Universidad Nacional de San Martín

Sistemas de Procesamiento de Datos

UNIDAD 6 = Memorias

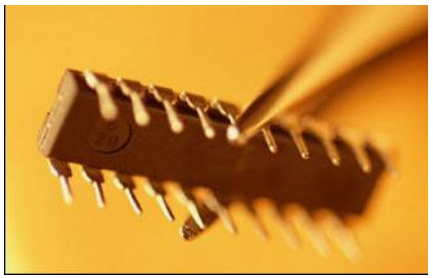
Tecnicatura en Programación Informática
Tecnicatura en Redes Informáticas

Profesor: Fabio Bruschetti
Ayudante: Pedro Iriso
2024 – 2C

Memorias

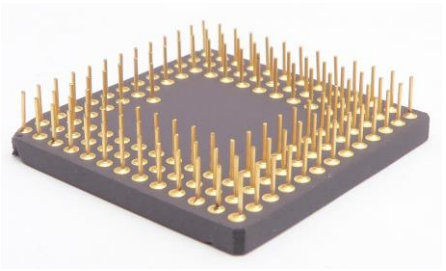
- Dispositivo utilizado para almacenar datos
- Características principales
  - Capacidad de almacenamiento
  - Velocidad de acceso (leer, escribir)
  - Permanencia de los datos (volátil, permanente)
  - Presentadas en circuitos integrados (IC's)

SMD: Surface-Mount Device



DIP: Dual In-Line Package

QFP: Quad Flat Package

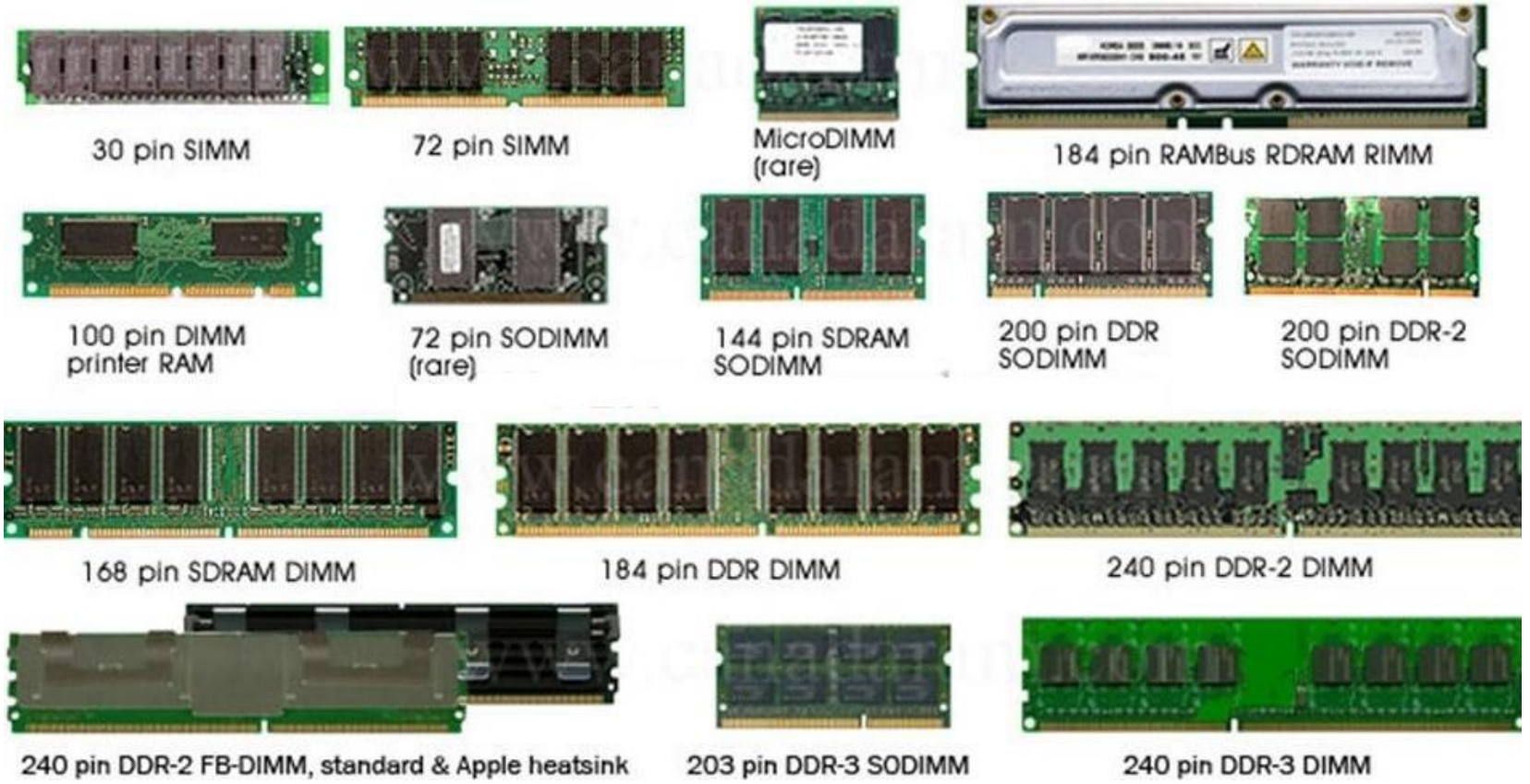


PGA: Pin Grid Array

Memorias

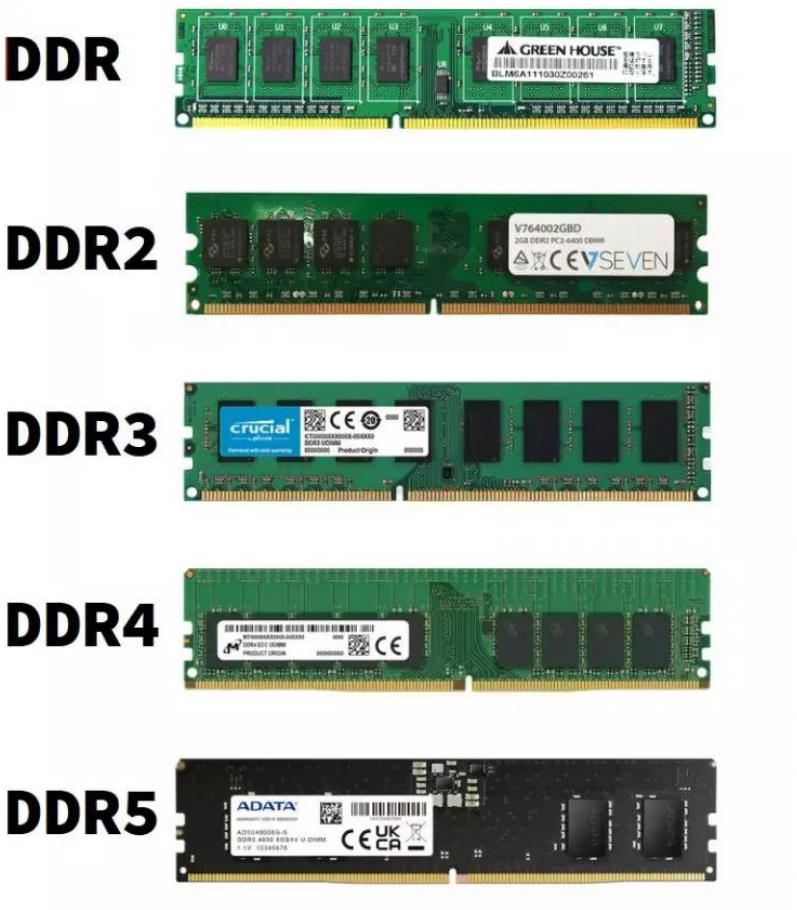
SIMM = Single In-Line Memory Module
DIMM = Dual In-Line Memory Module
RIMM = RAMBus In-Line Memory Module
SODIMM = Small Outline Dual In-Line Memory Module

- Presentadas en circuitos integrados (IC's o Chips)

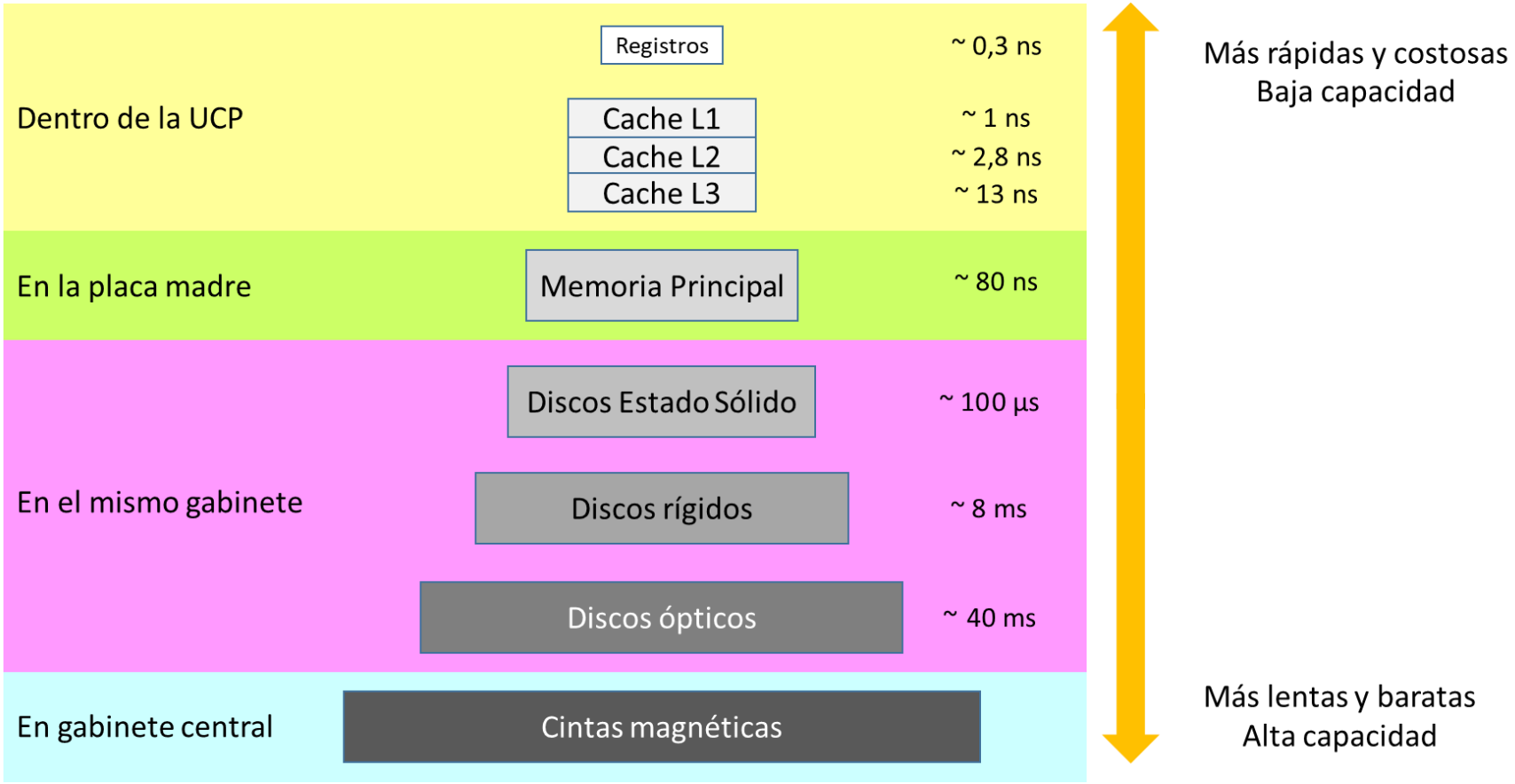


Memorias

- Presentación en bancos de memoria DDR

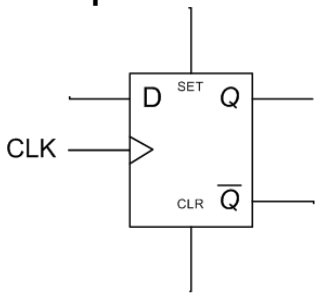


Jerarquía de memorias

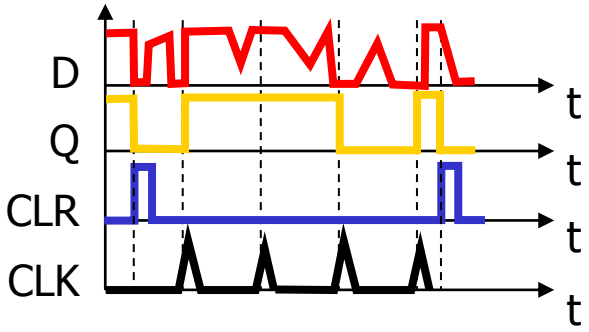


Registros: Flip-flops

- Son dispositivos de almacenamiento de *n* bits o celdas
- Cada celda de memoria almacena un solo bit y está compuesta por un "Flip-Flop" o biestable del tipo "D" que presentan el menor tiempo de acceso
- Un registro de *n* bits poseerá *n* flip-flops de este tipo
  - D = Entrada sincrónica
  - Q y Q = Salidas
  - CLK = Señal de clock
  - SET = Coloca Q en 1 en forma asincrónica
  - CLR = Coloca Q en 0 en forma asincrónica



- Para guardar un dato ("1" o "0") en esta celda de memoria, existen dos alternativas:
  - Asincrónicamente:
    - Guardar un "1" → SET = 1 y CLR = 0
    - Guardar un "0" → SET = 0 y CLR = 1
  - Sincrónicamente
    - Coloco en la entrada D el valor que quiero guardar
    - Coloco un pulso en la entrada CLK



Registros: lectura y escritura

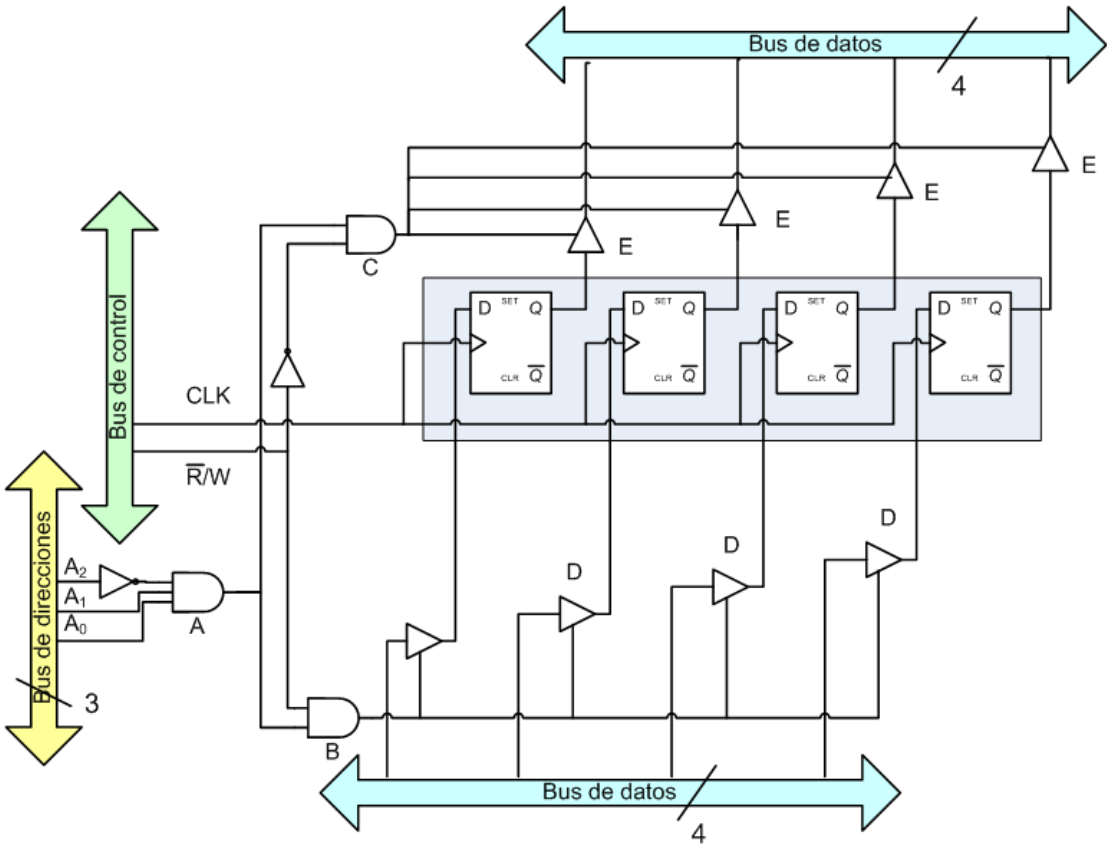
- **Lectura:**

1) se coloca la dirección "011_b" en el bus de direcciones. La salida de la compuerta "A" es "1"
2) La señal R/W se coloca en "0", con lo cual la salida de la compuerta "C" es "1" y habilita los buffers 3-state "E" disponiendo el dato en el bus
3) La compuerta "B" tiene un "0" en una de sus entradas lo que impide la escritura del registro

