Grupo ARCOS

uc3m | Universidad Carlos III de Madrid

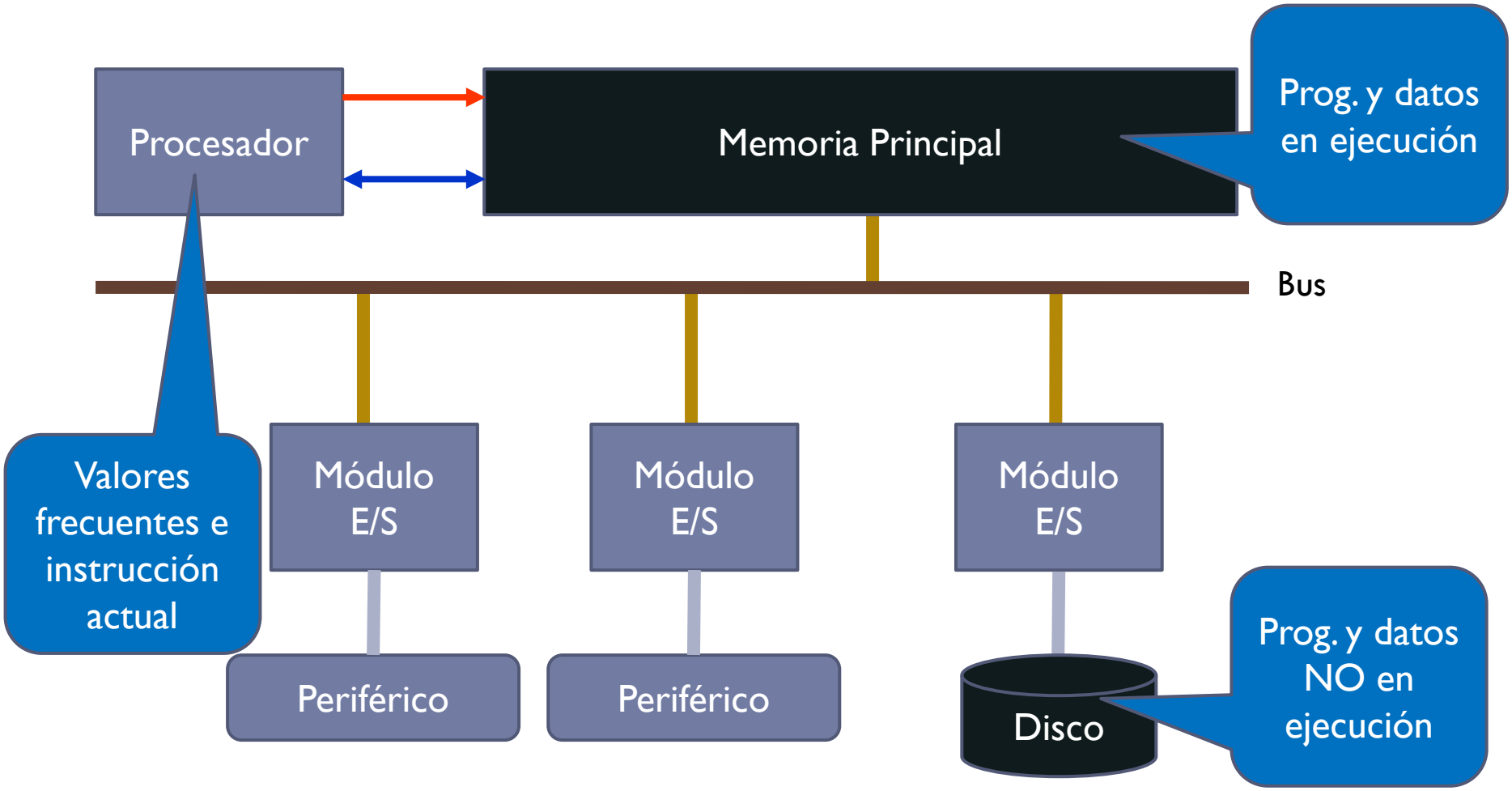
Tema 5
Jerarquía de Memoria

Estructura de Computadores
Grado en Ingeniería Informática

Contenidos

1. Tipos de memoria
2. Jerarquía de memoria
3. Memoria principal
4. Memoria caché
5. Memoria virtual

Visión general del computador

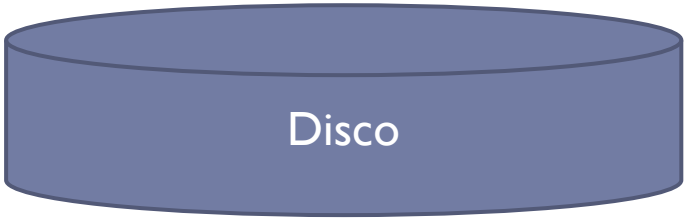


ARCOS @ UC3M
Félix García Carballeira, Alejandro Calderón Mateos

Tipos de memoria (hasta el momento)

- Almacena pocos datos
- Tiempo de acceso a un registro: orden de ns.

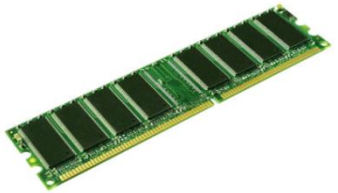
- Más capacidad (GB).
- Tiempo de acceso: 40-100 ns.
  - 1 acceso a memoria = muchos ciclos de reloj



- Capacidad de almacenamiento casi ilimitada.
- Tiempo de acceso lento: orden de milisegundos

Distintos tipos de dispositivos físicos

- Memorias semiconductoras
  - Circuitos electrónicos
  - Ej.: RAM, ROM y Flash



- Memorias magnéticas
  - Información sobre una superficie magnetizada
  - Ej.: Discos duros y cintas

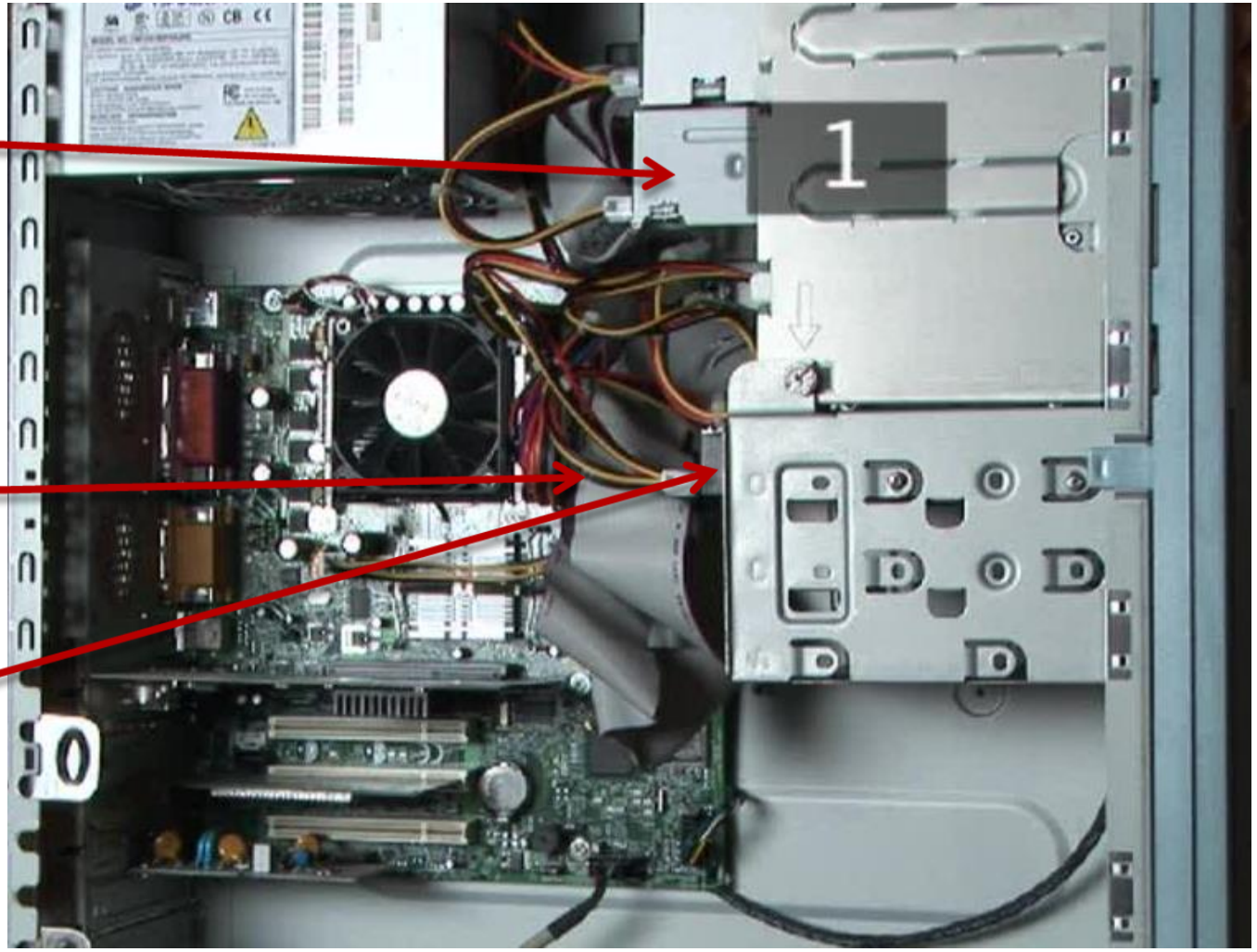
- Memorias ópticas
  - Información grabada con un láser que genera pequeñas perforaciones sobre una superficie
  - Ej.: CD, DVD y blu-ray

¿Dónde se encuentra?

