

MEMORIA

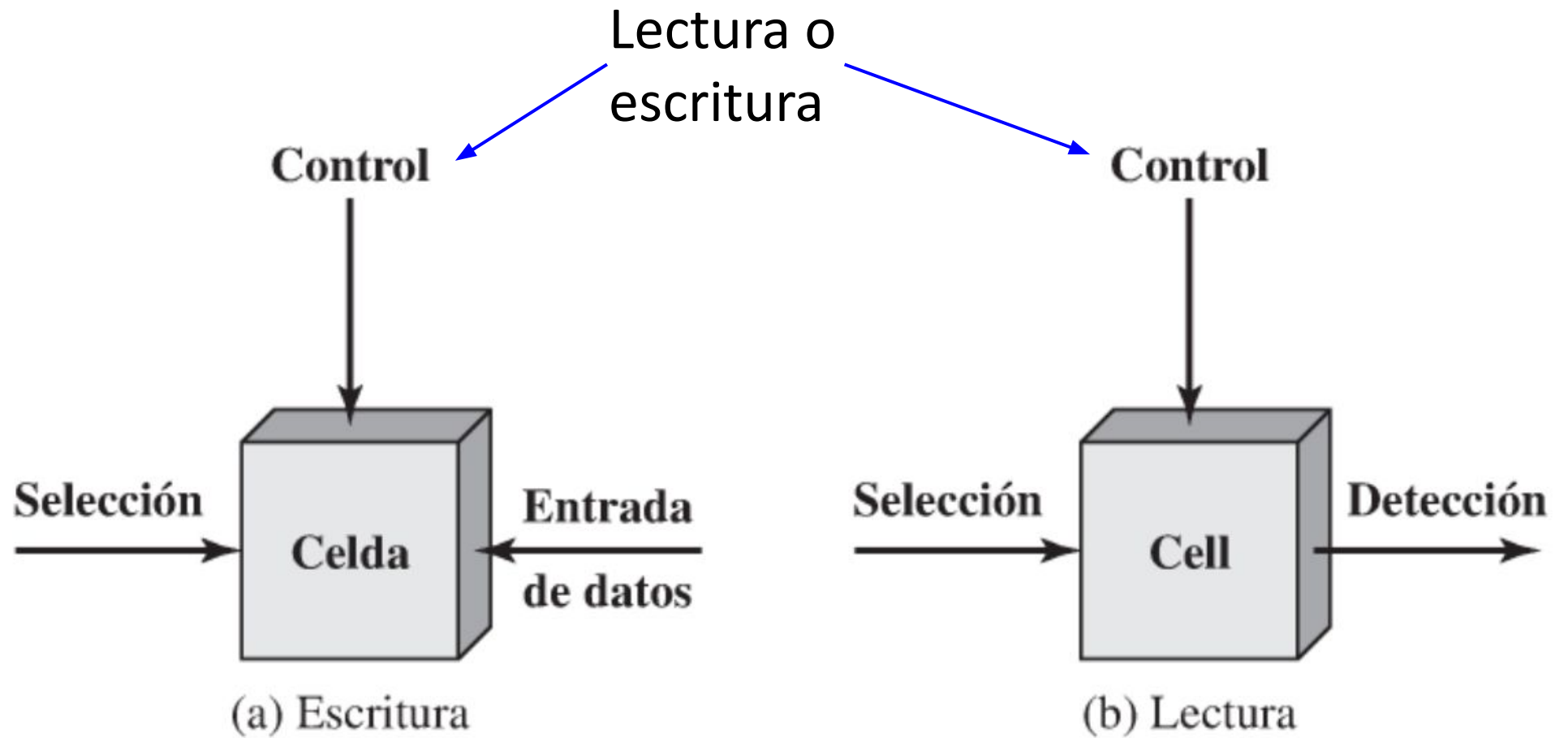
DEFINICIONES:

PUNTO DE MEMORIA (unidad mínima): “celda de bit”.

DIRECCIÓN DE MEMORIA: Etiqueta que permite identificar a cada celda.

DIRECCIONAMIENTO: Es el procedimiento por el cual es posible ubicar las posiciones de memoria para efectuar su lectura o su escritura.

ACCESO A MEMORIA: Se conoce con esta denominación al proceso genérico de lectura/escritura en memoria.



Buses

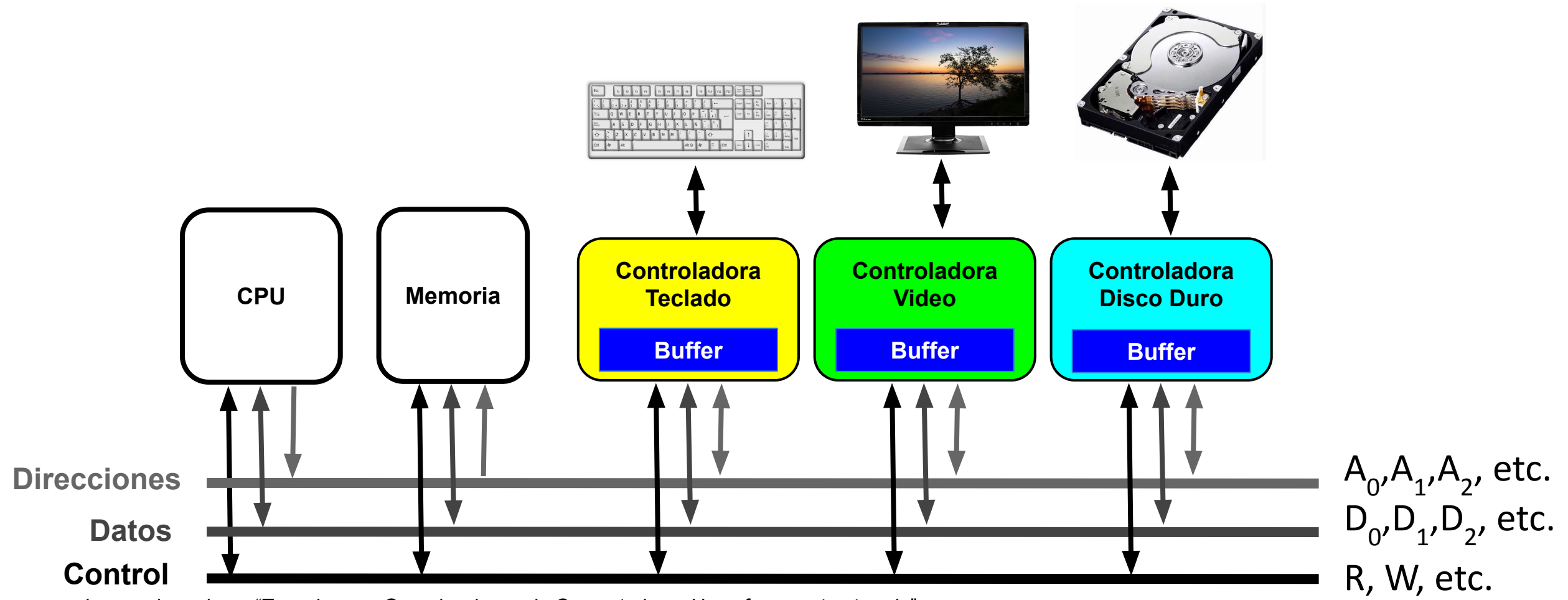


Imagen basada en "Tanenbaum - Organizaciones de Computadoras Un enfoque estructurado"

TIEMPO DE ACCESO: Desde que se indica una posición de memoria hasta que el dato está disponible para ser leído, o ha sido escrito.

$$t_a = \frac{1}{2} (t_l + t_e)$$

Donde: t_a = tiempo de acceso.
 t_l = tiempo de lectura.
 t_e = tiempo de escritura.

CICLO DE MEMORIA:



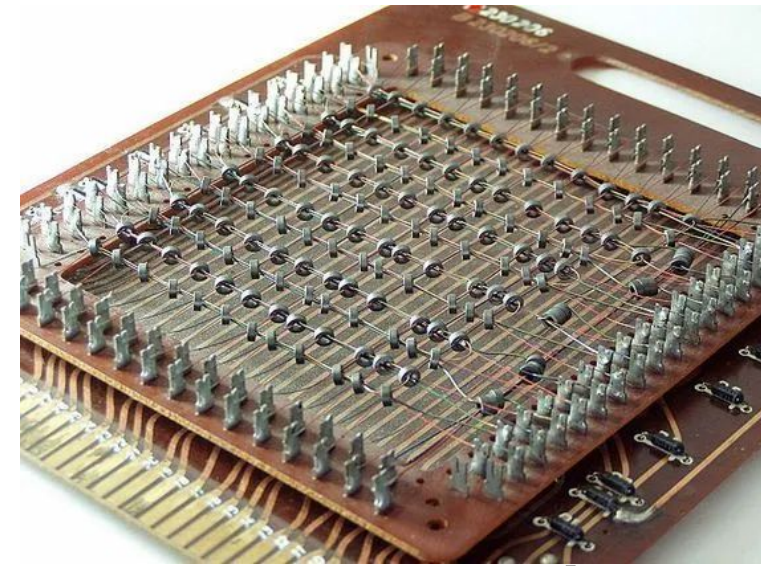
Tiempo de ciclo de memoria: Tiempo desde que comienza un acceso hasta que puede realizarse el siguiente acceso.

Palabras, Bytes, unidad direccionable.

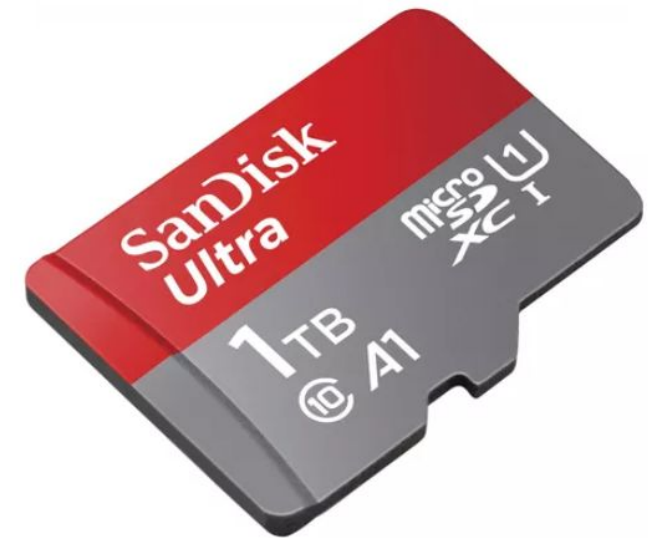
- **Palabra:** Unidad de datos básica con los que trabaja una computadora.
 - Coincide con el número de bits de los **registros** del procesador.
 - Suele coincidir con el número de bits del bus de datos.
 - Valores usuales: computadoras de escritorio y notebooks: **64 bits, 32 bits**. El ATmega 328P (Arduino UNO) es de 8 bits.
- **Unidad direccionable:** Unidad de datos que pueden direccionarse (cada uno posee una dirección)
 - En la gran mayoría de las computadoras de escritorio o notebook: **byte**.
- **Unidad de transferencia:** bits que pueden leerse o escribirse a la vez.
 - En la memoria principal suele coincidir con el tamaño de una palabra.
 - En discos duros suele ser mucho mayores, llamados bloques.

DIMENSIONES DE LAS MEMORIAS:

- **CAPACIDAD**, Cantidad de información que puede almacenar. Se mide en bits, bytes o palabras, y múltiplos (kilo, mega, giga). Ejemplo: 64 Megabytes.
- **CAUDAL**, Cantidad de información que puede transferir por unidad de tiempo, medida en bits o bytes por segundo. Ejemplo: 134 Gigabytes/segundo. También llamado **ancho de banda**.
- **DENSIDAD**, Puede ser lineal, superficial o volumétrica, y permite determinar la cantidad de información, en bits o bytes que puede ser almacenada por unidad de longitud, de superficie, o de volumen.
Ejemplo: 300 Kilobytes/cm², 250 Megabytes/cm³.



**Memoria magnética.
4KB por m³. 1960.**



**Memoria flash. Varios
GB por m³. Actual**

Clasificaciones

Según **tipo de acceso**:

- **Acceso Secuencial**: debe pasar desde la posición actual a la deseada, pasando por todas las posiciones intermedias.
 - Tiempo de acceso **variable**.
 - **Ejemplo: Cintas magnéticas.**
- **Acceso Aleatorio**: Cualquier posición puede seleccionarse aleatoriamente y direccionarse directamente. (**RAM: Random Access Memory**)
 - Tiempo de acceso **constante** e **independiente de la posición**.
 - **Ejemplo: memoria principal de una computadora.**
- **Acceso asociativo**: Se accede por “similitud”. Se proporciona parte de un dato, la memoria busca palabras que contengan la información proporcionada.
 - Tiempo de acceso **constante**.
 - **Ejemplo: Redes neuronales. Algunas cachés.**
- **Acceso cíclico**: El acceso es en ciclos (realiza ciclos, regresando al comienzo luego de cada ciclo). Discos duros magnéticos.



Clasificaciones

Según si conservan la información o no cuando se quita la energía eléctrica:

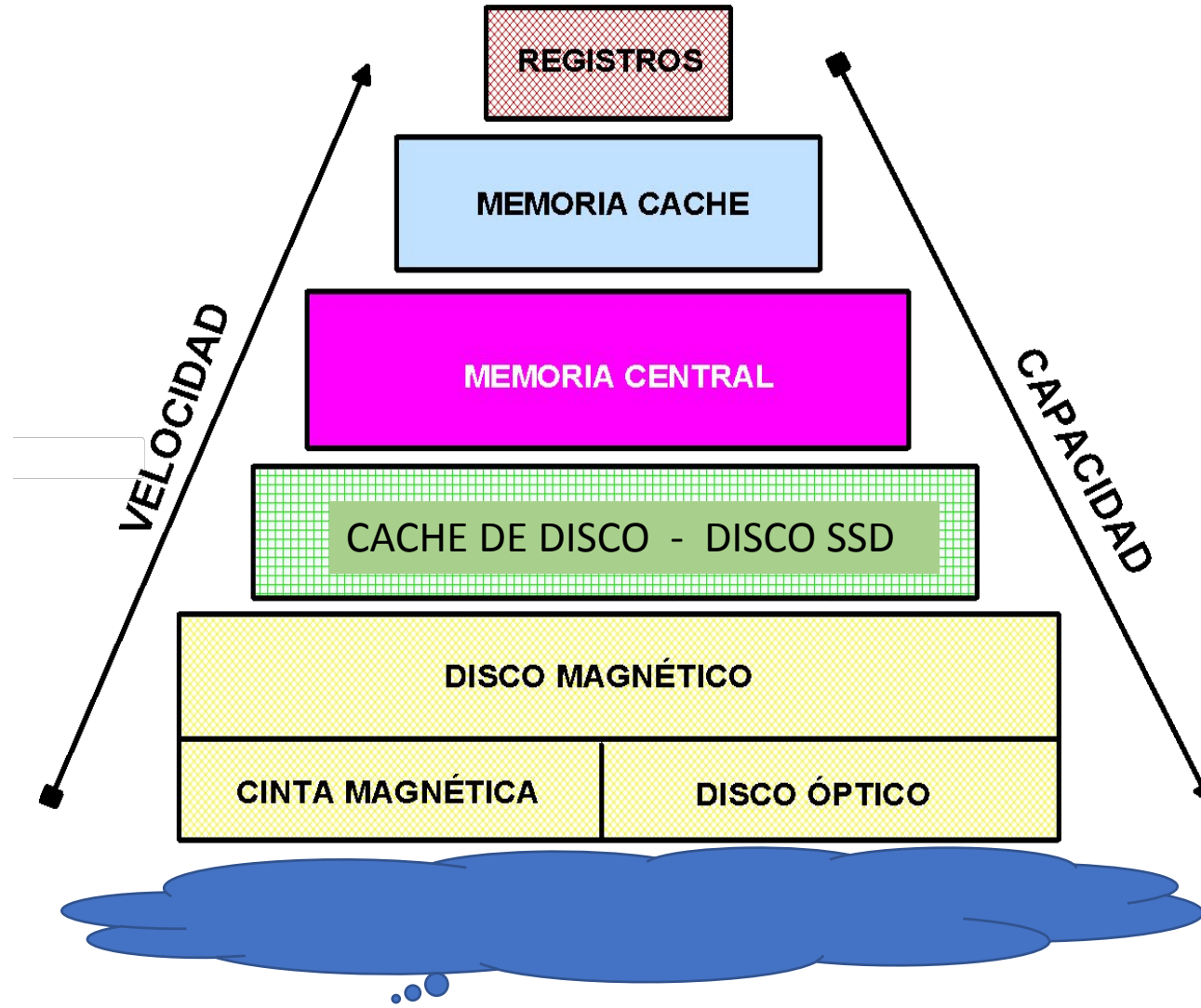
- **Volátil**: **No conservan** la información si se quita la energía eléctrica.
 - Ejemplo: memoria principal.
- **No volátil**: **Conservan** la información aún sin energía eléctrica.
 - Discos duros.

Memorias **no volátiles** eléctricas:

- **ROM (Read-Only Memory)**: Solo puede leerse (viene escrita de fábrica).
- **PROM (Programmable Read-Only Memory)**: Puede escribirse una sola vez.
 - Aplicaciones: Dispositivos de propósito específico.
 - También llamada **OTP EPROM: One-Time Programmable EPROM**.
- **EPROM (Erasable Programmable Read-Only Memory)**: Puede borrarse con luz ultravioleta. (También denominada **UV EPROM**)
- **EEPROM (Electrically Erasable Programmable Read-Only Memory)**: Borrable eléctricamente posición por posición.
 - Aplicaciones: Sistemas embebidos.
- **FLASH**: Borrable eléctricamente. Se escribe y borra por bloques.
 - Aplicaciones: SSD (discos de estado sólido), memorias USB, SD, etc.



CLASIFICACION POR JERARQUIA



CLASIFICACION

Tipo de Memoria	Tecnología	Tamaño	Tipo de acceso	Tiempo de acceso	Caudal
Caché	RAM de semiconductor	128 - 512 KB 8 Mb	Aleatorio	<10 ns	Depende del reloj central
Memoria Principal	RAM de semiconductor	4 – 32 GB	Aleatorio	<50 ns	Depende del reloj central
Disco Magnético	Disco Rígido	> 100 GB TB	Cíclico	<10 ms	10 MB/seg
Disco Óptico	CD-ROM	5 GB	Secuencial	300 ms	600 KB/seg
Cinta Magnética	Cinta	Varios TB	Secuencial	Varios segundos	10 MB/min

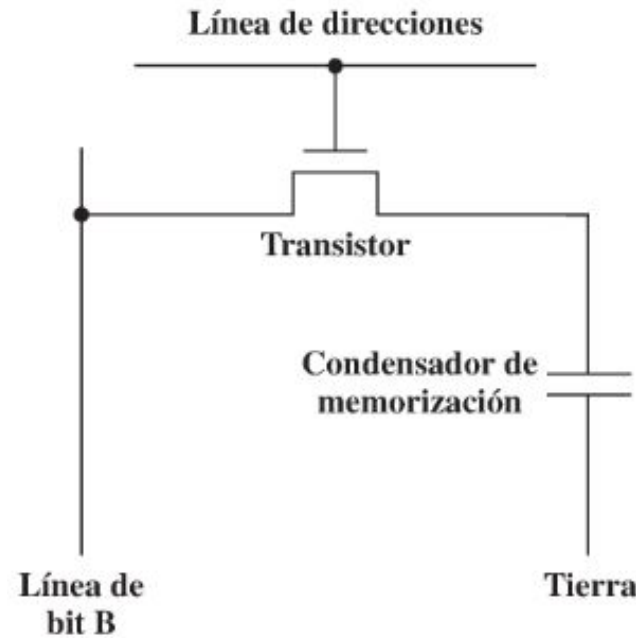
Memoria estática vs memoria dinámica

Memoria dinámica (DRAM):

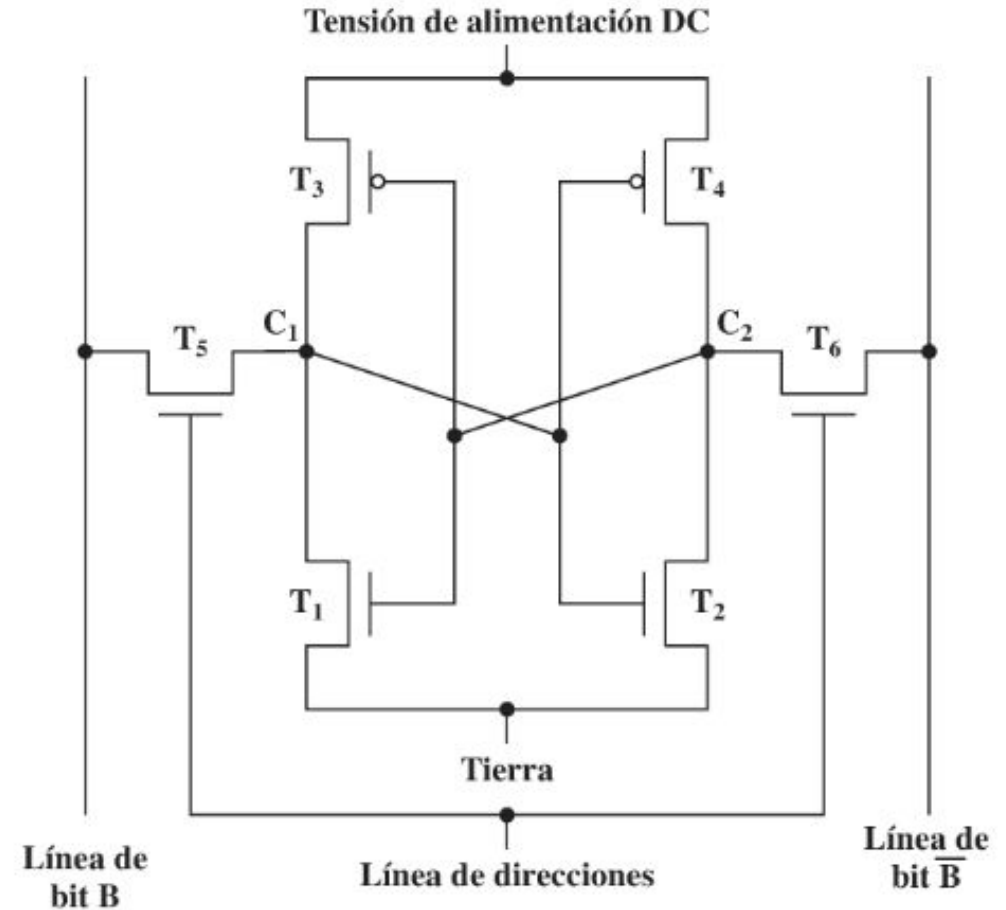
- Mayor densidad.
- Requiere refresco, **más lenta**.
- Direccionamiento por filas y columnas.
- Memoria principal.

