

Memorias

Segmentación

¿Qué son las Memorias?

Los dispositivos que contienen, almacenan, guardan y memorizan datos.

Tiene tres características

Ubicación

Internas

Externas

Velocidad

Rapidas

Lentas

Capacidad

Baja

Alta

Gestión de la Memoria (IA-32)

Introducción

El procesador 8086 fue el primer de 16 bits de ancho de palabra, sus registros tienen ese tamaño. Pero su bus de direcciones es de 20 líneas que permiten armar direcciones de 20 bits para disponer de un espacio en memoria de 1MByte.

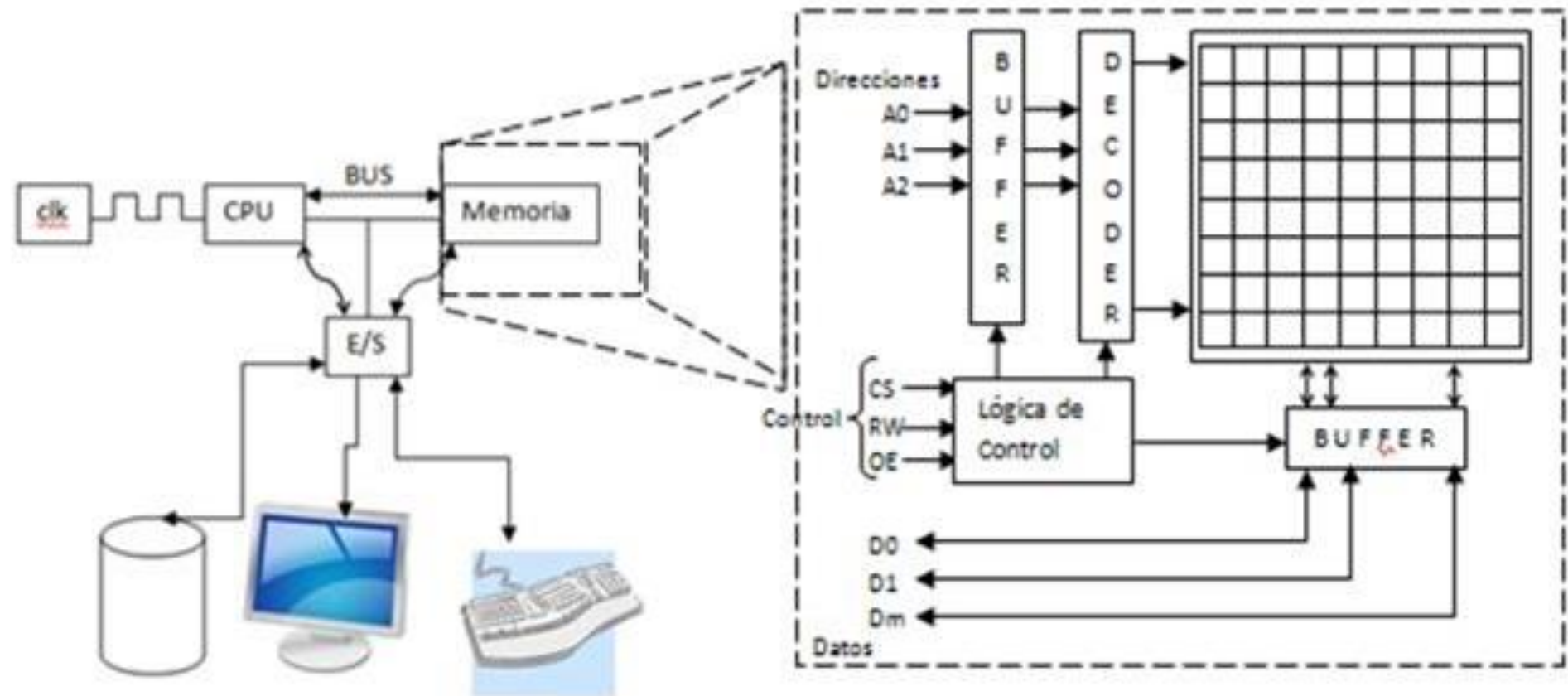
A partir del procesador 80286 (16bits) y del 80386 (32bits) se incluyó un nuevo modo de trabajo denominado Modo Protegido.

Los procesadores de 32bits tienen bus de datos de 32 líneas y un bus de direcciones de 32 líneas que permiten direccionar hasta 2^{32} (4Gbytes).

El Pentium utiliza direcciones virtuales de 46 bits podemos direccionar hasta 64TB como si fuera un espacio único de memoria principal.

¿Cuál es la función dentro de la arquitectura?

- Almacena datos e instrucciones de programas
- Interconecta la CPU y los dispositivos de E/S



Gestión de la Memoria (IA-32)

ESPACIO FÍSICO

La memoria está organizada como una secuencia de bytes, que se direccionan por el Bus de Direcciones.

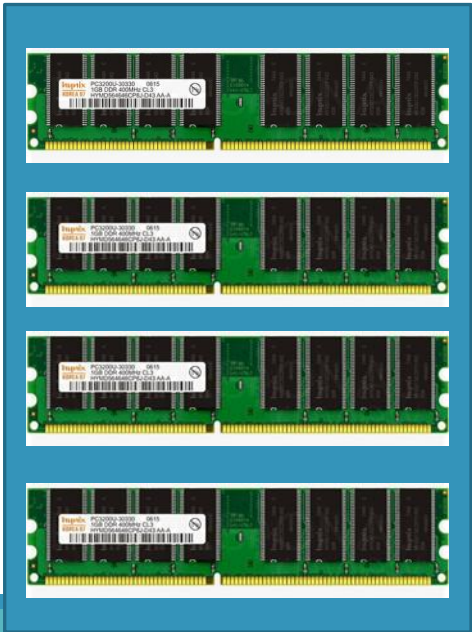
Denominamos **memoria física** a la memoria conectada a este bus y a las direcciones que van sobre este bus las denominamos **direcciones físicas**.

ESPACIO LÓGICO

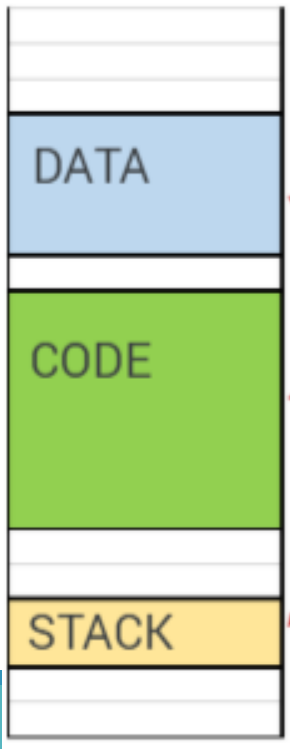
El espacio de direccionamiento está organizado en segmentos. Se necesitan 2 valores, la **base** del segmento dónde se encuentran las instrucciones o los datos y el **desplazamiento** a partir del inicio de ese segmento en dónde se encuentra efectivamente.

A la dirección formada por estos 2 valores la denominamos **dirección lógica**

Dirección Física a través del bus de direcciones



Memoria Física



Desplazamiento

Base

Gestión de la Memoria (IA-32)

¿Cómo pasar de la dirección lógica a la dirección física?

Estando en Modo Protegido se consigue con un proceso en dos niveles :

- Traslación de una dirección lógica
- Paginación del espacio lineal

Para hacer esta traducción existe una unidad denominada **MMU (Memory Management Unit)** que internamente tiene 2 unidades:

La Unidad de Segmentación que traslada la **dirección lógica** a una **dirección lineal** , y la Unidad de Paginación que traduce la **dirección lineal** en una **dirección física** que es la que se envía a la memoria física por el bus de direcciones.

TÉCNICAS DE DIRECCIONAMIENTO A MEMORIA



Espacio lógico de la memoria (Memoria Virtual)

Método de organización y gestión de la memoria que proporciona un espacio mucho mayor del que dispone físicamente. Es una “partición” del disco que anticipa las direcciones que necesita la CPU y con un buen algoritmo accedemos a ellas con mayor velocidad.

Dimensión de la Memoria

Bits,bytes,etc .Son las unidades que maneja la memoria.

La capacidad se expresa en Bits, Bytes, Kbytes o KB, GB, TB, PetaBytes, ExaByte, ZettaBytes, YottaBytes. En ese orden aumenta cada 2^{10} veces

No tiene que ver con los múltiplos físicos, por que :

$$2^{10} = 1.K = 1K = 1024 \neq 1000 = 10^3 = 1Kilo(fisico)$$

Capacidad del Módulo (Banco de memoria)

Es fácil calcular la capacidad de un módulo de memoria si se saben las capacidades de los chips que hay en éste.

Si hay 8 chips de 64 Mbits, es un módulo de 512 Mbits.

Sin embargo, debido a que la capacidad de un módulo se describe en megabytes, y no en megabits, se deben convertir los bits a bytes. Para hacer esto, se divide el número de bits entre 8.

En el caso del módulo de 512Mbit:

$$\frac{512\text{Mbits}}{8 \text{ bits por byte}} = 64\text{MB}$$

Tal vez haya escuchado que los módulos de memoria estándar en la industria se describen como: “4M x 32” (es decir, “4Meg por 32bits”), o “16M x 64” (“16Meg por 64bits”). En estos casos, calcule la capacidad del módulo exactamente como si fuera un chip:

4Mx32 es 128Mbits

$$\frac{128\text{Mbits}}{8 \text{ bits por byte}} = \text{módulo de } 16\text{MB}$$

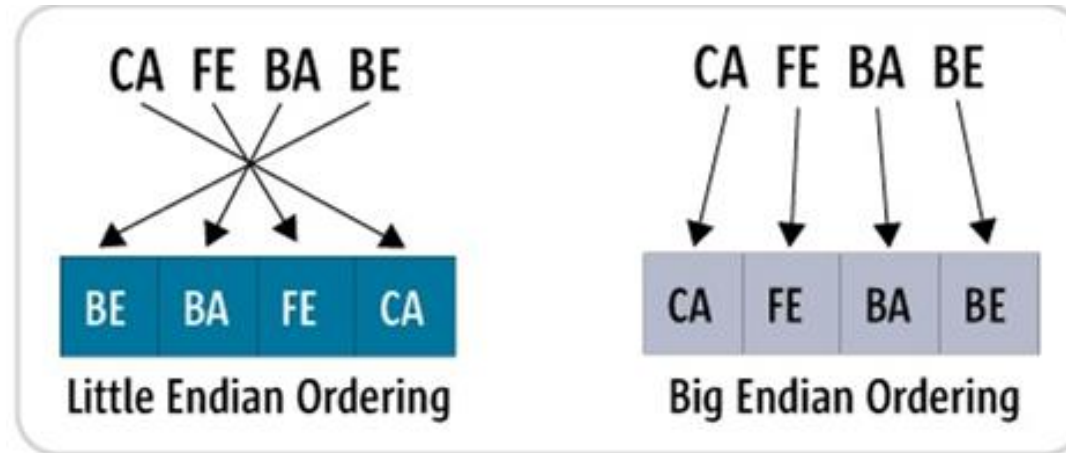
16Mx64 es 1024Mbits

$$\frac{1024\text{Mbits}}{8 \text{ bits por byte}} = \text{módulo de } 128\text{MB}$$

