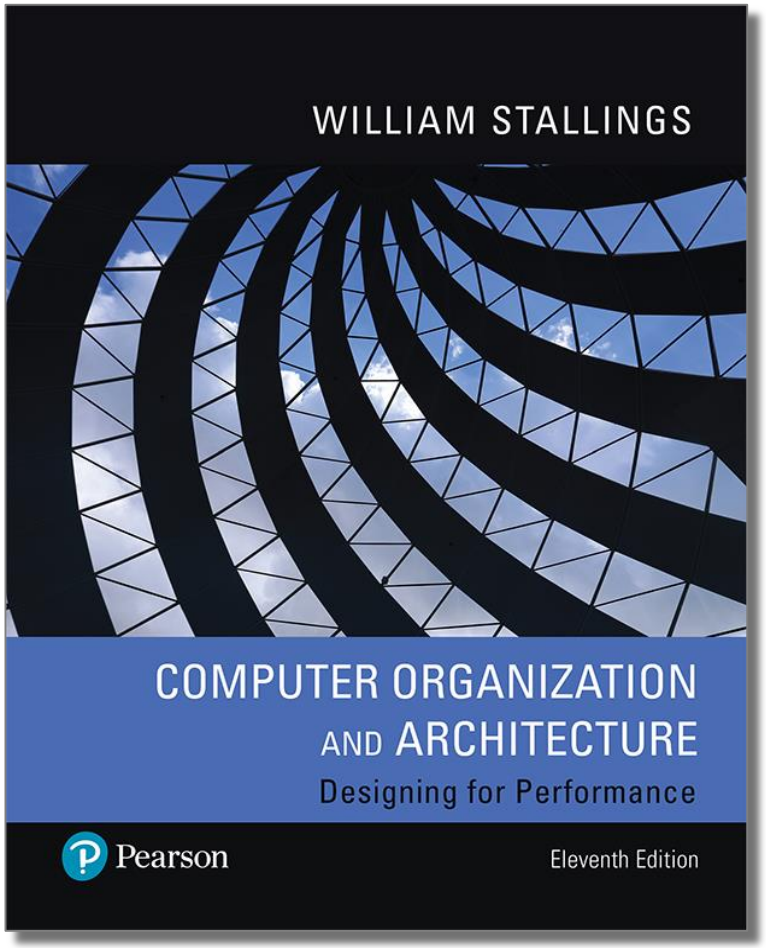
Computer Organization and Architecture

Designing for Performance

11th Edition



Chapter 5

Cache Memory

Table 5.1
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

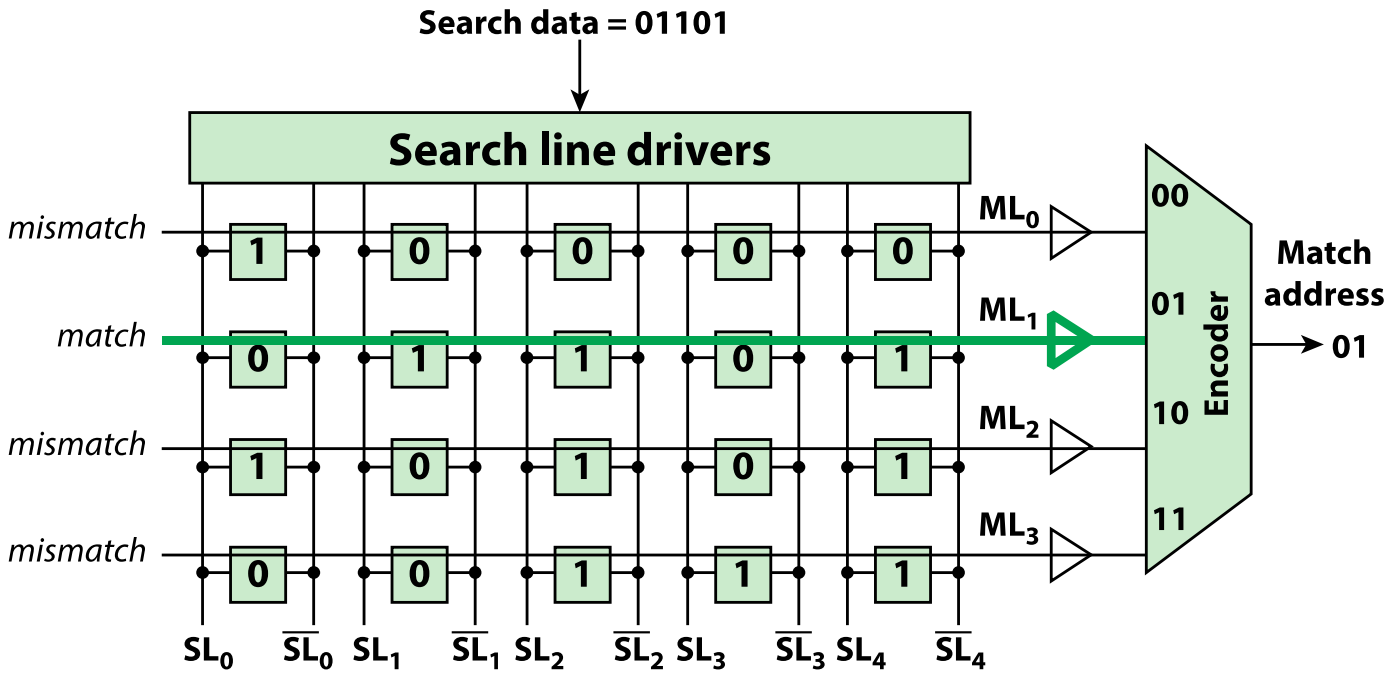
Line Size

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

Content-Addressable Memory (CAM)

- Also known as associative storage
- Content-addressable memory is constructed of static RAM (SRAM) cells but is considerably more expensive and holds much less data than regular SRAM chips
- A CAM with the same data capacity as a regular SRAM is about 60% larger