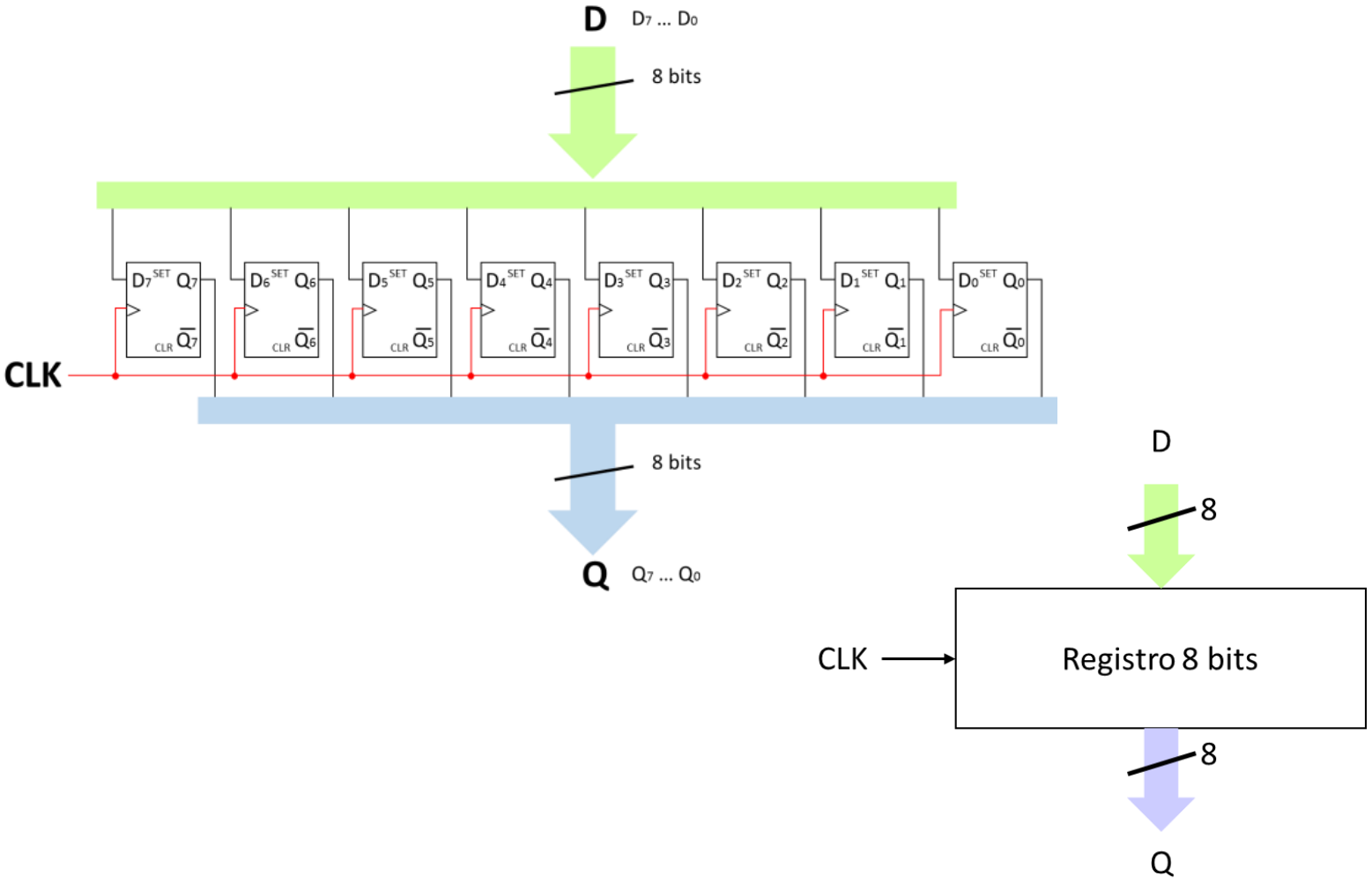
- **Escritura:**

1) se coloca la dirección "011_b"
2) Se presentan los datos a escribir en el bus de datos
3) La señal R/W se coloca en "1", con lo cual la salida de la compuerta "B" es "1" y habilita los buffers 3-state "D" disponiendo el dato en cada flip-flop
4) La compuerta "C" tiene un "0" en una de sus entradas lo que impide la lectura durante este ciclo

Conexión de un registro de 4 bits a los buses del computador

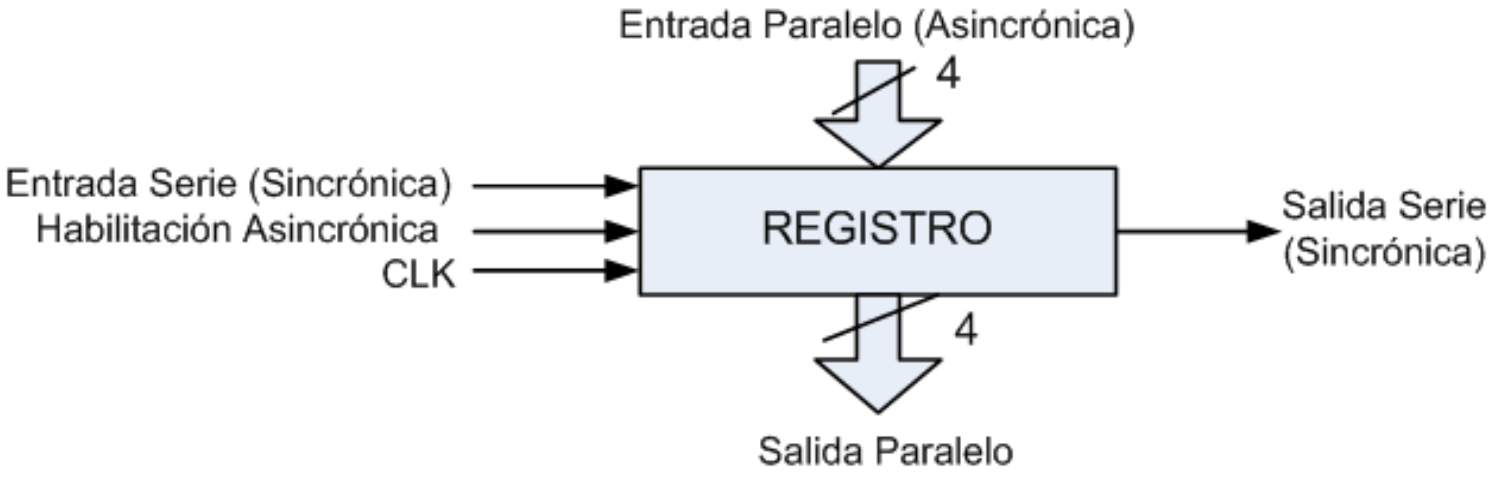


Registros de 8 bits

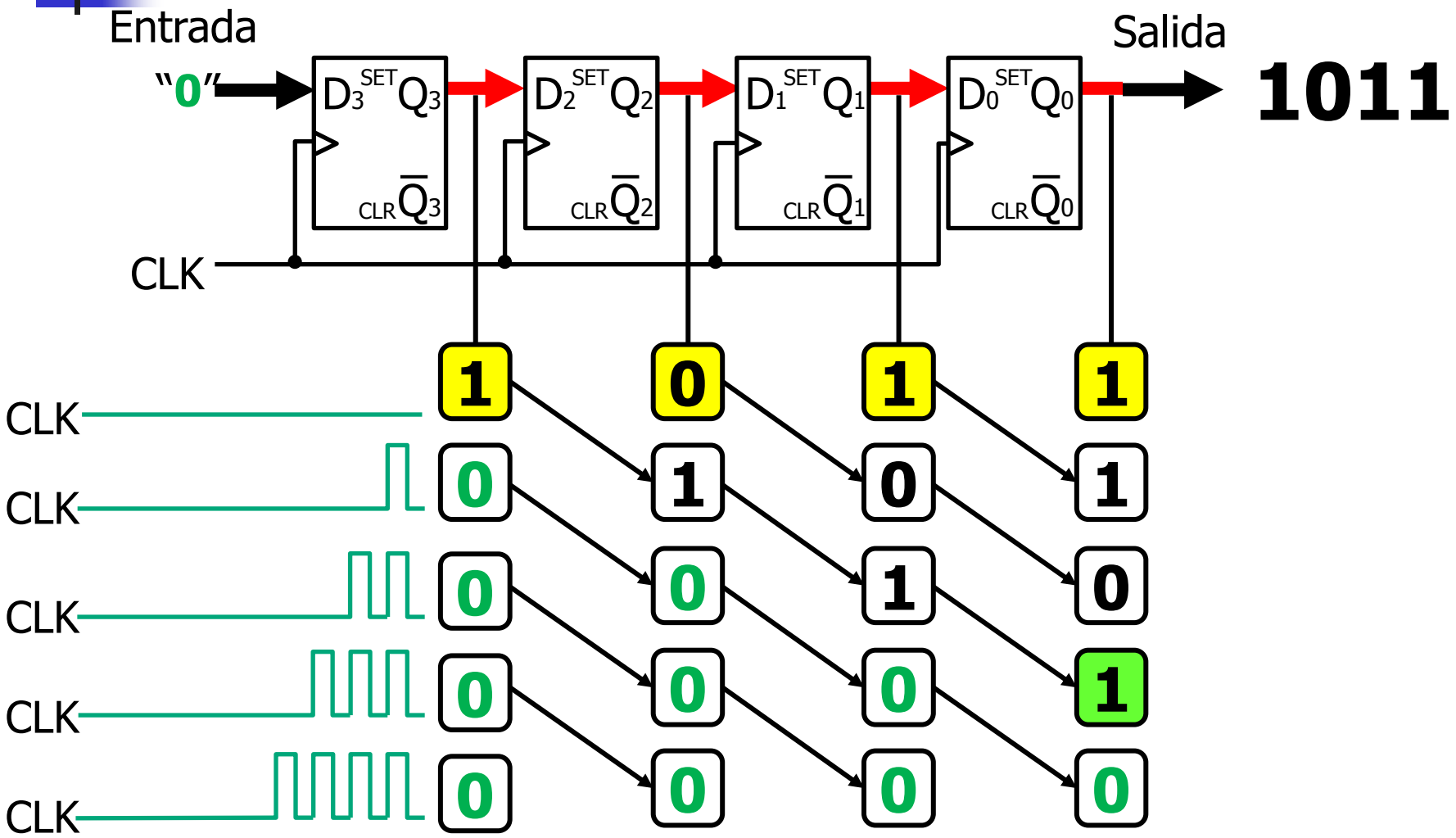


Registros de desplazamiento

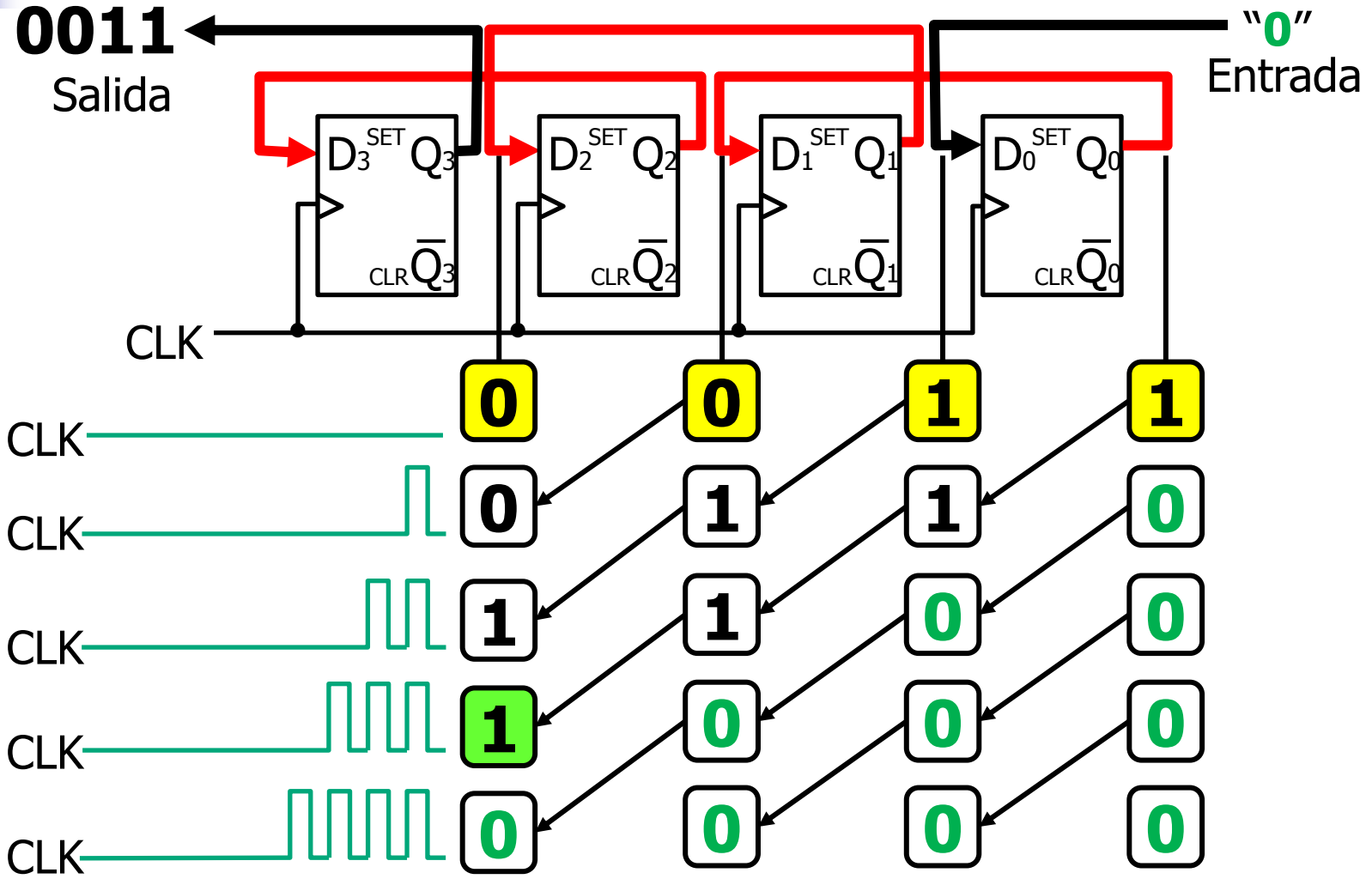
- Debidamente interconectados, los registros vistos pueden desplazar sus bits (alojados en cada uno de los Flip-Flops) hacia la derecha o hacia la izquierda
- Estos se encuentran en la ALU para hacer las operaciones producto y cociente
- El movimiento de los bits se realizará con cada pulso del CLK o reloj
- Se los puede cargar de forma asincrónica a través de SET y CLR o sincrónicamente a través de D y el CLK