Tipos de almacenamiento de bytes en memoria

LITTLE ENDIAN

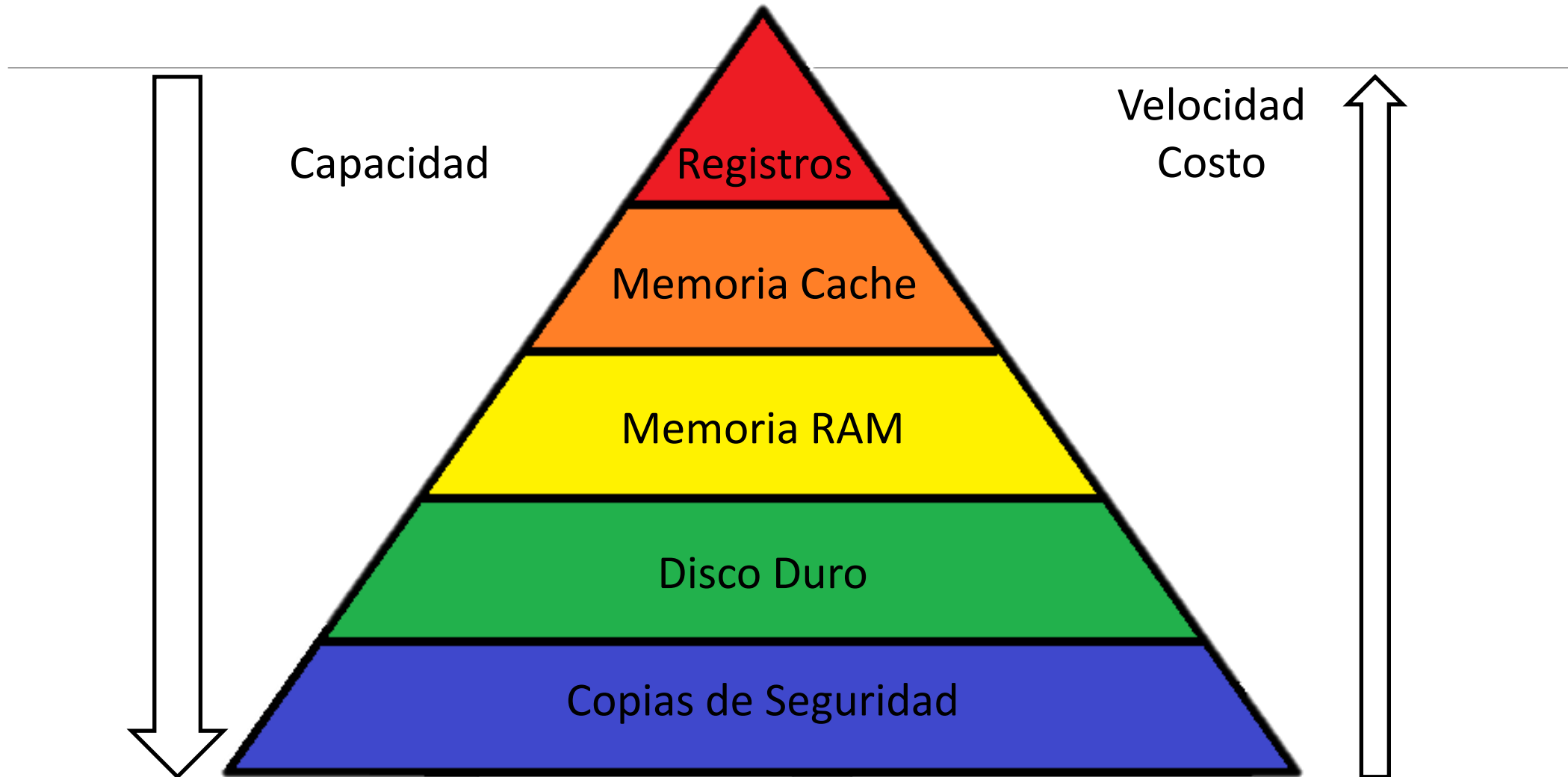
el byte menos significativo se almacena en la dirección más baja y cada octeto se almacena en memoria en forma invertida



BIG ENDIAN

el byte menos significativo se almacena en la dirección numéricamente más baja y el más significativo en la más alta

Jerarquía de memorias



Clasificación según modo de acceso

- Acceso aleatorio: Un componente de selección habilita una posición e inhabilita las demás. El componente de selección recibe “el identificador” único de la unidad de información implicada. Tiempo de acceso independiente a la posición.
- Acceso secuencial: para acceder a una unidad se establece una posición de referencia, a partir de ahí comienza el rastreo. Lectura hasta encontrar. Tiempo de acceso depende de esas posiciones.
- Acceso Asociativo: La búsqueda implica comparación de un grupo de bits de la unidad de información con el contenido de una posición. No se identifican por su dirección, chequear si este grupo de bits coincide con el contenido. Se envía grupo de bits, “rotulo” para comparación.

Clasificación según la duración de la información

- Volátiles: Pierden la información con el corte del suministro eléctrico.
- No volátiles/perennes/permanentes: no pierden la información post corte de suministro eléctrico.

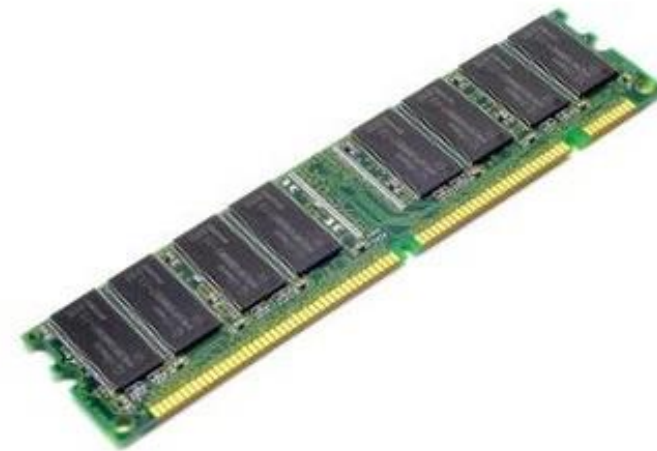
Tipos de memorias semiconductoras

Tipo de memoria	Clase	Borrado	Mecanismos de escritura	Volatilidad
Memoria de acceso aleatorio (RAM)	Memoria de lectura/escritura	Eléctricamente por bytes	Eléctricamente	Volátil
Memoria de sólo lectura (ROM)	Memoria de sólo lectura	No posible	Mediante máscaras	No volátil
ROM programable (PROM)			Eléctricamente	
PROM borrable (EPROM)	Luz ultravioleta, chip completo			
Memoria FLASH	Eléctricamente, por bloques			
PROM borrable eléctricamente (EEPROM)	Eléctricamente, por bytes			

Tipos de Memorias

La memoria de acceso aleatorio (RAM) se utiliza como memoria de trabajo de computadoras para el sistema operativo, los programas y la mayor parte del software. En la RAM se cargan todas las instrucciones que ejecuta la unidad central de procesamiento (procesador) y otras unidades del computador.

Se denominan «de acceso aleatorio» porque se puede leer o escribir en una posición de memoria con un tiempo de espera igual para cualquier posición, no siendo necesario seguir un orden para acceder (acceso secuencial) a la información de la manera más rápida posible.



Tipos de Memorias

Características de las RAM

En la industria de las memorias, las densidades de chips DRAM con frecuencia se describen por su organización de celdas. El primer número en la expresión indica la profundidad del chip (en ubicaciones) y el segundo número indica el ancho del chip (en bits). Si se multiplica la profundidad por el ancho, se obtendrá la densidad del chip. Aquí hay algunos ejemplos..

TECNOLOGÍA DE CHIPS DISPONIBLE ACTUALMENTE

	PROFUNDIDAD DE CHIPS EN MILLONES DE UBICACIONES	ANCHO DEL CHIP EN BITS	DENSIDAD DEL CHIP = PROFUNDIDAD x ANCHO
Chips de 16 Mbit			
4Mx4	4	4	16
1Mx16	1	16	16
2Mx8	2	8	16
16Mx1	16	1	16
Chips de 64 Mbit			
4Mx16	4	16	64
8Mx8	8	8	64
16Mx4	16	4	64
Chips de 128 Mbit			
8Mx16	8	16	128
16Mx8	16	8	128
32Mx4	32	4	128
Chips de 256 Mbit			
16Mx16	16	16	256
32Mx8	32	8	256
64Mx4	64	4	256

Tipos de Memorias

La memoria de sólo lectura, conocida también como ROM (acrónimo en inglés de read-only memory), es un medio de almacenamiento utilizado en ordenadores y dispositivos electrónicos, que permite solamente la lectura de la información y no su escritura, independientemente de la presencia o no de una fuente de energía.



Tipos de Memorias

La PROM es una memoria programable de solo lectura. Estas no vienen programadas de fábrica, sino que es el propio usuario el que se encarga de grabar la información y una vez que los datos se han grabado no se pueden borrar.



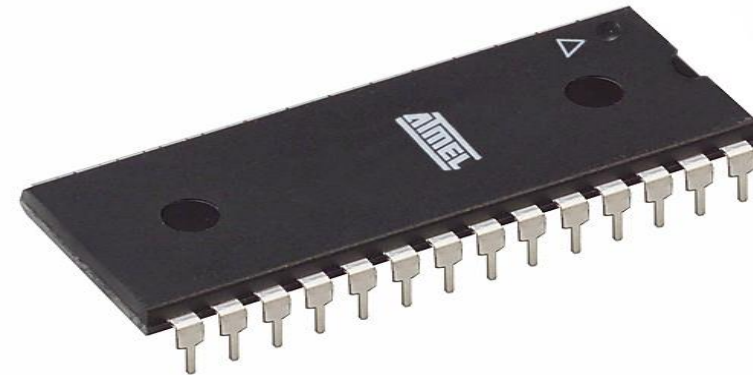
Tipos de Memorias

Las EPROM, o memorias de lectura reprogramables, se programan mediante impulsos eléctricos y su contenido se borra exponiéndolas a la luz ultravioleta para poder volver a ser grabadas



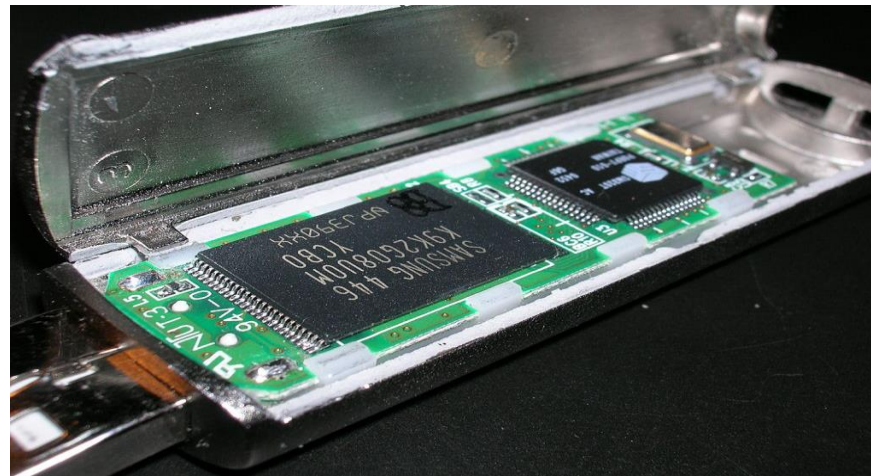
Tipos de Memorias

EEPROM o E²PROM (ROM programable y borrable eléctricamente). Es un tipo de memoria ROM que puede ser programada, borrada y reprogramada eléctricamente, a diferencia de la EPROM que ha de borrarse mediante un aparato que emite rayos ultravioleta. Son memorias no volátiles.



