

TEMA 4. MEMORIA INTERNA

1. Jerarquía de memoria

1.1. Espacio de direcciones

Se denomina espacio de direcciones a la cantidad de direcciones de memoria que un microprocesador puede referenciar. El espacio de direcciones está relacionado con el tamaño del bus de direcciones según la expresión:

$$\text{Esp. De Dir.} = 2^n$$

siendo n el tamaño del bus.

1.2. Principio de localidad

El principio de localidad puede enunciarse como:

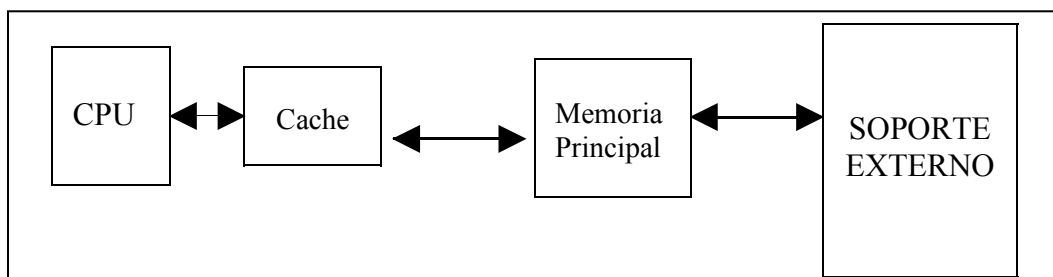
“Los programas acceden a una parte relativamente pequeña de su espacio de direcciones en un intervalo determinado de tiempo”

Hay dos tipos diferentes de localidad:

- *Localidad temporal.* Si un elemento es referenciado en el instante t , volverá a ser referenciado, presumiblemente, en un instante próximo a t .
- *Localidad espacial.* Si un elemento es referenciado en el instante t , los elementos cuyas direcciones están próximas tenderán a ser referenciado en un instante próximo a t .

Aprovechando el principio de localidad, la memoria de los ordenadores se construye de forma jerárquica. Una jerarquía de memoria consta de múltiples niveles de memoria con diferentes velocidades y tamaño. Las memorias más rápidas son, generalmente, más costosas y por tanto aparecen en menor cantidad. Debido a las diferencias en coste y tiempo de acceso, es útil construir la memoria principal como una jerarquía de niveles estando la memoria más rápida cerca del procesador y la más lenta situada al final de la jerarquía. El microprocesador siempre buscará la información en el nivel más próximo, si no está, se produce un fallo y se busca en el siguiente nivel produciéndose una transferencia de información entre ambas memorias. La cantidad mínima de información que se transfiere entre dos niveles se denomina bloque. A medida que nos alejamos de la CPU, el tamaño del bloque es mayor.

Un ejemplo típico de jerarquía de memoria puede ser:



Diseño Jerárquico de memoria

Las características de tamaño, tiempo de acceso y coste de los distintos niveles de memoria de una jerarquía vienen reflejados en la siguiente tabla:

MEMORIA	TAMAÑO	TIEMPO	COSTE
CACHE INTERNA	16 KB	2 ns	100\$/MB
CACHE EXTERNA	512 KB	8 ns	50\$/MB
DRAM	64 KB	60 ns	5\$/MB
DISCO	3,2 GB	10 ⁶ ns	100\$/GB

La residencia habitual de los datos e instrucciones de un programa en ejecución es la Memoria Principal. La Memoria Principal está formada por unos pocos chips de RAM dinámica, memoria que ofrece la mayor capacidad pero que no es particularmente rápida.

De hecho, la velocidad del procesador es mucho mayor que la velocidad de la Memoria Principal. O sea, que se necesitan muchos ciclos de procesador para acceder a un contenido situado en la Memoria Principal. Y no solo eso, ese número de ciclos se multiplica por un factor situado entre x1,5 y x2,0 cada dos años. Por tanto, desde el punto de vista del procesador la información está cada vez ‘más lejos’.

En este estado de las cosas, la Jerarquía de Memoria es la responsable de crearle al procesador la ilusión de que puede acceder en unos pocos ciclos a cualquier posición de la Memoria Principal.

La norma en los micros es organizar la memoria interna en dos niveles de memoria caché, el primero separado en datos e instrucciones, pequeño (~32KB) y muy rápido (1 o 2 ciclos del procesador). El segundo nivel es unificado (datos e instrucciones juntos), más grande (~512KB) y está más ‘alejado’ (~10 ciclos de procesador).

El movimiento de contenidos entre todos estos almacenes internos y la Memoria Principal es complejo, y debe realizarse de forma automática en unos pocos ciclos de procesador. El propósito de la Jerarquía es la colocación automática de las partes más ‘usadas’ de la memoria en los niveles más cercanos al procesador. Si algo falla en estos mecanismos de colocación, los accesos deberán servirse desde niveles lejanos, incluso desde la Memoria Principal, situada a cientos de ciclos. Durante el acceso a tan remotos almacenes el procesador se quedará inevitablemente sin trabajo y su rendimiento disminuirá de forma espectacular. Por tanto, al margen del tipo de procesador y de la clase de programa, la Jerarquía de Memoria puede ser un freno tremendo a la alta velocidad de ejecución.

