4. Cache Memory

Memory System

.Location
- CPU - registers
- Internal - main memory
- External - secondary memory (disk, tape)

.Capacity number of bytes or words

word size:
- 16 bits (Intel 8088)
- 32 bits (IBM S/370) Vax
- 64 bits (Supercomputers)

.Access Method

Sequential	tape, disk
Direct	disk
Random	main memory (uniform access time)
Associative	cache (content-addressable)

.Performance

Access time cpu: time to read/write
disk: time to position the head to the desired block

Memory cycle time: read - read (Note: refresh or regenerate time)

cf. Cache memory

Transfer rate

Main memory:	1/(cycle time)	
Disk:	Ta + N / R	N bits
	↑ access time	Rate in bps

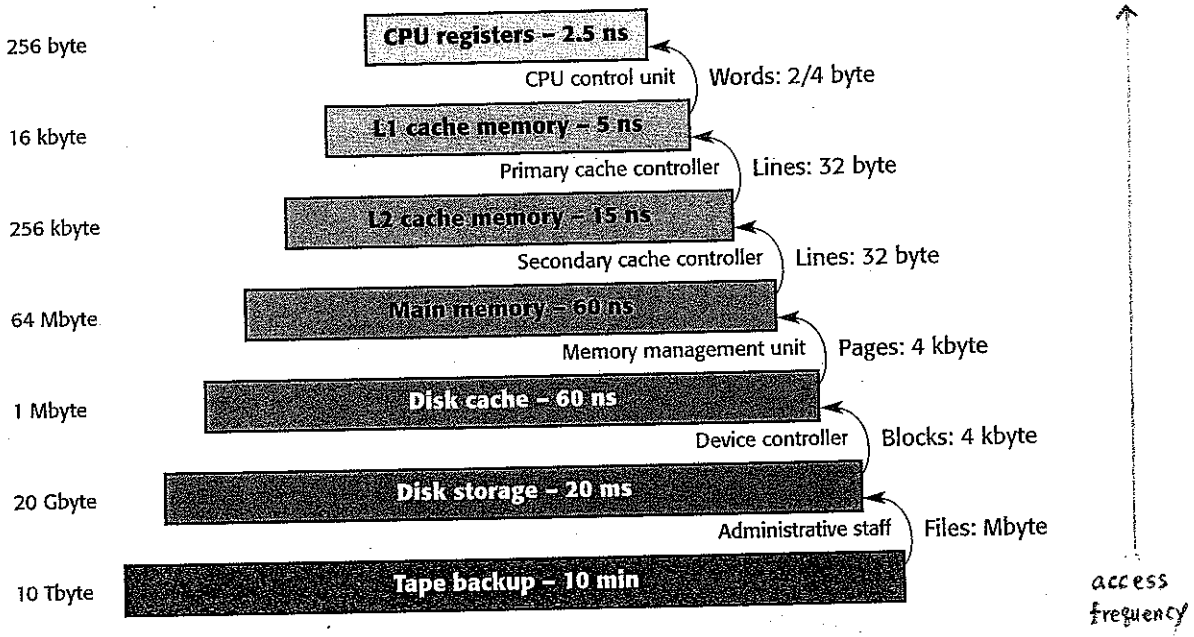
.Physical type
- magnetic core
- semiconductor
- magnetic surface: disk, tape
- GaAs: Cray-3

