

67
24



**UNIVERSIDAD NACIONAL
AUTONOMA DE MEXICO**

FACULTAD DE INGENIERIA

**"DISEÑO DE UN SISTEMA TUTORIAL SOBRE
MICROPROCESADORES UTILIZANDO
MULTIMEDIOS"**

T E S I S

**QUE PARA OBTENER EL TITULO DE:
INGENIERO MECANICO ELECTRICISTA
(AREA ELECTRICA Y ELECTRONICA)**

P R E S E N T A :

GERMAN FIGUEROA ESPINOZA

DIRECTOR: DR. FELIPE LARA ROSANO



MEXICO, D. F.

1996

**TESIS CON
FALLA DE ORIGEN**

**TESIS CON
FALLA DE ORIGEN**



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



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.

67
24



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE INGENIERIA

"DISEÑO DE UN SISTEMA TUTORIAL SOBRE
MICROPROCESADORES UTILIZANDO
MULTIMEDIOS"

T E S I S

QUE PARA OBTENER EL TITULO DE:
INGENIERO MECANICO ELECTRICISTA
(AREA ELECTRICA Y ELECTRONICA)

P R E S E N T A :

GERMAN FIGUEROA ESPINOZA

DIRECTOR: DR. FELIPE LARA ROSANO



MEXICO, D. F.

1996

TESIS CON
FALLA DE ORIGEN

DEDICATORIAS

**A mis padres:
Maria Cruz del Rosario y Germán Efraín**

**A mis hermanos:
MaryCruz y Bernardo**

A mis maestros

A Marta, in Memoriam

AGRADECIMIENTOS

A la Universidad Nacional Autónoma de México, por la intensidad
A la Facultad de Ingeniería de la UNAM, por la formación
Al Instituto de Ingeniería de la UNAM, por los espacios y recursos
Al Dr. Felipe Lara Rosano, por la oportunidad
Al Ing. Pedro Guerrero Briseño, por el apoyo
A Juan Cristobal Murillo, por el impulso

Diseño de un sistema tutorial sobre microprocesadores utilizando multimedios

Tesista: Germán Figueroa Espinoza
Director: Dr. Felipe Lara Rosano

ÍNDICE

ÍNDICE

CONTENIDO

	Página
Introducción	1
Objetivos	3
Estructura	6
1. Antecedentes de la Tecnología Multimedia	7
1.1. La Tecnología Multimedia	8
1.2. Arquitectura de las Aplicaciones Multimedia	16
1.3. Tipos de Aplicaciones Multimedia	20
1.4. La Multimedia en la Educación	25
1.5. Los Medios en las Comunicaciones	27
2. Sistemas Tutoriales en Computadora	29
2.1. Tutoriales en Computadora	30
2.2. La Comunicación y el uso de la Tecnología	31
2.3. Comunicación del Conocimiento	32
2.4. La Interfaz de Usuario	33
2.5. Decis	35
3. Microprocesadores	36
3.1. El Microprocesador	37
3.2. Memoria	58
3.3. Periféricos	72
3.4. Interrupciones	76
3.5. Aplicaciones	83
4. Diseño del Tutorial	89
4.1. Diseño de MICRO	90
4.2. Navegación del Tutorial	93
5. Implementación	98
5.1. Hardware y Software usados en la elaboración de MICRO	99
5.2. Hardware y Software necesarios para poder usar MICRO	100
5.3. Elaboración de la Interfaz de Usuario de MICRO	101
5.4. Metodología de Diseño y Desarrollo de Aplicaciones Educativas Multimedia	104
5.5. Funcionamiento del Tutorial	109
6. Conclusiones	110
Bibliografía	113

INTRODUCCIÓN

INTRODUCCIÓN

El presente trabajo de tesis consiste básicamente en la elaboración de un Sistema Tutorial para la enseñanza de Microprocesadores utilizando Multimedia. Para una abreviación mnemotécnica, el sistema fue bautizado como **MICRO**.

La introducción del microprocesador fue una pauta en la historia de la electrónica. En las distintas universidades, en carreras como Ingeniería en Electrónica, Ingeniería en Computación, Ingeniería en Sistemas, Ingeniería en Telecomunicaciones, Informática, etc., se imparten materias en las que se requiere una comprensión del tópico de los microprocesadores. En particular, en la asignatura de microprocesadores, que por ejemplo se imparte en la carrera de Ingeniería Mecánica Eléctrica área Electrónica de la Facultad de Ingeniería de la Universidad Nacional Autónoma de México, será de gran utilidad la inclusión de la computadora como una herramienta didáctica.

La aplicación de la computación a todas las actividades de la sociedad así como la competencia que actualmente se vive en el mercado de trabajo hacen que los estudiantes de Ingeniería, en especial de las ingenierías mencionadas en el párrafo anterior, se familiaricen con las tendencias que sigue el mundo del cómputo. Una de esas tendencias es la *Tecnología Multimedia* o *Tecnología Multimedia* como es comúnmente llamada. Esta tecnología parece tener aplicaciones ilimitadas, pues se extiende a todas las áreas del uso de computadoras.

Como parte del presente trabajo de tesis fue diseñado **MICRO**. Se espera que su utilización haga más fácil y amena la comprensión del tema de los microprocesadores, así como que introduzca a los alumnos en el mundo de la multimedia, de creciente aplicación en el mercado actual de cómputo.

OBJETIVOS

OBJETIVOS

El sistema **MICRO**, como fue bautizado este Tutorial sobre Microprocesadores utilizando **Multimedios**, está orientado a estudiantes de Ingeniería en Electrónica, Ingeniería en Computación, Ingeniería en Sistemas, Ingeniería en Telecomunicaciones, Informática y carreras afines. Es de especial utilidad en la enseñanza de la materia de Microprocesadores, impartida por ejemplo en la carrera de Ingeniería Mecánica Eléctrica área Electrónica de la Facultad de Ingeniería de la Universidad Nacional Autónoma de México (UNAM).

El principal objetivo de este trabajo de tesis es el siguiente:

- ◆ **Desarrollo de un sistema computarizado utilizando multimedios que sirva como auxiliar en la asignatura de microprocesadores.**

MICRO imparte un curso sobre microprocesadores basándose en describir las características de un microprocesador genérico de 8 bits, el cual contiene a su vez las características principales de los microprocesadores comerciales. Al terminar de recorrer el sistema **MICRO**, el estudiante habrá asimilado los conceptos de un curso básico de microprocesadores, y será capaz de entender lo siguiente:

- Que es un microprocesador
- Como surgió el microprocesador
- La relación de los microprocesadores con las computadoras y otros sistemas con microprocesador
- En Que consiste la arquitectura de un microprocesador y el funcionamiento de sus componentes
- La operación de un microprocesador
- Que es el modelo de programación
- Que es la memoria
- Que son las memorias ROM y RAM
- Que es un mapa de memoria
- Que es y como funciona el stack
- El repertorio de instrucciones de un microprocesador genérico
- Los modos de direccionamiento de un microprocesador genérico
- Que son los periféricos
- Que son los puertos
- Que son los puertos paralelo
- Que son los puertos serie
- Que son los temporizadores y contadores
- Que son las interrupciones
- Que son las interrupciones hardware
- Que son las interrupciones software
- Como se ha aplicado el microprocesador a distintos campos e industrias

Los objetivos particulares del sistema **MICRO** son los siguientes:

- Impartir en un curso sencillo, ameno y concreto el tópico de Microprocesadores, en base a un microprocesador genérico cuyas características sean semejantes a las de los microprocesadores en el mercado.
- El estudiante podrá, después de haber llevado a cabo el curso presentado en **MICRO**, comprender los conceptos básicos referentes a cualquier microprocesador.
- Comprendiendo los conceptos básicos de cualquier microprocesador, el estudiante será capaz de consultar y entender el manual de cualquier microprocesador.

- El alumno podrá manejar cualquier microprocesador, basándose en la comprensión de los elementos básicos de un microprocesador genérico y habiendo consultado el manual correspondiente.
- Proporcionar al estudiante un medio de aprendizaje autodidacta, interactivo y ameno sobre el tópico de microprocesadores, basándose en tecnología avanzada.
- Proporcionar a los profesores de asignaturas como microprocesadores, arquitectura de computadoras, etc., un auxiliar en la enseñanza de microprocesadores.
- Poner al estudiante en contacto con tecnología de punta al interactuar con hardware y software orientados a multimedia.
- El sistema tutorial **MICRO**, debe ser completamente transportable ya que debe permitir que los estudiantes lo puedan utilizar en cualquier IBM-PC o compatible que soporte el ambiente gráfico *Windows*.
- Explotar y aprovechar los recursos que brinda la tecnología multimedia para el diseño y construcción de aplicaciones interactivas orientadas a la educación.
- Propiciar el desarrollo de aplicaciones multimedia en general, y orientadas a la educación en cualquier ámbito en particular.
- Capacitar al alumno en el uso de la tecnología multimedia, al mismo tiempo que se despierta su interés no sólo por el uso de ésta, sino también en llegar a ser él mismo un diseñador, desarrollador y programador de aplicaciones de este tipo.
- Hacer uso de nuevas formas de interacción hombre-computadora a fin de mantenerse a la vanguardia tecnológica.
- Aprovechar la gran capacidad instalada en nuestro país de computadoras IBM-PC o compatibles con microprocesador 80386 o superior, y la gran difusión del ambiente gráfico *Windows*.
- Integrar el uso de sistemas y herramientas multimedia a los programas de estudio de las diversas carreras que se imparten en la Facultad de Ingeniería de la UNAM.

ESTRUCTURA

Este trabajo consta de 6 capítulos:

Capítulo 1.	Antecedentes de la tecnología multimedios.
Capítulo 2.	Sistemas tutoriales en computadora.
Capítulo 3.	Microprocesadores.
Capítulo 4.	Diseño del tutorial.
Capítulo 5.	Implementación.
Capítulo 6.	Conclusiones

El capítulo 1 presenta un esbozo general de la tecnología multimedia, desde su surgimiento, definición, categorías de multimedia, tipos de arquitecturas, tipos de aplicaciones multimedia, incursión en la educación y comunicación por parte de la multimedia. En el capítulo 2 se habla sobre los tutoriales en computadora, qué son, en qué campos se aplican, y cómo deben ser diseñados. En el capítulo 3 se expone la teoría básica de microprocesadores, misma que es la base teórica del sistema tutorial **MICRO**. En el capítulo 4 se muestra el diseño del tutorial en base a la estructura del mismo. En el capítulo 5 se ilustra cómo fue realizado el tutorial, siguiendo pautas, normas y tecnologías específicas. En el capítulo 6 se habla de las conclusiones que arrojó la realización de este trabajo de tesis.

Capítulo 1

Antecedentes de la tecnología multimedios

1. ANTECEDENTES DE LA TECNOLOGÍA MULTIMEDIOS

1.1. LA TECNOLOGÍA MULTIMEDIA

1.1.1. ANTECEDENTES

El proceso de asimilación y manejo de sistemas de cómputo, ha sufrido cambios considerables (aunque no vastos) en lo respectivo al tipo de usuarios que los utilizan. Los desarrolladores de hardware y software han realizado grandes esfuerzos en la "masificación" de la tecnología de cómputo, desarrollando interfaces y mecanismos de programación más "fáciles" de utilizar, necesidad producto de lo complejo que resultaba hace diez años desarrollar aplicaciones "integrales" (programación de sistemas) en la computadora. Los lenguajes de cuarta generación nacieron con el objetivo de permitirle a cualquier usuario de computadoras diseñar y desarrollar aplicaciones "integrales" mediante el manejo de instrucciones "sencillas" e interfaces gráficas (como las hojas de cálculo, procesadores de textos, desarrollo de bases de datos, etc.).

El Laboratorio de multimedia (MEDIALAB) del Instituto Tecnológico de Massachusetts (MIT), desde su fundación en 1980, ha destinado parte de sus esfuerzos al desarrollo de técnicas y metodologías para el desarrollo de aplicaciones docentes que utilicen la tecnología computacional. En 1987 el MIT se dividió en 11 departamentos de desarrollo, tres de los cuales se encuentran directamente relacionados con las actividades de transmisión y difusión del conocimiento así como la representación y manejo de interfaces hombre-máquina en la ejecución de aplicaciones computacionales para la educación. Los tres departamentos de desarrollo del MIT son:

- **Sistemas de Trabajo para Lenguaje Visual**, incursionando básicamente en el trabajo de computadoras gráficas y diseño de sistemas visuales.
- **La Escuela del Futuro**, donde la idea principal es determinar las características y los avances que conllevan el uso de computadoras en diferentes grados de estudio.
- **Interfaces Hombre-Máquina**. Desarrollo e investigación de máquinas que puedan leer labios y ojos, teniendo como objetivo la lectura de la mente (comportamiento neurofisiológico) mediante máquinas computarizadas.

Para incrementar el desarrollo tecnológico, es importante fomentar y acrecentar el acervo de talento humano, basado en el uso y manejo de tecnología de punta que le permita avanzar y desarrollarse eficientemente. La tecnología multimedia permite incrementar los niveles de eficiencia en la capacitación, así como el manejo e interacción de diversos tipos de información, integrando diferentes medios, para obtener un resultado unitario. Estudios hechos por la Universidad de California han demostrado que "la interactividad incrementa la inteligencia"; la multimedia es una de las tecnologías que permiten realizar una interactividad total entre el usuario y el conocimiento con el uso de computadoras.

1.1.2. DEFINIENDO MULTIMEDIA

En las tres últimas décadas los cambios que han sufrido los medios de comunicación e información giran en torno al uso de sistemas electrónicos, mismos que hacen más eficiente la reproducción y manipulación de información. En el caso de la fonografía los cambios de discos analógicos a discos digitales (CD) mejoraron las características de codificación, transmisión y fidelidad de la información auditiva. En el ámbito de la computación, las arquitecturas complejas: computadoras que necesitaban de grandes espacios y condiciones especiales para operar, se transformaron en sistemas pequeños con mayores capacidades, desarrollándose paralelamente nuevos mecanismos y dispositivos de almacenamiento y manipulación de información, de cintas magnéticas a discos magnéticos, y actualmente a CDROM. La computadora en la actualidad, se está transformando en uno de los pilares más importantes en los procesos de comunicación masiva o que

requieren del uso e integración de grandes volúmenes de información. Las computadoras como algunos autores lo consideran, han ocasionado la revolución tecnológica hacia la "era de la información". Las técnicas desarrolladas en torno al incremento de la eficiencia para manejar y transmitir información utilizando computadoras, han sustentado sus avances tecnológicos en la simultaneidad que los usuarios de la información puedan tener con el sistema que maneja dicha información. Los sistemas computarizados han pasado de ser mecanismos que almacenan y procesan información a mecanismos que interactúan con el usuario en ambientes cada vez más agradables y eficientes. En las últimas décadas se ha desarrollado una tecnología de cómputo orientada a la transmisión de información con altos niveles de interactividad, misma que permite incrementar la eficiencia en la difusión transmisión de conocimiento. Esta tecnología se denomina "**Multimedia**", y es común nombrarla también "**Tecnología Multimedia**".

El concepto de multimedia tiene dos raíces, "**multi**" muchos y "**media**" medios, por lo cual se puede decir que el concepto multimedia está asociado al uso, manejo, integración o manipulación de muchos medios o diferentes tipos de datos e información. Dentro de la tecnología multimedia, los *medios* están representados por diferentes tipos de información o datos; dicha información es integrada dentro de un sólo *medio*, por lo que los datos o información dentro del *medio* integrador adquieren un formato único. En sus inicios el concepto de multimedia fue asociado a la integración de más de un tipo de datos (provenientes de diferentes *medios*), como fue el caso de las presentaciones hechas utilizando audio y fotografías o diapositivas, diapositivas y video, etc. Dentro de los sistemas de cómputo algunos autores definieron como multimedia a las primeras aplicaciones que integraron texto y gráficas, pero considerando que la computadora es un *medio*, esta primera definición carece de consistencia en referencia a la definición derivada de las raíces etimológicas de la palabra *multimedia*. Actualmente en el ámbito de cómputo podemos definir como un sistema multimedia al que permite integrar y manipular datos e información provenientes de diferentes *medios*. Para ello podemos subdividir en dos rubros la información y datos de acuerdo a su origen:

- Datos e información provenientes de *medios* externos a la computadora como son fotografías, video, voz, música, etc.,
- Y datos e información creada dentro de la computadora como pueden ser: textos, gráficas, dibujos, animaciones, etc.

Ahora bien, los diferentes tipos de datos capaces de ser procesados por la computadora pueden clasificarse en cinco:

- Textos
- Imágenes fijas (fotografías, gráficas, dibujos, etc.)
- Video (imágenes en movimiento)
- Animaciones
- y Audio (voz, música, MIDI, etc.)

Los multimedia o la multimedia como es más comúnmente conocida, es un concepto definido a partir del uso de las computadoras, que básicamente radica en la integración de múltiples medios de información, en un ámbito único (computadora). Con sistemas multimedia, es posible interactuar con dispositivos físicos que facilitan la transmisión de conocimiento así como el manejo y control de sistemas que interactúan con el medio ambiente.

La multimedia es una tecnología. Actualmente, a pesar del amplio rango de tecnologías que la multimedia a su vez abarca, se considera que una aplicación es multimedia cuando hay de por medio un procesador digital, como por ejemplo en el caso de las computadoras digitales.

Una de las definiciones más popularizadas de la tecnología multimedia es la siguiente:

"Multimedia es la integración de un mínimo de tres de los cinco tipos de datos capaces de ser procesados por una computadora".

Esto quiere decir que una aplicación es *aplicación multimedia* cuando involucra por los menos tres tipos de datos capaces de ser procesados por computadora (hablando de los tipos de datos mencionados anteriormente).

Las primeras computadoras personales no tenían gráficas. Se tenía el sistema MDA de 80 columnas y 25 renglones de texto y luego vinieron el color y las distintas resoluciones: CGA, EGA, VGA.

La emergencia de la multimedia es similar a lo que pasó cuando el stereo reemplazó al mono. Esto sucedió porque el stereo es más natural. Lo mismo pasa con multimedia pues nos hace ver lo poco natural que es trabajar todo el día con una computadora sin sonidos, animaciones y música.

Para comprender el surgimiento de la multimedia hay que mirar a nuestro alrededor: todo en el mundo es un conjunto multisensorial de imágenes, sonidos y olores. La gente está "diseñada" para trabajar en ese tipo de ambiente. Los humanos necesitamos muchos tipos diferentes de sensaciones para comprender y sentirnos bien en el mundo.

Observando el desarrollo de la computación, se puede decir que lo que se tiene actualmente considerado como "normal" en computación, es "monomedia". En el futuro, las computadoras podrán hablar y escuchar, entretener y comunicar. La multimedia es un paso lógico en esa dirección.

Para el procesamiento de los diferentes tipos de datos, la generación de computadoras en turno (4a) requiere de una serie de aditamentos y dispositivos especiales para realizar las interfaces y transformación de los datos de audio, video e imagen fija a formatos digitales; aunado a esto, los sistemas multimedia cuentan con software que les permite *editar, crear e integrar* los diferentes tipos de datos.

La integración de datos del lenguaje natural en la tecnología multimedia, es una de las herramientas más importantes para su desarrollo y efectividad, teniendo en cuenta que la eficiencia de los flujos de información y/o conocimiento determinan los niveles de comunicación e interactividad. La siguiente gráfica (figura 1.1.) representa esquemáticamente las relaciones básicas entre los elementos del proceso de comunicación integrada multimedia, así como los recursos tecnológicos, metodológicos y de conocimiento necesarios para realizar la comunicación interactiva.

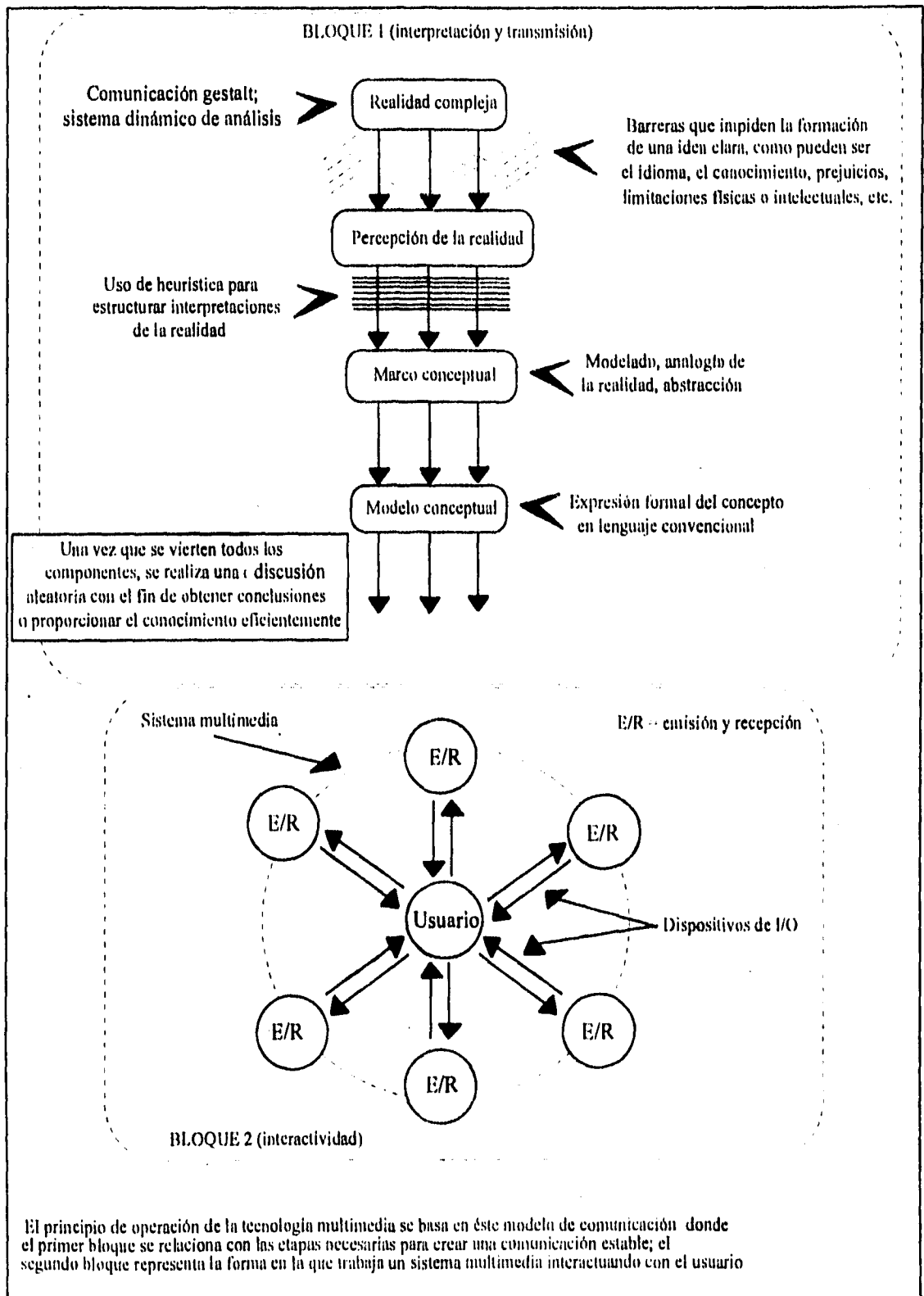


Figura 1.1. Modelo de comunicación Integrada, *multimedia*

Considerando las cantidades de información que una computadora es capaz de manipular, la tecnología multimedia promete ser una de las alternativas de transmisión de información, datos y conocimiento con mayor uso debido a la eficiencia sugerida por sus características intrínsecas. Aunado a lo anterior ésta tecnología incorpora principalmente información en formatos de lenguaje natural (audio, imágenes y video) así como también innovaciones tecnológicas en lo referente a hardware y software.

Dentro de los procesos enseñanza-aprendizaje convencionales, donde existe un instructor, la tecnología multimedia propone auxiliar y apoyar el desempeño de esta actividad, incrementando los niveles de asimilación, reduciendo los tiempos empleados para alcanzar los objetivos académicos, manipulando grandes bases de datos así como representaciones gráficas de problemas reales y casos prácticos. La multimedia es la tecnología que se encuentra definiendo una de las etapas importantes dentro de la historia de la tecnología de cómputo y de comunicación.

Paralelamente al proceso de asimilación y manejo de sistemas de cómputo, se fue desarrollando nueva tecnología que le permitió a más usuarios integrar las capacidades de las computadoras al desarrollo de actividades de propósito general; la multimedia como parte de esta nueva tecnología se encuentra albergada y en proceso de fortificación en muchos países, esta tecnología representa uno de los pilares en los procesos educativos y comerciales en muchas universidades y empresas del orbe, mismas que destinan gran parte de sus esfuerzos al desarrollo de tecnologías y métodos que incorporen las nuevas tecnologías en aplicaciones y sistemas modernos como es el caso de la multimedia.

1.1.3. CATEGORÍAS DE MULTIMEDIA

La multimedia se puede dividir en tres categorías:

- Material de diversión
- Material "de poder"
- Material creativo

El material de diversión consta de juegos, secuencias de animación y sonidos reales. El material "de poder" es software que permite hacer cosas a las computadoras que nunca habían hecho antes. Se tienen ejemplos de software "inteligente" como presentaciones multimedia sobre Beethoven, enciclopedias en CD-ROM, literatura y hasta revistas con sonidos e ilustraciones. El material creativo es software que permite al usuario hacer sus propios programas multimedia y presentaciones.

1.1.4. PLATAFORMAS PARA MULTIMEDIA

La multimedia puede ser desarrollada en muchas plataformas de cómputo. La computadora a utilizar depende del tipo de aplicaciones que se desee crear. Las IBM-PC's o compatibles, a pesar de las capacidades que se les pueden agregar por medio de tarjetas de audio y de procesamiento de video, son los equipos más limitados, mientras que las Macintosh y Silicon Graphics son plataformas ideales para hacer multimedia. La ventaja que presentan las IBM-PC's o compatibles es que representan la mayor capacidad instalada en cuanto a equipo de cómputo se refiere en nuestro país, aún cuando en años recientes el mercado de Macintosh y de estaciones de trabajo se ha incrementado.

El Sistema Tutorial sobre Microprocesadores utilizando Multimedia diseñado para el presente trabajo de tesis fue bautizado como MICRO, pues es común que los alumnos de la materia de Microprocesadores llamen a esta asignatura "Micros". Es por la gran capacidad instalada de IBM-PC's en México que se optó por desarrollar MICRO para la plataforma IBM-PC o compatible, sobre ambiente Windows 3.1, el cual es un ambiente gráfico que permite la implementación de aplicaciones multimedia.

Las plataformas para multimedia se pueden clasificar como:

- IBM-PC's y compatibles
- Apple Macintosh
- Commodore Amiga
- Estaciones de trabajo (como Silicon Graphics)
- Mainframes

1.1.5. CLASIFICACIÓN DEL EQUIPO IBM-PC PARA MULTIMEDIA

Las computadoras están diseñadas para trabajar con valores discretos en formato digital, lo que hace posible el procesamiento y almacenamiento masivo de información.

Una IBM-PC con capacidades multimedia puede integrar en una aplicación datos que le son naturales al ser humano, tales como imágenes fijas, animaciones, video, voces, música y sonidos diversos, siendo estos datos provenientes de diferentes medios: CD-ROM, compact disk musicales, videocasetes, cámaras de video, lasser disk (discos de video), imágenes fijas y fotografías digitalizadas por medio de *scanners*, así como de la misma computadora como es el caso de animaciones implementadas con softwares de animación.

El equipo de cómputo IBM-PC para multimedia se clasifica como:

- Equipo para desarrolladores de aplicaciones
- Equipo para usuarios de aplicaciones

En la actualidad, el equipo mínimo para desarrolladores de aplicaciones multimedia consiste en:

- CPU 80486 DX2/66 Mhz EISA (Extended Industry Standard Architecture)
- 20 Mb de RAM
- Resolución de monitor SVGA 1024x768 a 256 colores, local bus, con 1 Mb de RAM en tarjeta de video
- Disco duro de 1 Gb
- Tarjeta de audio de 16 bits con soporte para MIDI
- Unidad de CD-ROM
- Tarjeta de captura de video
- Tarjeta para exportar video
- Videocasetera y/o Lasser Disk indexados, es decir, que pueden ser controlados por la computadora.
- Cámara de video
- Scanner de cama a colores
- Quemador de CD-ROM (para el caso de distribución de aplicaciones en CD-ROM)

Actualmente el equipo mínimo para usuarios de aplicaciones multimedia desarrolladas para la plataforma IBM-PC's y compatibles consta actualmente de:

- CPU 80386 SX o superior
- 4 Mb de RAM
- Resolución de monitor VGA (640x480) a 256 colores (o mayor)
- Disco duro de 200 Mb
- Tarjeta de audio de 8 bits con soporte para MIDI
- Unidad de CD-ROM

1.1.6. LA INTERACTIVIDAD EN LA MULTIMEDIA

Uno de los elementos principales de gran importancia de la tecnología multimedia es la *interactividad*. Esto significa que el usuario tiene el control, es decir, lo que el usuario ve y escucha es el resultado de sus acciones. La interactividad requiere control. Las personas siempre se sienten a gusto cuando tienen el control y un ambiente de esa naturaleza las hace más receptivas a nueva información y situaciones. Por otro lado, la interactividad y el control hacen más agradable el trabajar con una aplicación. Un requisito muy importante de la multimedia es que se debe crear un ambiente de interactividad en el que el usuario tenga el control y al mismo tiempo se sienta cómodo. El impacto potencial de la multimedia es mayor que el de la televisión y de las publicaciones juntas, debido a que esta tecnología no sólo integra el poder audiovisual de la televisión con el poder de presentar texto, gráficos e imágenes fijas de las publicaciones, sino que además agrega la capacidad de interacción.

1.1.7. CREACIÓN DE APLICACIONES MULTIMEDIA

Las aplicaciones multimedia son creadas utilizando diversas metodologías por grupos multidisciplinarios de expertos, o por un solo desarrollador, dependiendo de la envergadura del proyecto. La realización de una aplicación multimedia generalmente requiere de las siguientes actividades:

- Producción y post-producción de audio
- Producción y post-producción de video
- Programación de computadoras
- Diseño artístico
- Escritura de guiones y scripts
- Procesamiento (digitalización y edición) de imágenes
- Diseño de interfaces de usuario
- Diseño de arquitecturas multimedia

Las capacidades de la tecnología multimedia pueden ser aprovechadas en muchas áreas del quehacer humano. Por ejemplo, para proveer soporte visual a un conferencista que realiza una presentación ante determinada audiencia. La facilidad que ofrece para crear presentaciones llamativas y excitantes y la flexibilidad proporcionada por la interactividad al momento de la presentación, harán que las conferencias y ponencias de cualquier índole sean uno de los mayores mercados de la multimedia.

En la actualidad, la mayoría de las conferencias se realizan con el apoyo de acetatos o diapositivas y a pesar de que, en muchos casos, la computadora se emplea ampliamente en su diseño y creación, estos medios presentan varias desventajas al compararlos con la multimedia. Esto se puede apreciar en la figura 1.2.

	ACETATOS	DIPOSITIVAS	MULTIMEDIA
Creación	Pueden ser diseñados en computadora. Requieren impresión en acetatos	Pueden ser diseñadas en computadora. Requieren procesamiento fotográfico	Diseño en computadora
Cambios	Requiere rediseño y reimpresión del acetato	Requiere rediseño y nuevo procesamiento fotográfico	Cambios directos en la computadora
Tiempo necesario para crear un cambio	Horas	Días	Minutos
Estilo de la presentación	No profesional	Profesional, pero con flexibilidad limitada	Profesional con alto grado de flexibilidad
Audio	No	No	Sí
Video	No	No	Sí

Figura 1.2. Comparación de acetatos y diapositivas con la tecnología multimedia.

1.2. ARQUITECTURA DE LAS APLICACIONES MULTIMEDIA

Las aplicaciones multimedia se construyen combinando actividades de presentación y de control, lo que tiene que ver con la interactividad. Las estructuras típicas de las aplicaciones multimedia son:

- Presentación lineal
- Motor controlado por datos
- Menú jerárquico
- Recuperación de información
- Hipermedia
- Simulación

1.2.1. PRESENTACIÓN LINEAL

Una presentación lineal es aquella que no hace otra cosa que presentar una secuencia de imágenes o un video. Sin embargo, una presentación útil necesita tener algún control. Las estructuras más comunes de este tipo alternan actividades de presentación y de control como se muestra en la figura 1.3.

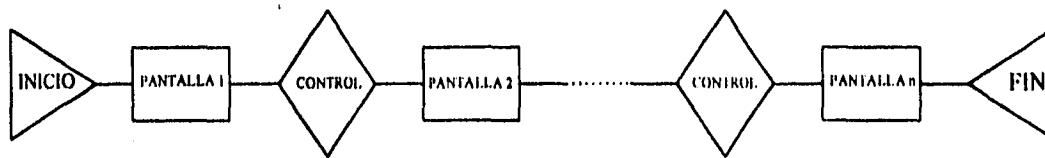


Figura 1.3. Estructura de una presentación lineal alternando actividades de presentación y de control sin posibilidad de elección.

La razón de las actividades de control es establecer las pausas necesarias en el flujo de la aplicación pero el usuario no tiene forma de elegir hacia donde ir. La forma de continuar la aplicación en cada actividad de control podría ser presionando una tecla, un botón en pantalla o mediante un límite de tiempo.

Una aplicación lineal más elaborada puede ofrecer algunas elecciones al usuario en cada puerto de control: la capacidad de ir hacia adelante o regresar, o bien, salir de la presentación. Esta estructura se muestra en la figura 1.4.

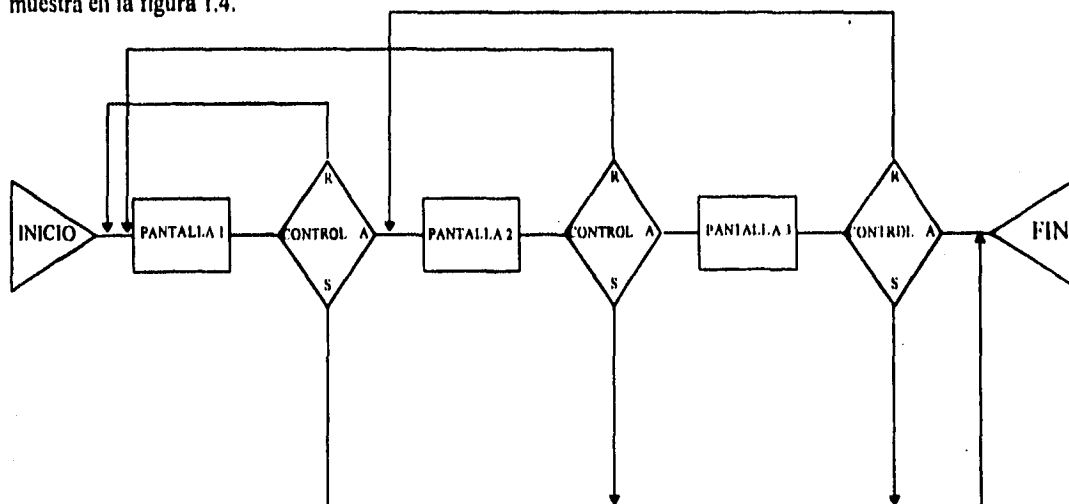


Figura 1.4. Estructura de una presentación lineal donde cada punto de control ofrece la opción de regresar a la pantalla anterior o salir de la presentación.

1.2.2. MOTOR CONTROLADO POR DATOS

El motor controlado por datos es una forma de simplificar las presentaciones lineales. Esta idea es hacer un programa, llamado *motor*, que lee instrucciones de un archivo de texto que especifica lo que debe suceder en cada punto de la presentación. El archivo de texto tiene una lista de los objetos a presentar en la secuencia deseada. Por lo tanto, se necesita una sola actividad de presentación para mostrar todo lo que se especifica en el archivo de texto y una sola actividad de control que espera los comandos del usuario para dar el siguiente paso. Por ejemplo, cuando se está mostrando una serie de imágenes en la presentación, la lista de secuencia sería exactamente una lista ordenada de los nombres de las imágenes a presentar.

Normalmente, el archivo de texto es creado de manera tal que presente un evento por línea. Si el usuario elige ir hacia adelante, simplemente avanza hacia la siguiente línea en el archivo de texto y repite la actividad de presentación; si elige salir simplemente termina la aplicación. La figura 1.5. muestra la estructura de una presentación lineal usando un motor controlado por datos.

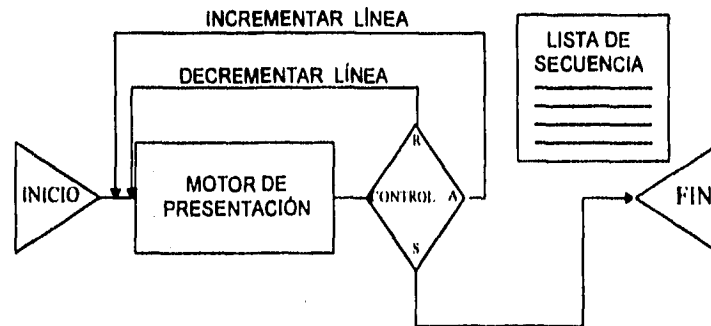


Figura 1.5. Estructura de una presentación lineal usando un motor controlado por datos.

La estructura de un motor controlado por datos es la misma sin importar la cantidad de objetos que sean presentados. Para cambiar el orden de una presentación, basta con modificar el archivo de texto de la misma.

La presentación utilizando esta estructura puede ser mucho más compleja que sólo una secuencia de imágenes. Agregando instrucciones en la línea de texto de cada objeto a ser presentado, se puede obtener control sobre la forma en que los objetos son presentados o el tipo de transición a usar entre un objeto y otro. De esta forma se pueden crear aplicaciones muy sofisticadas con solo modificar un archivo de texto. El archivo de texto se convierte, así, en un lenguaje de muy alto nivel para crear presentaciones.

Algunos sistemas de autoraje tienen motores controlados por datos integrados en sus módulos runtime (el módulo del software de autoraje que permite hacer ejecutable una aplicación diseñada con el software). Esto hace que las presentaciones sean extremadamente fáciles, pero limita sólo a los estilos de presentación integrados.

1.2.3. MENÚS JERÁRQUICOS

Un menú se construye usando una actividad de presentación para desplegar el menú y después una actividad de control para recibir la selección del usuario como se muestra en la figura 1.6. El resultado de la selección podría ir directamente a otra actividad de presentación o a otro menú de menor o mayor jerarquía.

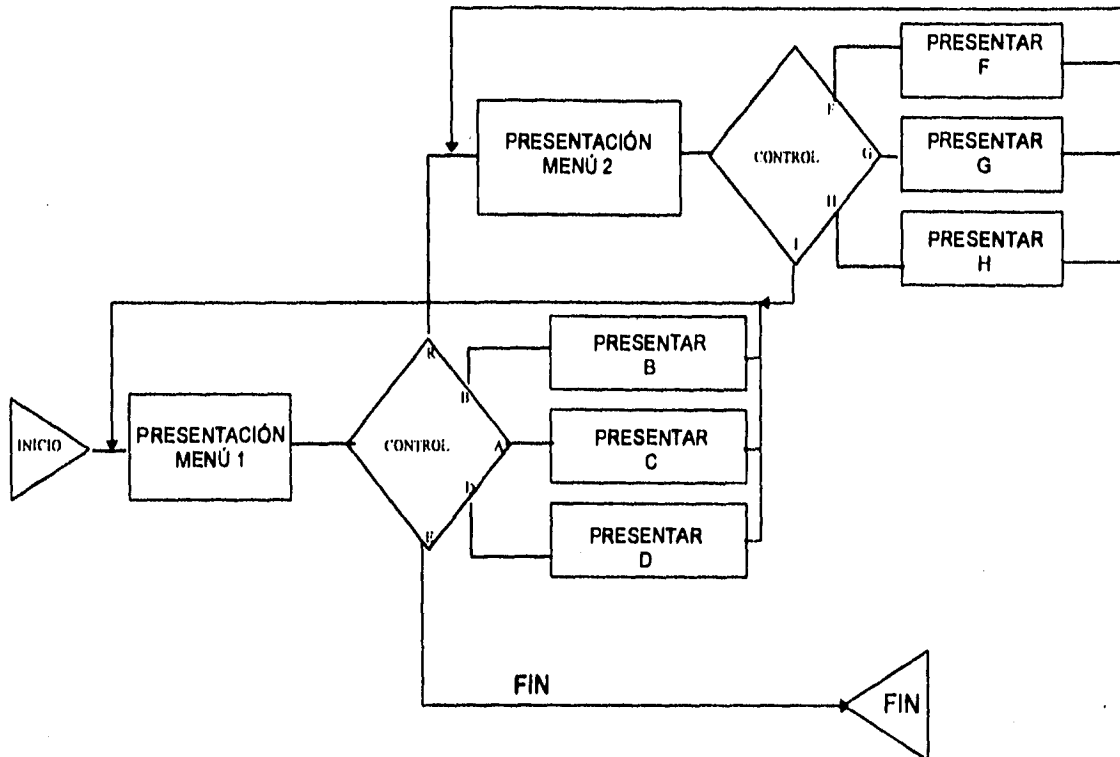


Figura 1.6. Estructura de una aplicación de menú jerárquico.