Memoria estática (SRAM):

- Menor densidad.
- No requiere refresco, **más rápida**.
- Memoria caché.



Celda de bit memoria dinámica



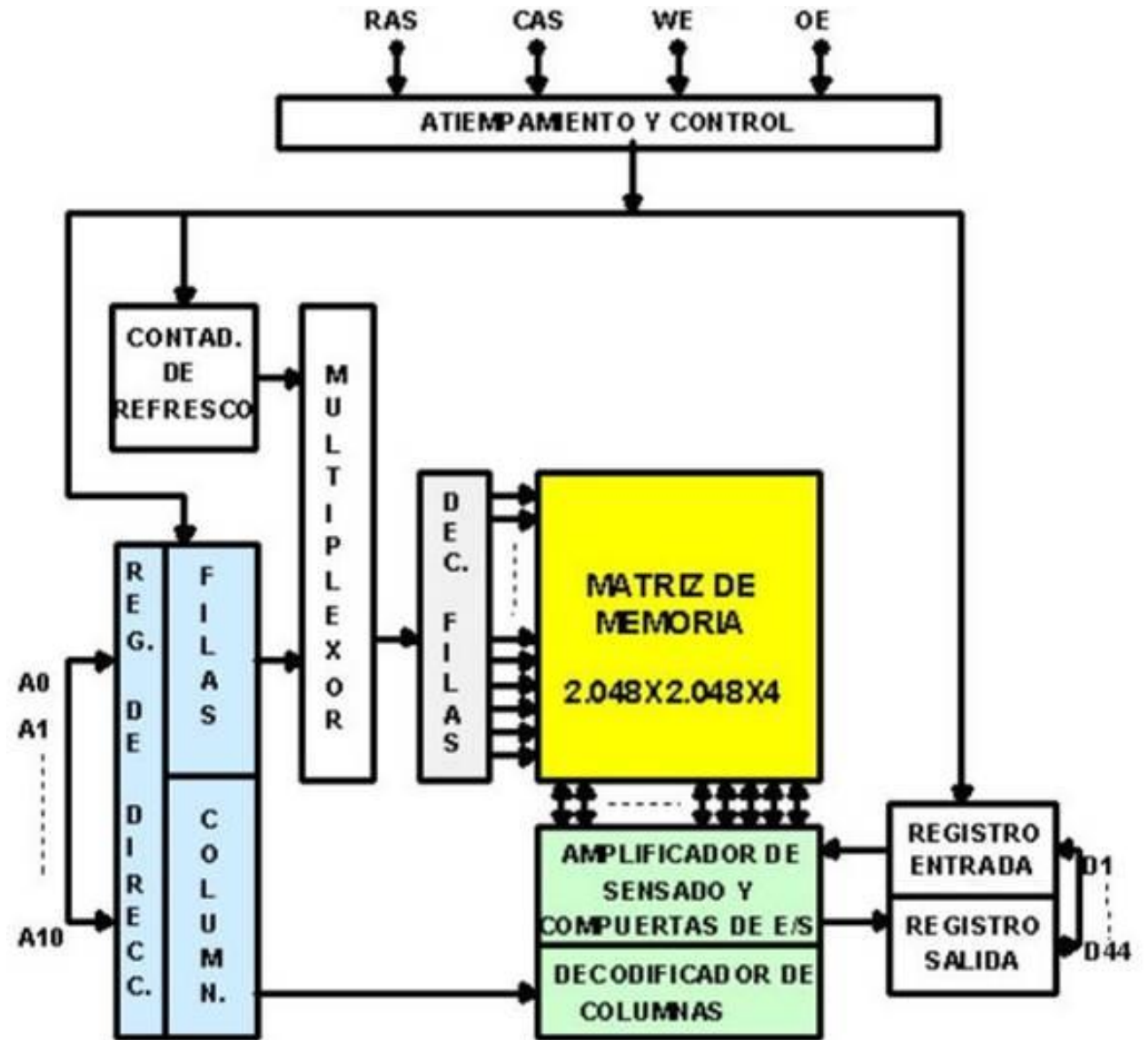
Celda de bit memoria Estática



Atención: ambas son **volátiles**

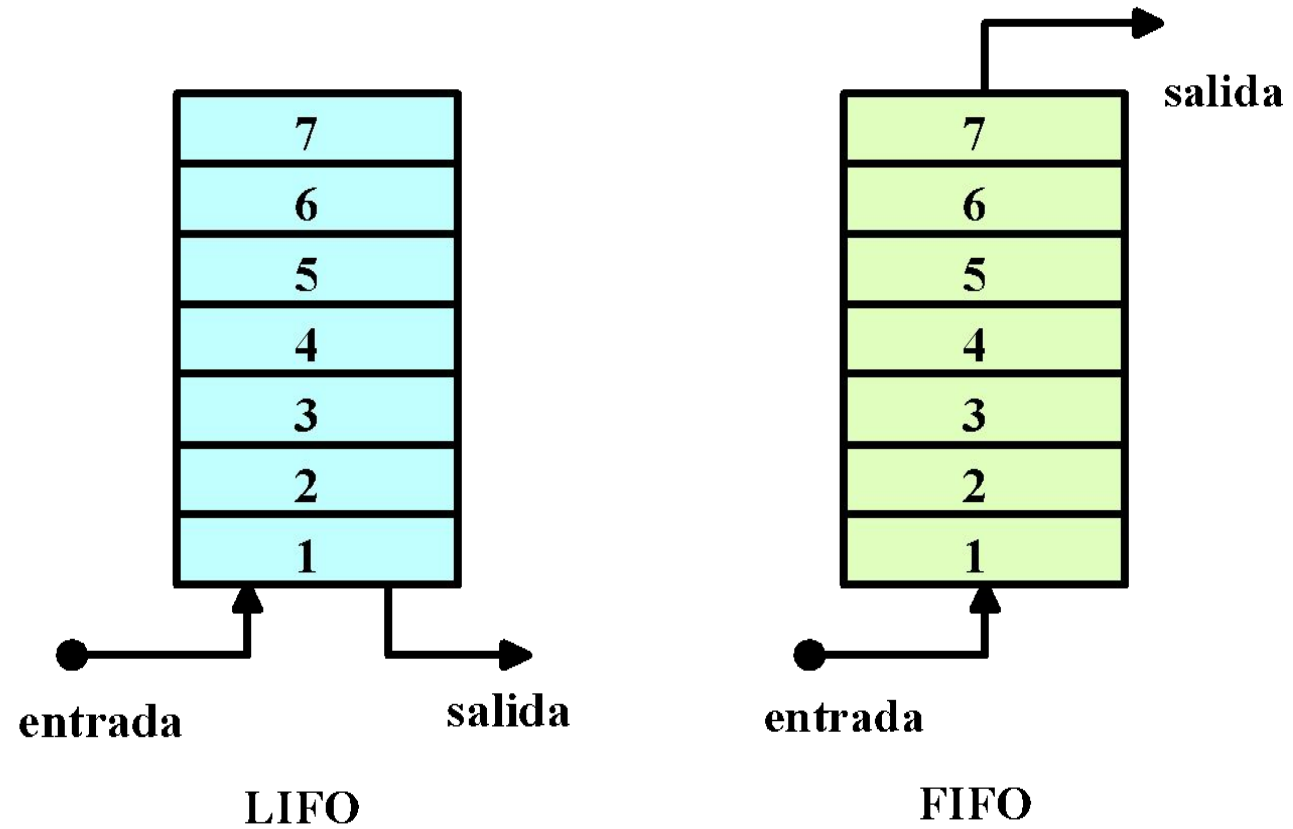
Mecanismo de refresco memoria dinámica.

- Acceso:
 - Primero se indica la fila, luego la columna.
 - con n bits, se direccionan 2^{2n} posiciones (ejemplo; con 10 bits se direccionan $2^{20} = 1\text{M}$ posiciones).
- **RAS** = Row Address Select (Elegir Dirección de Fila).
- **CAS** = Column Address Select (Elegir Dirección de Columna).
- Requiere refresco permanente.



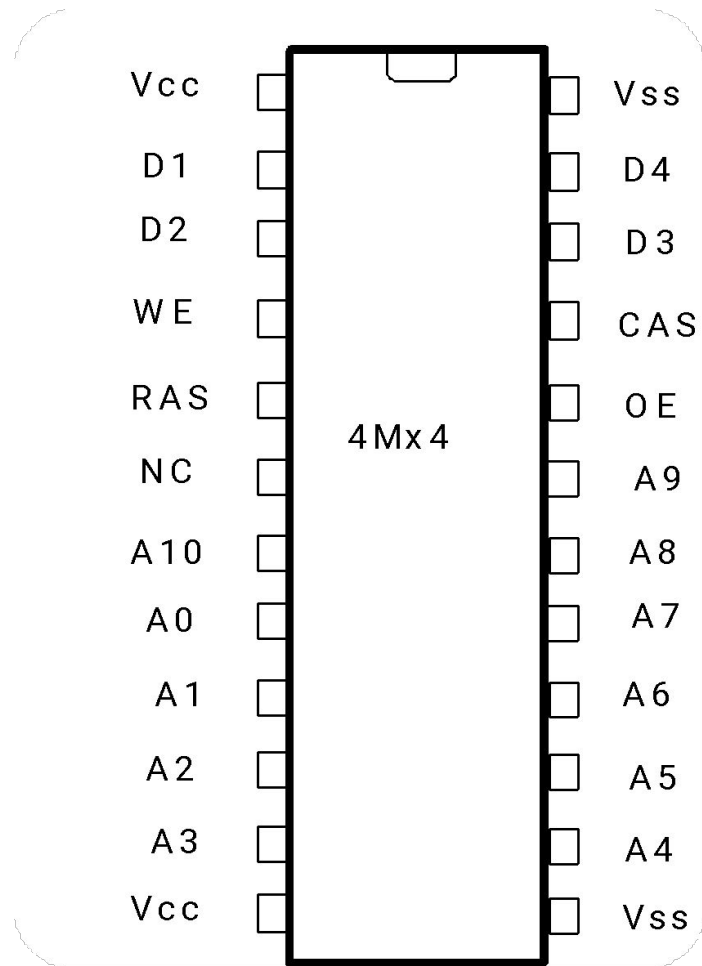
MEMORIAS ESPECIALIZADAS

MEMORIAS DE PILAS



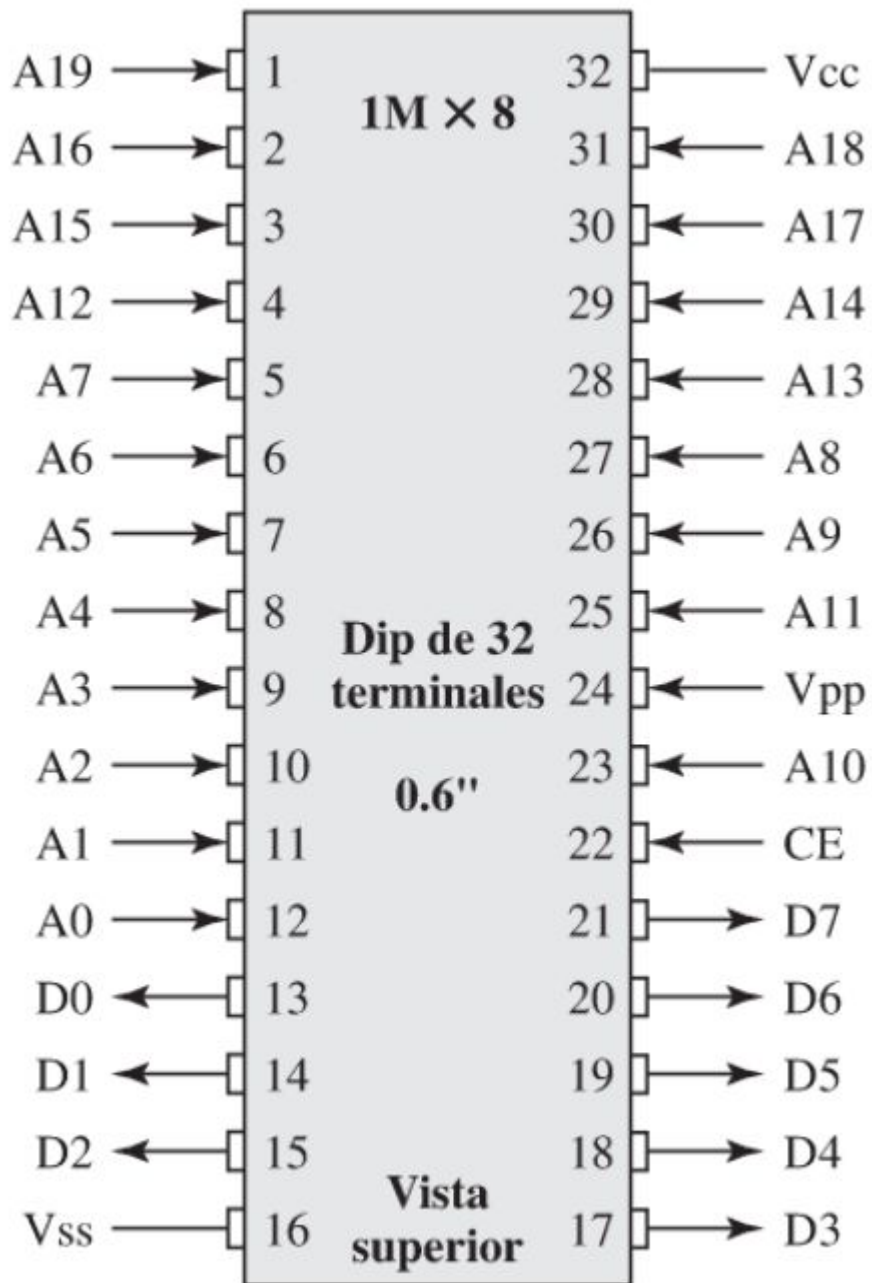
Tipo de memoria	Clase	Borrado	Mecanismos de escritura	Volatilidad
Memoria de acceso aleatorio (RAM)	Memoria de lectura/escritura	Eléctricamente por bytes	Eléctricamente	Volátil
Memoria de sólo lectura (ROM)	Memoria de sólo lectura	No posible	Mediante máscaras	No volátil
ROM programable (PROM)			Eléctricamente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

Figura obtenida de Organización y Arquitectura de Computadoras. William Stallings. 7ª ed. pag. 174.



- **RAS = Row Address Select (Elegir Dirección de Fila)**
- **CAS = Column Address Select (Elegir Dirección de Columna)**
- **WE = Write Enable (Habilitar Escritura)**
- **OE = Output Enable (Habilitar Salida)**

- Posee WE: lectura y escritura.
- No posee Vpp: volátil
- Posee RAS y CAS -> memoria dinámica.
- 11 líneas de direccionamiento (de A0 a A10) -> 2^{22} posiciones de memoria ($2^2 * 2^{20} = 4M$).
- 4 bits de datos (de D0 a D3)



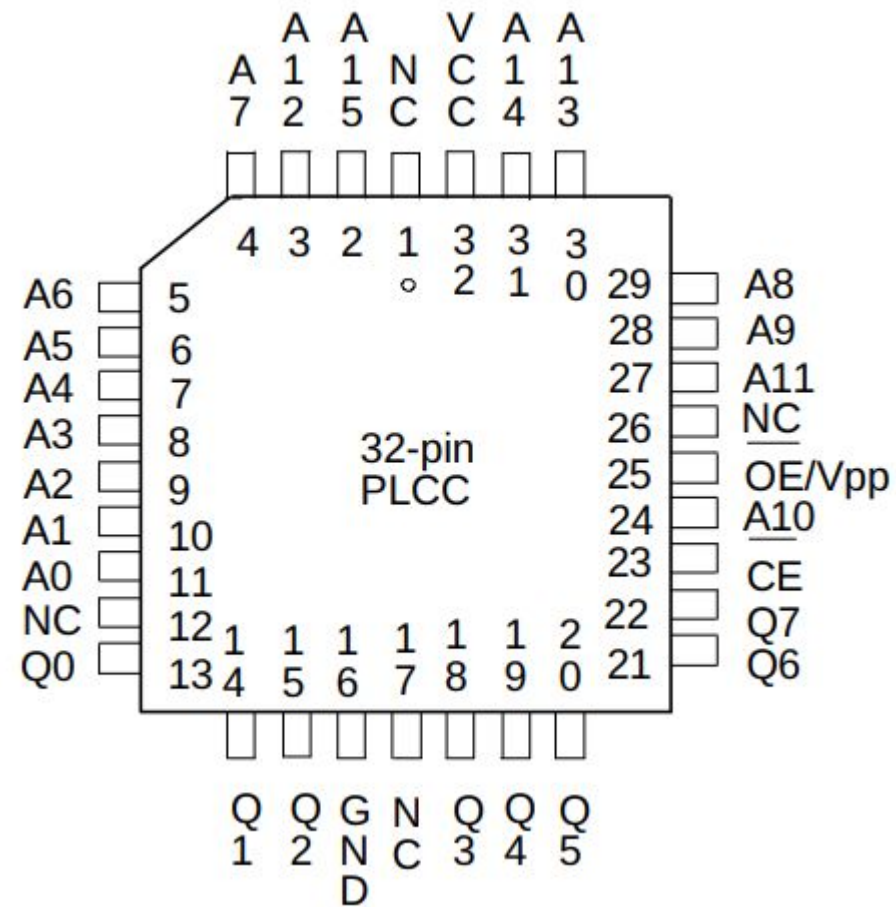
No posee WE: Solo lectura o sobre-todo lectura.

Bus direcciones:
 $A_0 - A_{19}: 2^{20}=1M$

Bus datos:
 $D_0 - D_7: 8$ bits.

Organización:
 1Mx8

Posee Vpp (tensión de programación/borrado no volátil): Se escribe y borra eléctricamente: EEPROM.



No posee WE: Solo lectura o sobre-todo lectura.

