

1. INTRODUCCIÓN

- **INTRODUCCIÓN**

- **Presentación del alumno y objetivos**

Este trabajo ha sido realizado por Luis Panzano Barbero, nº de matrícula 92064, alumno de quinto curso de la especialidad de Automática en la E.T.S.I.I.M. En él se pretende exponer una visión general de la arquitectura de un computador analizando sus distintos elementos.

- **Definiciones de: computador, arquitectura y organización del computador**

Se puede definir la **arquitectura de computadores** como el estudio de la estructura, funcionamiento y diseño de computadores. Esto incluye, sobre todo a aspectos de hardware, pero también afecta a cuestiones de software de bajo nivel.

Computador, dispositivo electrónico capaz de recibir un conjunto de instrucciones y ejecutarlas realizando cálculos sobre los datos numéricos, o bien compilando y correlacionando otros tipos de información.

- **Reseña histórica de los computadores**

- **La era mecánica de los computadores**

Podríamos decir que las máquinas mecánicas de calcular constituyendo la "era arcaica" o generación 0 de los computadores. Una evolución de estas máquinas son las máquinas registradoras mecánicas que aún existen en la actualidad. Otro elemento de cálculo mecánico que se utilizó hasta hace pocos años fue la regla de cálculo que se basa en el cálculo logaritmo y cuyo origen son los círculos de proporción de Neper. Ingenios clásicos de esa etapa fueran la máquina de Pascal, que podía realizar sumar, restas y, posteriormente, multiplicaciones y divisiones, y las dos máquinas de Charles Babbage: la máquina de diferencias y la **analítica**. Esta última fue la precursora de los computadores actuales.

La fase final de la en la mecánica de la informática y la constituyen los computadores electromecánicos basados en lógica de relés (década de los 30).

- **La era electrónica de los computadores**

Los computadores envasados en elementos mecánicos planteaban ciertos problemas:

La velocidad de trabajo está limitada **a inercia de la partes móviles**.

La transmisión de la información por medios mecánicos (engranajes, palancas, etcétera.) es poco fiable y difícilmente manejable.

Los computadores electrónicos salvan estos inconvenientes ya que carecen de partes móviles y la velocidad de transmisión de la información por métodos eléctricos no es comparable a la de ningún elemento mecánico.

El primer elemento electrónico usado para calcular fue la válvula de vacío y, probablemente, el primer computadores electrónicos de uso general fue el E.N.I.A.C. (Electronic Numerical Integrator Calculator) construido en Universidad de Pennsylvania (1943–46). El primer computador de programa almacenado fue el E.D.V.A.C. (Electronic Discrete Variable Computer, 1945–51) basado en la idea de **John Von Neumann**, que también participó en el proyecto E.N.I.A.C. de que el programa debe almacenarse en la misma memoria que los datos.

· Generaciones de ordenadores

En la evolución de las máquinas para el tratamiento automático de la información pueden distinguirse una aceria que y tos que marcan la diferencia entre las denominadas **generaciones de ordenadores**. Las generaciones habidas hasta la actualidad han sido:

1ª generación: (1946–1955) Computadores basados en válvula de vacío que se programaron en lenguaje máquina o en lenguaje ensamblados.

2ª generación: (1953–1964) Computadores de transistores. Evolucionan los modos de direccionamiento y surgen los lenguajes de alto nivel.

3ª generación: (1964–1974) Computadores basados en circuitos integrados y con la posibilidad de trabajar en tiempo compartido.

4ª generación: (1974–) Computadores Que integran toda la CPU en un solo circuito integrado (microprocesadores). Comienzan a proliferar las redes de computadores.

• **LA UNIDAD CENTRAL DE PROCESO**

• Funciones que realiza

La Unidad central de proceso o CPU, se puede definir como un circuito microscópico que interpreta y ejecuta instrucciones. La CPU se ocupa del control y el proceso de datos en los ordenadores. Habitualmente, la CPU es un microprocesador fabricado en un chip, un único trozo de silicio que contiene millones de componentes electrónicos. El microprocesador de la CPU está formado por una unidad aritmético–lógica que realiza cálculos y comparaciones, y toma decisiones lógicas (determina si una afirmación es cierta o falsa mediante las reglas del álgebra de Boole); por una serie de registros donde se almacena información temporalmente, y por una unidad de control que interpreta y ejecuta las instrucciones. Para aceptar órdenes del usuario, acceder a los datos y presentar los resultados, la CPU se comunica a través de un conjunto de circuitos o conexiones llamado bus. El bus conecta la CPU a los dispositivos de almacenamiento (por ejemplo, un disco duro), los dispositivos de entrada (por ejemplo, un teclado o un mouse) y los dispositivos de salida (por ejemplo, un monitor o una impresora).

• Elementos que la componen

- ◆ *Unidad de control:* controla el funcionamiento de la CPU y por tanto de el computador.
- ◆ *Unidad aritmético–lógica (ALU):* encargada de llevar a cabo las funciones de procesamiento de datos del computador.
- ◆ *Registros:* proporcionan almacenamiento interno a la CPU.
- ◆ *Interconexiones CPU:* Son mecanismos que proporcionan comunicación entre la unidad de control, la ALU y los registros.
- ◆ **Tipos**

Básicamente nos encontramos con dos tipos de diseño de los microprocesadores: RISC (Reduced–Instruction–Set Computing) y CISC (complex–instruction–set computing). Los **microprocesadores RISC** se basan en la idea de que la mayoría de las instrucciones para

realizar procesos en el computador son relativamente simples por lo que se minimiza el número de instrucciones y su complejidad a la hora de diseñar la CPU. Algunos ejemplos de arquitectura RISC son el SPARC de Sun Microsystems, el microprocesador Alpha diseñado por la antigua Digital, hoy absorbida por Compaq y los Motorola 88000 y PowerPC. Estos procesadores se suelen emplear en aplicaciones industriales y profesionales por su gran rendimiento y fiabilidad.

Los **microprocesadores CISC**, al contrario, tienen una gran cantidad de instrucciones y por tanto son muy rápidos procesando código complejo. Las CPU's CISC más extendidas son las de la familia 80x86 de Intel cuyo último micro es el Pentium II. Últimamente han aparecido otras compañías como Cirix y AMD que fabrican procesadores con el juego de instrucciones 80x86 y a un precio sensiblemente inferior al de los microprocesadores de Intel. Además, tanto Intel con MMX como AMD con su especificación 3D-Now! están apostando por extender el conjunto de instrucciones de la CPU para que trabaje más eficientemente con tratamiento de imágenes y aplicaciones en 3 dimensiones.

◆ **LA MEMORIA**

◆ **Funciones que realiza**

La memoria de un computador se puede definir como los circuitos que permiten almacenar y recuperar la información. En un sentido más amplio, puede referirse también a sistemas externos de almacenamiento, como las unidades de disco o de cinta.

Hoy en día se requiere cada vez más memoria para poder utilizar complejos programas y para gestionar complejas redes de computadores.

◆ **Elementos que la componen**

Una memoria, vista desde el exterior, tiene la estructura mostrada en la figura 3-1. Para efectuar una lectura se deposita en el bus de direcciones la dirección de la palabra de memoria que se desea leer y entonces se activa la señal de lectura (R); después de cierto tiempo (tiempo de latencia de la memoria), en el bus de datos aparecerá el contenido de la dirección buscada. Por otra parte, para realizar una escritura se deposita en el bus de datos la información que se desea escribir y en el bus de direcciones la dirección donde deseamos escribirla, entonces se activa la señal de escritura (W), pasado el tiempo de latencia, la memoria escribirá la información en la dirección deseada. Internamente la memoria tiene un registro de dirección (MAR, memory address register), un registro buffer de memoria o registro de datos (MB, memory buffer, o MDR, memory data register) y, un decodificador como se ve en la figura 3-2. Esta forma de estructurar la memoria se llama organización lineal o de una dimensión. En la figura cada línea de palabra activa todas las células de memoria que corresponden a la misma palabra.

Por otra parte, en una memoria ROM programable por el usuario con organización lineal, las uniones de los diodos correspondientes a lugares donde deba haber un "0" deben destruirse. También se pueden sustituir los diodos por transistores y entonces la célula de memoria tiene el esquema de la figura 3-3. en este caso la unión que debe destruirse para grabar un "0" es la del emisor.

