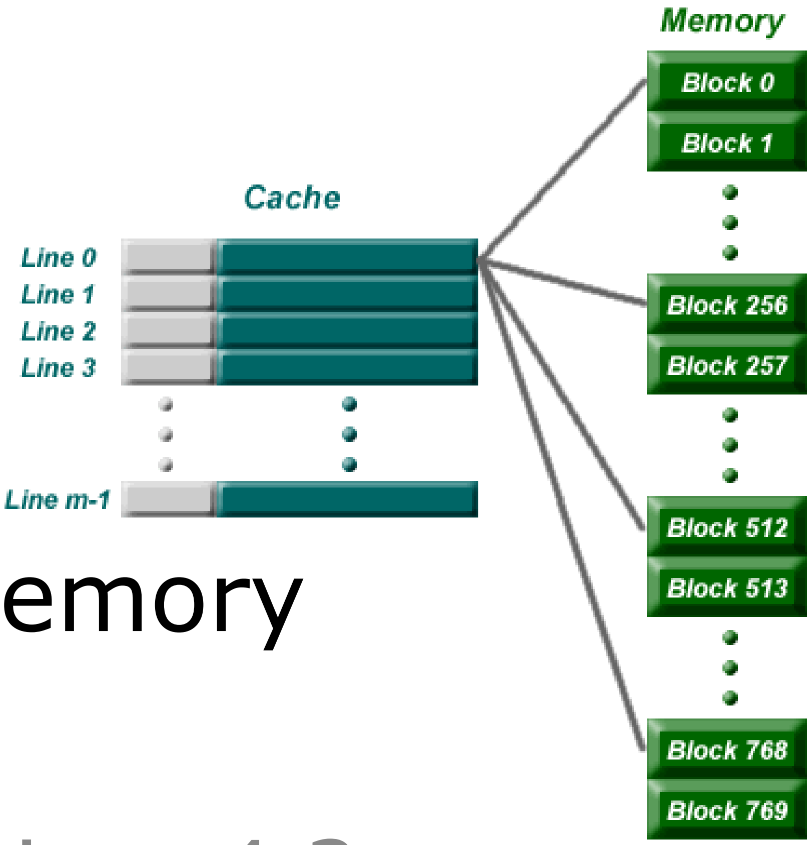
