



**UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO**

**FACULTAD DE ESTUDIOS SUPERIORES
ARAGÓN**

**“IMPLEMENTACION Y APLICACIÓN DE
GRABADORES PARA CIRCUITOS
INTEGRADOS”**

**MODALIDAD DE TITULACIÓN:
MEMORIA DE DESEMPEÑO DE SERVICIO SOCIAL**

**QUE PARA OBTENER EL TÍTULO DE
INGENIERO EN COMPUTACIÓN**

P R E S E N T A:

SERGIO ALBERTO GONZALEZ GALLEGOS

ASESOR:

ING. ENRIQUE GARCÍA GUZMÁN



MÉXICO

2009



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.



AGRADECIMIENTOS

Mi trabajo de titulación lo dedico con todo mi amor y cariño.

Doy infinitas gracias...

A ti DIOS que me diste la oportunidad de vivir y de regalarme una familia maravillosa.

A la vida... Por lo aprendido y aprehendido

Con mucho cariño principalmente a mis padres que me dieron la vida, a mi Madre que ha estado conmigo en todo momento. Gracias por haberme dado una carrera para mi futuro y por creer en mí, aunque hemos pasado momentos difíciles siempre has estado apoyándome y motivándome a salir adelante, por todo esto, te agradezco de todo corazón el que estés conmigo, aquí está lo que me has brindado, solamente te estoy devolviendo lo que me diste en un principio.

A mis hermanas Marcela y Rosa Isela gracias por estar conmigo, apoyarme y ayudarme siempre, por sus buenos ejemplos a seguir, sus consejos y palabras de apoyo, las quiero mucho. Y a Carlos Brindis por siempre motivarme a seguir adelante, hacerme ver mis errores y aconsejarme en todo momento, los quiero.

Ana Lilia, que te puedo decir Amor, te agradezco por todo este tiempo de conocernos, estar juntos en las buenas y en las malas, donde hemos compartido tantas cosas, pasado tantos momentos maravillosos en los cuales has estado conmigo, sobre todo en este día tan importante para mí. Solo quiero darte las gracias por todo el amor y apoyo que me has brindado para continuar con mi camino, recuerda que eres muy importante para mí.

A todos mis amigos y compañeros de la Facultad, por estar conmigo en todo este tiempo donde he vivido momentos felices y tristes, gracias por ser mis amigos y recuerden que siempre los llevare en mi corazón.

A Fernando Zayago, que siempre estuvo ayudándome, escuchándome, apoyándome y regañándome cuando era necesario, gracias por ser el mejor de los amigos.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS
INTEGRADOS



A Ricardo Perez y Georgina Beceril, por apoyarme en el ámbito laboral y darme la oportunidad de crecer profesionalmente, por escucharme y aconsejarme en los momentos difíciles.

A LA FACULTAD DE ESTUDIOS PROFESIONALES ARAGÓN

Por haberme brindado un lugar para estudiar y así poder desarrollarme profesionalmente y como persona.

A todos mis profesores que gracias a sus enseñanzas y conocimientos hoy me dan la oportunidad de titularme.

A todos mis Asesores que me ayudaron, apoyaron y brindaron facilidades con mi trabajo.

Y no me puedo ir sin antes decirles, que sin ustedes a mi lado no lo hubiera logrado, tantas desveladas sirvieron de algo y aquí está el fruto. Les agradezco a todos ustedes con toda mi alma el haber llegado a mi vida y el compartir momentos agradables y tristes, pero esos momentos son los que nos hacen crecer y valorar a las personas que nos rodean.

Los quiero mucho y nunca los olvidare.

A todos y cada uno de Uds. Mil gracias de todo corazón, que DIOS los bendiga por haber llegado a mi vida.

Detrás de cada línea de llegada, hay una de partida.

Detrás de cada logro, hay otro desafío.

Si extrañas lo que hacías, vuelve a hacerlo.

Sigue aunque todos esperen que abandones.

No dejes que se oxide el hierro que hay en ti.

He aprendido....Que la mejor aula de aprendizaje está a los pies de una persona mayor.

Sergio Alberto Gonzalez Gallegos 2009



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



INDICE

Planteamiento del problema	1
Justificación	2
Objetivo General	3
Objetivos Específicos	3
Capitulo 1. Marco Teórico	4
1.1 Historia de los Circuitos Integrados	4
1.2 Definición de conceptos	5
1.2.1 Circuito integrado	5
1.2.1.1 Clasificación de Circuitos integrados	6
1.2.1.2 Creación de Circuitos integrados	7
1.2.2 Transistor	8
1.2.3 Diodo	8
1.2.4 Resistencia	9
1.3 Detalle de los Circuitos Integrados	9
1.3.1 SSI	9
1.3.2 VLSI	10
1.3.3 MOS	11
1.3.4 CMOS	11
1.3.5 Principales fabricantes	11
1.4 Arquitectura Harvard y Von Neumann	13
1.5 Diagramas lógicos de Circuitos Integrados	15
1.6 Programa fuente	16
1.7 Lenguaje Ensamblador	16
1.7.1 Características	17
1.7.2 Ensambladores	17
1.7.3 Lenguaje	18
1.7.4 Código máquina (o lenguaje de máquina)	19
1.7.5 Ventajas del Ensamblador	21
1.8 Una Mirada Rápida al MPLAB	21
1.8.1 Las Herramientas del MPLAB	22
1.8.2 Software ensamblador	22
1.8.3 Simulación	25
1.8.4 Conversión a ASCII	25
1.8.5 Aritmética	26
1.9 Creación de un Grabador digital	27
1.9.1 Operación e Interconexión del grabador Digital	30



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



Capítulo 2. Implementación y Aplicación de los Grabadores	32
2.1 Introducción al Programador 844A BK PRECISION	32
2.1.1 Descripción en detalle del Programador y Software	32
2.1.2 Elementos del 844 ^a	35
2.1.3 Descripción del conector de ISP del 844A	36
2.1.3.1 Advertencias:	36
2.2 Requisitos mínimos de la PC, para la Implementación	37
2.2.1 Instalación del Hardware del programador	37
2.2.2 Instalación del Software del programador	37
2.2.3 Nuevas versiones del software del programador	38
2.2.4 Actualizando el software del programador	39
2.3 Aplicando el Programador 844A en la PC	39
2.3.2 Auto prueba y calibración	39
2.3.2 Extracción Correcta del Programador de la PC.	39
Capítulo 3. Utilización de grabadores con diferentes C.I.	40
3.1 Barra de Menú	40
3.2 Lista de Accesos Rápidos	40
3.3 Archivo	40
3.3.1 Archivo/Carga	40
3.3.2 Archivo/Guardar	41
3.3.3 Archivo/Carga del Proyecto	41
3.3.4 Archivo/Salvar Proyecto	41
3.3.5 Archivo/Recarga Proyecto	42
3.3.6 Archivo/Opciones del Proyecto	42
3.3.7 Archivo/Carga la Tabla de encriptación	43
3.3.8 Archivo/Salvar tabla de encriptación	43
3.3.9 Archivo/Salir sin salvar	43
3.3.9.1 Archivo/Salir y salvar	43
3.4 Dispositivo	43
3.4.1 Seleccionar dispositivo	43
3.4.1.1 Dispositivo/Selecciona de los dispositivos de default	43
3.4.1.2 Seleccionar Dispositivo/Todo	44
3.4.1.3 Selecciona Dispositivo/Solamente del tipo seleccionado	44
3.4.1.4 Seleccionar Dispositivo/Solamente del fabricante seleccionado	44
3.4.2 Dispositivo/Selecciona EPROM/flash por ID	45
3.4.3 Dispositivo/ Opciones de Dispositivo/ Opciones de Operación	46
3.4.3.1 Lista de Campos	46
3.4.3.2 Dispositivos/ Opciones del Dispositivo/ Serialización	47
3.4.3.3 Dispositivo/ Opciones del Dispositivo/ Serialización/ Modo de Incremento	48



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



3.4.3.4 Dispositivo/ Opciones del Dispositivo/ Serialización/ Modo del Archivo	48
3.4.4 Dirección (Address)	48
3.4.5 Valor Inicial	48
3.4.6 Paso	48
3.4.7 Estilo	49
3.4.8 Salvar en el buffer	49
3.5 Nombre de Archivo	49
3.5.1 Etiqueta de Inicio	49
3.5.2 Dispositivo/ Verifica Espacio en Blanco	49
3.5.3 Dispositivo/ Leer	50
3.5.4 Dispositivo/ Verificar	50
3.5.5 Dispositivos / Programar	50
3.5.6 Dispositivo/ Borrar	50
3.5.7 Dispositivo/ Prueba	50
3.5.8 Dispositivo / Información del Dispositivo	50
3.6 Opciones/ Opciones Generales	51
3.6.1 Opciones de Archivo	51
3.6.2 Opciones de Archivo Hexadecimal	52
3.6.3 Lenguaje	52
3.6.4 Sonido	52
3.6.5 Otros	53
3.7 Programando un dispositivo - La manera más corta	53
3.8 Manipulación con el dispositivo programado	53
CONCLUSIONES	54
BIBLIOGRAFIA	55



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



PLANTEAMIENTO DEL PROBLEMA

La Dirección General de la Jefatura de Carrera de Ingeniería en Computación (ICO) ha utilizado el laboratorio L3 para hospedar los grabadores y las aplicaciones para grabar circuitos Integrados (CI), utilizados en las diferentes materias de las carreras a fin, que utilizan este material, de esta institución (FES ARAGON).

Las características técnicas necesarias que se utilizan hasta el momento es una PC de escritorio con memoria de 256MB, capacidad en disco duro de 40 GB y un Sistema Operativo XP Profesional,

A través del tiempo se ha tenido mayor demanda de crecimiento en cuanto a usuarios debido a la demanda en diversos proyectos de las diferentes materias impartidas en la facultad. Esto se ha solucionado con la implementación de más grabadores y con la enseñanza a los alumnos y profesores de la utilización del material para el desarrollo de sus proyectos tanto personales, escolares y profesionales

La configuración de estos grabadores hasta el semestre anterior, era de 1 grabador para el uso de más de 600 usuarios por semestre, distribuido en una sala de préstamo de material para laboratorio en el laboratorio L3 con un horario de 4:00PM a 8:00PM

El inconveniente es que en el año 2008 se dieron a la tarea los diversos maestros de las diferentes materias en desarrollar sus proyectos al 100% y así dejar a los alumnos que tuvieran que ver con proyectos que integraran Circuitos Integrados Programables, por lo que ya no era posible abastecer las necesidades de todos y cada uno de los alumnos, incrementar la capacidad de permitir acceso a los grabadores y al conocimiento de su utilización, procesamiento del equipo para satisfacer las necesidades de crecimiento de aplicaciones de las materias de la Facultad.

La FES Aragón ha determinado que se han aumentado en un promedio de 700 alumnos por semestre, lo que implicaría en un futuro, mas demanda de recursos en el equipo.

Además, BK Precisión dio a conocer que existen diversos grabadores con tiempo de vida muy alto a los cuales se les puede dar Soporte Técnico con fines de mantenimiento preventivo y correctivo.

Con todos estos problemas detectados se determino renovar la infraestructura de la Facultad con el objetivo de tener un crecimiento escalable a través de los años para poder alojar más y nuevas aplicaciones para satisfacer la demanda de crecimiento de las actuales.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



JUSTIFICACIÓN

La Facultad de Estudios Profesionales Aragón en el área de Ingeniería en computación requiere una solución que le permita alojar a los grabadores con el equipo necesario para su adecuado funcionamiento, soportar la gran demanda que se generara cada semestre, así como un modelo que soporte un esquema de crecimiento a futuro, además de que los equipos son de última generación con arquitectura de compatibilidad de 64 bits.

La solución contempla los siguientes ambientes:

- Producción (servicio a sistemas operacionales).
- Desarrollo (empleado para la nueva aplicación y utilización de la existente).
- Garantía (relacionado al desarrollo de la nueva aplicación y significa una enseñanza estable y congelada obtenida en un desarrollo terminado).

La plataforma elegida para este proyecto es el BK Precisión Programador 844A que es de última generación con capacidad suficiente para soportar las aplicaciones estratégicas de la Facultad, satisfaciendo los niveles de servicio establecidos con la facilidad de brindar un mejor rendimiento y fiabilidad para los aplicativos.

Las características principales para instalar el BK Precisión Programador 844A son:

- PC Pentium a 100MHz o más velocidad.
- 512MB libres de Memoria RAM.
- Unidad de CD-ROM.
- 10 GB de espacio libre en Disco Duro.
- Sistema Operativo MS WINDOWS 95/98/Me/NT/2000/XP.
- Puerto paralelo libre (de la impresora) bus PCI, compatible con IEEE 1284 (ECP/EPP).



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



OBJETIVO GENERAL

Contar con una infraestructura capaz de cubrir la demanda de todos y cada uno de los proyectos requeridos por los alumnos para el desarrollo total de sus habilidades en las asignaturas dedicadas a mostrar el funcionamiento real de los Circuitos Integrados (CI).

OBJETIVOS ESPECIFICOS

- Incrementar el área para la utilización de grabadores.
- Implementar la mayor cantidad posible de grabadores en el área de trabajo.
- Dar a conocer a los alumnos y profesores el conocimiento de dichos grabadores, así como de su correcta utilización.
- Dar mantenimiento al software utilizado por las aplicaciones con las que se cuenta.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



Capítulo 1 Marco Teórico

1.1 Historia de los Circuitos Integrados

En Abril de 1949, el ingeniero alemán Werner Jacobi (Siemens AG) completa la primera solicitud de patente para circuitos integrados como semiconductores amplificadores de dispositivos. Jacobi realizó una típica aplicación industrial para su patente, la cual no fue reportada.

El primer circuito integrado (CI) fue desarrollado en 1958 por el ingeniero Jack Kilby justo meses después de haber sido contratado por la firma Texas Instruments. Se trataba de un dispositivo de germanio que integraba seis transistores en una misma base semiconductor para formar un oscilador de rotación de fase.

La historia de los circuitos integrados podría explicar un poco por qué nuestro mundo esta lleno de estos.

El microprocesador es un circuito integrado que procesa toda la información en una computadora; este mantiene un registro de las teclas que se han presionado y de los movimientos del ratón, cuenta los números y ejecuta los programas, juegos y el sistema operativo.

Los circuitos integrados también se encuentran en todos los aparatos electrónicos modernos como lo son los automóviles, televisores, reproductores de CD, reproductores de MP3, teléfonos móviles, etc.

Los circuitos integrados (CI) fueron posibles gracias a descubrimientos experimentales que demostraron que los semiconductores pueden realizar las funciones de los tubos de vacío. La integración de grandes cantidades de diminutos transistores en pequeños chips fue un enorme avance sobre el ensamblaje manual de los tubos de vacío (válvulas) y circuitos utilizando componentes discretos.

Con el transcurso de los años, los circuitos integrados (CI) están constantemente migrando a tamaños más pequeños con mejores características, permitiendo que mayor cantidad de circuitos sean empaquetados en cada chip. Al mismo tiempo que el tamaño se comprime, prácticamente todo se mejora (el costo y el consumo de energía disminuyen a la vez que aumenta la velocidad). Aunque estas ganancias son aparentemente para el usuario final, existe una feroz competencia entre los fabricantes para utilizar geometrías cada vez más delgadas.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



Este proceso, y el esperado en los próximos años, está muy bien descrito por el plan de trabajo para la Internacional de Tecnología de Semiconductores (International Technology Roadmap for Semiconductors, o, ITRS.)

La capacidad de producción masiva de circuitos integrados, confiabilidad y facilidad de agregarles complejidad, impuso la estandarización de los circuitos integrados (CI) en lugar de diseños utilizando transistores que pronto dejaron obsoletas a las válvulas o tubos de vacío.