Como primer eslabón de la Jerarquía debemos mencionar al Banco de Registros: un registro es un almacén rápido (acceso en 1 ciclo) que contiene un dato (p.e. un número de 64b). Veamos un ejemplo: para evaluar la expresión aritmética bastaría la ejecución de cuatro instrucciones: dos instrucciones de lectura de memoria para las variables y, una instrucción de suma y una instrucción de escritura en la posición asignada a la variable.

Las dos instrucciones de lectura leen la caché de datos y escriben los registros r1 y r2; la instrucción suma lee sus operandos del Banco de Registros y no accede a caché.

2. Registros

En el interior del micro existen unas celdas de memoria de alta velocidad que permiten a la CPU almacenar datos temporalmente mientras se efectúa alguna operación. Son los llamados registros internos, que constituyen la memoria interna del procesador. Están formados por un conjunto de bits que se manipulan en bloque. Este número varía dependiendo de la CPU, pero siempre son múltiplos de ocho y resultan imprescindible para su funcionamiento.

El tamaño del registro indica el número de bits que puede manipular a la vez el procesador; cuanto mayor sea, más potente será el micro, pues podrá trabajar con más cantidad de información a la vez. Las primeras CPUs tenían 8 bits. La CPU de los primeros PC disponían de registros de 16 bits: sólo podían ejecutar software de 16 bits (DOS, Windows 3.x) pues con este software sólo se pueden utilizar números de 16 bits para direccionar la memoria, lo que limita al procesador.

El 386 de Intel fue el primer modelo de CPU que incluyó registros de 32 bits y, por consiguiente, era capaz de ejecutar software de esa cantidad de bits.

Los registros de la CPU se pueden dividir en dos tipos: Registros visibles al usuario y Registros de control y de estado.

Registros visibles al usuario

Estos registros se definen como aquellos que pueden ser referenciados por lenguaje ensamblador o lenguaje máquina, con el fin de optimizar el uso de los recursos. Se distinguen 4 categorías:

Registros de Dirección. Contienen las direcciones de memoria donde se encuentran los datos. Algunos de los registros de dirección más usados son los registros índice y los punteros de pila.

Registros de datos. Se usan para contener datos. Esto hace que aumente la velocidad de proceso, sobre todo cuando un dato es solicitado, porque con frecuencia se deja en uno de estos registros y no es necesario acceder a la memoria.

Registros de condición. Son bits fijados mediante el hardware, que indican, por ejemplo, si una operación entrega un resultado positivo, negativo o nulo, si hay overflow (desbordamiento), etc.

Registros de control y de estado

Son los que intervienen en la ejecución de las instrucciones. Distinguimos los siguientes:

Contador de Programa (PC), también llamado Contador de Instrucciones. Contiene la dirección de la siguiente instrucción a ejecutar; su valor es actualizado por la CPU después de capturar una instrucción.

Registro de Instrucción (RI). Contiene el código de la instrucción actual. Aquí se analiza el código de operación.

Registro de Dirección de memoria (RMD – MAR). Contiene la dirección de una posición de memoria, donde se encuentra o va a ser almacenada la información; este intercambio se realiza a través del bus de direcciones.

Registro de Intercambio de memoria (RIM – MBR). Recibe o envía (dependiendo si es una operación de lectura o escritura) la información o el dato contenido en la posición apuntada por el RMD; el intercambio de datos con la memoria se realiza a través del bus de datos.

Estos cuatro registros funcionan de la siguiente forma:

- a) El registro contador de programa (CP) tiene la dirección de la próxima instrucción a ejecutar.
- b) Para buscarla, el contenido de esa posición es pasada al RDM.
- c) La instrucción apuntada por el RDM se carga en el RIM, y desde aquí al RI.
- d) El CP se incrementa en 1, para apuntar a la siguiente instrucción a ejecutar.

3. Memoria Caché

3.1. Estructura

Una caché es una memoria rápida situada entre la CPU y la RAM. Esta rapidez se consigue a partir de la tecnología que utiliza y del principio de localidad de referencias:

“La próxima referencia a la RAM que haga la CPU se encuentra localmente muy próxima a la referencial actual”.

Una caché está constituida por dos bloques: el *directorio* y la *zona de almacenamiento*.

- El directorio contiene información sobre las direcciones de memoria cuyos datos se encuentran en la caché. Está construido con memorias asociativas.
- La zona de almacenamiento contiene los datos de las direcciones incluidas en el directorio. Está construida con memoria RAM (generalmente estática, SRAM).

Cuando la CPU solicita un dato (referencia una dirección), el controlador de caché comprueba si esa dirección está en el directorio. Si está, lee el dato de la zona de almacenamiento y lo envía a la CPU, en caso contrario lee el dato de la RAM y lo guarda en la caché (junto con las direcciones que se encuentran a continuación) y envía una copia a la CPU.

3.2. Organización de una caché

Entendemos por organización de caché la estrategia de colocación de los datos en la zona de almacenamiento de forma que se optimice su uso.

