



UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO

FACULTAD DE ESTUDIOS SUPERIORES
ARAGÓN

GENERALIDADES DE LOS MICROCONTROLADORES Y SU APLICACIÓN EN CIRCUITOS Y
SISTEMAS DIGITALES

T E S I S

QUE PARA OBTENER EL TÍTULO DE
INGENIERO MECÁNICO ELECTRICISTA
P R E S E N T A :
DANIEL FALCÓN ECHEVERRÍA

ASESOR:
ING. ADRIÁN PAREDES ROMERO

ESTADO DE MÉXICO

2007





Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

Facultad de Estudios Superiores Aragón

DIRECCIÓN

DANIEL FALCON ECHEVERRIA
Presente

Con fundamento en el punto 6 y siguientes, del Reglamento para Exámenes Profesionales en esta Escuela, y toda vez que la documentación presentada por usted reúne los requisitos que establece el precitado Reglamento; me permito comunicarle que ha sido aprobado su tema de tesis y asesor.

TÍTULO:

"GENERALIDADES DE LOS MICROCONTROLADORES
Y SU APLICACIÓN EN CIRCUITOS Y SISTEMAS DIGITALES"

ASESOR: Ing. ADRIÁN PAREDES ROMERO

Aprovecho la ocasión para reiterarle mi distinguida consideración.

Atentamente
"POR MI RAZA HABLARÁ EL ESPÍRITU"

San Juan de Aragón, México, 21 de junio de 2006.

LA DIRECTORA

ARQ. LILIA TURCOTT GONZÁLEZ



C p Secretaria Académica
C p Jefatura de Carrera de Ingeniería Mecánica Eléctrica
C p Asesor de Tesis

LTG/AIR/csm



UNIVERSIDAD NACIONAL
AVENIDA DE
MEXICO

FACULTAD DE ESTUDIOS SUPERIORES
ARAGÓN UNAM

JEFATURA DE CARRERA DE
INGENIERÍA MECÁNICA ELÉCTRICA

OFICIO: FESAR/JAME/0737/2006.

LIC. ALBERTO IBARRA ROSAS
SECRETARIO ACADÉMICO
PRESENTE

FACULTAD DE ESTUDIOS SUPERIORES
ARAGON U. N. A. M.
SECRETARIA ACADEMICA
- 8 NOV 2006
RECIBIDO

Por este conducto me permito relacionar los nombres de los Profesores que sugiero integren el Sínoo del Examen Profesional del alumno: **C. FALCÓN ECHEVERRÍA DANIEL**, con Número de Cuenta: **09613912-7** con el tema: **"GENERALIDADES DE LOS MICROCONTROLADORES Y SU APLICACIÓN EN CIRCUITOS Y SISTEMAS DIGITALES."**

PRESIDENTE:	ING. JESÚS NÚÑEZ VALADEZ	ABRIL	77
VOCAL:	ING. FORTUNATO CERECEDO HERNÁNDEZ	JULIO	85
SECRETARIO:	ING. ADRIÁN PAREDES ROMERO	MAYO	90
SUPLENTE:	ING. JOSÉ LUÍS GARCÍA ESPINOSA	AGOSTO	98
SUPLENTE:	ING. RAMÓN PATIÑO RODRÍGUEZ	AGOSTO	03

Quiero subrayar que el Director de Trabajo de Titulación es el **ING. ADRIÁN PAREDES ROMERO**, quien está incluido basándose en lo que reza el Reglamento de Exámenes Profesionales de ésta Facultad.

Atentamente
"POR MI RAZA HABLARÁ EL ESPÍRITU"

Bosques de Aragón, Estado de México, 31 de octubre de 2006

EL JEFE DE CARRERA

M. en I. ULISES MERCADO VALENZUELA.



C.c.p.- Lic. Ma. Teresa Luna Sánchez.- Jefa del Depto. de Servicios Escolares.
C.c.p.- Ing. Adrián Paredes Romero. - Asesor de tesis.
C.c.p.- Alumno.
UMV/acl.

FACULTAD DE ESTUDIOS SUPERIORES
ARAGON
RECIBIDO
08 NOV 2006
DEPTO. SERVICIOS ESCOLARES
RECIBIDO

A:

EVA ECHEVARRIA PEREZ

Nuevamente vuelvo a ti como la noche regresa al ocultarse el Sol, porque no importa el lugar y la distancia siempre estas en mi pensamiento, porque el destino decidió que fuera yo parte de ti y tú parte de mí.

Porque no han importado tus sueños e ilusiones siempre nuestro bienestar fue primero; cuidaste cada uno de nuestros pasos, nuestros sueños en la enfermedad, te mantuviste despierta hasta que el último de tus hijos estuviera descansando en su cama. Gracias!!

Gracias porque desde el primer momento que decidiste ser madre, tomaste en serio cada uno de los consejos, las responsabilidades y los retos que implicaba tu posición como trabajadora independiente.

Gracias porque en ese afán de darnos lo necesario dejaste en un pequeño espacio tu vida, y así lo has hecho con cada uno de tus órganos.

Gracias por tus besos, por tus abrazos.

Gracias por tus consejos, por tus comentarios.

Gracias por tu abrigo, por los alimentos.

Gracias por tu comprensión, por tu apoyo.

Gracias por abrir las alas y dejar que también yo las abra.

Gracias por la VIDA.

Gracias MAMÁ!!

AGRADECIMIENTOS

A MI MADRE

Por ser el pilar que sostiene mi vida, por todas tus noches en vela, por tu apoyo, y por el entusiasmo que me das para seguir adelante.

JORGE TÉLLEZ +

A la memoria de un gran amigo que caminó conmigo durante muchos años proporcionándome enseñanza en cualquier momento. Donde estés, sigue iluminando mi camino.

PATRICIA JUÁREZ

Por todo tu cariño en mis momentos difíciles. Gran persona a la que respeto y admiro.

VICTOR CRUZ PAVÓN

Por creer en mí e involucrarme en nuevos proyectos que me aportaron gran enseñanza y entusiasmo cuando más lo necesitaba.

A MI FAMILIA

Por compartir conmigo momentos de alegría, tristeza y diversión.

Porque sé que en todo momento puedo contar con ustedes incondicionalmente.

ÍNDICE.

Pensamiento	4
Agradecimientos	5
Índice	6
Introducción	8
Objetivo General	10
Objetivos Particulares	10
Capítulo I.- <u>CIRCUITOS INCRUSTADOS, (EMBEDDED CIRCUITS)</u>	11
I.1.- Introducción	11
I.1.1.- Componentes de un Sistema Integrado	12
I.1.2.- Microprocesador y Sistemas Embebidos	12
I.1.3.- Arquitecturas de Computadoras más Empleadas	14
I.1.4.- Aplicaciones de un Sistema Embebido	15
I.1.5.- Ventajas de un Sistema Embebido sobre las Soluciones Industriales Tradicionales	16
Capítulo II.- <u>TEORÍA DE LOS MICROCONTROLADORES, (PIC)</u>	21
II.1.- Introducción	21
II.2.- Contenido Típico de un Microcontrolador	22
II.3.- Las Ventajas y Desventajas de los Microcontroladores	23
II.4.- Presentación de un Microcontrolador: El PIC16X84	24
II.4.1.- Descripción General	24
II.4.2.- Aspecto Externo	25
II.4.3.- La Frecuencia de Funcionamiento: El Reloj	26
II.4.3.1.- Tipos de Osciladores	26
II.4.4.- Reinicialización o RESET	27
II.5.- Arquitectura de los Microcontroladores PIC16X84	27
II.5.1.- Memoria de Programa	28
II.5.2.- Memoria de Datos RAM	29
II.5.3.- Direccionamiento de la Memoria de Datos	30
II.5.4.- El Registro de Estado	31
II.5.5.- Temporizador/Contador TMR0	32
II.5.6.- El Registro OPTION	34
II.5.7.- El Perro Guardián (WDT)	35
II.5.8.- Las Puertas de E/S	36
II.5.8.1.- La Puerta A	36
II.5.8.2.- La Puerta B	38
II.6.- La Palabra de CONFIGURACIÓN	39
II.7.- La Memoria EEPROM de Datos	39
II.8.- Interrupciones	40
II.8.1.- Causas de Interrupción	41
II.8.2.- El Registro de Control de Interrupciones INTCON	41
II.8.3.- Interrupción Externa INT	42
II.8.4.- Interrupción por Desbordamiento del TMR0	42
Capítulo III.- <u>PROGRAMACIÓN DE LOS MICROCONTROLADORES, PIC</u>	43
III.1.- Introducción	43
III.2.- Arquitectura del µPIC Trainer	46
III.2.1.- Componentes del µPIC Trainer	50
III.2.2.- El Módulo de Visualización LCD	52
III.2.3.- Software Recomendado	56

Capítulo IV.- APLICACIONES DE LOS MICROCONTROLADORES	57
IV.1.- introducción	57
IV.2.- Aplicaciones de los Microcontroladores	57
IV.2.1.- Domótica: Comunicaciones por Red Eléctrica	57
IV.2.2.- Mando a Distancia con Mensajes de Voz para Electrodomésticos	62
IV.2.3.- Plataforma para el Desarrollo de Sensores Inteligentes y Sistemas Microprocesados	67
IV.2.4.- Aplicaciones y Líneas de Investigación de los Microcontroladores	73
Conclusiones	81
Anexo: “Glosario de Términos”	82
Bibliografía	84

INTRODUCCIÓN.

El Microcontrolador es quizá, el componente electrónico más versátil que existe, sus aplicaciones están limitadas únicamente por la imaginación. Cada día es más frecuente encontrar aparatos que los utilicen como elemento de control; en el hogar por ejemplo, las cafeteras eléctricas, los hornos de microondas, las TV y las videograbadoras (entre otros), son tan sólo una pequeña muestra.

En el sector de las comunicaciones también se encuentran los Microcontroladores, los teléfonos convencionales, los teléfonos celulares, los conmutadores y las centrales telefónicas hacen parte de su campo de acción. Los automóviles tampoco se escapan de esta tendencia innovadora, en un vehículo de lujo, se pueden encontrar hasta 50 microcontroladores, los cuales desempeñan labores como el control de encendido, en los frenos ABS, la posición de los asientos, la radio, la temperatura individual de cada pasajero, etcétera.

Pero existen muchas áreas a las que esta innovación tecnológica no ha sido aplicada, de ahí, la importancia de conocer y aprender a manejar estos dispositivos. Un correcto uso y aplicación de los Microcontroladores son herramientas de vital importancia para enfrentar el reto que impone la creciente necesidad de modernización tecnológica, y para satisfacer un mercado que requiere personal idóneo en el área de diseño y desarrollo. Pero la mejor, y quizá, la única manera de adquirir estos conocimientos es realizando prácticas y aplicaciones reales, con las cuales, el Diseñador desarrollará la habilidad necesaria.

Entre un centenar de fabricantes de Microcontroladores que existen en el Mundo, es muy difícil seleccionar “el mejor”. En realidad, el mejor no existe, porque en cada aplicación son sus características específicas las que determinan el más conveniente.

Los Microcontroladores tienen “ángel” y una gran aceptación entre la comunidad de Ingenieros y Diseñadores de Sistemas que trabajan con PIC.

¿Cuánto durará este éxito? El tiempo que transcurra hasta que salga al mercado otro producto que mejore la imagen que actualmente se tienen de los PIC.

Es importante especificar en esta Introducción, qué es un Sistema Embebido: Un sistema embebido, también conocido como sistema empotrado, es una combinación de hardware, software y, eventualmente, componentes mecánicos diseñados para realizar una función específica.

Un **Sistema Embebido** puede o no ser de tiempo real dependiendo de los requerimientos específicos de la aplicación que se quiere implementar. De igual forma, el sistema será rápido si y sólo si se requiere de esta rapidez de acuerdo a la definición específica del sistema.

El hardware lo constituye un microcontrolador o un procesador de señales digitales (DSP), que es el corazón del sistema y de otros componentes electrónicos de acuerdo a la aplicación que se está implementando. Dentro de estos otros elementos de hardware se puede contar con componentes digitales (circuito digital que cumple una función específica sin tener su propio software de control), componentes analógicos (circuito analógico que cumple una función específica sin tener su propio software de control), bloques de lógica combinatoria y componentes en general (que realizan una función específica y que tienen su propio software de control). El tipo y la cantidad de componentes de hardware de un sistema dependen exclusivamente de la aplicación que se está implementando.

El software del sistema es el encargado de hacer que los distintos componentes de éste, realicen las funciones en los tiempos definidos y de acuerdo a sus requerimientos. Este software está incorporado en el microcontrolador o DSP e interactúa con los otros componentes a través de buses de datos, buses de control, sistemas DMA y puertos de control.

Los componentes mecánicos o de otra índole no son parte de este análisis por lo que no se referirá este trabajo a ellos. Por otra parte, sí se hará mención a la metodología que debe usarse para diseñar un sistema embebido. Este es un punto importantísimo para asegurarse del éxito del sistema que se está implementado y que éste cumpla con las expectativas funcionales y financieras del mercado.

Las etapas que deben ser consideradas en el diseño e implementación de un sistema embebido son las siguientes: definición de requerimientos del sistema, partición del sistema (definición de lo que se hará en hardware y software), definición del hardware, definición del software, implementación de modelos de simulación de los distintos componentes, simulación del sistema, diseño e implementación de hardware, diseño e implementación de software, pruebas de hardware, pruebas de software, integración y pruebas del sistema.

La primera etapa es vital para el éxito del sistema, ya que es en ella donde se define lo que hará el sistema que estamos implementando. La participación del área comercial y de marketing de la empresa es vital en este nivel. Un sistema que se define sin considerar los requerimientos funcionales y financieros del mercado tiene una probabilidad ínfima de éxito. Si el sistema está siendo definido para un cliente en particular, su participación activa es requerida en esta etapa. Una vez concluida, se genera un documento con los requerimientos del sistema, el cual debe ser aceptado por todas las partes involucradas y que es la base de todo desarrollo futuro.

OBJETIVO GENERAL.

Establecer las generalidades de los Microcontroladores y su Aplicación en Circuitos y Sistemas Digitales.

OBJETIVOS PARTICULARES.

- 1.- Establecer las características de los Circuitos Incrustados (*“Embedded Circuits”*).
- 2.- Establecer la Teoría de los Microcontroladores PIC.
- 3.- Establecer la Programación de los Microcontroladores PIC.
- 4.- Establecer las Aplicaciones de los Microcontroladores PIC en Circuitos y Sistemas Digitales.

CAPÍTULO I.

CIRCUITOS INCRUSTADOS, (EMBEDDED CIRCUITS).

1.1.- Introducción.

Como menciona [MBPES, Cap. I. Sec. 1], se entiende por sistemas embebidos a una combinación de hardware y software de computadora, sumado tal vez a algunas piezas mecánicas o de otro tipo, diseñado para tener una función específica. Un buen ejemplo de esto puede ser un horno microondas o un videogradora. Es común el uso de estos dispositivos pero pocos se dan cuenta que hay un procesador y un programa ejecutándose que les permite funcionar.

Esto ofrece un contraste con la computadora personal, que si bien también esta formada por una combinación de hardware y software mas algunas piezas mecánicas (discos rígidos, por ejemplo). Sin embargo la computadora personal no esta diseñada para un uso específico. Sino que es posible darle muchos usos diferentes. Por eso mucha gente la llama "*Computadora de propósito general*" para dejar clara esta diferencia.

Muchas veces un sistema embebido es un componente de un sistema mucho más grande, como por ejemplo los sistemas de frenos o el sistema de inyección de combustible, en automóviles actuales son sistemas embebidos.

Esta combinación de software y hardware puede ser reemplazada en muchos casos por un circuito integrado que realice la misma tarea. Pero una de las ventajas de los sistemas embebidos es su flexibilidad. Ya que a la hora de realizar alguna modificación resulta mucho más sencillo modificar una línea de código al software del sistema embebido que reemplazar todo el circuito integrado.

Un uso muy común de los sistemas embebidos es en los *sistemas de tiempo real*, entendiéndose por sistemas en tiempo real a aquellos sistemas en los que el control del tiempo es vital para el correcto funcionamiento. Los sistemas en tiempo real necesitan realizar ciertas operaciones o cálculos en un límite de tiempo. Donde ese limite de tiempo resulta crucial. Un ejemplo claro de un sistema de tiempo real es el control de tráfico aéreo.

Un **Sistema Embebido** (a veces traducido del inglés como *embebido*, *empotrado* o *incrustado*) es un sistema informático de uso específico construido dentro de un dispositivo mayor. Los sistemas integrados se utilizan para usos muy diferentes de los usos generales para los que se emplea un ordenador personal. En un sistema integrado la mayoría de los componentes se encuentran incluidos en la placa base (*motherboard*) (la tarjeta de vídeo, audio, módem, etcétera).

Dos de las diferencias principales son el precio y el consumo. Puesto que los sistemas integrados se pueden fabricar por decenas de millares o por millones de unidades, una de las principales preocupaciones es reducir los costos. Los sistemas integrados suelen usar un procesador relativamente pequeño y una memoria pequeña para reducir los costes. Se enfrentan, sobre todo, al problema de que un fallo en un elemento implica la necesidad de reparar la placa íntegra.

Lentitud no significa que vayan a la velocidad del reloj. En general, se suele simplificar toda la arquitectura del ordenador o computadora para reducir los costos. Por ejemplo, los sistemas integrados emplean a menudo periféricos controlados por interfaces síncronos en serie, que son de diez a cientos de veces más lentos que los periféricos de un ordenador o computadora personal normal. Los primeros equipos integrados que se desarrollaron fueron elaborados por IBM en los años 80. Los programas de sistemas integrados se enfrentan normalmente a problemas de tiempo real.

1.1.1.- Componentes de un Sistema Integrado.

En la parte central se encuentra el microprocesador, microcontrolador, DSP, etcétera. Es decir la CPU o unidad que aporta “*inteligencia*” al sistema. Según el sistema puede incluir memoria interna o externa, un micro con arquitectura específica según requisitos.

La comunicación adquiere gran importancia en los sistemas integrados. Lo normal es que el sistema pueda comunicarse mediante interfaces estándar de cable o inalámbricas. Así un SI normalmente incorporará puertos de comunicaciones del tipo RS232, RS485, SPI, I²C, CAN, USB, IP, WiFi, GSM, GPRS, DSRC, etcétera.

El subsistema de presentación tipo suele ser una pantalla gráfica, táctil, LCD, alfanumérico, etcétera. Denominamos actuadores a los posibles elementos electrónicos que el sistema se encarga de controlar. Puede ser un motor eléctrico, un conmutador tipo relé etcétera. El más habitual puede ser una salida de señal PWM para control de la velocidad en motores de corriente continua.

El módulo de E/S analógicas y digitales suele emplearse para digitalizar señales analógicas procedentes de sensores, activar diodos tipo LED, reconocer el estado abierto cerrado de un conmutador o pulsador, etcétera.

El módulo de reloj es el encargado de generar las diferentes señales de reloj a partir de un único oscilador principal. El tipo de oscilador es importante por varios aspectos: por la frecuencia necesaria, por la estabilidad necesaria y por el consumo de corriente requerido. El oscilador con mejores características en cuanto a estabilidad y coste son los basados en resonador de cristal de cuarzo, mientras que los que requieren menor consumo son los RC. Mediante sistemas PLL se obtienen otras frecuencias con la misma estabilidad que el oscilador patrón.

El módulo de energía se encarga de generar las diferentes tensiones y corrientes necesarias para alimentar los diferentes circuitos del SE. Usualmente se trabaja con un rango de posibles tensiones de entrada que mediante convertidores AC/DC o DC/AC se obtienen las diferentes tensiones necesarias para alimentar los diversos componentes activos del circuito.

Además de los convertidores AC/DC y DC/AC, otros módulos típicos son: filtros, circuitos integrados supervisores de alimentación, etcétera. El consumo de energía puede ser determinante en el desarrollo de algunos sistemas integrados que necesariamente se alimentan con baterías y es imposible su sustitución, con lo que la vida del SE suele ser vida de las baterías.

1.1.2.- Microprocesador y Sistemas Embebidos.

Un Microprocesador es una implementación en forma de circuito integrado (IC) de la Unidad Central de Proceso CPU de un ordenador. Frecuentemente nos referimos a un microprocesador como simplemente “CPU”, y la parte de un sistema que contiene al microprocesador se denomina subsistema de CPU. Los microprocesadores varían en consumo de potencia, complejidad y coste. Los hay de unos pocos miles de transistores y con coste inferior a 2 euros (en producción masiva) hasta de más de cinco millones de transistores que cuestan más de 600 euros.

Los subsistemas de entrada/salida y memoria pueden ser combinados con un subsistema de CPU para formar un ordenador o sistema integrado completo. Estos subsistemas se interconectan mediante los buses de sistema (formados a su vez por el bus de control, el bus de direcciones y el bus de datos).

El subsistema de entrada acepta datos del exterior para ser procesados mientras que el subsistema de salida transfiere los resultados hacia el exterior. Lo más habitual es que haya varios subsistemas de entrada y varios de salida. A estos subsistemas se les reconoce habitualmente como periféricos de E/S.

El subsistema de memoria almacena las instrucciones que controlan el funcionamiento del sistema. Estas instrucciones comprenden el programa que ejecuta el sistema. La memoria también almacena varios tipos de datos: datos de entrada que aún no han sido procesados, resultados intermedios del proceso y resultados finales en espera de salida al exterior.

Es importante darse cuenta de que los subsistemas estructuran a un sistema según funcionalidades. La subdivisión física de un sistema, en términos de circuitos integrados o placas de circuito impreso (PCBs) puede y es normalmente diferente. Un solo circuito integrado (IC) puede proporcionar múltiples funciones, tales como memoria y entrada/salida.

Un Microcontrolador (MCU) es un IC que incluye una CPU, memoria y circuitos de E/S. Entre los subsistemas de E/S que incluyen los microcontroladores se encuentran los temporizadores, los Convertidores Analógico a Digital (ADC) y de Digital a Analógico (DAC), y los canales de comunicaciones serie. Estos subsistemas de E/S se suelen optimizar para aplicaciones específicas (por ejemplo audio, video, procesos industriales, comunicaciones, etcétera).

Hay que señalar que las líneas reales de distinción entre microprocesador, microcontrolador y microcomputador en un solo chip están difusas, y se denominan en ocasiones de manera indistinta unos y otros.

En general, un Sistema Embebido (SE), consiste en un sistema con microprocesador cuyo hardware y software están específicamente diseñados y optimizados para resolver un problema concreto eficientemente. Normalmente un SE interactúa continuamente con el entorno para vigilar o controlar algún proceso mediante una serie de sensores. Su hardware se diseña normalmente a nivel de chips, o de interconexión de PCB, buscando la mínima circuitería y el menor tamaño para una aplicación particular. Otra alternativa consiste en el diseño a nivel de PCB consistente en el ensamblado de placas con microprocesadores comerciales que responden normalmente a un estándar como el PC-104 (placas de tamaño concreto que se interconectan entre sí "apilándolas" unas sobre otras, cada una de ellas con una funcionalidad específica dentro del objetivo global que tenga el SE). Esta última solución acelera el tiempo de diseño pero no optimiza ni el tamaño del sistema ni el número de componentes utilizados ni el coste unitario.

En general, un sistema embebido simple contará con un microprocesador, memoria, unos pocos periféricos de E/S y un programa dedicado a una aplicación concreta almacenado permanentemente en la memoria. El término embebido o empotrado hace referencia al hecho de que el microcomputador está encerrado o instalado dentro de un sistema mayor y su existencia como microcomputador puede no ser aparente. Un usuario no técnico de un sistema embebido puede no ser consciente de que está usando un sistema computador. En algunos hogares las personas, que no tienen por qué ser usuarias de un ordenador personal estándar (PC), utilizan del orden de diez o más sistemas embebidos cada día.

Los microcomputadores embebidos en estos sistemas controlan electrodomésticos tales como: televisores, videos, lavadoras, alarmas, teléfonos inalámbricos, etcétera Incluso un PC tiene microcomputadores embebidos en el monitor, impresora, y periféricos en general, adicionales a la CPU del propio PC. Un automóvil puede tener hasta un centenar de microprocesadores y microcontroladores que controlan cosas como la ignición, transmisión, dirección asistida, frenos antibloqueo (ABS), control de la tracción, etcétera.

Los sistemas embebidos se caracterizan normalmente por la necesidad de dispositivos de E/S especiales. Cuando se opta por diseñar el sistema embebido partiendo de una placa con microcomputador también es necesario comprar o diseñar placas de E/S adicionales para cumplir con los requisitos de la aplicación concreta.

Muchos sistemas embebidos son sistemas de tiempo real. Un sistema de tiempo real debe responder, dentro de un intervalo restringido de tiempo, a eventos externos mediante la ejecución de la tarea asociada con cada evento. Los sistemas de tiempo real se pueden caracterizar como blandos o duros.

Si un sistema de tiempo real blando no cumple con sus restricciones de tiempo, simplemente se degrada el rendimiento del sistema, pero si el sistema es de tiempo real duro y no cumple con sus restricciones de tiempo, el sistema fallará. Este fallo puede tener posiblemente consecuencias catastróficas.

Un sistema embebido complejo puede utilizar un Sistema Operativo como apoyo para la ejecución de sus programas, sobre todo cuando se requiere la ejecución simultánea de los mismos. Cuando se utiliza un sistema operativo lo más probable es que se tenga que tratar de un sistema operativo en tiempo real (RTOS), que es un sistema operativo diseñado y optimizado para manejar fuertes restricciones de tiempo asociadas con eventos en aplicaciones de tiempo real. En una aplicación de tiempo real compleja la utilización de un RTOS multitarea puede simplificar el desarrollo del software.

1.1.3.- Arquitecturas de Computadores más Empleadas.

Un PC embebido posee una arquitectura semejante a la de un PC. Brevemente éstos son los elementos básicos:

- ◆ **Microprocesador.**- Es el encargado de realizar las operaciones de cálculo principales del sistema. Ejecuta código para realizar una determinada tarea y dirige el funcionamiento de los demás elementos que le rodean, a modo de director de una orquesta.
- ◆ **Memoria.**- En ella se encuentra almacenado el código de los programas que el sistema puede ejecutar así como los datos. Su característica principal es que debe tener un acceso de lectura y escritura lo más rápido posible para que el microprocesador no pierda tiempo en tareas que no son meramente de cálculo. Al ser volátil el sistema requiere de un soporte donde se almacenen los datos incluso sin disponer de alimentación o energía.
- ◆ **Caché.**- Memoria más rápida que la principal en la que se almacenan los datos y el código accedido últimamente. Dado que el sistema realiza microtareas, muchas veces repetitivas, la caché hace ahorrar tiempo ya que no hará falta ir a memoria principal si el dato o la instrucción ya se encuentra en la caché. Dado su alto precio tiene un tamaño muy inferior (8 a 512 KB) con respecto a la principal (8 a 256 MB). En el interior del chip del microprocesador se encuentra una pequeña caché (L1), pero normalmente se tiene una mayor en otro chip de la placa madre (L2).
- ◆ **Disco Duro.**- En él la información no es volátil y además puede conseguir capacidades muy elevadas. A diferencia de la memoria que es de estado sólido éste suele ser magnético. Pero su excesivo tamaño a veces lo hace inviable para PC embebidos, con lo que se requieren soluciones como discos de estado sólido. Otro problema que presentan los dispositivos magnéticos, a la hora de integrarlos en sistemas embebidos, es que llevan partes mecánicas móviles, lo que los hace inviables para entornos donde estos estarán expuestos a ciertas condiciones de vibración. Existen en el mercado varias soluciones de esta clase (DiskOnChip, CompactFlash, IDE Flash Drive, etcétera) con capacidades suficientes para la mayoría de sistemas embebidos (desde 2 MB hasta más de 1 GB). El controlador del disco duro de PC estándar cumple con el estándar IDE y es un chip más de la placa madre.
- ◆ **Disco Flexible.**- Su función es la de un disco duro pero con discos con capacidades mucho más pequeñas y la ventaja de su portabilidad. Siempre se encuentra en un PC estándar pero no así en un PC embebido.
- ◆ **BIOS.-ROM.**- BIOS (Basic Input and Output System, sistema básico de entrada y salida) es un código que es necesario para inicializar el ordenador y para poner en comunicación los distintos elementos de la placa madre. La ROM (Read Only Memory, memoria de sólo lectura no volátil) es un chip donde se encuentra el código BIOS.
- ◆ **CMOS-RAM.**- Es un chip de memoria de lectura y escritura alimentado con una pila donde se almacena el tipo y ubicación de los dispositivos conectados a la placa madre (disco duro, puertos de entrada y salida, etcétera). Además contiene un reloj en permanente funcionamiento que ofrece al sistema la fecha y la hora.

- ◆ Chip Set.- Chip que se encarga de controlar las interrupciones dirigidas al microprocesador, el acceso directo a memoria (DMA) y al bus ISA, además de ofrecer temporizadores, etcétera. Es frecuente encontrar la CMOS-RAM y el reloj de tiempo real en el interior del Chip Set.
- ◆ Entradas al Sistema.- Pueden existir puertos para ratón, teclado, vídeo en formato digital, comunicaciones serie o paralelo, etcétera.
- ◆ Salidas del Sistema.- Puertos de vídeo para monitor o televisión, pantallas de cristal líquido, altavoces, comunicaciones serie o paralelo, etcétera.
- ◆ Ranuras de Expansión para tarjetas de tareas específicas.- Que pueden no venir incorporadas en la placa madre, como pueden ser más puertos de comunicaciones, acceso a red de ordenadores vía LAN (Local Area Network, Red de Área Local) o vía red telefónica: básica, RDSI (Red Digital de Servicios Integrados), ADSL (Asynchronous Digital Subscriber Loop, Lazo Digital Asíncrono del Abonado), etcétera. Un PC estándar suele tener muchas más ranuras de expansión que un PC embebido. Las ranuras de expansión están asociadas a distintos tipos de bus: VESA, ISA, PCI, NLX (ISA + PCI), etcétera.

Hoy en día existen en el mercado fabricantes que integran un microprocesador y los elementos controladores de los dispositivos fundamentales de entrada y salida en un mismo chip, pensando en las necesidades de los sistemas embebidos (bajo coste, pequeño tamaño, entradas y salidas específicas). Su capacidad de proceso suele ser inferior a los procesadores de propósito general pero cumplen con su cometido ya que los sistemas donde se ubican no requieren tanta potencia. Los principales fabricantes son *ST Microelectronics* (familia de chips STPC), National (familia Geode), Motorola (familia ColdFire) e Intel.

En cuanto a los sistemas operativos necesarios para que un sistema basado en microprocesador pueda funcionar y ejecutar programas suelen ser específicos para los sistemas embebidos. Así nos encontramos con sistemas operativos de bajos requisitos de memoria, posibilidad de ejecución de aplicaciones de tiempo real, modulares (inclusión sólo de los elementos necesarios del sistema operativo para el sistema embebido concreto), etcétera. Los más conocidos en la actualidad son Windows CE, QNX y VxWorks de WindRiver.

1.1.4.- Aplicaciones de un Sistema Embebido.

Los lugares donde se pueden encontrar los sistemas embebidos son numerosos y de varias naturalezas. A continuación se exponen varios ejemplos para ilustrar las posibilidades de los mismos:

- ✚ En una fábrica, para controlar un proceso de montaje o producción. Una máquina que se encargue de una determinada tarea hoy en día contiene numerosos circuitos electrónicos y eléctricos para el control de motores, hornos, etcétera, que deben ser gobernados por un procesador, el cual ofrece un interfase persona-máquina para ser dirigido por un operario e informarle al mismo de la marcha del proceso.
- ✚ Puntos de servicio o venta (POS, Point Of Service). Las cajas donde se paga la compra en un supermercado son cada vez más completas, integrando teclados numéricos, lectores de códigos de barras mediante láser, lectores de tarjetas bancarias de banda magnética o chip, pantalla alfanumérica de cristal líquido, etcétera. El sistema embebido en este caso requiere numerosos conectores de entrada y salida y unas características robustas para la operación continuada.
- ✚ Puntos de información al ciudadano. En oficinas de turismo, grandes almacenes, bibliotecas, etcétera; existen equipos con una pantalla táctil donde se puede pulsar sobre la misma y elegir la consulta a realizar, obteniendo una respuesta personalizada en un entorno gráfico amigable.