Tipos de Memorias

La memoria flash —derivada de las siglas EEPROM— permite la lectura y escritura de múltiples posiciones de memoria en la misma operación. Gracias a ello, la tecnología flash, siempre mediante impulsos eléctricos, permite velocidades de funcionamiento muy superiores frente a la tecnología EEPROM primigenia, que sólo permitía actuar sobre una única celda de memoria en cada operación de programación. Se trata de la tecnología empleada en los dispositivos denominados memoria USB.



Memorias RAM estáticas y dinámicas

DRAM: contiene celdas con cargas eléctricas (condensadores). La presencia o ausencia de carga representa un 0 o un 1 Binario.

Requieren refrescos periódicos

Ej: Memoria Principal

SRAM: La diferencia con la DRAM es que guardan en compuertas biestables y guardará los datos siempre y cuando se mantenga alimentada

No se requieren refrescos periódicos

Ej: Memoria Cache

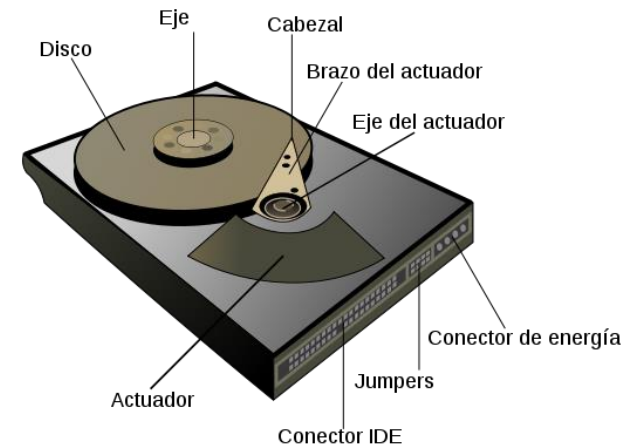
Memoria magnética

Las memorias magnéticas usan diferentes patrones de magnetización sobre una superficie cubierta con una capa magnetizada para almacenar información. Las memorias magnéticas son no volátiles. Se llega a la información usando uno o más cabezales de lectura/escritura. Como el cabezal de lectura/escritura solo cubre una parte de la superficie, el almacenamiento magnético es de acceso secuencial y debe buscar, dar vueltas o las dos cosas. En computadoras modernas, la superficie magnética será de alguno de estos tipos:

- **Disco magnético.**
- **Disco duro**
- **Cinta magnética**

Memoria magnética

La unidad de disco , es el dispositivo de almacenamiento de datos que emplea un sistema de grabación magnética para almacenar archivos digitales. Se compone de uno o más platos o discos rígidos, unidos por un mismo eje que gira a gran velocidad dentro de una caja metálica sellada. Sobre cada plato, y en cada una de sus caras, se sitúa un cabezal de lectura/escritura que flota sobre una delgada lámina de aire generada por la rotación de los discos. Es memoria no volátil.



Almacenamiento óptico

Son aquellos dispositivos capaces de guardar datos por medio de rayo láser, se almacenan por medio de ranuras microscópicas y la información queda grabada en la superficie de manera física. Solo el calor y las ralladuras pueden producir pérdida de datos. Se encuentran en discos

CD

Disco compacto. Un disco no borrable que almacena información de audio digitalizada. El sistema estándar usa discos de doce cm y puede grabar más de sesenta minutos de tiempo de ejecución ininterrumpido.

CD-ROM

Disco compacto de memoria de solo-lectura. Un disco no borrable usado como memoria de datos de un computador. El sistema estándar usa discos de doce cm y puede guardar más de 650 MB.

DVD

Disco versátil digital. Una tecnología para producir representación de información de vídeo digitalizada y comprimida, así como grandes cantidades de otros datos digitales. Se usan en formatos de ocho y doce cm de diámetro, con una capacidad con doble cara de hasta 17 GB. El DVD básico es de solo-lectura (DVD-ROM).

DVD-R

DVD grabable. Es similar al DVD-ROM. El usuario puede escribir en el disco solo una vez. Solo se utilizan discos de una cara.

DVD-RW

DVD grabable. Es similar al DVD-ROM. El usuario puede borrar y reescribir el disco varias veces. Solo se utilizan discos de una cara.

Modos de Direccionamiento u operación de los procesadores

MODO REAL

→ modo de trabajo desprotegido

MODO
PROTEGIDO

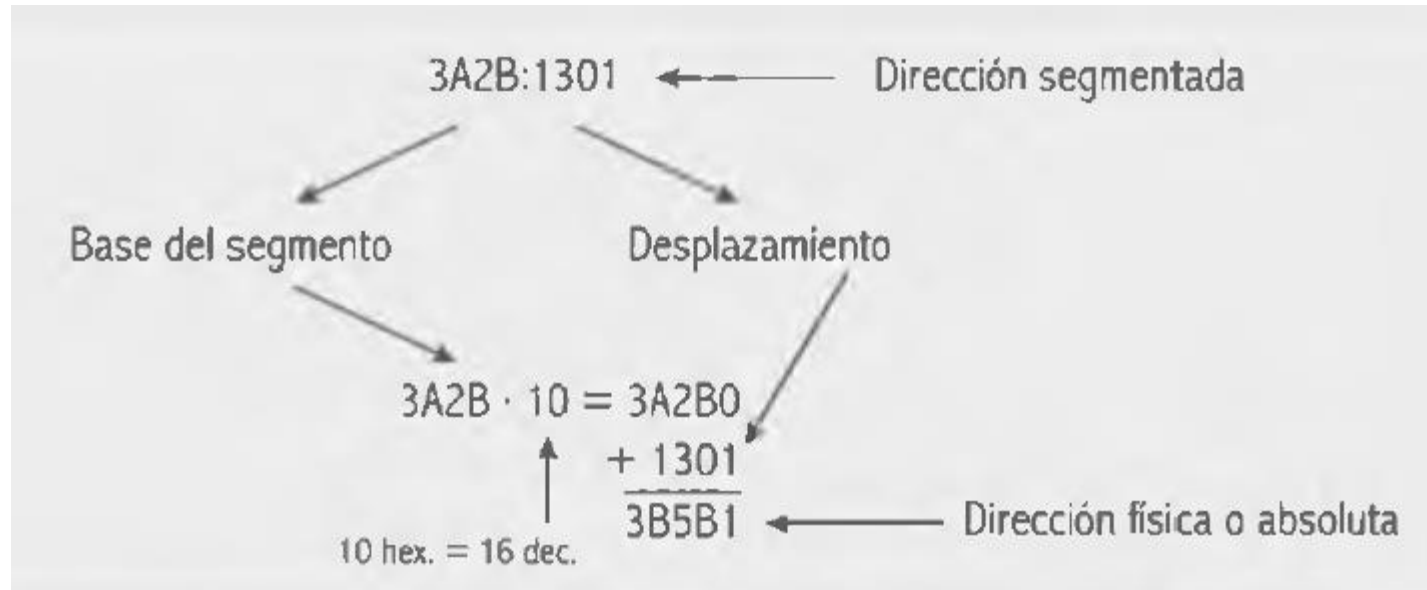
→ implica protección de la memoria

{
Espacio virtual o
lógico
Espacio lineal
Espacio físico

**No pueden estar
activados en
forma simultánea**

Modo Real

- Es un modo de operación del 80286 y posteriores cpu compatibles con la arquitectura x86
- Máxima cantidad de memoria direccionable: 1MB -> 20 bits
- No tiene conceptos de protección de memoria o multitarea a nivel de hardware
- Es Unitarea.
- No puede utilizar todos los recursos del CPU



Modo Protegido

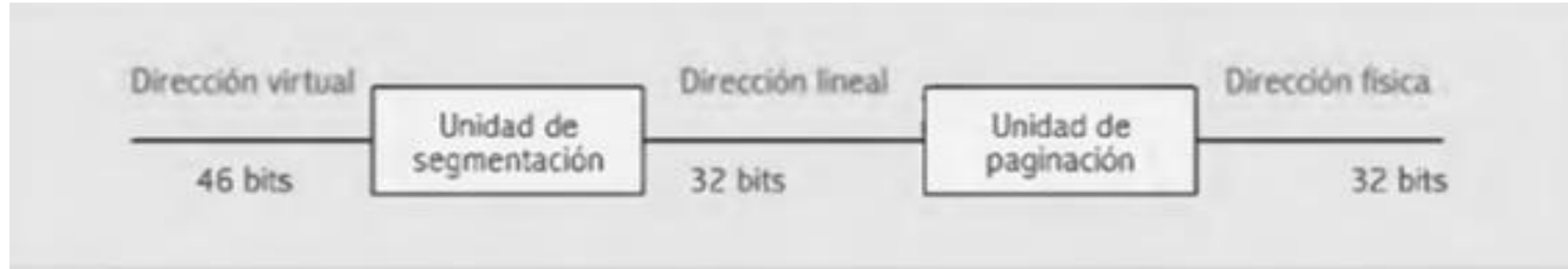
- El modo protegido es un modo operacional de los CPUs compatibles x86 de la serie 80286 y posteriores. El modo protegido tiene un número de nuevas características diseñadas para mejorar las multitareas y la estabilidad del sistema, como protección de memoria, y soporte de hardware para memoria virtual así como de conmutación de tareas.
- Trabaja con tablas de descriptores de Segmentos

Diferencias entre los modos

1. El modo protegido permite al sistema operativo utilizar características como memoria virtual, paginación y multitarea.
2. Todas las CPUs de la serie 80286 y posteriores empiezan en modo real al encenderse el computador.
3. La arquitectura 286 introdujo el modo protegido, permitiendo, entre otras cosas, la protección de la memoria a nivel de hardware.
4. El modo protegido requirió instrucciones de software adicionales no necesarias previamente.
5. Puesto que una especificación de diseño primaria de los microprocesadores x86 es que sean completamente compatibles hacia atrás con el software escrito para todos los chips x86 antes de ellos, el chip 286 fue hecho para iniciarse en 'modo real'.
6. Los sistemas operativos DOS (MS-DOS, DR-DOS, etc.) trabajan en modo real.

Direccionamiento

- Como se Relacionan estos 3 tipos de direccionamiento.



VIRTUAL

Abarca toda la dimensión de la memoria virtual y es el que maneja el programador de aplicaciones.