.Physical characteristics
- volatile / non-volatile
- erasable / non-erasable

Memory Hierarchy

Fast access time
Large capacity } **tradeoff**
Low unit cost

Solution → Memory hierarchy



Example. 2-level memory hierarchy

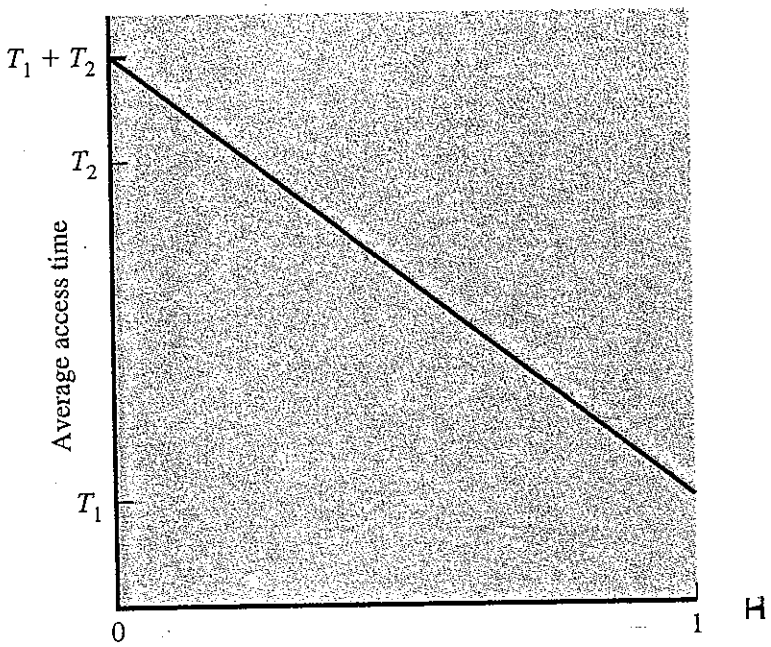
	access time	size	cost
Cache	1	1	10
Main	10	10	1

Hit ratio = 0.9

Effective access time: $T_S = 0.9 \times 1 + (1 - 0.9) \times (10 + 1) \doteq 2$

Average cost per bit:

$$C_S = \frac{10\,(1) + 1\,(10)}{1 + 10} \doteq 2$$

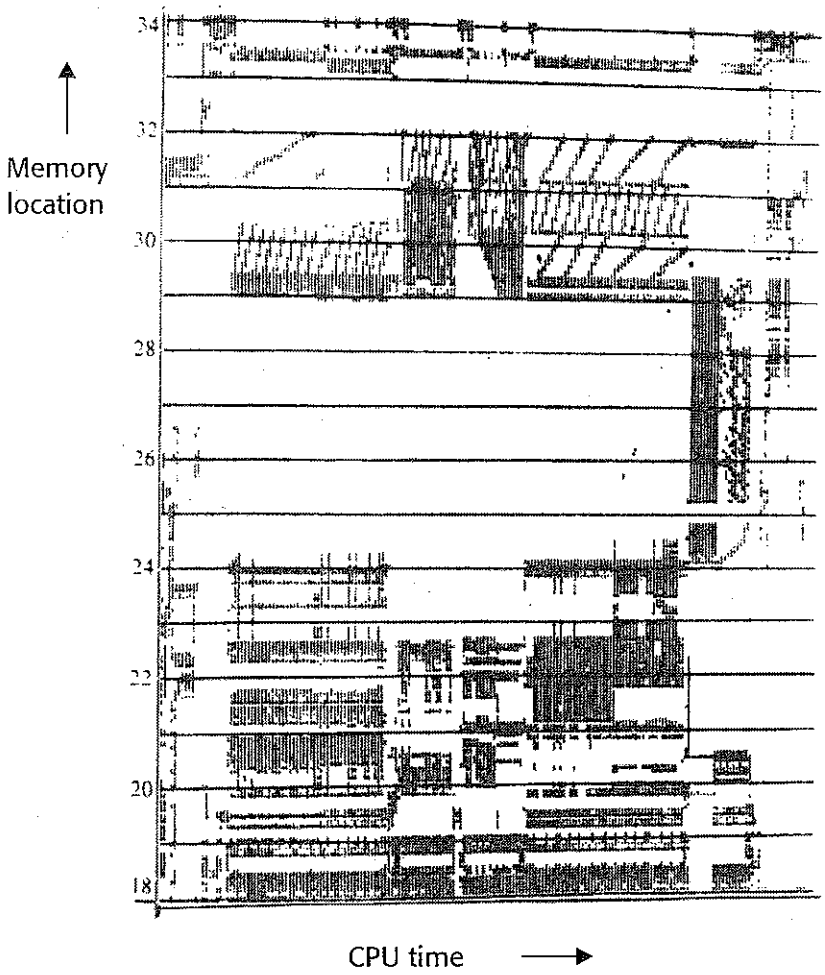


Locality of Reference (4A)