En el caso de una memoria RAM estática con organización lineal cada célula de memoria toma la forma mostrada en la figura 3-4. En este esquema las primeras puertas AND sólo son necesarias en el una de las células de cada palabra. Se debe comentar la necesidad de la puerta de tres estados a la salida del biestable: esta puerta se pone para evitar que se unan las salidas

de los circuitos de las células de diferentes palabras a través del hilo de bit. Si esa puerta no se pusiera (o hubiera otro tipo de puerta en su lugar, como una puerta AND) la información correspondiente a la palabra activa entraría por los circuitos de salida de las demás células, lo que los dañaría.

Organizar la memoria de esta forma, tiene el inconveniente de que la complejidad del decodificador crece exponencialmente con el número de entradas y, en una memoria de mucha capacidad, la complejidad del decodificador la hace inviable. Esto hace necesaria una alternativa que simplifique los decodificadores. Esta alternativa la constituye la organización en dos dimensiones en que los bits del registro de dirección se dividen en dos partes y cada una de ellas va a un decodificador diferente. En este caso, las líneas procedentes de ambos decodificadores (X e Y) se cruzan formando un sistema de coordenadas en que cada punto de cruce corresponde a una palabra de memoria. Dado que en cada decodificador sólo se activa una línea, sólo se activará la palabra correspondiente al punto de cruce de las dos líneas activadas. Fácilmente se puede comprender que los decodificadores se simplifican mucho ya que cada uno tiene la mitad de entradas que en el caso anterior. Hay que decir, sin embargo, que la célula de memoria se complica un poco porque hay que añadir una puerta AND en cada palabra para determinar si coinciden las líneas X e Y.

La organización de la memoria en dos dimensiones también es útil para las memorias dinámicas ya que el refresco de estas memorias se realiza por bloques y éstos pueden coincidir con una de las dimensiones (la que corresponda a los bits de dirección de mayor peso).

En la práctica, las memorias dinámicas son más lentas que las estáticas y además son de lectura destructiva, pero resultan más baratas, aunque necesitan circuitos de refresco, si la memoria no es de mucha capacidad.

◆ **Tipos**

• **Jerarquía de memoria**

En un ordenador hay una jerarquía de memorias atendiendo al tiempo de acceso y a la capacidad que. normalmente son factores contrapuestos por razones económicas y en muchos casos también físicas. Comenzando desde el procesador al exterior, es decir en orden creciente de tiempo de acceso y capacidad, se puede establecer la siguiente jerarquía:

- ◇ **Registros de procesador:** Estos registros interaccionan continuamente con la CPU (porque forman parte de ella). Los registros tienen un tiempo de acceso muy pequeño y una capacidad mínima, normalmente igual a la palabra del procesador (1 a 8 bytes).
- ◇ **Registros intermedios:** Constituyen un paso intermedio entre el procesador y la memoria, tienen un tiempo de acceso muy breve y muy poca capacidad.
- ◇ **Memorias caché:** Son memorias de pequeña capacidad. Normalmente una pequeña fracción de la memoria principal. y pequeño tiempo de acceso. Este nivel de memoria se coloca entre la CPU y la memoria central. Hace algunos años este nivel era exclusivo de los ordenadores grandes pero actualmente todos los ordenadores lo incorporan. Dentro de la memoria caché puede haber, a su vez, dos niveles denominados *caché on chip*, memoria caché dentro del circuito integrado, y *caché on board*, memoria caché en la placa de circuito impreso pero fuera del circuito integrado, evidentemente, por razones físicas, la primera es mucho más rápida que la segunda. Existe también una técnica, denominada Arquitectura Harvard, en cierto modo contrapuesta a la idea de Von Neumann, que utiliza memorias caché separadas para código y datos. Esto tiene algunas ventajas como se verá en este capítulo.
- ◇ **Memoria central o principal:** En este nivel residen los programas y los datos. La

CPU lee y escribe datos en él aunque con menos frecuencia que en los niveles anteriores. Tiene un tiempo de acceso relativamente rápido y gran capacidad.

- ◊ **Extensiones de memoria central:** Son memorias de la misma naturaleza que la memoria central que amplían su capacidad de forma modular. El tiempo de similar, a lo sumo un poco mayor, al de la memoria central y su capacidad puede ser algunas veces mayor.
- ◊ **Memorias de masas o auxiliares:** Son memorias que residen en dispositivos externos al ordenador, en ellas se archivan programas y datos para su uso posterior. También se usan estas memorias para apoyo de la memoria central en caso de que ésta sea insuficiente (memoria virtual). Estas memorias suelen tener gran capacidad pero pueden llegar a tener un tiempo de acceso muy lento. Dentro de ellas también se pueden establecer varios niveles de jerarquía.

- Clasificación de memorias semiconductoras de acceso aleatorio

Las memorias se clasifican, por la tecnología empleada y, además según la forma en que se puede modificar su contenido, A este respecto, las memorias se clasifican en dos grandes grupos:

1) Memorias RAM: Son memorias en las que se puede leer y escribir, si bien su nombre (Random access memory) no representa correctamente este hecho. Por su tecnología pueden ser de ferritas (ya en desuso) o electrónicas, Dentro de éstas últimas hay memorias estáticas (SRAM, static RAM), cuya célula de memoria está basada en un biestable, y **memorias dinámicas (DRAM, dinamic RAM**, en las que la célula de memoria es un pequeño condensador cuya carga representa la información almacenada. Las memorias dinámicas necesitan circuitos adicionales de refresco ya que los condensadores tienen muy poca capacidad y, a través de las fugas, la información puede perderse, por otra parte, son de lectura destructiva.

2) Memorias ROM (Read Only Memory): Son memorias en las que sólo se puede leer. Pueden ser:

- ◆ **ROM programadas por máscara**, cuya información se graba en fábrica y no se puede modificar.
- ◆ **PROM**, o ROM programable una sola vez.
- ◆ **EPROM** (erasable PROM) o **RPROM** (reprogrammable ROM), cuyo contenido puede borrarse mediante rayos ultravioletas para regrabarlas.
- ◆ **EAROM** (electrically alterable ROM) o **EEROM** (electrically erasable ROM), que son memorias que está en la frontera entre las RAM y las ROM ya que su contenido puede regrabarse por medios eléctricos, estas se diferencian de las RAM en que no son volátiles. En ocasiones a este tipo de memorias también se las denomina NYRAM (no volátil RAM).
- ◆ **Memoria FLASH**, denominada así por la velocidad con la que puede reprogramarse, utilizan tecnología de borrado eléctrico al igual que las EEPROM. Las memorias flash pueden borrarse enteras en unos cuantos segundos, mucho más rápido que las EPROM.

Básicamente las memorias ROM se basan en una matriz de diodos cuya unión se puede destruir aplicando sobre ella una sobretensión (usualmente comprendida ente -12.5 y -40 v.). De fábrica la memoria sale con 1's en todas sus posiciones, para grabarla se rompen las uniones en que se quieran poner 0's. Esta forma de realizar la grabación se denomina **técnica de los fusibles**.

- ◆ **BUSES DEL SISTEMA**
- ◆ **Funciones que realiza**

El bus se puede definir como un conjunto de líneas conductoras de *hardware* utilizadas para la transmisión de datos entre los componentes de un sistema informático. Un bus es en esencia una ruta compartida que conecta diferentes partes del sistema, como el microprocesador, la controladora de unidad de disco, la memoria y los puertos de entrada/salida (E/S), para permitir la transmisión de información.

En el bus se encuentran dos pistas separadas, el bus de datos y el bus de direcciones. La CPU escribe la dirección de la posición deseada de la memoria en el bus de direcciones accediendo a la memoria, teniendo cada una de las líneas carácter binario. Es decir solo pueden representar 0 o 1 y de esta manera forman conjuntamente el número de la posición dentro de la memoria (es decir: la dirección). Cuanto mas líneas haya disponibles, mayor es la dirección máxima y mayor es la memoria a la cual puede dirigirse de esta forma. En el bus de direcciones original habían ya 20 direcciones, ya que con 20 bits se puede dirigir a una memoria de 1 MB y esto era exactamente lo que correspondía a la CPU.

