



Arquitectura de computadores I

Memoria Cache

Septiembre de 2017

Memoria cache

- ◆ Sistemas de Memoria
- ◆ Principios de la memoria Cache
- ◆ Elementos del diseño de la Cache
- ◆ Organización de la Cache en Pentium
- ◆ Organización de la Cache en ARM



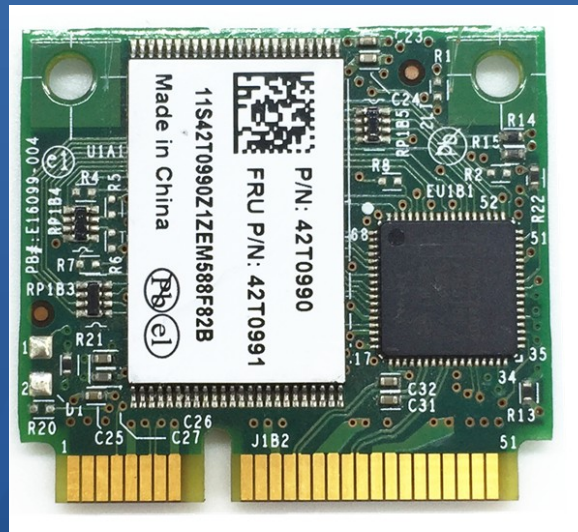
Sistemas de Memoria

Características sistemas memoria

- Ubicación
- Capacidad
- Transferencia
- Método de acceso
- Rendimiento
- Tipos y características físicas
- Organización

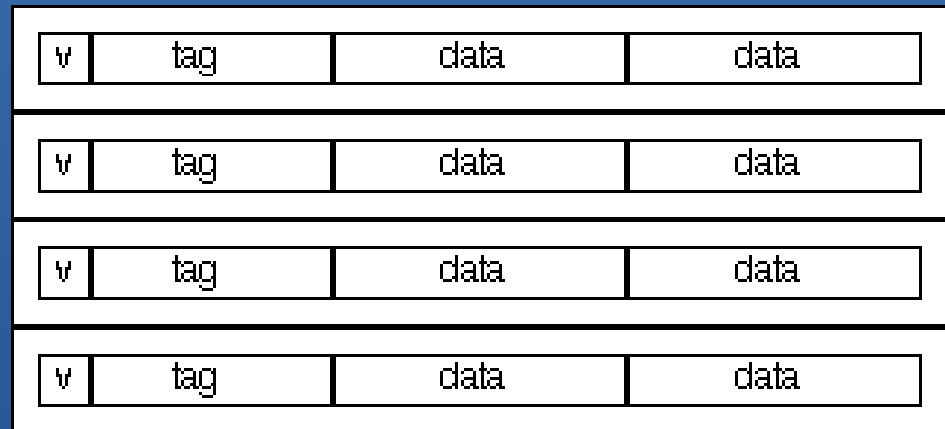
Ubicación

- CPU
- Externa
- Interna



Capacidad

- Tamaño de palabra
 - La unidad natural de organización
- Número de palabras
 - O bytes



Transferencia de datos

- Interna
 - Depende del tamaño del bus
- Externa
 - Se utilizan bloques que son agrupaciones de palabras
- Unidad direccionable
 - Menor localización que puede ser direccionada
 - Palabra

Métodos de acceso (1)

- Secuencial
 - Inicia en el comienzo de la memoria y va leyendo posición por posición
 - El tiempo de acceso depende de la localización
 - Ejemplo: Un caset
- Directo
 - Los bloques individuales tienen una única dirección
 - Acceso es un salto con respecto al acceso secuencial
 - El tiempo de acceso depende de la posición anterior
 - Ejemplo: Un disco

Métodos de acceso (2)

- Aleatorio

- Las direcciones individuales identifican direcciones exactamente
- El tiempo de acceso es independiente de la localización anterior o sucesos anteriores
- Ejemplo: RAM

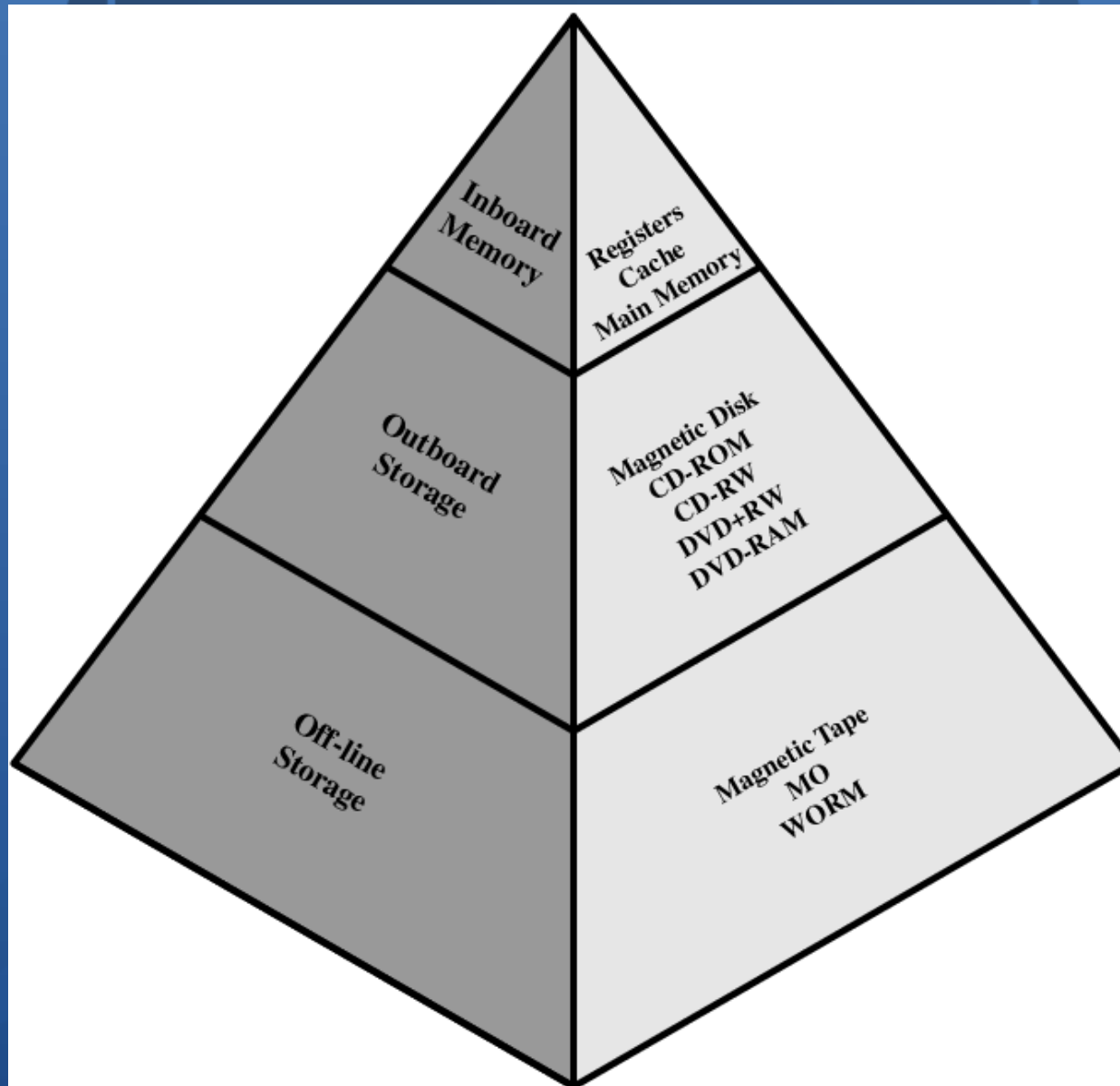
- Asociativo

- Los datos estan localizados por comparación con el contenido de una porción de los datos
- El tiempo de acceso es independiente de accesos previos
- Ejemplo cache

Jerarquía de memoria

- Registros
- Memoria interna o externa
 - Puede incluir uno o más niveles de cache
 - “RAM”
- Memoria externa
 - Almacenamiento

Diagrama de jerarquía de memoria



Rendimiento

- Tiempo de acceso
 - Tiempo entre la presentación de la dirección y la obtención de los datos
- Tiempo de ciclo de memoria
 - Tiempo que es requerido por la memoria para recuperarse antes del siguiente acceso
 - El tiempo del ciclo: Es acceso + recuperación
- Capacidad de transferencia
 - La capacidad está dada por cuantos datos pueden ser movidos

Rendimiento

- Capacidad de transferencia
 - Tiempo de acceso (Latencia)
 - Tiempo de ciclo de memoria
 - Tasa de transferencia

$$T_n = T_A + \frac{n}{R}$$

T_n Tiempo promedio de escritura o lectura de n bits

T_a Tiempo promedio de acceso

n Número de bits

R Tasa de transferencia (Bits por segundo bps)

Tipos físicos

- Semiconductor
 - RAM
- Magnético
 - Discos y casete
- Óptico
 - CD y DVD
- Otros
 - Burbuja
 - Holograma

Características físicas

- Durabilidad
- Volatibilidad
- Borrable
- Consumo de energía

Organización

- Arreglo físico de bits dentro palabras
- No siempre obvio
- Ejemplo: internivel

Lista jerárquica

- ◆ Registros
- ◆ L1 Cache
- ◆ L2 Cache
- ◆ Memoria principal
- ◆ Discos
- ◆ Opticos
- ◆ Casete

