CAPÍTULO VII: GESTIÓN DE LA MEMORIA

INTRODUCCIÓN

La memoria es la unidad donde se almacena la información que necesita el computador, es decir, tanto las instrucciones que forman los programas, como los datos.

Nos referiremos a la memoria principal, de tal forma que para seleccionar una **palabra** debe especificarse su dirección, por lo que se dice que es **memoria accesible por dirección**; aunque a veces también se utilizan **memorias accesibles por contenido o memorias asociativas**, donde en lugar de una dirección, se da una parte del contenido de la posición (denominada **clave**) y la memoria proporciona la totalidad del contenido de esa clave.

Un parámetro de vital importancia es la velocidad de respuesta, utilizándose usualmente tres parámetros relacionados con la velocidad:

✍ **Tiempo de acceso, t_A,** que es el tiempo máximo que se tarda en leer (tiempo de acceso de lectura) o escribir (tiempo de acceso de escritura) el contenido de una posición de memoria.

✍ **Tiempo de ciclo, t_c,** que es el tiempo mínimo entre dos lecturas consecutivas.

✍ **Ancho de banda, AB,** que es el número de palabras que se transfieren entre memoria y la CPU por unidad de tiempo: $AB = 1 / t_c$.

La memoria principal de un computador suele estar constituida por dos tipos de memoria: ROM y RAM, y aunque en este capítulo nos centraremos en cuestiones relativas a las memorias RAM, la mayoría de las ideas se pueden particularizar también para memorias ROM.

En la actualidad, las memorias RAM se estructuran en circuitos integrados (chips), existiendo dos tipos básicos de circuitos: RAM estáticos (SRAM) y RAM dinámicos (DRAM). Los primeros son más rápidos, pero su grado de integración es menor (se necesitan por cada bit más transistores); los segundos no son tan rápidos pero su densidad de integración es mucho mayor.

Jerarquía de la memoria.

La CPU capta las instrucciones y datos de la memoria principal, almacenando en ella los resultados de las operaciones, por tanto, es conveniente que las velocidades de funcionamiento de ambas unidades sean del mismo orden de magnitud; sin embargo, esto no suele ser así; problema que se palia en parte utilizando una memoria más veloz constituyendo una memoria caché.

Todo programa, para ser ejecutado, debe ser cargado en la memoria principal. Para posibilitar la ejecución de programas mayores que el tamaño de la memoria disponible se ha desarrollado la técnica de la **memoria virtual**, con la que el programa y sus datos se mantienen en disco (memoria secundaria o masiva), y sólo la parte de ellos implicada en la ejecución se mantiene en memoria.

Las prestaciones de un computador vienen en gran parte determinadas por la de su memoria. La memoria ha de cubrir cuatro objetivos básicos:

a) Tamaño o capacidad, **s**, de almacenamiento suficiente.
b) Tiempo de acceso, **t**, lo menor posible.
c) Ancho de banda alto, **b**.
d) Coste por bit reducido, **c**.

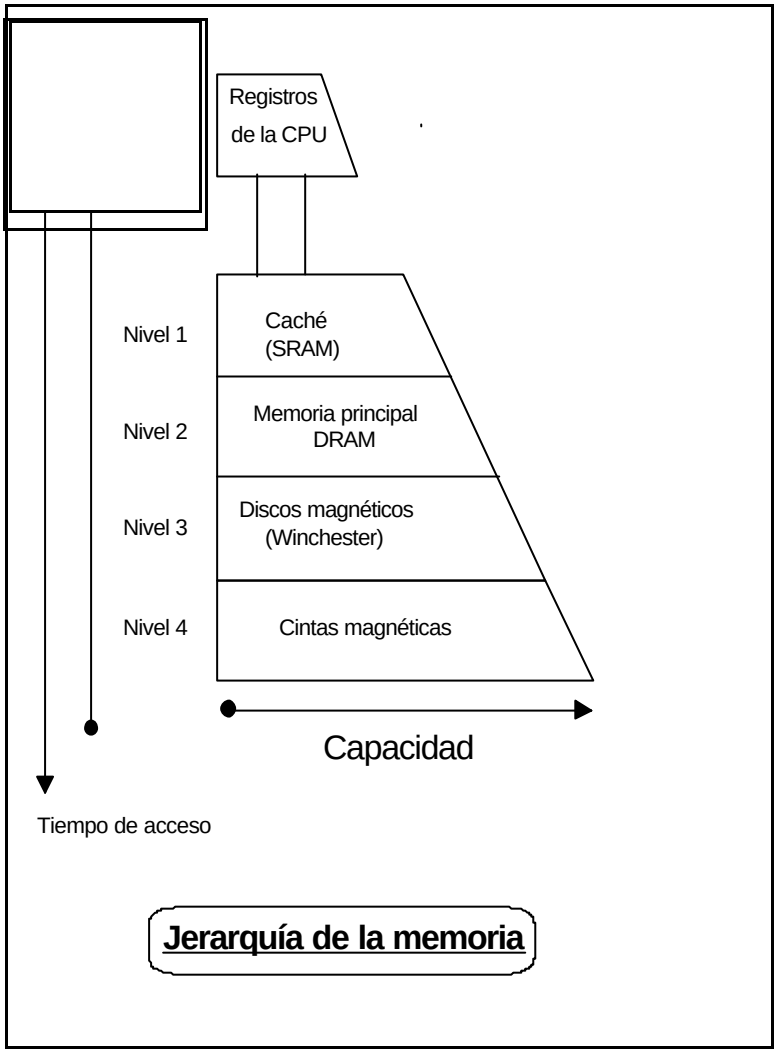
Desafortunadamente, no existe una tecnología concreta que reúna simultáneamente las cuatro características indicadas. Debido a ello se han desarrollado múltiples sistemas de almacenamiento auxiliar de la memoria principal, los cuales se han descrito en el Capítulo 4 "Periféricos". No obstante, en la siguiente tabla, se incluye un resumen de las características de los principales dispositivos de memoria.