- A CAM is designed such that when a bit string is supplied, the CAM searches its entire memory in parallel for a match
  - If the content is found, the CAM returns the address where the match is found and, in some architectures, also returns the associated data word
  - This process takes only one clock cycle

Figure 5.9
Content-Addressable Memory

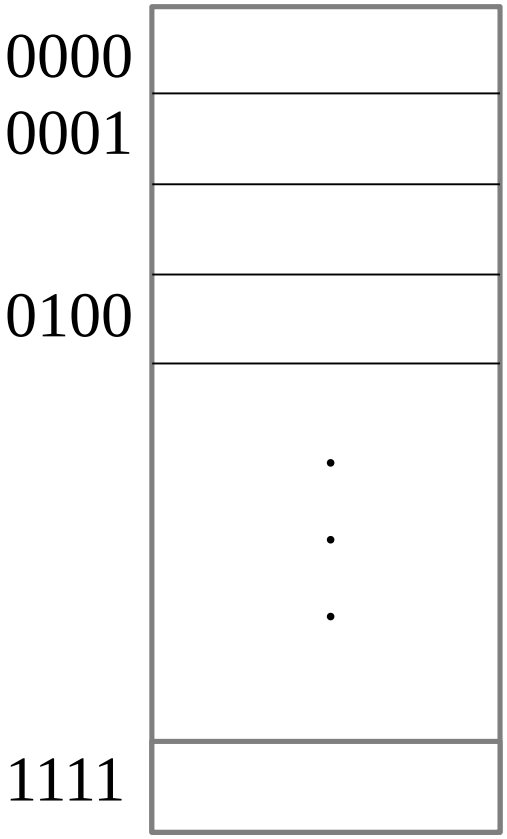


(a) Simplified CAM circuitry

Associative Mapping

Main memory (128 bytes)

Needs 7-bits address

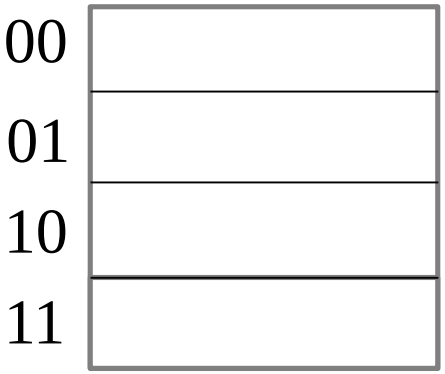


Block Size = 8 bytes

MBN = 128 / 8 = 16 (2^4)

CBN = 32 / 8 = 4 (2^2)