- ✚ Decodificadores y *Set-Top Boxes* para la recepción de televisión. Cada vez existe un mayor número de operadores de televisión que aprovechando las tecnologías vía satélite y de red de cable ofrecen un servicio de televisión de pago diferenciado del convencional. En primer lugar envían la señal en formato digital MPEG-2 con lo que es necesario un procesado para decodificarla y mandarla al televisor. Además viaja cifrada para evitar que la reciban en claro usuarios sin contrato, lo que requiere descifrarla en casa del abonado. También ofrecen un servicio de televisión interactiva o web-TV que necesita de un software específico para mostrar páginas web y con ello un sistema basado en procesador con salida de señal de televisión.
- ✚ Sistemas de Radar de Aviones. El procesado de la señal recibida o reflejada del sistema radar embarcado en un avión requiere alta potencia de cálculo además de ocupar poco espacio, pesar poco y soportar condiciones extremas de funcionamiento (temperatura, presión atmosférica, vibraciones, etcétera).
- ✚ Equipos de Medicina en Hospitales y Ambulancias UVI - Móvil.
- ✚ Máquinas de Revelado Automático de Fotos.
- ✚ Cajeros Automáticos.
- ✚ Pasarelas (Gateways) Internet-LAN.

Y un sin fin de posibilidades aún por descubrir o en estado embrionario como son las neveras inteligentes que controlen su suministro vía Internet, PC de bolsillo, etcétera.

1.1.5.- Ventajas de un Sistema Embebido sobre las Soluciones Industriales Tradicionales.

Los equipos industriales de medida y control tradicionales están basados en un microprocesador con un Sistema Operativo propietario o específico para la aplicación correspondiente. Dicha aplicación se programa en ensamblador para el microprocesador dado o en Lenguaje C, realizando llamadas a las funciones básicas de ese sistema operativo que en ciertos casos ni siquiera llega a existir. Con los modernos sistemas PC embebido basados en microprocesadores i486 o i586 se llega a integrar el mundo del PC compatible con las aplicaciones industriales. Ello implica numerosas ventajas: Posibilidad de utilización de sistemas operativos potentes que ya realizan numerosas tareas: comunicaciones por redes de datos, soporte gráfico, concurrencia con lanzamiento de *threads*, etcétera. Estos sistemas operativos pueden ser los mismos que para PC compatibles (Linux, Windows, MS-DOS) con fuertes exigencias en hardware o bien ser una versión reducida de los mismos con características orientadas a los PC embebidos.

Al utilizar dichos sistemas operativos se pueden encontrar fácilmente herramientas de desarrollo software potentes así como numerosos programadores que las dominan, dada la extensión mundial de las aplicaciones para PC compatibles. Reducción en el precio de los componentes hardware y software debido a la gran cantidad de PC en el mundo.





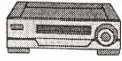




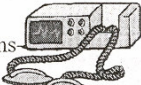


Anti-lock brakes	Modems			
Auto-focus cameras	MPEG decoders			
Automatic teller machines	Network cards			
Automatic toll systems	Network switches/routers			
Automatic transmission	On-board navigation			
Avionic systems	Pagers			
Battery chargers	Photocopiers			
Camcorders	Point-of-sale systems			
Cell phones	Portable video games			
Cell phone base stations	Printers			
Cordless phones	Satellite phones			
Cruise control	Scanners			
Curbside check-in systems	Smart ovens/dishwashers			
Digital cameras	Speech recognizers			
Disk drives	Stereo systems			
Electronic card readers	Teleconferencing systems			
Electronic instruments	Televisions			
Electronic toys/games	Temperature controllers			
Factory control	Theft tracking systems			
Fax machines	TV set-top boxes			
Fingerprint identifiers	VCR's, DVD players			
Home security systems	Video game consoles			
Life-support systems	Video phones			
Medical testing systems	Washers and dryers			

Figura I.1.- Lista de Equipos que utilizan Sistemas Embebidos.

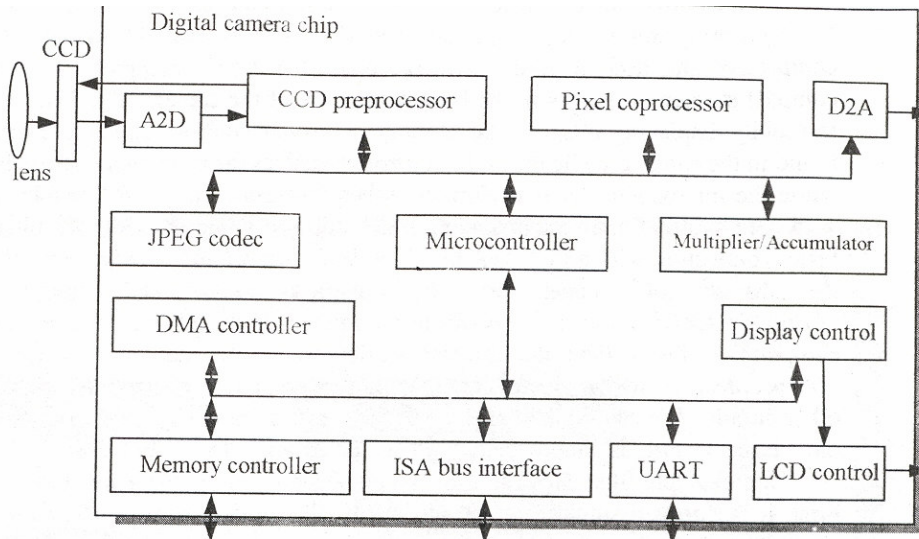


Figura I.2.- Ejemplo de un Sistema Embebido: Una Cámara Digital.

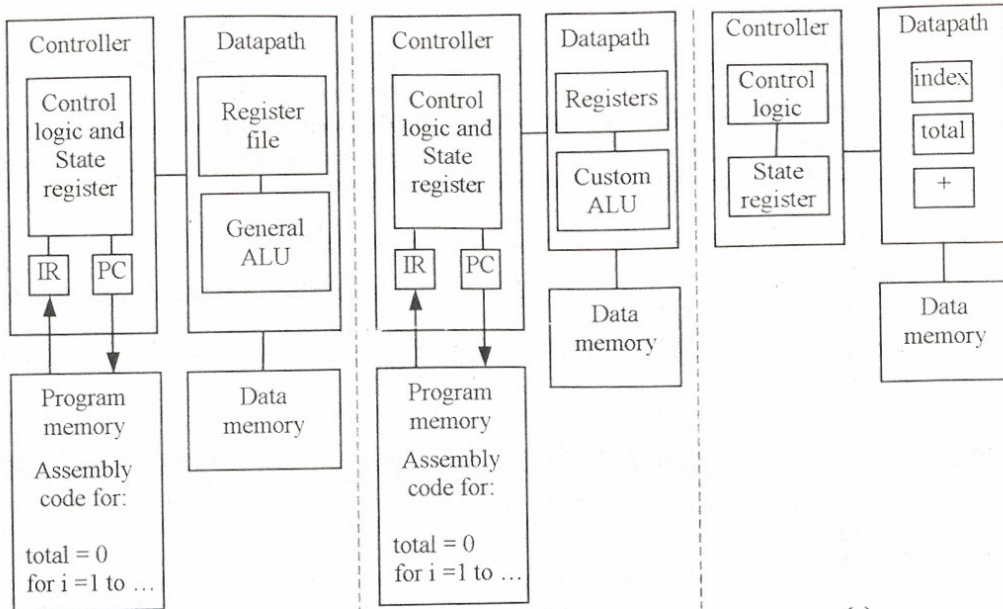


Figura I.3.- Implantación de diferentes tipos de Sistemas con Elementos Embebidos.

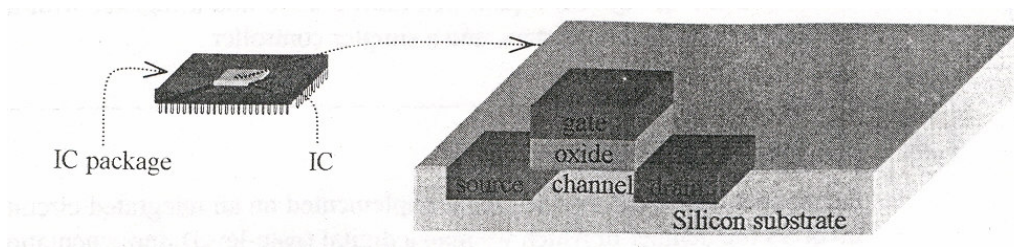


Figura I.4.- Los Circuitos Integrados poseen millones de Transistores.

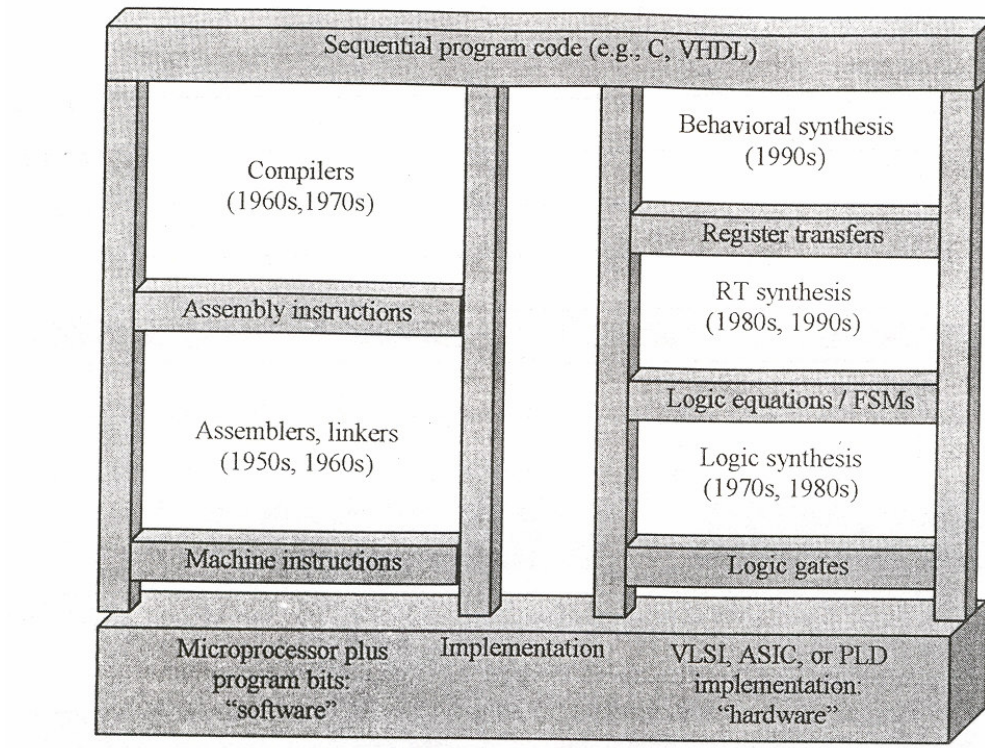


Figura I.5.- El Diseño Sintetizado de *Hardware* y *Software* permite la Operación de los Sistemas Embebidos.

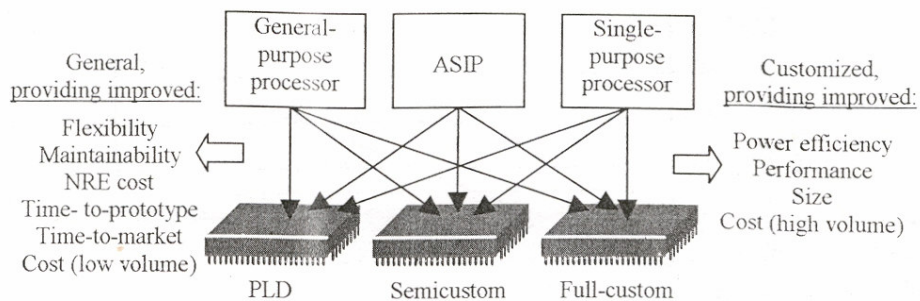


Figura I.6.- La Independencia del Procesador y la Tecnología de los Circuitos Integrados.

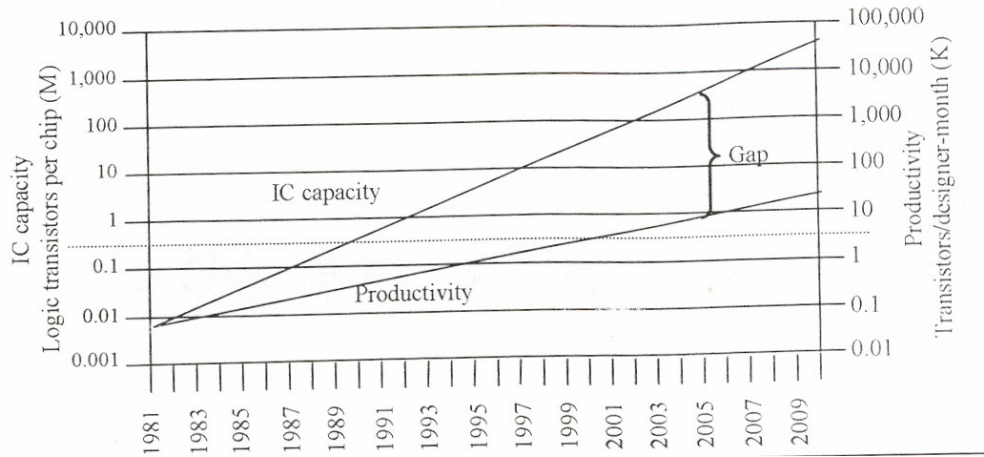


Figura I.7.- El "Crecimiento" en el uso de la Tecnología de los Sistemas Embebidos.

CAPÍTULO II.

TEORÍA DE LOS MICROCONTROLADORES, PIC.

II.1.- Introducción.

El Microcontrolador nace cuando las técnicas de integración han progresado lo bastante para permitir su fabricación; pero también porque, muy a menudo, tanto en las aplicaciones domésticas como industriales, se tiene la necesidad de sistemas “inteligentes” o, al menos programables. Un ejemplo muy simple es el programador de una lavadora, el cual debe controlar una cierta cantidad de elementos con ciclos y cadencias perfectamente definidas, pero variables en función del programa seleccionado. Otras aplicaciones más técnicas tienen, igualmente, necesidad de sistemas programables. Por ejemplo, una fotocopidora debe controlar permanentemente un gran número de elementos y de funciones. Gracias a la llegada de los microcontroladores, tarjetas que contenían varias decenas de circuitos lógicos clásicos se han visto reducidas a dos o tres microcontroladores.

Antes de ver qué es un microcontrolador y de analizar sus puntos fuertes y sus defectos, será útil hacer un repaso relativo a la estructura de cualquier sistema programable, que pueda hacer uso de un microcontrolador.

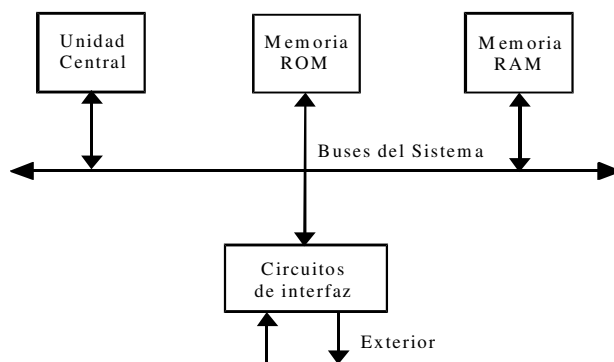


Figura II.1.

La Figura II.1 presenta el esquema tipo de cualquier sistema programable. Véase qué elementos son indispensables para su funcionamiento:

- ⑩ La unidad central
- ⑩ La memoria ROM
- ⑩ La memoria RAM
- ⑩ Los circuitos de interfase
- ⑩ Un bus de interconexión

La presencia de estos elementos básicos es indispensable y aun cuando no siempre se presenten tan claramente como en nuestro esquema siempre existen. Obsérvese, que son los mismos que los de un sistema informático clásico, pero dentro del marco de una aplicación que pueda ser tratada por un microcontrolador.

La unidad central, generalmente constituida por un microprocesador más o menos evolucionado, ejecuta el programa que da vida a la aplicación. Los programas pueden ser muy diversos, puesto que, como es evidente, el que asegura la gestión de un termostato inteligente no tiene nada que ver con el que controla el correcto funcionamiento de una fotocopidora. Sin embargo, estos programas tienen en común el hecho de que muy raramente necesitan cálculos complejos y, en cambio, sí suelen incluir numerosas manipulaciones de la información de entrada/salida.

El programa se almacena en un segundo elemento, que es la memoria ROM. Esta memoria puede constituirse de diferentes formas: UVPR0M, EEPROM u OTPROM, cualquiera que sea la que se utilice es una memoria no volátil desde la que se ejecutará el programa una vez alimentado el sistema. Para poder trabajar correctamente, nuestro microprocesador necesita, a menudo, almacenar datos temporales en alguna parte, y aquí es donde interviene la memoria RAM, que no necesita ser de grandes dimensiones.

El último elemento y que, generalmente, es el más importante en una aplicación susceptible de utilizar un microcontrolador es todo lo concerniente a los circuitos de interfase con el mundo exterior, que relacionará al microprocesador con elementos tan dispares como un motor paso a paso, un Display de cristal líquido o una botonera hexadecimal.

Después de este pequeño análisis nos podemos preguntar por qué se habla de microcontrolador y, no de un conjunto de elementos que realizan esta función. La respuesta es que el objetivo de los microcontroladores es integrar, tanto como sea posible, en un único encapsulado el conjunto de funciones de la Figura II.1.

II.2.- Contenido Típico de un Microcontrolador.

De lo descrito anteriormente, es evidente que un microcontrolador debe contener todos los elementos de la Figura II.1 en un único encapsulado; aunque no con un diseño tan simple. A la vista de los análisis de los sistemas realizados antes de la aparición de los microcontroladores, los fabricantes de circuitos integrados han perfilado la definición de lo que se debería integrar, para llegar al esquema de la Figura II.2.

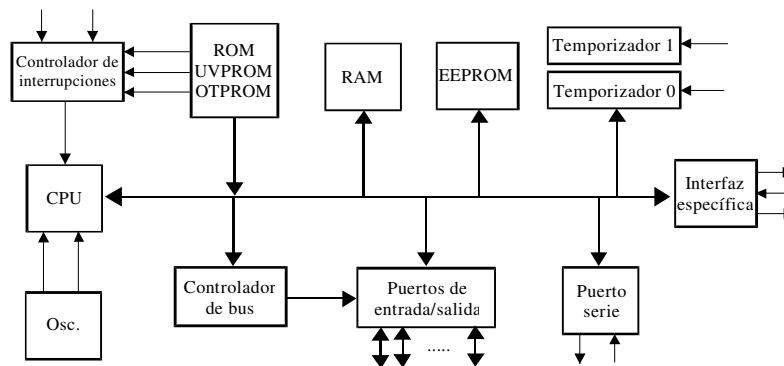


Figura II.2.

Evidentemente, encontramos en él nuestra unidad central pero, salvo casos particulares, frecuentemente se ha simplificado con respecto a los microprocesadores clásicos. En contrapartida se le han añadido instrucciones de manejo de bits, muy útiles para las entradas/salidas. En ciertos circuitos, esta unidad central está dotada de un gran número de registros internos, que sirven de memoria RAM, por lo que puede parecer que ésta última está ausente de algunos esquemas.

A continuación podemos ver la memoria ROM, aunque ésta no siempre aparece. En determinados encapsulados y hasta hace unos años, esta memoria no podía programarse más que mediante máscara durante la fabricación del circuito. Esto imponía al potencial usuario del microcontrolador comprar un número significativo de piezas idénticas, lo cual era aceptable para una serie grande, pero no para fabricaciones limitadas. Cierta número de microcontroladores estaban, y todavía están, disponibles sin ROM (versiones *ROMless* en los catálogos).

Posteriormente, los fabricantes han introducido en el chip una memoria programable eléctricamente y borrable mediante rayos ultravioleta (UVPROM) o, más recientemente, borrable eléctricamente (EEPROM). Como los encapsulados que contenían la memoria UVPROM eran relativamente caros (por la ventana de cuarzo), han aparecido otro tipo denominado OTPROM (One Time PROM), la memoria UVPROM existe siempre y se programa como cualquier circuito, pero debido a la ausencia de ventana, no se puede borrar. Es una solución interesante para la producción de series pequeñas y están disponibles en plásticos baratos.

Un último producto para almacenar de forma no volátil son las memorias FLASH, de bajo consumo, que se puede escribir y borrar en circuito al igual que las EEPROM, pero suelen disponer de mayor capacidad que estas últimas. El borrado solo es posible con bloques completos y no se puede realizar sobre posiciones concretas. Son muy recomendables en aplicaciones en las que sea necesario modificar el programa a lo largo de la vida del producto, como consecuencia del desgaste o cambios de piezas, como sucede con los vehículos.

En lo referente a la memoria RAM, suele utilizarse una del tipo SRAM (RAM estática) de pequeño tamaño, por qué generalmente la unidad central posee suficientes registros para realizar operaciones intermedias. En algunos casos, esta memoria se completa con una EEPROM de datos, que memoriza de forma semipermanente datos del usuario que se manejan como constante en la ejecución del programa y que de vez en cuando (pasados meses o años) deben ser modificados.

Algo más delicado es hacer un esquema tipo para los circuitos de interfase, ya que es un punto donde se distinguen los diferentes microcontroladores, en función de las aplicaciones que pretenden. No obstante se pueden encontrar los siguientes elementos básicos:

- ⑩ Líneas de entrada/salida paralelo, en cantidad variable, según la finalidad y el tamaño del encapsulado (se plantea un problema de número máximo de pines debido al crecimiento del número de estas líneas).
- ⑩ Al menos una interfase de entrada/salida serie asíncrona, más o menos evolucionada según los circuitos.
- ⑩ Uno o varios temporizadores internos cuyas posibilidades pueden ser muy variables pero que, generalmente, funcionan como contadores ascendentes y descendentes, generadores de impulsos programables, etcétera.
- ⑩ Uno o varios convertidores analógicos/digitales, precedidos o no de multiplexores para ofrecer varias vías.
- ⑩ A veces, pero es raro, un convertidor digital/analógico.

Por último, aunque no sea una verdadera interfase de entrada/salida en el sentido en que nosotros entendemos, ciertos microcontroladores disponen de un acceso a su bus interno. Esto permite conectar con otros circuitos destinados a cumplir funciones que faltan en el chip, lo que a veces resulta útil. Precisemos, aunque es evidente, que todos los microcontroladores sin memoria ROM interna disponen necesariamente de esta interfase, puesto que es necesario permitirle acceder a una memoria ROM externa.

II.3.- Las Ventajas y Desventajas de los Microcontroladores.

En primer lugar, un microcontrolador integra en un único encapsulado lo que antes necesitaba una o varias decenas de elementos distintos. Como resultado de estos, se ha obtenido una evidente disminución en el volumen del hardware y del circuito impreso. Esta integración también ha tenido como consecuencia inmediata la simplificación del diseño del circuito impreso, ya que no es necesario llevar los buses de direcciones y de datos de un componente a otro.

En segundo lugar, de todo lo anterior se deriva un aumento de la fiabilidad del sistema ya que, al disminuir el número de componentes, el número potencial de fuentes de error también disminuye. Además, la cantidad de conexiones entre componentes/zócalos o componentes/circuito impreso también disminuye, lo que aumenta la fiabilidad del sistema. Así mismo, la disminución de componentes reduce el consumo global de todo el sistema, lo que según en que aplicaciones y tipos de alimentación se utilice el microcontrolador puede resultar ventajoso.

Los mayores inconvenientes de los microcontroladores son bastante pocos y, principalmente, se encuentran en el nivel de la programación, pero en dos planos diferentes. El primer inconveniente es el sistema de almacenamiento de los programas que lo hacen funcionar, como ya hemos visto, las opciones de almacenamiento pasan por utilizar una memoria ROM en alguna de sus variantes (ROM por máscara, UV PROM, OTPROM, EEPROM, etcétera..), esto implica que la modificación de los programas realizados va a suponer un esfuerzo de borrado de la memoria completa (o de bloques en el mejor de los casos) o la sustitución del chip de memoria por uno nuevo, lo cual conlleva gastos adicionales en material o en esfuerzo.

El otro inconveniente es el de que en los microcontroladores, como cualquier sistema programado, hay que disponer de una herramienta o medio de desarrollo, es necesario escribir los programas, probarlos y comprobarlos sobre el hardware que rodea al microcontrolador, antes de instalarlos y hacer funcionar el sistema. Este sistema de desarrollo representa, por tanto, una inversión que hay que tener en cuenta en el coste del producto final. Si se prevé la realización de aparatos diversos que utilicen microcontroladores de la misma familia, es bastante fácil de amortizar; en caso contrario puede ser más difícil.

II.4.- Presentación de un Microcontrolador: El PIC 16X84.

II.4.1.- Descripción General.

Microcontrolador diseñado por la empresa Microchip (<http://www.microchip.com>), que se fabrica en varias versiones de las cuales las más simples, pero interesantes, son la 16C84 y la 16F84. Son idénticas en cuanto a su arquitectura interna a excepción de la memoria ROM y la memoria RAM. En el primer caso contiene una memoria EEPROM de 1Kbytes de 14 bits cada una, en el segundo diseño la memoria tiene la misma capacidad pero es de tipo Flash.

Tal y como se ha comentado, la memoria EEPROM y la Flash son eléctricamente grabables y borrables, lo que permite escribir y borrar el programa bajo prueba manteniendo el microcontrolador en el mismo zócalo y usando el mismo dispositivo para grabar y borrar. Esta característica supone una gran ventaja frente a la mayoría de los microcontroladores, que tienen como memoria de programa reescribible un tipo EPROM. Se graba eléctricamente, pero para borrarlas hay que someterlas durante cierto tiempo a rayos ultravioleta, lo que implica sacar del zócalo el circuito integrado y colocarlo en un borrador de EPROM.

Otra ventaja del PIC16X84 en cuanto a simplificar el proceso de escritura, borrado y reescritura de programas, tan necesario en la fase de diseño, es su sistema de grabación de datos, que se realiza en serie. Para escribir un programa en la memoria se manda la información en serie usando sólo dos patillas, una para la señal de reloj y otra para los datos serie. A continuación exponemos las características más significativas:

MEMORIA DE PROGRAMA: 1 K x 14
MEMORIA DE DATOS: 36 bytes (PIC16C84) y 68 bytes (PIC16F84)
MEMORIA DE DATOS EEPROM: 64 bytes
PILA (Stack): De 8 niveles
INTERRUPCIONES: 4 tipos diferentes
JUEGO DE INSTRUCCIONES: 35
ENCAPSULADO: Plástico DIP de 18 patillas
FRECUENCIA DE TRABAJO: 10 MHz máxima
TEMPORIZADORES: Sólo uno el TMR0. También tiene Perro Guardián (WDT)
LINEAS DE E/S DIGITALES: 13 (5 Puerta A y 8 Puerta B)
VOLTAJE DE ALIMENTACION (V_{DD}): De 2 a 6 V DC
VOLTAJE DE GRABACION (V_{PP}): De 12 a 14 V DC

Existen otras variantes que se comercializan y responden a la nomenclatura genérica PIC16X8X, atendiendo a diversas características como pueden ser la frecuencia máxima de funcionamiento, el tipo de oscilador externo o el margen del voltaje de alimentación.

II.4.2.- Aspecto Externo.

EL PIC16C(F)84 está fabricado con tecnología CMOS de altas prestaciones y encapsulado en plástico con 18 patillas, con la nomenclatura que se muestra en la Figura II.3. La misión de cada patilla comentada brevemente es:

V_{DD} : Tensión positiva de alimentación.

V_{SS} : Tensión conectada a tierra o negativa de alimentación.

OSC1/CLKIN: Entrada del circuito oscilador externo que proporciona la frecuencia de trabajo del microcontrolador.

OSC2/CLKOUT: Patilla auxiliar del circuito oscilador.

MCLR#: Patilla activa con nivel lógico bajo, lo que se representa con el símbolo #. Su activación origina la reinicialización o Reset del PIC. También se usa durante la grabación de la memoria para introducir por ella la tensión V_{PP} .

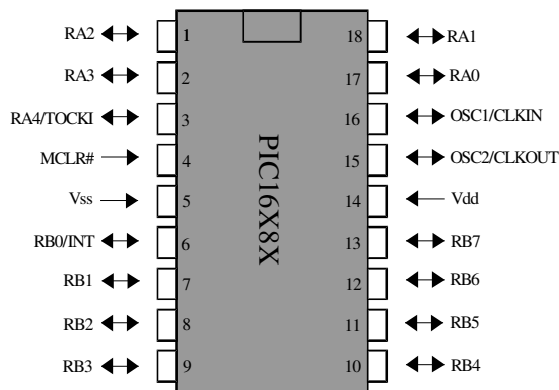


Figura II.3.- El PIC 16C(F)84.

RA0-RA4: Son las 5 líneas de E/S digitales correspondientes a la Puerta A. La línea RA4 multiplexa otra función expresada por TOCKI. En ese caso sirve para recibir una frecuencia externa para alimentar al temporizador TMR0.

RB0-RB7: Son las 8 líneas de E/S digitales de la Puerta B. La línea RB0 multiplexa la función de servir como entrada a una petición externa de una interrupción.

II.4.3.- La Frecuencia de Funcionamiento: El Reloj.

La frecuencia de trabajo del microcontrolador es un parámetro fundamental a la hora de establecer la velocidad de ejecución de instrucciones y el consumo de energía. Cuando un PIC16X84 funciona a 10 MHz, que es su máxima frecuencia, le corresponde un ciclo de instrucción de 400 ns, puesto que cada instrucción tarda en ejecutarse 4 períodos de reloj, o sea, $4 \times 100 \text{ ns} = 400 \text{ ns}$. Todas las instrucciones del PIC se realizan en un ciclo de instrucción, menos las de salto que tardan el doble.

Los impulsos de reloj entran por la patilla OSC1/CLKIN y se dividen por 4 internamente, dando lugar a las señales Q1, Q2, Q3 y Q4. Durante un ciclo de instrucción, que comprende las cuatro señales mencionadas, se desarrollan las siguientes operaciones:

Q1: Durante este impulso se incrementa el Contador de Programa.
Q4: Durante este impulso se busca el código de la instrucción en la memoria del programa y se carga en el Registro de Instrucciones.
Q2-Q3: Durante la activación de estas dos señales se produce la decodificación y la ejecución de la instrucción.

Para conseguir ejecutar cada instrucción en un ciclo de instrucción (excepto las de salto), se aplica la técnica de segmentación o *pipe-line*, que consiste en realizar en paralelo las dos fases que comprende cada instrucción.