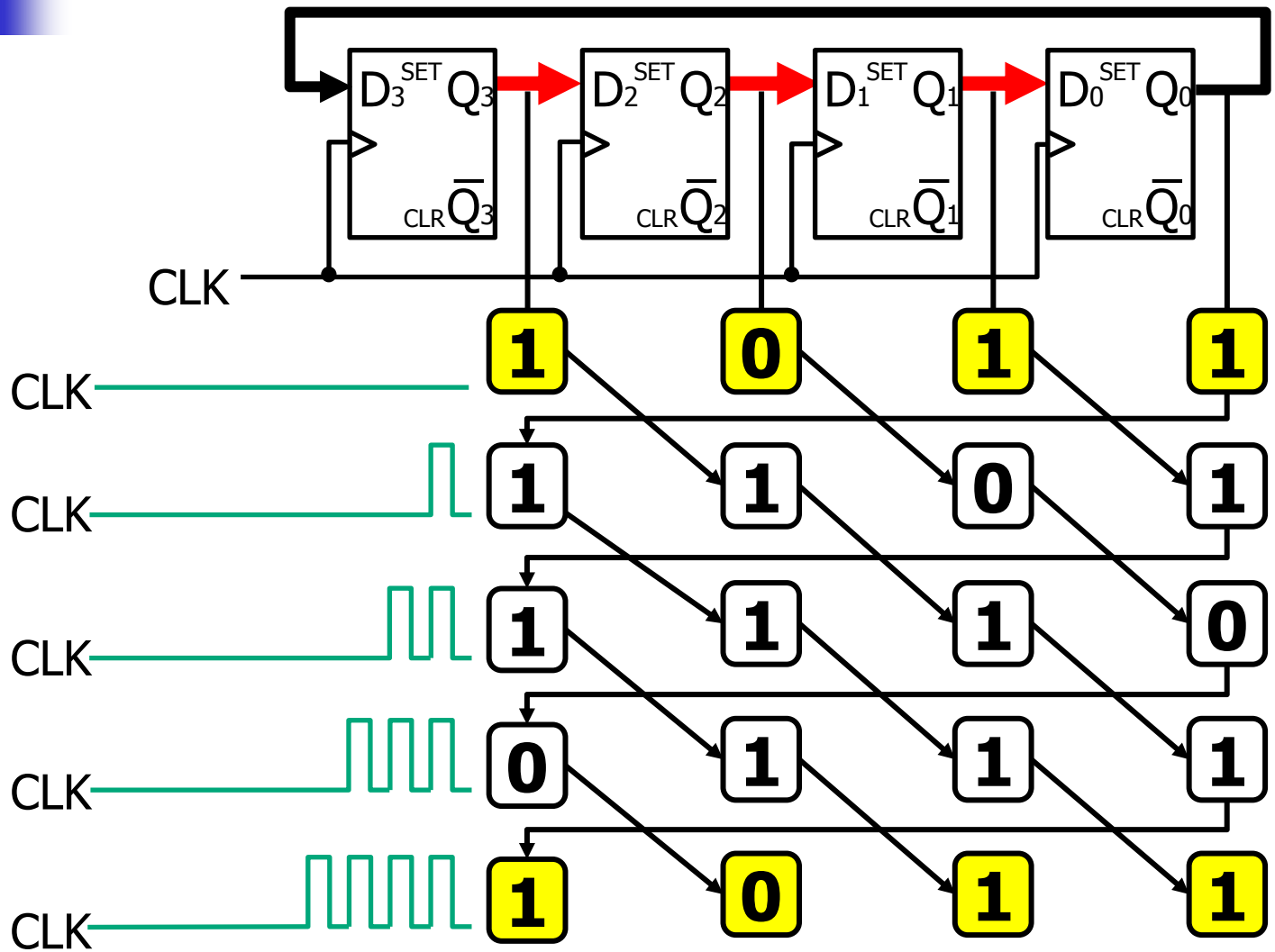
Registro de desplazamiento a derecha (sincrónicos)



Registro de desplazamiento a izquierda (sincrónicos)

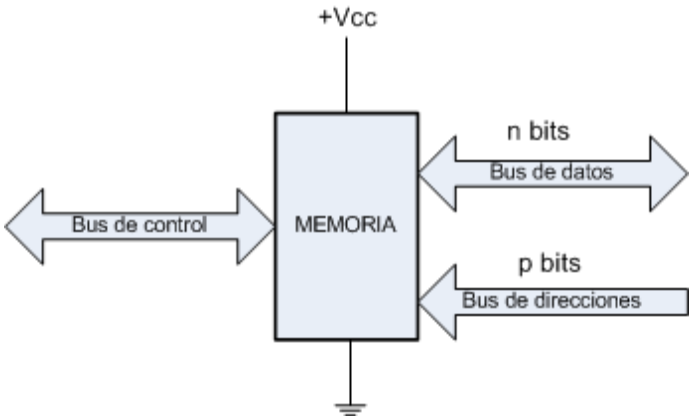


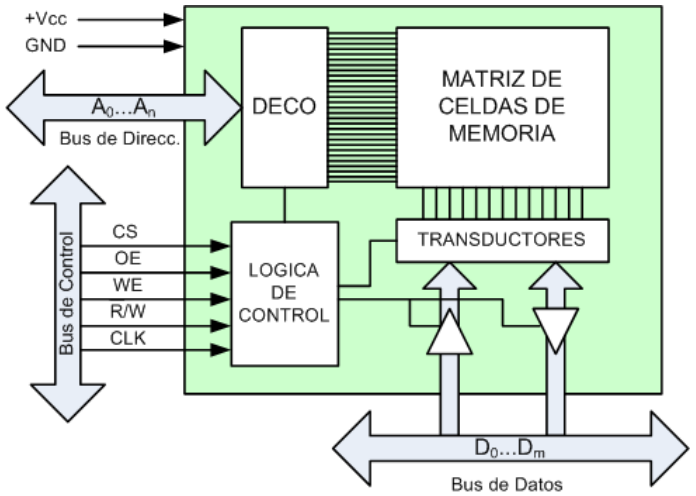
Registro de rotación a derecha (sincrónicos)



Memoria Principal

- Dispositivos rápidos conectados y controlados directamente por la CPU
- Suele estar organizada en base a palabras de *n* bits y se accede a cada palabra de
a una por vez y a través de una única dirección (de *p* bits)
- Los datos contenidos en una palabra son simplemente "0" y "1" y su interpretación dependerá de quien los lea o escriba
- El **Decodificador** selecciona a cuál de las 2^n palabras se quiere acceder
- La **Matriz de Celdas de Memoria** almacena los "0" y "1"
- La **Lógica de Control** gestiona las operaciones de lectura y escritura del dispositivo
- Los **Transductores** adaptan los niveles de señal necesarios entre el bus de datos y las celdas de la matriz de memoria





Ordenamiento

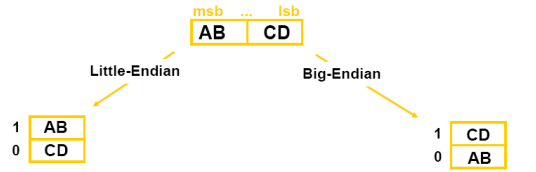
- Queremos guardar un dato de 16 bits contenido en un registro (como ser AX) en una memoria de longitud de palabra de 8 bits:

  9FCD = 1001111111001101

  AH AL

- Ordenamiento Little Endian (Intel)
  - Primero el LSByte (Less Significant Byte), luego el MSByte
  - Posición "n" → CD
  - Posición "n + 1" → 9F
- Ordenamiento Big Endian (Motorola)
  - Primero el MSBte, luego el LSByte
  - Posición "n" → 9F
  - Posición "n + 1" → CD

Ejemplo: El número Hexa de dos bytes $ABCD, a guardar en el lugar 0:

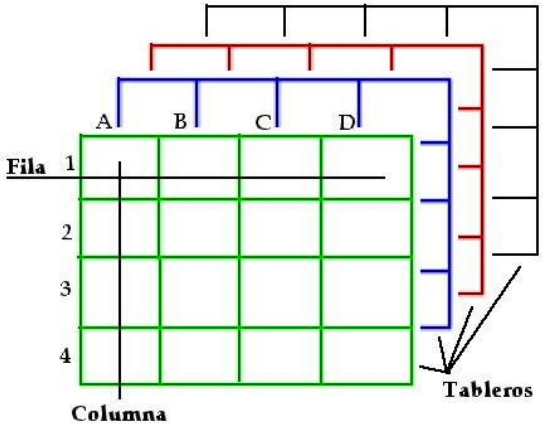


Latencias

- **Access Time** (Tiempo de Acceso)
  - Desde que se realiza un pedido de lectura hasta que el mismo queda satisfecho.
- **Cycle Time** (Tiempo de Ciclo)
  - Desde que se realiza un pedido hasta que se puede realizar el siguiente.
- **Write Time** (Tiempo de Escritura)
  - Al escribir un dato, tanto el dato y la dirección donde se escribirá deben estar presentes antes que llegue la señal de escritura (Setup Time)
  - Y deben mantenerse luego que se retire la señal de escritura (Hold Time)
- Los tiempos de acceso y de ciclo son muy importantes. Un CPU constantemente accede a la Memoria, y esto representa una traba a su velocidad.

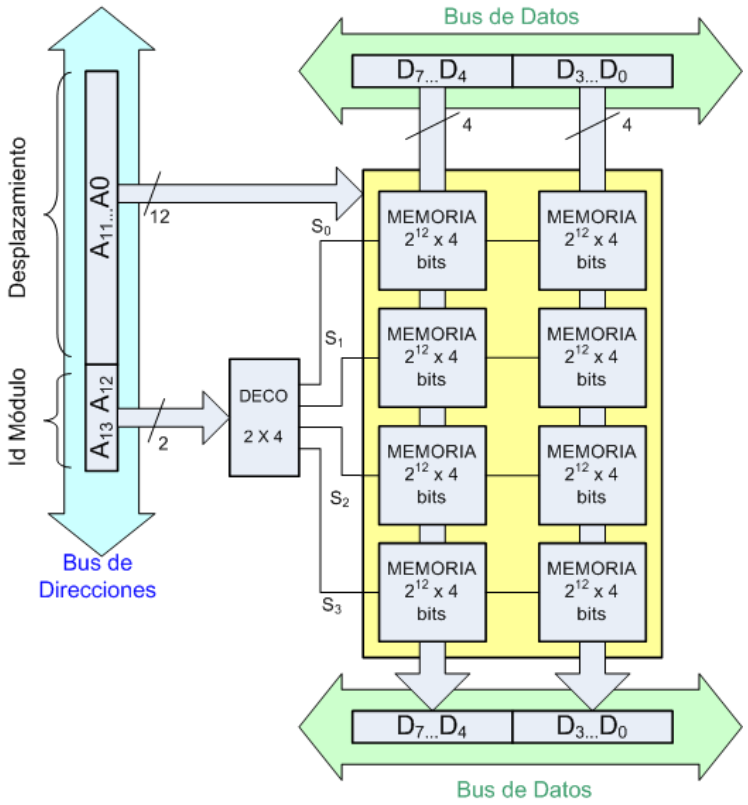
Latencias

- **CAS** (Column Access Strobe): indica el tiempo que tarda la memoria en colocarse sobre una columna.
- **RAS** (Row Access Strobe): indica el tiempo que tarda la memoria en colocarse sobre una fila.
- **ACTIVE**: indica el tiempo que tarda la memoria en activar un tablero.
- **PRECHARGE**: indica el tiempo que tarda la memoria en desactivar un tablero.



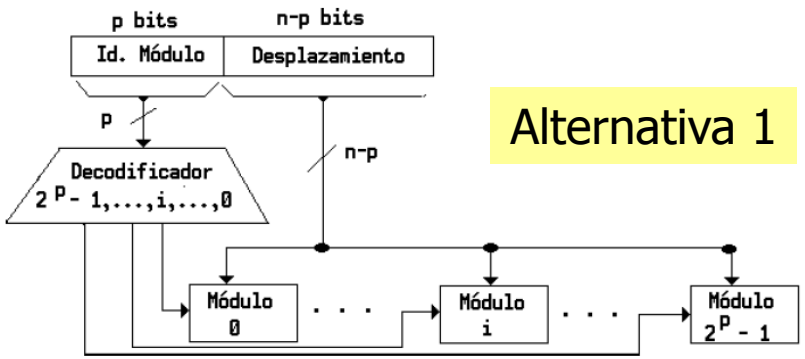
Bancos de memoria

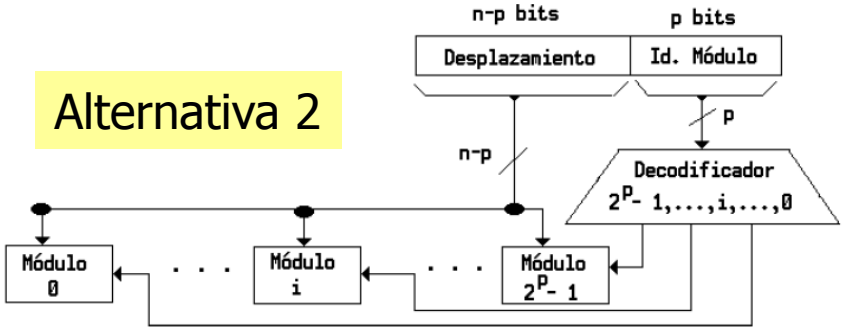
- La Memoria Principal se agrupa físicamente en varios módulos (Bancos)
- En un banco, la palabra direccionada podrá obtenerse:
  - Como la unión de varias partes de la palabras situadas cada una en módulos diferentes → encolumnar más de un módulo por dirección
  - En módulos distintos dependiendo de la dirección requerida → Más de una fila de módulos



Entrelazado de direcciones

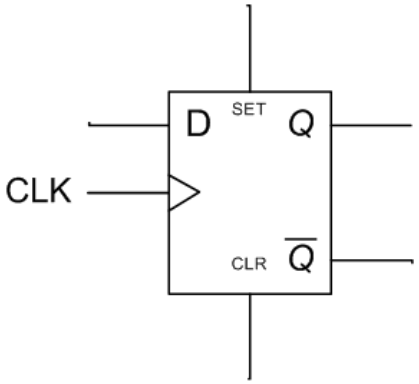
- Iterleave – Intercalado de Direcciones
  - 2^p módulos de $2^{(n-p)}$ direcciones de memoria
  - En cada módulo guardo $2^{(n-p)}$ direcciones de memoria
  - Alternativa 1: consecutivas
  - Alternativa 2: separadas 2^p





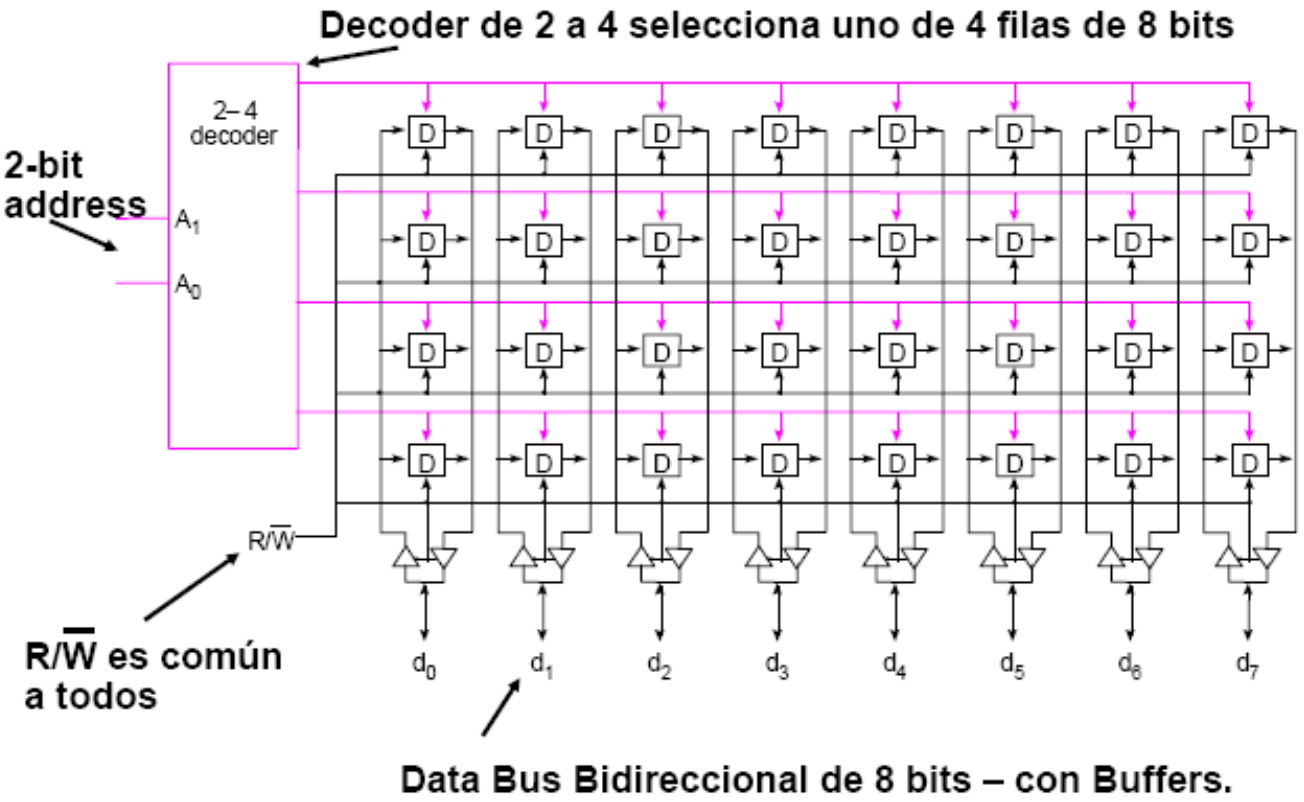
Memorias SRAM (estáticas)

