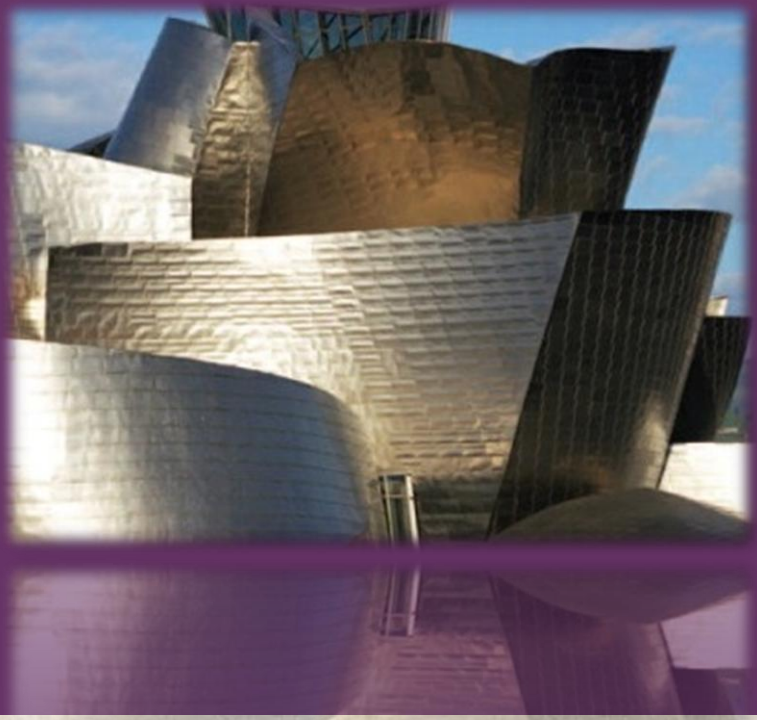
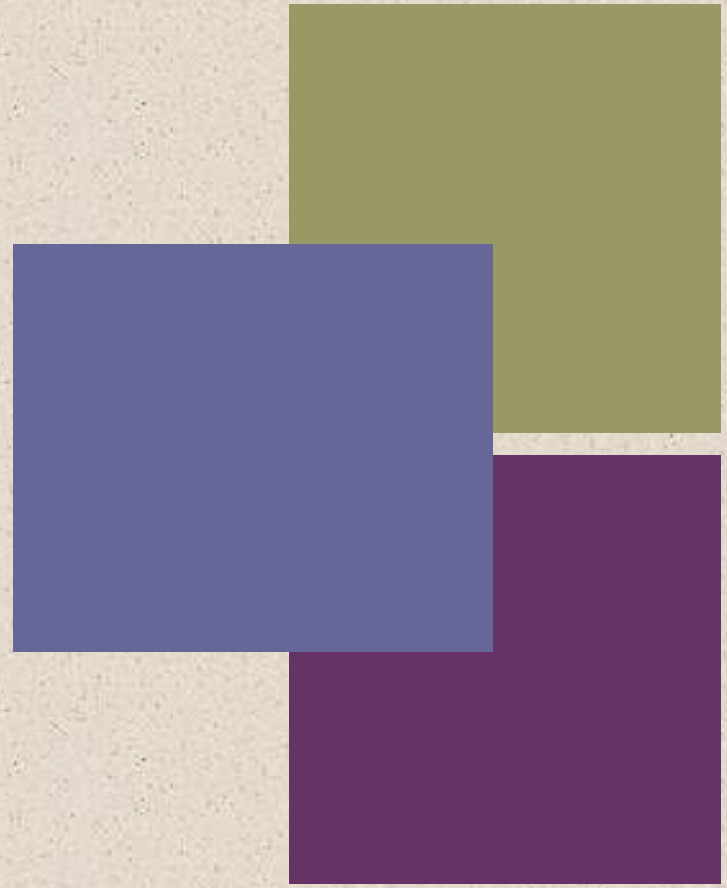