Una estructura de menús como ésta puede volverse complicada rápidamente, especialmente si alguna de las actividades posibles aparece en más de un menú. Como en el caso de las presentaciones lineales, esta estructura puede ser simplificada usando motores controlados por datos. Una forma de hacer esto es asociando un archivo de texto a cada menú describiendo los objetos del menú y los nombres de las presentaciones que deben ser desplegadas por éste. Se puede escribir un programa de motor controlado por datos para desplegar el menú y después pasar el nombre del objeto seleccionado a un simple programa de presentación también basado en motor -menú y presentación- y la compleja estructura de la aplicación es realmente guardada en un conjunto de archivos de texto.

1.2.4. RECUPERACIÓN DE INFORMACIÓN

Una aplicación de recuperación de información trabaja con una base de datos multimedia, que puede ser de texto con extensiones para audio, video o imágenes relacionadas a los objetos de texto. Por ejemplo, una enciclopedia multimedia tiene una base de datos de texto común para cada artículo, pero también incluye fotografías, sonidos e incluso segmentos de video.

Se proporcionan al usuario varias formas de entrar en la base de datos. Normalmente una de estas formas es búsqueda de palabras clave para temas en particular.

1.2.5. HIPERMEDIA

En una aplicación de recuperación de información, puede ser deseable permitir al usuario hacer selecciones directamente en los datos que han sido recuperados. Por ejemplo, si se ha recuperado una fotografía de la base de datos, el usuario sería capaz de marcar (hacer "click" con el botón izquierdo del mouse) sobre la fotografía y obtendría automáticamente más información referente al tema. A esto se llama hipermedia.

Un sistema hipermedia como éste requiere de una estructura de motor tanto para desplegar como para buscar. Ésta es una clase importante de aplicaciones multimedia y existen sistemas de autoraje especializados en hipermedia.

1.2.6. SIMULACIÓN

Este tipo de aplicaciones simula un sistema real, como la cabina de un camión o de un avión. Estas aplicaciones por lo general son muy específicas, por lo que no hay estructura genérica que se les aplique. La simulación de un panel de control eléctrico requerirá programación muy específica para permitir que las pantallas y los medidores respondan correctamente cuando el usuario mueve interruptores. Obviamente cada programa de simulación debe tener su propia estructura.

Muchas aplicaciones realmente serán mezclas de todos los tipos descritos. Por ejemplo, una aplicación puede empezar con una estructura de tipo menú, pero dependiendo de la opción elegida, puede seguir con presentaciones, recuperación de información, hipermedia, simulación o incluso todas ellas. Es importante que un sistema de autoraje sea lo suficientemente flexible para soportar esta mezcla de estructuras.

1.3. TIPOS DE APLICACIONES MULTIMEDIA

Actualmente las aplicaciones multimedia son consideradas como uno de los métodos de manejo de información más eficientes. Pueden clasificarse de acuerdo a sus objetivos y niveles de interactividad (navegación, interfaz de usuario y operación del sistema). Con base a sus objetivos, las aplicaciones multimedia pueden clasificarse como:

- **Aplicaciones corporativas.** Como puede ser el manejo de datos corporativos, procesos de diseño, automatización de oficinas, etc.
- **Aplicaciones educativas.** Estas aplicaciones están relacionadas con la instrucción, capacitación y adiestramiento. Las aplicaciones de este tipo tienen que ver con el proceso enseñanza-aprendizaje en cualquier tópico y nivel académico.
- **Aplicaciones de entretenimiento.** Juegos electrónicos, tours culturales digitales, sistemas de esparcimiento, sistemas digitales musicales, etc.
- **Aplicaciones para servicios de información.** Básicamente este tipo de aplicaciones se concreta a la transmisión de datos y la difusión de información. Los servicios de información pueden estar relacionados con cualquiera de los objetivos anteriores.

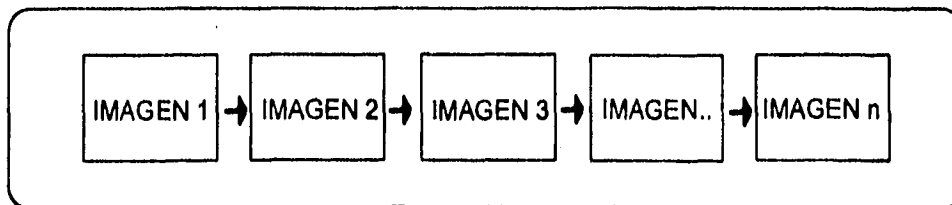
Las aplicaciones multimedia, cualquiera que sea su objetivo, también pueden clasificarse con base en su grado de interactividad, como:

- Presentadores de historias electrónicas
- Publicaciones electrónicas o presentaciones interactivas
- Kioscos multimedia

1.3.1. PRESENTADORES DE HISTORIAS ELECTRÓNICAS

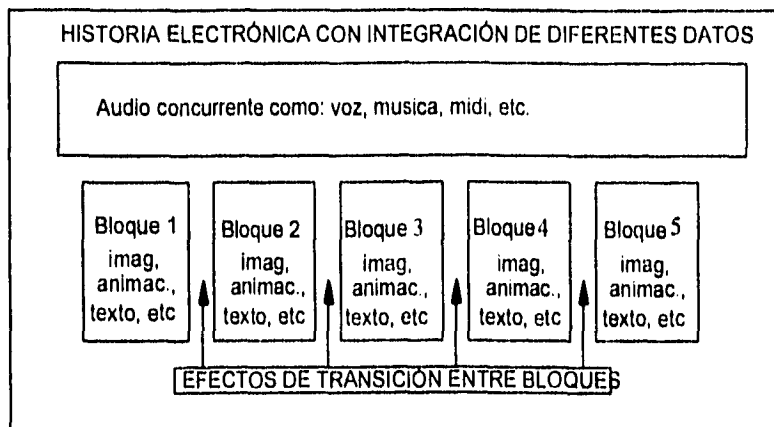
Son aplicaciones que tienen como fin mostrar una "historia" (presentación secuencial), para reforzar la presentación de un proyecto, exposición de productos, resultados de operación de una empresa, resultados de ventas de un departamento, etc. Este tipo de aplicaciones son consideradas como una herramienta de apoyo didáctico en los procesos de comunicación y transmisión de información. Las historias electrónicas son las aplicaciones con niveles de interactividad más bajos y el objetivo principal que persigue este tipo de aplicaciones es presentar una secuencia de pantallas conteniendo información. En estas presentaciones pueden incorporarse archivos de audio o animaciones sencillas en forma concurrente o secuencial mientras se realiza la presentación. En las siguientes figuras, 1.7. y 1.8., se presentan los esquemas de navegación de los presentadores de historias electrónicas.

HISTORIA ELECTRÓNICA BÁSICA



La historia electrónica más sencilla se basa en la presentación de imágenes o animaciones sin interactividad.

Figura 1.7. Historia electrónica básica.



La transición entre un bloque y otro puede efectuarse utilizando algunas técnicas de intercalado como disolución, intercalado, transposición, etc.

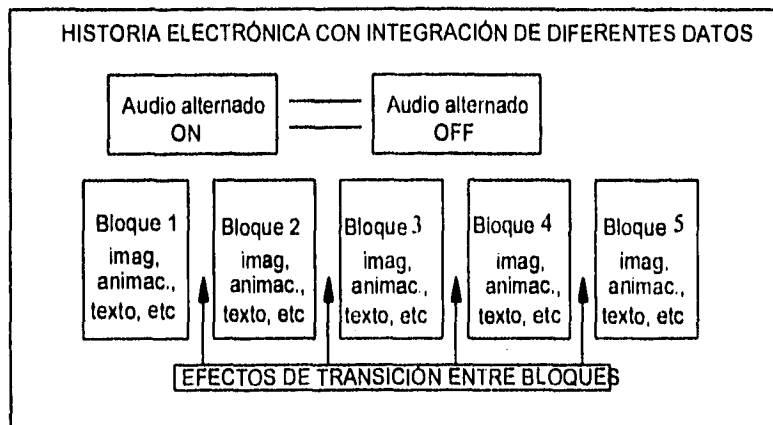


Figura 1.8. Historias electrónicas con integración de diferentes datos.

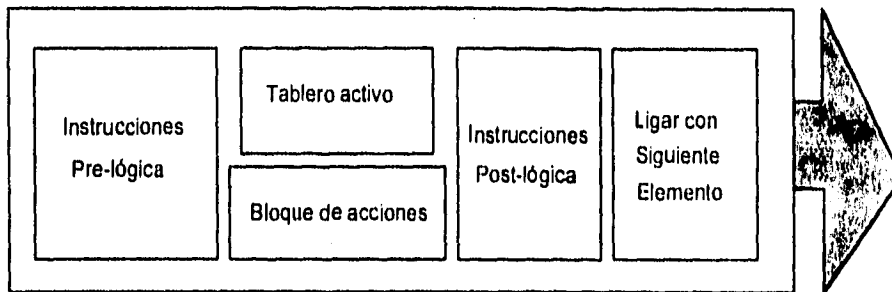
1.3.2. PUBLICACIONES ELECTRÓNICAS O PRESENTACIONES INTERACTIVAS

Al igual que las historias electrónicas, las presentaciones interactivas son aplicaciones que, por lo general, se almacenan en algún dispositivo transportable, ya sean disquetes o CD-ROM. El dispositivo en el cual se almacenan las publicaciones electrónicas depende del uso, formatos de información y objetivo de las mismas. El nivel de interactividad de las publicaciones electrónicas es mayor y permite al usuario seguir diferentes direcciones en el flujo de la información que éstas contienen. Las publicaciones electrónicas tienen niveles de interactividad basados en el concepto de "árbol" con diferentes niveles de "navegación".

La creación de una publicación electrónica o presentación interactiva está compuesta por cinco partes fundamentales que son:

1. La estructura pre-lógica, que no es más que la operación de una instrucción como *IF...THEN...ELSE* comúnmente utilizada en programación.
2. Los elementos que componen la presentación. Estos elementos están directamente relacionados a la instrucción pre-lógica.
3. Las características y funciones (llaves) para activar las instrucciones pre-lógicas con el uso del ratón o teclas clave.
4. Instrucciones post-lógicas, compuestas por una estructura similar a las pre-lógicas a diferencin que en esta parte se efectúan las operaciones posteriores a la terminación de la presentación interactiva o si es el caso a la ruptura de la misma (abortar la presentación).
5. Ramificaciones y ligas entre todos y cada uno de los elementos que componen la presentación, mismos que se recorrerán de acuerdo con la salida (instrucción) activada en el punto en el que se encuentre la presentación.

En las siguientes 3 figuras se aprecia el flujo de información en este tipo de aplicaciones multimedia.



Elemento de las presentaciones interactivas

Figura 1.9. Elemento de las publicaciones electrónicas o presentaciones interactivas.

Ejemplo de un bloque de una presentación interactiva

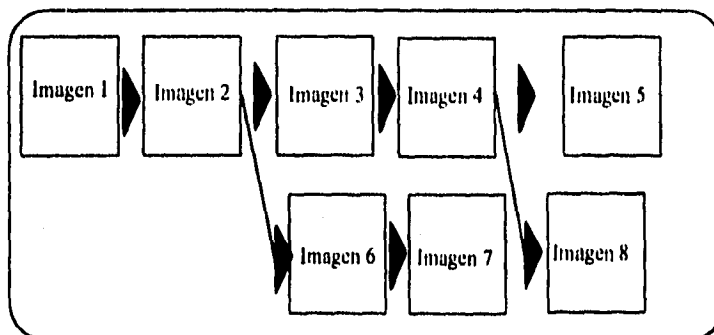
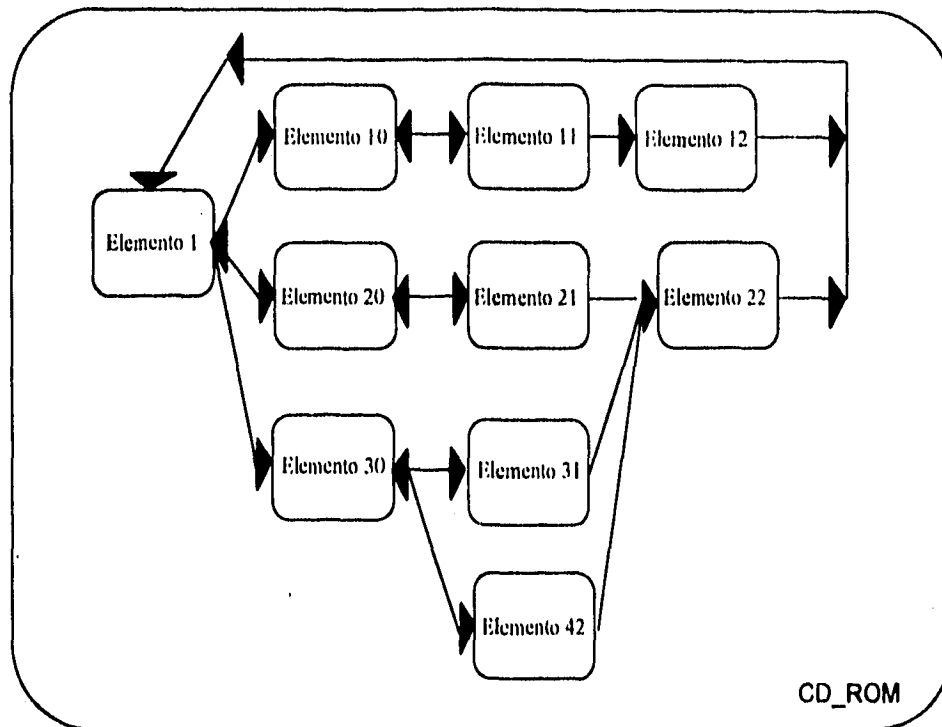


Figura 1.10. Bloque de una presentación interactiva.



Ejemplo de la navegación en las publicaciones interactivas

Figura 1.11. Ilustración de la navegación de las publicaciones electrónicas.

Para desarrollar presentaciones interactivas se pueden utilizar lenguajes de tercera y cuarta generación como los "authoring software". Una de las características principales de los historias electrónicas y la presentaciones interactivas es la transportabilidad.

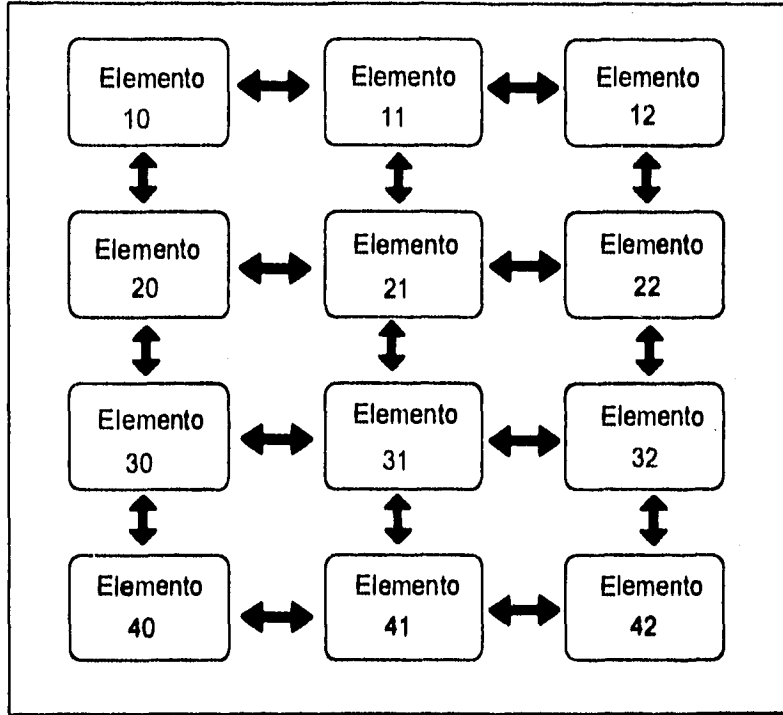
1.3.3. KIOSCOS MULTIMEDIA

Los kioscos multimedia son desarrollados bajo los mismos principios que una presentación interactiva, solo que los kioscos están pensados para no ser aplicaciones transportables debido a la cantidad de información que incluyen (más de 680 Mbytes, que es la capacidad de almacenamiento de un CD-ROM); Existen dos tipos de kioscos:

- **Espacios públicos.** Los espacios públicos tienen la finalidad de transmitir el conocimiento o la información con carácter general. Las aplicaciones de este tipo manejan, en la mayoría de los casos, grandes volúmenes de información
- **Estaciones de conocimiento.** Las estaciones de conocimiento están destinadas para instruir al usuario en un tema en particular. Algunos casos son sistemas de capacitación industrial, sistemas de capacitación académica, sistemas de entrenamiento en el uso y manejo de equipos, etc.

Los kioscos multimedia se desarrollan también utilizando authoring software o lenguajes de tercera generación. Un kiosco multimedia tiene control absoluto sobre el flujo de información. La navegación dentro del kiosco se realiza dentro de toda la información que posee.

La figura 1.12. ilustra un caso de estructura de kiosco multimedia.



Kiosco multimedia: espacio público

Figura 1.12. Estructura de un kiosco multimedia.

Basándonos en la clasificación de aplicaciones multimedia descrita anteriormente, se puede decir que el sistema MICRO se puede clasificar, dependiendo el uso que se le de, como una publicación electrónica o presentación interactiva si son distribuidas copias en disquetes para los usuarios en particular, o como una estación de conocimiento si es que es instalado en una computadora destinada al uso de varios estudiantes.

1.4. LA MULTIMEDIA EN LA EDUCACIÓN

Uno de los campos donde la multimedia tiene un gran futuro y que es de especial interés en este trabajo es el de la educación. El principio básico de la educación o capacitación apoyada en multimedia es que el estudiante capta y comprende los conceptos presentados de forma más eficiente que cuando solo lee o escucha acerca de ellos. El aprendizaje es más efectivo cuando el estudiante realmente interactúa con la información; en un ambiente multimedia aprende al ver, escuchar y hacer trabajando por sí mismo y a su propia velocidad.

La capacitación interactiva por computadora no es nada nuevo ya que ha sido empleada ampliamente por más de una década bajo el nombre de *entrenamiento basado en computadora* (CBT o *computer based training*). De hecho, multimedia nació del CBT. Los primeros sistemas de entrenamiento por computadora que eran capaces de reproducir audio y/o video requerían de un gran número de equipos analógicos conectados a la computadora digital. Los sistemas multimedia de la actualidad son más integrados, compactos y baratos debido a que ahora el audio y el video se reproducen utilizando dispositivos digitales y toda la información es almacenada, procesada y controlada por una computadora.

Una clase especial de capacitación en la que multimedia tiene importantes ventajas es la ayuda en línea o tutoriales, que se pueden encontrar en la mayoría de las aplicaciones de la actualidad. Estos sistemas están basados en texto casi en un 100%, no obstante el uso de *hipertexto* se está haciendo más común. En un sistema con hipertexto, el usuario puede señalar una palabra o una frase y saltar inmediatamente a información más detallada acerca del tema seleccionado. Sin embargo, los sistemas de ayuda se vuelven más agradables y versátiles cuando se usa información en forma de audio, video y/o animaciones de la misma forma que se usa el hipertexto (a esto se llama *hipermedia*).

No obstante las bases de datos de los sistemas de ayuda en línea pueden contener toda la información que el usuario necesita, de difícil presentarla de forma amigable utilizando solo texto, especialmente a usuarios que no están familiarizados con el uso de computadoras. Con multimedia, el texto puede ser reemplazado por voz (audio) en su mayor parte y, donde sea de utilidad, la presentación también puede incluir sonidos (audio), video o animaciones para explicar conceptos confusos. El resultado es que el usuario percibe que la aplicación entera es más amigable y es más fácil de aprender y operar con el sistema al reducirse la carga cognitiva (trabajo necesario para asimilar un tema). Por ejemplo, en el Laboratorio de Inteligencia Artificial y Multimedia del Instituto de Ingeniería (LIAMMII) de la Universidad Nacional Autónoma de México (UNAM) en colaboración con el Departamento de Fisiología de la Facultad de Medicina de la UNAM se desarrolló un sistema educativo multimedia sobre el ciclo cardíaco humano; los estudiantes que hicieron uso de este sistema requirieron entre 25% y 30% menos tiempo que los estudiantes que usaron métodos tradicionales para aprender el tema.

El uso de la tecnología de vanguardia en el proceso educativo brinda beneficios significativos a estudiantes y maestros. Uno de los elementos más importantes dentro de este nuevo esquema educativo es la computadora. Algunas de las ventajas del uso de la computadora como apoyo a la educación son la capacidad de interacción (acción → reacción → interacción) que ésta ofrece a los educandos. La computadora es una poderosa herramienta para la enseñanza por su versatilidad, misma que se ve incrementada al integrarse a ésta la tecnología multimedia.

La forma en la que aprende el ser humano está basada en los sentidos; por ello es importante presentar información y desarrollar conceptos utilizando técnicas que permitan el aprendizaje visual, auditivo y táctico. Por otro lado, es importante resaltar el hecho de que, para que un estudiante comprenda una idea nueva, debe primero relacionarla con sus propias experiencias. La compañía IBM ha realizado estudios a fin de determinar la cantidad de información que puede retener el ser humano a través de sus sentidos. Dichos estudios muestran los siguientes resultados: se retiene a corto plazo alrededor del 10% de lo que se ve, 20% de lo que se escucha, 40% de lo que se ve y se escucha y 75% de lo que se ve, se escucha y se lleva a la práctica.

Por otro lado, se ha desarrollado una serie de investigaciones a fin de cuantificar la efectividad de la tecnología multimedia en el ámbito educativo obteniendo el siguiente resultado: los estudiantes aprenden en 50% menos tiempo y retienen 25% más información.

De lo anterior se puede concluir que el proceso de aprendizaje se beneficia si la información que se desea consultar se presenta en forma tal que impacte los sentidos y exista a su vez posibilidad de interactuar con ella. Basta recordar que ver, escuchar y realizar es justamente lo relevante de la tecnología multimedia.

Las aplicaciones desarrolladas con esta tecnología muestran al usuario una interfaz de usuario más amigable (en el capítulo 2 se explicará más sobre la interfaz de usuario), así como un manejo más fácil y llamativo de la información; estos factores hacen de la multimedia un punto referencia para el mundo de la enseñanza, ya que los diferentes elementos empleados en el desarrollo de cursos y presentaciones multimedia atrapan la atención del estudiante involucrando sus sentidos.

Desde un aspecto educativo, las ventajas ofrecidas por la tecnología multimedia se pueden englobar en los siguientes puntos:

- La multimedia permite a los estudiantes *navegar* entre imágenes que ilustren una idea clave, escuchar y nombrar los conceptos relacionados con dicha idea. Esta dinámica da a las lecciones otra dimensión ya que promueve una amplia participación del estudiante, beneficiando su proceso de aprendizaje.
- Con multimedia el estudiante está listo para actuar, interactuando con la computadora al contestar cuestionarios y navegando en la información, creando su propio patrón de conocimientos. Una vez que los estudiantes han experimentado una lección multimedia, pasan a la fase del cuestionamiento en donde hacen preguntas relacionadas con los conceptos que vieron, desarrollando así su pensamiento crítico y pasando de la comprensión y aplicación de un tema a su análisis.
- Una clase en donde se usan aplicaciones multimedia es activa y estimulante por naturaleza; los estudiantes y los profesores no pueden adquirir un papel pasivo, ya que están interactuando entre ellos todo el tiempo.
- Cada persona tiene su propia forma de aprender; hay personas cuya capacidad de aprendizaje está orientada a material escrito. A otras les puede ser más fácil aprender utilizando medios visuales o auditivos. Algunas personas aprenden más rápido que otras. La tecnología multimedia soporta estas diferentes necesidades de aprendizaje, siendo posible de esta manera elevar la capacidad de aprendizaje de los alumnos al individualizar el proceso de enseñanza-aprendizaje.

Gracias a esta tecnología los estudiantes tienen la oportunidad de aprender haciendo. Por otro lado, el que los alumnos utilicen multimedia no sólo como usuarios sino que experimenten su poder como autores y desarrolladores de aplicaciones, les abrirá nuevos horizontes, los cuales les permitirán desarrollar su capacidad creativa, ya que al digitalizar imágenes, producir un video, escribir un texto y diseñar la integración de todos los elementos, se convierten en escritores, desarrolladores de artículos, diseñadores gráficos, guionistas, editores, correctores de estilo, productores y lo más importante: gente creativa capaz de investigar y de abstraer lo más importante de la información que se consulte.

1.5. LOS MEDIOS EN LAS COMUNICACIONES

Algo aconsejable para entender que es especial sobre los proyectos de laboratorios multimedia es examinar lo que está pasando en el mercado de las comunicaciones.

Cada año hay nuevas formas o medios de comunicación. Se tienen innovaciones como el MMDS (Sistema de distribución multipunto multicanal llamado también "TV por cable sin alambre") o el CD-I (Compact disk interactivo). Hay cambios revolucionarios en el servicio telefónico y las capacidades computacionales.

En su libro: "Technologies of freedom" la autora Ithiel de Sola Pool's dice que la primera era de las comunicaciones fue el discurso; la segunda fue la escritura; la tercera, la imprenta y otras formas de hacer muchas copias tales como fonógrafos y fotografía; y ahora, la cuarta era: todos los medios se están volviendo electrónicos.

La mayoría de los medios electrónicos se están volviendo digitales. Con la digitalización todos los medios se pueden traducir en otros. Una película, llamada telefónica, carta o artículo de revista, pueden ser enviados digitalmente vía línea telefónica, cable coaxial, fibra óptica, microondas, satélite, transmisión televisiva o medios físicos de almacenamiento como la cinta o el disco.

El medio digital está libre de ruido y puede corregir errores. Se considera que en el futuro todas las transmisiones serán digitales.

1.5.1. AUDIO DIGITAL

Los discos compactos (compact disk o CD) incorporan sonido digital de alta fidelidad. Son el producto electrónico de consumo más exitoso en la historia. Las ventas de CD's en Estados Unidos fueron el triple en 1985 que en 1984 y en 1986 triplicaron a las de 1985.

El advenimiento de la cinta digital de audio (DAT por sus siglas en inglés) es uno de los más extraños en la historia de la electrónica. En todo el mundo 84 compañías iban a introducir reproductores DAT y cintas DAT en el verano de 1986 y luego, universalmente, no lo hicieron.

El negocio de la música grabada se molestó, porque pareció que usuarios normales serían capaces de hacer copias tan buenas como las originales. Aún así, los DAT's se están introduciendo en la industria musical.

1.5.2. MEDIOS DENSOS DE INFORMACIÓN

Los reproductores CD son más baratos que los DAT y tienen acceso aleatorio rápido, lo que no se encuentra en las cintas. Los CD's son capaces de 630 megabytes de almacenamiento de datos de acceso aleatorio, lo que los hace apetecibles para los usuarios de computadora, que se sienten limitados con los floppy disk's de 1.4 megabytes de memoria.

El CD-ROM es un compact disk usado como memoria de solo lectura. "Solo lectura" significa que no se puede "escribir" nada en el CD-ROM.

Un CD puede albergar 250,000 páginas de texto, el equivalente de 500 libros. El publicante Barry Richman concluye que a largo plazo los medios densos reemplazarán al papel.

La primera publicación popular de CD-ROM entró al mercado en 1986 y fue la "Enciclopedia Académica Americana" de Grolier's; sus veinte volúmenes e índice se llevaron solo la quinta parte de un disco. Su costo fue de US \$200.00.

El CD-ROM es un medio interactivo multimedia. El compact disk interactivo (CD-I por sus siglas en inglés) puede tener en un disco mil cuadros de video, dos mil diagramas, seis horas de sonido de alta calidad y diez mil páginas de texto junto con un programa que hace a esto trabajar como un todo orgánico e intensamente interactivo con el usuario. La limitación del formato CD-I es que no puede hacer video de movimiento continuo ("Full motion video"). En 1987 Philips anunció un formato CD de video llamado "Video digital interactivo" (DVI por sus siglas en inglés) capaz de tener una hora de video de movimiento continuo en un compact disk, utilizando técnicas de compresión de señales.

1.5.3. CORREO ELECTRÓNICO (E-MAIL)

Las computadoras personales pueden conectarse con otras computadoras utilizando distintos medios: transmisión por satélite, fibra óptica, par trenzado, etc. o vía telefónica usando dispositivos como el módem. Últimamente se considera que una computadora personal sin una línea telefónica conectada a ella está, en cierto sentido, incompleta. Alrededor de la cuarta parte de los 10 millones de computadoras en Estados Unidos se han conectado al sistema telefónico ligando al correo electrónico teleconferencias por computadora y bases de datos. Más aún, si hablamos de las oficinas, el correo electrónico se está convirtiendo en una manera de vida en algunos negocios.

Se puede decir que el correo electrónico es una forma de conversación y se estima que de 250 a 1000 millones de mensajes por año en Estados Unidos son enviados por correo electrónico.

Existe también el llamado BBS (*Bulletin Board System*), donde varias computadoras se conectan a una computadora central o "Host" que multiplexa múltiples llamadas y permite acceso a mucha información.

Otro ejemplo es el llamado ISDN (*Integrated Services Digital Network* o red digital de servicios integrados), utilizando fibra óptica y del que se esperan muchos servicios innovadores como transmisión señales de video en tiempo real y procesamiento digital de sonido e imagen.

Capítulo 2

Sistemas tutoriales en computadora

2. SISTEMAS TUTORIALES EN COMPUTADORA

2.1. TUTORIALES EN COMPUTADORA

Desde la aparición de la computadora, cuando se tuvo acceso a terminales y con la aparición de las computadoras personales, estos programas fueron cobrando importancia en la implementación de sistemas educativos. Los tutoriales son programas de computadora educativos que sirven para hacer más fácil el aprendizaje de múltiples temas, que incluyen varios aspectos como la educación en muchos de sus niveles, ya sea cursos de determinadas materias, programas que ilustran el uso de ciertos implementos como pueden ser a su vez otros programas, o que ilustran didácticamente ciertos procedimientos (de control, industriales, de supervisión, etc.).

Los tutoriales en computadora son herramientas muy útiles que facilitan la educación, y cuya gama de acción es bastante amplia. Muchos programas de computación incluyen tutoriales que ilustran el uso del programa. En empresas son comunes los tutoriales que muestran distintos procedimientos. En escuelas y universidades es cada vez más frecuente encontrar tutoriales en computadora que facilitan la educación, pues sirven en la impartición de ciertos temas y asignaturas.

La tecnología, en constante avance y en especial la tecnología multimedios han potenciado la versatilidad de los tutoriales en computadora. Los multimedios o la multimedia como es más común nombrarla, multiplican el potencial educativo de los tutoriales en computadora, dadas las características de esta tecnología, como son el uso de distintos tipos de datos como texto, gráficos, animaciones, video y audio.

2.2. LA COMUNICACIÓN Y EL USO DE LA TECNOLOGÍA

La transferencia efectiva de información es el objetivo principal de todo mecanismo de comunicación, mismos que se dividen en cuatro etapas principales:

- a) Elaboración de la información
- b) Confección de la información
- c) Distribución y transmisión
- d) Recepción

La transmisión del conocimiento o información, se determina por los mecanismos que utiliza el proceso de comunicación; estos mecanismos están directamente relacionados con la simultaneidad entre la transmisión y la recepción, es decir, el nivel de acceso que el usuario (receptor) tenga con la información que recibe en el mismo momento de la difusión (transmisión).

La transmisión de la información por tanto, se puede dividir en dos rubros:

- I. La transmisión continua.
- II. La transmisión cuántica.

Se considera como transmisión continua desde las pláticas, conferencias, seminarios, cursos, etc. hasta el uso de mecanismos que permitan enviar y recibir información simultáneamente. Los mecanismos de información continua permiten la interacción del usuario con la información que recibe, que, en el caso de utilizar máquinas específicas se conoce como: servicios de información en línea, donde la interactividad con la información permite al usuario "navegar" y "cortar" la información de acuerdo a sus requerimientos. Particularmente las aplicaciones multimedios están encaminadas a cumplir con este tipo de mecanismos de comunicación, permitiéndole al usuario interactuar por completo con la aplicación, de forma tal que se establezca un diálogo "dinámico" entre la computadora (aplicación multimedios) y el usuario.

La transmisión cuántica está determinada por la cantidad de información que se transmite en un momento dado; en este caso, la información no puede ser transmitida continuamente, por lo que la información tiene que ser enviada en *quantums* o paquetes.

La transmisión de información dentro de la vida moderna depende en gran medida de los mecanismos utilizados, así como de las técnicas empleadas para cumplir este fin; uno de los medios para manejo de información es la computadora, misma que ha modificado casi todas las actividades del hombre para manejar información.

La computadora en la actualidad, es uno de los pilares más importantes en los procesos de comunicación masiva o de los que requieren del manejo de grandes volúmenes de información. Las técnicas desarrolladas en torno al incremento de la eficiencia para manejar y transmitir información utilizando computadoras han sustentado sus avances tecnológicos en la simultaneidad que los usuarios de la información puedan tener con el sistema que maneja dicha información. Los sistemas computarizados han pasado de ser mecanismos que almacenan y procesan información, a ser mecanismos que interactúan con el usuario de la información en ambientes cada vez más agradables y eficientes. Así mismo la generación y difusión de conocimientos se encuentra actualmente en el proceso de transición al uso de sistemas automáticos, computarizados y digitalizados, que le permitan elevar los niveles de desarrollo y eficiencia en el aprendizaje.

2.3. COMUNICACIÓN DEL CONOCIMIENTO

Wenger (1952) da una definición muy útil para la *Comunicación del conocimiento*: es la habilidad de ocasionar y/o apoyar la adquisición del conocimiento de alguien por alguien más, por medio de una serie restringida de operaciones de comunicación. Ejemplos de esta serie restringida de operaciones de comunicación son: lenguaje, gráficas en pantalla, series de ejercicios. Esta noción de *Comunicación del conocimiento* es muy amplia pues incluye muchas formas indirectas de que alguien tenga conocimiento adquirido gracias a alguien más. Con esta definición, se puede ver que todos los problemas relacionados con la elaboración de tutoriales son problemas envueltos en la comunicación del conocimiento.

Los tutoriales en computadora son un tipo de sistemas de comunicación de conocimiento.

El proceso de comunicación está organizado alrededor de *aprender*: esto es, alrededor de la integración exitosa de nuevo material y nuevas experiencias en el cuerpo de conocimiento del estudiante. Esto requiere piezas de información que son usadas específicamente para propósitos pedagógicos. Algunas piezas del dominio del conocimiento pedagógicamente orientadas son exposiciones razonadas para explicaciones en términos de metas y causas, así como relaciones conceptuales o taxonómicas entre piezas de conocimiento que faciliten el uso de analogías y abstracciones.

Las decisiones pedagógicas involucradas en el diseño, elaboración, implementación y desempeño de tutoriales en computadora reflejan una gran cantidad de conocimiento sobre cómo comunicar conocimiento.

El orden y la manera en que los temas son tratados pueden producir experiencias de aprendizaje muy distintas. En sistemas tutoriales, es algunas veces más efectivo dejar al estudiante buscar por un rato que estar interrumpiéndolo siempre. Por otro lado, cuando el estudiante se queda completamente por su cuenta, es común que se pierda, bloquee o atore. El aprendizaje efectivo y exitoso depende de una amplia variedad de factores y el sistema tutorial más esmerado en su información puede ser inútil si destruye la motivación personal del estudiante o su sentido de descubrimiento. La pedagogía es un arte que requiere una gran versatilidad: es probablemente más compleja y difícil que la mayoría de los temas a los que está aplicada en la enseñanza.

Las decisiones pedagógicas son tomadas respectivamente por el sistema tutorial y por el estudiante en el contexto de un ambiente que determina los grados de control sobre la actividad y sobre la interacción. Algunos sistemas *monitorean* la actividad del estudiante muy de cerca, adaptando sus acciones a las respuestas del estudiante pero nunca cediendo el control. En diálogos de *iniciativa mezclada*, el control es compartido por el sistema y el estudiante mientras intercambian preguntas y respuestas. El sistema tutorial debe ser capaz de responder al estudiante, pero puede también hacer preguntas cuyas respuestas ayuden a entender qué está tratando de hacer el estudiante, o qué sabe. En el *aprendizaje de descubrimiento guiado*, el estudiante tiene un total control de la actividad.

2.4. LA INTERFAZ DE USUARIO

La *interfaz de usuario* es en sí la forma de comunicación. Esta interfaz se encarga de la forma final de las acciones didácticas. Hablando más generalmente, la interfaz de usuario procesa el flujo de comunicación de entrada y de salida. En ambas direcciones traduce entre la representación interna del sistema y un *lenguaje de interfaz* entendible para el estudiante.

La *interfaz de usuario* es la forma en que la información será presentada al usuario y comprende la operatividad con que este último se relacionará con el programa. Hay interfaces de computadora que en su interacción con el usuario requieren de presionar distintas teclas. Se tienen también los llamados menús de comandos, que consisten en una barra en la parte superior de la pantalla donde aparecen distintos comandos, los cuales al ser seleccionados con el mouse desencadenan las acciones pertinentes. Hay otras, más amables, que se desarrollan en un ambiente gráfico, las cuales permiten una interacción en base a acciones del mouse, es decir, aparece un cursor en forma de flecha o mano y moviendo el mouse, éste se mueve a su vez; presionando el botón izquierdo del mouse (hacer "click") sobre cierta área de toque, se logran acciones. Tal es el caso de la interfaz de usuario de *Windows*, aplicación de *Microsoft* en la que se simulan botones en la pantalla. Al hacer "click" con el mouse sobre éstos, lo que es presionar en general el botón izquierdo del mismo cuando el cursor se encuentra sobre el botón de la pantalla, se desencadenan las acciones correspondientes. Los primeros en implementar interfaces de usuario de este tipo fueron *Apple Macintosh*, empresa que ha dictado los estándares necesarios para crear aplicaciones en computadoras de su tipo. Otro tipo de interfaz de usuario muy amable es el llamado *Touch Screen*, donde el usuario toca con el dedo ciertas áreas desplegadas en la pantalla del monitor, lo que desencadena las acciones deseadas.

La importancia práctica de la interfaz de usuario puede ser entendida en dos niveles. Primero, en cuanto a la forma final en que el sistema presenta un tema, la interfaz de usuario puede hacer esta presentación más o menos entendible. Dado que la interfaz de usuario es también la forma final en que un sistema se presenta a sí mismo, las cualidades como facilidad de uso y ser atractiva son cruciales para la aceptación del sistema por parte del estudiante. Segundo, el progreso en la tecnología provee herramientas cada vez más sofisticadas cuyo poder comunicativo puede ser útil para llevar a cabo el diseño de todo el sistema.

En el entorno tecnológico de la computación podemos encontrar como principales objetivos de la interfaz de usuario los siguientes:

- Realizar la interacción con los sistemas computarizados
- Facilitar la comunicación entre el usuario y el sistema
- Incrementar la eficiencia perceptiva del usuario

Todos los productos con los que tenemos contacto en la vida diaria tienen su propia interfaz de usuario, algunas más eficientes que otras. La mayoría de las interfaces de usuario permiten acumular un conocimiento básico aplicable a productos de similares patrones de operación. La primera reacción que tenemos cuando entramos en contacto con un sistema nunca antes operado, consiste en buscar dentro de nuestro conocimiento las relaciones de operación y los mecanismos de funcionamiento de sistemas similares, que nos permitan interactuar o asimilar más rápido y eficientemente la interfaz del nuevo sistema. La normatividad y estandarización de las interfaces de usuario ofrecen como principal ventaja la masificación y asimilación de sistemas nuevos dentro del mercado potencial de usuarios en un tiempo muy reducido. Un ejemplo concreto es el desarrollo de las habilidades necesarias para conducir un automóvil hasta aprender una vez para adquirir los conocimientos necesarios y utilizar cualquier otro automóvil.

La mayoría de las interfaces de usuario se evalúan efectuando una medición estadística del tiempo empleado por el usuario para crear las habilidades necesarias de operación del sistema con familiaridad, donde los niveles más complejos de asimilación están relacionados con el tiempo empleado en el aprendizaje del funcionamiento de dicho sistema. Los resultados obtenidos del estudio de la interfaz de usuario han

demostrado la importancia de contar con principios de diseño, dentro del proceso de creación de interfaces, basando éstas en la intuición humana (interfaces intuitivas), debido a que la primera reacción de los usuarios ante una interfaz muy complicada o tediosa es abandonar el sistema, por lo que la familiaridad con la que un usuario interactúe con un sistema es vital para el desarrollo del mismo.