- **Static RAM**: Memoria de lectura/escritura que mantiene sus datos siempre que tenga alimentación.
- Características
  - Son muy rápidas (devuelven el dato en 1 ns)
  - Muy caras
  - Consume mucha energía
- Usadas en dispositivos que requieren alta velocidad de operación
  - Memoria caché de microprocesadores
  - Procesamiento digital de imágenes

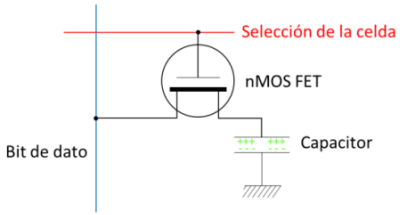


Memorias SRAM (estáticas)

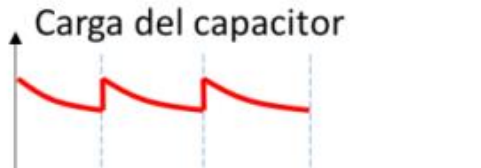
- Ejemplo Memoria 2D de 4 x 8

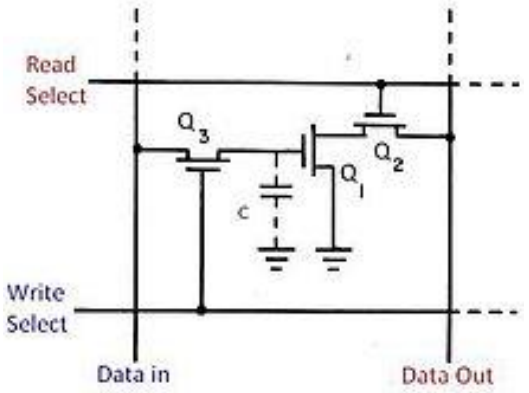


Memorias DRAM (dinámicas)



- **Dynamic RAM**: A diferencia de las SRAM, necesitan un pulso de energía periódico (refresco) para poder mantener los datos.
- Características
  - Más lentas que las SRAM (access time = 80 ns)
  - Más baratas que las SRAM
  - Consumo de energía mucho mas bajo
- Usadas como Memoria Principal en todo sistema con microprocesadores.
- Se convirtieron en el cuello de botella en un sistema moderno (PC).





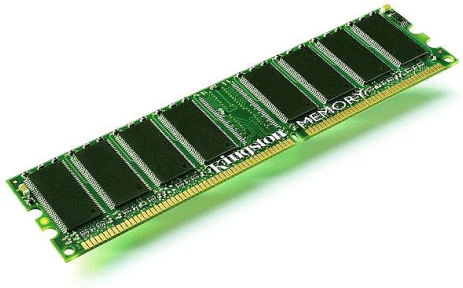
Memorias SDRAM (dinámicas)

- **Synchronous DRAM**: Permite el acceso a un bloque de datos que estén en fila, haciendo la transferencia sincronizada.

- Características
  - Permite transferencias en ráfagas (burst)
    - Se envía primero cantidad de datos a transferir
    - Luego la dirección donde se comenzará a almacenar
    - A partir de allí se transfieren varios bytes por ciclo.
  - Incorporan un circuito para hacer el refresco automáticamente haciendo que su velocidad de transferencia de datos sea más alta.
- Muy comunes en las PCs hasta hace un par de años.

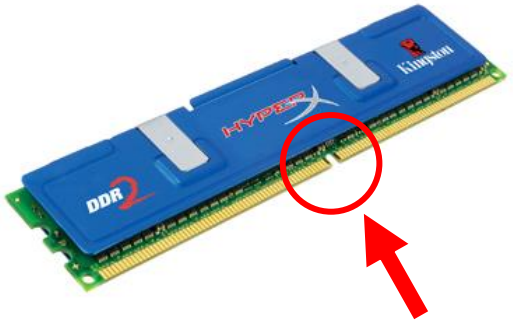
Memorias DDR (dinámicas)

- **Double Data Rate**: Fabricadas con tecnología SDRAM pero transfieren los datos en ambos flancos del reloj, duplicando la tasa de transferencia
- Trabajan con 2.5V en lugar de los 3.3V con que trabajan las SDRAM
- Adoptadas inicialmente por sistemas equipados con AMD mientras que Intel utilizaba RAMBUS
- Características
  - Poseen además técnicas avanzadas de optimización, que las hacen más veloces
    - Interleaving
    - Pipelining
  - El tiempo de acceso no es siempre igual, el primero dura más.
- Buffer interno de 2 bits (prefetch buffer)
- Velocidades del buffer desde 200 Mhz hasta 400 Mhz
- DIMM de 184 contactos

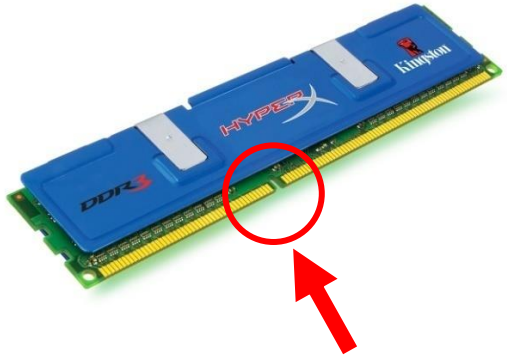


Memorias DDR2 y DDR3 (dinámicas)

- **Memorias DDR2**
  - Duplican la velocidad del bus respecto de las DDR (533 MHz a 1 GHz)
  - El buffer de prefetch es de 4 bits
  - Tensión de trabajo de 1.8V
  - DIMM de 240 contactos
  - Tienen menor latencia que las DDR

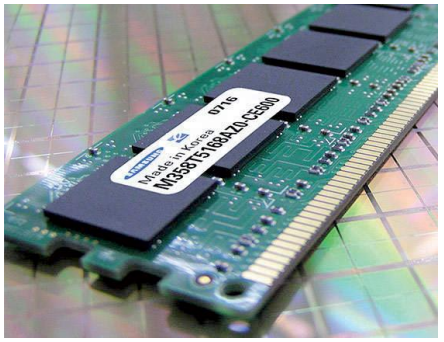


- **Memorias DDR3**
  - El buffer de prefetch es de 8 bits
  - Velocidad del bus de 800 Mhz a 2 Ghz
  - Tensión de trabajo de 1.5V
  - También el módulo DIMM es de 240 contactos pero se modifica la muesca
  - Tienen menor latencia que las DDR2!



Memoria DDR4 y DDR5 (dinámicas)

- **Memorias DDR4**
  - Duplican la velocidad del bus respecto de las DDR (2,13 GHz a 4,23 GHz)
  - El buffer de prefetch es de 4 bits
  - Tensión de trabajo de 1,2 a 1,05 V
  - DIMM de 288 contactos
  - No es compatible con las anteriores
  - Lanzada a finales de enero de 2014
  - Incremento del largo de la ráfaga de datos
  - Chequeo de paridad a nivel de bus de address y comandos



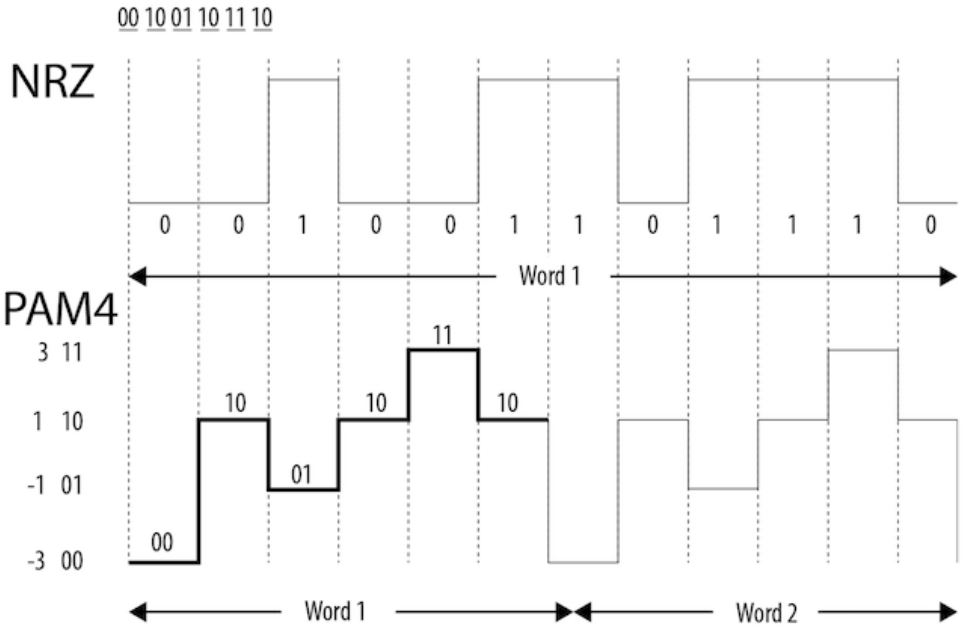
- **Memorias DDR5**
  - Velocidad del bus 5,2 GHz o más
  - Tensión de trabajo de 1,1 V
  - 128 GB o más en una placa
  - Duplicarán ancho de banda de las DDR4
  - Se comercializará cerca de julio 2020

Memorias gráficas GDDR (dinámicas)

- **Graphics-DDR**
- Utilizadas en tarjetas gráficas en conjunto con las GPU (Graphic-Processing-Unit)
- Las GPUs puede utilizarse tanto para tareas gráficas (como renderización) como para tareas de propósito general (HPC – High-Performance-Computing)
- Desde 2023 existen las GDDR6
  - Ancho de banda 2GB/seg
  - Densidad de hasta 16 Gb
  - Ancho de ráfaga de 16 bytes
  - 2 canales

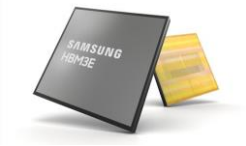
Memorias GDDR6X (dinámicas)

- **GDDR6** (usando NRZ) que es capaz de generar bits en forma de unos y ceros
- **GDDR6X** usa Pam4 que es capaz de 4 valores binarios para cada valor lo que resulta en el doble de ancho de banda. Vendrá en La placa Nvidia RTX 3090

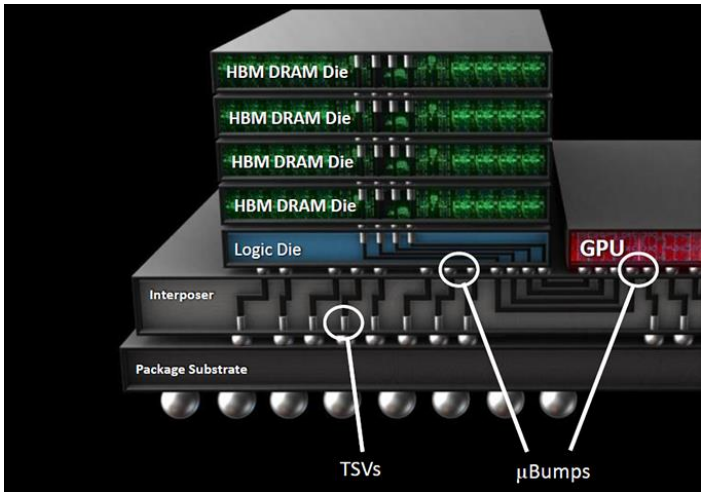


- Es mucho más eficiente energéticamente y también más barata
- PAM4: Es una técnica de modulación de señal de amplitud de pulso de 4 niveles que contine más información lógica en bits que las señales digitales tradicionales

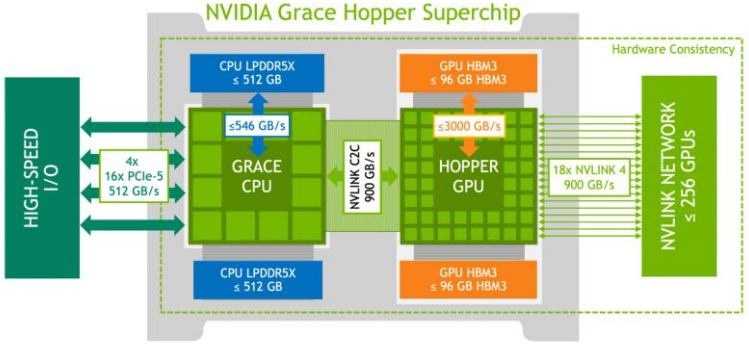
Memorias gráficas HBM (dinámicas)



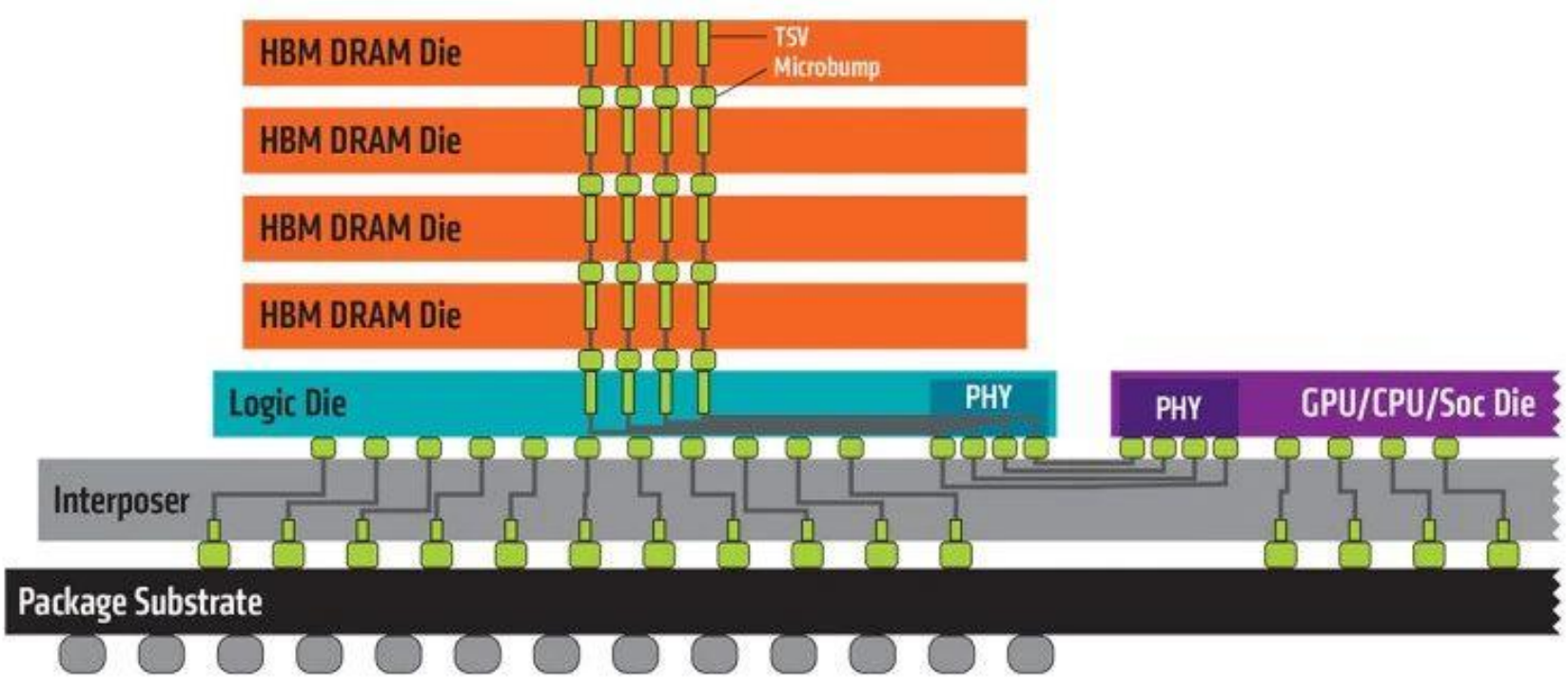
- **High-Bandwith Memory**
- Consisten en capas de memoria DRAM apiladas unas sobre otras
- Posee 4 conjuntos o pilas, cada una de 4 capas y cada capa (o Die) posee 2 canales



