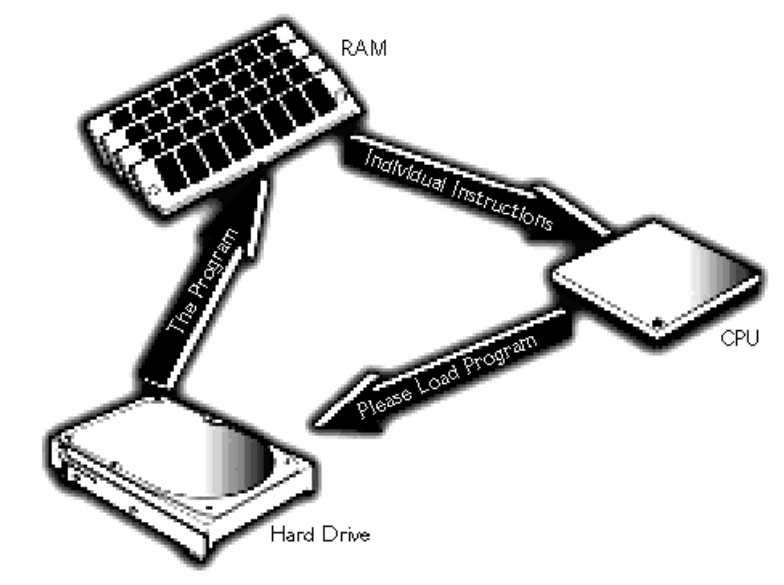
UNIDAD Nº 2
MEMORIA

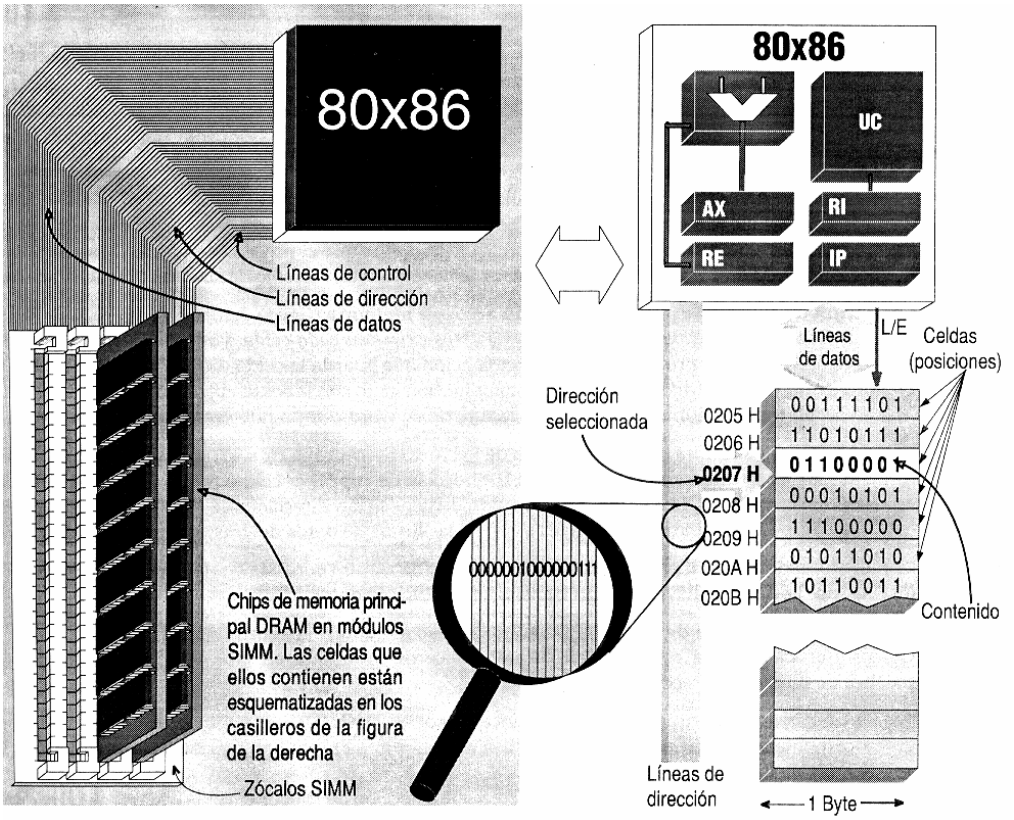
FUNDAMENTOS

La memoria principal es el espacio de trabajo del procesador de la computadora. Es un área de almacenamiento provisional en dónde deben residir los programas y datos utilizados por el procesador. El almacenamiento en memoria es considerado temporal porque los datos y programas permanecen en ella en tanto que la computadora tenga suministro eléctrico o no sea reiniciada. Antes de apagar o reiniciar la computadora, cualquier dato debe ser guardado en un dispositivo de almacenamiento permanente (generalmente un disco duro) con el fin de que pueda ser cargado en memoria cuando se requiera.

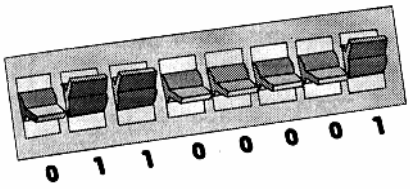


La memoria principal de la computadora almacena bits (unos y ceros) en celdas independientes, aisladas entre sí, que contienen un byte completo (8 bits) de información. Cada celda se localiza en el conjunto mediante un número binario identificatorio, que constituye su *dirección*, o indicación de su "posición" en ese conjunto. Este número no se puede alterar, pues está establecido circuitalmente. Por lo tanto, en relación con cada celda se tienen dos números binarios: un número fijo, la dirección, que presentado en los circuitos de la memoria permite acceder a una celda; y un número de 8 bits, que es el contenido informativo de esa celda, o sea la combinación de unos y ceros almacenada en ella. Este número puede cambiarse si la memoria es alterable.

Es costumbre representar las celdas de una memoria, o una porción de ella, mediante un conjunto de casilleros verticales formando una "escalera", siendo sus direcciones números binarios consecutivos. Estos números binarios se escriben en papel o en pantalla al lado de cada celda, en su equivalente hexadecimal, a fin de no tener que visualizar largas cadenas de unos y ceros. En la figura se supone que en las líneas del bus de direcciones (ampliado con la "lupa') se halla presente la dirección 0000 0010 0000 0111 = 0207h, en la cual está almacenado el byte 01100001 = 61h.



Puede ayudar a entender mejor el concepto de byte almacenado, si se piensa que en cada casillero existen 8 llaves del tipo "si-no", como las comunes de pared para encender la luz, cada una para retener un uno ("si") o un cero ("no"). Entonces, para una celda dada, como la que contiene 01100001, la combinación de unos y ceros que están formando las 8 llaves es la información contenida en dicha celda.



La información que almacena cada grupo de 8 llaves puede referirse a instrucciones o datos. El modelo de las llaves también es útil para tener presente que en cada posición de memoria siempre existe una cierta combinación de unos y ceros, o sea no es posible que ella no contenga "nada", pues las 8 llaves siempre están presentes, cada una en "si" o en "no". En cada dirección de memoria (celda) sólo pueden leerse o escribirse 8 bits por vez, sin posibilidad de operar menor cantidad de bits, o un bit aislado. Cuando los datos o instrucciones ocupen más de un byte, se almacenan fragmentados en varios bytes, los cuales deben estar contenidos en celdas correspondientes a posiciones consecutivas de memoria, o sea en direcciones sucesivas.

Las memorias residen en microcircuitos electrónicos construidos sobre una delgada capa de silicio semiconductor *(chip o circuito integrado)*. Éste se protege con un encapsulado cerámico o plástico, y se le conectan pines metálicos externos para el conexionado eléctrico. Se requieren varios chips para lograr el total de memoria necesaria *(ver módulos de memoria)*

Parámetros de las memorias

Tiempo de acceso: El tiempo de acceso es el tiempo transcurrido desde que se solicita un dato a la unidad de memoria hasta que ésta lo entrega. *Se mide en nanosegundos (ns) Ej:* $t_a = 60$ *ns.*
NOTA: 1 nanosegundo es la milmillonésima parte del segundo (1 .10^{-9} *seg)*

Velocidad de transferencia: Es la cantidad de información por unidad de tiempo que se transmite entre la memoria principal y la CPU. *Se mide en Megabytes por segundo o Megabits por segundo.* La velocidad también puede expresarse en MHz, y en ese caso MHz=1000/ns siendo Hz=ciclos / seg.

Capacidad de memoria: Es la cantidad total de bytes que pueden almacenarse en una memoria.

La unidad de la capacidad es el *Byte (B).* Sus múltiplos son: *el Kilobyte (KB), el Megabyte (MB), el Gigabyte (GB) y el Terabyte (TB).*

1 KB corresponde a 1024 Bytes *(*2^{10}*)*, que es la potencia de dos más cercana a 1000. El múltiplo siguiente es 1024 veces mayor.

1 KB = 1024 Bytes (2^{10})
1 MB = 1024 KB = 1048576 Bytes (2^{20})
1 GB = 1024 MB = 1048576 KB = 1073741824 Bytes (2^{30})

Para pasar de Bytes a KB se debe dividir por 1024. Para hacer el pasaje inverso, se debe multiplicar por 1024.

Es frecuente expresar también la capacidad de almacenamiento en bits. Para pasar de bits a bytes, se debe dividir la capacidad por 8. Ej: 64 Kb = 8 KB

Acceso aleatorio: Una memoria es de acceso aleatorio cuando el tiempo de acceso a cualquier dato es siempre el mismo (igual tiempo de acceso para cualquier dirección).

Volatilidad: Es la pérdida de la información ante la falta de alimentación eléctrica.

Memoria principal. Clasificación

La memoria principal es la memoria en dónde se almacenan los programas a ser ejecutados. La memoria principal se divide en:

RAM: *(Random Access Memory ó Memoria de Acceso Aleatorio*)* Es una memoria de lectura/escritura, volátil. (La información se pierde en ausencia de alimentación eléctrica)
** El término RAM, que significa memoria de acceso aleatorio no coincide el uso que se le da actualmente. El término se aplica a toda memoria volátil de lectura/escritura.*

ROM: *(Read Only Memory o Memoria de Lectura Solamente)* Es una memoria que sólo se puede leer, no volátil (No se puede escribir y su información permanece, aún en ausencia de alimentación eléctrica)

Memorias ROM

Las memorias ROM se usan comúnmente para almacenar programas que deban estar siempre disponibles en la computadora. El ejemplo más común es el BIOS *(Basic Input-Output System)*, que está grabado en una ROM llamada *ROM BIOS*. Una de las funciones más importantes de la ROM BIOS es iniciar el sistema durante el encendido de la máquina, momento en el cual la memoria RAM no está inicializada.

Nomenclatura de las memorias ROM:

Tipo de memoria	Nomenclatura
ROM	23nnnn (ya en desuso)
PROM	27nnnn
EPROM	27nnnn (con ventana de cuarzo)
EEPROM	28nnnn o 29nnnn

nnnn es la capacidad en Kb Ej: 27512 = 512 Kb = 64 KB

Las memorias ROM actuales se clasifican en:

ROM propiamente dichas, ROM verdadera o de máscara: Son memorias ROM programadas durante el proceso de fabricación e inalterables. Como un CD de audio estampado, sólo conviene en grandes cantidades.

Mientras que el término ROM supone la memoria no puede escribirse, hay algunas variantes que permiten cambiar su contenido ocasionalmente, estas son:

PROM (Programmable ROM): Las PROMs son un tipo de ROM en blanco, en su estado original, y deben ser programadas una sola vez con cualquier información que se requiera.

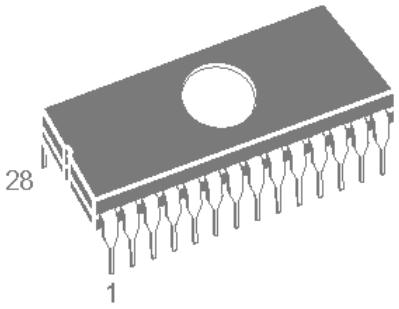
En su estado original (en blanco), las PROMs presentan 1 (unos) binarios en todas sus direcciones. Una PROM en blanco puede ser programada por medio de un programador ROM.

El proceso de programar una ROM se conoce como "quemado", una palabra que describe apropiadamente el proceso: La mayoría de los chips funcionan con 5 voltios; cuando se programa una PROM se utiliza un voltaje más alto (normalmente 12 V) en las distintas direcciones del chip. Este voltaje hace circular una corriente que quema fusibles en las direcciones deseadas, convirtiendo el 1 en 0. Este proceso no es reversible, aunque es posible pasar de 1 a 0 es imposible convertir un 0 en 1. La programadora de dispositivos examina el programa que se desea escribir en el chip y cambia los 1s a 0s, en dónde es necesario.

Los chips PROM son también conocidos como OTP *(One Time Programmable - Programables una sola vez)*. Es posible programarlos una sola vez y nunca se borran. La mayoría de las PROMs son muy baratas.

UVEPROM (UV Erasable Programmable ROM): Una EPROM es una PROM borrable. Un chip UVEPROM es una memoria PROM borrable por medio de radiación UV y puede ser reconocido fácilmente por la ventana de cristal de cuarzo colocada en el encapsulado, que permite ver directamente el chip. Las UVEPROM utilizan la misma nomenclatura que las PROM, y son funcionalmente idénticas. El cristal de cuarzo incrementa el precio de las EPROMs, si se las compara con las OTP. Este costo es

innecesario si no se quiere borrar el chip. El tiempo normal de borrado es de 5 minutos para una UVEPROM típica, ubicada a 3 cm. de una lámpara fluorescente UV de 8W (de tipo germicida). Para reprogramar estas memorias es necesario extraerlas, exponerlas a radiación UV y conectarlas a un grabador adecuado.

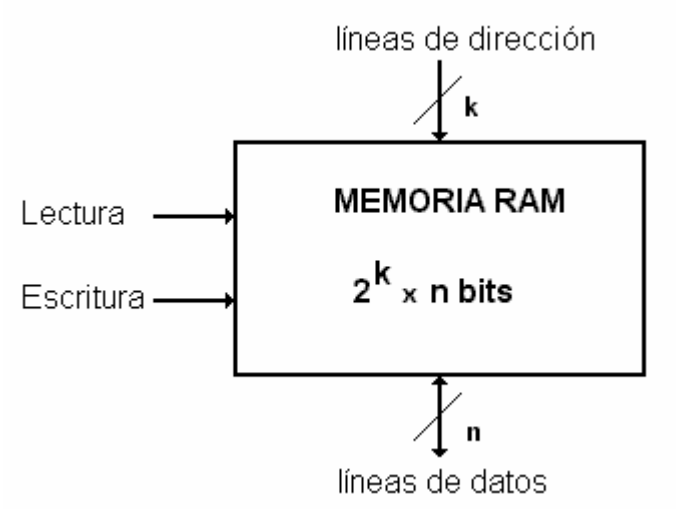


EEPROM - E²PROM - FLASH ROM: (Electrically Erasable Programable ROM):
Son memorias ROM que se pueden borrar eléctricamente y luego se pueden reprogramar. La diferencia entre las memorias EEPROM y FLASH es que las primeras pueden ser borradas y reprogramadas byte por byte, mientras que las últimas lo hacen de a bloques. A partir de 1994 las tarjetas madre *(motherboard)* comenzaron a usar ROM FLASH. Una memoria FLASH puede actualizarse fácilmente, sin tener que sustituirla o extraerla, como en el caso de las UVEPROM, pues obtienen la tensión eléctrica de borrado desde la misma placa en la que están conectadas. En la mayoría de los casos, se debe bajar información actualizada para la ROM de la tarjeta madre del sitio Web del fabricante, y después ejecutar un programa para grabarla.