- *Memory references tend to cluster*
- Dennis (1968)

Spatial locality

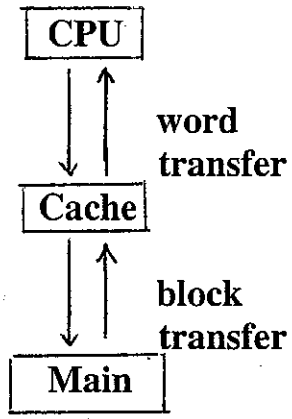
- sequential execution (Sequential locality)
- array: consecutive location

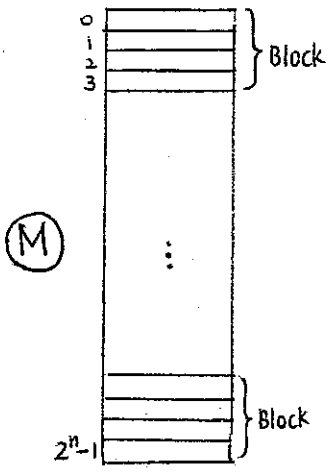


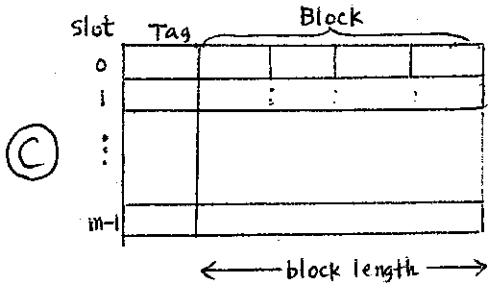
Temporal locality

- tend to access memory location again that have been used recently
- Keep recently used instruction/data in cache