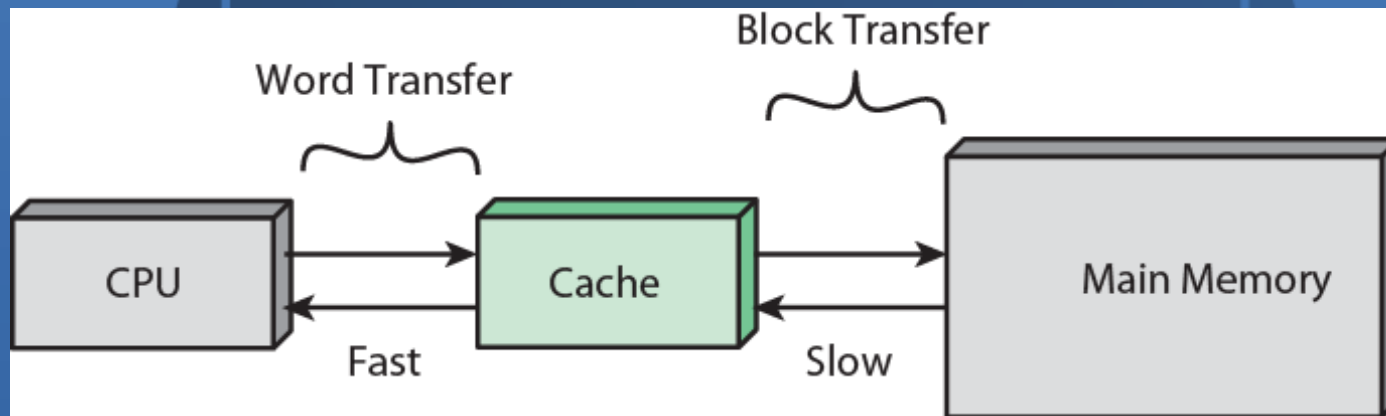


Principios de la memoria cache

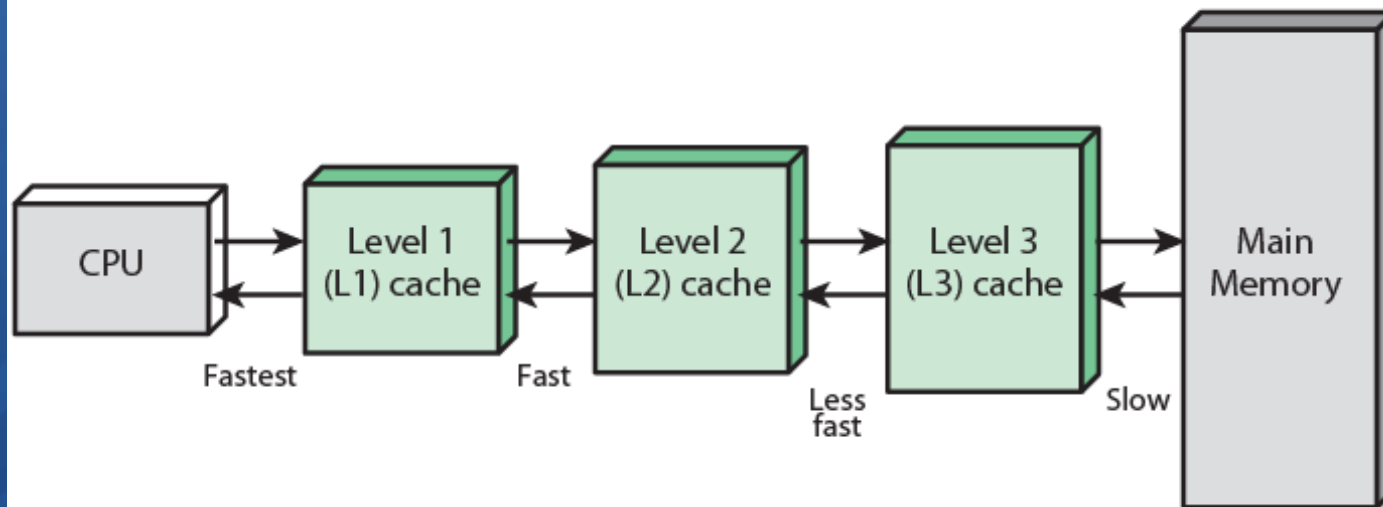
Cache

- ◆ Pequeña pero muy rápida
- ◆ Se encuentra entre la memoria principal y la CPU
- ◆ Podría estar localizada en el chip de la CPU o en un módulo