- ✓ La memoria principal se divide en bloque de información de tamaño fijo. Cada bloque está formado por varias palabras que se encuentran, localmente, en posiciones contiguas.
- ✓ La zona de almacenamiento de la caché se divide en marcos de bloque. Cada marco de bloque puede guardar un bloque de memoria.

- ✓ El directorio de caché se divide en entradas. Cada entrada se refiere a un marco de bloque y guarda información sobre el bloque que almacena.
- ✓ Cuando se transfiere información entre la caché y la RAM se envía un bloque completo. Este bloque se guarda en la zona de almacenamiento y su identificador se escribe en la entrada del directorio correspondiente.

Existen varias formas de organizar los datos en la caché. La organización directa es la más simple de implementar pero la menos eficiente, la organización asociativa es la más eficiente pero la más costosa. Por último, la organización asociativa por conjuntos es un compromiso entre las dos anteriores.

ORGANIZACION DIRECTA

En la organización directa, cada bloque de memoria principal siempre va a situarse en el mismo marco de bloque de la caché. La dirección de memoria principal (dirección física) se divide en tres campos.

- ✓ Palabra. Se refiere a la posición que ocupa la palabra solicitada dentro del bloque referenciado.
- ✓ Bloque. Se refiere al marco de bloque en el cual se encuentra el bloque.
- ✓ Marca. Es un identificador del bloque y se utiliza para almacenarlo en el directorio de caché.

Al conjunto Palabra, Bloque y Marca se le denomina *dirección lógica*.

El tamaño del campo *bloque* es $\log_2 n$ siendo n el número de entradas del directorio (igual al número de marcos de bloque de la caché). El tamaño del campo *palabra* es $\log_2 m$ siendo m el número de palabras que forman el bloque. El campo *marca* es un identificador del bloque y está formado por el resto de bits que forman la dirección física.



ORGANIZACION ASOCIATIVA

En la organización asociativa cada bloque de memoria puede ir a cualquier marco de bloque en la caché. Cuando la CPU referencia a una dirección, el controlador de caché busca en todas las entradas del directorio (búsqueda asociativa) para comprobar si los datos de esa dirección están en la caché. Para ello, la dirección física se descompone en dos campos. El campo *palabra* y el campo *marca* cuyo significado es similar al descrito anteriormente.

ORGANIZACION ASOCIATIVA POR CONJUNTOS

Es un compromiso entre la asignación directa y la asociativa. Se divide la caché en conjuntos y cada conjunto se divide a su vez en marcos de bloque. El acceso a cada

conjunto es directo mientras que la búsqueda dentro del conjunto es asociativa. De esta forma, un bloque de memoria principal va siempre al mismo conjunto de la caché, pero dentro de ese conjunto se puede insertar en cualquier marco de bloque.

3.3. Características de la memoria caché

Existen algunas características que definen la operatividad de una memoria caché. Estas características se pueden agrupar en:

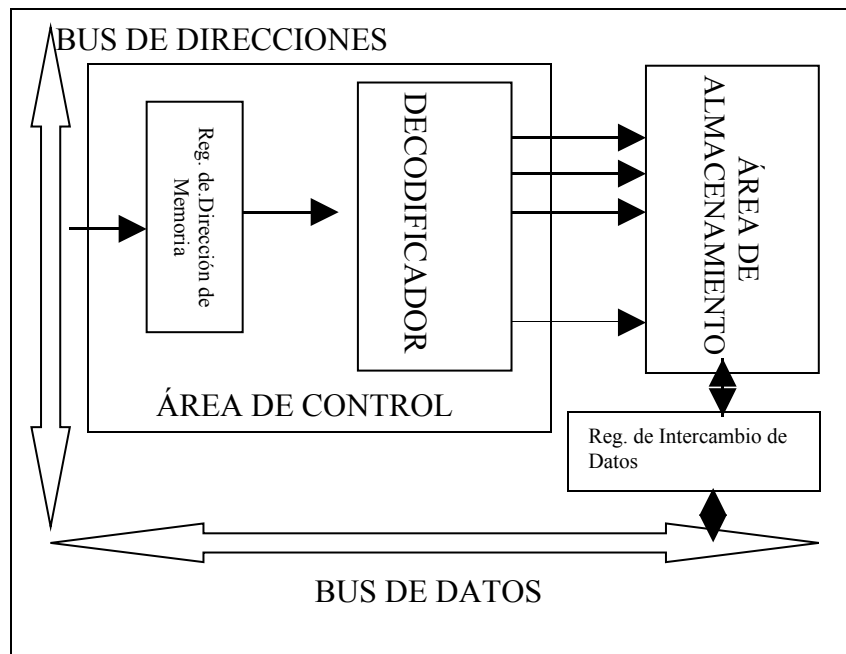
- *Ancho de banda.* Se define el ancho de banda como la velocidad de transferencia de datos entre la caché y la RAM. El ancho de banda aumenta incrementando el *bus* de conexión con la RAM de forma que en cada transferencia se envíe un bloque completo. Para ello debemos disponer de una memoria principal entrelazada con latch a la salida y con un orden de entrelazado igual al número de palabras del bloque.
- *Actualización de datos.* Las operaciones con datos las realiza la CPU en la caché. De esta forma, si modificamos el contenido de alguna dirección de memoria, esta modificación se actualizaría en la RAM.
- *Política de Reemplazo.* Cuando se produce un fallo de caché, hay que traerse de memoria principal un bloque nuevo; si hay que reemplazar uno de los antiguos, el problema es decidir cual de ellos reemplazamos. Existen varias estrategias basadas en experiencias previas:
 - Algoritmo menos reciente (MRU). Se sustituye el bloque que lleva más tiempo sin usarse.
 - Algoritmo más antiguo (FIFO). Se sustituye el bloque que lleva más tiempo en la caché.
 - Algoritmo de azar. Se elige de forma aleatoria un bloque.
- *Anticipación.* El rendimiento de una caché basa en reducir el número de fallos de caché. Basándose en el principio de localidad, la caché anticipa los datos que previsiblemente referenciará la CPU. Para ello, cada vez que se trae un bloque de memoria principal, se anticipa también el siguiente bloque. Existen dos formas de anticipación:
 - *Preextracción siempre.* Siempre que se referencia un bloque, la caché se trae el siguiente de la memoria principal. Tiene el inconveniente de que podemos tener varias copias de un mismo bloque en la caché, lo cual disminuye su rendimiento.
 - *Preextracción marcada.* Sólo traemos el siguiente bloque cuando se referencia por primera vez un bloque. Para saber si un bloque ya se ha referenciado con anterioridad, en su entrada en el directorio de caché existe un bit denominado *bit de marcado* que se activa cuando ese bloque se referencia por primera vez.
- *Rendimiento.* Se define el rendimiento de una caché para una secuencia de referencias como el cociente entre el número de aciertos y el número de referencias.