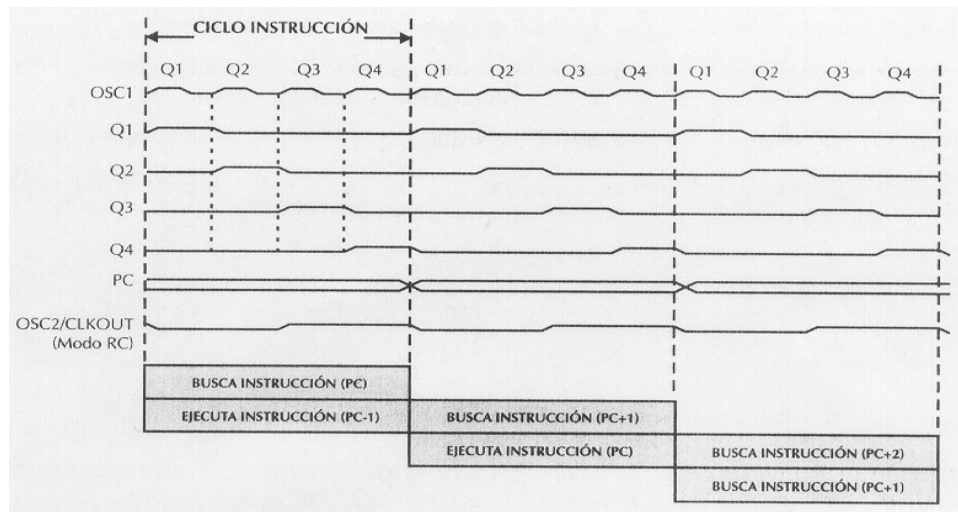


Figura II.4.- Pipeline del Microcontrolador.

La estructura segmentada del procesador permite realizar al mismo tiempo la fase de ejecución de una instrucción y la de búsqueda de la siguiente. Cuando la instrucción ejecutada corresponde a un salto no se conoce cuál será la siguiente hasta que se realice, por eso en esta situación se sustituye la fase de búsqueda por un ciclo "vacío", originando que las instrucciones de salto tarden en realizarse dos ciclos de instrucción.

II.4.3.1.- Tipos de Osciladores.

Los PIC admiten cuatro tipos de osciladores externos para aplicarles la frecuencia de funcionamiento, se colocan entre las patillas OSC1 y OSC2. El tipo empleado debe especificarse en dos bits de la "Palabra de Configuración", como se comentará más adelante. Los tipos que se pueden emplear son:

- ⑩ **Oscilador RC:** Es un oscilador de bajo coste formado por una simple resistencia y un condensador. Proporciona una estabilidad mediocre de la frecuencia, cuyo valor depende de los valores de los dos elementos R-C.
- ⑩ **Oscilador HS:** Es un oscilador que alcanza una alta velocidad comprendida entre 4 y 10 MHz y está basado en un cristal de cuarzo o un resonador cerámico.

- ⑩ *Oscilador XT*: Es un oscilador de cristal o resonador para frecuencias estándar comprendidas entre 100 KHz y 4 MHz.
- ⑩ *Oscilador LP*: Oscilador de bajo consumo con cristal o resonador diseñado para trabajar en un rango de frecuencias de 35 a 200 KHz.

II.4.4.- Reinicialización o RESET.

Cuando se aplica un nivel lógico bajo a la patilla MCLR# el microcontrolador reinicializa su estado. Dos acciones importantes se producen en la reinicialización o RESET:

1. El Contador de Programa se carga con la dirección 0, apuntando a la primera dirección de la memoria de programa en donde deberá estar situada la primera instrucción del programa de aplicación.
2. la mayoría de los registros de estado y control del procesador toman un estado conocido y determinado.

Se puede provocar el RESET de varias maneras, pero si se desea realizar manualmente, habrá que colocar, conectado a la patilla MCLR#, un circuito con un pulsador, que al ser apretado genere un nivel lógico bajo.

II.5.- Arquitectura de los Microcontroladores PIC16X84.

Para lograr una compactación de código óptima y una velocidad superior a la de sus competidores, los microcontroladores PIC incorporan en su procesador tres de las características más avanzadas en los grandes computadores:

- ⑩ Procesador tipo RISC.
- ⑩ Ejecución segmentada.
- ⑩ Arquitectura HARVARD.

El juego de instrucciones se reduce a 35 y sus modos de direccionamiento se han simplificado al máximo. Con la estructura segmentada se pueden realizar simultáneamente las dos fases en que se descompone cada instrucción. Con la arquitectura HARVARD se puede acceder de forma simultánea e independiente a la memoria de datos y a la de programa. El aislamiento y la diferenciación de los dos tipos de memoria permite que cada uno tenga la longitud y el tamaño más adecuado. De esta forma, en el PIC16X84 la longitud de datos es de un byte, mientras que la de las instrucciones es de 14 bits.

Otra característica relevante de los PIC es el manejo intensivo del Banco de Registros, los cuales participan de manera muy activa en la ejecución de las instrucciones. De igual forma, la memoria RAM complementa los registros internos implementando en sus posiciones registros de propósito específico y de propósito general.

La arquitectura interna del PIC16X84 se presenta en la figura II.5 y consta de siete bloques fundamentales.

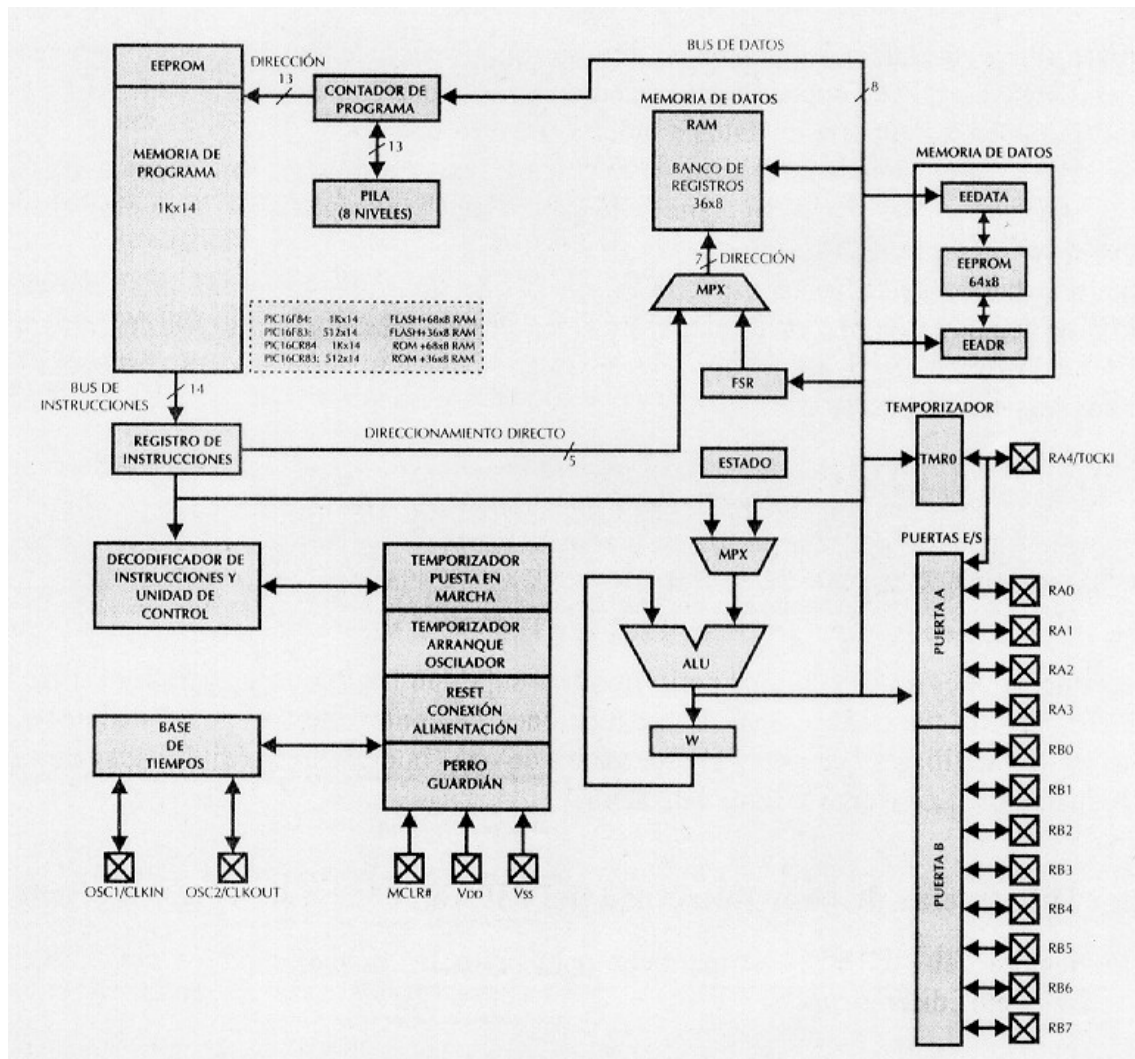


Figura II.5.- Arquitectura Interna del PIC16F84.

1. Memoria de programa EEPROM o Flash de 1 k x 14 bits.
2. Memoria de datos formada por dos áreas. Una RAM donde se alojan 22 registros de propósito específico (SFR) y 36 de propósito general (GDR), y otra de tipo EEPROM de 64 bytes.
3. Camino de datos con una ALU de 8 bits y un registro de trabajo W del que normalmente recibe un operando y envía el resultado. EL otro operando puede provenir del bus de datos o del propio código de la instrucción.
4. Diversos recursos conectados al bus de datos, tales como Puertas de Entrada/Salida, Temporizador TMR0, etcétera
5. Base de tiempos y circuitos auxiliares.
6. Direccionamiento de la memoria de programa en base al Contador de Programa ligado a una Pila de 8 niveles de profundidad.
7. Direccionamiento directo e indirecto de la memoria RAM.

II.5.1.- Memoria de Programa.

La arquitectura de los PIC de la gama media admite un mapa de memoria de programa capaz de contener 8192 instrucciones de 14 bits cada una. Este mapa se divide en páginas de 2048 posiciones, para direccionar 8 K posiciones se necesitan 13 bits, que es la longitud que tiene el Contador de Programa. Sin embargo el PIC16X84 solo tiene implementadas 1 K posiciones, por lo que se ignoran los tres bits de más peso del PC.

Al igual que todos los registros específicos que controlan la actividad del procesador, el Contador de Programa está implementado sobre un par de posiciones de la memoria RAM. Cuando se escribe el Contador de Programa como resultado de una operación de la ALU, los 8 bits de menos peso del PC residen en el registro PCL, que ocupa, repetido, la posición 2 de los dos bancos de la memoria de datos. Los bits de más peso, PC <12:8>, residen en los 5 bits de menos peso del registro PCLATH, que ocupa la posición 0A H de los dos bancos de la memoria RAM. En las instrucciones de salto, los 11 bits de menos peso del PC provienen del código de instrucción y los otros dos de los bits de PCLATH <4:3>, tal y como se muestra en la Figura II.6.

La Pila es una zona aislada de las memorias de instrucciones y datos. Tiene una estructura LIFO, en la que el último valor guardado es el primero que sale. Tiene 8 niveles de profundidad cada uno con 13 bits. Funciona como un buffer circular, de manera que el valor que se obtiene al realizar el noveno POP es igual al que se obtuvo en el primero.

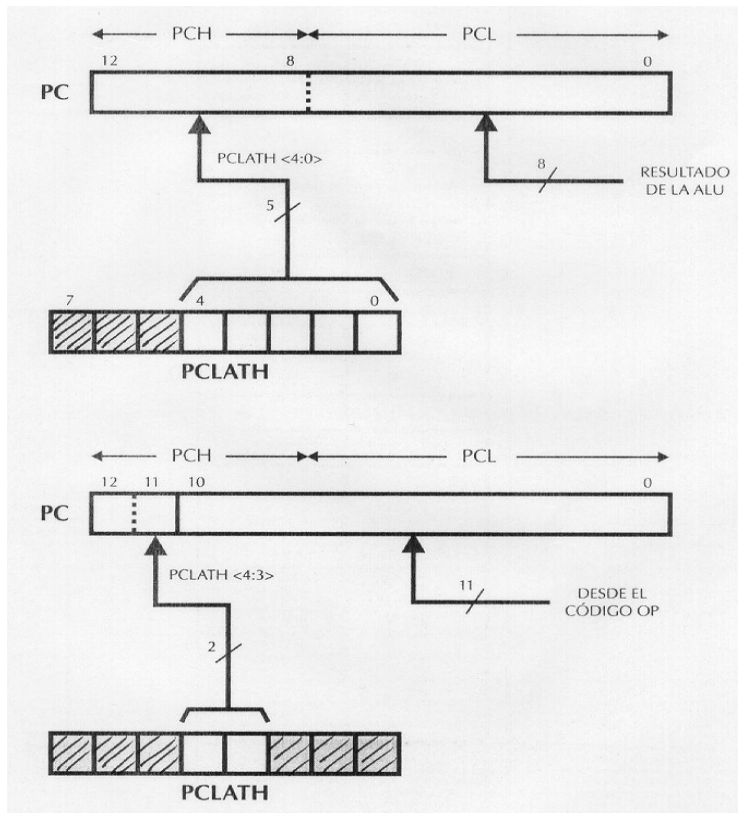


Figura II.6.- Estructura LIFO en un PIC.

II.5.2.- Memoria de Datos RAM.

La memoria de datos del PIC16X84 dispone de dos zonas diferentes:

1. **Área de RAM estática o SRAM**, donde reside el Banco de Registros Específicos (SFR) y el Banco de Registros de Propósito General (GPR). El primer banco tiene 24 posiciones de tamaño byte, aunque dos de ellas no son operativas, y el segundo 68.
2. **Área EEPROM**, de 64 bytes donde, opcionalmente, se pueden almacenar datos que no se pierden al desconectar la alimentación.

La zona de memoria RAM se halla dividida en dos bancos (banco 0 y banco 1) de 128 bytes cada uno. En el PIC16X84 sólo se hallan implementadas físicamente las 48 primeras posiciones de cada banco, de las cuales las 12 primeras están reservadas a los Registros de Propósito Específico (SFR), que son los encargados del control del procesador y sus recursos.

Algunos de dichos registros se hallan repetidos en la misma dirección de los dos bancos, para simplificar su acceso (INDF, ESTADO, FSR, PCLATH e INTCON). Las posiciones apuntadas por la dirección 7 H y la apuntada por la 87 H no son operativas. Los 36 registros restantes de cada banco se destinan a Registros de Propósito General y en realidad sólo son operativos los 68 del banco 0 porque los del banco 1 se mapean sobre el banco 0, es decir, cuando se apunta a un registro general del banco 1, se accede al mismo del banco 0, véase la Figura II.7.

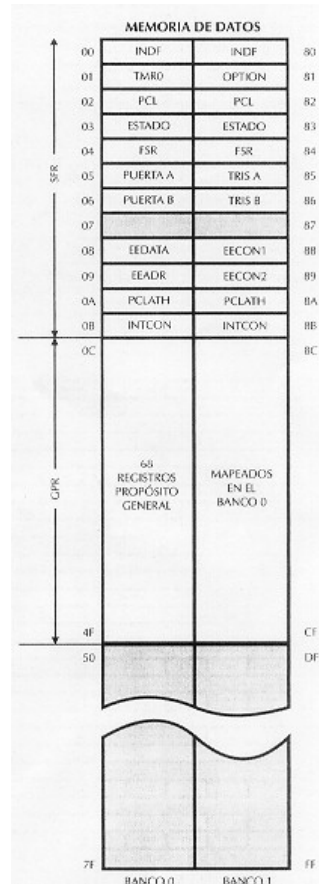


Figura II.7.- Memoria de Datos.

Para seleccionar el banco a acceder hay que manipular el bit 5 (RP0) del registro ESTADO. Si RP0=1 se accede al banco 1 y si RP0=0 se accede al banco 0. Tras un Reset se accede automáticamente al banco 0. Para seleccionar un registro de propósito general no hay que tener en cuenta el estado del bit RP0, porque al estar mapeado el banco 1 sobre el banco 0, cualquier direccionamiento de un registro del banco 1 corresponde a su homólogo del banco 0. En el direccionamiento directo a los registros GPR se ignora el bit de más peso, que identifica el banco y sus direcciones están comprendidas entre el valor 0x0C y 0x4F en Hexadecimal.

II.5.3.- Direccionamiento de la Memoria de Datos.

En los PIC de gama media la memoria de datos está organizada para alojar un máximo de 4 bancos de 128 bytes cada uno. Los PIC16C84 sólo tienen implementados los 48 primeros bytes de los bancos 0 y 1. En el resto de los PIC de esta familia se destinan dos bits del registro ESTADO (RP0 y RP1) para determinar el banco y otros siete para elegir una de las 128 posiciones del banco seleccionado, tal y como muestra la Figura II.8.

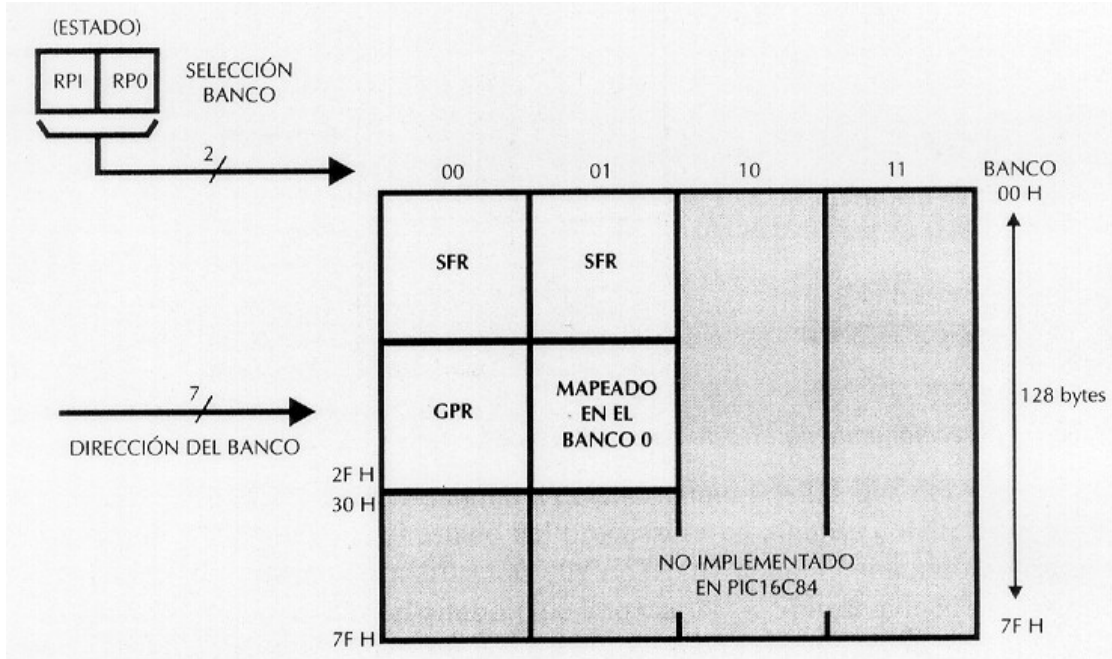


Figura II.8.- Direccionamiento de la Memoria de Datos de un PIC.

- ⑩ Direccionamiento Directo: El operando que utiliza la instrucción en curso se referencia mediante su dirección, que viene incluida en el código OP de la misma, concretamente en los 7 bits de menos peso. El banco a acceder lo determinan los bits RP0 y RP1 del registro ESTADO. En el caso del PIC16C84 sólo se usa el bit RP0 al tener implementados únicamente dos bancos.
- ⑩ Direccionamiento Indirecto: Este modo de direccionamiento se usa cuando en una instrucción se utiliza como operando el registro INDF, que ocupa la dirección 0 de ambos bancos. En realidad el registro INDF no está implementado físicamente y cuando se le hace referencia, se accede a la dirección de un banco especificada con los bits de menos peso del registro FSR. El bit de más peso de FSR junto al bit IRP del registro ESTADO se encargan de seleccionar el banco a acceder, mientras que los 7 bits de menos peso apuntan a la posición. Como sólo hay dos bancos en el PIC16C84 en este modo de direccionamiento, el bit IRP es 0 siempre.

II.5.4.- El Registro de Estado.

Hasta ahora ESTADO es el registro más usado y llega el momento de describirlo en su totalidad. Ocupa la dirección 3 tanto del banco 0 como del 1 de la memoria de datos RAM. Sus bits tienen tres misiones distintas:

1. Se encargan de avisar de la incidencias del resultado de la ALU (C, DC y Z).
2. Indican el estado de Reset (TO# y PD#).
3. Seleccionan el banco a acceder en la memoria de datos (IRP, RP0 y RP1).

Ahora, se muestra el diagrama de distribución de los bits del registro ESTADO, su misión es la siguiente:

C: Acarreo en el bit de más peso

- 1: Acarreo en el bit de más peso.
- 0: No acarreo en el bit de más peso.

DC: Acarreo en el 4º bit

- 1: Acarreo en el 4º bit.
- 0: No acarreo en el 4º bit.

Z: Cero

- 1: El resultado de una instrucción lógico-aritmética ha sido cero.
- 0: El resultado de una instrucción lógico-aritmética no ha sido cero.

PD#: Power Down

- 1: Se pone a este valor después de la conexión a la alimentación o al ejecutar *clrwdt*.
- 0: Se pone a este valor al ejecutar *sleep*.

TO#: Time Out

- 1: Se pone a este valor después de la conexión a la alimentación o al ejecutar *clrwdt* y *sleep*.
- 0: Se pone a este valor al desbordarse el Perro Guardián (*Watchdog*).

RP1-RP0: Selección de banco en direccionamiento directo

Como el PIC16X84 sólo tiene dos bancos únicamente emplea el bit RP0, de forma que cuando vale 1 se accede al banco 1 y cuando vale 0 se accede al banco 0. Después de un Reset RP0=0.

IRP: Selección del banco en direccionamiento indirecto

Este bit junto con el de más peso del registro FSR sirven para determinar el banco de la memoria de datos seleccionado. En el PIC16X84 al disponer de dos bancos no se usa este bit.

II.5.5.- Temporizador/Contador TMR0.

Una de las labores más habituales en los programas de control de dispositivos suele ser determinar intervalos concretos de tiempo, y recibe le nombre de "temporizador" (*timer*) el elemento encargado de realizar esta función. También suele ser frecuente contar los impulsos que se producen en el exterior del sistema, y el elemento destinado a este fin se denomina "contador". Si las labores del temporizador o contador las asignamos al programa principal robarían mucho tiempo al procesador en detrimento de actividades más importantes. Por este motivo se diseñan recursos específicamente orientados a estas misiones.

Los PIC16X84 poseen un temporizador/contador de 8 bits, llamado TMR0, que actúa de dos maneras diferentes:

- 1.^a Como contador de sucesos, que están representados por los impulsos que se aplican a la patilla RA4/T0CKI. Al llegar al valor FF H se desborda el contador y, con el siguiente impulso, pasa a 00 H, advirtiendo esta circunstancia con la activación de un señalizador y/o provocando una interrupción.
- 2.^a Como temporizador, cuando se carga en el registro que implementa el recurso un valor inicial se incrementa con cada ciclo de instrucción ($F_{osc}/4$) hasta que se desborda, o sea, pasa de FF H a 00 H y avisa poniendo a el bit señalizador y/o provocando una interrupción.

Para que el TMR0 funcione como un cantador de impulsos aplicados a la patilla T0CKI hay que poner a 1 el bit T0CS, que es el que ocupa la posición 5 del registro OPTION. En esta situación, el registro TMR0, que es el ubicado en la dirección 1 del banco 0 de la memoria de datos, se incrementa con cada flanco activo aplicado a la patilla T0CKI. El tipo de flanco activo se elige programando el bit T0SE, que es el que ocupa la posición 4 del registro OPTION. Si T0SE = 1, el flanco activo es el de bajada, y si T0SE = 0, es el de subida. Cuando se desea que TMR0 funcione como temporizador el bit T0CS = 0.

En realidad, los PIC16X84 disponen de dos temporizadores, el TMR0 y el Perro Guardián (*Watchdog*). El primero actúa como principal y sobre él recae el control de tiempos y la cuenta de impulsos. El otro vigila que el programa no se cuelgue, y para ello cada cierto tiempo comprueba si el programa se está ejecutando normalmente. En caso contrario, si el control está detenido en un bucle infinito a la espera de algún acontecimiento que no se produce, el Perro Guardián "ladra", lo que se traduce en un Reset que inicializa todo el sistema.

A menudo el TMR0 y el Perro Guardián precisan controlar largos intervalos de tiempo y necesitan aumentar la duración de los impulsos de reloj que les incrementa. Para cubrir este requisito se dispone de un circuito programable denominado Divisor de frecuencia, que divide la frecuencia utilizada por diversos rangos. Para programar el comportamiento del TMR0, el Perro Guardián y el Divisor de frecuencia se utilizan algunos bits del registro OPTION y de la Palabra de Configuración, que se verán más adelante. En la Figura II.9 se proporciona un esquema simplificado de la arquitectura del circuito de control de tiempos usado en los PIC16X84.

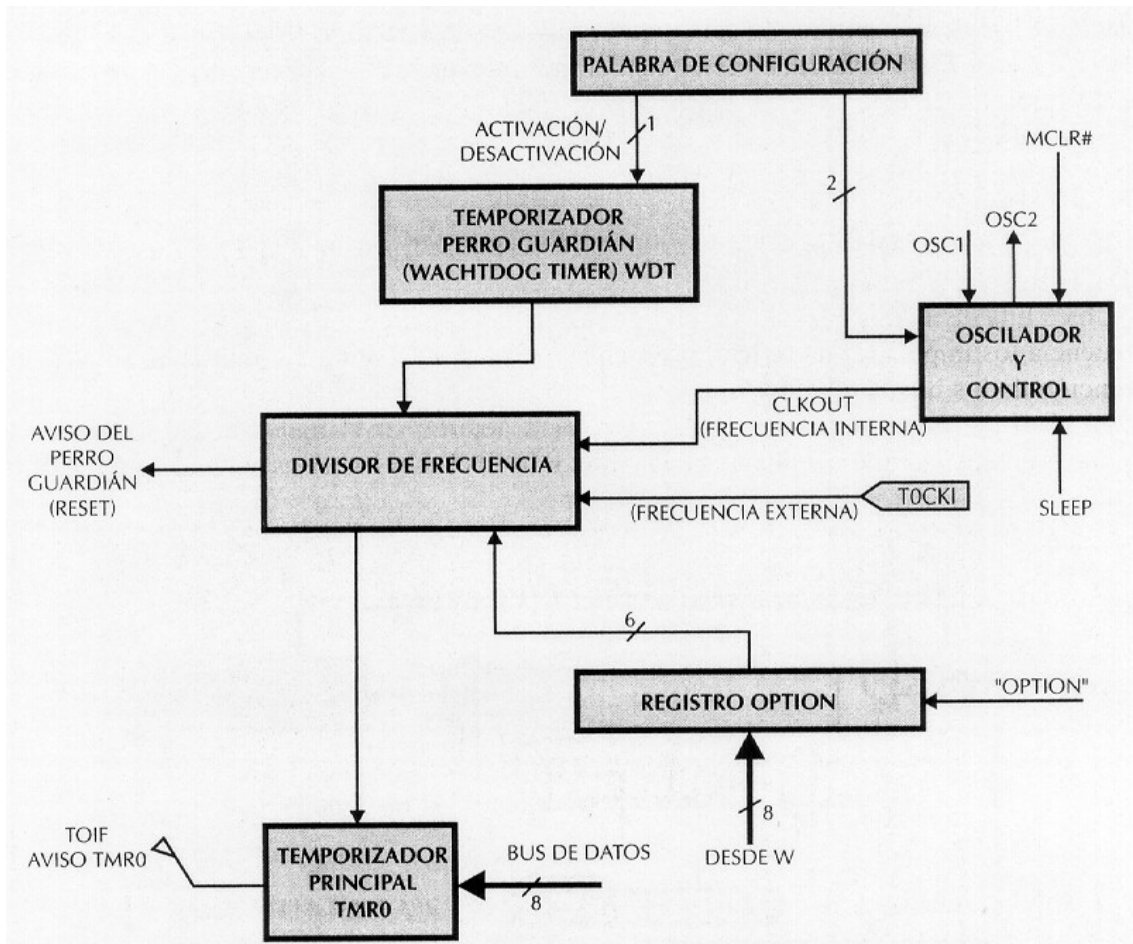


Figura II.9.- Temporizador/Contador de un Microcontrolador.

El Divisor de frecuencias puede usarse con el TMR0 o con el WDT. Con el TMR0 actúa como *Pre-divisor*, es decir, los impulsos pasan primero por el Divisor y luego se aplican al TMR0, una vez aumentada su duración. Con el WDT actúa después, realizando la función de *Post-divisor*. Los impulsos, que divide por un rango el Divisor de frecuencia, pueden provenir de la señal de reloj interna ($F_{osc}/4$) o de los que se aplican a la patilla TOCKI. El TMR0 se comporta como un registro de propósito específico (SFR) ubicado en la dirección 1 del banco 0 de la memoria de datos. En igual dirección, pero en el banco 1, se halla el registro OPTION.

TMR0 puede ser leído y escrito en cualquier momento al estar conectado al bus de datos. Funciona como un contador ascendente de 8 bits. Cuando funciona como temporizador conviene cargarle con el valor de los impulsos que se quiere temporizar, pero expresados en complemento a 2. De esta manera, al llegar al número de impulsos deseados se desborda y al pasar por 00 H se activa el señalizador TOIF y/o se produce una interrupción.

Para calcular los tiempos a controlar con TMR0 se utilizan las siguientes fórmulas prácticas.

$$\text{Temporización} = 4 \cdot T_{osc} \cdot (\text{Valor cargado en TMR0}) \cdot (\text{Rango del Divisor})$$

$$\text{Valor a cargar en TMR} = \text{Temporización} / 4 \cdot T_{osc} \cdot \text{Rango del Divisor}$$

En cualquier momento se puede leer el valor que contiene TMR0, sin detener su cuenta. En la Figura 10 se ofrece el esquema de funcionamiento de TMR0. Obsérvese que hay un bloque que retrasa 2 ciclos la cuenta para sincronizar el momento del incremento producido por la señal aplicada a T0CKI con el que se producen los impulsos internos de reloj. Cuando se escribe TMR0 se retrasan 2 ciclos su reincremento y se pone a 0 el Divisor de frecuencia.

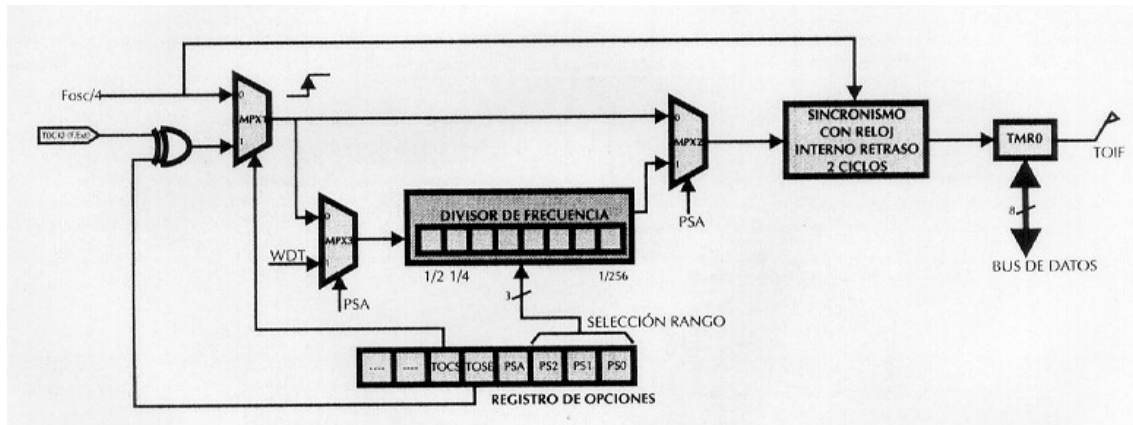


Figura II.10.- Funcionamiento del TMR0.

II.5.6.- El Registro OPTION.