Esto que en la teoría parece tan fácil es bastante mas complicado en la práctica, ya que aparte de los bus de datos y de direcciones existen también casi dos docenas más de líneas de señal en la comunicación entre la CPU y la memoria, a las cuales también se acude. Todas las tarjetas del bus escuchan, y se tendrá que encontrar en primer lugar una tarjeta que mediante el envío de una señal adecuada indique a la CPU que es responsable de la dirección que se ha introducido. Las demás tarjetas se desprecupan del resto de la comunicación y quedan a la espera del próximo ciclo de transporte de datos que quizás les incumba a ellas.

PROCESADOR	Bus de direcciones (bits)	Bus de datos (bits)
8086	20	16
8088	20	8
80186	20	16
80188	20	8
80286	24	16
80386 SX	32	16
80386 DX	32	32
80486 DX	32	32
80486 SX	32	32
PENTIUM	32	64
PENTIUM PRO	32	64

Este mismo concepto es también la razón por la cual al utilizar tarjetas de ampliación en un PC surgen problemas una y otra vez, si hay dos tarjetas que reclaman para ellas el mismo campo de dirección o campos de dirección que se solapan entre ellos.

Los datos en si no se mandan al bus de direcciones sino al bus de datos. El bus XT tenía solo 8 bits con lo cual sólo podía transportar 1 byte a la vez. Si la CPU quería depositar el contenido de un registro de 16 bits o por valor de 16 bits, tenía que desdoblarlos en dos bytes y efectuar la transferencia de datos uno detrás de otro.

De todas maneras para los fabricantes de tarjetas de ampliación, cuyos productos deben atenderse a este protocolo, es de una importancia básica la regulación del tiempo de las señales del bus, para poder trabajar de forma inmejorable con el PC. Pero precisamente este protocolo no ha sido nunca publicado por IBM con lo que se obliga a los fabricantes a medir las señales con la ayuda de tarjetas ya existentes e imitarlas. Por lo tanto no es de extrañar que

se pusieran en juego tolerancias que dejaron algunas tarjetas totalmente eliminadas.

♦ **Estructuras de interconexión**

Existen dos organizaciones físicas de operaciones E/S que tienen que ver con los buses que son:

Bus único

Bus dedicado

La primera gran diferencia entre estas dos tipos de estructuras es que el bus único no permite un controlador DMA (todo se controla desde la CPU), mientras que el bus dedicado sí que soporta este controlador.

El bus dedicado trata a la memoria de manera distinta que a los periféricos (utiliza un bus especial) al contrario que el bus único que los considera a ambos como posiciones de memoria (incluso equipara las operaciones E/S con las de lectura/escritura en memoria). Este bus especial que utiliza el bus dedicado tiene 4 componentes fundamentales:

- ◊ **Datos:** Intercambio de información entre la CPU y los periféricos.
- ◊ **Control:** Lleva información referente al estado de los periféricos (petición de interrupciones).
- ◊ **Direcciones:** Identifica el periférico referido.
- ◊ **Sincronización:** Temporiza las señales de reloj.

La mayor ventaja del bus único es su simplicidad de estructura que le hace ser más económico, pero no permite que se realice a la vez transferencia de información entre la memoria y el procesador y entre los periféricos y el procesador.

Por otro lado el bus dedicado es mucho más flexible y permite transferencias simultáneas. Por contra su estructura es más compleja y por tanto sus costes son mayores.

♦ **Tipos**

Ahora vamos a ver los distintos tipos de buses que se han ido desarrollando y los que se emplean en la actualidad.

• **EL BUS XT y EL BUS ISA (AT)**

Cuando en 1980 IBM fabricó su primer PC, este contaba con un bus de expansión conocido como XT que funcionaba a la misma velocidad que los procesadores Intel 8086 y 8088 (4.77 Mhz). El ancho de banda de este bus (8 bits) con el procesador 8088 formaba un tandem perfecto, pero la ampliación del bus de datos en el 8086 a 16 bits dejó en entredicho este tipo de bus (aparecieron los famosos cuellos de botella).

Dada la evolución de los microprocesadores el bus del PC no era ni mucho menos la solución para una comunicación fluida con el exterior del micro. En definitiva no podía hablarse de una autopista de datos en un PC cuando esta sólo tenía un ancho de 8 bits. Por lo tanto con la introducción del AT apareció un nuevo bus en el mundo del PC, que en relación con el bus de datos tenía finalmente 16 bits (ISA), pero que era compatible con su antecesor. La única diferencia fue que el bus XT era síncrono y el nuevo AT era asíncrono. Las viejas tarjetas de 8 bits de la época del PC pueden por tanto manejarse con las nuevas tarjetas de 16 bits en un mismo dispositivo. De todas maneras las tarjetas de 16 bits son considerablemente más

rápidas, ya que transfieren la misma cantidad de datos en comparación con las tarjetas de 8 bits en la mitad de tiempo (transferencia de 16 bits en lugar de transferencia de 8 bits).

No tan solo se amplió el bus de datos sino que también se amplió el bus de direcciones, concretamente hasta 24 bits, de manera que este se podía dirigir al AT con memoria de 16 MB. Además también se aumentó la velocidad de cada una de las señales de frecuencia, de manera que toda la circulación de bus se desarrollaba más rápidamente. De 4.77 Mhz en el XT se pasó a 8.33 Mhz. Como consecuencia el bus forma un cuello de botella por el cual no pueden transferirse nunca los datos entre la memoria y la CPU lo suficientemente rápido. En los discos duros modernos por ejemplo, la relación (ratio) de transferencia de datos ya es superior al ratio del bus.

A las tarjetas de ampliación se les ha asignado incluso un freno de seguridad, concretamente en forma de una señal de estado de espera (wait state), que deja todavía mas tiempo a las tarjetas lentas para depositar los datos deseados en la CPU.

Especialmente por este motivo el bus AT encontró sucesores de más rendimiento en Micro Channel y en el Bus EISA, que sin embargo, debido a otros motivos, no han tenido éxito.

- **BUS MICRO CHANNEL (MCA)**

Vistas las limitaciones que tenía el diseño del bus ISA en IBM se trabajó en una nueva tecnología de bus que comercializó con su gama de ordenadores PS/2. El diseño MCA (Micro Channel Architecture) permitía una ruta de datos de 32 bits, más ancha, y una velocidad de reloj ligeramente más elevada de 10 Mhz, con una velocidad de transferencia máxima de 20 Mbps frente a los 8 Mbps del bus ISA.

Pero lo que es más importante el novedoso diseño de bus de IBM incluyó un circuito de control especial a cargo del bus, que le permitía operar independientemente de la velocidad e incluso del tipo del microprocesador del sistema.

Bajo MCA, la CPU no es más que uno de los posibles dispositivos dominantes del bus a los que se puede acceder para gestionar transferencias. La circuitería de control, llamada CAP (punto de decisión central), se enlaza con un proceso denominado control del bus para determinar y responder a las prioridades de cada uno de los dispositivos que dominan el bus.

Para permitir la conexión de más dispositivos, el bus MCA especifica interrupciones sensibles al nivel, que resultan más fiables que el sistema de interrupciones del bus ISA. De esta forma es posible compartir interrupciones. Pero además se impusieron estándares de rendimiento superiores en las tarjetas de expansión.

Es cierto que el progreso conlleva un precio: La arquitectura de IBM era totalmente incompatible con las tarjetas de expansión que se incluyen en el bus ISA. Esto viene derivado de que los conectores de las tarjetas de expansión MCA eran más pequeños que las de los buses ISA. De esto se pueden sacar dos conclusiones. Por un lado el coste de estas tarjetas era menor y por otro ofrecía un mayor espacio interior en las pequeñas cajas de sobremesa.

Las señales del bus estaban reorganizadas de forma que se introducía una señal de tierra cada 4 conectores. De esta forma se ayudaba a reducir las interferencias.

- **EISA (Extended ISA)**

El principal rival del bus MCA fue el bus EISA, también basado en la idea de controlar el bus desde el microprocesador y ensanchar la ruta de datos hasta 32 bits. Sin embargo EISA

mantuvo compatibilidad con las tarjetas de expansión ISA ya existentes lo cual le obligo a funcionar a una velocidad de 8 Mhz (exactamente 8.33). Esta limitación fue a la postre la que adjudico el papel de estándar a esta arquitectura, ya que los usuarios no veían factible cambiar sus antiguas tarjetas ISA por otras nuevas que en realidad no podían aprovechar al 100%.