4. La memoria principal: la RAM

4.1. Estructura de una memoria

En la Memoria Principal o RAM es un dispositivo destinado a almacenar dos tipos de información: el programa o secuencia de instrucciones a ejecutar y los datos que manejan dichas instrucciones. La manipulación de los programas y los datos está dirigida por la CPU, y más concretamente por la Unidad de Control.

El esquema general de una RAM viene dado en la figura



El área de control es la encargada de localizar la posición de memoria que se corresponde con la dirección que se envía por el bus de direcciones. Consta de un Registro de Direcciones de Memoria y un Decodificador que tiene como entrada los n bits del bus de direcciones y 2ⁿ salidas para cada una de las posiciones de memoria.

El área de almacenamiento está formada por una matriz de celdas básicas de forma que cada fila se corresponde con una posición de memoria. Cada celda básica está formada por un dispositivo de almacenamiento binario que puede mantener un estado lógico (0 ó 1) durante un tiempo limitado y cuyo valor se transmitirá al Registro de Intercambio de Datos cuando la fila se activa para una operación de lectura. Cuando la operación es de escritura, el proceso es inverso.

La RAM está formada por un conjunto de casillas o posiciones de memoria capaces de almacenar un dato o una instrucción. Cada casilla contiene 8 bits, de manera que si la RAM es de 1 KB dispondrá de 1024 celdas de memoria y podrá almacenar 1024 caracteres. En la RAM es donde se almacenan los datos y programas que se

ejecutan en el ordenador. Cuando se apaga el ordenador el contenido de la RAM desaparece; por eso se dice que esta memoria es volátil.

Cada una de las casillas que forman la memoria se identifica con un número; es lo que se conoce como dirección de memoria. La finalidad es que la Unidad de Control pueda diferenciar unas casillas de otras.

Para poder realizar operaciones de lectura o de escritura de una celda de memoria o en una celda de memoria, utilizan: el registro de dirección de memoria (RDM), el registro de intercambio o de datos (RIM) y el selector de memoria o decodificador de direcciones, que es el dispositivo que conecta la celda de memoria cuya dirección figura en el RDM con el RIM, posibilitando la transferencia de datos en un sentido o en otro, dependiendo de la operación de lectura o de escritura.

Todas las operaciones de lectura y de escritura están dirigidas por la Unidad de Control. En una operación de lectura se efectúan los siguientes pasos:

- En el RDM se almacena la dirección de memoria de la celda que contiene la información
- El selector de memoria selecciona la dirección contenida en el RDM y carga en el RIM la información contenida en esa celda.
- Transfiere el contenido del RIM al registro de trabajo de la CPU para que allí se procese.

En una operación de escritura, se efectúan los siguientes pasos:

- Los datos a escribir en la memoria, que ya han sido procesados por la CPU, llegan al RIM
- En el RDM está la dirección de la celda destino de la información
- El selector de memoria selecciona la celda destino y se carga el contenido de RIM a la celda apuntada por el RDM.

En la memoria principal cada celda se implementa mediante un minúsculo condensador.

RAM Dinámica o DRAM, este nombre procede del hecho de que la carga del condensador cambia continuamente. Es decir, aunque el valor lógico de cada celda permanezca siempre a 1, el condensador va perdiendo internamente su carga a lo largo del tiempo, es decir, si ésta cae por debajo de un cierto umbral, el circuito de lectura no podrá distinguir entre carga positiva o negativa. Para evitar que esto ocurra, la carga debe refrescarse periódicamente a través de unos circuitos incluidos en el controlador de memoria.