Existen dos ventajas principales de los circuitos integrados (CI) sobre los circuitos convencionales: costo y rendimiento. El bajo costo es debido a que los chips, con todos sus componentes, son impresos como una sola pieza por fotolitografía y no construidos por transistores de a uno por vez.

Solo ha trascurrido medio siglo desde que se inició su desarrollo y los circuitos integrados (CI) se han vuelto casi omnipresentes. Computadoras, teléfonos móviles y otras aplicaciones digitales son ahora partes infaltables de las sociedades modernas. La informática, las comunicaciones, la manufactura y los sistemas de transporte, incluyendo Internet, todos dependen de la existencia de los circuitos integrados (CI). De hecho, muchos estudiosos piensan que la revolución digital causada por los circuitos integrados (CI) es uno de los sucesos más significativos de la historia de la humanidad.

1.2 Definición de conceptos

1.2.1 Circuito integrado

Un circuito integrado (CI) o chip, es una pastilla muy delgada en la que se encuentra una enorme cantidad (del orden de miles o millones) de dispositivos micros electrónicos interconectados, principalmente diodos y transistores, además de componentes pasivos como resistencias o condensadores. Su área es de tamaño reducido, del orden de un cm^2 o inferior. Algunos de los circuitos integrados más avanzados son los microprocesadores, que son usados en múltiples artefactos, desde computadoras hasta electrodomésticos, pasando por los teléfonos móviles. Otra familia importante de circuitos integrados la constituyen las memorias digitales.

Existen tres tipos de circuitos integrados (CI):

- **Circuitos monolíticos:** Están fabricados en un solo mono cristal, habitualmente de silicio, pero también existen en germanio, arseniuro de galio, silicio-germanio.

- **Circuitos híbridos de capa fina:** Son muy similares a los circuitos monolíticos, pero, además, contienen componentes difíciles de fabricar con tecnología monolítica.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



Muchos conversores A/D y conversores D/A se fabricaron en tecnología híbrida hasta que los progresos en la tecnología permitieron fabricar resistencias precisas.

- **Circuitos híbridos de capa gruesa:** Se apartan bastante de los circuitos monolíticos. De hecho suelen contener circuitos monolíticos sin cápsula, transistores, diodos, sobre un sustrato dieléctrico, interconectados con pistas conductoras. Las resistencias se depositan por serigrafía y se ajustan haciéndoles cortes con láser. Todo ello se encapsula, tanto en cápsulas plásticas como metálicas, dependiendo de la disipación de potencia que necesiten.

En muchos casos, la cápsula no está "moldeada", sino que simplemente consiste en una resina epoxi que protege el circuito. En el mercado se encuentran circuitos híbridos para módulos de transmisión, fuentes de alimentación, circuitos de encendido para automóvil.

1.2.2 Clasificación de Circuitos integrados

Atendiendo al nivel de integración - número de componentes - los circuitos integrados se clasifican en:

- SSI (*Small Scale Integration*) pequeño nivel: inferior a 12.
- MSI (*Medium Scale Integration*) medio: 12 a 100.
- LSI (*Large Scale Integration*) grande: 100 a 1 000.
- VLSI (*Very Large Scale Integration*) muy grande: 10 000 a 99 999.
- ULSI (*Ultra Large Scale Integration*) ultra grande: igual o superior a 100 000.

En cuanto a las funciones integradas, los circuitos se clasifican en dos grandes grupos:

- **Circuitos integrados analógicos.**

Pueden constar desde simples transistores encapsulados juntos, sin unión entre ellos, hasta dispositivos completos como amplificadores, osciladores o incluso receptores de radio completos.

- **Circuitos integrados digitales.**

Pueden ser desde básicas puertas lógicas (Y, O, NO) hasta los más complicados microprocesadores, estos son diseñados y fabricados para cumplir una función específica dentro de un sistema. En general, la fabricación de los circuitos integrados (CI) es compleja ya que tienen una alta integración de componentes en un espacio muy reducido de forma que llegan a ser microscópicos. Sin embargo, permiten grandes simplificaciones con respecto los antiguos circuitos, además de un montaje más rápido.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



- **Circuitos de aplicación específica:** circuitos diseñados para una función concreta (tarjeta de sonido, de video, amplificadores, temporizadores, reguladores.)
- **Circuitos de propósito general:** aquellos circuitos que pueden realizar diferentes funciones (microcontroladores, familia 74XX y 40XX).
- **Circuitos programables:** presentan características intermedias a los anteriores (Dispositivos Lógicos Programables (PLD), Arrays de Puertas Programables (FPGA)).

1.2.3 Creación de Circuitos integrados

Los circuitos integrados (CI) son unidades funcionales completas, esto no quiere decir que por si mismos son capaces de cumplir la función para los que estén diseñados, para ello serán necesarios unos componentes pasivos y activos para completar dicha funcionalidad, si los circuitos integrados no existieran las placas de circuito impreso para los aparatos serían muy grandes y además estarían llenos de componentes, este tipo de dispositivos, por su diseño, son capaces de albergar en su interior y de forma casi microscópica, gran cantidad de componentes sobre todo los semiconductores. No todos los componentes electrónicos se pueden integrar con la misma facilidad:

- Como antes se indicó los semiconductores, básicamente, los transistores y diodos, presentan menos problemas y menor costo en la integración. Igualmente tanto resistencias como condensadores se pueden integrar pero aumenta el costo.
- Por último las bobinas no se integran por la dificultad física que entrañan, así mismo ocurre con relés, cristales de cuarzo, displays, transformadores y componentes tanto pasivos como activos que disipan una potencia considerable respecto de la que podrían soportar una vez integrados.

El proceso de fabricación de un circuito integrado es como se observa en la figura 1.1 de un modo esquemático:

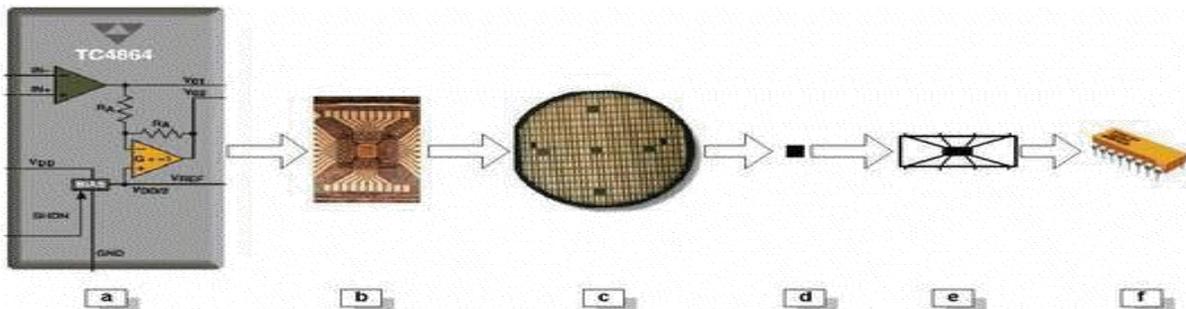


Figura 1.1 Elaboración de circuitos integrados (CI)

Ingeniería en Computación



1.3.1 Transistor

El Transistor es un dispositivo electrónico semiconductor que cumple funciones de amplificador, oscilador, conmutador o rectificador (figura 1.2). El término "transistor" es la contracción en inglés de transfer resistor ("resistencia de transferencia"). Actualmente se los encuentra prácticamente en todos los enseres domésticos de uso diario: radios, televisores, grabadores, reproductores de audio y vídeo, hornos de microondas, lavadoras, automóviles, equipos de refrigeración, alarmas, relojes de cuarzo, computadoras, calculadoras, impresoras, lámparas fluorescentes, equipos de rayos X, tomógrafos, ecógrafos, reproductores mp3, celulares.

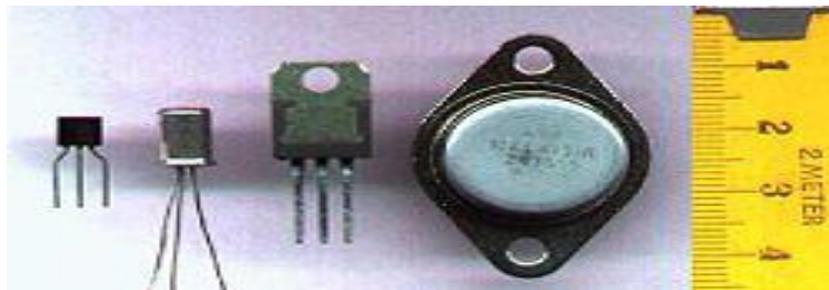


Figura 1.2 Diferentes tipos de transistores

1.2.3 Diodo

Un diodo (del griego "dos caminos") es un dispositivo semiconductor que permite el paso de la corriente eléctrica en una única dirección con características similares a un interruptor. De forma simplificada, la curva característica de un diodo consta de dos regiones: por debajo de cierta diferencia de potencial, se comporta como un circuito abierto (no conduce), y por encima de ella como un cortó circuito con muy pequeña resistencia eléctrica (figura 1.3).

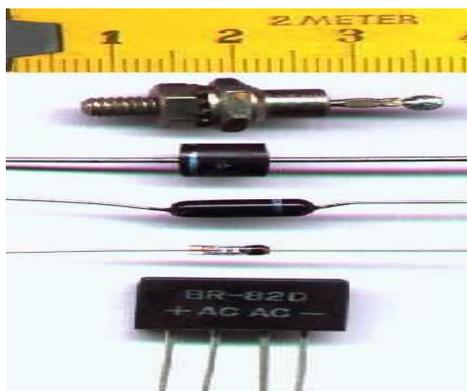


Figura 1.3 Tipos de diodos de estado sólido



1.2.4 Resistencia

Se denomina resistencia o resistor al componente electrónico diseñado para introducir una resistencia eléctrica determinada entre dos puntos de un circuito. En otros casos, como en las planchas, calentadores, etc., las resistencias se emplean para producir calor aprovechando el efecto Joule. Entre los técnicos es frecuente utilizar la palabra resistor como sinónimo de resistencia (figura 1.4).

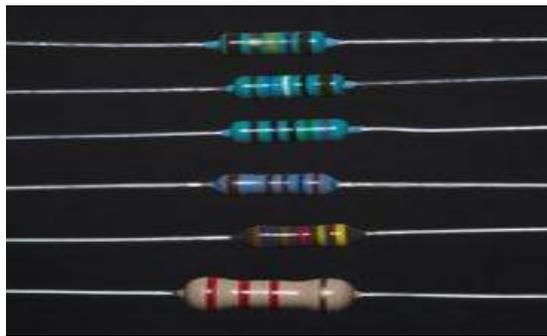


Figura 1.4 Tipos de Resistencias

1.3 Detalle de los circuitos integrados

El diseño de circuitos integrados (CI) es la parte de la electrónica que estudia distintas metodologías con el fin de desarrollar un circuito electrónico, que puede ser tanto analógico como digital, en función del número de componentes que forman el circuito integrado se habla de diferentes escalas de integración.

Las fronteras entre las distintas escalas son difusas, pero se denominan SSI (Small Scale of Integration) los circuitos de baja complejidad (algunas docenas de componentes en un mismo chip), MSI (Medium Scale of Integration) y LSI (Large Scale Integration) los circuitos de media y alta complejidad, y finalmente VLSI (Very Large Scale Integration) para circuitos extraordinariamente complejos, hasta cientos de millones de transistores. En esta última categoría entrarían los microprocesadores modernos.

1.3.1 SSI

SSI es acrónimo del inglés Small-Scale Integration (integración a baja escala) y hace referencia a los primeros circuitos integrados que se desarrollaron. Cumplían funciones muy básicas, como puertas lógicas y abarcan desde unos pocos transistores hasta una centena de ellos.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



Los circuitos SSI fueron cruciales en los primeros proyectos aeroespaciales, y viceversa, ya que los programas espaciales como Apollo o el misil Minuteman necesitaban dispositivos digitales ligeros. El primero motivó y guió el desarrollo de la tecnología de circuitos integrados (CI), mientras que el segundo hizo que se realizara una producción masiva.

Estos programas compraron prácticamente la totalidad de los circuitos integrados (CI) desde 1960 a 1963, y fueron los causantes de la fuerte demanda que originó un descenso de los precios en la producción de 1000 dólares la unidad (en dólares de 1960) hasta apenas 25 dólares la unidad (en dólares de 1963).

El siguiente paso en el desarrollo de los circuitos integrados, que tuvo lugar a finales de los 60, introdujo dispositivos que contenían cientos de transistores en cada chip y fue llamado MSI: Escala de Media Integración (Medium-Scale Integration).

1.3.2 VLSI

Acrónimo inglés de Very Large Scale Integration, integración en escala muy grande. La integración en escala muy grande de sistemas de circuitos basados en transistores en circuitos integrados (CI) comenzó en los años 1980, como parte de las tecnologías de semiconductores y comunicación que se estaban desarrollando.

Los primeros chips semiconductores contenían sólo un transistor cada uno. A medida que la tecnología de fabricación fue avanzando, se agregaron más y más transistores, y en consecuencia más y más funciones fueron integradas en un mismo chip. El microprocesador es un dispositivo VLSI.

La primera generación de computadoras dependía de válvulas de vacío. Luego vinieron los semiconductores discretos, seguidos de circuitos integrados. Los primeros circuitos integrados (CI) contenían un pequeño número de dispositivos, como diodos, transistores, resistencias y capacitores (aunque no inductores), haciendo posible la fabricación de compuertas lógicas en un solo chip. La cuarta generación (LSI) consistía de sistemas con al menos mil compuertas lógicas.

El sucesor natural del LSI fue VLSI (varias decenas de miles de compuertas en un solo chip). Hoy en día, los microprocesadores tienen varios millones de compuertas en el mismo chip.

Hacia principios de 2006 se están comercializando microprocesadores con tecnología de hasta 65 nanómetros (nm), y se espera en un futuro cercano el advenimiento de los 45 nanómetros (nm).



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



1.3.3 MOS

MOS, acrónimo de las siglas en inglés de (Metal Oxide Semiconductor) hace referencia a una de las familias de FET, del tipo de óxido metálico semiconductor (MOSFET). Comúnmente son utilizados en electrónica y se encuentran en diferentes tipos.

1.3.4 CMOS

CMOS (del inglés Complementary Metal Oxide Semiconductor, "Semiconductor Complementario de Óxido Metálico") es una de las familias lógicas empleadas en la fabricación de circuitos integrados (chips).

Su principal característica consiste en la utilización de conjunta de transistores de tipo pMOS y tipo nMOS configurados de tal forma que, en estado de reposo, el consumo de energía es únicamente el debido a las corrientes parásitas.

En la actualidad, la inmensa mayoría de los circuitos integrados que se fabrican son de tecnología CMOS. Esto incluye microprocesadores, memorias, DSPs y muchos otros tipos de chips digitales.

1.3.5 Principales fabricantes

La industria de los componentes es fundamental para la industria electrónica que a su vez lo es para el resto de industrias, el importante volumen de negocio de este tipo de industria en los países más desarrollados les hace jugar un importante papel en sus respectivas economías.

En la siguiente tabla se muestra un listado con las principales empresas fabricantes de componentes electrónicos, la mayoría son multinacionales en las que la fabricación de componentes electrónicos representa tan sólo una parte de campo de actuación.