Por lo anterior se reafirma la importancia de contar con una metodología para el diseño de interfaces de usuario así como con elementos y técnicas de evaluación para dichas interfaces.

Las interfaces para aplicaciones multimedia como en nuestro caso tutoriales multimedia en computadora, deben proporcionar al usuario la posibilidad de seleccionar una opción, aceptar ésta y permitirle regresar a la selección anterior.

Por otro lado, para la creación de software educativo multimedia como es el caso de tutoriales multimedia en computadora, es importante basar el trabajo de diseño en una metodología consistente que permita hacer eficiente el proceso de comunicación y transmisión de conocimiento. "La eficiencia en la transmisión del conocimiento o información se determina por los mecanismos que utiliza el proceso de comunicación, mismos que están directamente relacionados con la simultaneidad entre la transmisión y la recepción." (Lara y Guerrero, 1994). Así mismo, debido a las bondades que ofrece la tecnología multimedia para la representación de información o conocimiento a través de diferentes tipos de datos, como texto, audio, imágenes fijas, animaciones y video, el proceso enseñanza-aprendizaje auxiliado por computadora permite incrementar considerablemente la eficiencia de asimilación. "El proceso del aprendizaje consta de varias etapas, cuyo propósito es el establecimiento de nuevos estados o capacidades internas y destrezas motoras por medio de la comunicación del conocimiento" (Kemper, Lara, Guerrero y Monedero, 1994).

En una aplicación para el público en general no se puede suponer que el usuario tendrá experiencia en interfaces de usuario para computadora y por esto toda la información sobre qué hacer en el siguiente paso debe estar siempre disponible, ya sea en forma de texto o audio.

En el caso de las aplicaciones orientadas a usuarios que tienen contacto con computadoras, ellos están familiarizados con los estándares de las interfaces de usuario y saben cómo trabajar con un ambiente gráfico. Aún así, debido a la gran flexibilidad de la multimedia en aplicaciones como los tutoriales para computadora, es necesario poner mucho cuidado en el diseño de la interfaz de usuario ya sea que ésta puede salirse de los estándares fácilmente.

La clave del diseño de una buena interfaz de usuario es tener comprensión completa de las necesidades del público para el que se está trabajando. Esto incluye una apreciación completa de lo que el usuario hará con la interfaz y con la aplicación. El diseñador debe hacerse las siguientes preguntas: ¿Cómo trabajará con la aplicación? ¿Qué pasos seguirá para obtener los resultados que espera de la aplicación?

Una buena manera de contestar a estas preguntas es estudiando al usuario en acción. Tomar el punto de vista del usuario. A este proceso se le llama *Análisis de las tareas del usuario*. Conforme más compleja es la aplicación a desarrollar, más importante se vuelve este análisis.

La interfaz de usuario de una aplicación como un tutorial en computadora es tan importante como el contenido de la aplicación. Sin una buena interfaz, el usuario puede no ser capaz de interactuar con el tutorial.

2.5. DECIS

A manera de ejemplo se presentará de manera breve el sistema educativo DECIS. DECIS es un sistema educativo para la enseñanza de *toma de decisiones*, elaborado por Joel Silva en el Laboratorio de Inteligencia Artificial y Multimedia del Instituto de Ingeniería (LIAMMI) de la Universidad Nacional Autónoma de México (UNAM). Este sistema está orientado a estudiantes de ingeniería, matemáticas, actuaría, estadística o carreras afines que hayan cursado materias, o tengan los conocimientos equivalentes de cálculo diferencial e integral, probabilidad y estadística, álgebra lineal e investigación de operaciones.

La interfaz de usuario de DECIS es del tipo gráfico, orientada a ambiente *Windows*. Es mediante el uso de botones gráficos y el mouse que se toman las acciones pertinentes. Al entrar a DECIS se presenta al usuario un menú de botones que entre otras cosas explica el uso del sistema. Después se tiene acceso a un menú principal de donde se puede navegar a las distintas opciones de DECIS. Aquí están los siguientes botones: *Ayuda*, que explica la función de los botones; *Índice temático*, que sirve para ir directamente a cada uno de los temas que se exponen en la sección de consulta; *Bibliografía*, que presenta una selección de títulos sobre teoría de toma de decisiones; *Consulta*, que abre la sección de consulta, donde están los temas a ser estudiados por el usuario; *Ejercicios*, donde se encuentra un juego de ejercicios para cada tema de manera que el estudiante reafirme los conocimientos adquiridos; *Exámenes*, sección en la cual hay exámenes sobre los temas expuestos en la sección de consulta. En la parte inferior, a la izquierda hay un botón que da la hora actual el cual es ocultable al hacer "click" sobre el mismo, y a la derecha está el botón de *Salir*, que al ser presionado con el cursor y el mouse permite terminar la sesión y regresar al *Administrador de programas de Windows*. En la figura 2.1. se puede apreciar la interfaz de usuario de DECIS.



Figura 2.1. Interfaz de usuario de DECIS.

El curso de acción en el uso del sistema educativo DECIS es el siguiente: el usuario entra al sistema, se entera de cómo usarlo y de la función de los botones y comienza a estudiar la teoría de toma de decisiones en la sección de consulta. Después de haber estudiada cada lección realiza los ejercicios correspondientes y pasa a la sección de exámenes donde podrá realizar el examen del capítulo estudiado. DECIS permite realizar exámenes en un orden secuencial, es decir, no se puede realizar el tercer examen sin haber aprobado el segundo. DECIS lleva la cuenta de las calificaciones del estudiante en un archivo secreto para el usuario. Desde cualquier parte del sistema educativo se pueden presionar los demás botones, lo que hace que la navegación de DECIS sea muy versátil, pero durante la realización de un examen estos botones están inactivos de manera que el estudiante está obligado a terminar el examen sin abortar el sistema o consultar el capítulo.

DECIS es un desarrollo innovador en el aspecto educativo, pues se cuenta además con un programa para uso del profesor, donde este último lleva a cabo la administración de las calificaciones de los estudiantes. El Laboratorio de Inteligencia Artificial y Multimedia del Instituto de Ingeniería, LIAMMI, realiza constantemente nuevas aplicaciones como tutoriales educativos.

Capítulo 3

Microprocesadores

3. MICROPROCESADORES

3.1. EL MICROPROCESADOR

3.1.1. ¿QUE ES UN MICROPROCESADOR?

Los microprocesadores son actualmente el corazón de las computadoras digitales y de otros muchos sistemas electrónicos. Pueden ser encontrados en calculadoras de bolsillo, terminales de chequeo en tiendas, equipo de oficina, instrumentos científicos, equipo médico y video juegos, sólo por nombrar algunas aplicaciones. De hecho, cada día se encuentran nuevas aplicaciones y se desarrollan nuevos productos basados en microprocesadores.

El microprocesador es esencialmente el producto de veinte años de investigación y desarrollo en dos campos afines: computadoras digitales y tecnología de semiconductores. Consiste en varios miles de transistores incorporados en un circuito integrado. El microprocesador es un dispositivo capaz de llevar a cabo muchas de las operaciones básicas de una computadora digital y de otros muchos sistemas digitales.

Un microprocesador es un circuito combinacional y secuencial; es a su vez, un sistema digital síncrono programable de proceso. Esto significa que trabaja paso a paso realizando un proceso, activado por los impulsos de un reloj maestro y mediante una secuencia de operaciones elementales denominadas microoperaciones que son indicadas por las instrucciones de un programa, el cual sigue un determinado diagrama de flujo. Es digital, pues las instrucciones, datos y direcciones de memoria son representadas por cadenas de unos y ceros, lo que eléctricamente son voltajes altos y bajos.

Las funciones principales de un microprocesador son:

- Provee las señales de tiempo y control para todos los elementos del sistema que lo rodea.
- Busca instrucciones y datos desde la memoria.
- Transfiere datos desde y hacia dispositivos de entrada/salida (E/S).
- Decodifica instrucciones.
- Realiza operaciones lógicas y aritméticas solicitadas a través de instrucciones.
- Responde las señales de control de E/S, tales como RESET o INTERRUPCIÓN.

3.1.2. HISTORIA DEL MICROPROCESADOR

La historia de los microprocesadores viene a darnos una idea de la *electrónica antes del microprocesador*. A continuación se habla de cómo surgió el microprocesador y de algunos cambios que esto implicó en la tecnología.

La máquina llamada ENIAC (Electronic Numerical Integrator and Calculator) es reconocida como la primera computadora digital. Fue diseñada y construida en la Universidad de Pennsylvania en la década de los 40's. Usaba alrededor de 18,000 bulbos o tubos de vacío, se almacenaba en un cuarto de aproximadamente 20m x 10m y pesaba más de 30 toneladas. Desafortunadamente, el principal problema operacional de aquellas primeras computadoras era la falta de confiabilidad. Debido al gran número de componentes involucrados y a la vida finita de un bulbo, el tiempo entre fallas era típicamente de unos minutos.

William Shockley inventó el transistor en 1948 y esto pronto comenzó a influenciar la industria de las computadoras. Al principio de los 60's una segunda generación de computadoras digitales fue manufacturada usando la nueva tecnología de "estado sólido". Aunque los circuitos se basaban en el transistor

de germanio, que tiene muchas desventajas, la confiabilidad se mejoró substancialmente y la computadora digital comenzó a convertirse en una propuesta comercialmente viable.

A mitad de los 60's se dio un mayor progreso tecnológico en la industria de los semiconductores. Usando una versión avanzada de técnicas de fabricación de transistores se hizo posible producir en una sola "hostia" u "oblea" de silicón un circuito integrado (CI o IC por sus siglas en inglés) que consistía en muchos transistores interconectados. La nueva tercera generación basada en esta tecnología de circuitos integrados tenía mejoras como una mayor confiabilidad, reducido tamaño y un ahorro substancial en el costo económico.

Los tipos de computadoras que se diseñaron comenzaron a diversificarse, y esta tendencia continúa actualmente. Por un lado están las computadoras llamadas "main-frame", máquinas capaces de manejar y almacenar grandes cantidades de información, o de hacer cálculos complejos a gran velocidad. Por otro lado, el circuito integrado hizo posible una computadora de bajo costo, que es de operación más lenta y tiene características limitadas, pero que es perfectamente adecuada para muchas aplicaciones. Esta máquina se conoce como "minicomputadora". Ha crecido su popularidad al paso de los años y este tipo de computadora se ha desarrollado al grado que es ahora más rápida y flexible que la mayoría de sus contrapartes "main-frame" de los 60's.

Una de las partes más importantes de cualquier computadora digital es la Unidad Central de Proceso (CPU por sus siglas en inglés). Todos los cálculos aritméticos y decisiones lógicas se llevan a cabo en la CPU y de este dispositivo emanan la mayoría de las señales de control. En una minicomputadora la CPU consistiría típicamente en varios circuitos integrados conectados juntos en una tableta de circuito impreso. Sin embargo, en 1969 un fabricante de calculadoras propuso a la Corporación Intel discutir el diseño de un circuito de alta escala de integración (LSI por sus siglas en inglés) para usarse en una calculadora. El producto resultante consistió en un sólo circuito integrado o "chip" que realizaba todas las operaciones de control y aritméticas necesarias de acuerdo a un programa almacenado. Este CI tenía características de operación similares a las de la CPU de una minicomputadora y como consecuencia fue llamado *microprocesador*.

El primer microprocesador fue conocido por el número 4004 y tenía una longitud de palabra de instrucción y datos de 4 bits. Intel introdujo al mercado este dispositivo en aplicaciones que no eran calculadoras y la primera generación de sistemas basados en microprocesador o sistemas con microprocesador fue entonces desarrollada. El 4004 fue pronto seguido por un microprocesador de 8 bits denominado 8008. Este era un dispositivo más general y tenía un conjunto de palabras de instrucción muy similares a las de una minicomputadora. Un año después se hizo disponible el 8080: era un dispositivo de 8 bits, veinte veces más rápido que su predecesor, el 8008. En 1974, Motorola introdujo el microprocesador de 8 bits llamado 6800. Durante algunos años el 6800 y el 8080 fueron los líderes en el mercado a pesar de la entrada al campo de otros fabricantes. Compañías tales como Fairchild, Texas, Signetics, RCA, National Semiconductor, Zilog y Mostek produjeron y mejoraron dispositivos desde la aparición en 1971 del 4004. Estas mejoras consisten generalmente en que más tareas son disponibles en un CI; la operación es más veloz; la longitud de palabra es mayor y la capacidad aritmética se extiende. Desde 1971, la complejidad de los microprocesadores se ha duplicado cada dos años.

Conectando un chip microprocesador con otros CIs y componentes es posible producir una *microcomputadora*. Este dispositivo, que opera de manera similar a una minicomputadora, es considerablemente más barato pero más lento y menos completo. Sin embargo, estas diferencias están disminuyendo, y la última generación de microcomputadoras es tan rápida y tiene casi tantas capacidades como las minicomputadoras de hace algunos años.

Es posible reprogramar una microcomputadora de la misma manera que una minicomputadora o una main-frame. Sin embargo, otros tipos de "computadoras" pueden ser usadas como "sistemas dedicados" con programas fijos, y también es posible usar un microprocesador de esta manera. Debido al bajo costo de un chip microprocesador, un área de gran desarrollo potencial es en esas aplicaciones donde un microprocesador, junto con otros componentes, lleva a cabo una tarea fija bajo el control de un programa almacenado

permanentemente. Este sistema es comúnmente llamado "sistema basado en microprocesador" o "sistema con microprocesador implantado", dejando la palabra "microcomputadora" para ser aplicada a aquellos sistemas donde la reprogramación y la capacidad multitarea son pretendidas y fácilmente obtenibles.

Las computadoras digitales han tenido una gran influencia en nuestra sociedad y manera de vivir desde 1951, cuando la primera computadora digital comercial (Univac 1) se hizo disponible. Una nueva tecnología había emergido. Términos como "computación digital", "diseño lógico" y "programación" se convirtieron en conceptos integrales a la ciencia y la ingeniería. Sin embargo, la diversidad de estos conceptos frecuentemente causó una división de intereses. Por ejemplo, estaban aquellos cuyo interés era usar y programar las computadoras (software), mientras estaban también otros cuyo interés era diseñar computadoras (hardware). Aunque esta dicotomía en interés puede tener fundamentos reales en las computadoras de gran escala, los problemas que enfrentan programadores de aplicación y diseñadores de computadoras se interrelacionaron más con la introducción de las minicomputadoras en 1965. Estas computadoras no sólo estaban destinadas al procesamiento de datos y solución de problemas, sino que iban a formar parte de sistemas que requerían decisiones inmediatas de computadora, lo que se llama "Sistemas de tiempo real".

La introducción del microprocesador en 1971 acercó más estos dos tipos de interés. Resultó una era de diseño lógico de software o *lógica programada*. En esta era, los conceptos de programación y los principios de diseño lógico se mezclaron hasta el punto en que sus interacciones requieren científicos e ingenieros para tener una familiaridad completa con los principios computacionales de software y hardware que permita utilizar al máximo el potencial del microprocesador. El conocimiento de la construcción interna (hardware) del microprocesador y a su vez del sistema que interactúa con el mismo (muchas veces todo junto es un computador digital), permite al programador o creador de software un mayor alcance en cuanto al poder de acción de su trabajo.

3.1.3. EL MICROCOMPUTADOR Y EL SISTEMA CON MICROPROCESADOR

Para entender el funcionamiento de los microprocesadores, conviene, antes de estudiarlos intrínsecamente, dar un vistazo general a la forma en que se utilizan estos dispositivos.

El microcomputador

Sin pretender introducirnos mucho en el tema de los microcomputadores, diremos que son sistemas computacionales basados en microprocesador. Se utiliza uno de estos dispositivos para generar las señales necesarias de control del microcomputador, así como para efectuar las operaciones aritméticas y lógicas. De este modo, en el centro de todas las operaciones está la MPU (Unidad Microprocesadora), constituida por un chip microprocesador. Los microcomputadores pueden ser reprogramados para efectuar distintas tareas. En este sistema, el microprocesador toma parte de su programación de un bloque ROM (Memoria de sólo lectura), donde está un programa invariable que comanda distintas tareas del sistema, como son:

- Inicialización de la operación del microcomputador.
- Diagnóstico de encendido y revisión del sistema.
- Determinación de la configuración del sistema.
- Manejo y detección de dispositivos de E/S.
- Cargado de algún sistema operativo.

En un bloque de memoria RAM (Memoria de acceso aleatorio, también conocida como memoria de lectura-escritura) se almacenan datos utilizados durante los cálculos y también programas para realizar tareas de distinta índole. Actualmente todos hemos utilizado una microcomputadora. Estamos familiarizados con sus unidades de entrada, de las cuales la más común es el teclado, así como con las unidades de salida como son visualizadores de siete segmentos, monitores, impresoras, etc. Estos dispositivos son conocidos

como periféricos. Para nosotros es común teclear un programa en la computadora, apagarla, y en algún otro momento encenderla para introducir otro programa con el cual se efectuará una tarea distinta. En la figura 3.1 se aprecia la arquitectura generalizada y simplificada de una microcomputadora.

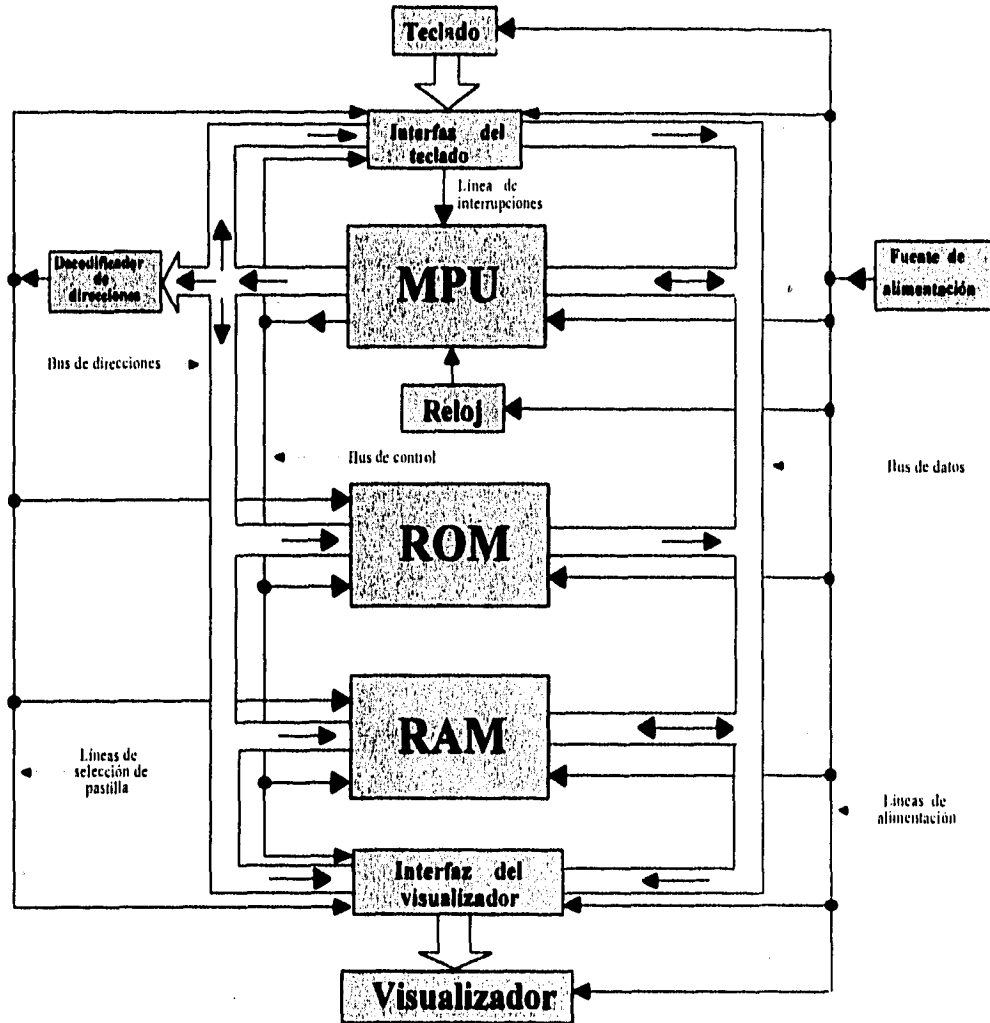


Figura 3.1.

La MPU necesita conexiones con la fuente de alimentación y el reloj. Este último puede ser un circuito separado o puede residir en la pastilla del microprocesador.

La arquitectura mostrada en la figura 3.1 tiene dos tipos de memorias semiconductoras. La ROM es la memoria permanente, que probablemente contiene el programa monitor del sistema. También se muestra una RAM como dispositivo de almacenamiento temporal de lectura/escritura.

El sistema microcomputador mostrado tiene un teclado como dispositivo de entrada. La interfaz del teclado almacena los datos y almacena las entradas del teclado. En el instante adecuado este circuito interrumpe la MPU vía la línea de interrupción especial. Esta señal de interrupción hace que la MPU (1) termine de ejecutar la instrucción actual, (2) suspenda la operación normal y (3) bifurque a un grupo especial de instrucciones del programa monitor que manipulan la entrada de datos desde el teclado. El circuito de interfaz del teclado tiene entradas de dirección, de selección de pastilla y de control para activar la unidad.

Una vez activada, la interfaz del teclado pondrá los datos del teclado en el bus de datos. La MPU acepta los nuevos datos de entrada vía el bus de datos.

Esta microcomputadora utiliza como salida un Visualizador que puede ser un grupo de visualizadores de siete segmentos. Cuando es activada por las entradas de selección de pastilla y bus de direcciones, la interfaz del visualizador acepta los datos por el bus de datos y los almacena. La interfaz excita entonces al visualizador para presentar la información de salida.

Para conectar o habilitar solamente el dispositivo ordenado, el decodificador de direcciones muestrea datos en el bus de direcciones. La lógica combinacional del decodificador de direcciones activa la línea adecuada de selección de pastilla, habilitando el dispositivo correcto.

El sistema con microprocesador

El microprocesador es, por supuesto, el componente central de cualquier sistema con microprocesador. Controla las funciones realizadas por otros dispositivos del sistema y proporciona al mismo su capacidad aritmética y lógica. El microprocesador busca las instrucciones en la memoria, las decodifica y las ejecuta. Referencia memoria y dispositivos de entrada-salida (E/S) en busca de datos y responde a señales de control procedentes de dispositivos externos.

Las señales de control procedentes de dispositivos externos pueden hacer que el microprocesador tome algunas de las siguientes acciones:

1. Reset, provoca que la ejecución del programa comience desde alguna posición inicial.
2. Esperar una cantidad de tiempo suficiente para acceder a una determinada posición de memoria.
3. Interrumpir la ejecución del programa en curso y bifurcar a una posición de memoria donde empieza una subrutina que da servicio al dispositivo externo causante de la interrupción.
4. Suspender su operación y poner sus patas de direcciones y datos en flotación, permitiendo que otros dispositivos lean o escriban directamente en memoria (DMA).

El propósito más fundamental de un sistema con microprocesador es procesar datos digitales que son introducidos desde el mundo exterior y dar como salidas datos digitales que son una función de los datos de entrada. Cuando el sistema con microprocesador debe procesar o generar datos en forma analógica, se utilizan subsistemas de conversión analógico-digital y digital-analógico con el fin de transformar la entrada analógica en la forma digital requerida y viceversa. En la figura 3.2 se aprecia un sistema con microprocesador (típico).

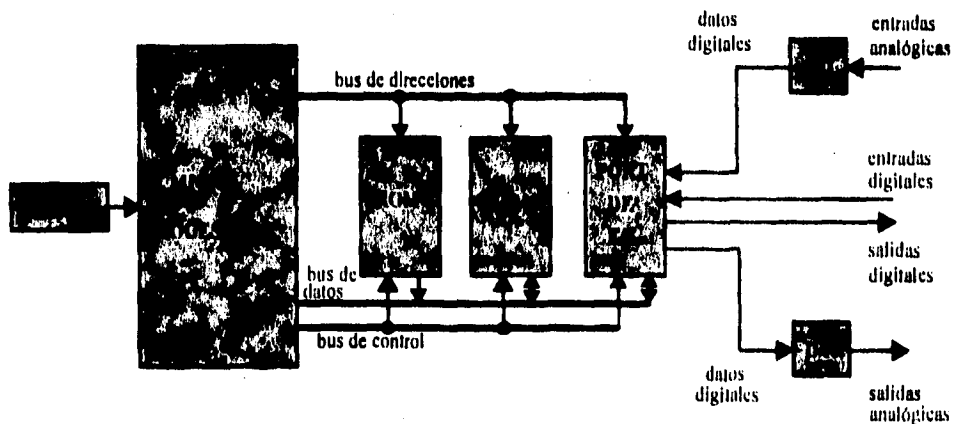


Figura 3.2.

En un sistema como éste, la memoria de datos se almacena en la RAM y la memoria de programa en la ROM. Este es un sistema dedicado, es decir, realiza una tarea indicada por el programa almacenado en ROM. En general, en el presente trabajo nos referiremos a sistemas como éste. Un sistema de esta naturaleza se puede utilizar en muchas aplicaciones como control automático de tránsito de trenes, sistemas de medición e instrumentación electrónica, equipo médico y científico, y en otros muchos campos.

Para entender los sistemas con microprocesadores, es ventajoso ver el sistema completo — microprocesador, ROM, RAM y puertos o ports de E/S— como una colección de registros direccionables. Los registros que residen en el interior del microprocesador se denominan *registros internos* y aquellos que existen en ROM, RAM y los puertos son los *registros externos*.

El conjunto de registros que constituyen un sistema en particular y las transferencias de datos que sean posibles entre ellos forman la *arquitectura del sistema*.

Los registros en los varios subsistemas del microprocesador se interconectan externamente por el *bus del sistema*, que incluye el bus de direcciones, el bus de datos y el bus de control.

El microprocesador comanda las operaciones de lectura y escritura de datos en memoria y puertos E/S por medio de un conjunto de señales generadas por la unidad de control. Los distintos microprocesadores presentan diferentes conjuntos de señales de control. Todas estas señales se transmiten por el bus de control del sistema.

3.1.4. ARQUITECTURA DEL MICROPROCESADOR

Los tipos de registros en el microprocesador (μP) y las posibles transferencias de datos entre ellos determinan la *arquitectura del microprocesador*.

Casi todos los microprocesadores contienen como mínimo lo siguiente:

1. Unidad de control y temporización.
2. Unidad aritmética y lógica (ALU).
3. Varios registros.
4. Contador de programa.
5. Circuitería de decodificación de instrucciones.
6. Varias entradas y salidas de control.
7. Líneas de control y buses internos.
8. Cerrojos y buffers.

Además de estos elementos, una pastilla microprocesadora puede contener algunas de las unidades funcionales siguientes:

1. Memoria ROM.
2. Memoria RAM.
3. Puertos de entrada/salida (E/S) serie.
4. Circuitería de reloj interna.
5. Temporizadores programables.
6. Circuitería de arbitración de prioridad de interrupciones.
7. Lógica de interfaz de comunicación de E/S serie a paralelo.
8. Lógica de control de acceso directo a memoria (DMA)

En general, cuando se contienen varios de estos últimos elementos, a la pastilla o chip se le llama *Microcontrolador* (Es posible abreviarlo μC , pero esto se puede confundir con "Microcomputador"), pues está capacitado para controlar la operación de un sistema programable.

Antes de describir los componentes del microprocesador, conviene hablar del *tamaño en bits del microprocesador*, el cual es denominado a veces *tamaño de palabra*, refiriéndose a la palabra de datos que maneja un microprocesador en especial. La longitud del acumulador es en general un buen indicador para conocer esta característica. Normalmente hay μP 's de 4, 8, 16 y 32 bits.

En la figura 3.3 se puede apreciar el diagrama de bloques que representa la arquitectura de nuestro microprocesador genérico. No se trata de un μP comercial real, sino de un microprocesador ideado para fines didácticos. Este es un μP de 8 bits, y entre paréntesis aparece el número de bits de sus componentes.

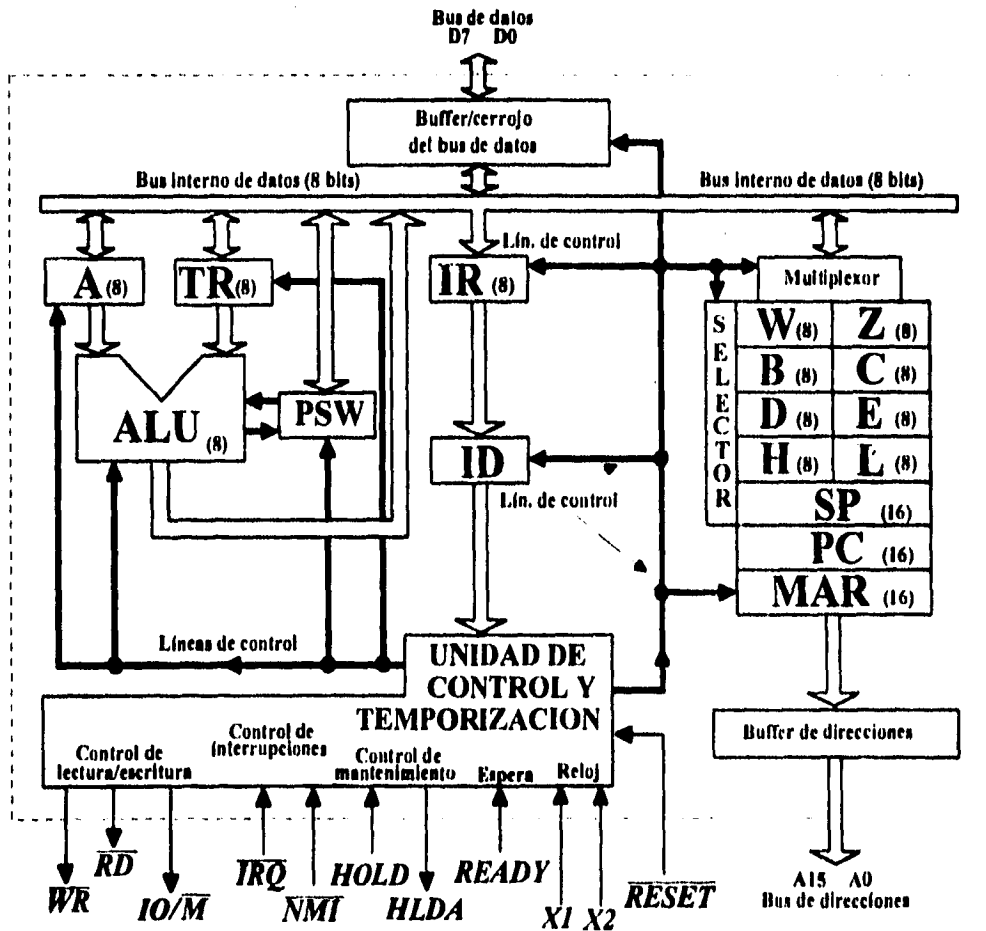


Figura 3.3. Diagrama de la arquitectura del microprocesador genérico.

Las iniciales se desglosan a continuación:

ALU. Unidad aritmético lógica.

A. Acumulador.

TR. Registro temporal para la ALU.

PSW. Registro de status.

IR. Registro de instrucción.

ID. Decodificador de instrucción.
W,Z. Registros temporales (par de registros WZ).
B,C,D,E. Registros de propósito general.
H,L. Registro par de datos/direcciones.
SP. Apuntador de pila.
PC. Contador de programa.
MAR. Registro de direcciones.

La unidad de control

La clave del funcionamiento del microprocesador es la unidad de control o controlador, también llamada *unidad de control y temporización*; ésta es un sistema secuencial síncrono de control, es decir, un sistema que al ser excitado por los pulsos del reloj recibe las señales del decodificador de instrucciones para determinar la naturaleza de la instrucción que se va a ejecutar y genera las señales adecuadas para gobernar la unidad aritmético lógica (donde se realizan todos los cálculos), las operaciones de lectura/escritura (a través del bus de control) en los registros utilizados en los cálculos, en la memoria de datos utilizada por el microprocesador y en los puertos de entrada/salida, y en general las demás partes del microprocesador. También incrementa o decrementa el contador de programa. Estas señales de temporización y control son enviadas a todo el microprocesador para coordinar la ejecución de las instrucciones. Cada instrucción indica distintas microoperaciones a realizar. Se entiende por microoperación una operación aritmética o lógica que se puede realizar durante un ciclo del generador de impulsos o reloj, equivalente a un único estado de la unidad de control, o a una transferencia de información entre registros realizada directamente o a través de la ALU. La secuencia de microoperaciones realizadas en cada uno de los estados de la unidad de control da lugar a un proceso de información.

Una unidad de control programable tiene dos ciclos diferenciados:

- a) Un ciclo en el cual genera los impulsos adecuados para leer la instrucción contenida en la memoria de instrucciones o memoria de programa y que se denomina ciclo de búsqueda o "Ciclo Fetch". De manera genérica el ciclo Fetch se divide en dos partes: en la primera se direcciona, es decir se accede a la dirección de la memoria de programa donde se encuentra la instrucción en turno y en la segunda parte se decodifica esa instrucción.
- b) Un ciclo en el cual genera los impulsos adecuados para ejecutar la instrucción y que se denomina "Ciclo de ejecución".

La unidad de control dirige la realización del proceso y la unidad aritmético lógica lo ejecuta (de manera general). Por ello ambos bloques, junto con algunos registros que se utilizan durante el proceso de información, se pueden reunir en uno solo que recibe el nombre de Unidad Central de Proceso (Central Processing Unit, abreviado como *CPU*).

La unidad aritmético lógica

Las operaciones aritméticas o lógicas en uno o dos operandos constituyen las transformaciones básicas de datos implementadas en un microprocesador. El microprocesador contiene una unidad aritmético y lógica, *ALU*, para dicho fin. La ALU, que se puede imaginar como una calculadora controlada electrónicamente, utiliza métodos binarios para representar y operar números. La ALU contiene un sumador que sólo realiza sumas binarias (la sustracción se hace complementando la adición). Los operandos le pueden ser suministrados por varios componentes del sistema, tales como: registros, acumuladores, memorias de datos o dispositivos de entrada/salida.

Lo más común en las operaciones que lleva a cabo la ALU es que el acumulador contenga uno de los operandos y que el segundo operando se encuentre en un registro temporal; de este modo, el resultado de una

operación aritmética o lógica se coloca en el acumulador al final de la operación, reemplazando a uno de los operandos originales.

La ALU es capaz de realizar los siguientes operaciones en datos binarios:

1. Suma y resta binaria.
2. AND, OR, EX-OR lógico.
3. Complemento.
4. Rotación a la izquierda y a la derecha.

Algunas ALU's pueden realizar otras operaciones.

La ALU contiene bits de estado que registran e indican ciertas condiciones específicas que pueden suceder durante las manipulaciones propias de la unidad (principalmente las aritméticas). Estos bits de estado están indicados por el estado de unos flip-flops y se denominan *flags*. Reunidos, dichos flags constituyen el "registro de flags" o "registro de estado", donde se agrupan las condiciones de estado. Algunas condiciones típicas indicadas en estos bits son:

- Rebose (overflow)
- Cero
- Signo negativo
- Acarreo

Por ejemplo, un flag puede indicar que el resultado de un cómputo fue 0 y el programador puede entonces usar la detección de esta condición para una toma de decisión. Siendo 0 el resultado, se realizan unos cálculos; siendo no 0, se realizan otros cómputos correspondientes a otra parte del programa.

La información del registro de flags o de estado, junto con más información de estado del microprocesador, forma parte de un registro especial llamado en muchos microprocesadores *PSW (program status word)*.

En muchas aplicaciones es apropiado representar datos en decimal codificado en binario, BCD. Muchas ALU's contienen lógica adicional para ajustar los resultados de operaciones de suma cuando los operandos son interpretados como datos BCD.

Registros y contadores

Los registros y contadores son partes fundamentales de todos los microprocesadores, computadores y demás sistemas con microprocesador.

Los *registros de almacenamiento* se usan para el almacenamiento temporal de bits binarios en el microprocesador. Un sistema con microprocesador implementa sus funciones a base de transferir y transformar datos en registros del sistema. Típicamente, las transformaciones en los datos tienen lugar en los registros internos, muchos de los cuales son registros de operación. Los *registros de operación* se diferencian de los *registros de almacenamiento* en que con su circuitería asociada implementan operaciones aritméticas o lógicas en los datos contenidos en dichos registros, transformándolos.

Es posible diseñar un registro que pueda realizar las siguientes funciones:

- Entrada serie salida serie (derecha o izquierda).
- Rotación derecha / izquierda.
- Entrada paralelo / salida paralelo.
- Entrada serie / salida paralelo.
- Entrada paralelo / salida serie.

En los sistemas digitales, los registros pueden ser de propósito general o dedicados. Los registros de propósito general pueden utilizarse para varias funciones bajo control del programa. Los registros dedicados sólo pueden realizar ciertas funciones. Están cableados en la máquina para hacer estas funciones y no están bajo control del programa. Normalmente, los registros se inicializan a cero al principio del programa.

Los registros dentro del microprocesador se llaman *registros internos* y los que están fuera de éste, formando parte del sistema con microprocesador, se llaman *registros externos*.

Los *contadores* son dispositivos que registran y mantienen la cuenta de un número de pulsos o bits. Un contador puede contar en sentido positivo o negativo. El contador positivo se borra igualando todos sus bits a cero y se actualiza su valor incrementándolo cada vez que llega un bit. Los contadores negativos se borran, igualando todos sus bits a uno (su máxima capacidad de contaje) y a cada bit se decrementa su valor en una unidad. En lugar de borrar el registro igualándolo a cero o a máximo, es posible cargar un número en el contador y después incrementarlo o decrementarlo. Un contador se lee siempre en paralelo para no destruir los bits originales del mismo.

El acumulador

Por regla general, el acumulador es el principal registro de trabajo del microprocesador. Es común abreviarlo *A* o *ACC*.

Usualmente, el acumulador se considera un registro de propósito general en la mayoría de los microprocesadores y almacena uno de los operandos que usa la ALU al realizar operaciones aritméticas o lógicas. El acumulador puede trabajar como registro fuente o registro de destino. Por ejemplo, el programa puede llamar el contenido de una cierta posición de memoria para ser sumado al actual contenido del acumulador y el resultado ser devuelto a la misma, o a cualquier otra posición de memoria. En este caso, el acumulador funciona como un registro fuente. Pero si el programa llama el contenido de una posición de memoria para sumarlo al contenido actual del acumulador y el resultado se mantiene en el acumulador, éste se convierte en un registro fuente y, a la vez, en un registro de destino.

Muchas veces el acumulador está diseñado para realizar varias funciones suplementarias tales como: complemento del contenido, desplazamiento a la izquierda o derecha del número y rotación del número a la derecha o izquierda. En muchas ocasiones, el acumulador está diseñado para contener un total acumulativo o progresivo de todos los números que se le transfieren. Cada número que se le transfiere se suma sucesivamente a la suma previa. Así mismo, los acumuladores pueden diseñarse para que, bajo control del programa, resten la cantidad entrada a la previamente existente.

Los registros de propósito general

Si el microprocesador los contiene, los también conocidos como *registros de trabajo*, sirven para almacenar internamente los resultados temporales procedentes de cálculos, en lugar de hacerlo en la RAM externa. En varios microprocesadores a estos registros se les asignan letras como *B, C, D, E*. Cada operación de lectura o escritura a memoria hecha por el microprocesador recibe el nombre de *referencia a memoria*. Las instrucciones que transfieren datos entre estos registros requieren pocos bits para hacer el direccionamiento pues el número de registros es limitado. Gracias a esto se requiere un menor tiempo de ejecución, pues se requieren menos bytes y menos referencias a memoria para buscar datos. Además la transferencia de datos entre el acumulador y estos registros no requiere una referencia a memoria, ya que la transferencia tiene lugar en el mismo microprocesador.

Los registros temporales

Un microprocesador (μP) maneja *registros temporales* para distintas tareas. Algunos μP 's contienen un par de registros usualmente denominados *W* y *Z*, que son utilizados por la unidad de control para mantener operandos o direcciones que son parte de una instrucción, hasta que son transferidos a otro registro en el microprocesador o utilizados como operandos en un cálculo. Los registros *W* y *Z* pueden ser usados como el par de registros *WZ*.

Otro caso es el de un registro temporal, que para propósito de esta explicación será denominado como *TR* (Temporal register), el cual es utilizado sólo por la ALU para sus operaciones, manteniendo un dato proveniente de otra fuente para realizar una operación relacionándolo con el acumulador.