William Stallings Computer Organization and Architecture 9th Edition



Chapter 4

Cache Memory

Direct Mapping Example

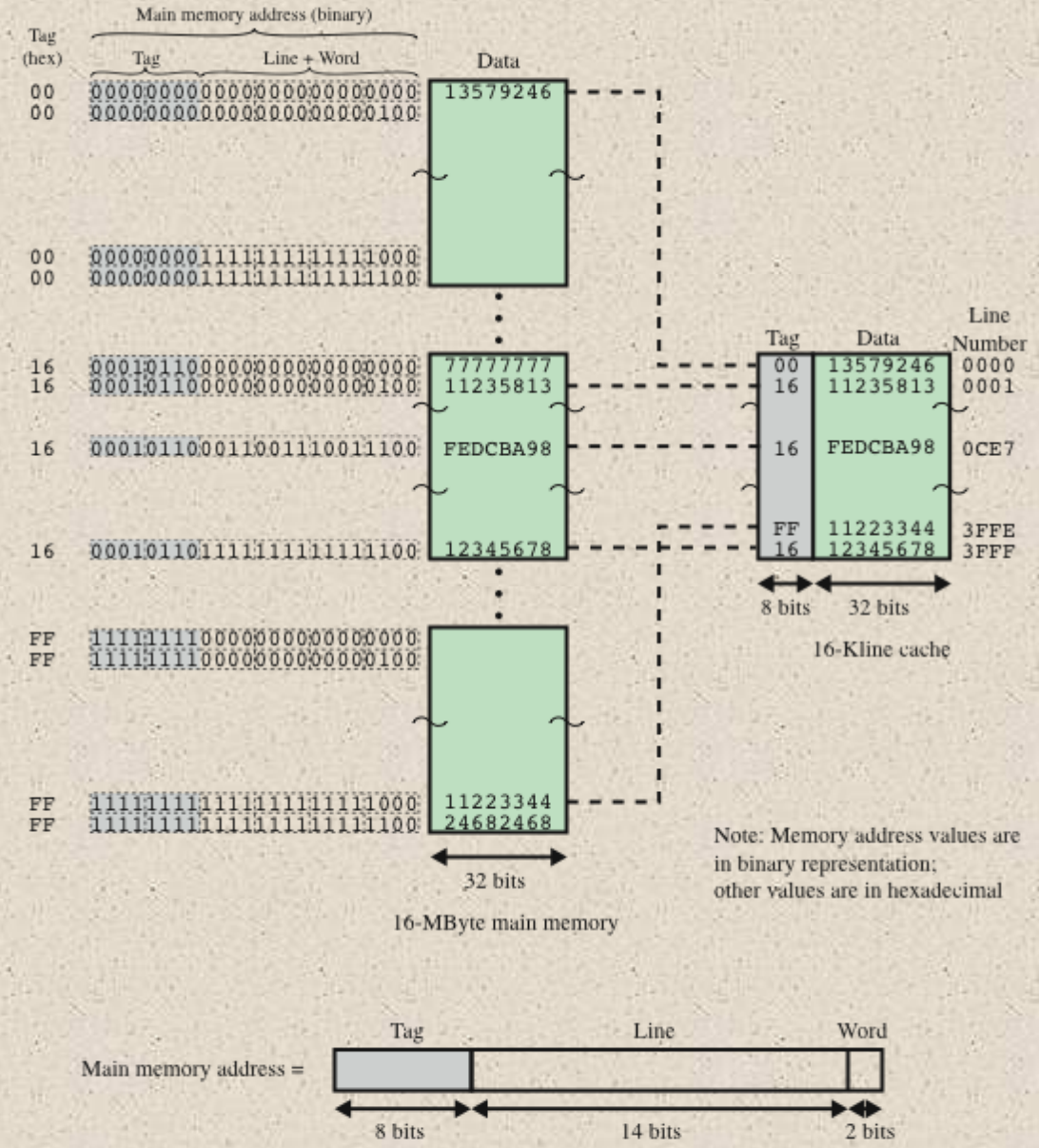
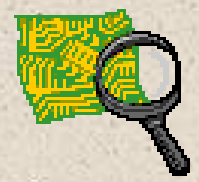


Figure 4.10 Direct Mapping Example

Direct Mapping Summary

- Address length = (s + w) bits
- Number of addressable units = 2^{s+w} words or bytes
- Block size = line size = 2^{w} words or bytes
- Number of blocks in main memory = $2^{s+w}/2^{w} = 2^{s}$
- Number of lines in cache = m = 2^{r}
- Size of tag = (s – r) bits



Advantage and Disadvantage

- Advantage:
  - Simple.
  - Inexpensive.