CD-ROM/
DVD-ROM/
BluRay/.

Memoria
RAM

Disco duro



http://www.videojug.com/film/what-components-are-inside-my-computer

Principales características

- Permanencia de los datos:
  - Volátiles (Ej.: RAM)
  - No volátiles (Ej.: ROM, Flash)
- Tipos de operaciones:
  - Lectura y escritura: RAM
  - De solo lectura: ROM
- Organización:
  - Unidad de almacenamiento:
    - Bits, bytes, palabras, bloques, etc.
  - Modo de acceso:
    - Secuencial (Ej.: cinta magnética),
    - Aleatorio (RAM): se puede acceder en cualquier orden. Mismo tiempo de acceso
- Prestaciones:
  - Tiempo de acceso: tiempo entre presentar dirección y obtener un dato
  - Ancho de banda o Velocidad de transferencia: cantidad de datos accedidos por unidad de tiempo
- Otras:
  - Capacidad: cantidad de datos que es posible almacenar
  - Coste: precio por unidad de dato almacenable

Unidades de tamaño

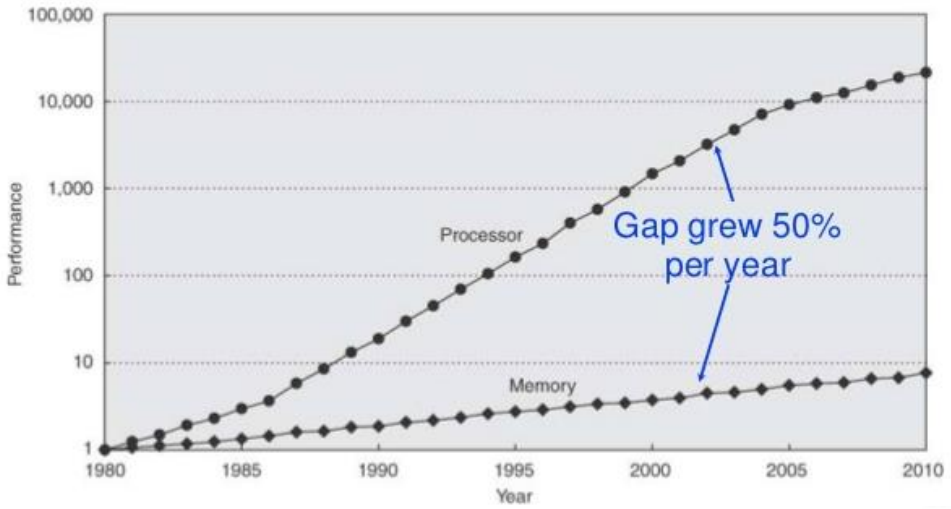
- Normalmente se expresa en octetos o bytes:
  - byte 1 byte = 8 bits
  - kilobyte 1 KB = 1.024 bytes 2^{10} bytes
  - megabyte 1 MB = 1.024 KB 2^{20} bytes
  - gigabyte 1 GB = 1.024 MB 2^{30} bytes
  - terabyte 1 TB = 1.024 GB 2^{40} bytes
  - petabyte 1 PB = 1.024 TB 2^{50} bytes
  - exabyte 1 EB = 1.024 PB 2^{60} bytes
  - zettabyte 1 ZB = 1.024 EB 2^{70} bytes
  - yottabyte 1 YB = 1.024 ZB 2^{80} bytes

Unidades de tamaño (cuidado)

- En comunicación se suele usar el kilobit y no el kilobyte (1 Kb <> 1 KB) y potencias de 10
  - 1 Kb = 1.000 bits
  - 1 KB = 1.000 bytes
- En almacenamiento (discos duros) algunos fabricantes no utilizan potencias de dos, sino potencias de 10:
  - kilobyte 1 KB = 1.000 bytes 10^3 bytes
  - megabyte 1 MB = 1.000 KB 10^6 bytes
  - gigabyte 1 GB = 1.000 MB 10^9 bytes
  - terabyte 1 TB = 1.000 GB 10^{12} bytes
  - …..

Evolución del rendimiento

- Procesadores
  - 1980-2000: Incremento medio del 60% anual.
- Memorias DRAM
  - 1980-2000: Incremento medio del 7% anual.
- La distancia entre memoria y procesador es mayor cada año



Source: Computer Architecture, A Quantitative Approach by John L. Hennessy and David A. Patterson

Número de accesos a memoria

```
int i;
int s = 0;
for (i=0; i < 1000; i++)
     s = s + i;
i=0;
```

- ¿Cuántos accesos a memoria se generan en este fragmento de código?

Número de accesos a memoria

```
int i;
int s = 0;
for (i=0; i < 1000; i++)
     s = s + i;
i=0;
```

```
          li   t0, 0      # s
          li   t1, 0      # i
          li   t2, 1000
bucle1:   bge  t1, t2, fin1
          add  t0, t0, t1
          addi t1, t1, 1
          beq  x0, x0, bucle1
fin1:     li   t1, 0
```

Número de accesos a memoria

```
int i;
int s = 0;
for (i=0; i < 1000; i++)
     s = s + i;
i=0;
```

```
          li   t0, 0      # s
          li   t1, 0      # i
          li   t2, 1000
bucle1:   bge  t1, t2, fin1
          add  t0, t0, t1
          addi t1, t1, 1
          beq  x0, x0, bucle1
fin1:     li   t1, 0
```

Solución: 3 + 4 × 1000 + 1 + 1 = 4005

Número de accesos a memoria

```
int i;
int s = 0;
for (i=0; i < 1000; i++)
     s = s + i;
i=0;
```

```
          li   t0, 0      # s
          li   t1, 0      # i
          li   t2, 1000
bucle1:   bge  t1, t2, fin1
          add  t0, t0, t1
          addi t1, t1, 1
          j    bucle1
fin1:     li   t1, 0
```

Solución: 3 + 4 × 1000 + 1 + 1 = 4005

Con una memoria de 60 ns el tiempo total sería 240300 ns

Un procesador típico dedicaría más del 98% de su tiempo a esperar datos de memoria

Número de accesos a memoria

```
int v[1000];  // global

int i;
for (i=0; i < 1000; i++)
     v[i] = 0;
```

Número de accesos a memoria

```
int v[1000];  // global

int i;
for (i=0; i < 1000; i++)
     v[i] = 0;
```

```
.data
        v: .space 4000

.text:
          li   t0, 0     # i
          li   t1, 0     # i de v
          li   t2, 1000 # componentes
bucle2: bgt  t0, t2, fin2
          sw   0,  v(t1)
          addi t0, t0, 1
          addi t1, t1, 4
          j    bucle2
fin2:
```

Número de accesos a memoria

```
int v[1000];  // global

int i;
for (i=0; i < 1000; i++)
     v[i] = 0;
```

```
.data
        v: .space 4000

.text:
          li   t0, 0     # i
          li   t1, 0     # i de v
          li   t2, 1000  # componentes
bucle2:   bgt  t0, t2, fin2
          sw   0,  v(t1)
          addi t0, t0, 1
          addi t1, t1, 4
          j    bucle2
fin2:
```

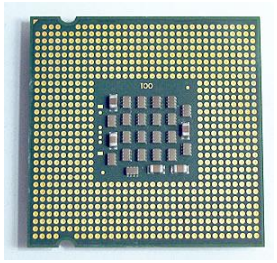
Solución:

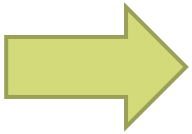
3 + 5 × 1000 + 1 + 1000 (acceso adicional de sw) = 6004

Contenidos

1. Tipos e memoria
2. Jerarquía de memoria
3. Memoria principal
4. Memoria caché
5. Memoria virtual

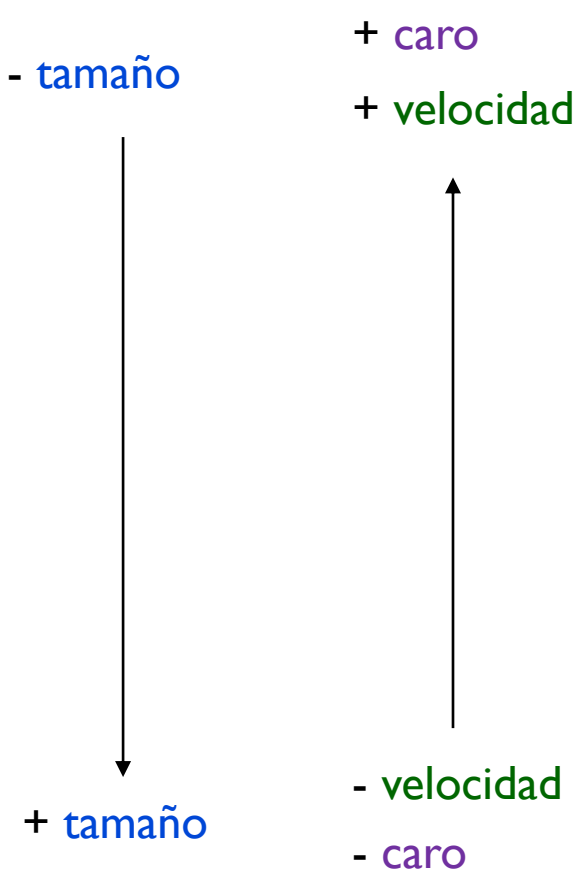
¿Cómo sería el sistema de memoria ideal?