Cache y memoria principal

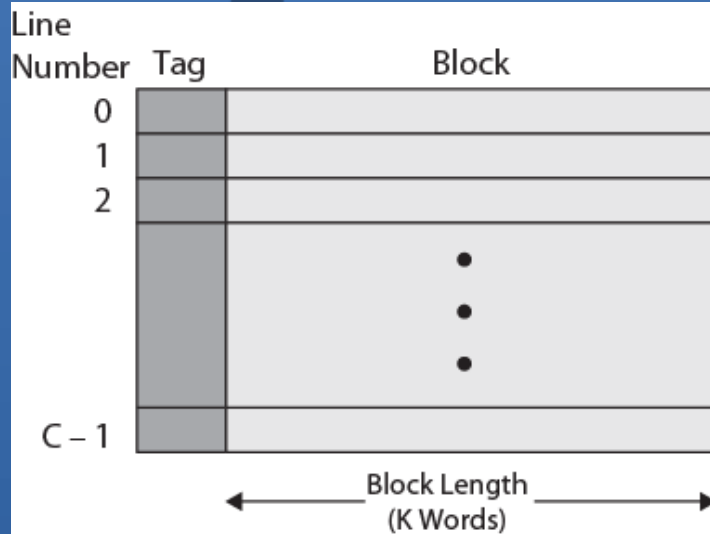


(a) Single cache

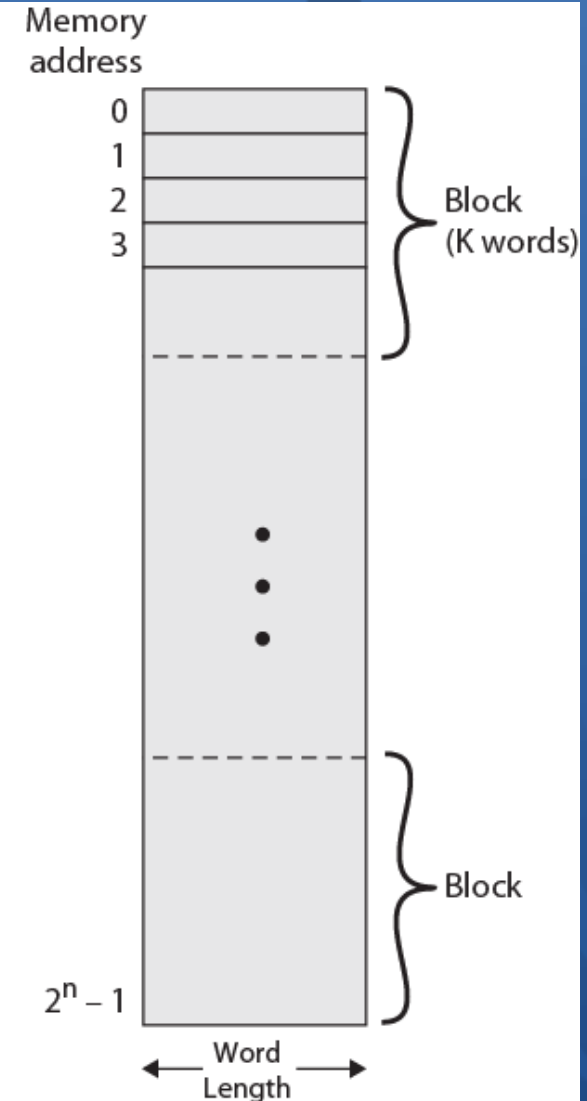


(b) Three-level cache organization

Estructura memoria principal y cache



(a) Cache

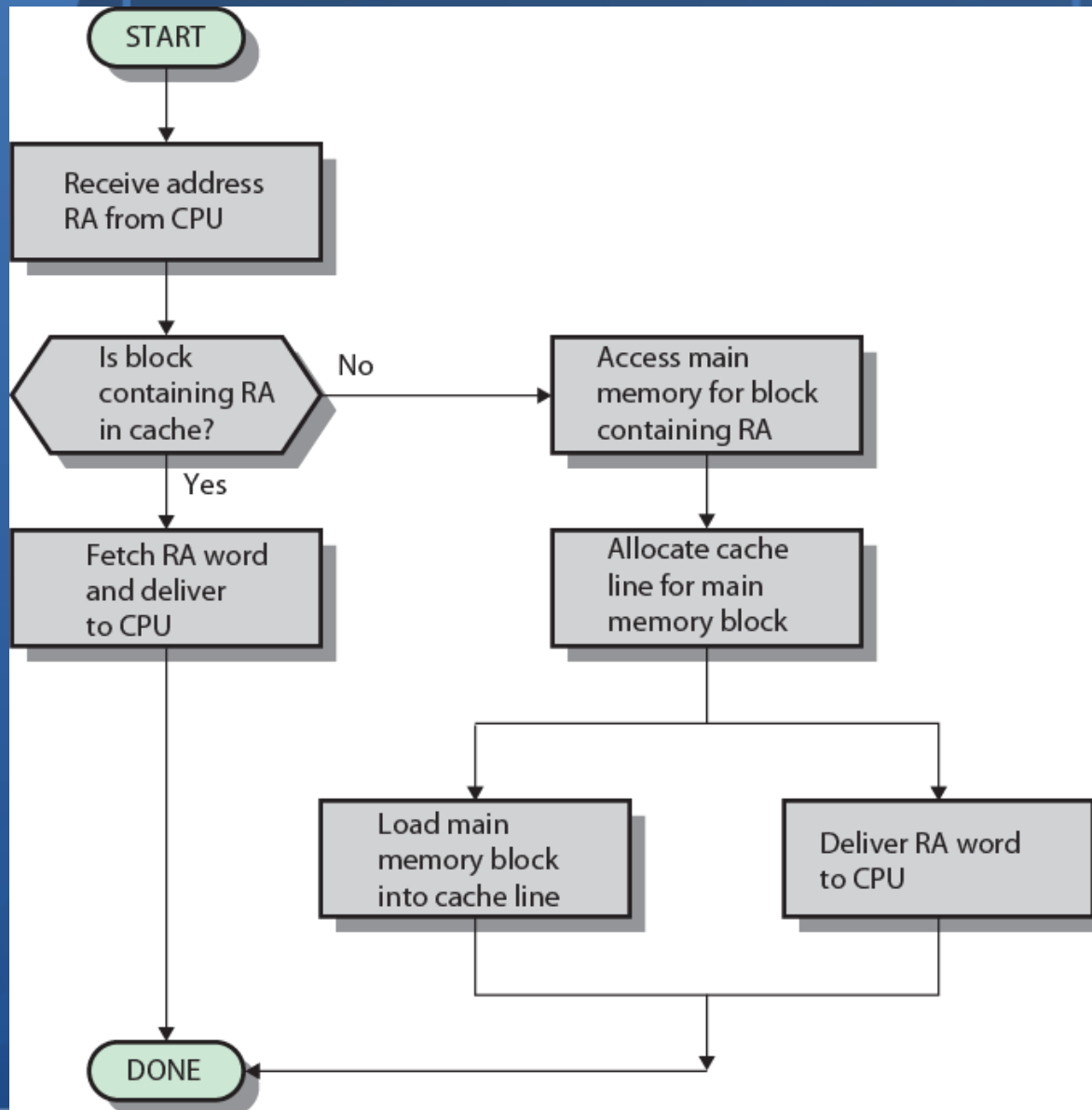


(b) Main memory

Operación de la cache

- ◆ La CPU requiere el contenido de una localización en memoria
- ◆ Se revisa la cache para estos datos
- ◆ Si está presente, se obtiene desde la cache (rápido)
- ◆ Si no está presente, se lee el bloque requerido desde la memoria principal hacia la cache
- ◆ Entonces, se envia desde la cache hacia la CPU
- ◆ La cache etiqueta cada bloque de memoria principal en cada slot de la cache

Operación de lectura de la cache



Diseño de la cache

- ◆ Direcccionamiento
- ◆ Tamaño
- ◆ Función de mapeo
- ◆ Algoritmo de reemplazo
- ◆ Política de escritura
- ◆ Tamaño del bloque
- ◆ Número de caches

Direccionamiento

- ◆ ¿Donde está la cache?
 - ◆ Entre el procesador y la unidad administradora de memoria (MMU)
 - ◆ Entre la MMU y la memoria principal
- ◆ La cache lógica (Cache virtual) almacena la información utilizando direcciones virtuales
 - ◆ Procesador accede la cache directamente, pero no la cache física
 - ◆ La cache se accede muy rápido
 - ◆ Las direcciones virtuales usan el mismo espacio para aplicaciones diferentes
- ◆ La caché física almacena información utilizando direcciones de memoria principal

Capacidad vs velocidad

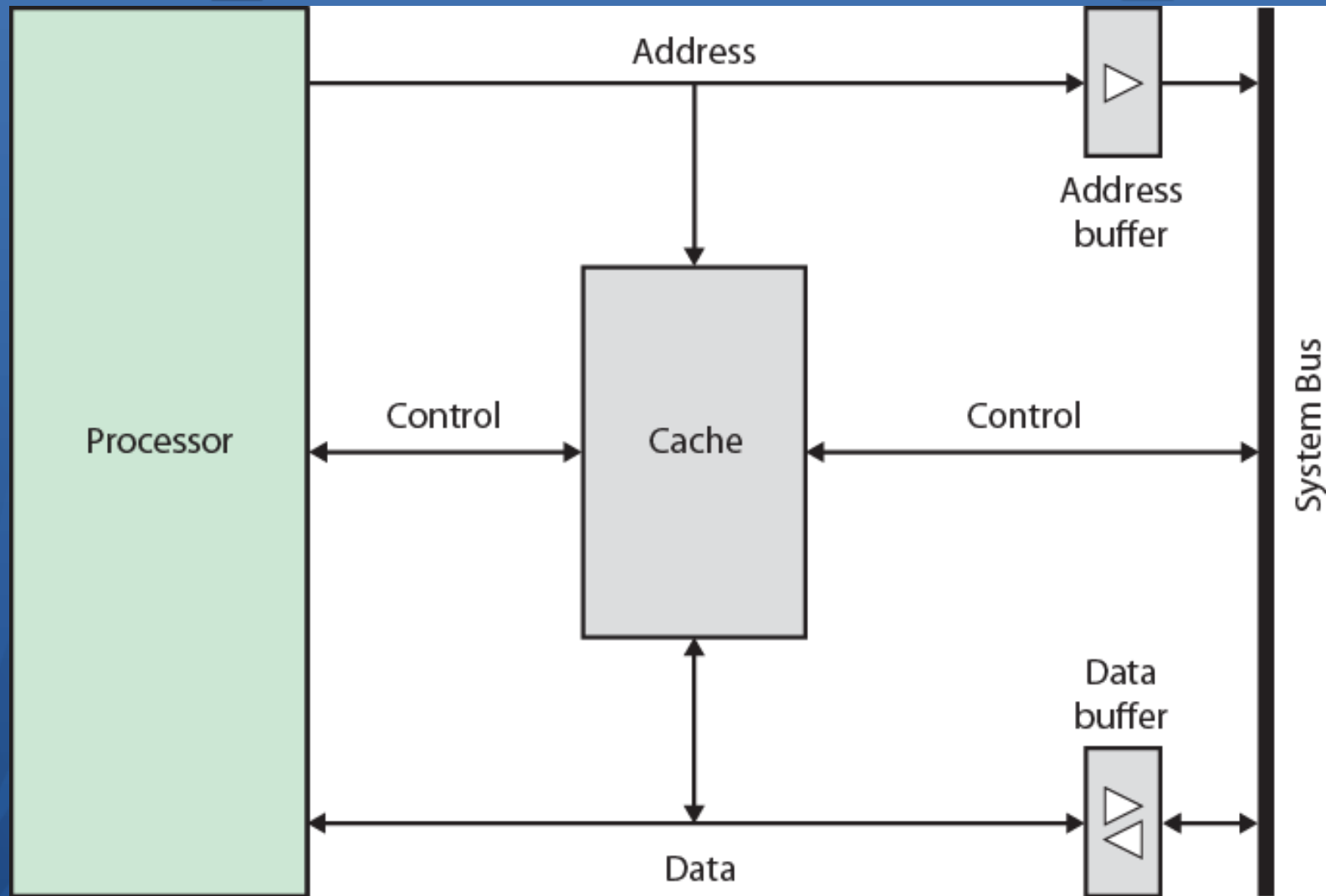
- ◆ Costo

- ◆ La cache es más costosa

- ◆ Velocidad

- ◆ Tener más cache implica mayor velocidad
- ◆ Verificar los datos de la cache toma tiempo

Organización de la cache





Elementos del diseño de la Cache

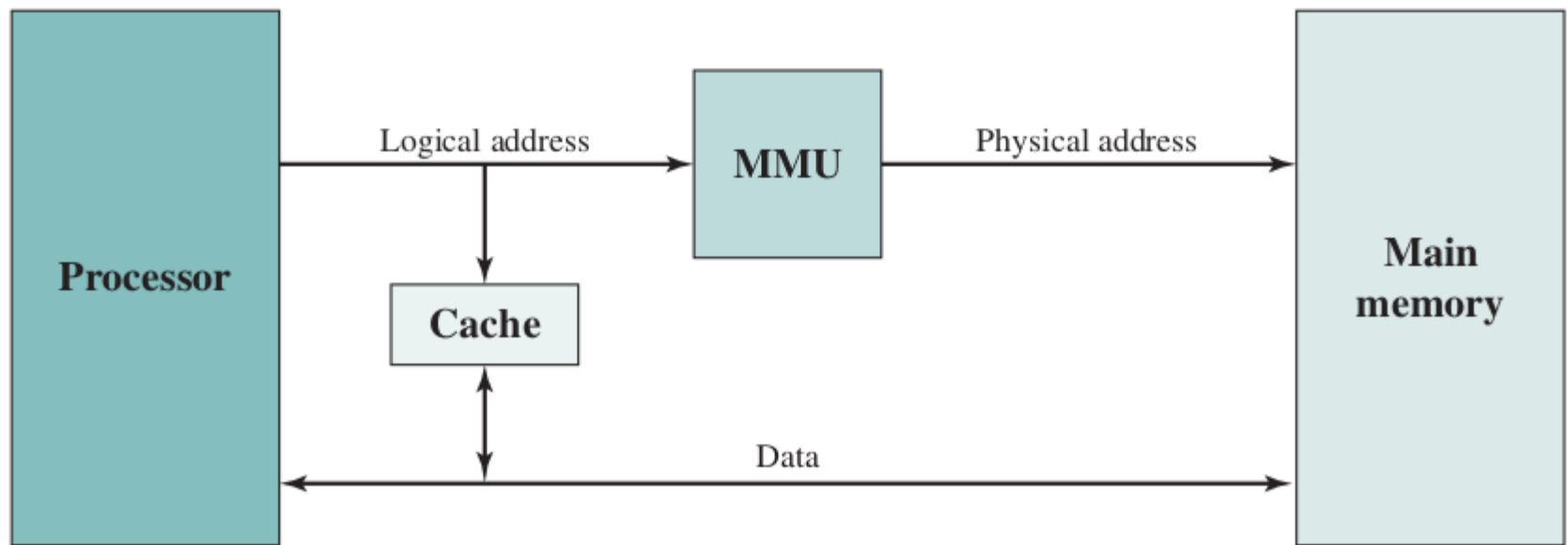
Direccionamiento

- ◆ Lógica: Es conocida como cache virtual, se almacena información utilizando direcciones virtuales. El procesador accede directamente a la memoria cache.
- ◆ Física: Almacena información usando la memoria principal

