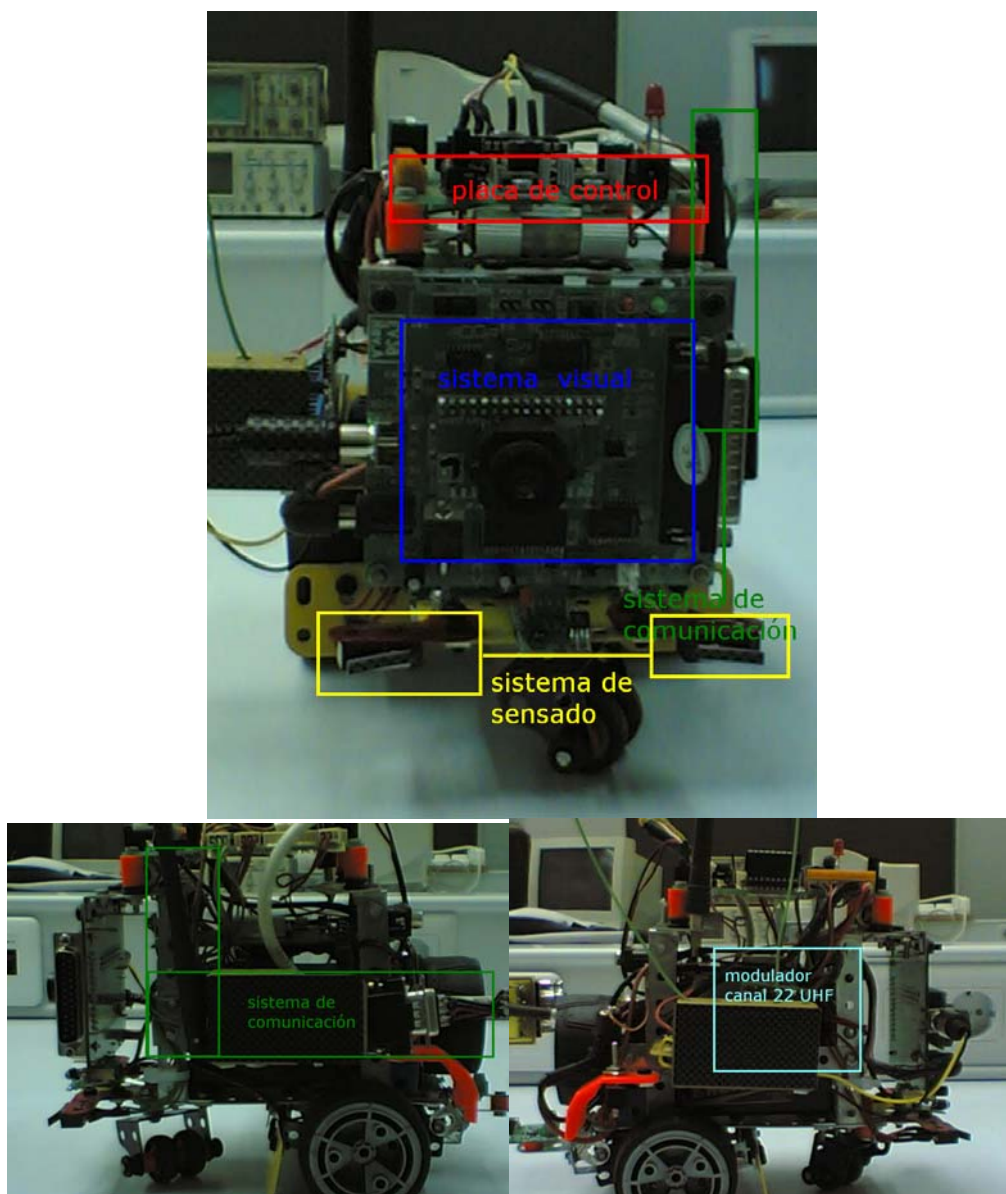


**Figura 3.9.** Ventana informativa

Además de los componentes que acabamos de explicar, la estación base también estará compuesta por un conjunto teclado-ratón comercial.

## MICRO-ROBOT MÓVIL

En este capítulo describiremos el vehículo que hemos desarrollado, el cual está formado por varios bloques, como puede ser el sistema microcontrolador, que se encarga de generar las señales de control de los actuadores, el sistema de sensado, encargado de proporcionar información acerca del entorno, el sistema visual, encargado de enviar imágenes a la estación base, tendremos los motores o actuadores y no podemos olvidar el desarrollo de la estructura móvil. Comentaremos todos los aspectos implicados, como el montaje e integración de placas de control y baterías. De cara a especificar los requisitos de la placa de control, describiremos brevemente las características de los motores, la micro-cámara, y el sistema de sensores incluido.



**Figura. 4.1.** Esquema general del vehículo móvil

Características físicas:

Dimensiones	Largo	15 cm.
	Ancho	10 cm.
	Alto	12 cm. (sin antena)
Peso		1.5 Kg.

**Tabla 4.1.** Características físicas del Micro-robot

Características electro-mecánicas:

Corriente máxima	400 mA
Corriente media	250 mA
Velocidad máxima	20 cm. / s
Carga máxima	1 Kg.
Autonomía	45 minutos
Alcance control remoto máximo sin obstáculos	100 metros
Alcance control remoto máximo con obstáculos	40 metros
Alcance señal enviada modulador RF canal 22 UHF	15 metros

**Tabla 4.2.** Características electro-mecánicas del Micro-robot

## 1 Descripción de los componentes empleados

### 1.1 Placa controlador MSx84

#### 1.1.1 Introducción

La tarjeta de control MSx84 de Microsystems Engineering es una tarjeta autónoma de propósito general y bajo coste que, controlada por un PIC 16x84, es capaz de controlar el estado de cinco sensores de entrada y gobernar dos motores DC o uno paso a paso (PAP) de salida, en función del software de aplicación grabado en el microcontrolador. De entre sus múltiples aplicaciones de carácter didáctico cabe citar las siguientes:

- Adaptación y control de diferentes tipos de sensores de entrada que pueden emplearse en aplicaciones de tipo industrial como son los mecánicos, ópticos, infrarrojos, magnéticos, ultrasónicos, etc. Se puede así experimentar con diferentes técnicas para la detección de flancos de señales de entrada, eliminación del efecto “rebote”, medida de anchura de pulsos, cuenta de pulsos de entrada, medida de la frecuencia de los pulsos de entrada, cadencia entre pulsos de entrada, etc.
- Control de motores de corriente continua (DC) mediante accionamiento digital (todo o nada), técnicas de PWM para la regulación de velocidad, control del sentido de giro, realimentación, posicionamiento, etc.



- Control de motores paso a paso (PAP), regulación de velocidad, sentido de giro, posicionamiento, etc.
- Microbótica. Control de una estructura móvil accionada mediante dos motores DC y varios sensores de entrada. Control de trayectorias, posicionamiento, trazado de rutas predefinidas, detección de obstáculos, seguimiento de objetos, bordear objetos, determinar perímetros o áreas, etc.

### ***1.1.2 Características generales***

De entre las características más notables de la placa MSx84 de Microsystems Engineering cabría citar las siguientes:

- Tarjeta autónoma de bajo coste y dimensiones reducidas (80 x 100 mm.).
- Alimentación de 15VAC mediante transformador de red o bien mediante baterías de 6, 9 o 12VDC, pudiendo en este último caso utilizarse una batería de 12VDC de plomo recargable.
- Circuitos de rectificación, filtrado y estabilización de alimentación incluidos en la tarjeta.
- Incluye circuito de carga para baterías de 12VDC de plomo ácido.
- Zócalo para microcontrolador PIC 16x84, circuito oscilador y pulsador de RESET incluido en la tarjeta.
- Conectores y circuitos para adaptación de hasta 5 sensores o dispositivos de entrada.
- Driver para el accionamiento de dos motores DC o uno paso a paso (PAP).
- Interruptor ON/OFF y bornas para la conexión tanto de los motores como para la batería externa.
- Mediante un jumper se puede seleccionar la tensión aplicada a los motores. Esta puede ser de +5VDC o bien de +12VDC o la tensión de la batería externa.
- Conjunto de 9 DIP-SWITCH que permiten habilitar o no cada uno de los sensores de entrada así como las salidas a los motores. De esta forma se puede dejar libres aquellas líneas del PIC asociadas a los sensores y motores que no se vayan a utilizar.
- Conector de expansión PIC-BUS compatible con la tarjeta  $\mu$ PIC Trainer de Microsystems Engineering. A través de él se obtienen todas las señales del PIC, pudiéndose de esta forma modificar o añadir nuevos periféricos.

### ***1.1.3 Listado de materiales***

Se muestra a continuación la lista de los materiales que componen la tarjeta MSx84<sup>1</sup>.

Referencia	Valor	Descripción
U1-U2	40106 2	Circuitos Trigger adaptadores de nivel.
U3	L293B	Driver para los motores
U4	UA7812	Estabilizador de tensión a 12VDC
U5	UA7805	Estabilizador de tensión a 5VDC
U6	PIC16x84	Microcontrolador PIC
D1	1 A	Puente rectificador de 1 A
D2-D5	1N4004	4 Diodos de silicio
D6	3mm	Diodo led rojo
R1-R5	47K	5 Resistencias de ¼ W
R6-R10	220	5 Resistencias de ¼ W
R11	10K	1 Resistencia de ¼ W
R12	220	1 Resistencia de ¼ W
C1-C2	100µF/25V	2 Condensadores electrolíticos
C3-C6 y C9-C10	100nF	6 Condensadores cerámicos de desacoplo
C7-C8	27pF	2 Condensadores cerámicos
Y1	4MHz	Cristal de cuarzo
JP1	Varios	Jumper de dos posiciones y caperuza.
J0-J4	Varios	5 Conectores MOLEX macho, codo de 4 vías y paso 2,54
J5-J7	Varios	3 Bornas de dos contactos paso 5mm
J8	Varios	Base de alimentación
J9	Varios	Conector macho, recto de 2x13 vías paso 2,54 Mm.
	Varios	6 Tuercas M3
	Varios	6 Tornillos M3 x 7 Mm.
	Varios	Cable PIC-BUS de 26 vías
	Varios	Porta pilas de 9V
	Varios	Porta pilas pack de 4 pilas 1.5
	Varios	Placa de C.Impreso de 80 x 100 Mm.
SW1	Varios	DIP-SWITCH de 9 contactos
SW2	Varios	Pulsador
SW3	Varios	Interruptor DPDT
	Varios	2 refrigeradores para los estabilizadores de tensión
	Varios	3 Zócalo de 18 pines
	Varios	1 Zócalos de 16 pines
	Varios	2 Zócalos de 14 pines
	Varios	4 Separadores 7 Mm. M3

1 Además de todos estos materiales, como mejora le hemos añadido un diodo led que nos indica cuando se está cargando la batería del robot.

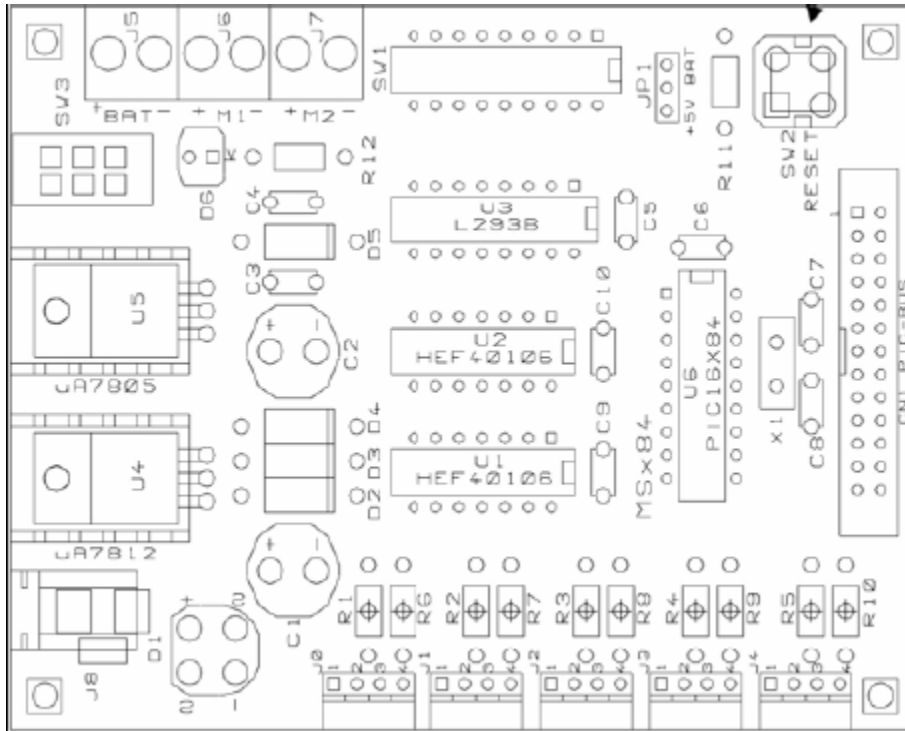


Figura 4.2 Tarjeta MSx84

#### 1.1.4 Descripción Funcional

En posteriores apartados se va a ir describiendo el hardware de la placa de control MSx84. De esta forma el usuario de la misma podrá entender sus aplicaciones así como las posibles modificaciones que se pueden realizar con objeto de controlar diferentes tipos de sensores y/o motores.

##### A LA FUENTE DE ALIMENTACIÓN

Esta integrada en la propia placa MSx84 y ofrece diferentes posibilidades de trabajo. Su esquema se ofrece en la figura 4.3.

Toda la electrónica de control, incluido el propio microcontrolador PIC 16x84, trabajan con una VCC de +5 V. Esta tensión se puede obtener a partir de una tensión alterna de 12VAC o bien de un conjunto de pilas o baterías.

Mediante el conector J8 se introduce una tensión alterna de 12VAC procedente de un transformador (no incluido). El puente rectificador D1 junto con C1 se encarga de rectificarla y filtrarla para obtener una tensión continua de unos 21VDC. Esta se estabiliza a unos 13 VDC gracias al regulador U4 (7812) y los diodos D2-D5. El transformador puede ser el mismo que se emplea en la  $\mu$ PIC Trainer. El interruptor SW3 de ON/OFF aplica esa tensión al controlador de motores (+VBAT) y por otra parte al regulador U5 (7805). Por la patilla 3 de U5 se obtiene la tensión +VCC de 5 VDC, que se aplica al resto del circuito. El led D6 pilota la presencia de dicha tensión.

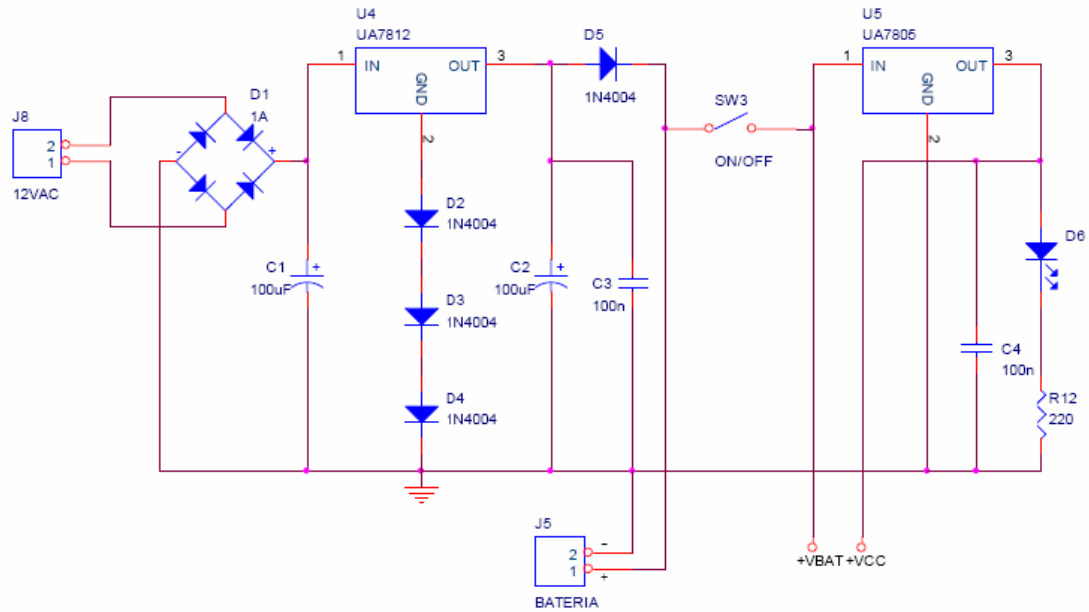


Figura 4.3 La fuente de alimentación

Otra posibilidad es aplicar una tensión continua procedente de una batería a través del conector J5, prestando atención a su polaridad. Al accionarse el interruptor ON/OFF SW3 dicha tensión es la que se aplica al controlador de motores (+VBAT) y la que permite obtener los +5VDC de VCC gracias al regulador U5.

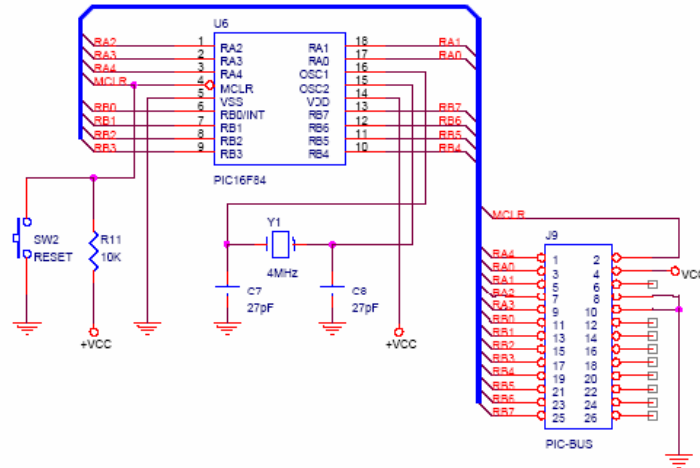
En el siguiente cuadro se reflejan las diferentes opciones de alimentación de la tarjeta de control MSx84.

OPCIONES	COMENTARIOS
Mediante transformador de 15VAC conectado en J8	No se conecta la batería. +VBAT=13VDC Y +VCC = 5 VDC
Mediante transformador de 15VAC conectado en J8	Batería recargable de plomo de 12V. +VBAT =13VDC y +VCC = 5VDC. El circuito formado por el regulador U4 y D5 permite recargar dicha batería
Batería recargable de plomo de 12V conectada en J5	El transformador se puede conectar en J8 para su recarga. +VBAT = 12 VDC y +VCC = 5VDC
Batería no recargable de 12 V conectada en J5	No conectar el transformador. +VBAT = 12 VDC y +VCC = 5VDC.
Batería no recargable de 9V conectada en J5	No conectar el transformador. +VBAT = 9 VDC y +VCC = 5VDC
Conjunto de 4 baterías de 1.5V en serie (6VDC) conectado en J5	No conectar el transformador. +VBAT = 6 VDC y +VCC = 5VDC

Tabla 4.3 Alimentación MSx84

## B EL MICROCONTROLADOR

En la figura 4.4 se muestran las conexiones del microcontrolador PIC 16x84. Es el “corazón” de la tarjeta MSx84. La función principal del sistema microcontrolador situado en el micro-robot móvil es la de interpretar los datos procedentes del sistema de control remoto y de los sensores, generando las oportunas señales de control de los motores del vehículo.



**Figura 4.4** El microcontrolador PIC 16x84

El cristal Y1 junto con los condensadores C7 y C8 genera la frecuencia de trabajo de 4MHz del microcontrolador PIC. Variando el valor de estos componentes se puede alterar dicha velocidad según las necesidades del usuario. Consultar las especificaciones técnicas de Microchip para seleccionar las diferentes frecuencias.

El circuito de RESET está formado por el pulsador SW2 y la resistencia R11. Esto permite la inicialización manual del sistema por parte del usuario.

El conector PIC-BUS es compatible pin a pin con el del sistema  $\mu$ PIC Trainer de Microsystems Engineering. A través de él, todas las señales del PIC están a disposición del usuario. Esto permitirá conectar y/o añadir otro tipo de periféricos distintos a los que soporta la tarjeta MSx84.

Igual de importante es el hecho de que gracias al PIC-BUS y el cable de conexión incluido, es posible grabar el PIC 16x84 ubicado en la tarjeta MSx84 mediante el sistema  $\mu$ PIC Trainer, así como aprovechar y hacer uso tanto de los periféricos de este último como de todo el software de grabación. Todo ello sin necesidad de extraer el PIC 16x84 de la tarjeta MSx84.

### 1.1.5 Adaptación de periféricos a la MSx84

#### Introducción

La tarjeta MSx84 está especialmente pensada para la conexión directa de sensores de tipo óptico infrarrojo, aunque es lo suficientemente flexible para que, con pequeñas modificaciones en los valores de ciertos componentes, se pueda adaptar fácilmente a otro tipo de sensores.

El esquema de la figura 4 muestra el circuito de los 5 conectores J0-J4 de la tarjeta MSx84 que permite conectar de la forma ya explicada hasta 5 sensores diferentes.

Las resistencias R1-R5 nominalmente son de 47K. Como ya se ha comentado, este valor se puede alterar en función del tipo de sensores que se vaya a emplear. Igualmente la placa está preparada para soportar resistencias ajustables o potenciómetros que permitan variar la sensibilidad de esos sensores.

Las resistencias R6-R10 nominalmente son de 220. Se tratan de las resistencias de polarización del led emisor. Su valor también se puede variar en función del tipo de sensor.

Las 5 señales de entrada van a parar a sendas puertas trigger inversoras para la adaptación de niveles, evitando en lo posible ruidos y deformación de las señales.

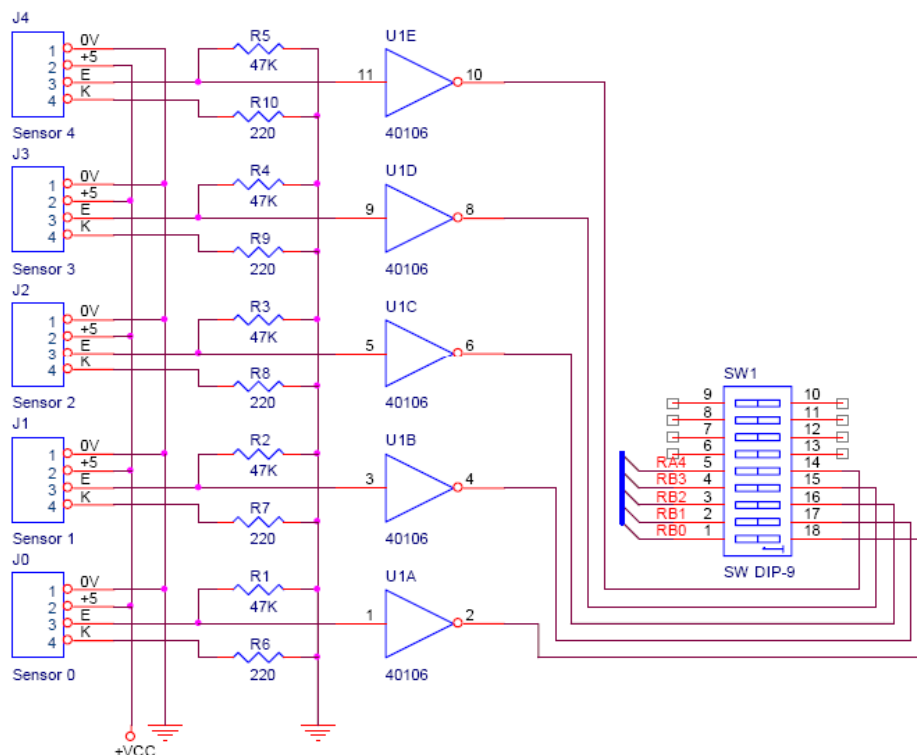


Figura 4.5 Las entradas de la tarjeta MSx84

Las salidas de los 5 inversores (U1A-U1E) van a parar a los micro-switches 1-5 del DIPSWITCH (SW1). Esto permite que cada una de las cinco señales de entrada pueda desconectarse de las líneas del microcontrolador PIC 16x84. De esta forma, si en una determinada aplicación ciertos sensores de entrada no se emplean, las líneas del PIC pueden quedar sin conexión y aprovecharlas así con otro tipo de periféricos.