- Minimiza tiempo de acceso
- Maximiza la capacidad
- Minimiza el coste

Realidad



- Objetivos incompatibles entre si:
  - + velocidad ⇨ - tamaño
- Se usan distintos tipos de memoria:
  - DRAM, Disco Duro, …
- Se organizan los distintos tipos de memoria por velocidad de acceso:
  - Jerarquía de memoria

Jerarquía de memoria

- tamaño

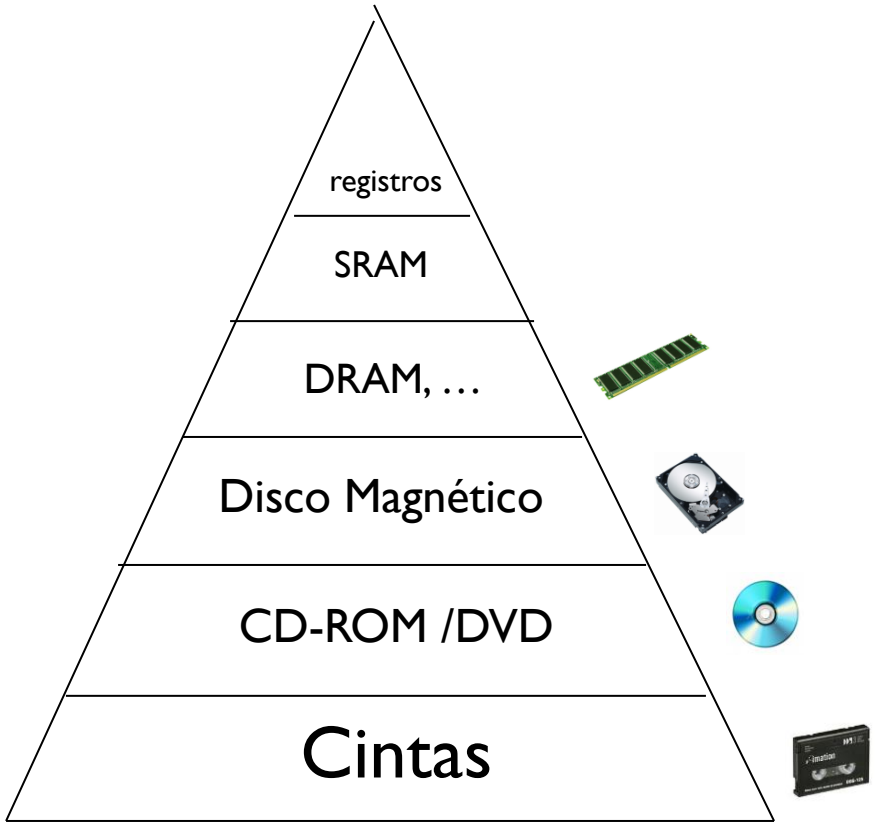
+ tamaño

+ caro

+ velocidad

- velocidad

- caro



Uso de la jerarquía de memoria:
diferentes tiempos de acceso

- T. acceso a registro
  - ~1 ns
- T. acceso a SRAM
  - ~2-5 ns
- T. acceso a DRAM
  - ~70-100 ns

La biblioteca de la UC3M...

La biblioteca de la UPC…

Una biblioteca en Florida…

Comparación

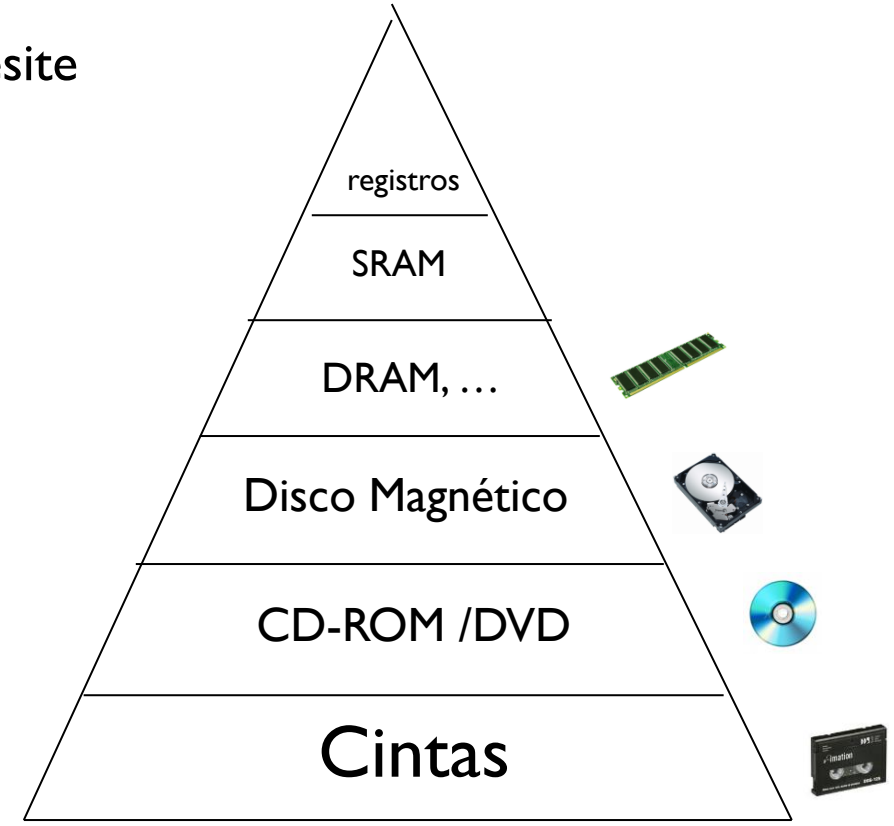
Technology	Bytes per Access (typ.)	Latency per Access	Cost per Megabyte[a]	Energy per Access
On-chip Cache	10	100 of picoseconds	$1–100	1 nJ
Off-chip Cache	100	Nanoseconds	$1–10	10–100 nJ
DRAM	1000 (internally fetched)	10–100 nanoseconds	$0.1	1–100 nJ (per device)
Disk	1000	Milliseconds	$0.001	100–1000 mJ

Memory Systems
Cache, DRAM, Disk
Bruce Jacob, Spencer Ng, David Wang
Elsevier

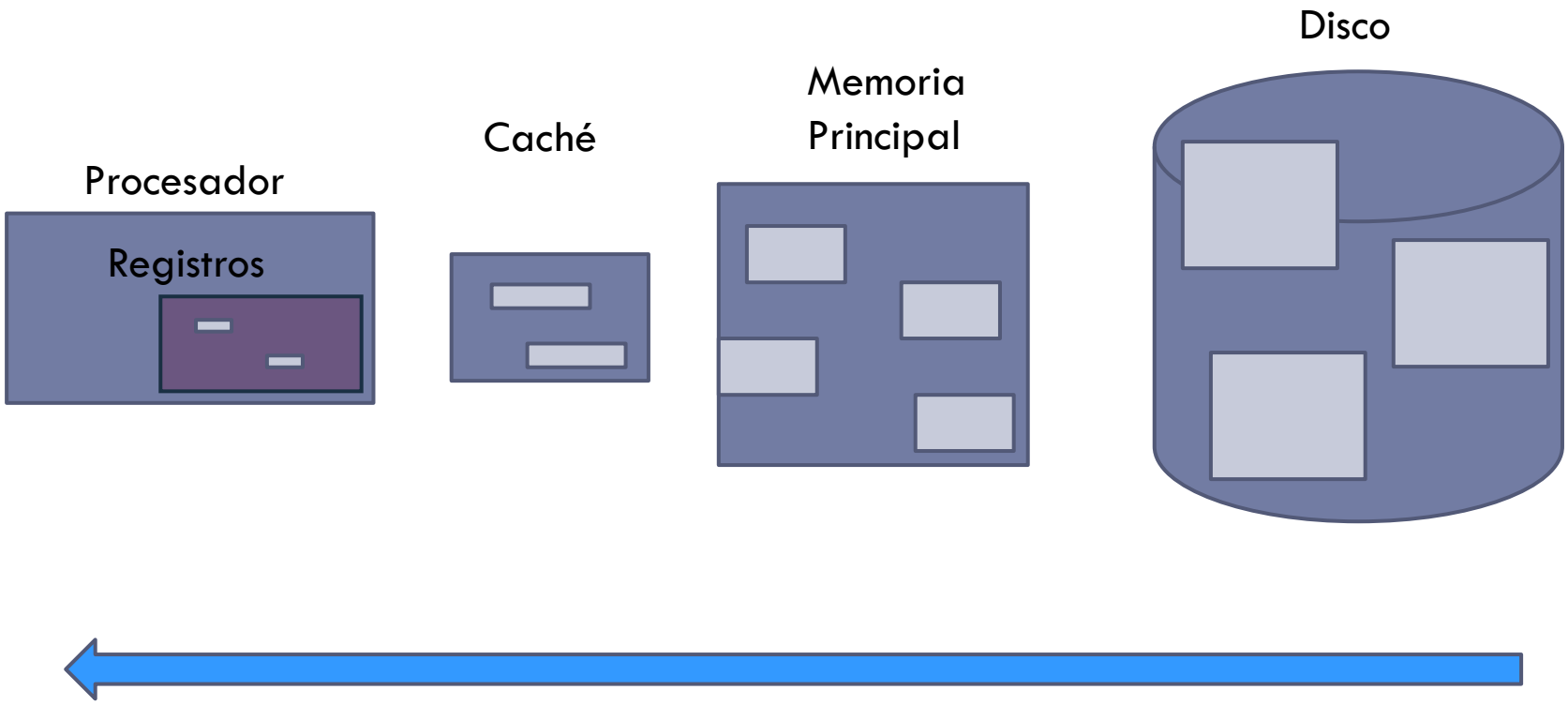
Uso de la jerarquía de memoria

- Solo en memoria lo que se necesite en un instante dado.
- Si no está, se copia de un nivel a otro la porción necesaria:
  - Ej.: cargar un programa en RAM
- Cuando no se necesite, se borra la copia realizada.