- Las conexiones se realizan con interconexiones TVS *Throug-Silicon Via* (conexiones a través del silicio) y *Microbumps* (micro conexiones con forma de bolilla) de 25 µm espaciadas 15 µm
- El estándar HBM3 logra un ancho de banda de 819 GB/s por pila
- GPU "Hoper" de NVidia prestará un BW de 3 TB/s

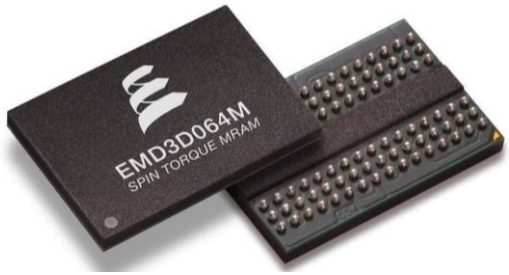


Memorias gráficas HBM (dinámicas)



Memorias especiales

- **XDR** (Extreme Data Rate)
  - Una versión mejorada de la memoria RAMBUS RDRAM (usada en Nintendo 64)
  - Permite un ancho de banda mucho más alto que las DDR2 u GDDR4
  - Ideal para placas de video o consolas de videojuegos (PlayStation 3)
- **MRAM** (Magnetoresistive RAM)
  - Usa discos ferromagnéticos separados por una capa aislante (en lugar de acumulación de cargas eléctricas) para guardar un bit
  - Ventajas: Menores tiempos de acceso y menor consumo de potencia
  - Desventajas: Menor densidad (180 nm) de integración
  - No volátil !!!! (Booteo instantáneo)
- **Otras Características**
  - ECC / NonECC
  - Buffered / Unbuffered



Clasificación: método de acceso

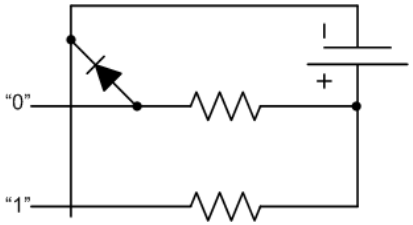
- **Aleatorias** (**R**andom **A**ccess **M**emory)
  - Permiten direccionar cualquier posición de la memoria de forma directa e independiente del lugar en donde se encuentre
- **Semi-aleatorias**
  - Discos floppy, discos duros, CD-ROM
- **Secuenciales**
  - Para acceder a una posición de memoria, debo recorrer la memoria desde el principio
  - Cintas magnéticas

Clasificación: volatilidad

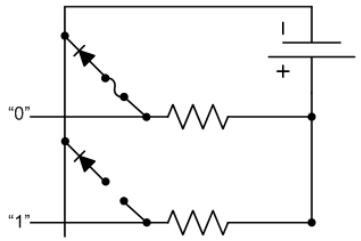
- **Volátiles**
  - Al desconectarlas del suministro eléctrico, se pierden los datos
  - Estáticas
    - Construidas con flip-flops
  - Dinámicas
    - Construidas con capacitores
- No volátiles
  - Los datos permanecen aún sin ser energizadas
    - ROM
    - PROM
    - EPROM
    - EEPROM
    - FLASH

Memorias ROM y PROM (solo lectura)

- ROM (**R**ead **O**nly **M**emory)
  - Se graban los datos durante el proceso de fabricación y no se puede alterar
  - Se coloca un diodo en donde se quieren guardar "ceros"

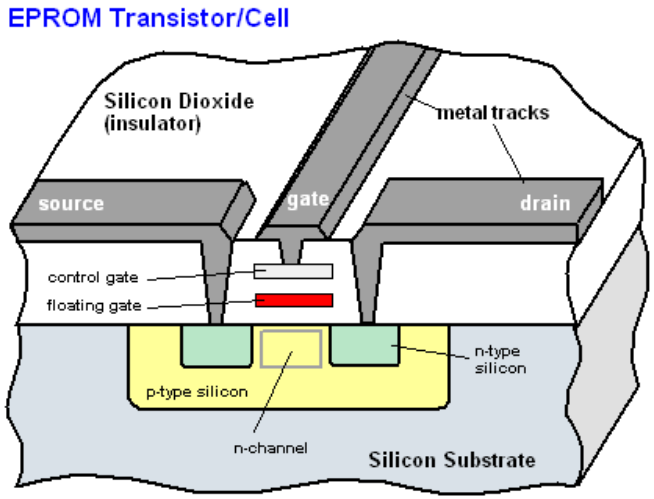


- PROM (**P**rogrammable **R**ead **O**nly **M**emory)
  - El fabricante coloca un diodo en cada celda de memoria, junto con un fusible en serie. De esta manera, en todas las celdas hay guardado un "cero"
  - En donde se necesita guardar un "uno", hay que quemar el fusible. Esto se realiza mediante un programa externo
  - Se programa una sola vez



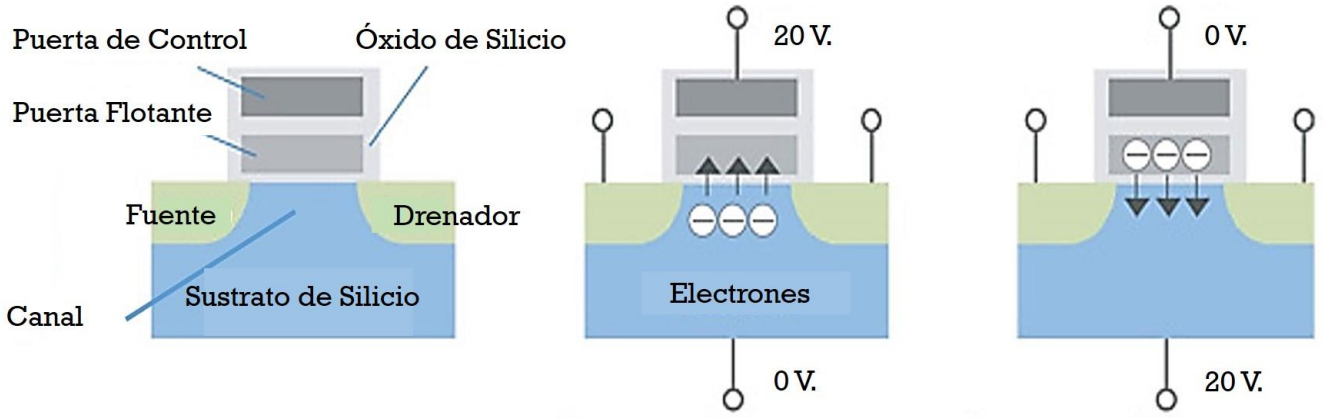
Memorias EPROM y EEPROM (Prioritariamente de lectura)

- EPROM (**E**lectrically **P**rogramable **R**ead **O**nly **M**emory)
  - Las programa el usuario en forma eléctrica
  - Se borra mediante luz ultravioleta



Memorias EPROM y EEPROM (Prioritariamente de lectura)

- EEPROM o E^{2}PROM (**E**lectrically **E**rasable **P**rogrammable **R**ead **O**nly **M**emory)
  - También las programa el usuario en forma eléctrica
  - Se borran de a celda por celda con una tensión eléctrica de polarización invertida

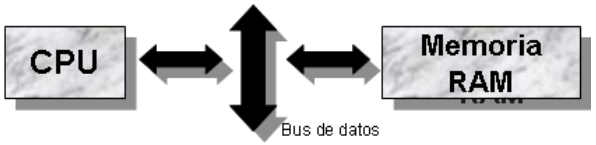


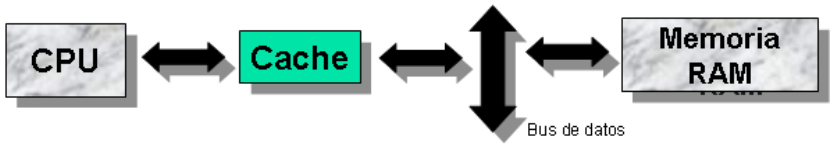
Memorias FLASH (lecto-escritura)

- Memoria del tipo EEPROM
- Se pueden escribir y borrar varias celdas simultáneamente
- Fabricadas con compuertas NOR y NAND para cada celda
- Barata, rápida (hasta 20 MB/s), de bajo consumo
- Durables: escribir y borrar su contenido una vez por día durante 27 años (Toshiba)

Memoria Cache

- Es una cantidad limitada de memoria de alta velocidad insertada entre el procesador y la memoria principal para mantener una porción del contenido de la memoria principal que está actualmente en uso
- El objetivo es incrementar la performance del sistema CPU ←→ Memoria

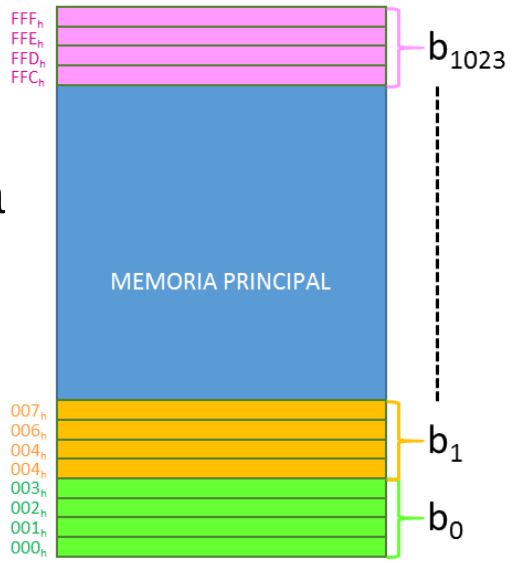




- Cuando se requiere acceder a una dirección de memoria, con altísima probabilidad se requerirá acceder a la siguiente → Se “cachean” bloques contiguos de memoria
- Su éxito puede ser atribuido a la propiedad de localidad de referencia
- Acierto (Hit): El procesador encuentra en la memoria Cache el dato que busca
- Falla (Miss): El procesador NO encuentra en la memoria Cache el dato que busca, debe buscarla en la memoria RAM común

Memoria Cache: Organización

- Indica las reglas para copiar datos de la memoria principal a la memoria Cache, como así también las reglas para alocar datos nuevos cuando la memoria Cache está llena
- Se asumirá:
  - La memoria común tiene una capacidad de 2^m bytes
  - Se la divide en bloques consecutivos de b palabras
  - El tamaño del bloque es una potencia de 2
  - Se tendrá la cantidad de 2^m / b bloques
- Existirán tres escenarios
  - Asociativa (Full Associative)
  - Mapeo Directo (Direct Mapping)
  - Asociativa Agrupada (Set Associative) solución de compromiso entre las dos anteriores



- 000_h – FFF_h
- 12 bits → 2^{12}
- 4096 direcciones

- B = 4
- 4096/4=1024 bloques

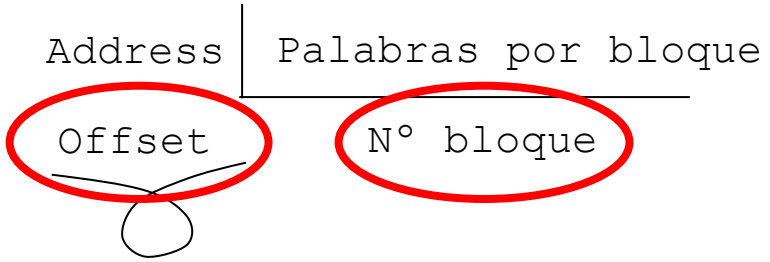
Memoria Cache: Organización **Asociativa**

- M cantidad de líneas o renglones
- Cada renglón tiene
  - B = Bit de validez
  - N = #bloque
  - V = Valores

Memoria Caché

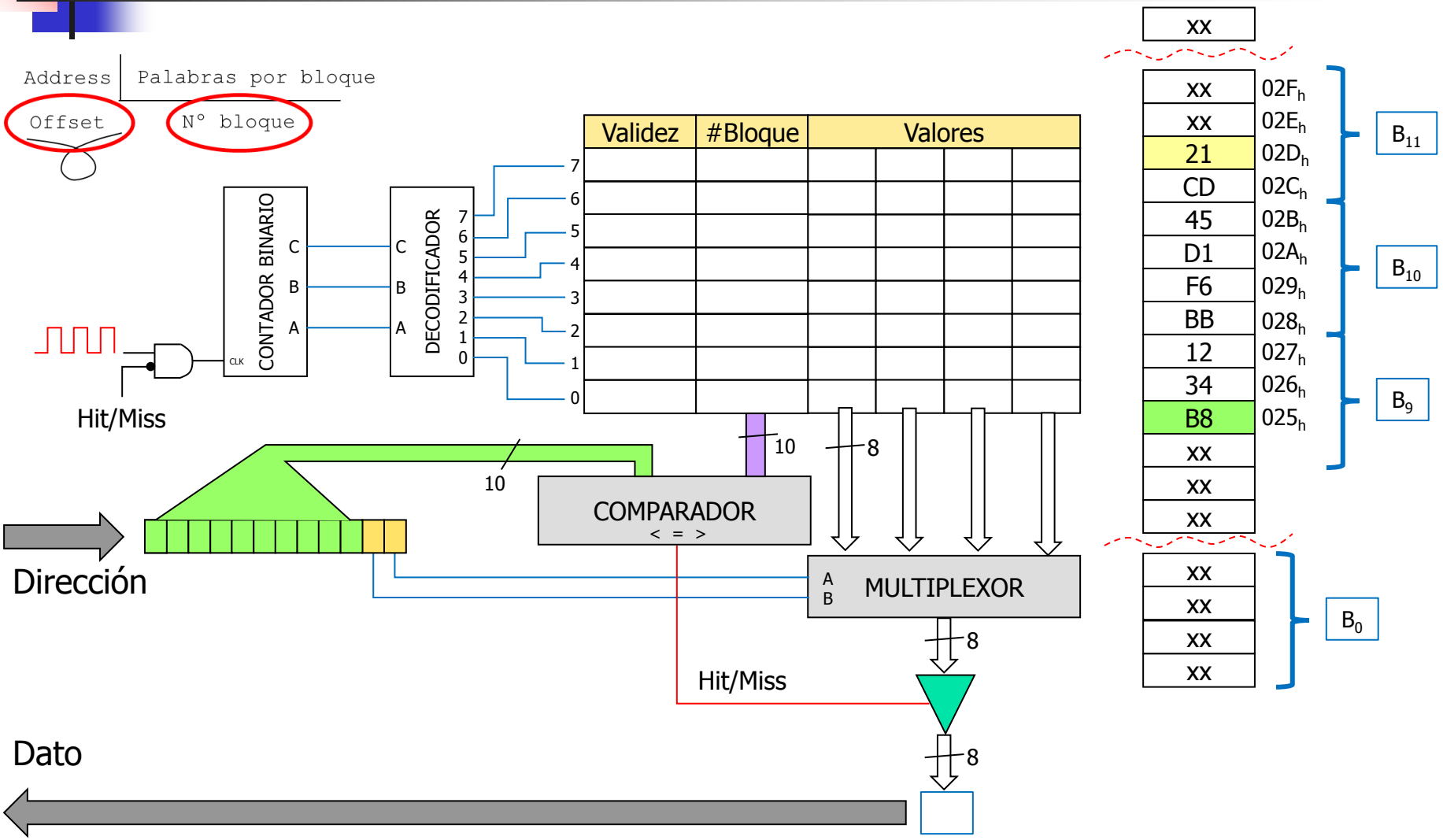
Renglón	B	N	V			
7						
6						
5						
4						
3						
2						
1						
0						

- Los datos de cualquier dirección de memoria común pueden ser almacenados en cualquier dirección de la memoria Cache
- El microcódigo debe
  - calcular el número de bloque
  - buscar ese número de bloque en la Cache
    - Circuitería adicional
    - Mas costosa