La misión principal de este registro es controlar TMR0 y el Divisor de frecuencia. Ocupa la posición 81 H de la memoria de datos, que equivale a la dirección 1 del banco 1. EL bit T0CS (*Timer 0 Clock edge Select*) selecciona en el multiplexor MPX1 la procedencia de los impulsos de reloj, que pueden ser del oscilador interno (Fosc/4) o los que se aplican desde el exterior por la patilla T0CKI. El bit T0SE (*Timer 0 clock Source sElect*) elige el tipo de flanco activo en los impulsos externos. El bit PSA del registro OPTION asigna el Divisor de frecuencia al TMR0 (PSA= 0) o al WDT (PSA = 1).

Los 3 bits de menos peso de OPTION seleccionan el rango por el que divide el Divisor de frecuencia los impulsos que se le aplican en su entrada. El bit 6 INTEDG (*INTerrupt EDGE*) sirve para determinar el flanco activo que provocará una interrupción externa al aplicarse a la patilla RB0/INT. Un 1 si es de subida y un 0 si es de bajada. El bit 7 RBPUP# (*RB Pull-Up*) activa, si vale 0, o desactiva, cuando vale 1, las resistencias Pull-Up que pueden conectarse en las líneas de la Puerta B. La Figura II.11 muestra la distribución y función de los bits de OPTION.

RBPO#	INTEDG	TOCS	TOSE	PSA	PS2	PS1	PS0
-------	--------	------	------	-----	-----	-----	-----

PS2: PS0 Valor con el que actúa el Divisor de frecuencia

PS2	PS1	PS0	División del TMR0	División del WDT
0	0	0	1:2	1:1
0	0	1	1:4	1:2
0	1	0	1:8	1:4
0	1	1	1:16	1:8
1	0	0	1:32	1:16
1	0	1	1:64	1:32
1	1	0	1:128	1:64
1	1	1	1:256	1:128

PSA: Asignación del Divisor de frecuencias
1 = El Divisor de frecuencia se le asigna al WDT
0 = El Divisor de frecuencia se le asigna al TMR0

TOSE: Tipo de flanco en T0CKI
1 = Incremento de TMR0 cada flanco descendente
0 = Incremento de TMR0 cada flanco ascendente

TOCS: Tipo de reloj para el TMR0
1 = Pulsos introducidos a través de T0CKI (contador)
0 = Pulsos de reloj interno Fosc/4 (temporizador)

INTEDG: Flanco activo interrupción externa
1 = Flanco ascendente
0 = Flanco descendente

RBPO#: Resistencias Pull-up Puerta B
1 = Desactivadas
0 = Activadas

Figura II.11.- El Registro OPTION.

II.5.7.- El Perro Guardián (WDT).

Se trata de un contador interno de 8 bits que origina un *Reset* cuando se desborda. Su control de tiempos es independiente del TMR0 y está basado en un simple circuito R-C. Su actuación es opcional y puede bloquearse para que no funcione programando el bit WDTE de la palabra de Configuración. La temporización nominal con la que se halla programado el Perro Guardián es de 18 ms, pero utilizando el Divisor de frecuencia puede aumentarse hasta alcanzar los 2,3 segundos.

Para evitar que se desborde el Perro Guardián hay que refrescarlo previamente. En realidad este refresco consiste en ponerle a cero mediante las instrucciones *clrwdt* y *sleep*. El programador debe analizar las instrucciones de la tarea y situar alguna de esas dos en sitios estratégicos por los que pase el flujo de control antes que transcurra el tiempo asignado al WDT. De esta manera si el programa se cuelga no se refresca el Perro Guardián y se produce la reinicialización del sistema.

La instrucción *clrwdt* borra al WDT y reinicia su cuenta. Sin embargo, la instrucción *sleep*, además de borrar WDT detiene al sistema y lo mete en un estado de reposo o de bajo consumo. Si no se desactiva el Perro Guardián al entrar en modo de reposo, al completar su cuenta provocará un *Reset* y sacará al microcontrolador del modo de bajo consumo. Para desactivar el Perro Guardián hay que escribir un 0 en el bit 2 (WDTE) de la Palabra de Configuración.

II.5.8.- Las Puertas de E/S.

Los PIC16X84 sólo disponen de dos puertas de E/S. La Puerta A posee 5 líneas, RA0-RA4, y una de ellas soporta dos funciones multiplexadas. Se trata de RA4/TOCKI, que puede actuar como línea de E/S o como la patilla por la que se reciben los impulsos que debe contar TMR0. La Puerta B tiene 8 líneas, RB0-RB7, y también tiene una con funciones multiplexadas, la RB0/INT, que, además de línea típica de E/S, también sirve como patilla por la que reciben los impulsos externos que provocan una interrupción.

Cada línea de E/S puede configurarse independientemente como entrada o como salida, según se ponga a 1 o a 0, respectivamente, el bit asociado del registro de configuración de cada puerta (TRISA y TRISB). Se llaman PUERTAA y PUERTAB los registros que guardan la información que entra o sale por la puerta y ocupan las direcciones 5 y 6 del banco 0 de la memoria de datos. Los registros de configuración TRISA y TRISB ocupan las mismas direcciones pero en el banco 1. Al reiniciarse el PIC todos los bits de los registros TRIS quedan a 1, con lo que las líneas de las puertas quedan configuradas como entradas. Cada línea de salida puede suministrar una corriente máxima de 20 mA y si es de entrada puede absorber hasta 25 mA. Al existir una limitación en la disipación máxima de la potencia del chip se restringe la corriente máxima de absorción de la Puerta A a 80 mA y la de suministro a 50 mA. La Puerta B puede absorber un máximo de 150 mA y suministrar un total de 100 mA.

II.5.8.1.- La Puerta A.

Las líneas RA3-RA0 admiten niveles de entrada TTL y de salida CMOS. La línea RA4/TOCKI dispone de un circuito *Trigger Schmitt* que proporciona una buena inmunidad al ruido y la salida tiene drenador abierto. RA4 multiplexa su función de E/S con la entrada de impulsos externos para el TMR0. En el circuito de la figura II.12 se muestra la adaptación de las patillas RA3-RA0 a las señales internas del procesador.

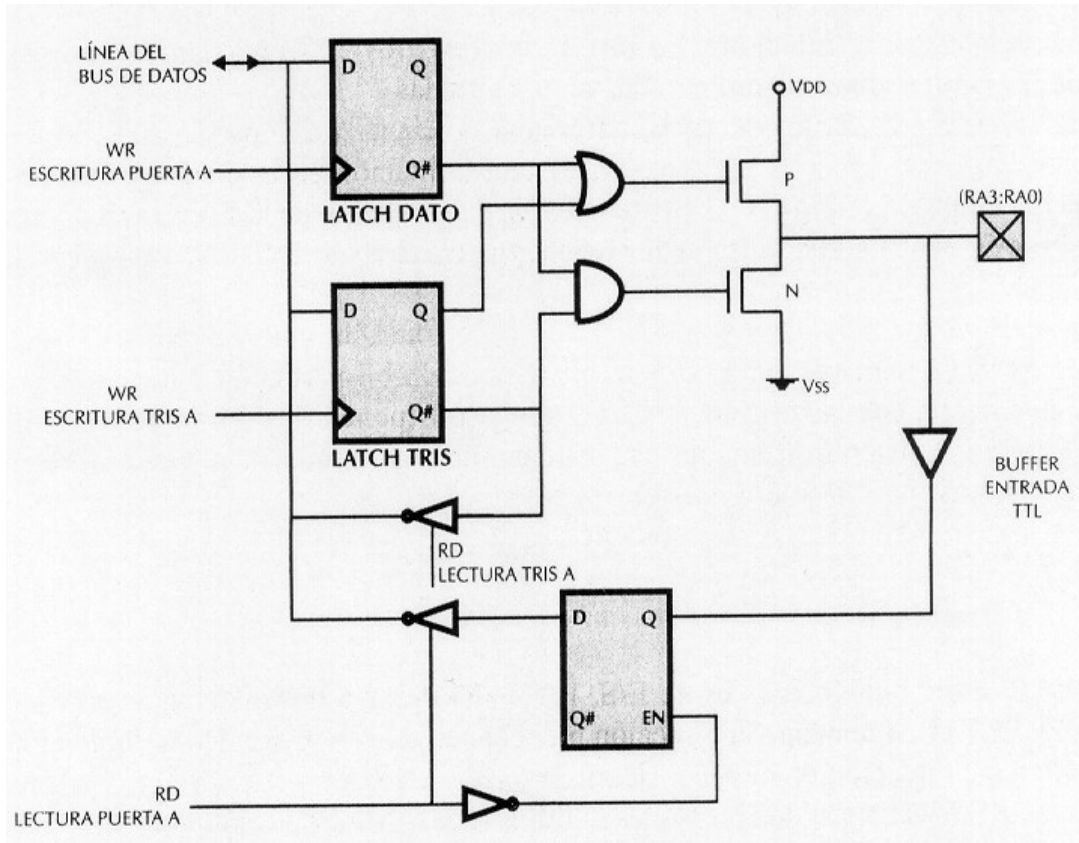


Figura II.12.- La Puerta A.

Cuando se lee una línea de la Puerta A se recoge el nivel lógico que tiene en ese momento. Las líneas cuando actúan como salidas están conectadas a unos biestables, lo que significa que sus patillas sacan el nivel lógico que se haya cargado por última vez en el registro PUERTAA. La escritura de una puerta, tal y como se ve en la Figura II.12, implica que primero se deposita el nivel lógico en la línea correspondiente del bus interno de datos y se activa la señal WRITE, lo que origina el almacenamiento de dicho nivel en el biestable. En esta situación, el biestable de configuración (TRIS) debería tener valor "0" lógico para que actuase como salida. Con estos valores la puerta OR tendría un "0" lógico a su salida y la AND también. Estos valores producen la conducción de los transistores MOS, en función del valor escrito, dando como salida final V_{DD} o V_{SS} ("1" lógico ó "0" lógico respectivamente).

Si una línea actúa como entrada, el nivel lógico depositado en ella desde el exterior pasa a la línea correspondiente del bus interno de datos cuando se activa la señal READ y se hace conductor el dispositivo triestado que les une. Al programarse como entrada, los dos transistores MOS de salida quedan bloqueados y la línea de alta impedancia. Téngase en cuenta que cuando se lee una línea de entrada se obtiene el estado actual que tiene su patilla correspondiente (almacenado en el tercer biestable) y no el que haya almacenado en el biestable de datos. La información presente en una línea de entrada se muestrea al iniciarse el ciclo de instrucción y debe mantenerse estable durante su desarrollo.

II.5.8.2.- La Puerta B.

Consta de 8 líneas bidireccionales de E/S, RB7-RB0, cuya información se almacena en el registro PUERTAB, que ocupa la dirección 6 del banco 0. El registro de configuración TRISB ocupa la misma dirección pero del banco 1. La línea RB0/INT tiene dos funciones multiplexadas; además de patilla de E/S actúa como la patilla para la petición de una interrupción externa, cuando se autoriza esta función mediante la adecuada programación del registro INTCON.

A todas las líneas de esta puerta se les permite conectar una resistencia *pull-up* de elevado valor con el positivo de la alimentación. Para este fin hay que programar en el registro OPTION el bit RBPUP# = 0, afectando la conexión de la resistencia a todas las líneas. Con el *Reset* todas las líneas quedan configuradas como entradas y se desactivan las resistencias de *pull-up*.

Las cuatro líneas de más peso, RB7-RB4, pueden programarse para soportar una misión especial. Cuando las 4 líneas actúan como entradas se las puede programar para generar una interrupción si alguna de ellas cambia su estado lógico. Esta posibilidad es muy práctica en el control de teclados. En la Figura II.13 se muestra el esquema de conexionado entre las patillas RB7-RB4 y las líneas correspondientes del bus interno.

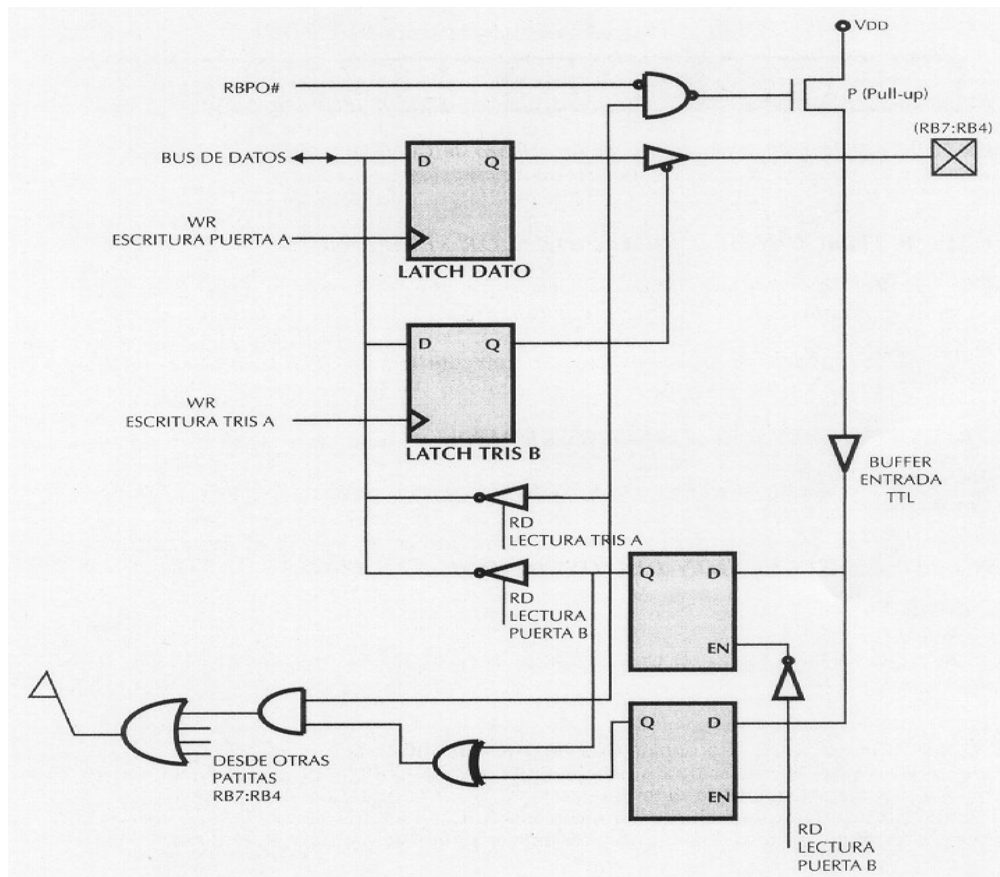


Figura II.13.- La Puerta B.

El estado de las patillas RB7-RB4 en modo de entrada se compara con el valor antiguo que tenían y que se había almacenado en un biestable durante la última lectura de la Puerta B. El cambio de estado en alguna de esas líneas origina una interrupción y la activación del señalizador RBIF.

La línea RB6 también se utiliza para la grabación serie de la memoria de programas y sirve para soportar la señal de reloj. La línea RB7 constituye la entrada de los datos en serie.

II.6.- La Palabra de CONFIGURACIÓN.

Se trata de una posición reservada de la memoria de programa situada en la dirección 2007 H y accesible únicamente durante el proceso de grabación. Al escribirse el programa de la aplicación es necesario grabar el contenido de esta posición de acuerdo con las características del sistema. En la Figura II.14 se muestra la distribución y asignación de los 14 bits de la Palabra de Configuración de los PIC16F84, que tienen las siguientes funciones:

CP	CP	CP	CP	CP	CP	CP	CP	CP	CP	PWRT E#	WDTE	FOSC 1	FOSC 0
----	----	----	----	----	----	----	----	----	----	------------	------	-----------	-----------

Figura II.14.- La Distribución y Asignación de los 14 Bits de la Palabra CONFIGURACIÓN de los PIC16F84.

CP: BITS DE PROTECCIÓN DE LA MEMORIA DE CÓDIGO

1: No protegida.

0: Protegida. El programa no se puede leer, evitando copias. Tampoco se puede sobrescribir. Además evita que pueda ser accedida la memoria EEPROM de datos y, finalmente, si se modifica el bit CP de 0 a 1, se borra completamente la EEPROM.

PWRT: ACTIVACIÓN DEL TEMPORIZADOR "POWER-UP"

El temporizador "power-up" retrasa 72 ms la puesta en marcha o Reset que se produce al conectar la alimentación al PIC, para garantizar la estabilidad de la tensión aplicada.

1: Desactivado

0: Activado

WDT: ACTIVACIÓN DEL PERRO GUARDIÁN

1: Activado el WDT

0: Desactivado

FOSC1-FOSC0: SELECCIÓN DEL OSCILADOR UTILIZADO

1-1: Oscilador RC

1-0: Oscilador HS

0-1: Oscilador XT

0-0: Oscilador LP

II.7.- La Memoria EEPROM de Datos.

Los PIC16X84 tienen 64 bytes de memoria EEPROM de datos, donde se pueden almacenar datos y variables que interesa que no se pierdan cuando se desconecta la alimentación al sistema. Soporta 1.000.000 de ciclos de escritura/borrado y es capaz de guardar la información sin alterarla más de 40 años. La memoria EEPROM no está mapeada en la zona de la memoria de datos donde se ubican los registros SFR y GPR. Para poder leerla y escribirla durante el funcionamiento normal del microcontrolador hay que utilizar 4 registros del banco SFR: EEDATA, EEADR y EECON1.

En el registro EEADR, ubicado en la dirección 9 del banco 0, se carga la dirección a acceder de la EEPROM de datos. las 64 posiciones de un byte ocupan las direcciones de un mapa que comienza en 00 H y termina en 3F H, por eso los 2 bits de más peso de este registro siempre valen 0. En el registro EEDATA, ubicado en la dirección 8 del banco 0, se depositan los datos que se leen o escriben. El registro EECON1, que ocupa la dirección 88 H de la memoria de datos, o la dirección 8 del banco 1, tiene misiones de control de las operaciones en la EEPROM y la distribución de sus bits se presenta en la Figura II.15, sirviendo cada uno de ellos para lo siguiente:

RD: Lectura

1: Se pone a 1 cuando se va a realizar un ciclo de lectura. Luego pasa a 0 automáticamente.

WR: Escritura

1: Se pone a 1 cuando se va a realizar un ciclo de ESCRITURA. Luego pasa a 0 automáticamente.

WREN: Permiso de escritura

1: Permite la escritura en la EEPROM.

0: Prohíbe la escritura.

WRERR: Señalizador de error en escritura

1: Se pone a 1 cuando una operación de escritura ha terminado prematuramente.

0: La operación de escritura se ha completado correctamente.

EEIF: Señalizador de final de operación de escritura

1: Cuando este señalizador se pone a 1 indica que la operación se ha completado con éxito.

Se pone a 0 por programa.

0: La operación de escritura no se ha completado.

-----	-----	-----	EEIF	WRERR	WREN	WR	RD
-------	-------	-------	------	-------	------	----	----

Figura II.15.- El Registro EECON1, que ocupa la Dirección 88 H de la Memoria de Datos, o la Dirección 8 del Banco 1, tiene Misiones de Control de las Operaciones en la EEPROM y la Distribución de sus Bits

II.8.- Interrupciones.

Las llamadas a las subrutinas mediante instrucciones del tipo CALL son desviaciones del flujo de control del programa originadas por instrucciones, por lo que se consideran síncronas. Se producen cada vez que se ejecuta dicha instrucción. Las interrupciones son desviaciones del flujo de control del programa originadas asincrónicamente por diversos sucesos que no se hallan bajo la supervisión de las instrucciones. Dichos sucesos pueden ser externos al sistema, como la generación de un flanco o nivel activo en una patilla del microcontrolador, o bien, internos, como el desbordamiento de un contador.

El comportamiento del microcontrolador ante la interrupción es similar al de la instrucción tipo CALL de llamada a subrutina. En ambos casos se detiene la ejecución del programa en curso, se salva la dirección actual del PC en la Pila y se carga el PC con una dirección, que en el caso de CALL viene acompañando a la misma instrucción, y en el caso de una interrupción es una dirección "reservada" de la memoria de código, llamada *Vector de Interrupción* que da paso a un trozo de código denominado Rutina de Servicio de la Interrupción (RSI).

La RSI suele comenzar guardando en la memoria de datos algunos registros específicos del procesador. Concretamente aquellos que la RSI va a emplear y va a alterar su contenido. Antes del retorno al programa principal se recuperan los valores guardados y se restaura completamente el estado del procesador. Algunos procesadores salvan estos registros en la Pila, pero los PIC no disponen de instrucciones para meter (*push*) y sacar (*pop*) información de la Pila, utilizando para este fin registros de propósito general de la memoria de datos.

Los PIC16X84 pueden ser interrumpidos por 4 causas diferentes, pero todas ellas desvían el flujo de control a la dirección 0004 H, por lo que otra de las operaciones iniciales de la RSI es averiguar cuál de las posibles causas ha sido la responsable de la interrupción en curso, para ello se exploran los señalizadores de las fuentes de interrupción. Otro detalle importante en la RSI de los PIC16X84 es que estos microcontroladores poseen un bit GIE (*Global Interrupt Enable*) que cuando vale "0" prohíbe todas las interrupciones.

Pues bien, al comenzar la RSI dicho bit GIE se pone automáticamente a 0, con objeto de no atender nuevas interrupciones hasta que se termine la que ha comenzado. En el retorno final de la interrupción, GIE pasa a valer automáticamente "1" para volver a tener en cuenta las interrupciones.

Antes del retorno conviene borrar el señalizador de la causa de interrupción que se ha atendido, porque si bien los señalizadores se ponen a "1" automáticamente en cuanto se produce la causa que indican, la puesta a 0 se hace por programa. En la Figura II.16 se muestra un organigrama de las fases más importantes que se desarrollan durante el proceso de ejecución de una interrupción.

II.8.1.- Causas de Interrupción.

Los PIC16X84 tienen 4 causas o fuentes posibles de interrupción:

- 1ª Activación de la patilla RB0/INT
- 2ª Desbordamiento del temporizador TMR0
- 3ª Cambio de estado en una de las 4 patillas de más peso (RB7-RB4) de la Puerta B
- 4ª Finalización de la escritura en la EEPROM de datos

Cuando ocurre cualquiera de los 4 sucesos indicados se origina una petición de interrupción, que si se acepta y se atiende comienza depositando el valor del PC actual en la Pila, poniendo el bit GIE = 0 y cargando en el PC el valor 0004 H, que es el Vector de Interrupción donde se desvía el flujo de control. Cada fuente de interrupción dispone de un señalizador o "flag", que es un bit que se pone automáticamente a "1" cuando se produce. Además cada fuente de interrupción tiene otro bit de permiso, que según su valor permite o prohíbe la realización de dicha interrupción.

II.8.2.- El Registro de Control de Interrupciones INTCON.

La mayor parte de los señalizadores y bits de permiso de las fuentes de interrupción en los PIC16X84 están implementados sobre los bits del registro INTCON, que ocupa la dirección 0B H del banco 0, hallándose duplicado en el banco 1. El significado de cada bit, que se muestra en la Figura II.16, es el siguiente:

GIE	EEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF
-----	------	------	------	------	------	------	------

Figura II.16.- muestra un organigrama de las fases más importantes que se desarrollan durante el proceso de ejecución de una interrupción.

GIE: Permiso Global de Interrupciones

0: Prohíbe todas las interrupciones.

1: Permite la ejecución de todas las interrupciones, cuyos bits de permiso individuales también las permitan.

EEIE: Permiso de Interrupción por fin de escritura en la EEPROM

0: Prohíbe que se produzca esta interrupción.

1: Permite que se origine esta interrupción cuando termina la escritura en la EEPROM de datos.

TOIE: Permiso de Interrupción por sobrepasamiento del TMR0

0: Prohíbe esta interrupción.

1: Permite una interrupción al desbordarse el TMR0.

INTE: Permiso de Interrupción por activación de la patilla RB0/INT

0: Prohíbe esta interrupción.

1: Permite la interrupción al activarse RB0/INT.

RBIE: Permiso de Interrupción por cambio de estado en RB7-RB4

0: Prohíbe esta interrupción
1: Permite esta interrupción.

T0IF: Señalizador de sobrepasamiento del TMR0

0: Indica que el TMR0 no se ha desbordado.
1: Toma este valor cuando ha ocurrido el desbordamiento.

INTF: Señalizador de activación de la patilla RB0/INT

0: Indica que RB0/INT aún no se ha activado.
1: Se pone a 1 al activarse RB0/INT.

RBIF: Señalizador de cambio de estado en las patillas RB7-RB4

0: No ha cambiado el estado de RB7-RB4.
1: Pasa a 1 cuando cambia el estado de alguna de esas líneas.

Siempre que se produzca una interrupción por cualquier causa, GIE=1 y el PC se carga con el valor 0004 H, que es el Vector de Interrupción. Para conocer que causa ha provocado la interrupción se exploran los señalizadores, tres de los cuales se ubican en el registro INTCON y el cuarto, EEIF, que se pone a 1 cuando finaliza la escritura en la EEPROM, se halla en el bit 4 del registro EECON1. Los señalizadores deben ponerse a 0 por programa antes del retorno de la interrupción y son operativos aunque la interrupción esté prohibida con su bit de permiso correspondiente.

11.8.3. Interrupción Externa INT.

Esta fuente de interrupciones es sumamente importante para atender los acontecimientos externos en tiempo real. Cuando ocurre alguno de ellos activa la patilla RB0/INT y se hace una petición de interrupción. Entonces, de forma automática, el bit INTF=1 y, si el bit de permiso INTE=1 se autoriza el desarrollo de la interrupción.

Mediante el bit 6, llamado INTDEG, del registro OPTION se puede seleccionar cual será el flanco activo en RB0/INT. Si se desea que sea ascendente se escribe un 1 en dicho bit, y si se desea descendente se escribe un 0. El procesador explora el señalizador INTF al final del primer ciclo de reloj de cada ciclo de instrucción. Recuérdese que cada ciclo de instrucción constaba de 4 ciclos de reloj: Q1, Q2, Q3 y Q4. Al terminar Q1 se exploran los señalizadores produciéndose un período de latencia de 3 ó 4 ciclos de instrucción desde el momento que hay un señalizador activado hasta que se inicializa la interrupción.

11.8.4.- Interrupción por Desbordamiento del TMR0.

Quando el TMR0 se desborda y pasa del valor FF H al 00 H, el señalizador T0IF se pone automáticamente a 1. Sí además, el bit de permiso de interrupción del TMR0 T0IE=1 y el bit de Permiso Global de Interrupciones GIE=1, se produce una interrupción.

CAPÍTULO III.

PROGRAMACIÓN DE LOS MICROCONTROLADORES, PIC.

III.1.- Instrucciones del PIC.

Como ya se ha visto, los PIC responden a la arquitectura RISC. Esta arquitectura contiene pocas instrucciones (35 en nuestro caso) simples y rápidas (la mayoría se ejecutan en un ciclo de instrucción); las instrucciones son ortogonales (las instrucciones pueden usar cualquier operando) y tienen una longitud constante (14 bits).

En la Tabla III.1, se pueden ver las 35 instrucciones diferentes que se pueden usar en la Programación del PIC 16F84. En la columna de sintaxis aparecen las siguientes letras:

- **f:** es el registro de la dirección (7 bits).
- **d:** si vale 1, el resultado de la instrucción se almacena en f. Si vale 0, va al acumulador (W).
- **b:** indica el numero de un bit dentro de un registro.
- **k:** es un valor inmediato (8 bits).
- **x:** indica que es un valor indeterminado.

En las últimas dos columnas, se indica el formato de la instrucción (14 bits) y los señalizadores que activa ésta (Bits). Para localizar mejor las instrucciones, se ha separado la Tabla para distribuir las instrucciones según su función (si manejan bits, registros, etcétera). La columna C indican la duración en ciclos de instrucción (en nuestro caso un ciclo dura 1 microsegundo) que tarda en ejecutarse una instrucción.

INSTRUCCIONES QUE MANEJAN REGISTROS				
SINTAXIS	OPERACIÓN	C	FORMATO (14 bits)	Bits
ADDWF f, d	W + f	1	00 0111 dfff ffff	C,DC,Z
ANDWF f, d	W AND f	1	00 0101 dfff ffff	Z
CLRF f	Borra f	1	00 0001 1fff ffff	Z
CLRW	Borra W	1	00 0001 0xxx xxxx	Z
COMF f, d	Invierte f	1	00 1001 dfff ffff	Z
DECF f, d	Decrementa f	1	00 0011 dfff ffff	Z
INCF f, d	Incrementa f	1	00 1010 dfff ffff	Z
IORWF f, d	W OR f	1	00 0100 dfff ffff	Z
MOVF f, d	Mueve f	1	00 1000 dfff ffff	Z
MOVWF f	Mueve W y f	1	00 0000 1fff ffff	
NOP	No operación	1	00 0000 0xx0 0000	
RLF f, d	Rota f a la izquierda	1	00 1101 dfff ffff	C
RRF f, d	Rota f a la derecha	1	00 1100 dfff ffff	C
SUBWF f, d	F – W	1	00 0010 dfff ffff	C,DC,Z
SWAPF f, d	Intercambia bits	1	00 1110 dfff ffff	
XORWF f, d	W XOR f	1	00 0110 dfff ffff	Z
INSTRUCCIONES QUE MANEJAN BITS				
SINTAXIS	OPERACIÓN	C	FORMATO (14 bits)	Bits
BCF f, b	Borra bit b de f	1	01 00bb bfff ffff	
BSF f, b	Pone a 1 bit b de f	1	01 01bb bfff ffff	
INSTRUCCIONES DE BRINCO				
SINTAXIS	OPERACIÓN	C	FORMATO (14 bits)	Bits
BTFSC f, d	Explora bit d de f y salta si es 0	1,2	01 10bb bfff ffff	
BTFSS f, d	Explora bit d de f y salta si es 1	1,2	01 11bb bfff ffff	
DECFSZ f, d	Decrementa f y si es 0 salta	1,2	00 1011 dfff ffff	
INCFSZ f, d	Incrementa f y si es 1 salta	1,2	00 1111 dfff ffff	

INSTRUCCIONES QUE MANEJAN OPERANDOS INMEDIATOS				
SINTAXIS	OPERACIÓN	C	FORMATO (14 bits)	Bits
ADDLW k	W + k	1	11 111x kkkk kkkk	C,DC,Z
ANDLW k	W AND k	1	11 1001 kkkk kkkk	Z
IORLW k	W OR k	1	11 1000 kkkk kkkk	Z
MOVLW k	Mueve a W el valor k	1	11 00xx kkkk kkkk	
SUBLW k	k – W	1	11 110x kkkk kkkk	C,DC,Z
XORLW k	W XOR k	1	11 1010 kkkk kkkk	
INSTRUCCIONES DE CONTROL Y ESPECIALES				
SINTAXIS	OPERACIÓN	C	FORMATO (14 bits)	Bits
CALL k	Llamada a subrutina	2	10 0kkk kkkk kkkk	TO,PD
CLRWDT	Borra o refresca el perro guardián	1	00 0000 0110 0100	
GOTO k	Salto incondicional a k	2	10 1kkk kkkk kkkk	
RETFIE	Retorno de interrupción	2	00 0000 0000 1001	
RETLW k	Retorno de subrutina y W=k	2	11 01xx kkkk kkkk	
RETURN	Retorno de subrutina	2	00 0000 0000 1000	
SLEEP	Pasa al modo de reposo	1	00 0000 0110 0011	TO,PD

Tabla III.1.- Juego de instrucciones del PIC16F84.

Por último, se observa más detenidamente el funcionamiento de las instrucciones representadas en la tabla anterior.

- **ADDLW k**: Añade el contenido de W al literal k y almacena el resultado en W. Esta instrucción activa los bits C, DC y Z del registro de estado. Se ejecuta en un ciclo de instrucción (todas se ejecutan en el mismo tiempo a menos que se especifique otro valor).

- **ADDWF f, d**: Añade el contenido de W al de f, y almacena el resultado en W si d=0, y en f si d=1. Activa los mismos señalizadores que la anterior.