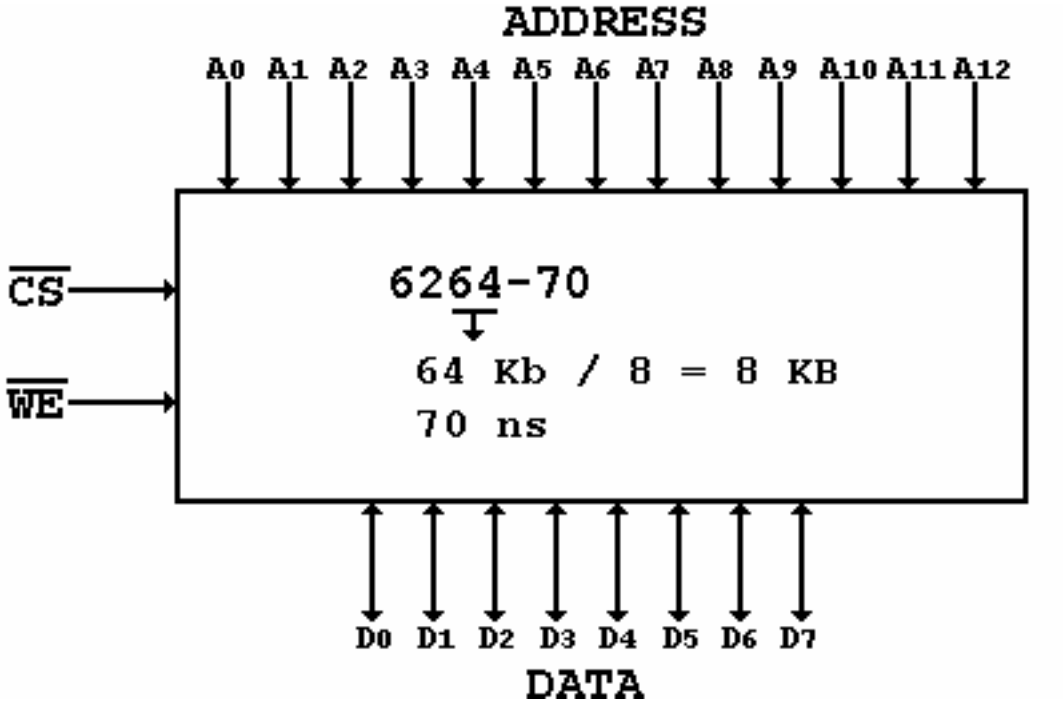
Por lo general, una memoria ROM BIOS puede actualizarse para:
- *Agregar discos duros mayores a 8 GB*
- *Agregar discos duros Ultra-DMA/33 ó Ultra-DMA/66*
- *Agregar CDROM ATAPI de arranque (especificación "El Torito")*
- *Agregar o mejorar la compatibilidad Plug and Play*
- *Corregir los errores del año 2000*
- *Corregir las imperfecciones con el hardware*
- *Agregar microprocesadores de nuevos tipos y velocidades*

El tiempo de acceso de la ROM es, en general, mucho mayor que el tiempo de acceso de la RAM (aproximadamente 150 ns), por esta razón, es frecuente que en las computadoras actuales, para acelerar la ejecución, se copie automáticamente el contenido de ROM a RAM en el arranque. La porción de RAM que se utiliza para ello se conoce como *"Shadow RAM" (RAM de sombra).* Esta posibilidad se encuentra en el SETUP de la motherboard; si el sistema funcionara exclusivamente bajo Windows 95, colocar memoria "Shadow" no mejoraría el rendimiento debido a que este sistema operativo no utiliza las rutinas de la BIOS y por ende el acceso a la ROM no es frecuente.

Memorias RAM



Memoria RAM genérica de 2^k x n bits



Memoria RAM tipo 6264, tiempo de acceso 70 ns

A_x = Address (Dirección)
D_x = Data (Dato)
CS# (*) = Chip Select (Selección de Chip)
WE# (**) = Write Enable (Habilitación de Escritura)

(*) La barra superior indica que la entrada se activa con estado lógico cero (0) o con una transición de alto a bajo. Puede ser reemplazada por el carácter #
(**) Para escritura, WE# = 0. Para lectura, WE# = 1.

Las dos operaciones que puede realizar una memoria RAM son: lectura y escritura.

Para extraer una palabra de la memoria (lectura), se debe:

1) Seleccionar el chip *(CS# = 0)*
2) Activar la entrada de lectura *(WE# = 1)*
3) Aplicar la dirección binaria en las líneas de dirección *(A_0 .. A_{12})*.

Luego de este proceso, se debe esperar un tiempo t_A *(tiempo de acceso)* para que la unidad de memoria entregue el dato binario *(D_0 .. D_7)* en las líneas de datos *(Datos Válidos)*. Una vez transcurrido un tiempo t_{RC} *(tiempo de ciclo de lectura)*, la memoria está dispuesta para comenzar un nuevo ciclo.

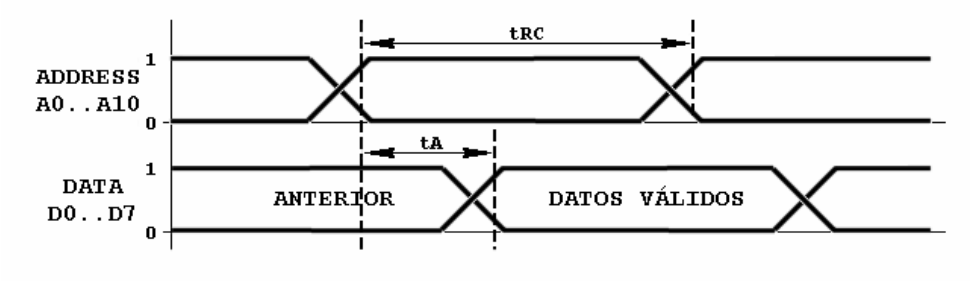


Diagrama de tiempos. Ciclo de lectura

t_A = Address Access Time
t_{RC} = Read Cycle Time

Para que una nueva palabra se almacene en una memoria (escritura), se debe:
1) Seleccionar el chip *(CS# = 1)*
2) Aplicar la dirección binaria en las líneas de dirección *(A_0 .. A_{12})*.
3) Activar la entrada de escritura *(WE# = 0)*.
4) Colocar los bits de datos en las líneas de datos *(D_0 .. D_7)*.

En ese momento, y luego de un tiempo t_{AW} *(tiempo de acceso a escritura)*, la memoria almacena en su interior el dato colocado en las líneas de datos (el contenido anterior se pierde). Luego de un tiempo t_{WC} *(tiempo de ciclo de escritura)*, la memoria está dispuesta para comenzar un nuevo ciclo.

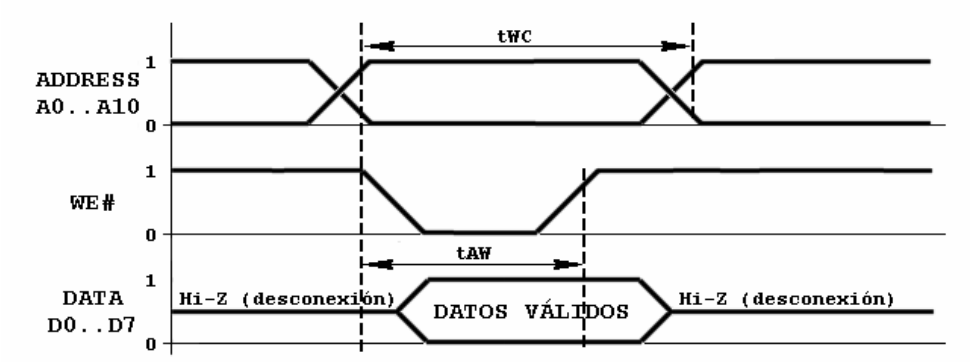


Diagrama de tiempos. Ciclo de escritura

t_{AW} = Address Valid to End of Write
t_{WC} = Write Cycle Time

NOTA: La acción de colocar la dirección binaria de la celda en las líneas de dirección se conoce como *"direccionamiento"*

Clasificación de las memorias RAM

Las memorias RAM se dividen en dos grandes categorías:

SRAM: *(Static RAM - RAM Estáticas)* Las memorias *RAM estáticas* son memorias de lectura/escritura de alta velocidad. Como contrapartida, presentan baja capacidad y alto costo. Para su fabricación, se necesitan 4 transistores por bit (6 transistores por bit en las memorias más confiables). Tienen baja densidad, cada celda ocupa 30 veces más lugar para la misma capacidad que la otra tecnología, DRAM.

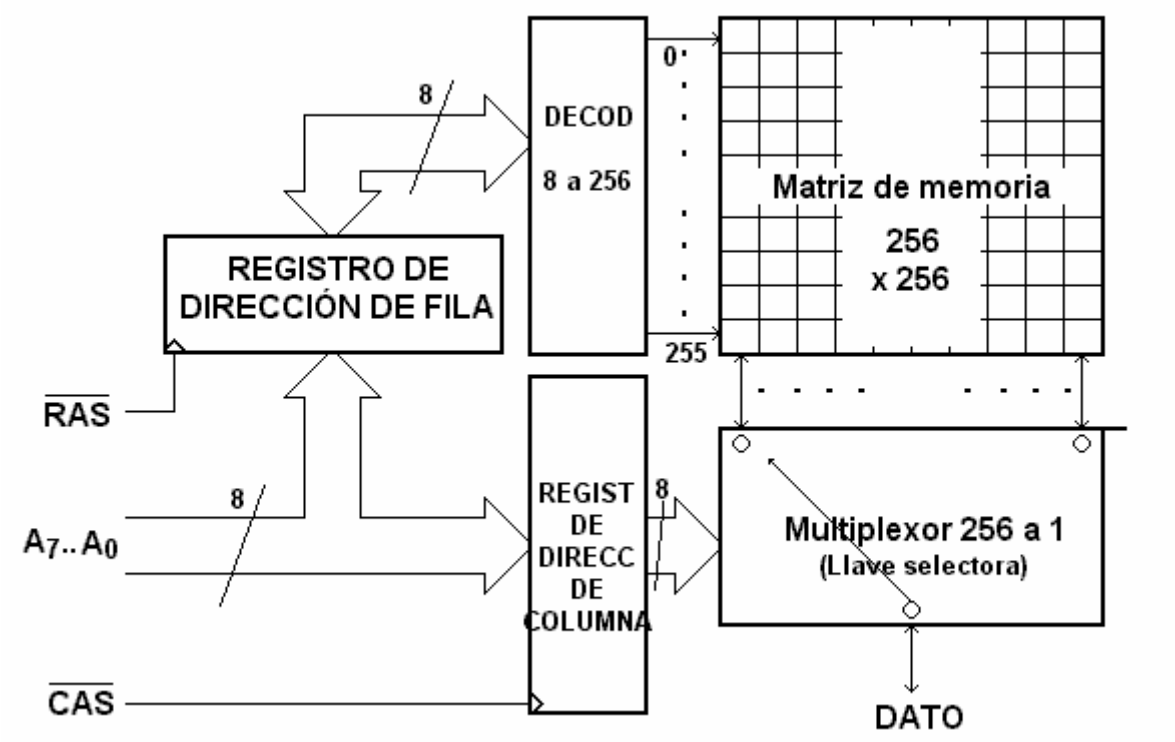
DRAM: *(Dynamic RAM - RAM Dinámicas)* Las memorias *RAM dinámicas* tienen alta capacidad y un costo mucho más bajo que las memorias estáticas, por lo que todos los fabricantes de computadoras las suelen usar como memoria principal.

Sin embargo, las memorias dinámicas son más lentas que las estáticas (Por ej.: 60 ns contra 10 ns). Cada bit de memoria utiliza un capacitor microscópico que necesita ser continuamente recargado. Para ello se requiere un circuito especial de "refresco" que actúa constantemente para evitar que cada celda de memoria dinámica pierda su contenido. Por este motivo son menos confiables que las RAM estáticas, pero por su alta capacidad son las memorias usadas con preferencia como memoria principal. Utilizan un transistor y un capacitor por bit, por ello presentan mayor densidad que las memorias estáticas. El período de refresco es entre 10 y 20 ms.

Cuadro comparativo:

DRAM	SRAM
Mayor densidad	Mayor velocidad
Bajo costo por MB	No precisa refresco. Más confiable
Uso como memoria principal	Uso en caché y registros del micro

Memorias DRAM



Arquitectura de un chip DRAM de 64K x 1 bit (Para completar 1 byte se requieren 8 chips)

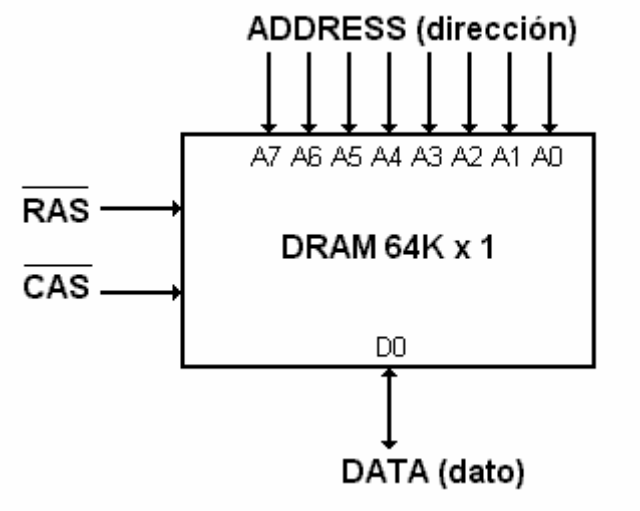


Diagrama de bloque del chip DRAM.

Internamente, las celdas de una memoria DRAM se encuentran dispuestas en forma de matriz, rectangular o cuadrada, de manera que cada celda corresponde a la intersección de una fila y una columna. La dirección completa, entonces, se divide en dos: una parte corresponde a una fila *("row")* y la otra a una columna *("column")*.
Una memoria de 1 Megabit de capacidad podría tener, por ej., 1024 filas por 1024 columnas *(1K·1K=1M)*

Tanto las filas como las columnas se encuentran multiplexadas, es decir, se transmiten por las mismas líneas de dirección, pero en diferentes instantes de tiempo. Dos señales externas indican cuándo se transmiten las filas y cuándo las columnas. Estas señales son *CAS# (Column Address Strobe o Selección de Columna) y RAS# (Row Address Strobe o Selección de Fila)*.