Direccionamiento

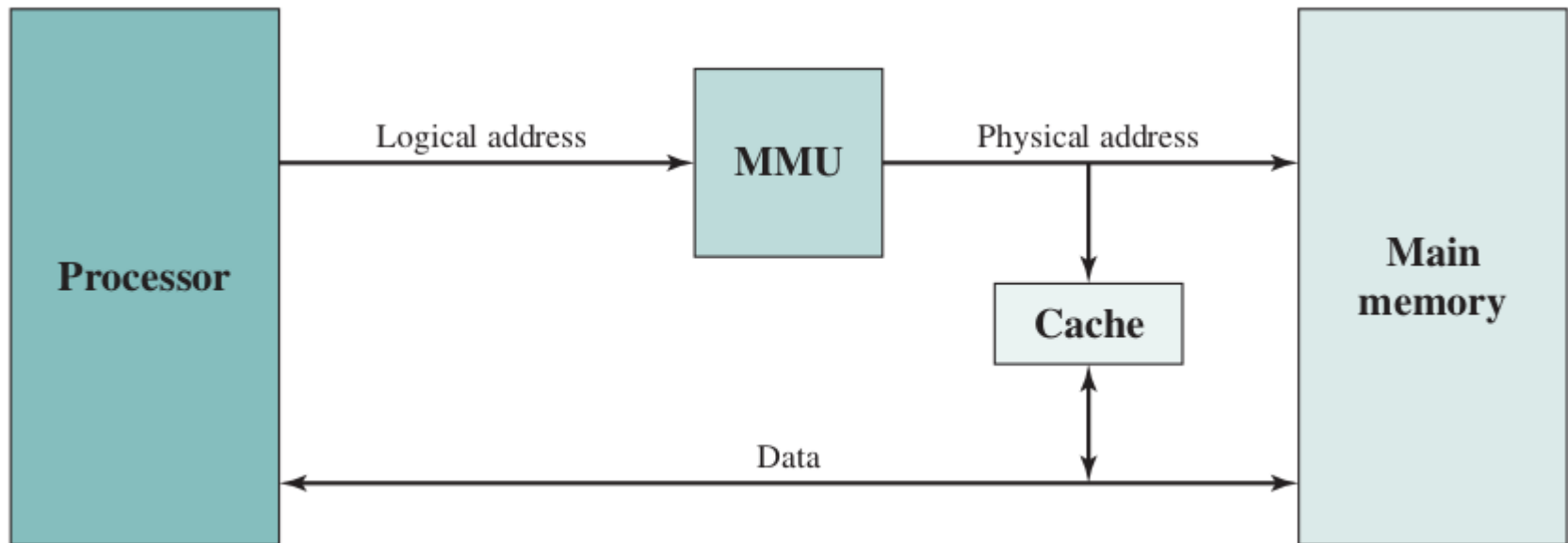
- ◆ Lógica: Es conocida como cache virtual, se almacena información utilizando direcciones virtuales. El procesador accede directamente a la memoria cache.
- ◆ Física: Almacena información usando la memoria principal

Direcccionamiento



(a) Logical cache

Direcccionamiento



(b) Physical cache

Comparación de cache

Processor	Type	Year of Introduction	L1 cache	L2 cache	L3 cache
IBM 360/85	Mainframe	1968	16 to 32 KB	—	—
PDP-11/70	Minicomputer	1975	1 KB	—	—
VAX 11/780	Minicomputer	1978	16 KB	—	—
IBM 3033	Mainframe	1978	64 KB	—	—
IBM 3090	Mainframe	1985	128 to 256 KB	—	—
Intel 80486	PC	1989	8 KB	—	—
Pentium	PC	1993	8 KB/8 KB	256 to 512 KB	—
PowerPC 601	PC	1993	32 KB	—	—
PowerPC 620	PC	1996	32 KB/32 KB	—	—
PowerPC G4	PC/server	1999	32 KB/32 KB	256 KB to 1 MB	2 MB
IBM S/390 G4	Mainframe	1997	32 KB	256 KB	2 MB
IBM S/390 G6	Mainframe	1999	256 KB	8 MB	—
Pentium 4	PC/server	2000	8 KB/8 KB	256 KB	—
IBM SP	High-end server/ supercomputer	2000	64 KB/32 KB	8 MB	—
CRAY MTA _b	Supercomputer	2000	8 KB	2 MB	—
Itanium	PC/server	2001	16 KB/16 KB	96 KB	4 MB
SGI Origin 2001	High-end server	2001	32 KB/32 KB	4 MB	—
Itanium 2	PC/server	2002	32 KB	256 KB	6 MB
IBM POWER5	High-end server	2003	64 KB	1.9 MB	36 MB
CRAY XD-1	Supercomputer	2004	64 KB/64 KB	1MB	—

Función de mapeo

- Cache de 64 KB
- Bloque de 4 bytes
 - Si la cache es de 16k (2^{14}) líneas de 4 bytes
- 16MBytes memoria principal
- Direcciones de 24 bit
 - ($2^{24}=16\text{M}$)

Mapeo directo

- ◆ Cada bloque de memoria principal se mapea solamente en una linea de cache
- ◆ Direcciones en dos partes
- ◆ W bits más significantes identifican la palabra
- ◆ S bits más significantes identifican un bloque de memoria
- ◆ Los bits más significantes son partidos dentro de la cache, una linea r tiene una etiqueta $s-r$

Mapeo directo

El mapeo directo está dado por

$$i = j \bmod m$$

i Número de la linea de la cache

j Número del bloque de memoria

m Número de lineas en la cache

Mapeo directo

La función de mapeo se implementa de la siguiente forma:

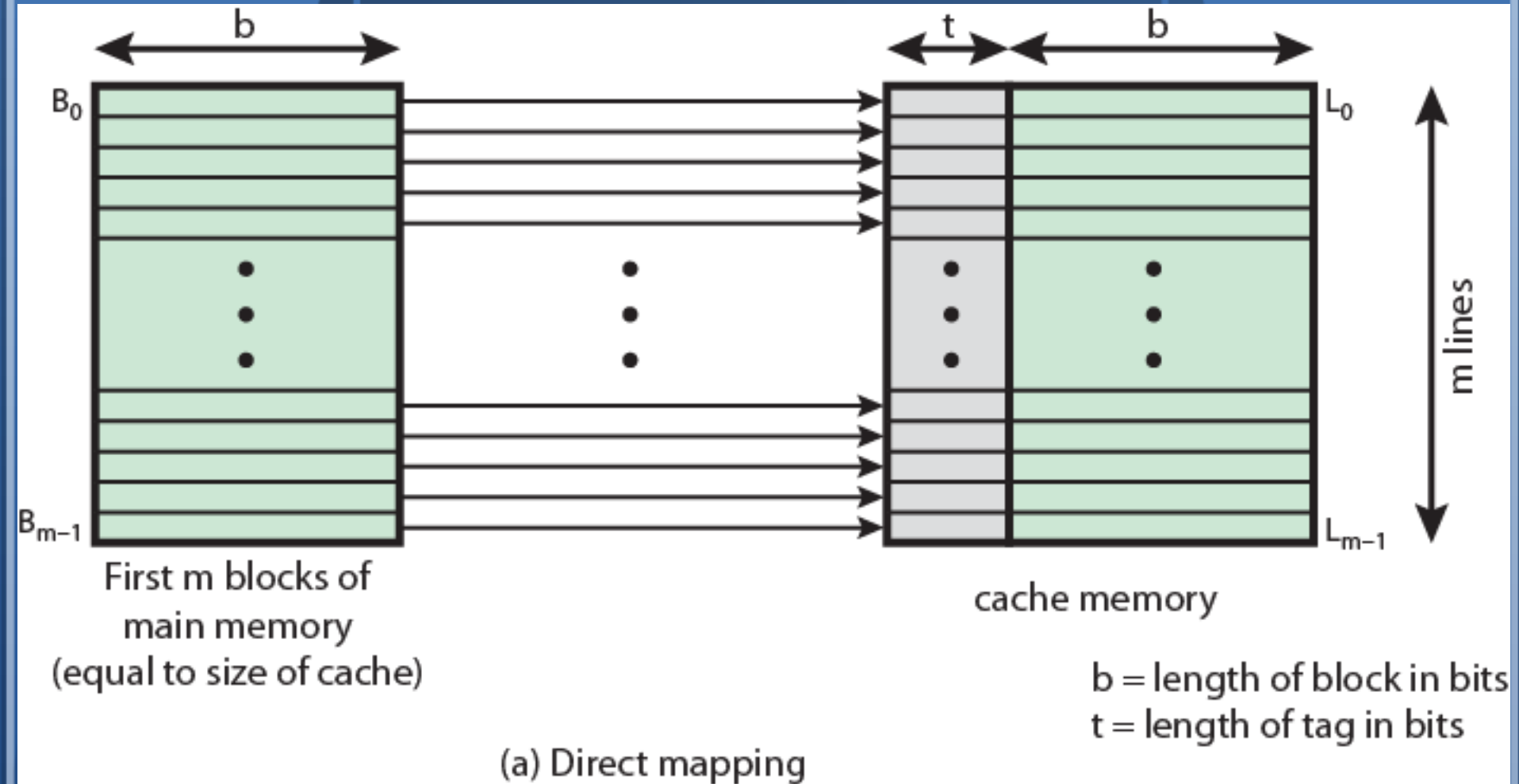
- ◆ Tamaño de la dirección $(s+w)$ bits
- ◆ Número de unidades direccionables 2^{s+w}
- ◆ Número de bloques en memoria principal 2^s
- ◆ Número de líneas en la cache = $m = 2^r$
- ◆ Tamaño de la cache 2^{r+w} palabras o bytes
- ◆ Tamaño de la etiqueta $(s - r)$ bits