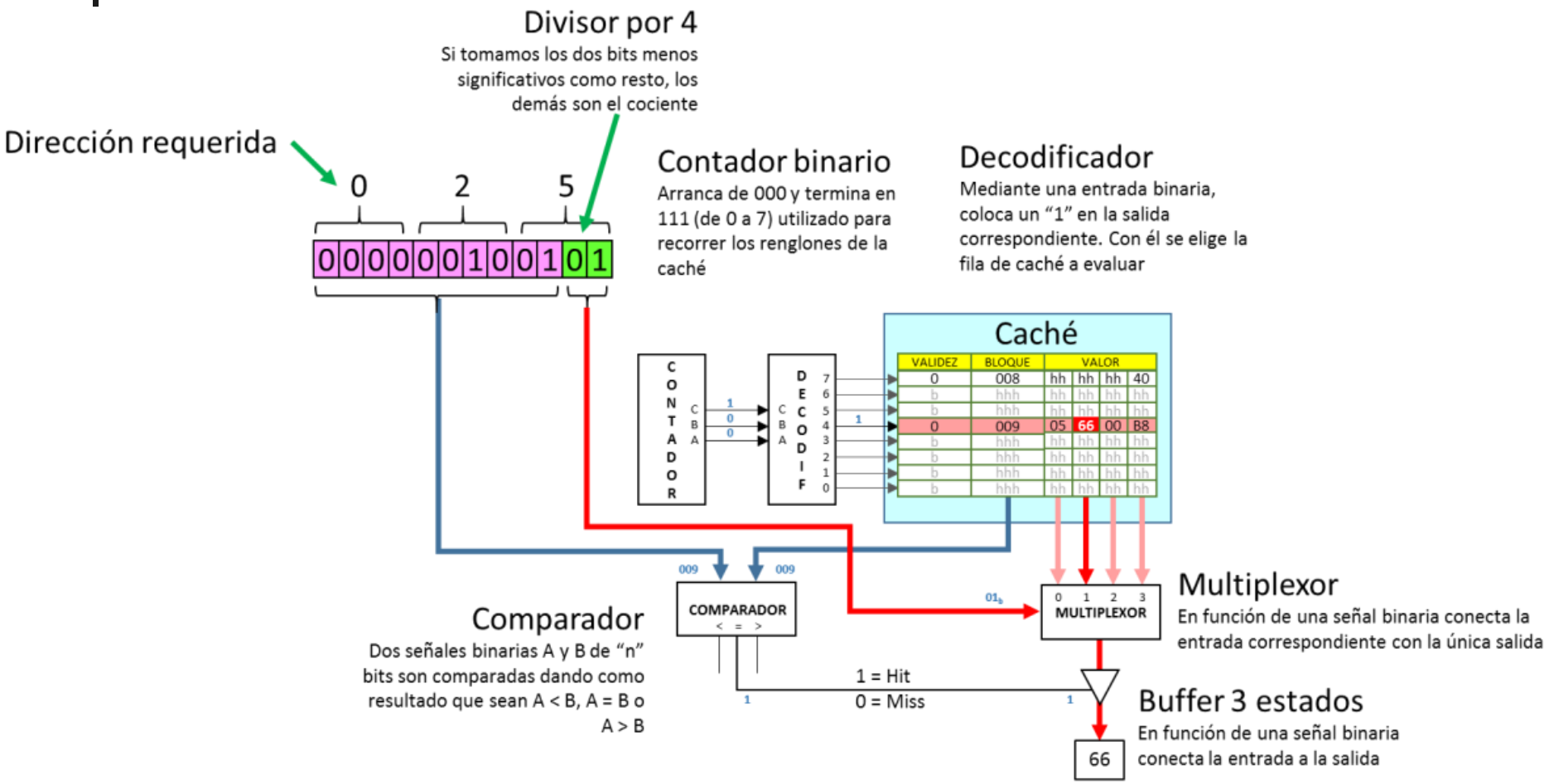


- Se necesitan sofisticados algoritmos de búsqueda
- Ofrece la mejor proporción teórica de aciertos

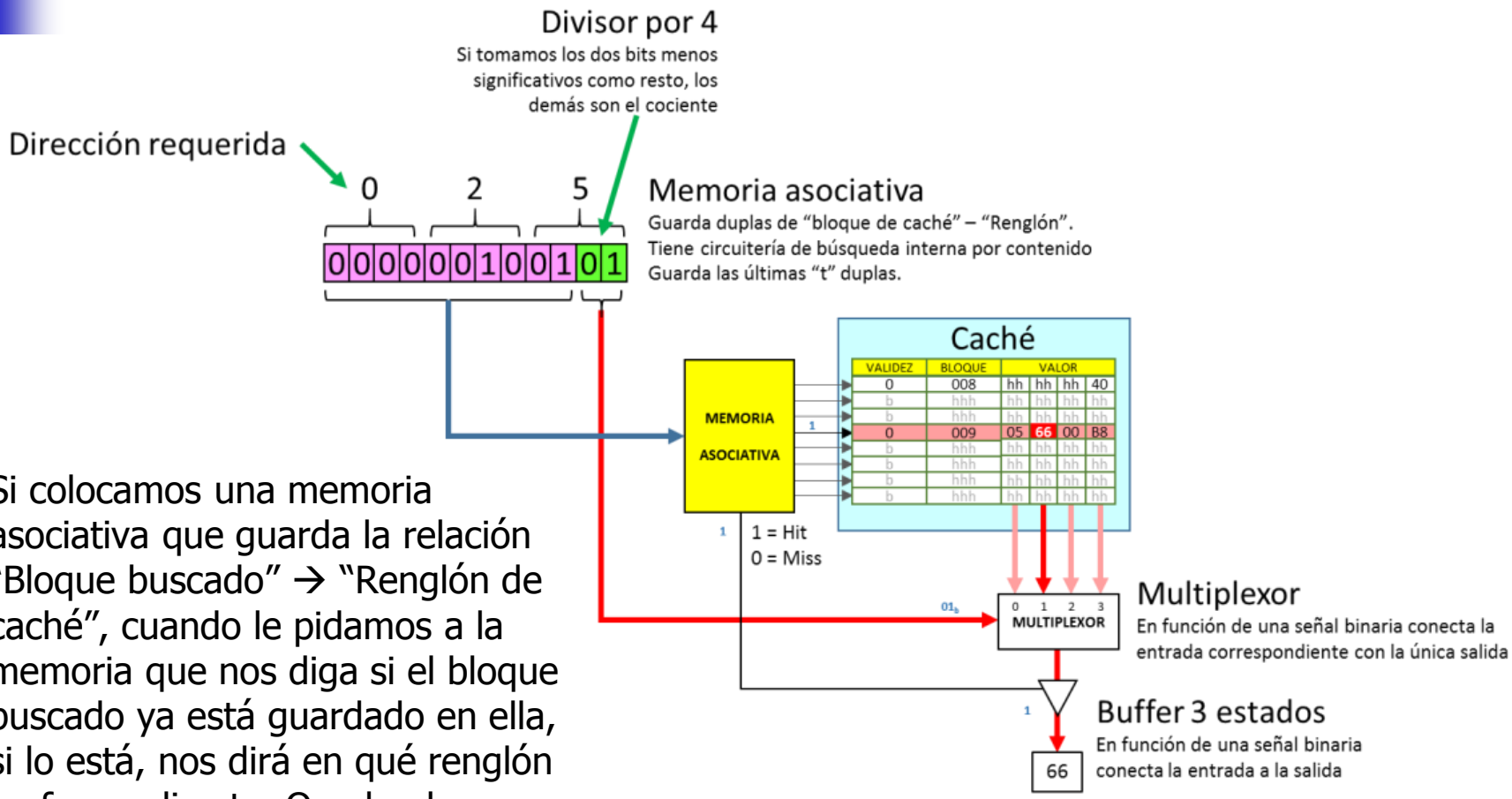
Memoria Cache: Organización **Asociativa**



Memoria Cache: Organización **Asociativa**



Memoria Cache: Organización **Asociativa**



- Si colocamos una memoria asociativa que guarda la relación "Bloque buscado" → "Renglón de caché", cuando le pidamos a la memoria que nos diga si el bloque buscado ya está guardado en ella, si lo está, nos dirá en qué renglón en forma directa. Quedando un circuito más reducido y eficiente

Memoria Cache: Organización **Mapeo Directo**

- M cantidad de renglónes
- b = palabras por bloque
- Cada renglón tiene
  - B = Bit de validez
  - E = Etiqueta
  - V = Valores

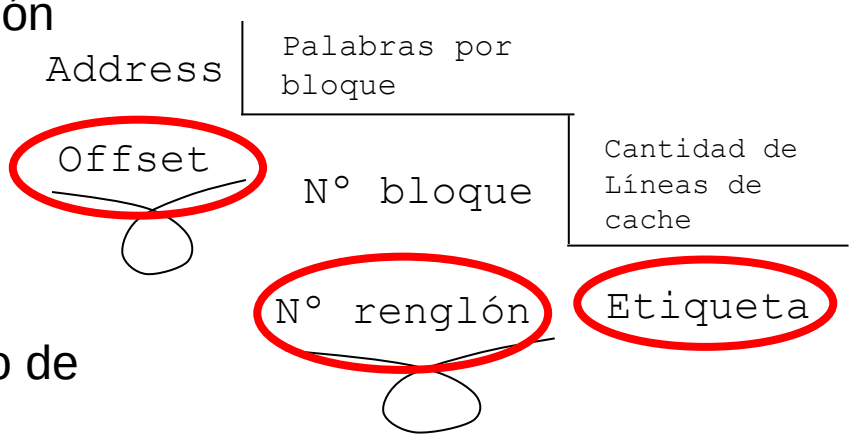
Memoria Caché

Renglón	B	E	V			
7						
6						
5						
4						
3						
2						
1						
0						

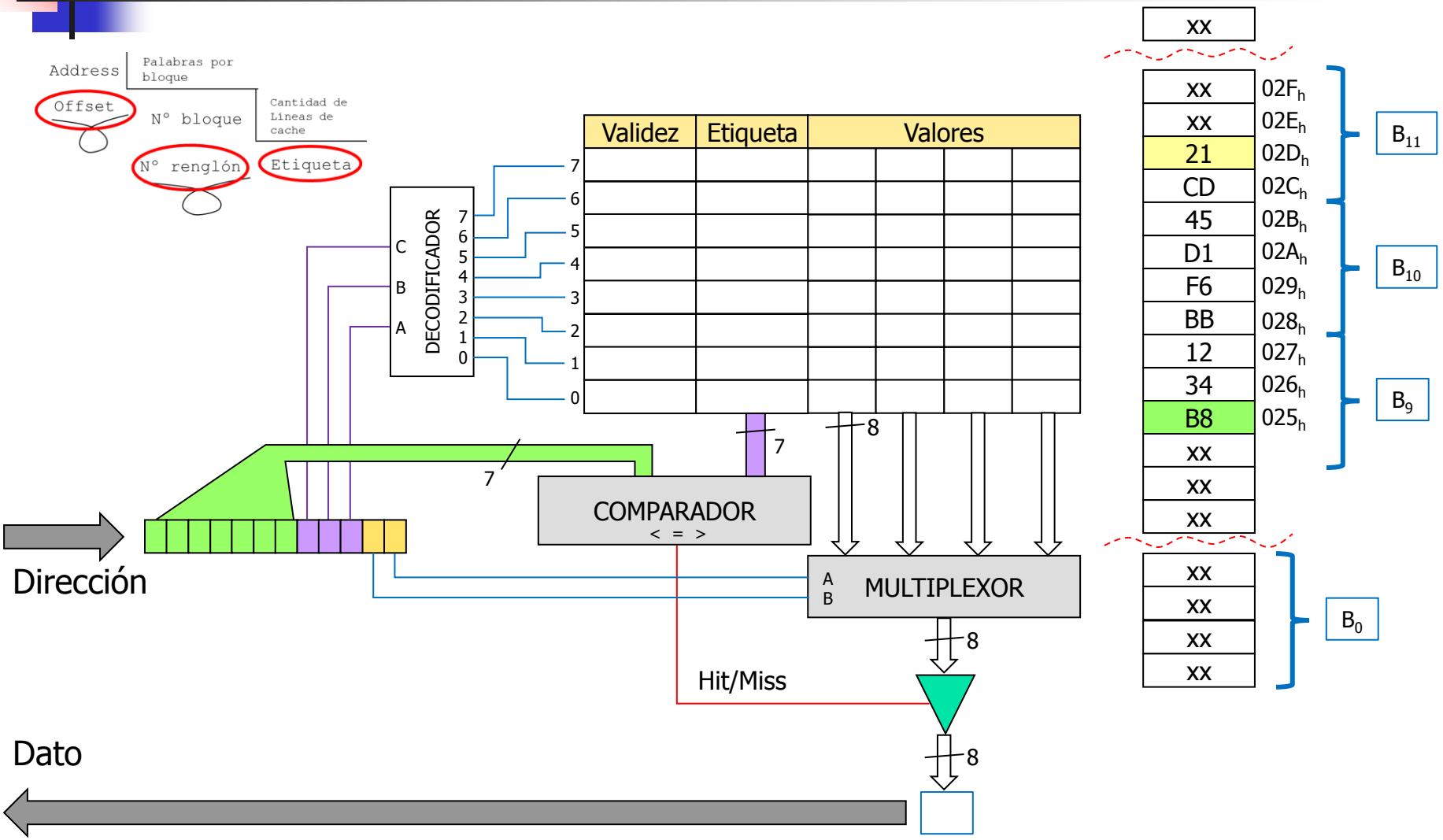
Nros. De bloque

7	15	23	31
6	14	22	30
5	13	21	29
4	12	20	28
3	11	19	27
2	10	18	26
1	9	17	25
0	8	16	24
Etiqueta →	0	1	2	3	.....

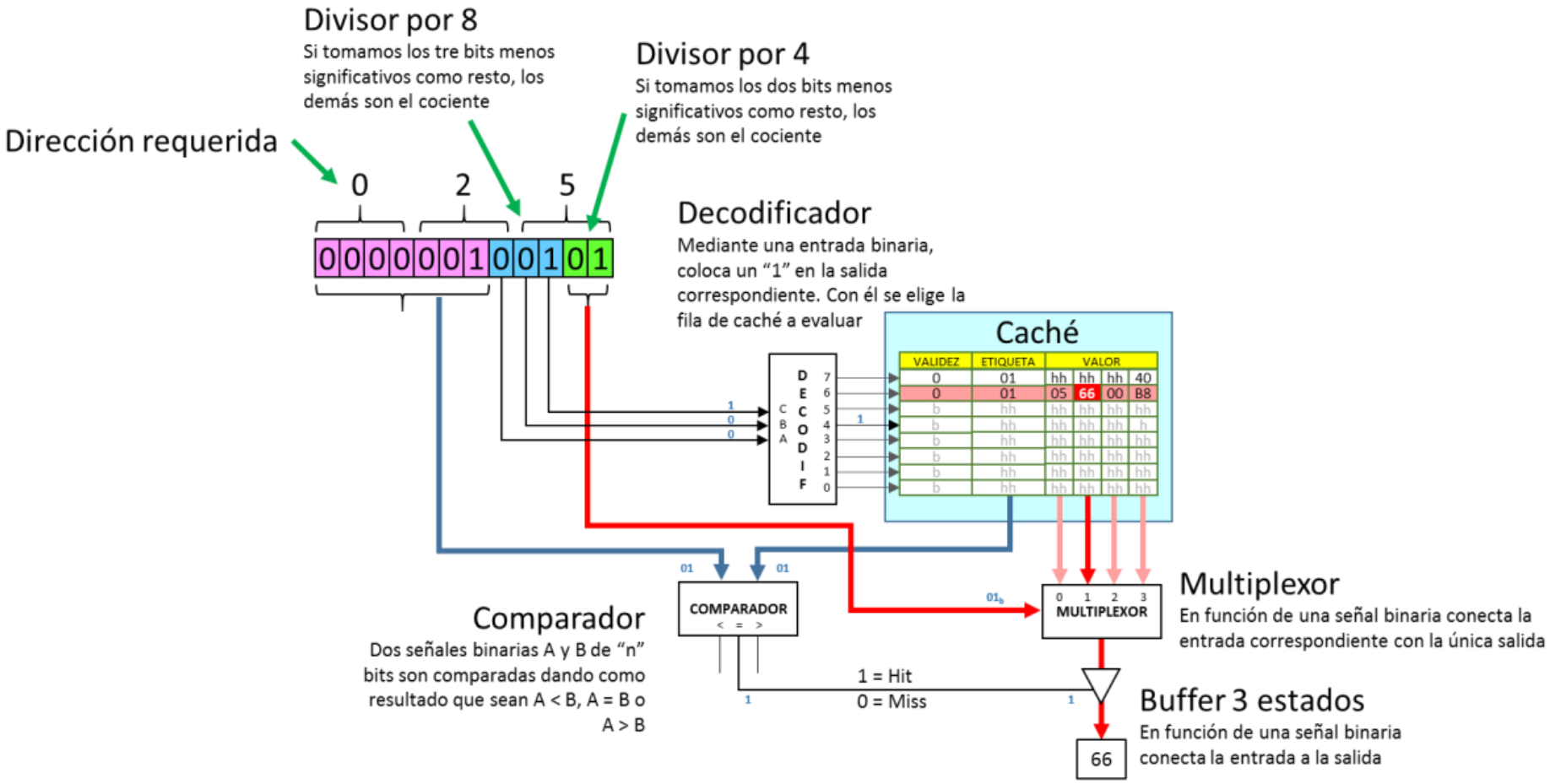
- Cada bloque tiene asignado un único renglón
  - Nro. de bloque = dirección / b
  - Etiqueta = #bloque / Cant. reng
  - Nro. de renglón = #bloque *mod* Cant. reng
- No hace falta buscar
- La etiqueta resuelve el problema de mapeo de direcciones con igual número de renglón
- Son más rápidas y sencillas