El coste y espacio físico por celda de las memorias implementadas con condensadores es bastante mejor que en el caso de las memorias estáticas, implementadas con condensadores.

El parámetro que describe la funcionalidad de la memoria es su tamaño (se suele indicar en MB) y los parámetros que mejor cuantifican el rendimiento son:

- a) **La latencia** (tiempo de acceso a la memoria), mide el tiempo que transcurre desde que el chip recibe la dirección de una celda hasta que devuelve el dato que se encuentra en ella (este parámetro se halla condicionado por la tecnología de integración de la memoria). La latencia también se puede medir como el tiempo

de ciclo (tiempo que transcurre entre que la memoria acepta una petición y queda lista para aceptar la siguiente).

- b) **Ancho de Banda:** Cantidad de información, expresada en bytes, que se puede enviar entre la Memoria Principal y la CPU por unidad de tiempo. Este parámetro depende del ancho de banda de los buses de la placa y su frecuencia de funcionamiento.
- c) **Ancho de banda del bus:** producto del ancho de bus por la frecuencia del funcionamiento. Se mide en $\text{MHz} * \text{Bytes} = \text{MB/s}$. El ancho de banda del bus determina el número de ciclos que se necesitan, tanto para enviar la dirección a la memoria como para traerse los datos de vuelta.
- d) **Fiabilidad:** muchas implementaciones comerciales añaden bits de información adicionales que permiten detectar errores producidos durante el proceso de lectura o escritura de la memoria. Estos bits se denominan **bits de paridad**.
 - **Bit de paridad par.** Se asocia normalmente con un byte de datos, y se complementará la pantalla de datos con un 0 o un 1 de manera que la información tenga siempre un número par de 1s.
 - **Bit de paridad impar.** Todas las palabras resultarán con un número impar de 1s.

Cuando se encuentra con una palabra que no cumpla con la convención utilizada, se detecta un error y se pide una nueva lectura del dato de la memoria.

Con este sistema solo puede detectarse un número impar de errores, ya que cuando se produce un número par de errores, el dato erróneo se camufla como correcto.

Las memorias con paridad vienen con chips adicionales para cubrir dichas funciones. La memoria con paridad puede pincharse en cualquier zócalo, lleve este paridad o no, pero las memorias sin paridad sólo funcionan en zócalos sin paridad. Para saber si un módulo dispone o no de paridad, basta con contar el número de chips que el módulo posee; si el número no es potencia de dos, entonces se trata de un módulo con paridad.

- **Memoria ECC.** ECC es la abreviatura de reconocimiento y corrección de errores. Las memorias que poseen esta característica son capaces de reconocer errores internos (errores de 1 bit o errores de 2 bits). En casos especialmente graves se puede llegar a detener por completo el ordenador. Los errores de 1 bit pueden corregirse en funcionamiento mediante la comparación de las sumas de comprobación. Las memorias con función ECC son más fiables, por lo que normalmente se emplean en servidores.

5. La memoria principal: la ROM

La ROM es una memoria de sólo lectura que contiene información vital para el funcionamiento correcto del ordenador. Algunas ROM son programables, es decir, parte de la información que contienen puede cambiarse por el usuario y por tanto, necesita estar alimentada constantemente. Para ello se utiliza una pila que actúe como fuente de

alimentación. Esta pila funciona mientras el ordenador está apagado y utiliza la alimentación de la red para recargarse cuando el ordenador está conectado. De esta forma, la pila puede durar indefinidamente (salvo problemas de humedad, cortocircuito, etc). Actualmente los ordenadores no emplean chip de memoria ROM, han sido reemplazados por las memorias EEPROM (Memorias ROM eléctricamente borrables y programables).

La **BIOS** es el elemento encargado de establecer la conexión entre el hardware y el software. Cuando se enciende el ordenador hay que establecer cuáles son los recursos disponibles y donde se encuentra el software del Sistema Operativo. Estas funciones las realiza la BIOS (Subrutinas Básicas de Entrada/Salida) a partir de cierta información que está almacenada de forma permanente en la placa base. La BIOS está dividida en varios chips que se reparten entre los distintos elementos hardware del ordenador:

- La **ROM BIOS** es la BIOS del sistema. Se trata de un chip que se encuentra en la placa base y que contiene un pequeño programa de arranque que chequea los recursos disponibles. También contiene las rutinas de E/S que permiten al procesador comunicarse con el exterior mediante el mecanismo de las interrupciones.
- La **RAM CMOS** es la parte configurable de la BIOS. Contiene información básica sobre algunos recursos del sistema que son susceptibles de ser modificados como el disco duro el tipo de disco flexible, etc. Esta información es almacenada en una RAM con tecnología CMOS (bajo consumo) alimentada por una pila que se encuentra en la placa base.
- La **ROM de dispositivos**. La mayoría de los dispositivos de E/S tienen su propia ROM para almacenar datos que necesitan para su correcto funcionamiento. Esta ROM suele estar incluida en la tarjeta controladora del dispositivo.