Símbolo	País	Tipos de componentes que fabrica	Web
AMD	Estados Unidos	Semiconductores	<u>AMD</u>
AD	Estados Unidos	Semiconductores	<u>Analog Devices</u>
CY	Estados Unidos	Semiconductores	<u>Cypress S.</u>
F	Estados Unidos	Semiconductores	<u>Fairchild</u>
FUJ	Japón	Semiconductores, condensadores,	<u>Fujitsu</u>



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



IBM	Estados Unidos	Memorias, microprocesadores, microcontroladores...	<u>IBM</u>
i	Estados Unidos	Memorias, microprocesadores y microcontroladores	<u>Intel</u>
	Japón	Semiconductores	<u>Mitsubishi</u>
M	Estados Unidos	Semiconductores	<u>Motorola</u>
NEC	Japón	Semiconductores, condensadores, relés...	<u>NEC</u>
OKI	Japón	Semiconductores	<u>OKI</u>
	Japón	Semiconductores, baterías, resistores...	<u>Panasonic</u>
	Holanda	Semiconductores	<u>NXP Semiconductors</u>
RMBS	Estados Unidos	Memorias	<u>Rambus</u>
	Sur Korea	Memorias, microcontroladores...	<u>Samsung</u>
ST	Suiza	Semiconductores	<u>ST</u>
	Japón	Memorias, microcontroladores, control de potencia...	<u>Sharp</u>
	Alemania	Semiconductores, reguladores...	<u>Siemens</u>
ti	Estados Unidos	Semiconductores	<u>TI</u>
	Estados Unidos	FPGA, CPLD	<u>Xilinx</u>
	Estados Unidos	Microcontroladores, microprocesadores, periféricos...	



1.4 Arquitectura Harvard y Von Neumann

Arquitectura Harvard

La arquitectura conocida como Harvard, consiste simplemente en un esquema en el que el CPU está conectado a dos memorias por intermedio de dos buses separados, una de las memorias contiene solamente las instrucciones del programa, y es llamada memoria de programa, la otra memoria solo almacena los datos y es llamada memoria de datos (figura 1.5). Ambos buses son totalmente independientes y pueden ser de distintos anchos. Para un procesador de Set de Instrucciones Reducido, o RISC (Reduced Instrucción Set Computer), el set de instrucciones y el bus de la memoria de programa pueden diseñarse de manera tal que todas las instrucciones tengan una sola posición de memoria de programa de longitud. Además, como los buses son independientes, el CPU puede estar accediendo a los datos para completar la ejecución de una instrucción, y al mismo tiempo estar leyendo la próxima instrucción a ejecutar. Se puede observar claramente que las principales ventajas de esta arquitectura son:

- a) Que el tamaño de las instrucciones no está relacionado con el de los datos, y por lo tanto puede ser optimizado para que cualquier instrucción ocupe una sola posición de memoria de programa, logrando así mayor velocidad y menor longitud de programa.
- b) Que el tiempo de acceso a las instrucciones puede superponerse con el de los datos, logrando una mayor velocidad de operación.

Una pequeña desventaja de los procesadores con arquitectura Harvard, es que deben poseer instrucciones especiales para acceder a tablas de valores constantes que pueda ser necesario incluir en los programas, ya que estas tablas se encontraran físicamente en la memoria de programa (por ejemplo en la EPROM de un microprocesador).

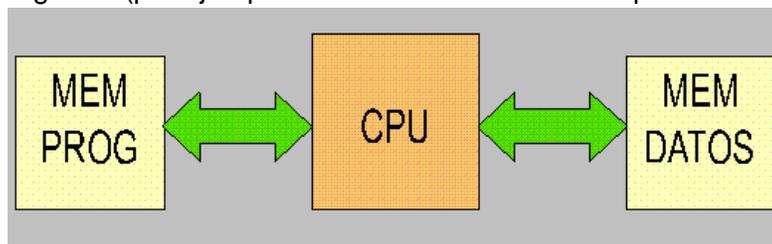


Figura 1.5 Arquitectura Harvard

Los micro controladores PIC 16C5X, 16CXX y 17CXX poseen arquitectura Harvard, con una memoria de datos de 1 byte, y una memoria de programa que, según el modelo, puede ser de 12 bits para los 16C5X, 14 bits para los 16CXX y 16 bits para los 17CXX.



Arquitectura Von Neumann

La arquitectura tradicional de computadoras y microprocesadores se basa en el esquema propuesto por John Von Neumann, en el cual la unidad central de proceso, o CPU, esta conectada a una memoria única que contiene las instrucciones del programa y los datos (figura 1.6). El tamaño de la unidad de datos o instrucciones esta fijado por el ancho del bus de la memoria. Es decir que un microprocesador de 8 bits, que tiene además un bus de 1 byte que lo conecta con la memoria, deberá manejar datos e instrucciones de una o más unidades de 8 bits (bytes) de longitud.

Por otro lado este bus único limita la velocidad de operación del microprocesador, ya que no se puede buscar de memoria una nueva instrucción, antes de que finalicen las transferencias de datos que pudieran resultar de la instrucción anterior. Es decir que las dos principales limitaciones de esta arquitectura tradicional son:

- a) Que la longitud de las instrucciones esta limitada por la unidad de longitud de los datos, por lo tanto el microprocesador debe hacer varios accesos a memoria para buscar instrucciones complejas.
- b) Que la velocidad de operación (o ancho de banda de operación) esta limitada por el efecto de cuello de botella que significa un bus único para datos e instrucciones que impide superponer ambos tiempos de acceso.

La arquitectura von Neumann permite el diseño de programas con código auto modificable, práctica bastante usada en las antiguas computadoras que solo tenían acumulador y pocos modos de direccionamiento, pero innecesaria, en las computadoras modernas.

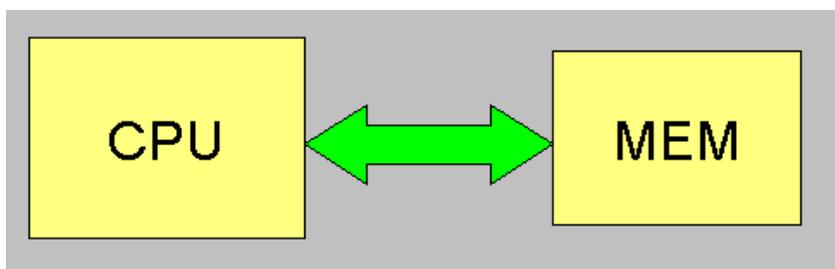


Figura 1.6 Arquitectura Von Newmann



1.5 Diagramas lógicos de Circuitos Integrados

Diagrama lógico del micro controlador (figura 1.7 y 1.8)

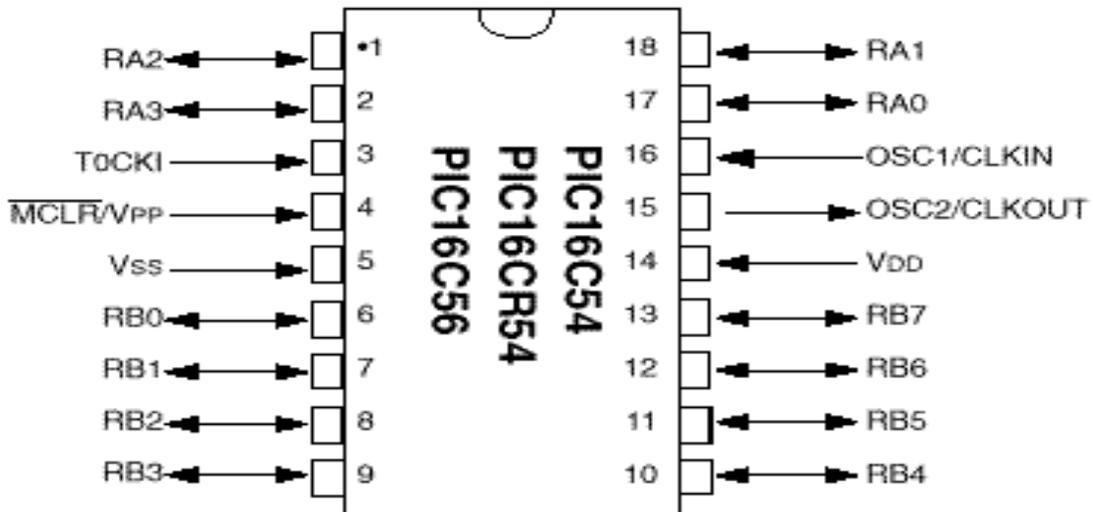


Figura 1.7 Representación de los pics 16C54, 16CR54 y 16C56

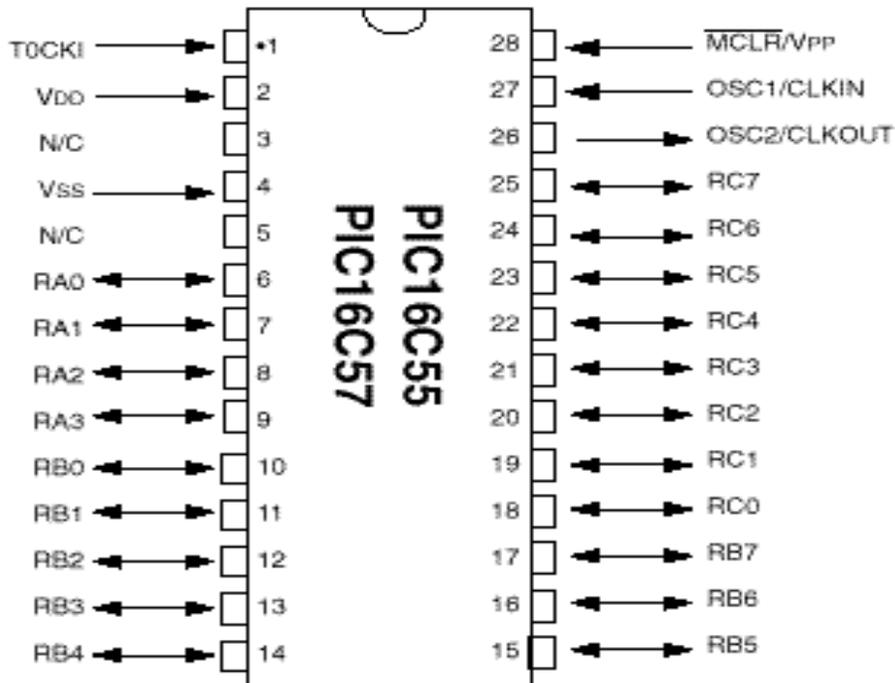


Figura 1.8 Representación de los pics 16C55 y 16C57



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



1.6 Programa fuente

El programa fuente está compuesto por una sucesión de líneas de programa. Cada línea de programa esta compuesta por 4 campos separados por uno o más espacios o tabulaciones. Estos campos son: [Etiqueta] Comando [Operando(s)] [; Comentario].

La etiqueta es opcional, el comando puede ser un mnemónico del conjunto de instrucciones, el operando esta asociado al comando, si no hay comando no hay operando, e inclusive algunos comandos no llevan operando. el comentario es opcional para el compilador aunque es buena práctica considerarlo obligatorio para el programador.

La etiqueta, es el campo que empieza en la primera posición de la línea. No se pueden insertar espacios o tabulaciones antes de la etiqueta sino será considerado comando. Identifica la línea de programa haciendo que el compilador le asigne un valor automáticamente. Si se trata de una línea cuyo comando es una instrucción de programa del microcontrolador, se le asigna el valor de la dirección de memoria correspondiente a dicha instrucción (location counter). En otros casos se le asigna un valor de una constante, o la dirección de una variable, o será el nombre de una macroinstrucción.

El comando puede ser un código mnemónico de instrucción del microcontrolador, o una directiva o pseudo instrucción para el compilador.

En el primer caso será directamente traducido a código de máquina, en el segundo caso será interpretado por el compilador y realizara alguna acción en tiempo de compilación como ser asignar un valor a una etiqueta.

El campo de parámetros puede contener uno o más parámetros separados por comas. Los parámetros dependen de la instrucción o directiva. Pueden ser números o literales que representen constantes o direcciones.

El campo de comentario debe comenzar con un carácter punto y coma. No necesita tener espacios o tabulaciones separándolo del campo anterior, e incluso puede empezar en la primera posición de la línea. El compilador ignora todo el texto que contenga la línea después de un carácter punto y coma. De esta manera pueden incluirse líneas que contengan solo comentarios, y es muy buena práctica hacer uso y abuso de esta posibilidad para que los programas resulten auto documentados.

1.7 Lenguaje Ensamblador

El lenguaje ensamblador es un tipo de lenguaje de bajo nivel utilizado para escribir programas informáticos, y constituye la representación más directa del código máquina específico para cada arquitectura de computadoras legible por un programador.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



Fue usado ampliamente en el pasado para el desarrollo de software, pero actualmente sólo se utiliza en contadas ocasiones, especialmente cuando se requiere la manipulación directa del hardware o se pretenden rendimientos inusuales de los equipos.

1.7.1 Características

- Programar en lenguaje ensamblador es difícil de aprender, entender, leer, escribir, depurar y mantener, por eso surgió la necesidad de los lenguajes compilados.
- A pesar de perder rendimiento en un proceso de compilación, en la actualidad la mayoría de las computadoras son suficientemente rápidas.
- El lenguaje ensamblador no es portable.
- Programar en lenguaje ensamblador lleva mucho tiempo.
- Los programas hechos en lenguaje ensamblador son generalmente más rápidos. Al programar cuidadosamente en lenguaje ensamblador se pueden crear programas de 5 a 10 veces más rápidos que con lenguajes de alto nivel.
- Los programas hechos en lenguaje ensamblador generalmente ocupan menos espacio. Un buen programa en lenguaje ensamblador puede ocupar casi la mitad de espacio que su contraparte en lenguaje de alto nivel.
- Con el lenguaje ensamblador se pueden crear segmentos de código imposibles de formar en un lenguaje de alto nivel.

1.7.2 Ensambladores

Un ensamblador crea código objeto traduciendo instrucciones mnemónicas a códigos operativos e interpretando los nombres simbólicos para direcciones de memoria y otras entidades. El uso de referencias simbólicas es una característica básica de los ensambladores, evitando tediosos cálculos y direccionamiento manual después de cada modificación del programa. La mayoría de los ensambladores también incluyen facilidades para crear macros, a fin de generar series de instrucciones cortas que se ejecutan en tiempo real, en lugar de utilizar subrutinas.

Los ensambladores son por lo general más fáciles de programar que los compiladores de lenguajes de alto nivel, y han estado disponibles desde la década de 1950. Los ensambladores modernos, especialmente para arquitecturas basadas en RISC, como por ejemplo MIPS, SPARC y PA-RISC optimizan las instrucciones para explotar al máximo la eficiencia de segmentación del CPU.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



Los ensambladores de alto nivel ofrecen posibilidades de abstracción que incluyen:

- Control avanzado de estructuras.
- Procedimientos de alto nivel, declaración de funciones.
- Tipos de datos que incluyen estructuras, registros, uniones, clases y conjuntos.
- Sofisticado procesamiento de macros.

1.7.3 Lenguaje

Un programa escrito en lenguaje ensamblador consiste en una serie de instrucciones que corresponden al flujo de órdenes ejecutables que pueden ser cargadas en la memoria de una computadora. Por ejemplo, un procesador x86 puede ejecutar la siguiente instrucción binaria como se expresa en código de máquina:

- Binario: 10110000 01100001 (Hexadecimal: 0xb061).

La representación equivalente en lenguaje ensamblador es más fácil de recordar:

- MOV al, 061h.

Esta instrucción significa:

- Mueva el valor hexadecimal 61 (97 decimal) al registro "al".

El mnemónico "mov" es un *código de operación* u "opcode", elegido por los diseñadores de la colección de instrucciones para abreviar "move" (mover). El opcode es seguido por una lista de argumentos o *parámetros*, completando una instrucción de ensamblador típica.

La transformación del lenguaje ensamblador en código máquina la realiza un programa ensamblador, y la traducción inversa la puede efectuar un desensamblador.