Las direcciones se componen de un segmento (segment) y un desplazamiento (offset). Segmentación de Memoria

LINEAL

Permite referenciar a 2 a la 32 = 4GB de memoria

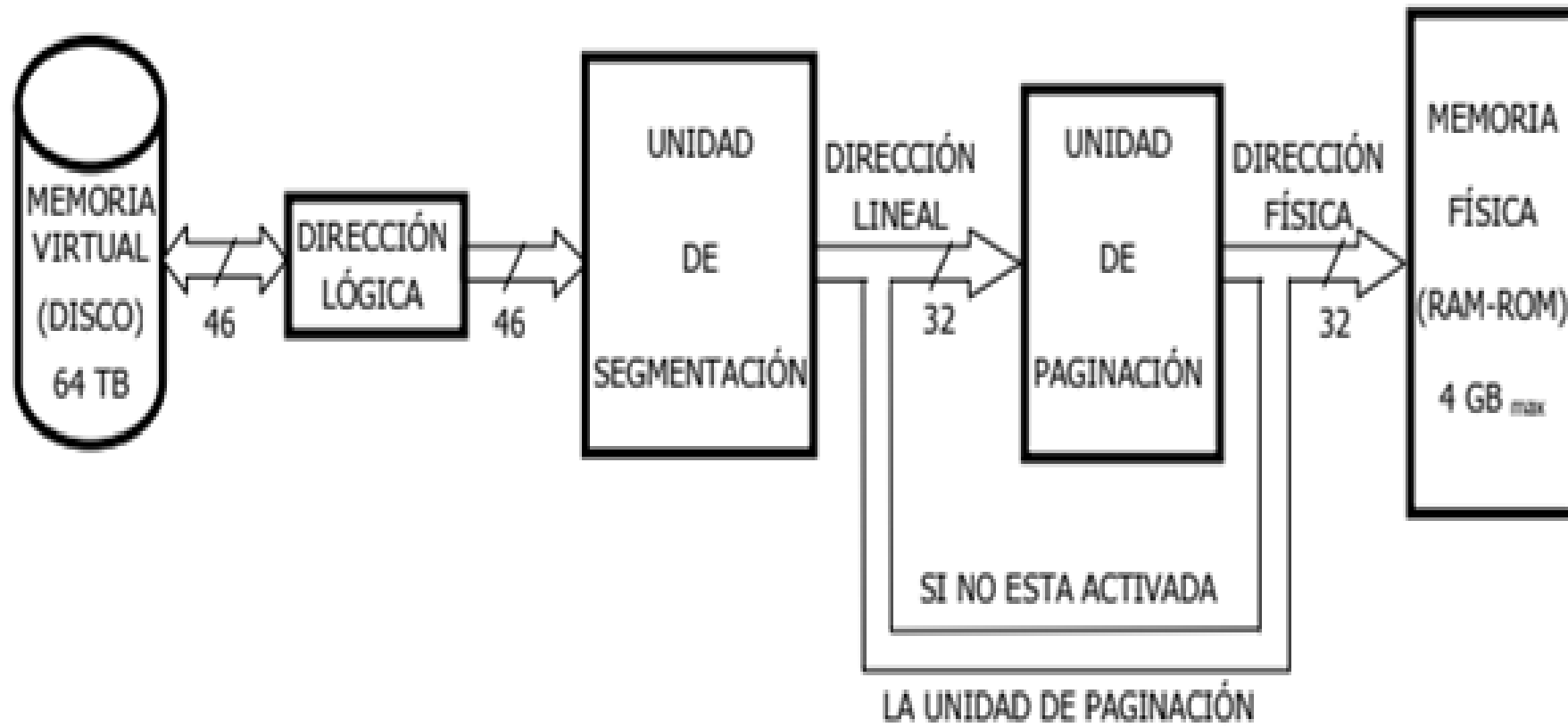
Tienen dispuestas todas sus posiciones en orden consecutivo o lineal.
Segmentación y Paginación

FÍSICO

Utilizado para acceder a las celdas de memoria RAM del sistema.

Para acceder a la memoria principal ligada directamente a la CPU, a la memoria física(EJ:RAM)

Memoria en modo protegido: espacios



Memoria en modo protegido: espacios

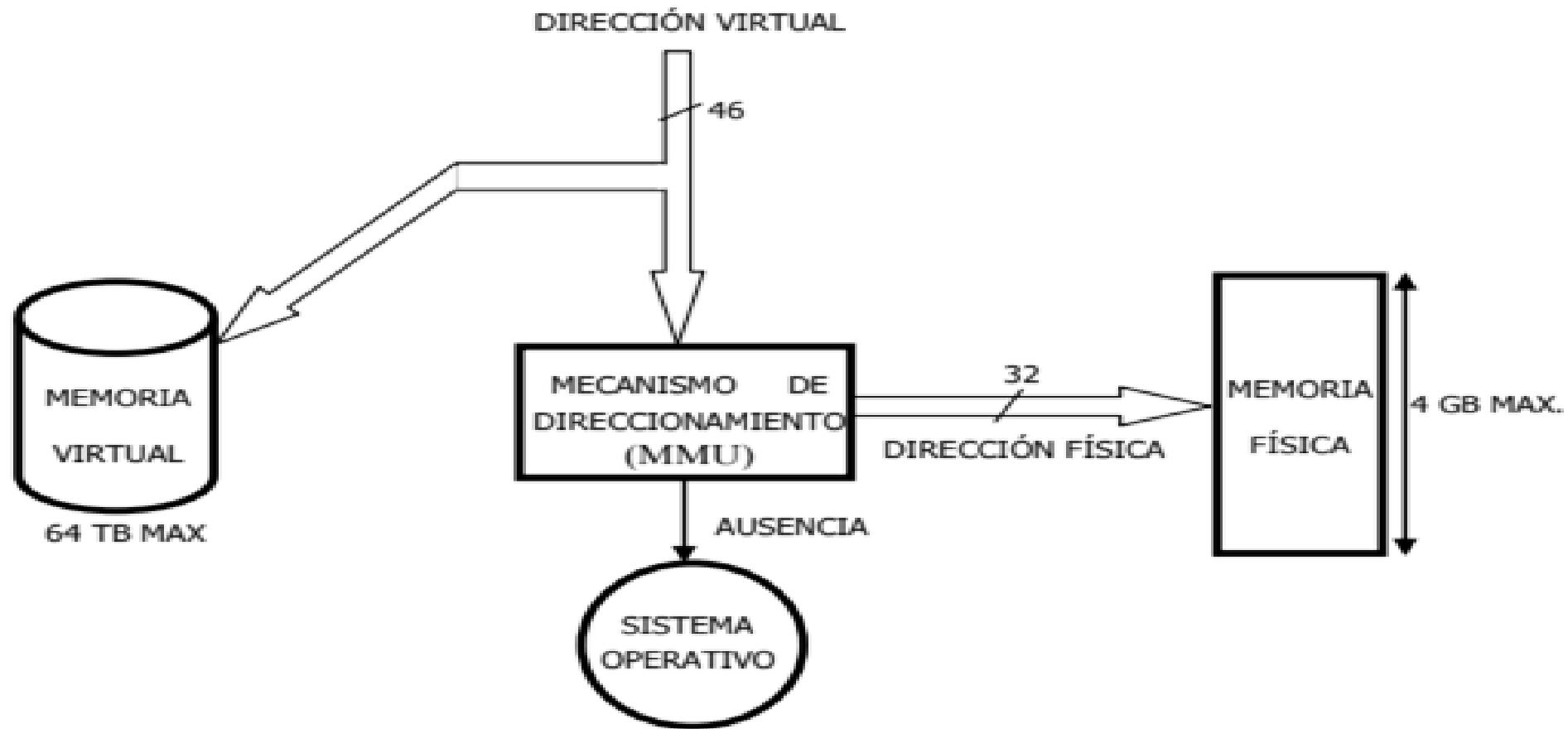


Figura 8.4. El mecanismo de direccionamiento traduce la dirección virtual a física, y, caso de no hallarse el elemento en la memoria, se detecta su 'ausencia' y se genera una excepción encargada de hacer la transferencia oportuna que es soportada por el Sistema Operativo.

Memoria virtual

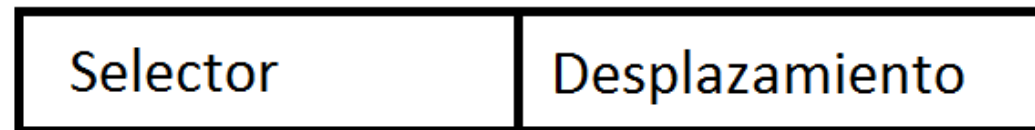
Forma de administrar la memoria para asignar mas de ella a cada tarea, utilizando almacenamiento en el disco, que suele ser el dispositivo de acceso directo mas rápido; así se simula que se dispone de una memoria mucho más amplia que la que el sistema tiene en realidad.

Dirección virtual o lógica

La dirección lógica o virtual se ve caracterizada por dos partes:

Selector: Selecciona un determinado segmento de la memoria virtual.

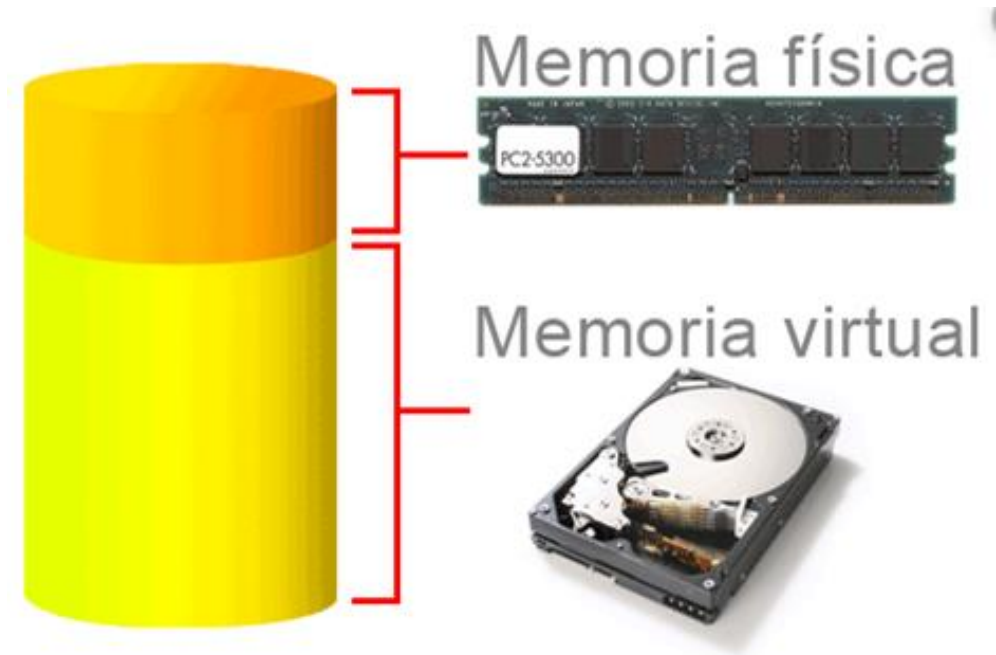
Desplazamiento: Determina la posición dentro del segmento.