Su mayor ventaja con respecto al bus MCA es que EISA era un sistema abierto, ya que fue desarrollado por la mayoría de fabricantes de ordenadores compatibles PC que no aceptaron el monopolio que intentó ejercer IBM. Estos fabricantes fueron: AST, Compaq, Epson, Hewlett Packard, NEC, Olivetti, Tandy, Wyse y Zenith.

Esta arquitectura de bus permite multiproceso, es decir, integrar en el sistema varios buses dentro del sistema, cada uno con su procesador. Si bien esta característica no es utilizada más que por sistemas operativos como UNIX o Windows NT.

En una máquina EISA, puede haber al mismo tiempo hasta 6 buses principales con diferentes procesadores centrales y con sus correspondientes tarjetas auxiliares.

En este bus hay un chip que se encarga de controlar el tráfico de datos señalando prioridades para cada posible punto de colisión o bloqueo mediante las reglas de control de la especificación EISA. Este chip recibe el nombre de Chip del Sistema Periférico Integrado (ISP). Este chip actúa en la CPU como un controlador del tráfico de datos.

El motivo para que ni MCA ni EISA hayan sustituido por completo a ISA es muy sencillo: Estas alternativas aumentaban el coste del PC (incluso más del 50%) y no ofrecían ninguna mejora evidente en el rendimiento del sistema. Es más, en el momento en que se presentaron estos buses (1987–1988) esta superioridad en el rendimiento no resultaba excesivamente necesaria: Muy pocos dispositivos llegaban a los límites del rendimiento del bus ISA ordinario.

• LOCAL BUS

Teniendo en cuenta las mencionadas limitaciones del bus AT y la infalibilidad de los buses EISA y MCA para asentarse en el mercado, en estos años se han ideado otros conceptos de bus. Se inició con el llamado Vesa Local Bus (VL–Bus), que fue concebido y propagado independientemente por el comité VESA, que se propuso el definir estándares en el ámbito de las tarjetas gráficas y así por primera vez y realmente tuviera poco que ver con el diseño del bus del PC. Fueron y son todavía las tarjetas gráficas quienes sufren la menor velocidad del bus AT. Por eso surgió, en el Comité VESA, la propuesta para un bus más rápido que fue el VESA Local Bus.

♦ **Vesa Local Bus**

Al contrario que con el EISA, MCA y PCI, el bus VL no sustituye al bus ISA sino que lo complementa. Un PC con bus VL dispone para ello de un bus ISA y de las correspondientes ranuras (slots) para tarjetas de ampliación. Además, en un PC con bus VL puede haber, sin embargo, una, dos o incluso tres ranuras de expansión, para la colocación de tarjetas concebidas para el bus VL, casi siempre gráficos. Solamente estos slots están conectados con la CPU a través de un bus VL, de tal manera que las otras ranuras permanecen sin ser molestadas y las tarjetas ISA pueden hacer su servicio sin inconvenientes.

El VL es una expansión homogeneizada de bus local, que funciona a 32 bits, pero que puede realizar operaciones a 16 bits.

VESA presentó la primera versión del estándar VL–BUS en agosto de 1992. La aceptación

por parte del mercado fue inmediata. Fiel a sus orígenes, el VL-BUS se acerca mucho al diseño del procesador 80486. De hecho presenta las mismas necesidades de señal de dicho chip, exceptuando unas cuantas menos estrictas destinadas a mantener la compatibilidad con los 386.

La especificación VL-Bus como tal, no establece límites, ni superiores ni inferiores, en la velocidad del reloj, pero una mayor cantidad de conectores supone una mayor capacitancia, lo que hace que la fiabilidad disminuya a la par que aumenta la frecuencia. En la práctica, el VL-BUS no puede superar los 66 Mhz. Por este motivo, la especificación VL-BUS original recomienda que los diseñadores no empleen más de tres dispositivos de bus local en sistemas que operan a velocidades superiores a los 33 Mhz. A velocidades de bus superiores, el total disminuye: a 40 Mhz solo se pueden incorporar dos dispositivos; y a 50 Mhz un único dispositivo que ha de integrarse en la placa. En la práctica, la mejor combinación de rendimiento y funciones aparece a 33 Mhz.

Tras la presentación del procesador Pentium a 64 bits, VESA comenzó a trabajar en un nuevo estándar (VL-Bus versión 2.0).

La nueva especificación define un interface de 64 bits pero que mantienen toda compatibilidad con la actual especificación VL-BUS. La nueva especificación 2.0 redefine además la cantidad máxima de ranuras VL-BUS que se permiten en un sistema sencillo. Ahora consta de hasta tres ranuras a 40 Mhz y dos a 50 Mhz, siempre que el sistema utilice un diseño de baja capacitancia.

En el nombre del bus VL queda de manifiesto que se trata de un bus local. De forma distinta al bus ISA éste se acopla directamente en la CPU. Esto le proporciona por un lado una mejora substancial de la frecuencia de reloj (de la CPU) y hace que dependa de la línea de control de la CPU y del reloj. A estas desventajas hay que añadirle que no en todos los puntos están bien resueltas las especificaciones del comité VESA, hecho que a la larga le llevará a que el éxito del bus VL se vea empañado por ello. En sistemas 486 económicos se podía encontrar a menudo, pero su mejor momento ya ha pasado.

- PCI (Peripheral Component Interconnect)

Visto lo anterior, se puede ver que el bus del futuro es claramente el PCI de Intel. PCI significa: interconexión de los componentes periféricos (Peripheral Component Interconnect) y presenta un moderno bus que no sólo está meditado para no tener la relación del bus ISA en relación a la frecuencia de reloj o su capacidad sino que también la sincronización con las tarjetas de ampliación en relación a sus direcciones de puerto, canales DMA e interrupciones se ha automatizado finalmente de tal manera que el usuario no deberá preocuparse más por ello.

El bus PCI es independiente de la CPU, ya que entre la CPU y el bus PCI se instalará siempre un controlador de bus PCI, lo que facilita en gran medida el trabajo de los diseñadores de placas. Por ello también será posible instalarlo en sistemas que no estén basados en el procesador Intel si no que pueden usar otros, como por ejemplo, un procesador Alpha de DEC. También los procesadores PowerMacintosh de Apple se suministran en la actualidad con bus PCI.

Las tarjetas de expansión PCI trabajan eficientemente en todos los sistemas y pueden ser intercambiadas de la manera que se desee. Solamente los controladores de dispositivo deben naturalmente ser ajustados al sistema anfitrión (host) es decir a su correspondiente CPU.

Como vemos el bus PCI no depende del reloj de la CPU, porque está separado de ella por el controlador del bus. Si se instalara una CPU más rápida en su ordenador, no debería preocuparse porque las tarjetas de expansión instaladas no pudieran soportar las frecuencias de reloj superiores, pues con la separación del bus PCI de la CPU éstas no son influidas por esas frecuencias de reloj. Así se ha evitado desde el primer momento este problema y defecto del bus VL.

El bus PCI emplea un conector estilo Micro Channel de 124 pines (188 en caso de una implementación de 64 bits) pero únicamente 47 de estas conexiones se emplean en una tarjeta de expansión(49 en caso de que se trate de un adaptador bus-master); la diferencia se debe a la incorporación de una línea de alimentación y otra de tierra. Cada una de las señales activas del bus PCI está bien junto o frente a una señal de alimentación o de tierra, una técnica que minimiza la radiación.

El límite práctico en la cantidad de conectores para buses PCI es de tres; como ocurre con el VL, más conectores aumentarían la capacitancia del bus y las operaciones a máxima velocidad resultarían menos fiables.

A pesar de presentar un rendimiento similar al de un bus local conectado directamente, en realidad PCI no es más que la eliminación de un paso en el microprocesador. En lugar de disponer de su propio reloj, un bus PCI se adapta al empleado por el microprocesador y su circuitería, por tanto los componentes del PCI están sincronizados con el procesador. El actual estándar PCI autoriza frecuencias de reloj que oscilan entre 20 y 33 Mhz.