Bus direcciones: $A_0 - A_{15}: 2^{16}=64K$

Bus datos: $Q_0 - Q_7: 8$ bits.

Organización: 64Kx8

Posee Vpp (tensión de programación/borrado): Se escribe y borra eléctricamente: EEPROM.



Memoria Eprom M27c512
10f1

\$ 4.000

[Ver los medios de pago](#)

Llega entre el 3 y el 8/abr por \$ 8.025¹

[Más formas de entrega](#)

Stock disponible

Cantidad: 1 unidad (2 disponibles)

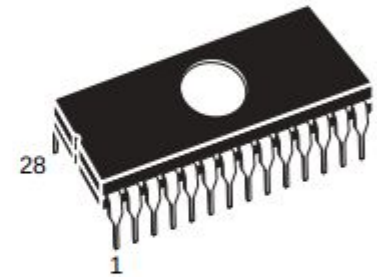
Comprar ahora

M27C512

512 Kbit (64K x8) UV EPROM and OTP EPROM

Features

- 5V \pm 10% supply voltage in read operation
- Access time: 45 ns
- Low power "CMOS" consumption:
 - Active current 30 mA
 - Standby current 100 μ A
- Programming voltage: 12.75 V \pm 0.25 V
- Programming time around 6 s.



FDIP28W (F)



UV EPROM: Borrable
por ultravioleta

OTP EPROM: (One-Time Programmable)
EPROM. Solo se programa una sola vez. No se
puede borrar. (=PROM)

Organización de la memoria

cantidad de palabras x **bits por palabra**

Relacionado con el número de bits del bus de direcciones

Relacionado con el número de bits del bus de datos



M27C512

512 Kbit (64K x8) UV EPROM and OTP EPROM

Organización: 64 k palabras.
8 bits cada palabra.



Nuevo

4 Memorias 93lc46b
93lc46b-i-p Eeprom Cmos 1k
Serial 64x16

\$ 8.075²²

en 6 cuotas de \$ 1.977⁸⁹
El precio por unidad es \$ 2.018,81

[Ver los medios de pago](#)

Llega el miércoles por \$ 7.361⁹⁹ ~~\$ 7.741⁹⁹~~

Comprando dentro de la próxima 1 h 23 min

[Más formas de entrega](#)

¡Última disponible!

Organización: 64 palabras x 16 bits por palabra.



16GB

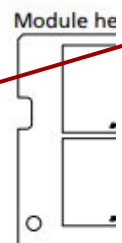
DDR4 SDRAM SODIMM

MTA16ATF2G64HZ - 16GB

Features

- DDR4 functionality and operations supported as defined in the component data sheet
- 260-pin, small-outline dual in-line memory module (SODIMM)
- Fast data transfer rates: PC4-3200, PC4-2666, or PC4-2400
- 16GB (2 Gig x 64)
- $V_{DD} = 1.20V$ (NOM)

Figure 1:

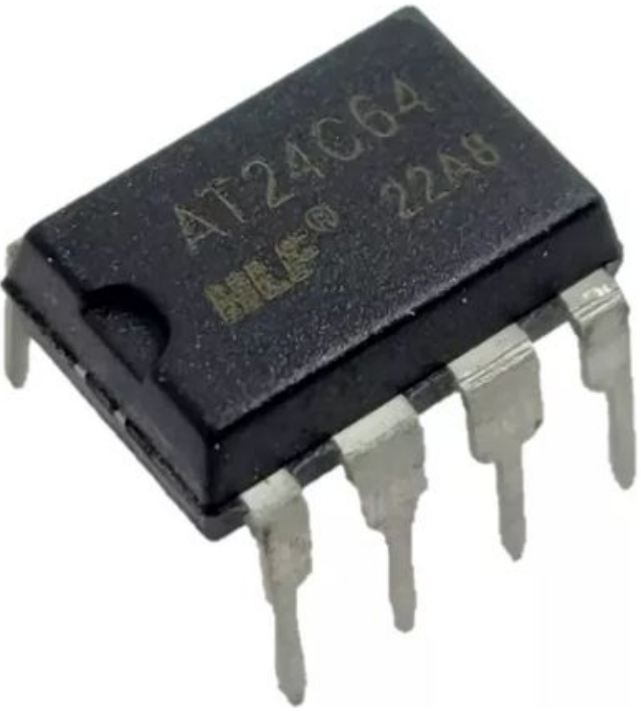
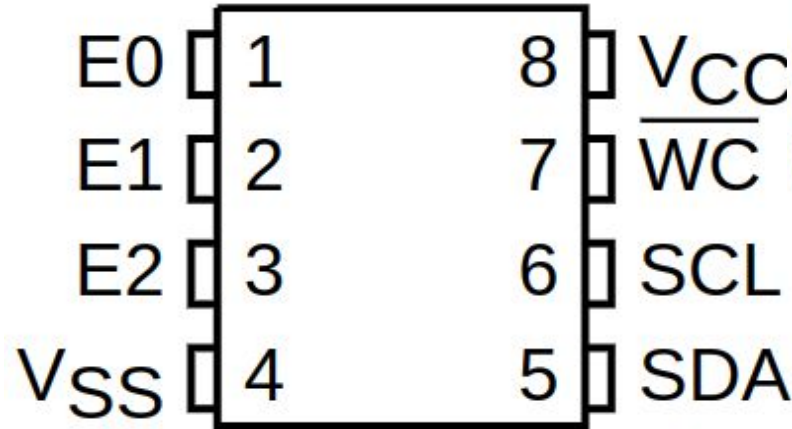


Organización: 2Gx64. 2G palabras por 64 bits por palabra
($2G * 64bits = 128Gb = 16GB$)

Memorias seriales

SUMMARY DESCRIPTION

These I²C-compatible electrically erasable programmable memory (EEPROM) devices are organized as 8192 x 8 bits (M24C64) and 4096 x 8 bits (M24C32).



E0, E1, E2	Chip Enable
SDA	Serial Data
SCL	Serial Clock
\overline{WC}	Write Control
V _{CC}	Supply Voltage
V _{SS}	Ground

8192 = 2^{13} palabras.

bus de dirección : 13 bits.

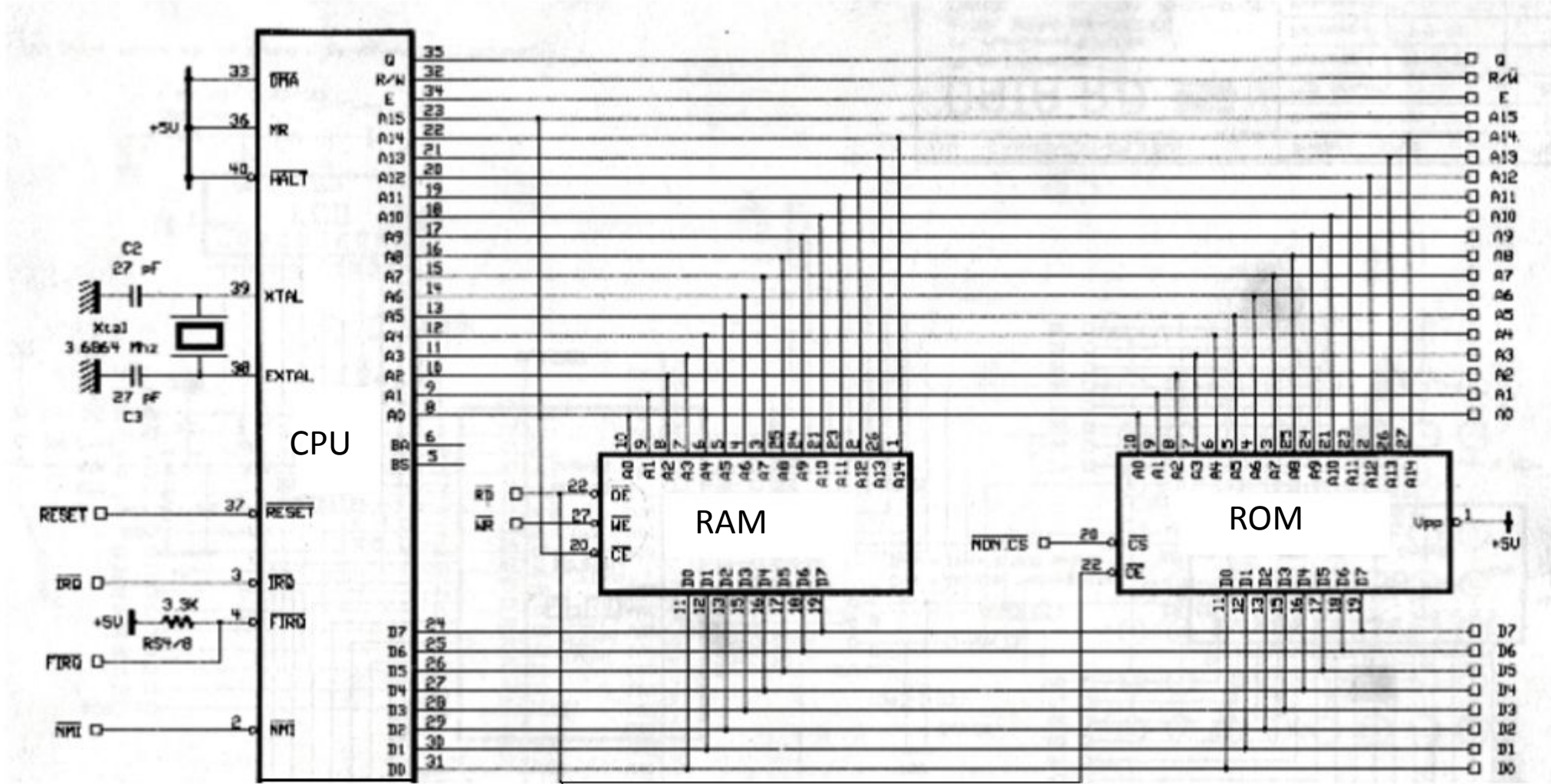
bus de datos: 8 bits.

Todos los bits van en serie (uno tras otro) por el mismo pin SDA.

Ventaja: Menos pines. Más pequeño.

Desventaja: Más lento.

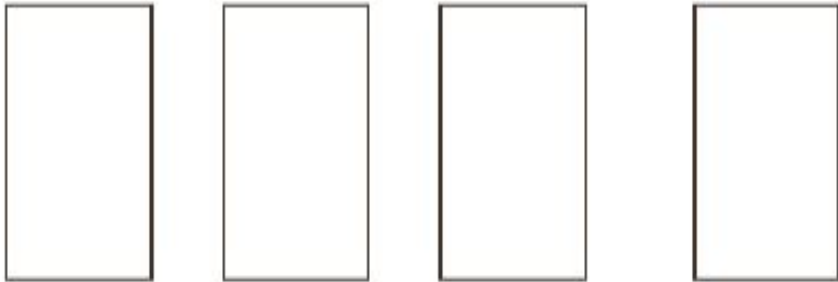
BANCOS DE MEMORIA



Banco de memoria

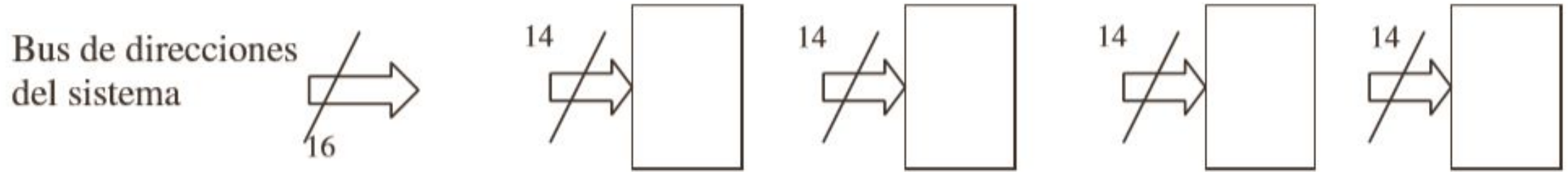
Supongamos un sistema que requiera una organización de memoria de 64 Kb x 8. Este sistema se puede implementar con un solo CI que tenga la organización requerida. En el caso de no disponer de estos circuitos integrados, se podría implementar la referida organización con otros Cis, como por ejemplo 4 CI de 16 Kb x 8.

$$16 \text{ Kb} + 16 \text{ Kb} + 16 \text{ Kb} + 16 \text{ Kb} = 64 \text{ Kb}$$



0000	
16383	
16384	
32767	
32768	
49151	
49152	
65535	

El bus de direcciones para direccionar 64 Kb tiene 16 líneas y el de cada circuito integrado tiene 14 líneas (16 Kb)

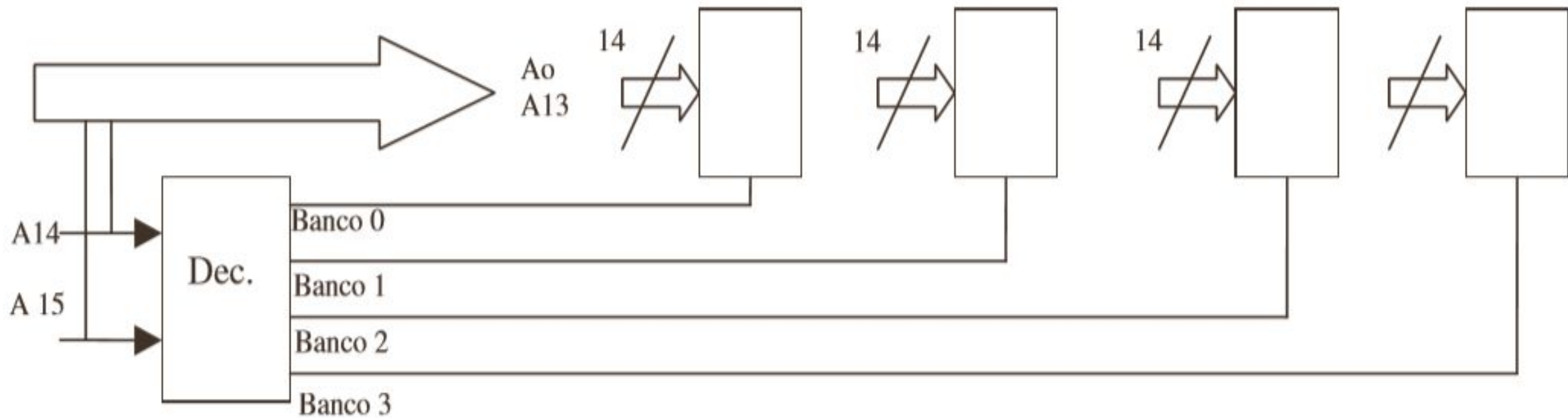


Si el contenido del bus de direcciones fuera XX00 0000 0000 1111, se está direccionando la dirección 15 de los cuatro Cis. Esto significa que se accede a la misma dirección de los 4 Cis. Para evitar esto se puede realizar lo siguiente:

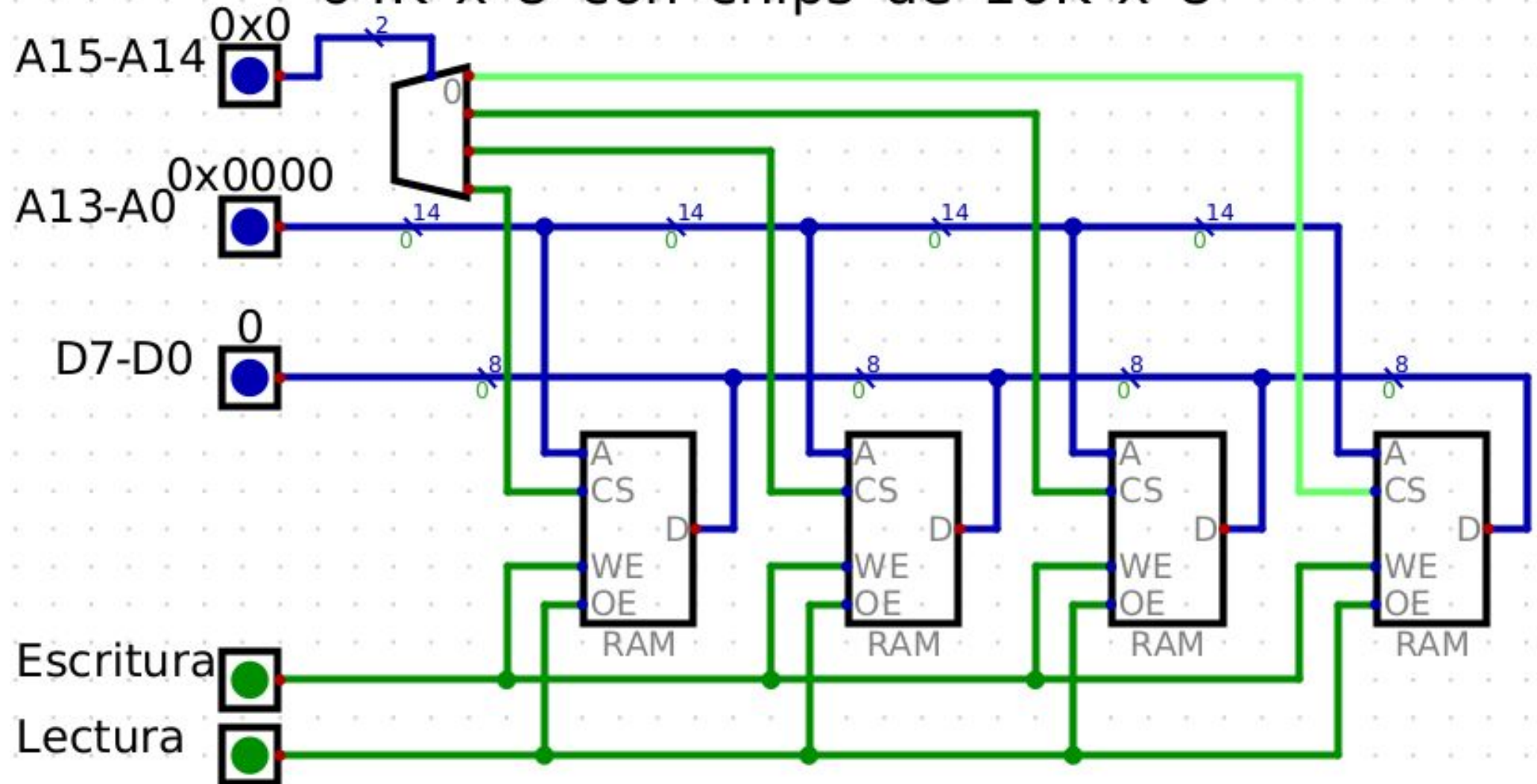
Utilizar los bits más significativos del bus de direcciones (msb) del sistema para seleccionar cual es el CI correspondiente a la dirección que el sistema quiere acceder. Por ejemplo:

	Contenido del bus de direcciones				Dirección
Primer CI	00 00	0000	0000	1111	15
Seg. CI	01 00	0000	0000	1111	16399
Ter. CI	10 00	0000	0000	1111	32783
Cuarto CI	11 00	0000	0000	1111	49167

	Contenido del bus de direcciones				Dirección
Primer CI	00 00	0000	0000	1111	15
Seg. CI	01 00	0000	0000	1111	16399
Ter. CI	10 00	0000	0000	1111	32783
Cuarto CI	11 00	0000	0000	1111	49167



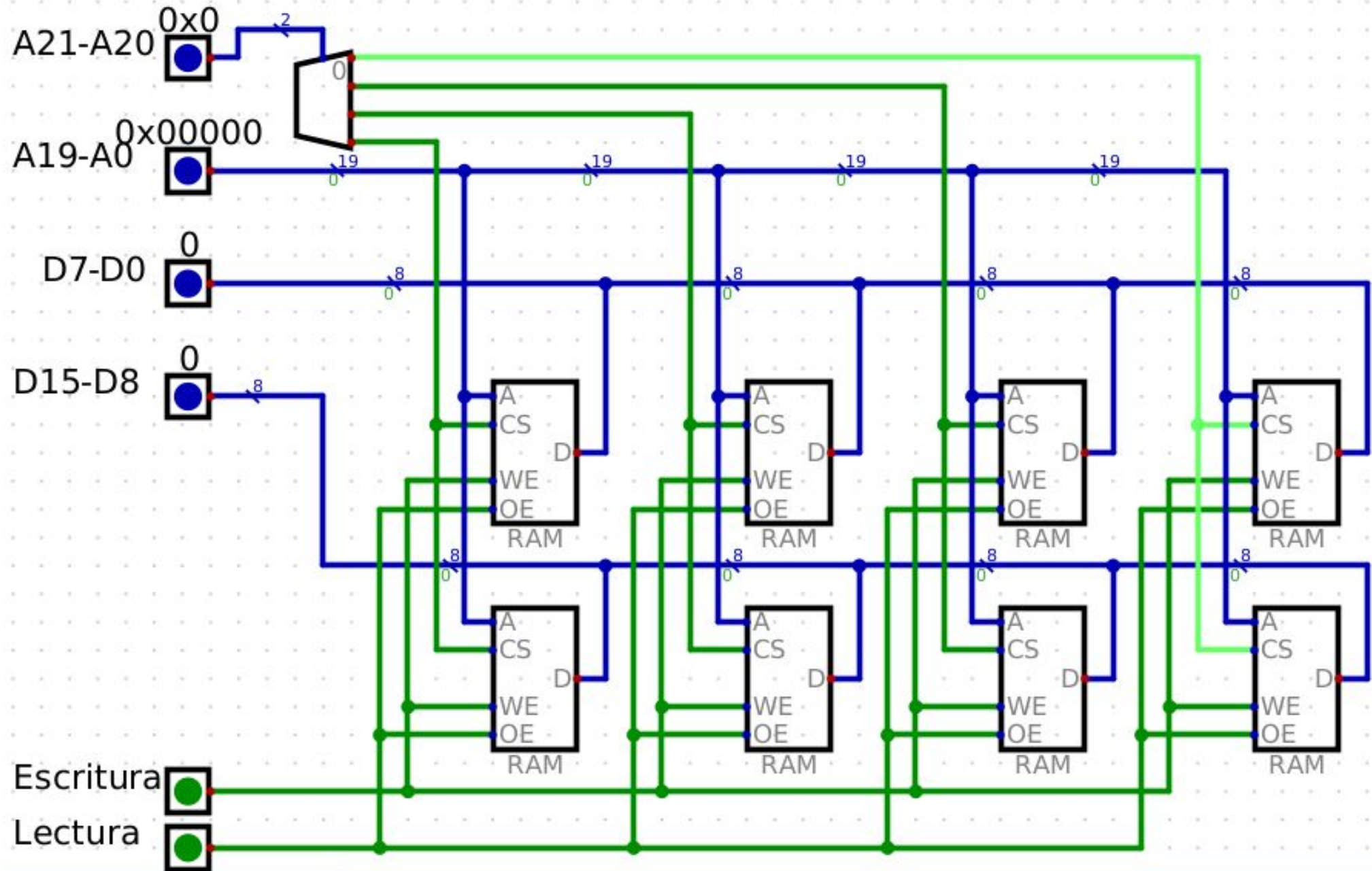
64K x 8 con chips de 16K x 8



Direcciones

49152	32768	16384	0
65535	49151	32767	16383

4M x 16 con chips de 1M x 8



Ejemplo:

16 GB

16 chips de 1 G x 8.