Cache Memory Principles







$C \ll M$

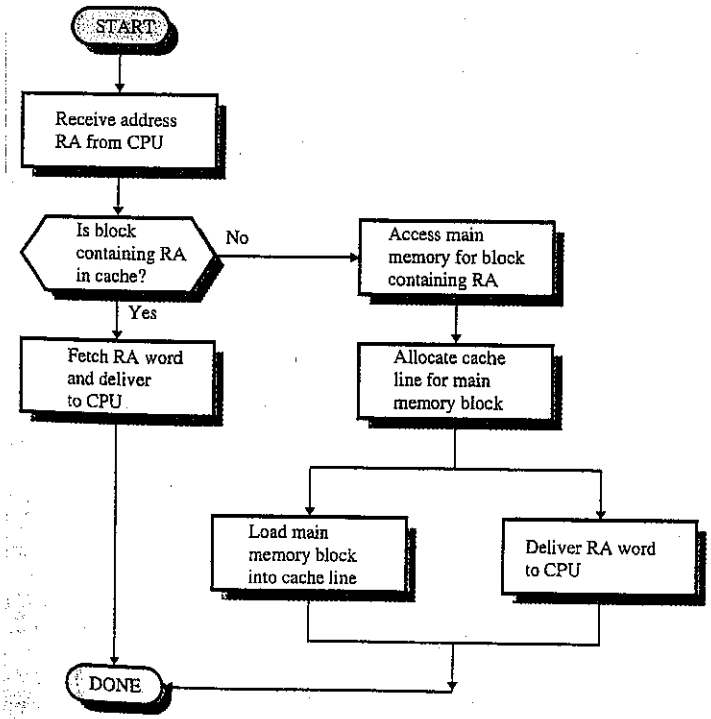


Figure 4.5 Cache Read Operation

Elements of Cache Design

(1) Cache size - tradeoff
1:100 ~ 1:500
1K ~ 512K

(2) Mapping Function
. direct
. fully associative
. set associative

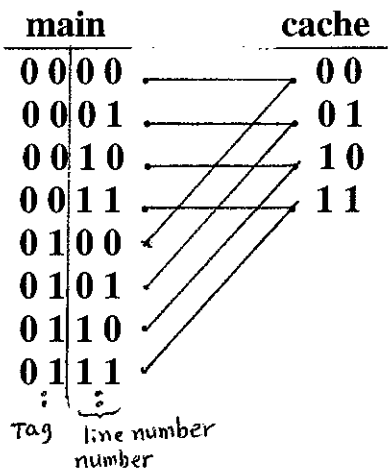
Direct Mapping

$i = j \bmod m$

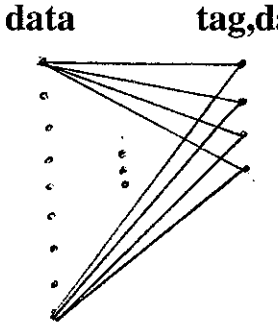
- i: cache line number
- j: main block number
- m: number of lines in the cache

cache line	memory block
0	0, m, 2m, …
1	1, m+1, 2m+1,..
:	: : :
m-1	m-1, 2m-1, …

contention problem



Fully Associative Mapping



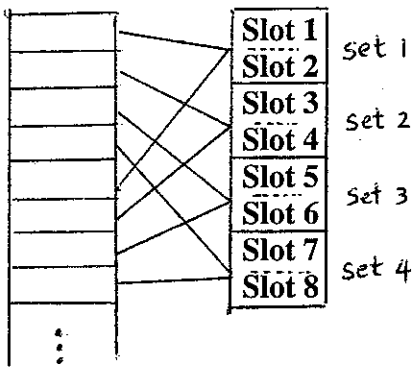
. cache control logic simultaneously examine every line's tag for a match