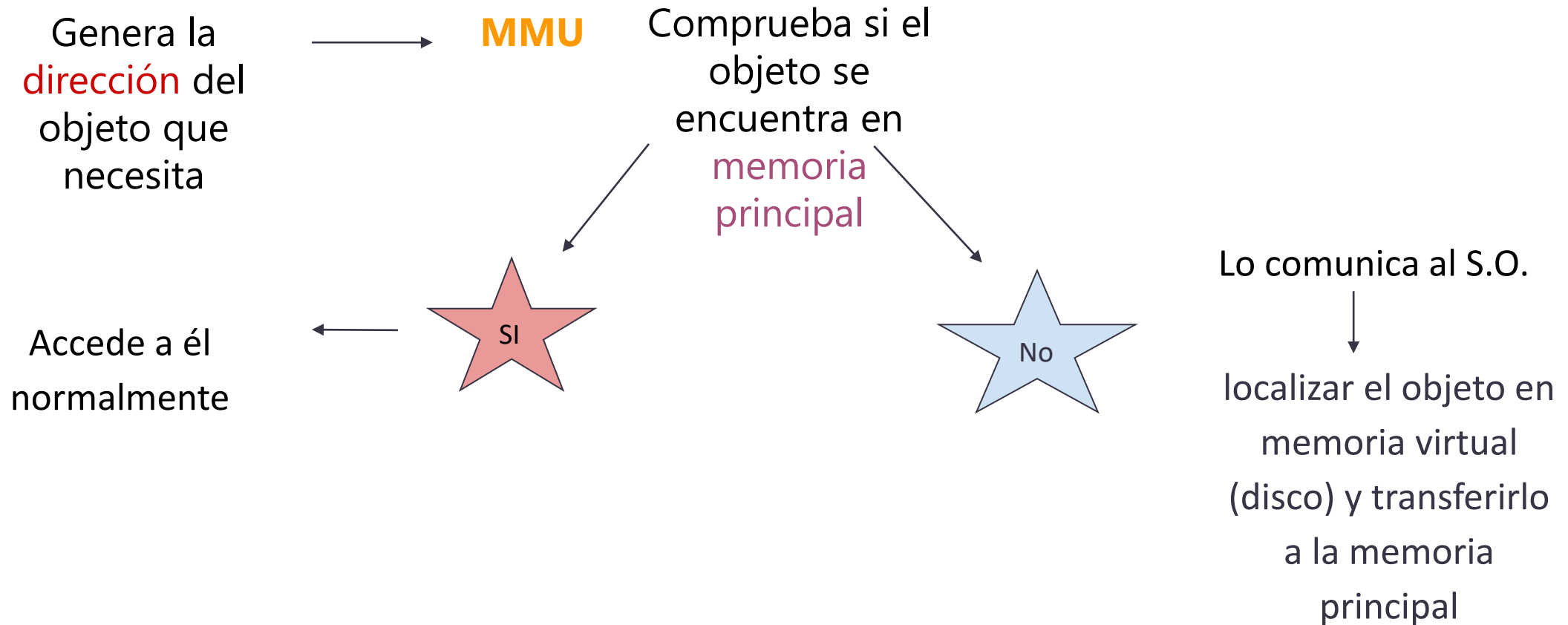
Memoria Virtual

- Método de **organización y gestión** de la memoria.
- Proporciona al programador de aplicaciones un **espacio mucho mayor** de que dispone físicamente.
- Da la **ilusión** a la CPU de que puede manejar mucha más memoria de la que la DRAM (memoria electrónica) tiene a su disposición.
- **Dirección virtual**: dirección lógica perteneciente al espacio de direccionamiento virtual

└─ debe traducirse ("mapeo") a dirección física



Funcionamiento de memoria virtual



Formas de organizar la Memoria Virtual (M.V.)

SEGMENTACIÓN

→ División de la memoria virtual en "trozos" (segmentos) de **tamaños variables**.

PAGINACIÓN

→ División del espacio de direcciones de cada proceso en bloques de **tamaño uniforme** llamados páginas

TIPOS



M.V.
segmentada

M.V.
paginada

M.V. con
segmentos
paginados

Criterios para fraccionar memoria

1. **FIFO** (First In – First Out): Se sustituye la fracción que más tiempo lleva en la memoria principal.
2. **LRU** (Last Recently Used): La porción que lleva en la memoria más tiempo sin haber sido usada.
3. **LIFO** (Last In – First Out): Se sustituye la fracción que menos tiempo lleva en la memoria principal.
4. **LFU** (Last Frequently Used): La porción que se accedido menos veces desde que se inició el proceso.
5. **RAND** (Random): Se elige una porción al azar.
6. **CLOCK**: Cuando se coloca un bit de uso en cada entrada de una cola FIFO y se establece un puntero que se convierte en circular.

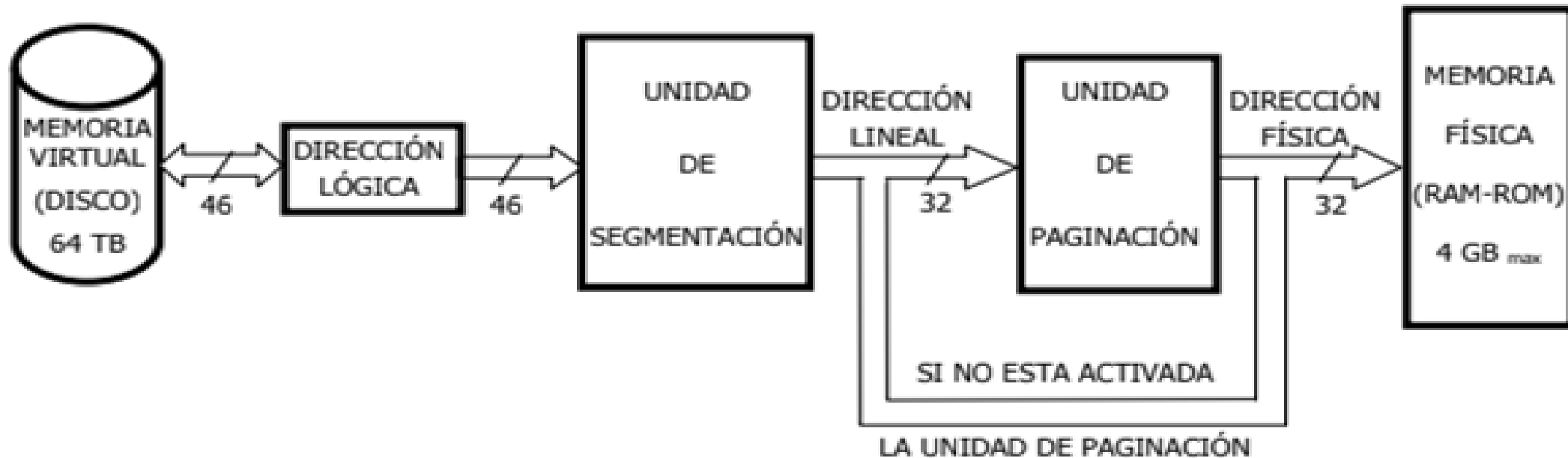
Segmentación de memoria

Se encarga de dividir la memoria en **segmentos**, cada uno de los cuales tiene una **longitud variable**, que está definida por el tamaño de ese segmento del programa

La Segmentación de memoria es un esquema de manejo de memoria mediante el cual la estructura del programa refleja su división lógica; llevándose a cabo una agrupación lógica de la información en bloques de tamaño variable denominados segmentos.

Cada uno de ellos tienen información lógica del programa: subrutina, arreglo, etc. Luego, cada espacio de direcciones de programa consiste de una colección de segmentos, que generalmente reflejan la división lógica del programa.

Esquema de las unidades de la MMU



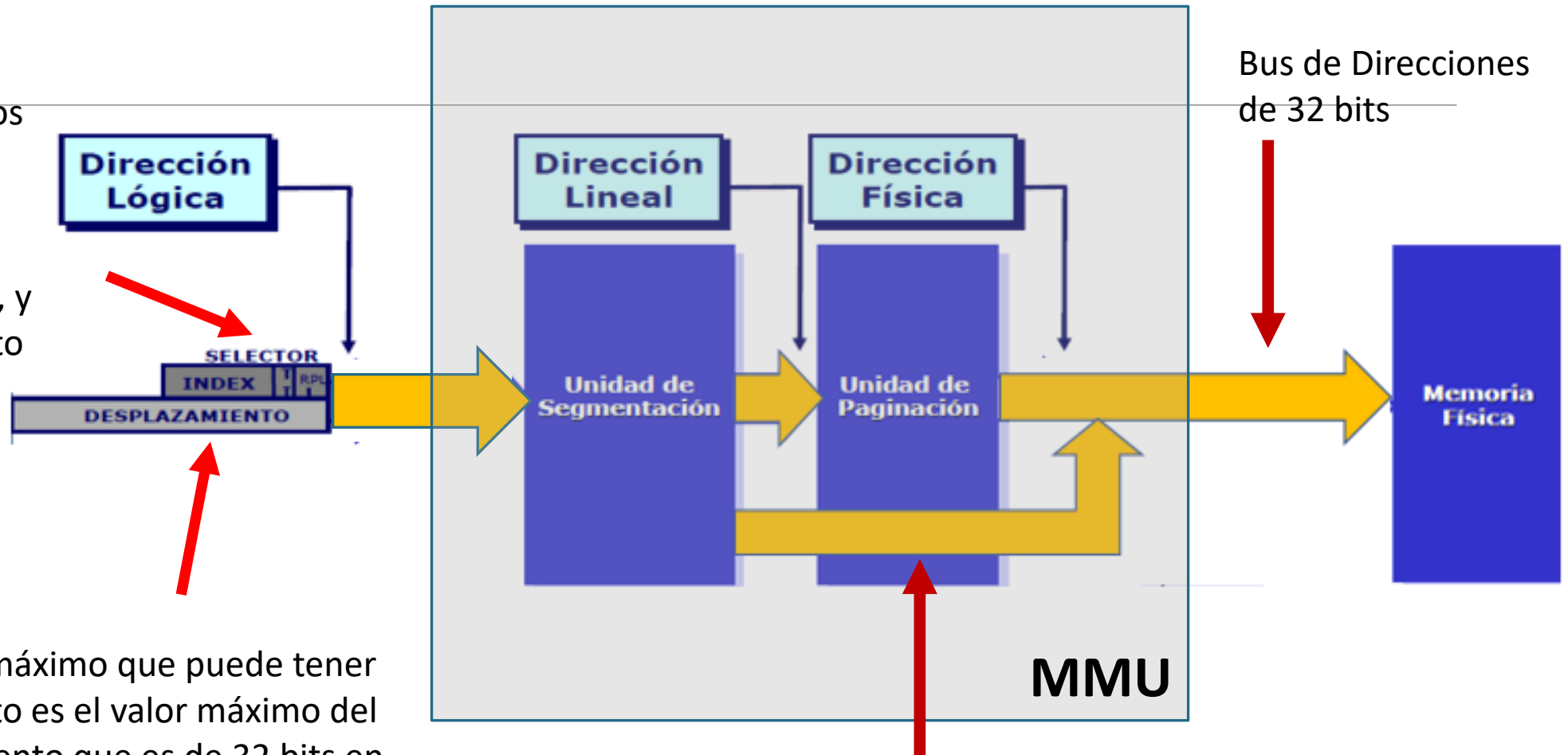
Segmentación: provee un mecanismo para separar el código, datos y el stack

Paginación: provee un mecanismo para implementar un sistema de memoria virtual de paginación bajo demanda

Esquema de las unidades de la MMU (IA- 32)

Los registros de segmentos de 16 bits se transforman en **selectores** (índices) de tablas que contienen la **Dirección Base**, el **Límite**, y los **Atributos** del segmento seleccionado. Esta estructura se denomina **Descriptor de Segmento**

El tamaño máximo que puede tener un segmento es el valor máximo del desplazamiento que es de 32 bits en modo protegido



La Unidad de Paginación es opcional, si no está activada la Dirección Lineal es igual a la Dirección Física

Segmentos de memoria

El segmento de código (tiene como base el contenido del registro CS)

En este segmento se encuentran las instrucciones que forman parte del programa. Para acceder a los datos contenidos en él, se usa el registro IP como desplazamiento.