Características de distintos dispositivos de memoria

Nivel	Dispositivo	Capacidad	Tiempo de acceso	Ancho de banda MB/s
0	Registros CPU	< 1 KB	3 a 100 ns	400 a 800
1	Caché (SRAM)	32 KB a 4 MB	10 a 40 ns	200 a 400
2	Memoria principal (DRAM)	1 MB a 1 GB	30 a 100 ns	100 a 200
3	Discos (Winchester)	100 MB-200 GB	8 a 18 ms	1 a 5
4	Disco óptico CD-ROM	680 MB	0,1 a 0,3 s	0,6
5	Disco magnetoóptico (WMRA)	0,5 a 1 GB	0,03 s	0,15
6	Disco óptico WORM	650 MB	0,1 a 0,3 s	0,15
7	Cinta magnética (DAT)	2,56 GB	60 s	0,21
8	Disquetes	2,88 MB	100 ms	0,05

En el nivel superior se encuentran los registros internos de la unidad de procesamiento, y en el inferior las cintas magnéticas, verificándose que cuanto más alto es el nivel, menor es su capacidad, pero la velocidad es mayor.

La CPU es el elemento principal del computador, ya que desde allí se controla su funcionamiento completo y en él se hace el procesamiento de datos. Por ello, interesa que los datos con los que en un momento dado va a operar la CPU estén en el nivel más alto de la jerarquía, ya que si no habrá que disponer de procedimientos (gestionados por el sistema operativo), para buscarlos en niveles inferiores, estas búsquedas (por **fallo** en los accesos) provocan una disminución del rendimiento del computador. Por lo general, toda información de un nivel se encuentra también almacenada en el nivel inmediato inferior, y así sucesivamente.



Jerarquía de la memoria

Existen estrategias para determinar qué información en un momento dado debe ubicarse en cada uno de los niveles superiores, de forma que se produzca el menor número de **fallos** en los accesos a datos en un determinado nivel. Así, si se está ejecutando un determinado programa o utilizando un grupo de datos es muy probable que si se referencia a un elemento, los elementos cercanos a él tiendan a ser referenciados pronto (principio denominado de **localidad espacial**). También, debido a que los bucles son muy frecuentes en programación, si se referencia un elemento, tenderá a ser nuevamente referenciado pronto (principio de **localidad temporal**). Estos dos principios han inspirado distintos procedimientos para gestionar la memoria caché y la memoria virtual.