. each data block can reside any cache line

. complex circuitry

Set Associative Mapping – compromise

Example. 2-way



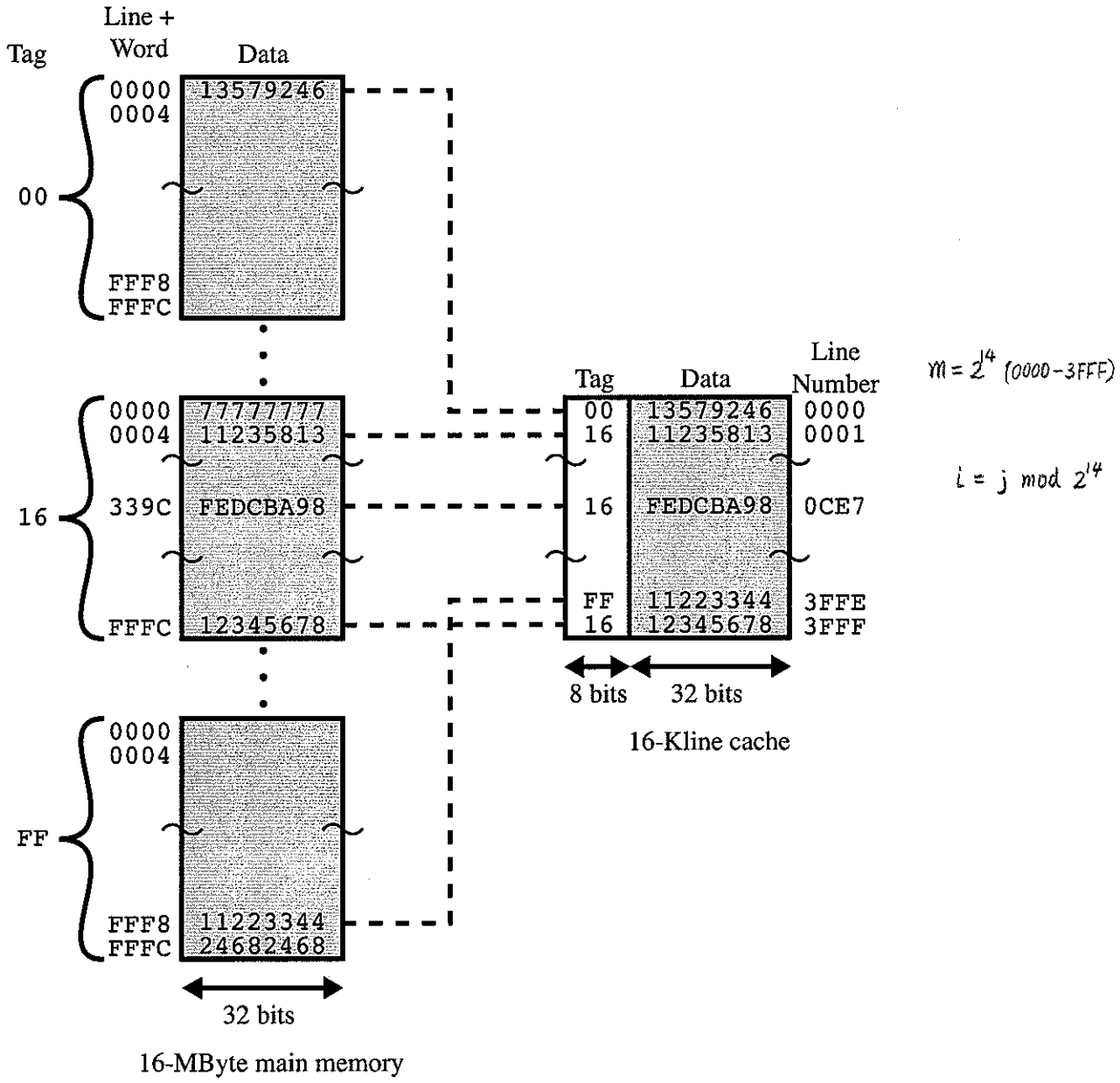


Figure 4.8 Direct Mapping Example

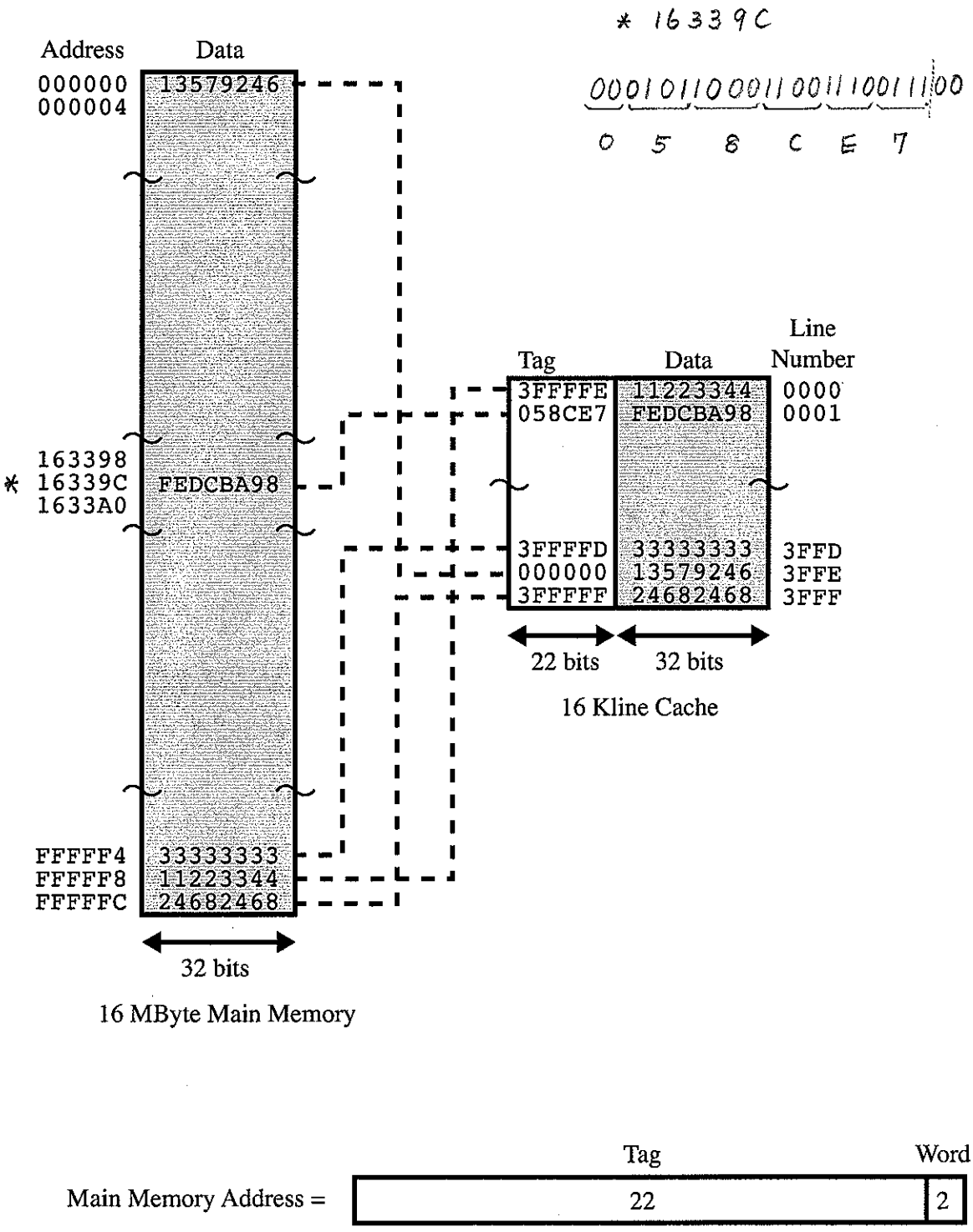


Figure 4.10 Associative Mapping Example