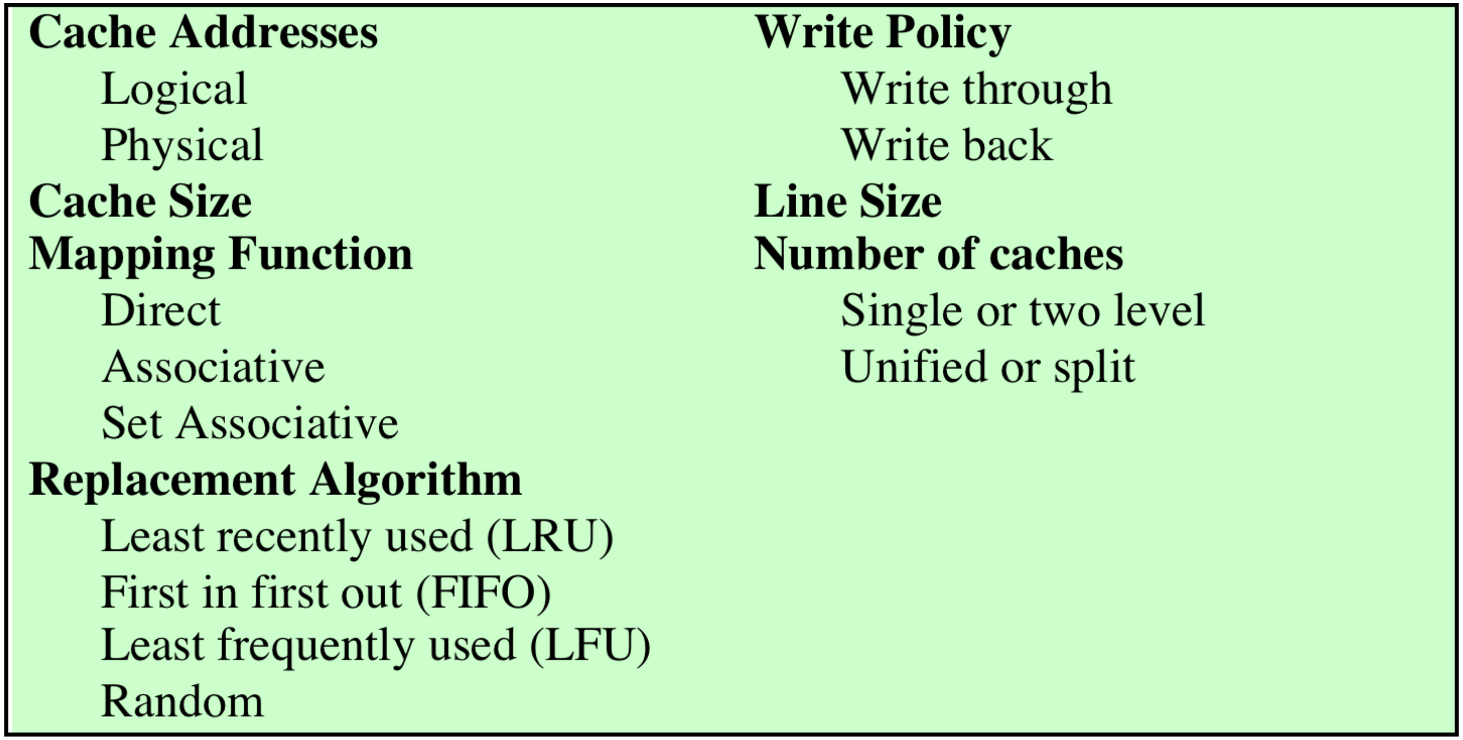