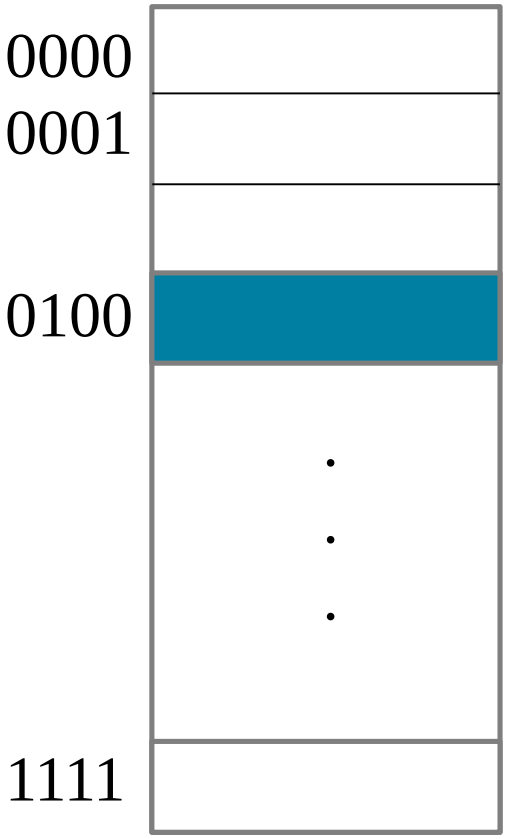
Cache memory (32 bytes)



Associative Mapping

Main memory (128 bytes)

Needs 7-bits address

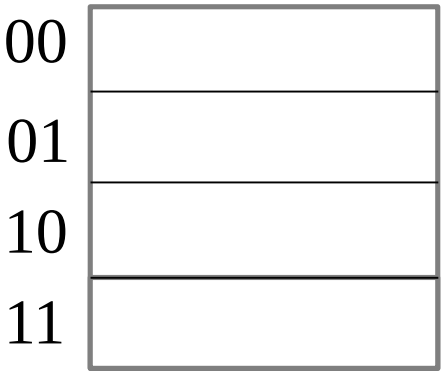


Block Size = 8 bytes

MBN = 128 / 8 = 16 (2^4)

CBN = 32 / 8 = 4 (2^2)

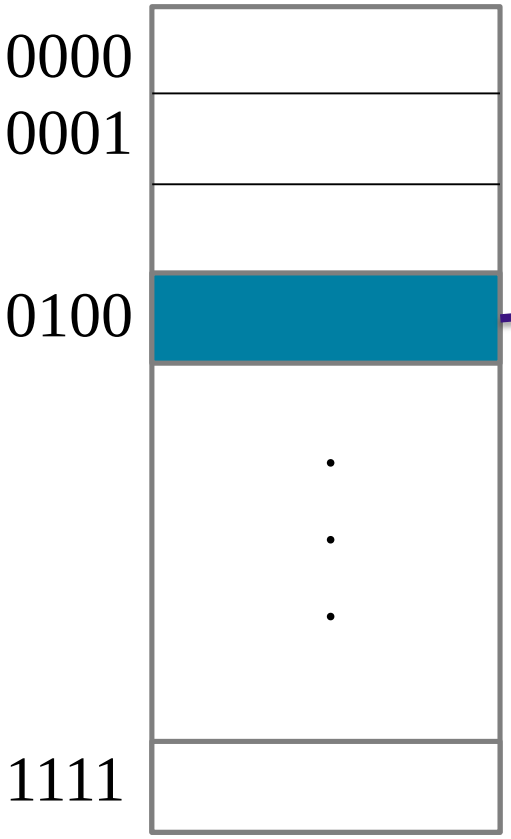
Cache memory (32 bytes)



Associative Mapping

Main memory (128 bytes)

Needs 7-bits address

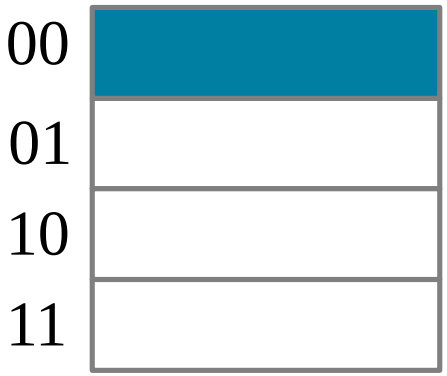


Block Size = 8 bytes

MBN = 128 / 8 = 16 (2^4)

CBN = 32 / 8 = 4 (2^2)

Cache memory (32 bytes)