El manejo de las señales de CAS# y RAS#, junto con el refresco de la memoria dinámica, es realizado por un circuito denominado *Controlador de Memoria Dinámica,* que se sitúa entre la CPU y la memoria principal, de manera que ésta última sea más fácil de acceder por la CPU. El controlador de memoria es parte del conjunto de chips *(chipset)* de la motherboard.

Para realizar una operación de lectura, el controlador envía la dirección de fila, luego baja la señal RAS# (que pasa de 1 a 0), en ese momento la dirección de fila es retenida en el interior del chip de memoria. Luego envía la dirección de columna, y baja la señal CAS# para retener dicha dirección. En base a la información de fila y columna, el chip puede ahora responder al pedido de lectura.

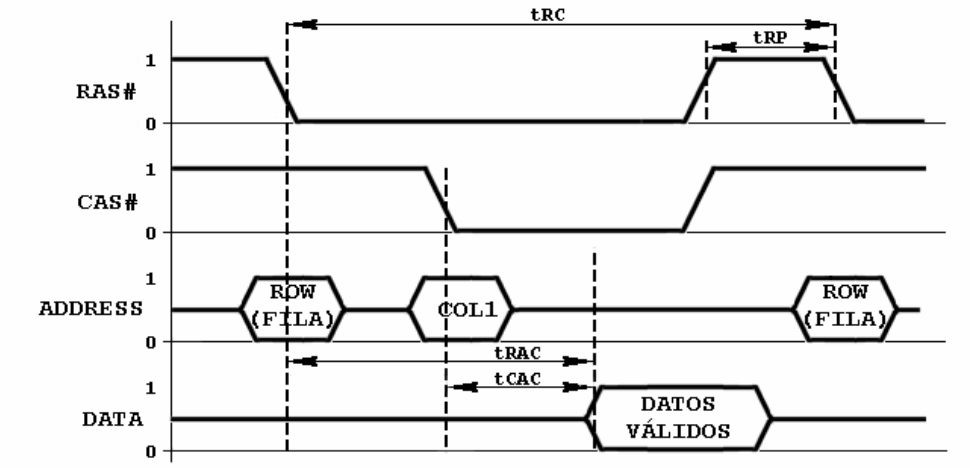


Diagrama de tiempos de una memoria DRAM convencional

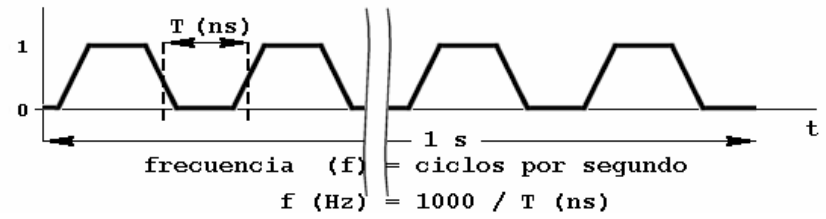
t_{RC} = *Random Read Cycle Time*
t_{CAC} = *Access Time for CAS#*
t_{RAC} = *Access Time for RAS#*
t_{RP} = *RAS Precharge Time*

En el diagrama podemos observar tiempos de acceso y tiempos de ciclo. Los tiempos de acceso se definen en relación al tiempo máximo que se demora en responder a un determinado evento. En una memoria dinámica convencional, el tiempo de acceso es el tiempo que pasa desde que se baja la señal de Selección de Fila (RAS#) hasta que la memoria entrega datos válidos en el Bus de Datos. El tiempo de ciclo es el tiempo requerido para realizar una operación más un tiempo de recuperación necesario para comenzar la siguiente.

t_{RAC} es el tiempo de acceso aleatorio, el tiempo requerido para leer una celda de memoria. Es el máximo permitido para, luego de que baja RAS#, seleccionar una dirección de fila y de columna, sensar la carga de la celda seleccionada y enviar la información a la salida.

t_{RC} es el tiempo asociado con un ciclo de acceso aleatorio. El t_{RC} mínimo es la suma del tiempo en que RAS# debe permanecer baja hasta leer una posición de memoria más el tiempo mínimo necesario para el refresco de la fila (llamado tiempo de precarga)

La velocidad y el rendimiento de la memoria pueden ser confusos, debido a que la velocidad de la memoria se expresa, frecuentemente, en nanosegundos, y la del procesador, en Megahertz. Un nanosegundo es un milmillonésimo de segundo, mientras que 1 MHz es un millón de ciclos por segundo. A medida que aumenta la velocidad expresada en MHz, disminuye proporcionalmente el tiempo de duración de cada ciclo expresado en ns, según la siguiente fórmula: ns=1000/MHz



Según estos cálculos, una memoria de 60 ns (16 MHz) resulta totalmente inadecuada si se la compara con las velocidades de procesador de 400 MHz y mayores. En los procesadores hasta 600 MHz, el bus de memoria (también llamado FSB: *Front Side Bus*) funciona a una velocidad mucho menor, de 66 MHz, lo que equivale a un tiempo de 15 ns por ciclo. Para una memoria de 60 ns o más, eso significa que el acceso a un dato lleva alrededor de 5 ciclos (cada uno de 15 ns). Estos ciclos se denominan estados de espera.

Los estados de espera requeridos para realizar los accesos a memoria se simbolizan x-y-y-y, donde x es el tiempo del primer acceso, e y representa el número de ciclos requeridos por cada acceso consecutivo. La DRAM convencional, en un bus de 66 MHz corre a 5-5-5-5 (5 ciclos para el primer acceso y 5 para los siguientes).

Memoria FPM DRAM (Fast Page Mode DRAM)

Los sistemas han evolucionado para mejorar las velocidades de acceso a la DRAM. Un cambio importante fue la instrumentación de los accesos en modo ráfaga. Los ciclos en modo ráfaga aprovechan la naturaleza consecutiva de la mayoría de los accesos a memoria. Las memorias FPM presentan un tiempo de acceso menor si el acceso se hace sobre la misma fila (una fila completa se denomina página). Ese tiempo menor se consigue evitando enviar nuevamente la dirección de fila, si ésta no varía en la siguiente lectura. De esta manera el tiempo de acceso a datos consecutivos disminuye.

La denominación de Fast o rápida es debido al menor tiempo de acceso en relación a las DRAM convencionales; en la actualidad estas memorias son las más lentas que se consiguen. El tiempo de acceso suministrado por el fabricante es el menor, es decir, el correspondiente a los datos secuenciales y no el tiempo inicial, que sigue siendo el mismo.

Después de especificar las direcciones de fila y columna, es posible acceder a las tres direcciones siguientes adyacentes sin necesidad de los estados de espera necesarios para especificar la dirección de fila, debido a que ésta ya fue retenida en el primer acceso.

Una DRAM FPM estándar de 60 ns normalmente corre a 5-3-3-3 (x333).

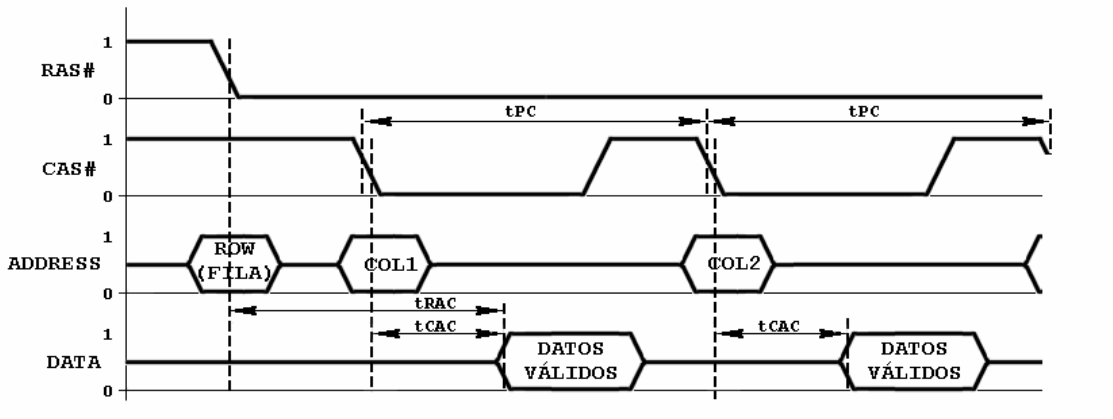
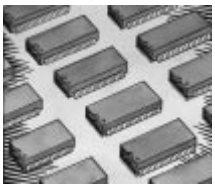


Diagrama de tiempos de una memoria FPM DRAM

Módulos de memoria SIMM

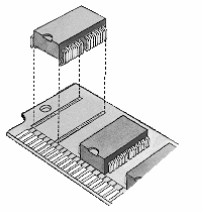
Las memorias FPM son las más antiguas en formato SIMM. Se suministran en 30-pin SIMM o 72 pin-SIMM, siendo el primer formato el más empleado.

En los comienzos, las computadoras poseían un lugar fijo para la memoria a instalar y generalmente la cantidad de memoria RAM era invariable. La memoria se instalaba en los sistemas chip por chip. El encapsulado era conocido como DIP *(Dual Inline Package)*. La PC original tenía 36 zócalos en la tarjeta madre (que alojaba CPU, memoria y los chips de soporte o chipset) agrupados en 4 bloques de 9 chips cada uno *(32 chips tipo 4164 = 4 x 8 x 64 K x 1 bit = 256 KB).* Además podían instalarse más de ellos en tarjetas de memoria adicionales.



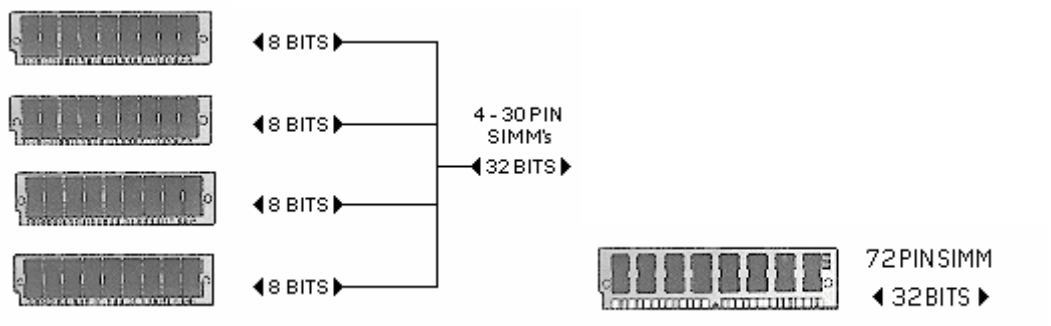
Además de presentar una instalación laboriosa, los chips DIP, en general, presentan un problema serio: con el tiempo, se desplazan de sus zócalos a raíz de los ciclos térmicos. Todos los días, al encender los equipos, éstos se calientan y después se enfrían y las contracciones y dilataciones hace que los chips vayan saliéndose de sus zócalos. A la larga se pierde el contacto y se presentan errores de memoria. Afortunadamente, basta asentar los chips para solucionar el problema, pero este método es bastante laborioso. La alternativa para esto, es soldar la memoria a la tarjeta madre. Sin embargo, si un chip se descompone es necesario desoldar el chip defectuoso y soldar uno nuevo. Además, no soluciona los problemas de actualización, pues con la utilización del mismo modelo de computadora para distintos requerimientos, se hizo necesario que la cantidad de memoria a colocar sea variable para cada equipo dependiendo de la aplicación y de las exigencias de los programas instalados.

Se necesita, entonces, que el chip esté soldado pero que pueda desmontarse fácilmente. Por ello se fijó un estándar para colocar los chips de memoria en pequeños módulos o placas, de manera que éstos sean fácilmente desmontables e intercambiables. Estos formatos de conectores ofrecen un método flexible para actualizar la memoria al mismo tiempo que ocupa menos espacio en la placa del sistema debido a su ubicación vertical.



A partir de las memorias FPM surge el módulo de memoria *SIMM (Single In-line Memory Module)*. El SIMM es tratado como si fuera un enorme chip de memoria. En caso de falla o de actualización, se debe quitar y sustituir todo el módulo.

Los módulos SIMM son módulos de memoria con una sola hilera de contactos eléctricos. Estos módulos, actualmente, están siendo reemplazados por los DIMM. Existen dos variantes de módulos SIMM:



*a) 30 contactos (**8 bits** más 1 bit opcional de paridad) 30 pin SIMM*
*b) 72 contactos (**32 bits** más 4 bits opcionales de paridad) 72 pin SIMM*

Los SIMM de 30 pines son más pequeños que los de 72. Las conexiones en los SIMM son iguales de cada lado.

Existen módulos, anteriores a los SIMM, llamados SIPP (Single Inline Pin Package), que son SIMMs de 30 contactos, pero con terminales o pines, para ser instalado en un zócalo estándar de circuito integrado. Los SIPP son menos confiables que los SIMM, mecánicamente. Puede convertirse un SIMM en un SIPP soldando los pines a un módulo en forma directa (o pasar de un SIPP a un SIMM desoldándolos)

Los SIMMs de 72 pines emplean un conjunto de 5 pines para indicar el tipo de memoria a la tarjeta madre. Estos pines se llaman de Detección de Presencia y están conectados a tierra o no conectados para indicar el tipo de SIMM a la tarjeta madre (desde 1MB, 100 ns a 32MB 50ns).

Memoria EDO DRAM (Extended Data Out DRAM)

Desde 1995 *(Motherboard con chipset Tritón FX)*, se puso a disponibilidad de las computadoras con Pentium un nuevo tipo de memoria conocida como EDO *(Extended Data Out o Salida Extendida de Datos)* La EDO es una forma modificada de memoria FPM, también se conoce como Modo de Hiperpágina.

La memoria EDO presenta registros que memorizan la salida de datos y evitan que éstos desaparezcan cuando el controlador de memoria quita la dirección de columna para enviar la siguiente. Esto permite que el siguiente ciclo se solape con el previo, lo cual ahorra 10 ns por ciclo (en otras palabras, permite introducir un nuevo dato mientras los anteriores están saliendo). En estas memorias, un registro adicional retiene los datos desde que éstos son válidos hasta que son leídos por la CPU y de esta manera, se extiende la salida de datos en el tiempo, mientras se hace la siguiente lectura, lo que da origen a su nombre.

La RAM EDO permite una serie de ciclos de 5-2-2-2 (x222) en modo ráfaga. Para efectuar cuatro transferencias de memoria, la EDO requeriría 5+2+2+2 = 11 ciclos

contra 5+3+3+3 = 14 ciclos de la FPM y 5+5+5+5=20 ciclos de la memoria convencional. Esto implica una mejora del 22% en duración general de ciclos, aunque en promedio la velocidad del sistema se incrementa en un 5%.

Aunque parezca pequeña la mejora general, el aspecto más importante del uso de EDO es que emplea el mismo diseño básico que la FPM, lo cual implica que no hay un costo agregado con relación a este último tipo de memoria.

La RAM EDO es ideal para sistemas con velocidades de bus de hasta 66 MHz (hasta 1997). Sin embargo, desde 1998, el mercado para las memorias EDO ha declinado rápidamente en presencia de las SDRAM, más rápida.