A diferencia de los lenguajes de alto nivel, aquí hay usualmente una correspondencia 1 a 1 entre las instrucciones simples del ensamblador y el lenguaje de máquina. Sin embargo, en algunos casos, un ensamblador puede proveer "pseudo instrucciones" que se expanden en un código de máquina más extenso a fin de proveer la funcionalidad necesaria.

Por ejemplo, para un código máquina condicional como "si X mayor o igual que", un ensamblador puede utilizar una pseudo instrucción al grupo "haga si menor que", y "si = 0" sobre el resultado de la condición anterior. Los ensambladores más completos también proveen un rico lenguaje de macros que se utiliza para generar código más complejo y secuencias de datos.

Cada arquitectura de computadoras tiene su propio lenguaje de máquina, y en consecuencia su propio lenguaje ensamblador.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



Los ordenadores difieren en el tipo y número de operaciones que soportan; también pueden tener diferente cantidad de registros, y distinta representación de los tipos de datos en memoria. Aunque la mayoría de las computadoras son capaces de cumplir esencialmente las mismas funciones, la forma en que lo hacen difiere y los respectivos lenguajes ensambladores reflejan tal diferencia.

Pueden existir múltiples conjuntos de mnemónicos o sintaxis de lenguaje ensamblador para un mismo conjunto de instrucciones, instanciados típicamente en diferentes programas ensamblador. En estos casos, la alternativa más popular es la provista por los fabricantes, y usada en los manuales del programa.

1.7.4 Código máquina (o lenguaje de máquina)

El lenguaje de máquina está formado por instrucciones sencillas, que -dependiendo de la estructura del procesador- pueden especificar:

- Registros específicos para operaciones aritméticas, direccionamiento o control de funciones.
- Posiciones de memoria específicas (*offset*).
- Modos de direccionamiento usados para interpretar un operando.

Las operaciones más complejas se realizan combinando estas instrucciones sencillas, que pueden ser ejecutadas secuencialmente o mediante instrucciones de control de flujo.

Las operaciones disponibles en la mayoría de los conjuntos de instrucciones incluyen:

- mover
 - ✓ llenar un registro con un valor constante.
 - ✓ mover datos de una posición de memoria a un registro o viceversa.
 - ✓ escribir y leer datos de dispositivos.
- Operaciones
 - ✓ sumar, restar, multiplicar o dividir los valores de dos registros, colocando el resultado en uno de ellos o en otro registro.
 - ✓ realizar operaciones binarias, incluyendo operaciones lógicas (AND/OR/XOR/NOT).
 - ✓ comparar valores entre registros (mayor, menor, igual).
- afectar el flujo del programa
 - ✓ saltar a otra posición en el programa y ejecutar instrucciones.
 - ✓ saltar si se cumplen ciertas condiciones (IF).
 - ✓ saltar a otra posición, pero guardar el punto de salida para retornar (CALL, llamada a subrutinas).



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



Algunas computadoras incluyen instrucciones complejas dentro de sus capacidades. Una sola instrucción compleja hace lo mismo que en otras computadoras puede requerir una larga serie de instrucciones, por ejemplo:

- salvar varios registros en la pila de una sola vez.
- mover grandes bloques de memoria.
- operaciones aritméticas complejas o de punto flotante (seno, coseno, raíz cuadrada).

El nivel de lenguaje ensamblador tiene aspectos importantes de los niveles de micro arquitectura, en los cuales se encuentra (ISA y sistema operativo) estos dos se utilizan para la traducción en lugar de la interpretación. Algunas características del lenguaje se describen a continuación Los programas que sirven para traducir algún programa para el usuario se llama traductores, el lenguaje en que esta escrito el programa original se llama lenguaje fuente, el lenguaje original que sea modificado se llama lenguaje objeto.

Se usa la traducción cuando se cuenta con un procesador (ya sea hardware o un intérprete) para el lenguaje objeto pero no para el lenguaje fuente,

Si la traducción se realiza correctamente, la ejecución del programa traducido dará exactamente los mismos resultados que habría dado la ejecución del programa fuente.

Hay dos diferencias entre traducción e interpretación, en la traducción no se ejecuta directamente el programa original, en el lenguaje fuente se convierte en un programa equivalente llamado programa objeto o programa binario ejecutable y este funciona solo cuando se ha acabado la traducción.

El código máquina, un simple patrón de bits, es hecho legible reemplazando *valores crudos* por símbolos denominados mnemónicos. Se inventó para facilitar la tarea de los primeros programadores que hasta ese momento tenían que escribir directamente en código binario. Inicialmente el código de ceros y unos (el programa) debía introducirse en una tarjeta perforada.

La posición ocupada por cada punto equivalía a un "1" o a un "0" según hubiera o no una perforación. Lo cual suponía una forma casi idéntica en la que hoy se escriben los datos binarios en soportes tales como los CD y DVD.

Mientras que una computadora reconoce la instrucción de máquina IA-32 10110000 01100001, para los programadores de microprocesadores x86 es mucho más fácil reconocer dicha instrucción empleando lenguaje ensamblador:

```
movb 0x61,%al
```

(Que significa mover el valor hexadecimal 61 (97 decimal) al registro 'al'.)



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



Cada instrucción de la máquina se transforma en una única instrucción en código simbólico, pero además, para mejorar la legibilidad del programa, el código simbólico introduce instrucciones adicionales, que no corresponden a ninguna instrucción de la máquina y que proporcionan información, se llaman "pseudo instrucciones".

El código simbólico puede parecer de difícil acceso, pero es más fácil de recordar e interpretar que el binario o el hexadecimal.

Los lenguajes simbólicos no resuelven definitivamente el problema de cómo programar un ordenador de la manera más sencilla posible. Para utilizarlos, hay que conocer a fondo el microprocesador, los registros de trabajo de que dispone, la estructura de la memoria, y muchas cosas más.

Además, el lenguaje ensamblador está muy ligado al microprocesador para que sea posible escribir programas independientes de la máquina en que van a ejecutarse. Este código simbólico no puede ser ejecutado directamente por un ordenador, por lo que es preciso traducirlo previamente. Pero la traducción es un proceso mecánico y repetitivo, que se presta a su realización por un programa de ordenador.

Los programas que traducen código simbólico al lenguaje de máquina se llaman ensambladores ("assembler", en inglés), porque son capaces de ensamblar el programa traducido a partir de varias piezas, procedimientos o subrutinas a código binario ("1" y "0") que entiende el procesador.

1.7.5 Ventajas del Ensamblador

- La primera razón para trabajar con ensamblador es que proporciona la oportunidad de conocer más a fondo la operación de su PC, lo que permite el desarrollo de software de una manera más consistente.
- La segunda razón es el control total de la PC que se tiene con el uso del mismo.
- Otra razón es que los programas de ensamblador son más rápidos, más compactos y tienen mayor capacidad que los creados en otros lenguajes.
- Por último el ensamblador permite una optimización ideal en los programas tanto en su tamaño como en su ejecución.

1.8 Una Mirada Rápida al MPLAB

EL MPLAB es un "Entorno de Desarrollo Integrado " (Integrated Development Environment, IDE) que corre en Windows, mediante el cual puede desarrollar aplicaciones para los diferentes micro controladores que existen dentro de las familias PIC.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



EL MPLAB le permite a Usted escribir, depurar y optimizar los programas (firmware) de sus diseños con PIC. EL MPLAB incluye un editor de texto, un simulador y un organizador de proyectos. Además, el MPLAB soporta el emulador PICMASTER y a otras herramientas de desarrollo de Microchip como el PICSTART - Plus.

De que forma le ayuda el MPLAB?

Con el MPLAB Usted puede:

- Depurar sus programas fuente.
- Detectar errores automáticamente en sus programas fuente para editarlos.
- Depurar los programas utilizando puntos de corte (break points) mediante valores de los registros internos.
- Observar el flujo del programa con el simulador MPLAB -SIM, ó seguirlo en tiempo real utilizando el emulador PICMASTER.
- Realizar medidas de tiempo utilizando un cronómetro.
- Mirar variables en las ventanas de observación.
- Encontrar respuestas rápidas a sus preguntas, utilizando la Ayuda en línea del MPLAB.

1.8.1 Las Herramientas del MPLAB

El organizador de proyectos (Project Manager) es parte fundamental de MPLAB. Sin crear un proyecto Usted no puede realizar depuración simbólica. Con el Organizador de Proyectos (Project manager) puede utilizar las siguientes operaciones:

- Crear un proyecto.
- Agregar un archivo de programa fuente de proyecto.
- Ensamblar o compilar programas fuente.
- Editar programas fuente.
- Reconstruir todos los archivos fuente, o compilar un solo archivo.
- Depurar su programa fuente.

1.8.2 Software ensamblador

El software ensamblador que presenta Microchip viene en dos presentaciones, una, para entorno DOS llamado MPASM.EXE y la otra, para entorno Windows llamado MPASMWIN.EXE



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



Las dos presentaciones soportan a TODOS los microcontroladores de la familia PIC de Microchip, el conjunto de instrucciones de los microcontroladores PIC es en esencia la base del lenguaje ensamblador soportado por este software para su correcta utilización.

Directivas de uso frecuente:

Son instrucciones para el compilador.

#DEFINE

ejemplo. #define <nombre> [<valor a remplazar>]

explicación: declara una cadena de texto como sustituto de otra

END

ejemplo. end

explicación: indica fin de programa

EQU

ejemplo. Status equ 05

explicación: define una constante de ensamble

INCLUDE

ejemplo. include <PIC16F84.h>

explicación: incluye en el programa un archivo con código fuente

ORG

ejemplo. org 0x100

explicación: ensambla a partir de la dirección especificada

Para información más completa referirse a la guía rápida del MPASM.

Una vez instalado adecuadamente el MPLAB, para realizar la simulación de un programa deben seguirse los siguientes pasos:

Edite en un archivo de texto el siguiente programa: ejemplo:

status equ 0x03; hace equivalencia entre el símbolo status indicándolo como 3 en hexadecimal

Cont equ 0x20

F equ 1

org 0 ; indica posición de memoria desde donde se ensambla

Inicio

movlw 0x0F ; carga de w con el valor constante 15 (literal)

movwf Cont ; el contenido de w se pasa al reg. CONT

Loop

decfsz Cont,F ;decremento de Cont y elude siguiente si=0

goto Loop ;salto incondicional a Loop

goto \$;Salto incondicional aqui mismo

end ;Fin del código



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



Lista de pasos:

1. Haga doble clic en el ícono correspondiente a MPLAB.
2. Crear el archivo fuente correspondiente (menú File...New Source).
3. Salve el archivo (con extensión .ASM) una vez terminada su edición (menú FILE...Save).
4. Debe a continuación crearse un nuevo proyecto (menú Project...New Project).
5. Cuando aparezca la ventana de New Project editar las cajas de texto: Project path and Name y Development Mode, hacer click en <OK>.
6. En la siguiente ventana Edit Project, hacer click en la sección Non-project files sobre el nombre del archivo fuente realizado en los pasos 2 y 3.
7. Haga click en el botón <=add y luego de que éste aparezca en la sección Project Files haga click sobre el botón <OK>.
8. Salvar el proyecto (en el menú Project...Save project).
9. Realizar la "construcción de todo el proyecto" (menú Project...Build All).
10. En esta etapa se realiza en forma automática el ensamble de nuestro programa fuente y el vaciado de éste en memoria de simulación. El proceso de ensamble generará un archivo de errores en caso de que estos existan, sí es así deben corregirse directamente sobre el archivo fuente, salvar las correcciones y reconstruir el proyecto (menú Project...Build All). <<<En esta etapa del proceso ya se tiene el entorno listo para la simulación>>>

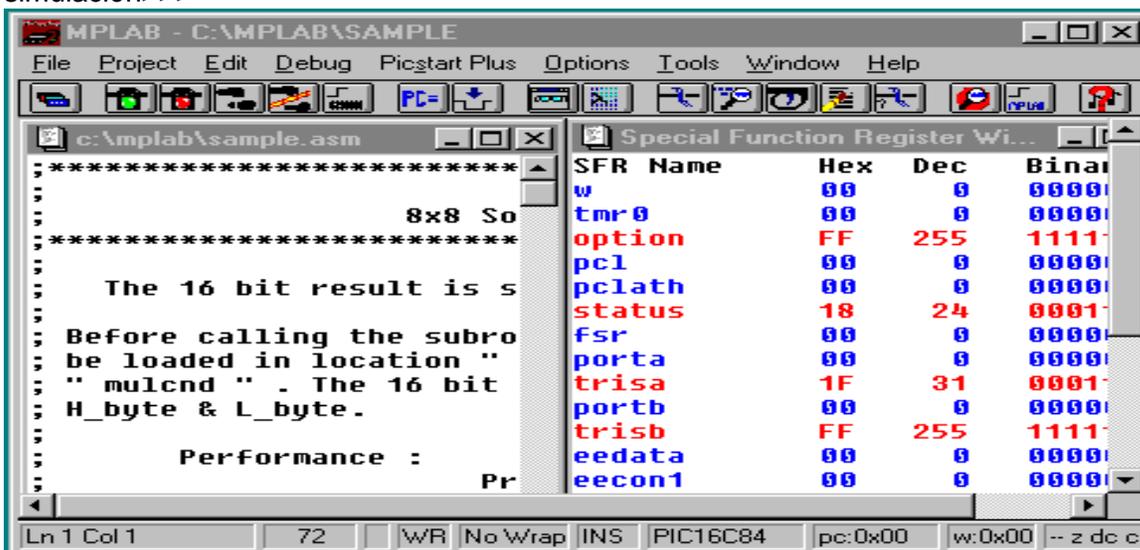


Figura 1.9 Pantalla del MPLAB

Vista típica del entorno MPLAB

Como en la mayoría de las aplicaciones Windows la pantalla se divide en varias secciones:

1. Barra de título: Se observa el nombre del proyecto
2. Barra de menús: Acceso a las diferentes opciones del entorno
3. Barra de herramientas: Cada ícono ejecuta las acciones correspondientes
4. Barra de estados: Indica el estado del entorno y sus ventanas

Ingeniería en Computación



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



1.8.3 Simulación:

1. Resetear el procesador (menú Debug...Run...Reset) ó con F6 ó con el ícono correspondiente en la barra de herramientas.
2. Crear una nueva ventana donde se incluyan las variables que queremos tener en cuenta (Window...New Watch Window)
3. Empezar a correr paso a paso el programa haciendo el seguimiento detallado de todos y cada uno de los pasos (menú Debug...Run...Step) ó con la tecla F7 ó con el ícono correspondiente en la barra de herramientas.

El proceso de simulación nos permite detectar y corregir problemas de lógica, problemas de situaciones que no hayamos tenido en cuenta que son errores que no pueden ser detectados en el momento del ensamble del programa.

Nota: El programa MPLAB puede obtenerse en forma gratuita de la dirección:
<http://www.microchip.com/10/Tools/mTools/MPLAB/index.html>

1.8.4 Conversión a ASCII

El conjunto de caracteres ASCII (American Standard Code for Information Interchange) es el código de representación en hexadecimal del alfabeto, los números del 0 al 9 y los principales símbolos de puntuación y algunos caracteres de control (Figura 1.10).