Memoria RAM DDR4 SODIMM 2666MHZ color verde 16GB 1 Crucial CB16GS2666

4.8 ★★★★★ (419)

MÁS VENDIDO 1° en Memorias RAM para Laptops

\$ 63.119

\$ 48.934 22% OFF

en 6 cuotas de \$ 11.985⁵⁷

[Ver los medios de pago](#)

OFERTA DEL DÍA

Lo que tenés que saber de este producto

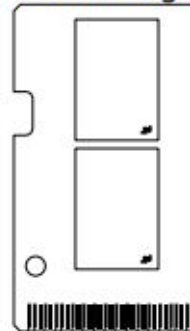
MTA16ATF2G64HZ – 16GB

Features

- DDR4 functionality and operations supported as defined in the component data sheet
- 260-pin, small-outline dual in-line memory module (SODIMM)
- Fast data transfer rates: PC4-3200, PC4-2666, or PC4-2400
- • 16GB (2 Gig x 64)
- $V_{DD} = 1.20V$ (NOM)
- $V_{PP} = 2.5V$ (NOM)
- $V_{DDSPD} = 2.5V$ (NOM)
- Nominal and dynamic on-die termination (ODT) for data, strobe, and mask signals
- Low-power auto self refresh (LPASR)
- Data bus inversion (DBI) for data bus
- On-die V_{REFDQ} generation and calibration
- Dual-rank
- On-board I²C serial presence-detect (SPD) EEPROM
- • 16 internal banks; 4 groups of 4 banks each
- Fixed burst chop (BC) of 4 and burst length (BL) of 8 via the mode register set (MRS)
- Selectable BC4 or BL8 on-the-fly (OTF)

Figure 1: 26

Module height



Options

- Operating – Comme
- Package – 260-pin
- Frequency – 0.62ns @
– 0.75ns @
– 0.83ns @

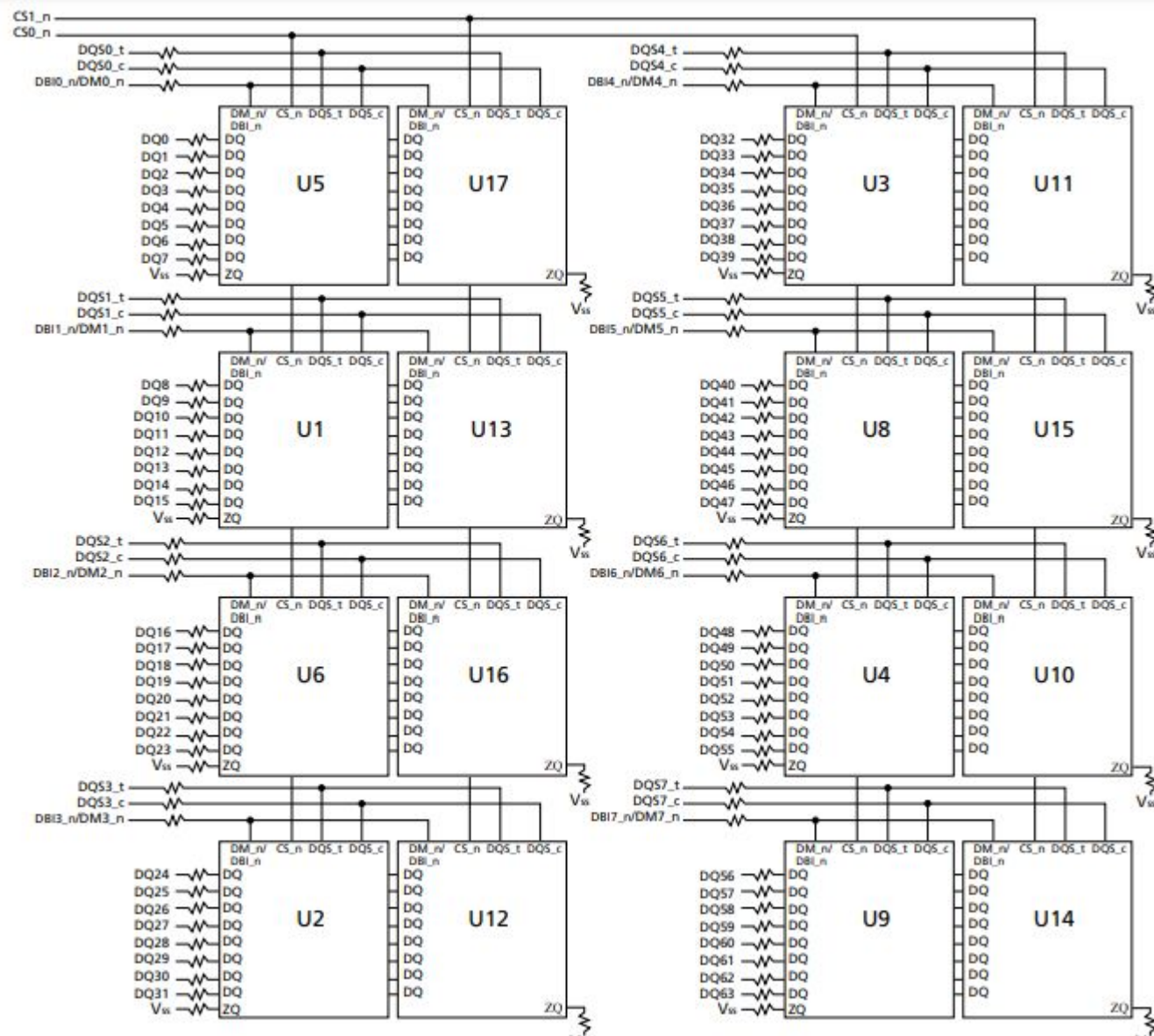
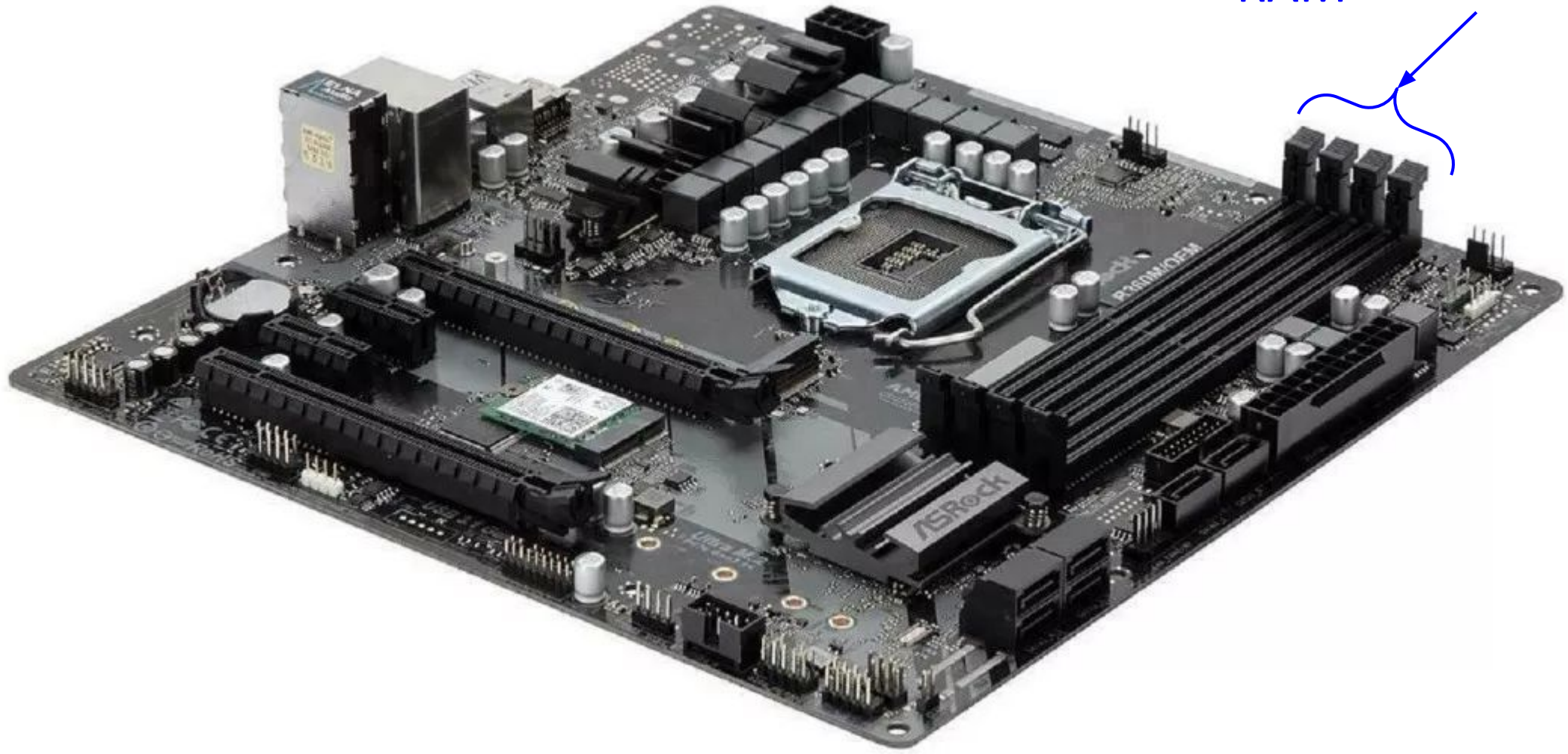


Figura obtenida de la hoja de datos de la memoria DDR4 SDRAM SODIMM, MTA16ATF2G64HZ – 16GB

<https://media-www.micron.com/-/media/client/global/documents/products/data-sheet/modules/sodimm/ddr4/atf16c2gx64hz.pdf?rev=f7a77116c7fe4fefbd87647143873c27>

Ejemplo de placa madre:

Slots para memoria
RAM





Direccionamiento físico y direccionamiento lógico

- Dirección **lógica**: Dirección presente en la instrucción. Indica una posición **relativa al comienzo del programa**.
- Dirección **física**: Dirección real en la memoria principal.

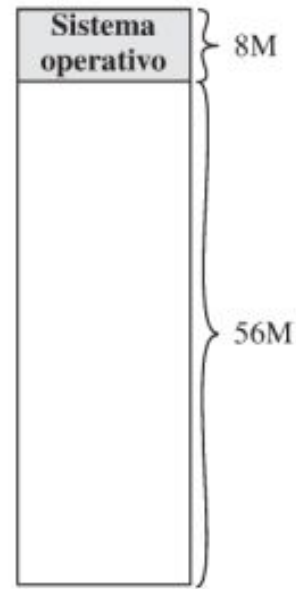
Mecanismos para convertir direcciones lógicas en direcciones físicas

- Direccionamiento lineal: La dirección lógica es igual a la dirección física.
- Direccionamiento segmentado.
- Direccionamiento paginado.
- Direccionamiento segmentado-paginado.

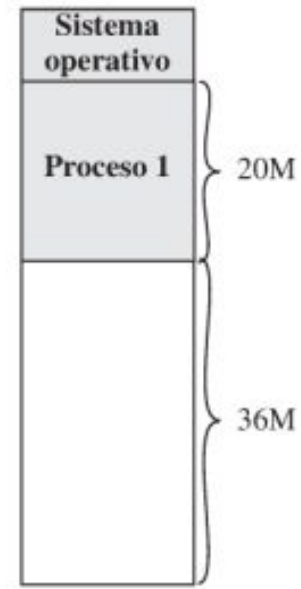
¿Por qué es necesario el **direccionamiento lógico**?

El direccionamiento lógico permite que la posición de los procesos (instrucciones y datos) en memoria en diferentes momentos de tiempo pueda variar.

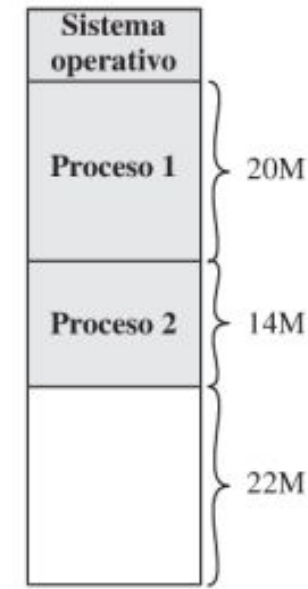
Nota:
Proceso: Instancias de un programa en ejecución.



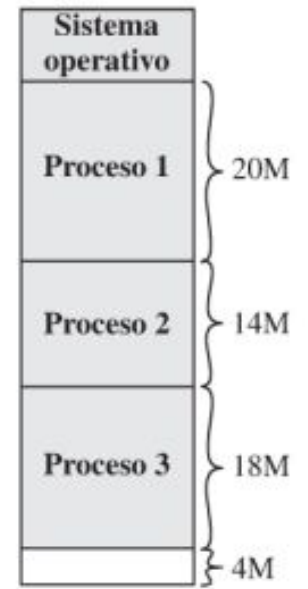
(a)



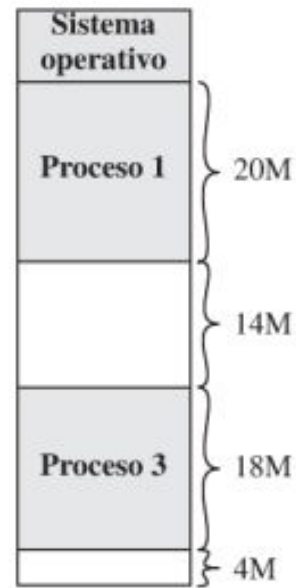
(b)



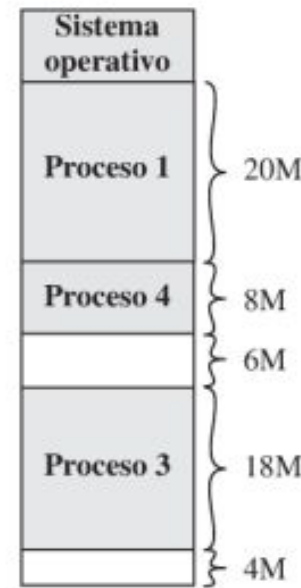
(c)



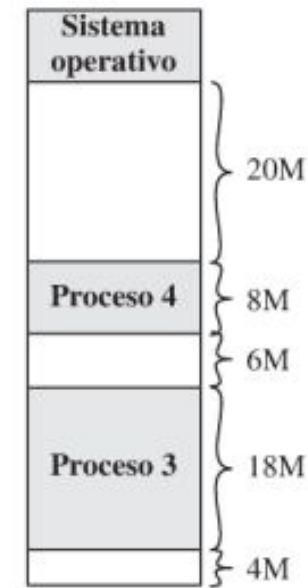
(d)



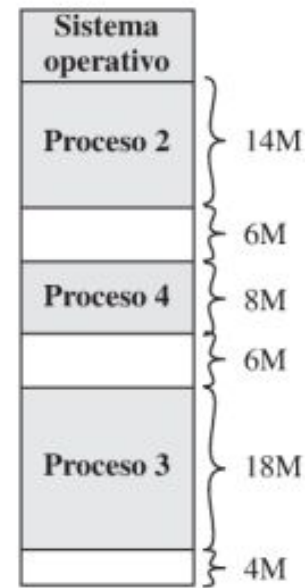
(e)



(f)



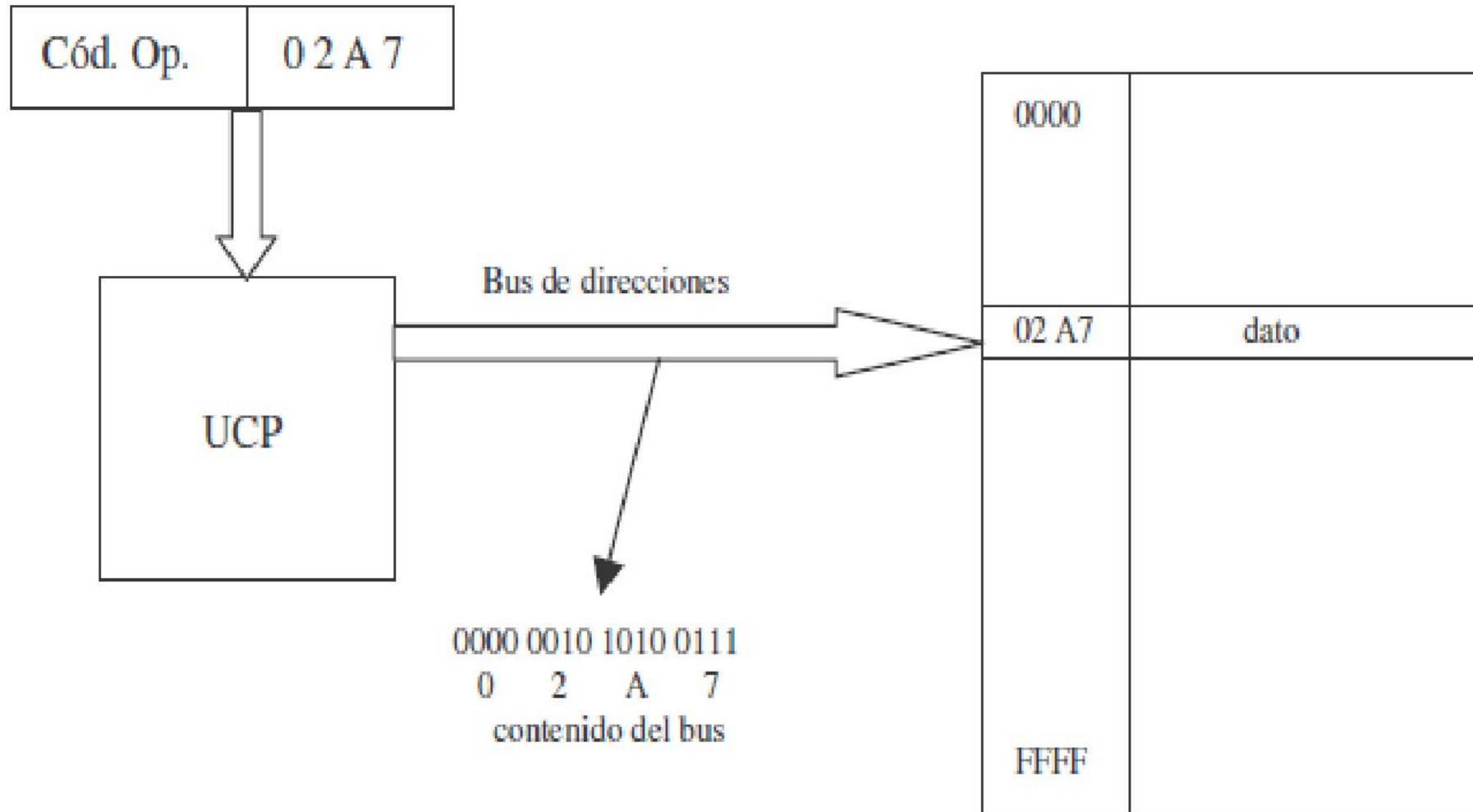
(g)