- Disadvantage:
  - There is a fixed cache location for any given block.
  - Thrashing: if a program happens to reference words repeatedly from two different blocks that map into the same line, then the blocks will be continually swapped in the cache and the hit ratio will be low.

Victim Cache

- Originally proposed as an approach to reduce the conflict misses of direct mapped caches without affecting its fast access time
- Fully associative cache
- Typical size is 4 to 16 cache lines
- Residing between direct mapped L1 cache and the next level of memory

Fully Associative Cache Organization

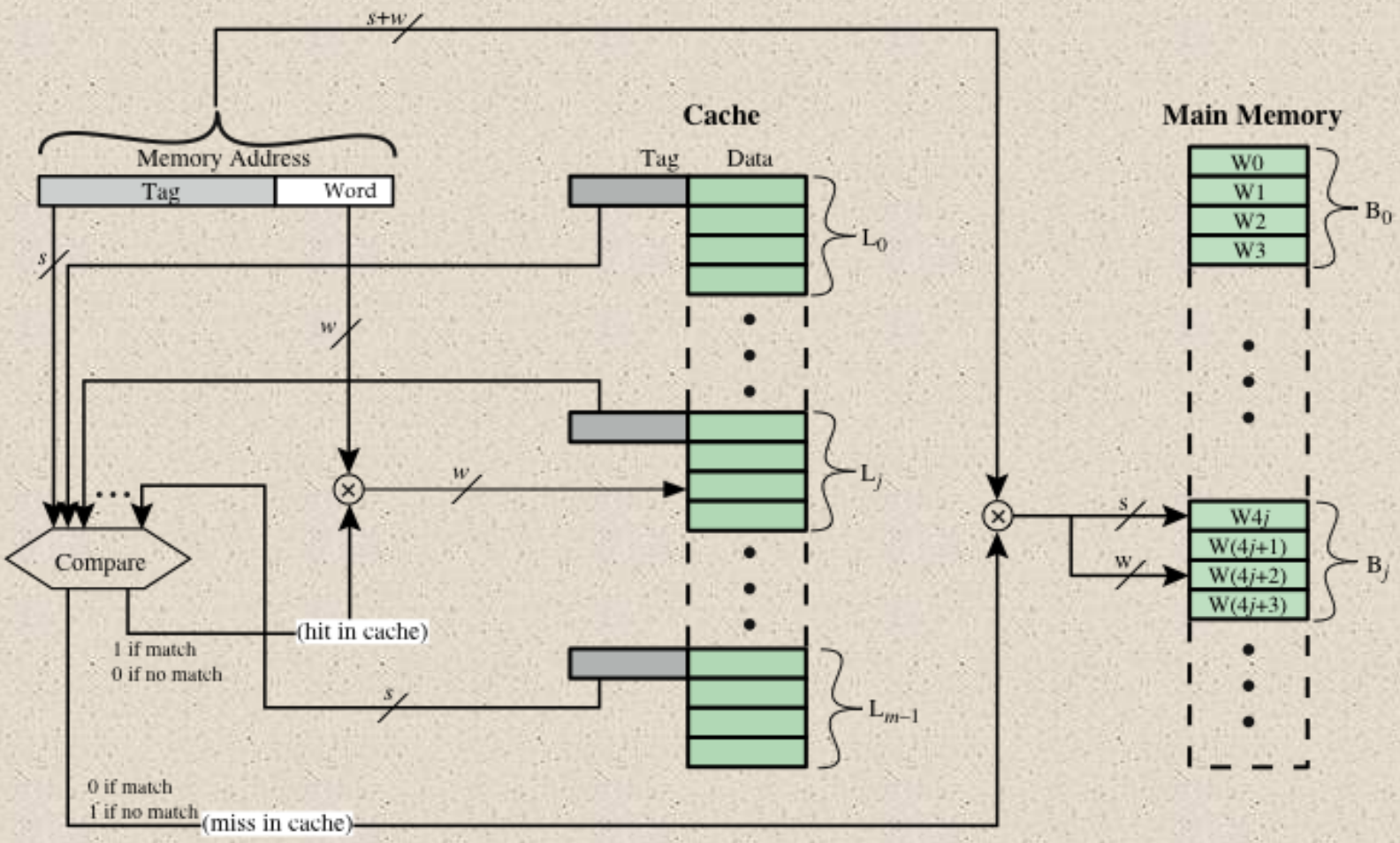


Figure 4.11 Fully Associative Cache Organization

Associative Mapping Example

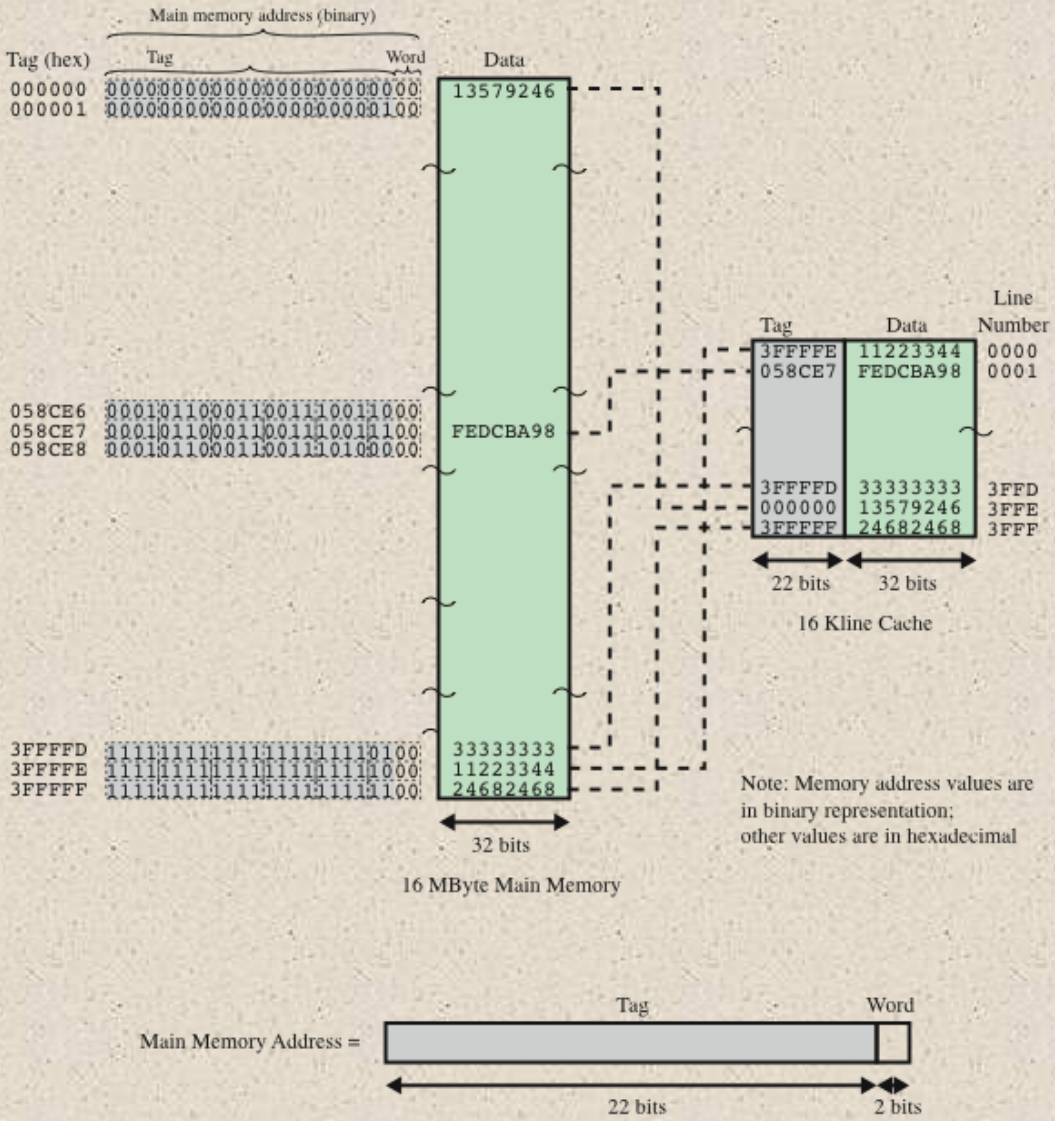
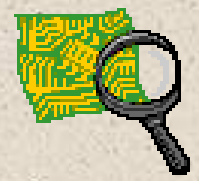