Toda la información de la BIOS no se almacena en la ROM sino que existe una zona de la RAM dedicada exclusivamente a guardar datos de la BIOS.

6. Memoria virtual

La memoria física que puede usar un microprocesador es, generalmente, mucho menor que su espacio de direcciones. Para resolver esta diferencia se utiliza una técnica denominada Memoria Virtual. Consiste en suponer que el espacio de direcciones está formado por la RAM y el disco. De esta forma, sólo una parte de este espacio de direcciones estaría disponible en la RAM y el resto de las direcciones se encontrarían en el disco. Cuando un dato no está presente en la RAM se produce un fallo y se cambia su contenido por un bloque del disco de forma similar al mecanismo de la memoria caché. Esta técnica permite resolver dos problemas:

- Sistemas multiusuario en los cuales cada usuario puede estar ejecutando un programa distinto
- Programas extensos que no pueden almacenarse en la RAM (Resuelve el problema de los overlays)

La organización de la memoria virtual es similar a la de la caché. El espacio de direcciones se divide en páginas de tamaño fijo y la memoria RAM en marcos de página. Algunas páginas se encontrarán en la RAM y otras en el disco. Cuando se

referencia a una dirección cuya página no está en memoria principal se produce un fallo y se transfiere la página desde el disco a la RAM reemplazando a una de las existentes.

Puesto que un fallo de página implica una penalización temporal elevada (se necesitan varios cientos de ciclos de reloj para cambiar de página), su tamaño debe ser suficientemente grande para reducir el número de fallos. Actualmente se utilizan tamaños de página de 16 a 64 KB.

Para gestionar la memoria virtual, el Sistema Operativo dispone de una tabla de localización de páginas (TLB) en la que se anotan en cada entrada la dirección de comienzo de la página que contiene.

Cuando se produce un fallo de página se transfiere del disco a la página solicitada y se sustituye por una de las que hay en memoria. Los algoritmos de sustitución de páginas son similares a los de gestión de caché: MRU, LRU, LIFO y FIFO.

7. Memorias entrelazadas

El principal problema con operaciones de memoria es el tiempo que se emplea en localizar la posición de almacenamiento pues la señal que contiene la dirección tiene que atravesar un elevado número de puertas con sus retrasos correspondientes. Para reducir este problema se ha diseñado varios métodos de implantación de memoria principal que permiten adelantar los datos de memoria antes de que la CPU los solicite. Una de esta técnica es la memoria entrelazada. Se trata de construir la memoria principal mediante bloques de memoria contiguos de forma que dos direcciones de memoria consecutivas se encuentran en dos módulos consecutivos.

Cuando se solicita una dirección, la RAM lee 4 direcciones, una de cada bloque (direcciones 0, 1, 2 y 3). Teniendo en cuenta que la mayoría de programas son secuenciales, la próxima dirección que solicitará la CPU será la 1, después la 2, etc. De esta forma, no hay que volver a decirle a la RAM que busque estas direcciones pues ya las ha leído. De esta forma se ha eliminado el tiempo de decodificación de las direcciones.

Mediante esta técnica, cada dirección de memoria (dirección física) se descompone en dos campos, el primero indica la dirección que hay que leer en cada módulo (es la misma para todos) y un segundo campo que indica a que módulo corresponde la petición de la CPU. Al conjunto de estos campos se le denomina dirección lógica.

Existen dos técnicas de entrelazar la memoria principal: entrelazado con latches a la entrada y entrelazado con latches a la salida. Ambas son similares pero poseen características diferenciadoras que hacen que cada una de ellas pueda ser usada en situaciones específicas.

7.1. Memorias entrelazadas con latches a la salida

En las memorias entrelazadas con latches a la salida, la dirección física se convierte en una dirección lógica con dos campos. El primero indica la dirección de memoria que deben leer todos los bloques de memoria. Cada bloque lee su dirección y deposita el dato obtenido en el registro que hay a la salida. El campo bloque actúa como control del multiplexor seleccionando una de las cuatro palabras que hay en los registros. De esta forma, aunque en el bus de datos sólo se deposita una palabra, las

demás ya están leídas y pueden seleccionarse en los siguientes ciclos de reloj eliminando el tiempo que se emplearía para decodificar la dirección.

7.2. Memorias entrelazadas con latches a la entrada

La organización de este tipo de entrelazado es similar a la anterior pero, en este caso, los registros se encuentran a la entrada de los bloques de memoria y no se utilizan para almacenar datos sino direcciones.

El campo dirección indica la dirección de memoria que debe leer el módulo pero, a diferencia del caso anterior, esta vez sólo lee el módulo que contiene la dirección referenciada por la CPU. El registro de entrada captura el dato y deja libre el bus de direcciones. De esta forma, la siguiente referencia de la CPU puede ser leída por otro bloque sin esperar a que termine el anterior. El campo bloque se utiliza como señal de control de multiplexor para seleccionar como salida la correspondiente al bloque que enviamos a leer.

7.3. Memorias asociativas

Una memoria asociativa es una memoria direccionable por contenidos. Es conceptualmente opuesto a la memoria convencional: se utiliza como entrada un dato y la memoria nos devuelve la dirección en la que se encuentra el dato.