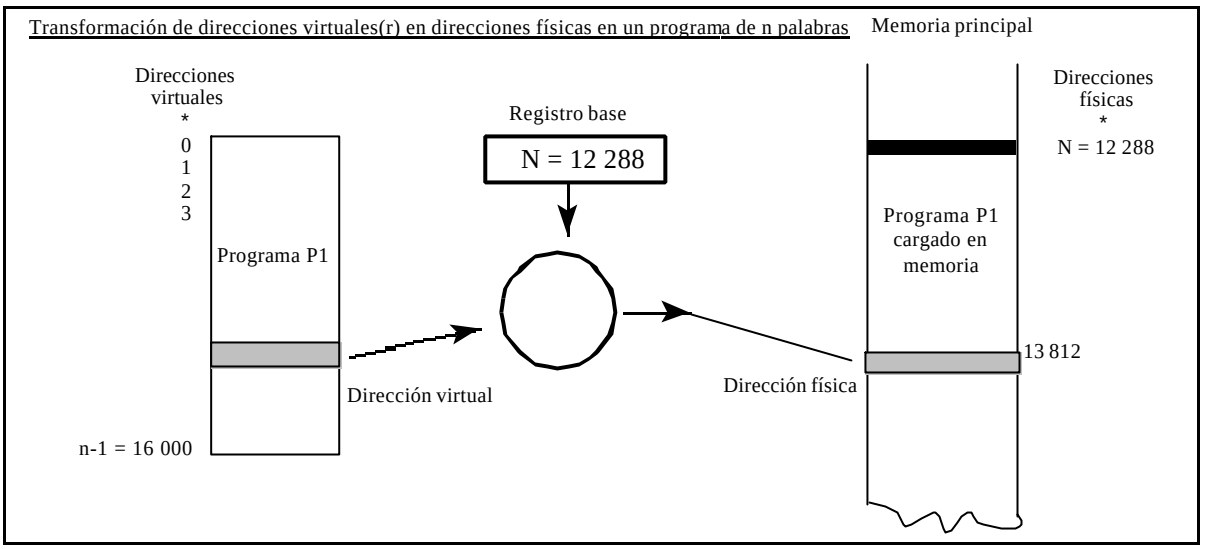
GESTIÓN DE LA MEMORIA.

Un programa máquina es un conjunto ordenado de instrucciones en código máquina. Estas instrucciones, en el momento de ejecutarse, "encajan" en palabras de memoria que pueden numerarse correlativamente de la 0 a la n-1 (suponiendo que el programa ocupa n palabras de memoria). Estas direcciones se denominan *direcciones virtuales o lógicas* .

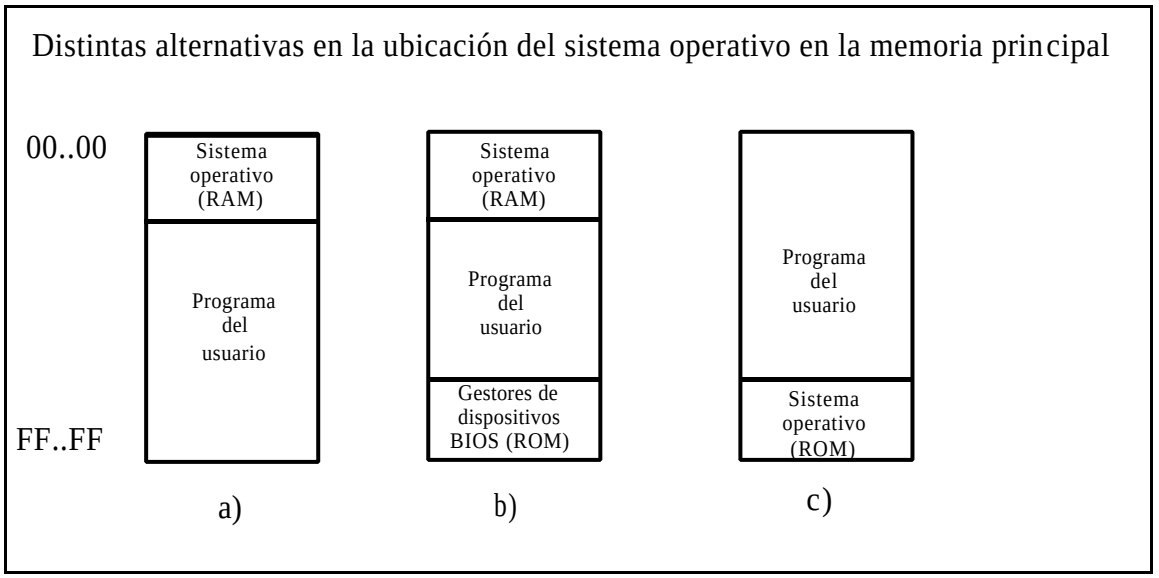
El programa anterior, al ser cargado en memoria, ocupará determinadas posiciones de la misma (en total n). Supongamos que las instrucciones del programa se almacenan consecutivamente; si se cargan a partir de la dirección N, quedará ubicado de la dirección N a la N + (n-1).

A *N* se le suele denominar *dirección base* y a las direcciones *f* en que realmente se almacenan las instrucciones (con direcciones lógicas *i*, por ejemplo) se les denominan *direcciones físicas*.