- El comportamiento de los accesos lo favorece:
  - Proximidad de referencias



Idea de la jerarquía de memoria



Diseño de la jerarquía de memoria

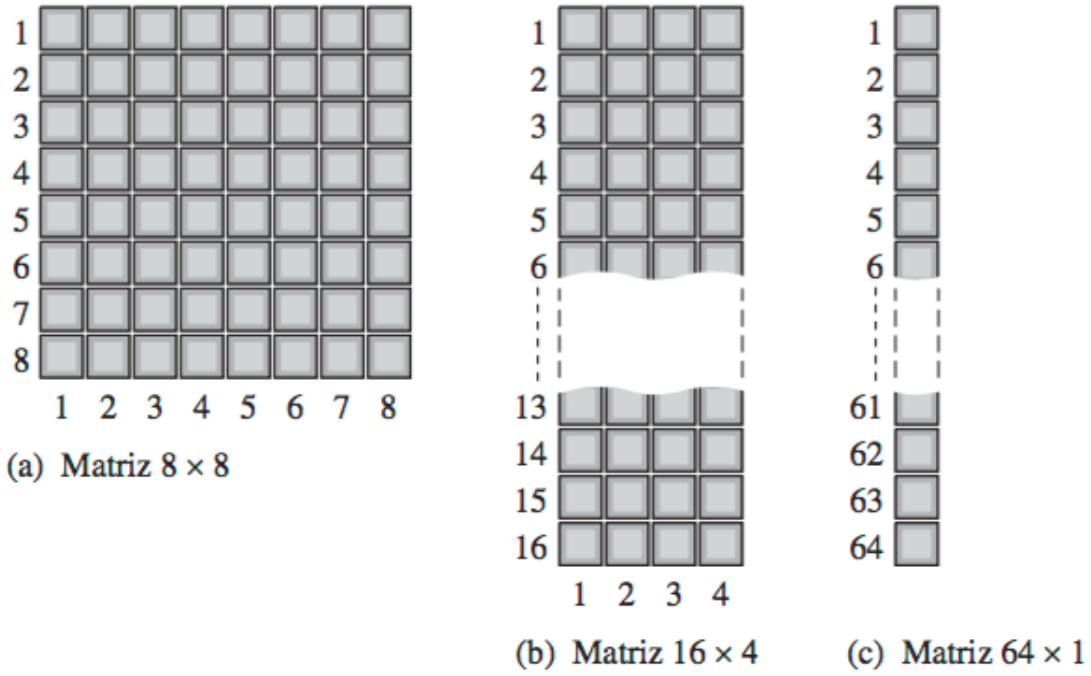
- El diseño de la jerarquía de memoria es crucial en procesadores multicore
- El ancho de banda crece con el número de cores
  - Un Intel Core i7 genera dos accesos a memoria por core y por ciclo de reloj
  - Con 4 cores y 3.2 GHz de frecuencia de reloj
    - 25.6 billones de accesos a datos de 64 bit/segundo +
    - 12.8 billones de accesos de 128 bits para instrucciones = 409.6 GB/s
  - Una memoria DRAM solo ofrece un 6% (25GB/s)
  - Se requiere:
    - Memorias multi puerto
    - Niveles de memoria caché

Memorias de semiconductores

- Memoria de solo lectura (ROM)
  - No necesita alimentación
  - Persistente
  - Ejemplo de uso: BIOS
- Memoria de lectura/escritura (RAM)
  - Necesita alimentación
  - No persistente
  - Más rápida que la ROM
  - Ejemplo de uso: memoria principal

Matriz de memoria semiconductora

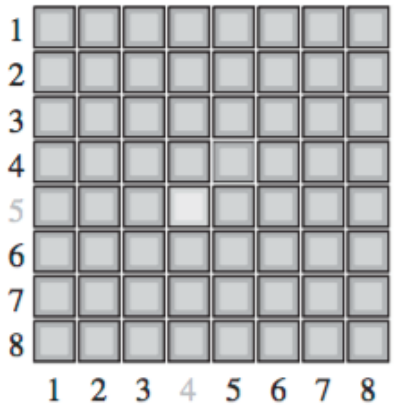
- Cada celda almacena un 1 o un 0



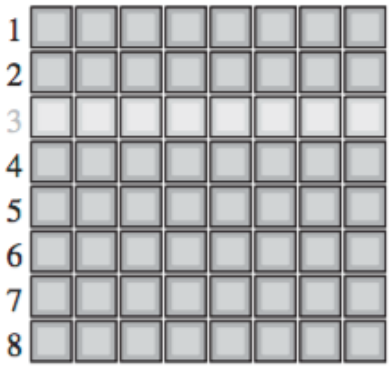
Fundamenros de Sistemas Digitales
Thomas L. Floyd

Direcciones y capacidad

- Dirección: posición de una unidad de datos en la matriz de memoria



(a) La dirección del bit gris claro es fila 5, columna 4.

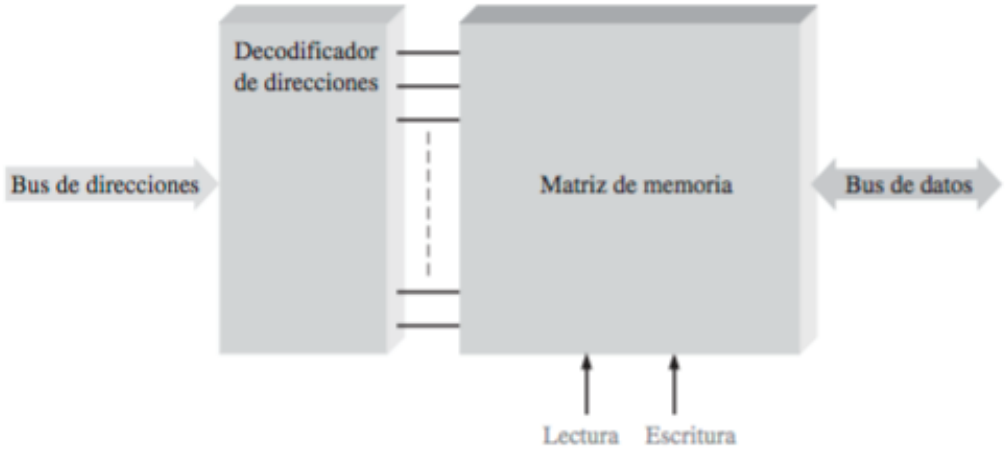


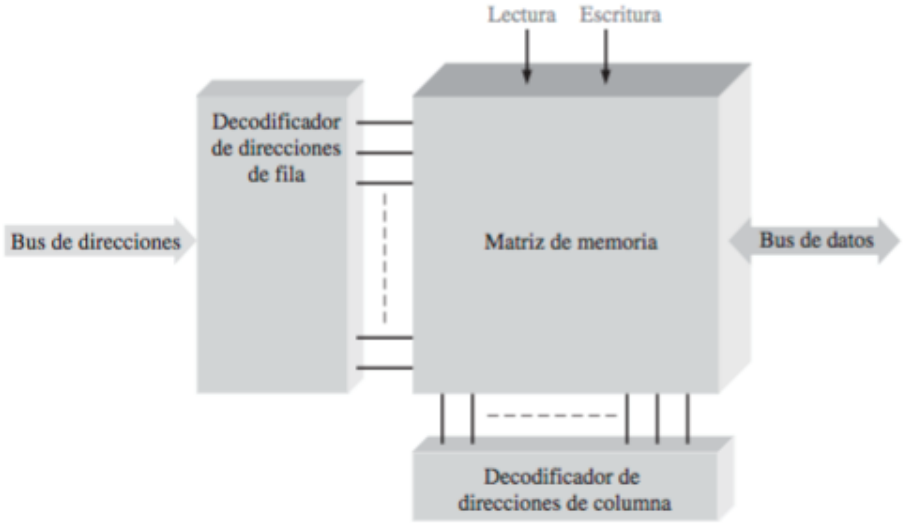
(b) La dirección del byte gris claro es la fila 3.