El registro de status

Comúnmente llamado "program status word" (*PSW*) o también "registro de códigos de condición" (*CCR* por sus siglas en inglés), este registro almacena, además de los bits de estado de la ALU, señalizadores tales como el de máscara de interrupción, información que es útil en instrucciones de bifurcación. Para nuestro μP genérico, este registro contiene los siguientes señalizadores: Arrastre *C* (bit 0), Overflow *V* (bit 1), Cero *Z* (bit 2), Negativo *N* (bit 3), Máscara de interrupción *I* (bit 4). Los demás bits del registro de ocho bits están cargados con 1's.

El registro de dirección

El registro de dirección es comúnmente conocido como *MAR* (Memory Address Register). Este es un dispositivo de almacenamiento temporal para mantener la dirección de lectura, para el caso de la *ROM* o de lectura/escritura de datos, para la *RAM*. Las salidas del registro de dirección son las patas de direcciones del microprocesador. Tratándose de la *ROM*, para obtener la primera palabra de la instrucción, la dirección contenida en el contador de programa se coloca en el bus de direcciones. Para ello, la unidad de control transfiere el contenido del contador de programa al *registro de dirección* y de ahí pasa al bus de direcciones. Si se trata de direccionar la *RAM*, este dispositivo puede presentar algunas cualidades y ventajas: en algunos microprocesadores este registro es programable, es decir, el programador puede modificar su contenido mediante ciertas instrucciones, como por ejemplo, incrementándolo o decrementándolo.

El registro de dirección consiste en general en flip-flops del tipo *D* que mantienen la dirección durante un periodo predefinido de tiempo. Normalmente los chips de memoria son más lentos que el microprocesador. El mantenimiento de la dirección en este registro durante un periodo de tiempo más largo permite usar chips de distintas velocidades.

El registro par de datos/direcciones

Algunos microprocesadores manejan un par de registros, *H* y *L*, para propósitos similares al *MAR*, que pueden ser utilizados de distintas maneras. Es común llamarlos par de registros *H/L* de datos/direcciones y pueden ser utilizados como punteros de direcciones donde se encuentran datos (puntero de datos). A veces es posible también usar los registros de propósito general como punteros de datos. De cualquier manera, al usarse de este modo, estos registros son fuente de direcciones para el *MAR* o algún cerrojo de direcciones en el μP . Los μP 's que incluyen el par de registros *H/L*, lo usan en varias instrucciones.

El registro de instrucción

El registro de instrucción (*IR* por sus siglas en inglés) es un registro dedicado que almacena el primer byte de una instrucción, que es el *código de operación u opcode*; la función del código de operación es indicar a la unidad de control las operaciones requeridas para ejecutar la instrucción. Esta información debe mantenerse temporalmente en el registro IR para poder ser decodificada y mandada a la unidad de control.

El decodificador de instrucción

Esta unidad, abreviada como *ID* (Instruction Decoder), interpreta el contenido del registro de instrucción, determina el programa exacto que se debe seguir para ejecutar la instrucción completa y dirige adecuadamente la sección de control.

El contador de programa

Las instrucciones que constituyen el programa están almacenadas en posiciones sucesivas de la memoria de programa que puede consistir en uno o varios chips de memoria ROM en los sistemas con microprocesador. A cada posición de memoria se le asigna un número o código único, llamado dirección. Para ejecutar el programa en la secuencia correcta, la unidad de control debe saber en qué posición de memoria buscar la próxima instrucción. Con este propósito, la unidad de control mantiene o gobierna un registro de propósito especial o dedicado, llamado contador de programa o *PC* (por sus siglas en inglés). Este es un registro de operación que siempre mantiene la dirección de la próxima instrucción a ejecutar o bien la dirección de la siguiente palabra de una instrucción multipalabra, la cual no ha sido buscada por completo. En cualquier caso, al completarse la ejecución de cualquier instrucción, el contador de programa contiene la dirección de la primera palabra de la siguiente instrucción a ejecutar. Es un registro de operación, pues su contenido puede ser incrementado por la unidad de control.

Cuando se tiene una instrucción de bifurcación, ésta dirige el contador de programa a una dirección distinta de la siguiente dirección secuencial. Esta instrucción contiene la dirección a la que se quiere dirigir el programa y esta dirección se inserta automáticamente en el contador de programa. Una instrucción de bifurcación puede ser un salto hacia delante o hacia atrás dentro del programa principal.

Otro caso es la instrucción de llamada a subrutina. Las subrutinas son "programas dentro del programa" que generalmente se utilizan más de una vez durante la ejecución del mismo. Para asegurar un retorno ordenado al programa principal después de cumplida la subrutina, es necesario que la próxima instrucción del programa principal que sigue a la llamada a subrutina sea guardada en algún sitio. Antes de ejecutar la subrutina, la unidad de control incrementa el contador de programa y almacena esta dirección en un grupo especial de registros en la RAM llamado *stack* o *pila* (este es el caso de nuestro microprocesador genérico, con un *stack tipo puntero*). Así, el contador de programa se libera para realizar su labor durante el desarrollo de la subrutina.

Una de las entradas de control es la de reset. Cuando el microprocesador es inicializado al habilitar esta entrada, la unidad de control coloca el contador de programa en cero. Este valor inicial establece la dirección de memoria de donde se obtendrá la primera instrucción.

Para nuestro microprocesador genérico, el PC es un registro de 16 bits, formado por un par de registros de 8 bits cada uno; uno para el byte superior y otro para el byte inferior del PC.

El stack pointer

También conocido como *puntero* o *apuntador de pila*, el *stack pointer* o *SP* es un registro que se encarga de seguir la pista de la siguiente posición de memoria disponible en el stack o pila, donde se puede encontrar una dirección de retorno de subrutina o bien de retorno después de un servicio de interrupción. El stack, que para nuestro microprocesador genérico es del tipo puntero, es un área reservada de la RAM, que también puede ser usado para almacenar datos.

Del mismo modo que el contador de programa, en nuestro microprocesador genérico el stack pointer es un registro de 16 bits formado por el registro para el byte superior del SP y el registro para el byte inferior del SP.

El stack pointer también es una fuente de direcciones para el MAR, como se aprecia en la figura 3.4:

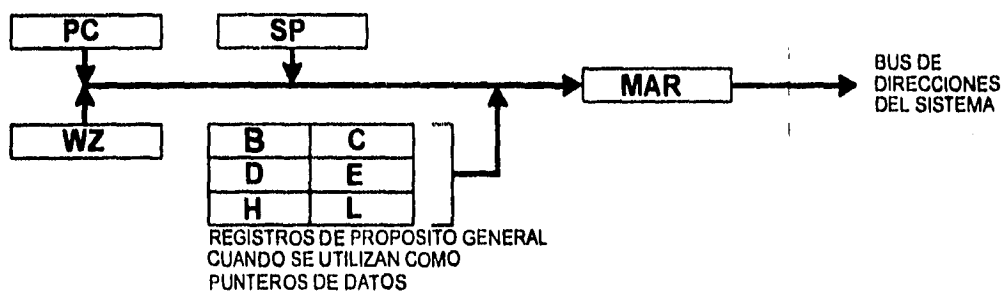


Figura 3.4. Fuentes de direcciones para el registro de direcciones.

Los buses

Los buses son grupos de líneas que interconectan los distintos dispositivos de un sistema con microprocesador, o de un computador. Dentro del microprocesador, también se pueden identificar buses que interconectan diferentes partes del chip; para efectos de esta explicación llamaremos a estos últimos *buses internos* y a los que sirven al microprocesador para conectarse con ROM, RAM y puertos de E/S los denominaremos *buses externos*.

Los buses pueden ser unidireccionales, en los que la información fluye en una sola dirección o bidireccionales, donde ésta fluye en ambas direcciones. Naturalmente al ser líneas de transmisión de impulsos eléctricos, tienen siempre capacidad bidireccional; la distinción unidireccional es definida, en realidad, por el dispositivo al que está conectado el bus o una determinada línea del bus.

Tradicionalmente se diferencian tres buses externos: *el bus de datos*, *el bus de direcciones* y *el bus de control*. El llamado *bus del sistema* incluye a su vez a estos tres buses.

El *bus de datos* parte lógicamente de las patillas de datos del microprocesador; es bidireccional, pues por éste circula información que entra al microprocesador desde la ROM (la conexión con la ROM es unidireccional pues no se puede escribir en ésta), o que entra y sale para comunicarse con la RAM y los puertos E/S o I/O.

El *bus de direcciones* es unidireccional, pues a través de éste el microprocesador indica posiciones de memoria a la RAM y la ROM, además de especificar el puerto específico de E/S con el que se va a comunicar.

Algunos microprocesadores multiplexan el bus de datos con parte del bus de direcciones; esto es conocido como bus multiplexado, y pasa cuando las patillas no son suficientes para todas las funciones del microprocesador. En estos casos se utiliza una salida de control que indica el modo de utilización del bus.

El *bus de control* es usualmente unidireccional por línea, esto es, algunas líneas son de salida y otras de entrada al microprocesador. Por medio de las *salidas de control* el microprocesador controla las operaciones de lectura y escritura en las memorias y puertos, habilitando el dispositivo deseado y especificando si es lectura o escritura. En lo que refiere a las *entradas de control*, a través de éstas el microprocesador puede recibir señales externas que le indiquen ciertas operaciones. Los distintos microprocesadores tienen diferentes entradas y salidas de control. A continuación se presenta un conjunto simplificado de entradas y salidas de control para nuestro microprocesador genérico.

Entradas y salidas de control

El microprocesador genérico tendrá las siguientes *entradas de control*:

Entradas de reloj. Muchos microprocesadores tienen dos patillas por medio de las cuales se conectan con las salidas de un cristal. Estas son comúnmente denominadas *X1* y *X2*, o también $\emptyset 1$ y $\emptyset 2$.

RESET. También llamado REINICIALIZACION, cuando esta entrada de control pasa a nivel bajo (en algunos microprocesadores esta entrada no está negada) el contador de programa se pone en 000011, o en una posición de reinicialización, buscando la primera instrucción de programa. En algunos casos, durante la reinicialización los buses de datos y de direcciones están en estado de *tres estados* (alta impedancia). El contenido de los registros internos también puede ser alterado durante una reinicialización, y esto depende del tipo de microprocesador. En algunos microprocesadores al activar y desactivar esta entrada se pasa a una rutina de servicio de interrupciones de reinicialización, contenida en alguna dirección específica.

Petición de interrupción. Cuando la entrada IRQ es activada con nivel bajo, el microprocesador procederá a un servicio de interrupción, dirigiéndose a una dirección de memoria donde se encuentra la subrutina destinada a este propósito. Esta interrupción es enmascarable.

Interrupción no enmascarable. Al activar con nivel bajo la entrada NMI el microprocesador responde de manera similar que con la entrada IRQ, yendo a la subrutina correspondiente de servicio de interrupción. Esta interrupción no se puede enmascarar.

Los distintos microprocesadores tienen diferentes tipos y número de interrupciones.

Petición de estado de espera. Algunos microprocesadores cuentan con entradas de control equivalentes a ésta, que se puede representar como **READY** y que proviene de un dispositivo periférico para informar al microprocesador que está listo para enviar o recibir datos. Si **READY** pasa a nivel bajo durante un ciclo de lectura o escritura, el microprocesador entra en *estado de espera*, donde permanecerá hasta que el periférico esté listo para transmitir o recibir datos, pasando **READY** a nivel alto, momento en el cual se procederá al ciclo de lectura/escritura. Esto es útil al usar memorias o periféricos demasiado lentos comparados con el microprocesador.

Mantenimiento. Al presentarse un nivel alto en la entrada **HOLD**, se completan las transferencias de datos en los buses; a continuación los buses de datos, de direcciones y las salidas de lectura/escritura para memorias y puertos se ponen en alta impedancia. De esta forma se notifica al microprocesador que otro dispositivo quiere utilizar los buses de direcciones y de datos para operaciones como el acceso directo a memoria (DMA). Así, no hay interferencia en las transferencias de datos en los buses. Si no se utiliza DMA esta entrada puede

conectarse a tierra. Algunos microprocesadores tienen entradas de control separadas para habilitar/inhabilitar el bus de datos y por otro lado el bus de direcciones junto con las salidas de lectura/escritura.

Generalmente, el microprocesador es el foco de todas las operaciones. Sin embargo, algunos controles (*control de mantenimiento*) liberan a los μP 's de los buses de direcciones y de datos durante el tiempo que un dispositivo periférico puede acceder directamente a la memoria principal del sistema sin ir a través del microprocesador. Esto se denomina *acceso directo a memoria (DMA)*.

Este microprocesador contará con las siguientes *salidas de control*:

Conocimiento de mantenimiento. Cuando la salida *HLDA* pasa a nivel alto, se está indicando que se ha recibido una petición de mantenimiento y que el microprocesador abandonará el control de los buses en el siguiente ciclo de reloj.

Control de lectura. El microprocesador genera una señal de control de lectura cuando activa con nivel bajo la salida \overline{RD} . En este momento está disponible para leer datos desde la memoria o un puerto de entrada.

Control de escritura. Cuando el microprocesador activa en nivel bajo la salida \overline{R} , quiere decir que ya colocó el dato en el bus de datos y que éste puede ser transferido a la memoria o a un puerto de E/S.

Status adicional. Esta salida de control, IO / \overline{M} , sirve para especificar si se está direccionando un dispositivo de E/S o de memoria por el microprocesador, ya sea para lectura o escritura.

Dependiendo del microprocesador, hay otros conjuntos de señales para las operaciones de lectura/escritura en memoria y puertos E/S, unos más complejos, en los que se manejan otras señales para especificar si se lee o se escribe ya sea en memoria o dispositivos de E/S, u otros más simples, en los que la lectura/escritura se coordina a través de una sola salida de control.

Líneas de control y buses internos

A través de las *líneas de control*, la unidad de control manda señales de control y temporización a las demás partes del microprocesador.

Dentro del microprocesador el flujo de datos y direcciones se da a través de los *buses internos*. Algunos microprocesadores comparten un bus interno de datos/direcciones y otros los tienen por separado.

Cerrosjos y buffers

Para el mantenimiento de datos y direcciones en sus respectivos buses y la habilitación de estados de alta impedancia o flotación, se cuenta con los *cerrosjos y buffers*. Los primeros sirven para el mantenimiento, y los últimos tienen salidas tri-estado para el propósito mencionado, y son controlados por la unidad de control.

3.1.5. OPERACIÓN DEL MICROPROCESADOR

El microprocesador es una máquina de proceso secuencial síncrona, que opera de forma cíclica; busca una instrucción del programa, la decodifica y la ejecuta realizando las operaciones pertinentes y transferencias de datos entre registros internos y externos, repitiendo este proceso hasta haber ejecutado todo el programa. La secuencia completa se sincroniza con un reloj maestro, es decir, la operación en general del microprocesador sigue la pauta marcada por los pulsos del reloj. De manera general, la frecuencia fundamental del reloj es generada por un cristal de cuarzo externo al μP . La circuitería de reloj interna utiliza esta frecuencia para

conectarse a tierra. Algunos microprocesadores tienen entradas de control separadas para habilitar/inhabilitar el bus de datos y por otro lado el bus de direcciones junto con las salidas de lectura/escritura.

Generalmente, el microprocesador es el foco de todas las operaciones. Sin embargo, algunos controles (*control de mantenimiento*) liberan a los μP 's de los buses de direcciones y de datos durante el tiempo que un dispositivo periférico puede acceder directamente a la memoria principal del sistema sin ir a través del microprocesador. Esto se denomina *acceso directo a memoria (DMA)*.

Este microprocesador contará con las siguientes *salidas de control*:

Conocimiento de mantenimiento. Cuando la salida *HLDA* pasa a nivel alto, se está indicando que se ha recibido una petición de mantenimiento y que el microprocesador abandonará el control de los buses en el siguiente ciclo de reloj.

Control de lectura. El microprocesador genera una señal de control de lectura cuando activa con nivel bajo la salida \overline{RD} . En este momento está disponible para leer datos desde la memoria o un puerto de entrada.

Control de escritura. Cuando el microprocesador activa en nivel bajo la salida \overline{R} , quiere decir que ya colocó el dato en el bus de datos y que éste puede ser transferido a la memoria o a un puerto de E/S.

Status adicional. Esta salida de control, IO / \overline{M} , sirve para especificar si se está direccionando un dispositivo de E/S o de memoria por el microprocesador, ya sea para lectura o escritura.

Dependiendo del microprocesador, hay otros conjuntos de señales para las operaciones de lectura/escritura en memoria y puertos E/S, unos más complejos, en los que se manejan otras señales para especificar si se lee o se escribe ya sea en memoria o dispositivos de E/S, u otros más simples, en los que la lectura/escritura se coordina a través de una sola salida de control.

Líneas de control y buses internos

A través de las *líneas de control*, la unidad de control manda señales de control y temporización a las demás partes del microprocesador.

Dentro del microprocesador el flujo de datos y direcciones se da a través de los *buses internos*. Algunos microprocesadores comparten un bus interno de datos/direcciones y otros los tienen por separado.

Cerrojos y buffers

Para el mantenimiento de datos y direcciones en sus respectivos buses y la habilitación de estados de alta impedancia o flotación, se cuenta con los *cerrojos y buffers*. Los primeros sirven para el mantenimiento, y los últimos tienen salidas tri-estado para el propósito mencionado, y son controlados por la unidad de control.

3.1.5. OPERACIÓN DEL MICROPROCESADOR

El microprocesador es una máquina de proceso secuencial síncrona, que opera de forma cíclica; busca una instrucción del programa, la decodifica y la ejecuta realizando las operaciones pertinentes y transferencias de datos entre registros internos y externos, repitiendo este proceso hasta haber ejecutado todo el programa. La secuencia completa se sincroniza con un reloj maestro, es decir, la operación en general del microprocesador sigue la pauta marcada por los pulsos del reloj. De manera general, la frecuencia fundamental del reloj es generada por un cristal de cuarzo externo al μP . La circuitería de reloj interna utiliza esta frecuencia para

realizar su función. En los primeros μP 's, se requería un chip de reloj externo. En algunos casos se puede recibir la frecuencia fundamental de otros dispositivos como circuitos RC y en otros se puede sincronizar directamente la señal de reloj con una señal externa, lo cual es opcional en algunos microprocesadores. Pensemos en la opción más generalizada, que es la de conectar un cristal a las entradas de reloj X1 y X2. El reloj puede generar un sólo tren de pulsos, siendo entonces un reloj monofásico. También se puede tener un reloj bifásico, caso bastante común, donde se tienen dos fases, o una circuitería con más fases. Las *fases de reloj* se denotan comúnmente con la letra ϕ , y pueden ser solapadas o sin solape. Esto se ilustra en la figura 3.5.

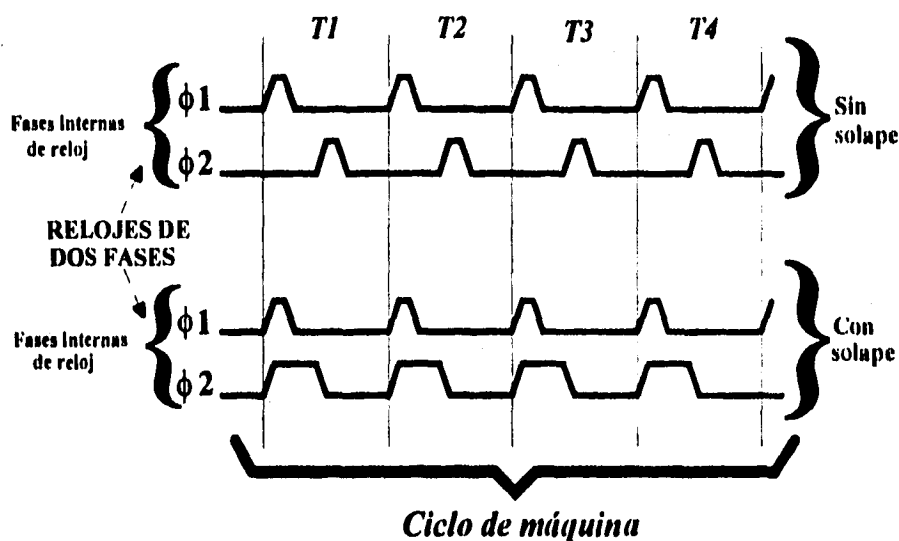


Figura 3.5.

Usualmente la operación del μP sigue una de las entradas de reloj, o si esto produce un reloj de más de una fase, a una de las fases de reloj.

El intervalo de tiempo entre dos pulsos de reloj adyacentes se denomina *período*. En la figura 3.5 se aprecia que las señales de reloj se dividen en *estados T* (T_1 , T_2 , etc.). Para este caso, cada estado consta de un período. En ciertos microprocesadores un estado puede estar constituido por dos períodos. En esta figura también se aprecia una división de tiempo mayor denominada *ciclo de máquina* y que para este caso consta de cuatro estados. Los ciclos de máquina se asocian a acciones del microprocesador, y de este modo hay distintos tipos de ciclos de máquina. La longitud del ciclo de máquina depende del tipo del mismo y del microprocesador del que se trate. En algunos μP 's todos los ciclos de máquina son de igual tamaño. Algunos tipos comunes de ciclo de máquina son:

1. Lectura (busca un código de operación).
2. Lectura de memoria o puerto E/S.
3. Escritura en memoria o puerto E/S.
4. Ejecución de una operación interna.

Una *fase del ciclo de máquina* (no confundir con fase de reloj) es una porción del mismo que puede identificarse con una acción definida. Hay fases de uno y, normalmente, de más períodos. Al ciclo de máquina lo constituyen varias fases.

Un *ciclo de instrucción* es el tiempo necesario para que el microprocesador busque una instrucción de la memoria de programa, la decodifique y realice todas las operaciones necesarias para ejecutarla. Hay ciclos de instrucción formados por varios ciclos de máquina, pero se dan los casos en que es suficiente un

ciclo de máquina para llevar a cabo la *secuencia direccionar, decodificar, ejecutar*. A su vez, hay instrucciones de más de un byte, y en la mayoría de los casos, es necesario un ciclo de máquina adicional para pasar cada byte de la memoria de programa al microprocesador, es decir, para procesar cada byte de la instrucción. Por todo esto, el tiempo del ciclo de instrucción es variable. El primer ciclo de máquina de toda instrucción se encarga de la búsqueda y lectura del código de operación.

La secuencia mencionada se divide en dos ciclos: *el ciclo Fetch, o ciclo de búsqueda*, en el cual se direcciona la instrucción llevándola al microprocesador, donde se decodifica, lo que indica a la unidad de control las operaciones que debe realizar; en el caso de instrucciones de 2 o 3 bytes, éstos deben ser traídos de la memoria. Se prosigue con el *ciclo de ejecución*, tiempo en el cual la unidad de control genera los impulsos necesarios para ejecutar la instrucción. En general, la diferencia entre las distintas instrucciones radica en el ciclo de ejecución. Todo esto se ilustra en las figuras 3.6a y 3.6b.



Figura 3.6a. Diagrama de flujo del microprograma de una instrucción.

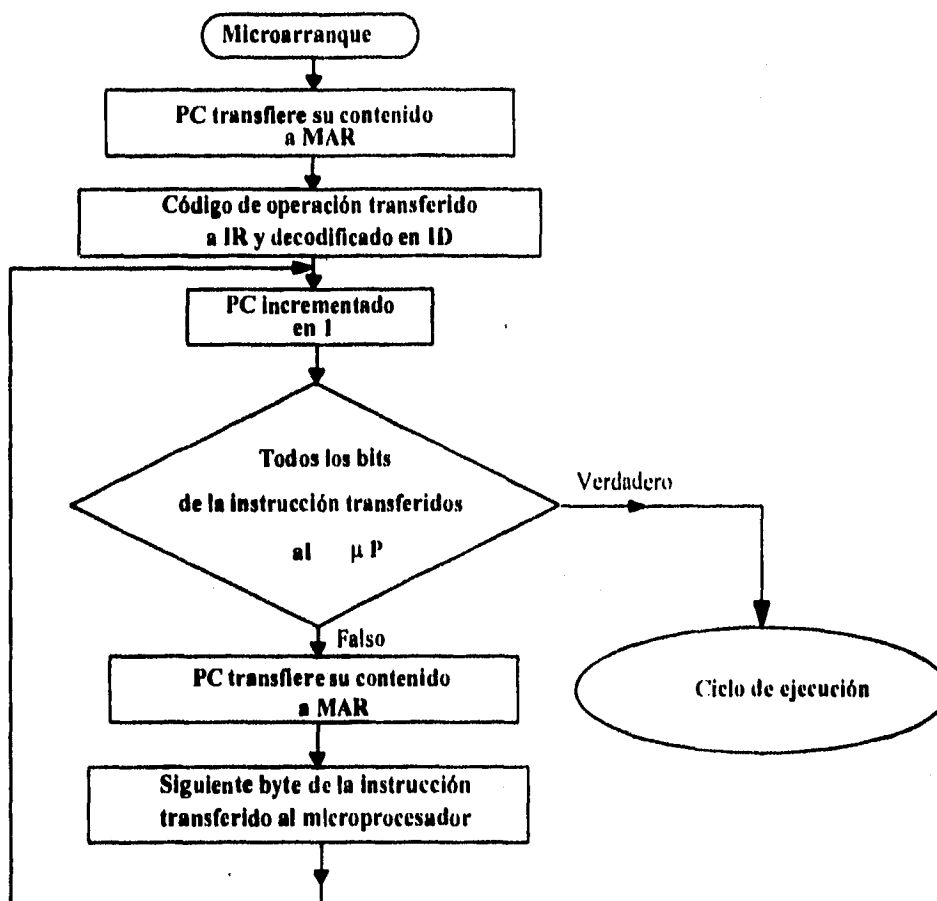


Figura 3.6b. Diagrama de flujo del Ciclo fetch.

Un ejemplo ilustrará, a continuación, estos conceptos.

Ejemplo 3.1. Se presentará el proceso seguido por el microprocesador en el caso de la instrucción de cargar el acumulador con el dato contenido en la posición de memoria especificada por los bits 2 y 3 de la instrucción. El primer bit contiene el código de operación, como en todas las instrucciones. Esta es una instrucción de 3 bytes y requiere 4 ciclos de máquina para realizarse. La instrucción se puede representar como sigue:

$$A \leftarrow (M)$$

*Los paréntesis quieren decir "contenido del registro o posición de memoria".

De manera común, las características de esta instrucción para nuestro microprocesador genérico serán:

LDA addr Cargar acumulador directo (load accumulator direct)		
A ← (byte3, byte2)		
El contenido de la posición de memoria cuya dirección está especificada en los bytes 2 y 3 de la instrucción es transferido al registro A (acumulador)		
Código binario	00111010	byte 1 (opcode)
	byte inferior de dirección	byte 2
	byte superior de dirección	byte 3
Código hexadecimal	3AH	

*La notación hexadecimal se indica con una "H" posterior al número.

Otros detalles que se usarán en el ejemplo son:

- La instrucción de 3 bytes mencionada estará almacenada en las primeras posiciones de la memoria, sean éstas 0000H, 0001H y 0002H (esta parte de la memoria corresponde a la memoria de programa, ROM para este caso).
- El dato a cargar en A será 42H (01000010 binario).
- Este dato está guardado en la posición de memoria 21AAH (memoria de datos, RAM).

A continuación se presentan los pasos para llevar a cabo el ciclo fetch:

No	Operación requerida	Resultados
1.	Inicializar el PC.	0000H → PC
2.	El contenido del PC pasa al MAR donde se mantiene durante la referencia a memoria.	0000H → MAR
3.	El contenido del MAR pasa al bus de direcciones.	0000H → bus de direcciones
4.	El contenido de la posición de memoria 0000H, el código de operación, se coloca en el bus de datos.	3AH → bus datos
5.	El contenido del bus de datos se carga en el cerrojo de datos, de donde pasa al IR.	3AH → cerr datos → IR
6.	El contenido del IR pasa al ID donde es decodificado.	IR → ID
7.	La unidad de control reconoce el tipo de instrucción, se da cuenta de que se requieren otros tres ciclos de máquina, y la lectura de otros dos bytes, entonces genera las señales apropiadas.	
8.	Se incrementa en 1 el PC.	0001H → PC
9.	El contenido del PC pasa al MAR donde se mantiene durante la siguiente referencia a memoria.	0001H → MAR
10.	El contenido del MAR pasa al bus de direcciones.	0001H → bus de direcciones
11.	El contenido de la posición de memoria 0001H se coloca en el bus de datos.	AH → bus datos
12.	El contenido del bus de datos pasa por el cerrojo de datos hacia el registro temporal Z.	AH → cerrojo datos → Z
13.	Se incrementa en 1 el PC.	PC → 0002H
14.	El contenido del PC pasa al MAR donde se mantiene durante la siguiente referencia a memoria.	0002H → MAR
15.	El contenido del MAR pasa al bus de direcciones.	0002H → bus de direcciones
16.	El contenido de la posición de memoria 0002H se coloca en el bus de datos.	21H → bus datos
17.	El contenido del bus de datos pasa por el cerrojo de datos hacia el registro temporal W.	21H → cerrojo datos → W
18.	Se incrementa en 1 el PC.	PC → 0003H

Al terminar el ciclo fetch, el estado de los registros del μP se encuentra como sigue:

- El IR mantiene en forma binaria el código de operación 3AH.
- El par de registros temporales WZ contiene la dirección de 16 bits (21AAH) de la posición de memoria donde está el dato que será cargado en el acumulador.
- El PC apunta a la siguiente instrucción.

Ahora se presentan los pasos necesarios para el ciclo de ejecución.

No	Operación requerida	Resultados
1.	El contenido del par de registros WZ pasa a MAR, donde se mantiene durante la referencia a memoria.	21AAH → MAR
2.	El contenido de MAR pasa al bus de direcciones.	21AAH → bus de direcciones
3.	El contenido de la posición de memoria 21AAH es leído y pasa al bus de datos.	4211 → bus de datos
4.	El contenido del bus de datos pasa por el cerrojo de datos hacia el acumulador.	4211 → cerrojo datos → A

Con esto se ha implementado la instrucción mencionada. A continuación se presenta un análisis de ciclos de máquina durante el desarrollo de esta instrucción.

Como se dijo antes, ésta es una instrucción que requiere de 4 ciclos de máquina para realizarse. El primer ciclo de máquina, correspondiente a la búsqueda del código de operación, estará formado por cuatro estados, de un período o pulso de reloj, cada uno. En el cuarto estado de este ciclo se decodifica el opcode. De este modo, el primer ciclo de máquina abarca desde el paso 1 al paso 6 del ciclo fetch. El siguiente ciclo de máquina está compuesto por tres estados y corresponde a la primera lectura que realiza el μP ; aquí se llevan a cabo desde el paso 7 hasta el paso 12 del ciclo fetch. Durante el tercer ciclo de máquina, de tres estados, se lleva a cabo la segunda lectura; esto sucede entre los pasos 13 y 18 del ciclo fetch. También consistente en tres estados, el último ciclo de máquina comprende el ciclo de ejecución de la instrucción, donde se hace la lectura de memoria de datos y se transfiere al acumulador. En la figura 3.7 se aprecia esta secuencia.

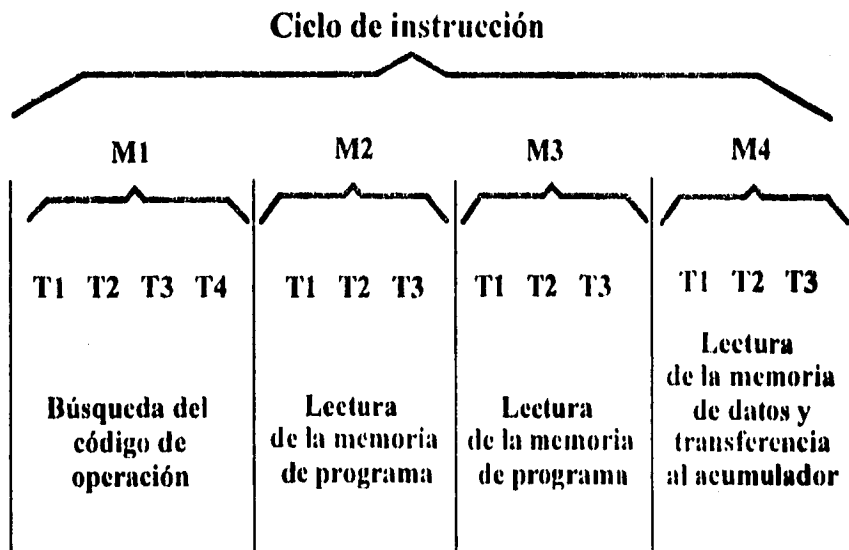


Figura 3.7.

3.1.6. MODELO DE PROGRAMACIÓN

El *modelo de programación* está comprendido por los registros disponibles al programador, es decir, por los registros cuyos contenidos pueden ser modificados usando las instrucciones disponibles en el repertorio de instrucciones de un microprocesador específico. En la figura 3.8a se aprecia el modelo de programación para nuestro microprocesador genérico.

PSW (8 bits)	A (8 bits)
B (8 bits)	C (8 bits)
D (8 bits)	E (8 bits)
H (8 bits)	L (8 bits)
SP (16 bits)	
PC (16 bits)	

Figura 3.8a.

Hemos llamado PSW al registro de status del microprocesador genérico. En la figura 3.8b se aprecian los señalizadores que contiene.

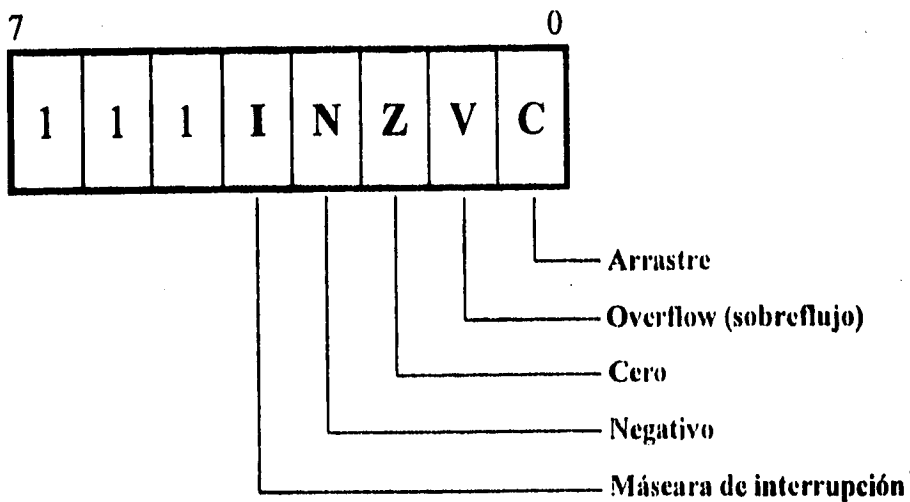


Figura 3.8b.

3.2. MEMORIA

3.2.1. ¿QUE ES LA MEMORIA?

La memoria es como un almacén con muchos estantes donde se guardan los bytes que el microprocesador utiliza. Estos estantes son las localidades de memoria.

La memoria está obligada a hacer trabajar al microprocesador. En la memoria deben estar todos los bytes que el microprocesador de la computadora o del sistema con microprocesador necesita para operar. Aquí se mantienen los datos primarios que requieren ser procesados y los resultados del procesamiento. La memoria también puede ser un canal de comunicación entre el microprocesador y sus periféricos. Hay muchos tipos de memoria descritos y delimitados por función y tecnología.

En la memoria se guardan programas y datos de uso constante y futuro. Sin este lugar de almacenamiento el microprocesador es inútil, pues sus registros internos sólo pueden mantener almacenados unos pocos bytes. La memoria pone cientos, miles y hasta millones de bytes a la disposición del microprocesador, suficientes para largas listas de instrucciones de programa o grandes bloques de memoria (como datos).

3.2.2. ROM Y RAM

El flip-flop, o cerrojo, es la celda básica de memoria utilizada en muchas memorias semiconductoras. Las memorias semiconductoras habitualmente se dividen en dos grupos: memorias de lectura/escritura o de acceso aleatorio (RAM) y memorias de sólo lectura (ROM). Los nombres implican las diferencias entre los dos tipos de memoria. Una ROM es una memoria no volátil con su patrón binario de ceros y unos programados permanentemente por el fabricante. La memoria de lectura/escritura es una memoria que puede ser fácilmente programada, borrada y reprogramada por el usuario. La programación se denomina "escribir" en memoria. Copiar datos de la memoria, sin destruir el contenido, se denomina "leer" de memoria. La memoria de lectura/escritura se denomina RAM (memoria de acceso aleatorio), con frecuencia. Generalmente, el programa de la RAM es volátil, lo cual significa que se perderá si se desconecta la alimentación del circuito integrado (CI), incluso durante un instante. Los sistemas microcomputadores normalmente contienen los dos tipos de memorias semiconductoras ROM y RAM. Las posiciones de almacenamiento ROM y RAM están habitualmente en CI's separados.

Existen actualmente cuatro posibilidades de memorias de sólo lectura. La ROM estándar es programada por el fabricante. La PROM (memoria de sólo lectura programable) puede ser programada permanentemente por el usuario o distribuidor utilizando un equipo especial. Puede ser programada solamente una vez. La EPROM (memoria de sólo lectura programable borrable) puede ser programada y borrada por el usuario. Los datos almacenados en la EPROM pueden ser borrados aplicando luz ultravioleta de alta intensidad, a través de una ventana especial transparente, en la parte superior del CI. Otra PROM borrable es la EAROM (memoria de sólo lectura alterable eléctricamente) que puede ser borrada y programada por el usuario con un equipo especial. La EAROM se borra eléctricamente y no con luz ultravioleta. La ROM, PROM, EPROM y EAROM son consideradas memorias no volátiles permanentes y no pierden sus datos cuando se desconecta la alimentación del CI.

Las memorias RAM se subdividen en dos grupos. Si la memoria de lectura/escritura contiene circuitos tipo "flip-flop" como celdas de memoria, se denomina RAM estática (SRAM). Una forma más simple de la celda de memoria de lectura/escritura es la base de la RAM dinámica (DRAM). Debido a que la celda de memoria de la RAM dinámica está basada en una capacitancia, las celdas deben ser refrescadas cientos de veces por segundo. La RAM estática no necesita refrescarse y mantiene su información binaria.

indefinidamente, todo el tiempo que esté alimentado el CI. Las RAM dinámicas tienen mayor capacidad y menos consumo de potencia que las estáticas.

Un dispositivo de memoria semiconductor más moderno es la RAM no volátil (NVRAM). La NVRAM combina RAM estáticas y PROM borrables eléctricamente (EEPROM, también conocidas como E²PROM) de idéntica capacidad. Los datos pueden ser transferidos entre la SRAM y la E²PROM y viceversa mediante operaciones de almacenamiento y rellamada. Los datos pueden almacenarse con seguridad en la EEPROM no volátil cuando se desconecta la alimentación. Cuando se conecta la alimentación, los datos de la EEPROM son rellamados automáticamente a la sección SRAM de la RAM no volátil. El concepto de RAM no volátil también está implementado por los fabricantes de microcomputadoras utilizando pastillas comunes de RAM con una batería de seguridad.

En microcomputadoras, las RAM se utilizan para almacenamiento temporal de programas y datos del usuario. Las ROM son utilizadas más frecuentemente para almacenar instrucciones del lenguaje máquina que pueden considerarse el "programa monitor". El programa monitor puede contener rutinas de inicialización no modificables, rutinas de entrada/salida y algoritmos aritméticos.

En microprocesadores, la RAM se utiliza normalmente como memoria de datos y la ROM como memoria de programa; es común nombrarlas de esta manera.

3.2.3. MAPAS DE MEMORIA

El *mapa de memoria* de un microprocesador define la ubicación de los diferentes grupos que forman la memoria que utiliza el μ P, como son la memoria de datos RAM, el stack (que forma parte de la última), la memoria de programa ROM, puertos de entrada/salida y áreas de memoria que no se utilizan (el acceso a esta área producirá resultados impredecibles ya que no está almacenado ningún programa ni ningún dato, pues para muchas de las aplicaciones de los μ P's no es necesario usar todas las direcciones posibles del sistema respecto de las celdas de su memoria). Los distintos microprocesadores presentan diferentes distribuciones para su mapa de memoria. En los casos de ROM y RAM, en los sistemas con microprocesador se puede tener uno o varios chips de cada uno; la dirección dentro del mapa de memoria identifica de qué chip se trata. De manera general, la representación hexadecimal del byte más significativo o de orden superior de la dirección de memoria es conocida como número de página. Para seleccionar el chip deseado de memoria o de E/S, es común que los sistemas con μ P utilicen un *decodificador de direcciones*. A este dispositivo se conectan las líneas de mayor orden del bus de direcciones del μ P, de modo que las salidas del decodificador sirven para seleccionar el chip deseado dependiendo de su ubicación en el mapa de memoria del μ P. En la figura 3.13 se aprecia el mapa de memoria de un microprocesador genérico.