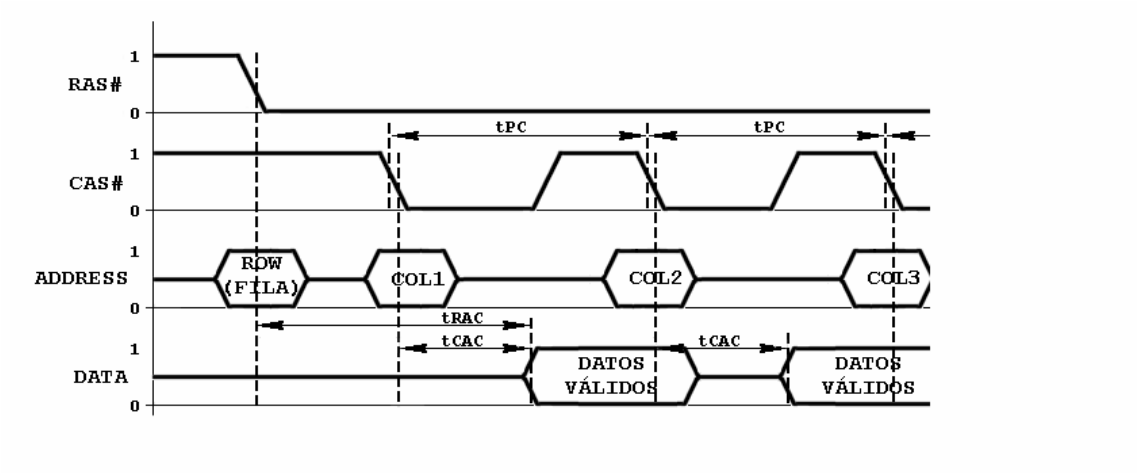


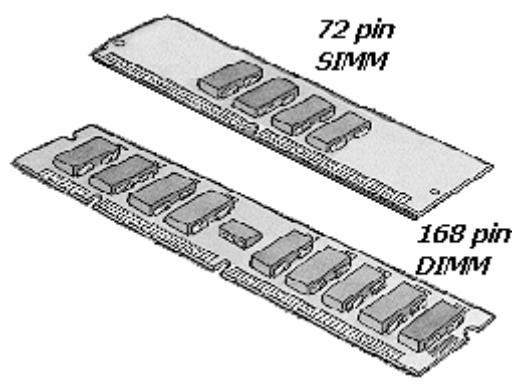
Diagrama de tiempos de una memoria EDO DRAM

Puede verse que con EDO, la elevación de CAS# no termina el acceso de datos como con FPM.

Memoria BEDO DRAM (Burst EDO DRAM)

El funcionamiento de este tipo de memorias es similar a la EDO. La única diferencia es que puede leer datos en ráfagas (burst) permitiendo accesos secuenciales muy rápidos, mediante la implementación de un contador interno. Una vez especificados la dirección de fila y columnas iniciales, la memoria Burst EDO no necesita del envío de la siguiente columna a acceder, si ésta es consecutiva. Un contador interno se incrementa y apunta a la siguiente columna con cada nueva lectura. La mejora se produce solamente en accesos de direcciones consecutivas, y el tiempo de acceso puede ser tan bajo como 10 ns. Su aparición coincidió con la SDRAM, que incluía la transferencia a ráfagas y la superaba en velocidad, por lo cual no se utilizó durante mucho tiempo. Sólo la soportaron las placas 440FX Natoma.

Módulos de memoria DIMM



A partir de las EDO y SDRAM, se comenzaron a utilizar módulos de memoria con doble hilera de terminales, llamados DIMM (Dual Inline Memory Module) con 168 contactos y un ancho de **64 bits** (el doble del correspondiente al SIMM de 72 pines). El DIMM usa un tipo completamente diferente de detección de presencia llamado SPD *(Serial Presence Detect)*. Consta de un pequeño chip EEPROM de 8 terminales, que contiene los datos de la memoria. Estos datos pueden ser leídos mediante dos pines (viajan en serie, un bit detrás de otro) y permite a la máquina configurar automáticamente la velocidad del DIMM instalado.

Bancos de memoria

Los chips de memoria (DIPs, SIMMs, SIPPs y DIMMs) están organizados en bancos colocados en la tarjeta madre. Para agregar memoria al sistema, necesitamos conocer la disposición de los bancos. Un banco es la cantidad menor de memoria que puede ser direccionada por el procesador de una sola vez, y generalmente corresponde al ancho del bus de datos del procesador. Antes de actualizar la memoria de un sistema, se debe conocer la disposición de los bancos.

Procesador	Bus de Datos	30-pin SIMM por banco	72-pin SIMM por banco	168-pin SIMM por banco
286	16 bits	2	-	-
386SX,SL,SLC	16 bits	2	-	-
486SLC,SLC2	16 bits	2	-	-
386DX	32 bits	4	1	-
486SX, DX,DX2,DX4,5x86	32 bits	4	1	-
Pentium,K6	64 bits	8 *	2	1
Pentium Pro, Pentium II, Celeron, Pentium III, Athlon	64 bits	8 *	2	1

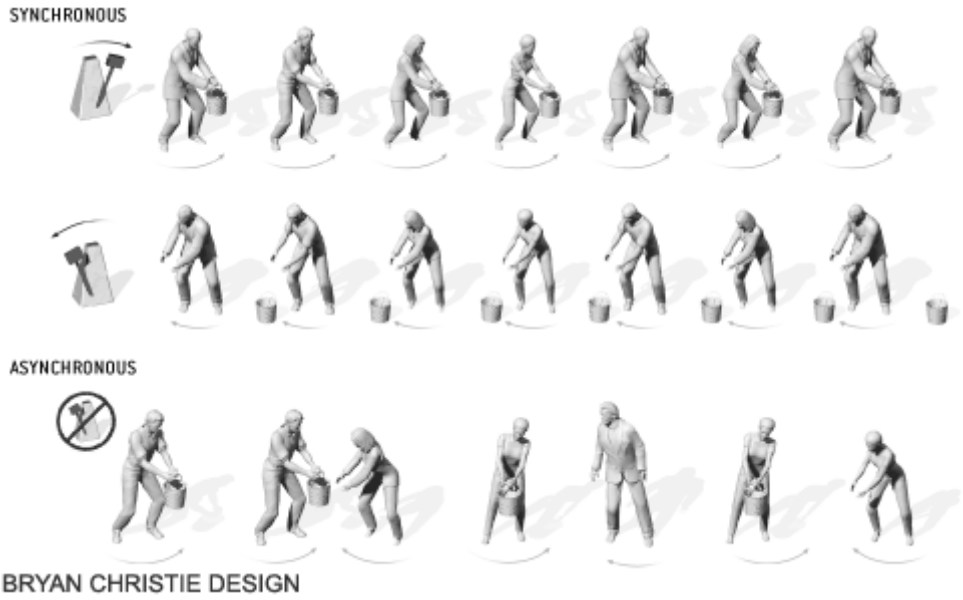
Todos los SIMMs de un mismo banco deben tener el mismo tamaño y el mismo tiempo de acceso y ser, en la medida de lo posible, del mismo tipo.

Una técnica que acelera la memoria FPM es conocida como memoria intercalada. En este diseño, dos bancos de memoria son usados como un solo conjunto, alternando el acceso: un banco se usa para almacenar los datos pares y otro para los impares. Mientras se accede a uno, el otro está siendo cargado, cuando se están seleccionando las direcciones de fila y de columna. Cuando el primer banco (par)

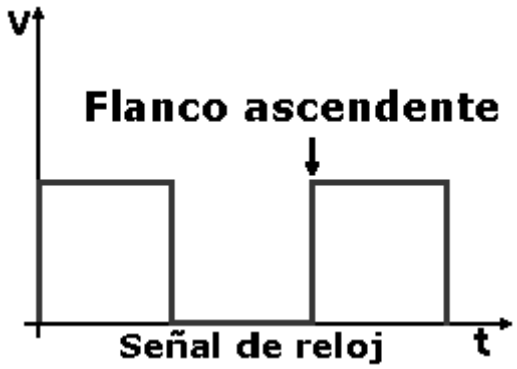
terminó de regresar datos, el segundo banco ha terminado el tiempo de latencia y está listo para devolver datos. Este solapamiento de bancos permite una recuperación más rápida. El único problema es que para usar el intercalado, se deben instalar dos bancos idénticos, lo cual duplica la cantidad de SIMMs requeridos.

Memoria SDRAM (Synchronous DRAM)

SDRAM son las siglas de DRAM sincrónica, un tipo de DRAM que funciona en sincronización con el bus de memoria. Hasta el momento, las memorias que hemos visto son dispositivos asincrónicos: cada módulo interno cambia de estado reaccionando al cambio de estado de un módulo anterior. Los tiempos involucrados constituyen la suma de los retardos de cada una de las etapas (el procesador debe esperar a que ocurran esos cambios). Los sistemas sincrónicos, en cambio, actúan en los instantes en que reciben un impulso común procedentes de un generador. Dicho generador de impulsos se llama reloj.



El cambio se suele producir en un flanco, es decir, en el momento en que se cambia de nivel: de subida o ascendente (cero a uno) o de bajada o descendente (uno a cero).



Todo el control interno de la memoria SDRAM está sincronizado con el bus de memoria, compartiendo el mismo reloj. El controlador de memoria ahora envía comandos en el flanco ascendente de reloj.

Mientras las DRAM FPM y EDO estaban diseñadas para permitir leer datos en ráfaga manteniendo una columna activa y seleccionando sólo columnas, SDRAM da un paso más asumiendo que se van a acceder a datos en ráfagas en secuencias predefinidas. Por ej: se puede programar una SDRAM de manera que cada vez que se le envíe una dirección de fila y de columna, automáticamente ésta entregue una secuencia de ocho columnas en el siguiente orden: 5-6-7-0-1-2-3-4 (dónde cero es la columna que se envía) ó 0-1-2-3 o toda una página entera. Para ello incorpora un registro interno, llamado *Registro de Modo* donde el controlador de memoria define la secuencia de la ráfaga. Una vez definida, sólo es necesario enviar fila y columna.

Además, para poder aprovechar las ventajas del intercalado, las memorias SDRAM tienen dos bancos internos, seleccionables mediante un pin del módulo DIMM. De esa manera se puede tener un banco ocupado con la precarga (refresco), mientras se está accediendo al otro banco.

El controlador de memoria se comunica con la SDRAM mediante comandos. Los comandos tienen funcionalidades similares a la activación individual de las líneas de control en las memorias asincrónicas vistas anteriormente (FPM y EDO)

Comandos:
`NOP` (No Operación)
`ACTIVATE` (Activa la fila, colocando la dirección de la fila en el bus de direcciones)
`READ` (Lee la columna de la fila activada colocando la dirección de columna en el bus)
`WRITE` (Escribe)
`LOAD MODE REGISTER` (Carga el Registro de Modo)

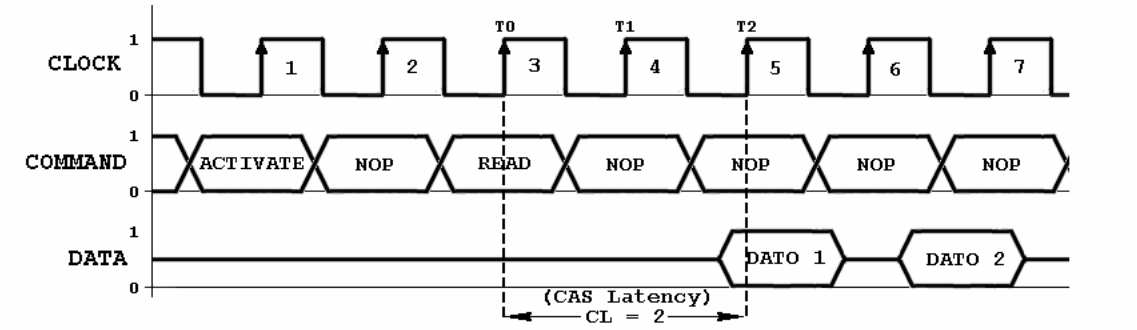


Diagrama de tiempos de una memoria SDRAM

En modo lectura, la SDRAM presenta los siguientes tiempos significativos:
Pulso 1: Se activa la fila *(Activate)*, a través de RAS#. Cuando se hace esto se coloca la dirección de la fila a activar.
Pulso 3: Se lee la columna *(Read)* seleccionada (CAS#)
Pulso 5 - 10: El dato de fila y columna sale por los pines de datos, seguidos por una secuencia de otras columnas, en el orden en que se dieron en la carga del registro de modo *(Mode Register)*.

La latencia de CAS# es el retardo, en ciclos de reloj, entre un comando de lectura *(Read)* y la disponibilidad del dato a la salida. La latencia puede estar entre 1 y 3 ciclos de reloj. La latencia CAS# es un tiempo de espera, por lo tanto, cuanto menor sea, mejor es la memoria (para una misma frecuencia de trabajo).

Las memorias SDRAM suelen marcarse con tres números a-b-c, siendo cada uno, respectivamente, CL *(CAS Latency)*, t_{RCD} *(RAS to CAS Delay)* y t_{RP} *(Row Precharge Time)*. Debido a que se trata de retardos, cuánto menores sean, más rápida es la memoria *(Ver Estándar PC100)*.

Estándares PC100 y PC133

Las velocidades de la memoria de las PCs varían de alrededor de 10 a 200 ns. Cuando se sustituye un módulo de memoria, debe instalarse uno del mismo tipo y la misma velocidad. Puede reemplazar un chip con otro de una velocidad diferente, sólo si la velocidad del nuevo es mayor.

Si bien las primeras computadoras IBM PC estaban limitadas a 10 MHz y el tiempo de acceso de las memorias DRAM cumplían con estos requerimientos, hacia finales del año 1997, el procesador funcionaba internamente a más de 200 MHz, mientras que externamente el bus de memoria permanecía en 66 MHz. A comienzos del año 1998, Intel introdujo la primer placa madre con un bus de memoria de 100 MHz *(440BX)*. Esto aumentó el rendimiento para la generación de Pentium II de 333 MHz en adelante.
Para ocuparse más detalladamente de la velocidad y de la confiabilidad, Intel creó, junto con la introducción del BX, un estándar para las memorias SDRAM de alta velocidad de 100 MHz, llamado PC100 (la memoria usada entonces era SDRAM, que funcionaba en forma sincrónica con el reloj del procesador, por lo que a partir de ellas los tiempos pudieron especificarse en ciclos y la velocidad en MHz).

Los estándares tienen como fin posibilitar la certificación de módulos de memoria como dispositivos capaces de ofrecer el rendimiento y la velocidad requeridos por las placas madre de Intel. Las especificaciones establecen parámetros mecánicos del módulo, anchos mínimos y máximos de pistas, número de capas, tiempos de acceso de chips, etc. Por ej., si bien 10 ns sería considerada una velocidad apropiada para operación a 100 MHz, la especificación estipula memoria más rápida, 8 ns, para asegurar el cumplimiento de todos los parámetros de temporización.

Aunque técnicamente no existe la especificación PC66 para memoria, cualquiera que no cumpla completamente con el estándar PC100, pero que funcione a 66 MHz, se considera PC66.