Figure 4.12 Associative Mapping Example

Associative Mapping Summary

- Address length = (s + w) bits
- Number of addressable units = 2^{s+w} words or bytes
- Block size = line size = 2^{w} words or bytes
- Number of blocks in main memory = $2^{s+w}/2^{w} = 2^{s}$
- Number of lines in cache = undetermined
- Size of tag = s bits



Set Associative Mapping

- Compromise that exhibits the strengths of both the direct and associative approaches while reducing their disadvantages
- Cache consists of a number of sets
- Each set contains a number of lines
- A given block maps to any line in a given set
- e.g. 2 lines per set
  - 2 way associative mapping
  - A given block can be in one of 2 lines in only one set

Mapping From Main Memory to Cache:

k-Way Set Associative

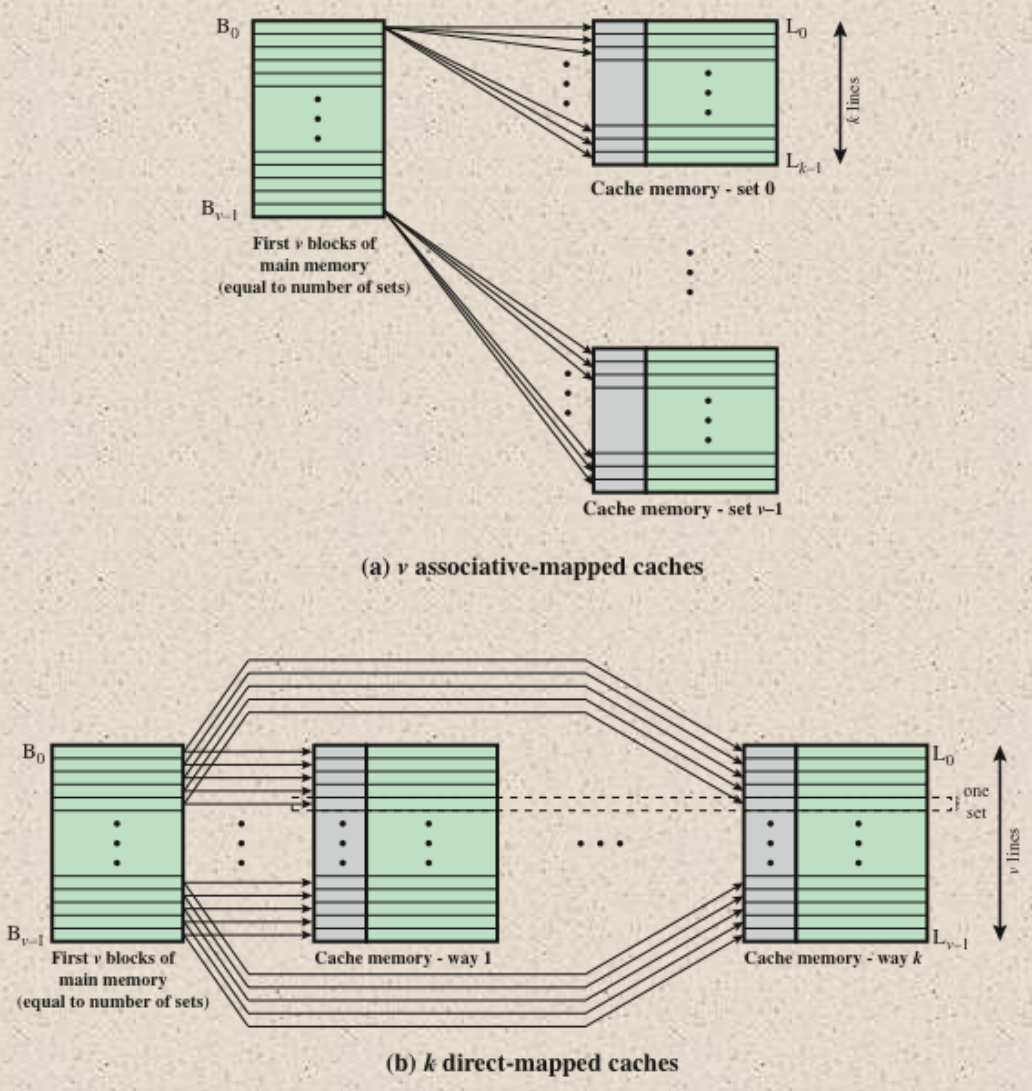


Figure 4.13 Mapping From Main Memory to Cache: *k*-way Set Associative

k-Way Set Associative Cache Organization

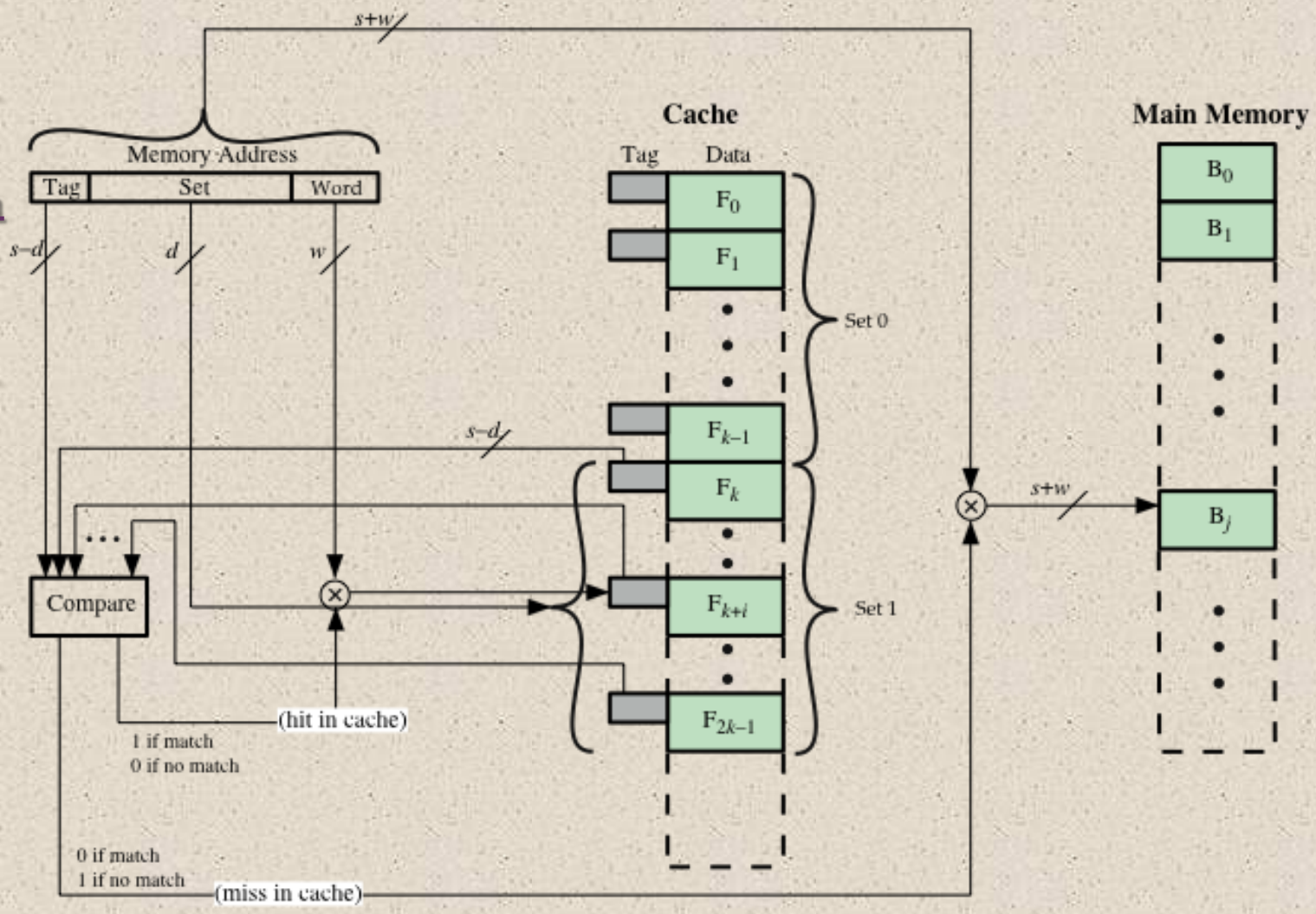
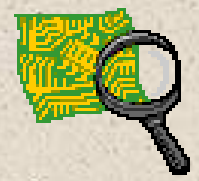