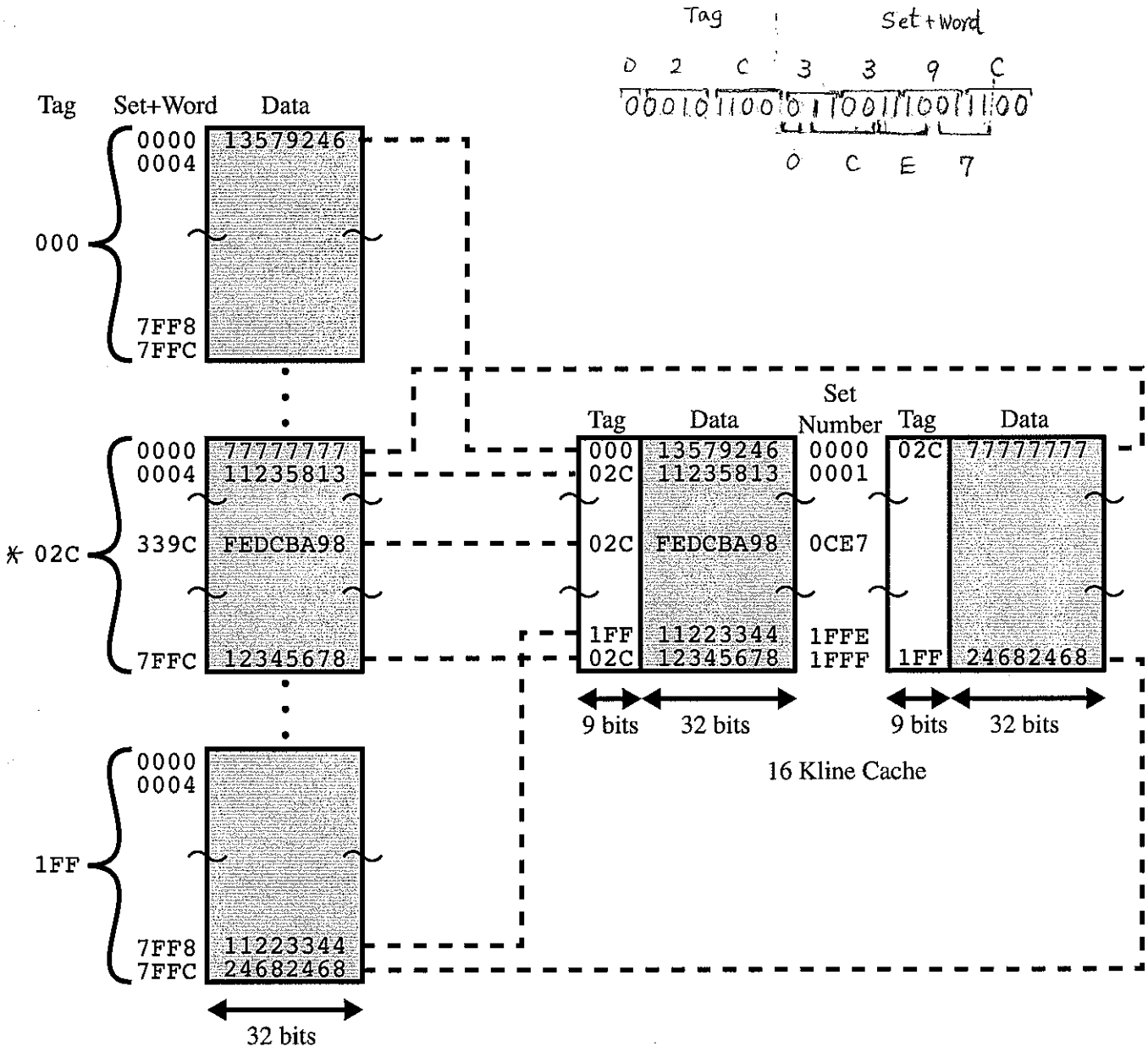


Figure 4.12 Two-Way Set Associative Mapping Example

Replacement Algorithm

- fully associative and set associative
- hardware implementation

LRU - use it (mark '1' whenever the block is used, mark '0' others)
FIFO
LFU - counter
Random

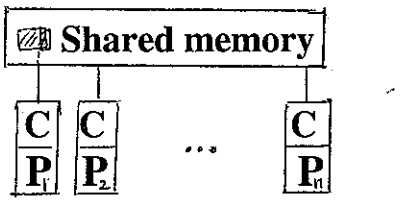
Write Policy

. Write through: All write operations are made to main memory as cache
. Write back: Updates are done on cache (update bit set). When a block is replaced, the block is written back to main memory iff update bit is set.