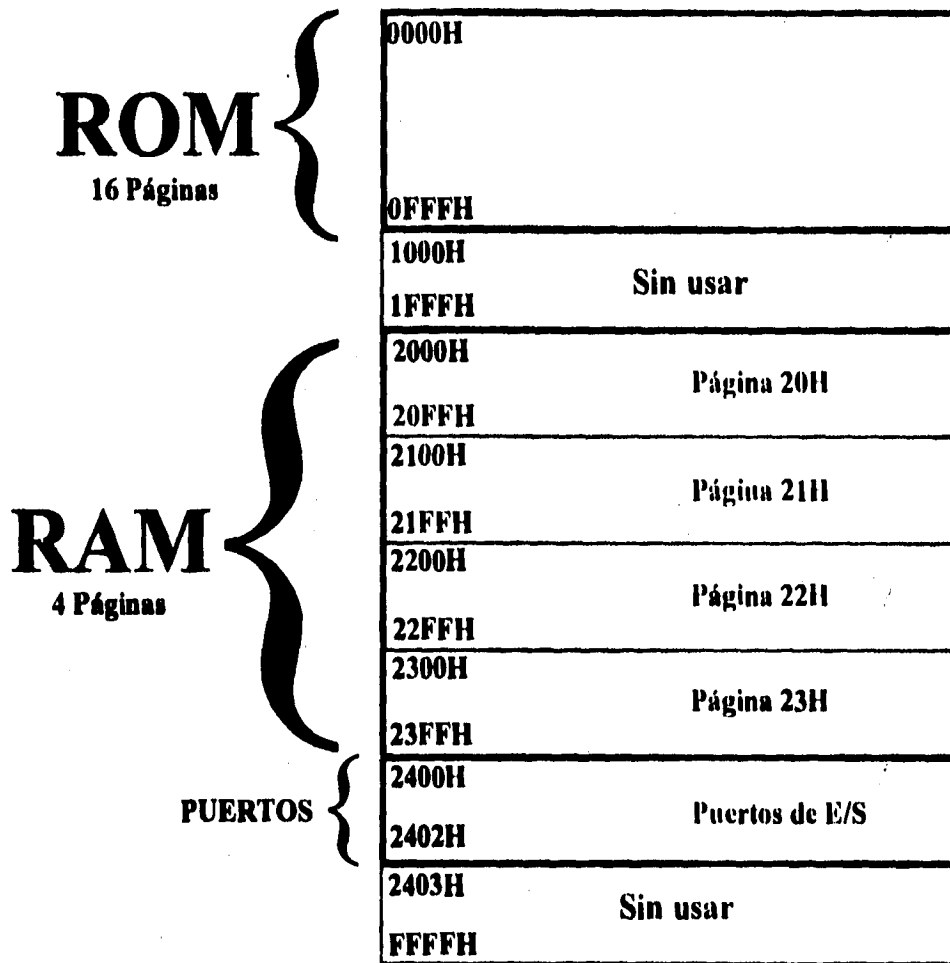


Figura 3.13.

3.2.4. EL STACK

El *stack*, también conocido como *pila*, se utiliza principalmente al implementar subrutinas o cuando se presenta una interrupción (la cual llama a algún tipo de subrutina de servicio de interrupción). La utilidad principal de un stack consiste en almacenar el contenido del contador de programa al presentarse un salto a subrutina, para que al finalizar ésta, sea posible regresar a la ejecución normal del programa; en el caso de una interrupción se suelen almacenar además los contenidos de otros registros que podrían ser alterados durante la subrutina de servicio de interrupción. Un stack es un conjunto de registros organizado de tal manera que el último dato escrito es el primero disponible para ser leído. Esto se conoce como *LIFO* (last in-first out). El único registro accesible para lectura o escritura en un momento dado es la *cabecera del stack* (tope del stack, o tope de la pila). La escritura de información en la cabecera del stack recibe el nombre de operación *push*. Cuando se lee información desde la cabecera del stack la operación se llama *pop*. Existen dos tipos de stacks, los cuales se describen a continuación.

El stack en cascada

Algunos microprocesadores internamente contienen un conjunto de registros conocido como *stack en cascada*, en el cual se tiene entrada paralela/salida paralela. Cuando se "empujan" datos en un stack en cascada, el contenido de cada registro se desplaza al registro de posición más baja. Un stack de este tipo es de longitud finita, es decir, el número de registros que lo forman, lo que se define como *profundidad del stack*, es limitado (típicamente entre 4 y 16 palabras). Es por esto que el dato en el registro del fondo se pierde después de cada push. Una operación pop lee el contenido del registro de cabecera del stack y el contenido de cada registro se desplaza hacia arriba hacia el siguiente registro de posición más elevada. Por convenio, el registro del fondo se borra después de cada operación pop. Cuando se ejecuta una instrucción de llamada a subrutina, tienen lugar los siguientes sucesos:

1. El contenido del contador de programa se incrementa normalmente.
2. El contenido del contador de programa se transfiere al registro más alto del stack.
3. La dirección de inicio de la subrutina, que está contenida en la propia instrucción de llamada a subrutina, se transfiere al contador de programa.
4. Se ejecuta la subrutina.
5. Después de completar la subrutina, el control se devuelve al programa principal con una instrucción de retorno de subrutina, transfiriendo el contenido de registro más alto al contador de programa.
6. El programa principal ejecuta su operación normal.

Si la primera subrutina A llama a una segunda subrutina B, se deben memorizar dos direcciones. La dirección de retorno al programa principal desde la subrutina A se memoriza primero en el registro superior del stack. Cuando aparece el segundo salto a la subrutina B, se inserta en el registro superior del stack la dirección de retorno a la subrutina A y la primera dirección de retorno baja en cascada al siguiente registro del stack. Este proceso se realiza mediante una operación push.

Cuando se completa la ejecución de la subrutina B, se consigue el retorno a la subrutina A transfiriendo el contenido del registro superior del stack al contador de programa. Simultáneamente, el contenido del siguiente registro del stack sube en cascada al registro superior del stack. Esto se realiza mediante una operación pop. Después de completar la ejecución de la subrutina A, el contenido del registro superior del stack se transfiere al contador de programa y puede continuar la ejecución del programa principal.

Para los μP 's con stack en cascada la operación push-pop es automática pues el programador no debe introducir en el programa ninguna instrucción especial para que se realice. En el siguiente ejemplo se muestra la operación de un stack en cascada.

Ejemplo 3.2. El programa principal de un microprocesador con un stack en cascada cuyos archivos son de 16 bits, va de la dirección 0000H a la 09F1H. Hay una subrutina C llamada por una subrutina B que a su vez es llamada por una subrutina A. Las direcciones de inicio y final son las siguientes:

Subrutina A → 0AB1H a 0B00H
Subrutina B → 0C0DH a 0D11H
Subrutina C → 0E11H a 0F4CH

Las direcciones de las instrucciones de llamada a subrutina en los distintos programas son:

Subrutina A → 08D2H
Subrutina B → 0ABDH
Subrutina C → 0CC8H

El contenido del stack después de cada operación de push y pop se puede mostrar en la tabla de la figura 3.9, (donde las direcciones son hexadecimales). Durante una instrucción de llamado a subrutina se

ejecuta una operación push y el stack contiene la dirección de retorno, que es la dirección donde se encuentra dicha instrucción incrementada en 1.

Número de registro en el stack	Operación push			Operación pop		
	Salto a subrutina			Retorno de subrutina		
	A	B	C	A	B	Programa principal
4	08D3	0ABE	0CC9	0ABE	08D3	0000
3	0000	08D3	0ABE	08D3	0000	0000
2	0000	0000	08D3	0000	0000	0000
1	0000	0000	0000	0000	0000	0000

Figura 3.9.

El inconveniente del stack en cascada es su tendencia a perder las direcciones situadas en la parte inferior, por lo que el programador debe observar estrictamente la limitación de la profundidad del stack. Además, el número de subrutinas que pueden llamar a otras queda limitado por la citada profundidad del stack.

El stack tipo puntero

El *stack tipo puntero* consiste en tener una parte de la memoria RAM acondicionada como stack. En este, que es el caso de nuestro microprocesador genérico, puede pensarse que el dato colocado en el stack permanece en una posición fija mientras que el stack se "mueve" —se expande o contrae— en función de la ejecución de instrucciones que impliquen operaciones push y pop. Esta implementación permite que el stack aumente si es necesario con la condición de que no exceda los límites de la RAM. Un registro interno, el *stack pointer* (*puntero o apuntador de pila*), abreviado como *SP*, que es en realidad un contador positivo/negativo, apunta en todo momento a la cabecera del stack. Esto se aprecia en la figura 3.10. Para el µP genérico analizado, el SP es un registro de 16 bits formado por dos registros de 8 bits, uno para el byte superior y otro para el byte inferior.

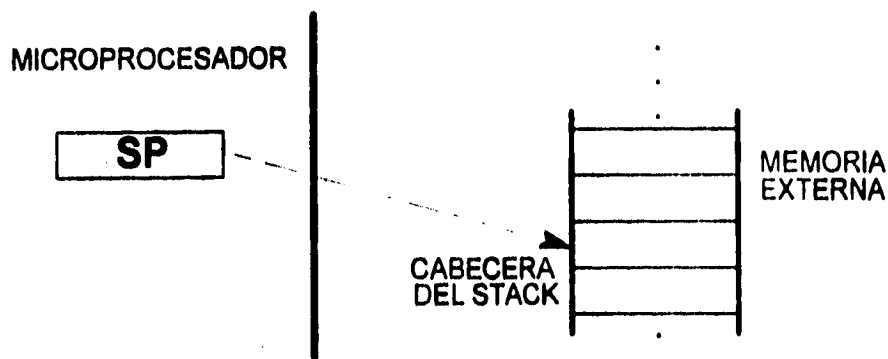


Figura 3.10.

En otros microprocesadores el SP apunta a una posición de memoria vacía (superior o inferior) "antes" de la cabecera del stack. Esto depende del microprocesador en especial, pues en unos casos una operación push mueve el stack pointer a una posición superior de memoria (se incrementa) y en otros μP 's lo mueve a una dirección inferior (se decrementa). El inicio del área de memoria dispuesta para el stack se especifica mediante la carga del stack pointer con un valor inicial. Esta especificación es necesaria debido a que los distintos sistemas tienen diferentes configuraciones de memoria. El stack pointer se inicializa mediante la instrucción de *carga de par de registros inmediato*, especificando la pareja de registros SP, que como ejemplo utilizaría el mnemotécnico:

LXI,data 16

Esta instrucción carga el SP con un dato de 16 bits (para el μP genérico de 8 bits), que representa la dirección de inicialización del stack. El stack pointer debe inicializarse antes de que se ejecute cualquier instrucción que utilice el stack. Por eso, la inicialización del stack es una de las primeras instrucciones en un programa. En los μP 's que decrementan el stack pointer al colocar datos en el stack, es decir, que ejecutan instrucciones que implementan operaciones push expandiendo el stack hacia posiciones de memoria con valores de dirección más bajos, el stack pointer se inicializa comúnmente hacia la posición de memoria RAM más alta disponible. Este es el caso de nuestro μP genérico.

Cuando el programador desea colocar el stack pointer a un valor calculado por el programa, este valor se coloca en H y L y se mueve a continuación al stack pointer, utilizando la instrucción *mover HL a SP*:

SPHL (SP) \leftarrow (H)(L)

*Los paréntesis quieren decir "contenido del registro o posición de memoria".

Hay varias instrucciones que provocan operaciones push y pop. Si aparece una interrupción, se producen operaciones push al guardar contenidos de registros en el stack; esto sucede también con una instrucción de llamada a subrutina cuyo mnemotécnico sería por ejemplo LCALL, y que guarda el contenido del PC. Una instrucción de retorno de interrupción produce varias operaciones pop, una por cada registro que se saca del stack. La instrucción de retorno de subrutina produce una operación pop. En todos estos casos, las operaciones push y pop son automáticas. Con las instrucciones PUSH (introducir o introducción) y POP (sacar) se provocan operaciones push y pop "manuales" y se puede usar el stack como área de almacenamiento temporal de datos. Por ejemplo, el contenido de las parejas de registros internos (BC, DE, o HL) se guardan mediante la instrucción PUSH.

PUSH rp ((SP) - 1) \leftarrow (rh) ((SP) - 2) \leftarrow (rl) (SP) \leftarrow (SP) - 2

rh = registro superior o byte superior

rl = registro inferior o byte inferior

PUSH transfiere el contenido de mayor peso de la pareja, rp, a la posición de memoria cuya dirección es una menos que el valor inicial del stack pointer y el contenido del registro de menor peso a la dirección de memoria cuya dirección es dos menos que su valor inicial. El stack pointer se deja con un valor dos unidades menor que su valor inicial. Como se observa, las instrucciones del stack del μP genérico ocasionan la transferencia de dos bytes de datos. Así, con esta implementación, la cabecera del stack consiste realmente en dos registros. Una operación push de este tipo se aprecia en la figura 3.11:

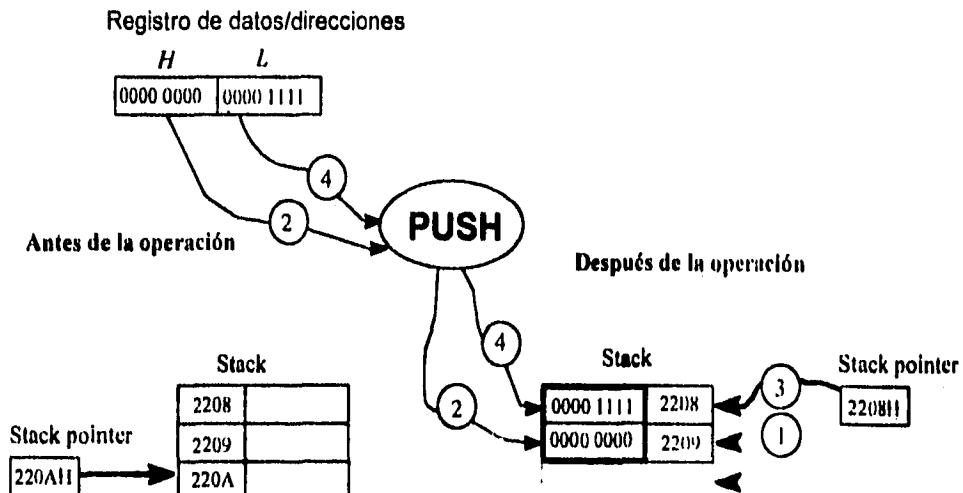


Figura 3.11.

Los números de las direcciones están en notación hexadecimal y los datos en notación binaria. Antes de esta instrucción PUSH el SP estaba en 220AH. Los números dentro de los círculos indican el orden de los pasos seguidos al ejecutarse una instrucción PUSH. Esta secuencia de acciones es la siguiente:

1. El SP del μP se decrementa de 220AH a 2209H.
2. El SP señala al área de memoria 2209H sobre el bus de direcciones del sistema y el byte de orden superior (0000 0000₂) es introducido en el stack.
3. El SP es decrementado, de nuevo, de 2209H a 2208H.
4. El SP señala al área de memoria 2208H sobre el bus de direcciones del sistema, y el byte de orden inferior (0000 1111₂) del registro de datos/direcciones es almacenado en el stack.

Los datos se transfieren del stack a la pareja de registros del microprocesador mediante la instrucción POP:

POP rp
(rl) ← ((SP))
(rh) ← ((SP + 1))
(SP) ← (SP + 2)

POP es la inversa de PUSH. No obstante, de manera diferente al stack en cascada, retirar elementos de un stack implementado en RAM no elimina datos del stack; simplemente los copia en una pareja de registros interna. Pero las posiciones de memoria, de las cuales se copian los datos, se consideran vacías y cuando ocurre una operación posterior de push, dichas posiciones se reescriben. Una operación pop producida por una instrucción POP de este tipo se ilustra en la figura 3.12:

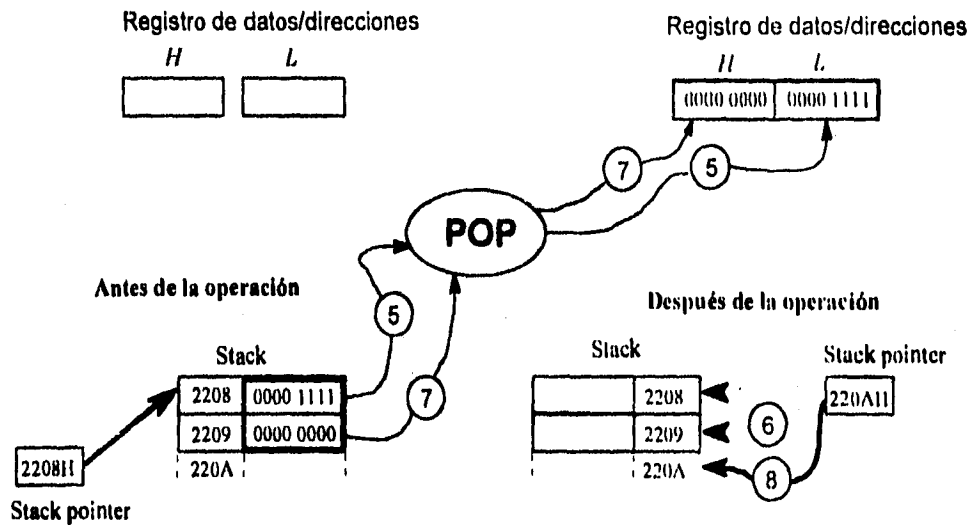


Figura 3.12.

La secuencia de eventos indicada por los números dentro de los círculos para esta instrucción POP es:

5. El SP direcciona la cabecera del stack (posición 2208H). Es sacado el contenido de esta posición y devuelto al byte de orden inferior del registro par HL.
6. El stack pointer es incrementado a 2209H.
7. El SP direcciona el tope del stack, que ahora es la posición 2209H. El contenido (0000 0000₂) es devuelto al byte de orden superior del registro par HL.
8. Finalmente, el stack pointer es incrementado de 2209H a 220AH en preparación para la siguiente instrucción de introducción (PUSH) o de llamada a subrutina (LCA1.1.).

Otra instrucción, la de intercambiar cabecera del stack con H y L, intercambia el dato en la cabecera del stack con el contenido de los registros H y L:

XHTL
(L) ↔ ((SP))
(H) ↔ ((SP + 1))

No cambia el valor del SP y por lo tanto, permite el acceso al contenido del stack sin cambiar la posición del SP o perder el contenido de H y L.

3.2.5. REPERTORIO DE INSTRUCCIONES

También llamado *conjunto de instrucciones*, el *repertorio de instrucciones* es el grupo de las mismas que un microprocesador específico puede ejecutar. Los repertorios de instrucciones de los microprocesadores pueden variar desde ocho hasta 200 instrucciones básicas. Los repertorios de instrucciones no están estandarizados, debido al individualismo de cada fabricante y a la diferencia de arquitectura y uso de los microprocesadores.

Las instrucciones de un repertorio pueden clasificarse de diversas formas. De manera genérica, se enumerarán las siguientes categorías:

1. Instrucciones aritméticas.
2. Instrucciones lógicas.

3. Instrucciones de transferencia de datos.
4. Instrucciones de bifurcación.
5. Instrucciones de llamada a subrutina.
6. Instrucciones de vuelta.
7. Instrucciones misceláneas.

Un sencillo repertorio de instrucciones de un microprocesador debería incluir las siguientes instrucciones aritméticas:

1. Sumar.
2. Restar.
3. Incrementar.
4. Decrementar.
5. Comparar.
6. Negar.

Otras instrucciones aritméticas utilizadas por algunos microprocesadores pueden incluir operaciones de suma con arrastre, resta con préstamo, multiplicación y división.

El mismo microprocesador simplificado debería tener las siguientes instrucciones lógicas:

1. AND.
2. OR.
3. OR exclusiva.
4. Not.
5. Desplazamiento a la derecha.
6. Desplazamiento a la izquierda.

Otras instrucciones lógicas utilizadas por algunos microprocesadores pueden incluir desplazamiento aritmético a la derecha, desplazamiento circular a la derecha, desplazamiento circular a la izquierda, desplazamiento circular a la derecha a través del arrastre, desplazamiento circular a la izquierda a través del arrastre y operación de test (prueba).

El microprocesador básico debería contener variaciones de las siguientes instrucciones de transferencia de datos:

1. Cargar.
2. Almacenar.
3. Transferir.
4. Entrar.
5. Salir.

Otras instrucciones de transferencia de datos utilizadas por algunos microprocesadores pueden incluir intercambio y diversas instrucciones de borrado e inicialización.

El microprocesador debería contener las siguientes instrucciones de bifurcación en su repertorio de instrucciones:

1. Bifurcación incondicional.
2. Bifurcación si cero.
3. Bifurcación si no cero.
4. Bifurcación si igual.
5. Bifurcación si no igual.
6. Bifurcación si positivo.
7. Bifurcación si negativo.

Otras instrucciones de bifurcación condicional utilizadas por algunos microprocesadores pueden depender de condiciones como mayor que o menor que, no arrastre o arrastre, "no overflow" u "overflow". Las operaciones de bifurcación son instrucciones de toma de decisiones.

Un sencillo microprocesador debería tener una instrucción de llamada a subrutina (referencia como CALL) para hacer que el programa bifurque o salte a un grupo especial de instrucciones que realizan una tarea específica. Todos los microprocesadores tienen la instrucción de llamada incondicional, y algunos también tienen instrucciones de llamada condicional. Las instrucciones de llamada condicional pueden incluir llamada si cero, llamada si no cero, llamada si positivo, llamada si no positivo, etc.

Al final de la subrutina el programa debe volver a donde originalmente abandonó el programa principal. Esta tarea la ejecuta una instrucción de vuelta. Las instrucciones de vuelta (retorno) pueden incluir el retorno de una subrutina o de las operaciones de interrupción. Las vueltas son habitualmente incondicionales, pero algunos microprocesadores contienen instrucciones de vuelta condicional.

Un repertorio de instrucciones simplificado de un microprocesador debería incluir las siguientes instrucciones misceláneas:

1. No operación.
2. Introducir.
3. Sacar.
4. Esperar.
5. Alto.

Otras instrucciones misceláneas pueden incluir habilitar interrupción, inhabilitar interrupción, ruptura y ajuste decimal.

3.2.6. MODOS DE DIRECCIONAMIENTO

Las instrucciones del microprocesador

Una *instrucción* es un "comando" al microprocesador para que éste realice una tarea específica o actividad. Una serie de tales instrucciones recibe el nombre de *programa*. El *código de operación* u *opcode* es la primera parte de cada instrucción, y al ser decodificado por el decodificador de instrucción en el μP indica a la unidad de control las operaciones pertinentes así como las transferencias de datos necesarias para ejecutar la instrucción. Para el microprocesador genérico de 8 bits, hay instrucciones de 1, 2 y 3 palabras de 8 bits, o bytes. Esta longitud depende del tipo de instrucción y del modo de direccionamiento.

Modos de direccionamiento

Para entender lo que es un modo de direccionamiento conviene analizar las *campos* que forman una instrucción. Estos son el *campo O*, correspondiente al código de operación, el *campo M* donde se especifica el modo de direccionamiento, y el *campo A*, conocido como *campo de desplazamiento* o *dirección local*. En la figura 3.14 se aprecian las distribuciones de campos dependiendo del tamaño de instrucción.

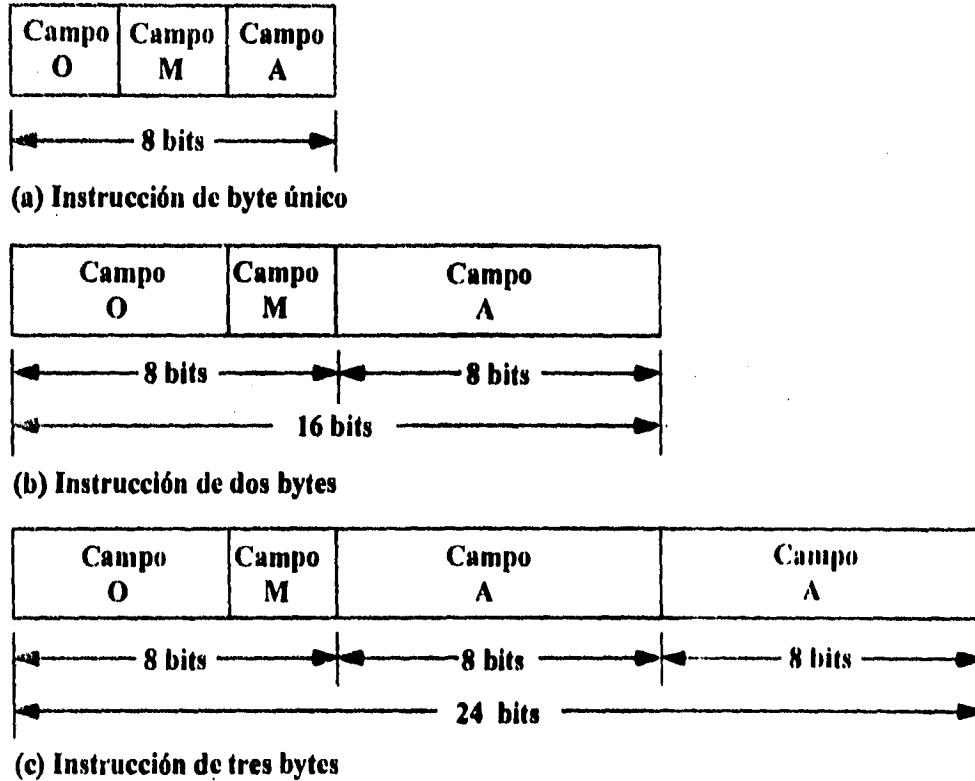


Figura 3.14. Distribuciones de campos en las distintas instrucciones.

Cabe aclarar que si se observan los códigos de operación de los microprocesadores comerciales actuales, éstos ya contienen el modo de direccionamiento (campo M), y en el caso de la instrucción de 1 byte, comprenden la instrucción completa a realizar, ya sea una transferencia de datos o una determinada operación. Pero, ¿Qué es el modo de direccionamiento? Comencemos por aclarar que para acceder a cualquier posición de memoria se requiere de dos procesos:

1. La generación de la dirección.
2. La utilización de la dirección.

Para explicar esto recordemos el campo A. Este campo es usualmente utilizado para generar la dirección donde se encuentra el operando de la instrucción, es decir, la *dirección efectiva*. La dirección es utilizada para rescatar o recuperar el dato-operando. Los distintos *métodos de recuperación* se denominan *modos de direccionamiento* del microprocesador. Los diferentes modos de direccionamiento influyen en la potencia de proceso de un μP , y cada fabricante proporciona distintos modos. A continuación se describen los más comunes.

Direccionamiento Inmediato

En este modo de direccionamiento el operando requerido por la instrucción es parte de la instrucción misma. Esto quiere decir que en el campo A de la instrucción se encuentra el dato en cuestión; no es necesario buscarlo en otro lado. Este modo de direccionamiento es común cuando se cargan registros internos con datos para distintos propósitos. Dependiendo del tamaño del registro interno en cuestión (8 o 16 bits), se necesitarán 2 o 3 bytes en la instrucción incluyendo el correspondiente al código de operación.

Direccionamiento inherente

También es conocido como *direccionamiento implícito*. Para este caso la instrucción sólo requiere un byte. Las instrucciones que utilizan este direccionamiento no requieren buscar operandos fuera del microprocesador, pues no se necesita ningún dato para completar la operación. Un ejemplo sería "poner a 1 el señalizador de arrastre del registro de status".

Direccionamiento de registro

En las instrucciones que utilizan este modo de direccionamiento, el operando es recuperado de un registro interno del μP . Este tipo de operaciones son siempre instrucciones de un byte, como en el caso del direccionamiento inherente, pues no se necesitan datos ni direcciones fuera del μP y todas las acciones ocurren dentro del mismo. Un ejemplo es sumar el contenido del registro D al contenido del acumulador (A) y guardar el resultado en A.

Direccionamiento directo

Para este modo de direccionamiento, que es considerado como el más primitivo, simple y fácil de usar, el campo A contiene la dirección donde se encuentra el operando requerido por la instrucción, es decir, la *dirección local* es la *dirección efectiva*. De esta forma, para este modo de direccionamiento usualmente se requieren un segundo y tercer bytes para almacenar los bytes menos y más significativos de la dirección mencionada. En algunos μP 's este modo se conoce también como *direccionamiento absoluto*. Otros μP 's hacen la distinción entre *direccionamiento directo* y *direccionamiento extendido*, donde en el primero se usan sólo 2 bytes, lo que implica que únicamente se pueden direccionar las primeras 256 posiciones de memoria, lo que es conocido como la página cero o página base (esto opera como el *direccionamiento relativo a la página cero*); para estos μP 's el último término coincide con la explicación que se dio inicialmente, es decir, requiere de 3 bytes y puede direccionar 65,536 posiciones de memoria. Esta explicación inicial es una definición más general del direccionamiento directo. Aún cuando es muy utilizado, se presenta el inconveniente de requerir de bytes adicionales lo que implica mayor tiempo de proceso.

Direccionamiento indirecto

El *direccionamiento indirecto* consiste en lo siguiente: la dirección local del campo A sirve como un puntero que indica una posición de memoria donde se puede encontrar la dirección efectiva. Hay varios tipos de direccionamiento indirecto.

Direccionamiento indirecto de registro

En este tipo de direccionamiento la dirección local apunta a algún registro intermedio; éste es un registro interno del microprocesador; que puede ser un registro de propósito general, como el par de registros H y L (registro par HL) en algunos μP 's, que se usa para apuntar direcciones. Si para esto se usa un par de registros se puede acceder a 65,536 direcciones de memoria de datos. Una fuerte ventaja de este modo es que sólo requiere un byte pues se trabaja con registros internos. Otra ventaja es que si se pueden usar varios registros internos como registros intermedios es posible prealmacenar varias direcciones efectivas y acceder a estas posiciones de RAM con instrucciones monopalabra usando *direccionamiento indirecto de registro*. Finalmente, de este modo se puede acceder repetidamente a una misma posición de memoria. Con este direccionamiento se están usando *punteros internos* del μP . El inconveniente radica en que el número de registros internos es limitado, lo que a su vez limita el número de direccionamientos indirectos de registro,

Direccionamiento inherente

También es conocido como *direccionamiento implícito*. Para este caso la instrucción sólo requiere un byte. Las instrucciones que utilizan este direccionamiento no requieren buscar operandos fuera del microprocesador, pues no se necesita ningún dato para completar la operación. Un ejemplo sería "poner a 1 el señalizador de arrastre del registro de status".

Direccionamiento de registro

En las instrucciones que utilizan este modo de direccionamiento, el operando es recuperado de un registro interno del μP . Este tipo de operaciones son siempre instrucciones de un byte, como en el caso del direccionamiento inherente, pues no se necesitan datos ni direcciones fuera del μP y todas las acciones ocurren dentro del mismo. Un ejemplo es sumar el contenido del registro D al contenido del acumulador (A) y guardar el resultado en A.

Direccionamiento directo

Para este modo de direccionamiento, que es considerado como el más primitivo, simple y fácil de usar, el campo A contiene la dirección donde se encuentra el operando requerido por la instrucción, es decir, la *dirección local* es la *dirección efectiva*. De esta forma, para este modo de direccionamiento usualmente se requieren un segundo y tercer bytes para almacenar los bytes menos y más significativos de la dirección mencionada. En algunos μP 's este modo se conoce también como *direccionamiento absoluto*. Otros μP 's hacen la distinción entre *direccionamiento directo* y *direccionamiento extendido*, donde en el primero se usan sólo 2 bytes, lo que implica que únicamente se pueden direccionar las primeras 256 posiciones de memoria, lo que es conocido como la página cero o página base (esto opera como el *direccionamiento relativo a la página cero*); para estos μP 's el último término coincide con la explicación que se dio inicialmente, es decir, requiere de 3 bytes y puede direccionar 65,536 posiciones de memoria. Esta explicación inicial es una definición más general del direccionamiento directo. Aún cuando es muy utilizado, se presenta el inconveniente de requerir de bytes adicionales lo que implica mayor tiempo de proceso.

Direccionamiento indirecto

El *direccionamiento indirecto* consiste en lo siguiente: la dirección local del campo A sirve como un puntero que indica una posición de memoria donde se puede encontrar la dirección efectiva. Hay varios tipos de direccionamiento indirecto.

Direccionamiento indirecto de registro

En este tipo de direccionamiento la dirección local apunta a algún registro intermedio; éste es un registro interno del microprocesador; que puede ser un registro de propósito general, como el par de registros H y L (registro par HL) en algunos μP 's, que se usa para apuntar direcciones. Si para esto se usa un par de registros se puede acceder a 65,536 direcciones de memoria de datos. Una fuerte ventaja de este modo es que sólo requiere un byte pues se trabaja con registros internos. Otra ventaja es que si se pueden usar varios registros internos como registros intermedios es posible prealmacenar varias direcciones efectivas y acceder a estas posiciones de RAM con instrucciones monopalabra usando *direccionamiento indirecto de registro*. Finalmente, de este modo se puede acceder repetidamente a una misma posición de memoria. Con este direccionamiento se están usando *punteros internos* del μP . El inconveniente radica en que el número de registros internos es limitado, lo que a su vez limita el número de direccionamientos indirectos de registro,

además de que los registros internos de propósito general suelen necesitarse para otras operaciones. Además, estar prealmacenando direcciones efectivas (es común hacer esto con direccionamiento inmediato) puede resultar engorroso y lento.

Direccionamiento indirecto a la página

Una página de memoria está definida por los bits más significativos de la dirección de memoria. Para efectos de esta explicación consideraremos a estos bits como el byte más significativo de la dirección. En este modo de direccionamiento, se utilizan *punteros de página*; esto es similar al uso de punteros internos, caso en el que se presentan ciertos inconvenientes por direccionamiento indirecto de registro. Los modos de direccionamiento indirecto a la página se basan en usar direcciones efectivas prealmacenadas en una página de memoria. Esto permite usar solo un byte en el campo A.

En el caso de *direccionamiento indirecto a la página cero (o página base)*, la dirección efectiva se encontrará en una posición de memoria cuyo byte más significativo es 0 y el byte menos significativo es la dirección local o desplazamiento indicado en el campo A de la instrucción.

En el *direccionamiento indirecto a la página actual*, la dirección efectiva se encontrará en una posición de memoria cuyo byte más significativo es el byte de mayor orden del contador de programa (PC), y el byte menos significativo es la dirección local o desplazamiento indicado en el campo A de la instrucción.

Utilizando estos dos modos es posible acceder a operandos localizados en otras zonas de la memoria. La ventaja de esto es acceder a un gran segmento de memoria usando instrucciones no tan largas.

Direccionamiento relativo a la página

Cuando se utiliza *direccionamiento relativo a la página actual*, la dirección efectiva del operando de la instrucción se obtiene concatenando el byte más significativo del PC con la dirección local de la instrucción, que para este caso, consta de un byte. De este modo, sólo se pueden acceder direcciones en la misma página de la instrucción.

En el *direccionamiento relativo a la página cero*, conocido también como *direccionamiento de página base*, la dirección efectiva es formada con un 0 en el byte más significativo y la dirección local de la instrucción como byte menos significativo. En este caso no se depende de la localización de la instrucción misma dentro de la memoria de programa.

Direccionamiento relativo al contador de programa

Este es también conocido simplemente como *direccionamiento relativo*. En este caso se considera el bit de signo del contenido del campo A. La dirección efectiva se obtiene sumando este contenido al contenido del contador de programa. Como se ve, si el signo es negativo se tendrá un salto hacia atrás, mientras que con signo positivo el salto será hacia adelante. Este modo de direccionamiento es útil cuando se ejecuta una instrucción de bifurcación o salto hacia una posición de memoria cercana; no es necesario especificar toda la dirección de ésta.

Direccionamiento indexado directo

Este es un modo de direccionamiento similar al direccionamiento relativo al contador de programa. En el *direccionamiento indexado directo* se utiliza el contenido de un registro especial llamado *registro índice*. Los μP 's de 8 bits que lo contienen usualmente cuentan con un registro índice de 16 bits o un par de registros

Índice de 8 bits cada uno. En otros casos, el registro índice se denomina *registro B*. Algunos μP s utilizan un puntero o apuntador de datos como registro índice. De cualquier manera, en este modo de direccionamiento la dirección efectiva se forma sumando el contenido del registro índice con el desplazamiento o dirección local. Esta dirección se carga en el MAR y se usa para acceder a la posición de memoria deseada. El modo de direccionamiento indexado es útil cuando se requiere un uso frecuente de tablas de consulta. Cada tabla tiene una dirección de inicio llamada dirección base. Esta dirección base está en los bits de desplazamiento del campo A de la instrucción. El parámetro cuyo valor se debe buscar en la tabla se precarga en el registro índice. La suma de estas dos cantidades es la dirección efectiva de la posición de memoria que se desea acceder en la tabla.

Direccionamiento indexado indirecto (preindexado)

Aquí se combinan los modos indirecto e indexado. En este caso el desplazamiento de la instrucción se suma a la cantidad prealmacenada en el registro índice. La suma resultante es la dirección donde está almacenado el puntero indirecto. En esta última posición de memoria se encuentra la dirección completa de la posición de memoria deseada, es decir, el puntero indirecto indica la dirección efectiva. Para el almacenamiento se pueden utilizar también los registros internos del μP . También es posible usar la página 0 de la memoria.

Direccionamiento indexado indirecto (postindexado)

En el modo postindexado, el desplazamiento de la instrucción contiene la dirección del puntero indirecto que puede ser un registro interno o estar en la página base. El contenido prealmacenado en el registro índice se suma al puntero indirecto para obtener la dirección efectiva, que es usada a través del MAR para acceder a la posición deseada de la memoria. Como el proceso de modificación con el contenido del registro índice tiene lugar después de llamar al puntero indirecto, este método recibe el nombre de postindexado.

Los distintos μP s, de acuerdo a su arquitectura, presentan diferentes modos de direccionamiento. Es posible combinar modos. Un μP comercial específico no contiene todos los mencionados y es posible que contenga otros más complejos.

3.3. PERIFÉRICOS

3.3.1. ¿QUÉ SON LOS PERIFÉRICOS?

Un periférico es un dispositivo de entrada o salida para la computadora. En el uso de computadoras todos hemos estado en contacto con unidades de entrada como el teclado para introducir texto y el mouse (ratón) para introducir dibujos y figuras y con unidades de salida como visualizadores de siete segmentos que como en las calculadoras nos permiten observar resultados numéricos, monitores que permiten desplegar visualmente el trabajo que se está realizando, impresoras con las que se puede imprimir texto y figuras, plotters que sirven para imprimir planos, etc. Estos son los periféricos y son indispensables y de gran utilidad para la interacción usuario-computadora.

3.3.2. ¿QUÉ SON LOS PUERTOS?

Los puertos reciben su nombre de la analogía con los puertos marítimos de un país dado. En el comercio de ese país, las mercancías entran y salen del mismo a través de los puertos. De la misma manera los puertos de un microprocesador son dispositivos por los cuales entran y salen datos del mismo. La conexión de los periféricos de una computadora internamente va los puertos del microprocesador de la misma, de manera que la información que entra es procesada y se entrega información de salida.

3.3.3. PUERTOS PARALELO

El acto de transferir datos desde o a un dispositivo periférico es conocido como *operación de entrada o salida*. En sistemas con microprocesador, el μP es el foco de toda operación, por lo que una *entrada* significa que el dato fluye hacia el μP y una *salida* significa que el dato fluye del mismo. Aquellas posiciones donde el dato entra o sale se denominan *puertos de entrada salida, puertos E/S o ports I/O*. Cuando al exterior del puerto esta transferencia se ejecuta transfiriendo una palabra de datos completa en un solo pulso de reloj, estamos hablando de un *puerto paralelo*.

Cuando se usan instrucciones de programa que controlan la transferencia de datos durante las operaciones de entrada/salida, se está haciendo una *transferencia controlada por programa*. Una transferencia de datos puede ser iniciada por el dispositivo periférico cuando éste dice "estoy listo para enviar o recibir datos". En este caso, estos dispositivos utilizan una *interrupción*. La subrutina de servicio de interrupción correspondiente puede incluir operaciones de entrada y salida.

Hay dos técnicas de E/S controladas por programa. Una de éstas se conoce como *entrada/salida aislada o por acumulador*. Se utilizan instrucciones como IN (Entrar) y OUT (Salir) para transferir datos a y desde los puertos de entrada/salida. Esta técnica y las instrucciones correspondientes se ilustran en la figura 3.15a.