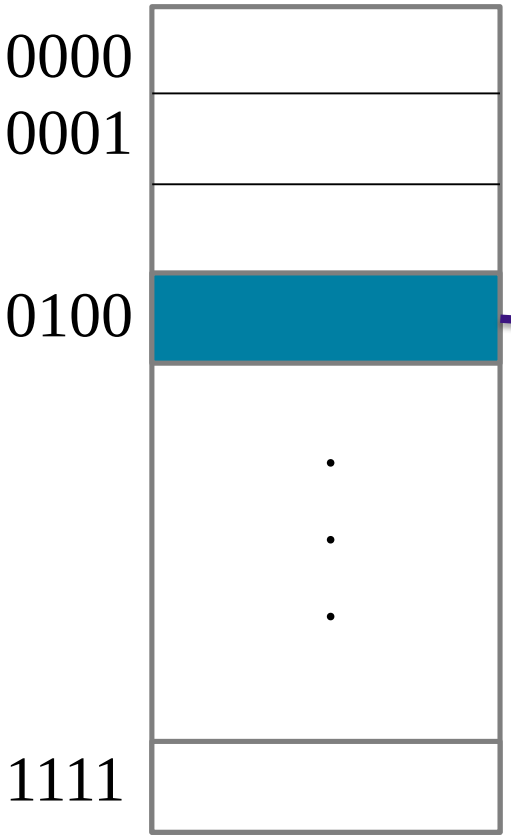


Any main memory block can be stored anywhere!

Associative Mapping

Main memory (128 bytes)

Needs 7-bits address



Block Size = 8 bytes

MBN = 128 / 8 = 16 (2^4)

CBN = 32 / 8 = 4 (2^2)

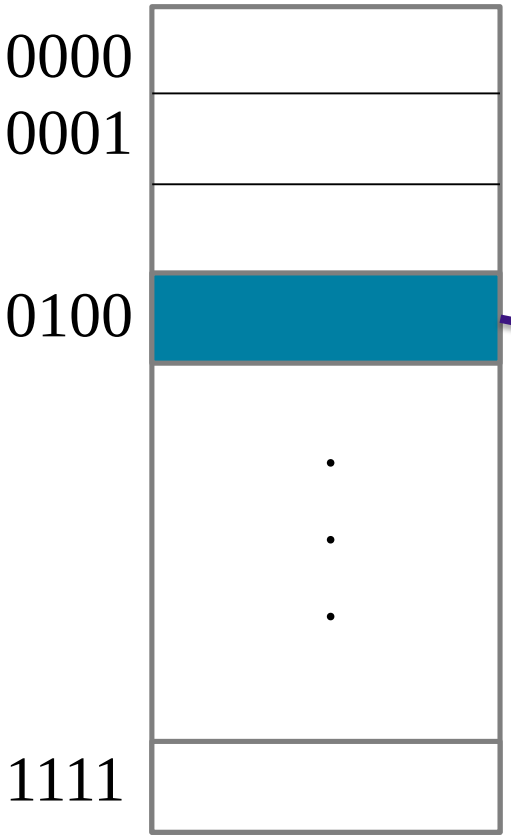
Cache memory (32 bytes)

Any main memory block can be stored anywhere!

Associative Mapping

Main memory (128 bytes)

Needs 7-bits address



Block Size = 8 bytes

MBN = 128 / 8 = 16 (2^4)

CBN = 32 / 8 = 4 (2^2)

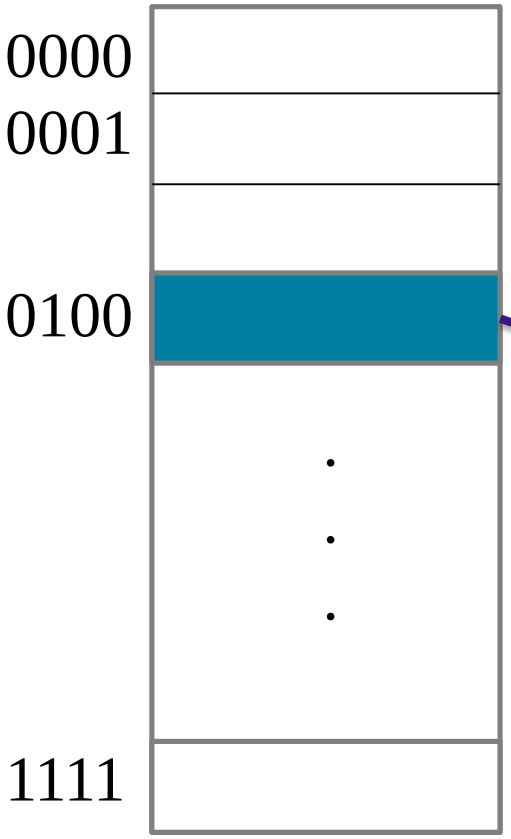
Cache memory (32 bytes)

Any main memory block can be stored anywhere!