Otra organización, JEDEC (Asociación de Tecnología de Estado Sólido) que crea la mayor parte de las especificaciones actuales de memoria, emitió en 1999 la especificación PC133, con parámetros de temporización mucho más rígidos.

Temporización	Velocidad real	Velocidad estipulada
15 ns	66 MHz	PC66
10 ns	100 MHz	PC66
8 ns	125 MHz	PC100
7.5 ns	133 MHz	PC133

El tiempo de acceso de la memoria se puede reconocer, en la práctica, observando los últimos caracteres que identifican a los chips de memoria. Frecuentemente se coloca un número, luego de un guión, que expresa el tiempo en nanosegundos o en décimas de nanosegundo.

Ejemplos:

-6 expresa tiempo de acceso 60 nanosegundos
-7 expresa tiempo de acceso 70 nanosegundos
-70 expresa tiempo de acceso 70 nanosegundos
-07 expresa tiempo de acceso 7 nanosegundos

Nomenclatura de módulos PC100 y PC133

PC100-abc-defm (ej:PC100-322-622R) ó ***PC100-abc-ddeefm (ej:PC100-322-54122R)***

a=CAS Latency (ciclos)
b=RAS to CAS delay t_{RCD} (ciclos)
c=Precarga t_{RP} (ciclos)
d=Tiempo de Acceso a Lectura t_{AC} (ns)
e=Versión del Chip SPD (2 corresponde a 1.2)
f=Versión de Diseño (2 corresponde a 1.2)
m=Tipo de módulo R: Registered (amplificado) U: Unbuffered (sin amplificar)

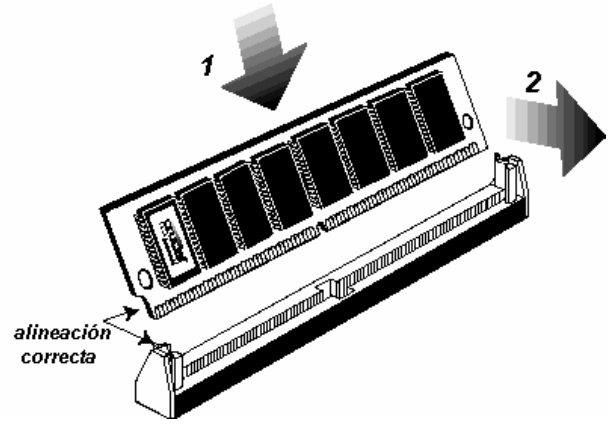
PC133m-abc-dde (ej: PC133U-222-452, PC133R-333-542)

a,b,c,m: Ídem PC100
dd=Tiempo de Acceso a Lectura (ns)
e=Versión del Chip SPD (2 corresponde a 2.0)

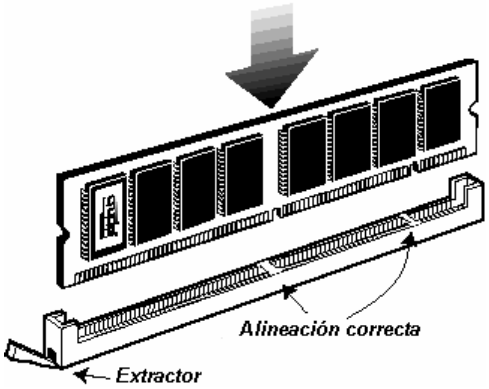
Ej: Texas Instruments 128 MB SDRAM PC133U-333-542 (CL3 upto 133 MHz): 133 MHz, Unbuffered, CL=3, t_{RCD}=3, t_{RP}=3, t_A=5.4 ns, Versión SPD=2.0)

Instalación de módulos de memoria

Al instalar módulos de tipo SIMM, coloque el módulo en un ángulo de 45 grados (1) y luego empuje levemente hasta que éste quede en posición vertical (2).



Los módulos de tipo DIMM se deben instalar directamente presionando en posición vertical. Tenga en cuenta que el módulo tiene muescas que deben coincidir con las del zócalo.



Nota: Las PC con ranuras SIMM y DIMM, generalmente Pentium MMX, poseen unos jumpers de color rojo para elegir el voltaje de trabajo de la memoria que suele ser de 3,3 o 5 voltios. Un error en la conexión podría deteriorar las memorias.

Errores de memoria. Paridad y ECC

Las memorias RAM dinámicas o DRAM, a causa de su naturaleza, pueden presentar errores de lectura ocasionales. Como almacenan unos y ceros en forma de cargas que deben ser refrescadas periódicamente pueden ser afectadas con facilidad.

Las memorias presentan dos tipos de errores:

- *ERRORES DE HARDWARE O REPETITIVOS (HARD ERRORS):* Son errores provocados por chips de memoria dañados.
- *ERRORES TRANSITORIOS (SOFT ERRORS):* Estos errores son los más frecuentes y dificiles de diagnosticar. Provocados por pérdidas temporales de carga.

Los errores transitorios pueden ser provocados por varios factores:

a) Partículas alfa irradiadas por el encapsulado del chip de memoria (Torio y Uranio) de baja energía
b) Rayos cósmicos (de alta energía)
c) Ruido en la alimentación eléctrica
d) Clasificación errónea del tipo o la velocidad

Además de las ya vistas, una fuente importante de errores de memoria son los contactos eléctricos. Los módulos están disponibles en versiones con contactos de oro o estañados (plateados).

Para contar con un sistema confiable, se deben instalar SIMMs o DIMMs con contactos estañados en zócalos estañados (plateados), de lo contrario, entre 6 meses y 1 año, se producen errores de memoria. En la actualidad, Intel y AMP no recomiendan mezclar metales en el sistema de memoria, de manera de tener el mismo material de contacto. La mejor configuración es oro con oro.

Cuando el estaño y el oro se ponen en contacto, la oxidación se va acumulando y no cede bajo presión como lo haría si ambas superficies fueran estañadas. La resistencia eléctrica va aumentando y esto produce con el tiempo fallas de memoria. El óxido de estaño es muy duro y a veces su eliminación requiere de un tratamiento abrasivo (se

suele utilizar una goma de borrar, aunque es recomendable rociar antes el módulo con líquido limpiacontactos)

¿Cómo enfrentar los errores de memoria? La mejor forma es incrementar la tolerancia del sistema a los errores. Existen tres técnicas de tolerancia a errores en una PC moderna:

1) Sin paridad
2) Paridad
3) ECC

Sin Tolerancia a Fallos. Memorias Sin Paridad

Las memorias sin paridad no tienen tolerancia a errores. La única razón de su uso es el bajo costo (no se requieren chips de memoria adicionales). Como la frecuencia de los errores depende de la densidad de la memoria, cuánto más memoria tenga un sistema, mayor es la probabilidad de fallas y por lo tanto el riesgo.

Detección de Error. Memorias con Paridad

Un estándar creado por IBM para la industria es la detección de errores mediante la utilización de 1 bit de control por cada 8 bits de datos almacenados. El bit de control incorporado contiene un dato que caracteriza a los otros ocho, lo que permite mantener la integridad del sistema. Las memorias con paridad añaden 1 bit extra por cada 8 bits de datos, ese bit extra se utiliza solamente para la detección del error.

Módulo	*Líneas de Datos (Sin Paridad)*	*Líneas de Datos (Con Paridad)*
30 pin SIMM	8 bits	9 bits
72 pin SIMM	32 bits	36 bits
168 pin DIMM	64 bits	72 bits

Proceso de detección de error. Paridad impar.

1) El bit de paridad se coloca en 0 si los bits de datos contienen un número impar de unos y se coloca en 1 en caso contrario. (Ej: Si el dato es 01001000, contiene un número par de unos, y el controlador de memoria coloca el bit de paridad en 1).
2) El bit de paridad se escribe a DRAM, junto con el dato.
3) Se lee la memoria. Antes de que el dato sea enviado a la CPU, es interceptado por el controlador de memoria. Si contiene un número par de unos, el dato es considerado inválido y se se dispara una interrupción no enmascarable (no evitable) provocando la detención del sistema.

NOTA: Los módulos de memoria también pueden usar paridad par. El sistema más común es el de paridad impar.

Corrección de Error. Memorias con ECC (Error Correction Code)

Las memorias con ECC permiten detectar y corregir automáticamente errores de 1 bit, utilizando un código de corrección de errores, tal como el código de Hamming. También pueden detectar (pero no corregir) errores de 2,3 ó 4 bits. Para ello añaden solamente 1 bit por cada 8, y evitan la detención del sistema, lo que ocurriría en el caso de memorias con paridad.

Cómo el 98% de los errores suelen ser errores de un bit, el sistema presenta un alto rendimiento. Es posible detectar errores de un bit, pero no corregirlos. El sistema se considera como SEC-DED (Single Error Correction - Double Error Detection)

El ECC hace que el controlador de memoria calcule los bits de verificación en una operación de escritura a memoria, efectúe la comparación entre los bits de verificación leídos y los calculados en una operación de lectura, y, si es necesario, corrija los bits erróneos. El ECC afecta el rendimiento de la memoria en escritura, debido a que se debe esperar el cálculo de los bits de comprobación, y, cuando el sistema espera datos corregidos, su lectura.

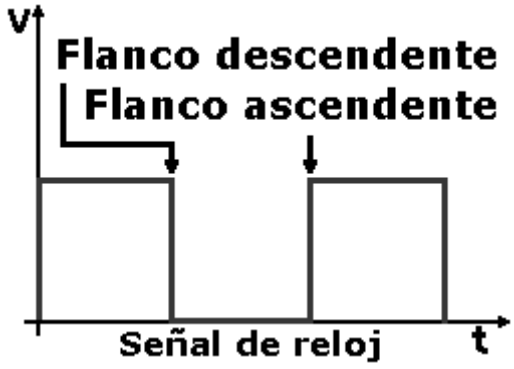
La mayoría de los errores de memoria son de un solo bit, y pueden ser corregidos por el ECC. Un equipo con ECC es una buena elección para servidores, estaciones de trabajo u otras aplicaciones críticas. Tienen un costo de un 30% mayor a las memorias sin paridad.

Sin Tolerancia a Fallos. Paridad Lógica

Hace algunos años, cuando la memoria era más cara, unas cuantas compañías estuvieron vendiendo SIMMs con chips de "paridad lógica" o "emulación de paridad". Este chip ignoraba cualquier paridad que el sistema estuviera tratando de almacenar en el SIMM, pero cuando se recuperaba información, siempre aseguraba que la paridad correcta fuese devuelta, haciendo creer al sistema que todo estaba bien, mediante un generador de paridad. El generador de paridad no parece un chip de memoria, y tiene un rótulo diferente al resto de los chips del módulo. Es evidente que el sistema no tiene ninguna utilidad como detector o corrector de errores, pues siempre entrega paridad correcta, aún en caso de errores de memoria.

Memoria DDR SDRAM (Double Data Rate SDRAM)

DDR SDRAM es un avance sobre la tecnología SDRAM, producido hacia finales de 1999. Si consultamos los diagramas de tiempos de la SDRAM convencional, vemos que la transferencia de comandos, direcciones y datos se hace en el flanco ascendente de reloj. Como la SDRAM normal, la DDR transfiere comandos, direcciones y datos en el flanco ascendente, pero a diferencia de ésta, lo hace también en el flanco descendente de reloj. Por lo tanto, DDR puede transferir dos palabras de datos por ciclo de reloj, doblando la velocidad de lectura o escritura en circunstancias óptimas.



DDR significa "Double Data Rate", un nombre que describe su habilidad para transferir el doble de datos que una SDRAM convencional. Para implementar un sistema que utilice ambos flancos de reloj, se necesita una circuitería extra que ocupe lugar en el chip.

Además de incrementar la velocidad de transferencia, la memoria DDR SDRAM utiliza cuatro bancos internos. La ventaja de tener múltiples bancos es que cada uno puede tener ya una fila seleccionada (o activa). Ello evita tener que pasar direcciones de fila y gastar tiempo extra. Manteniendo una fila activa el mayor tiempo posible, se hacen sólo accesos rápidos a columnas y en ráfagas, transfiriéndose un dato de un banco en un flanco ascendente y un dato en el descendente.

Al aumentar la velocidad efectiva de transferencia sin tener que aumentar la velocidad de reloj, el sistema DDR permite aumentar el rendimiento sin los problemas que ocasiona el aumento de frecuencia. Por ello, a partir de este sistema, se considera que, si el bus corre a 100 MHz, la transferencia de datos se hace a 2 veces la frecuencia de reloj: 200 MHz efectivos o equivalentes.

La DDR SDRAM puede funcionar a frecuencias de 100, 133 ó 166 MHz (físicos) con una frecuencia equivalente de 200, 266 o 333 MHz. A cada chip de memoria se lo llama, de acuerdo a su frecuencia de trabajo, DDR 200, DDR 266, DDR 333, DDR 400 ó DDR 533.

Sin embargo, de acuerdo a su máxima velocidad de transferencia, los módulos de memoria DIMM se denominan PC1600, PC2100, PC2700, PC3200 y PC4200. El cálculo de la velocidad de transferencia se realiza de la siguiente manera: Como cada módulo DIMM transfiere 64 bits de datos, es decir, 8 bytes, se multiplican los 8 bytes a transferir por la velocidad efectiva del bus. Ej: Para DDR 200, tenemos 8 x 200 = 1600 Bytes/s, para DDR400, tenemos 8 x 400 = 3200 Bytes/s.

Las memorias trabajan con 2,5 Voltios (en lugar de 3,3 Voltios) y utilizan un empaquetado DIMM de 184 pines. Varios fabricantes han establecido a DDR como un estándar (**http://www.jedec.org**).

Nomenclatura:

PCxm-aabc-dde (ej: PC266R-2533-750 ó PC1600U-2220-81)

x=frecuencia efectiva (la velocidad del bus debe ser la mitad o menos)
m=tipo de módulo (R: Registered, U: Unbuffered)
aa=CAS Latency
b=RAS to CAS Delay
c=Row precharge

dd=Read Data Access Time
e=SPD Chip Revision (0 equivale a SPD 1.0)

RAMBUS DRAM (RDRAM)

La memoria RDRAM rediseña la tecnología de memoria dinámica y cómo se integra dentro del sistema. Se comenzó a utilizar desde finales de 1999.

La celda básica de memoria dinámica se mantiene, pero la forma en que un sistema accede a cada celda es muy diferente. RAMBUS desarrolló un bus de memoria de tipo serie para la comunicación entre chips a velocidades muy altas.