Mapeo directo

Etiqueta s-r	Linea r	Palabra w
8	14	2

- ◆ Direcciones de 24 bit
- ◆ Identificador de palabra de 2 bits (Bloque de 4 bytes)
- ◆ Identificador de bloque de 22 bits
 - ◆ Etiqueta de 8 bits ($=22-14$)
 - ◆ 14 bits de linea
- ◆ No hay dos bloques en la misma linea que tengan la misma etiqueta
- ◆ Tenemos una cache de 64KB

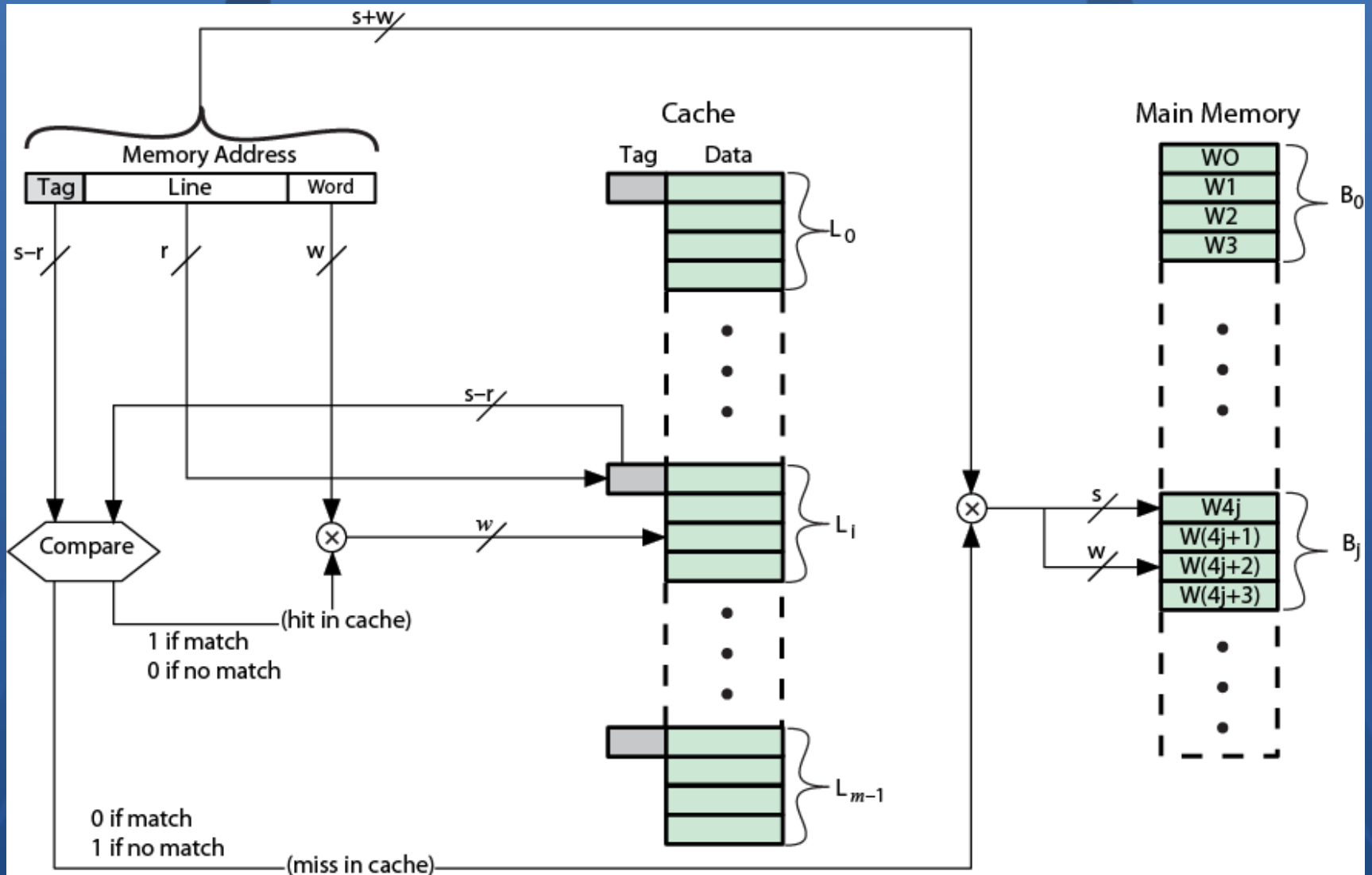
Mapeo directo desde la cache – memoria principal

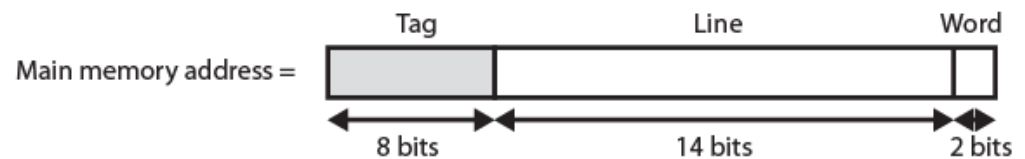
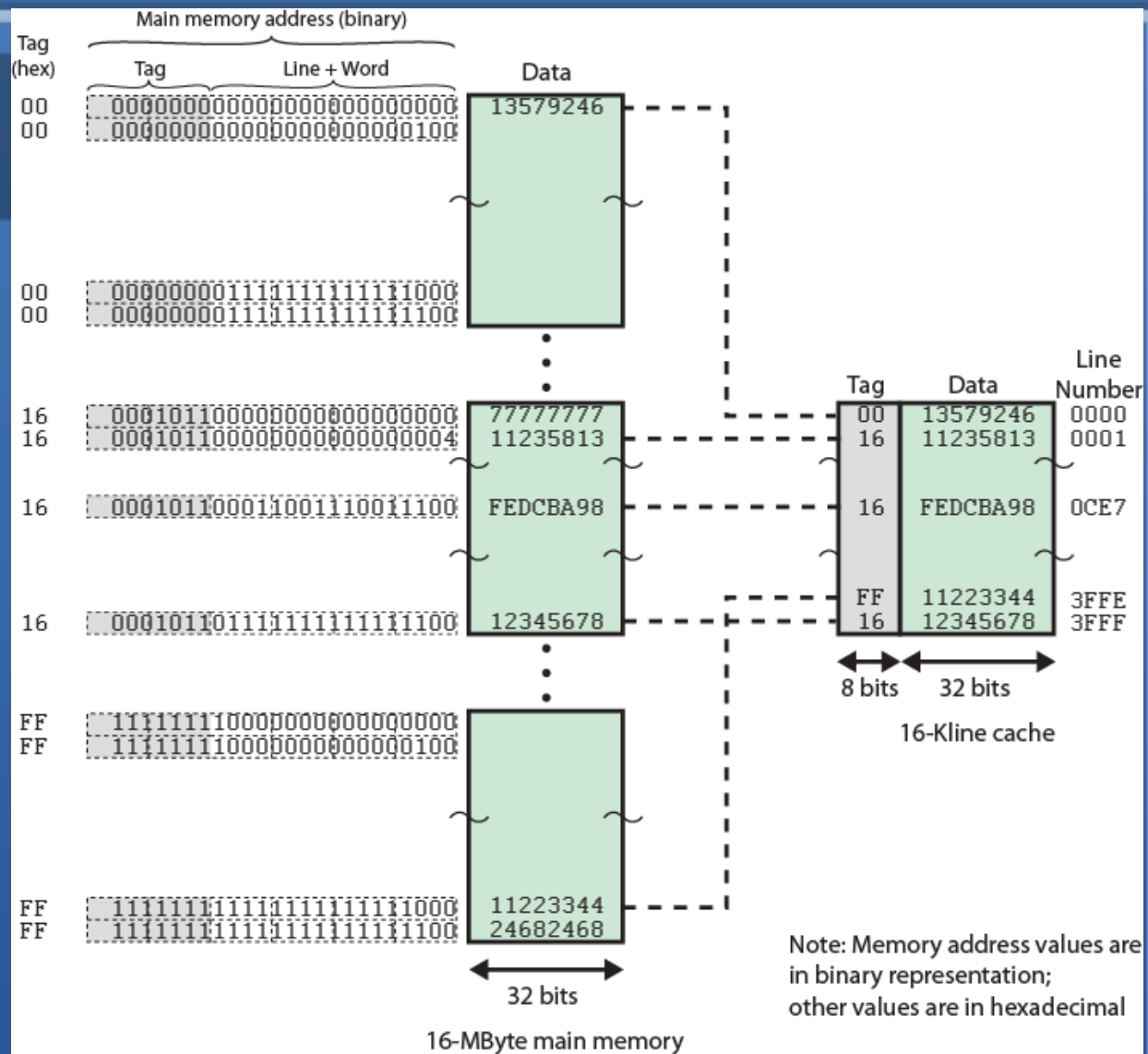


Mapeo directo

Línea de cache	Bloques de memoria principal
0	0, m, 2m, 3m...2s-m
1	1,m+1, 2m+1...2s-m+1
...	
m-1	m-1, 2m-1,3m-1...2s-1