		Most Significant Character								
		Hex	0	1	2	3	4	5	6	7
Least Significant Character	0	NUL	DLE	Space	0	@	P	`	p	
	1	SOH	DC1	!	1	A	Q	a	q	
	2	STX	DC2	"	2	B	R	b	r	
	3	ETX	DC3	#	3	C	S	c	s	
	4	EOT	DC4	\$	4	D	T	d	t	
	5	ENQ	NAK	%	5	E	U	e	u	
	6	ACK	SYN	&	6	F	V	f	v	
	7	Bell	ETB	'	7	G	W	g	w	
	8	BS	CAN	(8	H	X	h	x	
	9	HT	EM)	9	I	Y	i	y	
	A	LF	SUB	*	:	J	Z	j	z	
	B	VT	ESC	+	;	K	[k	{	
	C	FF	FS	,	<	L	\	l		
	D	CR	GS	-	=	M]	m	}	
	E	SO	RS	.	>	N	^	n	~	
	F	SI	US	/	?	O	_	o	DEL	

Conjunto de caracteres ASCII Como se observa en la figura 1.10, podemos dividir a cada carácter representado en hexadecimal como una parte alta de 3 bits (Most significant character) y una parte baja de 4 bits (Least significant character), o sea, la representación total se hace con solo 7 bits.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



De los problemas usuales en la programación está el convertir un número hexadecimal representado en 8 bits a dos caracteres ASCII los cuales sean la representación de dicho número para permitir la visualización en terminales de datos tales como Monitores de video o Impresoras entre otras.

Ejemplo:

Se quiere representar el número hexadecimal 70h cuya representación en binario es 01110000b como los dos caracteres ASCII "7" y "0", gráficamente:

7	0	en hexadecimal (8 bits)
-	-	
"7"	"0"	en ASCII (16 bits)
-	-	
37h	30h	ASCII en hexadecimal (16 bits)

Transportándolo a un programa:

```

NumHex equ 0Ch ; posición donde se almacena el número a convertir
AsciiH equ 0Dh ; posición donde se almacena el resultado parte alta
AsciiL equ 0Eh ; posición donde se almacena el resultado parte baja
.....
movlw 0Fh ; dato para enmascarar parte alta
andwf NumHex,0 ; se enmascara la parte alta del número hexa y va a W
iorlw 30h ; convierte el número en ascii
movwf AsciiL ; el número queda salvado en la variable de salida
movlw 0F0h ; dato para enmascarar parte baja
andwf NumHex,1 ; se enmascara la parte baja del número hexa
swapf NumHex,0 ; se invierten parte alta y baja
iorlw 30h ; convierte el número en ascii
movwf AsciiL ; el número queda salvado en la variable de salida.
.....

```

Se debe tener en cuenta que el ejemplo anterior funciona en forma correcta siempre y cuando los nibbles del número hexadecimal a convertir, estén en el rango de 0 a 9, debe realizarse un tratamiento adicional a estos si se encuentran en el rango de Ah a Fh.

1.8.5 Aritmética

Dentro de los microcontroladores PIC se cuenta con instrucciones aritméticas tales como ADDWF y ADDLW, SUBWF y SUBWF, para efectuar operaciones de suma y resta respectivamente e instrucciones tales como RLF y RRF para realizar operaciones de rotación a través del carry con las cuales se pueden realizar divisiones entre 2 y multiplicaciones por 2 respectivamente, hasta este punto podríamos ver el conjunto de instrucciones un poco limitado, sin embargo, utilizando técnicas avanzadas de programación podemos obtener operaciones más complejas.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



1.9 Creación de un Grabador digital

Para la construcción del Grabador Digital son precisos los componentes que a continuación se mencionan.

C1	100 nF
C2	100 nF
C3	1 μ F 25V
C4	1 μ F 25V
C5	4.7 μ F 50V
C6	100 nF
C7	220 μ F 25V
C8	100 nF
C9	2 μ F 25V
C10	100 μ F 25V (2)
C12	10 μ F 25V (2)
D1	DIODO LED
M1	MICROFONO
Q1	BC557
R1	10 K 1/4W
R2	47 K 1/4W
R3	1 K 1/4W
R4	1 K 1/4W
R5	220 K 1/4W
R6	47 K 1/4W
R7	470 K 1/4W
R8	10 K 1/4W
R9	2.2 K 1/4W
R10	100 K 1/4W
R11	1 K 1/4W
S1	PULSADOR
S2	PULSADOR
SP1	ALTAVOZ
U1	LM7805
U2	ISD1016
1	ZOCALO 29 PIN
12	TERMINALES
1	CAJA RM.04
4	SEPARADORES
4	PATILLAS ADHESIVAS
1	METRO CABLE R/N
1	CIRCUITO IMPRESO



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



Para la construcción del Grabador Digital utilizaremos un circuito impreso (figura 1.11). Para su construcción seguiremos cualquiera de los métodos habituales.

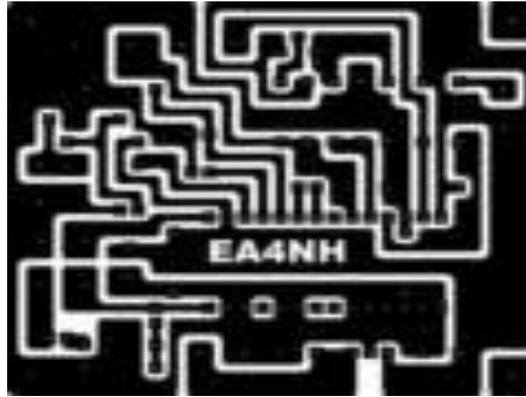


Figura 1.11 Elaboración de un circuito impreso

Una vez en posesión de la placa de circuito impreso y del resto de los componentes, procederemos al montaje, utilizaremos un soldador de punta fina y estaño de calidad con alma de resina, comenzaremos soldando las resistencias, siguiendo por los condensadores, antes de soldar el zócalo en su posición no hay que olvidar de realizar, con un trozo de hilo desnudo, el puente marcado "BR", ya que una vez colocado el zócalo es imposible soldar dicho puente.

Completaremos el montaje con los condensadores electrolíticos, transistor, regulador y terminales. En la figura 1.12 tenemos la placa de circuito impreso con todos los componentes colocados.



Figura 1.12 Placa del circuito impreso con todos los componentes soldados

Una vez soldados todos los componentes en la placa de circuito impreso y antes de su instalación y cableado en la caja, procederemos a la comprobación del funcionamiento del circuito. Conectaremos un altavoz en los terminales marcados SPKR. En los terminales correspondientes soldaremos los pulsadores S1 y S2.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



Con un par de trozos de hilo rígido soldaremos el micrófono en su posición. Igualmente soldaremos el diodo LED en los terminales correspondientes. El cable de alimentación rojo-negro irá soldado en los terminales marcados + y -.

Alimentaremos el circuito con una tensión de doce voltios. Accionaremos el pulsador S1 y manteniéndolo pulsado, accionaremos el pulsador S2. En este momento comenzaremos la grabación del mensaje de prueba. Hablaremos a una distancia entre 10 y 20 centímetros del micrófono. Una vez terminado el mensaje, soltaremos el pulsador S2 y a continuación el S1.

Pulsando de nuevo brevemente el pulsador S2 escucharemos en el altavoz el mensaje grabado. Al final del mensaje el diodo LED se encenderá brevemente. Si esto no ocurre, cambiar la posición de los terminales del diodo. Después de algunas pruebas conseguiremos determinar la distancia más adecuada para la grabación del mensaje.

Una vez comprobado el perfecto funcionamiento del circuito, procederemos a mecanizar la caja metálica. Se utiliza una caja modelo RM.04 de Retex, aunque por supuesto se puede utilizar cualquier otra de dimensiones similares.

En la parte frontal realizaremos los taladros correspondientes a los dos pulsadores, S1 y S2, el micrófono y el diodo LED. En la parte trasera daremos dos taladros, para la entrada de alimentación y conector de salida de audio.

Si se quiere utilizar un altavoz exterior, habrá que dar otro taladro para el conector de dicho altavoz. Es muy importante tener en cuenta que los dos terminales del altavoz deben estar aislados de masa, por lo que sí se utiliza un jack hembra, será necesario aislarlo completamente de masa utilizando arandelas aislantes. De no proceder así, se corre el riesgo de deteriorar el circuito integrado ISD1016.

En el fondo de la caja daremos cuatro taladros para la sujeción del circuito mediante cuatro separadores de 5 mm de longitud.

Realizados todos los taladros, sujetaremos el circuito a la caja con los mencionados separadores y mediante cortos trozos de hilo uniremos los pulsadores S1 y S2, el micrófono y el diodo LED a sus correspondientes terminales.

En la parte trasera colocaremos el conector de salida de audio y el conector de salida de altavoz, convenientemente aislado, que también uniremos con unos trozos de cable blindado a los correspondientes terminales.

En el taladro destinado a la entrada de alimentación colocaremos una goma pasa chasis y através pasaremos el cable rojo-negro de alimentación, que irá soldado a los terminales marcados + y - (más y menos).



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



En la figura 1.13 podemos ver el circuito impreso sujeto a la caja y todas las conexiones realizadas.



Figura 1.13 Circuito Impreso sujeto a caja con todas las conexiones

Una vez realizado el cableado y comprobado que las conexiones han sido realizadas correctamente, el Grabador Digital queda listo para su utilización.

1.9.1 Operación e Interconexión del grabador Digital.

La interconexión entre el Grabador Digital y el transmisor se realiza aplicando la señal de salida en paralelo con el micrófono del equipo. Prepararemos un trozo de cable blindado y en uno de sus extremos soldaremos el conector macho (normalmente un jack miniatura), adecuado al conector hembra que hayamos destinado en el Grabador Digital a la salida de audio.

El extremo libre de este cable blindado irá soldado directamente sobre el micrófono del transmisor. Como se puede observar en el esquema teórico, se ha dispuesto la resistencia R10 de 100K en serie con la salida de audio. Esta resistencia tiene dos funciones. Por un lado, debido a su alto valor impedimos que el Grabador Digital presente una impedancia demasiado baja, que al estar en paralelo con el micrófono del equipo disminuiría su señal de salida.

Por otro lado, como el Grabador Digital entrega una señal relativamente alta, es preciso reducir el valor de esta señal a unos pocos milivoltios para igualar su nivel al entregado por el micrófono del transmisor.

Una vez realizadas todas las conexiones, daremos tensión al Grabador Digital. Procederemos a grabar el mensaje pulsando S1 y a continuación S2. Grabaremos el mensaje y soltaremos S2 y S1 en este orden.

Con un altavoz conectado en el conector trasero, escucharemos el mensaje pulsando S2. Si todo es correcto procederemos a transmitir el mensaje. Debemos monitorizar nuestra transmisión para comprobar que el nivel de audio es el correcto.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



Esto lo podremos hacer escuchando la señal de salida en un receptor, escáner, etc. Si esto no es posible deberemos contar con la colaboración de un amigo que nos indique si el nivel de audio es correcto.

Si dicho nivel no es el adecuado, deberemos cambiar el valor de la resistencia R10 para que el nivel de audio cuando se habla por el micrófono o cuando se transmite el mensaje grabado, sea similar.

En las anteriores líneas se ha propuesto la construcción y puesta en servicio de un Grabador Digital, adecuado para transmitir pequeños mensajes previamente grabados y sobre todo muy útil para aquellos alumnos que se dedican asiduamente a la realización de proyectos.



Figura 1.14 Grabador concluido



Capítulo 2. Implementación y Aplicación a los Grabadores

2.1 Introducción al PROGRAMADOR 844A BK PRECISION



Figura 2.1 Grabador 844^a BK Precisión

Programador Universal con Base de 40 pines con capacidad de ISP Cómo utilizar este manual

- Aquí se explica cómo instalar el programa de control y cómo utilizar el programador.
- Se asume que el usuario tiene una cierta experiencia con las PC y la instalación del software.
- Una vez que usted ha instalado el programa de control le recomendamos consulte la AYUDA (HELP) que se encuentra dentro del menú del programa de control.
- Usted encontrará información específica con respecto a la instalación del programa de control y al uso del programador.

2.1.1 Descripción en detalle del Programador y Software

Usted encontrara algunas características menos relevantes del programador descritas aquí, pero todas las características del programador se describen en esta sección junto con los detalles con respecto a la instalación del programa de control. Lea esta sección para explorar todas las características que le proporciona su programador.

El 844A es el siguiente miembro de la nueva generación de programadores universales de BK PRECISION basados en Windows 95/98/Me/NT/2000/XP.

El programador está diseñado para satisfacer las demandas de laboratorios del desarrollo e ingenieros de campo, por ser un programador portátil.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



El 844A es un programador pequeño, rápido y poderoso en las diferentes clases de dispositivos programables. Al Usar BUILD-IN en Programación Serial en Circuitos (ISP) el conector del programador está capacitado para programar ISP.

El 844A no es solamente un programador, también es un probador de circuitos integrados lógicos TTL/CMOS y memorias.

El Programador 844A proporciona un precio muy competitivo y un diseño de hardware excelente para una programación confiable, muy rápida debido a la alta velocidad proporcionada por su hardware FPGA y la ayuda de la alta velocidad IEEE1284 (ECP/EPP) puerto paralelo. Seguramente más rápidamente que competidores en esta categoría.

El 844A interconecta con ordenadores personales de escritorio con cualquier puerto estándar (de la impresora) (ninguna tarjeta de interfaz especial necesitada). Por lo tanto usted puede tomar el programador y moverlo a otra PC fuera (assembly/disassembly) de la PC.

El 844A hace que 40 pin drivers de gran alcance de la TTL proporcionen H/L/pull_up/pull_down y capacidad leída para cada perno de zócalo, los conductores del perno funcionan bajo 1.8V.

El programador realiza la prueba de la inserción del dispositivo (incorrecta la posición del dispositivo en zócalo), antes de cargar el programa. Estas capacidades, apoyan a la prevención de algún daño de la viruta, debido al error del operador.

El hardware del programador ofrece bastantes recursos para la auto prueba, ya que el programa de control en cualquier momento pueda comprobar pin drivers, el actual y correcto nivel de todos los voltajes, la sincronización y comunicación entre el programador y la PC.

El programador 844A realiza la verificación de programación en nivel marginal del voltaje de fuente, que, mejora obviamente la producción de programación, y retención larga de los datos de las garantías.

El 844A es conducido por un programa de control fácil de utilizar con menús y ayuda en línea. El seleccionar del dispositivo es realizado por su clase, por el fabricante o simplemente mecanografiando a fragmento del nombre del vendedor y/o del número de pieza.

La función Rich-Featured Autoincrement permite asignar números de serie individuales a cada dispositivo programado o simplemente incrementa un número de serie, o la función permite leer los números de serie o cualquier identificación de dispositivo programado asignado por el archivo.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



Varios convertidores del socket están disponibles para manejar el dispositivo dentro de PLCC, SOIC y otros paquetes.

El diseño avanzado, incluyendo circuitos de protección, componentes originales de fábrica y la fabricación cuidadosa permite que se proporcione una garantía anual en piezas y funcionamiento del 844A (garantía limitada a 25,000 ciclos en el socket ZIF).

La instalación del software del programador es muy fácil. Inserte en CD en su unidad de de CD-ROM e instale el programa que se iniciará automáticamente. Instale el programa (setup.exe), el cual le guiará a través del proceso de instalación y hará todos los pasos necesarios antes de que usted pueda correr por primera vez el programa de control.

Si usted utiliza el sistema operativo MS-DOS, corra el programa de Instalación adecuado del CD.

Los Programas PG4U.EXE (para MS-DOS) y PG4W.EXE (para Windows) son programas de control comunes para estos Programadores de BK Precisión. Garantizamos el funcionamiento de éstos programas bajo todos los sistemas operativos antes mencionados sin ningún problema. También el funcionamiento bajo Windows está libre de errores.

Para las versiones de MS-DOS del programa de control que funciona bajo WINDOWS 3,11/95/98, sugerimos lo siguiente: (ajuste las "Propiedades" en el icono en su pantalla): Corra el programa de control en el MS-DOS en pantalla completa (en modo texto). En modo gráfico el programa de control puede estar un poco más lento.

Si usted quiere que el programa de control corra de fondo, es necesario fijar: (WIN95/98) el Botón del ratón DERECHO en el icono de MSDOS/Propiedades/otros/fondo (Background)/ Habilita; (WIN 3xx) en el archivo PIF apropiado, fije la opción para fondo (Background).