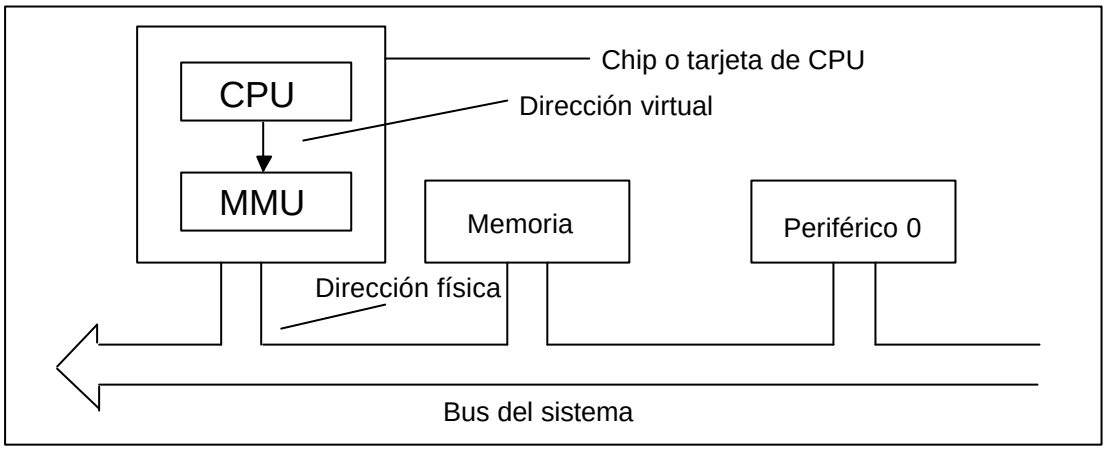
Se verifica que $f = N + i$; para todo $0 \odot i \odot n-1$.



En los sistemas operativos de monoprogramación la memoria principal se puede organizar de diversas maneras. El sistema operativo puede ocupar las primeras posiciones de memoria o las últimas (que puede ser de tipo ROM) o incluso parte del sistema operativo puede estar en la zona RAM de direcciones bajas, y otra parte de él (los gestores de periféricos) y el cargador inicial en ROM, en las direcciones altas. Esta última situación se corresponde con la de los PC's compatibles, en los que la ROM se denomina sistema básico de entradas y salidas o BIOS (Basic Input Output System). En cualquier caso, cuando el usuario da una orden, si el proceso que la implementa no está en memoria, la concha (shell o intérprete de órdenes) del sistema operativo se encarga de cargarlo en memoria desde disco y el distribuidor da paso a su ejecución. Cuando finaliza el proceso, el sistema operativo visualiza el indicador de petición de orden y espera a que le dé una nueva, en cuyo caso libera la zona de memoria ocupada, sobreescribiendo el nuevo programa sobre el anterior proceso.



En un sistema de multiprogramación, cuando un programa se carga en memoria para ser ejecutado (o continuar su ejecución), el sistema operativo, de acuerdo con los espacios libres de memoria, le asigna una dirección base, y transforma direcciones virtuales en direcciones físicas, según podemos comprobar en el primer cuadro de la página anterior. Estas transformaciones suelen ser efectuadas por circuitos especializados que constituyen la **unidad de gestión de memoria (MMU - Memory Management Unit -)**, que en la actualidad se suele integrar en el mismo chip de la CPU. La asignación de memoria para distintos procesos sigue ejecutándose concurrentemente suele hacerse, dependiendo del sistema operativo, de alguna de las formas que se indican a continuación:



- Particiones estáticas.
- Particiones dinámicas.
- Paginación.
- Segmentación.
- Memoria virtual.

Un concepto importante que se utiliza es el segmento. Se denomina *segmento* a un grupo lógico de información, tal como un programa, una subrutina, una pila, una tabla de símbolos, un array y una zona de datos. Esta unidad no es de un tamaño preestablecido, pues depende totalmente del programa de que se trate. Un programa ejecutable (listo para ser ejecutado por la CPU) es una colección de segmentos.

La memoria virtual permite a los usuarios hacer programas de una capacidad muy superior a la que físicamente tiene el computador. Por ejemplo, el microprocesador 80386 admite hacer programas de hasta 64 Terabytes (2^{46}), utilizando una memoria principal mucho menor (por ejemplo, de 2 Megabytes). En realidad, la memoria virtual hace posible que la capacidad máxima de los programas venga determinada por el espacio que se reserve en disco para ella y no por la memoria principal. En definitiva, los sistemas con memoria virtual presentan al usuario una memoria principal "aparente" mayor que la memoria física real.