A la vista del esquema anterior se puede apreciar la asociación que hay entre los conectores de los sensores de entrada, los micro-switches y las líneas del PIC. Esta asociación queda resumida en la siguiente tabla.

Hay que destacar que el sensor conectado a través de J4 queda asociado a la línea RA4 del PIC. Esta línea es también la señal T0CKI que accede al Timer 0 (TMR0). Este, actuando en modo contador de eventos externos, puede contar los pulsos procedentes de ese sensor. Se puede así desarrollar aplicaciones consistentes en contar pulsos, medir frecuencias, medir anchura de pulsos, intervalos entre pulsos, etc.

Igualmente, el sensor conectado a través de J0, está asociado a la línea RB0 del PIC. Esta línea es también la señal INT de entrada de interrupción externa. Puede ser programada para ser activa al flanco ascendente o descendente. Se puede de esta manera dar una prioridad a los eventos ocurridos en los sensores de entrada para realizar el tratamiento oportuno.

CONECTOR	SW <sub>n</sub>	CONEXIÓN
J0	1 = ON	RB0 / INT
J0	1 = OFF	Libre
J1	2 = ON	RB1
J1	2 = OFF	Libre
J2	3 = ON	RB2
J2	3 = OFF	Libre
J3	4 = ON	RB3
J3	4 = OFF	Libre
J4	5 = ON	RA4 / T0CKI
J4	5 = OFF	Libre

**Tabla 4.4** Asociación conectores MSx84- $\mu$ Controlador

Se debe insistir que la posibilidad de desconectar las señales no empleadas y, teniendo en cuenta que todas las señales del PIC 16x84 están disponibles mediante el conector PIC-BUS, nos ofrece los recursos necesarios para, implementando un hardware adicional, poder gobernar todo tipo de periféricos no contemplados en la tarjeta MSx84.

### ***Los Motores***

Una de las orientaciones que se quiere dar a la tarjeta MSx84 es la del accionamiento y control de hasta 2 motores de DC o bien uno paso a paso.

### ***El Driver L293B***

La tarjeta MSx84 dispone de un driver del tipo push-pull de 4 canales integrados sobre el chip L293B de SGS-THOMSON y especialmente indicado para activar cargas de elevado consumo (p.e. motores de DC). Encargado de amplificar la corriente de alimentación de los motores, y de acondicionar las señales de control de los mismos, que provienen del microcontrolador. Básicamente, se trata de una etapa de potencia. Las características más relevantes son:

- Corriente de salida de 1 A por canal.
- Corriente de salida de pico de 2 A por canal
- Señal para la habilitación de las salidas
- Alta inmunidad al ruido
- Alimentación para las cargas separada de la alimentación de control.
- Protección contra sobre – temperaturas.

*Descripción*

El L293B es un driver de 4 canales capaz de proporcionar una corriente de salida de hasta 1A por canal. Cada canal es controlado por señales de entrada compatibles TTL y cada pareja de canales dispone de una señal de habilitación que desconecta las salidas de los mismos.

Dispone de una patilla para la alimentación de las cargas que se están controlando, de forma que dicha alimentación es independiente de la lógica de control. La figura 4.6 muestra el encapsulado de 16 pines, la distribución de patillas y la descripción de las mismas.

Pin N°	Nombre	Descripción
1	EN1	Habilitación de los canales 1 y 2
2	IN1	Entrada del canal 1
3	OUT	Salida del canal 1
4	GND	Tierra de alimentación
5	GND	Tierra de alimentación
6	OUT2	Salida del canal 2
7	IN2	Entrada del canal 2
8	VDD	Alimentación de las cargas
9	EN2	Habilitación de los canales 3 y 4
10	IN3	Entrada del canal 3
11	OUT3	Salida del canal 3
12	GND	Tierra de alimentación
13	GND	Tierra de alimentación
14	OUT4	Salida del canal 4
15	IN4	Entrada del canal 4
16	VCC	Alimentación de +5VDC

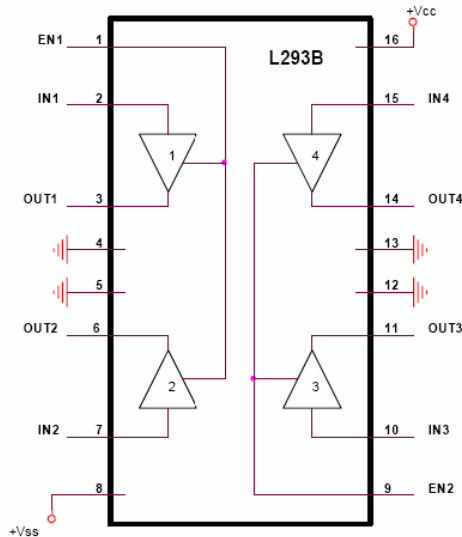
  

**Figura 4.6** Descripción de pines del L293B



*Diagrama por bloques*

Es el mostrado en la figura 4.7. La señal de control EN1 activa la pareja de canales formada por los drivers 1 y 2. La señal EN2 activa la pareja de drivers 3 y 4. Las salidas OUTn se asocian con las correspondientes INn. Las señales de salida son amplificadas respecto a las de entrada tanto en tensión (hasta +Vss) como en corriente (máx. 1 A).



**Figura 4.7** Esquema por bloques del L293B

En la siguiente tabla se muestra la tabla de la verdad que es igual para cada canal del driver L293B.

VINn	VOUTn	VENn
H	H	H
L	L	H
H	Z	L
L	Z	L

H = nivel "1" L = nivel "0" Z = off

**Tabla 4.5.** Tabla de verdad L293B

*Parámetros eléctricos*

**RANGOS ABSOLUTOS MÁXIMOS**

Símbolo	Parámetro	Valor	Unidad
Vss	Tensión de alimentación para las cargas	36	V
Vcc	Tensión de alimentación de la lógica	36	V
Vi	Tensión de entrada	7	V
Vinh	Tensión de habilitación	7	V
Iout	Intensidad de pico de salida	2	A
Ptot	Disipación total de potencia	5	W

**Tabla 4.6.** Parámetros eléctricos driver L293B

## CARACTERÍSTICAS ELÉCTRICAS

Para cada canal:

$V_{ss}=24V$

$V_{cc}=5V$

$T^a$  ambiente  $25^{\circ}C$

Símbolo	Parámetro	Mín.	Típ.	Máx.	Unidad
$V_{ss}$	Alimentación de las cargas	$V_{ss}$	--	36	V
$V_{cc}$	Alimentación de la lógica	4.5	--	36	V
$V_{IL}$	Tensión de entrada a nivel "0"	-0.3	--	1.5	V
$V_{IH}$	Tensión de entrada a nivel "1"	2.3	--	$V_{cc}$	V
$I_{IL}$	Corriente de entrada a nivel "0"	--	--	-10	$\mu A$
$I_{IH}$	Corriente de entrada a nivel "1"	--	30	100	$\mu A$
$V_{inhL}$	Tensión de habilitación a "0"	-0.3	--	1.5	V
$V_{inhH}$	Tensión de habilitación a "1"	2.3	--	$V_{cc}$	V
$I_{inhL}$	Corriente de habilitación a "0"	--	-30	-100	$\mu A$
$I_{inhH}$	Corriente de habilitación a "1"	--	--	+/-10	$\mu A$

Tabla 4.7 Características eléctricas L293B

### Motores DC

El esquema de la figura 4.8 muestra el principio de funcionamiento de un circuito de control para motores DC conocido como 'puente en H' (Aunque como veremos mas adelante, los motores utilizados no son motores DC, hacemos una descripción de los mismos, ya que los servomotores utilizados han sido trucados para que su funcionamiento sea similar al de estos motores).

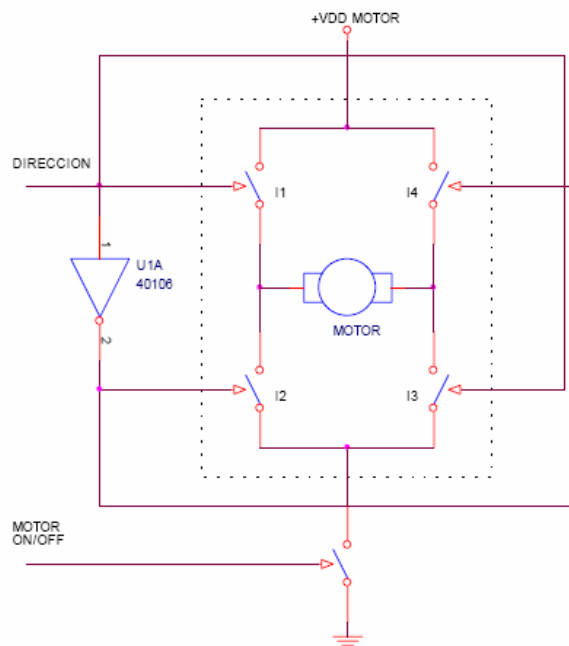


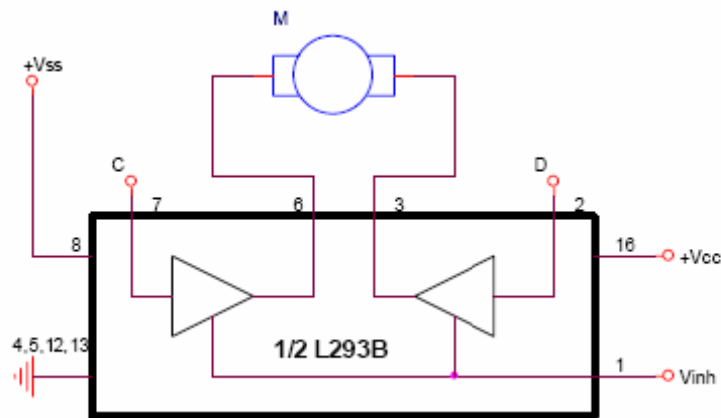
Figura 4.8 El puente en H

Los 5 interruptores emulan a sendos transistores de conmutación o drivers. El de ON/OFF se activa, cerrando circuito con tierra, mediante una señal que se puede considerar como señal de habilitación.

Estando desconectada, el motor queda sin tensión. El motor puede girar libremente. El accionamiento del motor se realiza activando los interruptores por parejas. Así, al activar I1 e I3, al motor le llega una tensión cuya polaridad estará a la derecha y Vdd a la izquierda. Esto producirá el giro en un sentido.

Activando la pareja de interruptores I2 e I4, la tensión aplicada adquiere la polaridad opuesta por lo que el giro se realizará en sentido contrario. Naturalmente nunca deberán estar ambas parejas activadas pues se produciría un cortocircuito en la alimentación.

Empleando dos de los cuatro drivers que posee el circuito L293B se puede realizar un sistema que accione el motor y lo haga girar en ambos sentidos. Puede servir como ejemplo el mostrado en la figura 4.9.



**Figura 4.9** Accionamiento de un motor de DC

La señal introducida por la patilla 1 (Vinh) equivale a la señal ON/OFF anterior. Si está a nivel "0", ambos drivers permanecen inhabilitados y el motor sin tensión. Las señales C y D controlan a las parejas de interruptores anteriormente citadas en la figura 4.8 deben trabajar en oposición.

Activando C y desactivando D, el motor recibe una tensión cuya polaridad queda con el + a la izquierda y el - a la derecha, provocando el giro en un sentido. Invertiendo el estado de las señales C y D se conseguirá una polarización opuesta y un giro en sentido contrario.

La figura 4.10 muestra el empleo del circuito L293B en la tarjeta MSx84. Gracias a él se pueden gobernar dos motores de corriente continua (DC)

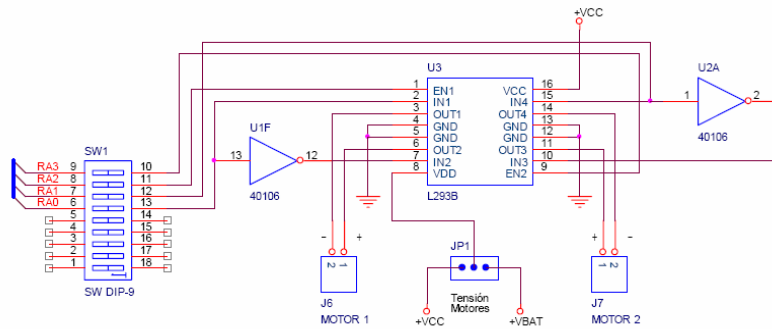


Figura 4.10 Conexión del L293 en la tarjeta MSx84

La señal RA0 del PIC va a parar, a través del switch 6, a la señal IN1 y, mediante el inversor U1F, a la señal IN2 del L293B. Esta señal gobierna las salidas OUT1 y OUT2 que controlan el motor M1. Según su nivel lógico, las salidas OUT1 y OUT2 aplicarán una determinada polaridad al motor, haciéndolo girar en un sentido u otro.

La misma explicación sirve respecto a la señal RA1 del PIC. En este caso van a parar a IN3 e IN4 que gobiernan OUT3 y OUT4 y estas a su vez al motor M2.

Las señales RA2 y RA3 del PIC quedan conectadas con las entradas de habilitación EN1 y EN2 del circuito L293B. Con ellas se consigue habilitar o no a cualquiera de los dos motores M1 y M2.

Mediante el Jumper JP1 se selecciona la tensión con la que funcionarán los motores. Esta puede ser +VCC de +5V que se obtienen del circuito de estabilización general del sistema, o bien +VBAT, que es de +13V si se obtienen desde la fuente de alimentación o dependiente de la tensión externa aplicada mediante baterías.

La siguiente tabla de la verdad sintetiza las diferentes combinaciones que, un software de aplicación, puede sacar por las líneas RA0-RA3 del PIC.

RA3	RA2	RA1	RA0	Descripción
0	0	X	X	Ambos motores desconectados
0	1	X	0	M1 en ON, giro horario. M2 en OFF
0	1	X	1	M1 en ON, giro anti horario. M2 en OFF
1	0	0	X	M1 en OFF. M2 en ON, giro horario
1	0	1	X	M1 en OFF. M2 en ON, giro anti horario
1	1	0	0	M1 en ON, giro horario. M2 en ON, giro horario
1	1	0	1	M1 en ON, giro anti horario. M2 en ON, giro horario
1	1	1	0	M1 en ON, giro horario. M2 en OFF, giro antihorario
1	1	1	1	M1 en ON, giro anti-horario. M2 en ON, giro-antihorario

Tabla 4.8 Combinaciones Jumper MSx84-L293B

Finalmente, los switches 6, 7, 8 y 9 de SW1 permiten, al igual que en el caso de los sensores de entrada ya estudiados, desconectar las señales RA0-RA3 del driver de motores L293B. De esta forma, aquellas aplicaciones que no requieran el empleo de dicho driver, puede utilizar esas señales para controlar otro tipo de periféricos no contemplados en la tarjeta MSx84. Se recuerda que todas las señales del PIC están disponibles en el conector PICBUS.

#### *Empleo desde $\mu$ PIC Trainer*

El conector PIC-BUS de la tarjeta MSx84 es totalmente compatible pin a pin con el conector PIC-BUS de la  $\mu$ PIC Trainer. Esto permite que ambas tarjetas puedan compartir entre sí sus respectivos recursos, conectándolas mediante el cable de conexión incluido en el KIT. Así, los periféricos sitos en la  $\mu$ PIC Trainer pueden ser usados por la tarjeta MSx84 y viceversa. Entre ambas tarjetas se aumenta considerablemente el número de aplicaciones posibles.

Especialmente importante es el destacar la posibilidad de grabar mediante la  $\mu$ PIC-Trainer el PIC 16F84 que en ese momento esté insertado en la tarjeta MSx84, utilizando el mismo software PICME-TR con el que el usuario de  $\mu$ PIC-Trainer está ya familiarizado.

Para proceder a grabar el PIC ubicado en la tarjeta MSx84 desde la  $\mu$ PIC-Trainer, proceder realizando los siguientes pasos:

- Quitar, si lo hubiera, el PIC de la  $\mu$ PIC-Trainer. Insertar el que se desea grabar en la tarjeta MSx84
- Conectar el cable plano entre el PIC-BUS de la  $\mu$ PIC-Trainer y el PICBUS de la tarjeta MSx84.
- Abrir los jumpers J5, J6 y J7 de la  $\mu$ PIC-Trainer. Igualmente es aconsejable desconectar los distintos periféricos que hubiera conectados en ambas tarjetas. De esta forma se disminuye el consumo eléctrico, especialmente si se anda escaso de alimentación.
- Conectar el transformador de alimentación de 12-15VAC a la  $\mu$ PIC-Trainer. El PIC-BUS alimenta así a la tarjeta MSx84.
- Conectar el cable del canal paralelo del PC a la  $\mu$ PIC-Trainer. Ejecutar el programa PICME-TR.
- Hacer uso normal de los distintos botones y opciones de este programa, según la función a realizar: Leer, Abrir fichero, Programar, Verificar, etc.
- Una vez grabado el PIC con el programa de la aplicación, retirar el cable del PIC-BUS. La tarjeta MSx84 es autónoma. Conectar en ella los periféricos a gobernar así como la tensión de alimentación.

## 1.2 Módulos Bluetooth

En un principio la comunicación se iba a realizar mediante dos transceptores RF de la casa AUREL modelo RTLSAW, mucho más baratos que los módulos bluetooth empleados y además debían de proporcionar la misma funcionalidad (en teoría) pero estos tenían las pruebas realizadas demostraron que los primeros presentan peor inmunidad frente al ruido y menor alcance.

### 1.2.1 MÓDULO RF AUREL TRANSCEPTOR RTF-SAW

Módulo emisor/receptor (transceptor). Permite la comunicación half-duplex vía radio entre dos estaciones con una velocidad de 2400 baudios. Tensión de alimentación +5Vdc, consume de 4.5mA el transmisor y de 2.5 mA el receptor, frecuencia 433.92 MHz. Conforme a la norma EN 300-220 con disponibilidad del correspondiente certificado.

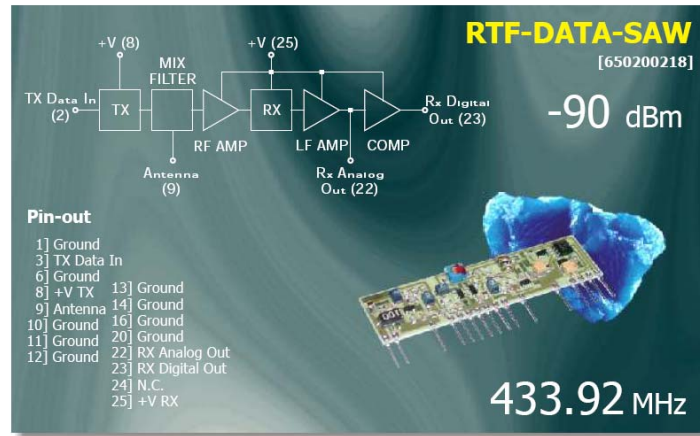
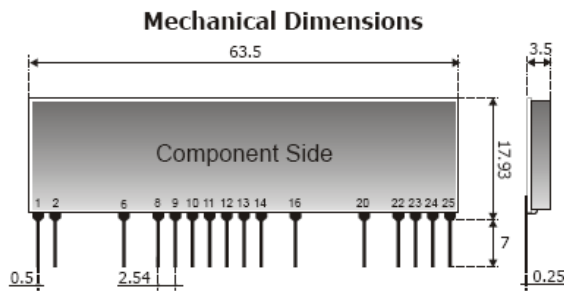


Figura 4.11 Esquema general conexión módulos RF aurel



### Technical Specification

Ta = 25 °C

CHARACTERISTICS	MIN	TYP	MAX	UNIT
V <sub>s</sub> Supply Voltage • Alimentazione		5		Vdc
I <sub>s</sub> Supply Current • Corrente Assorbita [TX ON]			4.5	mA
I <sub>s</sub> Supply Current • Corrente Assorbita [RX ON]			2.5	mA
I <sub>s</sub> Supply Current • Corrente Assorbita [RX/TX OFF]			0	mA
F <sub>w</sub> Reception frequency • Frequenza di ricezione		433.92		MHz
S <sub>1</sub> RF sensitivity • Sensibilità RF		-90		dBm
B <sub>w</sub> LF Bandwidth • Banda passante BF		5		KHz
P <sub>o</sub> TX Power • Potenza TX		9		dBm±2dB
Z <sub>1</sub> Antenna impedance • Impedenza d'antenna		50		Ω
T <sub>on</sub> Switch-on time • Tempo di accensione			100	ms
T <sub>op</sub> Operating temperature range • Temperatura di lavoro	-20		+80	°C

Figura 4.12 Especificaciones mecánicas y técnicas módulos RF aurel

La imagen que presentamos a continuación, corresponde a cada una de las placas con las cuales realizábamos las pruebas del módulo.

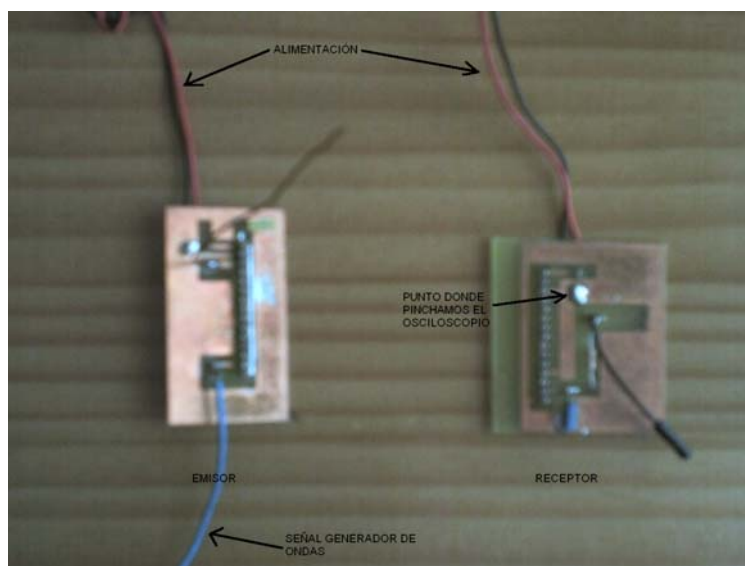


Figura 4.13 Placas prueba módulos RF aurel

## 1.2.2 BLUETOOTH RS232 INTERFACE PROMI SD202

### INTRODUCCIÓN

Promi-SD fue creado para hacer comunicaciones seriales inalámbricas, de bajo coste, fácil de realizar y de un alcance extendido. Se trata de una conexión inalámbrica sin cables estándares RS-232.