10. Cache Memory

Chapter 4: sections 4.3

Elements of Cache Design



Cache Addresses
- Logical
- Physical

Cache Size

Mapping Function
- Direct
- Associative
- Set Associative

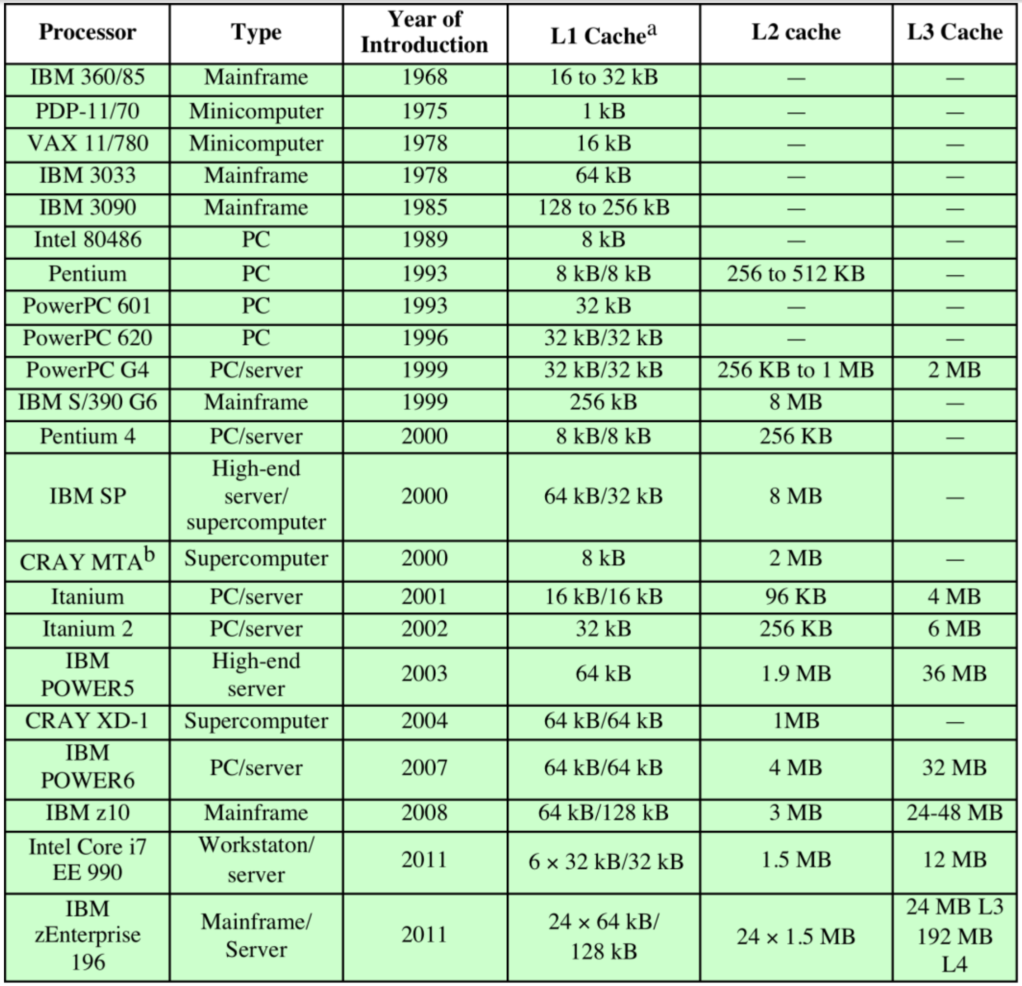
Replacement Algorithm
- Least recently used (LRU)
- First in first out (FIFO)
- Least frequently used (LFU)
- Random