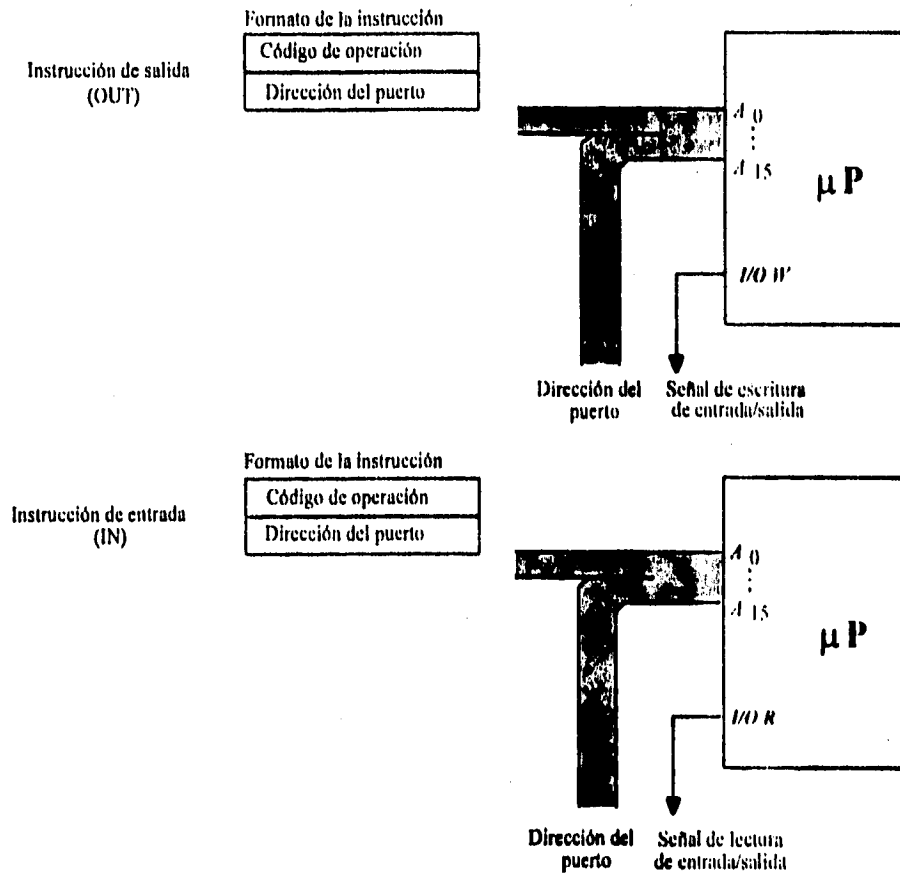


Figura 3.15a.

Con el formato de instrucción indicado se puede seleccionar uno de 256 (2^8) puertos, pues la dirección del puerto proviene de las 8 líneas de dirección menos significativas (A_7-A_0). La figura 3.15a muestra dos señales de control de salida que pueden ser generadas a partir de las señales de control de entrada/salida de datos de nuestro microprocesador genérico (\overline{RD} , \overline{WR} e IO/\overline{M}). Al utilizar la operación OUT, se activa una señal especial de *escritura de entrada/salida* (\overline{IOW}); para la operación IN se tiene una señal de *lectura de entrada/salida* (\overline{IOR}). Ambas señales se activan a nivel bajo.

La otra técnica trata a las posiciones de entrada y salida como direcciones regulares de memoria. Esta técnica se denomina *E/S de mapa de memoria* (o con correspondencia en el espacio de direcciones de memoria). Con esta técnica se usan las direcciones regulares de acceso a memoria. En la figura 3.15b la instrucción "almacena A directo" se utiliza para sacar datos a un puerto de salida. En la misma figura, la instrucción "carga A directo" se utiliza para introducir datos desde un puerto o dispositivo. Las líneas de dirección deben ser decodificadas y utilizadas para la selección exacta de un puerto de E/S. Las señales habituales de escritura (\overline{WR}) y lectura (\overline{RD}) también se utilizan para entrar o sacar datos. Con esta técnica cualquier instrucción de acceso de memoria se puede usar para entrar o sacar datos.

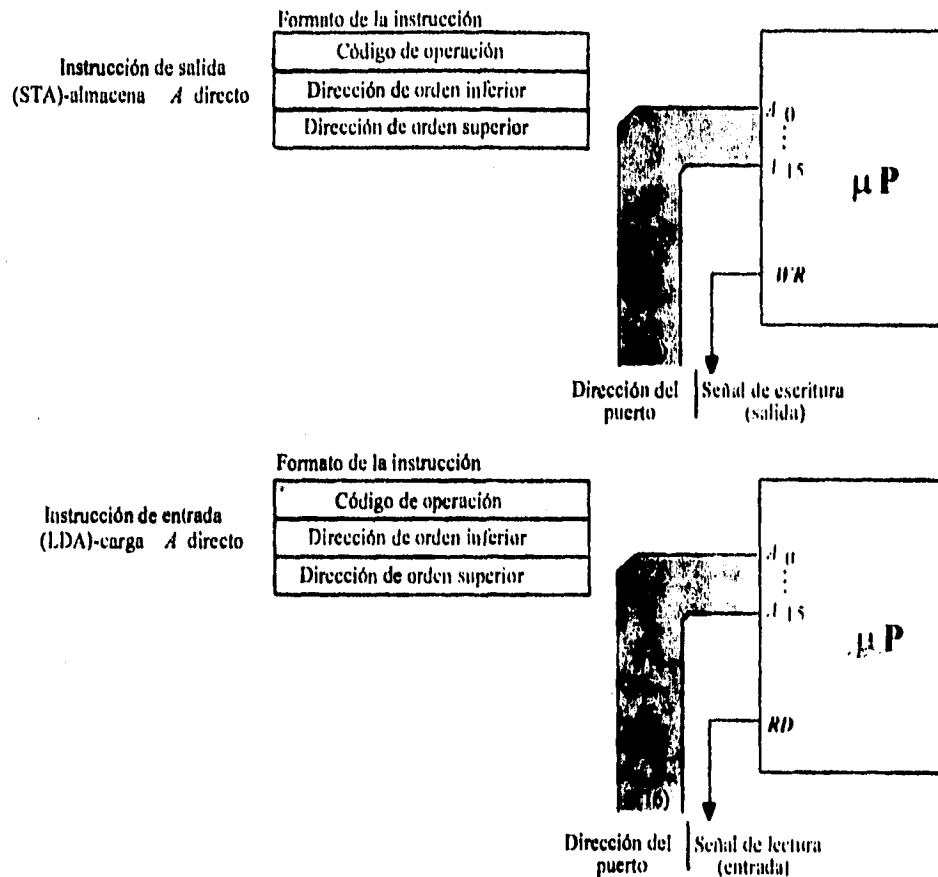


Figura 3.15b.

La instrucción STA almacena el contenido del acumulador en la dirección de memoria especificada por los dos bytes que siguen al opcode. La instrucción LDA carga en el acumulador el dato contenido en el segundo y tercer bytes de la instrucción.

La técnica de E/S de mapa de memoria es, probablemente, la más común y puede utilizarse con cualquier microprocesador. La técnica de E/S aisladas puede usarse sólo con μP 's que tengan instrucciones IN y OUT separadas, entradas/salidas especiales y salidas de control de lectura y escritura.

3.3.4. PUERTOS SERIE

Por los puertos serie entran y salen datos al microprocesador de manera serial. Esto significa que dada una palabra de ocho bits o un byte como es el caso de nuestro microprocesador genérico, la palabra es transmitida ya sea de entrada o de salida, bit por bit sucesivamente. Dado que la transmisión es bit por bit en un puerto serie, ésta es más lenta que en un puerto paralelo, pero por otro lado, no son necesarias tantas conexiones.

3.3.5. TEMPORIZADORES

Para llevar a cabo el conteo de tiempo y eventos, existen dispositivos que se suelen incluir en los sistemas con microprocesador llamados temporizadores y contadores. Los temporizadores, como su nombre lo indica, cuentan tiempo en unidades muy pequeñas y exactas del mismo, a la manera de un reloj electrónico. Esto sirve por ejemplo, para evaluar la duración de pulsos de voltaje que eventualmente son conectados a alguna de las patas del puerto paralelo, o por otro lado, para limitar la duración de un proceso dado.

Para contar eventos, se tienen los contadores. Estos dispositivos sirven para contar pulsos, dado un voltaje intermitente conectado a la pata del contador, lo cual es muy útil en muchos sistemas con microprocesador.

3.4. INTERRUPCIONES

3.4.1. ¿QUÉ ES UNA INTERRUPCIÓN?

Una *interrupción* es una forma de trabajar con señales que ocurren aleatoriamente en el tiempo y que deben afectar las tareas que está realizando el microprocesador. Existen *interrupciones software*, cuando la señal aludida es generada por una instrucción del microprocesador, e *interrupciones hardware*, las más comunes, cuando una señal externa interrumpe el trabajo de un microprocesador. Este último, es un método de informar al μP que un dispositivo de E/S está "listo" y de la acción que necesita tomarse.

Cuando se cuenta con varios y distintos tipos de interrupción, es necesario atender a la *prioridad de la interrupción*, pues varias señales de interrupción pueden presentarse al mismo tiempo y es necesario decidir cual será atendida primero.

Durante la subrutina de servicio de interrupción pueden alterarse los contenidos de ciertos registros, lo que al volver al programa principal podría ocasionar problemas. Es por esto que al ser activada alguna interrupción, el contenido de algunos registros se introduce automáticamente al stack (en la gran mayoría de los μP 's), y durante la instrucción de retorno de interrupción estos contenidos se sacan de la pila, restaurándose así los valores almacenados antes de la interrupción en los registros. Los registros cuyos contenidos se almacenan en el stack son generalmente el contador de programa, el registro de status, y en algunos casos el acumulador y, si existe, el registro índice.

3.4.2. INTERRUPCIONES HARDWARE

Una entrada de interrupción, si es habilitada, permite a una señal externa decirle al microprocesador que después brinque a una subrutina que responda a la necesidad indicada por la interrupción. Al final de esta subrutina de servicio de interrupción, una instrucción de retorno envía al μP de regreso al programa que estaba corriendo antes de ser interrumpido.

Habilitar una entrada de interrupción significa que mediante alguna instrucción de programa se indica al μP que "haga caso" a la señal externa o que la ignore. Comúnmente las interrupciones se referencian como enmascarables o no enmascarables. Una *interrupción no enmascarable* es la que no puede ser inhabilitada por el sistema. Una *interrupción enmascarable* puede ser inhabilitada por el sistema. Las interrupciones no enmascarables tienen mayor prioridad que las enmascarables. El mecanismo real para habilitar e inhabilitar selectivamente las interrupciones se denomina *máscara de interrupción*, lo cual es válido también cuando sólo se tiene una interrupción enmascarable. Si una interrupción es *enmascarada*, no es reconocida por el microprocesador. Esto se logra poniendo a 1 (enmascarar) o 0 (no enmascarar) un flip-flop llamado *señalizador de máscara de interrupción*.

Se pueden tener distintos sistemas de interrupciones. Estos se describen a continuación.

Interrupciones por línea

Se dice que un μP maneja interrupciones por línea, cuando tiene una línea o si acaso dos (enmascarable y no enmascarable), por la(s) que sólo un dispositivo externo puede interrumpir la ejecución del programa principal. En estos μP 's, el señalizador de máscara de interrupción / se encuentra en el registro de status del microprocesador. Tal es el caso de nuestro microprocesador genérico.

Cuando se activa con nivel BAJO la entrada de control \overline{NMI} o *interrupción no enmascarable* del μP genérico, éste salta a la secuencia de interrupción no enmascarable mostrada en la figura 3.16.

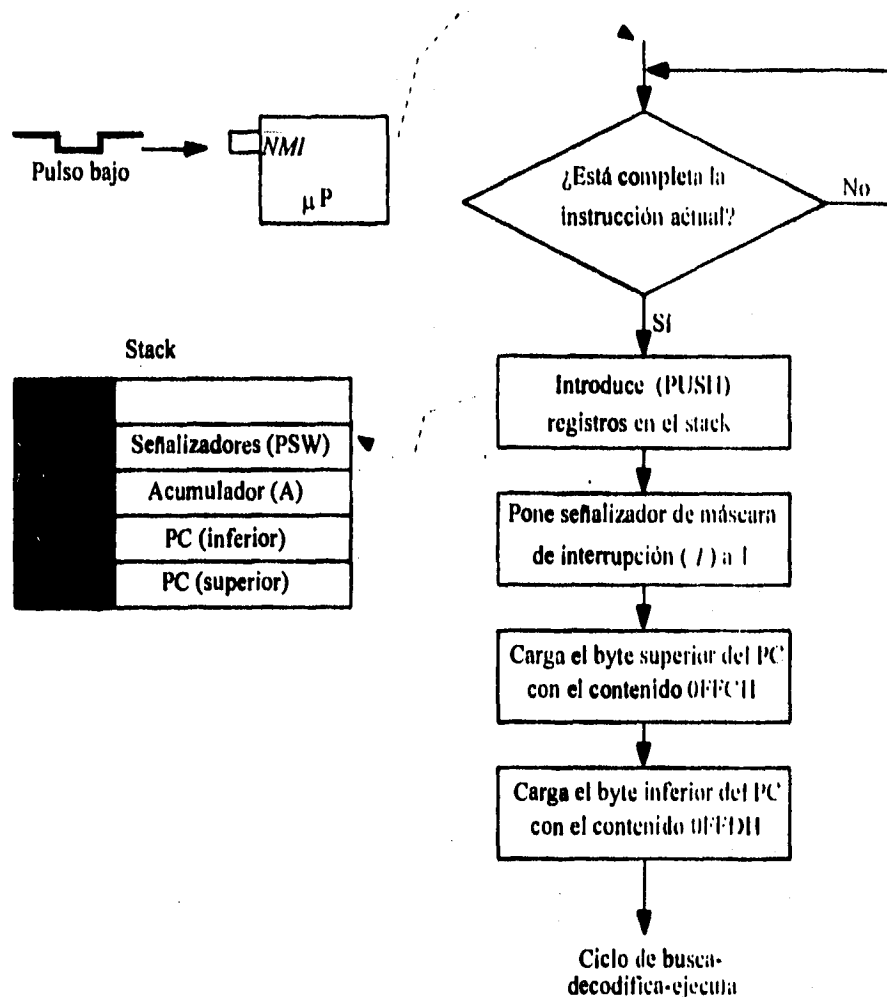


Figura 3.16.

La instrucción actual se completa, el contenido de los registros del μP genérico se guarda en el stack, el señalizador I se pone a 1 y la dirección de la subrutina de servicio de interrupción no enmascarable se busca en las posiciones de memoria 0FFCH y 0FFDH, cargando el PC con el contenido de esas direcciones. El μP salta entonces a esta subrutina.

Como se puede observar en la figura 3.17, al activar con nivel BAJO la entrada de *petición de interrupción*, \overline{TRQ} , la cual es una interrupción enmascarable, el μP salta a la secuencia de petición de interrupción mostrada a la derecha de la figura. Se completa la instrucción actual y se comprueba el señalizador I . Si éste está a 1, el μP ignora la petición de interrupción y continúa con el programa regular. Si el señalizador I está a 0, el μP introduce entonces el contenido de los registros en el stack como se observa. A continuación se pone a 1 el señalizador I , pues así no se puede presentar otra interrupción enmascarable mientras se realiza la subrutina deseada. Como paso siguiente, el PC se carga con la dirección contenida en las posiciones de memoria ROM 0FF8 (byte superior) y 0FF9 (byte inferior). El proceso se traslada a la subrutina de servicio de petición de interrupción.

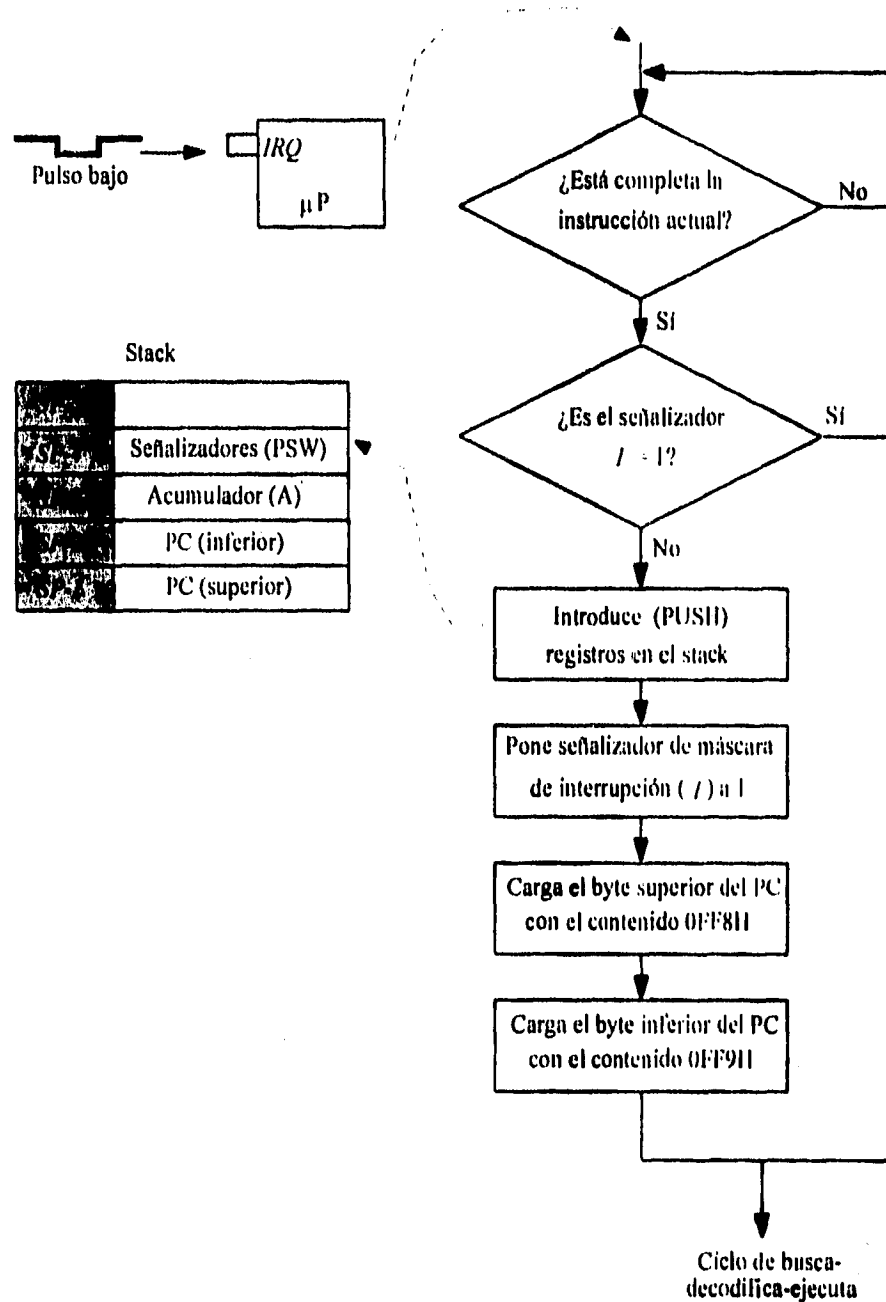


Figura 3.17.

En algunos microprocesadores la entrada \overline{RESET} es considerada una interrupción, pues después de activarse por pasar por un tiempo a nivel BAJO y luego a nivel ALTO, se sigue un proceso similar, es decir, se pone a uno el señalizador I , y el PC se carga con el contenido de unas posiciones de memoria ROM específicas. Se ejecuta entonces una subrutina de reinicialización. De cualquier manera \overline{RESET} tiene prioridad mayor que las demás interrupciones.

Interrupciones multilínea

Las *interrupciones multilínea* se pueden considerar como un *vector de interrupciones*, pero buscando mayor claridad se hace la distinción entre las dos categorías. Estas interrupciones son el caso de los microprocesadores que cuentan con varias líneas de petición de interrupción. Estas se numeran y son enmascarables; cada una tiene distinta prioridad. Puede tenerse además una entrada de interrupción no enmascarable, con prioridad mayor a la de las enmascarables. Cada interrupción enmascarable se puede enmascarar con un señalizador de máscara de interrupción individual, contenido en un registro de interrupciones. Este registro puede contener también un señalizador general que habilite o inhabilite a todas las interrupciones en conjunto. De esta manera, además de revisar el señalizador de máscara de interrupción individual, se revisa también durante la secuencia de interrupción el señalizador general, el cual pudo haber sido activado o desactivado mediante una instrucción de programa. La activación de cada línea de petición de interrupción produce un salto a una posición de memoria de programa distinta para cada línea.

Vector de interrupciones

También conocido como *esquema de interrupciones vectorizadas*, de esta manera se puede lograr que uno de varios dispositivos interrumpa al microprocesador, usando una sola entrada de interrupción (podemos distinguirla como *INTR*). Para lograr esto, se requieren *circuiterías de arbitraje de prioridad de interrupciones* externas al microprocesador. Estos circuitos tienen dos objetivos:

1. Resolver problemas de prioridad de interrupciones, para los casos en que se presente más de una señal de interrupción a la vez.
2. Informar al microprocesador cuál dispositivo está interrumpiendo, de modo que el μP salte a la correspondiente subrutina de servicio de interrupción.

Para hacer posibles estos esquemas, es necesario que el μP cuente con una salida de control de *conocimiento de interrupción* (la distinguiremos como \overline{INTA}). La señal proporcionada por dicha salida de control para habilitar una entrada al bus de datos (una dirección) de manera que el microprocesador sabe que tarea debe desempeñar en atención a un determinado dispositivo, después de haberse tomado la decisión correspondiente a la prioridad.

A continuación se describen brevemente estos esquemas.

Interrupciones vectorizadas basadas en direcciones distintas. Este esquema se basa en enviar al microprocesador a través del bus de datos la dirección de la subrutina de servicio de interrupción correspondiente a un determinado dispositivo externo. Esta dirección es determinada por una red externa que selecciona uno de los dispositivos que quieren interrumpir de acuerdo a un esquema de prioridad.

En este esquema se asocia una circuitería a cada dispositivo, donde es posible si es que ya fue seleccionado por el esquema de prioridad y una vez recibida la señal \overline{INTA} que indica que el μP está listo para atender a una interrupción, poner en el bus de datos la dirección correspondiente a la subrutina que desencadenará una interrupción ocasionada por ese dispositivo en especial como se aprecia en la figura 3.18.

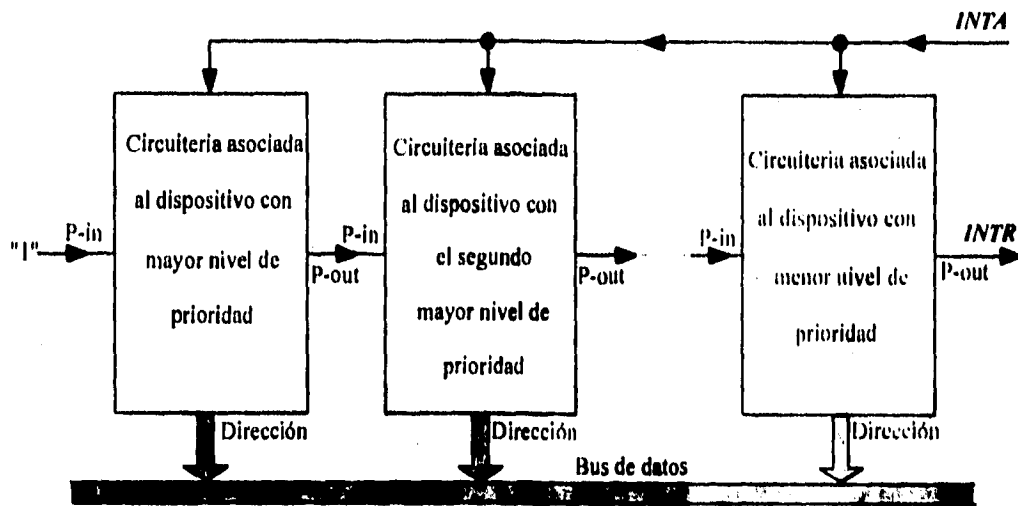


Figura 3.18.

Codificadores de prioridad para vectores de interrupciones. Este segundo esquema de interrupciones vectorizadas provee al microprocesador un código numérico único que representa al dispositivo causante de la interrupción. En este esquema, una interrupción ocasiona un salto a una posición determinada donde un servicio-maestro de interrupciones comienza. Esta rutina introduce al μP el contenido de un puerto especial al que se ha proporcionado el número de código del dispositivo que interrumpe. Basándose en este número de código un salto subsecuente se hace a otra rutina, particular para el dispositivo en cuestión.

La manera más común de implementar este esquema para que resuelva los problemas relacionados con la prioridad al presentarse a un tiempo varias peticiones de interrupción, es como sigue: un codificador, cuya representación se puede observar en la figura 3.19, es diseñado de tal forma, que no solo provea un código dependiendo de la entrada que se pone a 1, sino que cuando varias entradas se ponen a 1, entregue a la salida el código de la entrada más "alta" del codificador.

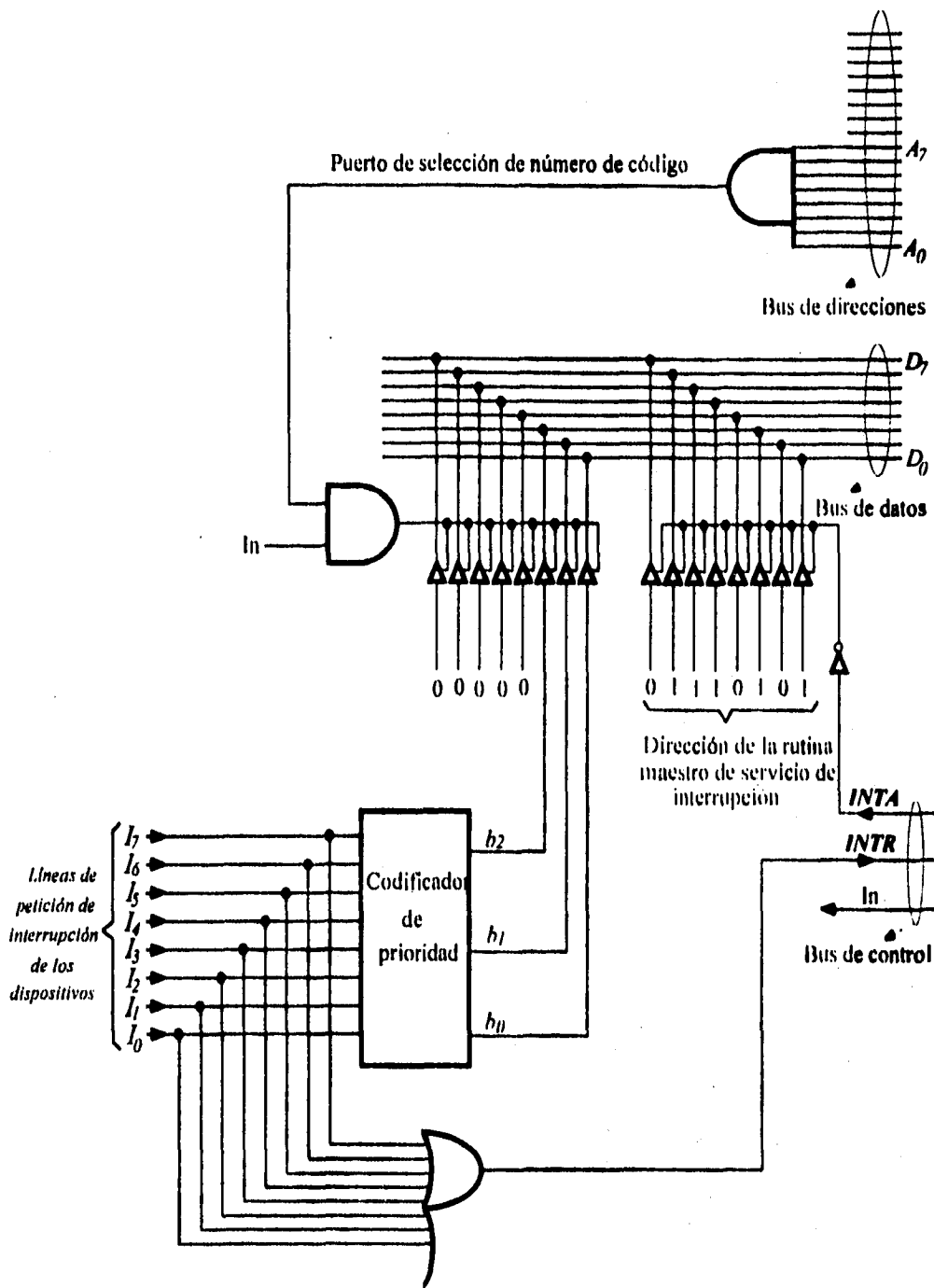


Figura 3.19.

El número de código resultante, representado por 3 bits, se presenta a un puerto paralelo de entrada, de manera que el μP tiene acceso a él. Las líneas de petición de los dispositivos se conectan a una compuerta OR, lo que permite que al activarse una o más de éstas, la salida de la compuerta active la entrada $INTR$ del μP . Un pulso en el momento adecuado aparecerá en la línea de control \overline{INTA} , en respuesta al reconocimiento de una interrupción por el microprocesador. Este pulso se usa para poner la dirección de la rutina maestro de

interrupción en el bus de datos, de modo que sea transferida al μP . La ejecución de la rutina mencionada comienza y se introduce primero (activando la salida de control *In* o las correspondientes salidas para lectura de puertos I/O) el número de código de la petición de interrupción de más alta prioridad que esté demandando interrumpir. La rutina usa el código como una base para un brinco a una rutina que atienda al dispositivo en cuestión.

Interrupciones vectorizadas usando instrucciones originadas en los dispositivos. Algunos microprocesadores usan esquemas de interrupciones vectorizadas que involucran una instrucción mandada al μP desde el dispositivo que interrumpe. Cuando se reconoce una petición de interrupción por estos microprocesadores, la respuesta es buscar la siguiente instrucción a ser ejecutada en la red de interfaz para el dispositivo que interrumpe, en lugar de buscarla en la memoria. El contenido del PC no se guarda automáticamente cuando ocurre una interrupción en estos esquemas.

Algunas veces se trata tan sólo de una instrucción pero las más de las veces se requieren varias, de modo que la instrucción obtenida del dispositivo externo puede ser un salto a una subrutina especial para atender a dicho dispositivo. De cualquier manera, después de todo esto, pues las subrutinas al ser terminadas regresan a donde se quedó el programa principal, se reanuda la operación normal del programa.

Esquema de escrutinio de interrupciones

Para el esquema de *escrutinio (polling) de interrupciones*, sólo es necesaria una entrada de control al microprocesador. Esta entrada proviene de la salida de una compuerta OR a la que están conectadas todas las líneas de petición de interrupción de los dispositivos externos. Asociado a cada dispositivo hay un puerto de status de entrada, donde la posición de un bit es usada para indicar una petición de interrupción hecha por ese dispositivo. Cuando una señal de interrupción de cualquier dispositivo aparece en la línea de control de petición de interrupciones hacia el μP , se hace un salto a una dirección específica que alberga una rutina-maestro de servicio de interrupciones, que sucesivamente introducirá el contenido de cada puerto de status para probar su bit de petición de interrupción. Cuando se encuentra el dispositivo que está demandando la interrupción, la rutina-maestro salta a una subrutina de servicio para ese dispositivo.

Para resolver los problemas de prioridad, este esquema da mayor prioridad a los dispositivos que busca primero en la rutina-maestro de servicio de interrupción. La desventaja de este esquema es que el escrutar interrupciones le lleva tiempo, que en algunos sistemas es vital.

3.4.3. INTERRUPTIONES SOFTWARE

Una *interrupción software* es comandada por una correspondiente instrucción de interrupción. Los microprocesadores que contienen este tipo de instrucciones usualmente siguen la siguiente secuencia:

1. Se activa el señalizador correspondiente, que suele estar en el registro de status del μP . Esto indica que la interrupción fue provocada por una instrucción y no por una interrupción hardware.
2. Se introduce el contenido de los registros adecuados en el stack.
3. Se inicializa el señalizador de máscara de interrupción para que no se presente una interrupción enmascarable, usualmente de menor prioridad que una instrucción software.
4. Se carga el contador de programa con el contenido de unas direcciones específicas de memoria, lo que es la dirección de comienzo de tratamiento de la interrupción.

Como se ve, la interrupción software hace aproximadamente lo mismo que las interrupciones hardware y es parecida a una subrutina. Se diferencia de esta última en que como en las otras interrupciones, al retorno de la subrutina de interrupción se restauran los valores de los registros, los cuales pueden ser alterados en una subrutina normal. No se permite otra interrupción mientras se está procesando la subrutina de servicio de interrupción software, lo que puede ser usado para ciertos propósitos.

3.5. APLICACIONES

3.5.1. APLICACIONES DEL MICROPROCESADOR

El microprocesador es el desarrollo electrónico individual más importante desde el transistor y, al igual que éste, ha revolucionado todas las industrias en toda su amplia gama de aplicaciones. Su popularidad se deriva de su bajo costo, la pequeña área que ocupa, su bajo consumo de energía y su alto nivel de confiabilidad. Estas características han fomentado nuevas aplicaciones tanto en la industria como en el hogar. Se han lanzado al mercado ciertos paquetes de software listos para usarse, los que simplifican la programación y reducen el costo de los sistemas.

Los sistemas con microprocesador, están llamados a ser aplicados en la mayoría de los productos terminados, que alcancen cierto grado de complejidad y de precio. En la actualidad, los precios de algunos chips que componen las familias que se aplican en sistemas con microprocesador, en países no fabricantes de los mismos, no es alto, lo que unido a la tendencia a la baja de los precios del mercado, hace suponer que en breve, el costo añadido a un producto terminado al que se introduzca un sistema con microprocesador, será de poca importancia. También merece ser destacado el hecho de que la constante investigación en el área de la microelectrónica favorece las innovaciones que continuamente ofrecen los fabricantes de componentes, que cada vez van siendo más potentes y sofisticados al mismo tiempo que va disminuyendo su precio.

A continuación se muestran los campos de aplicación general de los microprocesadores:

- Industria en general
- Industria de automoción
- Control del tráfico de automóviles
- Control del tráfico de ascensores
- Industria eléctrica y de electrodomésticos
- Electromedicina
- Instrumentación y medida
- Terminales inteligentes
- Juegos
- Industria musical

En los siguientes puntos se detallarán algunos de estos campos de aplicación.

3.5.2. INDUSTRIA EN GENERAL

Los microprocesadores cada vez van teniendo más aplicaciones en la industria. Se espera que esta tendencia continúe pues son de gran utilidad en la automatización y control de procesos. Algunos casos en los que se encuentran aplicaciones de sistemas con microprocesador son: Laminación, adquisición de datos, transductores, control numérico de máquinas herramientas, control parcial y total de procesos, dosificación de básculas, autómatas programables, sustitución de electrónica convencional, clasificación de chapas, pesaje electrónico, comprobación de componentes activos y pasivos, comprobación de tarjetas con circuitos integrados, toma y tratamiento múltiple de temperaturas, control de procesos químicos, control de estaciones emisoras-receptoras de cartuchos neumáticos, control de hornos de gratificación, control de válvulas, dosificación en fábricas de hormigón, regulación de cadenas de hornos, cromatógrafo de procesos, sistemas de supervisión y telemando. A continuación se explica una de estas aplicaciones.

Control numérico de máquinas-herramientas

Los sistemas con microprocesador se han empleado con éxito a diversas partes de los dispositivos que controlan a las máquinas-herramientas.

Una de las posibilidades de empleo del microprocesador en este campo la constituye el bloque de la máquina-herramienta llamado "controlador", que es el que se encarga de codificar la información que recibe desde la memoria o el operador y luego enviarla a la máquina para mover la herramienta o la pieza en los 3 ejes del espacio. Los movimientos en la máquina son controlados por una serie de transductores que informan al controlador y constituyen un bucle cerrado de realimentación. La figura 3.20. muestra un diagrama por bloques del controlador.

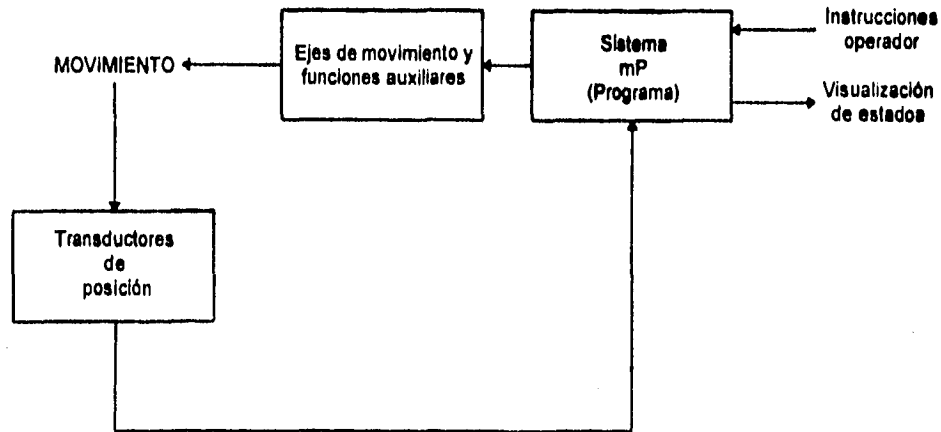


Figura 3.20. Diagrama por bloques del controlador.

Otra aplicación del microprocesador en la máquina-herramienta es la que se dedica a realizar todos los cálculos geométricos en los 3 ejes, según los programas de tipo matemático establecidos.

También se emplean los sistemas con microprocesador en el control punto a punto, con el cual se almacenan todas las posiciones de la matriz de puntos en la memoria del sistema, formada con incrementos, lo que le confiere una gran flexibilidad, pues el cambio de programas de trabajo sólo requiere la sustitución de la memoria. La figura 3.21. muestra una matriz de puntos.

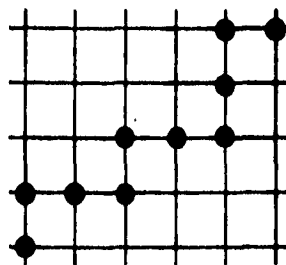


Figura 3.21. Matriz de puntos

También puede implementarse, usando un microprocesador, otra versión de control denominada "por trazo continuo", aunque en este caso y dado que el perfil de la pieza es dependiente de la posición de la máquina y se ha de trabajar con tolerancias reducidísimas, el sistema microprocesador ha de realizar un trabajo más rápido y potente.

El control numérico de los 3 ejes del espacio se realiza mediante un proceso de interpolación, cuya precisión depende de la potencia de cálculo del microprocesador y del algoritmo que se emplee, lo que otorga un papel preponderante al software en estos sistemas y por este motivo una gran operatividad en futuros cambios. Además de estas ventajas, el microprocesador es el elemento idóneo en esta área, porque también permite una reducción del costo del equipo y una mejora sustancial del mantenimiento.

3.5.3. INDUSTRIA DE AUTOMOCIÓN

Quizá sea esta área y, sobre todo, la relacionada con la fabricación y control de automóviles, la más idónea y una de las pioneras de la implantación de los sistemas con microprocesadores en los productos terminados. Las características que permiten esta adaptación son los grandes volúmenes de producción, el alto precio relativo del producto, la responsabilidad y popularidad del mismo y la potenciación de que puede dotarse un vehículo mediante un microprocesador. Por la relación de proyectos que se exponen, no sólo los automóviles, sino también camiones, barcos y aviones presentan grandes expectativas para el empleo de sistemas con microprocesador.

La industria del automóvil está impulsando a otras hacia la utilización de sistemas microinformáticos y no sólo en el campo del producto terminado, sino también en la automatización de los diferentes procesos de la producción.

Los logros conseguidos por los japoneses, pioneros en la robotización e incorporación de autómatas programables a todo tipo de maquinaria, obligan a los restantes fabricantes mundiales a adoptar estas mismas soluciones con el fin de abaratar los costos de la mano de obra e incrementar substancialmente la productividad y la calidad de los coches.

Los problemas que plantea el costo creciente del petróleo están siendo enfrentados y amortiguados por los fabricantes de coches aumentando considerablemente las prestaciones de los mismos, en aspectos tan fundamentales como la seguridad, la comodidad, el control total de consumo, etc. Por otra parte, las legislaciones particulares de los estados son cada vez más exigentes en cuestiones como contaminación, consumo y seguridad.