El nuevo sistema intenta resolver algunos problemas remanentes de la tecnología DRAM. Son, entre otros, los siguientes:

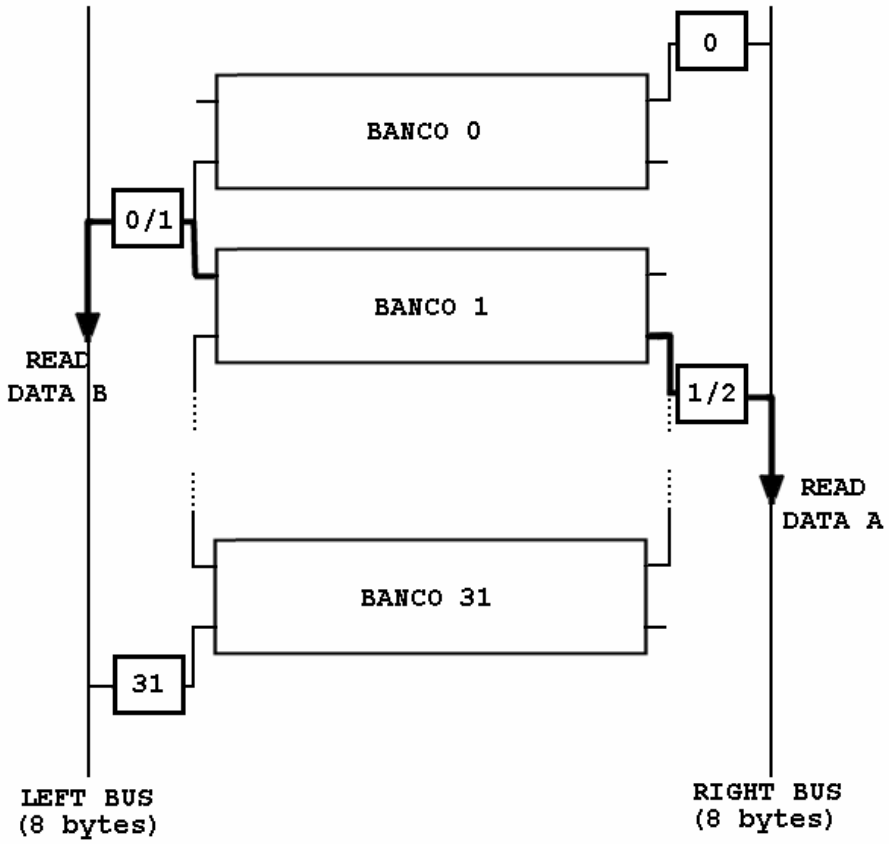
1) Restricción en el número de terminales de los chips
Debido al incremento en el tamaño del bus de memoria y de la capacidad, ha debido incrementarse proporcionalmente el número de terminales de los módulos de memoria, multiplicado por el número de bancos disponibles por módulo. Sin embargo, a medida que el número de pines de un chip de memoria se incrementa, el costo de producirlo, encapsularlo y soldarlo aumenta significativamente, por lo que cada chip SDRAM puede aportar como máximo 16 bits de datos (el número de pines para datos está limitado a 16). Para formar un módulo DIMM SDRAM necesitamos de 4 chips por banco como mínimo (4x16 = 64 bits). (Para un DDR precisamos el doble)

2) Número de bancos limitado
El total de bancos está limitado por el número de chips del módulo DIMM. Generalmente no supera los cuatro.

3) Pobre granularidad
La restricción en el número de pines es causante, entre otros, de que las memorias deban actualizarse mediante grandes incrementos. Se puede actualizar, por ejemplo, de 64 MB a 128 MB, pero no de 64 MB a 70 MB. Una tecnología que permita agregar RAM en pequeños incrementos tiene buena granularidad.

La memoria RDRAM intenta disminuir los costos de fabricación empleando chips con menor cantidad de terminales y a la vez incrementar la velocidad de transferencia. Una de las características más importantes de los chips RDRAM es que pueden contener hasta 32 bancos de memoria, individuales y autocontenidos (es decir, dentro del chip), a diferencia de SDRAM que utiliza varios chips para formar un banco.

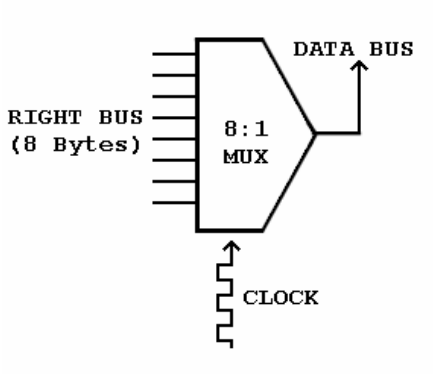
En el diagrama se representa una memoria de 32 MB.



Cada banco almacena 1 MB. En cada uno se aloja una matriz de 512 filas de 128 datos de 16 bytes cada uno (llamados dualocts: Un dualoct es la más pequeña unidad que la memoria RDRAM puede direccionar y corresponde a 16 bytes).
Cuando la CPU lee 16 bytes de un banco, el dato deja el banco y se divide en dos datos de 8 bytes cada uno, A y B.

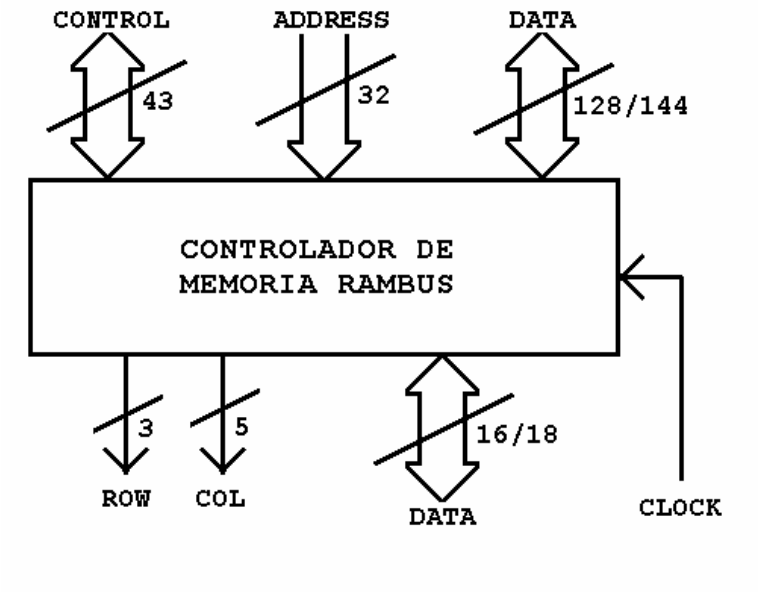
A pesar de poseer 32 bancos, sólo 16 pueden estar activos en un determinado tiempo, la mitad del número de bancos del chip debido a que dos comparten el acceso al mismo amplificador. Cada amplificador incrementa el tamaño y costo del chip, el compartirlos hace menor el costo total.

Cada chip RAMBUS tiene sólo 16 pines de datos, es decir, sólo 2 bytes de ancho (En oposición a los 8 bytes de SDRAM) ¿Cómo provee 16 bytes a partir de 2 bytes de datos? A través de la multiplexión. Usando dos multiplexores (llaves selectoras) entre el núcleo de la memoria con sus buses internos de 8 bytes y el bus externo, los datos se transfieren de 1 byte por vez (En el dibujo se muestra un solo bus, el derecho).

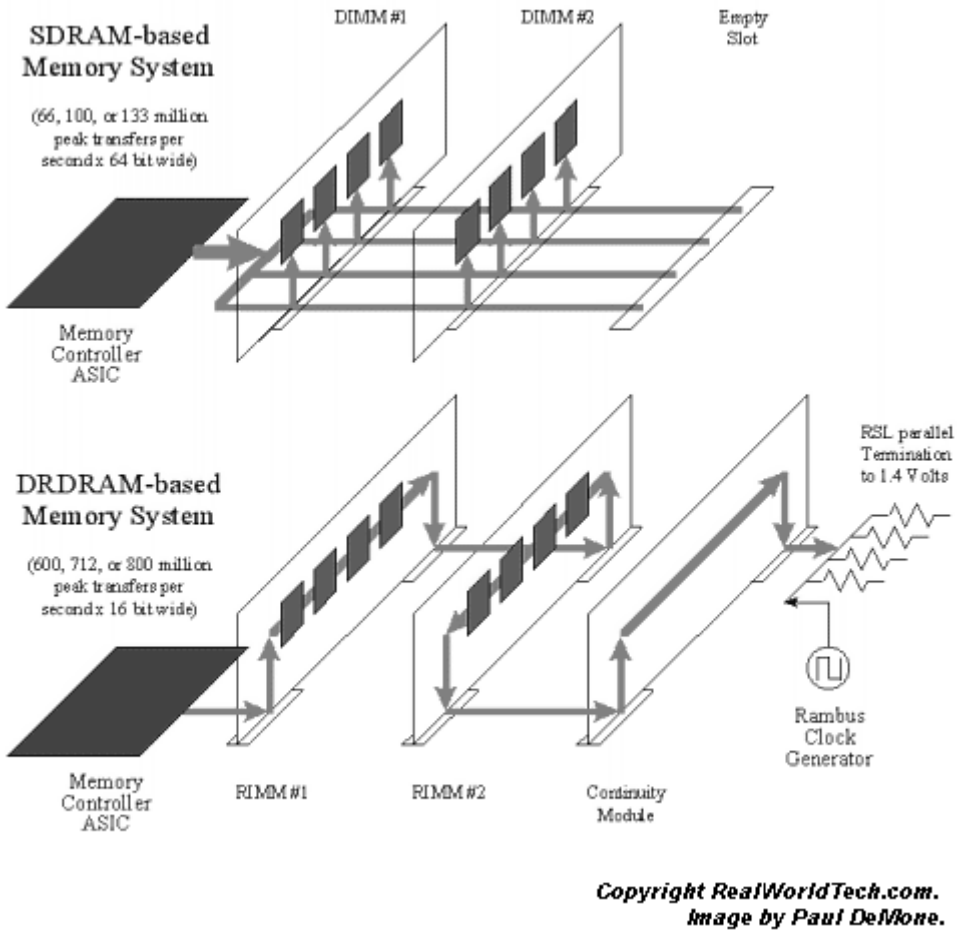


La transferencia de datos se realiza tanto en los flancos ascendentes como descendentes de reloj. Debido a que los buses izquierdo y derecho (Data A y Data B) transfieren datos al mismo tiempo, por cada pulso de reloj se transfieren 4 bytes, y el paquete de datos entero de 16 bytes (1 dualoct) se transmite en 4 ciclos de reloj. En escrituras (write) el proceso es similar, sólo que en lugar de utilizar multiplexores se utilizan dispositivos que realizan el proceso inverso, llamados demultiplexores.

La memoria RDRAM añade complejidad al controlador de memoria, el cual recibe los datos en paralelo (todos juntos en forma simultánea) y debe enviarlos en serie, uno detrás de otro, contando para ello con multiplexores y demultiplexores, como sucede con las memorias.

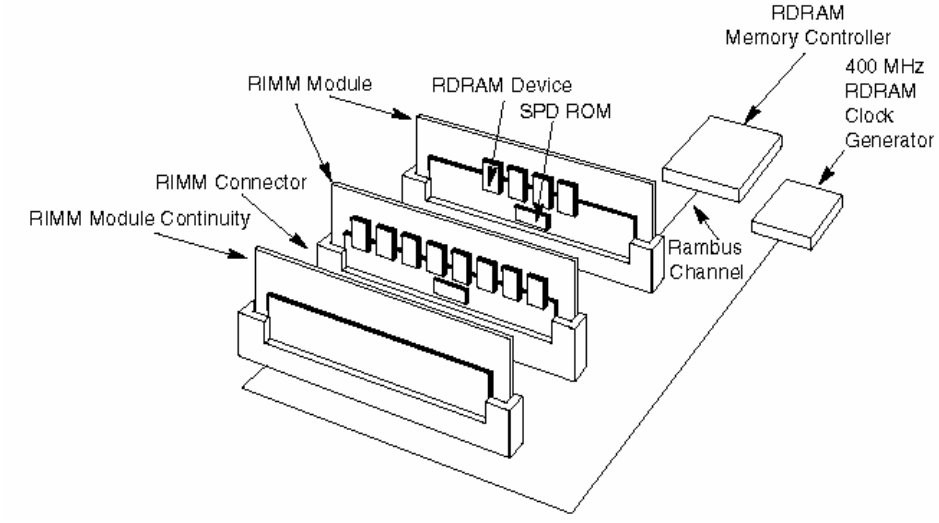


Los bancos de memoria RDRAM se conectan en forma diferente a los SDRAM. Cada chip RDRAM se conecta a un conjunto de 30 líneas de alta calidad llamado Canal RAMBUS que maneja direcciones, datos y señales de control. Cada línea se origina en el controlador de memoria, pasa por cada RDRAM del sistema y termina en una resistencia.



Las memorias RDRAM se sueldan a módulos de memoria llamados RIMM (RAMBUS Inline Memory Module) similares a los DIMM, sólo que cada chip se conecta en serie, de manera que todas las señales de un canal pasen por cada chip. Los zócalos RIMM vacíos deben llenarse con módulos de continuidad (CRIMM) para que el canal termine adecuadamente. Cada RIMM posee 184 pines y dos muescas.

Se pueden tener canales RAMBUS independientes en un sistema, cada uno de los cuales aportaría 1,6 GB/s. Se recomienda instalar el módulo de memoria lo más cerca posible del procesador, para aumentar la velocidad de transferencia.



La RDRAM posee un sistema de control de energía con cuatro modos de consumo: *activo, standby, dormido y apagado*. En el modo activo el consumo es mayor y la respuesta es la más rápida. El modo puede ser seleccionado individualmente, y sólo un dispositivo por canal puede estar en estado activo, el cual disipa todo el calor (a diferencia de las SDRAM, dónde el calor se reparte en cada chip) Por ello, los módulos RIMM requieren de disipadores de calor, lo que los hace más costosos. Aún así, debido a la posibilidad de desactivación y a funcionar con tensiones de sólo 2,5 Voltios y variaciones de 0,8 Voltios (El estado lógico cero es 1V y el uno es 1,8 V) la memoria tiene un consumo menor que las SDRAM.