El segmento de datos (tiene como base el contenido del registro DS)

Contiene los datos que utiliza el programa (variables, etc). Para acceder a los datos contenidos en él, se suelen usar los registros SI y DI como desplazamiento.

Segmentos de memoria

El segmento de pila (tiene como base el contenido del registro SS)

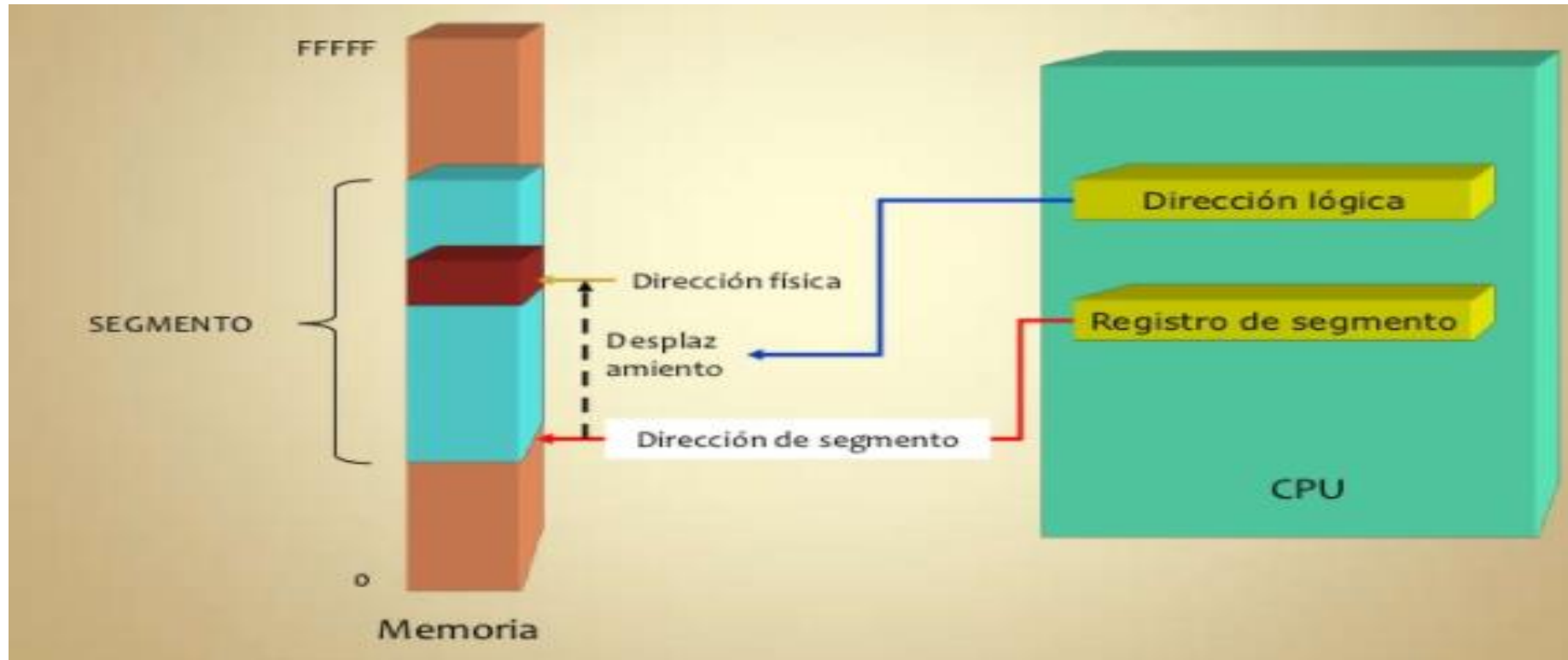
En él se desarrolla la pila del programa, utilizada para almacén temporal de datos, llamadas a funciones, etc. Debe estar presente en todos los programas EXE de forma obligada. Se utiliza el registro SP para acceder a los datos de este segmento.

El segmento extra (tiene como base el contenido del registro ES)

Su uso es opcional, y en él se encuentra un segmento definido por el usuario y que, regularmente, contiene datos adicionales. Al igual que ocurre con el segmento de datos, para acceder a los datos contenidos en él, se suelen utilizar los registros SI y DI

En este segmento se encuentran las instrucciones que forman parte del programa. Para acceder a los datos contenidos en él, se usa el registro IP como desplazamiento

Segmentación



Ejemplo: algoritmo de cálculo que realiza la MMU (Memory Manager Unit) del procesador en modo real

Unidad de segmentación

- ★ Activación obligada
- ★ Traduce las direcciones virtuales a lineales
- ★ El **direccionamiento** en el espacio lógico es realizado a través de un **selector de segmento y un desplazamiento**.
- ★ Las tablas pueden contener hasta 8192 (2 a la 13) entradas de descriptores.
- ★ Hay dos registros que indican el comienzo de las tablas (gdtr y ldtr).

¿Como se divide el segmento?

Selectores y descriptores

Hay dos tablas de descriptores que se utilizan con los registros de segmento: una contiene descriptores globales y la otra contiene descriptores locales.

Los **descriptores globales** contienen definiciones de segmentos que se aplican a todo los programas, mientras que los **descriptores locales** son por lo general únicos para una aplicación.

Al descriptor global lo podemos llamar **descriptor de sistema** y al descriptor local lo podemos llamar **descriptor de aplicación**.

Cada tabla contiene 8192 descriptores, por lo que hay un total de 16,384 descriptores disponibles para una aplicación en un momento dado.

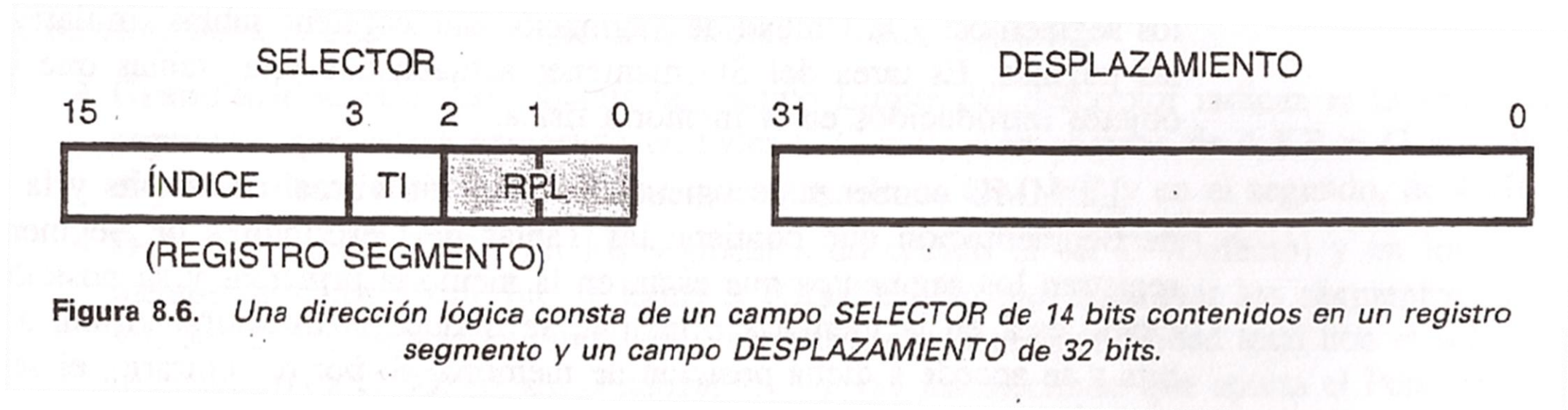
Como el descriptor describe un segmento de memoria, esto permite que se describan hasta 16,384 segmentos de memoria para cada aplicación. Como un segmento de memoria puede tener hasta 4 Gbytes de longitud, esto significa que una aplicación podría tener acceso a $4 \text{ G} \times 16,384 \text{ bytes}$ de memoria, o 64 Tbytes.

¿Como se divide el segmento?

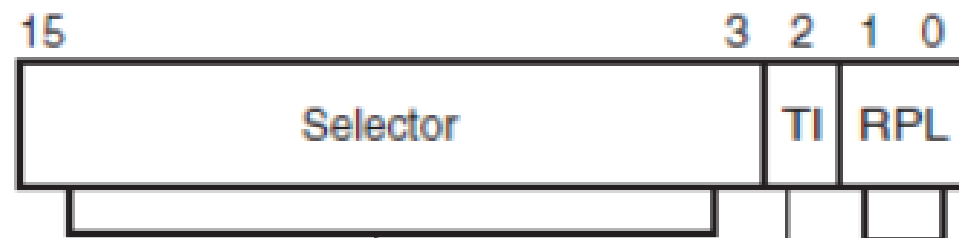
Ejemplo:

- El campo selector está compuesto de 3 campos (16 bits)
 - Índice (13 bits): es el número de entrada en una tabla de descriptores de segmentos.
 - Indicador de tabla (1 bit): define si es sobre la tabla global (GDT – Global Descriptor Table) o local (LDT – Local Descriptor Table).
 - Nivel de privilegio (2 bits): modo en el q estoy trabajando (Real o Protegido)
- El desplazamiento(16 bits)

Unidad de Segmentación



Modo Protegido



RPL = nivel de privilegio solicitado,
en donde 00 es el mayor y 11 es el menor

TI = 0 Tabla de descriptores globales
TI = 1 Tabla de descriptores locales

Selecciona uno de 8192 descriptores, ya sea en la tabla
de descriptores globales o en la de descriptores locales

Descriptor del 80386 al Pentium 4

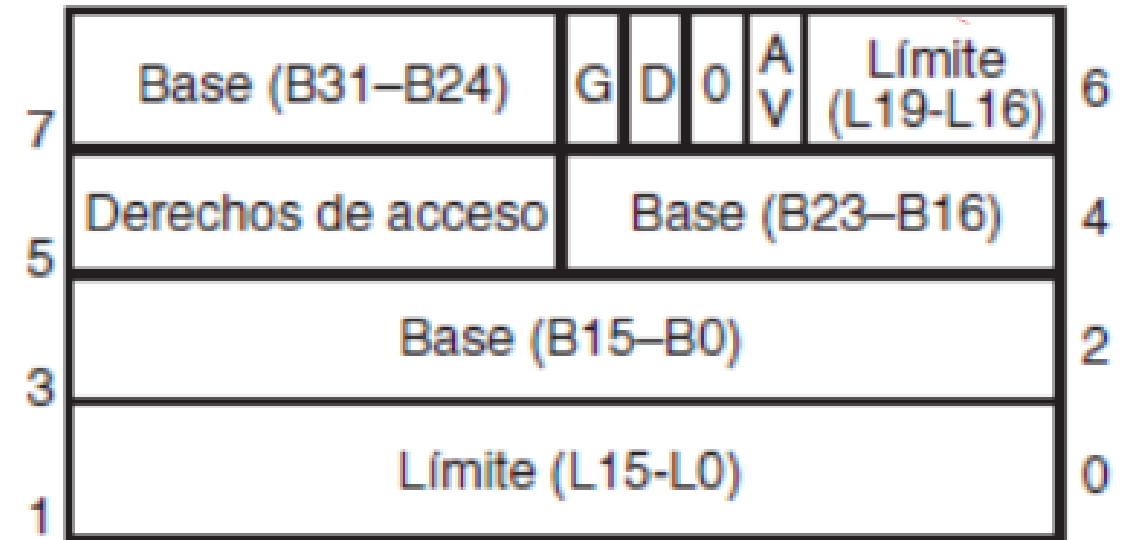
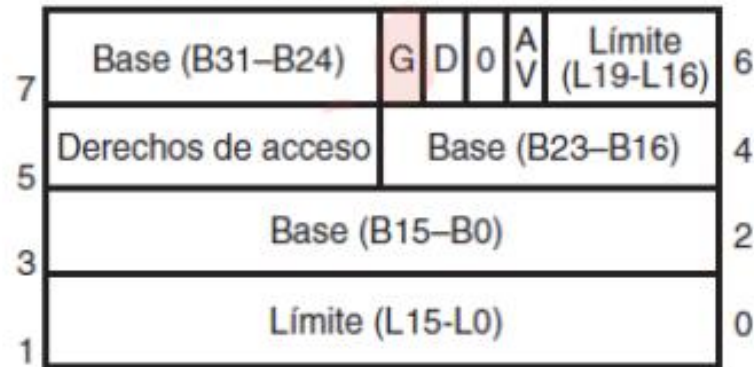