Entre los elementos que dispondrá el coche del futuro, algunos ya los incorporan ciertos modelos hoy existentes. Así en la actualidad ya se conocen los computadores de viaje, el cambio de marcha automático, los controles para la optimización del gasto del combustible y la contaminación, los sistemas antirrobo sofisticados, los frenos antideslizantes y los radios digitales con exploración automática de sintonía y su memorización. A continuación se destacan algunos aspectos en los que intervienen microprocesadores:

- Control de la inyección de gasolina y tiempo de ignición
- Control de la salida de gases de combustión
- Instrumentación y panel de mando
- Alarmas de aviso de averías
- Multiplexado del cableado eléctrico
- Mantenimiento y diagnóstico
- Cambio automático de marchas
- Controles para la presión de los neumáticos
- Pantallas digitales de video
- Sistemas de seguridad, tales como el control sobre los cinturones de seguridad y empleo de bolsas de aire
- Sintetizador de voz
- Controles de clima
- Asientos anatómicos adaptables
- Distribución del peso para regular la carga sobre los ejes

3.5.4. ELECTROMEDICINA

La electromedicina constituye uno de los campos más a propósito para la introducción de computadores y sistemas con microprocesador. Esto se puede encontrar en sistemas de monitorización de parámetros fisiológicos, mesas de radiología, sistemas microcomputadores para el control de un generador de rayos X, regulación del motor que guía el ánodo del tubo de rayos X, marcapasos microcomputarizado, sistemas de diagnóstico. A continuación se explica un sistema para monitoreo de parámetros fisiológicos de los pacientes.

Sistema para monitoreo de parámetros fisiológicos

Se utiliza un microprocesador para monitorear en forma continua el estado de los signos vitales de cada paciente, incluyendo el ritmo cardíaco, presión de sangre, pulso, electrocardiograma (ECG), respiración y temperatura corporal. El sistema proporciona datos a un registro ECG y una impresora, la cual da la identificación del paciente y su estado general de salud. En caso de cualquiera de las pruebas indique condiciones anormales, los parámetros fallidos se transmitirán a un puerto de salida para indicación de alarma, el cual cuenta con una alarma de audio y un indicador en el panel de control del estado del paciente en la sección asignada a las enfermeras. Esto se ilustra en la figura 3.22.

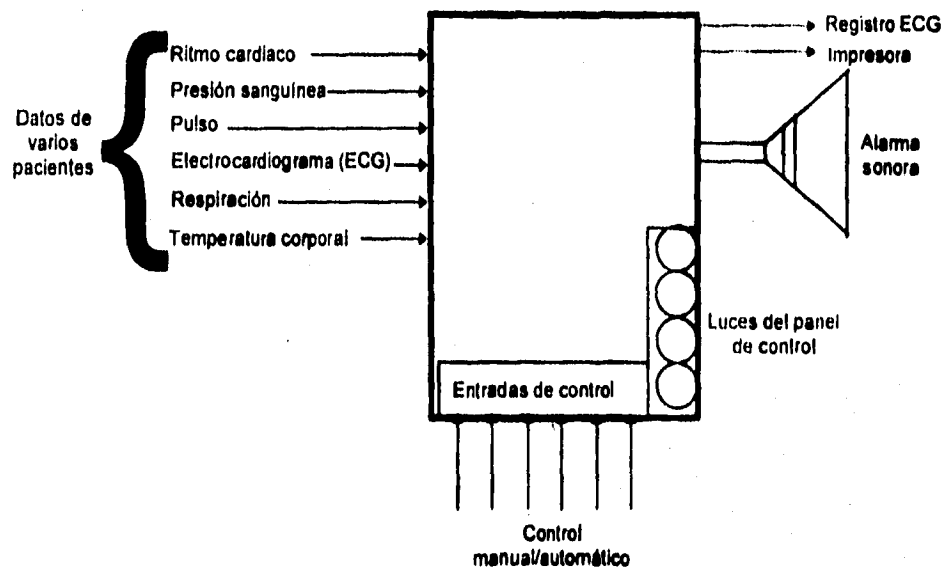


Figura 3.22. Sistema para monitoreo de pacientes.

3.5.5. INSTRUMENTACIÓN

Los fabricantes de aparatos de medida e instrumentación más importantes del mundo, como Hewlett Packard, han comenzado a emplear masivamente el microprocesador como parte integrante de sus productos terminados. A continuación se presentan algunos instrumentos existentes en el mercado que incorporan microprocesadores.

Balanza peso-precio

Esta balanza mide exactamente el peso, calcula los importes parciales y totales y memoriza el conjunto de operaciones realizadas.

Puente de medida para componentes

Sirve para medir resistencia óhmica, capacitancia e inductancia, con una precisión del 0.1 %.

Paneles visualizadores

Los paneles visualizadores son empleados para informar al público de resultados, puntuaciones, tiempos o acontecimientos en espectáculos, bancos, campos de deporte, etc.

Osciloscopio de doble trazo

La inclusión del microprocesador en los osciloscopios les ha añadido importantes ventajas. Tal es el caso del modelo 1722 B de Hewlett Packard, que cubre una banda de 275 Mhz, con una velocidad de barrido de 1 ns/div. y 5 funciones de visualización, para la medida exacta de tiempos, proporcionando el cálculo directo de la frecuencia, el voltaje de corriente continua, voltaje instantáneo y porcentaje de amplitud.

En el campo de los osciloscopios también puede citarse el modelo PM 3263 de PHILIPS, que permite medidas exactas de incrementos de tiempo en una señal o en señales distintas. Además realiza directamente el cálculo de la frecuencia y detecta cualquier tipo de error en el manejo del aparato, así como en el comportamiento de los circuitos internos.

Voltímetros digitales

Actualmente se pueden encontrar en el mercado diversas versiones de multímetros digitales, dotados de un sistema microcomputador, entre los que se citan:

- Multímetro digital Fluke, de 4 ½ dígitos, que permite el funcionamiento en modo Offset y en modo dB. En el primero se realiza la compensación de la resistencia de los terminales o establece un voltaje como referencia para la medida. En el modo dB indica el nivel absoluto (dBm) de señales de corriente continua y corriente alterna directamente.
- El multímetro MX 550, de Metrix, proporciona la conversión analógica-digital y la conmutación automática de las escalas y además permite medir capacitancias, frecuencias y temperaturas.
- El voltímetro digital 3455A, de Hewlett Packard, de 5 ½ ó 6 ½ dígitos, con autocalibración. La autocalibración asegura las medidas de ohms y volts en corriente continua y simplifica la calibración de estas magnitudes, corrigiendo cualquier error. También dispone de autoverificación, para ayuda al servicio de mantenimiento y reparación. Las medidas de tensiones continuas (DCV) pueden ser hechas con una sensibilidad de 1 microvolt y las de tensiones alternas (ACV) alcanzan frecuencias comprendidas entre 30 Hz y 250 kHz.

Contadores cronológicos

La inclusión del microprocesador en estos instrumentos les confiere unas características hasta hoy insospechadas. Así, por ejemplo, el modelo 5370A de Hewlett Packard dispone de una resolución de 20 pseg. y alcanza frecuencias de hasta 100 Mhz. Posee calibración automática para errores sistemáticos y visualiza desviaciones estándar, máximos, mínimos, etc.

Generadores de funciones

Estos aparatos están destinados ante todo a generar cualquier tipo de ondas (senoidales, diente de sierra, cuadradas, etc.), con diferentes frecuencias. Una variante muy interesante, atendiendo al principio básico de funcionamiento, para aplicación del microprocesador, es el *sintetizador*, el cual proporciona las diferentes señales tratando aritméticamente la frecuencia fundamental originada por un oscilador controlado por un cristal de cuarzo. A este tipo corresponde el sintetizador de Hewlett Packard modelo 8062A. Este generador cubre una banda de frecuencias comprendida entre 10 kHz y 1,280 Mhz, tanto en AM como en FM y con una resolución de frecuencia de 0.1 Hz. Entre 13 y -120 dB la precisión alcanza 1 dB mediante la corrección del sistema microcomputador.

Comprobador de circuitos integrados

Mediante un sistema de 3 microprocesadores, Gen Rad ha diseñado un equipo capaz de verificar cualquier tipo de circuito integrado, como microprocesadores, en tecnología bipolar, PMOS o NMOS. Dicho instrumento dispone de una pantalla de visualización, un teclado y diferentes zócalos como base de pruebas. Los programas específicos para cada tipo de circuito integrado se almacenan en cassette y trabaja con una velocidad de 2 Mhz.

Programador de memorias PROM

La inclusión de un microprocesador a los instrumentos encargados de la grabación de las memorias del tipo PROM, les dota de relevantes características, que simplifican enormemente su manejo.

Estos programadores pueden trabajar con diferentes formatos de datos y pueden ser controlados, además del teclado propio, a través de control remoto y por computador. También son capaces de grabar varias memorias simultáneamente.

Capítulo 4

Diseño del tutorial

4. DISEÑO DEL TUTORIAL

4.1. DISEÑO DE MICRO

El Sistema Tutorial sobre Microprocesadores utilizando Multimedia diseñado para el presente trabajo de tesis fue bautizado como **MICRO**. Dado que este sistema está destinado al uso de estudiantes de Ingeniería donde se requiera aprender el tema de microprocesadores, como por ejemplo Ingeniería Electrónica, Ingeniería en Computación, Ingeniería en Sistemas, etc., **MICRO** está elaborado de manera didáctica. La plataforma sobre la cual está diseñado **MICRO** es *Windows*, de manera que el usuario se mueve a través de la *interfaz de usuario* del tutorial oprimiendo botones que aparecen en la pantalla, haciendo "click" sobre éstos con el botón izquierdo del mouse.

Después de una pantalla de presentación, aparece una pantalla de bienvenida donde se pide al usuario que escriba su nombre. A continuación, el usuario se encuentra con el *Menú Inicial*, donde aparece una *Ayuda* que muestra la función de los botones ahí desplegados. También se tiene la llamada *Utilización del Tutorial*, que explica cómo usar **MICRO**. Además se cuenta con un *Índice Temático*, desde el cual se puede acceder directamente a los diferentes temas que expone el tutorial. Para que el usuario pueda incrementar sus conocimientos sobre microprocesadores se cuenta con una *Bibliografía*, que contiene títulos de libros sobre el tema. En la barra inferior de botones se hay un botón de *Hora*, que se puede presionar en cualquier parte del tutorial para ocultarlo y que da la hora actual y un botón marcado con una "mu" griega (la μ es el símbolo de **MICRO**) que se puede presionar también desde cualquier parte del tutorial y que accede al *Menú Inicial*. En esta barra además se aprecia el botón de *Salir*, también presionable desde cualquier parte del tutorial y que como su nombre lo dice, sirve para salir del sistema; al presionarlo se termina la sesión, se presenta una pantalla de créditos y se va al *Administrador de Programas de Windows* (o al *Administrador de Archivos*, si es que de ahí se ejecutó el programa).

Desde la barra inferior de botones se puede presionar el botón de *Sección de Consulta*, y de manera inmediata se accede dicha sección donde el usuario podrá consultar uno a uno los diferentes temas que **MICRO** expone sobre microprocesadores. En la barra inferior de botones de esta sección aparecen un botón de *Notas*, que al presionarlo permite anotar recordatorios, un botón de *Ayuda*, marcado con un signo de interrogación (?), que ilustra sobre la función de los botones que aparecen en la *Sección de Consulta*, y el botón de *Retorno*, marcado con una flecha estilizada, que sirve cuando se accedió a un tema de la *Sección de Consulta* desde el *Índice Temático* del *Menú Inicial*, y cuya función es regresar al *Índice Temático*.

Entrando a la *Sección de Consulta* aparece el *Menú de Consulta*, con cinco subsecciones que son:

- **El Microprocesador.** Donde se presentan los principales conceptos referentes al microprocesador.
- **Memoria.** En esta subsección se pueden acceder seis diferentes temas relacionados con la memoria.
- **Periféricos.** Subsección en la que se localiza la información referente a los periféricos.
- **Interrupciones.** Se explica qué son las interrupciones y se habla de sus diferentes variantes.
- **Aplicaciones.** Aquí se habla de la gama de aplicaciones de los microprocesadores y se dan ejemplos explicados.

Presionando los botones correspondientes se puede acceder a dichas subsecciones.

Al acceder a la subsección de *El Microprocesador*, aparece el llamado *Menú del Microprocesador* con los siguientes temas:

- **¿Qué es un microprocesador?**. Se explica de manera sencilla y didáctica en qué consiste un microprocesador.
- **Historia del microprocesador**. Este tema habla de la historia de los microprocesadores desde su creación hasta el momento actual, mencionando las principales pautas de este desarrollo.
- **El microcomputador y el sistema con microprocesador**. Como una introducción pedagógica a los microprocesadores, se explican los conceptos básicos del microcomputador, principal aplicación de los microprocesadores hoy en día, y a manera de establecer una comparación que clarifique la manera de usar estos dispositivos, se habla de la estructura básica de un sistema con microprocesador.
- **Arquitectura del microprocesador**. Este es un extenso tema que se desarrolla explicando en qué consiste la arquitectura del microprocesador, partiendo de un microprocesador genérico con características didácticas. Se enumeran los componentes de la arquitectura y en un diagrama de la misma el usuario o estudiante puede hacer "click" con el mouse sobre las diferentes partes del esquema, de forma que aparecerán explicaciones con gráficos, texto y animaciones sobre las diferentes partes de un microprocesador.
- **Operación del microprocesador**. Aquí se explica paso a paso como opera un microprocesador, explicando conceptos como el *Ciclo de Máquina*, *Ciclo de Instrucción*, etc., y mediante animaciones se ilustra un ejemplo que muestra como es el manejo de información de un microprocesador con la memoria a través de sus componentes internos.
- **Modelo de programación**. En este tema se explica lo que es el modelo de programación y se encuentran detallados los registros que sirven al programador de microprocesadores para realizar su labor.

Accediendo a la subsección de **Memoria**, aparece el **Menú de Memoria** y se despliegan los siguientes temas:

- **¿Qué es la memoria?**. De manera didáctica este tema explica qué es la memoria y su utilidad para el microprocesador.
- **ROM y RAM**. Se detallan los diferentes aspectos que ilustran el trabajo de las memorias ROM y RAM, las cuales son fundamentales en el desempeño de los microprocesadores.
- **Mapas de memoria**. Aquí se detalla la distribución generalizada de memoria atendiendo al microprocesador genérico ilustrado en **MICRO**.
- **El stack**. En este tema se explica lo que es el stack y mediante texto, gráficos y animaciones se explican sus dos variantes, el stack en cascada y el stack tipo puntero.
- **Repertorio de instrucciones**. Todos los microprocesadores tienen un set o repertorio de instrucciones. Aquí se detalla el repertorio de instrucciones del microprocesador genérico ilustrado en **MICRO**, atendiendo al tipo de instrucción.
- **Modos de direccionamiento**. Se ilustra la composición de una instrucción de microprocesador detallando sus diferentes campos, para así poder explicar uno a uno los distintos modos de direccionamiento usados en sistemas electrónicos que incluyen microprocesador.

Entrando a la subsección de **Periféricos**, se despliega el **Menú de Periféricos** de donde se puede acceder a los siguientes temas:

- **¿Qué son los periféricos?**. En este tema se explica sencillamente qué son los periféricos y se dan algunos ejemplos.

- **¿Qué son los puertos?**. Aquí se explica de manera didáctica qué son los puertos de los microprocesadores.
- **Puertos paralelo**. Se explica mediante texto y gráficos qué son los puertos paralelo y las diferentes maneras de implementar su funcionamiento.
- **Puertos serie**. Se da una explicación que ilustra qué son los puertos serie.
- **Temporizadores**. Se habla de una manera sencilla y explicativa sobre los temporizadores y contadores.

Accediendo a la subsección de *Interrupciones* aparece el *Menú de Interrupciones* que despliega los siguientes temas:

- **¿Qué son las interrupciones?**. En este tema se explica qué son las interrupciones, las cuales son de un uso muy importante en el manejo de microprocesadores.
- **Interrupciones hardware**. Se explica qué son las interrupciones hardware, ilustrando sus diferentes variantes mediante texto y gráficos.
- **Interrupciones software**. En este tema se habla claramente sobre las interrupciones software.

Y finalmente, al acceder a la subsección de *Aplicaciones*, aparece el *Menú de Aplicaciones* que da paso a los siguientes temas:

- **Aplicaciones del microprocesador**. Se habla sobre el impacto del microprocesador en muchas áreas, de lo que derivan aplicaciones en industrias muy diversas.
- **Industria en general**. En la industria en general hay múltiples aplicaciones de los microprocesadores; aquí se enumeran estas aplicaciones y se explica un ejemplo de aplicación.
- **Industria de automoción**. Se habla de las múltiples aplicaciones de los microprocesadores en esta industria.
- **Electromedicina**. En este tema se enumeran las aplicaciones de microprocesadores a la electromedicina y se ilustra su uso mediante un ejemplo.
- **Instrumentación**. Se ilustra el uso de los microprocesadores en la instrumentación mediante la explicación de las características de aparatos de instrumentación y medida existentes en el mercado.

Los temas de la *Sección de Consulta* son explicados con pantallas de texto, gráficos y animaciones, de manera que al ser consultados el usuario recibe la información de una manera muy didáctica.

4.2. NAVEGACIÓN DEL TUTORIAL

La navegación del sistema es, como su nombre lo indica, la forma y posibilidades de circulación dentro del mismo. Para visualizarla se presentan dos tipos de diagramas: el primero es el de *diagrama de flujo*, que como en cualquier programa muestra, usando bloques, el flujo de la información durante la ejecución del mismo. El segundo tipo es el *diagrama de árbol*, que indica mediante círculos el despliegue de la información, así como las variables utilizadas en el sistema para efectuar saltos. Ambos diagramas plantean el esqueleto del diseño general del sistema.

Por la complejidad del sistema, los diagramas de flujo y de árbol se dividen entre sí en sucesivos diagramas de la misma naturaleza. En la figura 4.1. se muestra la navegación general del sistema representada con un diagrama de flujo.

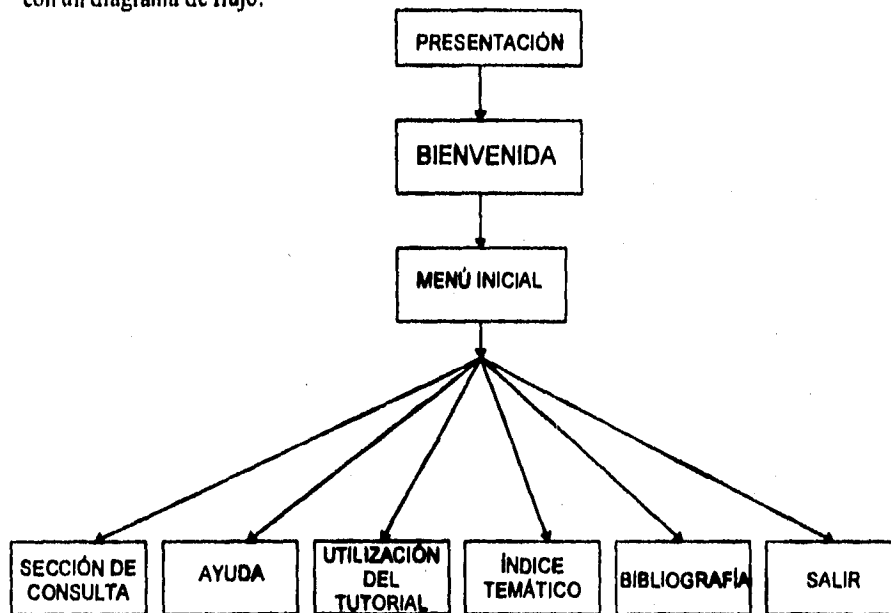


Figura 4.1.

En la siguiente figura se puede apreciar el diagrama de flujo correspondiente a la *Sección de Consulta*.

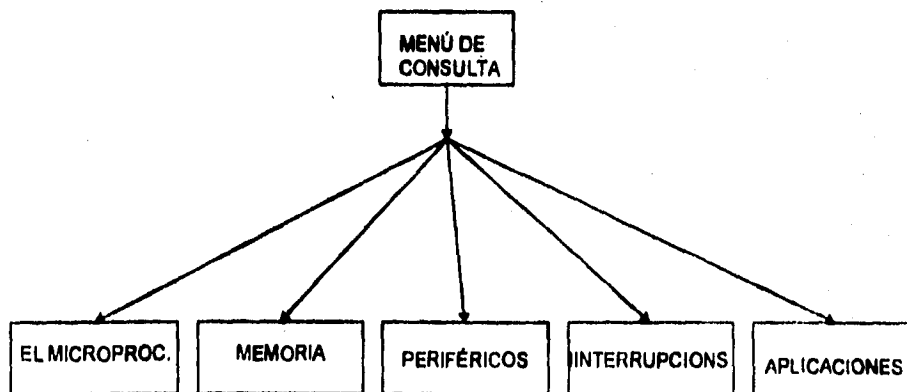


Figura 4.2.

Los diagramas de flujo individuales para las subsecciones de la *Sección de Consulta* se muestran en las siguientes figuras. A continuación se muestra el diagrama de flujo de la subsección de *El microprocesador*.

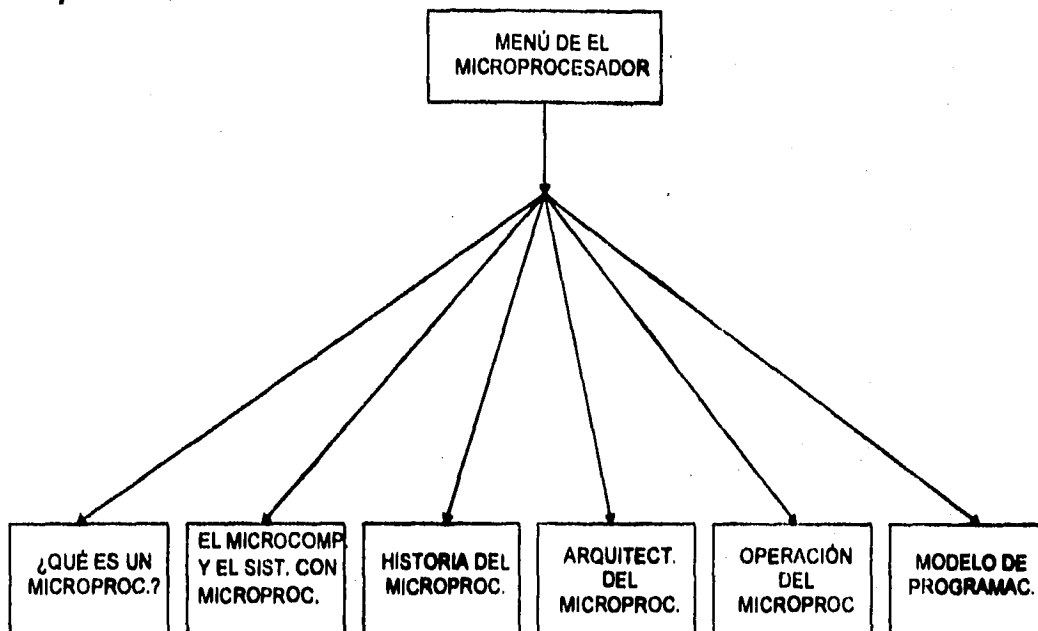


Figura 4.3.

En la siguiente figura se muestra el diagrama de flujo de la subsección de *Memoria*.

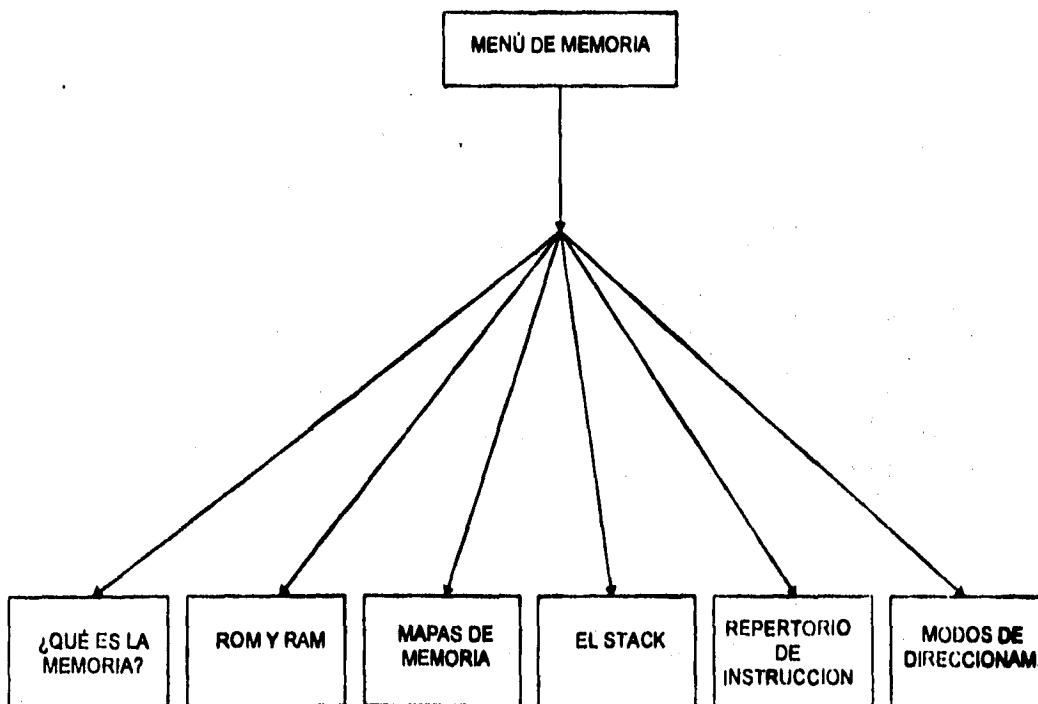


Figura 4.4.

En la siguiente figura se aprecia el diagrama de flujo de la subsección de *Periféricos*.

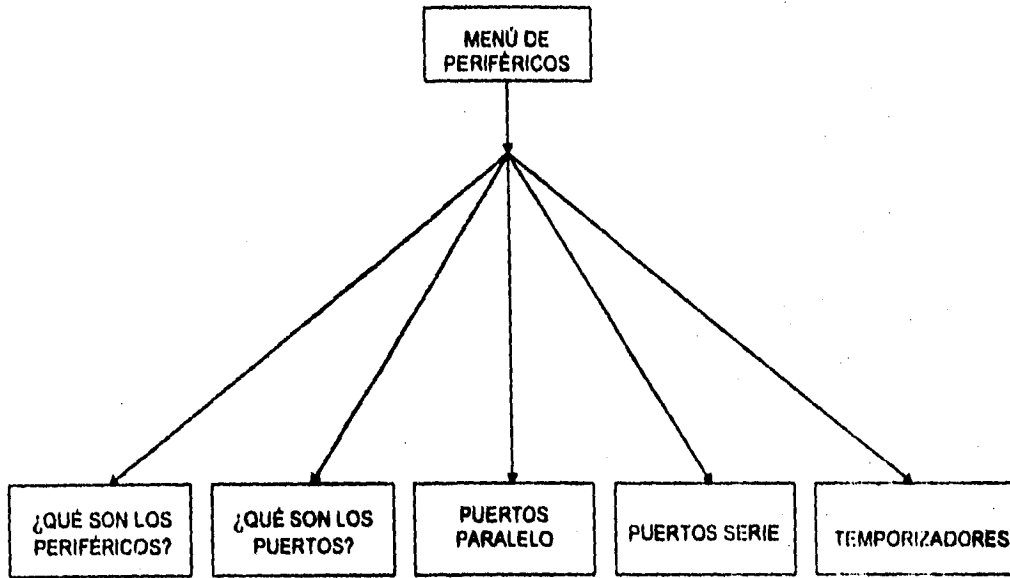


Figura 4.5.

A continuación se presenta el diagrama de flujo de la subsección de *Interrupciones*.

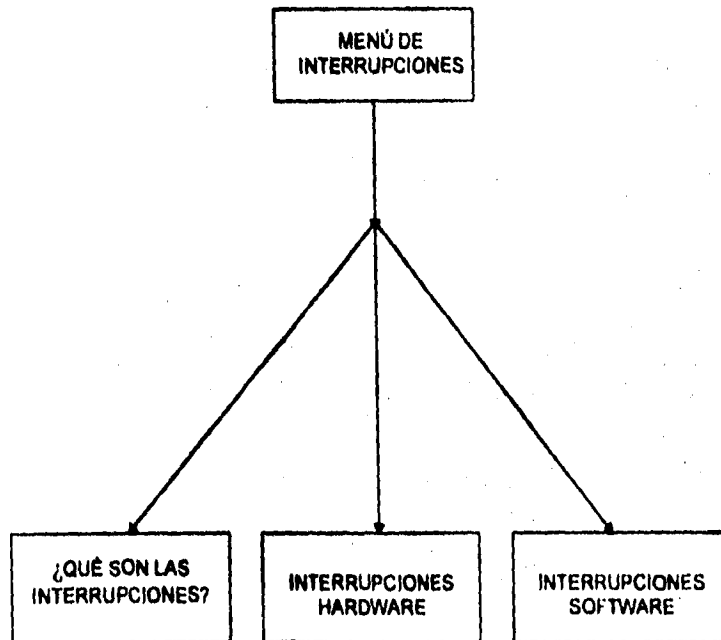


Figura 4.6.

A continuación se presenta el diagrama de flujo de la subsección de *Aplicaciones*.

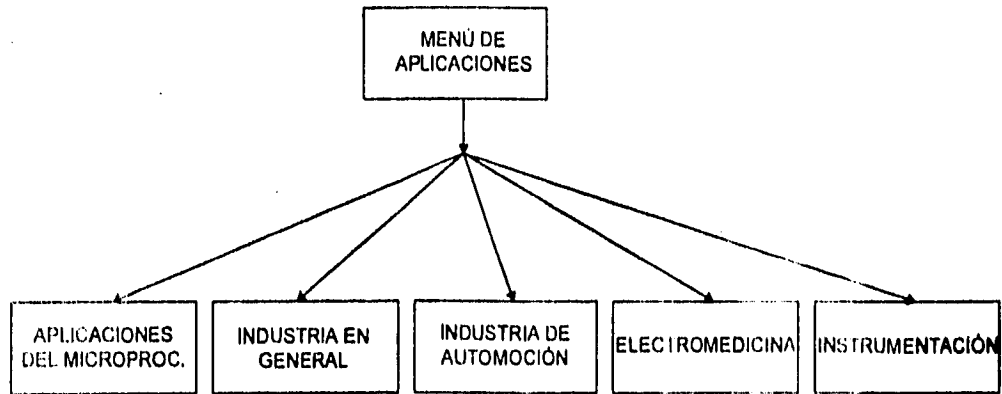


Figura 4.7.

La navegación general del sistema tutorial se muestra en el diagrama de árbol de la siguiente figura.

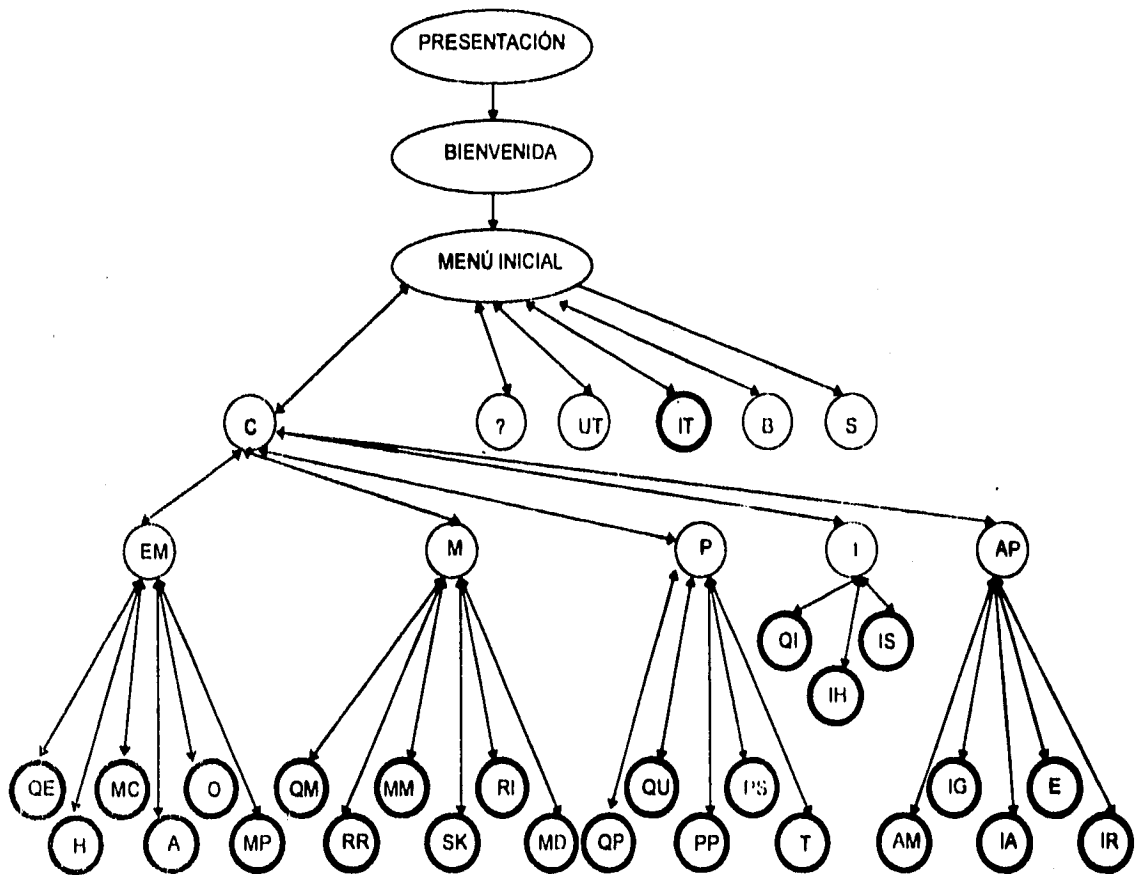


Figura 4.8.

En la figura 4.8. el *Índice Temático* y todos los temas de la *Sección de Consulta* están remarcados; esto quiere decir que a partir del *Índice Temático* el usuario puede navegar hacia todos esos temas y de regreso. En la siguiente tabla se indican los significados de las abreviaciones de la figura anterior mediante una leyenda donde aparecen las iniciales contenidas en los círculos del diagrama de árbol.

C	Sección de Consulta
?	Ayuda
UT	Utilización del Tutorial
IT	Índice Temático
B	Bibliografía
S	Salir
EM	El microprocesador
M	Memoria
P	Periféricos
I	Interrupciones
AP	Aplicaciones
QE	¿Qué es un microprocesador?
H	Historia del microprocesador
MC	El microcomputador y el sistema con microprocesador
A	Arquitectura del microprocesador
O	Operación del microprocesador
MP	Modelo de programación
QM	¿Qué es la memoria?
RR	ROM y RAM
MM	Mapas de memoria
SK	El stack
RI	Repertorio de instrucciones
MD	Modos de direccionamiento
QP	¿Qué son los periféricos?
QU	¿Qué son los puertos?
PP	Puertos paralelo
PS	Puertos serie
T	Temporizadores
QI	¿Qué son las interrupciones?
IH	Interrupciones hardware
IS	Interrupciones software
AM	Aplicaciones del microprocesador
IG	Industria en general
IA	Industria de automoción
E	Electromedicina
IR	Instrumentación

Capítulo 5

Implementación

5. IMPLEMENTACIÓN

5.1. HARDWARE Y SOFTWARE USADOS EN LA ELABORACIÓN DE MICRO

MICRO fue construido para trabajar en la interfaz gráfica *Windows*, pues en México la mayoría de las computadoras utilizadas son IBM PC's o compatibles y actualmente la mayoría de éstas cuentan con procesador 80386 o superior lo que hace que la interfaz gráfica más popular sea *Windows*. Por lo antes mencionado, en cuestión de Hardware fue necesario lo siguiente:

- Computadora personal compatible IBM 486, reloj de 66 Mhz
- Mínimo de 16 Mbytes de RAM.
- Monitor VGA a color o superior
- Tarjeta de video VGA o superior
- Mouse
- Drive de 3 ½" de alta densidad
- Disco duro de 1Gb

Para la elaboración de **MICRO** se utilizó el programa de autoraje *Authorware* que trabaja sobre plataforma *Windows 3.1* o superior, de manera que en general fueron necesarios los siguientes recursos software:

- MS-DOS 6.2.
- Windows 3.1.
- Authorware Professional
- Word para Windows 2.0
- Microsoft Draw
- Corel Draw
- Paintbrush
- Corel Photo Paint
- Paint Shop Pro
- 3D Studio

5.2. HARDWARE Y SOFTWARE NECESARIOS PARA PODER USAR MICRO

Como se mencionó anteriormente, **MICRO** fue elaborado para trabajar sobre plataforma *Windows*. Esto hace necesarios para su uso ciertos recursos de hardware y software.

MICRO presenta ciertas limitantes, pues en caso de que la velocidad de procesamiento del microprocesador de la computadora con la que se está trabajando sea baja (como es el caso de computadoras con procesador 80286 y 80386), la presentación gráfica será algo lenta, lo que afecta en consecuencia el desempeño del tutorial.

Otra limitante de **MICRO**, dado lo anteriormente mencionado, es que el usuario deberá contar con *Windows 3.1*. Esto último acarrea una limitante más referente a los recursos de hardware con que deberá contar la máquina del usuario para poder correr *Windows 3.1*.

Los recursos hardware requeridos para usar **MICRO** son:

- IBM PC o compatible con procesador 80386 o superior.
- Un mínimo de 2 Mbytes en memoria RAM (de preferencia más de 4 Mb)
- Mouse
- Un mínimo de 4.2 Mb de espacio en disco duro para instalación
- Drive de 3 1/2" de alta densidad
- Monitor VGA (640 x 480 a 16 colores)

El software necesario para utilizar **MICRO** es:

- MS-DOS 3.3, o superior
- Windows 3.1.

Como se puede ver, los recursos necesarios para hacer utilizar **MICRO** son los de una PC típica, lo que significa que no se requiere de ningún equipo especial o muy costoso. Esto permite que el sistema sea 100% transportable entre una PC y otra, de manera que el usuario no tiene que depender de una computadora en particular para hacer uso de **MICRO**.

5.3. ELABORACIÓN DE LA INTERFAZ DE USUARIO DE MICRO

Par diseñar interfaces de software educativo multimedia, la clave consiste en el pleno conocimiento del usuario al que estará orientada la aplicación. No solamente es importante la información; también hay que tomar en consideración las expectativas y la apreciación completa de la interfaz por parte del usuario "común" del sistema, para lo cual se pueden formular preguntas como: ¿Cuáles son los pasos que deberá realizar el usuario para obtener resultados de las expectativas que tiene sobre el sistema?, ¿Cómo usará el usuario la aplicación?, etc. Un camino muy eficiente para contestar estas preguntas, como se mencionó en el capítulo 2 de este trabajo, es estudiar al usuario en acción. Se desarrollan diferentes escenarios que describen diversos caminos (interfaces) de trabajo con la aplicación, estableciendo básicamente lo que el usuario necesitará hacer para trabajar con el sistema. Este proceso de análisis es comúnmente llamado *Análisis de Tareas del Usuario (ATU)* donde el usuario es el punto principal de análisis. Cuando el diseñador ha generado un "mapa o árbol de navegación" -estructura usada en el diseño de software multimedia (Lara y Guerrero, 1994)- producto del ATU, donde se encuentran contempladas todas las variantes del trabajo desarrollado por el usuario para navegar a través de la aplicación, entonces estará listo para realizar la interfaz óptima de dicho sistema. En el caso de **MICRO**, los árboles de navegación mencionados fueron realizados pensando en el comportamiento de los estudiantes al usar el sistema; estos árboles utilizados en el ATU se pueden apreciar en el capítulo 4. Par implementar una aplicación educativa multimedia es necesario seguir una metodología en el diseño de la interfaz de usuario. En el caso de esta metodología, la interfaz de usuario debe contar con las siguientes características:

- Hacer fácil la interacción del usuario con ella
- Consistencia
- Simplicidad
- Uso de metáforas
- Secuencia y jerarquías funcionales
- Dar el control al usuario
- Confirmación de tareas y retrocesos

5.3.1. FACILIDAD DE INTERACCIÓN

En el diseño de interfaces de usuario es necesario ocupar cierto tiempo en considerar los problemas que se puedan presentar en la interacción usuario-interfaz. En el caso de **MICRO**, esta es una de las razones por las que se eligió el ambiente de trabajo *Windows*, pues al ser un ambiente gráfico permite el uso de botones gráficos; la interfaz de **MICRO** está elaborado en base a botones gráficos y áreas de toque, lo que hace que el aprendizaje de uso del sistema sea más intuitivo.