Las aplicaciones más usuales de las memorias asociativas son en la construcción de los ficheros de índices de las grandes bases de datos indexadas y en los directorios de las memorias Caché.

8. Descomposición de la memoria

8.1. Bancos

La memoria del PC se encuentra distribuida en bancos. Estos pueden tener distinta longitud, pero deben tener la misma anchura que además deberá coincidir con el bus de datos.

Ventajas de descomponer la memoria en bancos:

- 1) Mientras uno de los bancos está funcionando, el resto de ellos se encuentran desactivados. Se comparte circuitos sin producir conflicto.
- 2) Dota de mayor flexibilidad al sistema de memoria a la hora de repartir el espacio total de almacenamiento.
- 3) Permite la configuración de un mapa de memoria heterogéneo, que permite direcciones de memoria de diferentes características. Es conveniente que las memorias más rápidas se encuentren en los bancos con una numeración más baja, ya que suelen ser los más usados.

8.2. Módulos

Cada banco se descompone en módulos de la misma anchura y longitud, pero también con la misma velocidad y características. Puede haber bancos con un solo módulo. Los más populares son SIMM, DIMM y RIMM.

8.3. Chips

Los módulos están formados por chips. Todos los de un mismo módulo tendrán que ser idénticos, salvo el encargado de albergar los bits de paridad (suele ser menos ancho).

La anchura del módulo será la suma de las anchuras de cada uno de los chips que lo componen.

Cuando se quiere aumentar la longitud doblando el número de chips, cada mitad actuará siempre por separado, pero cuando se duplica la anchura, todos los chips deberán actuar al mismo tiempo.

8.4. Celdas

Cada chip está formado por una matriz de celdas a las que se accede por dirección. Cada dirección está formada por su dirección de columna, y su dirección de fila. En todos los chips se accede al mismo tiempo a la misma dirección de celda, y entre todos nos da la palabra.

8.5. Entrelazado

El entrelazado de factor k consiste en dividir la memoria en k bloques de anchura o longitud con objeto de poder acceder a todos ellos simultáneamente por la misma dirección.

Cuanto mayor es k , mayor rendimiento se puede obtener del sistema, pero también se encarece el controlador de memoria.

Esquemas:

- a) **Superior.** Consiste en que cada bloque de los k contiene direcciones de memoria consecutivas. No se usa mucho. Se hace más fácil la expansión de memoria y la reparación de rangos defectuosos.
- b) **Inferior.** Las direcciones consecutivas se van repartiendo circularmente entre los k bloques. Muy extendido. Muy eficiente ya que los programas cumplen la propiedad de localidad espacial.

9. Conexión a la placa base

La memoria puede presentarse en distintos formatos atendiendo a su integración física.

DIP (Doble Patilla Alineada). Chip de memoria rectangular con 16 patas de metal. Formato utilizado por los ordenadores más antiguos (8088 y llegó a utilizarse en los PCs hasta el procesador 80386).

SIMM (Módulos de memoria simple alineada). Son pequeñas láminas de circuito impreso denominadas PCB que tienen chips de memoria soldados y que se insertan en unos zócalos dispuestos para tal efecto en la placa base. Las láminas pueden contener un número variado de chips. Los chips pueden ir montados sobre una sola cara o sobre ambas. Los contactos pueden fabricarse en metal u oro.

DIMM (Módulos de memoria doble alineadas). Puede verse como un par de SIMMs de 72 contactos implementados conjuntamente.

SIPP. Son realmente SIMMs pero con patillas en lugar de contactos. Estas patillas se diseñan para ser insertadas en un zócalo longitudinal que es mucho más barato que los zócalos SIMM estándar, pero ocupan un mayor espacio en la placa base y dificultan la ventilación de los chips de memoria. Resulta difícil encontrar chips de memoria SIPP en los computadores actuales.

Actualmente, todos los computadores del entorno doméstico utilizan el esquema de módulos SIMM / DIMM para la memoria convencional.

10. Formato

Los módulos DIP, SIMM y SIPP se fabrican en distintos formatos que determinan la anchura de la palabra de memoria.

En el entorno PC los módulos SIMM / DIMM se han impuesto sobre los otros.

- c) **SIMM de 30 contactos.** Estos módulos eran frecuentes en los sistemas bajo procesadores con bus de datos de 8 y 16 bits. Sus capacidades comerciales son: 256 Kb, 1 Mb y 4 Mb.
- d) **SIMM de 72 contactos.** Comenzó a utilizarse a partir del procesador 80846 de Intel. Su uso se ha propagado hasta las placas del Pentium y para los primeros Pentium II.
- e) **DIMM de 168 contactos.** Entran en escena con la llegada de los Pentium. Los 168 contactos están dispuestos 84 por cada cara. Una memoria DIMM es parecido a dos memorias SIMM.
- f) **DIMM de 184 contactos.** Se emplea con la memoria SDRAM DDR.
- g) **RIMM.** Aparecen con la llegada de la memoria RDRAM de Rambus Inc. A mediados de los años 90 con el fin de introducir un módulo de memoria con niveles de rendimiento muy superiores a los módulos de memoria SDRAM de 100 MHz y 133 MHz disponible en aquellos años.