Tabla de Descriptores de Segmentos

- En el sistema hay dos tipos de tablas: GDT y LDT
- La GDT es única para todo el sistema (Intel)
- La dirección base de GDT está almacenada en el registro GDTR
- El GDTR tiene dos partes: Dirección Lineal Base y límite
- En el sistema puede haber mas de una LDT, pero sólo una activa
- Las LDT están descriptas en la GDT
- La dirección base de la LDT está almacenada en el registro LDTR
- El LDTR es un índice a la GDT

Descriptor del 80386 al Pentium 4

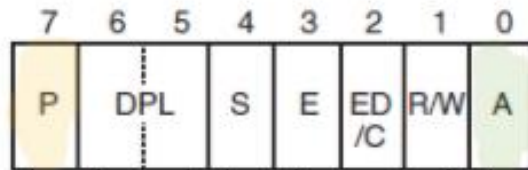


G: Granularidad 0=bytes 1=pág 4K

D: 0=seg. 16bits 1=seg. 32bits

O: seg. code 64bits (IA-32e)

AV: Disponible para ser usado por software del sistema



A = 0 No se ha accedido al segmento
A = 1 Se ha accedido al segmento

E = 0 El descriptor describe un segmento de datos
ED = 0 El segmento se expande hacia arriba (segmento de datos)
ED = 1 El segmento se expande hacia abajo (segmento de pila)
W = 0 No pueden escribirse datos
W = 1 Pueden escribirse datos

E = 1 El descriptor describe el segmento de código
C = 0 Se ignora el nivel de privilegio del descriptor
C = 1 Se obedece el nivel de privilegio
R = 0 No puede leerse el segmento de código
R = 1 Puede leerse el segmento de código

S = 0 Descriptor del sistema
S = 1 Descriptor de segmento de código o de datos

DLP = Establece el nivel de privilegio del descriptor

P = 0 El descriptor está indefinido
P = 1 El segmento contiene una base y un límite válidos

Nota: Algunas de las letras utilizadas para describir los bits en los bytes de derechos de acceso varían en la documentación de Intel.

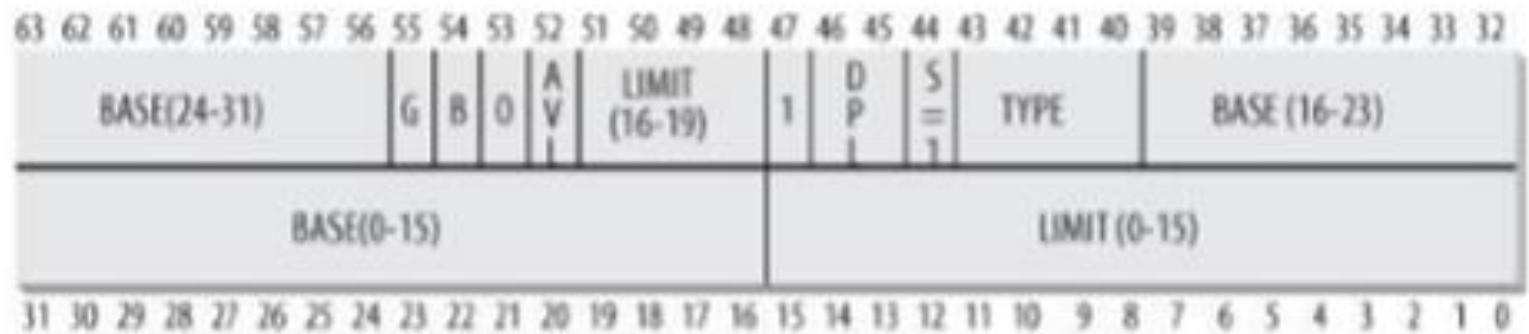
Descriptores de segmentos

Hay diferentes tipos de segmentos:

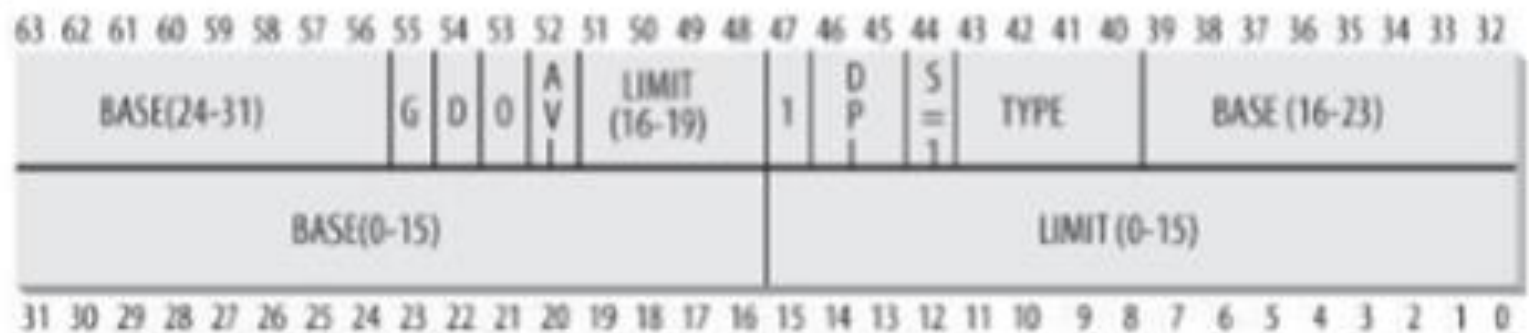
- Datos (DSD - Data) (GDT o LDT).
- Código (CSD - Code) (GDT o LDT).
- Estado de la tarea (TSSD - Task State) (GDT).

Contienen la dirección base y el límite del segmento en el espacio lógico

Data Segment Descriptor



Code Segment Descriptor



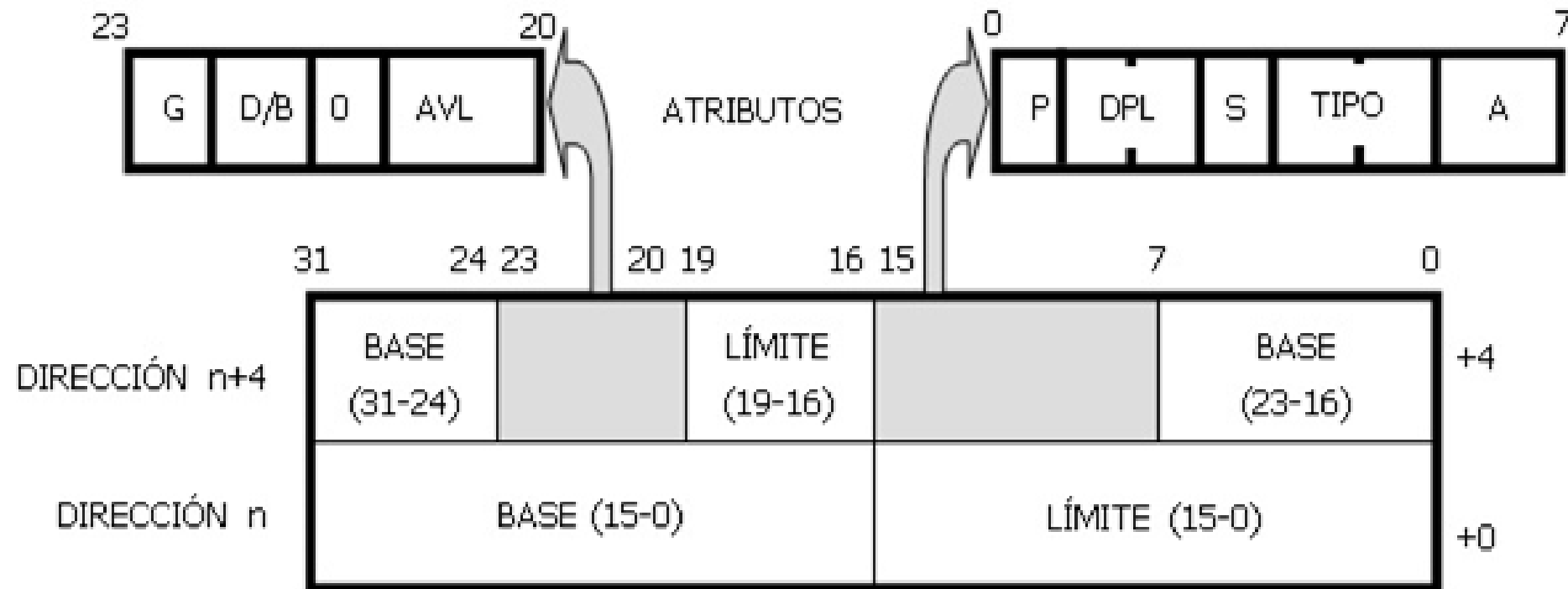
Parámetros que determinan un segmento

1. **Base**: Campo de 32 bits que contiene la dirección lineal donde comienza el segmento.
2. **Límite**: Campo de 20 bits que expresa el tamaño del segmento. Como con 20 bits el tamaño máximo es de 1 MB, hay otro bit complementario en el campo de atributos, llamado de **granularidad** que indica si el límite está expresado en bytes (G=0) o en páginas de 4 KB (G=1).
3. **Atributo o derechos de acceso**: campo de 12 bits, que proporciona las características relevantes del segmento.