Fundamenros de Sistemas Digitales
Thomas L. Floyd

- Capacidad: número total de unidades de datos que se pueden almacenar

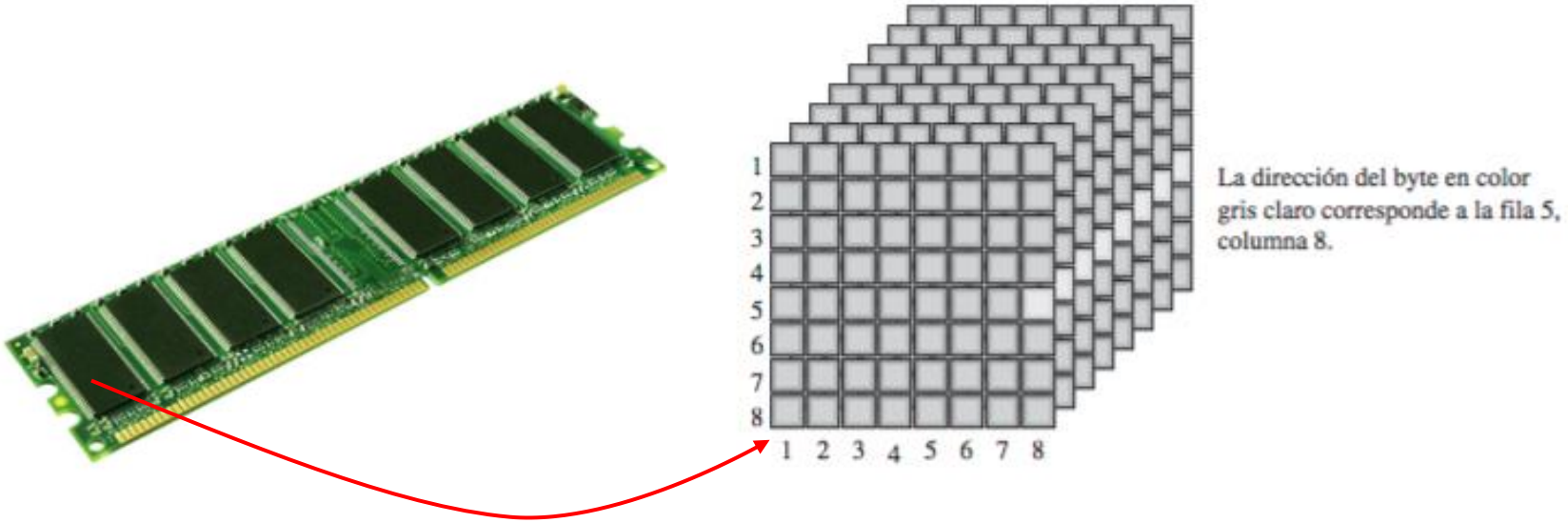
Tipos de direccionamientos



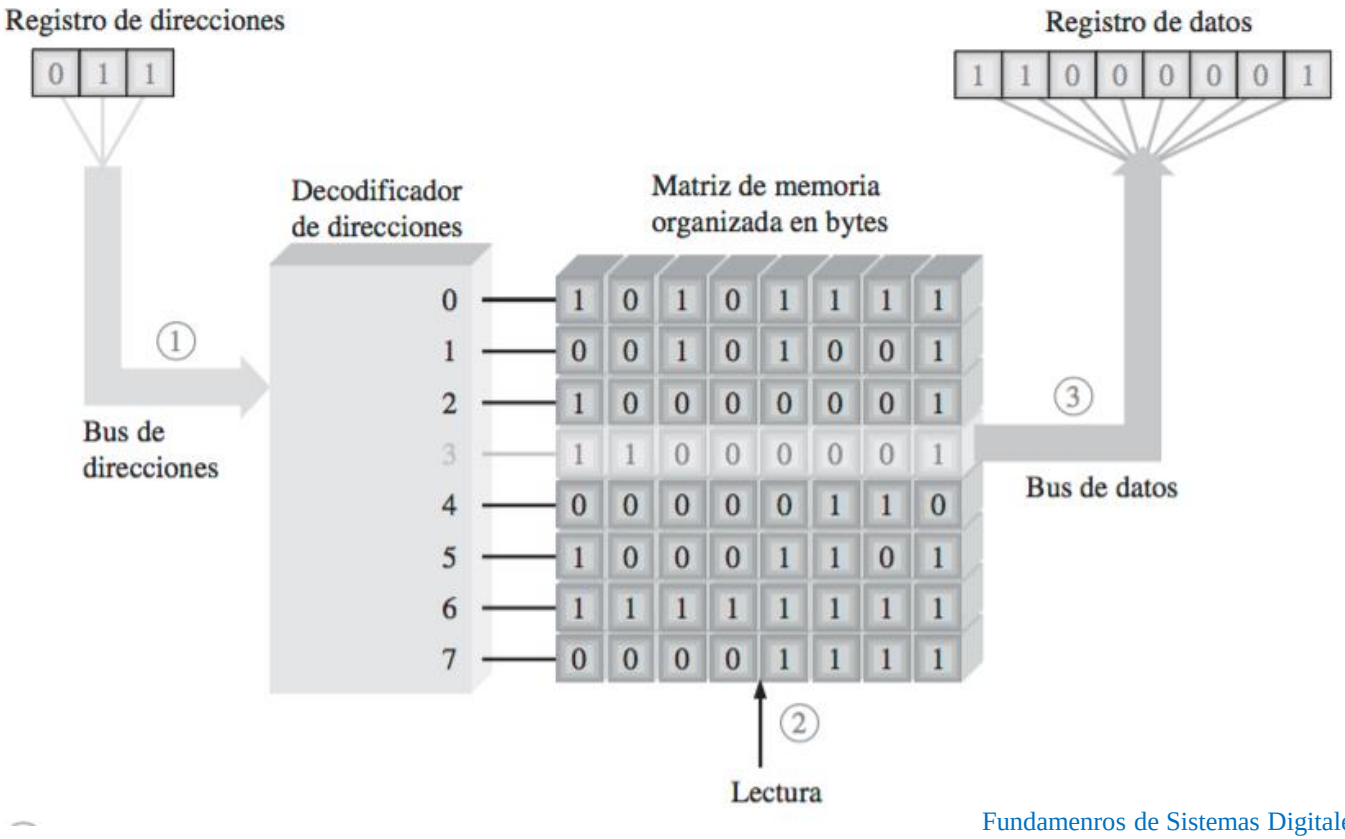


Fundamenros de Sistemas Digitales
Thomas L. Floyd

Ejemplo de organización

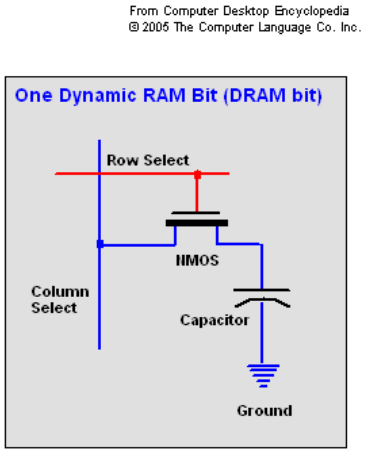


Operación de lectura

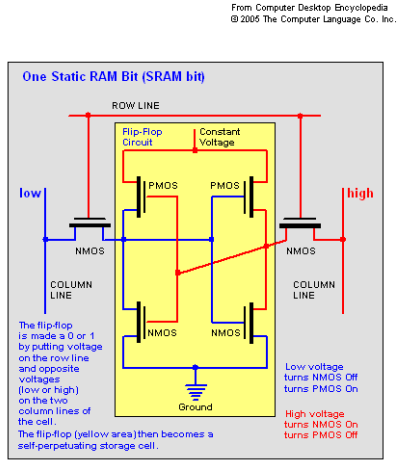


Fundamenros de Sistemas Digitales
Thomas L. Floyd

Memoria RAM (memorias de acceso aleaotorio)



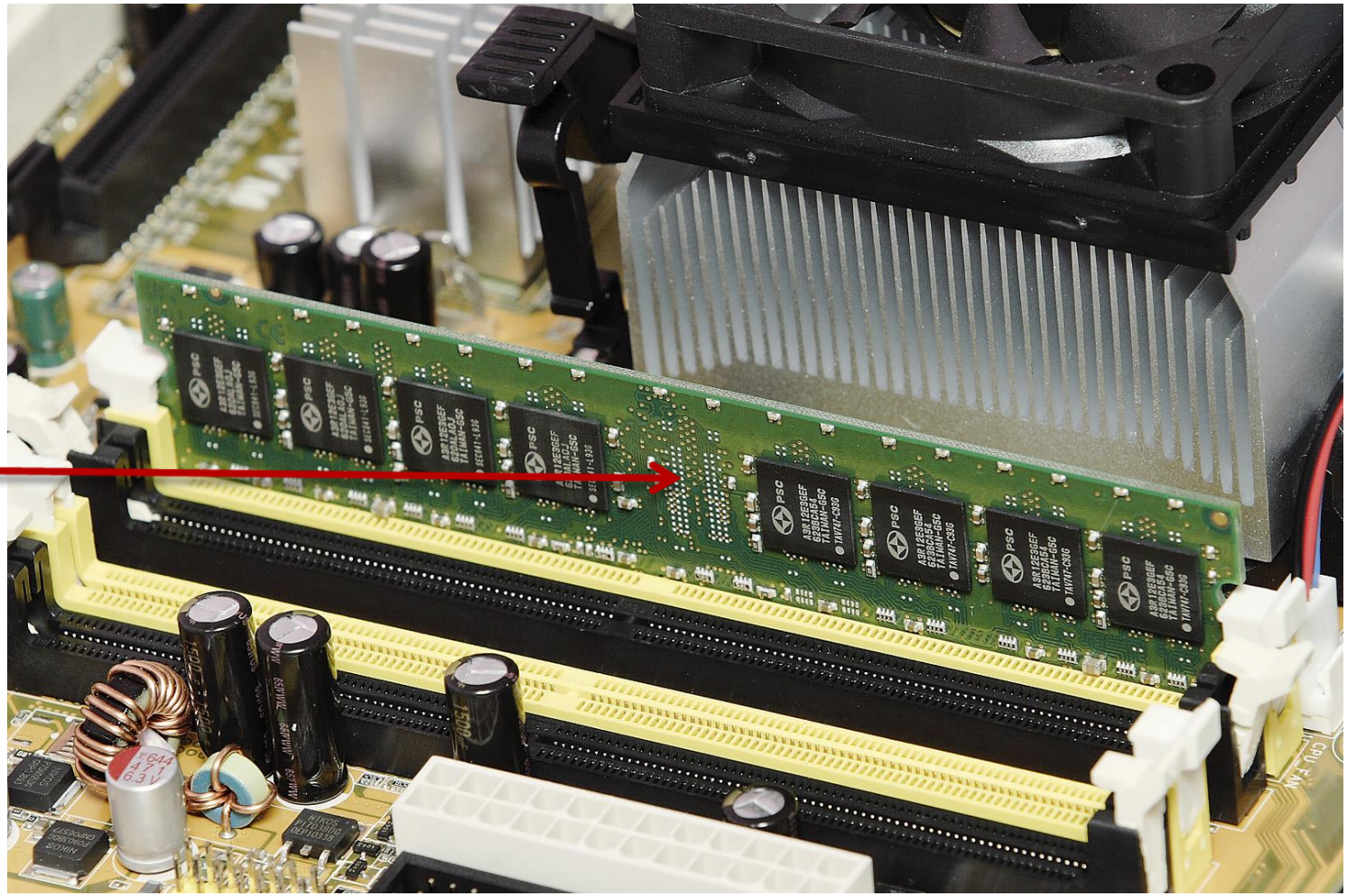
- RAM dinámica (DRAM)
  - Almacena bits como carga en condensadores.
  - Tiende a descargarse: necesita refrescos periódicos.
    - Ventaja: construcción más simple, **más almacenamiento**, más económica
    - Inconveniente: necesita circuitería de refresco, **más lenta**.
      - 2%-3% de los ciclos de reloj consume el refresco
    - Utilizada en memorias principales