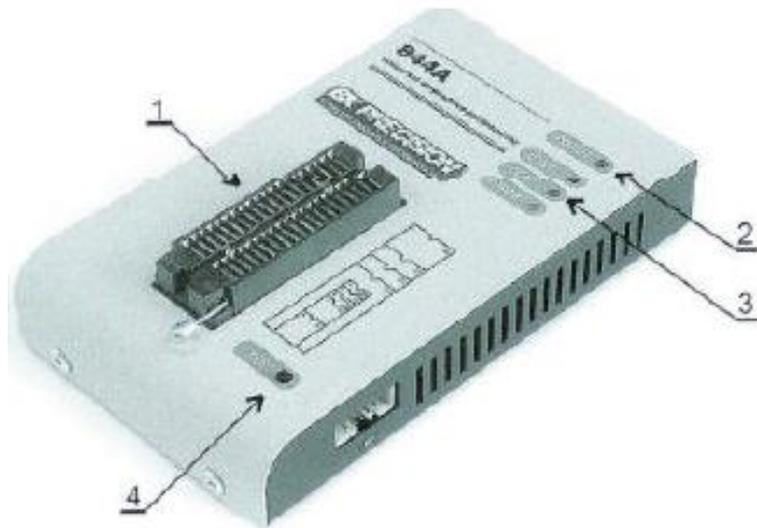


2.1.2 Elementos del 844A

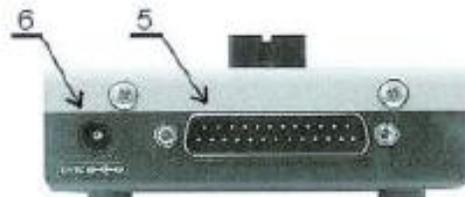
Los siguientes elementos componen al grabador digital 844^a (figura 2.2).

1. Socket ZIF de 40 pines
2. LED Encendido/En espera
3. LED, que indican resultado del trabajo
4. Botón de ¡SI!

Nota: debido al consumo bajo de energía del 844A en estado inactivo, no requiere el interruptor.



5. Conector para el cable de comunicación de la PC
6. Conector de la Fuente de Alimentación



7. Conector 7 para la ISP



Figura 2.2 Elementos del 844A BK Precisión



2.1.3 Descripción del conector de ISP del 844A

Descripción de los pines de la conexión ISP (figura 2.3).

- 1.- VCCP para la tarjeta, con sentido
- 2.- 10 H/L/Lectura, GND, VCCP, VPP
- 3.- H/L/Lectura, GND, VPP
- 4.- H/L/Lectura, GND, VCCP, VPP
- 5.- NC
- 6.- H/L/Lectura, GND, VCCP
- 7.- 9 GND
- 8.- H/L/Lectura, GND



Figura 2.3 Descripción de pines de 844A

La especificación de los pines del conector de ISP depende de la programación del dispositivo y es desplegado en la ventana adicional de Información (figura 2.4). Esta especificación corresponde con las notas de aplicación publicadas por los fabricantes del dispositivo.

Usted puede encontrar en www.bkprecision.com, una sección de notas de Aplicación.



Figura 2.4 Cable ISP del 844A

2.1.3.1 Advertencias:

Cuando use el 844A como programador ISP, no inserte en dispositivo en el socket ZIF
Cuando programe dispositivos en el socket ZIF, no inserte el cable para ISP al conector de ISP.

Use solamente el cable para ISP proporcionado. Cuando usted utiliza otro cable de diferente material y/o longitud para ISP pueden ocurrir errores en la programación.

El 844A puede proveer a la tarjeta (sistema tarjeta con limitación), pero el sistema de tarjeta no puede proveer al 844A.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



El 844A aplica voltaje de programación a la tarjeta y comprueba su valor (el sistema tarjeta puede modificar el voltaje de programación).

Si el voltaje de programación es diferente a como se espera, no se ejecutará ninguna acción con la tarjeta.

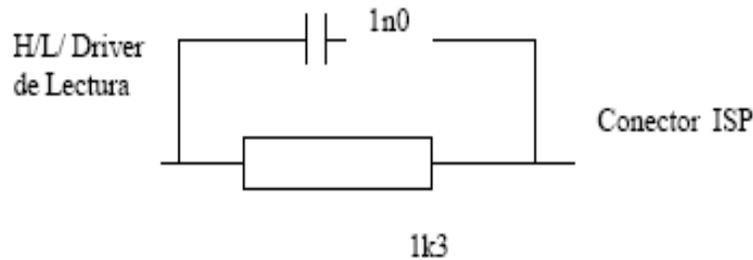


Figura 2.5 Controlador del 844A de H/L/Lectura

2.2 Requisitos mínimos de la PC, para la Implementación

PC Pentium a 100MHz o más veloz

512MB libres de RAM

Unidad de CD-ROM

10MB de espacio libre en HDD

Sistema Operativo MS WINDOWS 95/98/Me/NT/2000/XP

Puerto paralelo libre (de la impresora) bus PCI, compatible con IEEE 1284 (ECP/EPP)

Nota: Para su conveniencia, sugerimos que utilice una tarjeta adicional de puerto paralelo para una impresora adicional (LPT2 por ejemplo), con el propósito de evitar compartir el mismo LPT entre la impresora y el programador.

2.2.1 Instalación del hardware del programador

- Apague la PC y el programador.
- Conecte el puerto de comunicación del programador al Puerto Paralelo de la PC utilizando el cable proporcionado.
- Encienda la PC.
- Conecte el adaptador de alimentación al programador.

2.2.2 Instalación del software del programador

- El paquete del programador contiene un CD con el control del programa, utilerías e información adicional.
- El permiso de copiar libremente el contenido del CD se concede a razón de demostrar que se ocupará para trabajo con programadores de BK Precisión.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



- Ejecute el programa de la instalación del CD (DOS: Install.exe, WINDOWS: Setup.exe) y siga las instrucciones en pantalla.
- Teclee PG4U.EXE (DOS) o PG4UW.EXE (WINDOWS) para entrar en el programa de control. La opción FILE (Archivo) del menú contiene los comandos para la manipulación de archivos y directorios.
- La opción Buffer del menú debe ser utilizado para la manipulación del buffer de almacenamiento.

El programa de control proporcionado por BK PRECISION, incluido en el CD del paquete, está garantizado de estar libre de cualquier virus en el momento de la entrega. Para incrementar su seguridad nuestros programas incluyen un algoritmo especial para la detección de posibles infecciones de virus. Corra el programa de control, de la línea de comando de MS/PC-DOS escriba: PG4U.

Dentro del ambiente Windows: dé doble clic al icono de PG4UW.

Después del comienzo, el programa de control PG4U/PG4UW, explora automáticamente todos los puertos existentes en busca de cualquier Programador BK Precisión conectado. El programa PG4U/PG4UW es común para estos programadores de BK Precisión, por lo tanto el intentará encontrar cualquiera de estos programadores 865, 864, 844A, 848, 849, 862, 863 y 861.

Notas: Cuando comienza el programa de PG4U/PG4UW, el programa comprueba su integridad, la exhibición del menú del programa espera instrucciones del usuario, si el programa de control no puede comunicarse con el programador, un mensaje de error aparecerá en la pantalla, incluyendo código de error y la descripción de razones posibles (programador desconectado, mala conexión, falta de la fuente de alimentación, puerto de impresora incompatible).

Elimine la fuente del error y presione cualquier tecla, si todavía existe la condición de error, el programa reasume su operación en modo de demo y el acceso al programador no es posible, si usted no puede encontrar la causa del error, siga las instrucciones en la sección de localización de averías, en la adición, el programa de control comprueba la comunicación con programador antes de cualquier operación con el dispositivo programado.

2.2.3 Nuevas versiones del software del programador

Para explotar todas las capacidades del programador le recomendamos usar la versión más reciente de PG4U/PG4UW.

Usted puede descargar la última versión del software del programador (archivo PG4UARC.EXE /PG4UWARC.EXE) del sitio de Internet, en la sección Downloads.

- Por favor, para la información más reciente sobre el hardware y el software del programador visite www.bkprecision.com.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



2.2.4 Actualizando el software del programador

Copie PG4UARC.EXE/PG4UWARC.EXE a un temporal el directorio después de bajarlo. Después de la extracción usted verá todos los archivos disponibles necesarios para el proceso de instalación.

Siga los mismos pasos para una instalación estándar (Corra el Setup/Instale el programa). Usted puede eliminar todos los archivos de la carpeta temporal después de que el proceso de instalación se haya completado

2.3 Aplicando el 844A en la PC

Apague la PC y programador. Inserte el cable de conexión, incluido en el set del programador 844A, a un puerto paralelo libre de la PC, si su computadora sólo cuenta con un puerto paralelo, substituya el cable del programador por el cable de la impresora, conecte el extremo opuesto del cable con el programador, atornille en ambos conectores.

Esto es muy importante, principalmente para el extremo que va hacia el programador, aunque sustituir el cable de la impresora por el cable del programador cada vez que este se va a utilizar, es incómodo, no se recomienda utilizar el programador a través de un interruptor mecánico de impresora, el uso del interruptor electrónico no es posible, conecte el adaptador de corriente etiquetado con 15VDC a la toma de corriente y al programador.

En todo este tiempo LED de Resultado (y el LED de Energía) se encenderá, una vez que las luces del leed de energía estén en un punto bajo de brillo entonces el 844A está listo para funcionar, encienda la PC e inicie el programa de control.

2.3.2 Auto prueba y calibración

Si usted tiene la sensación, de que su programador no se comporta según sus expectativas (por lo menos cada 3 meses), por favor corra el auto prueba del programador usando el diagnóstico POD (figura 2.6), que contiene la configuración estándar del programador. Para el óptimo desempeño del programador 844A se recomienda cada 6 meses (no más) hacer la prueba extendida y comprobar la calibración. Todas estas actividades usted puede encontrarlas en el menú del diagnóstico de PG4UW.



Figura 2.6 Calibrador del 844A

2.3.2 Extracción Correcta del Programador de la PC.

Si usted no desea apagar su PC cuando conecte el 844A, hágalo de la forma siguiente: Cuando el programador esté conectado con la PC: PRIMERO inserte el cable de comunicación al programador y DESPUÉS suminístrele energía. Cuando desconecte el programador de la PC: PRIMERO desconecte el suministro de energía y DESPUÉS el cable de comunicación.



Capítulo 3. Utilización de grabadores con diferentes C.I.

En este capítulo se describirá la forma correcta para poder utilizar el programador 844A con los diversos tipos de Circuitos Integrados (CI) existentes, para poder realizar diversos proyectos y se desarrolle el conocimiento necesario para utilizar al máximo sus dispositivos.

3.1 Barra de Menú

Lista de funciones básicas de la barra de menú

Ventana del Archivo/Nombre del Archivo

Información del archivo cargado actualmente en el buffer

Ventana de Estado

Información acerca del estado del programador y del PG4U/PG4UW

Ventana de Direcciones, organización, tamaño, comienzo y el fin de direcciones del dispositivo, buffer y archivo.

3.2 Lista de Accesos Rápidos

<F1> Ayuda (Help) Llama al menú de Ayuda

<F2> Salvar (Save) Salva archivo

<F3> Cargar (Load) Carga un archivo dentro del buffer

<F4> Edita (Edit) Ver/Editar del buffer

<F5> Selección por Default Selecciona dispositivos de los últimos 10 seleccionados

<Alt+F5> Selección Manual Selección del Dispositivo por el tipo de dispositivo o nombre del fabricante

<F6> En blanco (Blank) Comprueba el espacio en Blanco

<F7> Leer Lee en contenido del dispositivo en el buffer

<F8> Verifica Compara el contenido del dispositivo con el del buffer

<F9> Programa el dispositivo

<Alt+Q> Salir sin guardar Termina PG4U/PG4UW

<Alt+X> Salir y Salvar Termina PG4U/PG4UW y también guarda los cambios.

<Ctrl.+F1> Despliega información adicional acerca del dispositivo seleccionado

<Ctrl.+F2> Borrar Llena el buffer con un valor dado

<Ctrl +Shift+F2> Llena el buffer con un valor aleatorio

3.3 Archivo

Este submenú se utiliza para la manipulación de archivos fuente, ajustes y visión de directorios, cambio de unidad, cambios de dirección de inicio y término del buffer para cargar y salvar archivos en formato binario, MOTOROLA, tecnología MOS, Intel Hexadecimal (extendido), Tektronix, ASCII, JEDEC, y POF.

3.3.1 Archivo/Carga

Analiza el formato del archivo y carga los datos del archivo especificado al buffer. Usted puede elegir el formato deseado (binario, MOTOROLA, tecnología MOS, Tektronix, Intel HEX extendido, ASCII, JEDEC y POF).



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



El programa de control almacena la última máscara válida para el listado del archivo. Usted puede salvar la máscara dentro del archivo config por el comando (Options/Save) Opciones/Salvar Opciones.

La habilitación de la opción Reconocimiento Automático de Formato de Archivo del menú de Comprobación (Check) dice al programa detectar el formato del archivo automáticamente. Cuando programa no puede detectar el formato del archivo a partir del formato que soporta, se asume que el formato del archivo es binario.

Cuando el Reconocimiento Automático de Formato de Archivo está deshabilitado en el programa se permite que el usuario seleccione manualmente el formato del archivo deseado de la lista de formatos de archivo disponibles en el panel de Formato de archivo seleccionado. Cuando el Formato Binario es seleccionado, puede ser especificado el valor de inicio del buffer. El valor inicial del buffer es la dirección del buffer del cual los datos leídos del archivo serán escritos en el buffer.

El acceso rápido < F3 > desplegará este menú de cualquier menú y en cualquier momento.

3.3.2 Archivo/Guardar

Salva los datos en el buffer, que se hayan creado, modificado o leído de un dispositivo en el destino especificado. Usted puede elegir el formato deseado (binario, MOTOROLA, tecnología MOS, Tektronix, HEX extendido de Intel, ASCII, JEDEC y POF).

El acceso rápido < F2 > desplegará este menú de cualquier menú y en cualquier momento

3.3.3 Archivo/Carga del Proyecto

Esta opción se utiliza para cargar el archivo del proyecto el cual contiene los datos salvados del buffer de configuración del dispositivo y la configuración de interfaz de usuario.

La leyenda LOAD PROJECT (Carga Proyecto) contiene la ventana adicional - (Project Description) Descripción del Proyecto – ubicada al final del diálogo. Esta ventana está para desplegar información acerca del archivo del proyecto seleccionado en el diálogo Cargar Proyecto.

La información del proyecto consiste en:

- Nombre y hora de la creación del proyecto.
- Versión del programa por el cual fue creado.
- Descripción escrita del usuario del proyecto (esta puede ser texto arbitrario, generalmente autor del proyecto y algunas notas más).

3.3.4 Archivo/Salvar Proyecto

Esta opción se utiliza para salvar el archivo de proyecto, que contiene ajustes de la configuración del dispositivo y datos salvados del buffer. Los Datos salvados al archivo de proyecto pueden ser restaurados en cualquier momento por el comando del menú File/Project.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



El diálogo Save Project (Salvar Proyecto) contiene tres ventanas adicionales dentro del panel de Descripción del Proyecto ubicadas al final del diálogo Save Project (Salvar Proyecto).

Las ventanas están para desplegar la información acerca del archivo de proyecto actualmente seleccionado en diálogo Save Project (Salvar Proyecto) e información sobre el proyecto actual, el cual tiene que ser salvado.