Con Promi-SD, puedes efectuar conexiones inalámbricas Bluetooth hacia cualquier dispositivo Bluetooth que soporte el SPP (Perfil Puerto Serie). Pues, cuando utilizas el Promi-SD como cable de sustitución, aprovecha la conexión automática del Promi-SD. Cuando dos Promi-SD están instalados a éste propósito, se conectan automáticamente cuando se encienden. Colocados en su rayo cobertura, sirven de cable virtual RS-232.

### CARACTERÍSTICAS

- Serial Interface:  
RS232, Female DSUB-9, 1200~230400 baud  
Automatic Detection Feature of Hardware flow control  
DTR/DSR for loop-back & full transfer
- Bluetooth Interface:  
Bluetooth Specification: V 1.1  
Radio: Class 1  
Output Power: 63mW  
Level: 16~18 dBm  
Range: ~100m  
Bluetooth protocols: RFCOMM, L2CAP, SDP

- Supported Profiles: Serial Port Profile
- Recommended operating conditions:  
Temperature: -20'C~70'C  
Humidity: 90% Non-condensing
  - Dimensions:  
60 x 26 x 16 (mm)
  - RF Range:  
Up to 100m
  - Power Connector:  
DC plug or 9 pin
  - Power supply:  
5V~12V
  - Dip switch:  
No
  - Battery:  
No
  - Applicable Antenna:  
Stub Antenna (included)  
Dipole Antenna (optional)  
Patch Antenna (optional)
  - Type Approved:  
TELEC  
MIC  
CE  
FCC
  - Includes:  
Bluetooth Serial Adaptor (Class 1)  
Stub Antenna  
DC Power Cable with bare wire leads

**Tabla 4.9** Características módulo bluetooth

De todos los dispositivos disponibles de esta misma marca, el módulo PROMI SDXXX es el más adecuado para el proyecto ya que proporciona mayor inmunidad al ruido y a las interferencias y un mayor alcance y fiabilidad. Entre los modelos disponibles se elige el SD202 cuya única diferencia con el SD101 es que tiene la posibilidad de ser alimentado por corriente continua, en un rango entre 5-12 V.

**SD202**



**Figura 4.14** Módulo bluetooth PROMI-SD 202



### 1.3 Sistema de Sensado

Un sistema de sensado es un sistema de percepción, y en general, el sistema de percepción más implementado en los robots móviles es aquel que se encarga de la detección de obstáculos. Generalmente, se suelen utilizar dispositivos de telemetría (medición de distancias) ópticos, ultrasónicos, o electromecánicos.

Como sistema de sensado se ha implementado un sistema de sensores electromecánicos encargado de proporcionarle “visión” al micro-robot. La utilidad que le vamos a dar a este sistema de sensado es hacer que el micro-robot móvil, de forma autónoma, cambie su dirección cuando detecte que ha chocado y a su vez nos informe de dicho choque, enviando un mensaje al PC.

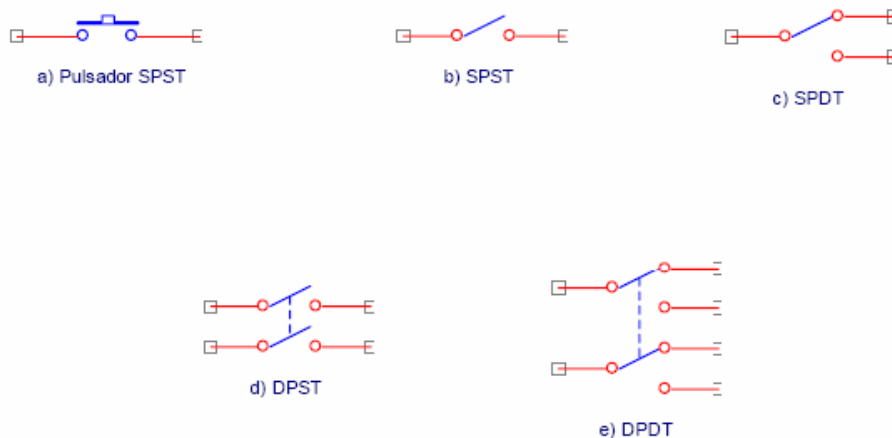
#### 1.3.1 Bumpers

Se trata de dispositivos que, mediante un determinado mecanismo, son capaces de abrir o cerrar un circuito eléctrico generando así una señal para su posterior tratamiento.

Aunque existen múltiples modelos adaptados a otras tantas aplicaciones, cabría citar por su sencillez y bajo coste los pulsadores, interruptores, finales de carrera, de mercurio, magnéticos, bumpers, etc.

Su funcionamiento se basa en una o varias láminas metálicas que mediante algún dispositivo de accionamiento cierran o abren un circuito eléctrico dejando pasar o no la tensión y obteniéndose así un determinado nivel lógico.

Según el tipo de mecanismo así como el número de circuitos que abre o cierra, nos podemos encontrar con diferentes símbolos, algunos de los cuales están representados en la siguiente figura.



**Figura 4.15** Símbolos más usuales de los bumpers

Un mecanismo muy conocido y empleado como final de carrera, detector de obstáculos, etc., es el conocido ‘Bumper’ cuyo aspecto se muestra en la figura 4.16

### 1.3.2 Sensores o Bumpers electromecánicos

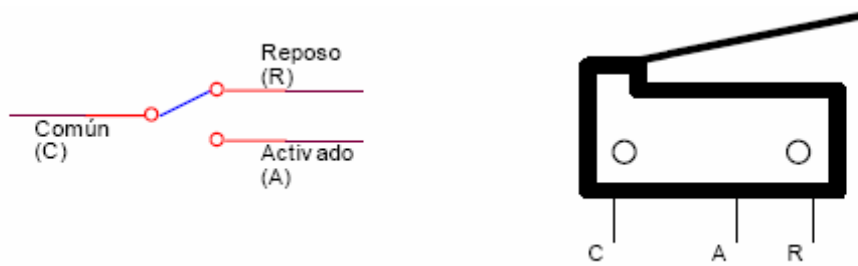


Figura 4.16 Los Bumpers

Se trata de un mecanismo tipo pulsador de simple circuito y doble posición (SPDT). Su accionamiento se provoca presionando sobre la lengüeta o varilla metálica.

Los conectores de entrada J0-J4 de la tarjeta MSx84 permiten la conexión de este tipo de mecanismos o de cualquier otro similar. El circuito típico se muestra en la figura 4.11. Un mecanismo tipo SPDT en reposo conecta con tierra la entrada del inversor. Si fuera un mecanismo tipo SPST en reposo habría un circuito abierto pero la resistencia de 47K sita en la tarjeta MSx84 se encargará de mantener la entrada del inversor a tierra. En ambos casos la salida del inversor estará a nivel "1" cuando los mecanismos estén en reposo.

Al ser accionados, la entrada del inversor se conecta con +VCC por lo que a la entrada del PIC el nivel lógico será "0".

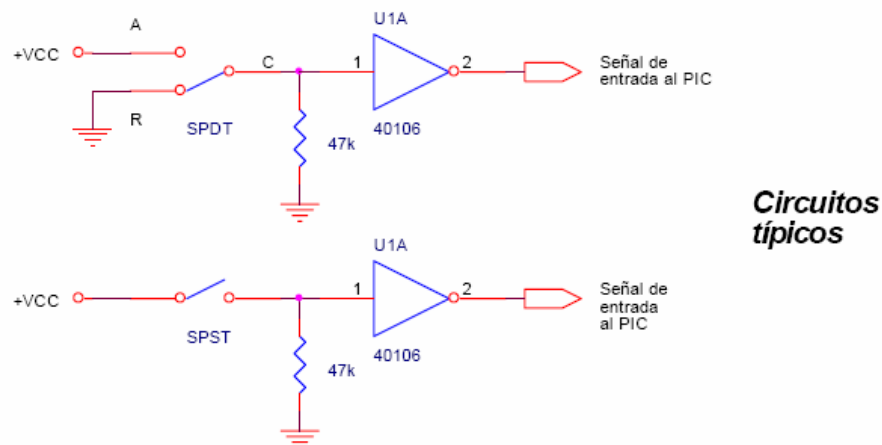


Figura 4.17 Conexión de mecanismos con MSx84

Teniendo en cuenta la distribución de las señales que transportan los conectores J0-J4 de la tarjeta MSx84, la conexión práctica de un bumper se presenta en la figura 4.18.

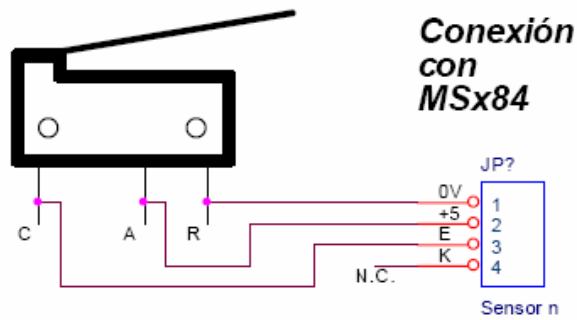


Figura 4.18 Conexión de un bumper con MSx84

El circuito general de conexión de los cinco conectores J0-J4 de la tarjeta MSx84 es el siguiente:

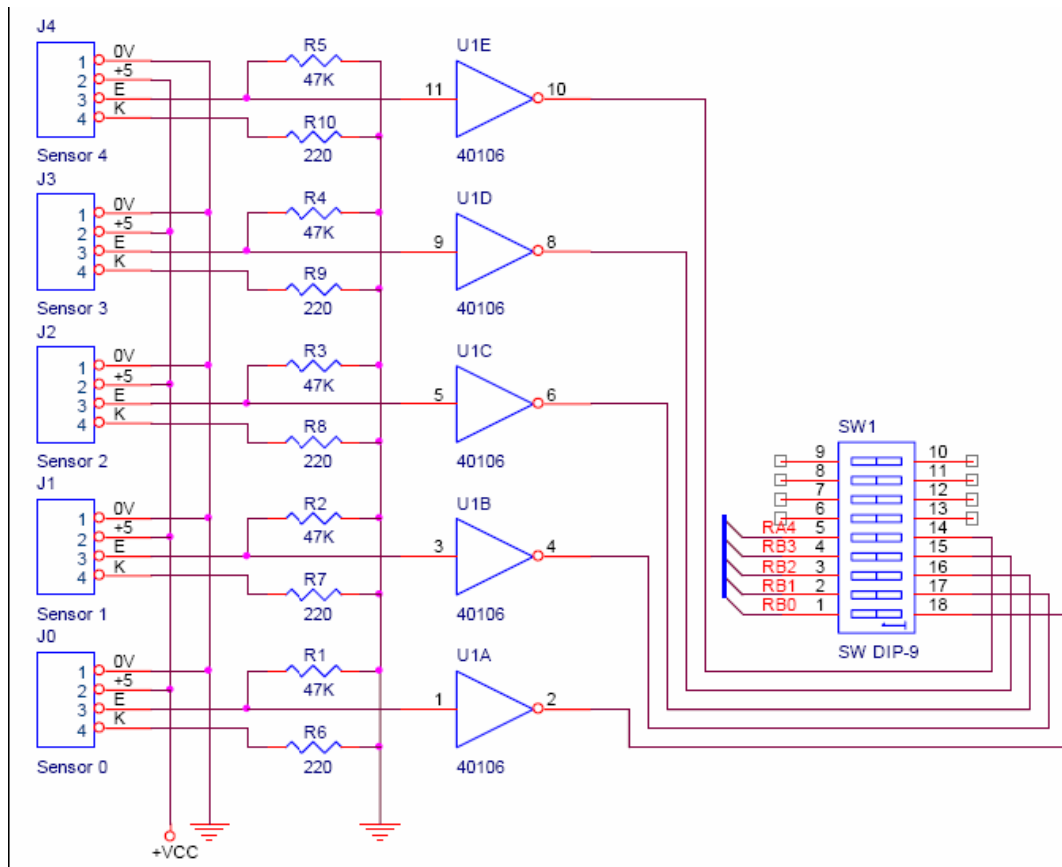


Figura 4.19 Conexión conectores J0-J4 a la tarjeta

Las imágenes que aparecen a continuación representan tanto la imagen real de los bumpers de choque que hemos utilizado, como la colocación estratégica en la estructura del robot.



Figura 4.20 Imagen real bumper utilizado

## 1.4 Motores

La placa MSx84 permite la utilización de dos motores de corriente continua (DC) o de uno paso a paso (PAP). En nuestro proyecto lo que se necesita es la primera opción, es decir, dos motores de corriente continua para el accionamiento de las ruedas motrices del robot. Así se utilizará dos servo-motores conectados a la placa MSx84. En concreto utilizamos dos servo-motores marca HITEC modelo HS422, cuyas características técnicas son:

Servo Hitec HS422	
Sistema de Control	Control por Anchura de Pulso. 1,5 ms al centro
Tensión de funcionamiento	4,8V a 6 V
Velocidad a 6V	0,16 Seg /60 grados sin carga
Fuerza a 6V	4,1 Kg · cm
Corriente en reposo	8 mA
Corriente en funcionamiento	150 mA sin carga
Corriente Máxima	1100 mA
Zona Neutra	8 $\mu$ sec
Rango Trabajo	1100 a 1900 $\mu$ sec
Dimensiones	40,6 x 19,8 x 36,6 mm
Peso	45,5 g
Rodamiento Principal	Metálico
Engranajes	Plástico
Longitud del cable	300 mm

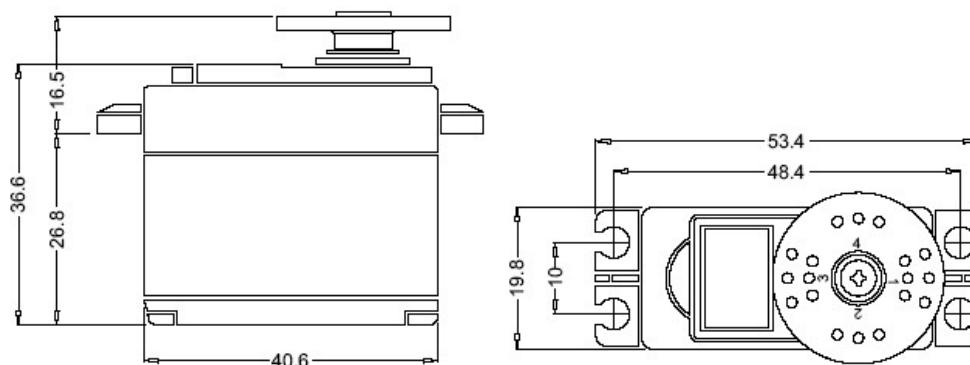


Figura 4.21 Características electro-mecánicas y dimensiones servos utilizados

Ahora bien, el funcionamiento de uno de estos servo-motores, no es el mismo que el de un motor DC. A continuación explicamos como hemos realizado dicha modificación para que el comportamiento del servo sea el mismo que el de un motor de corriente continua.

Una de las mejores características del servo Hitec HS-422 es su facilidad para ser modificado para rotación continua, con el fin de usarlo como motor propulsor en los robots. Este servo tiene unas características técnicas y físicas muy parecidas a las del popular HS300, pero con la salvedad de su mayor facilidad de modificación. Veamos uno a uno los pasos a seguir.

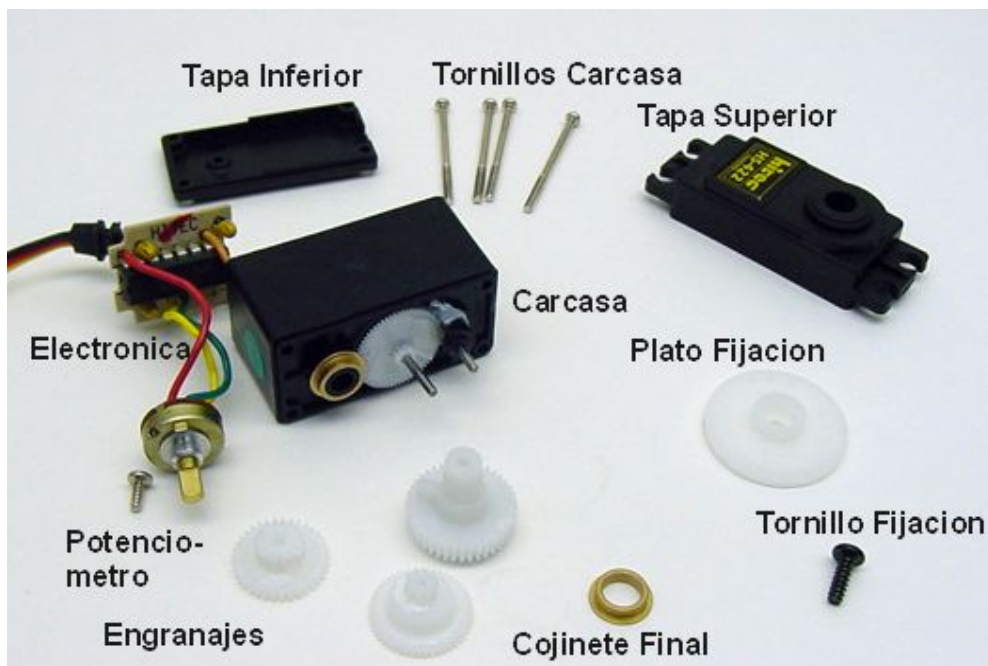


Figura 4.22 Imagen servo desmontado

1. Desmontamos el servo, aflojando los cuatro tornillos que fijan la carcasa y desmontamos los diferentes elementos, tal y como aparece en la figura 4.22.

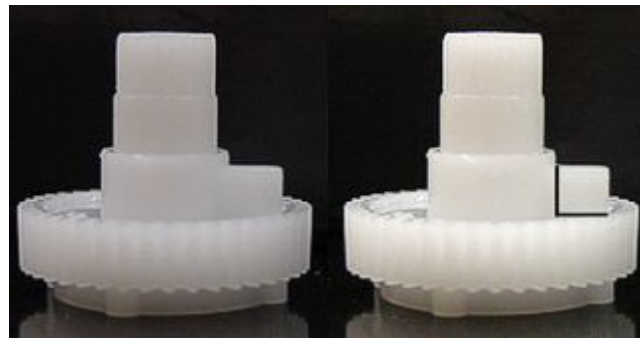
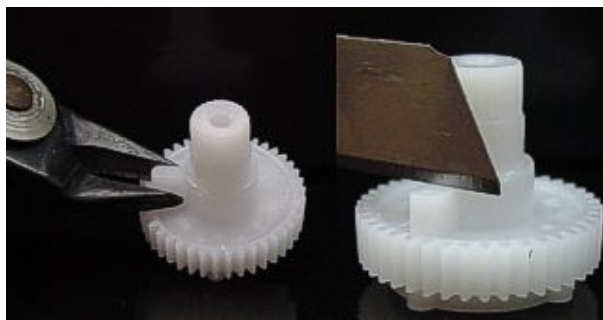


Figura 4.23 Detalle del tope que hay que cortar.

2. Localizamos el engranaje principal, el que tiene el eje estriado, y procedemos a cortar el tope limitador (Figura 4.23) con la ayuda de un cúter o unos alicates de corte. Ver figura 4.24.



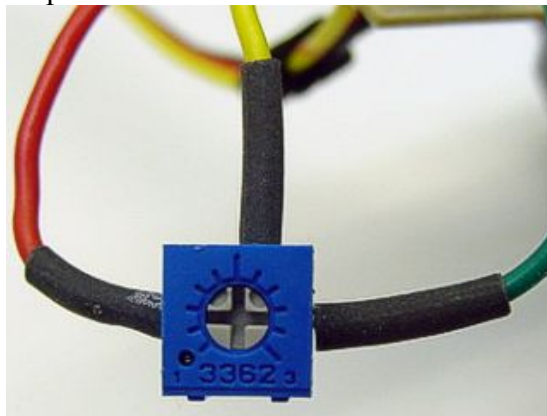
**Figura 4.24** Posibles métodos de corte.

3. Volvemos a montar todo el engranaje en la carcasa superior con la ayuda de la figura 4.24. Hay que procurar no eliminar la grasa lubricante. Evitar que queden restos de nylon entre los dientes de los engranajes.



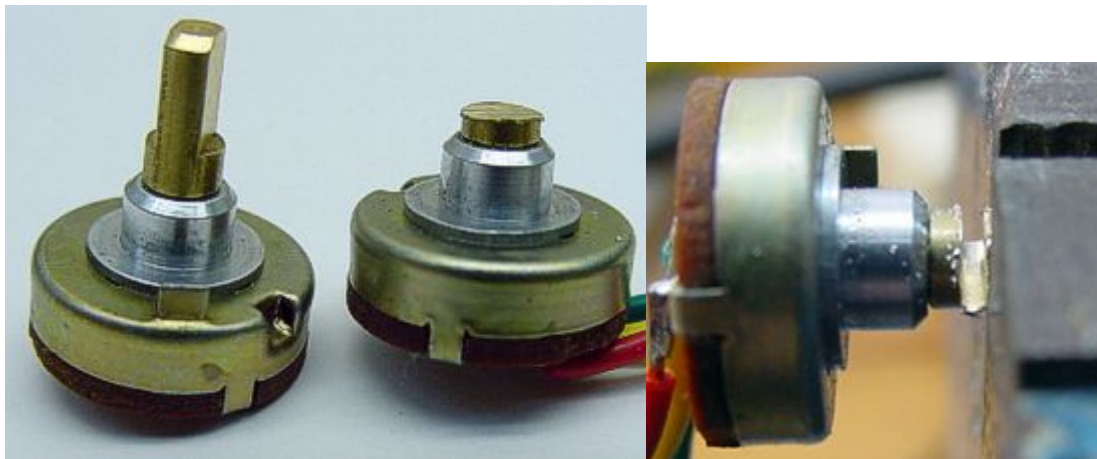
**Figura 4.25** Montaje de los engranajes.

4. Ahora hay que reemplazar el potenciómetro de control por otro de igual valor (Figura 4.26) o bien por dos resistencias de 2K7. El potenciómetro tiene la ventaja de permitir ajustar el punto neutral del servo al valor deseado. Hay que tener en cuenta el color de los cables especialmente el conectado a la patilla central.



**Figura 4.26** Conexión de un nuevo potenciómetro

5. Otra posibilidad es la de utilizar el mismo potenciómetro (Figura 4.27), pero cortando su eje con una sierra para que no sea alterado por el engranaje al girar. Esta solución tiene la ventaja de no tener que soldar nada. El eje puede cortarse fácilmente con la ayuda de un tornillo de fijación, tal y como muestra la figura 4.28.



**Figuras 4.27 y 4.28** Detalle de modificación del potenciómetro original.

6. Por último solo queda volver a montarlo todo. Hay que tener en cuenta que la posición en la que haya quedado el potenciómetro es la que marcará el valor neutral, por lo que antes de montar el servo es conveniente conectarlo al circuito de control de servos y enviar los pulsos correspondientes a la posición de reposo. A continuación giraremos el eje del potenciómetro ya cortado, hasta que este se pare. Luego montamos el potenciómetro en su sitio con cuidado de no mover el eje.