A pesar de que las tarjetas ISA no pueden ser instaladas en una ranura PCI, no debería renunciarse a la posibilidad de inserción de una tarjeta ISA. Así pues, a menudo se puede encontrar en un equipo con bus PCI la interfaz «puente» llamada «PCI-To-ISA-Bridge». Se trata de un chip que se conecta entre los distintos slots ISA y el controlador del bus PCI. Su tarea consiste en transponer las señales provenientes del bus PCI al bus ISA. De esta manera pueden seguir siendo utilizadas las tarjetas ISA al amparo del bus PCI.

A pesar de que el bus PCI es el presente, sigue habiendo buses y tarjetas de expansión ISA ya que no todas las tarjetas de expansión requieren las ratios de transferencia que permite el bus PCI. Sin embargo las tarjetas gráficas, tarjetas SCSI y tarjetas de red se han decantando cada vez más fuertemente hacia el bus PCI. La ventaja de la velocidad de este sistema de bus es que este hardware puede participar del continuo incremento de velocidad de los procesadores.

- SCSI (Small Computer System Interface)

Además de todas las arquitecturas mencionadas anteriormente, también hay que mencionar a SCSI. Esta tecnología tiene su origen a principios de los años 80 cuando un fabricante de discos desarrollo su propia interface de E/S denominado SASI (Shugart Associates System Interface) que debido a su gran éxito comercial fue presentado y aprobado por ANSI en 1986.

SCSI no se conecta directamente al microprocesador sino que utiliza de puente uno de los buses anteriormente nombrados.

Podríamos definir SCSI como un subsistema de E/S inteligente, completa y bidireccional. Un solo adaptador host SCSI puede controlar hasta 7 dispositivos inteligentes SCSI conectados a él.

Una ventaja del bus SCSI frente a otros interfaces es que los dispositivos del bus se direccionan lógicamente en vez de físicamente. Esto sirve para 2 propósitos:

- ◆ Elimina cualquier limitación que el PC-Bios imponga a las unidades de disco.
- ◆ El direccionamiento lógico elimina la sobrecarga que el host podría tener en manejar los aspectos físicos del dispositivo como la tabla de pistas dañadas. El controlador SCSI lo maneja.

Es un bus que a diferencia de otros buses como el ESDI puede tener hasta 8 dispositivos diferentes conectados al bus (incluido el controlador). Aunque potencialmente varios dispositivos pueden compartir un mismo adaptador SCSI, sólo 2 dispositivos SCSI pueden comunicarse sobre el bus al mismo tiempo.

El bus SCSI puede configurarse de tres maneras diferenciadas que le dan gran versatilidad a este bus:

- ◆ **Único iniciador/Único objetivo:** Es la configuración más común donde el iniciador es un adaptador a una ranura de un PC y el objetivo es el controlador del disco duro. Esta es una configuración fácil de implementar pero no aprovecha las capacidades del bus SCSI, excepto para controlar varios discos duros.
- ◆ **Único iniciador/Múltiple objetivo:** Menos común y raramente implementado. Esta configuración es muy parecida a la anterior excepto para diferentes tipos de dispositivos E/S que se puedan gestionar por el mismo adaptador. Por ejemplo un disco duro y un reproductor de CD-ROM.
- ◆ **Múltiple iniciador/Múltiple objetivo:** Es mucho menos común que las anteriores pero así es como se utilizan a fondo las capacidades del bus.

Dentro de la tecnología SCSI hay 2 generaciones y una tercera que está a punto de generalizarse. La primera generación permitía un ancho de banda de 8 bits y unos ratios de transferencia de hasta 5 MBps. El mayor problema de esta especificación fue que para que un producto se denominara SCSI solo debía cumplir 4 códigos de operación de los 64 disponibles por lo que proliferaron en el mercado gran cantidad de dispositivos SCSI no compatibles entre sí.

Esto cambió con la especificación 2.0 ya que exigía un mínimo de 12 códigos, por lo que aumentaba la compatibilidad entre dispositivos. Otro punto a favor de SCSI 2.0 es el aumento del ancho de banda de 8 a 16 y 32 bits. Esto se consigue gracias a las implementaciones wide (ancho) y fast (rápido). Combinando estas dos metodologías se llega a conseguir una transferencia máxima de 40 Mbps con 32 bits de ancho (20 Mbps con un ancho de banda de 16 bits).

El protocolo SCSI 3.0 no establecerá nuevas prestaciones de los protocolos, pero si refinará el funcionamiento de SCSI.

• AGP (Accelerated Graphics Port)

La tecnología AGP, creada por Intel, tiene como objetivo fundamental el nacimiento de un nuevo tipo de PC, en el que se preste especial atención a dos facetas: gráficos y conectividad.

La especificación AGP se basa en la especificación PCI 2.1 de 66 Mhz (aunque ésta apenas se usa, dado que la mayoría de las tarjetas gráficas disponibles tan sólo son capaces de utilizar la velocidad de bus de 33 Mhz), y añade tres características fundamentales para incrementar su rendimiento: operaciones de lectura/escritura en memoria con *pipeline*, demultiplexado de datos y direcciones en el propio bus, e incremento de la velocidad hasta los 100 Mhz (lo que supondría unos ratios de transferencia de unos 800 Mbytes por segundo, superiores en más de 4 veces a los alcanzados por PCI).

Pero el bus AGP es también un bus exclusivamente dedicado al apartado gráfico, tal y como se deriva de su propio nombre, *Accelerated Graphics Port* o bus acelerado para gráficos. Esto tiene como consecuencia inmediata que no se vea obligado a compartir el ancho de banda con otros componentes, como sucede en el caso del PCI.

Otra característica interesante es que la arquitectura AGP posibilita la compartición de la memoria principal por parte de la aceleradora gráfica, mediante un modelo que Intel denomina *DIME* (*Direct Memory Execute*, o ejecución directa a memoria) y que posibilitará mejores texturas en los futuros juegos y aplicaciones 3D, al almacenar éstas en la RAM del sistema y transferirlas tan pronto como se necesiten.

◆ **ENTRADA Y SALIDA**

◆ **Funciones que realiza**

Vamos a señalar las funciones que debe realizar un computador para ejecutar trabajos de entrada/salida:

- ◇ **Direccionamiento o selección del dispositivo** que debe llevar a cabo la operación de E/S.
- ◇ **Transferencia de los datos** entre el procesador y el dispositivo (en uno u otro sentido).
- ◇ **Sincronización y coordinación** de las operaciones.

Esta última función es necesaria debido a la deferencia de velocidades entre los dispositivos y la CPU y a la independencia que debe existir entre los periféricos y la CPU (por ejemplo, suelen tener relojes diferentes).

Se define una transferencia elemental de información como la *transmisión de una sola unidad de información* (normalmente un byte) *entre el procesador y el periférico o viceversa*. Para efectuar una transferencia elemental de información son precisas las siguientes funciones:

- ◇ **Establecimiento de una comunicación física** entre el procesador y el periférico para la transmisión de la unidad de información.
- ◇ **Control de los periféricos**, en que se incluyen operaciones como prueba y modificación del estado del periférico. Para realizar estas funciones la CPU gestionará las líneas de control necesarias.

Definiremos una operación de E/S como el *conjunto de acciones necesarias para la transferencia de un conjunto de datos* (es decir, una transferencia completa de datos). Para la realización de una operación de E/S se deben efectuar las siguientes funciones:

- ◇ **Recuento de las unidades de información** transferidas (normalmente bytes) para reconocer el fin de operación.
- ◇ **Sincronización de velocidad** entre la CPU y el periférico.
- ◇ **Detección de errores** (e incluso corrección) mediante la utilización de los códigos necesarios (bits de paridad, códigos de redundancia cíclica, etc.)
- ◇ **Almacenamiento temporal de la información**. Es más eficiente utilizar un *buffer* temporal específico para las operaciones de E/S que utilizan el área de datos del programa.
- ◇ **Conversión de códigos**, conversión serie/paralelo, etc.

◆ **Dispositivos externos**

Una de las funciones básicas del computador es comunicarse con los dispositivos exteriores, es decir, el computador debe ser capaz de enviar y recibir datos desde estos dispositivos. Sin

esta función, el ordenador no sería operativo porque sus cálculos no serían visibles desde el exterior.