(h)

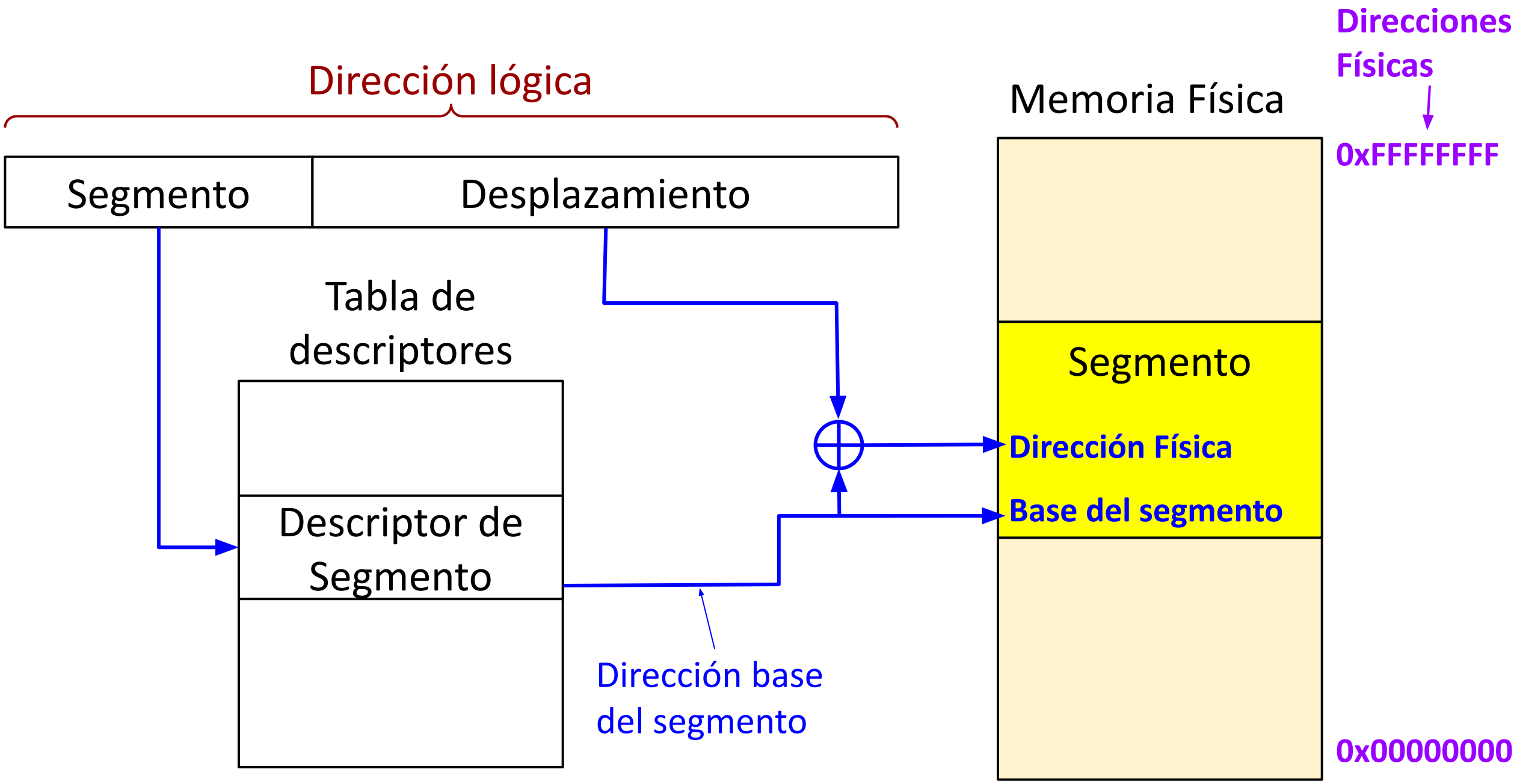
Direccionamiento Lineal

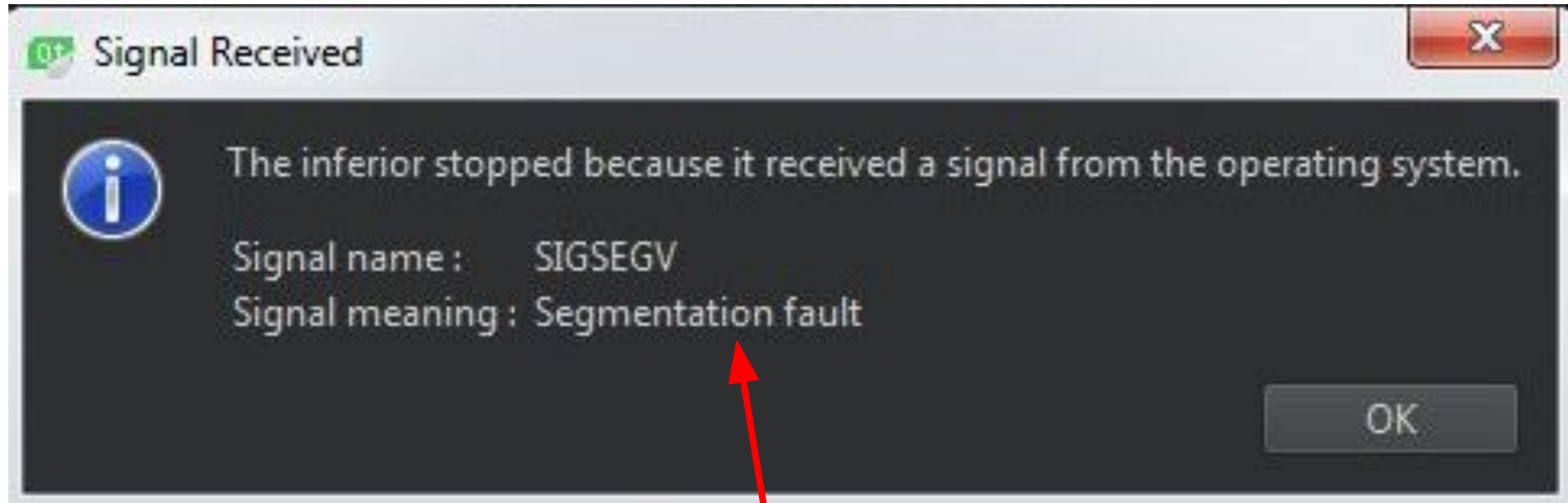
Dirección física = dirección lógica



Direccionamiento segmentado

- La memoria se divide en bloques llamados **segmentos**.
- Los segmentos son de **diferentes tamaños**. El tamaño es modificable.
- **Descriptor de segmento**: conjunto de datos que define las propiedades del segmento:
 - **Base del segmento**.
 - **Límite**.
 - **Atributos** (puede ser de solo lectura, de datos, de programa).
 - **Permisos** de uso y de acceso.
- Pueden haber segmentos compartidos (ejemplo, donde está el software de acceso a la red).
- **La dirección lógica puede tener más bits que la dirección física**.
 - Permite guardar segmentos en memoria principal y otros en el disco duro (memoria virtual).

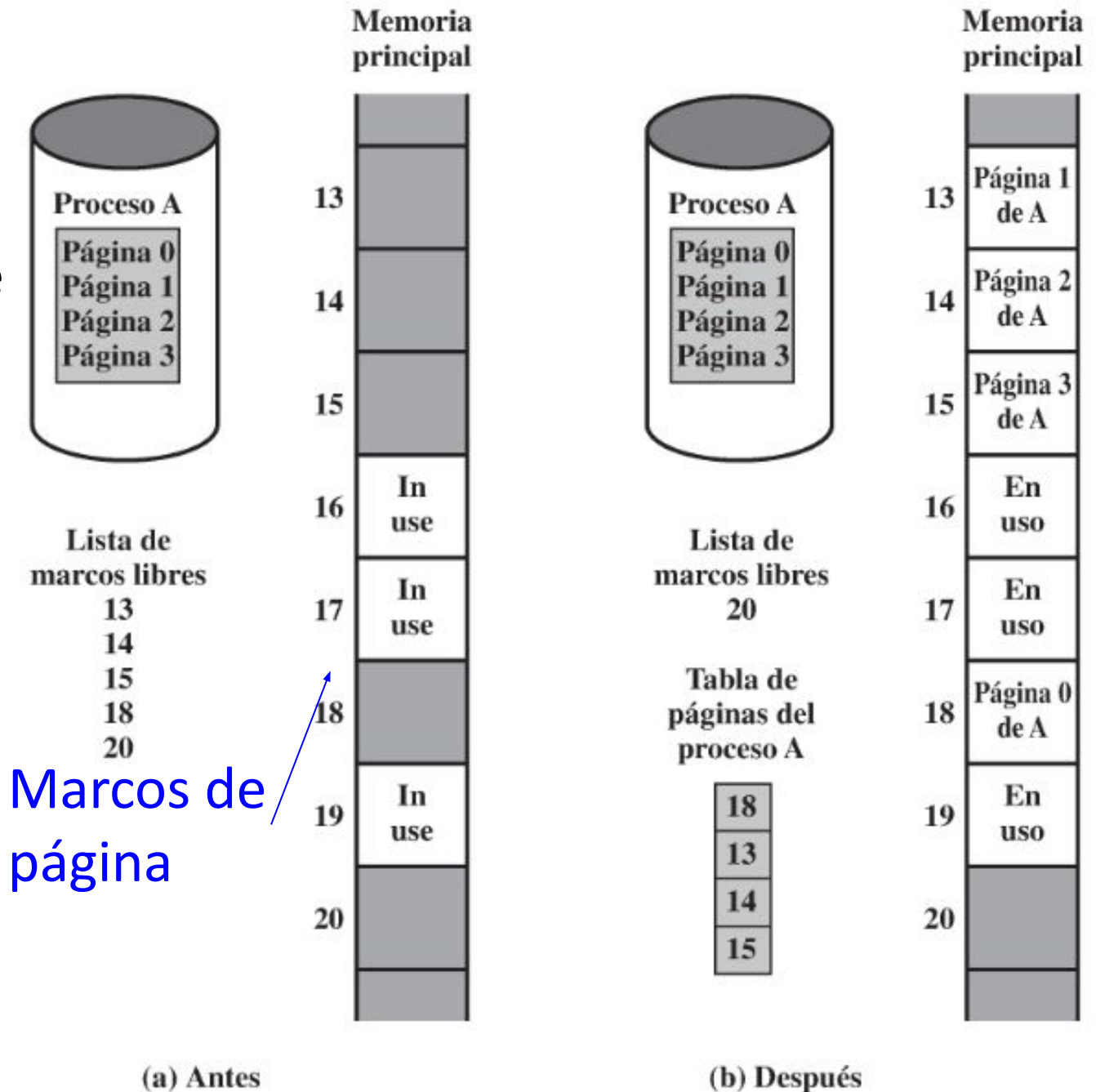




Error producido cuando un programa quiere acceder a direcciones de memoria fuera del segmento asignado, o cuando quiere escribir pero no tiene permisos de escritura

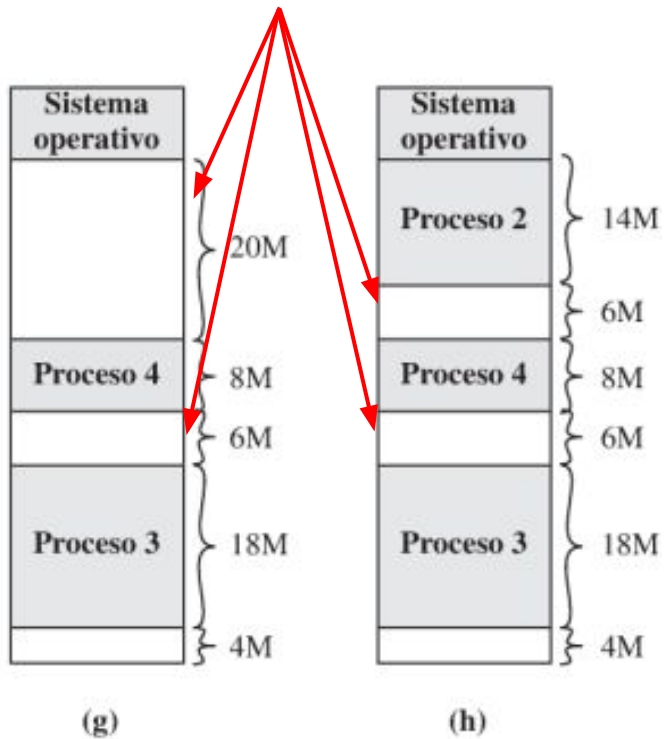
Memoria paginada

- La memoria se divide en **marcos de página**. Todos de **igual tamaño**.
- Los procesos se dividen en **páginas** de igual tamaño.
- La página es la unidad de transferencia mínima con discos duros.
- Objetivo: Hacer uso eficiente de memoria (Bloques grandes de memoria contigua hacen uso ineficiente de memoria).
- Al cargar un proceso, el sistema operativo le asigna marcos de página disponibles, adyacentes o no adyacentes.

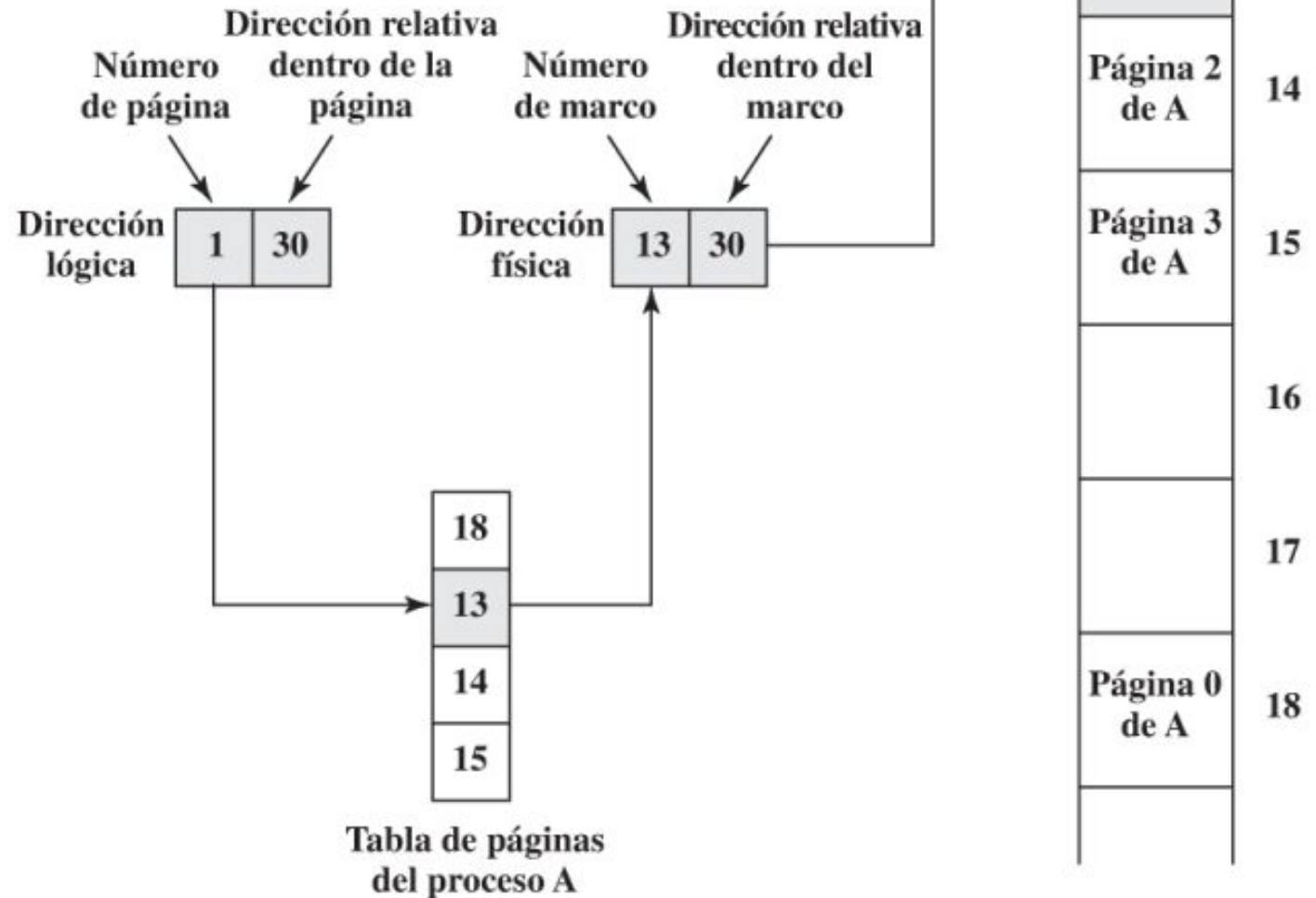


Ventaja de la paginación:

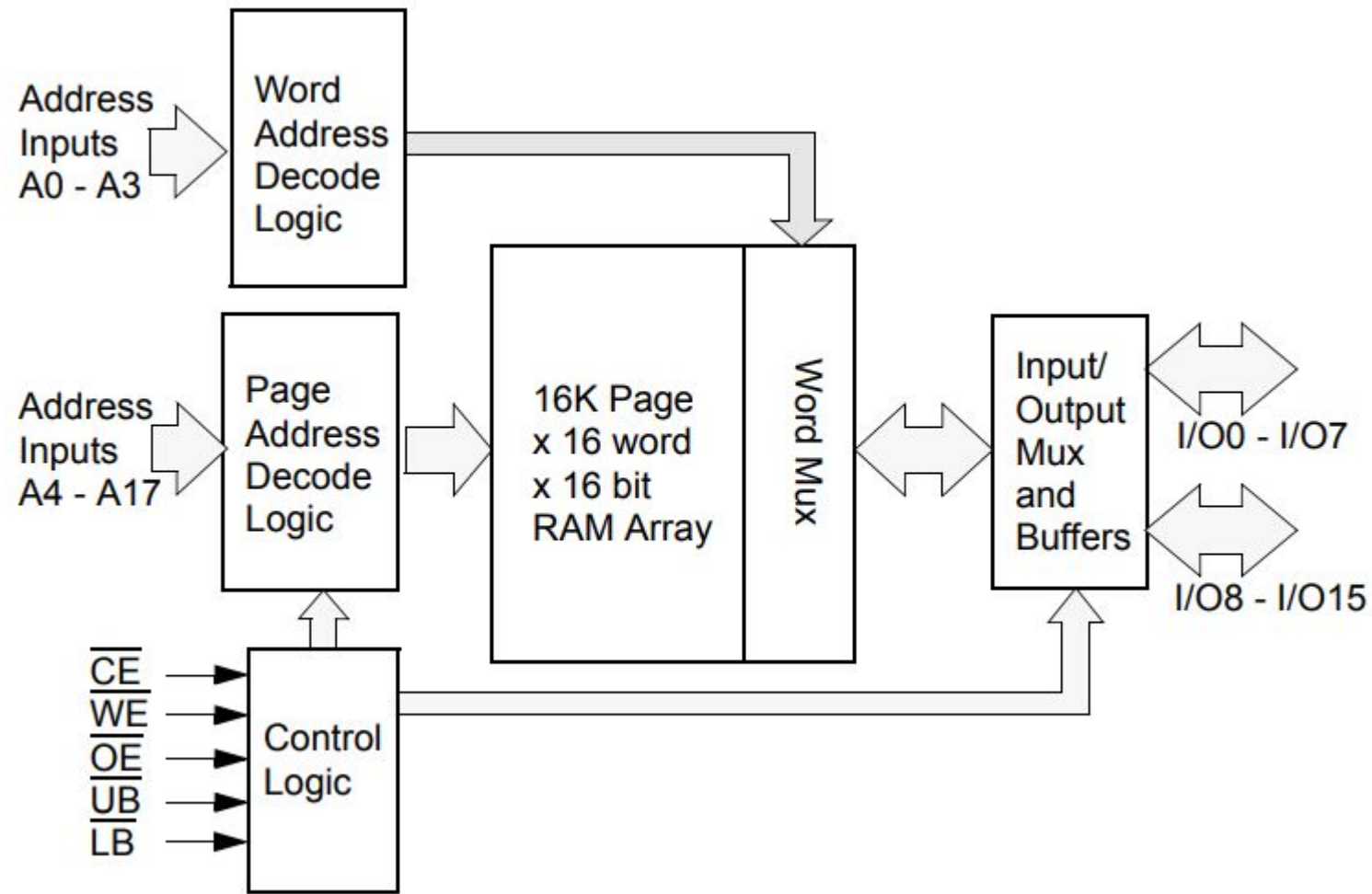
Si los procesos se almacenan en grandes bloques de memoria, al cerrar procesos y abrir otros, quedan “huecos” difíciles de llenar.