Figure 4.14 *k*-Way Set Associative Cache Organization

Set Associative Mapping Summary

- Address length = (s + w) bits
- Number of addressable units = 2^{s+w} words or bytes
- Block size = line size = 2^{w} words or bytes
- Number of blocks in main memory = $2^{s+w}/2^{w}=2^{s}$
- Number of lines in set = k
- Number of sets = v = 2^{d}
- Number of lines in cache = m=kv = k * 2^{d}
- Size of cache = $k * 2^{d+w}$ words or bytes
- Size of tag = (s – d) bits



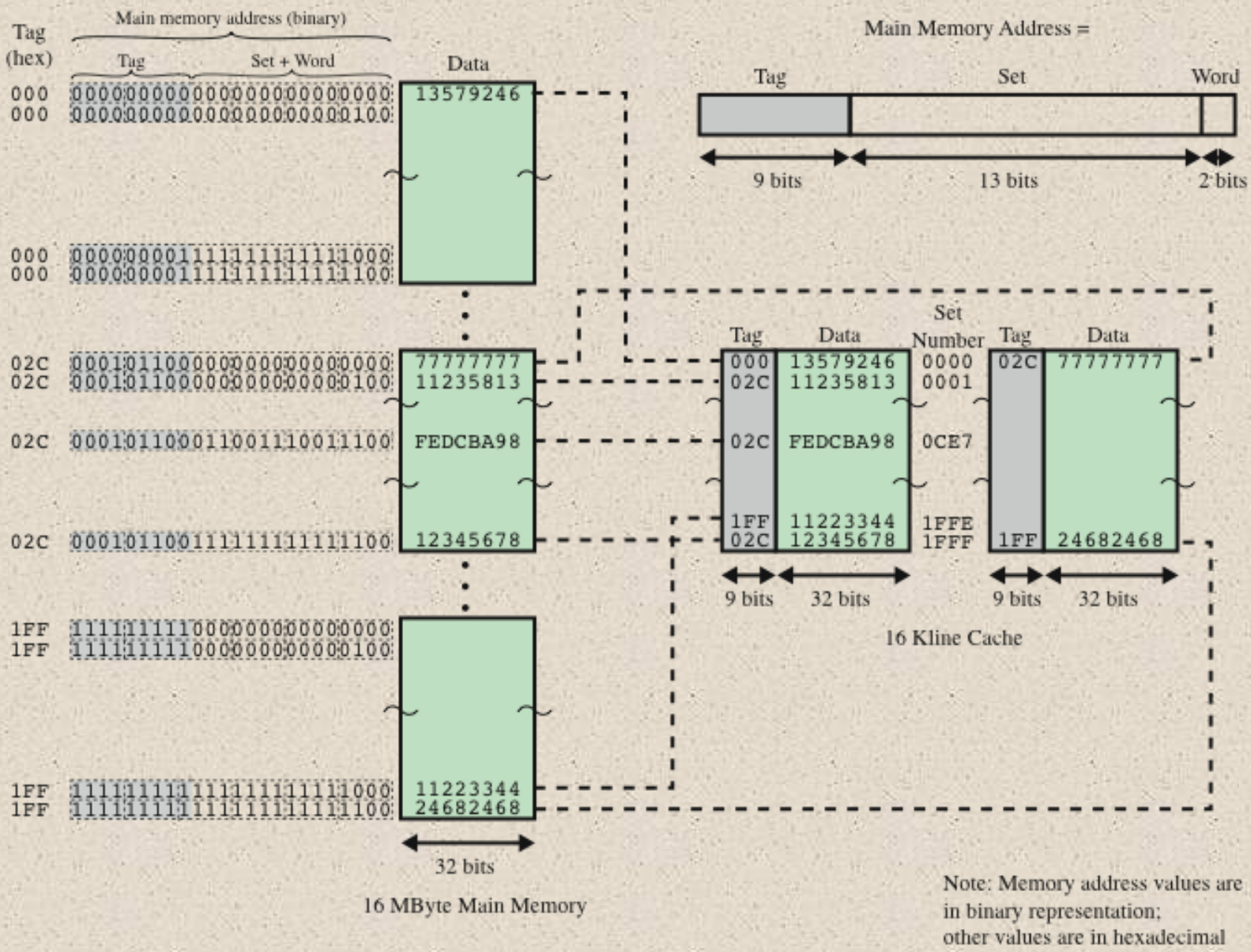


Figure 4.15 Two-Way Set Associative Mapping Example

Varying Associativity Over Cache Size

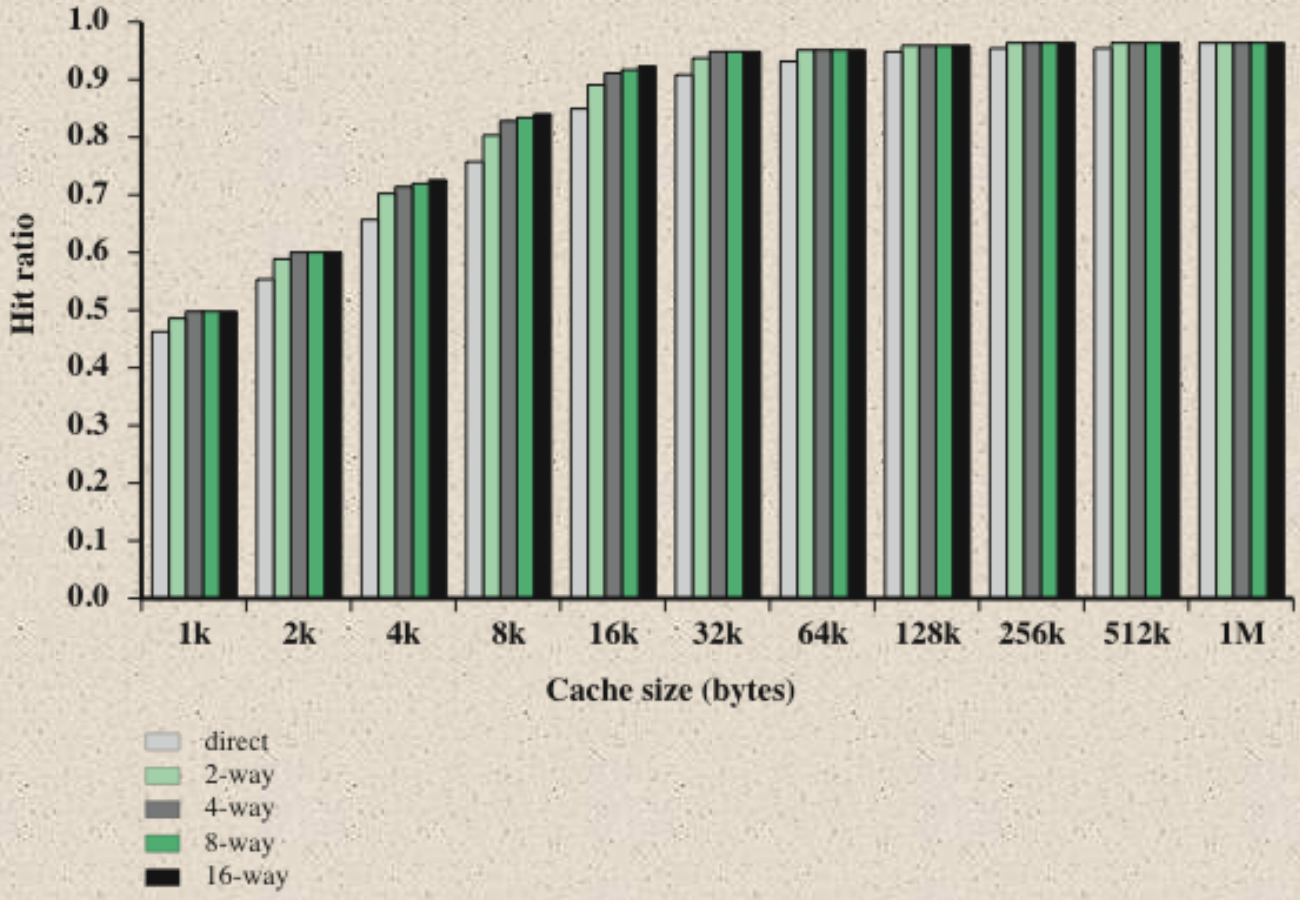


Figure 4.16 Varying Associativity over Cache Size