Existe una gran variedad de dispositivos que pueden comunicarse con un computador, desde los dispositivos clásicos (terminales, impresoras, discos, cintas, etc.) hasta convertidores A/D y D/A para aplicaciones de medida y control de procesos. De todos los posibles periféricos, algunos son de lectura, otros de escritura y otros de lectura y escritura (es importante resaltar que este hecho siempre se mira desde el punto de vista del proceso). Por otra parte, existen periféricos de almacenamiento también llamados memorias auxiliares o masivas.

La mayoría de los periféricos están compuestos por una parte mecánica y otra parte electrónica. Estas partes suelen separarse claramente para dar una mayor modularidad. A la componente electrónica del periférico se le suele denominar controlador del dispositivo o, también, adaptador del dispositivo. Si el dispositivo no tiene parte mecánica (como, por ejemplo, la pantalla de un terminal), el controlador estará formado por la parte digital del circuito. Frecuentemente los controladores de los dispositivos están alojados en una placa de circuito impreso diferenciada del resto del periférico. En este caso es bastante habitual que un mismo controlador pueda dar servicio a dispositivos de características similares.

El principal problema planteado por los periféricos es su gran variedad que también afecta a las velocidades de transmisión. Por tanto, el mayor inconveniente que encontramos en los periféricos es la diferencia entre sus velocidades de transmisión y la diferencia entre éstas y la velocidad de operación del computador.

♦ Uso de interrupciones

Un computador debe disponer de los elementos suficientes para que el programador tenga un control total sobre todo lo que ocurre durante la ejecución de su programa. La llegada de una interrupción provoca que la CPU suspenda la ejecución de un programa e inicie la de otro (rutina de servicio de interrupción). Como las interrupciones pueden producirse en cualquier momento, es muy probable que se altere la secuencia de sucesos que el programador había previsto inicialmente. Es por ello que las interrupciones deben controlarse cuidadosamente.

De esta forma, podemos resumir todas las etapas **seguidas ante una interrupción** en un sistema dotado de vectorización. Estos pasos son los siguientes:

- ♦ El dispositivo envía la solicitud de interrupción mediante la línea INTR.
- ♦ El procesador termina la ejecución de la instrucción en curso y analiza la línea de petición de interrupción, INTR. Si esta línea no está activada continuará normalmente con la ejecución de la siguiente instrucción, en caso contrario se pasa a la etapa siguiente.
- ♦ La CPU reconoce la interrupción, para informar al dispositivo de ello, activa la línea de reconocimiento de interrupción, INTA.
- ♦ El dispositivo que recibe la señal INTA envía el código de interrupción por el bus de datos.
- ♦ La CPU calcula la dirección de memoria donde se encuentra la rutina de servicio de interrupción (vector de interrupción).
- ♦ El estado del procesador, y en particular el contador de programa, se salva en la pila de la misma forma que en una llamada a procedimiento.
- ♦ La dirección de la rutina de servicio de interrupción se carga en el contador de programa, con lo que se pasa el control a la citada rutina.
- ♦ La ejecución continúa hasta que el procesador encuentre la instrucción de retorno de interrupción.
- ♦ Cuando se encuentre la instrucción de retorno de interrupción se restaura el estado del

procesador, en especial el contador de programa, y se devuelve el control al programa interrumpido.

Normalmente la primera instrucción de la rutina de servicio tendrá como fin desactivar las interrupciones para impedir el anidamiento, por otra parte, antes de devolver el control al programa interrumpido se volverán a habilitar si es necesario.

◆ Tipos

• Dispositivos de entrada

Estos dispositivos permiten al usuario del ordenador introducir datos, comandos y programas en la CPU. El dispositivo de entrada más común es un **teclado** similar al de las máquinas de escribir. La información introducida con el mismo, es transformada por el ordenador en modelos reconocibles. Otros dispositivos de entrada son los **lápices ópticos**, que transmiten información gráfica desde tabletas electrónicas hasta el ordenador; **joysticks** y el **ratón** o mouse, que convierte el movimiento físico en movimiento dentro de una pantalla de ordenador; los **escáneres** luminosos, que leen palabras o símbolos de una página impresa y los traducen a configuraciones electrónicas que el ordenador puede manipular y almacenar; y los módulos de **reconocimiento de voz**, que convierten la palabra hablada en señales digitales comprensibles para el ordenador. También es posible utilizar los dispositivos de almacenamiento para introducir datos en la unidad de proceso. Otros dispositivos de entrada, usados en la industria, son los **sensores**.

• Dispositivos de Entrada/Salida

Los dispositivos de almacenamiento externos, que pueden residir físicamente dentro de la unidad de proceso principal del ordenador, están fuera de la placa de circuitos principal. Estos dispositivos almacenan los datos en forma de cargas sobre un medio magnéticamente sensible, por ejemplo una cinta de sonido o, lo que es más común, sobre un disco revestido de una fina capa de partículas metálicas. Los dispositivos de almacenamiento externo más frecuentes son los **disquetes** y los **discos duros**, aunque la mayoría de los grandes sistemas informáticos utiliza bancos de unidades de almacenamiento en **cinta magnética**. Los discos flexibles pueden contener, según sea el sistema, desde varios centenares de miles de bytes hasta bastante más de un millón de bytes de datos. Los discos duros no pueden extraerse de los receptáculos de la unidad de disco, que contienen los dispositivos electrónicos para leer y escribir datos sobre la superficie magnética de los discos y pueden almacenar desde varios millones de bytes hasta algunos centenares de millones. La tecnología de **CD-ROM**, que emplea las mismas técnicas láser utilizadas para crear los discos compactos (CD) de audio, permiten capacidades de almacenamiento del orden de varios cientos de megabytes (millones de bytes) de datos. También hay que añadir los recientemente aparecidos **DVD** que permiten almacenar más de 4 Gb de información.

• Dispositivos de salida

Estos dispositivos permiten al usuario ver los resultados de los cálculos o de las manipulaciones de datos de la computadora. El dispositivo de salida más común es la unidad de visualización, que consiste en un **monitor** que presenta los caracteres y gráficos en una pantalla similar a la del televisor. Por lo general, los monitores tienen un tubo de rayos catódicos como el de cualquier televisor, aunque los ordenadores pequeños y portátiles utilizan hoy pantallas de cristal líquido (LCD, acrónimo de Liquid Crystal Displays) o electroluminiscentes. Otros dispositivos de salida más comunes son las **impresoras**, que permiten obtener una copia impresa de la información que reside en los dispositivos de almacenamiento, las **tarjetas de sonido** y los **módem**. Un módem enlaza dos ordenadores transformando las señales digitales en analógicas para que los datos puedan transmitirse a través de las líneas telefónicas convencionales.

♦ MODOS DE DIRECCIONAMIENTO Y FORMATOS

♦ Introducción

El campo de operación de una instrucción especifica la operación que se debe realizar. Esta debe ser ejecutada sobre algunos datos almacenados en registros del computador o en palabras de memoria, es decir, sobre los operandos. El **modo de direccionamiento** especifica la forma de interpretar la información contenida en cada campo de operando para localizar, en base a esta información, el operando.

Los ordenadores utilizan **técnicas de direccionamiento** con los siguientes fines:

♦ **Dar versatilidad** de programación al usuario proporcionando facilidades tales como índices, direccionamientos indirectos, etc., esta versatilidad nos servirá para manejar estructuras de datos complejas como vectores, matrices, etc.

♦ **Reducir el número de bits del campo de operando.**

Al usuario que tiene poca experiencia, la variedad de modos de direccionamiento en un ordenador le puede parecer excesivamente complicada. Sin embargo, la disponibilidad de diferentes esquemas de direccionamiento le da al programador experimentado flexibilidad para escribir programas que son más eficientes en cuanto a número de instrucciones y tiempo de ejecución.

Es tal la importancia de los modos de direccionamiento que la potencia de una máquina se mide tanto por su repertorio de instrucciones como por la variedad de modos de direccionamiento que es capaz de admitir.

Definición: Los modos de direccionamiento de un ordenador son las *diferentes formas de transformación del campo de operando de la instrucción en la dirección del operando.*

En esta definición el término *dirección* debe interpretarse en su sentido más general de localización del operando, en cualquier lugar, y no en el sentido más estricto de dirección de memoria.

A la dirección obtenida de las transformaciones anteriores la llamaremos dirección efectiva. Esta dirección, en el caso de tratarse de una dirección de memoria, es la que se cargará en el M.A.R. o registro de dirección de memoria.