Associative Mapping

Main memory (128 bytes)

Needs 7-bits address



Cache memory (32 bytes)

Block Size = 8 bytes

MBN = 128 / 8 = 16 (2^4)

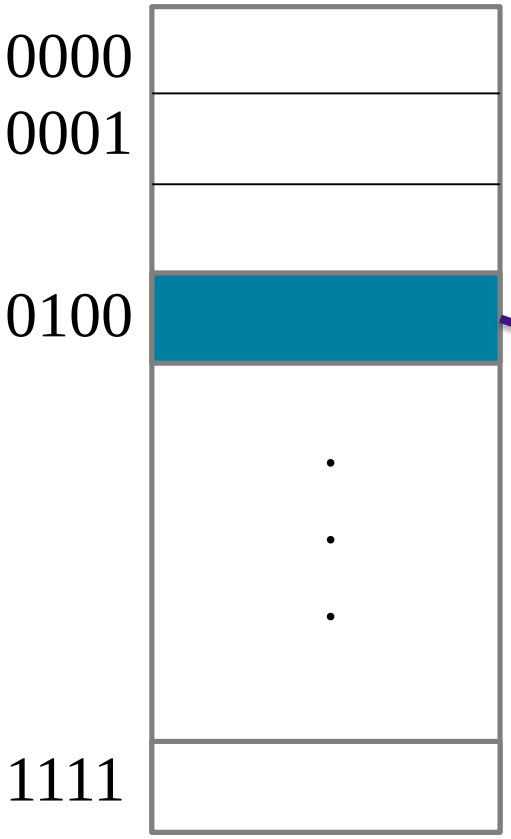
CBN = 32 / 8 = 4 (2^2)

Any main memory block can be stored anywhere!

Associative Mapping

Main memory (128 bytes)

Needs 7-bits address



Block Size = 8 bytes

MBN = 128 / 8 = 16 (2^4)

CBN = 32 / 8 = 4 (2^2)

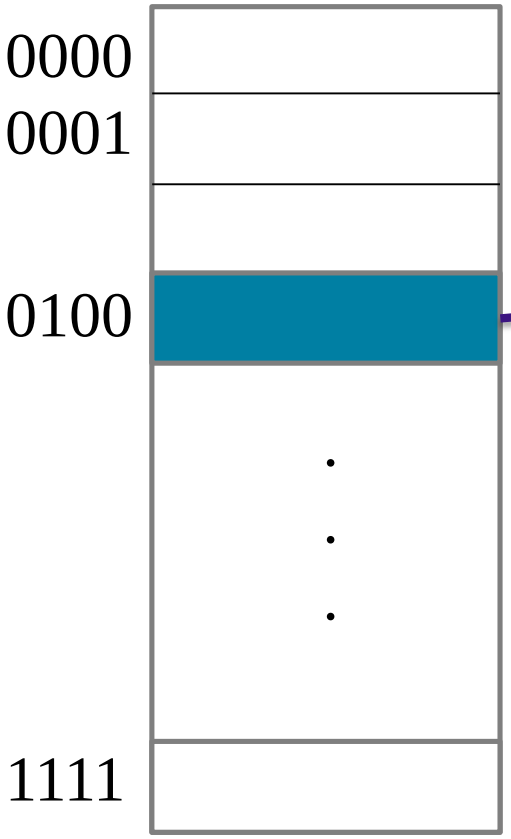
Cache memory (32 bytes)

How mapping occurs?

Associative Mapping

Main memory (128 bytes)

Needs 7-bits address

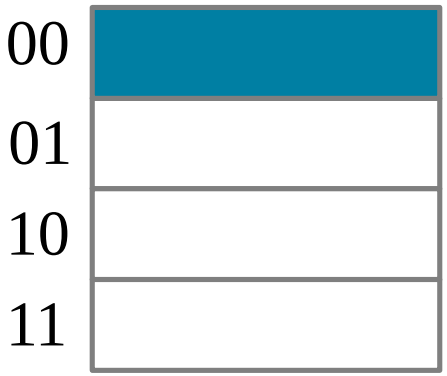


Block Size = 8 bytes

MBN = 128 / 8 = 16 (2^4)