## 1.5 Micro-cámara y modulador RF canal 22 UHF

### 1.5.1 MICRO-CÁMARA

Utilizaremos una micro-cámara en B/N. El chip de la cámara<sup>4</sup> es un OV7120. Detalles de configuración y características tanto de la cámara como de la placa de pruebas, se pueden encontrar en los anexos 6 y 14 respectivamente. La cámara irá conectada a una placa de pruebas EV38, por lo que no tendremos que realizar ningún tipo de conversión analógica-digital, ni nada por el estilo, ya que de eso se encarga dicha placa de pruebas. La EV38 va equipada con un puerto paralelo, que permite su control desde el PC. También cuenta con una salida analógica en B/N que será la que utilizaremos para enviar los datos al modulador RF.

### 1.5.2 MODULADOR RF CANAL 22 UHF AUREL MODELO MAV-UHF 479

Circuito transmisor de audio y vídeo que opera en la banda de UHF, en el canal 22 a 479.55 MHz. Admite señal de video compuesto estándar PAL a 1.2 Vpp de nivel. La señal de audio con modulación FM debe tener un nivel de 1 Vpp (típico). Circuito sencillo, de bajo coste y con alimentación única de +5Vdc.



Fig. 4.29 MODULADOR RF CANAL 22 UHF

Aunque las características técnicas de este modulador las podemos ver mas detalladamente en el datasheet que incluimos, cabe destacar como ventajas la facilidad de conexionado y sus escasas dimensiones, y como desventajas, la escasa potencia con la que emite ( 2mW) y su poca fiabilidad frente a ruidos e interferencias.

#### *Forma y dimensiones de la antena*

Existen diversos tipos de antenas: longitudinal, helicoidal, en lazo, etc. Después de examinar las características básicas de cada tipo, se ha optado por utilizar una antena longitudinal porque son de fácil construcción y tienen un gran alcance. El rendimiento de este tipo de antenas disminuye a cortas distancias entre el emisor y el receptor. En nuestra aplicación, y después de diversas pruebas, esto no ha supuesto un gran inconveniente. La forma y dimensión de la antena de ambos dispositivos es la misma.

En cuanto a la longitud, cabe decir que la máxima eficiencia de la antena se consigue cuando la impedancia equivalente de la misma es puramente resistiva (resonancia).

La longitud de resonancia de la antena con el par Tx-Rx empleado y con este tipo de diseños es  $\lambda/4$ , siendo  $\lambda$  la longitud de onda para las frecuencias de trabajo de los módulos de RF. El parámetro  $\lambda$  se calcula de la siguiente forma:

$$\lambda = T \times C = 0.62 \text{ [m]}$$

donde:

$$T = \text{Periodo [s]} = 1 / 479,5 \text{ MHz}$$

$$C = \text{Velocidad de la luz} = 3E + 8 \text{ [m/s]}$$

De forma que la longitud teórica de la antena a emplear es  $0.62 / 4 = 0.16$  metros aproximadamente. Se ha comprobado el funcionamiento del prototipo con antenas de diferentes tamaños, obteniéndose el mejor resultado con una longitud de 15 cm. Esta longitud puede ser menor si se diseñan planos de masa perpendiculares a la dirección de la antena.

## 1.6 DISEÑO ESTRUCTURA MÓVIL

Para el desarrollo de la estructura mecánica del vehículo móvil se ha optado por emplear un kit comercial de MECANO, el cual contiene todas las partes mecánicas que constituyen el prototipo. Como ya justificamos antes, el empleo del kit comercial se

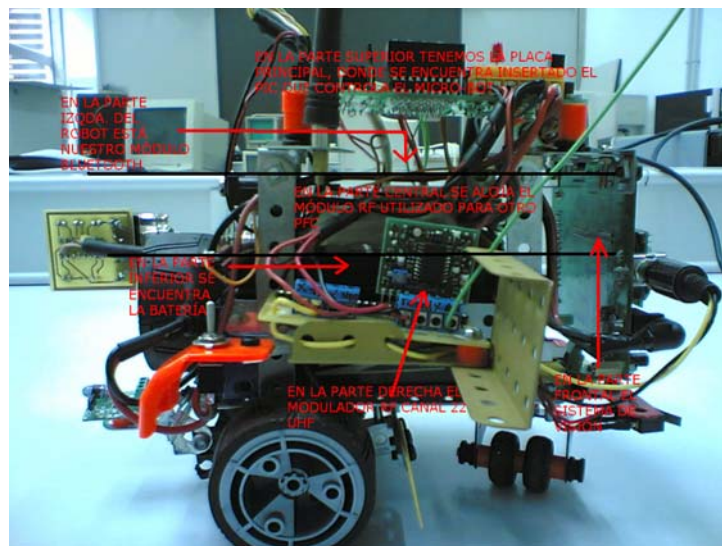


debe a la facilidad que esto ofrece frente a tener que diseñar las piezas mecánicas uno mismo.

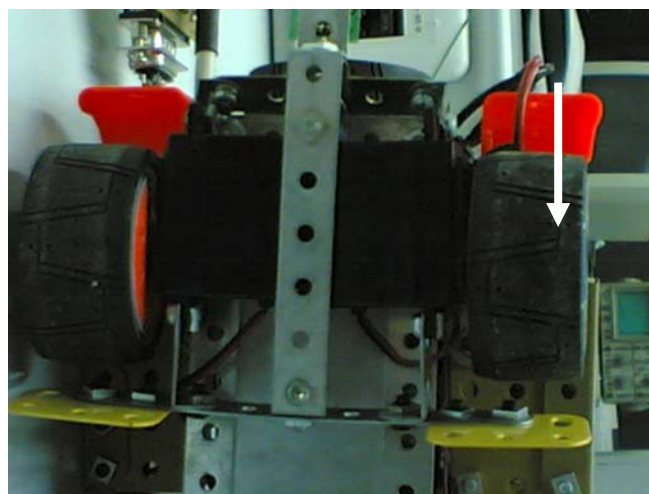
El chasis está compuesto completamente de piezas plásticas y metálicas del kit de Mecano. Su diseño es libre, sólo se ha tenido en cuenta no utilizar demasiadas piezas para restarle complejidad y peso al prototipo.

Las cavidades para las baterías y el sistema de acoplamiento de las placas también han sido diseñados con las mismas piezas. Esto, y el reducido tamaño de las placas de control y sensado, nos han ahorrado tener que insertar una plataforma sobre la que montar las placas.

Todos los ejes, acopladores y engranajes utilizados son también de plástico. Para poder adaptar ruedas de este tamaño, se ha acoplado al eje de cada servomotor un piñón, que se acopla perfectamente a la llanta de la rueda utilizada. Este mecanismo hace que la velocidad del vehículo sea también pequeña, pero en cambio, permite realizar un mejor control sobre el vehículo.



**Figura 4.30** Detalle de sistemas de acoplamiento y distribución de elementos en el vehículo



**Figura 4.31** Sistema de transmisión-reducción del vehículo (ejes y engranajes)

La estructura del vehículo se ha diseñado de tal forma que permita que se realice un giro sobre su propio eje, girando para ello los motores de un lado en contramarcha respecto a los del otro lado. En el diseño del prototipo, como ya hemos avanzado, hemos empleado dos servomotores.

El hitec HS422 es un servo de dimensiones estándar pero con una mayor potencia gracias al cojinete de salida metálico que transfiere toda la potencia al eje de salida con precisión y suavidad. El servo hs422 destaca por sus excelentes características de centrado y resolución.

También tenemos que comentar que ha sido necesario realizar distintas manipulaciones para anclarlos al chasis, así como una serie de modificaciones electromecánicas que comentaremos en el apartado correspondiente.



Figura 4.32 Servomotor hitec HS422

## 1.7 SISTEMA DE ALIMENTACIÓN

Para elegir el tipo de baterías se hizo un estudio para analizar las características de los diferentes modelos y marcas existentes en el mercado. Se pensó en cuatro tipos diferentes de baterías:

- Recargables de Ni-Cd, cuya principal ventaja es la de poder ser recargadas, y su gran inconveniente es el escaso almacenaje de carga, proporcionando por tanto, poca corriente, descargándose muy rápido.
- Alcalinas, aunque su peso y tamaño son los de una pila normal (AA) y el precio es relativamente barato, no almacenan la carga suficiente como para proporcionarle una buena autonomía al prototipo.
- Baterías de aeromodelismo, las cuales almacenan una carga media, son recargables y pesan poco, pero en cambio, son grandes y muy caras.
- Baterías de ácido, cuya mejor ventaja es la elevada carga que almacenan. El mayor inconveniente es que pesan bastante.

Finalmente, nos decidimos por baterías de ácido, ya que aunque pesan bastante tienen las características idóneas para nuestra aplicación. Concretamente, usamos una batería de Plomo 12V 6A5 HR1224W-F, de forma que la tensión de alimentación del prototipo es de 12 voltios con la batería totalmente cargada. Con este tipo de batería se ha conseguido una excelente relación autonomía / velocidad en el vehículo.

La batería se ha ubicado en piso inferior de la estructura móvil, aumentando así la estabilidad del vehículo, como se muestra en la figura 4.33.

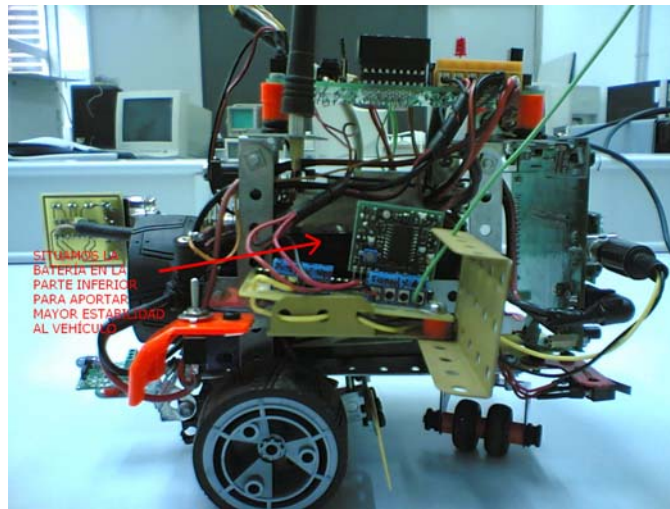


Figura 4.33 Detalle de ubicación de la batería en el vehículo

### 1.8 MAX 232

En el Mercado hay muchos circuitos integrados que permiten la conversión entre niveles ttl y niveles Rs232, Entre ellos destaca el Transceptor MAX232, fabricado por Dallas Semiconductor-MAXIM. El MAX232 convierte los niveles RS232 (cerca de +12 y -12 V ) a voltajes TTL (0 a +5 V) y viceversa sin requerir nada más que una fuente de +5V. El chip contiene dos drivers TTL -> RS232 y dos RS232 -> TTL. Necesita cuatro condensadores externos de unos pocos microfaradios para generar el voltaje RS232 internamente tal como se muestra en las figuras 4.25 y 4.26.

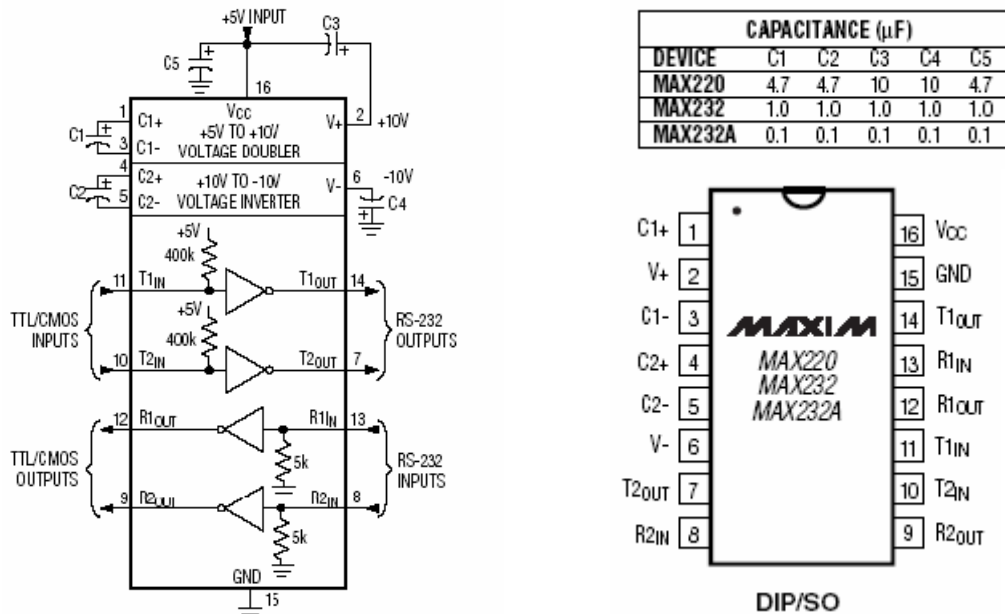


Figura 4.34 Conexiones interiores

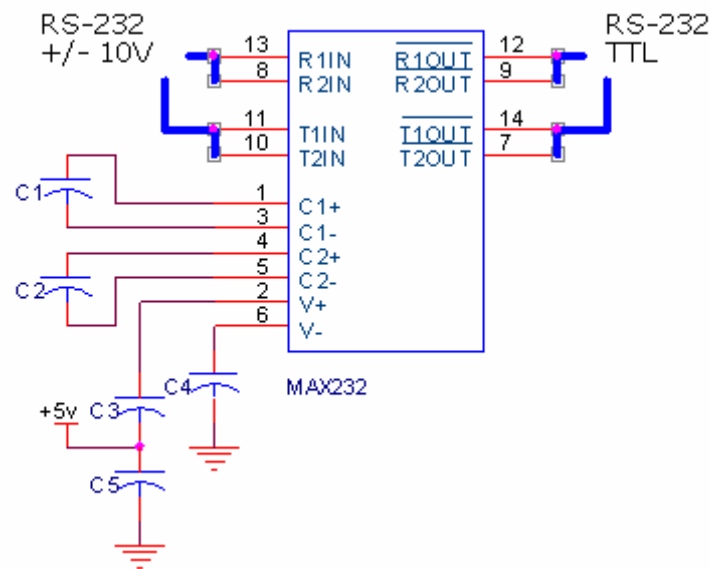


Figura 4.35 Esquema eléctrico

### 1.8.1 CONEXIÓN PUERTO RS232 Y PIC16F84

Para comunicarse con un microcontrolador se pueden utilizar sólo tres patillas del puerto serie RS232. Éstas son:

- Pin de transmisión (TxD)
- Pin de recepción (RxD)
- Pin de masa (SG)

La figura 4.36 muestra un ejemplo de circuito para comunicar un ordenador y un microcontrolador PIC16F84A a través del puerto serie RS232.

El primer problema a resolver es que los niveles lógicos TTL que salen del microcontrolador no son compatibles con los niveles lógicos RS232 del puerto serie del ordenador, razón por la cual se debe introducir en el circuito un interface que traduzca los datos del microcontrolador al puerto y viceversa.

Este chip puente puede ser el MAX232 o alguno similar. En el esquema de la figura 30 se aprecia como se utiliza un par de drivers de los cuatro disponibles.

La línea TxD (pin 3 del conector DB9) del puerto serie RS232 transmite información con niveles RS232, por tanto se conecta al pin R1IN del MAX232 para convertir estos niveles a TTL y transmitírselos al PIC16F84A a través de la patilla R1OUT.

Del mismo modo la información que envía el PIC16F84A con niveles TTL entra en la línea T1IN del MAX232 para convertirla en niveles RS232 y poder ser recibida por el puerto serie RS232 a través de su línea RxD, pin 2 del conector DB9. Hay que

tener en cuidado para no equivocarse con la polaridad de los condensadores electrolíticos al conectarlos en el circuito.

Para el control de la comunicación del ordenador con el microcontrolador es necesario el siguiente software:

- Unas subrutinas que gestionen el software del microcontrolador, como las utilizadas en la librería RS232.INC que se explicará en otro apartado.
- Un Programa para el ordenador que gestione el control del sistema, tal como el Hyperterminal de Windows que emplearemos en las primeras fases de prueba.

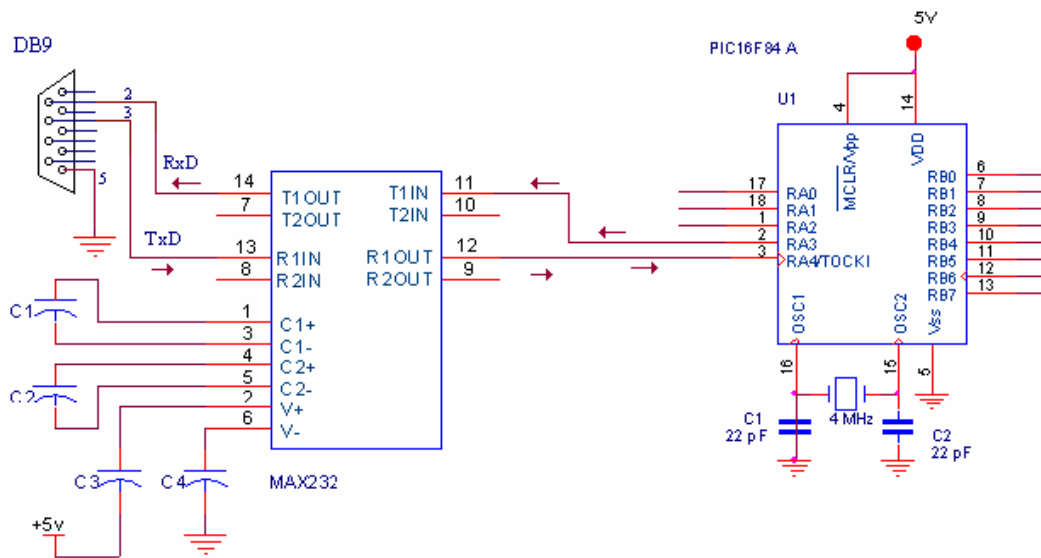


Figura 4.36 Diagrama eléctrico que muestra la conexión PIC con Rs232



---

---

# SOFTWARE DEL SISTEMA

---

---

## 1 Introducción.

En los anteriores capítulos se ha explicado la estructura del micro-robot, describiendo los bloques hardware que lo componen. En este capítulo nos centraremos en la descripción de los programas que ejecutan el microcontrolador, y el pc para gestionar las diferentes maniobras que realiza el prototipo.

El software del sistema está constituido por dos programas, uno residente en la estación base (pc), al que llamamos software base (Sfb), y el otro, residente en el microcontrolador de la placa del vehículo, al que llamamos microcontrolador micro-robot ( $\mu$ Cm).

En este capítulo comenzaremos describiendo cual son las funciones desempeñadas por el Sfb, presentando un flujograma y la explicación correspondiente de cada subrutina software que ejecute el pc Posteriormente, procederemos de igual forma con las funciones desempeñadas por el  $\mu$ Cm.

Para el desarrollo de ambos programas se ha utilizado diferentes lenguajes. Para programar la aplicación de la estación base, se ha utilizado el lenguaje de programación de alto nivel Java. Se ha utilizado este lenguaje porque facilita la programación, la detección de errores en el código y la simulación del funcionamiento de los programas (como pudimos ver en la asignatura de Fundamentos de Programación).

Por otra parte, para el  $\mu$ Cm hemos utilizado lenguaje ensamblador. Utilizamos este lenguaje por ser un lenguaje bastante conocido para nosotros (en la asignatura de Sistemas electrónicos digitales). Cabe destacar que la utilización de este lenguaje no exime de la posibilidad de programar en lenguaje picBasic o similar, alguna subrutina en la que se deba ser muy estricto con los tiempos de cómputo del microcontrolador.

Se ha empleado la herramienta RealJ para la programación en Java, porque es un buen entorno integrado de trabajo que nos facilita el desarrollo software del sistema.

Para el caso de la programación en ensamblador se ha utilizado la herramienta MPLab. Este entorno esta formado por varias aplicaciones:

- MPLab editor: Permite editar, lincar y simular.
- Mpsasm: Permite compilar y generar el fichero de código ensamblado.

## 2 Funciones desempeñadas por el Software Base

Como ya vimos en anteriores capítulos, la estación base es el bloque encargado de generar los comandos que controlan el micro-robot. Recordemos que la estación base está compuesta por un conjunto teclado-ratón comercial, una tarjeta de tv y un interfaz bluetooth RS232.

Las funciones principales que realiza el Sfb son la obtención de los comandos de control de trayectoria del vehículo y la transmisión de datos a través de su puerto serie. También realiza otras tareas como pueden ser, escuchar constantemente el puerto serie para recibir mensajes en caso de que el vehículo móvil, envíe información indicando que ha chocado, o visualizar la imagen que captura la cámara del robot.

Siguiendo un buen modelo de programación, todo el programa en Java se ha estructurado en funciones y en subrutinas. De esta forma se organiza y entiende mejor el código (presentado en el Anexo 16) y se tarda menos en depurar errores.

A continuación explicaremos cada uno de los métodos de las clases que componen nuestra aplicación.