Write Policy
- Write through
- Write back

Line Size

Number of caches
- Single or two level
- Unified or split

Cache Sizes



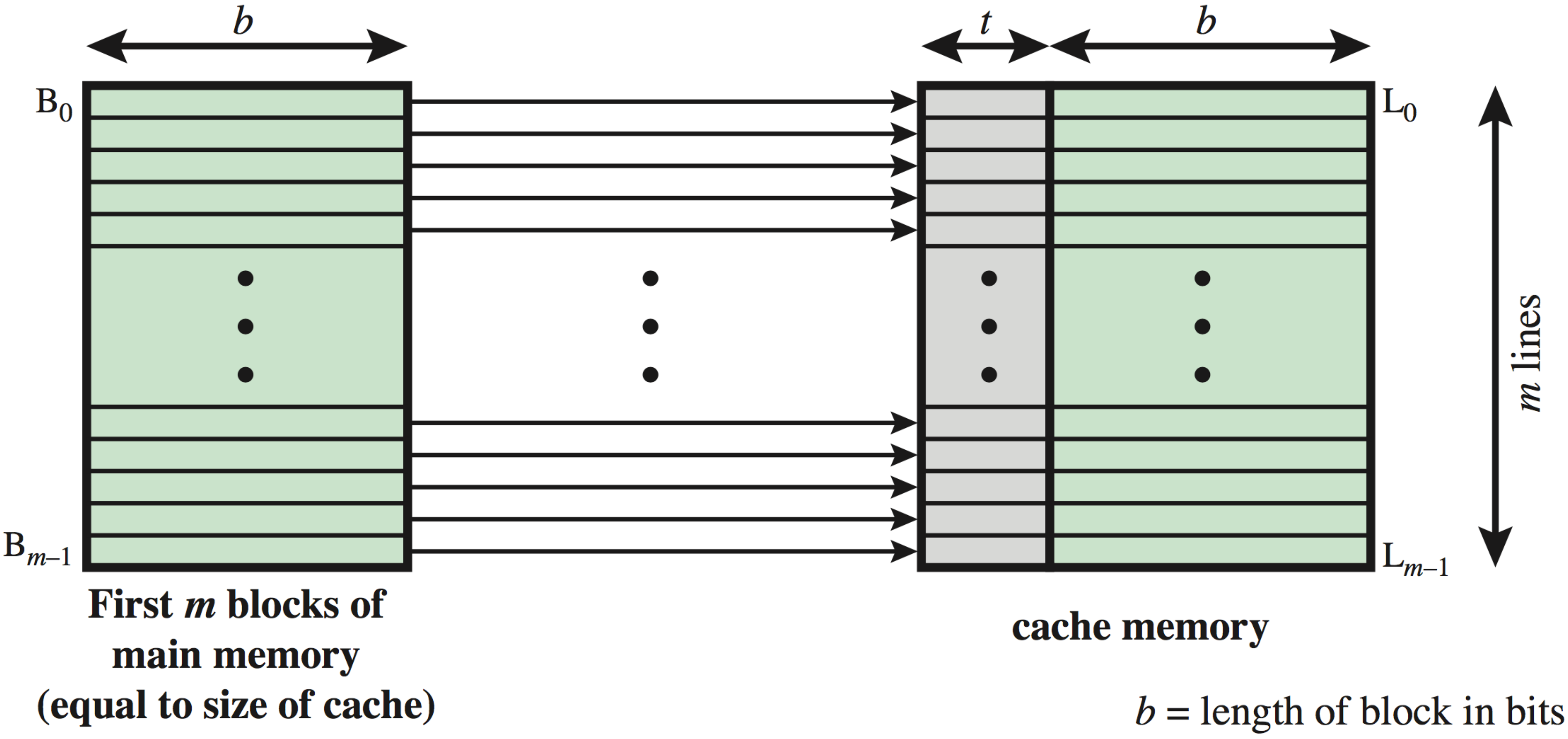
Processor	Type	Year of Introduction	L1 Cache[a]	L2 cache	L3 Cache
IBM 360/85	Mainframe	1968	16 to 32 kB	—	—
PDP-11/70	Minicomputer	1975	1 kB	—	—
VAX 11/780	Minicomputer	1978	16 kB	—	—
IBM 3033	Mainframe	1978	64 kB	—	—
IBM 3090	Mainframe	1985	128 to 256 kB	—	—
Intel 80486	PC	1989	8 kB	—	—
Pentium	PC	1993	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 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
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	—
IBM POWER6	PC/server	2007	64 kB/64 kB	4 MB	32 MB
IBM z10	Mainframe	2008	64 kB/128 kB	3 MB	24-48 MB
Intel Core i7 EE 990	Workstaton/ server	2011	6 × 32 kB/32 kB	1.5 MB	12 MB
IBM zEnterprise 196	Mainframe/ Server	2011	24 × 64 kB/ 128 kB	24 × 1.5 MB	24 MB L3 192 MB L4