Llamando x a la información del campo de operando y $A_{ef.}$ a la dirección efectiva, la función f que a partir de x nos da $A_{ef.}$ constituirá el modo de direccionamiento empleado:

$$A_{ef.} = f(x)$$

En la evaluación de la función f pueden intervenir otras informaciones además de la información presente en el campo de operando de la instrucción. Estas informaciones pueden residir en registros del procesador o en memoria.

La especificación del modo de direccionamiento puede ir en el código de operación o en el campo de cada operando. Normalmente se codifica en el código de operación si el número de modos es pequeño, en caso contrario se codifica con cada operando, esta última forma de codificación favorece la ortogonalidad.

♦ Modos de direccionamiento más usuales

En los párrafos siguientes se irán describiendo los modos de direccionamiento más frecuentes. La mayoría de estos modos son comunes a muchas máquinas, sin embargo, hay otros que sólo se usan en determinados procesadores.

- Direccionamiento implícito

En este modo, llamado también inherente, *el operando se especifica en la misma definición de la instrucción*. El modo implícito se usa para hacer referencia a operandos de dos tipos:

- ◊ **Registros:** En el caso de que el código de operación se refiera en particular a un registro.
- ◊ **Operandos en la pila:** En el caso de que la operación se realice siempre sobre el dato situado en la cima de pila.

El primer caso es típico de las organizaciones de un solo acumulador. Generalmente en un ordenador de este tipo todas las instrucciones que actúan sobre el acumulador utilizan direccionamiento implícito.

En el segundo caso están la mayoría de las instrucciones de los ordenadores con organización de pila. Estas operaciones llevan implícitos los operandos que son los elementos de la cima de pila. Esto se debe a que en este tipo de máquinas la mayoría de las operaciones no tienen campos de dirección. También están en este caso las instrucciones PUSH y POP de la mayoría de los ordenadores cuyo operando implícito también es, como en el caso anterior, la cima de pila.

- Direccionamiento inmediato (o literal)

En este modo es el operando el que figura en la instrucción no su dirección. En otras palabras *el campo de operando contiene él mismo, sin transformación alguna, la información sobre la que hay que operar*. Este modo es útil para inicializar registros o palabras de memoria con un valor constante.

- Direccionamiento directo por registro

Se mencionó anteriormente que el campo de dirección de una instrucción puede especificar una palabra de memoria o un registro M procesador. Cuando se da este último caso se dice que el operando está especificado con direccionamiento directo por registro, en tal caso, *el operando reside en uno de los registros del procesador* que es seleccionado por un campo de registro de k bits en la instrucción. Un campo de k bits puede especificar uno de 2k registros. Este modo es típico de los ordenadores con organización de registros de uso general.

Las **ventajas** de este modo son:

- ◊ **El acceso a los registros es muy rápido**, por tanto el direccionamiento por registro debe usarse en las variables que se usen con más frecuencia para evitar accesos a memoria que son más lentos, un ejemplo muy típico del uso de este direccionamiento son los índices de los bucles.
- ◊ **El número de bits necesarios para especificar un registro es mucho más pequeño** que el necesario para especificar una dirección de memoria, esto es debido a que el número de registros del procesador es muy pequeño comparado con el número de direcciones de memoria. Sin embargo, hay que tener en cuenta que en los ordenadores modernos el número de registros ha aumentado considerablemente.

- Direccionamiento directo (o absoluto)

Este es el modo de direccionamiento más sencillo. El campo *de dirección no necesita transformación alguna para dar la dirección efectiva*, es decir la función que transforma el campo de operando en la dirección efectiva es la identidad. Esto significa que el campo de

operando es ya la dirección efectiva.

Este direccionamiento sólo se usa en ordenadores pequeños en que el programa siempre se sitúa en la misma zona de memoria ya que dificulta la relocalización de los programas, es decir que el código de los programas no dependa de su situación en memoria. Ordinariamente este modo sólo se usa para acceder a direcciones del sistema que normalmente se refieren a operaciones de entrada y salida ya que estas direcciones no dependen del programa.

- Direccionamiento indirecto

En este modo *el campo de operando de la instrucción indica la localización de la dirección efectiva del operando*. El modo de direccionamiento indirecto puede adquirir diferentes formas según cuál sea el lugar donde se encuentre la dirección del operando. En general, todos los modos de direccionamiento tienen su versión indirecta que añade un eslabón más a la cadena del direccionamiento. Por ejemplo existe el direccionamiento indirecto por registro, en el que el registro especificado contiene la dirección del operando, no el operando mismo.

Este direccionamiento es útil cuando se trabaja con **punteros** ya que los punteros son variables que contienen las direcciones de los operandos, no los operandos mismos.

- Direccionamiento relativo

Hay algunos modos de direccionamiento en que se hace uso de una propiedad muy generalizada de los programas denominada **localidad de referencia**, esta propiedad consiste en que *las direcciones referenciadas por los programas no suelen alejarse mucho unas de otras* y, por tanto, suelen estar concentradas en una parte de la memoria. Estas consideraciones nos llevan a la conclusión de que no es necesario utilizar todos los bits de la dirección de memoria en el campo de operando, basta utilizar los bits precisos para cubrir la parte de memoria donde estén incluidas las direcciones a las que el programa hace referencia. Esto puede hacerse tomando como referencia un punto de la memoria y tomando como campo de operando la diferencia entre ese punto y la dirección efectiva del operando. La dirección que se toma como punto de referencia puede residir en un registro de la CPU y, por tanto, sumando el contenido de ese registro con el campo de operando obtendremos la dirección efectiva. Hay varios direccionamientos basados en esta técnica que reciben diferentes nombres dependiendo de cuál sea el registro en el que radica la dirección tomada como referencia. Todos ellos podrían catalogarse como direccionamientos **relativos a un registro**.

El direccionamiento denominado habitualmente relativo toma como valor de referencia el **registro contador de programa**. Cuando se usa este modo de direccionamiento, el campo de operando consiste en un número (normalmente con signo) que expresa la diferencia entre la dirección del dato y la dirección siguiente a la instrucción en curso (contenida en el contador de programa). Si el campo de operando, llamado en este caso **desplazamiento u offset**, es positivo el operando residirá en una dirección posterior a la de la instrucción y si es negativo, en una dirección anterior.

Este modo de direccionamiento es usado muy frecuentemente en programas cuyo código deba ser independiente de la posición de memoria donde estén situados (programas relocalizables) ya que el desplazamiento es independiente de la localización del programa. También se usa con mucha frecuencia en instrucciones de bifurcación.

Los apartados siguientes se refieren a diferentes versiones de direccionamientos

relativos a registros.

- Direccionamiento por base y desplazamiento

Este modo de direccionamiento se fundamenta en la propiedad de localidad de referencia mencionada anteriormente. La dirección que se toma como referencia de la zona de memoria en la que están localizados los datos se deposita en un registro denominado registro base y el campo de operando indica la diferencia entre el registro base y la dirección M operando. Normalmente se toma como referencia (registro base) la dirección de comienzo de la zona de memoria ocupada por un programa. Por tanto, *la dirección efectiva del operando se calculará sumando el contenido del registro base con el campo de operando.*

Este modo de direccionamiento se usa en ordenadores que pueden mantener en memoria varios programas ya que, de esta forma, los diferentes registros base pueden contener las direcciones de comienzo de cada uno de los programas. Esto es muy útil porque facilita la relocalización de los programas. para situar el programa en una zona de memoria diferente bastará con cambiar el contenido de su registro base, no será necesario cambiar ninguno de los campos de operando.