Memoria Cache: Organización **Mapeo Directo**



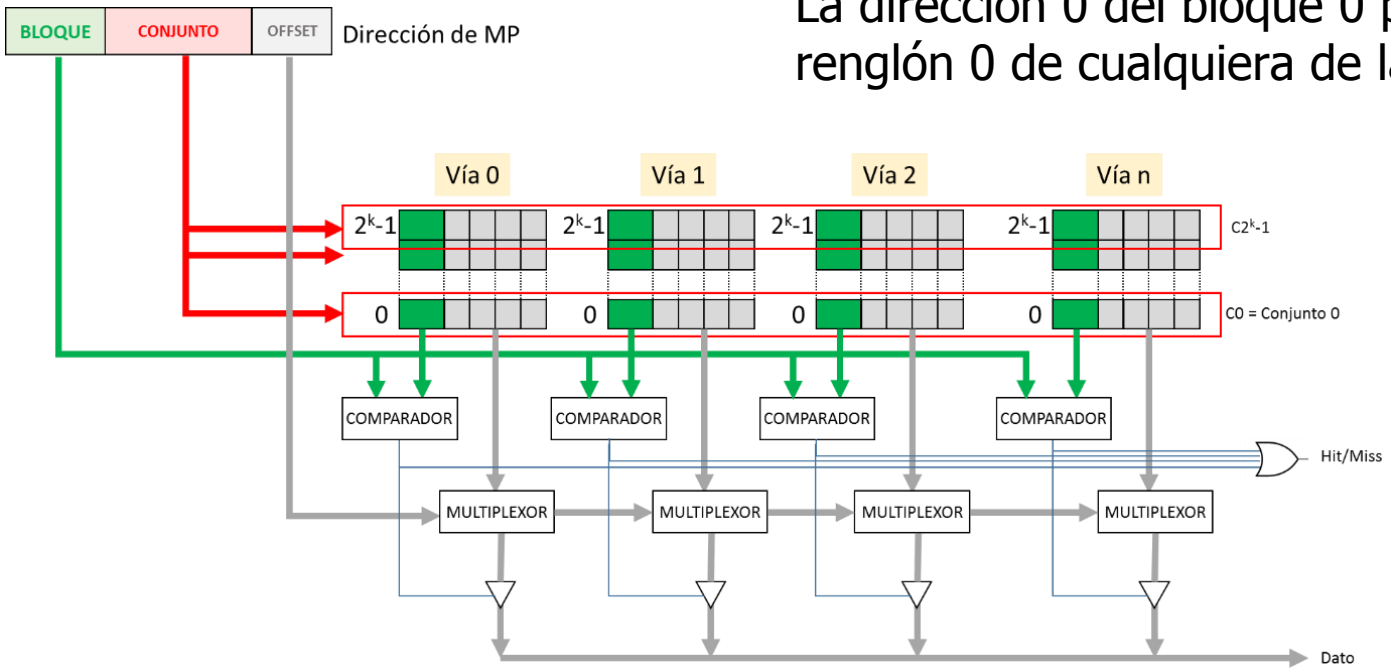
Memoria Cache: Organización **Mapeo Directo**



Memoria Cache: Organización **Asociativa por Conjuntos**

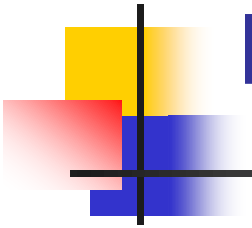
- Es una solución de compromiso entre ambas estrategias. Toma lo mejor de ambas para dar una mejor proporción de aciertos. Básicamente consiste en definir tamaño de bloques de MP de igual cantidad de palabras que cada paquete de direcciones o vía de memoria caché

La dirección 0 del bloque 0 puede ir en el renglón 0 de cualquiera de las vías



Memoria Cache: Políticas de escritura a Memoria Principal

- Determinan como se administrará las escritura de la memoria RAM común con los datos contenidos en la memoria Cache.
- **Escritura a Memoria** (Write-Through Cache)
  - Cuando se modifica la Cache, se modifica el mismo dato en la memoria principal
  - No tiene problemas de consistencia
  - Beneficioso para procesos de lectura intensiva
- **Retrograbado** (Write-Back Cache)
  - Se modifica solamente la Cache
  - Se modifica el dato en la memoria principal común cuando se necesita desocupar un bloque de la Cache que tiene datos modificados
  - Mas performante que el anterior
  - Beneficioso para procesos de escritura intensiva
  - Es necesario usar 1 bit más para indicar si el renglón de la Cache fue modificado o no – Dirty Bit
- ¿Qué bloque saco para que entre uno nuevo?
  - LRU (**L**east **R**ecently **U**sed) – RANDOM – FIFO (**F**irst-**I**n-**F**irst-**O**ut)



Memoria virtual

- Contar con un espacio de memoria mucho más grande que la capacidad de nuestra memoria principal
- Ese espacio está en las unidades de almacenamiento secundario: HDD, SSD
- Mecanismo: permitirle al procesador trabajar con direcciones de memoria (virtuales) mayores que las existentes (reales) y cada vez que haya que accederlas, traducirlas
- Como en memoria caché, una parte de esa memoria virtual estará copiada en la memoria principal (y una parte de ésta, en la memoria caché)

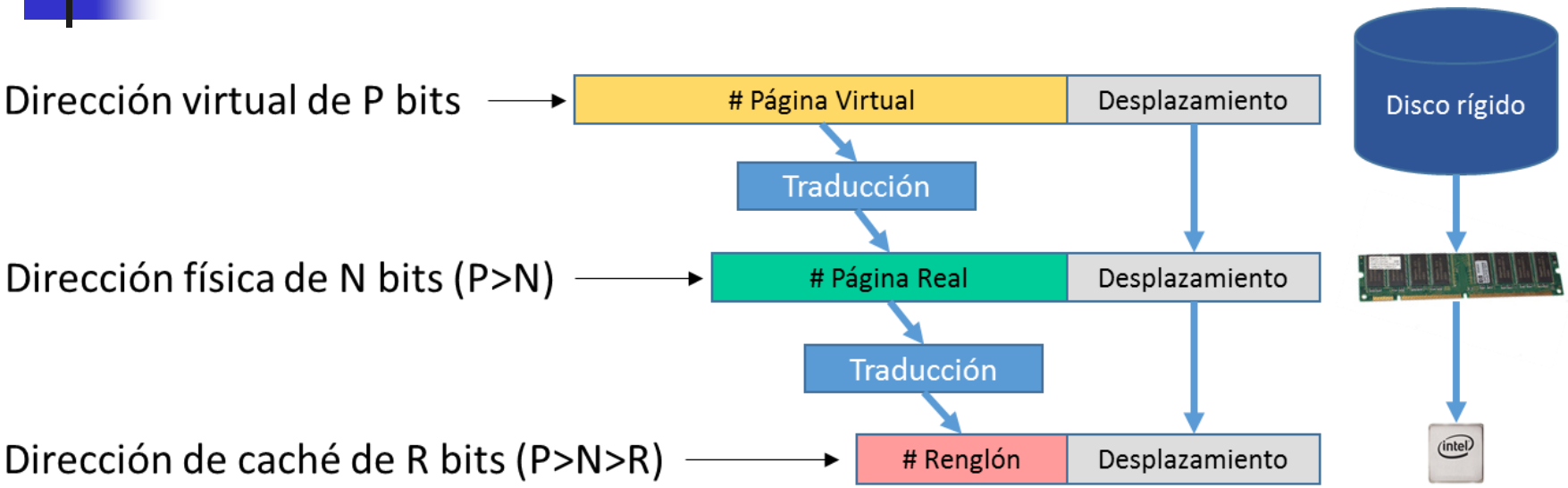
Memoria virtual: Espacios

- Espacio Virtual
  - Definido en la memoria secundaria (discos externos, cintas, etc.)
  - La capacidad de direccionamiento en la memoria virtual supera las posibilidades definidas por la cantidad de bits del bus de direcciones (Address Bus)
  - Su capacidad está restringida a:
    - La capacidad de la memoria secundaria
    - La posibilidad de direccionamiento virtual de la CPU o del Sistema Operativo
- Espacio real
  - Definido en la memoria principal (MP)
  - La capacidad de direccionamiento está dada por la cantidad de bits del bus de direcciones
  - Su capacidad está restringida a la cantidad de memoria RAM física disponible

Memoria virtual: Segmentación y paginación

- EL S.O. asigna un conjunto de posiciones de memoria virtual contiguas de tamaño variable llamado ***Segmento***
- Cada segmento se lo divide en partes iguales llamadas ***Páginas***
- La memoria virtual será de ***Segmentos paginados***
- Proceso:
  - El usuario requiere ejecutar un programa dentro del S.O.
  - El S.O. lo lee del disco y lo debiera cargar en una dirección inexistente
  - Le asigna una porción de memoria virtual y su número de segmento
  - El S.O. divide el segmento en páginas desde la página 0 hasta la N
  - El S.O. registra la dirección de inicio y fin de cada segmento
  - El S.O. toma la página 0 del segmento y lo copia en la memoria principal
  - La UGM (MMU) toma la porción de memoria principal y hace el cálculo de bloque (y etiqueta si corresponde) para copiar parte del mismo en caché
  - El primer dato del programa viaja a la CPU

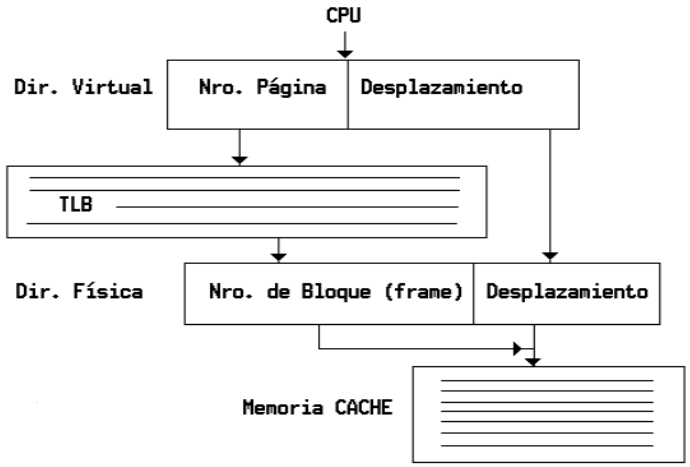
Memoria virtual: Segmentación y paginación



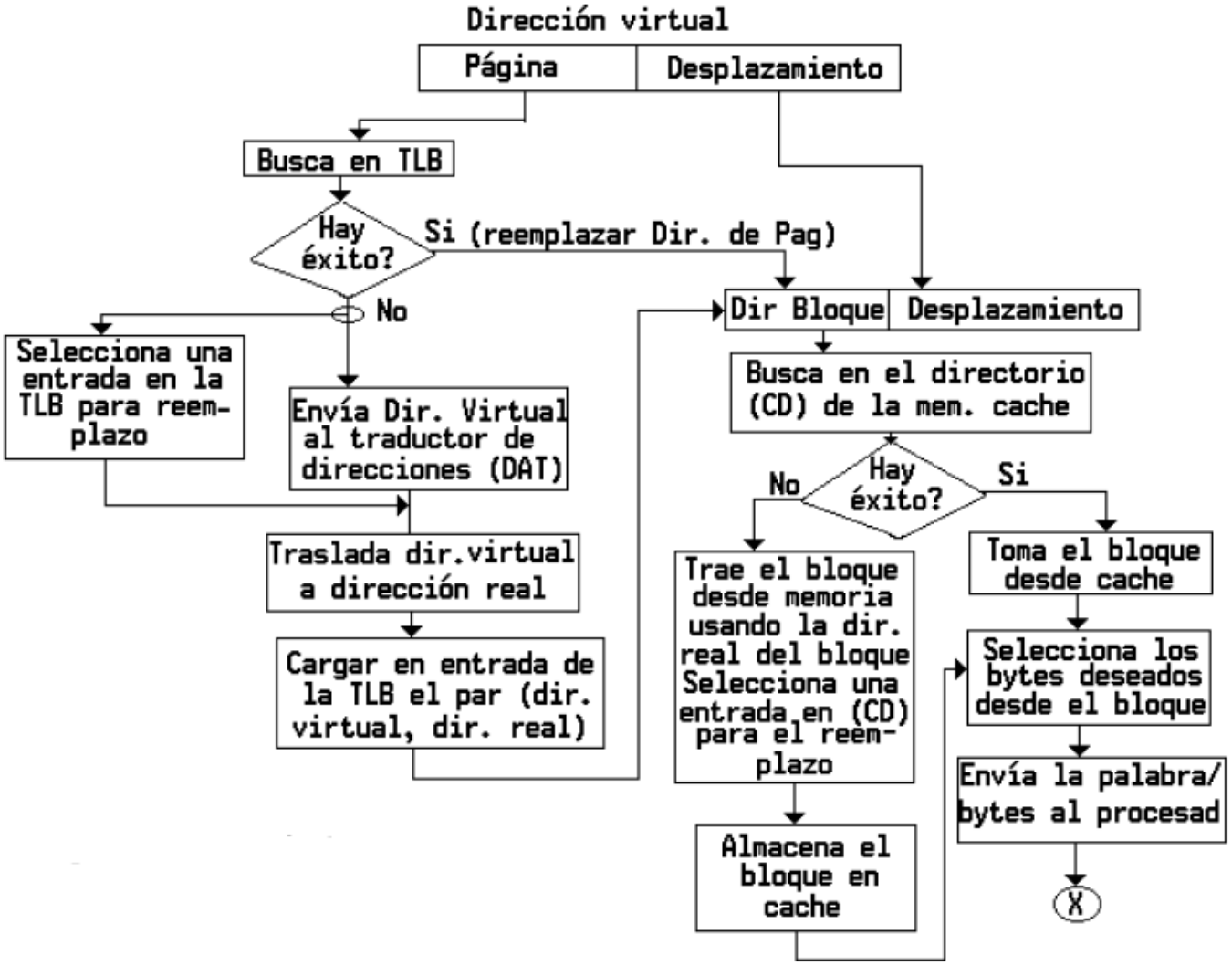
- Si la página se busca en MP no existe, nos encontraremos con una falla de página o ***Page Fault***. En cambio si está en MP, será un éxito o ***Page Hit***.
- Ante un ***Page Fault***, se deberá traer desde la MV la página faltante en la MP. ***Swapping*** es el intercambio de la página "saliente" de la MP y la nueva o "entrante" a la MP

Memoria virtual: Tabla de consulta rápida (TLB)