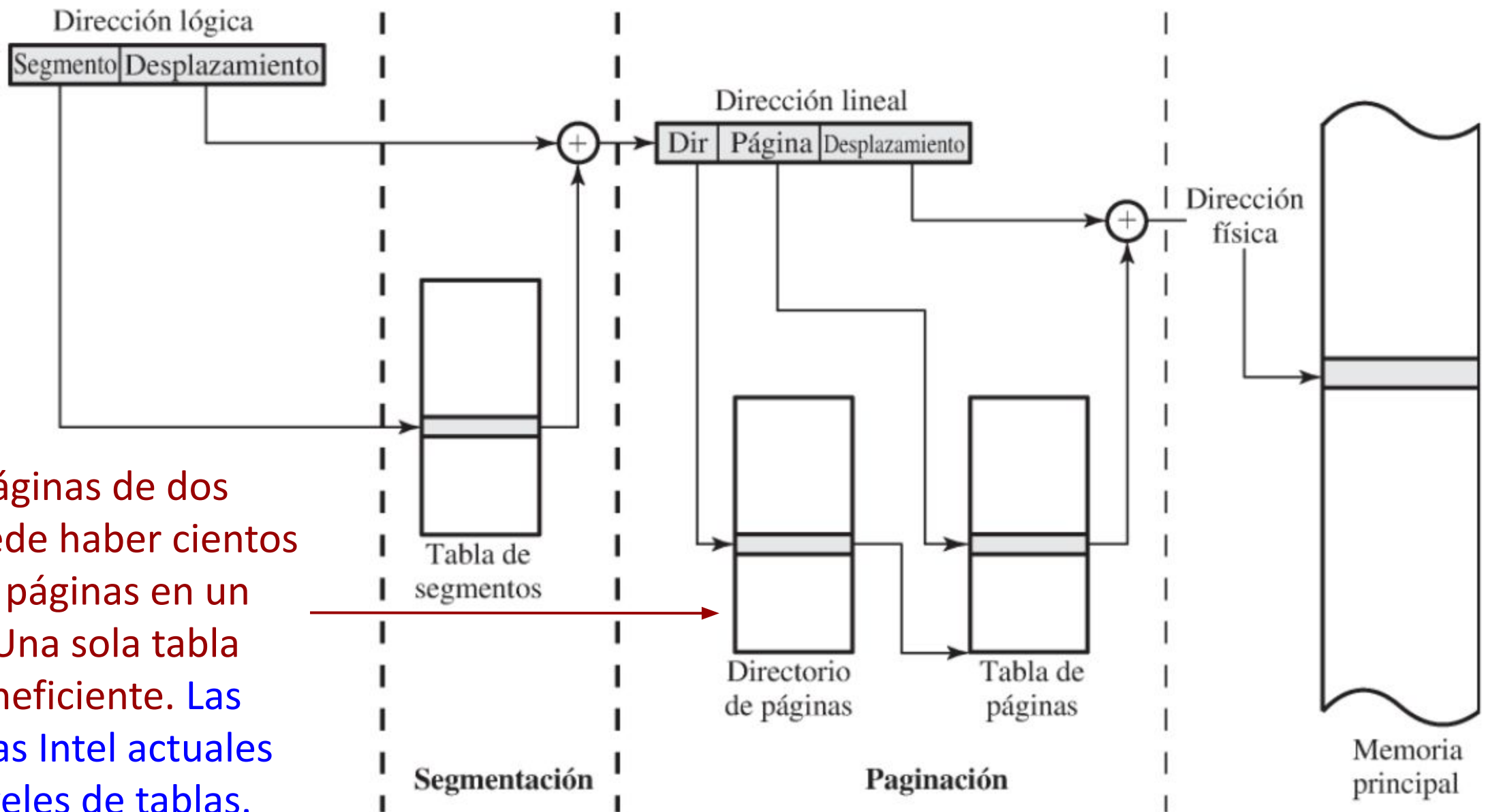


Al direccionar, el procesador debe buscar en qué marco de página está una página.



Ejemplo de memoria comercial paginada: Memoria N04L63W1A de Onsemi



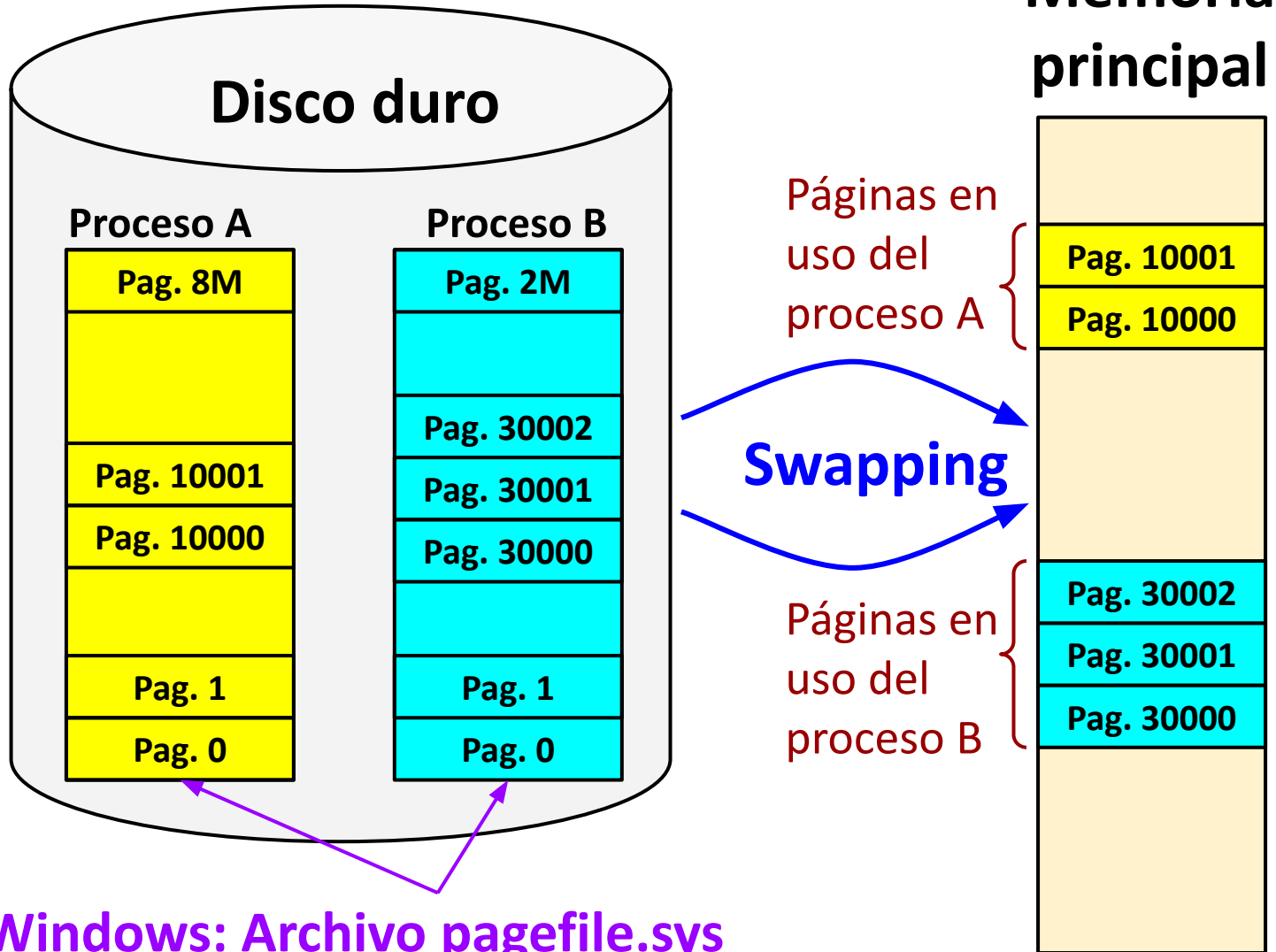


Tablas de páginas de dos niveles: Puede haber cientos de miles de páginas en un segmento. Una sola tabla puede ser ineficiente. Las arquitecturas Intel actuales tienen 5 niveles de tablas.

Figura obtenida de Organización y Arquitectura de Computadoras. William Stallings. 7ª ed. pag. 288.

Memoria virtual

swapping (intercambio)



Ejemplo:

- Páginas de 4KB
- Proceso A: $8 * 2^{20} * 4 * 2^{10} = 32\text{GB}$
- Proceso B: $2 * 2^{20} * 4 * 2^{10} = 8\text{GB}$
- Memoria principal de 4 GB.

¿Se pueden ejecutar los procesos?

Si, gracias a la **memoria virtual**:
Se copian en memoria principal sólo las páginas en uso, las otras se almacenan en el disco duro.

- La memoria virtual es más grande que la dirección física.

Windows: Archivo pagefile.sys

Linux: Partición Swap o archivo swap

El procesador utiliza varios bits de bandera para brindar o recuperar información sobre los segmentos o las páginas. Cada procesador tiene sus banderas. Algunos de uso común son:

Banderas (Bits) de información de segmentos

- Bit de segmento presente: Indica si el segmento está en memoria.

Banderas (Bits) de información de páginas

- **Bit de presencia:** Indica si la página está en memoria principal (también llamado bit de validez).
- **Bit de modificado** (también llamado bit de suciedad): Indica que la página ha sido escrita. Debe actualizarse al disco duro.
- **Bits de permisos:** Indica si las aplicaciones de usuario tienen acceso, solo el sistema operativo.

Ejemplo

Procesador i7-14650HX

Essentials		Download Specifications ↓	
Product Collection	Intel® Core™ i7 Processors (14th gen)	Marketing Status	Launched
Code Name	Products formerly Rap	Launch Date ⓘ	Q1'24
Vertical Segment	Mobile	Embedded Options Available ⓘ	No
Processor Number ⓘ	i7-14650HX	Memory Specifications	
Lithography ⓘ	Intel 7	Max Memory Size (dependent on memory type) ⓘ	192 GB
Recommended Customer Price ⓘ	\$495.00	Memory Types ⓘ	Up to DDR5 5600 MT/s Up to DDR4 3200 MT/s
		Max # of Memory Channels ⓘ	2
		Max Memory Bandwidth ⓘ	89.6 GB/s

Memoria física máxima: 192 GB
Memoria virtual: 48 bits (todos los x86_64 tienen 48 bits para direccionar memoria virtual).

<https://www.intel.com/content/www/us/en/products/sku/235996/intel-core-i7-processor-14650hx-30m-cache-up-to-5-20-ghz/specifications.html>

Ejemplo

Para el procesador anterior, indicar:

- 1) El formato de dirección virtual.
- 2) La cantidad máxima de marcos de página (suponiendo que instalamos la memoria máxima posible).
- 3) El número máximo de páginas que un proceso puede tener asignado y el tamaño máximo que puede ocupar en memoria.
- 4) La dirección física que corresponde a las direcciones virtuales a) 0x0000011A02 y b) 0x00000621B2

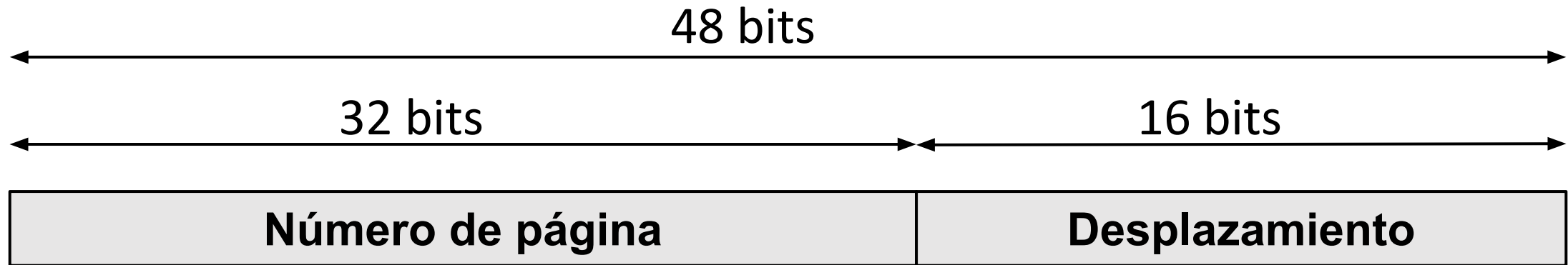
Página	Bit Presencia	Bit suciedad	Marco de página/disco
0	0	0	0xF423F
1	1	0	0xF4240
2	1	1	0xF4241
3	1	1	0xF4242
4	0	0	0xF423A
5	0	0	0xF423B
6	0	0	0xF423C
7	1	1	0xF423D
8	1	0	0xF423E

Memoria virtual (48 bits): $2^{48} = 2^8 * 2^{40} = 256$ TB

Memoria física máxima: 192 GB = $192 * 2^{30}$ B = $3 * 2^6 * 2^{30} = 3 * 2^{36}$

Tamaño página: 64KB (todos los x86_64) = $64 * 2^{10} = 2^6 * 2^{10} = 2^{16}$ (16 bits)

1) Formato dirección virtual



2) Cantidad de marcos de página =

tamaño de memoria física / tamaño página = $192 * 2^{30} / 64 * 2^{10} = 3 * 2^{20} =$
= 3M páginas.

3) Cantidad de páginas virtuales =

tamaño de memoria virtual/tamaño página = $2^{48}/2^{16} = 2^{32} = 2^2 * 2^{30} = 4G$ páginas.

4 a)

$$0000011A02_{16} = \underbrace{0000\ 0000\ 0000\ 0000\ 0000\ 0001}_{\text{página virtual}} \underbrace{0001\ 1010\ 0000\ 0010}_2 \text{ desplazamiento (16 bits)}$$

La página virtual 1 tiene bit de presencia 1, por lo tanto, está en la memoria física. Le corresponde el marco de página 0xF4240, por lo que la dirección física será: $F42401A02_{16}$

$$\underbrace{0000\ 1111\ 0100\ 0010\ 0100\ 0000}_{\text{marco de página (0xF4240)}} \underbrace{0001\ 1010\ 0000\ 0010}_{\text{desplazamiento}} = F42401A02_{16}$$

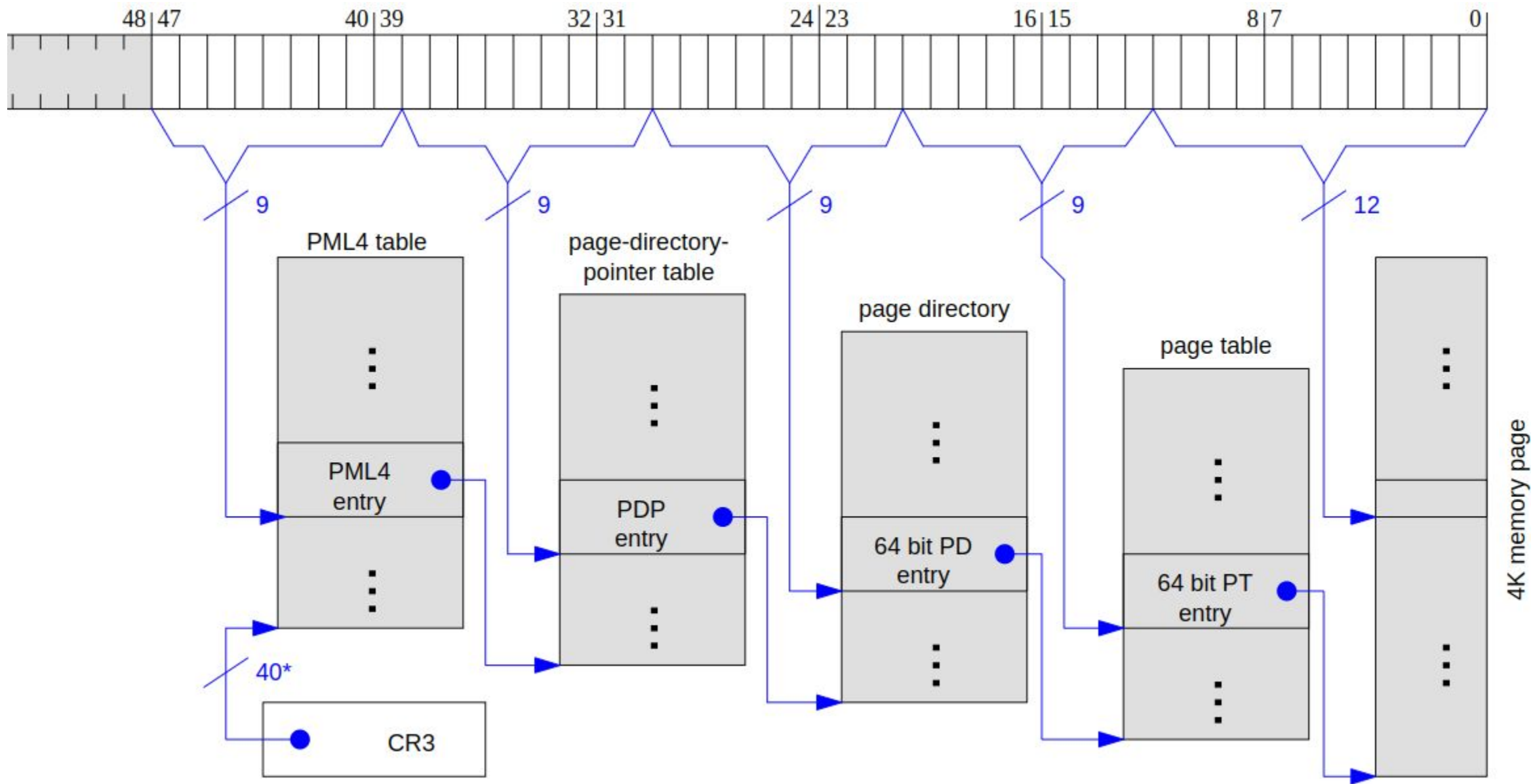
Se reemplaza la página virtual por el marco de página

4 b)

$$00000621B_{16} = \underbrace{0000\ 0000\ 0000\ 0000\ 0000\ 0110}_{\text{página virtual}} \underbrace{0010\ 0001\ 1011\ 0010}_2$$

desplazamiento

La página virtual $110_2 = 6$ tiene bit de presencia 0, por lo tanto, no está en la memoria física.



4K memory page

MMU (Memory Management Unit)

- **Traduce** direcciones lógicas a físicas (trabaja con segmentos y páginas).
- Gestiona memoria virtual.
- Gestiona los permisos (permisos de lectura, escritura y ejecución).

Cache de traducción de direcciones (TLB: Translation Lookaside Buffer):

- Almacena traducciones recientes de direcciones lógicas a físicas, para realizar traducciones futuras más rápido.

Algoritmo de reemplazo de páginas

- Elige qué página en memoria principal reemplazar cuando es necesario traer una nueva página desde el disco duro.