Cache Mapping Functions

- Direct Mapping - simplest of the cache mapping schemes

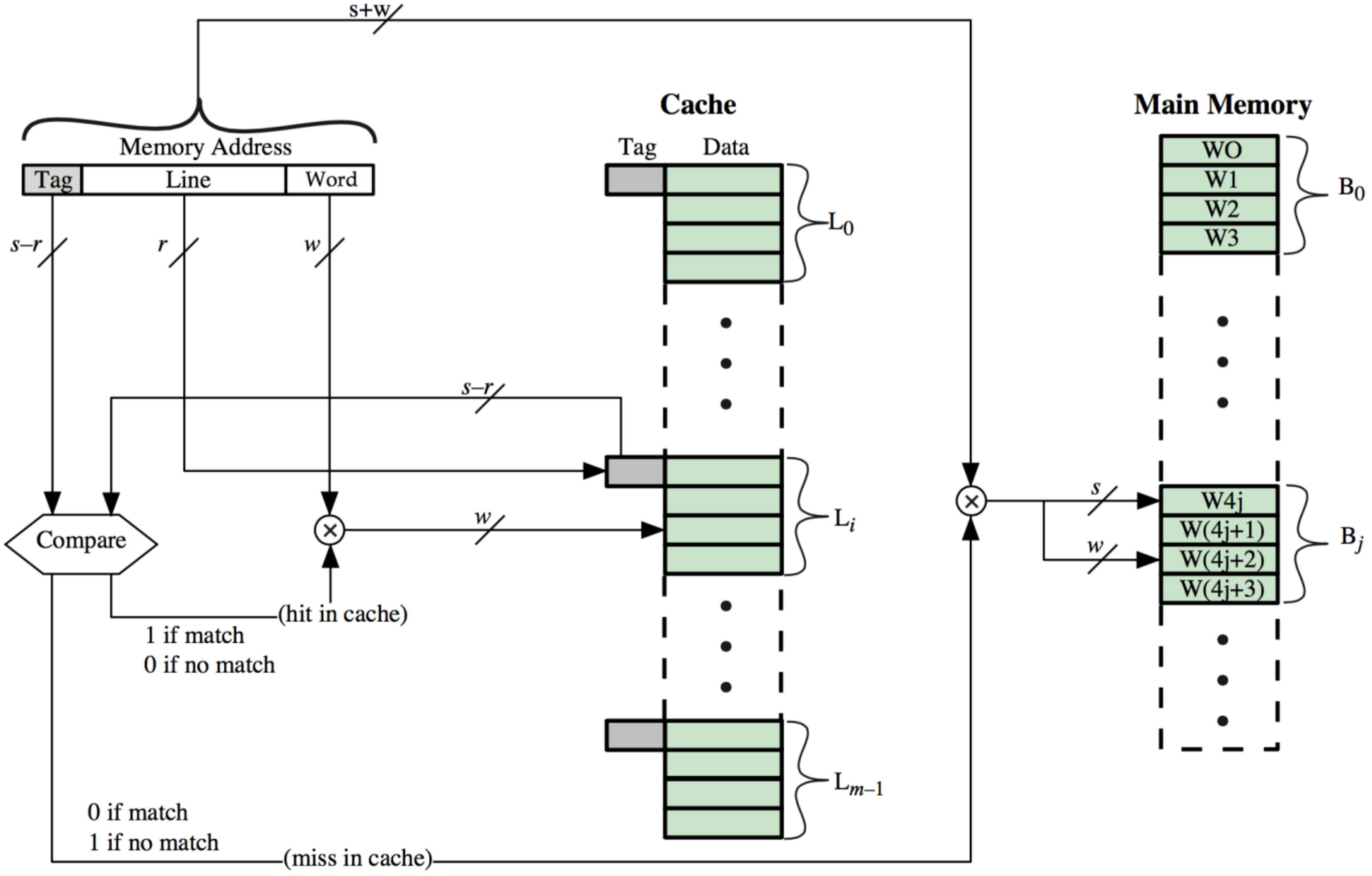
$$i = j \; modulo \; m \; where$$
$$i = cache \; line \; number$$
$$j = main \; memory \; block \; number$$
$$m = number \; of \; lines \; in \; the \; cache$$