CBN = 32 / 8 = 4 (2^2)

Cache memory (32 bytes)

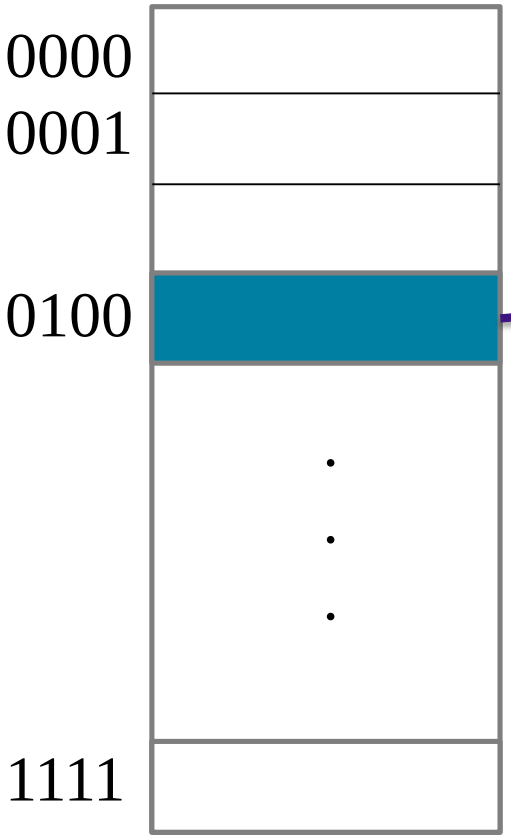


How mapping occurs?

Associative Mapping

Main memory (128 bytes)

Needs 7-bits address

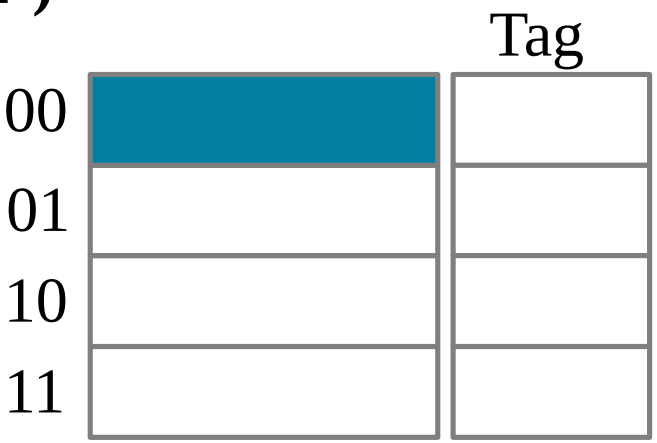


Block Size = 8 bytes

MBN = 128 / 8 = 16 (2^4)

CBN = 32 / 8 = 4 (2^2)

Cache memory (32 bytes)

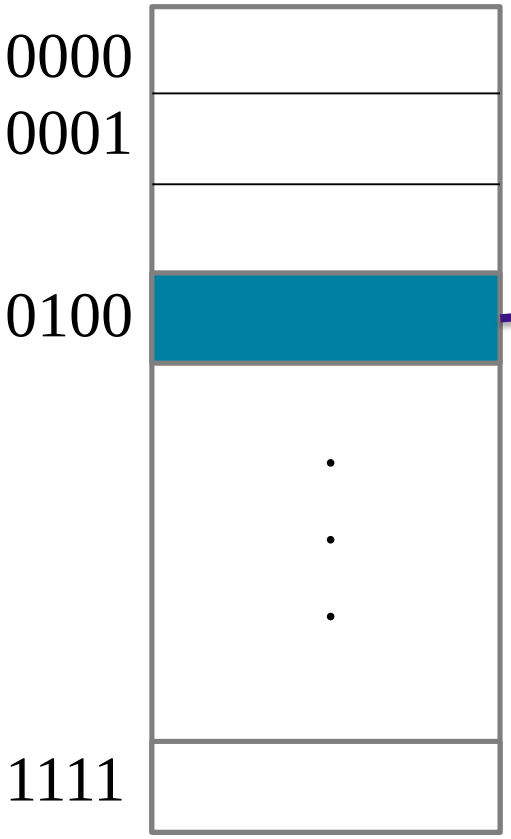


How mapping occurs?