*SimpleSerial.java: Implementa la comunicación por el puerto serie*

```
// CONSTANTES
public static final int NOPARITY      = 0;
public static final int ODDPARITY    = 1;
public static final int EVENPARITY   = 2;
public static final int MARKPARITY   = 3;
public static final int SPACEPARITY  = 4;
public static final int ONESTOPBIT   = 0;
public static final int ONE5STOPBITS = 1;
public static final int TWOSTOPBITS  = 2;

// DATOS PUBLICOS
public DataInputStream  mi_DataInputStream;
public DataOutputStream mi_DataOutputStream;
int m_Port = 0;
Stack m_ReadQueue = new Stack();

// #####CONSTRUCTORES#####

//NUEVO PUERTO DE SERIE PASANDO COMO ARGUMENTO EL NUMERO DE
PUERTO
SimpleSerial(int comPort);

// IGUAL QUE EL ANTERIOR, PERO ADEMÁS PODEMOS CONFIGURAR
PARAMETROS DEL PUERTO
SimpleSerial(int comPort, int baud, int dataBits, int stopBits, int parity);

#####
// #####FUNCIONES PÚBLICAS#####
#####

// CHEQUEA QUE EL PUERTO DE SERIE FUE INICIALIZADO CORRECTAMENTE
public boolean isValid();

//ENVIA UN CARACTER POR EL PUERTO SERIE
public boolean writeByte(int val) ;

// ENVIA UNA CADENA DE CARACTERES(STRING) POR EL PUERTO SERIE
public boolean writeString(String string);

// Espera aquí a que lleguen los datos.
public void waitForData();

// Espera aquí hasta que lleguen 'minNumBytes' de datos
```



```

public void waitForData(int minNumBytes);

// lee un byte de datos. Sino lo hay espera hasta que haya datos disponibles
// Si ocurre un error devuelve -1.
public int readByte();

// lee una cadena de caracteres(String).De no haber ninguna disponible
// devuelve una cadena vacía.
// Al contrario que la anterior, esta rutina no espera la llegada de datos.
public String readString();

//Devuelve cuantos bytes hay listos para ser recibidos
//Durante el tiempo que se realiza la lectura, también
// pueden llegar nuevos bytes listos para ser recibidos.
public int available();

/*Cierra el puerto de serie. Normalmente el puerto se cierra
automáticamente; esta rutina se utiliza sobre todo cuando se
quiere cerrar el puerto y reabrir otro */
public void close();

// Devuelve un input stream para este puerto.Permite un gran control de la IO de datos
public InputStream getInputStream();

// Devuelve un output stream para este puerto.Permite un gran control de la IO de datos
public OutputStream getOutputStream();

#####
// #####MIEMBROS NO PUBLICOS#####
#####

// ABRE EL PUERTO DE SERIE Y DEVUELVE UN MANEJADOR PARA ESE
PUERTO
// EL MANEJADOR ES GUARDADO PARA SER LLAMADO EN LAS _read/_write
native int _openSerialPort(String comPort, int baud, int dataBits, int stopBits, int parity);

// Escribe los bytes de la cadena 'string'.Devuelve el nºde bytes escritos.
native int _writeSerialPort(int port, String string);

// Lee todos los bytes disponibles en el buffer. Sino hay nada disponible
//devuelve un array de longitud 0.
// Esta función nunca se bloquea, y devuelve el array de datos inmediatamente.
native byte[] _readSerialPort(int port);
native void _closeSerialPort(int port);

JavaTerm.java: Ejemplariza la utilización de SimpleSerial.java.

JFrame m_Frame; // ventana principal frame
JTextArea m_TextInput; // parte de arriba de la ventana donde el texto será leído
JCheckBoxMenuItem m_PortMenuItem[] = new JCheckBoxMenuItem[5];

// menú para seleccionar puerto com
JMenuItem m_ClearMenuItem = new JMenuItem("Limpiar Buffers");

// 'limpiar buffers' menu
SimpleSerial m_SerialPort = null; // Puerto de serie

Int m_PortIndex; // Indica qué com vamos a utilizar

```

```

static final String m_PrefsFileName = new String("JavaTerm.pref");    // nombre del
archivo de preferencias

JPanel panelPantalla;          //constituyen la parte central de la ventana, y se utilizan
para enviar texto al robot

Imagelcon der = new Imagelcon("derecha.gif");
Imagelcon izq = new Imagelcon("izquierda.gif");
Imagelcon arr = new Imagelcon("arriba.gif");
Imagelcon aba = new Imagelcon("abajo.gif");
Imagelcon sto = new Imagelcon("stop.gif");
JButton adelante = new JButton(arr);
JButton atras = new JButton(aba);
JButton derecha = new JButton(der);
JButton izquierda = new JButton(izq);
JButton parar = new JButton(sto);
JPanel panelBotones;
robot pantallazo;

//MÉTODOS-

//CONSTRUCTOR:
JavaTerm();

//PÚBLICOS:
// Inicializamos el puerto de serie y le asociamos Input/Output streams
private void initSerialPort() throws IOException;

// Cuando el usuario cambia el puerto desde el menú, se salta a esta parte del código
public void itemStateChanged(ItemEvent e);

// Si el usuario selecciona "Limpiar Buffers" saltamos aqui.
public void actionPerformed(ActionEvent e)

robot.java: Esta parte de la aplicación es la encargada de realizar la captura de video.

//VARIABLES:
JTabbedPane tabbedPane;
static Player player;
Component visualComponent;
CaptureDeviceInfo camara, cam;
MediaLocator loc;
DataSource formattedSource = null;
FrameGrabbingControl fgc = null;
Buffer buf = null;
BufferToImage bti = null;
static RenderedImage entrada = null;
Vector dispVideo = new Vector(), lista = null, nombres = new Vector();
Format[] formatos;
JPanel pVisual = new JPanel();
int cuenta = -1;
Border borde;

//MÉTODOS

//CONSTRUCTOR
public robot();

```

```

//PÚBLICOS
public static void display(String titulo, RenderedImage imag);

//MONTARÁ LA PANTALLA QUE REALIZA LA CAPTURA, AÑADIENDOLE UN
TÍTULO.

JPanel createPane(String s); //DEVUELVE EL JPANEL DONDE SE HARÁ LA
SELECCIÓN DEL DRIVER DE CAPTURA

public void crearPlayer(); // CREA EL PLAYER DE CAPTURA Y LO INICIA

```

### **3 Funciones desempeñadas por el Microcontrolador del Micro-robot**

Este microcontrolador es el encargado del guiado del micro-robot y, principalmente, realiza las siguientes funciones:

- Atender comandos remotos recibidos por el puerto serie.
- Comandos de guiado.
- Informar a la estación base de posibles colisiones e inmediatamente cambiar el sentido de la marcha, utilizando información proporcionada por sensores, y generando para ello las señales de alimentación, control y habilitación de los drivers de los motores del vehículo móvil.

Al igual que en el apartado anterior, todo el programa en ensamblador se ha estructurado en funciones y en subrutinas de atención a la interrupción para una mejor organización del código presentado en el Anexo 15.

En primer lugar, se presenta un flujograma general en el que se puede observar la secuencia de tareas y subrutinas que ejecuta este programa. A continuación, se procede a describir cada subrutina de forma separada y en el orden en el que se van ejecutando, presentándose en primer lugar un flujograma particular y posteriormente explicándose el desarrollo de la subrutina.

La lista de subrutinas es la siguiente: ComprobarSensorD(), RS232\_LeeDato(), TesteaTeclado(),EjecutaOrden(),y la subrutina de atención a la interrupción Rb0/Int (TesteaSensores()). Además de estas subrutinas, en el archivo RS232\_MEN.inc, se encuentran las subrutinas encargadas de enviar y recibir caracteres por el puerto de serie, y las subrutinas del archivo RS232.inc. Estas subrutinas permiten realizar las tareas básicas de control de la transmisión serie asíncrona según normas RS-232.

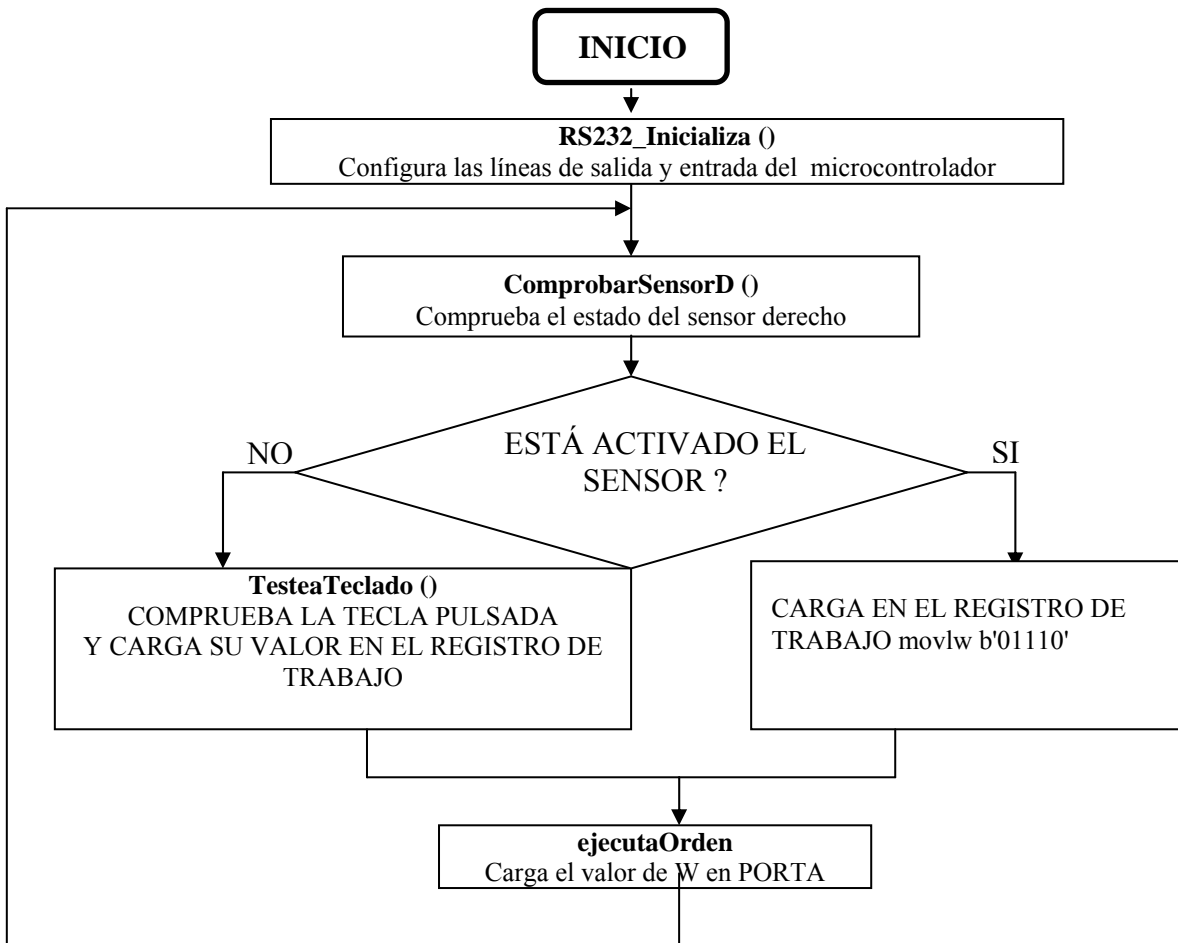


Figura 5.1 Flujograma general programa Microcontrolador del Micro-robot

### 3.1 ComprobarSensorD ()

Mediante esta función el microcontrolador comprueba continuamente el estado de la línea 1 del PORTB, línea a la que se conecta el sensor derecho.

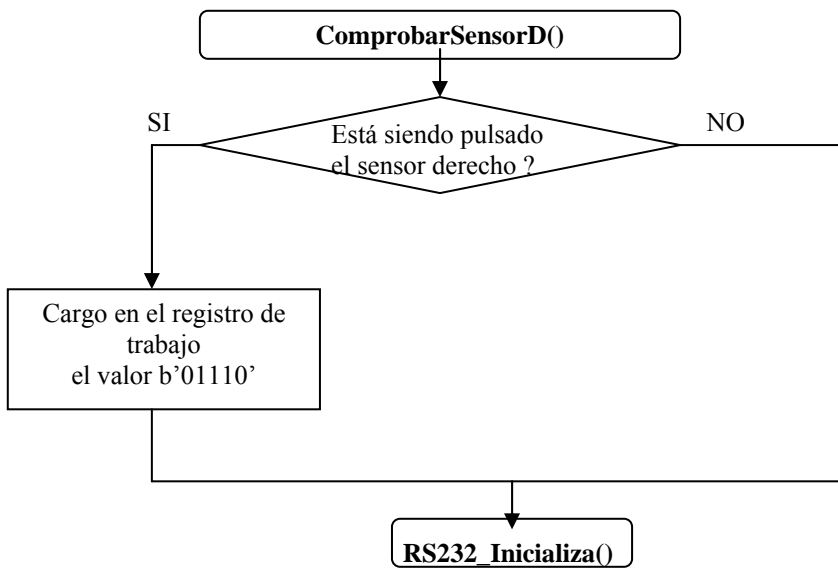


Figura 5.2 Flujograma de ComprobarSensorD()

Como vemos en el flujograma, si el sensor derecho es pulsado, se cargará en el registro de trabajo W, el valor que hace que el robot gire hacia la izquierda, ya que supuestamente ha chocado por la derecha. Este valor será cargado posteriormente en el PORTA, que es el puerto que controla los motores, mediante la llamada a EjecutaOrden.

### 3.2 RS232\_Inicializa()

Esta función configura los valores de los registros del 16f84 que controlan las líneas de entrada y salida.

En primer lugar, se configura la línea 5 del PORTB como entrada. A continuación, configuramos como salida la línea 2 del PORTB. Esta subrutina no se encargará de configurar la recepción serie. Para ello utilizaremos otras subrutinas del archivo RS232.inc, como Retardos\_100micros(), estableciendo el modo de funcionamiento (8 bits a 9600 Baudios).

Como hemos dicho no se encarga de configurar la recepción serie, pero si de leer los bits que recibe el micro-controlador por la patilla de entrada. El microcontrolador lee el dato por la línea de entrada comenzando por el bit de menor peso. El dato leído se envía finalmente en el registro de trabajo W.

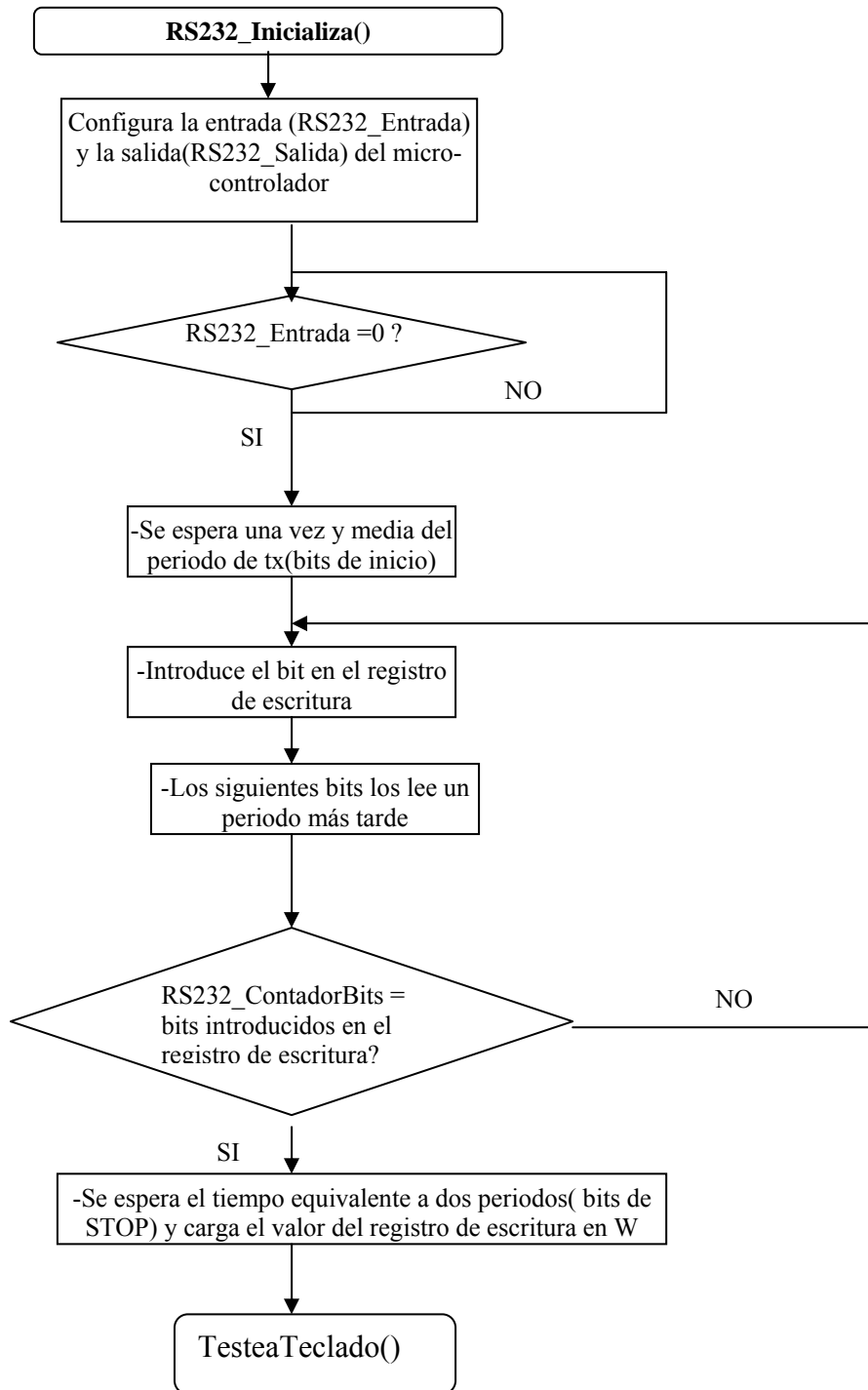


Figura 5.3 Flujograma de RS232\_Inicializa()

### 3.3 TesteaTeclado():

Esta subrutina va comparando el valor de la tecla pulsada, con cada uno de los valores que tenemos almacenados que corresponden a las órdenes que le podemos mandar al robot:

```

TeclaAdelante = 'W'
TeclaAtras   = 'S'
TeclaIzqda   = 'A'
TeclaDrecha  = 'D'
TeclaStop    = ''

```

Si el valor coincide, carga en el registro W el valor en binario, que después cargaremos en PORTA, correspondiente al giro de los motores deseado. Si por el contrario no coincide, pasamos a compararlo con el siguiente valor disponible.

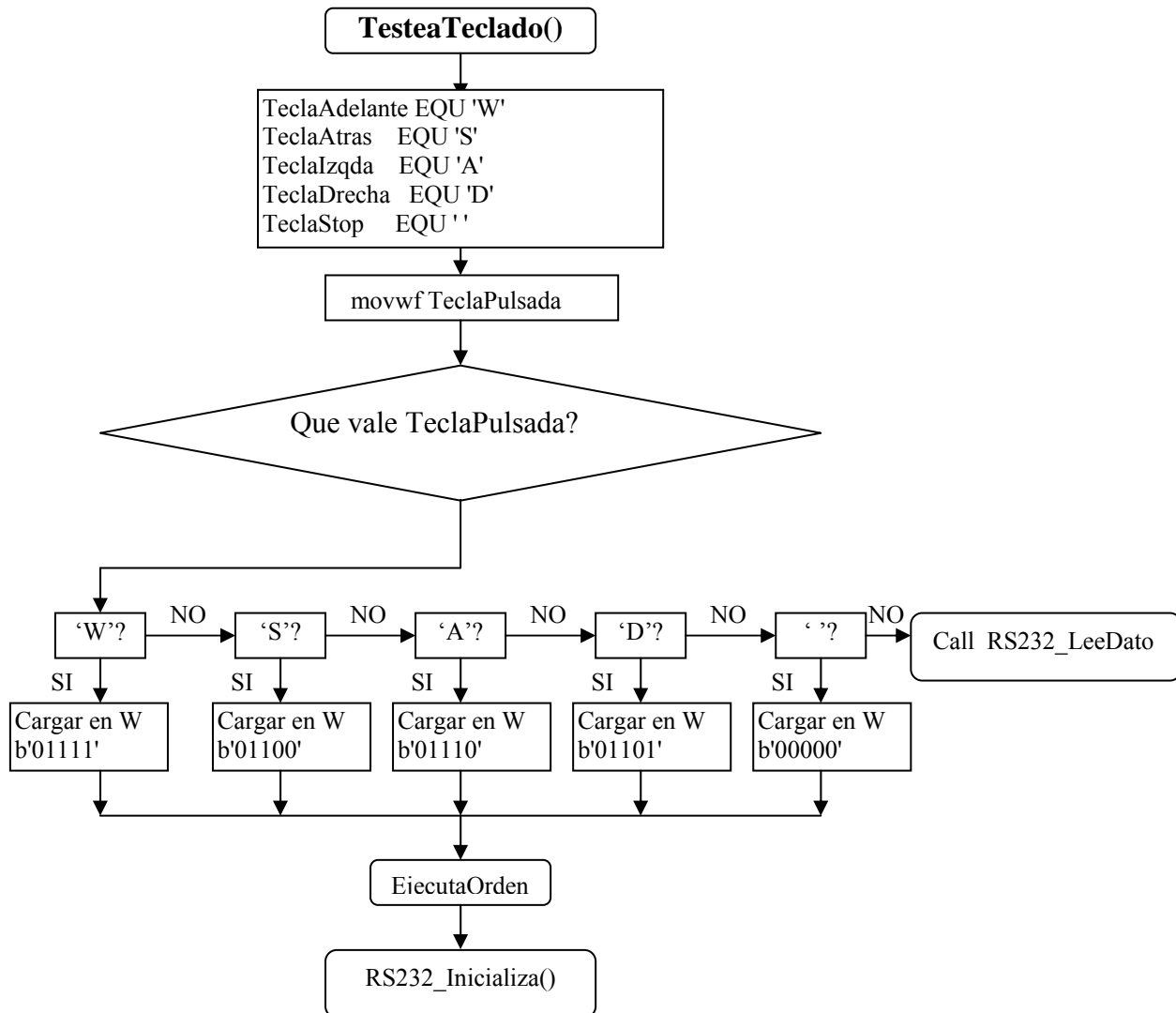
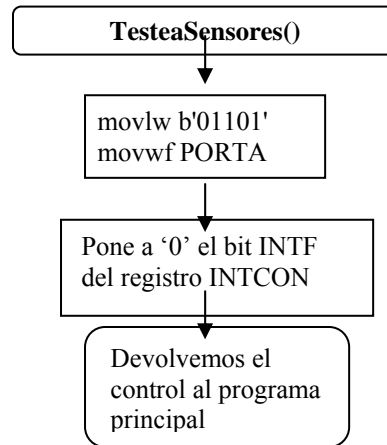


Figura 5.4 Flujograma de TesteaTeclado()

### 3.4 Subrutina de atención a la interrupción del rb0/Int (TesteaSensores()):

La función que cumple esta subrutina es la de leer constantemente el estado del sensor Izquierdo, y en caso de que este sea activado, ejecutar la rutina de la interrupción como explica el siguiente diagrama (si el sensor ha sido activado, el robot tiene que girar hacia la derecha, ya que el choque se ha producido por la izquierda):



**Figura 5.5** Flujograma de subrutina interrupción



## **6. FASES DEL PROYECTO**

---

---

### **1 FASE 1**

#### **1.1 Recopilar información Hardware**

Esta es la primera etapa del proyecto, en la cual se debe realizar un profundo estudio de las distintas posibilidades que existen en el mercado a la hora de realizar nuestro microrobot. Así debemos decidir que tipos de sensores son los mas adecuados para nuestros proyecto, que tipo de motores son los necesitamos para nuestra aplicación, de que forma vamos a comunicar el robot con el Pc y que con que tipo de conexiones deberemos trabajar.

##### *1.1.1 Sensores telemétricos.*

Según la especificación del proyecto nuestro robot debe evitar estrellarse contra un objeto, para ello se instalará algún tipo de sensor que evite el choque o que en su defecto, nada más chocar con alguna parte prevista para ello cambie inmediatamente de sentido. Dadas las características de nuestro proyecto vemos que los tipos de sensores mas adecuados para lograr el objetivo impuesto serían tres:

- Sensores mecánicos (Bumpers)
- Sensores de Infrarrojos
- Sensores de Ultrasonidos

##### *BUMPERS*

Los sensores de contacto son los dispositivos más simples de todos los sensores que podemos encontrarnos, ya que son interruptores que se activan o desactivan si se encuentran en contacto con un objeto, por lo que de esta manera se reconoce la presencia de un objeto en un determinado lugar.

Su simplicidad de construcción añadido a su robustez, los hacen muy empleados en la construcción de micro robots. La figura 6.1 muestra esquema básico de cómo es uno de estos sensores.

La descripción de este tipo de sensores es muy simple, el bumper es un conmutador de 2 posiciones con muelle de retorno a la posición de reposo y con una palanca de accionamiento más o menos larga según el modelo elegido.



Figura 6.1 Esquema básico bumper

En cuanto a su funcionamiento, también es muy sencillo: en estado de reposo la patilla común (C) y la de reposo (R) están en contacto permanente hasta que la presión aplicada a la palanca del bumper hace saltar la pequeña pletina acerada interior y entonces el contacto pasa de la posición de reposo a la de activo (A), se puede escuchar cuando el bumper cambia de estado, porque se oye un pequeño clic, esto sucede casi al final del recorrido de la palanca.