Los módulos **RIMM RDRAM** cuentan con 184 pins y debido a sus altas frecuencias de trabajo requieren de difusores de calor consistentes en una placa metálica que recubre los chips del módulo. Se basan en un bus de datos de 16 bits y están disponibles a velocidades de 300 MHz, 356 MHz, 400 MHz y 533 MHz que por su pobre bus de 16 bits tenía un rendimiento 4 veces menor que la DDR.

Inicialmente los módulos RIMM fueron introducidos para su uso en servidores basados en Intel Pentium III.

A pesar de tener la tecnología RDRAM niveles de rendimiento muy superiores a la tecnología SDRAM y las primeras generaciones de DDR RAM, debido al alto costo de esta tecnología no han tenido gran aceptación en el mercado de PCs.

- h) **SODIMM**. Es una versión compacta del módulo DIMM convencional. Viene en dos tamaños, 44 y 72 pines. Se utiliza en ordenadores portátiles.

11. Tipos de memoria

11.1. Caché

CACHÉ ASÍNCRONA

- Primera caché que se usó como caché L2 en los 80386
- Mismo tipo de caché que existía desde el principio
- Su latencia oscilaba entre 20 y 12 ns. Demasiado lenta como para poder acceder de manera síncrona con el microprocesador.

CACHÉ SÍNCRONA

Todas las señales que controlaban el funcionamiento del microprocesador pueden funcionar a la misma frecuencia que éste, dándole mayor rapidez y simplificando su diseño. Tipos:

a) CACHÉ EN MODO RÁFAGA

Guarda la dirección a la entrada de caché y genera las direcciones consecutivas hasta llenar la línea de caché. Entonces se almacena.

El número de direcciones generadas sería 4 ya que cada línea de caché solía tener tamaño de 32 bytes, y los buses un tamaño de 8 bytes.

El coste era inferior al de la caché segmentada. Ahora es muy raro encontrarlo.

Tiempo de acceso entre 12 y 8 ns.

b) CACHÉ SEGMENTADA

Añade circuitos tanto a la entrada de la dirección como en la salida de los datos para desacoplar estas dos operaciones y poder hacer un pipeline de dos etapas.

Su latencia oscila entre 7 y 2,5 ns.

11.2. RAM

DRAM ó FPM DRAM

Es una memoria de gran capacidad de almacenamiento. Este tipo de memoria necesita refrescarse para que la información que contiene no se pierda. Presenta el inconveniente de que no se puede acceder a ella mientras se está refrescando, otro inconveniente que presenta esta memoria es que es una memoria volátil. Tiene la ventaja de que es una memoria de acceso directo, para acceder a un dato no hace falta pasar por los demás datos.

EDO ó EDO-RAM

Evoluciona de la FPM. Permite empezar a introducir nuevos datos mientras los anteriores están saliendo, lo que la hace algo más rápida.

BEDO

Es una evolución de la EDO RAM que mejora su velocidad gracias a que envía los datos en “ráfagas”. Casi no se usa. Compite en prestaciones con la SDRAM

SRAM

Es una memoria de menor capacidad que la DRAM pero es más cara y más rápida. La diferencia que existe con la DRAM es que no necesita refrescarse para conservar la información. Es una memoria más pequeña que la anterior pero también es volátil y de acceso directo. Un ejemplo es la memoria caché.

SDRAM

La memoria SDRAM es una memoria que incorpora la capacidad de la memoria DRAM y la velocidad de la memoria SRAM.

DDRAM o DDR (memoria de doble carga)

Esta memoria se refresca el doble de veces que la memoria DRAM. Tiene la ventaja de ser el doble de rápida que las anteriores memorias. También llamada SDRAM-II o DDR SDRAM.

Rambus DRAM o RDRAM

La memoria Rambus es un tecnología exclusiva de la empresa Rambus, es una memoria a la que se puede acceder de manera muy rápida, es hasta 10 veces más rápida que la memoria DRAM, pero con el inconveniente de que es muy cara. Es utilizada en el Pentium 4.

11.3. ROM

PROM

La memoria PROM puede ser programable por el usuario, una vez que el usuario la ha programado ya no se puede cambiar más su información, con lo cual se convierte en una memoria ROM, permitiéndonos sólo su lectura.

EPROM

También es conocida con el nombre de PROM – regrabable, se llama así porque nos permite grabar y borrar tantas veces como se quiera, su información se borra mediante radiaciones ultravioleta. Presenta el inconveniente que no se puede borrar sólo una parte de memoria, sino que se borra toda la información.

EEPROM

Es una memoria que puede borrar su información mediante elevadas corrientes eléctricas. La diferencia que existe entre las EPROM y EEPROM es que en esta última sí podemos borrar sólo una parte de la memoria, quedando el resto de la información intacta. Presenta el inconveniente que sólo se puede cambiar un byte de información cada vez.

Flash Memory

Es un tipo de EEPROM que se puede reprogramar en bloques. Se usa en la BIOS de los equipos.