- direccionamiento indexado

En este modo de direccionamiento, *la dirección del operando también se calcula sumando un registro de la CPU al campo de operando, este registro es un registro específico para este uso llamado registro índice.* En los ordenadores con organización de registros generales, el registro índice puede ser cualquiera de los registros de la CPU. En los ordenadores en que el contador de programa es considerado como un registro de uso general (PDP-11 y VAX) el modo relativo es un caso particular del direccionamiento indexado. A la cantidad que hay que sumar al registro índice para conseguir la dirección del operando también se le llama **desplazamiento u offset**. Este modo de direccionamiento es especialmente útil para el direccionamiento de vectores y matrices en bucles ya que, si se quieren direccionar elementos consecutivos del vector o matriz, basta mantener en el desplazamiento la dirección del primer elemento e ir incrementando el registro índice. También sirve para acceder de forma relativa a elementos de vectores cercanos a uno dado, para ello, se carga la dirección del elemento de referencia en el registro índice y después se accede mediante direccionamiento indexado, con el desplazamiento adecuado, al anterior, al siguiente, etc., esto mismo también es aplicable a pilas, en que, en ocasiones, hay que acceder a datos cercanos, por encima o por debajo, al dato señalado por algún apuntador. Una consecuencia de todo esto es una modalidad de direccionamiento indexado de –que disponen algunos ordenadores, denominada **autoindexación**, que hace que el registro índice sea incrementado o decrementado en el tamaño del operando antes o después de acceder al mismo. Los ordenadores que poseen autoindexación incorporan los modos de direccionamiento descritos en los dos apartados siguientes. En algunos ordenadores existen variantes del direccionamiento indexado en que se *obtiene la* dirección del operando sumando el contenido de varios registros con el desplazamiento, esto puede servir para especificar el comienzo de un vector mediante un desplazamiento respecto a un registro y el elemento del vector mediante un registro índice.

- Direccionamiento autoincremental o postincremental

En este modo, *la dirección del operando se encuentra en un registro y éste es incrementado, después de acceder al operando, en el tamaño del mismo.*

Este modo es útil para manejar vectores y matrices como se veía en el apartado anterior. También se puede utilizar para extraer datos de pilas (que crezcan hacia direcciones bajas) ya que, si el registro sobre el que se aplica este modo es el apuntador de pila, después de la operación el apuntador señalará al siguiente elemento de la pila.

- Direccionamiento autodecremental o predecremental

En este modo *para obtener la dirección del operando hay que decrementar un registro en el tamaño del operando; el nuevo contenido del registro después de efectuar esa operación, es la dirección del operando.*

Para finalizar, veamos un resumen de las utilidades de los modos de direccionamiento usados con más frecuencia.

MODOS	UTILIDADES
Inmediato	Operaciones con constantes.
Directo por registro	Variables locales de procedimientos no recursivos
Indirecto por registro	Variables referenciadas a través de punteros
Absoluto	Direcciones de sistema
Relativo	Variables globales
Indexado	Acceso a vectores, matrices y cadenas
Autoincremental	Desapilar parámetros de procedimientos
	Recorrido de vectores y cadenas
Autodecremental	Apilar parámetros de procedimientos
	Recorrido de vectores y cadenas hacia atrás

♦ JUICIO CRÍTICO

En este trabajo se ha tratado de describir los distintos elementos que forman parte de un computador y la relación que existe entre ellos. Se ha tratado de abordar el tema desde un punto de vista generalista, ya que el sector de la microelectrónica está en continuo desarrollo y lo que hoy puede ser un estándar, es muy posible que esté desfasado dentro de unos pocos años. Este dinamismo del sector, en principio, tiene muchos efectos positivos para las empresas y los consumidores como son el abaratamiento de los costes y el aumento de la potencia de los equipos. De esta forma, equipos que antes ocupaban un armario entero son mucho menos potentes que la gran mayoría de los ordenadores personales que cualquiera puede adquirir hoy por poco más de 100.000 ptas. Además, gran parte del éxito de estos ordenadores ha sido el que existan una serie de especificaciones comunes que todos los fabricantes respetan (PCI, SCSI, SVGA, x86, etc.) de forma que se puede construir un PC a partir de componentes de distintas marcas sin demasiados problemas lo que evita los sistemas propietarios siempre caros de mantener al estar sujeto a los diseños del fabricante. Incluso monopolios de hecho como el de Intel en el desarrollo de CPUs empiezan a tambalearse ante el afianzamiento de microprocesadores como el K6-2 de la empresa americana AMD que consiguen rendimientos similares al Pentium II con precios mucho más reducidos y una total compatibilidad con todo el software y hardware actual.

Por otro lado, en la informática corporativa, las cosas no son siempre así. Este mercado está dominado por empresas como Hewlett Packard, Compaq (gracias a la reciente adquisición de Digital), Sun Microsystems o IBM. Los sistemas profesionales integran procesadores muchísimo más potentes que cualquier Pentium II (SPARC, Alpha, etc.), corriendo bajo sistemas operativos Unix o NT y en los que la fiabilidad es un factor fundamental. En estos computadores hay una gran incompatibilidad entre las distintas plataformas, pues al ser equipos tan específicos y especializados, los fabricantes optan por producir el mejor sistema operativo (Sun con Solaris o Compaq con Digital–Unix) y los mejores componentes hardware para sus sistemas.

En definitiva, el sector informático aparece como uno de los más competitivos y con mayor desarrollo en el mundo actual tanto a nivel empresarial, industrial como en el ámbito doméstico. Es más, se podría decir que estamos en la *era del chip*. Hoy, se hace pues inconcebible, el intentar llevar a cabo cualquier proyecto empresarial sin contar con el componente informático y el capital humano necesario para trabajar con él.

Jamás se descubriría nada si nos considerásemos satisfechos con las cosas descubiertas.

Séneca el Filósofo, Lucius Annaeus (c.5 a. C.–65 d. C.).

◆ BIBLIOGRAFÍA

- ◇ *PC Interno*. MARCOMBO. Data Becker, 1995.
- ◇ *Computer Architecture and Organization, Second Edition*. HAYES, J. McGraw–Hill, 1988.
- ◇ *Computer System Architecture*. MANO, M. Prentice Hall, 1993.
- ◇ *Fundamentos de Informática*. FERNANDO SAEZ VACAS. Alianza Informática, 1991.
- ◇ *Apuntes sobre arquitectura de computadores*. JUAN ÁNGEL GARCÍA MARTÍNEZ.
- ◇ *Arquitectura de computadores* JOSÉ ANTONIO DE FRUTOS REDONDO. Alcalá de Henares : Departamento de Automática, Universidad de Alcalá de Henares, 1994.
- ◇ *Arquitectura de ordenadores*. ENRIQUE HERNÁNDEZ HERNÁNDEZ S.L. : s.n., 1998.
- ◇ *Arquitectura de ordenadores* . EDUARDO ALCALDE LANCHARRO. McGraw–Hill, Interamericana de España, 1996.
- ◇ *Arquitectura de computadores : de las técnicas básicas a las técnicas avanzadas* / GÉRARD BLANCHET, BERTRAND DUPOUY Masson, D.L. 1994
- ◇ *Arquitectura de computadores : un enfoque cuantitativo* / JOHN L. HENNESSY, DAVID A. PATTERSON. McGraw–Hill, D.L. 1993
- ◇ *Organización y arquitectura de computadores*. WILLIAM STALLINGS. Prentice Hall, 1997.

ARQUITECTURA DE COMPUTADORES

– 1 –

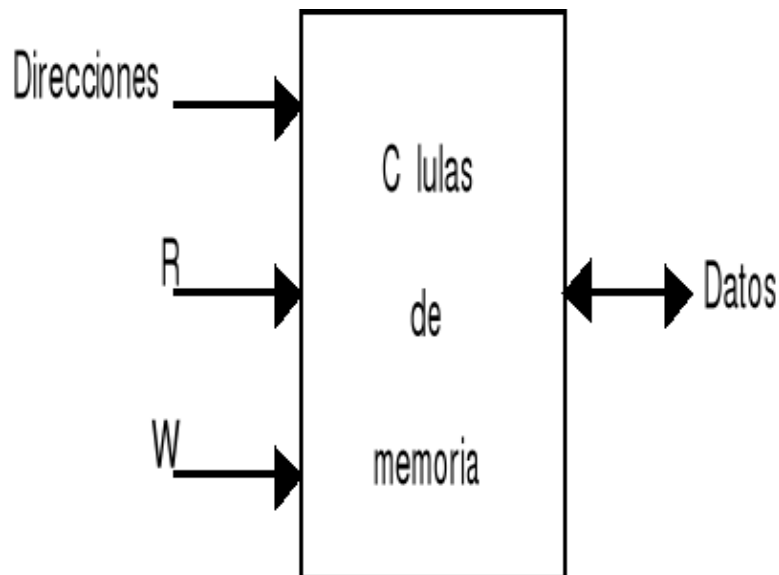


Figura 3-2

Figura 3-1

Figura 3-3

Figura 3-4

Figura 4-1

◇