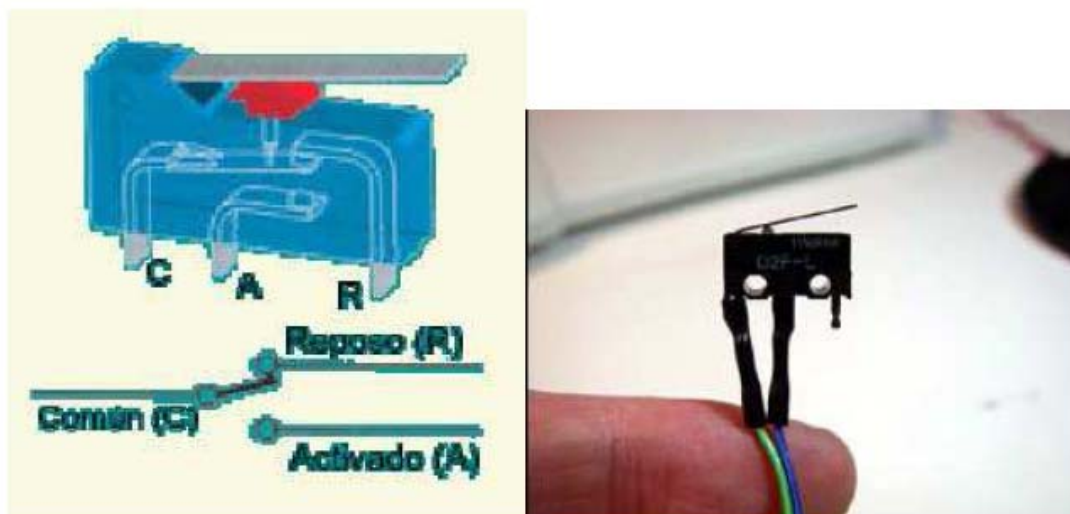


Figura 6.2 Imagen bumpers comerciales

#### *Bumper comerciales:*

Sensor electromecánico (Bumper) de simple circuito y doble posición (DPST) con leva.

#### *SENSORES IR*

En el mercado existen múltiples sensores de infrarrojos para la medición de distancias. Vemos las posibles opciones para el robot.



Figura 6.3 Sensor infrarrojo sharp GP2D02

El Sharp GP2D02 es un sensor medidor de distancias por infrarrojos con un rango de trabajo de 10 a 80 cm., con un consumo mínimo cuando esta en reposo. El sensor utiliza solo una línea de entrada y otra de salida para comunicarse con el procesador principal. Su utilización es tan sencilla como mandar un pulso bajo en la entrada de control, esperar 70 ms y mandar 8 pulsos de reloj para leer el valor de la distancia medida. El sensor se entrega con un conector de 4 pines y un diodo para poder conectarlo a cualquier circuito CMOS/TTL. Tensión de funcionamiento 5V, Temperatura funcionamiento :-10 a 60°C, Consumo Max - Min: 35 mA, 3uA.

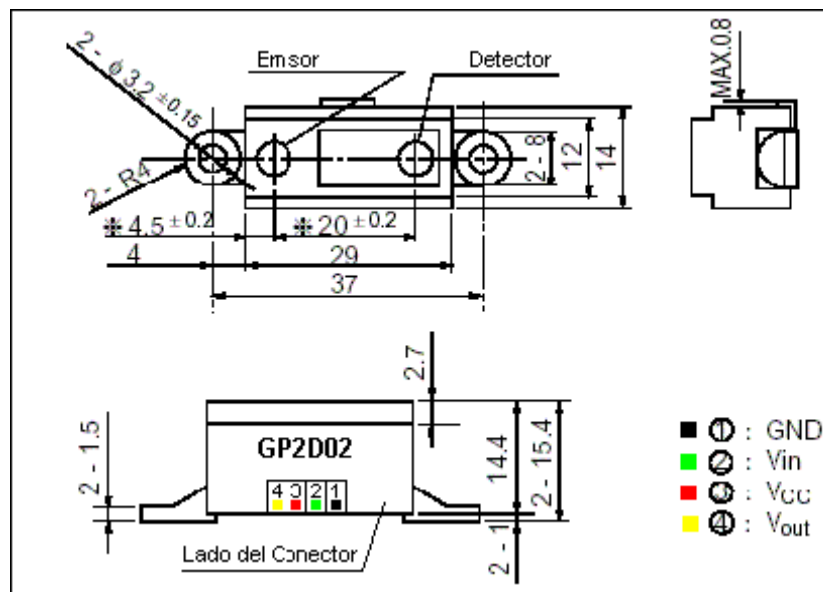


Figura 6.4 Diagrama de conexiones sensor infrarrojo

### SENSOR I.R. DE OBSTACULOS MSE-S135

El circuito MSE-S135 es un detector IR de obstáculos que detecta la presencia de un objeto sin contacto físico con el mismo. Consiste en un emisor/detector de luz infrarroja modulada. Esta característica lo hace prácticamente inmune a interferencias provocadas por otras fuentes de luz. Un diodo emisor emite una haz infrarrojo modulado a una frecuencia de 7.7 KHz. El rebote de dicho haz sobre un objeto, es captado por un foto transistor detector que acondiciona la señal recibida, compara si

corresponde con la señal emitida y, en caso afirmativo, genera una señal de salida activa por flanco descendente. Si no se detecta ningún rebote de la señal emitida, la salida se mantiene en reposo, a nivel lógico "1". El dispositivo dispone de un orificio que permite una flexible instalación y sujeción del mismo sobre cualquier tipo de estructura.



**Figura 6.5** Imagen real sensor I.R.

### *SENSORES ULTRASONIDOS*

#### *SRF235*



**Figura 6.6** Sensor ultrasonidos alta frecuencia

SRF235 es un medidor de distancias por ultrasonidos de alta frecuencia y haz estrecho con conexión I2C de múltiples aplicaciones dentro de la robótica. Este sensor de ultrasonidos destaca por utilizar un sensor ultrasónico industrial que funciona a una frecuencia de 235Khz y tiene un haz de solo 15 grados, lo que proporciona una medida de altísima precisión y gran direccionabilidad. Debido a su alta frecuencia, puede efectuar hasta 100 mediciones por segundo, siendo toda la memorización totalmente interna liberando al microcontrolador de retardos y tiempos de espera. Gracias a su bus I2C se pueden conectar hasta 16 unidades con solo dos líneas de entrada y salida, permitiendo montar un completo sonar perimetral en cualquier robot con solo dos pines libres.

Este sensor se puede utilizar a la vez que otros sensores de 40 KHz como el srf8 ya que no se hacen interferencias entre si.

El sensor de distancias por ultrasonidos SRF235 dispone de un transductor industrial de alta frecuencia que opera a 235KHz utilizado para transmitir y recibir. El

patrón del haz de detección que se muestra a continuación compara los 15 del Sensor SRF235 con los 55 del patrón común de SRF04.

SRF04

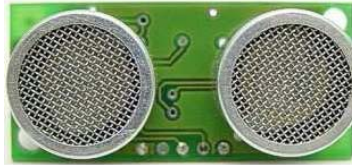


Figura 6.7 Sensor distancias por ultrasonidos

SRF04 es un sensor de distancias por ultrasonidos capaz de detectar objetos y calcular la distancia a la que se encuentra en un rango de 3 a 300 cm. El sensor funciona por ultrasonidos y contiene toda la electrónica encargada de hacer la medición. Su uso es tan sencillo como enviar el pulso de arranque y medir la anchura del pulso de retorno. De muy pequeño tamaño, SRF04 destaca por su bajo consumo, gran precisión y bajo precio por lo que esta reemplazando a los sensores polaroid en los robots mas recientes.

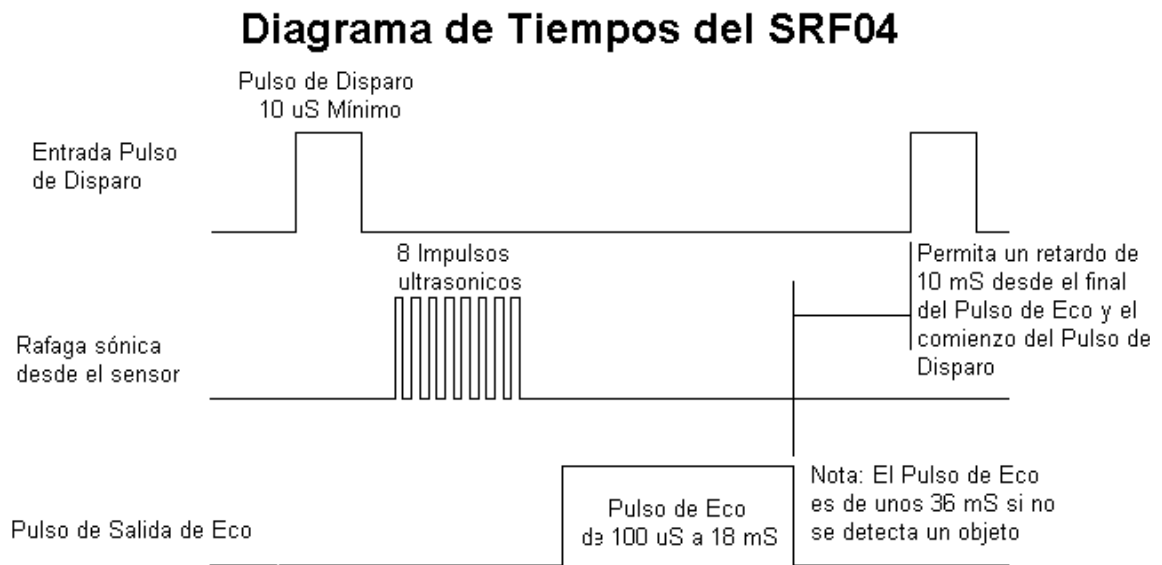


Figura 6.8 Diagrama de tiempos SRF04

*FUNCIONAMIENTO*

El sensor SRF04 funciona emitiendo impulsos de ultrasonidos inaudibles para el oído humano. Los impulsos emitidos viajan a la velocidad del sonido hasta alcanzar un objeto, entonces el sonido es reflejado y captado de nuevo por el receptor de ultrasonidos. Lo que hace el controlador incorporado es emitir una ráfaga de impulsos ya continuación empieza a contar el tiempo que tarda en llegar el eco. Este tiempo se

traduce en un pulso de eco de anchura proporcional a la distancia a la que se encuentra el objeto.

### **1.1.2 Motores**

En un micro-robot se precisan motores para el funcionamiento de poleas o engranajes internos en la carcasa del robot así como para el simple, giro de sus ruedas. Se utilizan básicamente dos tipos de motores:

- Motores de corriente continua.
- Motores paso a paso.

#### *Motores DC*

Los motores de corriente continua ( DC )suelen controlar el giro de las ruedas del micro robot e incluyen un grupo reductor que proporciona un buen par de fuerza y una gran estabilidad en la velocidad de giro. Son motores de baja inercia en los instantes de arranque y parada. Suelen admitir un amplio rango de alimentación en DC y tienen un peso y tamaño reducidos. Este tipo de motores proporcionan un par proporcional al voltaje de entrada, lo que significa que tienen una regulación precisa. Sin embargo no disponen de recursos para saber el ángulo girado y tiene que trabajar en lazo cerrado con sensores para controlar el ángulo girado.

#### *Motores PAP*

Los motores paso a paso (PAP) gira ángulo muy exactos y su control es de tipo digital. Se alimentan en DC V. Estos motores tienen un conjunto de sensores con tratados con señales digitales en los extremos del motor para que gire un ángulo determinado. El motor de PAP trabaja en lazo abierto ya que el giro del eje es conocido por el propio control. Una desventaja de este tipo de motores es que su eje no recorre todas las posiciones y en consecuencia no se cubre toda el área de trabajo.

### **1.1.3 Comunicación Inalámbrica**

#### *Transceptor multi punto sin caja WIZ2-434-RS / RSB*

Este tipo de módulos transceptores están preparados para la transferencia de datos vía radio entre uno o mas módulos. Puede crearse una red de transceptores con un módulo principal o master y hasta 255 esclavos. Cada módulo se puede configurar como master o como esclavo asignando a cada uno una dirección. Cada vez que el master transmite un paquete de datos debe indicar a qué esclavo va dirigido. Cuando recibe un paquete de datos se recibe también de qué esclavo procede. Estos paquetes de datos pueden ser de hasta 96 bytes. Se alimentan con una F.A. externa de entre 9 y 15Vdc con positivo al centro del conector. La velocidad se selecciona entre 9600, 19200, 57600 y 115200 baudios y el alcance aproximado es de 500 m en exteriores, con una frecuencia de 433.92 MHz.

Cumple con las especificaciones ETSI 300-220.

La única diferencia entre el transceptor WIZ2-434-RS y el WIZ2-434-RSB es que este último va introducido en una caja de plástico que le confiere una mayor solidez. En ambos casos se incluye la antena exterior.

Desde nuestra página se puede bajar el software de demostración WIZ-2 Pack que suministra el fabricante y que permite la configuración de cada módulo así como la comunicación vía radio entre dos o mas PC's.

También se puede emplear los conocidos programas de comunicación del tipo Hyper Terminal de windows o similares o bien desarrollar un software específico para aplicaciones específicas.

#### *TRANSCEPTOR AUREL mod. RTL-DATA-SAW-IA*

Módulo emisor/receptor (transceptor) con una única antena. Permite la comunicación half-duplex vía radio entre dos estaciones con una velocidad máxima de 2400 baudios. Tensión de alimentación +5Vdc, consumo de 4.5mA el transmisor y de 2.5 mA el receptor, frecuencia 433.92 MHz. Conforme a la norma EN 300-220 con disponibilidad del correspondiente certificado.



**Figura 6.9** TRANSMISOR AUREL mod. TX-4M50SA05IA

Módulo transmisor con antena integrada y tensión de alimentación de +5 Vdc. Frecuencia de transmisión 433.92 MHz, consumo de 26 mA, potencia de salida +6dB. Ideal para aplicaciones donde sea necesario modular y transmitir señales digitales. Conforme con el estándar Europeo EN 300 220.

#### *INTERFAZ BLUETOOTH RS232 PROMI SD202*

**SD202**



**Figura 6.10** INTERFAZ BLUETOOTH RS232 PROMI SD202

Con Promi-SD, puedes efectuar conexiones inalámbricas Bluetooth hacia cualquier dispositivo Bluetooth que soporte el SPP (Perfil Puerto Serie). Pues, cuando utilizas el Promi-SD como cable de sustitución, aprovecha la conexión automática del

Promi-SD. Cuando dos Promi-SD están instalados a éste propósito, se conectan automáticamente cuando se encienden. Colocados en su rayo cobertura, sirven de cable virtual RS-232.

### 1.1.4 Comunicación puerto serie PC.

#### Introducción

Una manera de conectara dos dispositivos es mediante comunicaciones serie asíncronas. En ellas los bits de datos se transmiten "en serie" (uno detrás de otro) y cada dispositivo realiza tiene su propio reloj. Previamente se ha acordado que ambos dispositivos transmitirán datos a la misma velocidad.

#### Comunicaciones serie asíncronas

Los datos serie se encuentran encapsulados en tramas de la forma:



Figura 6.11 Encapsulación de las tramas

Primero se envía un bit de start, a continuación los bits de datos (primero el bit de mayor peso) y finalmente los bits de STOP. El número de bits de datos y de bits de Stop es uno de los parámetros configurables, así como el criterio de paridad par o impar para la detección de errores.

Normalmente, las comunicaciones serie tienen los siguientes parámetros: 1 bit de Start, 8 bits de Datos, 1 bit de Stop y sin paridad.

En esta figura se puede ver un ejemplo de la transmisión del dato binario 10011010. La línea en reposo está a nivel alto:

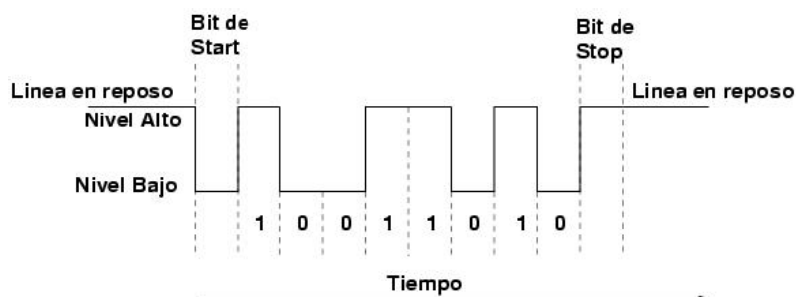


Figura 6.12 Tx del dato binario 10011010

#### Norma RS232

La Norma RS-232 fue definida para conectar un ordenador a un MODEM. Además de transmitirse los datos de una forma serie asíncrona son necesarias una serie



de señales adicionales, que se definen en la norma. Las tensiones empleadas están comprendidas entre +15/-15 voltios.

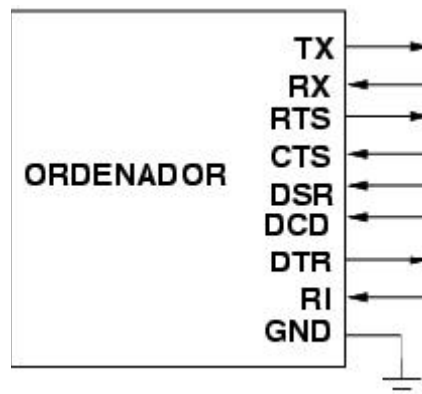


Figura 6.13 Conexión de un microcontrolador al puerto serie del PC

Para conectar el PC a un microcontrolador por el puerto serie se utilizan las señales Tx, Rx y GND. El PC utiliza la norma RS232, por lo que los niveles de tensión de los pines están comprendidos entre +15 y -15 voltios. Los microcontroladores normalmente trabajan con niveles TTL (0-5v). Es necesario por tanto intercalar un circuito que adapte los niveles:

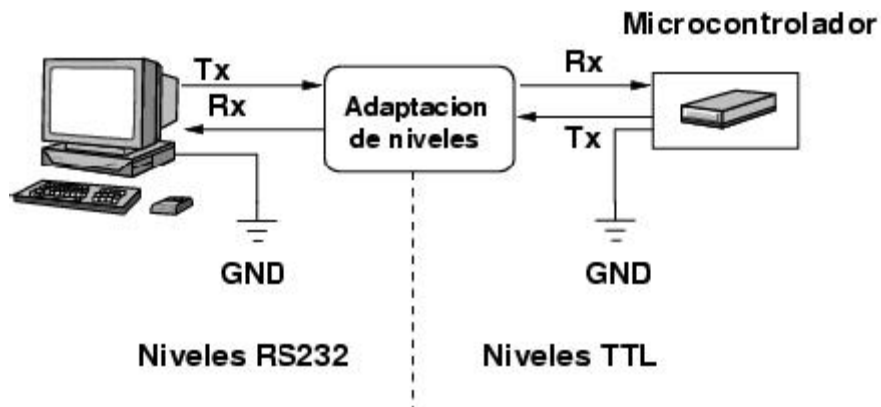


Figura 6.14 Conexión circuito adaptador de niveles

Uno de estos circuitos, que se utiliza mucho, es el MAX232.

*El conector DB9 del PC*

En los PCS hay conectores DB9 macho, de 9 pines, por el que se conectan los dispositivos al puerto serie. Los conectores hembra que se enchufan tienen una colocación de pines diferente, de manera que se conectan el pin 1 del macho con el pin 1 del hembra, el pin2 con el 2, etc..

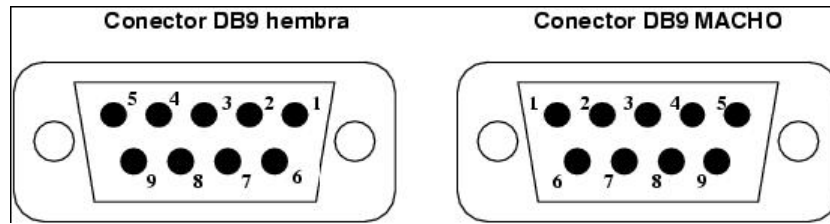


Figura 6.15 Conectores DB9 macho y hembra

La información asociada a cada uno de los pines es la siguiente:

Número de pin	Señal
1	DCD (Data Carrier Detect)
2	RX
3	TX
4	DTR (Data Terminal Ready)
5	GND
6	DSR (Data Sheet Ready)
7	RTS (Request To Send)
8	CTS (Clear To Send)
9	RI (Ring Indicator)

Tabla 6.1 Información pines DB9

### El chip MAX 232

Este chip permite adaptar los niveles RS232 y TTL, permitiendo conectar un PC con un microcontrolador. Sólo es necesario este chip y 4 condensadores electrolíticos de 22 micro-faradios. El esquema es el siguiente:

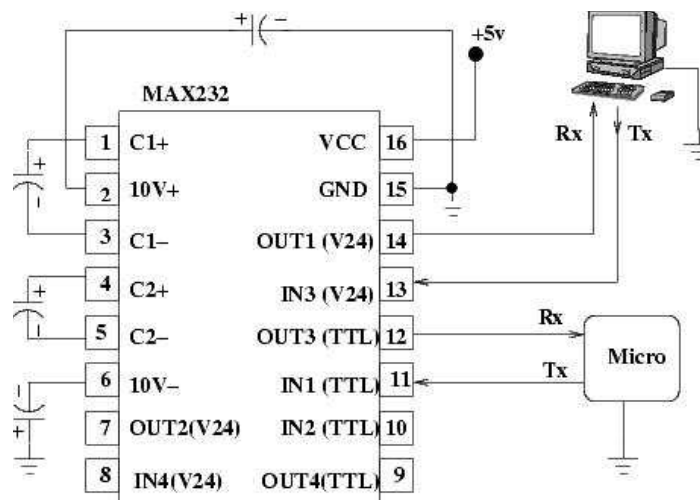
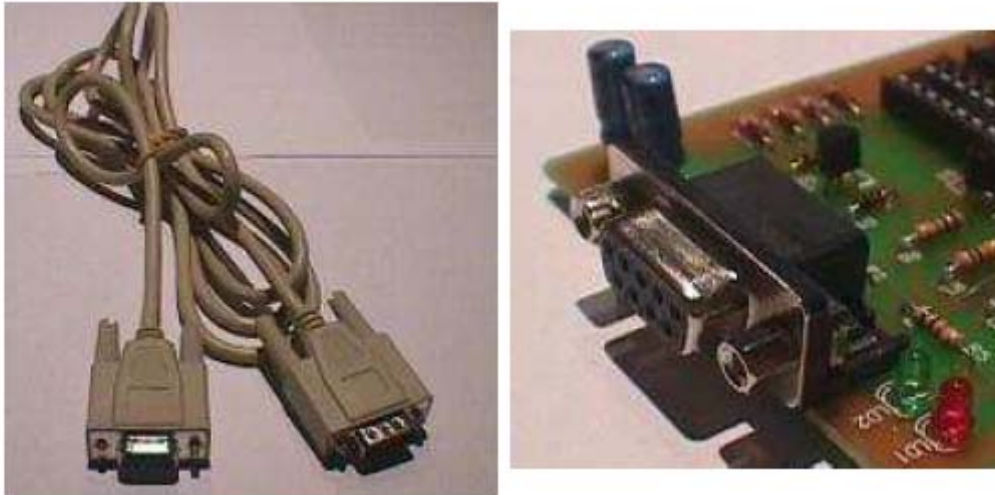


Figura 6.16 Esquema conexión max232

*Cable de conexión*

Para realizar la conexión entre el PC y nuestro circuito podemos usar diferentes alternativas. Una manera es utilizar un cable serie macho-hembra no cruzado, y en el circuito un conector hembra db9 para circuito impreso:



**Figura 6.17** Cable y programador conexión PC-microcontrolador

Cuando conectamos un microcontrolador al PC normalmente sólo usamos los pines TX, RX y GND, sin embargo en este tipo de cables se llevan los 9 pines.