MEMORIA CACHÉ

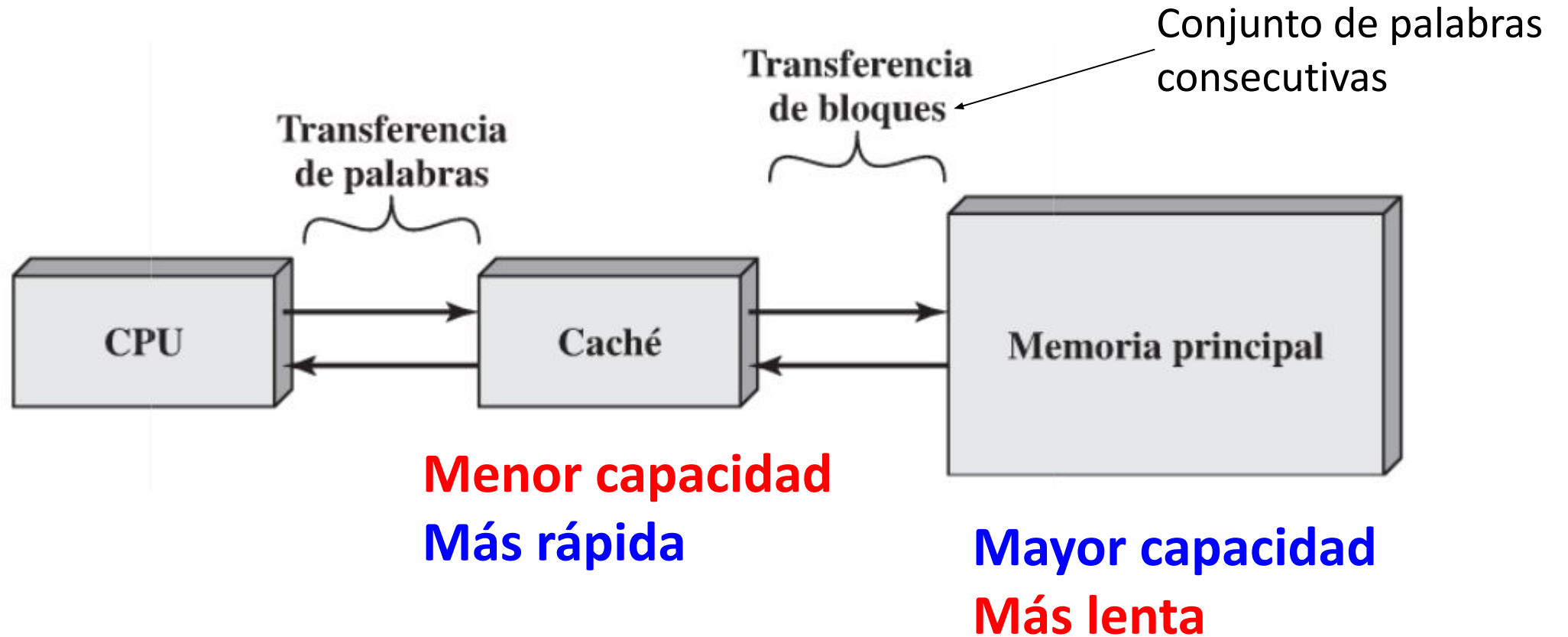
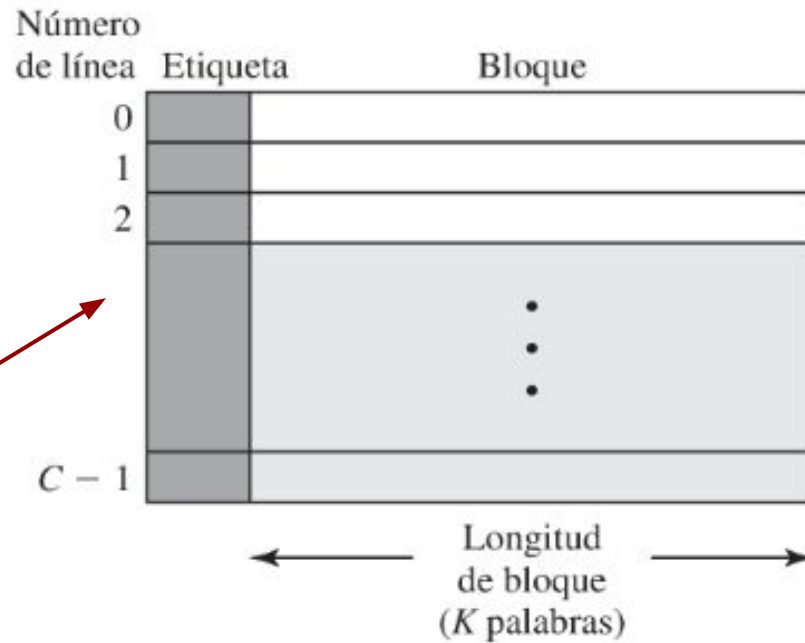


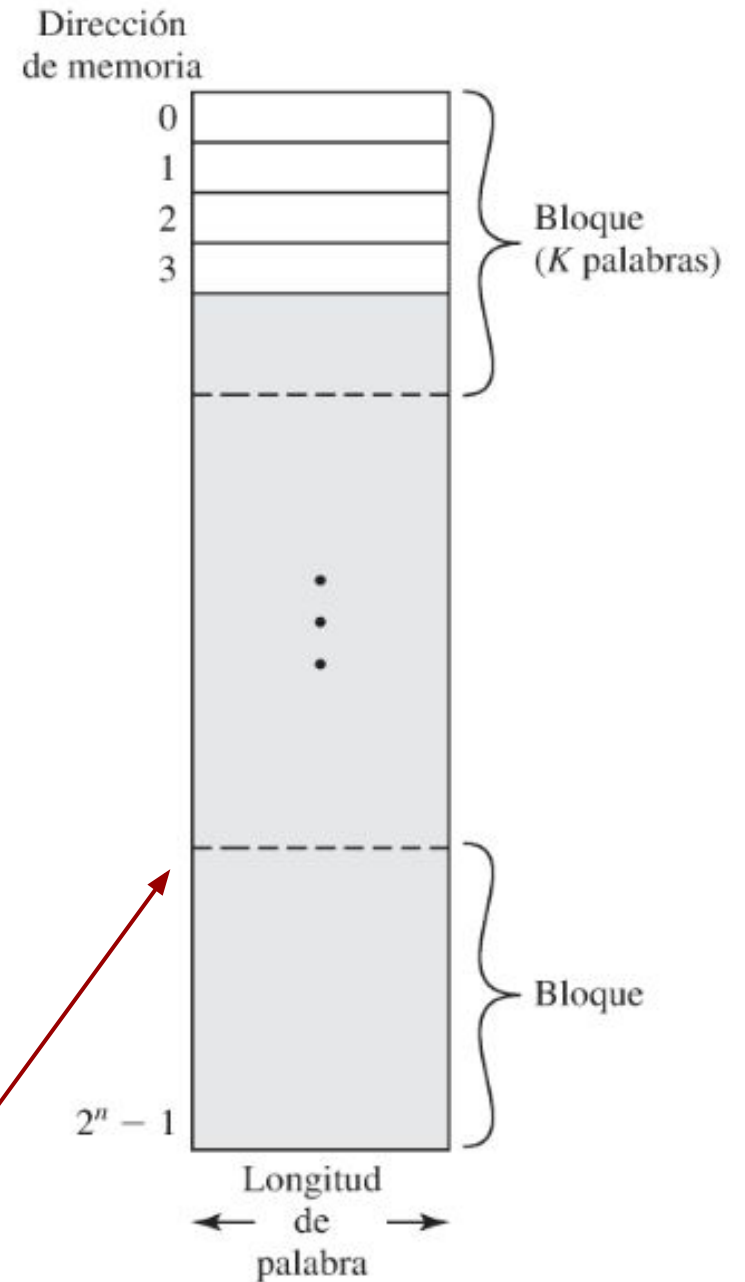
Figura obtenida de Organización y Arquitectura de Computadoras. William Stallings. 7ª ed. pag. 133.

Cada línea almacena un bloque



(a) Caché

Menos bloques



(b) Memoria principal

Mucho más bloques

El procesador direcciona una **palabra**:

- **Si está en caché**
 - El procesador accede a la **palabra** en la caché.
- **Si no está en caché:**
 - Se busca y lleva el **bloque** donde está la palabra desde la memoria principal a la caché.
 - El procesador accede a la **palabra** en la caché.

¿Por qué bloques?: Dos principios

Vecindad espacial: los programas suelen requerir los datos o instrucciones que se encuentran en las direcciones cercanas a las que se procesan.

Vecindad temporal: los programas suelen requerir los datos más recientes.

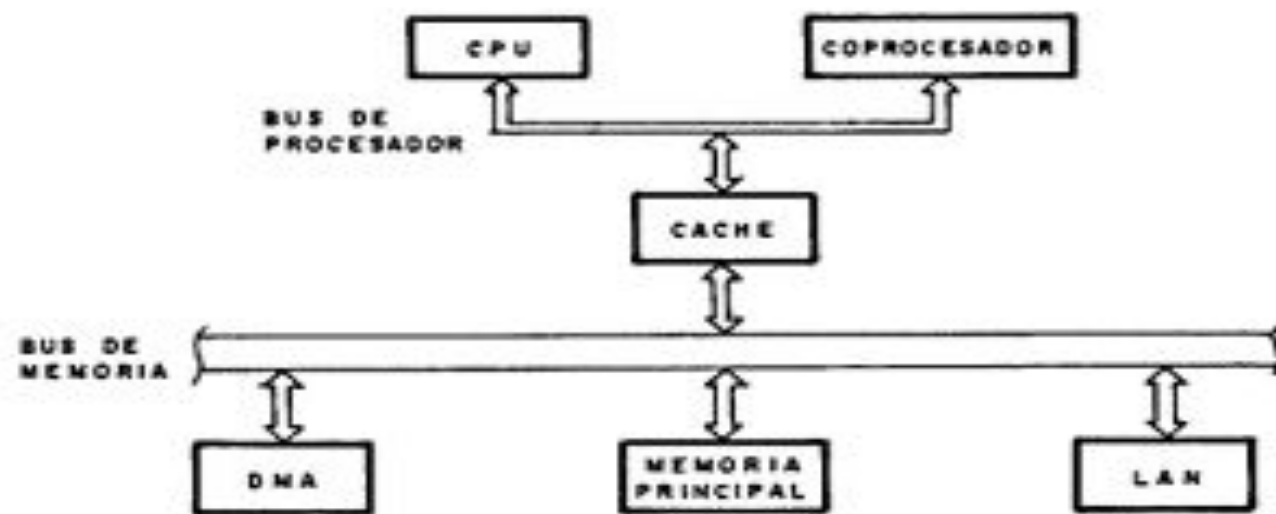
Medida de la eficacia.

Cuando el procesador solicita datos o instrucciones y los encuentra en la caché, se dice que hay presencia (hit), sino decimos que hay ausencia (falla). La eficacia de la caché se puede expresar como

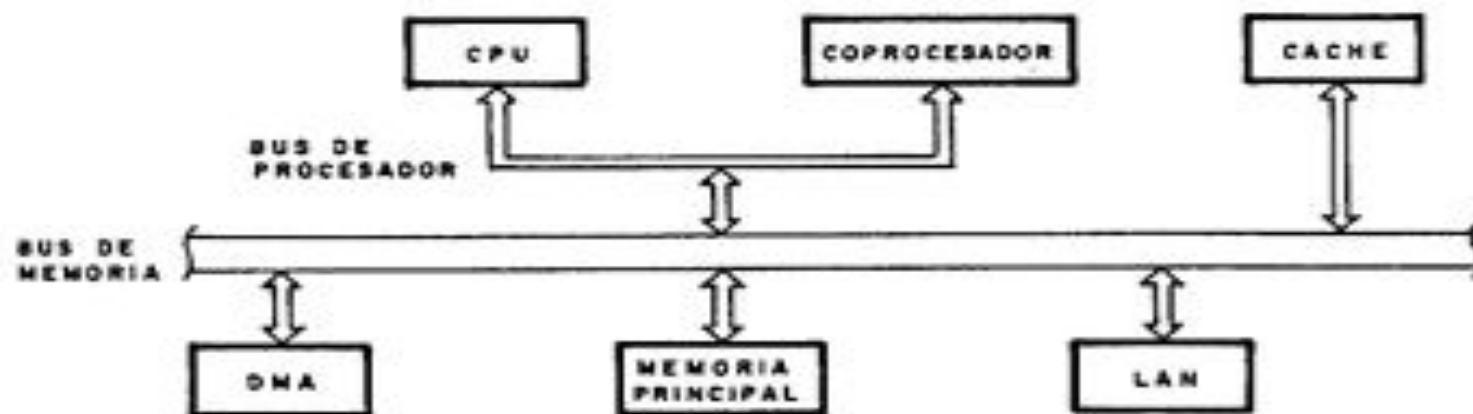
$$\text{Probabilidad de presencia} = \frac{\text{Nº de presencias en la caché}}{\text{Nº total de peticiones a memoria}}$$

La eficacia también depende del software, debido a los principios de la vecindad de las referencias.

Tipos de conexión



a) CONEXION SERIE



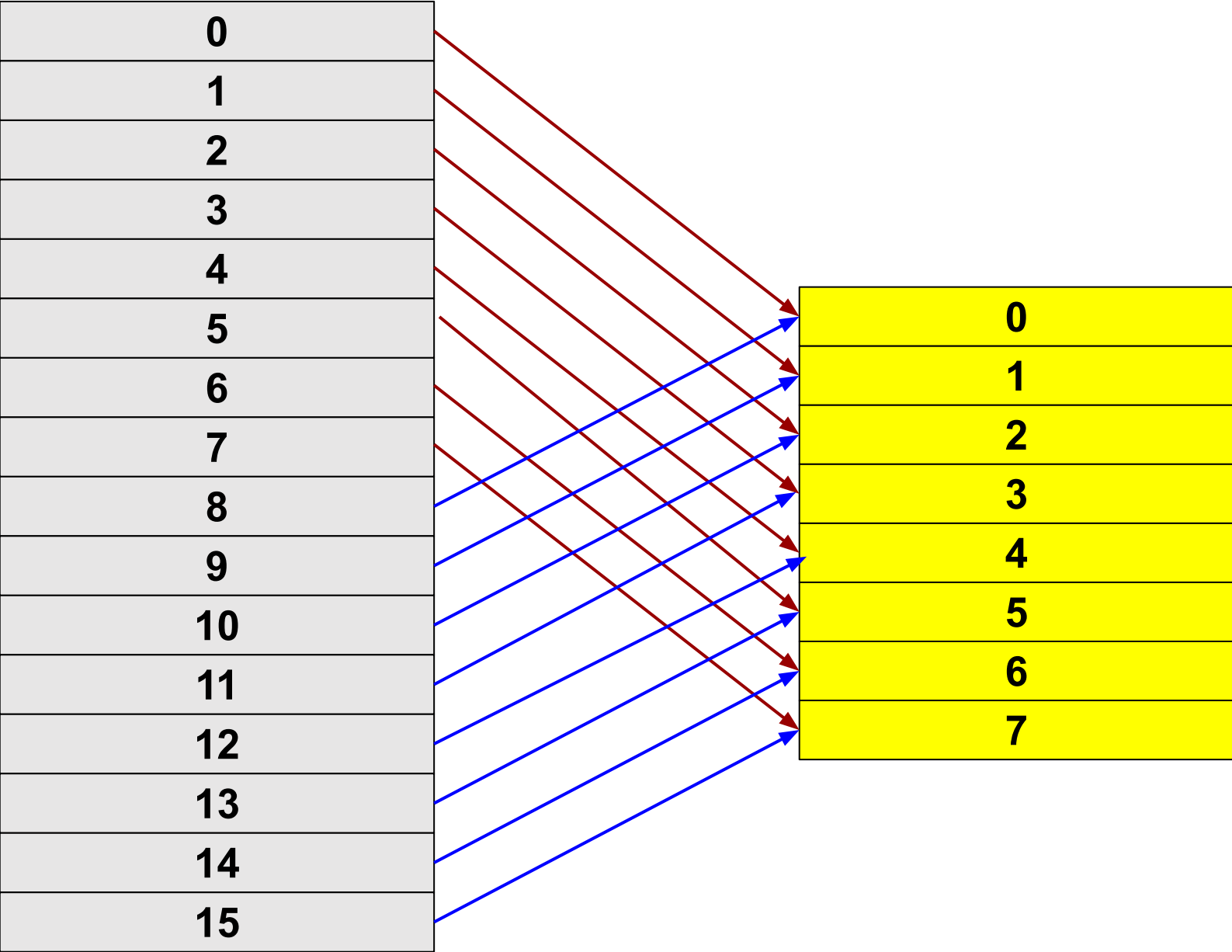
b) CONEXION PARALELO

Función de correspondencia

La caché almacena menos bloques que la memoria principal. ¿Cómo se decide en qué línea de caché se almacena un bloque?

- **Correspondencia directa**: Cada bloque de memoria principal se almacena siempre en la misma línea de caché.
 - Poco eficiente.
- **Correspondencia asociativa**: Los bloques de memoria principal pueden almacenarse en cualquier línea de caché.
- **Correspondencia asociativa por conjuntos**: Combinación de las técnicas anteriores. La caché se divide en conjuntos de k líneas. Cada bloque puede almacenarse en cualquier línea dentro de un conjunto determinado. Se dice que es una caché asociativa por conjuntos de k vías.

Memoria caché de correspondencia directa



Dirección

Etiqueta

Correspondencia asociativa

Línea

Etiqueta 0_{16} { 00000000000000000000=0x0000

 000000000000111111=0x003F

Etiqueta 100_{16} { 0001100100000000=0x1900

 0001100100111111=0x193F

Etiqueta 452_{16} { 0111000100000000=0x7100

 0111000100111111=0x713F

Etiqueta 711_{16} { 1011000111000000=0xB1C0

 1011000111111111=0xB1FF

0xA0
 ...
 0x21

0xB1
 ...
 0x54

0x23
 ...
 0x7C

0x65
 ...
 0x80

711₁₆

452₁₆

0₁₆

100₁₆

0xB1
 ...
 0x54

0xB1
 ...
 0x54

0xA0
 ...
 0x21

0xB1
 ...
 0x54

0

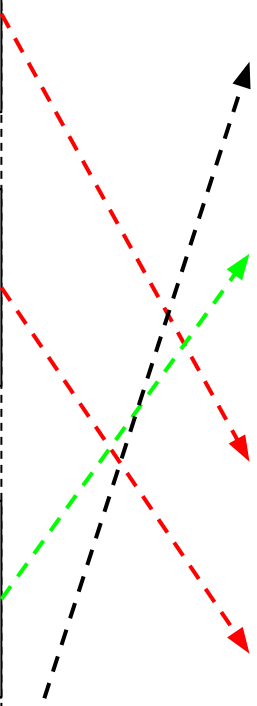
1

2

3

Memoria principal

Memoria caché



Estructura interna caché asociativa

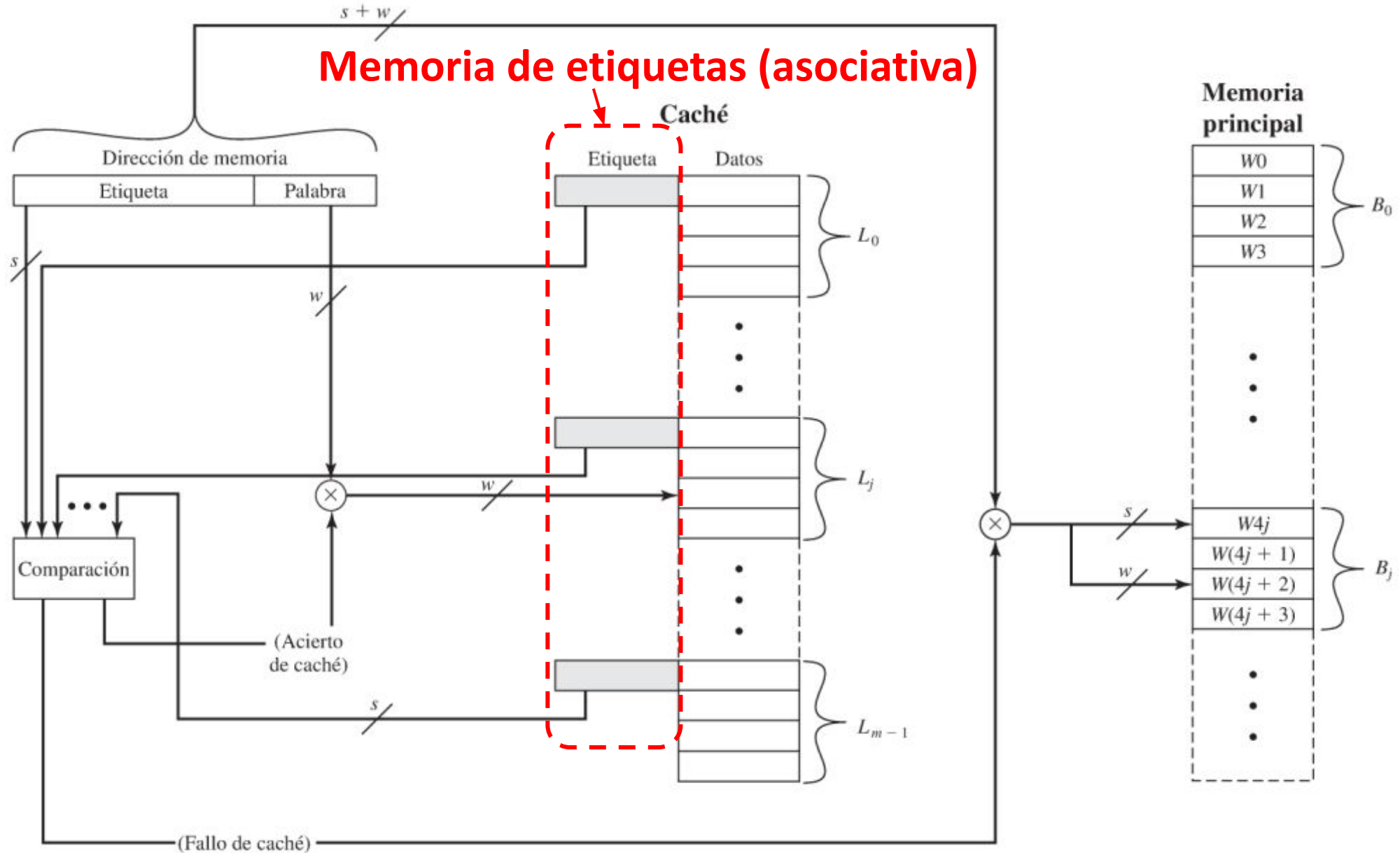


Figura obtenida de Organización y Arquitectura de Computadoras. William Stallings. 7ª ed. pag. 121.