- **ANDLW k**: Efectúa un AND lógico entre el contenido de W y el literal k, y almacena el resultado en W. Si el resultado es nulo, activa el bit Z.

- **ANDWF f, d**: Efectúa un AND lógico entre el contenido de W y el de f y coloca el resultado en W (d=0) o en f (d=1).

- **BCF f,b**: Pone a cero el bit número b de f.

- **BSF f,b**: Pone a uno el bit número b de f.

- **BTFSC f, d**: Si el bit número b de f es nulo, la instrucción que sigue a ésta se ignora y se trata como un NOP. En este caso (solo en este caso), la instrucción precisa de dos ciclos para ejecutarse.

- **BTFSC f, d**: Si el bit número b de f es 1, la instrucción que sigue a ésta se ignora y se trata como un NOP. En este caso (solo en este caso), la instrucción precisa de dos ciclos para ejecutarse.

- **CALL k**: Salvaguarda la dirección de vuelta en la pila y después llama a la subrutina situada en la dirección cargada en el PC. Los pasos que sigue para su ejecución son: PC + 1 → pila, k → PC (0 – 10), PCLATH (3,4) → PC (11,12)

- **CLRF f**: Pone el contenido de f a cero y activa el bit Z.

- **CLRW**: Pone el registro W a cero y activa el bit Z.

- **CLRWDT**: Pone a cero el registro contador del temporizador *Watchdog* (WDT), así como el previsor. Pone a 1 los bits TO y PD.

- **COMF f, d:** Hace el complementario de f bit a bit. El resultado se almacena de nuevo en f si d=1 y en W si d=0 (en este caso, f no varía). Si el resultado es nulo, activa Z.

- **DECF f, d:** Decrementa el contenido de f en una unidad. El resultado lo almacena de nuevo en f (d=1) o en W (d=0), no variando f en este último caso. También activa Z si el resultado es nulo.

- **DECFSZ f, d:** Decrementa el contenido de f en una unidad. El resultado se almacena según el valor de d. Si el resultado es nulo, se ignora la siguiente instrucción y, en ese caso, esta instrucción dura dos ciclos.

- **GOTO k:** Llama a la subrutina situada en la dirección cargada en el PC. La ejecución de la instrucción sería: $k \rightarrow PC (0-10)$, $PCLATH (3,4) \rightarrow PC (11,12)$

- **INCF f, d:** Incrementa el contenido de f en una unidad. El resultado se almacena de nuevo en f o W (en este caso f no varía). Activa Z si el resultado es nulo.

- **INCFSZ f, d:** Incrementa el contenido de f en una unidad. El resultado se almacena en f (d=1) o W (d=0 y f no varía). Si el resultado es nulo, se ignora la siguiente instrucción y, en este caso, esta instrucción dura dos ciclos.

- **IORLW k:** Efectúa un OR lógico inclusivo entre el contenido de W y el literal k, y almacena el resultado en W. Si la operación da cero, se activa el bit Z.

- **IORWF f, d:** Efectúa un OR lógico inclusivo entre el contenido de W y el contenido de f, y almacena el resultado en W (d=0) o en f. Si la operación da cero, se activa el bit Z.

- **MOVF f, d:** Desplaza el contenido de f a f si d=1 ó a W si d=0. El desplazamiento de f a f permite comprobar el contenido de f con respecto a cero, ya que esta instrucción actúa sobre el bit Z.

- **MOVLW k:** Carga W con el literal k.

- **MOVWF f:** Carga f con el contenido de W.

- **NOP:** Sólo consume tiempo de máquina, un ciclo en este caso.

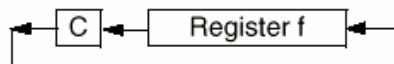
- **RETFIE:** Carga el PC con el valor que se encuentra en la parte superior de la pila, asegurando así la vuelta de la interrupción. Pone a 1 el bit GIE con el fin de autorizar de nuevo que se tengan en cuenta las interrupciones.

- **RETLW k:** Carga W con el literal k, y después carga el PC con el valor que se encuentra en la parte superior de la pila, efectuando así un retorno de subrutina.

- **RETURN:** Carga el PC con el valor que se encuentra en la parte superior de la pila, efectuando así una vuelta de subrutina. Es como la anterior pero simplificada.

- **RLF f, d:** Rotación de un bit a la izquierda del contenido de f, pasando por el bit de acarreo C (ver Figura III.1). Si d=1 el resultado se almacena en f, si es 0 se almacena en W.

Figura III.1.- Desplazamiento a la Izquierda



- **RRF f, d:** Rotación de un bit a la derecha del contenido de f, pasando por el bit de acarreo C (es como en la figura 5.1 pero con las flechas hacia la derecha). Si d=1 el resultado se almacena en f, si es 0 se almacena en W.

- **SUBLW k:** Sustraer el contenido de W del literal k ($k - W$) y almacena el resultado en W. La sustracción se realiza en complemento a dos.

- **SUBWF f, d:** Sustraer el contenido de W del contenido de f ($f - W$) y almacena el resultado en W o f (según valor de d). La sustracción se realiza en complemento a 2. Tanto esta como la anterior activan los bits C, DC, y Z según el caso.

- **SWAPF f, d:** Intercambia los cuatro bits de mayor peso con los cuatro de menor peso de f y almacena el resultado en f o W (según d).

- **XORLW k:** Efectúa un OR lógico exclusivo entre el contenido de W y el literal k, y almacena el resultado en W.

- **XORLW f, d:** Efectúa un OR lógico exclusivo entre el contenido de W y el contenido de f, y almacena el resultado en f o W (según d). Tanto esta como la anterior activa el bit Z si se da el caso.

- **SLEEP:** Pone el circuito en modo SLEEP y para el oscilador. El procesador queda congelado, no ejecutando instrucciones y manteniendo el mismo valor en las puertas de E/S. Pone a cero el WDT (aunque sigue funcionando), el predivisor y PD; también pone a 1 el bit TO. Para que el micro salga del modo SLEEP, se tiene que dar una de estas situaciones: que la patilla MCLR se active externamente (reset), que se desborde el WDT (en caso de que estuviera activado al entrar en el modo reposo) y que se de una interrupción (como TMR0 está desactivado, solo podrá producirse una de los otros tipos).

El sistema μ pic trainer consiste en un equipo completo para la evaluación para aplicaciones basadas en los microcontroladores PIC de gama media. Dispone de una serie de periféricos básicos de E/S con los que poder verificar el funcionamiento de una aplicación, así como la circuitería necesaria para la grabación de todos los modelos del PIC de gama media de 18 y 28 patillas.

III.2.- Arquitectura del μ PIC Trainer.

En la Figura III.2, se puede ver el aspecto externo de la placa. En ella aparecen representados los elementos más importantes mediante las siguientes notas:

(1): Dos zócalos para insertar cualquier microcontrolador de gama media tanto de 18 (abajo) como de 28 patillas (arriba).

(2): Cristal de cuarzo junto con dos condensadores. Nos proporciona la frecuencia de trabajo de 4 MHz.

(3): Pulsador que genera un reset en el PIC.

(4): 5 conmutadores correspondientes a 5 entradas digitales.

(5): 4 potenciómetros correspondientes a las 4 entradas analógicas.

(6): Jumpers (J8) para selección de entradas analógicas (jumpers arriba), digitales (abajo) o para desactivar las entradas al puerto A.

(7): Conector de expansión PIC-BUS.

(8): Barra de leds conectada a la puerta B. Arriba está su pack de resistencias.

(9): Display 7 segmentos conectado a puerta B.

(10): Jumpers (J5, J6, J7) para la activación del Display, los leds y el LCD.

(11): Módulo de visualización LCD.

(12): Potenciómetro de ajuste (P1) para el LCD.

(13): Conector DB25 para conexión con el canal paralelo de cualquier PC.

(14): Led (D5) que indica la presencia de la tensión de programación (V_{pp}).

(15): Capacitores (C1 y C2) y puente rectificador (D1) para transformar la tensión alterna (12VAC) en continua (el transformador se conecta por arriba).

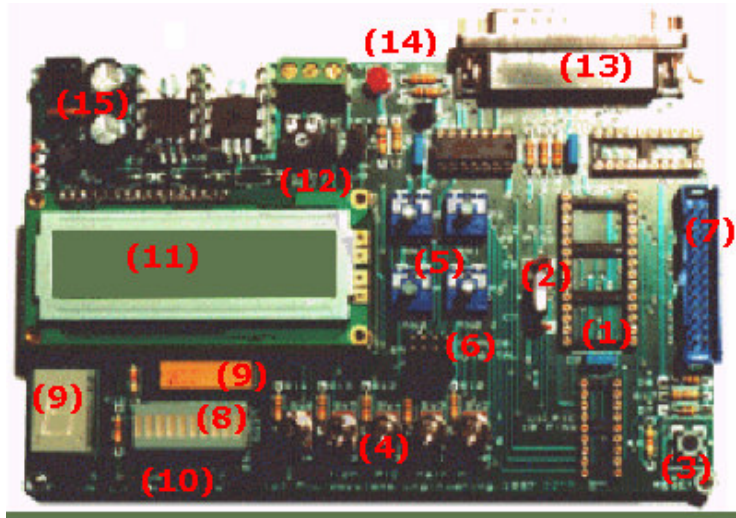


Figura III.2.- Aspecto Exterior del μ PIC Trainer.

En la Figura III.3, aparecen estas referencias pero según su distribución en el esquemático de la placa. Las características del μ PIC Trainer son las que se explican a continuación:

- **Alimentación** desde un transformador de 12VAC conectado a J1. El circuito de estabilización está incluido en la placa y nos da la tensión general de +5V (Vcc) y la de grabación de 13,8V. La tensión alterna proporcionada por el transformador, se rectifica mediante el puente de diodos D1 y se filtra mediante los condensadores C1 y C2 (obteniéndose así una tensión continua sin estabilizar). La tensión obtenida se estabiliza a +5V mediante el regulador U6 (μ A7805) y a +13,8V con el regulador μ A7812 junto con los tres diodos D2-D4.

- Soporta cualquier **Microcontrolador** de gama media tanto de 18 como de 28 patillas; esta es la razón de la existencia de dos zócalos en la placa (ver Figura III.2). La frecuencia de trabajo es de 4 MHz, que se puede modificar cambiando el cristal de cuarzo y los valores de C6 y C7.

- Generación automática de **Reset** y manual, mediante un pulsador de la placa (ver Figura III.2).

- Conector de expansión **PIC-BUS**. Se trata de un conector de 26 contactos en el que están disponibles las señales de las puertas E/S además de la alimentación y MCLR.

- Posee **5 Entradas Digitales** implementadas mediante 5 conmutadores (SW2-SW6) y **4 Entradas Analógicas** variables mediante los potenciómetros P2-P5 (nos dan una tensión variable de 0-5V). Están conectados a las líneas RA0-RA4 de la puerta A. Los conmutadores son capaces de introducir niveles lógicos 1 y 0 por esas líneas. Estas pueden programarse para actuar de diferentes formas (RA0-RA3 pueden actuar como entradas analógicas o digitales y RA4 como entrada exterior de pulsos para el timer 0 o como entrada digital). Los jumpers J8(0)-J8(3) seleccionan individualmente si por RA0-RA3 se van a introducir entradas digitales (interruptores SW2-SW5), analógicas (potenciómetros P2-P5) o si se van a quedar desconectadas para usarlas con otros periféricos. Con el jumper J9 se selecciona si RA4 actúa como entrada digital (SW6), como señal de reloj (aplicada desde el exterior mediante conector J2(1)) o si está desconectada.

- Posee **8 Salidas Digitales** conectadas a 8 indicadores luminosos tipo LED y/o a Display 7 segmentos. Están conectadas a la puerta B y consisten en una barra de diodos leds luminosos (D8) que representan el estado lógico de las señales RB0-RB7 y en un Display numérico de 7 segmentos. Están conectados entre sí en paralelo a la puerta B; tanto los leds como el Display disponen de los jumpers J7 y J5 respectivamente con los que se les puede desconectar si la aplicación no los necesita. La barra de leds es de 10, ocho de los cuales van a las puertas, uno no se utiliza y el décimo nos indica la presencia de tensión +5V (Vcc ON). Los leds están conectados al pack de resistencias RPACK 1 (8 resistencias de 330Ω que actúan como limitadoras). La línea RB0 también puede actuar como entrada de interrupción INT (aplicada mediante conector J2(3)). Con el jumper 10 la señal RB0 se conecta a los leds y al Display.

- Tiene conectado un módulo de visualización tipo **LCD** (Liquid Cristal Display) **de 2x16** (dos líneas de 16 caracteres cada una) y control de contraste mediante potenciómetro (P1). Las 8 líneas de datos del LCD (D0-D7) están conectadas a la puerta B; por ella se aplican los diferentes códigos de control para realizar los distintos efectos de visualización en el LCD (en este caso las puertas actúan como salida); en algunos casos las puertas B deben actuar como entradas para comprobar el estado interno del LCD. Mediante el jumper J6 activamos o desactivamos el LCD.

- Contiene un **Circuito Grabador** que nos permite (con el software de control necesario) grabar el PIC insertado en uno de los zócalos (el PIC podrá ser grabado, borrado, ... desde el mismo zócalo de trabajo). Mediante el conector J3 se conecta al canal paralelo de cualquier ordenador personal. El circuito integrado U3 se encarga de verificar el hardware del µpic trainer y la comunicación entre este y el PC. Los bits de datos se leen o escriben en serie a través de RB7 y son aplicados al puerto paralelo mediante los amplificadores U4D y U4F. La señal RB6 (entrada de reloj para sincronizar la lectura/grabación) es generada por el software de control y se aplica por el canal paralelo a través del amplificador U4C. El software también genera una señal que, mediante U4B, se aplica al transistor de conmutación Q1 para que envíe los 13,8V a la patilla MCLR/Vpp del pic; al mismo tiempo el diodo led D5 se ilumina indicando la presencia de esta tensión. Finalmente, a través de U4A, el software genera la señal MCLR que resetea al micro. Durante la lectura/escritura, hay que tener desconectados los jumpers J5, J6 y J7 para que no se altere la información leída o grabada.

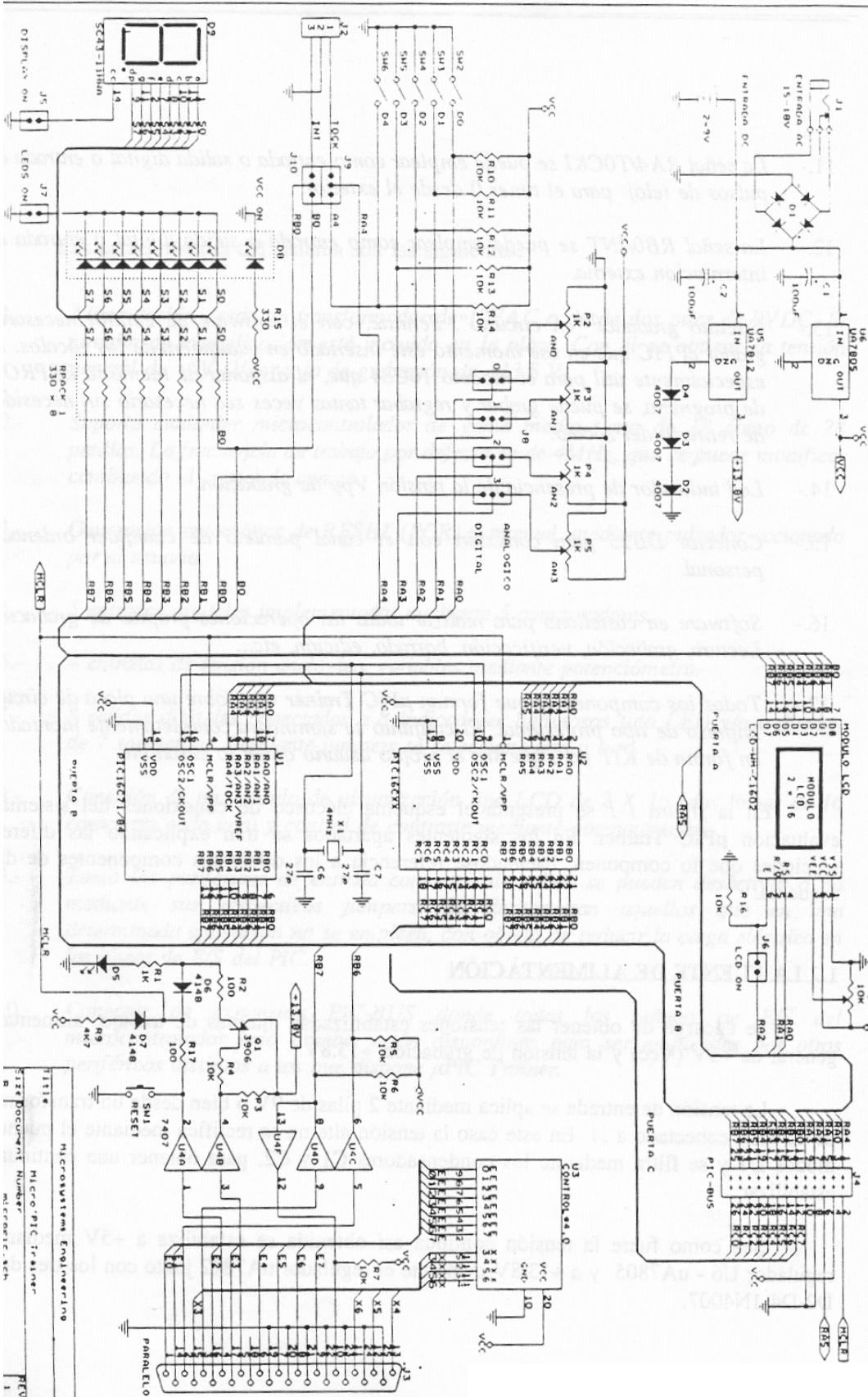


Figura III.3.- Esquema Eléctrico del μ PIC Trainer.

III.2.1.- Componentes del μ PIC Trainer.

El μ PIC Trainer está montado sobre una placa de circuito impreso tamaño *Eurocard* de 100x160 de doble cara. En la Figura III.4 se presenta la disposición y orientación de los componentes sobre la placa de circuito impreso.

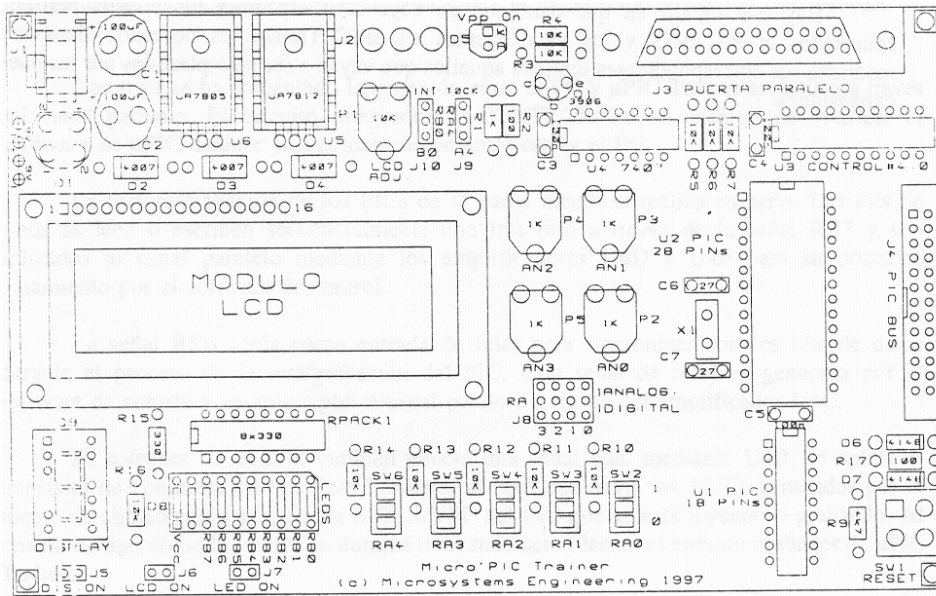


Figura III.4.- Disposición de Componentes en la Placa.

En la Tabla III.2, se presentan los valores de los distintos componentes utilizados en la placa (ver Figura III.4).

CIRCUITOS INTEGRADOS		DIODOS	
U1	PIC de 18 patillas	D1	Puente rectificador de 1 ^a
U2	PIC de 28 patillas	D2-D4	Diodos 1N4007
U3	Circuito de control	D5	Diodo luminoso tipo led rojo
U4	Amplificador SN7407	D6-D7	Diodos 1N4148
U5	Estabilizador de tensión (12V) 7812	D8	Barra de 10 diodos led
U6	Estabilizador de tensión (5V) 7805	D9	Display 7 segmentos cátodo común
RESISTENCIAS		CONDENSADORES	
R1	1K – ¼ W	C1-C2	Electrolítico de 100µRF / 63V
R2	100 – ¼ W	C3-C5	100nF
R3-R7	10K – ¼ W	C6-C7	27pF
R9	4K7 – ¼ W	OTROS	
R10-R14	10K – ¼ W		
R15	330 ¼ W	X1	Cristal de cuarzo 4MHz
R16	10K – ¼ W	SW1	Pulsador de reset
R17	100 – ¼ W	SW2-6	Conmutadores de palanca
RPACK1	Pack de 8 resistencias DIL 330	Q1	Transistor PNP 3906
P1	Resistencia ajustable 10K		
P2-P5	Resistencias ajustables 1K		
JUMPERS			
J1	Base de alimentación para entrada de corriente alterna	J3	Conector DB25 macho con codo de bajo perfil para c. impreso
J2	Borna de 3 contactos para circuito impreso de paso 5.08	J4	Conector 26 contactos (2x13), macho recto

Tabla III.2.- Lista de Materiales Empleados en el µPIC Trainer.

En la placa también se han utilizado otros materiales como zócalos, cables, tornillos, refrigeradores (2 para los circuitos de estabilización), ejes de potenciómetros, módulo LCD,...

En la Tabla III.3, se ve el resultado obtenido según la posición ocupada por los distintos jumpers.

JUMPER	POSICIÓN	DESCRIPCIÓN
J5	Abierto	Display D9 desconectado
	Cerrado	Display D9 conectado a la puerta B
J6	Abierto	Módulo LCD desconectado
	Cerrado	Módulo LCD conectado a la puerta B
J7	Abierto	Barra de leds D8 desconectada
	Cerrado	Barra de leds D8 conectada a la puerta B
J8(0)	Abierto	Línea RA0 sin conexión
	Analógico	Línea RA0 entrada analógica AN0
	Digital	Línea RA0 entrada digital
J8(1)	Abierto	Línea RA1 sin conexión
	Analógico	Línea RA1 entrada analógica AN1
J8(2)	Digital	Línea RA1 entrada digital
	Abierto	Línea RA2 sin conexión
	Analógico	Línea RA2 entrada analógica AN2
J8(3)	Digital	Línea RA2 entrada digital
	Abierto	Línea RA3 sin conexión
	Analógico	Línea RA3 entrada analógica AN3
J9	Digital	Línea RA3 entrada digital
	Abierto	Línea RA4 sin conexión
	TOCKI	Línea RA4 entrada de reloj para el Timer 0
J10	A4	Línea RA4 entrada digital
	Abierto	Línea RB0 sin conexión
	INT	Línea RB0 entrada de interrupción externa
	B0	Línea RB0 salida digital

Tabla III.3.- Posición de los Jumpers.

III.2.2.- El Módulo de Visualización LCD.

Se trata de un módulo microcontrolado capaz de representar 2 líneas de 16 caracteres cada una. A través de 8 líneas de datos se le envía el carácter ASCII que se desea visualizar así como ciertos códigos de control que permiten realizar diferentes efectos de visualización. Mediante estas líneas, el módulo devuelve información sobre su estado interno. Contiene otras tres líneas para controlar el flujo de información. Las señales empleadas en el módulo LCD son las representadas en la Tabla III.4, así como su número de terminal correspondiente.

La interfase entre el módulo y el μ PIC Trainer se realiza uniando las líneas D0-D7 del LCD con la puerta B del pic; de igual forma, la señal RS se conecta a RA0, la señal R/W se conecta a la RA1 y la señal E a RA2. Tanto RA0 como RA2 deben programarse como salidas, mientras que RA1 será entrada o salida, según si se lee o se escribe.

PIN Nº	SÍMBOLO	DESCRIPCIÓN
1	Vss	Patilla de tierra de alimentación
2	Vdd	Patilla de alimentación de +5V
3	Vo	Patilla de contraste del cristal líquido.
4	RS	RS=0 Selección del registro de control o RS=1 de datos
5	R / W	R/W=0 el LCD es escrito y R/W=1 es leído
6	E	E=0 módulo desconectado o E=1 conectado
7 – 14	D0 – D7	Bus de datos Bi-direccional.

Tabla III.4.- Terminales (Pines) del Módulo LCD.

El juego de instrucciones permite configurar el LCD según los efectos que se quiera. En la Tabla III.5, se ven los valores que toman las líneas para darnos una estas instrucciones:

RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	INSTRUCCIÓN
0	0	0	0	0	0	0	0	0	1	Clear Display
0	0	0	0	0	0	0	0	1	X	Home
0	0	0	0	0	0	0	1	I/D	S	Entry mode set
0	0	0	0	0	0	1	D	C	B	Display ON/OFF
0	0	0	0	0	1	S/C	R/L	X	X	Cur. or Dis. shift
0	0	0	0	1	DL	N	F	X	X	Set
0	0	0	1	Dirección de la CG RAM					Set address CG R	
0	0	1	Dirección de la DD RAM					Set address DD R		
0	1	BF	Dirección de la CG RAM o DD RAM					Read busy flag		
1	0	Código ASCII o byte del carácter gráfico							Write data to Ram	
1	1	Código ASCII o byte del carácter gráfico							Read data de Ram	
ABREVIATURAS EMPLEADAS										
S	Si es 1 desplaza la visualización al escribir un dato. Si es 0 modo normal									
I/D	Si es 1 el cursor se va incrementando, si es 0 decrementa									
S/C	Si es 1 desplaza el Display, si es 0 mueve el cursor									
R/L	Si es 1 desplazamiento a la derecha, si es 0 a la izquierda									
BF	Si es 1 el módulo está ocupado, si es 0 está disponible									
DL	Si es 1 el bus de datos es de 8 bits, si es 0 será de 4 bits									
N	Si es 1 el LCD usa las dos líneas, si es 0 usa una									
F	Si es 1 el carácter es de 5x10 puntos, si es 0 es de 5x7									
B	Si es 1 el cursor parpadea									
C	Si es 1 el cursor está activado									
D	Si es 1 el Display está conectado									
X	Valor indeterminado									

Tabla III.5.- Juego de Instrucciones del LCD y Descripción de Palabras utilizadas.

A continuación se va a explicar brevemente el uso de cada una de las instrucciones representadas en la tabla anterior.

- **Clear Display:** Borra el módulo LCD y coloca el cursor en la primera posición (dirección 0). Pone el bit I/D (ver Tabla III.5) a 1 por defecto.

- **Home:** Coloca el cursor en la posición inicio y hace que el Display comience a desplazarse desde la posición original. Tanto esta instrucción como la anterior, tardan 1.64 msg en ejecutarse, al contrario que todas las demás que tardan 40 µsg.

- **Entry Mode Set:** Establece la dirección de movimiento del cursor y especifica si la visualización se va desplazando a la siguiente posición de la pantalla o no.

- **Display ON/OFF Control:** Activa o desactiva poniendo en ON/OFF tanto el Display como el cursor y si este debe o no parpadear.

- **Cursor or Display Shift:** Mueve el cursor y desplaza el Display sin cambiar el contenido de la memoria de datos de visualización DD RAM.

- **Set:** Establece el tamaño de interfase con el bus de datos, número de líneas del Display y tipo de carácter.

- **Set the CG RAM address:** el LCD, además de tener definidos todo el conjunto de caracteres ASCII, permite definir 4 u 8 caracteres gráficos. La composición de estos caracteres se va guardando en una memoria llamada CG RAM con capacidad para 64 bytes. Cada carácter definido se compone de 16 u 6 bytes que se almacenan en sucesivas posiciones de la CG RAM. Mediante esta instrucción se establece la dirección de memoria CG RAM a partir de la cual se irán almacenando los bytes que definen un carácter gráfico.

- **Set the DD RAM Address:** Los caracteres o gráficos que se van visualizando, se van almacenando previamente en una memoria llamada DD RAM para de aquí pasar a pantalla. Mediante esta instrucción se establece la dirección de memoria DD RAM a partir de la cual irán almacenando los datos a visualizar. Las direcciones de la 80h a la 8Fh corresponden a los 16 caracteres de la primera línea del LCD y de la C0h a la CFh al segundo.

- **Read Busy Flag&Address:** Esta instrucción informa del estado del flag BUSY además de proporcionar el valor del contador de direcciones de la CG o DD RAM (según la última que se haya empleado). El flag BUSY indica si se está ejecutando una instrucción (módulo ocupado) o no.

- **Write Data to CG or DD RAM:** Escribe en la memoria DD RAM los datos que se quieren presentar en pantalla. Igualmente se escribe en la memoria CG RAM los diferentes bytes que permiten confeccionar caracteres gráficos. El escribir en una u otra depende de si se ha usado antes la instrucción de direccionamiento DD o CG RAM.

- **Read Data from CG or DD RAM:** Igual que la anterior pero leyendo. En la Figura III.6, se muestran todos los caracteres gráficos disponibles para el LCD. Por ejemplo, para que salga la "A", tendremos que introducir por las líneas de datos el código 41h (ver Figura III.6).

		4 bits de más peso del carácter (hex)																
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
4 bits de menos peso del carácter	0	CG RAM (1)	00P`P -03xp															
	1	CG RAM (2)	!100a4 07#43q															
	2	CG RAM (3)	"2BRbr riyxpe															
	3	CG RAM (4)	#3C5ca 107E3															
	4	CG RAM (5)	\$4DTdt \Ibtpo															
	5	CG RAM (6)	%5EUeu 0at1c0															
	6	CG RAM (7)	&6FUfu 002apz															
	7	CG RAM (8)	'7GWgw 7#39gr															
	8	CG RAM (1)	(8HXhx 1070r															
	9	CG RAM (2)	>9IYiy 0710'y															
	A	CG RAM (3)	*#JZjz 000kjp															
	B	CG RAM (4)	+;KkK< 09E0* 5															
	C	CG RAM (5)	, <L#11 02700P															
	D	CG RAM (6)	-3M1m) 0202t÷															
	E	CG RAM (7)	.>N^n+ 0200n															
	F	CG RAM (8)	/?O_0+ 00700															

Figura III.6.- Juego de Caracteres ASCII del Módulo LCD.

Las posiciones marcadas como CG RAM (1-8) corresponden a los caracteres gráficos definidos por el usuario.

En la Tabla III.7, se muestra un ejemplo de configuración del carácter "A". En este caso se ha usado un carácter 5x7 que necesita 8 octetos en la CG RAM para ser definido (si fuera 5x10 necesitaría 16). Los octetos son introducidos en las 8 primeras posiciones de la CG RAM (si fuera 5x10 serían 16). Cada bit de cada octeto que valga 1 implica que su correspondiente pixel se active en el LCD. Si es el primer carácter creado de la CG RAM, se seleccionará aplicando el código 00h en la DD RAM como si fuera cualquier otro código ASCII. Si siguiéramos creando caracteres, estos se cargarían con las direcciones 01h, 02h, 03h,..., 0Fh de la DD RAM.