- RAM estática (SRAM)
  - Almacena bits como interruptores en *on* y *off*.
  - Tiende a no descargarse: **no** necesita refresco.
    - Ventaja: No necesita circuitería de refresco, **más rápida**.
    - Inconveniente: Construcción compleja, **menos almacenamiento**, más cara.
    - Utilizada en memorias cachés

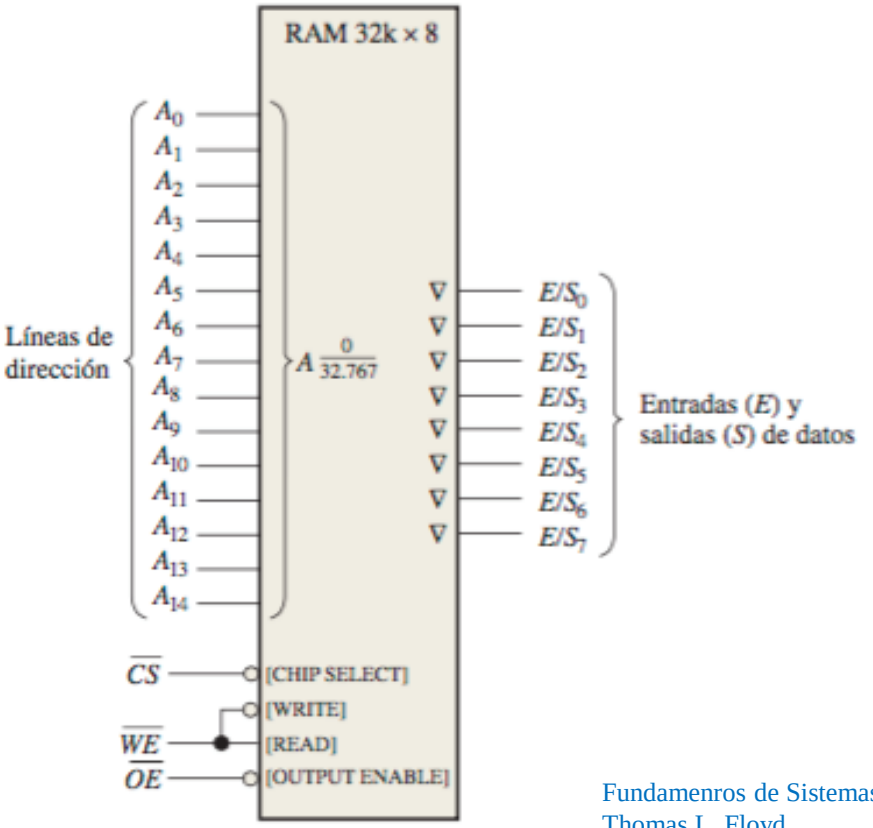
¿Dónde se encuentra la memoria DRAM?

Memoria DRAM

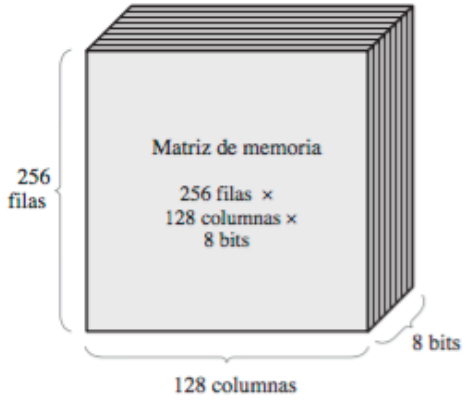


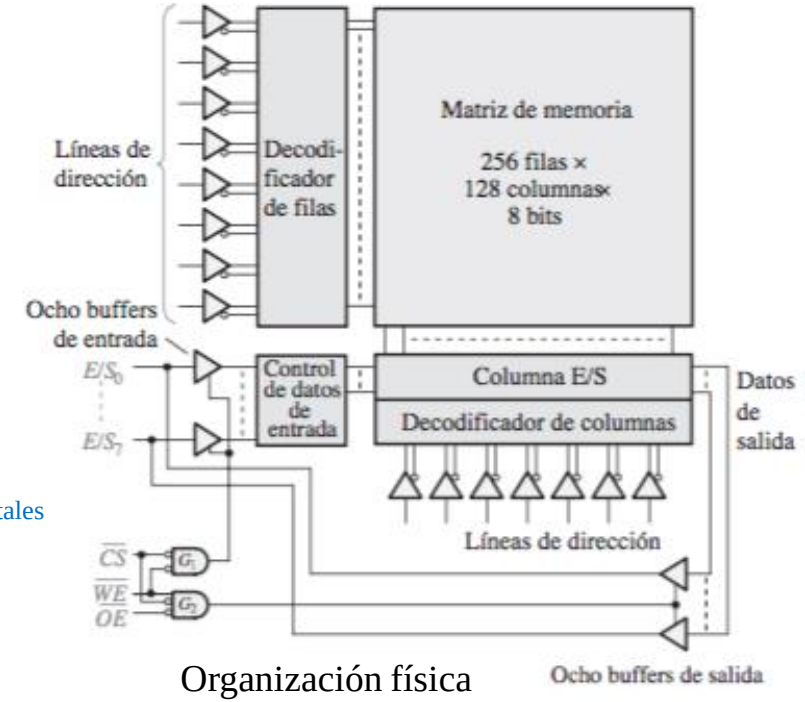
http://en.wikipedia.org/wiki/Primary_storage#Primary_storage