Direct Mapping



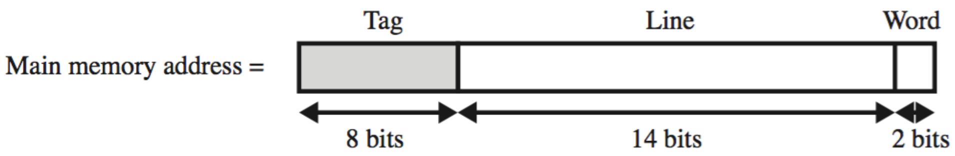
(a) Direct mapping

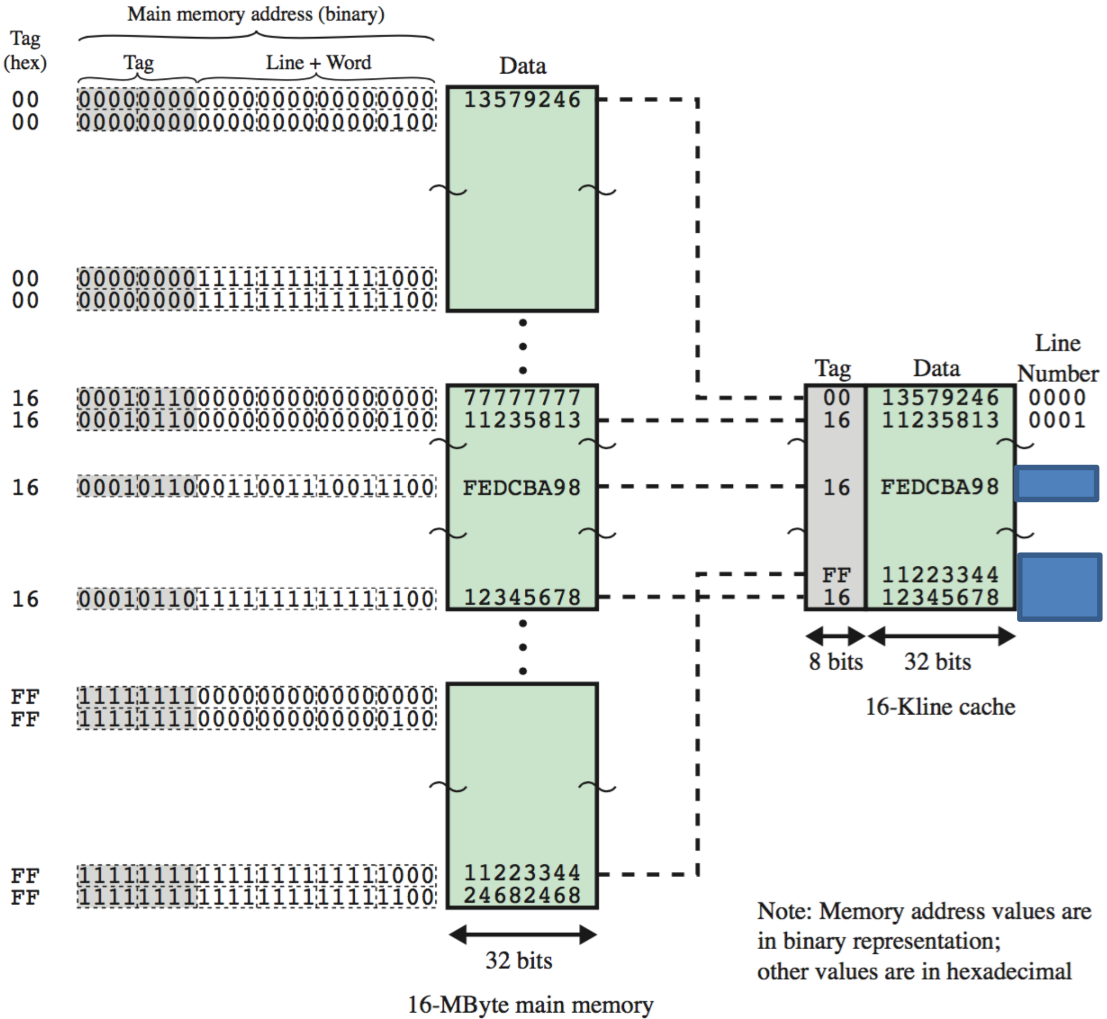
Direct Mapping



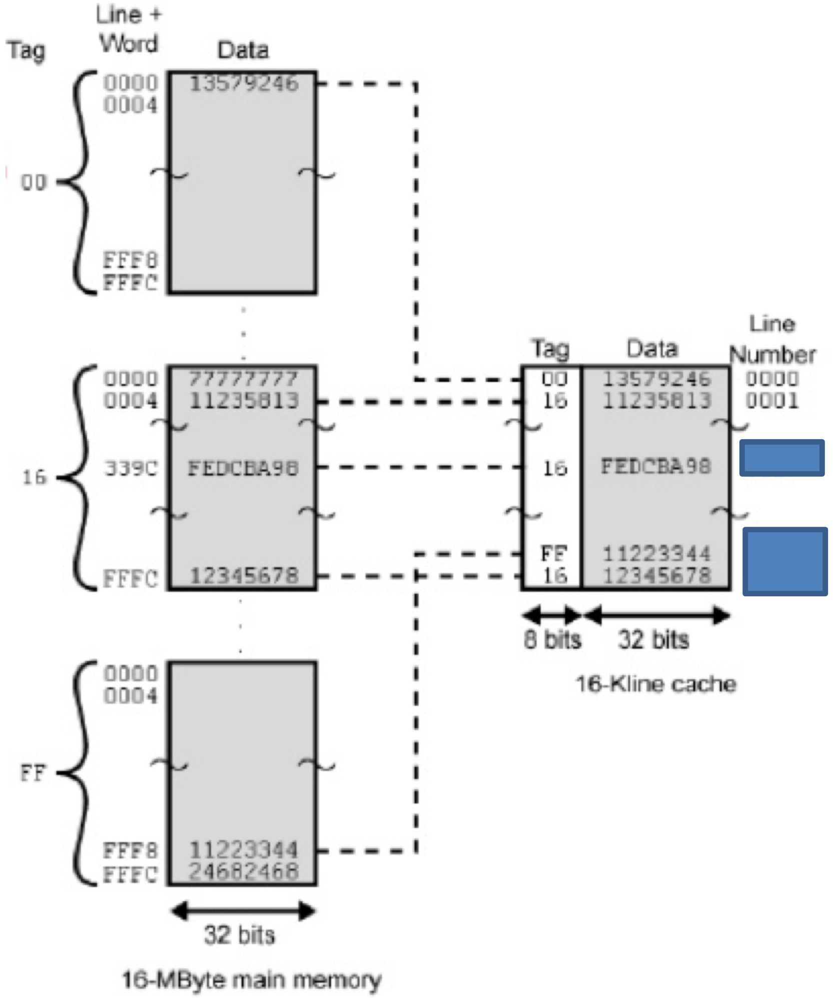
Direct Mapping

- To summarize:
  - Address Length is (s + w) bits
  - Number of addressable units is 2^{s+w} bytes
  - Block size = line size = 2w bytes
  - Number of blocks in main memory is = 2^{s}
  - Number of cache lines is 2^{r}
  - Tag size is (s - r) bits


Main memory address =
Tag
Line
Word
8 bits
14 bits
2 bits


Main memory address (binary)
Tag (hex)
Tag
Line + Word
Data
00 000000000000000000000000 13579246
00 000000000000000000000100
00 000000001111111111111000
00 000000001111111111111100
16 000101100000000000000000 77777777
16 000101100000000000000100 11235813
16 000101100011001110011100 FEDCBA98
16 000101101111111111111100 12345678
FF 111111110000000000000000
FF 111111110000000000000100
FF 111111111111111111111000 11223344
FF 111111111111111111111100 24682468
32 bits
16-MByte main memory
Tag
Data
Line Number
00 13579246 0000
16 11235813 0001
16 FEDCBA98
FF 11223344
16 12345678
8 bits
32 bits
16-Kline cache
Note: Memory address values are in binary representation; other values are in hexadecimal

Direct Mapping



Direct Mapping Summary

- Advantages of direct mapping:
  - The technique is simple
  - The mapping scheme is easy to implement
- Disadvantage of direct mapping:
  - Each block of main memory maps to a fixed location in the cache which could lead to thrashing