Código del carácter DD RAM								Dirección de la CG RAM						Patrones del carácter CG RAM																																																																							
7	6	5	4	3	2	1	0	5	4	3	2	1	0	7	6	5	4	3	2	1	0																																																																
0 0 0 0 0 0 0 0								0 0 0						<table border="1"> <tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr> </table>								1	1	1	1	1	1	0	0	1	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0	1	1	1	1	1	1	0	0	1	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	0	0																																																																														
1	0	0	0	0	1	0	0																																																																														
1	0	0	0	0	1	0	0																																																																														
1	1	1	1	1	1	0	0																																																																														
1	0	0	0	0	1	0	0																																																																														
1	0	0	0	0	1	0	0																																																																														
1	0	0	0	0	1	0	0																																																																														
0	0	0	0	0	0	0	0																																																																														

Tabla III.7.- Generación del Carácter Gráfico "A"

El módulo LCD ejecuta automáticamente la siguiente secuencia de inicio interna en el instante de aplicarle la tensión de alimentación (esta debe estabilizarse de 0.2V hasta los 4.5V en un tiempo comprendido entre 0.1 mseg y 10 mseg; igualmente el tiempo de desconexión debe ser como mínimo de 1 mseg antes de volver a conectar):

- **Clear Display:** El Flag BUSY se mantiene a 1 durante 15 mseg hasta que finaliza la inicialización.
- **Set:** Bus de Datos = 8 y Reglones del Display = 1.
- **Display ON/OFF Control:** Display OFF, cursor OFF y parpadeo OFF.
- **Entry Mode Set:** Incremento del cursor y modo normal del Display.
- Se selecciona la **Primera Posición** de la DD RAM.

Si no se satisfacen las condiciones de alimentación, la secuencia de inicialización habría que realizarla por software.

Para terminar, en la Tabla III.8, se presentan los tiempos de las señales recomendados para el módulo LCD utilizado en la placa. En la Figura III.7, se presenta el diagrama de tiempos (con los tiempos representados en la Tabla III.8) para escritura o lectura.

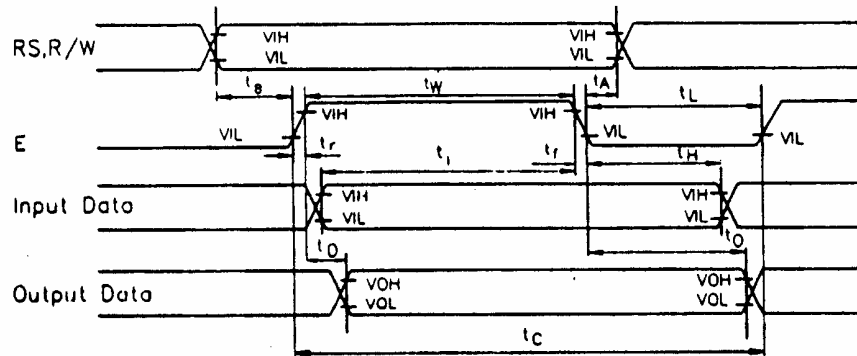


Figura III.7.- Diagrama de Tiempos para la Lectura/Escritura.

PARÁMETRO	SÍMBOLO	TIEMPO (nsg)
Enable cycle time	t_C (min)	1000
Enable pulse Width	t_W (min)	450
	t_L (min)	450
E. Raise time	t_r (max)	25
E. Fall Time	t_f (max)	25
Set-up time	t_B (min)	140
Data set-up time	t_l (min)	195
Data delay time	t_D (max)	320
Address hold time	t_A (max)	10
Hold time	t_H (min)	10
	t_O (min)	20

Tabla III.8.- Tiempos Recomendados.

III.2.3.- Software Recomendado.

El software recomendado para la programación de los PIC, es el **MPASM** y el **PIME-TR**. Los pasos a seguir a la hora de realizar y grabar un programa en el PIC son los que siguen:

- En primer lugar, se escribe el programa utilizando cualquier editor de texto con la particularidad de ser grabado con extensión ASM. El editor de MS-DOS nos puede servir para la escritura del programa.

- Tras ello, se compila el programa utilizando el archivo MPASM.EXE. El programa únicamente nos pide la ubicación exacta del fichero a compilar; tras seleccionar los ficheros deseados a la salida (archivo.ERR, archivo.LIST,...) y el tipo de PIC se pulsa F10 y se obtiene el archivo compilado con extensión HEX. Si el programa tiene errores en la compilación, estos aparecen en el archivo .ERR (si se ha seleccionado su salida). Otro fichero de interés es el .LST, el cual contiene el listado completo de la compilación (tanto este como el .ERR están seleccionados por defecto).

- Por último queda cargar el archivo PICME-TR.EXE. El programa funciona con ventanas y es muy simple; a la derecha se seleccionan las distintas opciones de programación (tipo de PIC, tipo de oscilador,...) y a la izquierda nos aparece el menú para abrir el archivo (con extensión HEX) y programarlo; desde este menú de la izquierda, también podemos leer el PIC, borrarlo y verificarlo. Mediante mensajes el programa nos informará de cómo ha ido la programación.

CAPÍTULO IV.

APLICACIONES DE LOS MICROCONTROLADORES.

IV.1.- Introducción.

En este capítulo se desarrollan algunas de las aplicaciones importantes de los Microcontroladores. De igual forma, se establece al final, las Líneas de Investigación más importantes en este campo de los Microcontroladores y los PIC.

IV.2.- Aplicaciones de los Microcontroladores.

IV.2.1.- Domótica: Comunicaciones por Red Eléctrica.

El Primer ejemplo de aplicación es el llamado: **“Domótica: Comunicaciones por Red Eléctrica”**.

El término Domótica procede de la palabra francesa *Domotique*; que se define como: *El uso de la Tecnología como medio para satisfacer las necesidades básicas de Seguridad, Comunicación, Gestión Energética y Confort del hombre y de su entorno más cercano”*.

Dado que las aplicaciones en el campo de la Domótica son prácticamente ilimitadas, los productos existentes en el mercado son muy variados, y tiene prestaciones muy avanzadas. Este apartado, no pretende centrarse en la implantación de funciones de regulación complejas, sino en el aspecto que suele considerarse más importante: el de las comunicaciones entre módulos por medio del empleo de corrientes portadoras.

En la actualidad, existen diversos sistemas de comunicación domóticos, los cuales se clasifican dependiendo del canal utilizado para dicha transmisión.

1.- *Sistemas Tradicionales* donde Emisor y Receptor están unidos físicamente: estas soluciones requieren infraestructura previa, debiendo ser previstos con anterioridad a la construcción del edificio, de forma que sea posible la introducción del cableado necesario. Un sistema estandarizado (Normalizado) que utiliza esta Tecnología es el EIB (*European Instalation Bus*), el cual permite conectar hasta 11520 elementos, que se comunican entre sí a través de dos únicos hilos.

2.- *Sistemas por Radio Frecuencia*, estos métodos presentan como ventaja el hecho de que no necesitan cableado. Sin embargo, tienen como inconveniente la limitación en el alcance, debido a su elevada susceptibilidad a las perturbaciones del medio o simplemente causados por la propia distribución de las paredes del edificio.

3.- *Sistemas basados en Corrientes Portadoras*, permiten comunicar los diversos elementos utilizando las líneas eléctricas existentes como soporte de información.

Esta última técnica, que apenas ha sido utilizada en Europa (2006), está actualmente muy extendida en aplicaciones Domóticas en los Estados Unidos de América, de forma que los buenos resultados que allá se han obtenido, hacen presagiar una pronta implantación a nivel mundial.

El interés de este sistema (ver la siguiente información) es evidente: la gratuidad del soporte, los numerosos accesos y salidas, la transmisión digital, la facilidad de instalación y traslado, etcétera.

- ✚ Las comunicaciones con corrientes portadoras no requieren el uso de cableado externo, con lo que el precio de la instalación será menor y el montaje de la misma, mucho más sencillo.
- ✚ Los nodos de comunicación pueden ser “pinchados” en cualquier toma de corriente sin interferir en el funcionamiento del resto.

- ✚ Las magnitudes captadas por los sensores son enviadas por la red en formato digital, lo que consigue una cierta inmunidad ante el ruido y ante la presencia de armónicos en la red.
- ✚ En los sistemas basados en portadoras, existen Protocolos estandarizados (Normalizados) como por ejemplo X.10; con lo que la mayoría de las acciones a realizar, tramas de transmisión, etcétera, ya están definidas.

1.- **Fundamentos de X.10.**- El formato de codificación X.10, es un Estándar (Norma) de transmisión por corrientes portadoras (*Power Line Carrier*) que, como tal, permite que todos los equipos que lo utilizan sean compatibles entre sí. El elemento básico y fundamental de la Técnica de Corrientes Portadoras, es el aprovechamiento doble de la instalación eléctrica ya existente como conductor de energía y de información. Con los componentes X.10, la red, además, del suministro de corriente, se encarga también de la transmisión de señales de mando para los diversos aparatos eléctricos. Con ello, se puede enviar señales de Corrientes Portadoras a cualquier punto de la instalación que se desee y, a su vez, se puede solicitar de dicho punto, la información pertinente. El sistema permite el accionamiento a distancia y control remoto de diversos receptores eléctricos, desde uno o desde varios puntos, y trabaja tanto en redes de corriente alterna monofásica como trifásica.

2.- **Principio de Funcionamiento.**- El funcionamiento de las Corrientes Portadoras se basa en la superposición sobre la señal de 220V/50Hz en Europa (ó 120V/60Hz en el Continente Americano) de una serie de pulsos de 120 kHz que serán los que transporten la información (ver la Figura IV.1), en formato digital a todos los componentes del sistema.

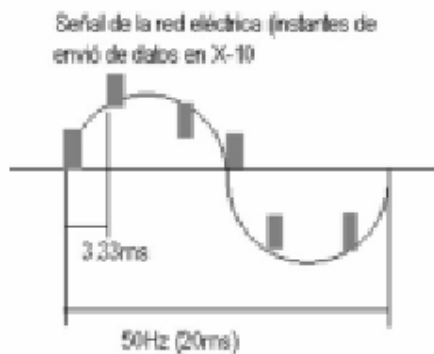


Figura IV.1.

Cada bit de información en X.10 se representa por un pulso de 120 kHz durante 1 ms en el punto de paso por cero de la corriente alterna. La exactitud de la duración del pulso no suele ser un factor crítico en la mayoría de las aplicaciones. Cada pulso deberá ser enviado tres veces para que coincida con el paso por cero de las tres fases de un posible sistema trifásico; es decir, deberá ser enviado en el paso por cero a los 3.333 y 6.666 mseg.

Es importante garantizar el sincronismo con el paso por cero de la señal alterna, ya que en ese preciso instante la línea presenta menos distorsiones para el receptor, esto es, se da la mayor relación Señal/Ruido para la transmisión de datos, siendo para el receptor más fácil el filtrado de la señal portadora. El valor de tensión de la línea coincidirá con el de la señal de 120 kHz, que podrá ser leída con mayor exactitud y menor probabilidad de error.

Debido a que la red eléctrica no es el medio más idóneo para la transmisión de señales digitales, en este Protocolo se recurre a añadir un código redundante para disminuir los posibles errores de comunicación:

- Cada bit, a excepción de los del código de inicio, se envía dos veces, una en su verdadero valor y seguidamente negado. Esto es, para enviar un “1” se enviará un “1” seguido de un “0”, y al contrario para enviar un “0”.
- Cada código X.10 será enviado dos veces.
- La transmisión completa de un Código X.10 necesita 11 ciclos de corriente: Los dos primeros representan el Código de inicio 1110. Los cuatro siguientes contienen el Código de Casa, letras A-P. Los siguientes cinco representan bien el Código Numérico (1-16); o bien, el Código de Función (*Encoger*, *Apagar*, *Aumento de Intensidad*, etcétera).

Tal y como se ha advertido, este bloque completo se transmite siempre dos veces, separando cada dos códigos por tres ciclos de la corriente, excepto para funciones de la regulación de intensidad, que se transmiten de forma continua (por lo menos dos veces), sin separación entre códigos.

3.- Diseño de una Aplicación Real.- Para evaluar realmente las posibilidades de esta Tecnología, se ha recurrido a la realización de un prototipo de un Sistema Domótico típico en el cual se controla el estado de dos variables tradicionalmente ligadas con el Confort, como son: Iluminación y Temperatura.

Con este fin se han repartido las tareas de control en una serie de tarjetas distribuidas, interconectadas entre sí a través de la Red Eléctrica. Dichas tarjetas se encuentran basadas en microelectrónica programada, siendo cada una de ellas gobernada por un Microcontrolador de gama media (el PIC16F84), que proporcionará la potencia de cálculo necesaria para traducir las tramas X.10 que reciben en las acciones de control pertinentes. De esta forma se dispondrá de:

- ◆ Un módulo de sensores: Esta tarjeta está formada básicamente por dos captadores analógicos (luz y temperatura), cuya señal, previamente convertida a valores digitales, es introducida en un Microcontrolador convenientemente programado para mostrarla a través de unos Displays o para transmitirla al módulo de control central, si éste la solicitara.
- ◆ Dos módulos de Actuadores.
- ◆ Un *Trimmer* destinado a permitir la regulación de la iluminación de bombilla incandescente, a través del control del valor de la tensión que cae entre sus terminales.
- ◆ Un Control Digital que permite el encendido o apagado del Actuator que estuviera conectado a dicha toma (ventilador, calefactor, etcétera).

La monitorización, tanto del valor actual de las variables como de su valor deseado, es realizada a través de un Ordenador, el cual, al disponer de una aplicación C++ específicamente diseñada, permite incluir funciones añadidas como son el control por medio de temporizadores.

Para evitar la necesidad de mantener permanentemente encendido el Ordenador, se ha procedido al diseño de una placa de control central externa a éste. Dicho módulo, gobernado por un Microcontrolador PIC16F84, dispone de los Algoritmos necesarios, por un lado para realizar comunicaciones bidireccionales de tramas con el Ordenador, y por otro, para enviar o recibir datos y señales de control de los módulos de sensores y actuadores conectados a la toma eléctrica a través de su correspondiente placa de comunicaciones. La Figura IV.2, muestra la apariencia final del prototipo. En esta representación todos los módulos se encuentran montados sobre una misma base, en la que la parte de la izquierda está destinada a albergar los distintos sensores y actuadores, mientras que la parte inferior derecha muestra la unidad de control del sistema.

Esta unidad será la que, a través de una conexión paralela, permita la interconexión entre el Ordenador (interfase con el Usuario) y los diferentes módulos mostrados en la fotografía de la cita Figura IV.2.

Las comunicaciones internas de las dos regiones en las que se divide el conjunto se realizan a través de dos placas de comunicaciones idénticas que colocadas una en cada renglón y conectadas a tomas de corriente diferentes, permiten la transferencia tanto de datos como de órdenes de control.



Figura IV.2.- Vista de los diferentes Módulos de que consta el Prototipo.

Las *placas de comunicación* (ver el esquema eléctrico de la Figura IV.3) son los elementos que permitirán transformar las salidas binarias TTL (0-5 volts), de un Microcontrolador en una señal senoidal de 120 kHz de similar amplitud, capaz de propagarse a través de la Red Eléctrica.

Esta señal modulada deberá ser remodulada por otra placa de comunicaciones de similares características, localizada junto al elemento destinatario de la comunicación. Las placas de comunicación informan al Microcontrolador (a cual están conectadas) de cualquier señal o trama de comunicación depositada en la Red Eléctrica, siendo éste el que deberá compara si el *código de elemento* incluido en el mensaje coincide con el suyo propio, atendiendo o desatendiendo la orden o los datos, incluidos en dicha trama.

Este hecho permite que varios módulos, cada uno con su respectivo Microcontrolador, pueden hacer uso de una única placa, si se coordina de forma adecuada el acceso al medio de cada uno de ellos a la hora de transmitir sus mensajes.

Estas placas tienen dos partes claramente diferenciadas: filtrado y acoplamiento a la Red Eléctrica, y sistema generador de señal portadora 120 kHz.

4.- Filtrado y Acoplamiento a la Red Eléctrica.- El filtrado de la señal eléctrica es realizado por medio de una serie de etapas LC sintonizadas, en las cuales se extiende una banda de paso en torno a la frecuencia de la señal portadora, eliminándose tanto la propia señal de 50 Hz como el efecto de sus armónicos.

El aislamiento de la parte de potencia (220V ó 120 V) de la parte de señal se consigue a través de un pequeño transformador.

5.- **Sistema Generador de Señal Portadora de 120 kHz.-** La modulación de los “1” y “0” lógicos, así como su posterior demodulación, es realizada a través de un CI de 16 terminales TDA5051A cuyo diagrama de bloques se muestra en la Figura IV.4.

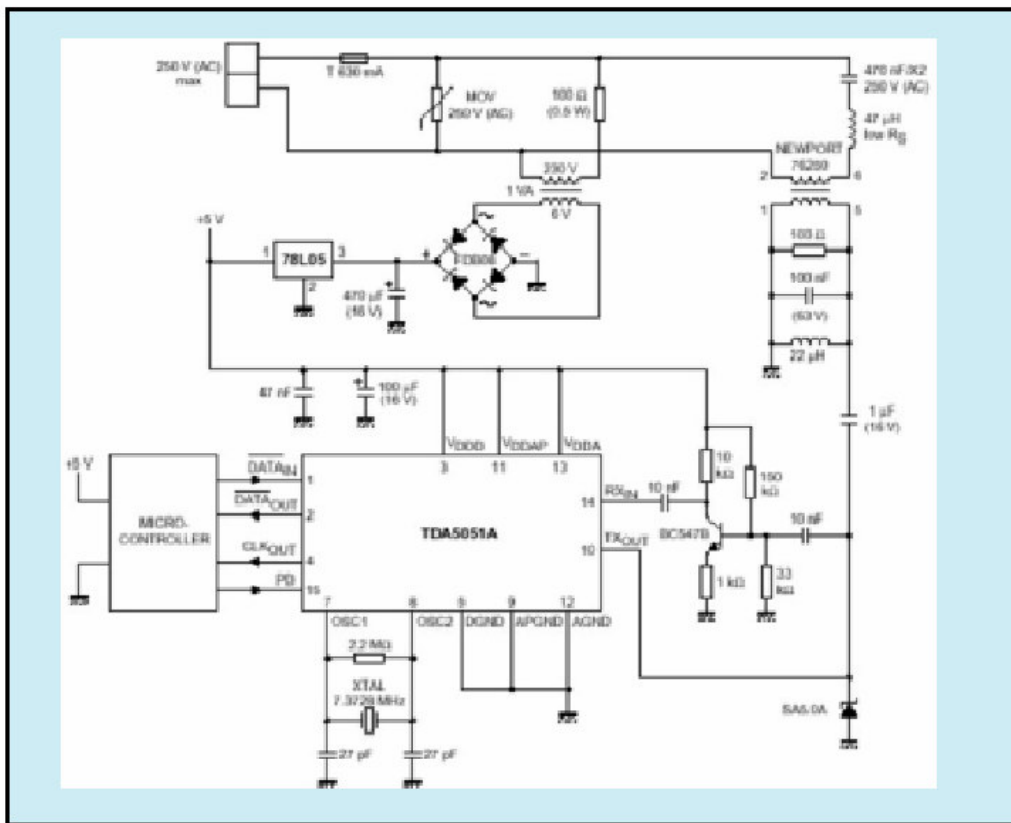


Figura IV.3.- Esquema Eléctrico de la Placa de Comunicaciones.

Este CI (de la marca Philips) realiza las funciones de un Módem con transmisión y recepción FSK a través de la red eléctrica convencional. Es por tanto, capaz de realizar comunicaciones bidireccionales “Half-Dúplex” entre distintos dispositivos a una velocidad de transmisión que puede establecerse mediante interruptores (*switches*) en 600 ó 1200 baudios (bits/segundos) a partir de una alimentación única de 5 volts.

El Módem se comunica directamente con el Microcontrolador por medio de dos terminales (DATA_IN y DATA_OUT), de forma que un nivel bajo en DTA_IN implicará la introducción en la red de la señal portadora; mientras que un nivel bajo en DATA_OUT informará al Microcontrolador de la presencia de una señal portadora.

6.- **Conclusiones Finales del Proyecto.-** A la vista de los resultados obtenidos en la experimentación que se lleva a la práctica, el sistema de comunicación a través de la red eléctrica utilizando Corrientes Portadoras, es un medio muy adecuado para conseguir la comunicación de pequeñas cantidades de datos, a bajas velocidades y en redes eléctricas con bajo contenido armónico, permitiendo la interconexión de diversos módulos por medio de una sencilla instalación, y sin necesidad de obra alguna.

Sus limitaciones, que pudieran hacer al sistema no deseable para uso industrial, no impiden que sea totalmente válido para aplicaciones domóticas, en las que las necesidades de control en tiempo real y el número de datos a enviar, son mucho menos restrictivas. Aunque los siguientes valores dependen mucho del lugar donde se realicen las pruebas, los resultados obtenidos en ambiente no hostil fueron éstos:

- ⊕ Tiempo transcurrido desde el envío de una orden (ON/OFF), hasta su ejecución: 0.2 segundos.
- ⊕ Distancia Máxima: Variable entre 50 y 75 metros, dependiendo de las perturbaciones.

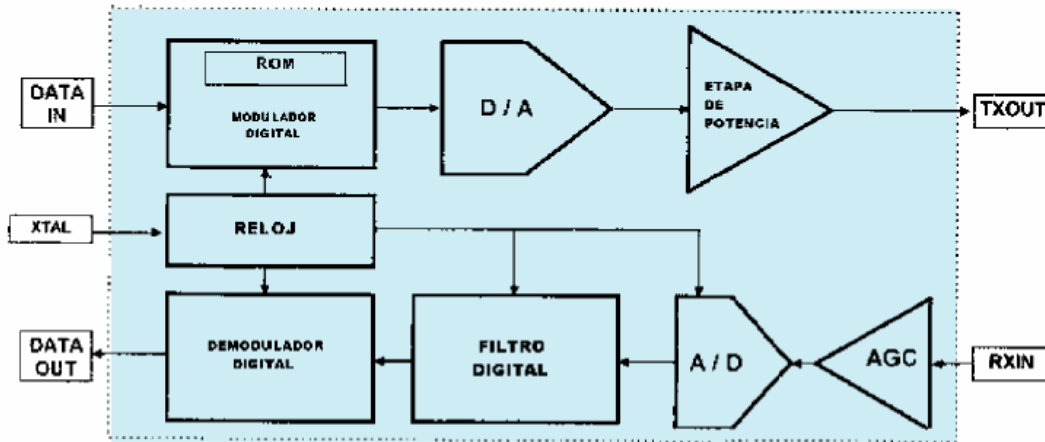


Figura IV.4.- Diagrama de Bloques del TDA5051A.

IV.2.2.- Mando a Distancia con Mensajes de Voz para Electrodomésticos.

El Segundo ejemplo de aplicación es el llamado: **“Mando a Distancia con Mensajes de Voz para Electrodomésticos”**.

1.- Descripción Funcional de los Electrodomésticos con Comunicación Bluetooth™.

En primer lugar hay que decir que este desarrollo de *Fagor™*, en colaboración con la ONCE, pretende mejorar la accesibilidad a sus electrodomésticos de colectivos que tienen algún tipo de discapacidad visual, para los que ciertas evoluciones del escenario actual en este tipo de productos están suponiendo barreras adicionales, como luego se explicará. Estos electrodomésticos de *Fagor™*, disponen de la capacidad de transmitir por radiofrecuencia toda la información que permite conocer, instantáneamente, cual es el estado de los mismos, tanto cuando un usuario los está programando, como cuando va avanzando o termina la función que el usuario ha puesto en marcha.

La transmisión por radiofrecuencia se realiza utilizando un estándar comercial, el *Bluetooth™*, de manera que el usuario pueda utilizar terminales estándar, como unos auriculares, o un teléfono móvil, para poder recibir, mediante mensajes hablados, la información enviada por el electrodoméstico.

De esta manera, cuando un usuario empieza a manejar un electrodoméstico, y actúa sobre cualquiera de sus mandos, la acción realizada, además de quedar reflejada en los indicadores o textos del portamandos del aparato, se oirá con el correspondiente mensaje hablado en los auriculares o la terminal portátil que el usuario llevará con ese fin. Así, paso a paso podrá programar el electrodoméstico en la función deseada, hasta ponerlo en marcha. Una vez en marcha, podrá solicitar el estado del mismo para conocer el tiempo que resta hasta la finalización de la función. Una vez terminada esta, el electrodoméstico avisará con un mensaje. También avisará al usuario con mensajes, si el aparato detecta en alguna ocasión algún tipo de anomalía.

Como continuación de esta línea de productos, se están desarrollando funciones más avanzadas, de manera que desde una terminal portátil (un teléfono móvil; una terminal específico; o, una terminal tipo PC), se podrá realizar la programación y la puesta en marcha de cualquiera de los electrodomésticos.

Con esta nueva prestación, se logra una ventaja de gran importancia, como es la de que cualquiera de los electrodomésticos se podrá manejar y programar utilizando una única interfase, la de la terminal portátil que está acostumbrado a utilizar el usuario, sin necesidad de tener que conocer los interfaces de usuario de cada uno de los electrodomésticos. Estas nuevas prestaciones, que marcan un paso importante en la apuesta de *Fagor™* por lograr aparatos accesibles para todos los usuarios, se incorporan a los productos estándar del catálogo de productos de *Fagor™* Electrodomésticos.

Para ilustrar lo que acabamos de exponer, en la siguiente tabla mostramos una sencilla secuencia de acciones y mensajes, durante la programación y puesta en marcha de una lavadora:

ACCIÓN DEL USUARIO	MENSAJE DEL TERMINAL
Pulsar tecla de encendido	<i>"ring" + Lavadora encendida</i>
Girar el programador	<i>Programa normal; 30 grados; 1100 revoluciones; duración estimada 120 minutos</i>
Pulsar tecla "Fácil planchado"	<i>Fácil planchado activado</i>
Pulsar tecla "Aclarado extra"	<i>Aclarado extra activado"</i>
Volver a pulsar tecla "Aclarado extra"	<i>Aclarado extra desactivado</i>
Pulsar tecla "Marcha"	<i>En marcha; tiempo restante estimado: 87 minutos</i>

2. Escenario Actual, Evolución de Equipamientos Domésticos y Relación con otros Escenarios.

En el mundo de los electrodomésticos, se está dando en los últimos años una generalización acelerada de controles electrónicos que gobiernan el funcionamiento de estos aparatos.

Las circunstancias que han hecho que los controles electrónicos, presentes en este sector desde hace ya bastantes años, se hayan generalizado de forma rápida en estos últimos años, sustituyendo a los controles electromecánicos, están probablemente relacionadas con el coste competitivo al que han llegado este tipo de componentes; las posibilidades de mayores y más ajustadas prestaciones que permiten; la necesidad de competir y diferenciarse que ha hecho que la introducción masiva de este tipo de controles por unas marcas haya sido seguido por otras; y probablemente, otros muchos factores difíciles de determinar.

Esto ha traído consigo que las interfaces de usuario se hayan modificado también sustancialmente, de manera que los mandos giratorios con selecciones de posición por saltos, han sido sustituidos por pulsadores, o por botones giratorios, cuya acción es mostrada en indicadores luminosos de dígitos, o con algún carácter. Las teclas de arranque, paro y similares también han sido sustituidos por pulsadores con indicadores luminosos (pilotos o dígitos). Y así sucesivamente. Hasta llegar a la creciente aparición de pantallas tipo LCD, con textos que sirven de menú, donde seleccionar los distintos programas y parámetros de los mismos, mediante la utilización de pulsadores o mandos giratorios.

La generalización de este tipo de interfases de usuario se está dando también en otros ámbitos, bien por evolución, como ha sucedido en el campo de los electrodomésticos o bien porque son nuevos servicios que han nacido con este tipo de interfases.

En los escenarios en los que debemos desenvolvemos habitualmente cualquiera de nosotros, encontraremos este tipo de interfaces, como pueden ser los ascensores; los terminales informáticos de bancos; transportes públicos; terminales de información de edificios públicos; vía pública (semáforos, terminales de información); y por supuesto en teléfonos (móviles, y también fijos) y en terminales tipo PC.

Se puede decir que los avances tecnológicos (a costos asumibles por el gran público) han sido los que han motivado la aparición de toda esta diversidad de terminales electrónicos con interfaces visuales basados en textos o en iconos, y los que, en cierta manera, han impuesto nuevas barreras para aquellos que tienen algún tipo de discapacidad visual, en mayor o menor grado. Pero también podemos decir, que esos mismos avances tecnológicos, adecuadamente aplicados, pueden contribuir, no solo a que estos avances no supongan una barrera, sino incluso a facilitar el acceso a todos estos servicios y aparatos de una manera más cómoda y ventajosa de lo que se podía hacer anteriormente.

En este sentido, la utilización de la transmisión inalámbrica de datos, en combinación con terminales estándares, que cada vez más, están extendidos a una gran mayoría de personas, puede ser una de las vías para superar estas barreras. Para que estas soluciones sean realmente útiles y viables, otra condición de suma importancia es que las aplicaciones que se realicen en los diferentes sectores de productos y de servicios, utilicen tecnologías y protocolos estandarizados, de manera que con cualquier terminal de una amplia gama de terminales estándar, se pueda acceder a cualquiera de estas aplicaciones.

La estandarización de las tecnologías, protocolos y vías de acceso a estas prestaciones y servicios, también será un factor importante que favorecerá tanto a los usuarios, al poder acceder a los mismos en cualquier escenario con una terminal estándar, como a los fabricantes de este tipo de aparatos y servicios, que podrán ofertar desarrollos dirigidos a un más amplio sector, y en suma favorecerá su extensión y generalización.

En el escenario en el que *Fagor™* Electrodomésticos incide, el del hogar en general, y con especial atención en el entorno de la cocina, la integración de diferentes aparatos, sistemas y servicios es un paso importante para facilitar la accesibilidad a todos ellos. En este sentido, *Fagor™* ya ha dado los primeros pasos en esta dirección al integrar todos los electrodomésticos en una única red, la Red Domótica *Fagor™*, gobernados por una inteligencia central o *Maior-Domo*.

En estos momentos se están desarrollando las pasarelas ("gateway") y aplicaciones necesarias para que diferentes sistemas instalados en el hogar, puedan relacionarse y cooperar entre si. Tales como sistemas de seguridad, sistemas de sensores y actuadores, sistemas de videovigilancia, y sistemas multimedia.

También se están desarrollando las aplicaciones necesarias para que la inteligencia central o *Maior-Domo*, tenga una conexión con el mundo exterior a través de una conexión Internet, que lo comunicará con un servidor específico de *Fagor™* que suministrará servicios dirigidos al hogar y a los electrodomésticos, y que le permitirá conectarse con cualquier otro servidor que desarrolle servicios de este tipo.

La importancia de la integración de diferentes sistemas y de la comunicación con el mundo exterior a través de Internet, para facilitar la accesibilidad a cualquier tipo de aparatos, prestaciones y servicios, viene de que una vez realizada esta integración, es suficiente con desarrollar una terminal que facilite este acceso, para resolver la accesibilidad a todo el conjunto de aparatos y servicios. Por otra parte, la conexión con el mundo de Internet, nos permitirá tener a nuestra disposición las aplicaciones necesarias para que se pueda instalar en nuestra terminal, en nuestro *Maior-Domo* o en nuestros aparatos domésticos, aquellas funciones necesarias para implementar esta accesibilidad, así como su actualización o la incorporación de nuevas funciones.

3. Descripción Técnica de los Electrodomésticos *Fagor™* con *Bluetooth™*.

En todos los electrodomésticos *Fagor™* que tienen su funcionamiento gobernado por un control electrónico, se ha dispuesto sobre la tarjeta electrónica en que se materializa dicho control, un conector en el que está accesible toda la información del funcionamiento del aparato en tiempo real, y desde el que se pueden también introducir datos que actúen sobre el funcionamiento de dicho aparato electrodoméstico.

Por otra parte, se ha desarrollado un pequeño circuito electrónico, denominado Kit *Bluetooth™* que conectado a la tarjeta de control del electrodoméstico mediante el conector antes citado, transforma dicha información desde el protocolo BMF en que lo recibe, al Protocolo *Bluetooth™*, transmitiéndolo entonces al aire por radiofrecuencia. Este es el comportamiento de este Kit cuando los datos que recibe por BMF los envía en forma de datos por *Bluetooth™*.