Ejemplo de memoria SRAM



Organización lógica

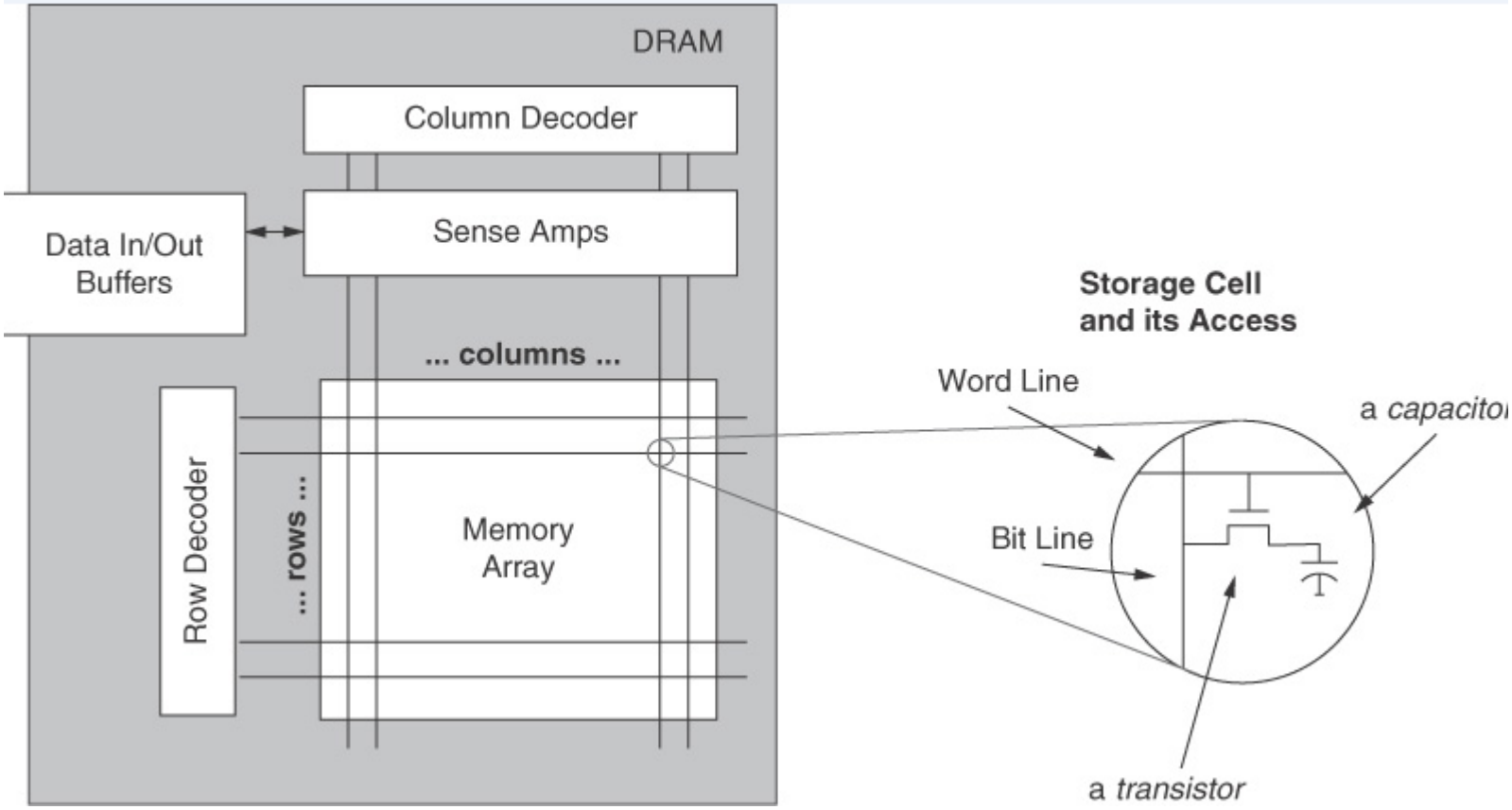




Organización física

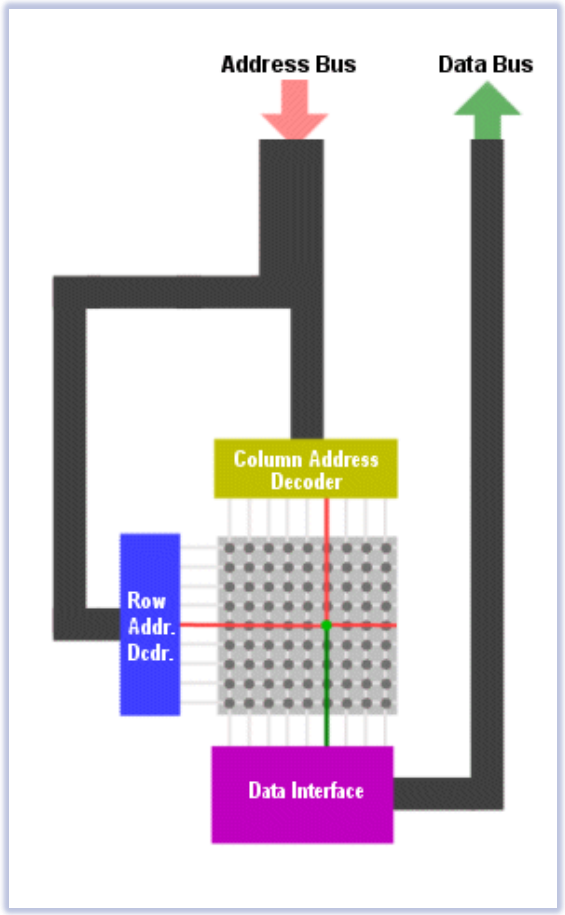
Fundamenros de Sistemas Digitales
Thomas L. Floyd

Estructura de una memoria DRAM

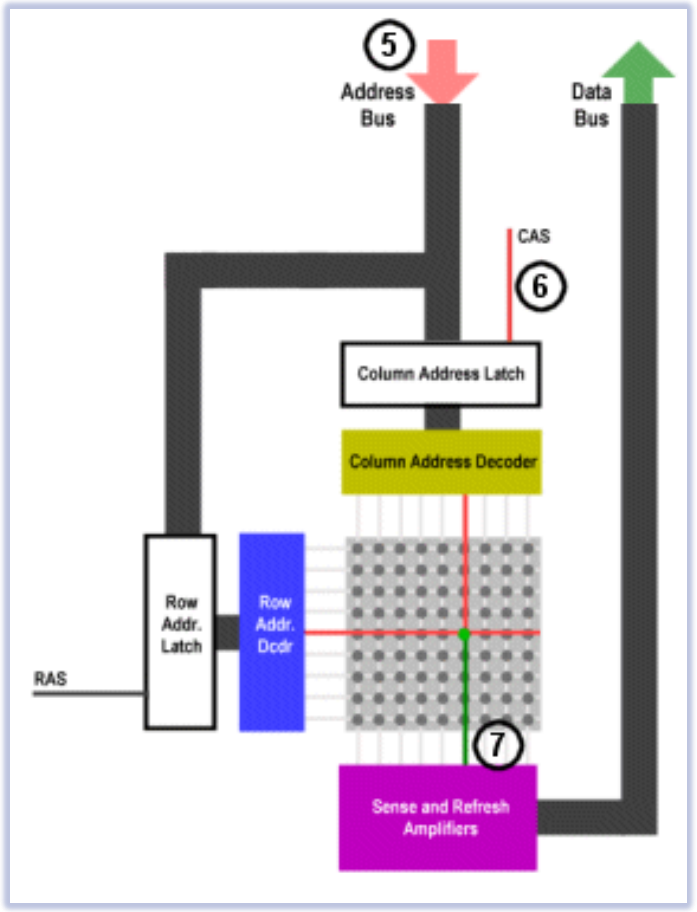


Memory Systems
Cache, DRAM, Disk
Bruce Jacob, Spencer Ng, David Wang
Elsevier

Multiplexción de direcciones en DRAM

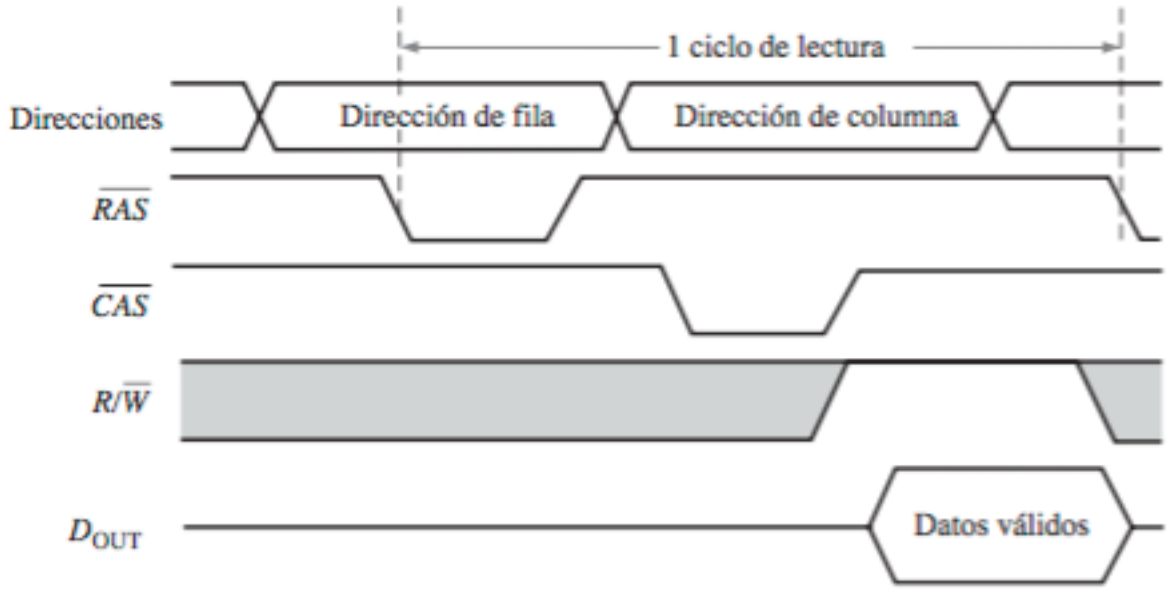


Direccionamiento
por fila/columna



Direccionamiento
por fila/columna con CAS/RAS

Operación de lectura con CAS/RAS



Ciclos de refresco

- Una DRAM almacena un bit en un condensador
- Esta carga se degrada con el tiempo y la temperatura
- Necesario refrescar cada bit
- Típicamente una DRAM se debe refrescar cada pocos milisegundos
- Una operación de lectura refresca todas las direcciones de una fila
- Una DRAM utiliza ciclos de refresco

Velocidad de las memorias DRAM

Production year	Chip size	DRAM Type	Slowest DRAM (ns)	Fastest DRAM (ns)	Column access strobe (CAS)/ data transfer time (ns)	Cycle time (ns)
1980	64K bit	DRAM	180	150	75	250
1983	256K bit	DRAM	150	120	50	220
1986	1M bit	DRAM	120	100	25	190
1989	4M bit	DRAM	100	80	20	165
1992	16M bit	DRAM	80	60	15	120
1996	64M bit	SDRAM	70	50	12	110
1998	128M bit	SDRAM	70	50	10	100
2000	256M bit	DDR1	65	45	7	90
2002	512M bit	DDR1	60	40	5	80
2004	1G bit	DDR2	55	35	5	70
2006	2G bit	DDR2	50	30	2.5	60
2010	4G bit	DDR3	36	28	1	37
2012	8G bit	DDR3	30	24	0.5	31