La información del proyecto consiste en:

- Nombre y fabricante del primer dispositivo seleccionado en el proyecto.
- Fecha y Hora de la creación del proyecto.
- Versión del programa por el cual el proyecto fue creado
- Descripción escrita del usuario del proyecto (puede ser texto arbitrario, generalmente autor del proyecto y de algunas notas más).

La primera ventana (la de más arriba) contiene la información alrededor actualmente el archivo de proyecto seleccionado en diálogo Salvar Proyecto (Save Project).

La segunda ventana (la del centro) contiene la información acerca de la configuración del programa real incluyendo el dispositivo actualmente seleccionado, programador activo, fecha y hora y versión del programa. Estos ajustes del programa real se utilizan para la creación de la descripción del proyecto líder.

La tercera ventana (la de abajo) es editable para el usuario y contiene la descripción del proyecto (texto arbitrario), que consiste generalmente del autor del proyecto autor y algunas notas.

3.3.5 Archivo/Recarga Proyecto

Elija esta opción para recargar un archivo recientemente usado.

Cuando usted utiliza un archivo, se agrega a la lista de Reload File (Archivo de recarga). Los archivos son numerados en orden dependiendo de la fecha en que fueron usados. Los archivos usados anteriormente usados se enlistan antes que los que se han utilizado con mayor anterioridad.

Para recargar un archivo:

1. Del menú file (Archivo), elija el archivo de la recarga.
2. La lista de archivos utilizados anteriormente se despliega.

Haga click sobre el archivo que desea recargar.

Nota: Al recargar un archivo se utiliza el formato del archivo, con el cual se guardó la última vez.

3.3.6 Archivo/Opciones del Proyecto

Esta opción se utiliza para desplegar/Editar las opciones del proyecto actualmente cargado. Las Opciones del Proyecto contienen una descripción básica del proyecto incluyendo los siguientes datos:

- Nombre y fabricante del dispositivo.
- Fecha de creación del proyecto.
- Versión del programa en que fue creado.

Descripción del proyecto definido por el usuario (texto arbitrario), ejemplo. Autor del proyecto y otros datos de texto para mayor descripción del proyecto.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



El usuario puede editar directamente sólo la descripción del proyecto definida por el usuario. El nombre del dispositivo, el fabricante, la fecha y la versión del programa son generados automáticamente por el programa.

3.3.7 Archivo/Carga la Tabla de encriptación

Este comando carga los datos de archivo binario del disco y los salva en la parte de memoria, reservada para una tabla de encriptación (de seguridad).

3.3.8 Archivo/Salvar tabla de encriptación

Este comando escribe el contenido de la parte de la memoria, reservado para una tabla de encriptación, en el archivo en el disco como un dato binario.

3.3.9 Archivo/Salir sin salvar

El comando distribuye la pila, cancela el buffer en el disco (si existe) y regresa al sistema de la operación.

3.3.9.1 Archivo/Salir y salvar

El comando distribuye la pila, cancela el buffer en el disco (si existe), salva el ajuste actual de los últimos 10 dispositivos al disco y regresa al sistema de la operación.

3.4 Dispositivo

Las funciones para un trabajo con los dispositivos programables seleccionados - Selector de dispositivo, lee datos del dispositivo, verifica el espacio en blanco del dispositivo, programa el dispositivo, verifica el dispositivo y borra el dispositivo.

3.4.1 Seleccionar dispositivo

Esta ventana permite seleccionar el tipo deseado del dispositivo de todos los dispositivos soportados por el programador actual. Es posible elegir el dispositivo por nombre, por el tipo o por fabricante. El dispositivo seleccionado es automáticamente guardado en el buffer dispositivos de default (máximo 10 dispositivos). Este buffer está accesible con el comando Dispositivo/Selecciona Dispositivos de default.

Si usted desea desplegar la información adicional sobre el dispositivo actual, utilice el acceso rápido <Ctrl+F1>. Este comando proporciona el tamaño del dispositivo, la organización, el algoritmo de programación y de una lista de programadores (incluyendo módulos auxiliares) que soportan este dispositivo.

Usted puede encontrar aquí información del paquete y también otra información de carácter general sobre el dispositivo actual.

3.4.1.1 Dispositivo/Selecciona de los dispositivos de default

Esta ventana permite seleccionar el tipo deseado del dispositivo de la lista de dispositivos de default. Éste es un buffer cíclico en el cual están almacenados los últimos 10 dispositivos incluyendo sus opciones de dispositivo. Esta lista es guardada al disco por el comando Archivo/ Salir y Salvar (File / Exit and Save).



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



Si usted desea desplegar información adicional sobre el dispositivo actual, utilice el acceso rápido <Ctrl+F1>. Este comando proporciona el tamaño del dispositivo, la organización, el algoritmo de programación y una lista de programadores (incluyendo módulos auxiliares) que soportan este dispositivo. Usted puede encontrar aquí información sobre el paquete y también otra información de carácter general sobre el dispositivo actual.

Utilice un para la cancelación del dispositivo actual de la lista de dispositivos de default. No hay posibilidad de vaciar esta lista, si usted repite esta operación. El último dispositivo permanece en el buffer y el no es aceptado.

3.4.1.2 Seleccionar Dispositivo/Todo

Esta ventana permite seleccionar el tipo deseado del dispositivo de todos los dispositivos soportados por el programador actual. Los dispositivos soportados se exhiben en una lista. El dispositivo puede ser seleccionado por un doble click en la línea del dispositivo de la lista con el nombre del fabricante deseado y el número del dispositivo deseado o introduciendo el nombre del fabricante y/o el número del dispositivo en la caja de búsqueda (use la barra espaciadora <Space> como un carácter de separación) y presione <Entrar> o dé un click en el botón de OK.

Presione la tecla <Esc> o dé un click en el botón de Cancelar en cualquier momento para cancelar la selección del dispositivo sin afectar el dispositivo actualmente seleccionado. El dispositivo seleccionado es guardado automáticamente en el buffer de dispositivos de default (máximo 10 dispositivos). Este buffer está accesible con el comando Dispositivo/Seleccionar dispositivos de default.

Si usted desea desplegar la información adicional sobre el dispositivo actual, utilice el acceso rápido <Ctrl+F1>. Este comando proporciona el tamaño del dispositivo, la organización, el algoritmo de programación y de una lista de programadores (incluyendo módulos auxiliares) que soportan este dispositivo. Usted puede encontrar aquí información del paquete y también otra información de carácter general sobre el dispositivo actual.

3.4.1.3 Seleccione dispositivo/Solamente del tipo seleccionado

Esta ventana permite el seleccionar del tipo deseado del dispositivo.

Primero - usted debe seleccionar un tipo de dispositivo (ej. EPROM) y el subtipo del dispositivo (ej. 64Kx8 (27512)), con el ratón o el teclado. Esto generará una lista de fabricantes y los dispositivos serán exhibidos.

El dispositivo puede ser seleccionado por doble click en la línea del dispositivo de la lista con el nombre del fabricante y el número del dispositivo deseado o introduciendo el nombre del fabricante y/o número del dispositivo en caja de búsqueda (use la barra espaciadora <Space> como carácter de separación) y presione <Entrar> o dé un click al botón de OK.

Presione la tecla <Esc> o dé un click en el botón de cancelar en cualquier momento para cancelar la selección del dispositivo sin afectar el dispositivo actualmente seleccionado.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



El dispositivo seleccionado es guardado automáticamente en el buffer de dispositivos de default (máximo 10 dispositivos). Este buffer está accesible con el comando Dispositivo/Seleccionar dispositivos de default.

Si usted desea desplegar la información adicional sobre el dispositivo actual, utilice el acceso rápido <Ctrl+F1>. Este comando proporciona el tamaño del dispositivo, la organización, el algoritmo de programación y de una lista de programadores (incluyendo módulos auxiliares) que soportan este dispositivo. Usted puede encontrar aquí información del paquete y también otra información de carácter general sobre el dispositivo actual.

3.4.1.4 Seleccionar Dispositivo/Solamente del fabricante seleccionado

Esta ventana permite seleccionar el tipo de dispositivo deseado por el fabricante. Primero es requerido que seleccione un fabricante en la caja de Fabricante usando el mouse o el teclado. Esto desplegará una lista de los dispositivos del fabricante.

El dispositivo puede ser seleccionado por doble click en la línea del dispositivo de la lista con el nombre del fabricante y el número del dispositivo deseado o introduciendo el nombre del fabricante y/o número del dispositivo en caja de búsqueda (use la barra espaciadora <Space> como carácter de separación) y presione <Entrar> o dé un click al botón de OK. Presione la tecla <Esc> o dé un click en el botón de Cancelar en cualquier momento para cancelar la selección del dispositivo sin afectar el dispositivo actualmente seleccionado.

El dispositivo seleccionado es guardado automáticamente en el buffer de dispositivos de default (máximo 10 dispositivos). Este buffer está accesible con el comando Dispositivo/Seleccionar dispositivos de default.

Si usted desea desplegar la información adicional sobre el dispositivo actual, utilice el acceso rápido <Ctrl+F1>.

Este comando proporciona el tamaño del dispositivo, la organización, el algoritmo de programación y de una lista de programadores (incluyendo módulos auxiliares) que soportan este dispositivo. Usted puede encontrar aquí información del paquete y también otra información de carácter general sobre el dispositivo actual.

3.4.2 Dispositivo/Selecione EPROM/flash por ID

Utilice este comando para la autoselección de una EPROM o una Flash como dispositivo activo por la lectura de la ID del dispositivo.

El programador puede identificar ciertos dispositivos automáticamente por la lectura leer el fabricante y la ID del dispositivo que están grabados dentro del chip.

Este sólo aplica para EPROM o Flash que soportan esta característica.

Si el dispositivo no soporta una ID del Chip y la ID del fabricante, un mensajes será desplegado en pantalla indicando a este como dispositivo desconocido o dispositivo no soportado.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



Si más dispositivos con ID de Chip e ID del fabricante idénticos son detectados, la lista de estos dispositivos será desplegada. Un dispositivo puede ser elegido de esta lista por seleccionar su número (o nombre de fabricante) de la lista y presionar <Entrar> (o dar clic k al botón de OK). Presione la tecla <Esc> o dé click al botón de Cancelar en cualquier momento para cancelar la selección del dispositivo sin afectar al dispositivo actualmente seleccionado.

ADVERTENCIA: El programa de control sólo soporta EPROM's y Flash con 28 y 32 pines. Cualquiera de los programadores determina el número de pines automáticamente. Para otros programadores usted debe de introducir el número de pines manualmente.

El programador aplica un alto voltaje a los pines apropiados en el socket. Esto es necesario para habilitar al sistema para leer la ID del dispositivo. No inserte dentro del socket un dispositivo que no sea un EPROM o Flash. Podría ser dañado cuando el programador aplique dicho alto voltaje. No recomendamos aplicar este comando a EPROM de tipo 2764 y 27128, porque la mayoría de ellos no soporta la ID.

3.4.3 Dispositivo/ Opciones de Dispositivo/ Opciones de Operación

Todos los parámetros de este comando son utilizados para el control del proceso de programación, serialización y control del archivo asociado. Este es un medio flexible el cual contiene campos asociados con el dispositivo actual y el tipo de programador. Los campos, que son válidos para el dispositivo actual pero que no son soportados por el programador, son desactivados. Estos parámetros son salvados en el disco junto con el dispositivo asociado por el comando Archivo/ Salir y Salvar (File/ Exit and Save).

3.4.3.1 Lista de Campos:

- Grupo de Direcciones:

Dirección inicial del dispositivo	(default 0)
Dirección final del dispositivo	(Tamaño de Disp. default -1)
Dirección inicial de Buffer (default 0)	

- Grupo de test de inserción:

Test de inserción	(HABILITADO por default)
Compruebe los datos de ID	(HABILITADO por default)

- Grupo de Ejecución de comando:

Verifica el espacio en blanco Antes de programar	(INHABILITADA por default)
Borrar antes de Programar	(INHABILITADA por default)
Verificar después de la lectura	(HABILITADA por default)
Verificar después de programar	(UNA VEZ, DOS VECES)
Verificar las opciones	(nominal VCC el 5%, nominal VCC el10%, VCC min VCC máx.)



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



3.4.3.2 Dispositivos/ Opciones del Dispositivo/ Serialización

La serialización es el modo espacial del programa. Cuando un modo es activado, un valor especificado es automáticamente insertado en la dirección predefinida dentro del buffer antes de programar cada dispositivo. Cuando más dispositivos son programados uno por uno, el valor del número de serie es cambiado para cada dispositivo automáticamente e insertado en el buffer antes de programar el dispositivo, entonces cada dispositivo tiene un número de serie único.

Hay dos tipos de serialización:

1. Modo de Incremento
2. Modo del Archivo

Si un dispositivo nuevo es seleccionado, la función de serialización se posiciona en un estado de default, es decir, inhabilitado. Los parámetros de serialización actual para el dispositivo actualmente seleccionado son guardados en el disco junto con el dispositivo asociado por el comando Archivo/ Salir y Salvar (File / Exit and Save)

Cuando el Modo de incremento es activado los siguientes parámetros actuales son guardados en el archivo de configuración: Dirección, Tamaño, Valor Serial, paso de incremento y ajuste de modo ASCII/ BIN, DEC / HEX, LS byte / MS Byte first.

Cuando el Modo de el Archivo es activado los siguientes parámetros actuales son salvados en el archivo de configuración: nombre del archivo de serialización de entrada y etiqueta actual, la cual indica la línea con el actual número de serie en el archivo de entrada.

Cuando el programa está en Modo de Multiprogramación (el programador de socket múltiple está actualmente seleccionado) la sección especial – Acción sobre los valores seriales no programados debido a error- es desplegada en el diálogo de serialización.

En esta sección dos opciones son habilitadas:

- 1.- Ignora los valores seriales no programados
- 2.- Agrega los valores seriales no programados a el archivo

Ignore los valores seriales no programados significa que los valores seriales no programados son ignorados y no se acciona con ellos.

Agrega los valores seriales no programados al archivo significa que los valores seriales no programados son agregados al archivo. El archivo de valores seriales no programados tiene el mismo formato de texto como el archivo de serialización para el modo de serialización “del archivo”. Entonces hay posibilidad para programar los valores seriales más tarde en el modo de serialización “del archivo”.

Si la programación del dispositivo es detenida por el usuario, el programa no cambiará los valores seriales listos para el siguiente paquete de dispositivos. La misma situación es si el programa del dispositivo está incompleto.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



Ejemplo. Por error de la prueba de inserción del dispositivo. Ignorando o escribiendo los valores seriales no programados se utiliza solamente cuando por lo menos uno de los dispositivos del actual paquete de dispositivos en el programador de módulo de socket múltiple está completamente programado y verificado sin errores.

3.4.3.3 Dispositivo/ Opciones del Dispositivo/ Serialización/ Modo de Incremento

El modo de Incremento habilita a asignar números de serie individuales a cada dispositivo programado. Un número de inicio introducido por el usuario será incrementado por el paso especificado para cada operación del programa del dispositivo y cargado en el formato seleccionado para especificar la dirección del buffer antes de la programación de cada dispositivo.

Hay las siguientes opciones:

Que el usuario pueda modificar por el modo de Incremento.

Tamaño del Número de Serie (S/N)

La opción del tamaño del S/N define el número de bytes del valor serial que serán escritos en el buffer.

Para los modos de serialización Binaria (Bin) los valores 1-4 son validos para el tamaño del S/N y para los modos de serialización ASCII los valores 1-8 son validos para el tamaño del S/N.

3.4.3.4 Dispositivo/ Opciones del Dispositivo/ Serialización/ Modo del Archivo

Usando el Método de el Archivo, los valores seriales son leídos del archivo de entrada especificado por el usuario y escritos en el buffer en la dirección especificada en el archivo de entrada.