Organización del mapeo directo





Mapeo asociativo

- Un bloque de memoria puede ser cargado en una posición de la cache
- La dirección de memoria es interpretada como una etiqueta y una palabra
- Una etiqueta identifica una dirección de memoria
- Cada etiqueta es examinada para una coincidencia
- La búsqueda en cache es costosa

Mapeo asociativo

- Tamaño de dirección = $(s + w)$ bits
- Número de unidades direccionables = 2^{s+w}
- Tamaño bloque = tamaño línea = 2^w
- Número de bloques en memoria principal = 2^s
- Número de líneas en memoria = no determinado
- Tamaño de la etiqueta = s bits

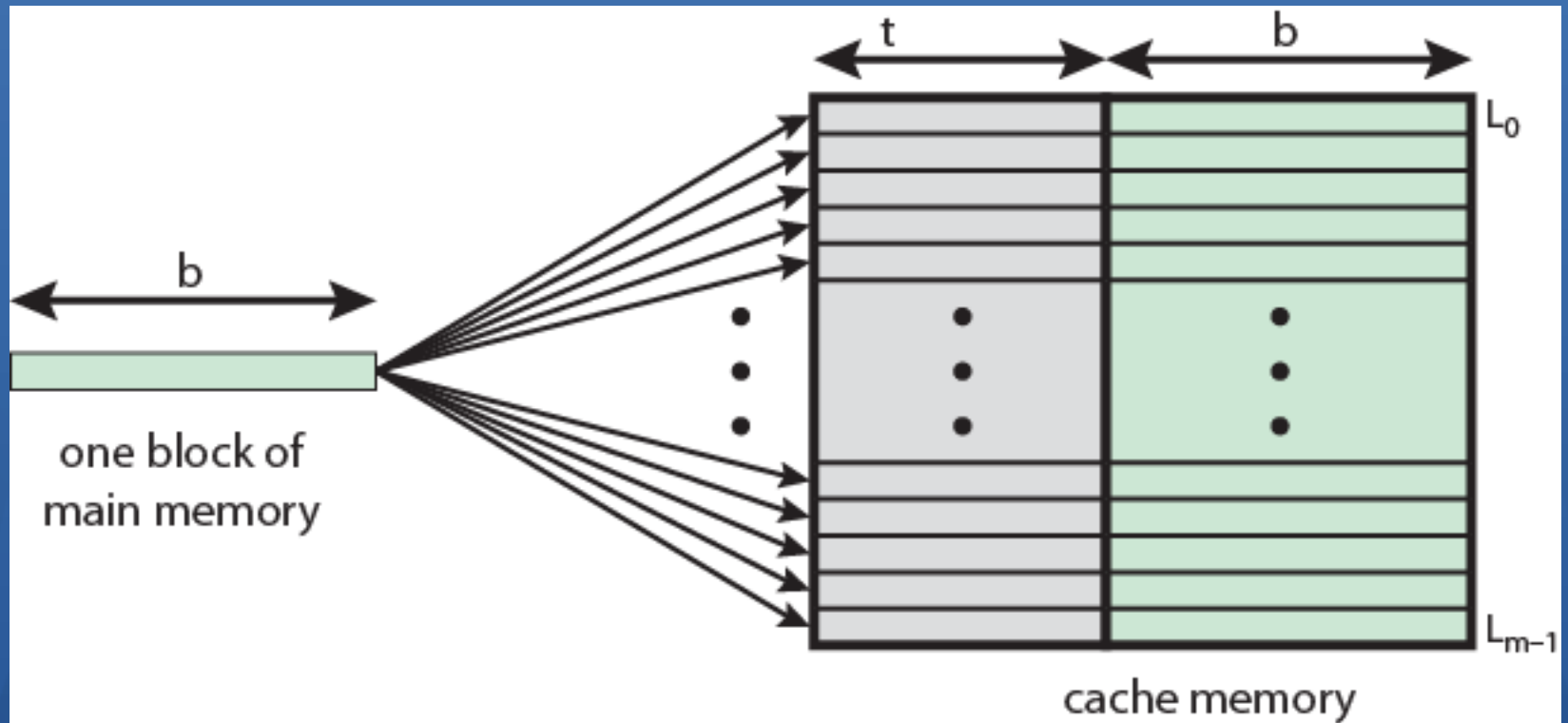
Mapeo asociativo: Estructura de las direcciones

Etiqueta 22 bit	Palabra 2 bit
-----------------	---------------

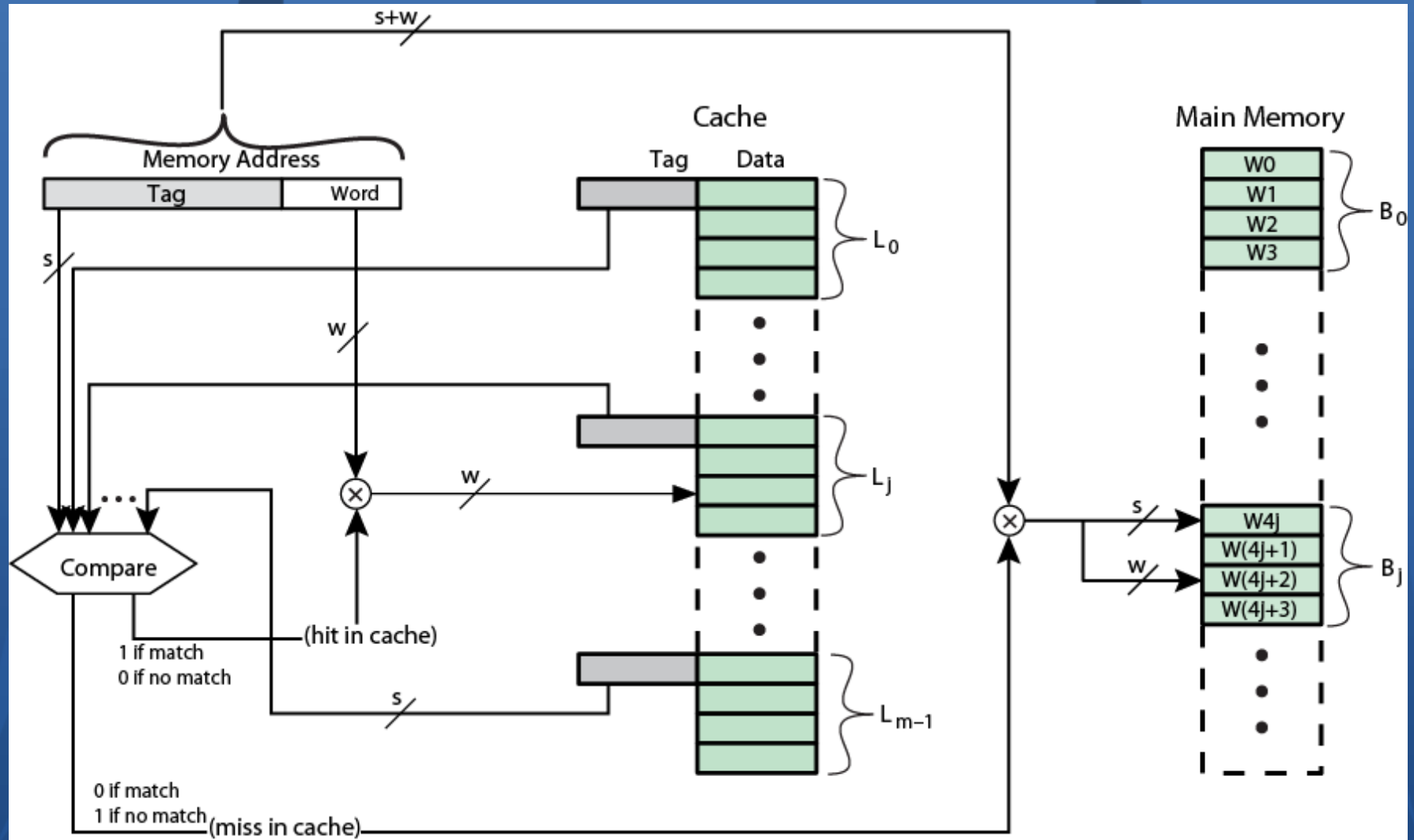
- ◆ Etiqueta de 22 bit, en un bloque de 32 bits
- ◆ Se compara la etiqueta con la etiqueta de la entrada para chequear colisiones
- ◆ 2 bits menos significantes identifican cual palabra de 16 bits es requerida desde un bloque de 32 bits
- ◆ Ejemplo

Dirección	Etiqueta	Datos	Linea de cache
FFFFFC	FFFFFC	24682468	3FFF

Mapeo asociativo



Organización mapeo asociativo





Mapeo asociativo por conjuntos

El mapeo asociativo está dado por

$$m = v * k$$

$$i = j \bmod m$$

- i Número de la linea de la cache
- j Número del bloque de memoria
- m Número de lineas en la cache
- v Número de conjuntos
- k Número de lineas en cada conjunto