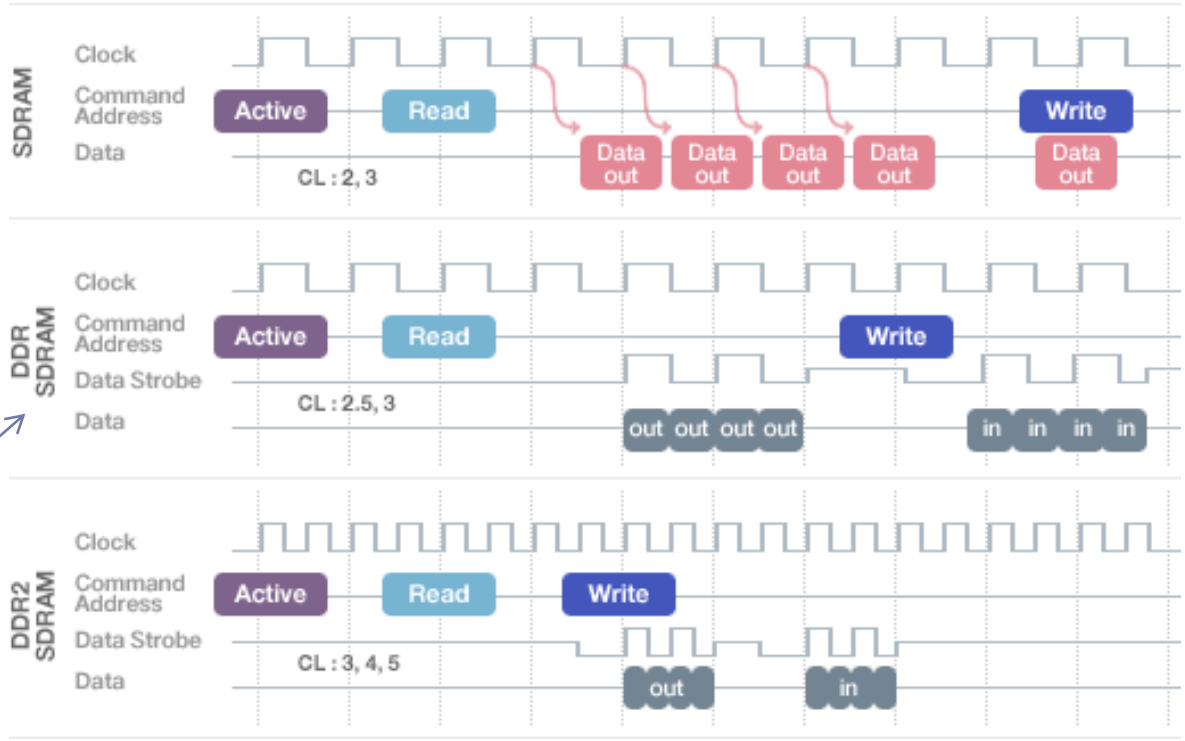
Figure 2.13 Times of fast and slow DRAMs vary with each generation. (Cycle time is defined on page 95.) Perfor-

Patterson y Hennesy

Tipos de memoria RAM

- DRAM
  - EDO
  - FPM
- SDRAM
  - DDR
  - DDR2

(double data rate)



SDRAM (Synchronous DRAM): sincronizadas con el reloj del sistema

Tipos de memoria DDR

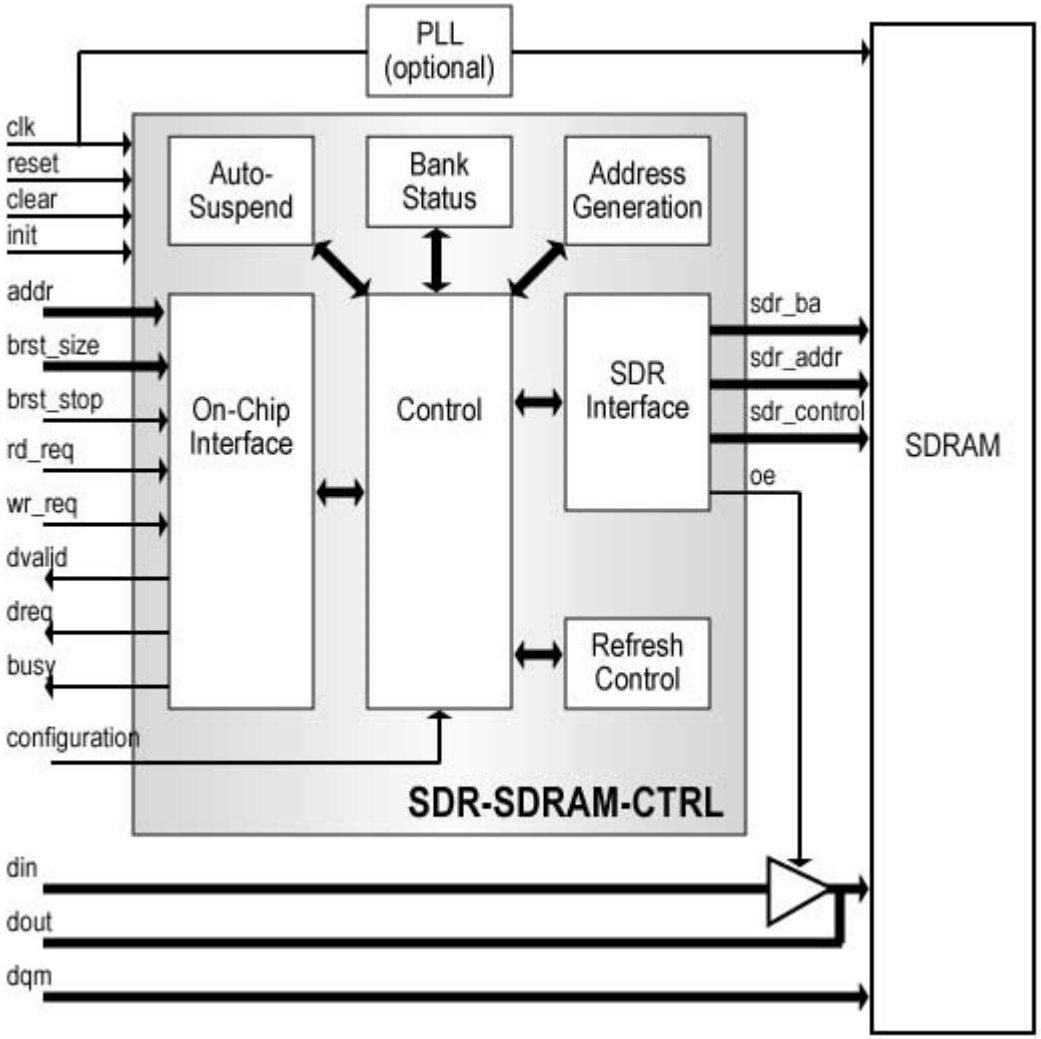
Standard	Clock rate (MHz)	M transfers per second	DRAM name	MB/sec /DIMM	DIMM name
DDR	133	266	DDR266	2128	PC2100
DDR	150	300	DDR300	2400	PC2400
DDR	200	400	DDR400	3200	PC3200
DDR2	266	533	DDR2-533	4264	PC4300
DDR2	333	667	DDR2-667	5336	PC5300
DDR2	400	800	DDR2-800	6400	PC6400
DDR3	533	1066	DDR3-1066	8528	PC8500
DDR3	666	1333	DDR3-1333	10,664	PC10700
DDR3	800	1600	DDR3-1600	12,800	PC12800
DDR4	1066–1600	2133–3200	DDR4-3200	17,056–25,600	PC25600

Figure 2.14 Clock rates, bandwidth, and names of DDR DRAMS and DIMMs in 2010. Note the numerical relation-

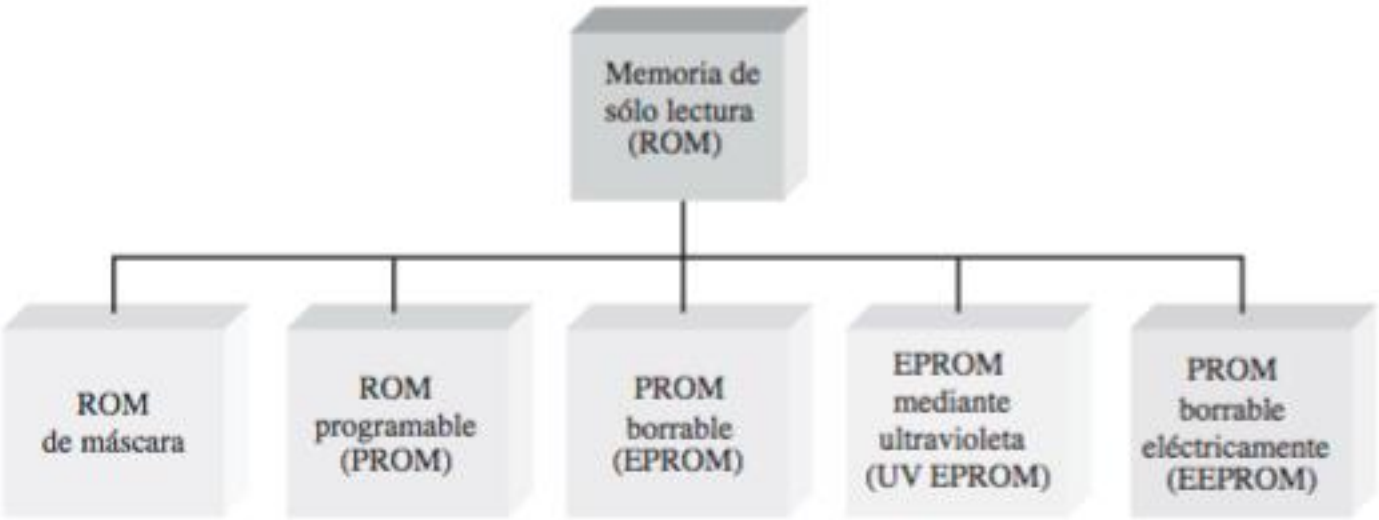
Patterson y Hennesy

Controlador de memoria DRAM

- Controlador se encarga del refresco y particularidades de la DRAM
- Oculta todo esto al procesador y le ofrece una interfaz simple
  - Procesador **no** dependiente de la tecnología de la memoria



Memorias ROM



Fundamenros de Sistemas Digitales
Thomas L. Floyd