3.4.4 Dirección (Address)

La opción de Dirección especifica del buffer, donde el valor serial ha de ser escrito. Note que el rango de dirección debe estar dentro de las direcciones inicial y final del dispositivo. La dirección debe de ser correctamente especificada entonces el último (más alto o más bajo) byte del valor serial debe de estar dentro del rango de direcciones de inicio y fin del dispositivo.

3.4.5 Valor Inicial

La opción de valor inicial especifica el valor inicial, del cual la serialización empezará. Generalmente el valor máximo para la serialización es \$1FFFFFFF en una palabra larga de 32 bits.

Cuando el valor serial actual excede el valor máximo, los tres bits más significativos del número de serie son puestos a cero.

Después de esta acción el número esta siempre dentro del intervalo 0.

\$1FFFFFFF (este es un estilo básico de Handling Overflow).

3.4.6 Paso

La opción de paso especifica el paso de incremento del incremento del valor serial.

Modo del S/N

La opción de S/N define la forma por la cual el valor serial ha de ser escrito en el buffer.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



Las dos opciones Son habilitadas: **ASCII y Bin.**

ASCII.- Significa que el número de serie está escrito en el buffer como una cadena ASCII. Por ejemplo, el número \$0528CD está escrito en modo ASCII en el buffer como 30h 35h 32h 38h 43h 44h ('0' '5' '2' '8' 'C' 'D'), es decir, seis bytes.

BIN.- Significa que el número de serie es escrito directamente en el buffer. Si el número de serie tiene más de un byte de longitud, puede ser escrito en uno de los dos posibles ordenes del Byte. El orden del byte puede ser cambiado in el campo "Salvar en el buffer" (Save to Buffer).

3.4.7 Estilo

Lo opción de Estilo define la base del número de serie. Hay dos opciones: **Decimal y Hexadecimal**

Decimal: Los números son introducidos y desplegados usando los caracteres '0' al '9'.

Hexadecimal: Los números y también usa los caracteres de la 'A' a la 'F'. El caso especial es el Decimal a Binario, que significa que el estilo de números BCD (BCD significa que el número decimal está almacenado dentro del número hexadecimal, es decir, cada nibble (equivale a 4 bits) debe de tener el valor de 0 a 9). Los valores "A" a la "F" no son considerados como nibbles de los números BCD. Seleccionar la base en las opciones de "Estilo" antes de introducir los números del valor serial inicial o el paso.

3.4.8 Salvar en el buffer

La opción de Salvar en el buffer especifica el orden del byte del valor serial para escribir al buffer. Esta opción es utilizada para el modo de S/N Binario (para el modo ASCII este no tiene efecto).

Las dos opciones disponibles:

El primer LSByte (utilizado por los procesadores Motorola) colocará el primer byte más significativo a la dirección más baja en el buffer.

El primer MSByte (utilizado por los procesadores Motorola) colocará el primer byte más significativo a la dirección más baja en el buffer.

3.5 Nombre de Archivo

La opción de Nombre de Archivo especifica el nombre del archivo del cual las direcciones seriales y los valores serán leídos. El archivo de entrada debe de tener el formato especial, el cual está descrito en el formato de archivo bajo la serialización del archivo.

3.5.1 Etiqueta de Inicio

La etiqueta de inicio define la etiqueta de inicio en el archivo de entrada. La lectura de los valores seriales de los archivos iniciales de la etiqueta de inicio definida.

3.5.2 Dispositivo/ Verifica Espacio en Blanco

Este comando permite verificar el espacio en blanco de todos los dispositivos o su parte en blanco si existe. El programa de control reporta un resultado de esta acción por una escritura en una mensaje de Precaución en la ventana de INFO.

El comando del Menú Dispositivo/ Opciones de Dispositivo/ Opciones de Operación permite ajustar otra área de trabajo como la Estándar.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



3.5.3 Dispositivo/ Leer

Este comando permite leer todos los dispositivos o su parte dentro del buffer. El programa de control reporta un término de esta acción por un mensaje escrito en la ventana de INFO.

El comando del menú Dispositivo/ Opciones de Dispositivos/Opciones de Operación permite ajustar otra área de trabajo como la estándar. Ajustando una opción de verifica datos después de leer en este comando del menú significa una alta confiabilidad para la lectura del dispositivo.

3.5.4 Dispositivo/ Verificar

Este comando comparte los datos programados de todos los dispositivos o de su parte con los datos en el buffer. El programa de control reporta un resultado de esta acción por un mensaje escrito en la ventana de INFO.

El comando del menú Dispositivo/ Opciones de Dispositivo/Opciones de Operación permite ajustar otra área de trabajo como la estándar.

Por el ajuste en el menú Opciones / Despliega los errores del comando permite escribir los errores encontrado en la pantalla o escribir los mismos en el archivo VERIFY.ERR. En el Modo de mostrar los errores a la pantalla puede desplegar máximo 45 del programa el primero las diferencias encontradas, las cuales están localizadas por las direcciones donde estas fueron encontradas.

3.5.5 Dispositivos / Programar

Este dispositivo permite la programación de todos los dispositivos o en parte de ellos por los datos en el buffer. El programa de control reporta un resultado de esta acción por un mensaje de error escrito en la ventana de INFO.

El comando del menú Dispositivo/ Opciones de Dispositivo/ Opciones de Operación permite ajustar otra área de trabajo como la estándar, ajustar otras opciones de operación para el control del proceso de programación.

3.5.6 Dispositivo/ Borrar

Este comando permite borrar todos los dispositivos programables. El programa reporta el término sin errores o el término con errores por el reporte escrito de Precaución en la pantalla.

3.5.7 Dispositivo/ Prueba

Este comando ejecuta una prueba con el dispositivo seleccionado de la lista de dispositivos soportados (ejemplo. RAM estática), en programadores, que soportan esta prueba.

3.5.8 Dispositivo / Información del Dispositivo

El comando provee información adicional acerca del dispositivo actual- tamaño del dispositivo, organización, algoritmo de programación y una lista de programadores (incluyendo los módulos auxiliares) que soportan este dispositivo. Usted puede encontrar aquí un paquete de información y también otra información de carácter general acerca del dispositivo actual.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



La llave reservada <Ctrl+F1> mostrará este menú de cualquier menú y en cualquier momento.

Buffer.- Este submenú es utilizado para la manipulación del buffer, el bloque de operación, llenar una parte del buffer con secuencia, borrado, comprobación y el curso de editar y visualización con otros campos (encontrar y reemplazar secuencia, imprimir).

3.6 Opciones/ Opciones Generales

El menú de opciones contiene los comandos que permiten ver y cambiar varios parámetros de default.

El diálogo de opciones generales permite al usuario controlar las siguientes opciones del programa.

3.6.1 Opciones de Archivo

La página de Opciones de Archivo le permite ajustar las máscaras del archivo, auto-recarga del archivo actual y elegir el reconocimiento del formato de archivo para los archivos cargados.

Máscaras de formato de archivo: es utilizado para ajustar las máscaras del Nombre de Archivo para utilizarse como un filtro para el listado de archivo: En Archivo/ Salvar (In File/ Save) y la ventana de Archivo/ Cargar Archivo para todos los formatos de archivo. La máscara debe de contener uno de los comodines (*,?) por lo menos y debe de ser aplicado correctamente a la sintaxis.

Extensión default del archivo del proyecto: Es utilizado para ajustar la extensión de archivo de proyecto utilizado como extensión de default en los diálogos Archivo/ Cargar Proyecto y Archivo/ Salvar

Proyecto.

En el grupo cuando el archivo actual es modificado por otros procesos puede ser puesto el modo de recarga de el archivo (actual) cargado actualmente.

Hay tres opciones:

- 1.- El prompt antes de recargar el archivo.
- 2.- Recargar automáticamente.
- 3.- Ignorar cambios escanear el archivo actual.

Cargar Formato de Archivo permite ajustar el modo de reconocimiento de archivo de formato para los archivos cargados.

Cuando el formato de archivo automático es seleccionado, el programa analiza el formato del archivo a cargar y prueba el archivo para cada uno de los formatos soportados, que están disponibles en el programa. Si el formato es compatible con uno de los formatos soportado, el archivo es leído en el buffer en el formato detectado.

El formato de Archivo manual permite al usuario seleccionar explícitamente el formato de archivo deseado de la lista de formatos de archivo soportados. El archivo puede ser cargado incompletamente o incorrectamente, si el formato de archivo no es compatible con en formato de archivo seleccionado por el usuario.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



3.6.2 Opciones de Archivo Hexadecimal

Esta página contiene algunas opciones para el control de carga de cualquiera de los formatos Hexadecimales.

La primera opción fija Borrar buffer (con el valor deseado) automáticamente antes de cargar por cualquiera de los formatos Hexadecimales.

La segunda opción fija un Offset negativo, que es utilizado para la modificación de direcciones de datos por el cargar cualquier archivo Hexadecimal entonces, esos datos pueden ser escritos a las direcciones del buffer existentes.

Ejemplo:

Un archivo contiene los datos del Formato Motorola S. Un bloque de datos iniciado en la dirección FFFF0H. Es un formato S2 con longitud de arreglo de dirección de 3 bytes. Para toda lectura de datos usted puede fijar un valor de offset negativo a FFFF0H. Esto significa, que el offset será restado de las direcciones reales actuales y entonces los datos serán escrito de la dirección del buffer 0.

Precaución: El valor de offset negativo se resta de la dirección real y por lo tanto el resultado de la resta puede ser un número negativo. Porque toma cuidado de el ajuste correcto de este valor.

La tercera opción fija Re direccionar datos de el archivo de el formato Hexadecimal extendido de Intel a las direcciones físicamente existentes del buffer. Eso significa, que el usuario será incitado a introducir el segmento, que está asociado con el segmento más bajo en el archivo (Tipo de Grabado 02) y los otros segmentos serán modificados con la diferencia. Esta opción no es válida para el Formato Hexadecimal "Simple" de Intel. Default significa estado inactivo.

El archivo contiene dos grabados del tipo 02 con direcciones F000H y F800H. Cuando introducimos el nuevo segmento por ejemplo 0H, los datos del segmento F000H serán dirigidos al segmento 0H y los datos similares de en segmento F800H a el segmento 800H. El parámetro Default es No activo.

3.6.3 Lenguaje

Esta página le permite seleccionar otro lenguaje para la interfaz de usuario tales como menú, botones, diálogos, información y mensajes. También permite seleccionar la el archivo de ayuda deseado en otro lenguaje. Para el soporte de otro lenguaje de interfaz de usuario el archivo de definición de lenguaje es requerido.

3.6.4 Sonido

La página de sonido permite al usuario seleccionar el modo de sonido del programa. El programa genera sonidos después de algunas actividades, Ejemplo. Actividades sobre dispositivos (programar, verificar, lectura). El programa genera sonidos también cuando un mensaje de Error o Precaución es desplegado en pantalla,



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



El usuario puede seleccionar ahora sonido de el sonido del Sistema de MS Windows (requiere de tener la tarjeta de sonido instalada), bocina de la PC o ningún sonido.

3.6.5 Otros

La página Otros permite al usuario manejar otro parámetros del programa. En el panel de Botones de Herramientas, las opciones de pantalla indirectas en los botones de la barra de herramientas en la ventana principal del programa pueden ser modificadas. En el panel de Directorio de Inicio puede ser seleccionado el modo de directorio de seleccionar el directorio cuando el programa inicia.

El directorio default de inicio significa que el directorio, del cual el programa el llamado. El directorio en el cual el programa fue últimamente terminado significa que el último directorio actual donde el programa fue últimamente cerrado. Este directorio asume el primer directorio de la lista del historial de directorios.

3.7 Programando un dispositivo - La manera más corta

Utilice el acceso rápido <Alt+F5> para introducir el nombre de dispositivo y/o fabricante para seleccionar el tipo deseado de dispositivo o tarjeta.

Si usted desea copiar un dispositivo existente, insértelo en el socket ZIF del programador y entonces presione < F7>.

Si usted desea programar una tarjeta con datos de un disco o Diskette presione <F3 > y seleccione el archivo apropiado en el buffer.

Después inserte su dispositivo en el ZIF socket.

Para comprobar si el dispositivo está en blanco - presione < F6 >.

Ahora usted puede programar el dispositivo presionando < F9 >.

Después de programar puede realizar una verificación adicional presionando < F8 >.

3.8 Manipulación con el dispositivo programado

Después de seleccionar el dispositivo deseado para su trabajo, usted pueda insertarlo en el socket ZIF abierto (palanca arriba) y cierre el socket (palanca abajo). La orientación correcta del dispositivo programado en el socket ZIF se muestra en el cuadro cerca del socket ZIF en la cubierta del programador. No remueva el dispositivo programado hasta que el leed de BUSY se apague.

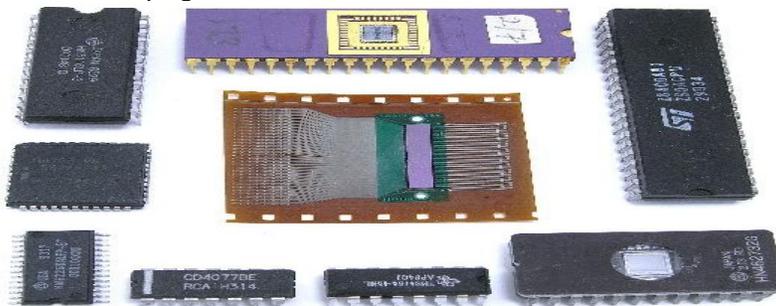


Figura 3.1 Circuitos Integrados más comunes para utilizarlos en los grabadores



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



CONCLUSIONES

El/los Objetivo(s) primordial/es se cumplieron en su totalidad, debido a que al final de este proyecto podemos presumir de:

- Contar con una infraestructura capaz de cubrir la demanda de todos y cada uno de los proyectos requeridos por los alumnos para el desarrollo total de sus habilidades en las asignaturas dedicadas a mostrar el funcionamiento real de los Circuitos Integrados.
- Contamos con un área dedicada exclusivamente para el desarrollo del mismo.
- Se cuenta con 3 Programadores los cuales son capaces de solventar la gran demanda de alumnos para la creación de diversos proyectos.
- Se cuenta con el conocimiento suficiente para operar los programadores y también distribuir el conocimiento adquirido tanto a los alumnos, como a los profesores dedicados a las materias a fin de la Facultad.
- Se es capaz de solventar cualquier problema que se presente y dar respuesta pronta al problema.

Al final se cumplen todos y cada uno de los objetivos establecidos y se logra una satisfacción total así como personal, por haber podido colaborar y aprender para el bien de la Facultad.



IMPLEMENTACION Y APLICACIÓN DE GRABADORES PARA CIRCUITOS INTEGRADOS



BIBLIOGRAFIA

- David Salomon, Assemblers and Loaders 1993.
- ANGULO, José M. Microcontroladores PIC, Diseño práctico de aplicaciones. Segunda edición. Mc Graw Hill, 1999
- Principles of Computer Architecture (POCA) – ARC Tools.
- Murdocca, Miles J. y Huring, Vincent P. Principles of Computer Architecture (2000), Prentice Hall, ISBN 0-201-43664-7.
- Dominic Sweetman, See MIPS Run. Morgan Kaufmann Publishers. ISBN 1-55860-410-3.
- Robert Britton: MIPS Assembly Language Programming. Prentice Hall. ISBN 0-13-142044-5.
- John Waldron: Introduction to RISC Assembly Language Programming. Addison Wesley. ISBN 0-201-39828-1.
- Luis Sánchez Pérez. EA4-NH Apartado 421, TOLEDO
- URUÑUELA, José M. Microprocesadores, Principios y Aplicaciones. Mc. Graw Hill, 1989.
- GARCIA, Hugo. Teoría de programación de autómatas. Ed. Limusa, 1990.