Pero el Kit *Bluetooth™* también puede enviar los datos recibidos por BMF en forma de mensajes de audio. Para ello, el Kit transforma los datos en mensajes y los envía al aire por el canal audio *Bluetooth™*. Con esto se permite una mayor diversidad de posibles terminales receptores.

En el caso de la utilización de una terminal como mando a distancia, el Kit *Bluetooth™* realiza las operaciones inversas, partiendo de los datos enviados por el terminal, es decir: la recepción de comandos procedentes de la terminal, en Protocolo *Bluetooth™*; su conversión a al protocolo BMF; y el traspaso, en este formato, de la información, a la electrónica del electrodoméstico.

A continuación se va a realizar una descripción mas detallada de cada una de las partes fundamentales que conforman esta nueva funcionalidad de los electrodomésticos *Fagor™*.

3.1 Tarjeta de Control del Electrodoméstico y Bus BMF.

Los controles electrónicos de los electrodomésticos están basados en un Microcontrolador que gobierna las entradas y las salidas necesarias para que el aparato realice las funciones correspondientes. En algunos casos es un único microcontrolador el que gobierna todas las entradas/salidas, y en otros casos hay un segundo microcontrolador dedicado específicamente al control de una interfase de usuario más o menos sofisticado, dependiendo de la filosofía adoptada para el producto y también de la gama a la que pertenece el aparato.

En el caso de *Fagor™*, la idea es tender a la modularidad de los controles electrónicos, lo que posibilita la flexibilidad de estos controles, facilitando la realización de diferentes gamas y el desarrollo de diferentes prestaciones, sin modificar la tarjeta de control, entre otras ventajas.

Elemento esencial para lograr esta flexibilidad, permitiendo conectar interfases de distinto tipo y gama, y módulos electrónicos que implementan diferentes prestaciones (Kit Domótico; *Kit Bluetooth™*; etcétera), es el bus de comunicaciones desarrollado por *Fagor™* para este fin, el Bus Multipunto *Fagor™* (BMF). Tanto las características físicas, como el protocolo de este bus, se han diseñado optimizando sus exigencias y prestaciones para su utilización dentro de los electrodomésticos. Así podemos resumir estas características en las siguientes:

- Es un bus de tres hilos. Se ha buscado un bus con un número mínimo de hilos para facilitar el cableado y reducir costes, tanto del propio cable como de los conectores de sus extremos. Estos tres hilos son el de masa, alimentación e información.
- Es un bus síncrono que puede transferir información a 1200 baudios o a 9600 baudios según requerimientos de la aplicación. También se han habilitado dos maneras de acceder a la información del bus desde el microcontrolador: mediante *pulling*, para los microcontroladores sin UART y por interrupción para los que lo tienen.

- Es un bus multipunto para permitir que varios módulos puedan estar conectados a la tarjeta de control del electrodoméstico simultáneamente.
- El software correspondiente al protocolo de comunicaciones ocupa poco más de 1Kbyte de memoria. Se ha procurado que el protocolo de comunicaciones ocupe el mínimo espacio en memoria, pero satisfaciendo con fiabilidad todas las prestaciones, para que no penalice el coste de los microcontroladores a utilizar.
- Todos los electrodomésticos *Fagor™* con control electrónico incorporan el Bus Multipunto *Fagor™*, lo que quiere decir que el microcontrolador lleva el bloque de software correspondiente a dicho protocolo, y que el circuito electrónico de control lleva un conector de tres pines, en el que se accede e inserta información relacionada con el funcionamiento del electrodoméstico, mediante el BMF.

Como se ha indicado, este bus permitirá que un determinado electrodoméstico lleve un portamandos u otro; que sea un modelo con prestaciones domóticas; o que sea un modelo con facilidad de acceso para discapacitados visuales. En este último caso, el electrodoméstico llevará conectado al BMF un módulo electrónico al que se ha denominado *Kit Bluetooth™*.

3.2 Módulo Electrónico Kit Bluetooth™.

Este módulo electrónico es el encargado de recibir la información proveniente del control electrónico del electrodoméstico a través del BMF, extraer los datos y componerlos según el Protocolo *Bluetooth™*, y a continuación transmitirlos por radiofrecuencia. Cuando está funcionando con una terminal que realiza las funciones de mando a distancia, realizará el proceso inverso, es decir, captar la información de radiofrecuencia que llega con Protocolo *Bluetooth™*; extraer la información; y, volverla a componer según el protocolo BMF para pasarla al control del electrodoméstico.

Como ya se ha expresado, para lograr una mayor diversidad de posibilidades en cuanto a terminales que puedan servir al usuario como acceso a esta prestación, se ha desarrollado una aplicación paralela sobre este Kit. Va dirigido a que se puedan utilizar auriculares inalámbricos *Bluetooth™* para captar, mediante mensajes de voz, las manipulaciones realizadas sobre el electrodoméstico, así como su estado. En este caso, la información recibida por el Kit a través del BMF, es extraída y a continuación emparejada con el mensaje hablado que le corresponde, el cual se encuentra almacenado en una memoria de mensajes; estos mensajes hablados son transmitidos por radiofrecuencia utilizando el canal de voz del protocolo *Bluetooth™*. A continuación vamos a indicar las principales características de este módulo electrónico denominado *Kit Bluetooth™*:

- Los principales componentes físicos de este Kit son una fuente de alimentación conmutada, el chip *Bluetooth™*, un microcontrolador de 8 bits y una memoria para los mensajes de voz.
- La incorporación en el Kit de su propia fuente de alimentación se debe a que los electrodomésticos incorporan fuentes ajustadas a sus necesidades básicas, de manera que los módulos suplementarios deben de llevar su propia fuente de alimentación.
- El *Chip Bluetooth™* utilizado es Clase A, lo que garantiza un alcance que puede cubrir sin problemas cualquier vivienda normal (100 metros en espacio abierto). El stack (Blue Core 2.7 CSR), está implementado hasta el nivel RFCOM y SDP.
- El Microcontrolador de 8 bits es el encargado de realizar la extracción de datos y su transformación de un bus a otro, así como de gestionar los mensajes de voz. Lleva 2 UART, un puerto síncrono, una memoria flash de 60Kb y una memoria RAM de 2 Kb.
- La memoria de voz es una memoria RAM de 1 Mb. Lleva grabados el conjunto de palabras que permitirán componer los múltiples mensajes necesarios para comunicar al usuario los procesos de programación, activación y avisos de todos los electrodomésticos. El número de palabras grabadas es del orden de 500 y se han realizado mediante grabación de voz natural, por resultar de mejor calidad que la voz sintetizada. Están muestreadas a 8Khz y cada muestra es de 8 Bits.
- Los bloques de software embarcados en el Microcontrolador son fundamentalmente: el protocolo BMF; extracción de datos del BMF y del *Bluetooth™*; composición de datos en la trama *Bluetooth™* y en la trama BMF; software de interfase con el *Chip Bluetooth™*; y la gestión de los mensajes de voz.

- En la memoria propia del *Chip Bluetooth™* se incorporan los bloques de software correspondientes a los perfiles de *Bluetooth™* de Datos y de Audio; y, el software de interfase con el microcontrolador NEC.
- Otros aspectos de gran importancia han sido el diseño de la antena, para que integrada en el circuito que se ha desarrollado, tuviera una buena transferencia de energía; el diseño del tamaño y la forma del Kit para lograr una buena integración física en los electrodomésticos; y la elección de la ubicación en los mismos, para que la interferencia del propio electrodoméstico en la transmisión de la radiofrecuencia fuera la mínima posible.

3.3 Terminales Receptores y Mando a Distancia.

Para el desarrollo de las aplicaciones necesarias, para acceder a los electrodomésticos desde los diferentes terminales, ha sido imprescindible la colaboración de la ONCE, como representante de colectivos que son potenciales usuarios de las prestaciones desarrolladas.

Fagor™ Electrodomésticos ha aportado al desarrollo de estas aplicaciones, su conocimiento de los electrodomésticos y de las diferentes secuencias necesarias para programarlos, activarlos y monitorizarlos, y por lo tanto ha intervenido en la definición de las interfaces de usuario de los terminales.

Por su parte, la ONCE se ha encargado de desarrollar las diferentes aplicaciones que incorporadas a los correspondientes terminales, serán capaces de reflejar en mensajes de voz las manipulaciones realizadas sobre los electrodomésticos, en función de los mensajes recibidos por *Bluetooth™*; así como de desarrollar la aplicación sobre la terminal que lo convierte en un Mando a Distancia de los electrodomésticos.

Como terminales que estarán inmediatamente disponibles para poder interactuar con los electrodomésticos de *Fagor™*, gracias a los desarrollos realizados por la ONCE se tienen dos modelos estándares determinados de teléfonos móviles de *Nokia™* y de *Ericsson™* y un teléfono móvil de *Owasys™* específicamente desarrollado para discapacitados visuales.

También son operativos para la funcionalidad desarrollada por *Fagor™*, la mayoría de los auriculares estándares *Bluetooth™* para teléfonos móviles, que se encuentran en el mercado.

En un corto periodo de tiempo, también estarán disponibles otros terminales, como los PDA, PC estándar o el *Maior-Domo Fagor™*, para los que se están adaptando las aplicaciones desarrolladas.

Para una descripción más detallada del contenido y de cómo se ha realizado el desarrollo de cada una de las aplicaciones para los diferentes terminales, será la ONCE la más indicada para hacerlo, por ser quien se ha encargado de materializarlos. Para utilizar esta funcionalidad basada en electrodomésticos con *Bluetooth™* y terminales con mensajes hablados, es necesario que antes de empezar a utilizarlos, se haga un proceso de emparejamiento entre la terminal y el electrodoméstico. Este sencillo proceso de emparejamiento se realiza utilizando la propia terminal y solo hay que realizarlo una vez.

IV.2.3.- Plataforma para el Desarrollo de Sensores Inteligentes y Sistemas Microprocesados

La Tercera aplicación se refiere a una: **“Plataforma para el Desarrollo de Sensores Inteligentes y Sistemas Microprocesados”**.

1. Descripción del Sistema.

El sistema implementado es un prototipo domótico que se articula en torno a un bus de campo (Kaschel y Pinto, 2002) que es el elemento que le confiere las principales ventajas al sistema. Como paso previo, se definieron cuatro criterios que debían ser respetados en el diseño del sistema:

- ✿ Ampliable. Para permitir la inclusión de nuevos sensores y actuadores de diferente naturaleza
- ✿ sin tener que modificar el protocolo o software de los elementos existentes.
- ✿ Abierto. Con especificaciones claras para que se puedan desarrollar nuevas aplicaciones sobre su propio hardware e incluirlo en el sistema y ofreciendo la posibilidad de que de este proyecto surjan otros que lo complementen.
- ✿ Controlable de forma remota. Para permitir la monitorización del estado del hogar inteligente y
- ✿ la intervención en el sistema desde cualquier ordenador conectado a Internet y desde terminales móviles (GSM/GPRS).
- ✿ Con maqueta. Para lograr un entorno más cercano a la realidad donde poder realizar pruebas
- ✿ sobre el sistema.

El segundo paso importante fue la elección del bus de campo más apropiado. En la actualidad existen múltiples estándares de buses de campo específicos para domótica como por ejemplo X10, EHS, EIB, Lonworks, Batibus, etcétera. Cada uno de estos buses tiene características diferentes y están enfocados a distintos sectores del mercado domótico. X10 por ejemplo es un estándar que aprovecha la infraestructura eléctrica para transmitir señales de control, es barato y sencillo de instalar, pero la velocidad de transmisión es muy baja (50 bit/s) y su funcionalidad se reduce a poco más que encender y apagar dispositivos.

Otros buses como EHS también pueden aprovechar la infraestructura eléctrica o utilizar otros medios de transmisión como el par trenzado, y son capaces de alcanzar velocidades algo mayores (del orden de los 9600 bit/s). En el otro extremo se encuentra LONWORKS (Hertel, 2001), que puede considerarse como un bus domótico de altas prestaciones. Tiene una estructura jerárquica, puede transmitirse por múltiples medios de transmisión y alcanza velocidades de hasta 1.25 Mbit/s aunque su elevado coste ha hecho que la implantación de LonWorks sólo haya tenido éxito en hoteles y grandes edificios.

A pesar del gran número de buses de campo disponibles para aplicaciones domóticas, se optó por utilizar CAN (Bosch, 1991), bus de campo versátil ampliamente utilizado en sectores como el automovilístico y en la industria. Evidentemente esta decisión no se atiene a las tendencias de convergencia de buses domóticos, pero la realidad muestra que a pesar de esto, las empresas siguen apoyando el desarrollo de sus propios productos, resultando en un cúmulo de buses diferentes compitiendo entre sí en lo que se ha denominado como *“guerra de los buses”* (Kaschel y Pinto, 2002).

Las características de CAN lo hacen adecuado para su uso en aplicaciones domóticas, logrando un rendimiento comparable al de los buses de campo de altas prestaciones. CAN reúne muchas de las ventajas que tienen el resto de buses de campo bajo un estándar verdaderamente abierto, sencillo y económico que permite una mayor libertad a la hora de definir los protocolos de las capas superiores. Esto, junto con las características propias de CAN, fue uno de los motivos que nos llevaron a escogerlo. El sistema diseñado hereda de CAN dos características fundamentales:

- ✿ En el nivel de enlace, el protocolo de la tecnología CAN no utiliza los identificadores como direcciones de los nodos de la red sino que éstos indican el tipo de contenido del mensaje. Por ejemplo: podríamos asignar un identificador para los datos de temperatura, otro para los de velocidad, etcétera. De esta manera se pueden establecer comunicaciones punto a multipunto ya que sólo los nodos a los que interese el mensaje lo recibirán.
- ✿ El tipo de acceso al medio es basado en contienda pero con la particularidad de que no se desperdicia tiempo, debido a que las colisiones se resuelven mediante un arbitraje de bits en el que gana el nodo con más prioridad.

Si a nivel funcional las ventajas son notables, a nivel educativo lo son aún más puesto que se permite al alumno observar un desarrollo completo utilizando buses de campo. En la siguiente figura se muestra un esquema general del sistema en el que se muestra la interconexión de los diferentes elementos que lo componen:

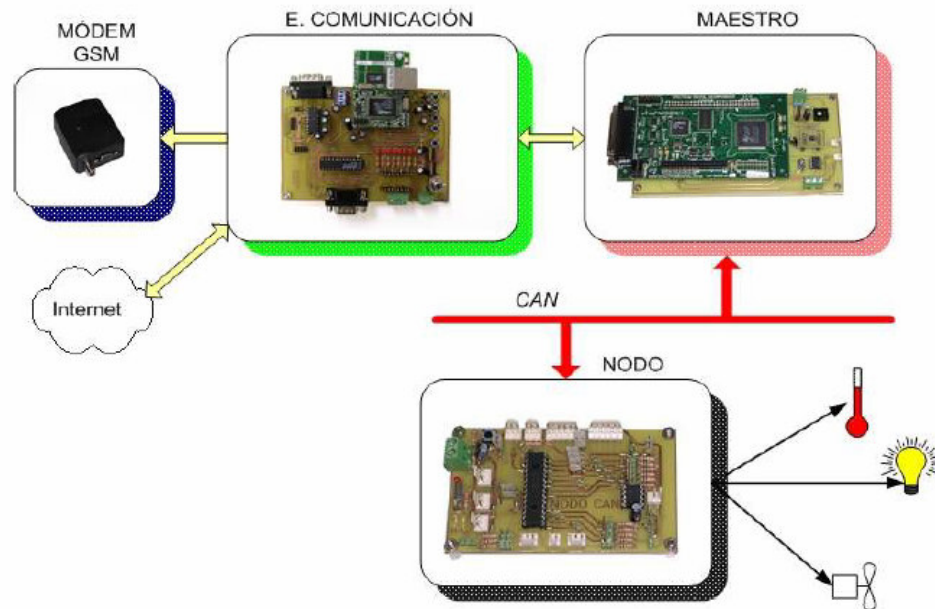


Figura IV.5.- Esquema General del Sistema en donde se muestra la Interconexión de los diferentes Elementos que lo Configuran.

El sistema está gobernado por el “elemento maestro”, basado en un DSP TMS320F2812 de *Texas Instruments*®. Este elemento está constituido por un DSP de Texas Instruments, en particular, el dispositivo TMS320F2812. El elemento maestro se encarga, entre otras tareas, de:

- ✚ Almacenar y administrar la información general del sistema como la distribución de sensores y
- ✚ actuadores en los sectores o habitaciones, dirección IP, máscara de subred, contraseñas para el servidor, números de teléfono a los que enviar las alarmas, etcétera.
- ✚ Gestionar los números de identificación de los nodos y los elementos sensores y actuadores.
- ✚ Trasladar las órdenes del usuario a los elementos de la red para que actúen en consecuencia y recoger los datos solicitados por el usuario.
- ✚ Generar alarmas si se dan las condiciones correspondientes.

Otro de los elementos destacados del sistema es el elemento de comunicación con el exterior. Se trata básicamente de un módulo RCM2200, que contiene un procesador Rabbit 2000. Este módulo está preparado para ser conectado en redes Ethernet 10/100BaseT, razón fundamental por la que se escogió este módulo para desempeñar las tareas de comunicación con el exterior. Este módulo controla, además, el módem GSM/GPRS GM29 de *Sony-Ericsson*™ a través de uno de sus puertos serie y está permanentemente comunicado por otro de sus puertos serie con el elemento maestro. El software que se ha creado para este módulo se encarga principalmente de hacer las veces de servidor unificado para el acceso remoto. De esta manera, el mismo servidor da acceso al sistema tanto si el usuario utiliza un ordenador conectado a Internet como si lo hace a través de una terminal móvil GPRS.

El módem GSM/GPRS se encarga básicamente de canalizar los mensajes de alarma a través de SMS al usuario y al servicio de emergencias correspondiente. Se ha optado por utilizar SMS para enviar las alarmas debido a que su recepción es casi instantánea.

Los elementos sensores y actuadores se agrupan en “nodos”. Los nodos han sido implementados mediante microcontroladores PIC18F248 sobre una placa genérica que se adapta a multitud de sensores y actuadores gracias a sus líneas de entrada analógicas y sus entradas/salidas digitales. Como muestra de la capacidad del sistema, al prototipo se le ha dotado de sensores y actuadores tales como luces, sensores de temperatura, sensores de luz, sensores de proximidad por ultrasonidos, interruptores, etcétera.

Para coordinar los elementos del sistema se han diseñado dos protocolos, uno para la red interna de bus CAN al que se ha denominado protocolo interno y otro para la comunicación entre el servidor y la aplicación cliente sobre Internet al que se ha denominado protocolo externo. El primero de ellos aprovecha las ventajas de CAN que se citaron anteriormente para permitir el intercambio de información entre los actuadores y sensores de la red. Gracias a las características de este protocolo, el sistema está dotado de una gran funcionalidad. Por su parte, el protocolo externo gestiona la comunicación entre el usuario y el sistema a través de Internet. El protocolo externo es un protocolo de la capa de aplicación y se transmite sobre TCP. Este protocolo permite controlar el sistema desde un PC conectado a Internet o desde una terminal móvil GPRS.

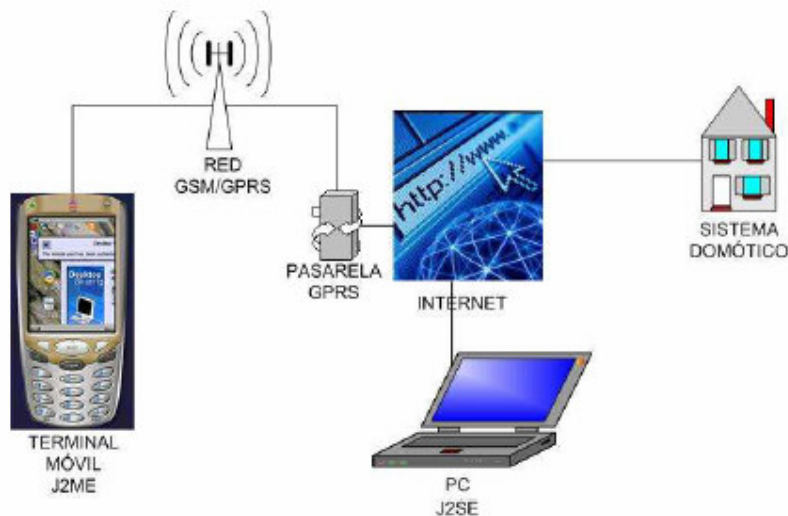


Figura IV.6.- Acceso al Sistema.

Para que la aplicación cliente pueda manipular los elementos instalados en el sistema, se ha creado la figura de los ficheros de manejo, que juegan un papel análogo a los manejadores (*drivers*) para un sistema operativo convencional. Haciendo uso de estos ficheros, la aplicación cliente permite al usuario influir directamente sobre el comportamiento de los elementos (por ejemplo, encender una luz), pero también permite hacer que otros elementos influyan indirectamente sobre el comportamiento de un elemento (por ejemplo, hacer que la luz se encienda cuando se detecte movimiento o cuando la luminosidad sea baja).

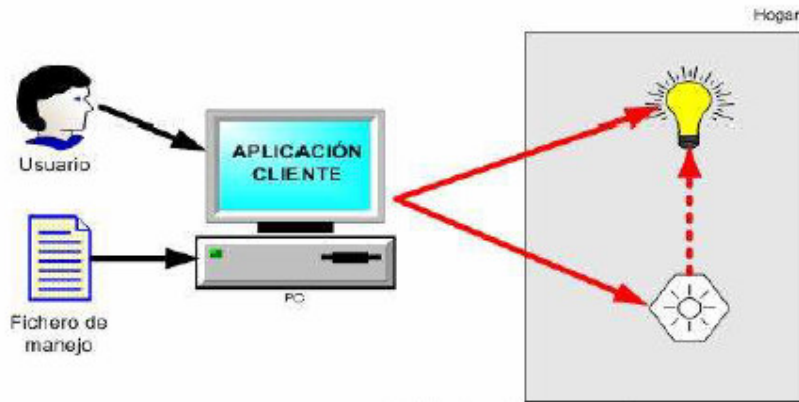


Figura IV.7.- Posibilidades de Actuación.

Una ventaja que se ha añadido al sistema es la posibilidad de insertar nuevos elementos en la red mediante un mecanismo de autorregistro (*plug & play*) con lo que se facilitan las ampliaciones. El sistema se completa con las aplicaciones cliente, que en principio están siendo implementadas en forma de programa ejecutable específico. En el caso de la aplicación para PC, el programa está realizado en JAVA (Schildt, 2001), mientras que la versión para terminales móviles está realizándose en JAVA 2 Micro Edition (J2ME). Existen otras arquitecturas cliente-servidor posibles y de hecho se está estudiando la posibilidad de sustituir la aplicación cliente específica por una página web que implemente la interfase para el usuario y le evite a éste tener que instalar programas o tener que disponer de la aplicación en un soporte físico. De este modo se mejoraría la accesibilidad.

3. Aplicación al Laboratorio de Instrumentación Electrónica.

Este sistema se puede emplear en un Laboratorio de Instrumentación Electrónica cualesquiera. El laboratorio de instrumentación electrónica es una asignatura práctica en la que los alumnos trabajan en grupos usando diferentes equipos de instrumentación como osciloscopios, analizadores de espectro, multímetros, analizadores lógicos, etcétera. Las prácticas están estructuradas en dos grupos. El primero, contiene prácticas guiadas, explicadas paso por paso. En estas prácticas los alumnos aprenden y revisan el manejo básico de los aparatos de instrumentación, usándolos para obtener medidas en circuitos sencillos. El segundo grupo contiene prácticas donde los alumnos tienen que usar todo el conocimiento que han aprendido durante el primer grupo de prácticas, junto con otros conocimientos adquiridos durante la carrera, para solucionar los problemas propuestos en la práctica.

Dado que para cursar este Laboratorio, los alumnos ya tienen conocimientos de Instrumentación, y aunque se supone que son capaces de manejar la mayoría de los aparatos, es aconsejable recordarles su uso. En esta estructura se han detectado algunos problemas, entre ellos destaca la falta de aplicación directa de los circuitos sobre los que se trabaja.

Utilizando este prototipo doméstico se pretende dar solución a una de las reivindicaciones de los alumnos: trabajar sobre circuitos con aplicación directa. El hecho de trabajar con sistemas cercanos a la realidad motiva el interés del alumno. El sistema será utilizado en el laboratorio de instrumentación con una doble finalidad: por un lado, se han previsto algunas sesiones prácticas con dos vertientes diferentes. La primera versa sobre circuitos de adaptación de sensores y actuadores. En particular se utilizan circuitos para adaptar termopares y sensores de proximidad. En estas prácticas se fomenta el uso de multímetros y osciloscopios y, además, se enseña la naturaleza y principios de funcionamiento de estos sensores.

La segunda vertiente se centra en el manejo del bus CAN y en la programación de los microcontroladores. En estas prácticas el alumno aprende algunas nociones sobre el manejo y programación de los microcontroladores y sobre el funcionamiento y aplicaciones de los buses de campo. Se pretende que sean capaces de manejar los sensores y actuadores utilizando los periféricos que incorporan estos dispositivos, tales como convertidores analógico-digitales, puertos serie, líneas de entrada/salida, así como los módulos CAN para establecer comunicación con el resto de elementos y desarrollar estrategias de control distribuido. Esta práctica fomenta el uso de osciloscopios, analizadores lógicos y entornos de desarrollo para microprocesadores.



Figura IV.8.- Muestra de la Maqueta.

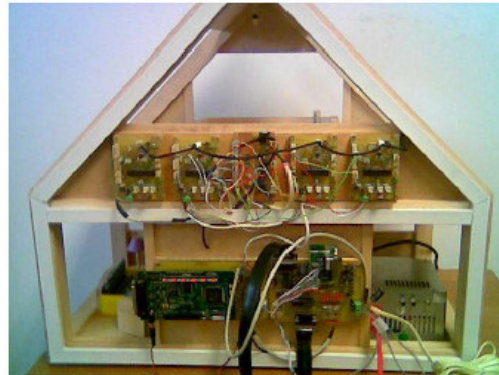


Figura IV.9.- Hardware del Sistema.

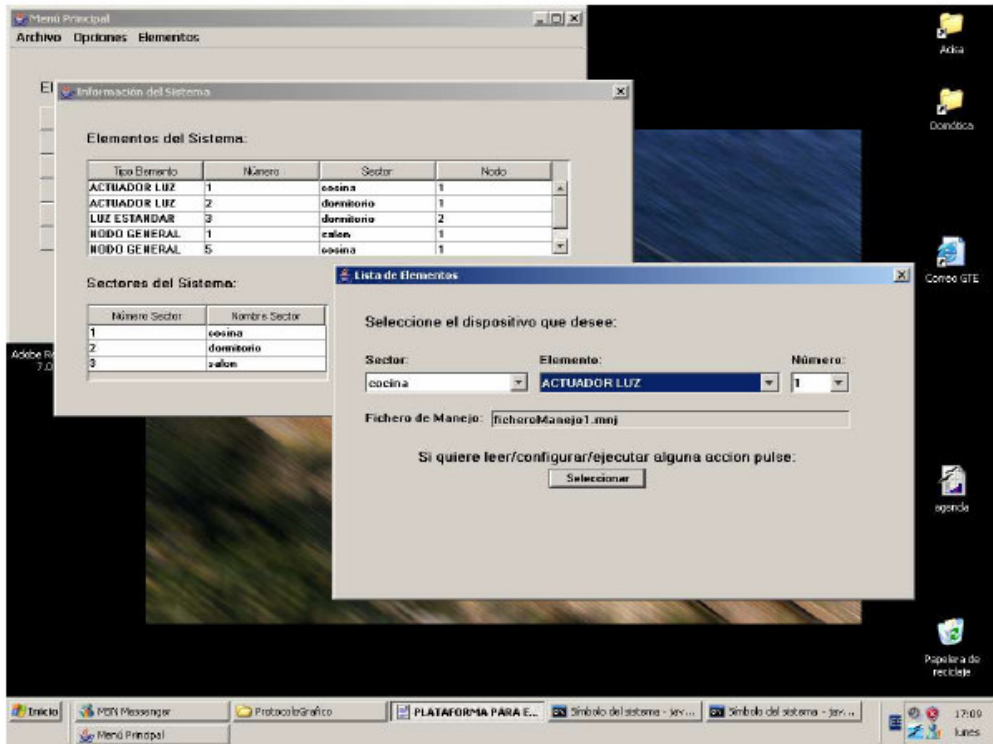


Figura IV.10.- Aplicación Cliente para Ordenador basada en JAVA.

Por otra parte, también se pretende aprovechar el sistema para la realización de trabajos tanto voluntarios como obligatorios, iniciativa que ya viene funcionando desde cursos pasados y que los alumnos encuentran bastante útil. Lo que se pretende con estos trabajos es enseñar al alumno a hacer diseños de esquemáticos, PCB, a buscar y seleccionar componentes electrónicos, a manejarse con documentación técnica y a depurar el hardware que han desarrollado, haciendo uso de la instrumentación electrónica disponible. Como resultado, los alumnos obtienen una visión general del proceso de diseño de sistemas electrónicos reales y adquieren una experiencia muy útil que no suelen recibir en otras asignaturas.

Alternativamente se pueden plantear nuevas prácticas sobre el sistema orientadas a las comunicaciones móviles y al desarrollo de aplicaciones en red a través de Ethernet. La previsión es que el sistema evolucione progresivamente gracias a las contribuciones de los propios estudiantes. Aunque la expansión natural del sistema pasa por la creación de nuevos tipos de sensores y actuadores, también se plantea la incorporación de tecnologías inalámbricas (Ferrari, Flammini, Marioloi, Sisinni y Taroni, 2005), como *Bluetooth™* o Zigbee, que sustituyan en algunos casos al bus de campo.

4. Conclusión.

El sistema propuesto constituye una herramienta docente novedosa y útil en asignaturas de últimos cursos, como en el Laboratorio de instrumentación presentado en este apartado, y adecuada para acercar aplicaciones de tipo práctico y reales al alumno. Trabajar con tecnologías reales permitirá atraer la atención de los alumnos, ayudándoles a comprender mejor la utilidad de los conocimientos adquiridos a lo largo de la carrera. Su amplia funcionalidad les permitirá desarrollar nuevas aplicaciones domóticas con posibilidades casi ilimitadas. Pero más allá de lo meramente técnico, los alumnos pueden adquirir conocimientos sobre cómo se estructuran sistemas domóticos, algo que les puede ser de utilidad cuando tengan que abordar proyectos en los que los ingenieros de telecomunicación tienen competencia, como son los proyectos de ICT y domótica.

IV.2.4.- Aplicaciones y Líneas de Investigación de los Microcontroladores. (**Estado del Arte de los Microcontroladores**).

En este apartado se muestran algunas de las aplicaciones que desarrollan Arquitectura de Sistemas (*Hardware*) y Programación de Sistemas (*Software*) de sistemas Microelectrónicos Programables (Microcontroladores, DSP¹, FPGA², etcétera); incluyendo tecnología inalámbrica en los equipos Microelectrónicos.

Las principales **Líneas de Investigación** se desarrollan en los siguientes incisos.

1.- *Desarrollo de Equipos de Macrobiótica con Sensores Avanzados.*- Investigación y desarrollo de equipos de macrobiótica integrados con sistemas de sensores, actuadores y comunicaciones que permiten realizar tareas complejas.

2.- *Desarrollo de Equipos Didácticos para Microelectrónica.*- Investigación y desarrollo de equipamiento microelectrónica que permita su integración en Plataformas de Aprendizaje.