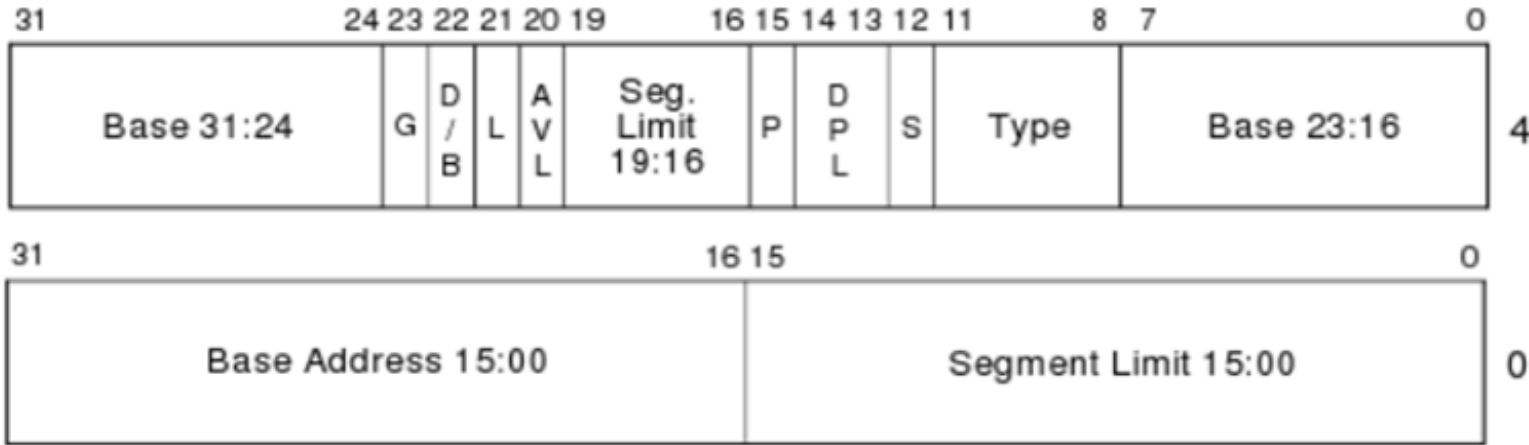
Los campos de Descriptores de Segmentos

- **Bit de presencia (P):** indica si el segmento al que hacen referencia esta en memoria.
- **Nivel de privilegio (DPL):** es un campo que consta de 2 bits indica el privilegio de dicho segmento.
- **Clase de segmento(S):** indica si es un segmento normal (de pila, dato, de código) o si es un segmento de sistema.
- **Tipo:** distingue si los segmentos normales son de código, de dato o de pila.
- **Accedido (A):** indica si el segmento fue accedido. El SO lo utiliza para tener una cuenta de cuantas veces se accedió para luego utilizar el algoritmo LRU.
- **Granularidad (G):** indica si el campo limite es expresado en unidades de Bytes o paginas de 4 KB.
- **Defecto / grande (D/B):** utilizado para la compatibilidad con el 8086.
- **Disponible (AVL):** a disposición del programador de SO.

Formato de un descriptor de segmento compuesto por 8 bytes,
que ocupan dos posiciones de memoria de 32 bits cada una:



Estructura de un descriptor de 32 bits.



- L — 64-bit code segment (IA-32e mode only)
- AVL — Available for use by system software
- BASE — Segment base address
- D/B — Default operation size (0 = 16-bit segment; 1 = 32-bit segment)
- DPL — Descriptor privilege level
- G — Granularity
- LIMIT — Segment Limit
- P — Segment present
- S — Descriptor type (0 = system; 1 = code or data)
- TYPE — Segment type

Función de los Bits del Campo de Atributos

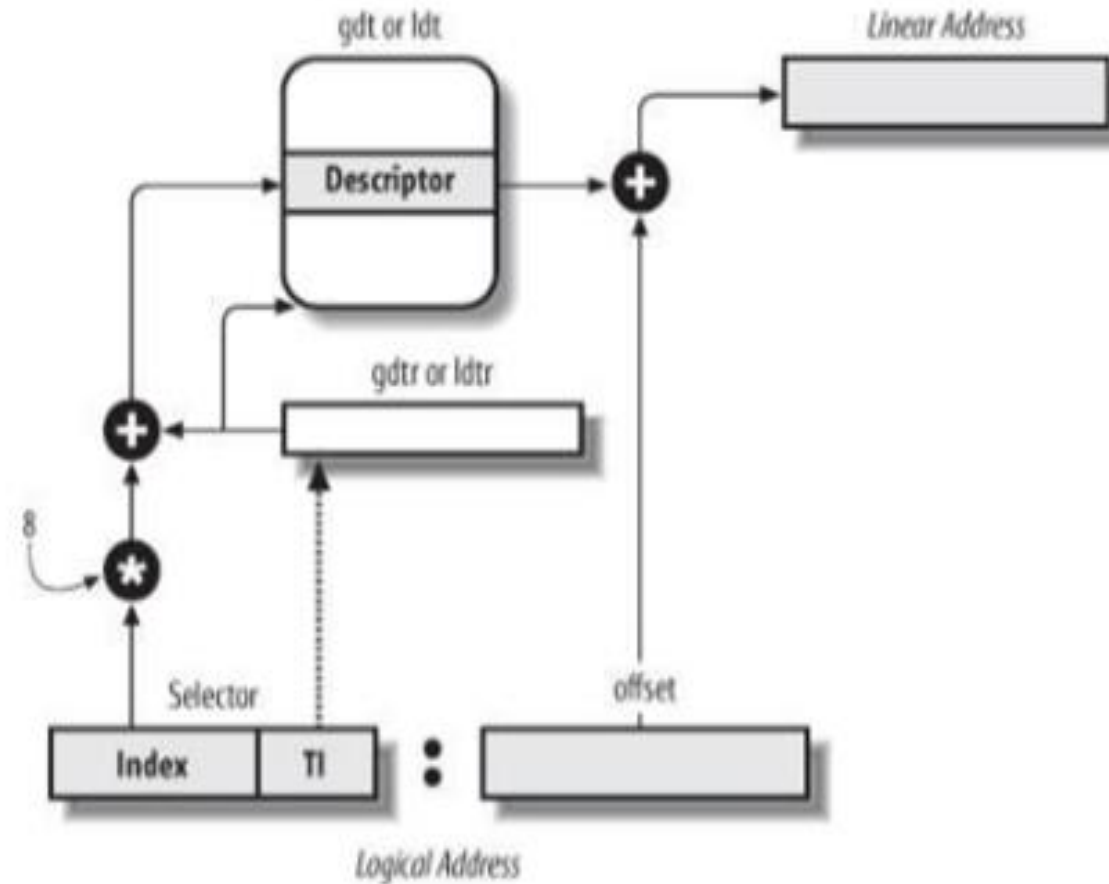
- ★ **Bit de presencia (P):** Indica si el segmento al que referencia el descriptor está cargado, o sea, se halla presente en la memoria principal (P=1), o bien, está ausente (P=0).
- ★ **Nivel de privilegio (DPL):** Indica el nivel de privilegio del segmento al que referencia el descriptor. Su valor puede variar entre el 0 y el 3 y es un campo que consta de dos bits.
- ★ **Tipo de segmento (S):** Si S=1, el segmento correspondiente al selector es “normal”, o sea, un segmento de código, de datos o de pila. Si S=0, se refiere a un segmento del sistema, que referencia un recurso especial del sistema.
- ★ **Tipo:** Los tres bits de este campo distinguen en los normales si se trata de uno de código, de datos o de pila. Además, determinan el acceso permitido: lectura/escritura/ejecución.
- ★ **Accedido (A):** Este bit se pone automáticamente a 1 cada vez que el procesador accede al segmento.

Función de los Bits del Campo de Atributos

- ★ **Granularidad (G):** Los 20 bits del campo LÍMITE del descriptor indican el tamaño del segmento, que estará expresado en bytes si G=0 (tamaño del segmento 1MB), y en páginas de 4 KB si G=1 (4 GB).
- ★ **Defecto/grande (D/B):** En los segmentos de código el bit D (Defecto) y en los segmentos de datos de este mismo bit, llamado B (Grande), permiten distinguir los segmentos nativos de 32 bits para el Pentium. Si D=1 tanto las direcciones efectivas como los operandos son de 32 bits. Si D=0 se toman sólo 16. (es posible manejar conjuntamente segmentos pertenecientes al 80286 con otros del Pentium)
- ★ **Disponible (AVL):** bit en disposición del usuario para poder diferenciar ciertos segmentos que contengan un tipo determinado de información o que cubran alguna función específica.

Traducción de direcciones lógicas a lineales

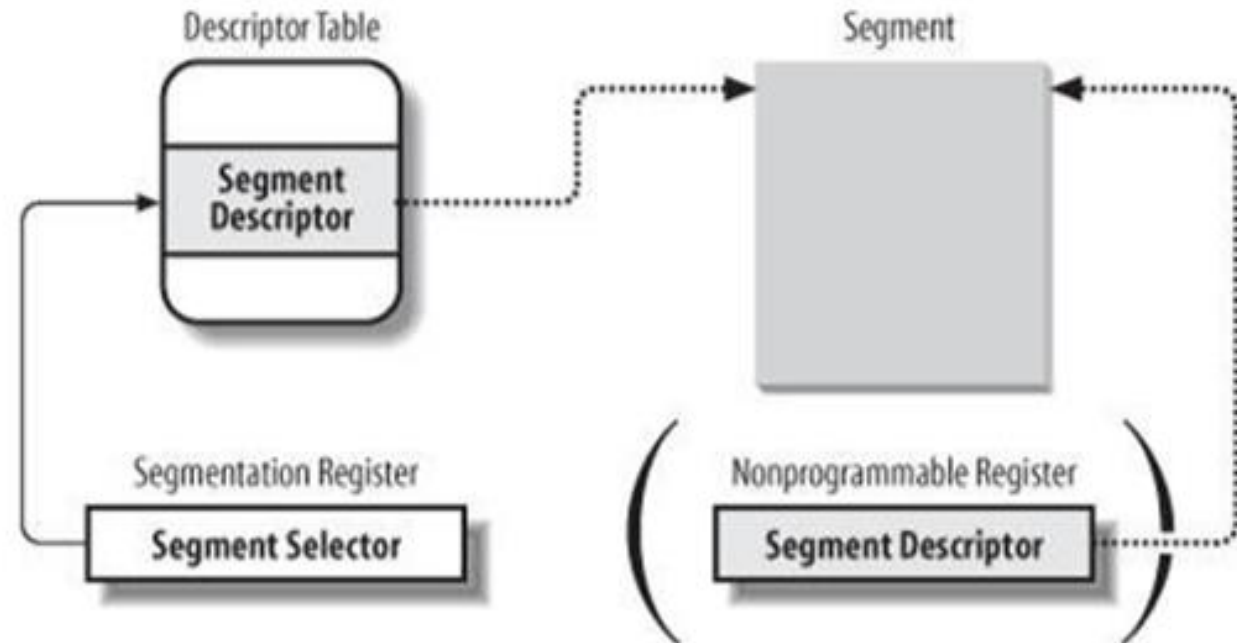
- El selector determina el tipo de tabla (gdt o ldtr).
- Se selecciona el descriptor correspondiente al índice
- Para poder posicionarse en un descriptor de segmento se calcula así:
$$\text{Dirección} = (\text{LDTR o GDTT}) + \text{INDICE} * 8$$
- El 8 es por q se le agregan 3 ceros ,1 bit es usado para saber a q tabla corresponde (GDT o LDT) y 2 bits para nivel de privilegio.
- Se selecciona el descriptor correspondiente con la dirección.
- A partir del descriptor se toma la dirección base y se le suma el desplazamiento.
- Es necesario 2 accesos a memoria.



Traducción de direcciones lógicas a lineales

Intel agrega registros no programables (cargados por la arquitectura) que contienen los descriptores de segmentos.

Esto permite que la traducción se haga en un solo acceso.



GRACIAS!