## 1.2 Conclusión y Elección de Dispositivos

Tras estudiar las diferentes posibilidades y las características y necesidades del proyecto se eligen los dispositivos mas adecuados.

### *SENSORES*

Se descartan los sensores de ultrasonidos por que tienen un precio muy elevado y porque consumen demasiado, lo que implicaría la necesidad de añadir una batería extra lo incrementaría aun más el proyecto. Los sensores de infrarrojos son los que mejores características ofrecen, pero esta tampoco será la opción por la que nos inclinemos.

Se instalarán un par de Bumpers ya que tienen un coste muy Bajo.

### *MOTORES*

La placa MSx84 permite la utilización de dos motores de corriente continua (DC) o de uno paso a paso (PAP). En nuestro proyecto lo que se necesita es la primera opción, es decir, dos motores de corriente continua para el accionamiento de las ruedas motrices del robot.

Así se utilizará dos servomotores trucados para funcionar como motores DC, conectados a la placa MSx84.

## COMUNICACIÓN INALÁMBRICA

El módulo PROMI SD 202 es el más adecuado para el proyecto ya que son los que proporcionan mayor inmunidad al ruido y a las interferencias y un mayor alcance y fiabilidad.

### COMUNICACIÓN con PC

Finalmente para la comunicación con el pc se usará la interfaz RS232 ya que está muy extendida, tiene compatibilidad directa con microcontroladores y placas y existen múltiples librerías que facilitan enormemente la programación del PIC16F84A. Además los módulos bluetooth disponen de ésta inferfaz.

## 1.3 Diseño funcional del Robot

Tras recopilar la información sobre los diferentes dispositivos, sensores y conexiones con el PC se debe elegir el tipo de robot que se va a construir. Las posibilidades son muy amplias y según las decisiones que se tomen la construcción del robot será más o menos complicada.

### 1.3.1 Tipo de estructuras móviles

Lo primero que se tiene que elegir es el tipo modo de desplazamiento que tendrá el robot, encontramos, entre otras, las siguientes posibilidades:

#### *Robot con estructura vehículo con Ruedas*

Es la estructura más sencilla y práctica. El robot adoptaría la forma de vehículo terrestre. La forma de desplazarse también sería la mas sencilla, únicamente harían falta dos servomotores con dos ruedas que puedan girar sin limitación. Y una rueda loca para permitir los giros.

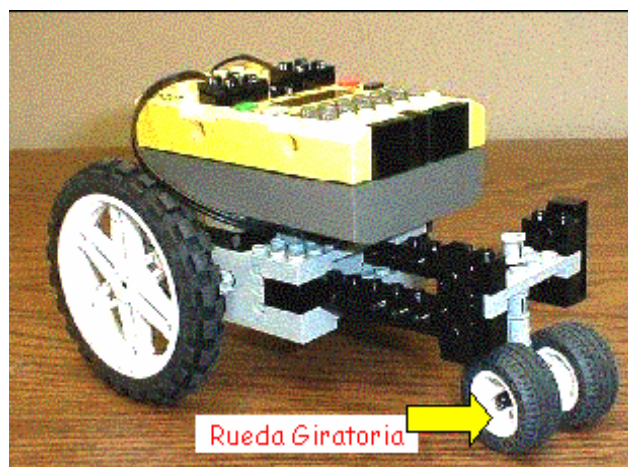
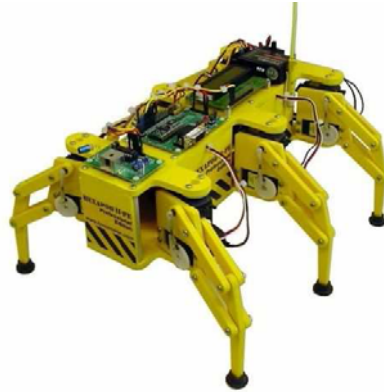


Figura 6.18 Robot estructura con ruedas

*Robot con estructura insecto con múltiples patas*

Es el siguiente paso en la dificultad. Ahora el robot tendría una estructura tipo insecto, y en lugar de las típicas ruedas dispondría de múltiples patas, cada una con un servomotor muchos mas complejos que requieren mayor control.



**Figura 6.19** Robot estructura con múltiples patas

*Robot con estructura humanoide*

Por último estarían los robots humanoides que tratan de imitar el andar de humano, para ello utilizan dos piernas articuladas con varios servomotores también y múltiples sensores para detectar velocidad, ángulo, inclinación, etc.



**Figura 6.20** Robot con estructura humana bípedo.

Se elige el primer tipo, robot con estructura tipo coche y ruedas motrices por ser el más sencillo, barato y cumple de sobra con los requerimientos del proyecto.

### 1.3.2 *Tamaño*

Otro apartado a tener en cuenta es el tamaño de nuestro prototipo. Cuanto más grande sea el robot, más piezas necesitará para su construcción, lo que a su vez lo hará más pesado. Esto es importante porque al ser más pesado se necesitará más energía para moverlo, y llegando a necesitar múltiples baterías para abastecer a todos los servomotores necesarios para moverlo con agilidad.

Por todas estas razones, se opta con un tamaño lo más reducido posible, aunque lo suficientemente grande como para que se pueda trabajar cómodamente con él y que pueda albergar todas las piezas requeridas para cumplir la funcionalidad esperada. Así el robot tendrá unas dimensiones aproximadas de 15 cm. de largo x 10 cm. de ancho x 12 cm. de alto aproximadamente.

#### *Forma y Materiales*

De la forma y los materiales se hablará con más detalle en los apartados siguientes, Optimización de espacio en el robot. Aunque se puede adelantar que tendrá forma de prisma rectangular y que se construirá con piezas mecano.

### 1.4 **Recopilar información Software:**

Para desarrollar el proyecto necesitamos varias herramientas software. Las tareas que se deben realizar y el software más adecuado para ello, están descritas en la siguiente tabla:

TAREA	SOFTWARE
Desarrollo del programa del PIC	MP-LAB IDE
Programación del PIC	IC-PROG
Testeo de la aplicación	HYPERTERMINAL
Programación de la interfaz Gráfica	NET-BEANS 5.0 y RealJ

**Tabla 6.2** Software desarrollo del proyecto

#### 1.4.1 *Programación comunicaciones PC-Puertos.*

El control del robot se gestionará a través de una aplicación gráfica diseñada y programada en lenguaje java. Este lenguaje no soporta directamente la programación de los puertos COM. Para ello se debe utilizar un paquete externo llamado javax.comm que se encarga de añadir a java esta funcionalidad. Toda la información relativa a este API se encuentra ampliamente descrita en el ANEXO 4: API DE JAVAX.COMM

**1.5 Diseño físico y construcción del robot:**

**1.5.1 Ubicación de componentes.**

La estructura del robot tendrá forma rectangular y con tres pisos. En primer lugar, en el chasis irán los servomotores y la rueda loca.

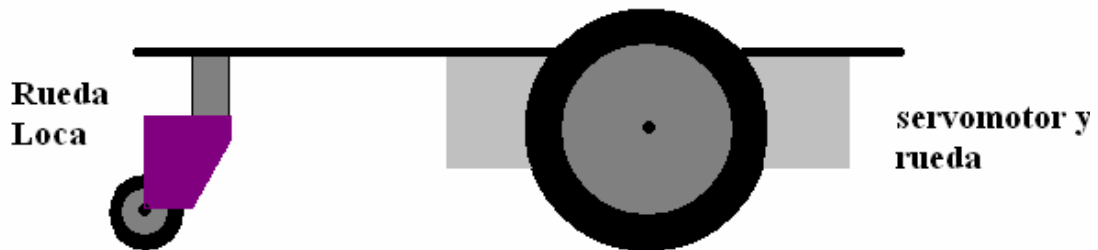


Figura 6.21 Fase 1 construcción estructura robot

Sobre el chasis iría alojada la batería, directamente sobre los servomotores que controlan las ruedas. La estructura partiría de aquí, de donde saldrían 4 piezas a modo de columnas, sobre las cuales se asentaría el primer piso.

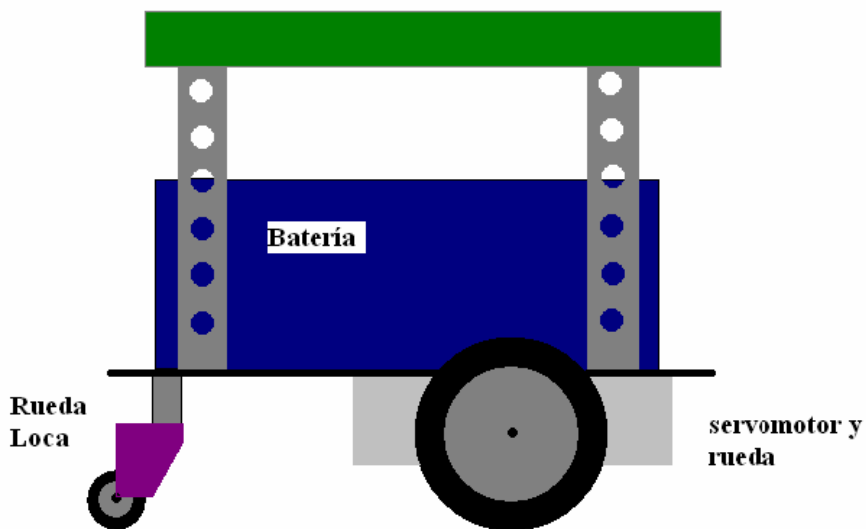


Figura 6.22 Fase 2 construcción estructura robot

Colocado en el lado izquierdo, en posición horizontal, iría colocado el módulo bluetooth.

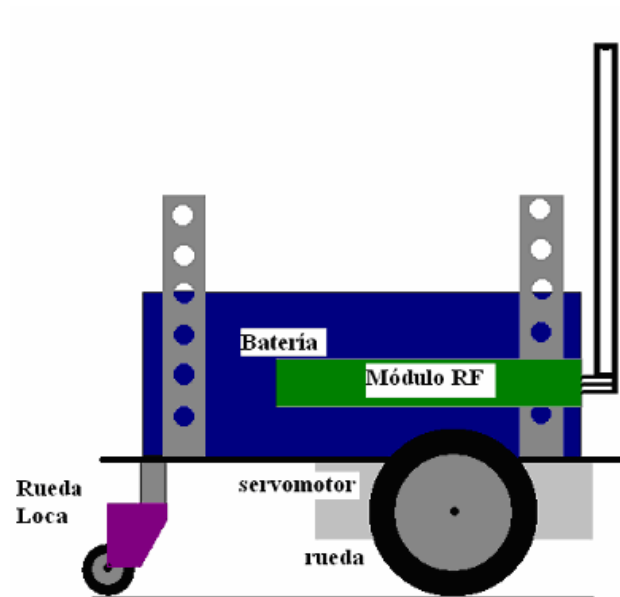


Figura 6.23 Fase 3 construcción estructura robot

En la parte superior, y más despejada, iría alojada la placa MSx84 facilitando así todo el cableado hacia los sensores, batería, módulo RF etc.

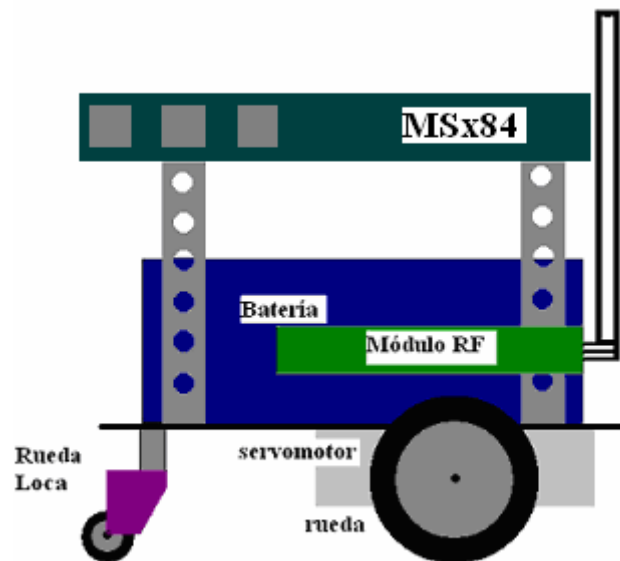


Figura 6.24 Fase 4 construcción estructura robot

En la parte frontal del microrobot irían situados los bumpers de choque y la micro-cámara. El MAX232 se colocará en la parte posterior.



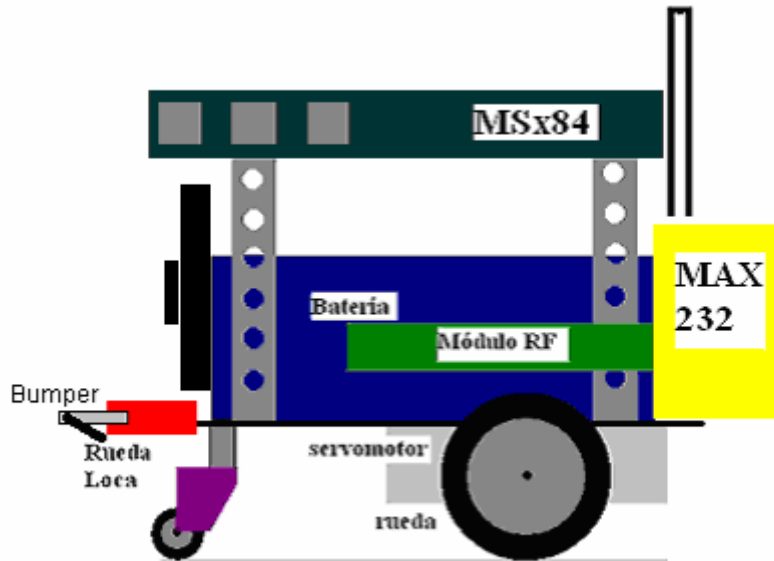


Figura 6.25 Fase 5 construcción estructura robot

Por último situaremos el modulador del canal 22 UHF en el lado derecho del robot

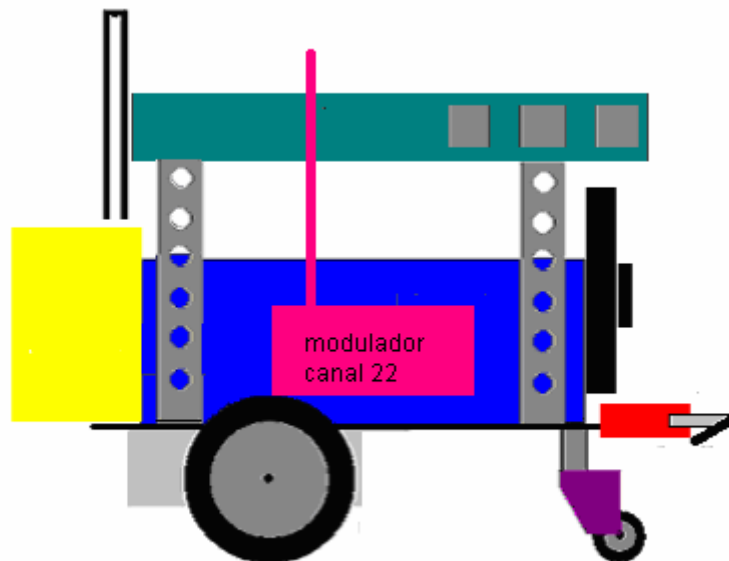


Figura 6.26 Fase 6 construcción estructura robot

### 1.5.2 Optimización de espacio en el robot.

Para optimizar el espacio en el robot y que todos los componentes se pudieran alojar en un espacio contenido y compacto se ha empleado una estructura rectangular con tres pisos. Esto facilita la disposición de los elementos unos encima de otros haciendo su conexión muy sencilla. Y un cableado claro y fácil de rectificar durante el proceso de construcción y diseño del robot.

Las piezas de mecano ofrecen grandes posibilidades de montaje, lo que hace posible ir situando los sensores de infrarrojos, Bumpers o cualquier dispositivo que se pudiera añadir después situándolos alrededor de la estructura principal.

La estructura rectangular permite situar los dispositivos que se van añadiendo en cualquier cara del prisma. Así podemos ir añadiendo pisos hacia arriba según vaya haciendo falta, situar sensores en la parte frontal de la estructura o situar el circuito MAX232 con todo su cableado en la parte posterior.

### ***1.5.3 Optimización del consumo.***

La optimización del consumo se ha tenido en cuenta desde el principio del proyecto y por ello se han ido eligiendo componentes que tuvieran un bajo consumo. Así por ejemplo se descartaron el uso de sensores de ultrasonidos por su elevado consumo, sustituyéndolos por unos bumpers con un consumo nulo.

Todo el robot irá alimentado con una única batería de plomo de 12 v que se recargará a través de un adaptador de corriente conectado directamente a la placa MSx84. Para alimentar la micro-cámara, se ha utilizado un rectificador 7805, para conseguir los 5V necesarios para su funcionamiento a partir de los 12 V proporcionados por la batería. La energía para alimentar el módulo bluetooth se ha extraído del pic-bus de la controladora MSx84. Por último la alimentación del modulador, la obtenemos de uno de los puertos destinados a la conexión de un sensor.

## **2 FASE 2**

### **2.1 Construcción del robot**

Una vez que está definidos los materiales a utilizar, la ubicación de los componentes y con los planos conexionado se procede a la construcción del robot. El proceso será el siguiente:

1. Se truncan los servomotores para eliminar el bloqueo y permitir una rotación completa de la rueda.
2. Se unen los servos usando las piezas meccano. Y se coloca la pieza que hará de chasis.
3. Se atornillan los servos con las ruedas a la base, se monta y añade la rueda loca y se colocan las cuatro varillas que harán de columnas para el piso superior.
4. Se coloca la batería, ya con cables de tensión para luego no tener que desmontar, ya que la batería quedará rodeada de piezas por todos lados y tendrá un difícil acceso para su manipulación.
5. Se colocan piezas sobre la base para conseguir que la batería no se mueva, estas piezas servirán después para fijar los sensores y el max232.

6. Se coloca el módulo bluetooth PROMI SD202.
7. Se colocan piezas auxiliares y sobre éstas la placa MSx84.
8. Se colocan los bumpers y el max232 de forma provisional.
9. Se cablea todo: batería, placa, servomotores, sensores, etc.
10. Se endereza la antena para que quede perpendicular.
11. Se fijan con los Bumpers, se añade retráctil en las conexiones, se fijan bien todos los componentes y se coloca de forma compacta el max232.
12. Colocamos la micro-cámara en la parte frontal de robot.
13. Conectamos el modulador a la micro-cámara y lo situamos en la parte derecha del robot.

## **2.2 Software Microcontrolador**

El desarrollo del código del PIC que controla en el robot está dividido en dos apartados: Flujogramas de la aplicación y el código en lenguaje ensamblador. Estos dos apartados están ampliamente desarrollados en el ANEXO 14: Software del Microcontrolador.

## **2.3 Software PC**

En las primeras fases del proyecto, el robot se controlará utilizando el teclado a través del hyperterminal de Windows, pues éste programa controla de forma inmediata los puertos COM así como los diferentes parámetros que influyen en la transmisión como tasa de transferencia, bits de parada, etc.

En la última fase del proyecto, la tarea del control remoto del robot se realizará a través de una aplicación en lenguaje java que mostrará una interfaz gráfica.

El código de este programa en java, así como las correspondientes aclaraciones se encuentran desarrolladas en el ANEXO 15: Software Aplicación Gráfica.

## **3 FASE 3**

Una vez que se ha realizado el montaje de las piezas y las conexiones de los cables se debe comprobar el funcionamiento del robot. Pero esta fase se dividirá en tres etapas donde se irá comprobando el funcionamiento del robot paso a paso.

Esto se hará así porque resulta mucho más sencillo localizar los posibles problemas que irán surgiendo y solucionarlos mucho más rápidamente.

Si no se hicieran estas etapas, y no funcionará el robot, no sabríamos si el problema esta en la programación del PIC, en los cables, en la alimentación, en algún

pieza defectuosa, etc. De este modo se van acotando los posibles problemas determinando que piezas funcionan perfectamente. El esquema general sería este:

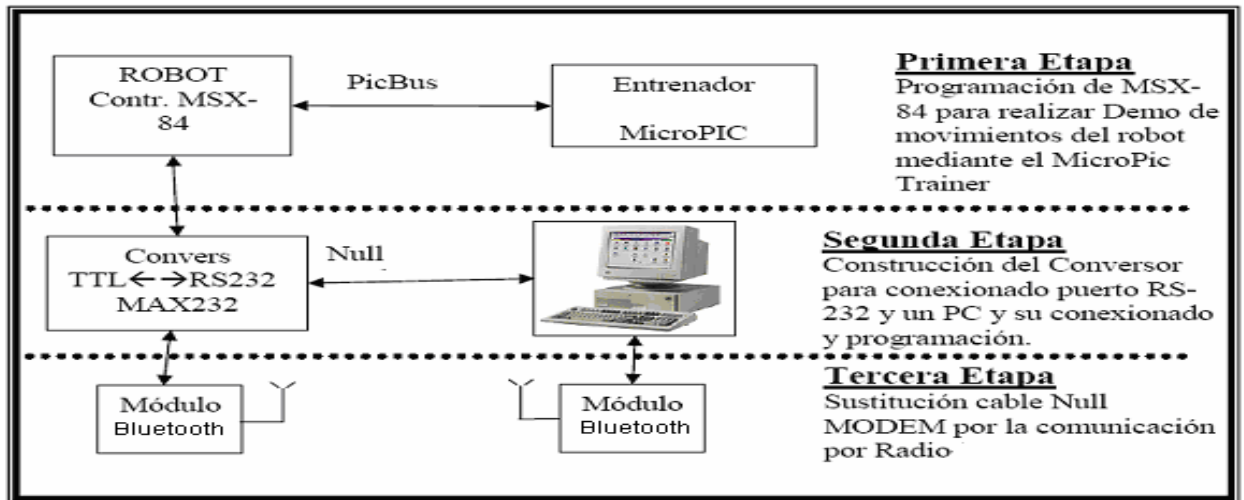


Figura 6.27 Descripción etapas desarrollo robot

### 3.1 Primera Etapa

Primero compilamos el software del PIC en el programa MPLAB, si la compilación es correcta generará un archivo con extensión .HEX que se usará para programar el PIC mediante un programador cargando ese archivo con el programa ICPROG. Ya estaría cargado el programa de control del PIC y ahora se pincharía el chip sobre el MICROPIC TRAINER.

El objetivo de esta primera prueba es comprobar que los comandos que enviamos desde el programa hyperterminal de Windows a través del teclado llegan al entrenador y son correctamente interpretados por el PIC encendiendo los leds adecuados. Así al pulsar en el teclado se debe mostrar como salida una combinación de 5 bits correspondientes a RA3-RA0 que posteriormente gobernarán los motores. En la siguiente tabla vemos las correspondencias:

Tecla pulsada	Combinación RA3-RA0
A	1110
S	1100
D	1101
W	1111
Espacio	0000

Tabla 6.3 Correspondencia tecla-combinación RA3-RA0

Problema:

Algunas de las teclas no producen la combinación que se esperaba

Solución

Tras reprogramar el software, y volver a repetir el proceso de complicación y programación se vuelve a repetir el test obteniendo ya el resultado esperado.

RESULTADO

Tras realizar la prueba se verifica que los datos llegan y que son interpretados correctamente por el PIC. Se confirma que el software está bien diseñado.