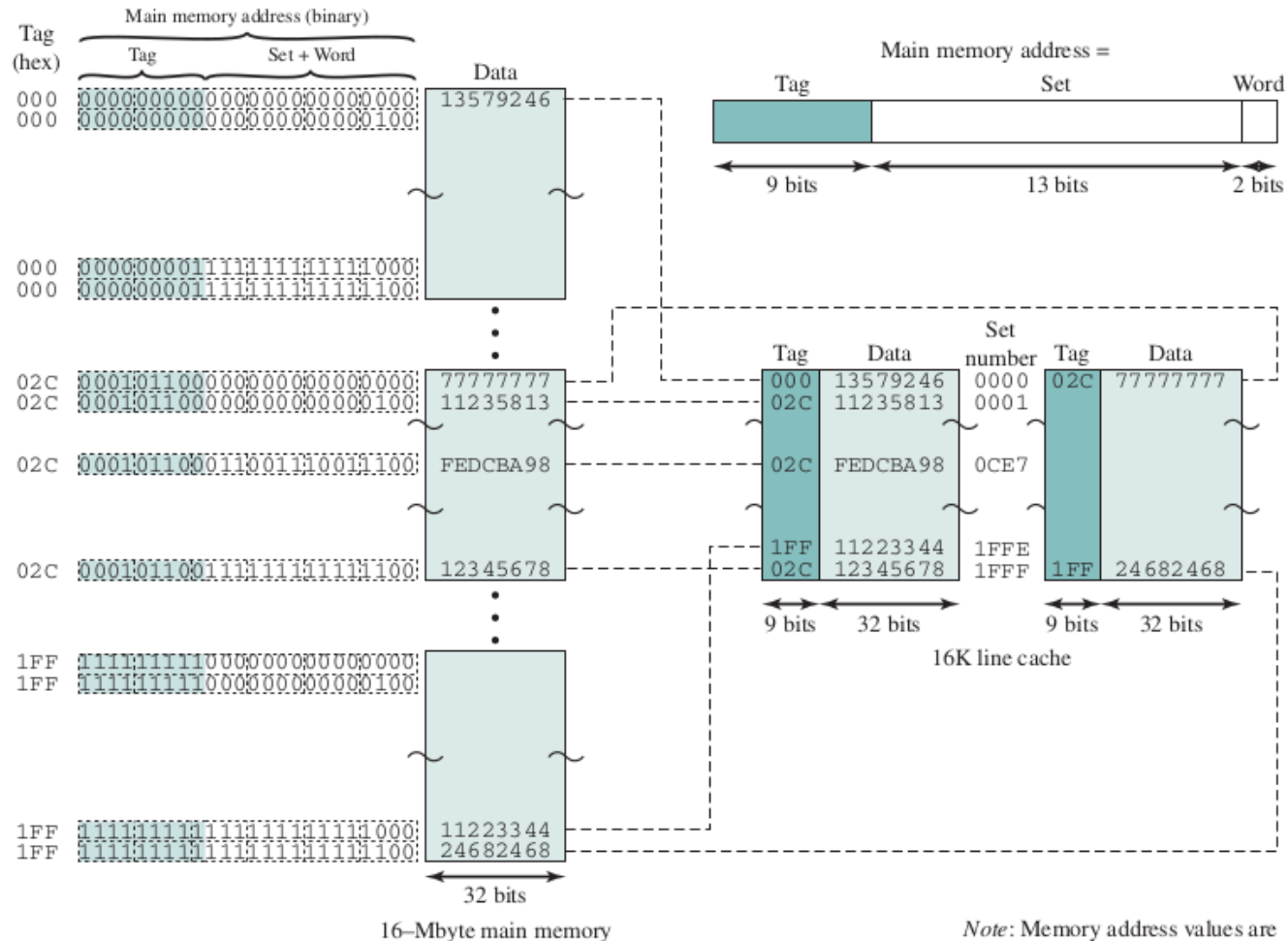
Mapeo asociativo por conjuntos

- Tamaño de dirección = $(s + w)$ bits
- Número de unidades direccionables = 2^{s+w}
- Tamaño bloque = tamaño línea = 2^w
- Número de bloques en memoria principal = 2^s
- Número de líneas en el conjunto = k
- Número de conjuntos $v = 2^d$
- Número de líneas en memoria = $kv = k 2^d$
- Tamaño de la etiqueta = $(s-d)$ bits

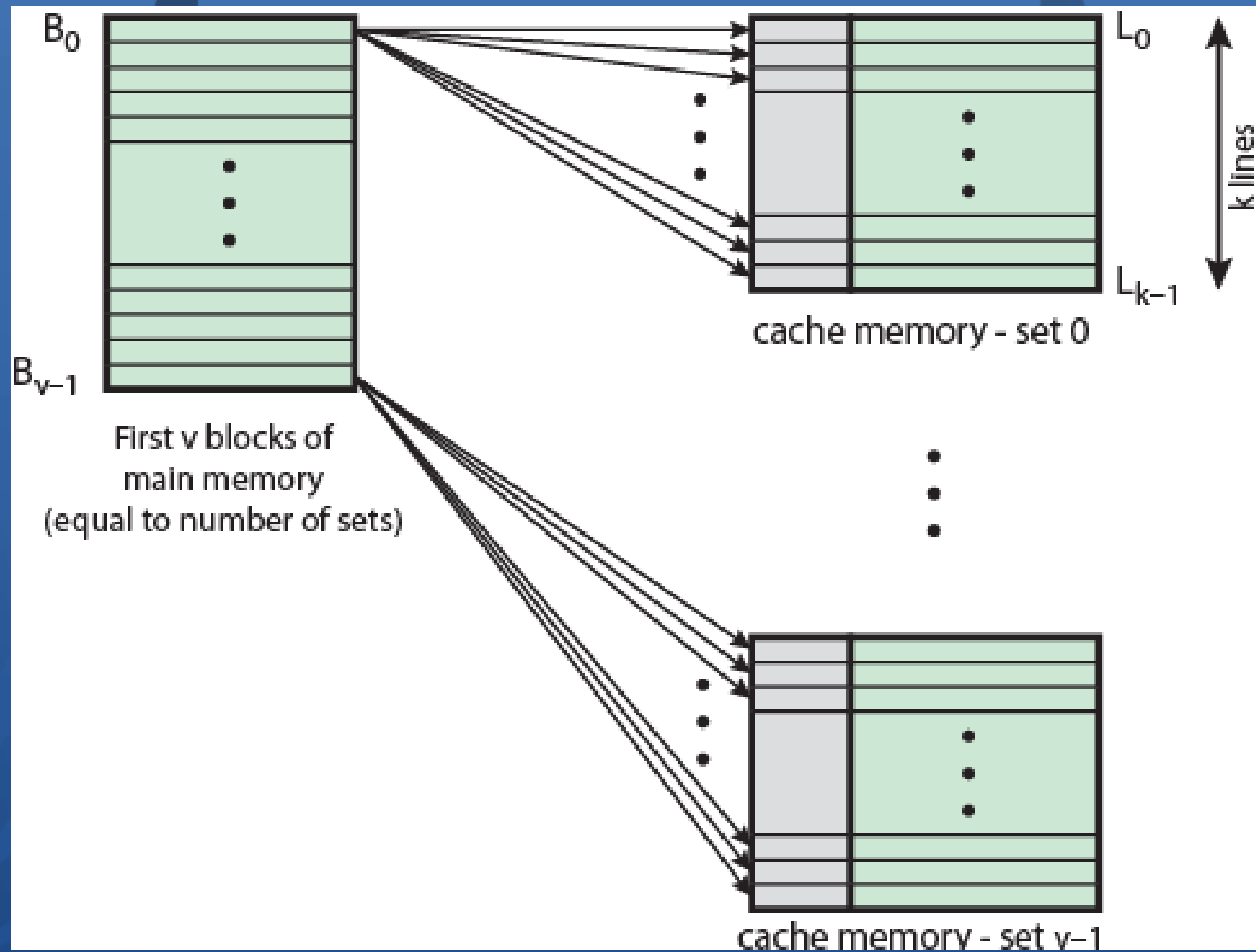
Mapeo asociativo por conjuntos

- ◆ La cache es dividida en un número de conjuntos
- ◆ Cada conjunto contiene un número de líneas
- ◆ Un bloque mapea una línea en un conjunto dado
 - ◆ Ejemplo. Bloque B puede estar en una línea de un conjunto i
- ◆ Ejemplo 2 líneas por conjunto
 - ◆ Un bloque dado puede estar en una de dos líneas en solo un conjunto

Mapeo asociativo por conjuntos



Mapeo asociativo



Algoritmos de reemplazo (1)

Mapeo directo

- ◆ Cada bloque es mapeado en una sola línea
- ◆ Reemplaza esta línea

Algoritmos de reemplazo (2)

Asociativo

- ◆ Primero en entrar primero en salir(FIFO)
 - ◆ Reemplace los bloques por el orden que ingresaron
- ◆ Menos frecuentemente usado
 - ◆ Reemplce el bloque que tiene menos colisiones
- ◆ Aleatorio

Política de escritura

- ◆ No se debe sobrecribir la memoria a menos que la memoria principal esté actualizada
- ◆ CPUs multiples puede tener caches individuales
- ◆ E/S podrían direccionar memoria principal directamente