¹ DSP.- Es el acrónimo de *Digital Signal Processor*, que significa **Procesador Digital de Señal**, un nombre bastante descriptivo, pues su función no es otra sino recibir una señal como entrada, hacer unas operaciones sobre esa señal y sacar a su salida una nueva señal.

² FPGA.- Internamente una **FPGA** es una serie de pequeños dispositivos lógicos, que algunos fabricantes llaman **CLB**, organizados por filas y columnas. Entre los **CLB** (Configurable Logic Block, Bloque Lógico Configurable) hay un gran número de elementos de interconexión, líneas que pueden unir unos **CLB** con otros y con otras partes de la **FPGA**. Puede haber líneas de distintas velocidades. También hay pequeños elementos en cada una de las patillas del chip para definir la forma en que ésta trabajará (entrada, salida, entrada-salida...). Se suelen llamar **IOB** (Input/Output Block). Aparte de esta estructura, que es la básica, cada fabricante añade sus propias ideas, por ejemplo hay algunos que tienen varios planos con filas y columnas de **CLB**. Los **CLB** contienen en su interior elementos hardware programables que permiten que su funcionalidad sea elevada. También es habitual que contengan dispositivos de memoria.

3.- *Desarrollo de Equipos de Control Inalámbricos.*- Experimentación e Integración de capacidades de comunicación inalámbrica en equipos y sistemas electrónicos.

4.- *Procesadores Digitales de Señal DSP.*- Amplia experiencia en programación de DSP, en especial de las casas Analog Devices™ y Motorola™, habiendo desarrollado diferentes proyectos con estos DSP; en especial, con la Familia SHARC de Analog Devices™, utilizando como Herramienta de Desarrollo el IDE VisualDSP. Experiencia en el procesado de señal, tanto de voz como de imagen. En el área de procesado de voz, es de destacar un particular interés en los campos de síntesis, reconocimiento de voz, y especialmente, en regeneración de la voz.

5.- *Lógica Programable (FPGA y EPLD³).*- En el campo de la Lógica Programable se posee experiencia de trabajo con las FPGA de las casas ALTERA™ y XILINX™. La orientación principal son Sistemas Digitales de Procesado de Señal y Controladores de Transmisión de Datos.

6.- *Microcontroladores.*- Experiencia en el campo de los microcontroladores de Microchip™, tanto en la Programación en Ensamblador, como en Lenguaje C.

7.- *Diseño y Prototipo de Circuitos Impresos.* Experiencia de Diseño de Circuitos Impresos mediante la Herramienta Orcad Layout™ y Protel™, e implantación de los circuitos impresos en prototipo mediante procesos químicos y mediante procesos mecánicos (ProtoMat C20™).

Los principales **Proyectos de Investigación** se desarrollan en los siguientes incisos.

1.- *Automatización Integral de Piscifactorías.*- El objetivo del Proyecto consiste en medir todos los parámetros fundamentales que determinan el crecimiento de *truchas* para que puedan ser comercializadas en el mercado de consumo. Los parámetros que se controlan con el proyecto son: la temperatura, el pH, la cantidad de oxígeno presente en el agua, la turbiedad y la calidad del agua.

El sistema está basado en un Ordenador Maestro, en donde reside el Modelo Biológico destinado al preengorde de las *truchas*. El sistema se conecta a la Red a través de la interfase RS-485, ésta se conecta con los diferentes módulos de E/S basados en Microcontroladores PIC16F876.

³ EPLD.- Denominadas EPLD o simplemente PLD son circuitos lógicos programables eléctricamente y borrables por exposición a radiación ultravioleta. Denominadas EPLD o simplemente PLD son circuitos lógicos programables eléctricamente y borrables por exposición a radiación ultravioleta.

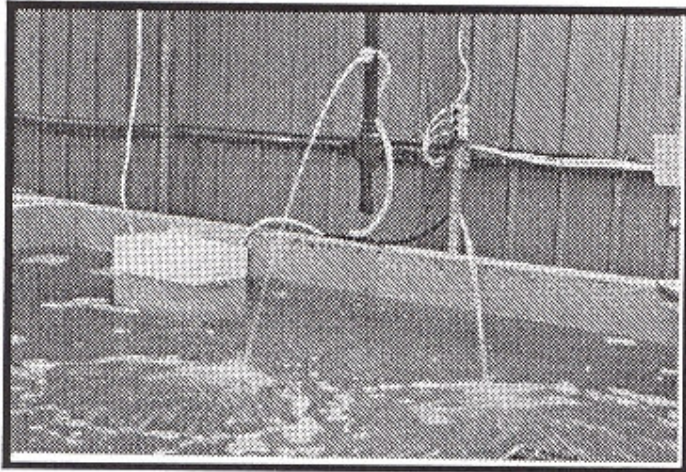


Figura IV.11.- Automatización Integral de Piscifactorías.

2.- *Análisis y Aplicaciones de los Microcontroladores PIC Avanzados.*- Con la colaboración de la Editorial Paraninfo Thomson Learning S.A., se desarrolló este proyecto que cubre tres objetivos. En primer lugar, la edición de un libro que se tituló "*Microcontroladores PIC: La Solución en un Chip*"; en el que se recoge de forma clara y didáctica toda la información técnica necesaria para comprender el funcionamiento de los Microcontroladores PIC de Altas Prestaciones, incluyendo los PIC18CXXX. El segundo objetivo que se alcanzó con este Proyecto, fue el desarrollo de una serie de aplicaciones prácticas que ayudan a comprender el diseño de aplicaciones con Microcontroladores. El tercer objetivo fue implantar un módulo de grabación y emulación de aplicaciones con el que se conseguía implantar los proyectos propuestos en el segundo objetivo.

3.- *Desarrollo e Implantación de un nuevo Algoritmo de Simplificación Booleana.*- Este Proyecto nace del estudio de los métodos clásicos de simplificación Booleana, para aportar uno nuevo, cuyas principales características son:

- Menor costo en tiempo para calcular los implicantes/implicadis primos de una Función.
- Menor costo en tiempo para calcular la expresión Booleana simplificada.
- Mejor bondad de las expresiones Booleanas simplificadas.
- Los resultados anteriores provienen de un estudio comparativo con los principales métodos de simplificado tradicionales.

Este Proyecto conjuga un estudio teórico con una aplicación informática compleja, tanto en representación, como en manipulación de datos en árboles binarios.

4.- *Desarrollo e Implantación de un nuevo Algoritmo para la Optimización del Área en un "Floorplan" No Particionado.*- El principal objetivo que se fijó en esta Investigación, consistió en determinar una implantación lo más adecuada y óptima posible, para cada uno de los módulos no particionados que conforman el "*Floorplanning*"₁, de modo que al unirse todos ellos se distribuyesen en un área lo mínimo posible. Para ello, se desarrolló e implantó un nuevo Algoritmo no Heurístico que optimice la resolución de este problema, empleando un tiempo de proceso menor que el mínimo requerido para ello en la actualidad.

5.- *Diseño y Simulación de Ordenadores Básicos.*- Este Proyecto de carácter decididamente didáctico tiene como objetivos:

- ◆ Diseñar Ordenadores básicos: Máquina sencilla, Máquina Plus y Máquina PlusPlus, MS, M+ y M++.
- ◆ Desarrollar simuladores interactivos basados en Tecnología Web. Las aplicaciones *jsp* desarrolladas, permiten simular a nivel de programa, a nivel de instrucción y microinstrucción o nivel de estado, cualquier programa del usuario.
- ◆ Implantar en FPGA los ordenadores mencionados en el primer inciso. Los ordenadores implantados deben poder ejecutar el programa cargado, visualizar los resultados en una terminal VGA, y comunicarse con un Ordenador Personal de Escritorio.
- ◆ Implantar un equipo didáctico que bajo el mando de un Microcontrolador PIC simule una Máquina MIPS, y permita su control y visualización por parte del usuario.

En este Proyecto, se reúnen tecnologías didácticas, de Internet y Java™, de implantación de FPGA y PIC de comunicaciones. El Proyecto muestra la capacidad del grupo para integrar tecnologías diversas en un Proyecto Didáctico.

6.- *Diseño de PathFinder: Un MicroRobot Inteligente.*- El Proyecto consiste en el diseño e implantación del Robot denominado *PathFinder* que complementa una obra de 100 fascículos para la Editorial F&G de España. De forma complementaria, se confeccionaron todos los manuales y textos necesarios para el seguimiento y construcción de dicho Robot móvil. Este Robot es un MicroRobot móvil que puede desplazarse con ruedas y con patas, mediante un motor de controla la tracción de las ruedas motrices y otro motor controla la dirección. Se programa con tarjetas inteligentes y, dispone de una cámara Web por la que envía imágenes a la TV o al Ordenador Personal vía Radio Frecuencias (RF); un sintetizador de voz que genera frases adecuadas a las acciones en curso, sensores infrarrojos, sonar brazo articulado y varios elementos que daban a la máquina un cierto grado de "inteligencia" que le permitía dedicarse a diferentes tareas.



Figura IV.12.- Diseño de un Micro-Robot Inteligente.

7.- *Diseño de un Entrenador Didáctico para Microelectrónica Programable.*- Para complementar un pedido de la Editorial Mc Graw-Hill, se diseñó un Entrenador Profesional y económico que servía de soporte para la realización de prácticas de diversos temas de la Microelectrónica Programable: Electrónica Digital, Electrónica Analógica, Semiconductores, Microcontroladores PIC Básicos, Microcontroladores PIC Avanzados, Lenguaje PBASIC y PLD.

El Entrenador se presenta en una maleta transportable, conteniendo todas las fuentes de alimentación, generadores de señales y conectores necesarios para que junto a una tarjeta tipo Protoboard muy amplia, y una serie de sensores de entrada (interruptores, pulsadores, sensor de temperatura, sensor de luminosidad, sensor de proximidad, etcétera) y actuadores de salida (LED, zumbador, Display de 7 Segmentos, Display LCD, etcétera) se puedan desarrollar colecciones de prácticas de carácter didáctico sobre las materias antes citadas. La finalización del proyecto supuso la edición de 2 libros que constituyen la obra "Laboratorio de Prácticas de Microelectrónica" de la Editorial Mc Graw-Hill y la comercialización del Entrenador Universal Trainer™.



Figura IV.13.- Diseño de un Entrenador Didáctico para Microelectrónica Programable.

8.- *EGC-Electrocardiógrafo Educativo Computarizado.*- Se pretende diseñar el material didáctico Electrocardiográfico necesario para su empleo en actividades de enseñanza, control deportivo o investigación. Este material se compone de dos elementos fundamentalmente:

- ◆ El Electrocardiógrafo propiamente dicho, que constituirá la parte de la Arquitectura del Sistema (*Hardware*). Este dispositivo será de tamaño reducido y se le conectarán los electrodos a las clavijas de su parte trasera para hacer las lecturas correspondientes. Dispondrá también de una interfase RS-232C para poder interactuar con un Ordenador haciendo uso de su conexión serial.
- ◆ La aplicación del Programa (*Software*) que gobierne al Electrocardiógrafo. Esta aplicación consistirá de un Programa a la medida, para el manejo por Ordenador del Electrocardiógrafo; además de mostrar en pantalla, los resultados de las pruebas. El programa estará realizado según los requerimientos de la Ingeniería de Programación, y se basará en la Plataforma de *Windows* de Microsoft™.

El ámbito de aplicación estará compuesto principalmente por alumnos relacionados con las Ciencias de la Salud, la Medicina y la Bioingeniería. Además de su uso en Centros Deportivos.

9.- *Entorno Computacional de Análisis y Diseño de Sistemas Digitales a Nivel de Bit.*- Este Entorno enfoca desde una variante Académica (pero también Profesional), el desarrollo de Aplicaciones Informáticas capaces de:

- ① Analizar y diseñar Sistemas Digitales Combinacionales a nivel de Bit: Expresiones Booleanas, Tablas de Verdad, Diagramas de Veitch-Karnaugh, Expresiones Simplificadas, Circuitos Lógicos, etcétera.
- ① Analizar y diseñar Sistemas Digitales Secuenciales a nivel de Bit como Autómatas: Diagramas de Moore y Mearly, Reducción de Autómatas, Simplificación de Ecuaciones, Flip-Flop J-K yD, Simulación de Autómatas, Circuitos Lógicos, etcétera.
- ① Tanto para Sistemas Combinacionales como Secuenciales, el Entorno Computacional es capaz de generar Código VHDL o ficeros *jedec* directamente grabables en un PLD, lo que permite un uso profesional e industrial de la Herramienta.

El Entorno Computacional actualmente está distribuido por Centros de Enseñanza de varios países europeos.

10.- *Guiado de Invidentes por medio del Sistema Global de Posicionamiento.*- El principal proyecto consistió en lograr un sistema de guiado para discapacitados; es un sistema autónomo y fácilmente portable y está basado en el cálculo de posicionamiento a través de GPS.

Para su desarrollo se partió del diseño de un prototipo de posicionador e interfase hablada con el usuario, que permite el guiado del usuario a través de un camino preestablecido. El prototipo consta de un receptor portátil GPS12 que entrega las coordenadas de situación, respecto a la red de satélites en longitud y latitud a un dispositivo IG1. Ambos dispositivos se conectan a través de un canal RS-232C. El dispositivo IG1 es también portátil, constituido por un Microcontrolador, un Micromódem y un Generador de Voz. Se emplea para cubrir tres misiones: conexión a la base de datos que contiene información para el cálculo del trayecto solicitado, calcular la posición exacta con respecto al recorrido almacenado y finalmente, dar las indicaciones necesarias al usuario a través del Generador de Voz. La Base de Datos con todas las características del recorrido es accedida automáticamente a través de la línea telefónica (móvil o fija).

El sistema se adapta a cualquier minusvalía: por una parte el trayecto generado reúne todas las características de interés para el minusválido como son: la situación de los semáforos, el rayado del paso de peatones, obras temporales en las aceras y las calles, mobiliario urbano en las aceras, escaleras, etcétera. Por otra parte, el trayecto almacena la ubicación de sitios de interés como paradas de taxi, autobús, estaciones del Metro, etcétera; lo que permite escoger recorridos alternativos, repetición de mensajes, imprevistos, etcétera.

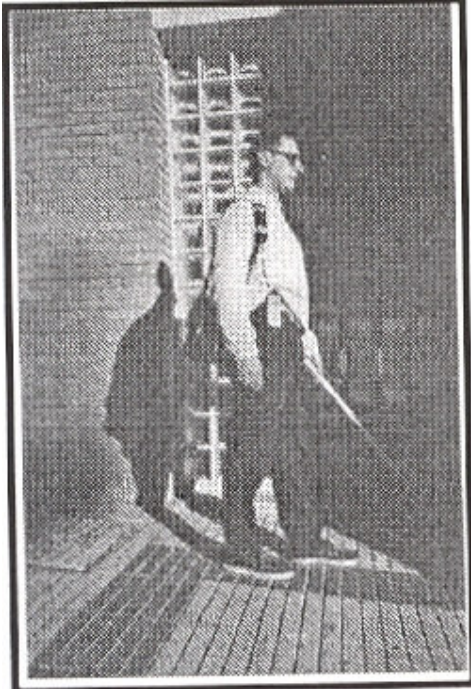


Figura IV.14.- Guiado de Invidentes por medio del Sistema Global de Posicionamiento, (GPS).

11.- *Implantación de un Sistema de Frenado.*- El Proyecto consistió en diseñar una tarjeta de control electrónico de frenado para el sistema de frenado de un Aerogenerador. La Empresa que lo requirió, solicitó los siguientes requisitos:

- Diseñar un sistema de medida de la velocidad de giro de la tripala del Aerogenerador. Este sistema debía ser distinto y redundante al ya existente.
- El sistema de frenado debía dispararse de modo automático al superar la tripala una velocidad de seguridad.
- El frenado no podía ser brusco, para de esta forma, evitar roturas y situaciones de emergencia. La rampa de frenado debía ser según una pendiente seleccionada por el usuario.
- El rearme del freno y liberación de la tripala debía ser manual.

12.- *COMMBOTS.*- El Proyecto COMMBOTS, tiene como objetivo la creación de una Planta de Experimentación con Robots Móviles para la ejecución de diversas tareas como la obtención y medición de datos mediante sensores; la realización de trabajos en diferentes entornos, la recolección de muestras o la grabación y transmisión de vídeo hacia una estación central de control mediante cámaras incorporadas. Dichos Robots estarán dotados de módulos de comunicación móvil GSM/GPRS que les permitirá recibir órdenes del puesto de control, así como, enviar resultados y datos de las actividades que estén realizando.

13.- *WebLab-PLD.*- El proyecto WebLab-PLD tiene como objetivo desarrollar una Tecnología capaz de abrir el laboratorio a los alumnos y a la Sociedad en general, a través de Internet. El objetivo del WebLab-PLD es potenciar el uso de los laboratorios con un menor costo y una mayor disponibilidad. El objetivo práctico es permitir que gracias a Internet el alumno/usuario acceda desde su casa u otro sitio, al equipo didáctico para poder programarlo, controlar sus entradas y observar sus salidas; es decir, WebLab-PLD permite que el alumno lleve el laboratorio a su casa.

14.- *LOGBOT*.- El objetivo de LOGBOT es la obtención de equipos de desarrollo didácticos, potentes, profesionales y baratos, incluyendo programación, grabación, implantación y sensores con CPLD, FPGA y VHDL. Como demostración, se diseñará un Microbot Industrial, basado y controlado por CPLD y FPGA con aplicaciones a la Microtecnología dotado de Microsensores, Microactuadores y Sistemas de Guiado por Rayos Infrarrojos para soportar tareas en el área de Microfabricación.

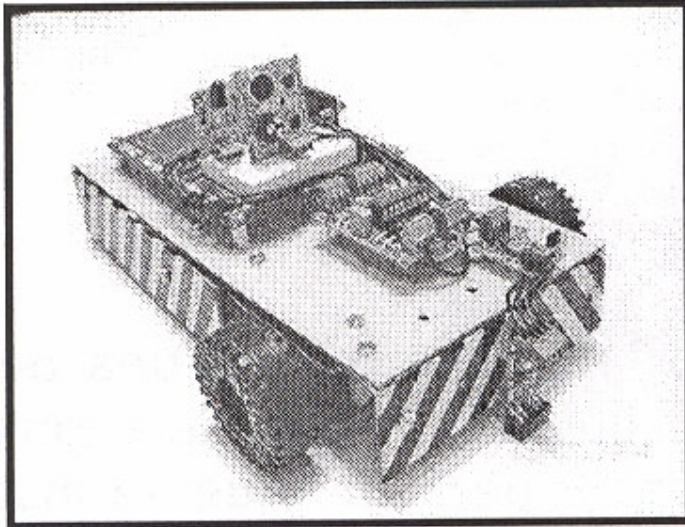


Figura IV.15.- Diseño de un LOGBOT para uso Didáctico.

15.- *Desarrollo de un Dispositivo Selector Automático del Operador telefónico más Rentable*.- Se ha diseñado un dispositivo al que se le introducen vía un Ordenador Personal, las tarifas telefónicas correspondientes a las Compañías Telefónicas con las que el Usuario tenga Contrato. Con estos datos, cada vez que se descuelgue el teléfono y se marque un número, el sistema detecta el tipo de llamada que se desea realizar (local o de larga distancia) y, en función de los datos de que dispone, selecciona la Compañía que permita realizar dicha llamada al mejor precio, marcando él automáticamente el prefijo correspondiente.

CONCLUSIONES.

- En algunos sistemas el cableado se realiza con un par trenzado, generando ahorro de mano de obra y gastos, además de una simplificación de proyecto.
- En otros sistemas, la instalación utiliza la red eléctrica de 120V evitando reformas físicas del recinto y posibilitando una rápida instalación. Para reformas de edificios, estos sistemas son ideales, ya que también evitan que el edificio esté en obra nuevamente.
- Controla todos los servicios, iluminación, aire acondicionado, alarmas, bombas, etcétera.
- Las unidades pueden ser programadas y reprogramadas, para optimizar el manejo de energía en cualquier momento con facilidad.
- El Sistema Inteligente es instalado y programado por un Ingeniero de acuerdo a los requerimientos del usuario y una vez instalado, no es necesario ningún conocimiento técnico para operarlo.
- Dada la alta flexibilidad que posee el sistema, en cualquier momento podrá actualizarlo y expandirlo con costos mínimos. Garantiza la funcionalidad de la edificación.
- El Sistema Inteligente permite integrar cualquier dispositivo que no sea inteligente al sistema.
- Simplifica enormemente el diseño de una obra, que podría ser dificultosa y costosa usando el cableado y dispositivos convencionales.

El proceso de planificación se reduce significativamente al igual que los cambios que demandan las edificaciones más modernas.

ANEXO: “GLOSARIO DE TÉRMINOS”.

ACOMETIDA ELÉCTRICA.- Instalación eléctrica de entrada a un Edificio, local o equipo.

ACS.- Agua caliente sanitaria.

AISLAMIENTO GALVÁNICO.- Aislamiento eléctrico a la corriente conducida de dos circuitos o partes metálicas.

ARMÓNICOS.- Efecto producido sobre la onda senoidal al conectarle cargas no lineales. Pueden ser de Intensidad de Corriente o de Tensión de Voltaje. La distorsión armónica es la medida del contenido de armónicos en una onda senoidal. Es la suma cuadrática de los mismos.

AUTOMATIZACIÓN.- Operación automáticamente controlada de un proceso o sistema realizada por dispositivos mecánicos o electrónicos que reemplazan al operador humano en las tareas de observación, realización de esfuerzos y toma de decisiones.

BABL.- Control de Longitud Ilegal (prueba de cableados de comunicaciones).

BACK-UP.- Copia o elementos de reserva o respaldo.

BT.- Baja Tensión (hasta 1,000 Volts).

BARRAS DE DISTRIBUCIÓN.- En general, conductor metálico rígido, casi siempre desprovisto de aislamiento, utilizado para conducir fuertes corrientes o para servir de conductor común a varios circuitos.

BATERÍAS.- Asociación de elementos electroquímicos que pueden transformar la energía química en energía eléctrica. Generalmente, se clasifican en dos tipos para su aplicación a los Sistemas de Alimentación Ininterrumpida (SAI): Plomo y Níquel-Cadmio. Las de Plomo además, pueden ser herméticas (sin necesidad de relleno de electrolito).

CARGAS.- Circuito, equipo o sistema que recibe alimentación eléctrica. En función del perjuicio económico que puede acarrear su alimentación incorrecta, se clasifican en críticas y no críticas.

CAÍDA DE ALIMENTACIÓN.- Disminución más o menos brusca de la alimentación eléctrica a un circuito o sistema.

CAÍDA DE TENSIÓN.- Diferencia de tensión entre dos puntos debida a la circulación de una corriente a través de una impedancia.

CCTV.- Circuito Cerrado de Televisión.

CEL.- Cuadro de Energía Limpia. (SAI).

CES.- Cuadro de Energía Sucia. (Red o grupo electrógeno).

CPD.- Centro de Proceso de Datos.

CRC.- Código de Paridad.

CS.- Centro de Seccionamiento.

CT.- Centro de Transformación.

ENERGÍA REACTIVA.- Energía que es incapaz de desarrollar trabajo.

ERGONOMÍA.- Ciencia que estudia el diseño de equipamiento del puesto de trabajo, para crear las mejores condiciones.

FACTOR DE POTENCIA DE LA CARGA.- Relación entre la potencia activa y la aparente de la carga, con tensión senoidal pura, con carga lineal el factor de potencia es igual al Coseno del ángulo de defasamiento.

FIABILIDAD.- Término genérico que denota la capacidad de un dispositivo, equipo o sistema para llevar a cabo de una misión determinada con ciertas condiciones.

GRADO DE INDEPENDENCIA.- En un edificio, es la capacidad del mismo para que en condiciones extremas de suministros exteriores pueda permanecer funcionando sin merma alguna.

GRADO DE REDUNDANCIA.- Se define en los sistemas de alimentación ininterrumpida como la capacidad de seguir funcionando al 100%, con fallas de una, dos o “n” módulos en un sistema de equipos múltiples.

IBR.- Manta de Fibra de Vidrio.

INTERRUPTOR.- Elemento destinado a la apertura de un circuito eléctrico. Si el circuito está en vacío, se denomina seccionador en vacío. Si el circuito está en carga, se tendrá dos tipos: el Disyuntor, para operación automática (incorpora relevadores) y el Seccionador en carga, para operación manual.

JAULA DE FARADAY.- También conocida como Pantalla de Faraday. Red de cables en paralelo conectados a un conductor común en un extremo para proporcionar un apantallamiento electrostático sin que le afecten las ondas electromagnéticas. El conductor común se pone a tierra generalmente.

PLAN DE CONTINGENCIA.- Plan de previsión ante desastres de todo tipo.

POTENCIA.- En los equipos eléctricos, se define como la capacidad de los mismos para producir trabajo.

PROTECCIONES.- Dispositivos que desconectan un circuito cuando las condiciones de diseño no son mantenidas. Existen de tensión y de intensidad.

RED DE DATOS.- Interconexión de una serie de puntos mediante instalaciones de comunicaciones. La arquitectura de sistemas que lo soporta está generalmente formado por multiplexores, módems, adaptadores de líneas y ordenadores.

RED EQUIPOTENCIAL.- Conjunto De elementos que están todos conectados al mismo punto de puesta a tierra.

PVF.- Factor de Velocidad de Propagación.

SAI.- Sistema de alimentación ininterrumpida (también denominado UPS. Equipo electrónico para mantener alimentadas las cargas denominadas críticas, de forma segura y de buena calidad, tanto en presencia de red como ausencia de ésta. En este caso, durante un período de tiempo determinado por el tiempo de autonomía de las baterías.

SELECTIVIDAD.- Cuando se disponen de varios dispositivos de protección en serie se requiere que éstos sean selectivos. La protección de la red se considera selectiva si solamente se desconecta el dispositivo de protección más próximo al punto de defecto por delante del mismo. Existen dos tipos de selectividad: la cronometría (tiempo) y la intensidad.

SQE.- Error de Calidad de la Señal.

TENSIÓN DE ENTRADA NOMINAL.- Valor eficaz de la tensión que alimentará el equipo y que se toma como indicativo.

TENSIÓN NOMINAL DE LA BATERÍA.- La que resulta de multiplicar el número de vasos de la batería por su tensión nominal (2 Volts para Plomo y 1.2 Volts para Ni-Cd).

TRANSCIVERS.- Órganos activos en sistemas de comunicaciones (voz y datos).

TRANSFORMADORES DE AISLAMIENTO.- Transformador que tiene un devanado primario aislado galvánicamente del secundario.

TRIGENERACIÓN.- Coproducción conjunta de electricidad, calor y frío (es una variante de la cogeneración).

UTA.- Unidad de Tratamiento de Aire.

VIDA MEDIA.- Tiempo durante el cual el edificio funciona correctamente bajo las condiciones normales de servicio.

VRV.- Volumen de Refrigerante Variable.

BIBLIOGRAFÍA.

Angulo Usategui, J. M. (1997). **Microcontroladores PIC. Diseño Práctico de Aplicaciones.** España: McGraw-Hill.

Betham, J. (2002). **TCP/IP Lean Web Servers for Embedded Systems.** USA: CMP Books, 2nd edition.

Bosch, R. (1991). **CAN Specification Version 2.0.** USA: CAN Edition.

Braden R., ISI, Borman, Cray Research, Partridge C., BNN Laboratories. (1988). **Computing the Internet checksum.** USA: RFC 1071.

Cisco System. (s/f). **Internetworking Technology Handbook.** [En línea o URL], en http://www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/index.htm

DP8390/NS32490D, NIC Network. (1995). **Interface Controller.** USA: National Semiconductors

Dunkel A., Alonso J., Voigt T. (2004). **Making TCP/IP Viable for Wireless Sensors Networks.** Swedish Institute of Computer Science.

Ferrari, P., Flammini, D., Marioli, E., Sisinni, E, y Taroni, A. (2005). **A Bluetooth™-Based Sensor Network with Web Interface.** USA: IEEE Transactions on Instrumentation and Measurement.

Floery, S. (s/f). **Proyecto PICNIC. Version: 2.1 Build 7.** [En línea o URL], en <http://picnic.sourceforge.net>

Fox R. (1989). **TCP Big Windows and NAK Options.** USA: RFC 1106. Networking Working Group.

Griffiths, R. T. (2002). **History of the Internet, Internet for Historians (and just about everyone else).** (Universiteit Leiden) [En línea o URL], en http://www.let.leidenuniv.nl/history/ivh/frame_theorie.html

Hersch R. (1999). **68CH11 Microcontroller FAQ.** [En línea o URL], en <ftp://rtfm.mit.edu/faqs/microcontroller-faq/8051/>

Hersch R. (1997). **8051 Microcontroller FAQ.** [En línea o URL], en <http://www.faqs.org/faqs/microcontroller-faq/8051/>

Hertel, J. W. (2001). **LONWORKS in Industrial.** USA: Proceedings, 8TH IEEE International Conference on Emerging Technologies and Factory Automation.

Internet Society (ISOC), (s/f). **All About the Internet: History of the Internet.** [En línea o URL], en http://www.isoc.org/internet/history/brief.shtml#Initial_Concepts

J.C.R. Licklider & W. Clark, (1962). **On-Line Man Computer Communication.** USA: Mc Graw-Hill.

Jacobson, V., Braden, R., Borman D. (1992). **TCP Extensions for High Performance.** RFC 1323. USA: Networking Working Group.

Kaschel, H. y Pinto, E. (2002). **Análisis del Estado del Arte de los Buses de campo Aplicados al Control de Procesos Industriales.**

Kolm , Pavlu , Staub , Zlabinger. (s/f). **Proyecto MONTPI. Version: 0.1f.** [En línea o URL] en <http://plain.at/monpti/>

Michael, B. (2001). **Programming Embedded Systems in C and C++**. USA: O' Reilly Publisher.

(s/a). (2001). **PIC 16F877XA DataSheet 28/40 pin Enhanced Flash Microcontrollers**. USA: Microchip Technology Inc.

(s/a). (2001). **PIC 18FXX2 DataSheet High Performance, Enhanced Flash Microcontrollers with 10-bit A/D**. USA: Microchip Technology Inc.

(s/a). (2001). **PICDEM.net User Guide**. USA: Microchip Technology Inc.

(s/a). (1997). **PICmicro mid-range MCU family**. USA: Microchip Technology Inc.

Plummer, D. C. (1982). **Ethernet Address Resolution Protocol or Converting Network Protocol Addresses to 48 Bit Ethernet Address for Transmission on Ethernet Hardware**. USA: Mc Graw-Hill.

Postel, J. (1981). **Assigned Numbers**. USA: RFC 790. (USC/Information Sciences Institute. DARPA Internet Program).

Postel, J. (1981). **Internet Message Control Protocol**. USA: RFC 792. (USC/Information Sciences Institute. DARPA Internet Program).

Postel, J. (1981). **Transmission Control Protocol**. USA: RFC 793. (USC/Information Sciences Institute. DARPA Internet Program).

Postel, J. (1983). **Character Generator**. USA: RFC 864. (USC/Information Sciences Institute. DARPA Internet Program).

Postel, J. (1983). **Echo Protocol**. USA: RFC 862. (USC/Information Sciences Institute. DARPA Internet Program).

Postel, J. (1981). **Internet Protocol Specification**. USA: RFC 791, (USC/Information Sciences Institute. DARPA Internet Program).

Postel, J. (1980). **User Datagram Protocol**. USA: RFC 768 (USC/Information Sciences Institute. DARPA Internet Program).

Schildt, H. (2001). **JAVA 2: The Complete Reference**. USA: Mc Graw-Hill, 4TH edition.

Stevens, R. (1995). **TCP/IP Illustrated**. Vol. I y II. USA: Adisson Wesley.

Tanenbaum, A., S. (2000). **Computers Networks**. USA: Prentice Hall, 4TH edition.