Associative Mapping

Main memory (128 bytes)

Needs 7-bits address

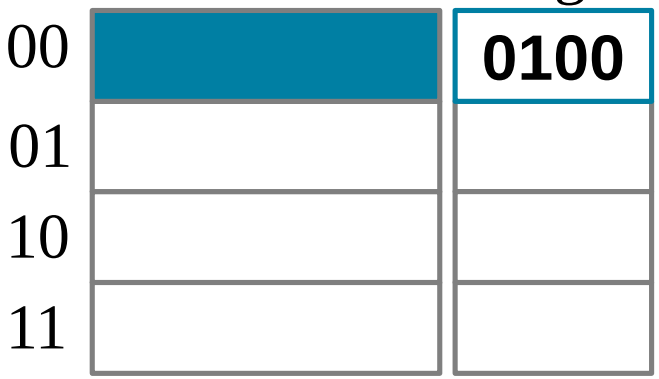


Block Size = 8 bytes

MBN = 128 / 8 = 16 (2^4)

CBN = 32 / 8 = 4 (2^2)

Cache memory (32 bytes)



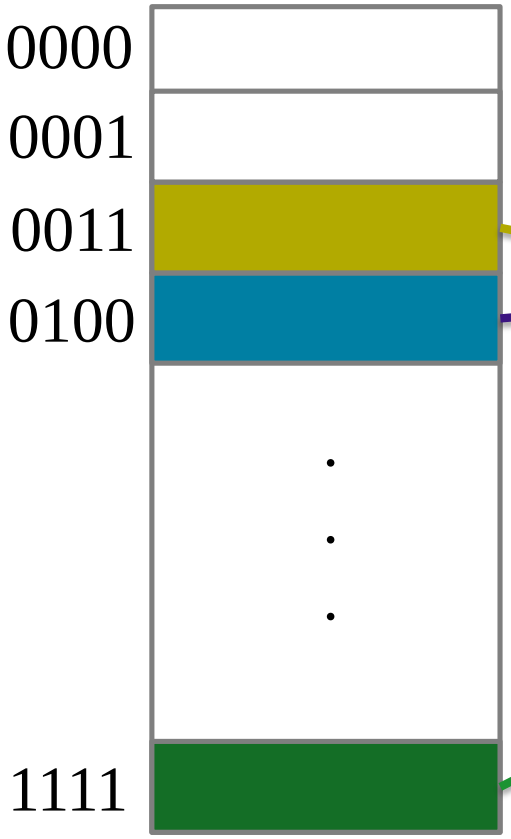
How mapping occurs?

Entire MBN must be stored in the tag field (Tag bits = MBN)

Associative Mapping

Main memory (128 bytes)

Needs 7-bits address

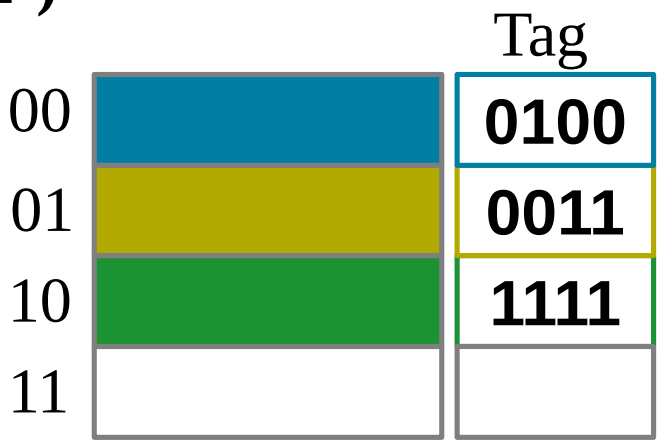


Block Size = 8 bytes

MBN = 128 / 8 = 16 (2^4)

CBN = 32 / 8 = 4 (2^2)