Organización de la Cache en Pentium

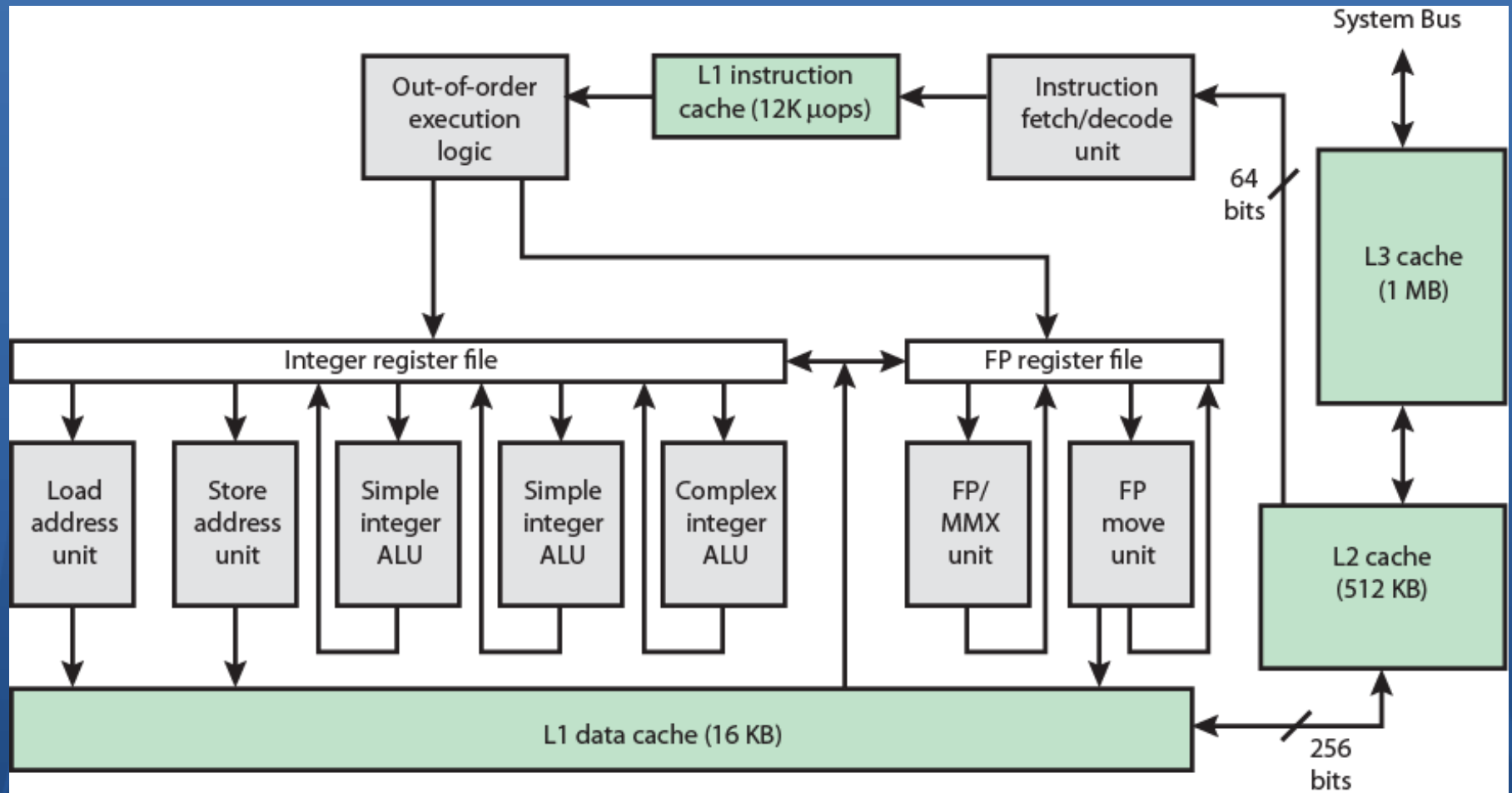
Pentium 4 Cache

- ◆ 80386 – no on chip cache
- ◆ 80486 – 8k using 16 byte lines and four way set associative organization
- ◆ Pentium (all versions) – two on chip L1 caches
 - ◆ Data & instructions
- ◆ Pentium III – L3 cache added off chip
- ◆ Pentium 4
 - ◆ L1 caches
 - ◆ 8k bytes
 - ◆ 64 byte lines
 - ◆ four way set associative
 - ◆ L2 cache
 - ◆ Feeding both L1 caches
 - ◆ 256k
 - ◆ 128 byte lines
 - ◆ 8 way set associative
 - ◆ L3 cache on chip

Evolución de cache de Intel

Problem	Solution	Processor on which feature first appears
External memory slower than the system bus.	Add external cache using faster memory technology.	386
Increased processor speed results in external bus becoming a bottleneck for cache access.	Move external cache on-chip, operating at the same speed as the processor.	486
Internal cache is rather small, due to limited space on chip	Add external L2 cache using faster technology than main memory	486
Contention occurs when both the Instruction Prefetcher and the Execution Unit simultaneously require access to the cache. In that case, the Prefetcher is stalled while the Execution Unit's data access takes place.	Create separate data and instruction caches.	Pentium
Increased processor speed results in external bus becoming a bottleneck for L2 cache access.	Create separate back-side bus that runs at higher speed than the main (front-side) external bus. The BSB is dedicated to the L2 cache.	Pentium Pro
	Move L2 cache on to the processor chip.	Pentium II
Some applications deal with massive databases and must have rapid access to large amounts of data. The on-chip caches are too small.	Add external L3 cache.	Pentium III
	Move L3 cache on-chip.	Pentium 4

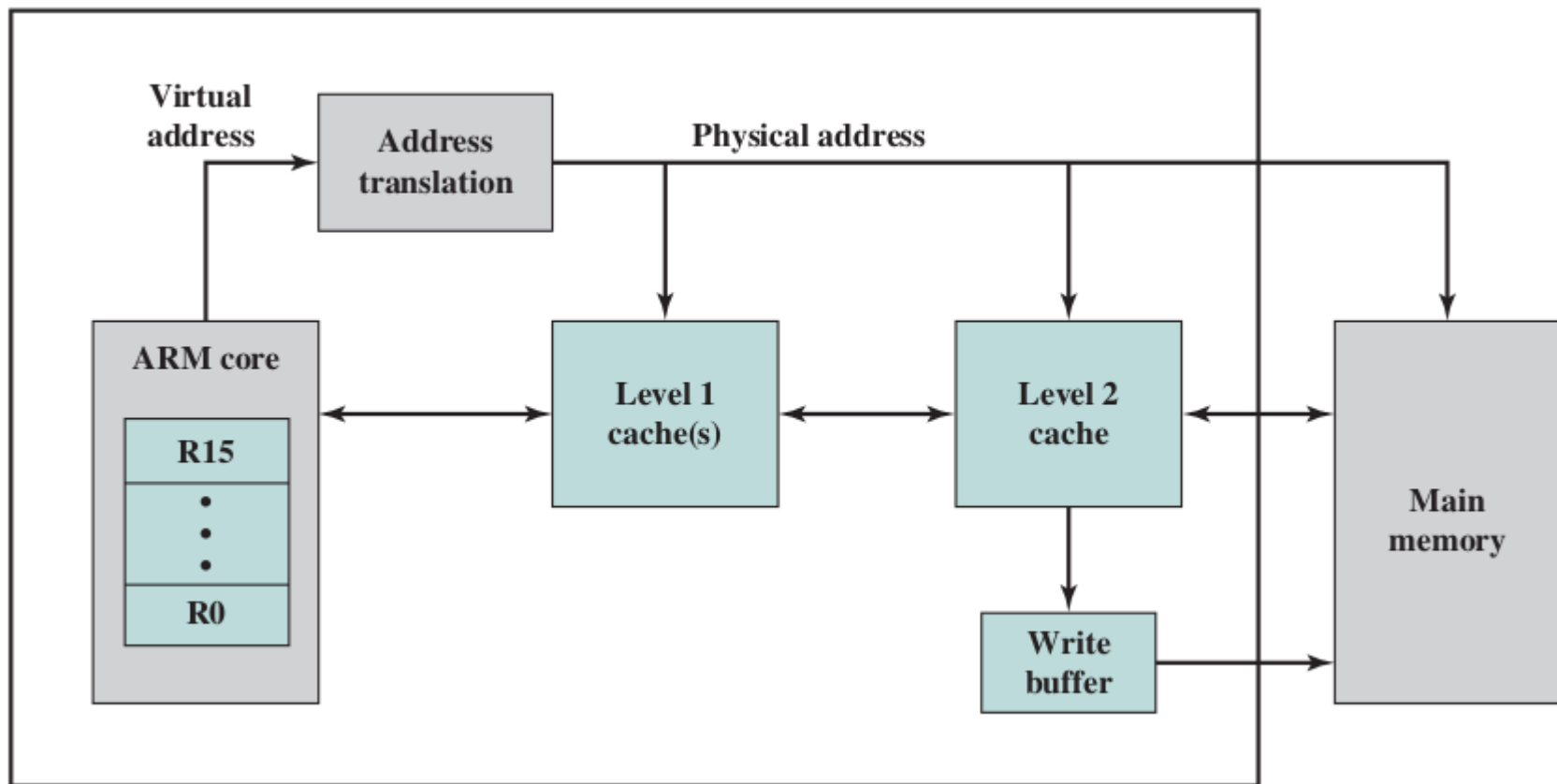
Pentium 4





Organización de la Cache en ARM

Características de cache de ARM



Características de cache de ARM

Core	Cache Type	Cache Size (kB)	Cache Line Size (words)	Associativity	Location	Write Buffer Size (words)
ARM720T	Unified	8	4	4-way	Logical	8
ARM920T	Split	16/16 D/I	8	64-way	Logical	16
ARM926EJ-S	Split	4-128/4-128 D/I	8	4-way	Logical	16
ARM1022E	Split	16/16 D/I	8	64-way	Logical	16
ARM1026EJ-S	Split	4-128/4-128 D/I	8	4-way	Logical	8
Intel StrongARM	Split	16/16 D/I	4	32-way	Logical	32
Intel Xscale	Split	32/32 D/I	8	32-way	Logical	32
ARM1136-JF-S	Split	4-64/4-64 D/I	8	4-way	Physical	32

Referencias

William Stallings. 2013. Computer Organization and Architecture: Designing for Performance (9th Edition). Prentice-Hall, Inc., Upper Saddle River, NJ, USA. Chapter 4

Gracias

¿Preguntas?

Próxima clase: Memoria cache

Gracias

¿Preguntas?

Próximo tema: Memoria principal