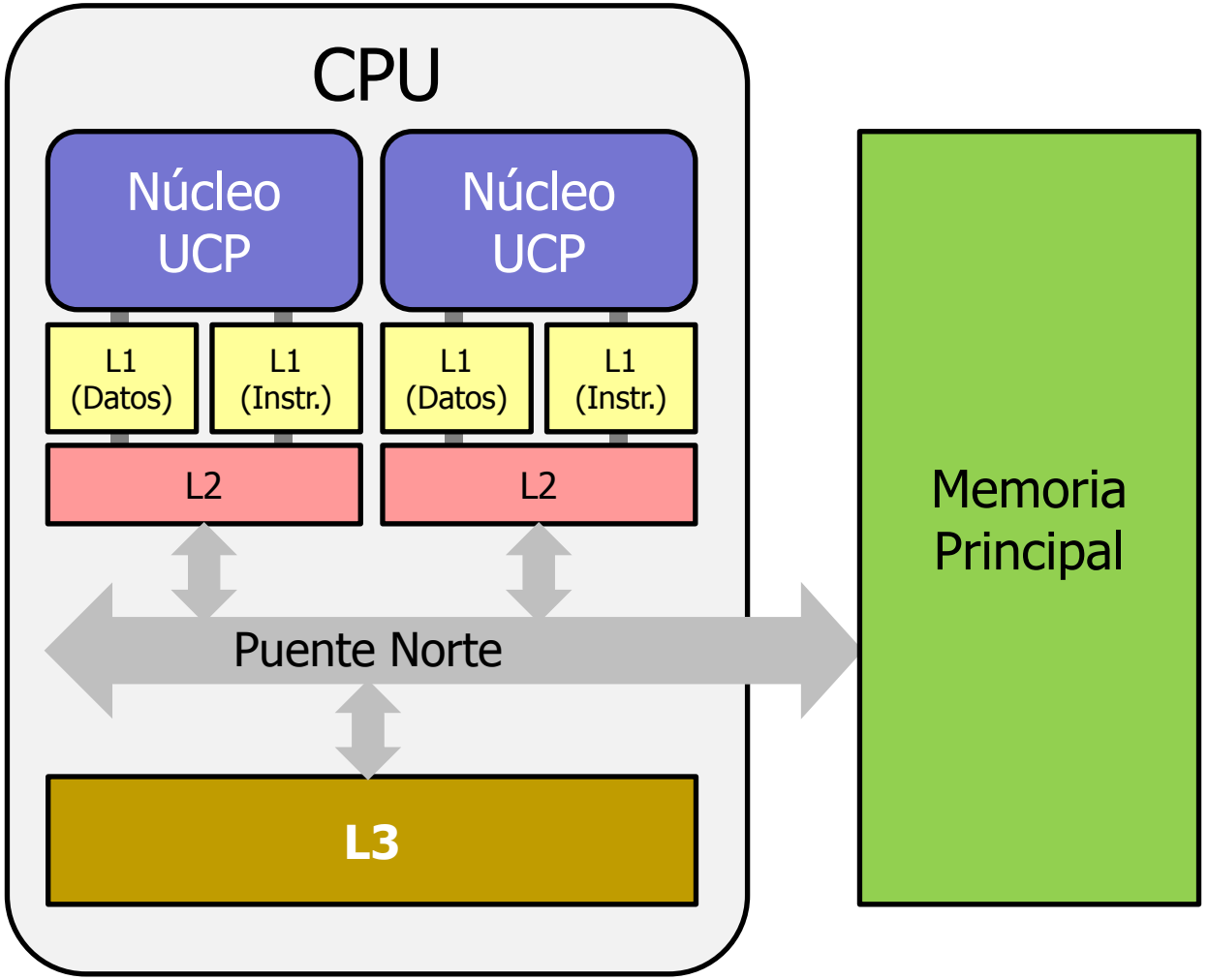
- Uso de la TLB (**T**able-**L**ookside-**B**uffer)
  - Las CPUs pueden direccionar espacios de memoria más allá de los *n* de bits (2^n) que posee en el bus de direcciones → Espacio virtual de memoria
  - La CPU traducirá esa dirección virtual de memoria (compuesta por un número de página + un desplazamiento) en una dirección real o física (compuesta por un número de bloque + un desplazamiento) para poder ubicarla en la memoria
  - La TLB es una memoria asociativa que almacena las últimas *p* traducciones
  - Si dirección virtual a traducir no está en la TLB, se resolverá a través de circuitería adicional



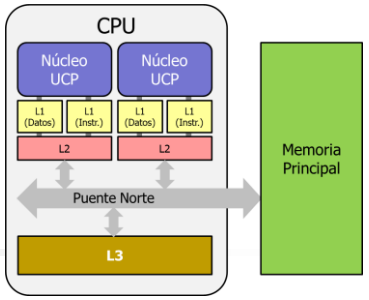
Memoria virtual: Flujograma de aciertos y fallos



Niveles de memoria caché

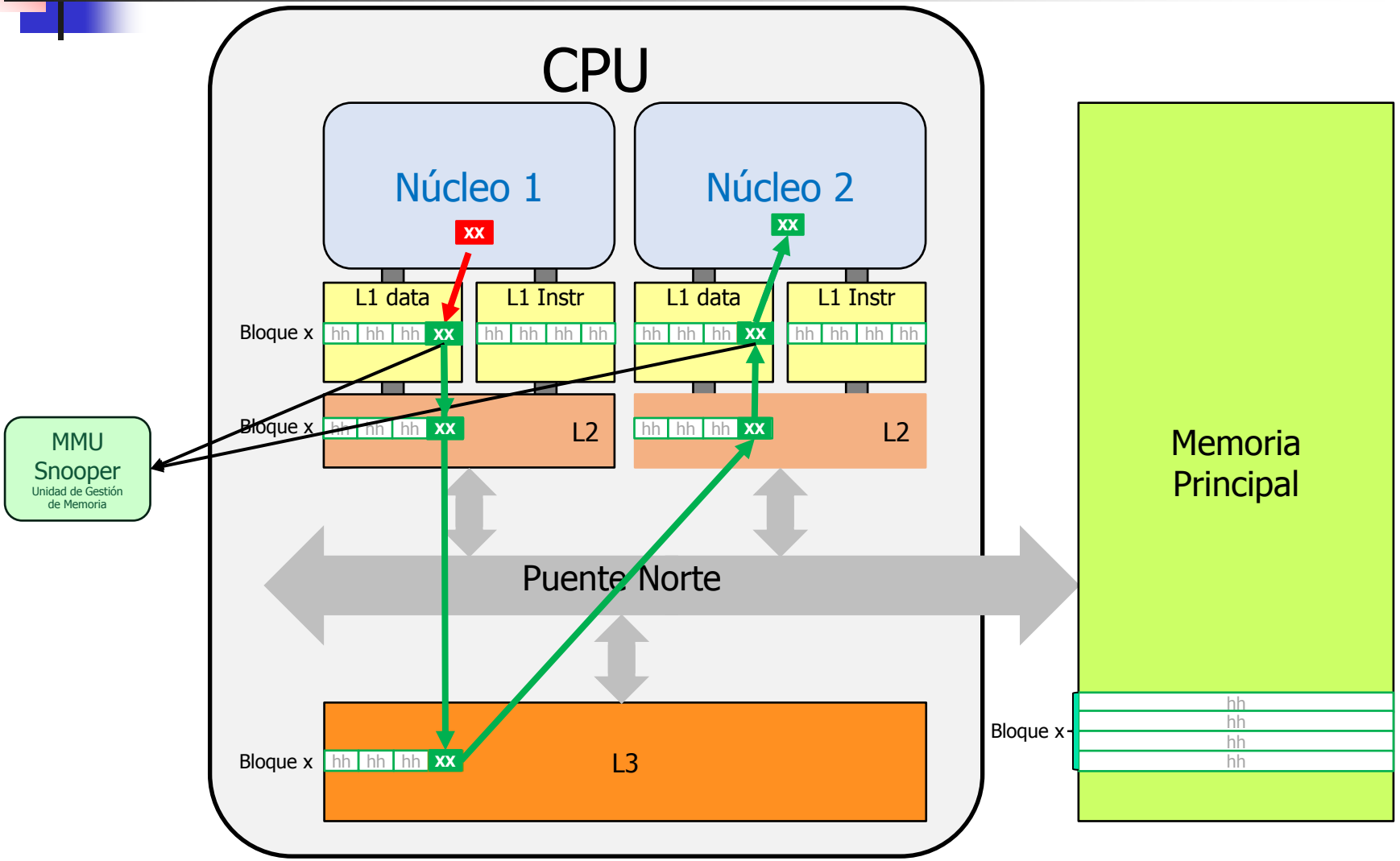


Memoria caché - Coherencia

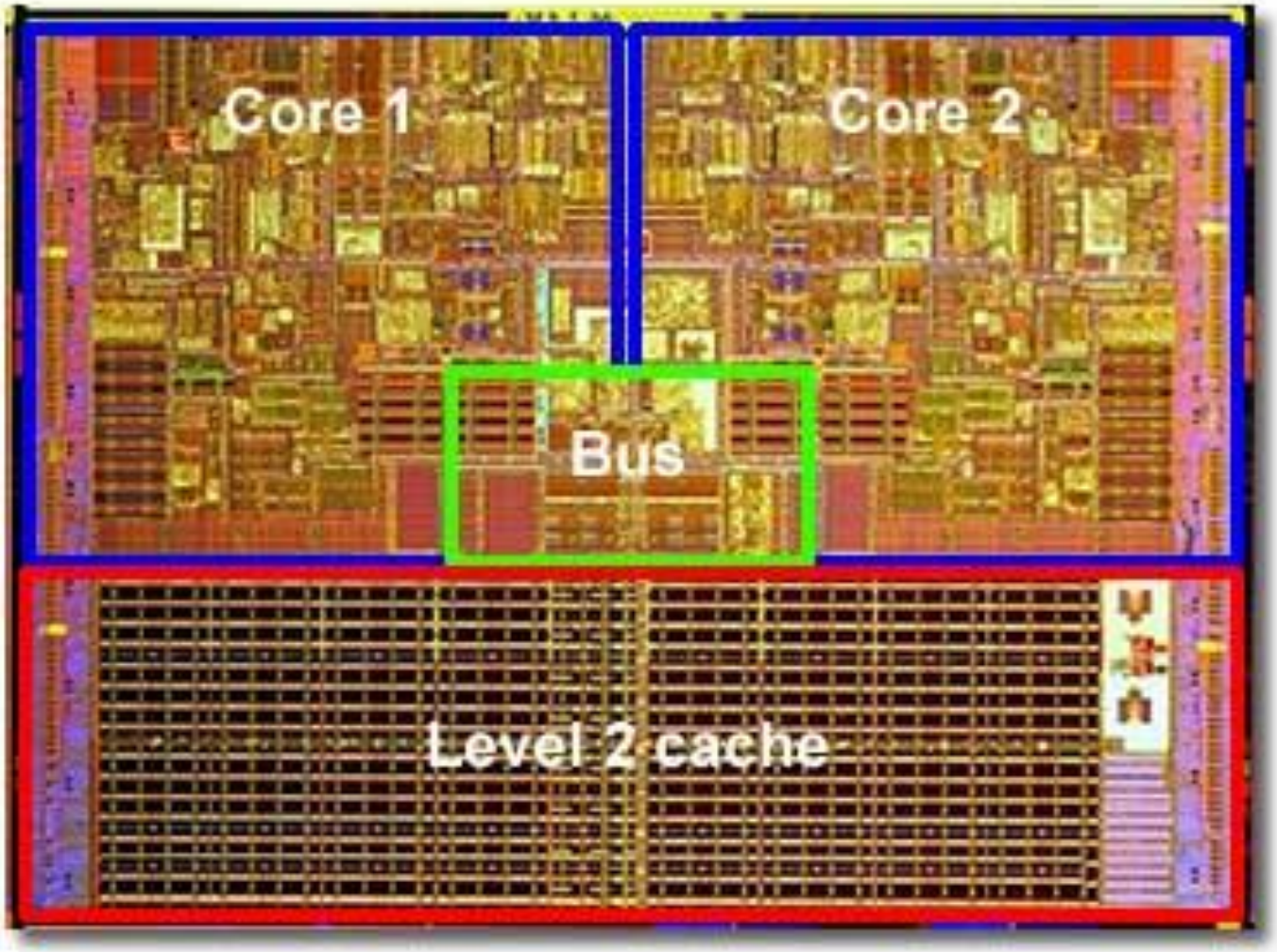


- Lo procesadores actuales son multi-núcleo (multicore)
- El nivel 3 (L3) de caché común a todos los núcleos posee una copia de la Memoria Principal de "m" cantidad de bloques
- A nivel 2 (L2), cada núcleo tiene su propio caché. Este nivel aloja "p" cantidad menor de bloques que "m", copiados de L3
- A nivel 1 (L1) vemos que cada núcleo tiene dos espacios de caché diferentes y físicamente separados: *Data* e *Instruction.* Por su parte, este nivel aloja "t" cantidad menor de bloques que "p", copiados de L2. Un proceso interno separa los códigos de las instrucciones de sus datos u objetos en cada caché L1
- Un bloque en "p" puede estar copiado en más de un núcleo a la vez según se haya requerido
- Si un núcleo modifica un dato de un bloque en L1 y luego otro núcleo requiere leerlo, la MMU realizará todos los retrograbados necesarios hasta asegurar la consistencia entre todos los niveles de caché
- Si se necesita desalojar un bloque "sucio" de L3, recién allí se realizará el retrograbado necesario en MP

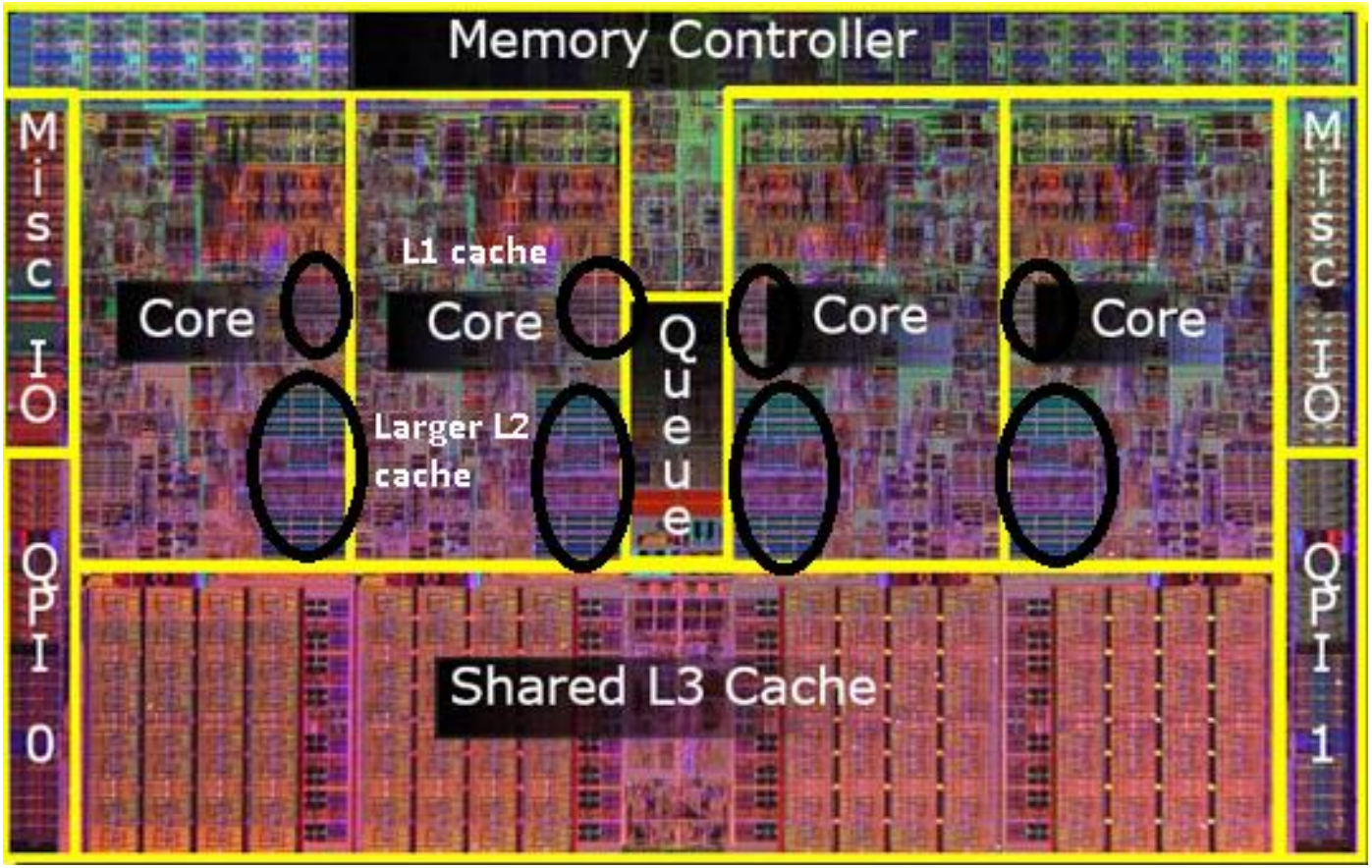
Memoria caché - Coherencia



Core 2 Duo – Caché L2 compartido



Procesador 4 núcleos – Caché L3 compartido



L1 = 32 KB

L2 = 256 KB

L3 = 8 MB