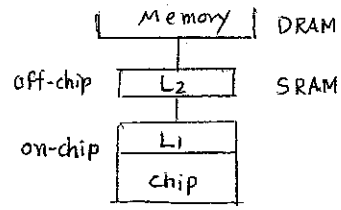
Block size

hit ratio – tradeoff
4 – 8 words/block

Cache Coherency



Single – vs Two-level Cache

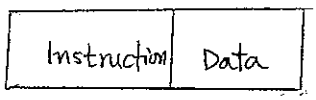


Unified – vs. Split Cache

Power PC

high hit ratio

instruction pipeline



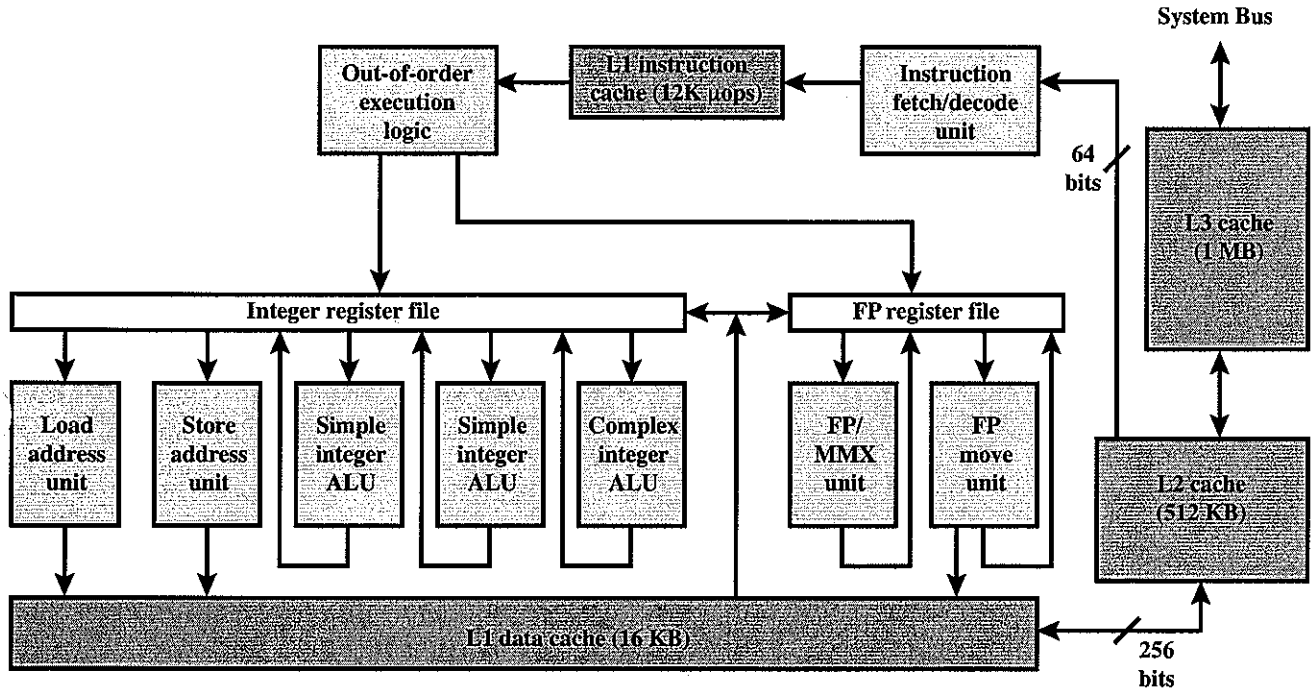


Figure 4.13 Pentium 4 Block Diagram

Problems

4.1 A set associative cache consists of 64 lines, or slots, divided into four-line sets. Main memory contains 4K blocks of 128 words each. Show the format of main memory addresses.

4.2 A two-way set associative cache has lines of 16 bytes and a total size of 8 kbytes. The 64-Mbyte main memory is byte-addressable. Show the format of main memory addresses.

4.8 Consider a machine with a byte addressable main memory of 2^{16} bytes and block size of 8 bytes. Assume that a direct mapped cache consisting of 32 lines is used with this machine.

a. How is a 16-bit memory address divided into tag, line number, and byte number?

b. Into what line would bytes with each of the following addresses be stored?
0001 0001 0001 1011
1100 0011 0011 0100
1101 0000 0001 1101
1010 1010 1010 1010

c. Suppose the byte with address 0001 1010 0001 1010 is stored in the cache. What are the addresses of the other bytes stored along with it?

d. How many total bytes of memory can be stored in the cache?

e. Why is the tag also stored in the cache?

4.19 Consider a memory system with the following parameters:

$$T_c = 100 \text{ ns} \qquad C_c = 10^{-4} \text{ \$/bit}$$
$$T_m = 1{,}200 \text{ ns} \qquad C_m = 10^{-5} \text{ \$/bit}$$

a. What is the cost of 1 MByte of main memory?

b. What is the cost of 1 MByte of main memory using cache memory technology?

c. If the effective access time is 10% greater than the cache access time, what is the hit ratio H?

4.22 A computer has a cache, main memory, and a disk used for virtual memory. If a referenced word is in the cache, 20 ns are required to access it. If it is in main memory but not in the cache, 60 ns are needed to load it into the cache, and then the reference is started again. If the word is not in main memory, 12 ms are required to fetch the word from disk, followed by 60 ns to copy it to the cache, and then the reference is started again. The cache hit ratio is 0.9 and the main memory hit ratio is 0.6. What is the average time in ns required to access a referenced word on this system?