Nomenclatura:
xMB / a b PCd (ej: 256MB/16 ECC PC800)
x=capacidad del módulo a=Número de chips RDRAM b=ECC d=Velocidad

---

Cuadro comparativo de velocidades

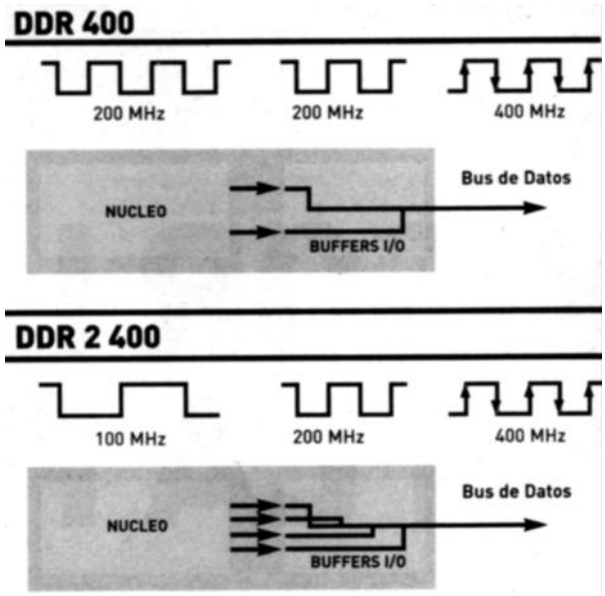
Nomenclatura	Memoria	Vel. Efect. (MHz)	Transfiere	Vel. de transf. (Ancho de Banda)
PC66	SDRAM	66	8 bytes	0,5 GB/s
PC100	SDRAM	100	8 bytes	0,8 GB/s
PC133	SDRAM	133	8 bytes	1,06 GB/s
PC1600	DDR200	200 (100 FSB)	8 bytes	1,6 GB/s
PC2100	DDR266	266 (133 FSB)	8 bytes	2,1 GB/s
PC2700	DDR333	333 (166 FSB)	8 bytes	2,7 GB/s
PC3200	DDR400	400 (200 FSB)	8 bytes	3,2 GB/s
PC4200	DDR533	533 (266 FSB)	8 bytes	4,2 GB/s
PC800	RDRAM Dual	400 (100 FSB)	2 bytes	3,2 GB/s
PC1066	RDRAM Dual	533 (133 FSB)	2 bytes	4,2 GB/s

Ejemplo de módulo de memoria RAM
(Datos obtenidos por medio de la lectura del chip SPD)
[128 MB PC2100 DDR SDRAM]
Tamaño del módulo 128 MB (1 rows, 4 banks)
Tipo de módulo Unbuffered
Tipo de memoria DDR SDRAM
Velocidad de la memoria PC2100 (133 MHz)
Anchura del módulo 64 bit
Voltaje del módulo SSTL 2.5
Método de detección de errores Ninguno
Tasa de refresco Reducido (7.8 us), Self-Refresh
Latencia CAS máxima 2.5 (7.5 ns @ 133 MHz)
Segunda latencia CAS máxima 2.0 (10.0 ns @ 100 MHz)

Memoria DDR 2

Las memorias DDR 2 hicieron su aparición como memorias de vídeo en las tarjetas de vídeo de gama alta, antes que como memoria principal. La tecnología empleada en estas memorias no es compatible con las memorias DDR, no sólo aumenta la velocidad de operación, sino que también disminuye el consumo. Las memorias DDR 2 trabajan con tensiones eléctricas menores a las DDR (1,8 V contra 2,5 V) lo que reduce notablemente el consumo. Utiliza módulos de 240 pines (contra 184 de las DDR convencionales), con lo que los zócalos de memoria no son compatibles entre sí.

Las memorias DDR 2 tienen hasta ocho bancos. Los circuitos internos trabajan con dos velocidades distintas: El acceso a filas y columnas (núcleo) de la memoria se hace a una velocidad, se almacenan en registros (buffers) al doble de velocidad y salen al exterior en ambos flancos de la señal de reloj (DDR) al cuádruple de la velocidad del núcleo. Es decir, una memoria DDR 2 de 400 MHz accedería a sus bancos a una frecuencia de 100 MHz, entregando datos a los buffers de salida a 200 MHz y finalmente pasando estos datos por el bus de memoria en los flancos ascendentes y descendentes de reloj, a 400 MHz efectivos. Una memoria DDR 2 de 400 MHz, por lo tanto, tendría un rendimiento igual o menor a una memoria DDR convencional de la misma frecuencia. Estas memorias están diseñadas para funcionar con velocidades más altas de reloj que las DDR, cercanas al GHz.



Las señales de muy alta frecuencia deben estar adecuadamente terminadas por dispositivos llamados resistores para evitar interferencias y señales reflejadas. En las memorias DDR 2 cada módulo que no esté activo funciona como terminador, evitando que se produzcan interferencias con las frecuencias elevadas que utilizan. Esto se conoce como "On-Die Termination" (ODT).

En forma similar a las memorias DDR, un chip DDR 2 se nombra como DDR2-xxx donde xxx es la velocidad efectiva en MHz. DDR2-400, DDR2-533, DDR2-667 y DDR2-800. También, en relación con el módulo, pueden identificarse como PC2-yyyy donde yyyy es la frecuencia efectiva multiplicada por 8: PC2-3200, PC2-4200, PC2-5300 y PC2-6400.

Módulo	Memorias	Velocidad del módulo	Velocidad del sistema DDR2 con doble canal
PC2-3200	DDR2-400	3,2 GB/s	6,4 GB/s
PC2-4200	DDR2-533	4,2 GB/s	8,4 GB/s
PC2-5300	DDR2-667	5,3 GB/s	10,6 GB/s
PC2-6400	DDR2-800	6,4 GB/s	12,8 GB/s

LA MEMORIA CACHÉ

Introducción

El esquema más básico y sencillo de lo que es un sistema de ordenador consta de una CPU, o unidad central de proceso, un sistema de memoria y una unidad de entrada/salida. Los modernos microprocesadores, procesan una gran cantidad de datos a muy alta velocidad. La memoria es un elemento fundamental en todo ordenador y limita en gran medida la potencia total del sistema. Como vimos, la memoria física de las computadoras está construida con memorias DRAM. Sin embargo, cuando los chips DRAM no son suficientemente rápidos para el microprocesador, éste tiene que incluir períodos de inactividad *(wait states)* en todos los accesos a la memoria.

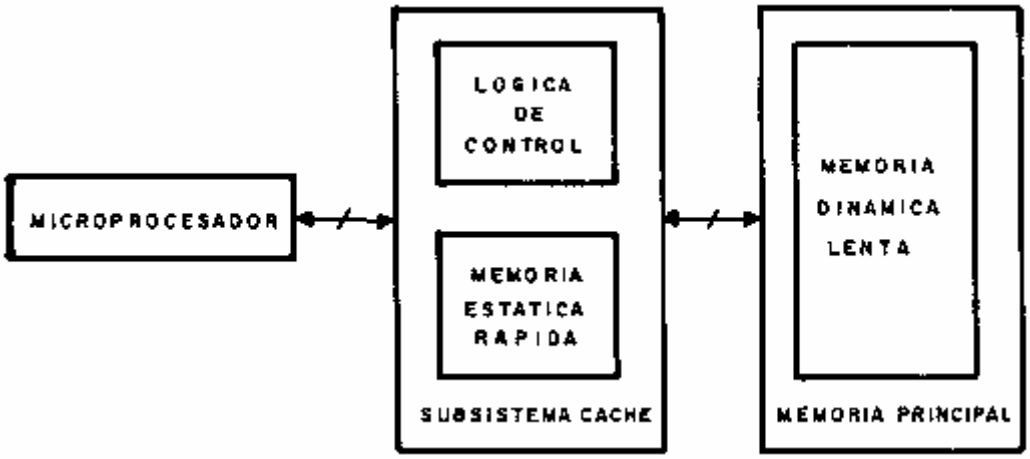
De esta forma, el microprocesador prolonga su tiempo de acceso a la memoria hasta que ésta esté lista para aceptar la escritura de datos en ella, o para proporcionarle los datos que se le han solicitado.

Conceptos generales

Una memoria caché es un subsistema de memoria intermedia, situado entre el microprocesador y la memoria principal, compuesto por memoria estática SRAM de alta velocidad.

Con la memoria caché, la unidad de memoria consta de una memoria principal DRAM, barata y de alta capacidad y una pequeña memoria SRAM de alta velocidad. El sistema se diseña para que el microprocesador pueda operar la mayor parte del tiempo desde la caché, construida con memoria SRAM. Ésta tiene almacenada copias de datos de la memoria principal DRAM que el microprocesador utiliza frecuentemente.

De esta forma, se consiguen las prestaciones de un sistema con memoria rápida estática a un precio ligeramente superior que si toda la memoria fuese realizada con memoria dinámica.



Principio de funcionamiento

La memoria caché consta de tres componentes básicos:

- **La memoria RAM de datos**
- **La memoria RAM de etiquetas**
- **La lógica de control o controlador de la caché.**

En la memoria de datos *(data memory)* se almacenan los datos que son usados más frecuentemente por el microprocesador.

En la memoria de etiquetas *(tag memory)*, se almacenan las direcciones de memoria de los datos que están almacenados en la caché.

La lógica de control *(cache controller)* consulta las direcciones almacenadas en las etiquetas, para saber si un dato está en la caché o no. Compara la dirección de memoria recibida con las direcciones almacenadas en la memoria de etiquetas, y deduce si el dato buscado está en la caché o no.

Cada acceso a memoria es interceptado por la lógica de control de la caché: Si el dato está disponible, se lo envía al microprocesador desde la caché, si no, se pasa el pedido a la memoria principal. Una vez que la memoria principal envía el dato pedido al microprocesador, la caché lo almacena para un posible uso posterior.

El éxito de una caché depende de la habilidad que tenga para almacenar datos que el microprocesador va a solicitar posteriormente, es decir, anticiparse a los pedidos del microprocesador. Para ello, existe un principio totalmente empírico, que la experiencia demuestra que se cumple muy ampliamente, denominado **Vecindad de las referencias** *(locality of reference)***.** Consta, a su vez, de dos subprincipios: Vecindad temporal y vecindad espacial.

- **Vecindad espacial**: Los programas solicitan datos o instrucciones cuyas direcciones en memoria están cercanas a los datos o instrucciones recientemente direccionados. Este principio se cumple, ya que los programas se escriben de forma secuencial y los datos suelen ser adyacentes.
- **Vecindad temporal**: Los programas tienden a usar los datos más recientes. Cuanto más antigua sea la información, más improbable es que un programa la solicite.

(Intentar mostrar la vecindad temporal y espacial en un lazo sencillo de programa)

Medida de la eficacia

Cuando el microprocesador solicita datos de la memoria, la caché comprueba si esta información está disponible en la caché. Si esto ocurre, se dice que hay presencia y la caché entrega el dato. Si los datos pedidos no están en la caché, se dice que se ha producido una ausencia. Al detectar esta situación, el controlador de la caché pasa el pedido a la memoria principal, y al mismo tiempo que el microprocesador recibe los datos solicitados, la caché los debe almacenar para tenerlos en caso de futuras referencias (por el principio de vecindad temporal). La caché carga datos adicionales hasta llenar una línea. Una línea es la mínima cantidad de información que el controlador lee desde la memoria principal para actualizar la RAM de datos de la caché. El tamaño de línea es mayor que el tamaño del dato pedido (por el principio de vecindad espacial es probable que sean requeridos los datos adyacentes).

La eficiencia de la caché se mide mediante la probabilidad de presencia *(hit rate)*.

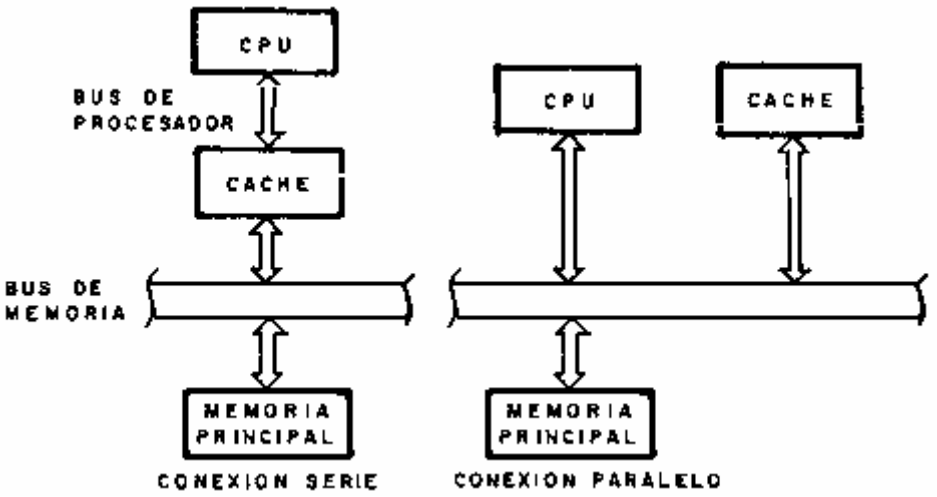
PROB. PRESENCIA = Nº de presencias en la caché / Nº total de peticiones a memoria.

Este parámetro, expresado en tanto por ciento, se denomina porcentaje de presencias. A veces, también se habla de probabilidad de ausencia *(miss rate)* que se obtiene haciendo *1-PP*. Si la probabilidad de presencia es muy cercana a uno, el microprocesador pasa la mayor parte del tiempo trabajando desde la RAM de la caché, muy rápida, y realiza muy pocos accesos a la memoria principal, mucho más lenta. Las prestaciones globales de la computadora aumentan **muy significativamente**. Es usual trabajar con subsistemas de caché cuyo porcentaje de presencias sea superior al 80% (de cada 10 datos pedidos, 8 están en la caché).

La probabilidad de presencia depende del cumplimiento del principio de vecindad de las referencias. Y esto depende de cómo esté escrito y estructurado el software. Por lo tanto, la eficacia de una caché depende, en gran medida, del programa que se esté ejecutando. En algunos casos, como por ejemplo, los saltos en las secuencias de programa y las conmutaciones de tarea, el principio de vecindad no se cumple.

Tipos de conexión

Hay dos formas principales de conectar una caché: En serie y en paralelo.



Caché Serie

La colocación de la caché se hace en serie entre el microprocesador y la memoria principal. El microprocesador envía el pedido en primer lugar a la caché, y si ésta no tiene el dato, se pasa a la memoria principal.

La principal ventaja de la disposición de la caché en serie es que, en caso de presencia, no se producen pedidos a la memoria principal, liberando al bus de memoria, el cual queda libre para que otro microprocesador pueda acceder a la memoria. Sólo cuando se produce una ausencia, el pedido se transmite a la memoria principal. En este caso, hay un retardo en el acceso (debido al tiempo inicial empleado para decidir) y éste se hace más lento que si no existiera la caché. Este es el inconveniente más grave de la memoria caché en serie. Además, la caché serie no puede quitarse del sistema, ya que está en el camino de transferencia de datos entre la CPU y la memoria principal.

Caché Paralelo

La caché se sitúa en paralelo con la memoria principal. El pedido llega simultáneamente a la memoria principal y a la caché. Si la caché tiene el dato pedido, se lo suministra al microprocesador y envía una señal a la memoria principal para que interrumpa el ciclo de lectura. En caso de que se produzca una ausencia, la caché no puede suministrar el dato, y lo hace la memoria principal.

La principal ventaja de la caché paralelo es que se puede quitar del sistema sin tener que hacer modificación alguna, ya que no afecta a la interconexión entre el microprocesador y la memoria principal. De esta forma la caché es un elemento opcional, que puede ponerse o no, dependiendo de los requerimientos del sistema. No hay penalización en el tiempo de acceso en caso de ausencia, debido a que el pedido se envía simultáneamente a la memoria principal. Es mucho más sencilla de fabricar que la caché serie. La desventaja es que la utilización del bus no se reduce, debido a que todas las peticiones de datos provocan accesos a memoria.

Arquitectura

Tamaño

Evidentemente, cuánto mayor sea el tamaño de una caché, mayor será la probabilidad de presencia. Sin embargo, el aumento del costo limita el tamaño de la caché. Además, por el principio de vecindad, la eficiencia de la caché no aumenta proporcionalmente al aumento de tamaño. Almacenar en la caché mucha cantidad de código y datos que no están próximos a los referenciados recientemente no mejora el rendimiento ya que, al salirse del entorno de vecindad, no es probable que sean requeridos.