5.3.2. CONSISTENCIA

Una de las consideraciones sobreentendidas en el diseño de la interfaz de usuario es la consistencia. Esta consistencia se refleja en el desempeño del usuario en la aplicación, así como en las alternativas que tiene para avanzar. Dentro de las aplicaciones multimedia es importante que se cuente con una estructura basada en el mapa de navegación, que le permita al diseñador de la aplicación definir los nodos o bloques en los cuales se desarrollará una acción de control, donde la intervención del usuario es la que determinará la ruta de trabajo del sistema. Estos nodos deberán estar diseñados con los mismos mecanismos de trabajo, de tal forma que el usuario, en cualquier nivel en el que se encuentre (nodo), cuente con las herramientas cognitivas necesarias para avanzar, retroceder o abortar el sistema.

Una forma de explicar fácil y sencillamente la consistencia es la siguiente: la interfaz de usuario debe verse y operar exactamente igual para cada parte de la aplicación. Una vez que el usuario ha descubierto como funciona la interfaz en una sección, espera que funcione de la misma forma en toda la aplicación.

MICRO fue diseñado e implementado atendiendo rigurosamente a la característica de consistencia. En sus distintas partes cuenta con ayudas que explican como navegar en el tutorial, y las secciones del mismo nivel cuentan con interfaces similares, de manera que una vez que el usuario ha recorrido una sección, de manera intuitiva podrá recorrer las demás.

5.3.3. SIMPLICIDAD

Un principio general de todos los tipos y estilos de sistemas es que el diseño simple siempre es el mejor, principio que es aplicable a las interfaces de usuario de aplicaciones multimedia. Si el concepto de la aplicación es simple, entonces la interfaz también puede serlo y por consiguiente de fácil manejo para el usuario. Si la aplicación es suficientemente compleja es necesario tratar de hacer que su operatividad sea más simple. Uno de los mejores caminos para lograrlo es crear una lista de las actividades y un conjunto de herramientas que auxilien al usuario en la operación; posteriormente a la creación de la lista, sus elementos deberán ser priorizados de acuerdo a la frecuencia de recurrencia del usuario. Las actividades y herramientas de las aplicaciones deben ser utilizadas mediante el menor número de pasos posible. Todas aquellas actividades y herramientas que sean de menor prioridad, pueden ser incluidas en diferentes grupos de operación o trabajo como: sub-menús, opciones, barras de herramientas, configuraciones, etc.

Si se tiene un concepto simple de aplicación, entonces la interfaz también será simple y, en consecuencia, más fácil para el usuario. Con una aplicación y una interfaz simples, el usuario no tiene que preocuparse de lo que hace la aplicación a fin de dar su siguiente paso. El comportamiento debe ser natural y la siguiente acción a realizar debe ser obvia para el usuario basado en lo que ve y escucha.

Dado que **MICRO** fue elaborado principalmente para el uso de un conjunto de estudiantes, no se incorporó audio en el sistema tutorial, pues no es común que en sus computadoras los estudiantes promedio cuenten con tarjeta de audio. De cualquier manera, la interfaz de usuario de **MICRO** está basada en las acciones que ésta conlleva, esto es, los botones gráficos están indicados con texto, las iniciales de la acción que se desea realizar, o bien, dibujos que ilustran la acción. Además, en el *Menú Inicial* de **MICRO** se cuenta con la llamada *Utilización del Tutorial*, donde se explica cómo navegar a través del sistema tutorial, y con botones de *Ayuda*, que explican la función generalizada de los botones.

5.3.4. USO DE METÁFORAS

Un principio importante en el diseño de las interfaces de usuario es el uso de las representaciones asociadas con un conocimiento generalizado, esto es, el aprovechamiento de las similitudes que puede tener una interfaz con algunos dispositivos de uso común. Las ventajas que esto conlleva se reflejan principalmente en la adaptación casi instantánea que tiene el usuario con las aplicaciones. Por ejemplo, si en una aplicación requerimos que el usuario active una secuencia de imágenes ya sean fijas, animaciones o video, es factible desarrollar una interfaz, que está relacionada con un dispositivo que tenga un alto grado de probabilidad de ser conocido por el usuario; podríamos crear una interfaz que sea similar a la de los dispositivos VCR (video cassette recorder): un botón de avance, uno de retroceso, pausa, etc. Teniendo un diseño de esta naturaleza, el usuario sabrá inmediatamente después de percibir la interfaz (visualmente), como trabaja el sistema, pues asocia la interfaz con el funcionamiento que representan los símbolos (gráficos). En muchos casos, los diseños de interfaces también pueden recurrir a otro tipo de metáforas, que pueden ser las barras de scroll o los menús de acción para activar el funcionamiento del sistema. Aunque estas interfaces son probablemente menos conocidas (usadas) por los usuarios, el sistema debe estar diseñado para instruir al usuario en la forma de usarlas, recordando que el principio de consistencia es siempre más importante que el uso de metáforas.

En el caso de **MICRO**, se utilizan metáforas como los botones de avance y retroceso, similares a los de una grabadora, utilizados para ir en el despliegue de páginas hacia delante o hacia atrás.

5.3.5. SECUENCIA Y JERARQUÍAS FUNCIONALES

Una de las principales tareas es llevar al usuario, mediante la interfaz, a realizar una o más selecciones del sistema. Existen varias estrategias a este respecto que dependen de cuántas selecciones tenga el sistema o la dificultad de explicarle al usuario qué representa cada selección. Normalmente no se deben presentar al usuario dos selecciones dentro de una, pues le causaría abrumación. Si se tiene un gran número de selecciones, se puede usar una organización jerárquica para presentarlas.

Un menú estructurado jerárquicamente realiza varias acciones ordenadas en cada selección, mismas que dependen del número de niveles jerárquicos. En un menú de trabajo es conveniente no tener muchas selecciones. Esto se fundamenta en la importancia que tiene la identificación adecuada de cada alternativa. Generalmente un menú de trabajo debe contener sólo la información suficiente para que el usuario comprenda lo que implica cada selección. Esto es muy importante en los menús de primer nivel, donde las alternativas de selección que tiene el usuario son completamente diferentes una de otra. Conforme el usuario vaya escalando a niveles jerárquicos superiores, las selecciones tendrán una relación más estrecha.

En el diseño de la interfaz de usuario de MICRO, las selecciones para el usuario están ordenadas jerárquicamente. Por ejemplo, en el primer nivel de la *Sección de Consulta* sólo hay 5 opciones. Sigue otro nivel de selección, donde se exponen los temas y en algunos casos, estos temas se bifurcan en subtemas, caso en el cual usualmente hay más selecciones disponibles.

5.3.6. DAR EL CONTROL AL USUARIO

Los usuarios de aplicaciones pueden estar más seguros frente a ellas cuando sienten que tienen el control. El sistema deberá responder a sus órdenes, de tal forma que arranque cuando el usuario quiera y esté preparado, se detenga, se adelante, se retrase, etc. cuando el usuario lo desee. El usuario nunca debe estar frente a alguna acción o proceso inesperado. Comúnmente sucede que alguna aplicación contiene una instrucción que implica abortar el sistema; si esta instrucción se activa cuando el usuario espera alguna respuesta de trabajo y operación de la aplicación, su reacción será la pérdida de interés por el sistema. En estos casos es importante habilitar controles específicos, bien documentados y explicados al usuario, para que haga uso de ellos, siempre y cuando así lo desee. Esta característica debe ser considerada desde el inicio del diseño.

MICRO da un control total de las acciones al usuario, pues su interfaz está planeada para responder a todas las acciones que desencadene el usuario, sin reaccionar de manera inesperada. En el caso de que el usuario no sepa como avanzar, retroceder o abortar el sistema, se cuenta con una *Ayuda* que explica como llevar a cabo estos procedimientos.

5.3.7. CONFIRMACIÓN DE TAREAS Y RETROCESOS

El resultado de la selección de un usuario debe tomar cierto tiempo en realizarse. Dentro de este retardo el sistema debe indicarle al usuario la alternativa que ha seleccionado y que está en proceso de ejecución. Por otro lado, este tiempo es importante para el usuario, pues tendrá la oportunidad de reflexionar y asegurarse de su selección para continuar o retroceder en la navegación.

En MICRO se ha tomado en cuenta esta característica. La interfaz de usuario del sistema tutorial hace cambios de cursor que permiten identificar las selecciones y su tiempo de retardo. Al ubicar el cursor sobre las áreas de toque y la mayoría de los botones de MICRO, el cursor pasa de ser una flecha a una mano; una vez presionando el botón izquierdo del mouse, el botón gráfico se hunde, o, en la mayoría de los casos, cambia a colores inversos. Durante el tiempo de retardo, el cursor pasa a ser un reloj de arena, lo cual indica al usuario que debe esperar para que se complete la acción; de esta manera el usuario no cree que la aplicación se ha bloqueado y no la abandona.

5.4. METODOLOGÍA DE DISEÑO Y DESARROLLO DE APLICACIONES EDUCATIVAS MULTIMEDIA

La implementación de **MICRO** se llevó a cabo tomando en cuenta una metodología de diseño y desarrollo. Ésta fue la *metodología de diseño y construcción de aplicaciones multimedia para la educación* desarrollada en el Laboratorio de Inteligencia Artificial y Multimedia (LIAMMI) de la Universidad Nacional Autónoma de México (UNAM). Esta metodología consta de los siguientes pasos:

- I. Análisis y estructuración del contenido de la aplicación
- II. Generación, creación y recopilación de la información
- III. Diseño y programación de la aplicación
- IV. Digitalización y edición de la información
- V. Integración y validación de la aplicación
- VI. Producción

5.4.1. ANÁLISIS Y ESTRUCTURACIÓN DEL CONTENIDO DE LA APLICACIÓN

Para que el contenido informativo de una aplicación multimedia para la educación sea de máxima calidad, es necesario trabajar en colaboración con un experto en el tema sobre el que versará la aplicación. Independientemente de las investigaciones bibliográficas que se realicen, el sistema basará su funcionamiento, en cuanto a contenido teórico, en las recomendaciones y experiencia del experto. En esta etapa se deben definir los medios de despliegue y reproducción de información (audio, video, animaciones, imágenes fijas y texto) factibles de ser usados en la aplicación así como la configuración de hardware y software con que deberán contar los usuarios potenciales para poder correr la aplicación en óptimas condiciones. Esto permitirá al desarrollador o diseñador de la aplicación y al experto en el tema proponer el uso de uno u otro medio de despliegue y reproducción para presentar información específica.

Dado que **MICRO** está orientado a estudiantes comunes de ciertas ingenierías como electrónica, sistemas y computación, los cuales usualmente no cuentan en sus computadoras con tarjetas de audio y video, se definieron tres tipos de datos a utilizar: gráficos, texto, animaciones y audio. Se especificaron las características mínimas de hardware y software para el usuario, mencionadas anteriormente en el punto 5.2. de este capítulo.

En esta etapa, el desarrollador y el experto deben diseñar en conjunto un diagrama que defina las características generales y de funcionamiento del sistema.

Por el caso de **MICRO**, se buscó la asesoría por parte de un experto en el tema de los microprocesadores, el Ing. Eduardo Ramírez Sánchez. Con él se definieron los alcances académicos, teóricos e informativos del sistema, llegándose a la conclusión de que debían exponerse los siguientes temas:

1. El microprocesador
2. Memoria
3. Periféricos
4. Interrupciones
5. Aplicaciones

Se trazó el diagrama mostrado en el capítulo 4 y que aquí se reproduce para mayor ilustración.

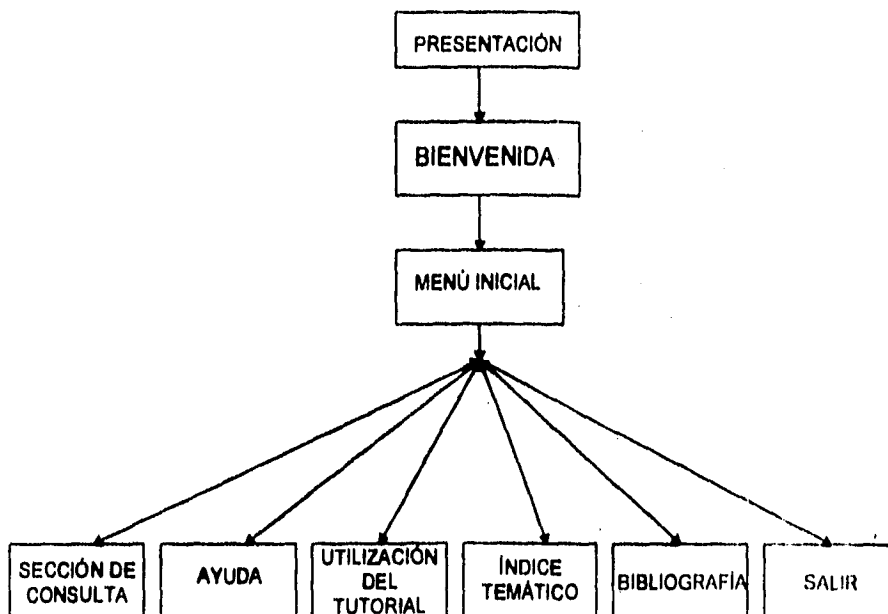


Figura 5.1. Organización jerárquica nodal del sistema MICRO

Después de la pantalla de presentación de la aplicación, aparece una pantalla de bienvenida que pregunta al usuario su nombre. Luego aparece un *Menú Inicial*, cuya función es familiarizar al usuario con el uso de MICRO. De ahí se puede pasar a la *Sección de Consulta*, donde se podrán estudiar uno a uno los temas que constituyen el acervo informativo del tutorial.

5.4.2. GENERACIÓN, CREACIÓN Y RECOPIACIÓN DE LA INFORMACIÓN

En esta etapa el trabajo del asesor académico o experto es fundamental, pues deberá estructurar la información con la secuencia, formatos y características predefinidas en la primera etapa, con base en el diagrama diseñado (Figura 5.1.), también llamado *gráfica de escritorio*. La información así estructurada debe responder a los objetivos académicos, así como a las expectativas de desarrollo de la aplicación. En esta etapa es conveniente contar con la colaboración de personal de disciplinas como la pedagogía y el diseño gráfico que puedan asesorar al desarrollador o diseñador de la aplicación en cuanto a las características de presentación, manejo de colores, diseño de pantallas, despliegue didáctico de la información y demás características indispensables en una aplicación educativa para computadora de calidad. Se consulta y extrae información de diferentes fuentes, es decir, se lleva a cabo la investigación bibliográfica y, junto con la información proporcionada por el experto, se genera un primer guión que será la base teórica del sistema. Después de este primer guión el desarrollador y el experto junto con los asesores en pedagogía y diseño gráfico deben proponer qué partes del mismo pueden ser representadas utilizando audio, video, animaciones, imagen fija o texto. Con estas propuestas y el guión inicial se diseñan nuevos guiones y scripts que serán utilizados en la etapa de integración para vaciar la información en el sistema.

En la elaboración de MICRO, en base a la estructura propuesta por el Ing. Eduardo Ramírez, se llevó a cabo la investigación bibliográfica correspondiente. Se elaboró un guión con los resultados de esta investigación y se presentó al Ingeniero Ramírez y otros asesores académicos para su revisión. Estas personas propusieron otros temas, lo que desencadenó en una nueva investigación bibliográfica, y un segundo guión. Con la información recopilada se elaboró el capítulo 3 de este trabajo. Esta información es la base teórica del sistema.

Con el segundo guión, en colaboración de asesores como la Lic. en pedagogía Fernanda Pimentel, el Ing. Pedro Guerrero, el Ingeniero industrial y Maestro en Artes Gráficas Felipe Lara García, se hicieron las proposiciones de uso de imágenes, animaciones para sustituir texto en la presentación de distintos temas. Se

tomaron en cuenta también las proposiciones de estos asesores para el diseño de la interfaz de usuario de MICRO. Las propuestas mencionadas ayudaron a conformar el tercer guión.

5.4.3. DISEÑO Y PROGRAMACIÓN DE LA APLICACIÓN

Paralelamente al desarrollo de la segunda etapa, el diseñador deberá implementar en la computadora la estructura nodal de la aplicación, basándose en la técnica "fast-prototyping" (esta técnica precisa realizar un prototipo rápido e "esqueleto" de la navegación completa del sistema sin contener información final), de forma tal que le permita ejemplificar de manera operativa las restricciones de la aplicación final. En el desarrollo de esta etapa es importante que el diseñador elabore un "demo" operativo básico del sistema. Las jerarquías en el diseño de la aplicación pueden ser:

- | | |
|----------------------|---|
| <i>Primer nivel</i> | 1. Agrupaciones jerárquicas
2. Nodos de interactividad jerárquicos |
| <i>Segundo nivel</i> | 3. Agrupaciones particulares jerárquicas
4. Nodos de interactividad inter-particulares |
| <i>Nivel "n"</i> | n. Agrupaciones jerárquicas
n+1. Nodos de interactividad jerárquicos |

Una vez teniendo la estructura nodal del sistema, se genera la estructura básica de despliegue de pantallas, lo cual, dependiendo del nivel y el tipo de información de despliegue, requerirá del auxilio de pedagogos y diseñadores gráficos para la distribución de la pantalla.

El "esqueleto" del sistema o diagrama de navegación fue expuesto en el capítulo 4 y aquí se reproduce en la figura 5.2. para ilustración.

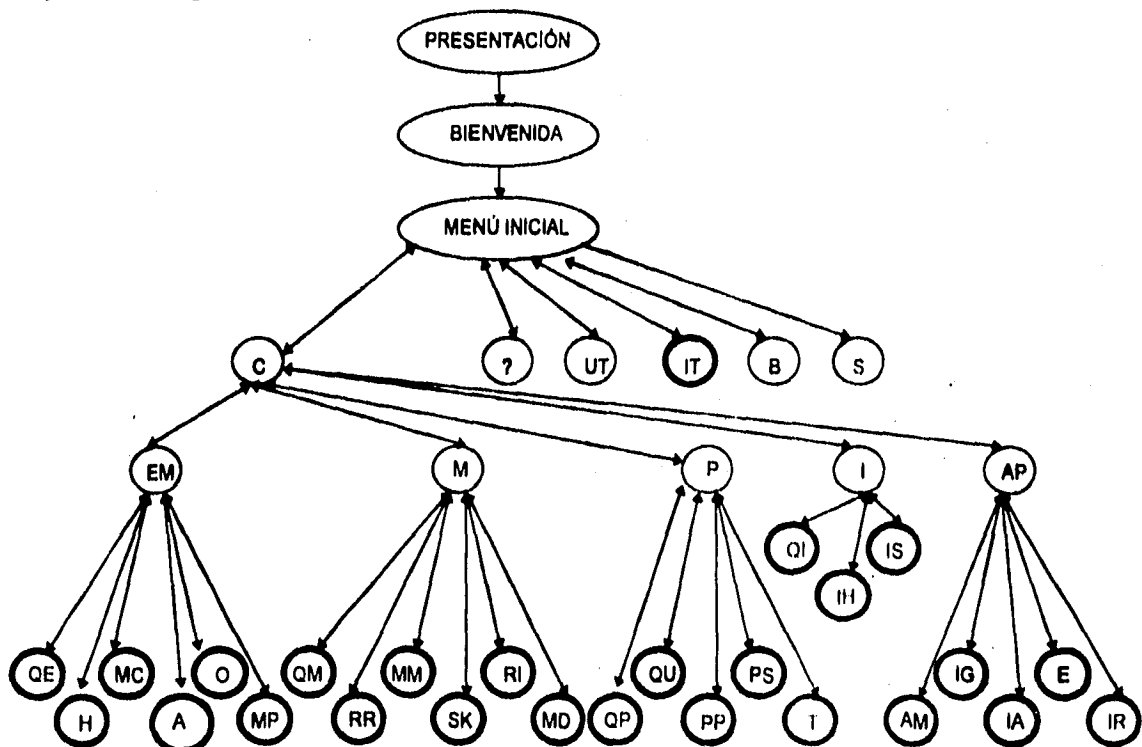


Figura 5.2. Diagrama de navegación del sistema

tomaron en cuenta también las proposiciones de estos asesores para el diseño de la interfaz de usuario de MICRO. Las propuestas mencionadas ayudaron a conformar el tercer guión.

5.4.3. DISEÑO Y PROGRAMACIÓN DE LA APLICACIÓN

Paralelamente al desarrollo de la segunda etapa, el diseñador deberá implementar en la computadora la estructura nodal de la aplicación, basándose en la técnica "fast-prototyping" (esta técnica precisa realizar un prototipo rápido e "esqueleto" de la navegación completa del sistema sin contener información final), de forma tal que le permita ejemplificar de manera operativa las restricciones de la aplicación final. En el desarrollo de esta etapa es importante que el diseñador elabore un "demo" operativo básico del sistema. Las jerarquías en el diseño de la aplicación pueden ser:

- | | |
|----------------------|---|
| <i>Primer nivel</i> | 1. Agrupaciones jerárquicas
2. Nodos de interactividad jerárquicos |
| <i>Segundo nivel</i> | 3. Agrupaciones particulares jerárquicas
4. Nodos de interactividad inter-particulares |
| <i>Nivel "n"</i> | n. Agrupaciones jerárquicas
n+1. Nodos de interactividad jerárquicos |

Una vez teniendo la estructura nodal del sistema, se genera la estructura básica de despliegue de pantallas, lo cual, dependiendo del nivel y el tipo de información de despliegue, requerirá del auxilio de pedagogos y diseñadores gráficos para la distribución de la pantalla.

El "esqueleto" del sistema o diagrama de navegación fue expuesto en el capítulo 4 y aquí se reproduce en la figura 5.2. para ilustración.

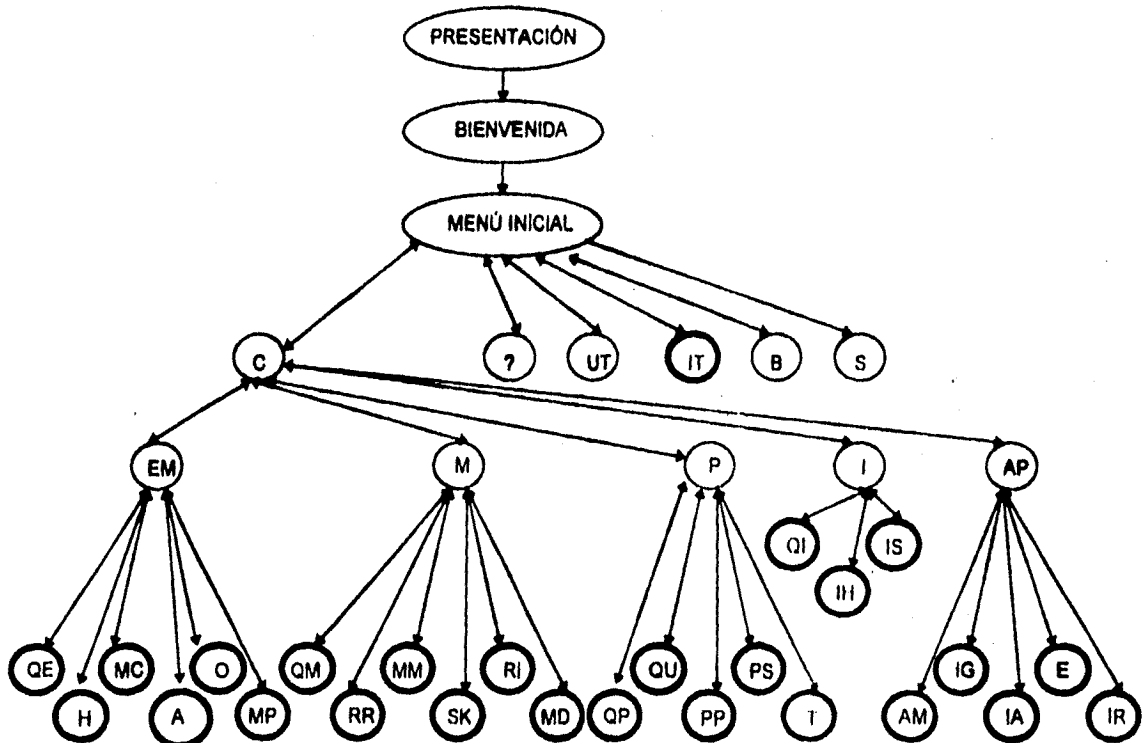


Figura 5.2. Diagrama de navegación del sistema

En el diagrama de la figura 5.2. hay círculos remarcados; esto significa que el *Índice Temático* tiene conexión con todos los temas expuestos en el tutorial y éstos a su vez con el *Índice Temático*. Otra parte importante de esta etapa es el diseño de la interfaz de usuario. En el punto 5.3. de este capítulo se detallan las características con que fue elaborada la interfaz de usuario de **MICRO**. Como información adicional, diremos que dado que uno de los elementos principales de cualquier programa, en especial tutoriales para computadora, es la interfaz de usuario pues de ésta depende que tan fácil le será al usuario interactuar con el programa, las aplicaciones multimedia deben presentar interfaces intuitivas, amigables y por lo tanto fáciles de usar pero no por todo esto ineficientes o faltas de elegancia. Lo anterior debe cumplirse en especial para las aplicaciones educativas multimedia. Todo esto hace que en esta etapa el diseñador de la aplicación o desarrollador deba hacer especial énfasis en el diseño de la interfaz de usuario del sistema.

Existen diversos estándares para el diseño de interlaces de usuario; en la actualidad los más populares son las interfaces gráficas (*GUI, Graphical User Interface*) que se pueden encontrar en todas las plataformas de cómputo (*Windows* en la IBM-PC o compatible, *Finder* en equipos Macintosh, *X-Windows* en estaciones de trabajo Unix). Este tipo de interfaces gráficas resultan ideales para el desarrollo de aplicaciones multimedia ya que son muy intuitivas y los usuarios finales no necesitan tener mucha experiencia en el uso de computadoras para interactuar con ellas a un nivel básico.

La interfaz de usuario de la aplicación debe cumplir los estándares de la GUI sobre la que trabajará (por ejemplo, si se trata de *Windows* para IBM-PC o compatible, la aplicación debe verse y trabajar de forma parecida a como lo hace *Windows*, lo que similarmente debe cumplirse para cualquier otra GUI) para evitar que el usuario sufra confusiones al interactuar con ella.

Dado que **MICRO** es un sistema tutorial diseñado para trabajar sobre *Windows*, se decidió utilizar la GUI *Windows* para IBM-PC's y compatibles. De esta manera, la interfaz de usuarios de **MICRO** presenta botones gráficos de color gris, a la manera de *Windows*. En la figura 5.3. se presenta la interfaz de usuario de **MICRO**.

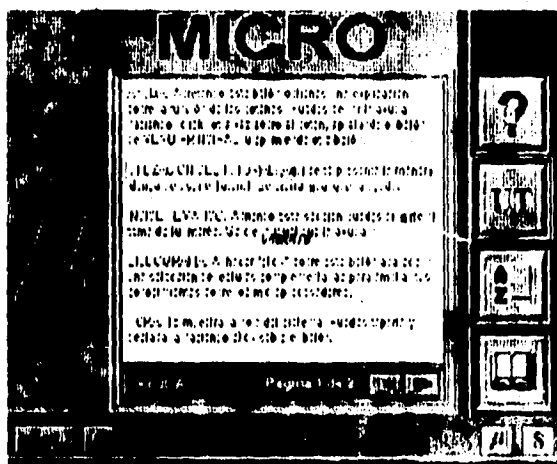


Figura 5.3. Interfaz de usuario de MICRO

El sistema se desarrolló basado en los diagramas de la figura 5.1. en lo que a jerarquía de nodos se refiere y la figura 5.2. en lo que refiere a navegación del sistema.

La herramienta de programación de aplicaciones multimedia conocida como *Authoring Software* (software de autoraje) utilizada para realizar el sistema tutorial fue *Authorware Professional*.

5.4.4. DIGITALIZACIÓN Y EDICIÓN DE LA INFORMACIÓN

Esta etapa consiste en transformar la información a formatos digitales, transcribir textos en procesadores de palabra, diseñar las pantallas principales del sistema, digitalizar y editar imágenes, voz música video, desarrollar animaciones, etc.

Esta etapa se concreta a realizar dos actividades: creación, y, principalmente, edición de los datos a integrar en el sistema. Teniendo los datos en formato digital, deberán ser revisados para hacer las modificaciones o ajustes necesarios, con el objetivo de poder integrar los datos en el sistema.

En el caso de MICRO, los textos fueron capturados en *Word for Windows 2.0* y *6.0*. Posteriormente fueron editados en *Authorware*. Los gráficos fueron realizados en *Microsoft Draw*, *Corel Draw*, *Corel Photo Paint*, *Paintbrush* y algunos fueron capturados en *Paint Shop Pro*. Algunos gráficos fueron realizados en *Authorware*. Las animaciones del sistema se realizaron en *Authorware* y *3D Studio*.

5.4.5. INTEGRACIÓN Y VALIDACIÓN DE LA APLICACIÓN

En la misma estructura nodal diseñada en la primera y especialmente tercera etapa, se integra la información en sus respectivos nodos y pantallas, haciendo pruebas al sistema de: funcionamiento, operatividad, interfaz de usuario, interfaces con dispositivos, flexibilidad de información, calidad del contenido, etc.

En lo que refiere a la validación, es recomendable que el sistema sea evaluado por personal relacionado por el área académica en la que se encuentra la aplicación, así como también realizar pruebas controladas con usuarios potenciales.

En la elaboración de MICRO, se integraron los datos como texto, imagen fija (gráficos) y animaciones en las pantallas y nodos correspondientes.

Además, se integró en MICRO una herramienta llamada *Black de Notas*, donde el usuario puede apuntar recordatorios mientras está trabajando con el sistema.

Para la validación del sistema, éste se sometió a evaluación con alumnos de ingeniería de la Universidad Nacional Autónoma de México y del Instituto Tecnológico de San Luis Potosí. Los usuarios que evaluaron presentaron reportes de sus observaciones y comentarios en base a los cuales se realizaron algunas modificaciones en el sistema.

5.4.6. PRODUCCIÓN

Habiendo realizado la validación del sistema, se estará en condiciones de producir el master o prototipo de producción. Éste puede ser realizado ya sea en CD-ROM, disquetes, VCR, etc.

MICRO fue hecho ejecutable con *Authorware Professional*, y se elaboró un programa de instalación con *Visual Basic*. De esta manera, el sistema quedó listo en disquetes de 3 ½ pulgadas para su distribución al usuario final.

5.5. FUNCIONAMIENTO DEL TUTORIAL

El programa fuente del sistema tutorial MICRO consta de 5 archivos:

- micro.apw
- micprin.apw
- miccons.apw
- micsalir.apw
- miclibl.apl

El archivo *micro.apw* es el programa fuente de la entrada del tutorial. El archivo *micprin.apw* es el programa fuente del menú inicial. El archivo *miccons.apw* es el programa fuente de la sección de consulta. El archivo *micsalir.apw* es el programa fuente de la salida del tutorial y, por último, el archivo *miclibl.apl*, es una librería fuente, es decir, modificable en programación, donde se encuentran algunos gráficos y animaciones del tutorial que ocuparían de otro modo mucha memoria.

Estos archivos, fueron transformados en un paquete ejecutable mediante la opción "Package" de *Authorware Professional*, quedando los siguientes archivos:

- micro.exe
- micprin.app
- miccons.app
- micsalir.app
- miclib.apr

Donde el archivo *micro.exe* es el ejecutable con el que se puede dar inicio a la operación del tutorial sobre el ambiente *Windows*; después de la instalación del tutorial, éstos son los archivos que quedan guardados en el disco duro. Posteriormente, se usó un programa en *Visual Basic* para comprimir estos archivos y tener un programa de instalación. Los disquetes de instalación de **MICRO** contienen los siguientes programas:

- micro.ex_
- micprin.ap_
- miccons.ap_
- micsalir.ap_
- miclibl.ap_
- setup.exe
- setup.lst
- setupl.ex_
- setupkit.dl_
- vbrun300.dl_
- ver.dl_

Donde *micro.ex_*, *micprin.ap_*, *miccons.ap_*, *micsalir.ap_* y *miclibl.ap_* son los archivos constituyentes de **MICRO**, pero comprimidos. El ejecutable *setup.exe* lleva a cabo la instalación del tutorial (descomprimiendo los archivos pertinentes) y los demás archivos son parte del programa de instalación. Para instalar **MICRO**, se ejecuta *setup.exe* (en el disco 1 de instalación), desde el *Administrador de Programas* o desde el *Administrador de Archivos de Windows*. El programa de instalación pide por su cuenta el disco 2 y crea el grupo de programas correspondiente y el elemento de programa **MICRO**, con el debido icono. Una vez instalado, el tutorial se podrá ejecutar haciendo doble click sobre el icono creado, o ejecutando el archivo *micro.exe*.

Capítulo 6

Conclusiones

6. CONCLUSIONES

La impartición del tópico de los microprocesadores se verá beneficiada con nuevas formas de enseñanza. Recuerdo cuando llevé la asignatura de microprocesadores: durante la primera parte del curso el profesor estuvo dibujando cuadros y rectángulos en el pizarrón, que representaban los diferentes registros y partes del microprocesador. En todo ese tiempo entendí muy poco de lo que se me trataba de enseñar, pues debía incorporar muchos conceptos nuevos y esto era difícil. Fue durante la segunda parte del curso, cuando tuvimos que desarrollar una aplicación práctica con microprocesador, que entendí el funcionamiento de los llamados "micros".

La inclusión de un tutorial en la enseñanza de microprocesadores será de gran ayuda, pues mediante el uso de la computadora el estudiante irá entrando en el conocimiento de los diferentes tópicos que tienen que ver con microprocesadores, de una manera pausada, didáctica y gráfica. En la impartición del curso de microprocesadores, será de gran utilidad, además de las partes teórica y práctica del curso, incorporar un recorrido por el Tutorial sobre Microprocesadores utilizando Multimedia MICRO, pues el estudiante verá de una manera sencilla, didáctica y gráfica los distintos conceptos básicos del tópico de los microprocesadores.

La emergencia de la computadora y sus múltiples aplicaciones como herramienta de trabajo han modificado las tendencias en cuanto a forma de trabajar en prácticamente todos los ámbitos. Actualmente es común interactuar con computadoras en terminales bancarias y restaurantes. Como consecuencia se ha presentado el surgimiento de diversas tecnologías que permiten la creación y utilización de herramientas y programas cada vez más poderosos, aplicables a muchos campos como: arte por computadora, telecomunicaciones, ingeniería, medicina, administración, investigación científica, entretenimiento, finanzas y educación. Por otro lado, las interfaces desarrolladas cada vez son más amables, lo que facilita el uso de la computadora a usuarios no experimentados. Una de las tecnologías que ha surgido en los últimos años es la Multimedia.

La multimedia está provocando una revolución en el procesamiento y transmisión de la información, dada la versatilidad que sus herramientas confieren a los desarrolladores de aplicaciones. Esta revolución ha alcanzado el ámbito de los procesos de enseñanza-aprendizaje, pues el uso de la multimedia en estos procesos permite a los estudiantes aprender a su propio ritmo, a la vez que estimula sus sentidos haciéndolos más receptivos a nuevas ideas y conceptos.

MICRO se presenta así como uno de los primeros esfuerzos de incorporar multimedia a la enseñanza en la Facultad de Ingeniería de la UNAM, y a mi parecer, *cumple satisfactoriamente con los objetivos que persigue*, pues presenta un apoyo a los estudiantes para aprender los conceptos básicos del tópico de los microprocesadores y es una herramienta didáctica útil al profesor para facilitar la enseñanza de la asignatura.

Sería muy interesante y necesario realizar una investigación exhaustiva en base a experimentos para ver en qué porcentaje la utilización de MICRO reduce el tiempo de aprendizaje comparándolo con investigación personal o enseñanza tradicional. Esta investigación está fuera de los alcances del presente trabajo de tesis.

El desarrollo del sistema **MICRO** fue muy formativo en varias de sus facetas, como el caso de la investigación bibliográfica donde se tuvieron que efectuar múltiples lecturas y razonamientos para poder explicar de manera sencilla y didáctica los diferentes temas. Otro caso fue el diseño del tutorial, pues aprendí mucho sobre diseño de aplicaciones multimedia y diseño de interlaces de usuario. Por otro lado, en la programación e implementación se utilizaron herramientas software, como el *Authorware Professional*, paquete con que se programó el tutorial, *Visual Basic*, con el cual se realizó el programa de instalación de **MICRO** y programas de diseño gráfico por computadora como el *Painbrush*, *Paint Shop Pro*, *Microsoft Draw*, *Corel Draw*, *Corel Photo Paint*, etc.

Algo que se puede concluir es que **MICRO** ayuda a los estudiantes a aprender los conceptos básicos del tópico de microprocesadores de manera más amena y amable que con los métodos tradicionales.

Bibliografía

BIBLIOGRAFÍA

- ☐ Angulo, José Ma. (1984) *Microprocesadores (Curso sobre aplicaciones en sistemas industriales)*. Paraninfo. Madrid, España.
- ☐ Angulo, José Ma. (1989) *Microprocesadores (Fundamentos, diseño y aplicaciones en la industria y en los microcomputadores)*. Paraninfo. Madrid, España.
- ☐ Arrch C., Luther (1992) *Designing Interactive Multimedia*. Bantman Books. EUA.
- ☐ Boyce, Jefferson C. (1979) *Microprocessor and Microcomputer Basics*. Prentice Hall. New Jersey, EUA.
- ☐ Brant, Stewart (1988) *The Media Lab: Inventing the Future at M.I.T.* Penguin Books. EUA.
- ☐ Fohl, Mark E. (1978) *A Microprocessor Course*. Petrocelli Books. New York, EUA.
- ☐ Freeman, W. H. and Company (1976) *Recent Progress in Perception*. Scientific American. EUA.
- ☐ García, Hugo G. (1988) *Microprocesadores-Teoría y Práctica*. Limusa. Mexico. D. F., México.
- ☐ Gilmore, Charles M. (1989) *Principios de Microprocesadores*. Limusa. México. D. F., México.
- ☐ Givone, Donald D.; Roesser, Robert P. (1980) *Microprocessors/Microcomputers: An Introduction*. McGraw-Hill. EUA.
- ☐ Goleman, D.; Kaufman, P.; Ray, M. (1992) *The Creative Spirit*. Dutton Book. EUA.
- ☐ Hall, Douglas V. (1983) *Microprocessors and digital systems*. McGraw-Hill. New York, EUA.
- ☐ Hall, Douglas V. (1992) *Microprocessors and Interfacing*. Macmillan McGraw-Hill. EUA.
- ☐ Halsall, F.; Lister, P. F. (1984) *Fundamentos de Microprocesadores*. Paraninfo. Madrid, España.
- ☐ Hayes, John P. (1984) *Digital System Design and Microprocessors*. McGraw-Hill. New York, EUA.
- ☐ Heffer, D. E.; King, G. A.; Keith, D. C. (1986) *Basic Principles and Practice of Microprocessors*. Edward Arnold. London, RU.
- ☐ Hilburn, John L. (1976) *Microcomputers/microprocessors: hardware, software and applications*. Prentice Hall. New Jersey, EUA.
- ☐ Hill, Fredrick J.; Peterson, Gerald R. (1984) *Digital Logic and Microprocessors*. Wiley. New York, EUA.
- ☐ Jacobson, Ivar (1993) *Object Oriented Software Engineering*. Addison Wesley Publishing Company. EUA.
- ☐ Kemper, N.; Lara, F; Guerrero, P.; Monedero, A. (1994) "Sistema Tutorial Multimedia para Apoyar la Capacitación Técnica en Cogeneración" en *Memoria Técnica, XI Seminario Nacional sobre el Uso Racional de la Energía*. México, D.F., México.

- Khambata A. J. (1987) *Microprocesadores/Microcomputadores*. Gustavo Gili. México D. F., México.
- Lara, F. y Guerrero, P. (1994) "Metodología básica para el desarrollo de aplicaciones multimedia en la educación" en *Memorias del II Congreso Internacional de Informática*. Mendoza, Argentina.
- Mompin, José y otros autores (1984) *Microprocesadores y Microcomputadores*. Marcombo. México, D. F., México.
- Ramírez, Edward V. (1986) *Introducción a los Microprocesadores*. Limusa. México, D. F. México.
- Ray E., Eberts (1994) *User Interface Design*. Prentice Hall. EUA.
- Rosh, Winn L. (1992) *The Winn L. Rosh Hardware Bible*. Brady Publishing. New York, NY, EUA.
- Short, Kenneth L. (1985) *Microprocesadores y lógica programada*. Gustavo Gili. Barcelona, España.
- Tocci, Ronald J.; Laskowski, Lester P. (1979) *Microprocessors and Microcomputers (Hardware and Software)*. Prentice-Hall. New Jersey, EUA.
- Tokheim, Roger L. (1991) *Fundamentos de los Microprocesadores*. Mc Graw Hill. Madrid, España.
- Wenger, Etienne (1987) *Artificial Intelligence and Tutoring Systems*. Morgan Kaufmann. Los Altos, California, EUA.
- Wodaski, Ron (1992) *Multimedia Madness*. Sams Publishings. EUA.