En la siguiente prueba en lugar de pinchar el chip sobre el entrenador, lo dejaremos en la placa MSx84, ahora conectamos el PC con hyperterminal al entrenador y el entrenador al robot. Es necesario hacerlo así porque el Entrenador dispone del circuito MAX232 necesario para la adaptación de las señales a niveles TTL. El objetivo ahora es que en el lugar de obtener la salida con los leds sean las ruedas del propio robot las que se muevan acorde con el comando mandado.

Ahora la salida esperada es la siguiente:

Tecla pulsada	Movimiento de las ruedas
A	Giro hacia la izquierda
S	Hacia delante
F	Hacia atrás
E	Giro hacia la derecha
Espacio	Parada

**Tabla 6.4** Correspondencia Tecla pulsada- movimiento ruedas

Problema

Las ruedas se mueven pero no de la forma esperada. Algunas de las teclas no producen movimiento o es un movimiento contrario al previamente programado.

Solución

Puesto que el experimento anterior función correctamente se descarta que el fallo esté en el software, tras revisar detenidamente el manual de la placa comprobamos que existe un conjunto de 9 DIP-SWITCH que permiten habilitar o no cada uno de los sensores de entrada así como las salidas a los motores. De esta forma se puede dejar libres aquellas líneas del PIC asociadas a los sensores y motores que no se vayan a utilizar. Tras colocarlos en las posiciones adecuadas obtenemos el movimiento de las ruedas acorde con la tecla pulsada.

RESULTADO

Una vez que esto funciona queda vigente que la placa y el PIC están bien programados y alimentados. Que el driver L293 funciona correctamente y que los servomotores están alimentados y funcionan correctamente.

### 3.2 Segunda Etapa

En esta segunda etapa el objetivo es prescindir del Micropic Trainer para ello se debe dotar al robot con el circuito MAX232 necesario para la conversión entre niveles TTL y niveles Rs232 (y viceversa).

Para ello se implementará el circuito necesario en una placa de silicio con los componentes citados en el apartado MAX232. De esta placa saldrán tres cables (TX, Tensión y Tierra) a los 3 pines correspondientes de los 26 del PIC-bus de la placa MSx84.

Se puede ver el esquema en el apartado Planos – Esquema de conexiones.

Durante el proceso se deben resolver varios problemas entre ellos la inexistencia de un interfaz macho-macho necesaria para conectar la placa de silicio con el módulo de radio frecuencia. Éste adaptador se tendrá que fabricar de forma artesanal soldando con estaño las patillas del conector macho. Una vez construido el dispositivo tendrá el siguiente aspecto:



Figura 6.28 Adaptador para max232

Tras instalarlo en el robot se procede a comprobar que todo funciona exactamente igual que funcionó al usar el chip max232 del Entrenador.

#### Problema

No funciona, las ruedas no se mueven.

#### Solución

Está claro que el problema se encuentra en el dispositivo max232 que se ha fabricado, tras repasar concienzudamente los planos se determina que el circuito está bien montado. Se cambia el chip max232 pensando que estaba dañado pero no era eso.

También se comprueba si los 3 pines del pic bus correspondían a TX, Tensión y tierra comprobando que el esquema del libro PIC16F84A desarrollo de proyectos estaba al revés.

Tras hacer ese cambio se comprueba que tampoco funcionaba. Finalmente el error estaba en la pieza macho-macho que se había fabricado soldando los pines uno a uno. Para que funcionase los pines de transmisión y retransmisión deben de cruzarse, una vez echo esto, todo funciona correctamente.

### RESULTADO

Ahora el robot funciona perfectamente dándole órdenes desde el teclado, ya sin necesidad de la presencia del entrenador. El siguiente paso es introducir los módulos bluetooth para que el robot funcione de forma inalámbrica.

## **3.3 Tercera Etapa**

Ahora hay que usar los dos módulos PROMI SD202 para que el robot reciba las ordenes vía bluetooth. En primer lugar se comprará que los módulos funcionan bien, para ello se empleará el software que se proporciona junto con los dispositivos. Conectando cada módulo a un PC y verificando que se puede transmitir texto entre ambos.

### RESULTADO

Ahora el robot se mueve perfectamente guiado de forma inalámbrica, con las órdenes que se le van mandando al pulsar las teclas correspondientes en el teclado del PC.

### **3.3.1 Puesta en marcha del robot**

Finalmente para la puesta en marcha del robot no hace falta ningún software especial, tan solo un programa tal como RealJ, que nos proporcione la máquina virtual de Java, para poder ejecutar, y en caso de que sea necesario, compilar nuestra aplicación.

Además y como paso previo, debemos de abrir el programa de la capturadora de televisión para sintonizar el canal 22 de UHF, quedando así preestablecido cuando ejecutemos la aplicación.

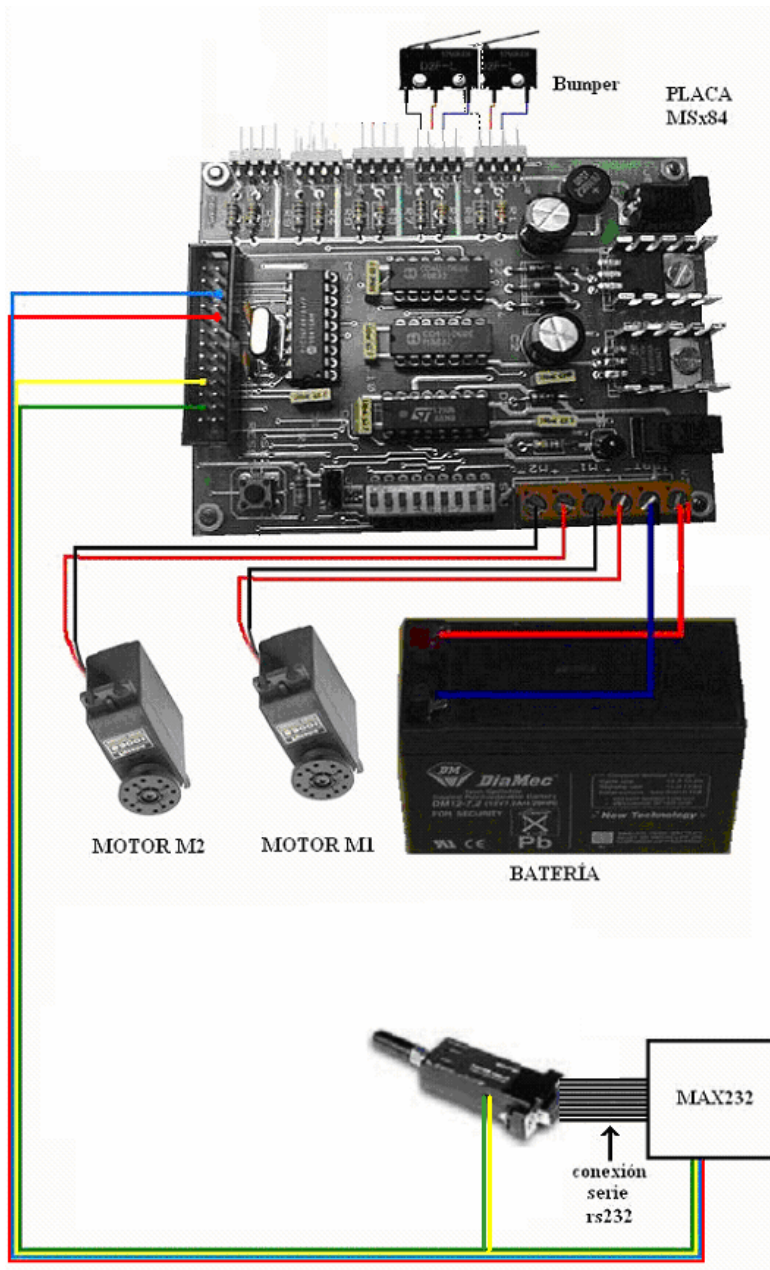
Como diremos en el apartado de pruebas, debido a la sensibilidad de la tarjeta capturadora de televisión, utilizaremos un televisor convencional para capturar las imágenes de la cámara. Tan sólo es necesario sintonizar el canal 22 de UHF, que el canal por el que emite el modulador instalado en el robot.



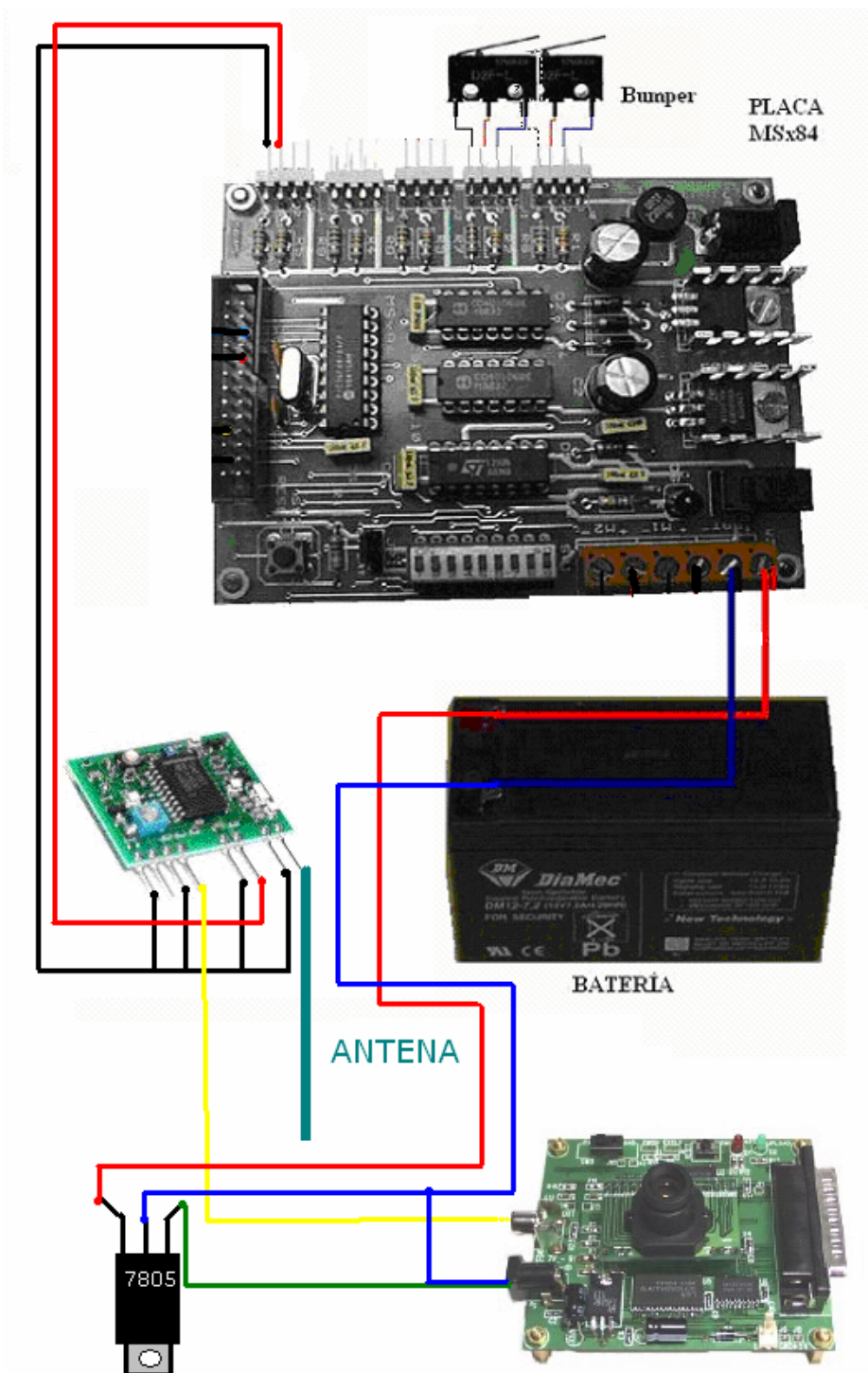


# PLANOS

## 1 ESQUEMA INSTALACIÓN SISTEMAS DE SENSADO, MOTORIZACIÓN Y COMUNICACIÓN



## 2 ESQUEMA INSTALACIÓN SISTEMA VISIÓN



## PRESUPUESTO

Componente	Precio Unidad	Unidades	Total
Tarjeta de control de motores MSX84	69,00 €	1	69,00 €
Bluetooth RS232 Interface Promi-SD	89,00 €	1	89,00 €
Batería de Plomo 12V 6A5 HR1224W-F	32,53 €	1	32,53 €
Cargador Batería plomo 6 Y 12V 0	11,40 €	1	11,40 €
Servo motor HITEC HS422	13,41 €	2	26,82 €
Sensor electromecánico (Bumper)	2,20 €	2	4,40 €
TV CHANNEL AUDIO-VIDEO MODULATOR	27 €	1	27 €
TARJETA CÁMARA ev38	55 €	1	55 €
TARJETA TV PINNACLE PCTV STEREO	59 €	1	59 €
MICROCAMARA OMNIVISION OV7120	48 €	1	48 €
Adaptador de niveles lógicos TTL a niveles RS232 MAX232	1,50 €	1	1,50 €
Placa de silicio y cables para el MAX232	5,00 €	1	5,00 €
Par rueda neopreno para robot 38 X 12 mm	4,18 €	2	8,36 €
Antena recta para 433 MHZ	6,25 €	1	6,25 €
KIT Piezas MECANO	60,00 €	1	60,00 €
Cables,tornillos y adaptadores para el robot	8,00 €	1	8,00 €
<b>PRECIO TOTAL</b>			<b>511,26 €</b>

TOTAL PRESUPUESTO: 511,26 €

El presente Presupuesto, asciende a la cantidad de QUINIENTOS ONCE CON VEINTISÉIS EUROS.

Cartagena, \_\_\_\_ de \_\_\_\_ de 2006

El proyectista.



# PLIEGO DE CONDICIONES

---

---

## 1 Introducción

El presente pliego de condiciones corresponde al proyecto Robot controlado desde ordenador vía Radio Bluetooth. La fabricación y programación de este robot se ejecutará de acuerdo al presente pliego de condiciones y demás documentos del proyecto.

## 2 Condiciones de los componentes

En primer lugar debe realizarse un pedido que nos provea de todos los componentes electrónicos y mecánicos necesarios para la realización del proyecto. Todos los componentes necesarios para la construcción del robot están especificados en el presupuesto.

Todos los componentes electrónicos se ajustarán a las condiciones que previenen los documentos que se integran en este proyecto. Ha de prestarse especial atención al estado de los pines de los microcontroladores y el buen estado de todos los demás componentes electrónicos y alimentarlos a la tensión adecuada ya que cualquier sobrealimentación puede llegar a dañar el circuito.

## 3 Condiciones de realización del robot

La construcción del robot se hará siguiendo los pasos detallados en la memoria y se utilizan los materiales citados en la misma. Los componentes serán colocados y soldados según se especifica en los planos.

## 4 Condiciones de software

El software que se usará durante el desarrollo del proyecto es el siguiente:

- MPLAB IDE: Este programa se utilizará para la edición y compilación de los programas.
- IC-PROG: Este se utilizará para la programar el microcontrolador a través de una cable serie.
- HYPERTERMINAL: Terminal de Windows que nos permite la comunicación con el robot en las primeras fases del proyecto. Configurando adecuadamente los parámetros de tasa de transferencia y el puerto, asignamos a cada tecla a, s, d, w, y espacio el correspondiente movimiento del robot. Mas tarde de esto se encargará una aplicación en java.

- RealJ : Programa para el diseño y compilación de programas en lenguaje Java. Se utilizará para el desarrollo de la aplicación con interfaz gráfica que controlará el robot.

# PRUEBAS, CONCLUSIONES Y TRABAJO FUTURO

---

---

## 1 Fase de pruebas.

Después de efectuar diversas pruebas parciales del funcionamiento de cada parte por separado, se procedió a hacer pruebas con el sistema completo integrado. En esta fase se comprueba el correcto funcionamiento del sistema implementado, verificándose que realice correctamente todas las tareas.

Se verifica que el micro-robot sea correctamente teleoperado por el usuario. El micro-robot debe corregir su trayectoria basándose tanto en la información que le proporcione el usuario a través de una interfase, como de la información proporcionada por el sistema de sensado, ya que el robot también debe corregir su trayectoria en función de la información proporcionada por el sistema de sensores electromecánicos.

### 1.1 Pruebas del control remoto y sistema de sensado.

La prueba principal a la que se somete el robot, es seguir el camino que le vaya marcando el usuario. Se ha comprobado visualmente que el robot no se pierde, es decir, obedece en todo momento las indicaciones del control remoto.

Para comprobar el alcance máximo del robot (el alcance máximo de los módulos bluetooth), realizamos distintas pruebas.

La primera prueba se realizó en el patio principal del cuartel de Antigones, prueba que se realizó para determinar el alcance máximo del mismo en espacios abiertos. Tras instalar el PC desde el que se iba a controlar el robot en un extremo del patio, se le dio la orden al mismo, de avanzar hacia delante, ordenándole también en intervalos cortos de tiempo que detuviera su marcha, para comprobar que efectivamente todavía obedecía a las ordenes del PC. Realizada esta prueba pudimos determinar que el alcance del robot en espacios abiertos era superior a 100 metros.

La segunda prueba realizada, se llevó a cabo en los sótanos del cuartel de Antigones, cuyas paredes son de hormigón armado de unos 20 cms. Desde un PC instalado en el laboratorio donde se ha llevado a cabo la ejecución del proyecto, fue guiándose el robot por los pasillos del sótano. Para ello contamos con la ayuda de un compañero de laboratorio, que por teléfono móvil fue guiándonos para no chocar con ningún objeto. El alcance máximo que se logró fue de unos 40 metros aproximadamente.



Figura 10.1 Descripción situación pruebas control remoto

En cuanto al alcance del modulador UHF incorporado en el robot, hay que destacar que la potencia de transmisión es tan pequeña que aunque la aplicación Java que hemos programado incorpora una ventana donde captura las imágenes enviadas por la cámara, hemos tenido que capturar las imágenes usando un televisor convencional, sintonizando en el mismo el canal 22 de UHF. Esto es debido a que mientras que la tensión de excitación de un televisor convencional es de unos pocos milivoltios, la tensión de excitación de una tarjeta capturadora de televisión, en este caso una Pinnacle pctx stereo, es mucho mayor. Por tanto, una vez realizada la prueba del alcance del modulador en una televisión convencional, llegamos a la conclusión que sin obstáculos el alcance para que la captura de la imagen fuera medianamente buena era de unos 15 metros, y si había cualquier tipo de obstáculo la distancia se reducía drásticamente.

Además de todas las pruebas anteriores, realizamos diversas pruebas del sistema de sensado, y comprobamos que tras chocar con obstáculos colocados intencionadamente, el robot cambiaba su trayectoria, además de informar de dicha colisión, enviando la correspondiente información a la aplicación Java programada.

## 2 Conclusiones.

Después de la realización de diferentes pruebas parciales de cada módulo y de pruebas globales del sistema integrado, se ha comprobado que el prototipo realizado satisface perfectamente los objetivos marcados en el presente proyecto:

- Diseño y construcción de un micro-robot móvil y de un sistema de control remoto que permita que el robot sea teleoperado por un usuario.



- Diseño e implementación de un sistema de enlace radio que permita comunicar una estación base (interfase con el usuario) con una plataforma móvil.
- Diseño e implementación de un sistema de sensado que proporcione autonomía para cambiar su trayectoria y posibilite que éste realice la función requerida (prueba de colisiones).

Las principales conclusiones a las que se ha llegado son las siguientes:

- Correcto funcionamiento del prototipo en modo teleoperado.
- El sistema de comunicaciones funciona correctamente, sin ningún tipo de interferencias, aunque existan obstáculos entre el emisor y el receptor.
- La plataforma móvil responde correctamente a las consignas del control remoto y a la información proporcionada por los sensores electromecánicos.
- El software desarrollado tanto para el microcontrolador, como para la aplicación gráfica es correcto. Hemos observado la facilidad de programación del microcontrolador 16f84A en lenguaje ensamblador, y de la potencia y portabilidad de Java.
- Por último, cabe decir que el prototipo es ampliable, pudiéndose añadir otro tipo de sensores complementarios, añadir motores para el control de la cámara, etc.



# BIBLIOGRAFÍA

---

---

- [1] Joseph L. Jones, Anita M. Flynn. “MOBILE ROBOTS, inspiration to implementation”, ed. Sales, 1993.
- [2] PIC16F87X Data Sheet 24/40-Pin 8-Bit CMOS FLASH Microcontrollers, Microchip Technology Inc.
- [3] M. H. Rashid. “Power electronics: circuits, devices and applications”, ed. Prentice-Hall, 1993.
- [4] Ramón Pallás Areny. “Sensores y acondicionadores de señal”, ed. Marcombo, 1994.
- [5] Microcontrolador PIC16F84. Desarrollo de proyectos. Ed. E. Palacios, F. Remiro. Rama. 2004.
- [6] Microcontroladores PIC: Diseño de aplicaciones. 3ª Ed. J.M. Angulo, I. Angulo. McGraw-Hill. 2003.
- [7] Manual de usuario controladora de motores y sensores MX-84.
- [8] Ángela Nombela Piqueras. “Desarrollo e implementación de un sistema sensorial y de comunicaciones para un robot móvil”, PFC Universidad Carlos III de Madrid, 1999.
- [9] Michael Tischer. “PC interno 2.0”, ed. Marcombo, 1995.
- [10] Manual de usuario controladora radiofrecuencia MSx84.
- [11] Manuel Martínez Alguacil. “ORCAD PCB II”, ed. Paraninfo, 1993.
- [12] DEITEL & DEITEL Java™ How to Program, 6/e (©2005)
- [13] [www.dalsemi.com](http://www.dalsemi.com)
- [14] <http://www.msebilbao.com/tienda/default.php>
- [15] <http://www.superrobotica.com/>
- [16] <http://www.planetaelectronico.com/tienda/default.php>
- [17] <http://www.learobotics.com/proyectos/cuadernos/ct1/ct1.html>
- [18] <http://www.javahispano.org/download.download.action?type=articles&id=3>
- [19] <http://www.x-robotics.com/sensores.htm#Bumper>
- [20] [http://www.lawebdelprogramador.com//news/mostrar\\_new.php?id=58&texto=Microcontroladores&n1=366639&n2=2&n3=1&n4=0&n5=0&n6=0&n7=0&n8=0&n9=0&n0=0](http://www.lawebdelprogramador.com//news/mostrar_new.php?id=58&texto=Microcontroladores&n1=366639&n2=2&n3=1&n4=0&n5=0&n6=0&n7=0&n8=0&n9=0&n0=0)
- [21] [http://www.scmstore.com/robotic/biorobots/zona\\_robotologica.htm](http://www.scmstore.com/robotic/biorobots/zona_robotologica.htm)
- [22] <http://www.geocities.com/ikerbot/index.html>
- [23] <http://www.itapizaco.edu.mx/paginas/JavaTut/froufe/parte19/cap19-3.html>
- [24] [http://www.geocities.com/victorhugo\\_dira/Captura\\_de\\_Video.htm](http://www.geocities.com/victorhugo_dira/Captura_de_Video.htm)



Cartagena,

Fdo. : Matías Pérez Ruiz  
Ingeniero Técnico de Telecomunicaciones Esp. Telemática