Ejemplo

Memoria virtual de 48 bits (procesadores i3, i5, i7 entre otros). 8 bits por dirección de memoria.

Bloques de 64 bytes (valores usuales son 32, 64 o 128 bytes).

a) Indique la etiqueta del bloque que contiene la dirección de memoria 0x01AB29243A12 e indique el desplazamiento en el bloque.

Bloques de 64 bytes = 2^6 bytes. Se requieren 6 bits por bloque.

$01AB29243A12_{16} =$

$0000\ 0001\ 1010\ 1011\ 0010\ 1001\ 0010\ 0100\ 0011\ 1010\ 0001\ 0010_2$

Etiqueta

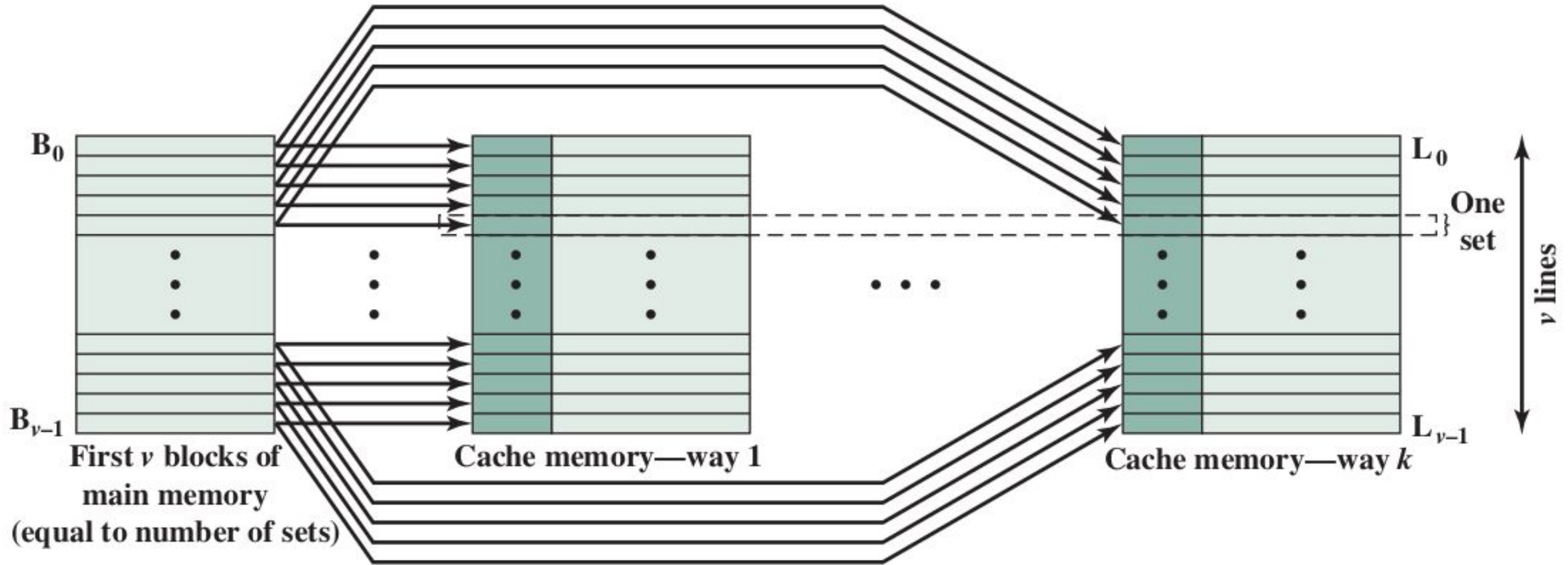
Desplazamiento

Etiqueta: $0000\ 0001\ 1010\ 1011\ 0010\ 1001\ 0010\ 0100\ 0011\ 1010\ 00_2 =$

$6ACA490E8_{16}$

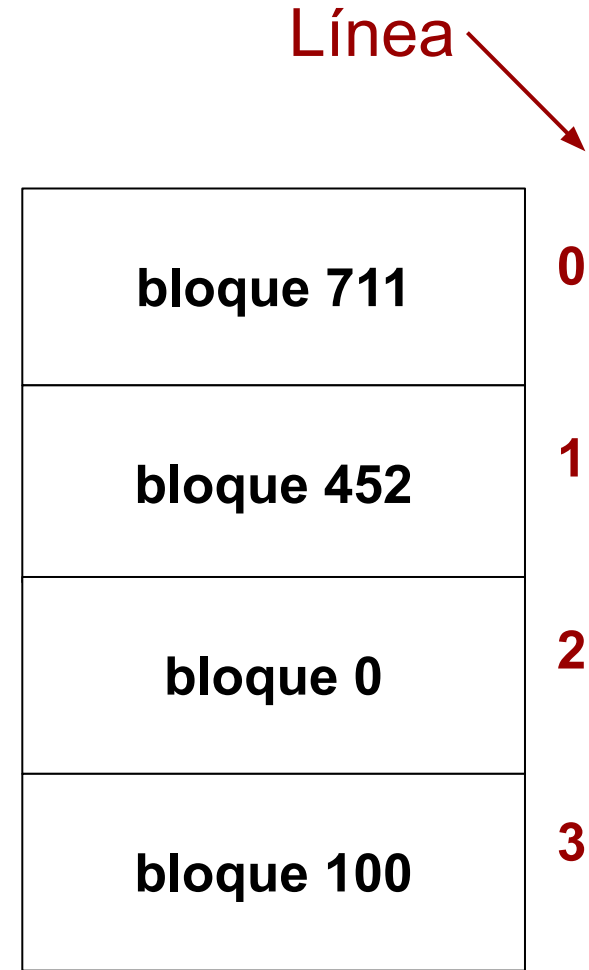
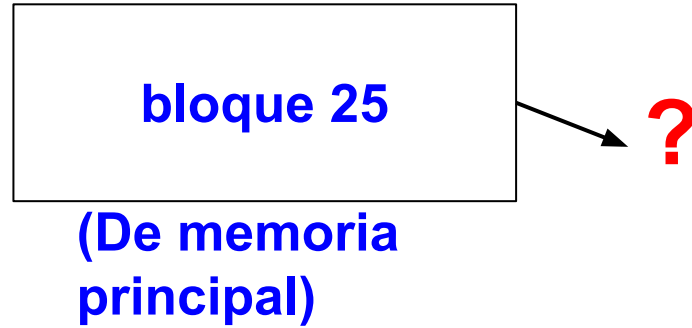
Desplazamiento: $010010_2 = 12_{16}$

Memoria caché asociativa de k vías.



Actualización de la caché (Algoritmos de sustitución)

Los algoritmos de sustitución eligen cuál posición de la caché sustituir cuando debe traerse un bloque de memoria principal a memoria caché.



En las de correspondencia directa sólo puede almacenarse en una posición determinada de la caché.

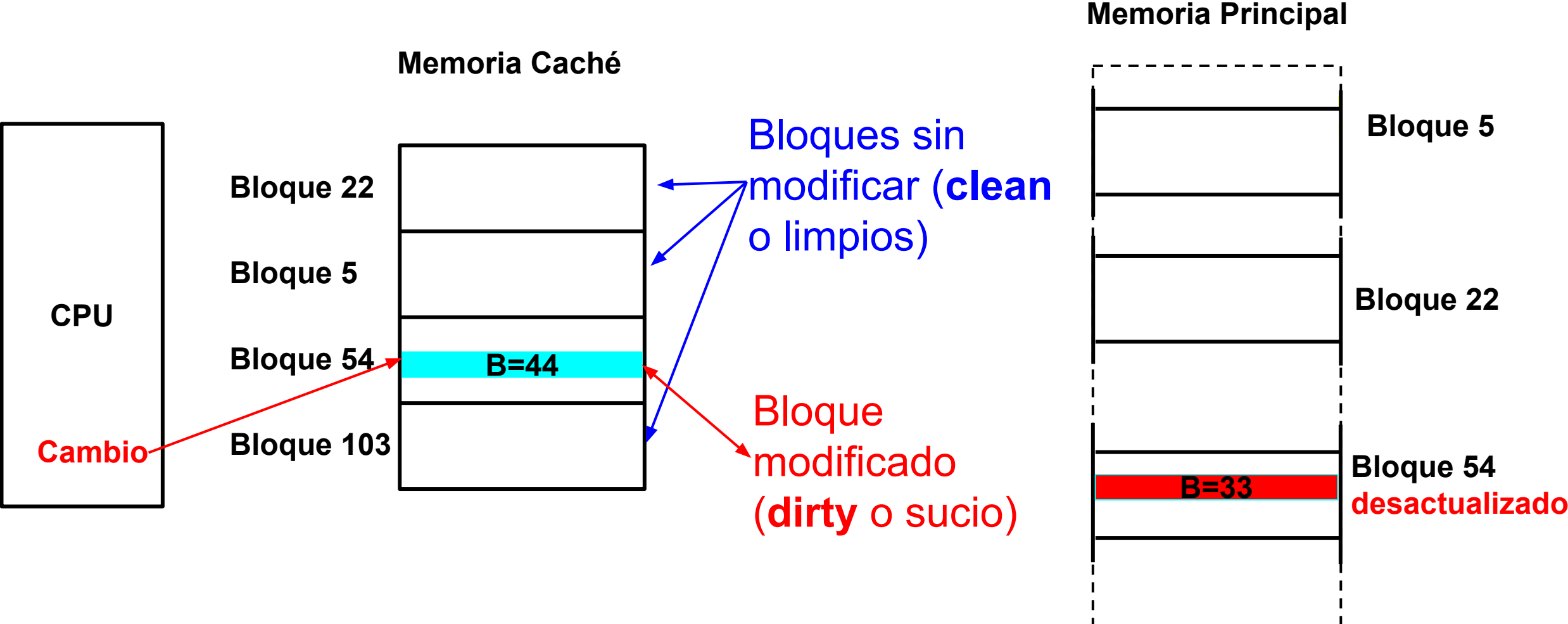
En las otras organizaciones se puede hacer la sustitución según distintos algoritmos. Los más usados son:

- **Random**: Actualiza en forma aleatoria.
- **LRU** ("least recently used" o "menos usado recientemente"): sustituye aquella que lleva más tiempo sin ser accedida.

Memoria caché
llena

Actualización de la memoria principal (Políticas de escritura)

Si el procesador ha modificado un dato en la caché, el bloque correspondiente en memoria principal está **desactualizado** (obsoletos).



Actualización de la memoria principal (Políticas de escritura)

Políticas de actualización de la memoria principal: Definen cuando actualizar un bloque desactualizado en memoria principal. Tres métodos:

Write through (escritura inmediata)

Todas las escrituras del procesador en la cache son traspasadas inmediatamente a la memoria principal. Baja la performance.

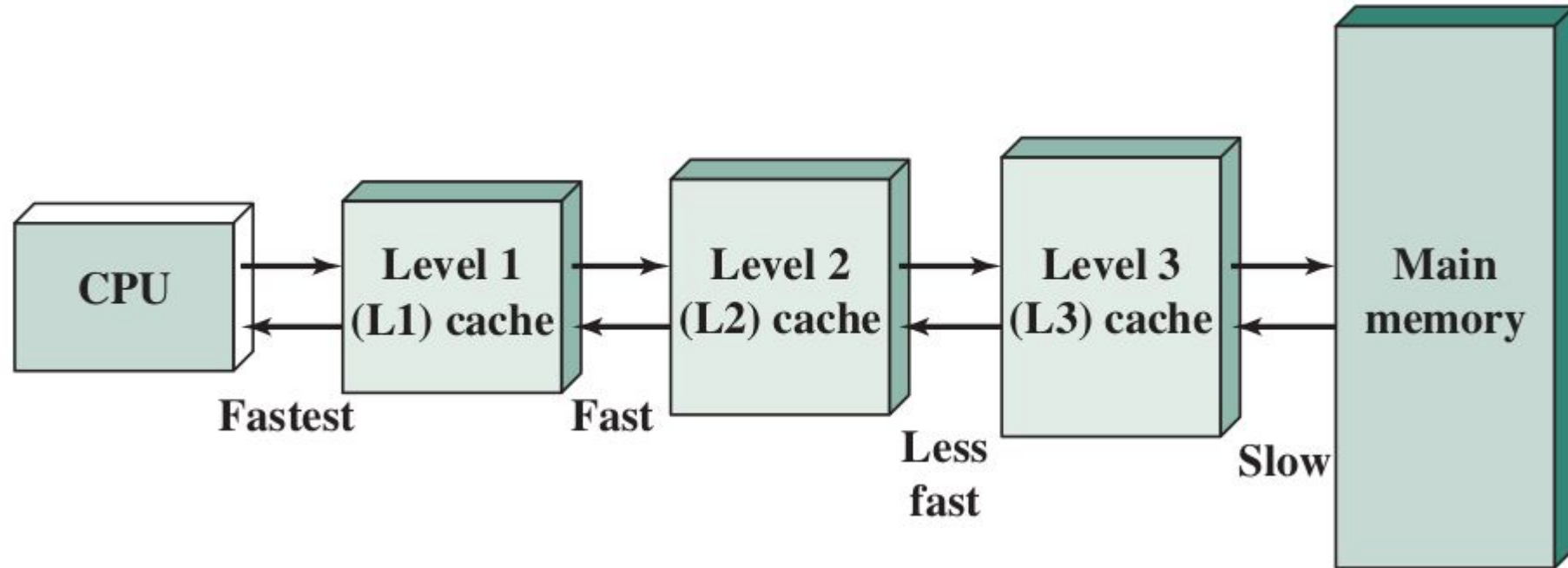
Buffered/posted write through (escritura diferida)

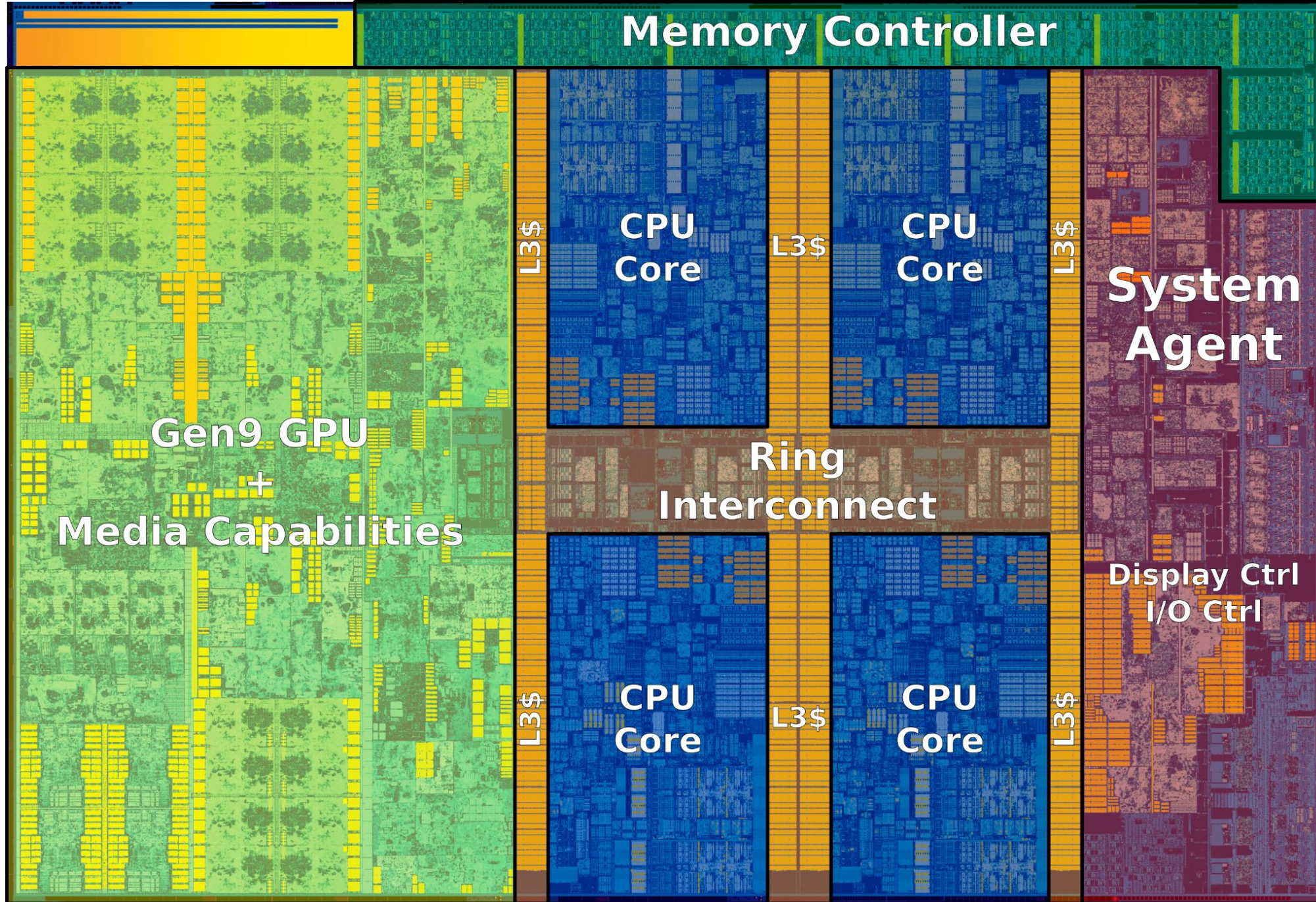
Incorpora registros intermedios (3 a 5), que permiten que el controlador de la caché actualice cuando el bus está ocioso.

Write back (escritura obligada)

Las escrituras del procesador en la caché solamente se hacen en la memoria principal si son estrictamente necesarias (cuando otro procesador accede a la memoria principal o cuando se reemplaza una posición de la caché que ha sido previamente modificada).

Caché multinivel





Microarquitectura Skylake de Intel. Procesadores i3, i5, i7 y Xeon E3.

Core de la microarquitectura Skylake