El tamaño ideal depende de muchos factores, entre ellos, la posibilidad de multitarea, el tipo de programas a ejecutar, el tamaño de la memoria principal, etc. y debe ser evaluado en cada caso particular. Por lo general, el tamaño de la caché varía entre 4 KB y 512 KB, pudiendo utilizarse cachés de hasta tres niveles (con tamaños mayores para los niveles superiores). El tiempo de acceso aumenta para cachés más grandes debido a que el controlador de caché debe explorar mayor cantidad de posiciones para detectar presencia.

Organización

Hay tres tipos principales de organización para una memoria caché: Totalmente asociativa, de Correspondencia directa y Asociativa de n vías.

Caché totalmente asociativa

En las cachés totalmente asociativas *(fully associative cache)*, un dato ubicado en una posición de memoria principal puede almacenarse en cualquier posición en la memoria caché.

Cuando el microprocesador solicita un dato o instrucción, el controlador de la caché tiene que comparar cada una de las etiquetas presentes en la caché con la dirección del dato pedido en la memoria principal, lo que implica hacer una gran cantidad de comparaciones. Esta es una desventaja ya que dichas comparaciones deben hacerse simultáneamente, para mantener un tiempo de acceso reducido. Por ello, para

disminuir la cantidad de etiquetas a comparar, estas cachés no suelen ser muy grandes (de menos de 4KB).

Aunque esta organización es la que proporciona los mejores rendimientos, ya que los datos pueden almacenarse en cualquier celda libre de la caché, no suelen utilizarse en las computadoras, a causa de su complejidad y costo.

Caché de correspondencia directa

En las cachés de correspondencia directa *(direct mapped cache)*, un dato ubicado en una posición de memoria principal sólo puede almacenarse en una posición específica en la memoria caché.

La lógica de control de la caché "ve" a la memoria principal dividida en bloques o páginas del mismo tamaño que la RAM de datos de la caché. Para almacenar un dato en la caché se toma el valor del desplazamiento de la dirección de la memoria principal respecto del comienzo de bloque. El desplazamiento identifica el lugar de la memoria caché dónde el dato debe ser almacenado. En la figura se aprecia esta correspondencia.

Las celdas de memoria con iguales valores de desplazamiento, respecto del comienzo de bloque respectivo, se deben ubicar en el mismo lugar en la memoria caché. Este es el principal inconveniente de este tipo de organización, ya que aunque haya celdas libres, si los desplazamientos de dos datos utilizados frecuentemente coinciden, en cada pedido del procesador se produce una ausencia (analizar la causa).

Esta organización es la más simple de llevar a cabo. Necesita una única comparación para saber si el dato pedido está en la caché o no. Se compara el contenido de la etiqueta correspondiente al desplazamiento con el número de la página. Si coinciden, es una presencia. Además, las etiquetas son muy simples: no necesitan alojar la dirección completa (como en las memorias totalmente asociativas), solamente almacenan el número de página.

Aunque esta organización sea sencilla, el rendimiento es muy bajo en entornos multitarea, donde es probable que los datos e instrucciones tengan desplazamientos coincidentes, por lo que no suelen utilizarse.

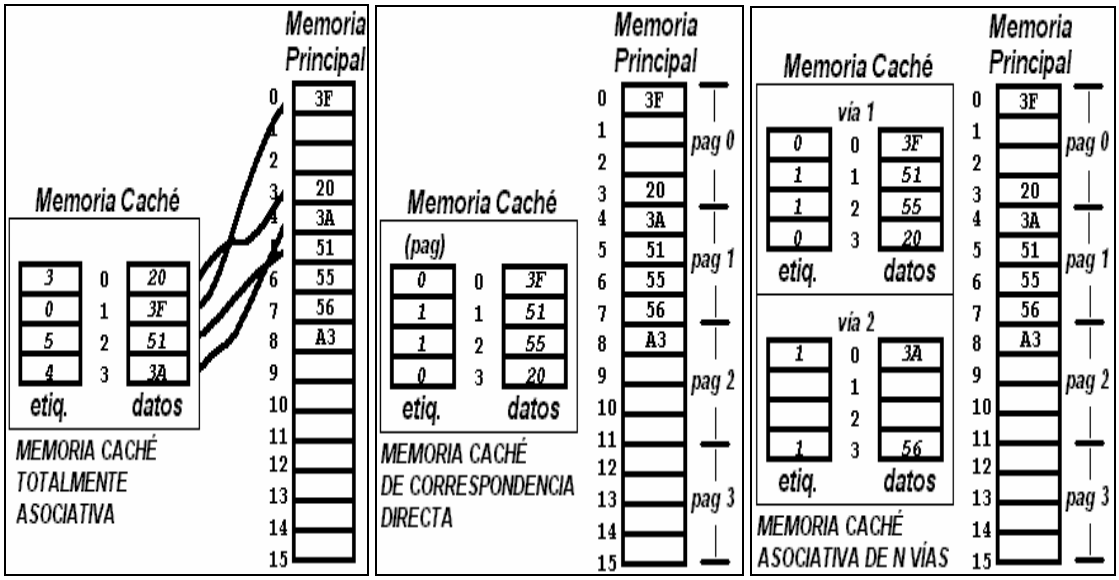
Caché asociativa de n vías

La caché asociativa de n vías *(n-way set associative cache)* tiene dividida su memoria en n cachés de correspondencia directa que operan en paralelo (simultáneamente). Se llama de n vías porque se accede directamente a las n cachés por n vías al mismo tiempo.

Un dato puede ir a cualquiera de las vías. En la figura se representa una memoria caché asociativa de 2 vías. En el caso de que la organización fuera de 4 vías, en vez de dos bancos de memoria habría cuatro. Las memorias de correspondencia directa pueden considerarse asociativas de una vía. Cuantas más vías existan, más complicada será la lógica de control de la caché, al tener que hacer más comparaciones simultáneamente para saber si un dato está presente o no.

La caché asociativa de n vías también "ve" a la memoria principal dividida en páginas, como lo hacía la de correspondencia directa. Cada página es del mismo tamaño que una vía o grupo de la caché. El tamaño de cada vía es igual al tamaño de la caché dividido por el número de vías. Un dato se almacena en la caché con el mismo desplazamiento respecto del comienzo de página en cualquiera de los bancos o vías. Para saber si un dato está almacenado en la caché, se comparan tantas etiquetas como vías tenga con el número de página de dicho dato. Dichas comparaciones son simultáneas para evitar retardos de tiempo.

La caché asociativa de n vías mejoran el rendimiento de las memorias de correspondencia directa y tienen un costo y complejidad inferiores a las memorias totalmente asociativas, por lo que suelen ser las más usadas en las computadoras.



NOTA: Los ejemplos son sólo ilustrativos. Contemplan un tamaño de línea (tamaño del bloque asociado a una etiqueta) de sólo 1 byte, y una memoria principal de 16 bytes.

Actualización de la caché

La actualización de la caché debe llevarse a cabo cuando se produce una ausencia. En ese caso, el dato pedido no está en la caché, por lo que se carga desde la memoria principal; la lógica de control debe guardar ese dato en la caché para posibles referencias futuras; pero para hacerlo debe desalojar una información presente en la RAM de datos de la caché.

En el caso de las cachés de correspondencia directa no hay alternativa posible: la dirección de memoria buscada sólo puede almacenarse en una posición determinada dentro de la caché. Sin embargo, las cachés con nivel de asociatividad mayor que uno se encuentra con la disyuntiva de qué posición específica de la caché se va a sustituir. Existen diversos algoritmos de reemplazo para una caché: LRU, FIFO, RANDOM, etc. La elección del algoritmo de reemplazo, junto con la organización de la caché, son dos factores que afectan de forma muy importante al rendimiento total.

Los más usados son: el RANDOM y LRU. El primero actualiza de forma *aleatoria* cualquier posición de la caché. Es un algoritmo sencillo, pero puede dar lugar a sustituir precisamente la posición que acaba de ser actualizada, lo que haría disminuir la eficacia del subsistema de caché. En el algoritmo LRU *(least recently used)* se

sustituye aquella información que lleva menos tiempo sin ser accedida. Es eficiente pero difícil de implementar.

Cada vez que la caché necesita ser actualizada, se debe transferir desde memoria principal, una línea completa (es decir, más información de la realmente pedida, que por el principio de vecindad espacial es probable que sea requerida). Las líneas suelen tener más de 32 bits. El llenado de la línea de la caché se puede hacer de dos formas:

- Dato pedido en último lugar: El procesador debe esperar hasta que se llene la línea para disponer del dato solicitado. Es muy simple de implementar.
- Dato pedido en primer lugar: El procesador recibe el dato y puede seguir operando. La línea se llena a partir del dato pedido por el procesador. El rendimiento es mayor, pero la implementación es más complicada.

Actualización de la memoria principal

También la memoria principal necesita ser actualizada. Como el microprocesador trabaja con el subsistema de caché, las escrituras se realizan sobre la RAM de la caché. El problema se plantea desde el momento en que ésta no es, precisamente, la memoria principal. Si el sistema es multiprocesador y otro procesador quiere acceder a esa posición de memoria, obtendrá un dato incorrecto (no actualizado). Como consecuencia, es importante mantener una política coherente de actualización de la memoria principal para evitar el acceso involuntario a datos obsoletos.

Existen tres métodos diferentes: Escritura Inmediata *(Write Through)*, Escritura Diferida *(Posted Write Through)* y Escritura Obligada *(Write Back)*.

Escritura Inmediata

En este método, todas las escrituras del microprocesador sobre la memoria caché son pasadas inmediatamente a la memoria principal. La memoria principal siempre contiene datos válidos.

La ventaja del método es que es simple de implementar, pero el rendimiento se reduce mucho debido al tiempo empleado para escribir los datos en la memoria principal. Además, la utilización del bus es elevada, lo cual es un inconveniente grave en sistemas multiprocesador.

Escritura Diferida

En este otro método, la caché incorpora una serie de Registros Intermedios de Escritura. Las escrituras en la caché se envían a estos registros, liberándose el procesador inmediatamente. A medida que el bus de memoria queda ocioso, el controlador de caché escribe los datos en la memoria principal.

La ventaja de este método es que el microprocesador no disminuye sus prestaciones debido a los largos tiempos de escritura en memoria principal. La desventaja es que el tiempo de utilización del bus es el mismo que en el método anterior, aunque esté mejor repartido.

Escritura Obligada

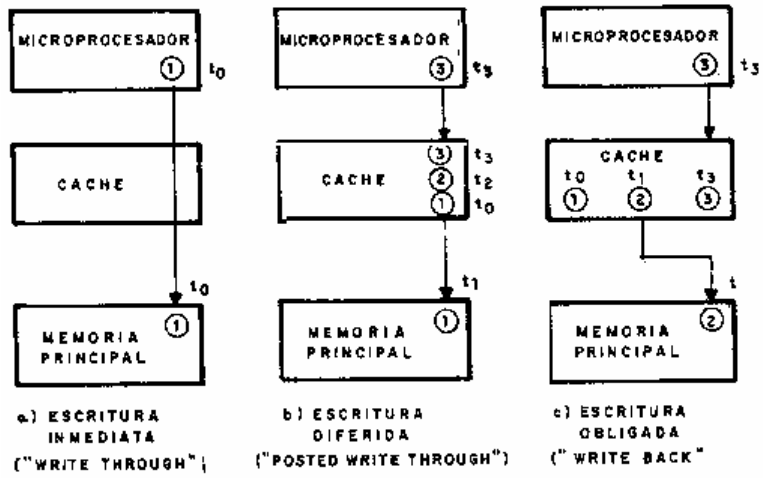
En este caso, las escrituras del procesador se hacen en la memoria caché, y sólo se hacen en la memoria principal si resulta estrictamente necesario.

Pueden darse dos situaciones diferentes que obliguen a la caché a actualizar la memoria principal:

La primera se produce cuando otro microprocesador va a leer un dato de la memoria principal. En este caso, se detiene el acceso y se actualiza la posición de memoria que va a ser leída con la información contenida en la caché.

La segunda situación tiene lugar cuándo se reemplaza un determinado dato de la memoria caché que ha sido modificada mientras estuvo en la caché. En ese caso, primero se almacena la información en la memoria y luego se sustituye por los nuevos datos. Para detectar cuándo se produce esta situación cada etiqueta de la caché tiene unos bits que indican si el dato ha sido modificado. Si un dato en caché no ha sido modificado, no necesita escribirse en memoria principal.

Este método reduce las escrituras en memoria principal a las estrictamente necesarias, por lo que disminuye en gran medida la utilización del bus. Es el más eficiente pero el más difícil de implementar.



Ejemplo: memoria caché del microprocesador 80486 (interna):
Tamaño: 8-Kbytes
Organización: Asociativa de cuatro vías (four-way set associative)
Tamaño de línea: 16-bytes (128 bits)
Actualización de la caché: pseudo-LRU
Actualización de la memoria principal: Escritura Inmediata (Write-Through)

Memoria caché

(Datos obtenidos de manuales)

"Cache memory size of 32 KB, 16 bytes cache line size" (386 DX)
"33 MHz CPU requires 20 ns (tag) and 25 ns (data) SRAM chips" (386 DX)
"L1 cache WriteBack CPU system" (486)
"Direct Mapped L2 cache controller" (486)
"Level 2 WriteBack Policy for high performance" (486)
"DRAM is cacheable to 64 MB" (386DX)
"Pipeline Burst SRAM" (Pentium)
"8 KB, 4-Way Set Associative, 64-Byte line"

Bibliografía:

1) *Scott Mueller - Manual de Actualización y Reparación de PCs - Prentice Hall (General)*
2) *Mario Ginzburg - La PC por dentro - Biblioteca Técnica Superior (General)*
3) *Angulo, Funke - Microprocesadores Avanzados 386 y 486 - Paraninfo (Memoria caché)*
4) *Kingston Technology - The ultimate memory guide - http://www-kingston.com (Guía completa: requerimientos de memoria, instalación de módulos)*
5) *IBM - Understanding Static RAM Operation (Memoria SRAM)*
6) *IBM - Understanding DRAM Performance Specifications (FPM, EDO, SDRAM)*
7) *Jon Stokes - RAM Guide - http://arstechnica.com/ (DDR SDRAM, RDRAM)*
8) *TomsHardware - http://www.tomshardware.com (Guía de memorias, DDR SDRAM, RDRAM)*
9) *Sharky Extreme - SDRAM Memory Performance Guide - http://www.sharkyextreme.com/ (SDRAM)*
10) *SysOpt - RAM Technology Primer: CAS Latency - http://www.sysopt.com/ (SDRAM)*
11) *ExtremeTech - Inside RAM - http://www.extremetech.com (SDRAM, DDR SDRAM, RDRAM)*
12) *Aces Hardware - Ace's Guide to Memory Technology - http://www.aceshardware.com/ (RDRAM, DDR SDRAM)*
13) *Micron – DDR2 Offers New Features and Functionality (Documento PDF – DDR2)*
14) *Power USERS nº 15 (General)*
15) *Power USERS nº8 (DDR2)*