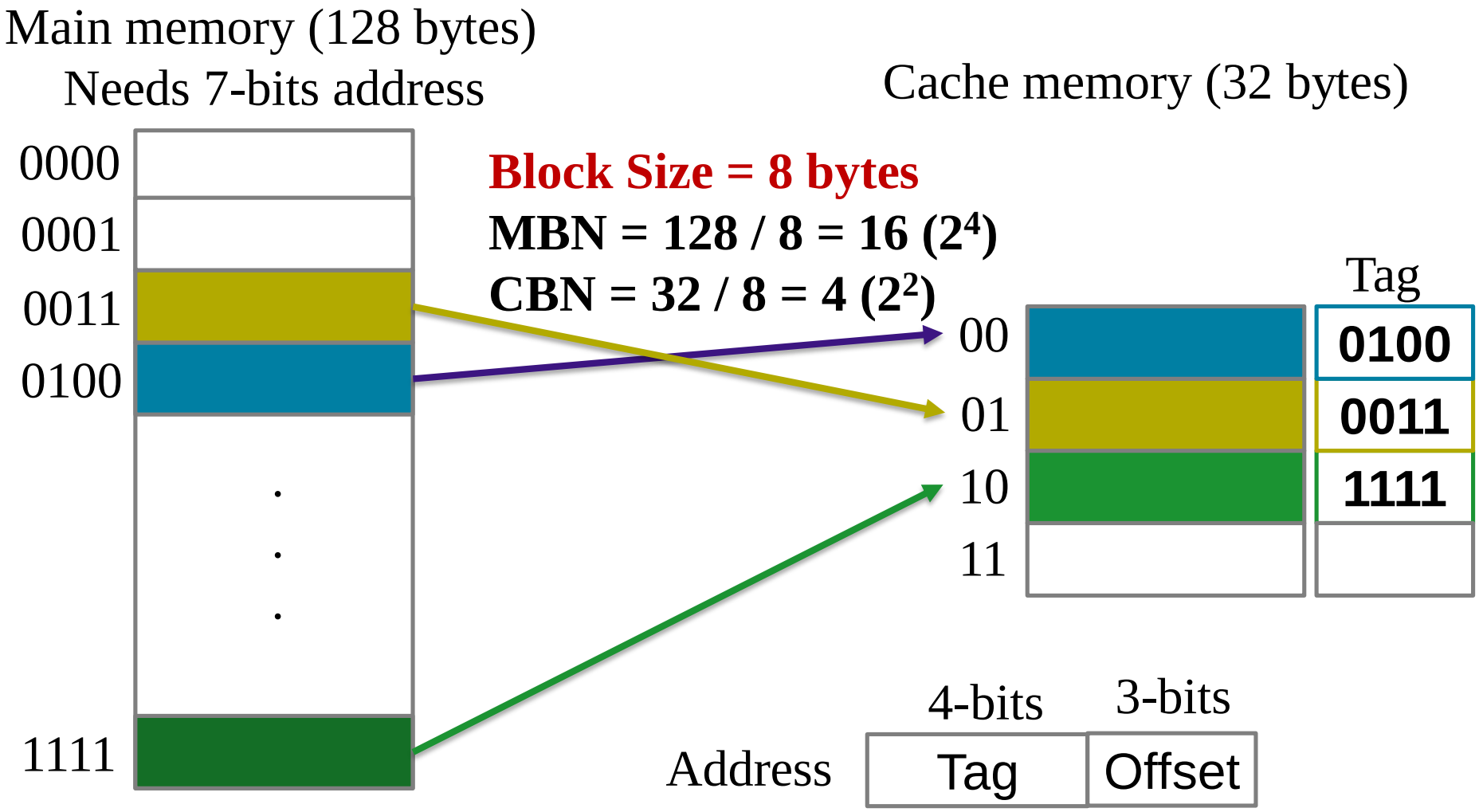
Cache memory (32 bytes)



How mapping occurs?

Entire MBN must be stored in the tag field (Tag bits = MBN)

Associative Mapping



Associate Mapping - Example

- Consider an associate cache of size 8KB and a block size of 32B, and a main memory of size 1GB.
  - What is the number of Tag bits?

BS = 32B (2^5) MBN = 1GB (2^{30})

  - How many comparisons to check if a block exists?

Associate Mapping - Example

- Consider an associate cache of size 8KB and a block size of 32B, and a main memory of size 1GB.
  - What is the number of Tag bits?

BS = 32B (2^5) MBN = 1GB (2^{30})

  - How many comparisons to check if a block exists?

We need to check the tag of all cache lines, so

Associate Mapping - Example

- Consider an associate cache of size 8KB and a block size of 32B, and a main memory of size 1GB.
  - What is the number of Tag bits?

BS = 32B (2^5) MBN = 1GB (2^{30})

  - How many comparisons to check if a block exists?

We need to check the tag of all cache lines, so

CBN = 8KB (2^{13}) / BS (2^5) = (2^8) = 256

Associate Mapping - Example

- Consider an associate cache of size 8KB and a block size of 32B, and a main memory of size 1GB.
  - What is the number of Tag bits?

BS = 32B (2^5) MBN = 1GB (2^{30})

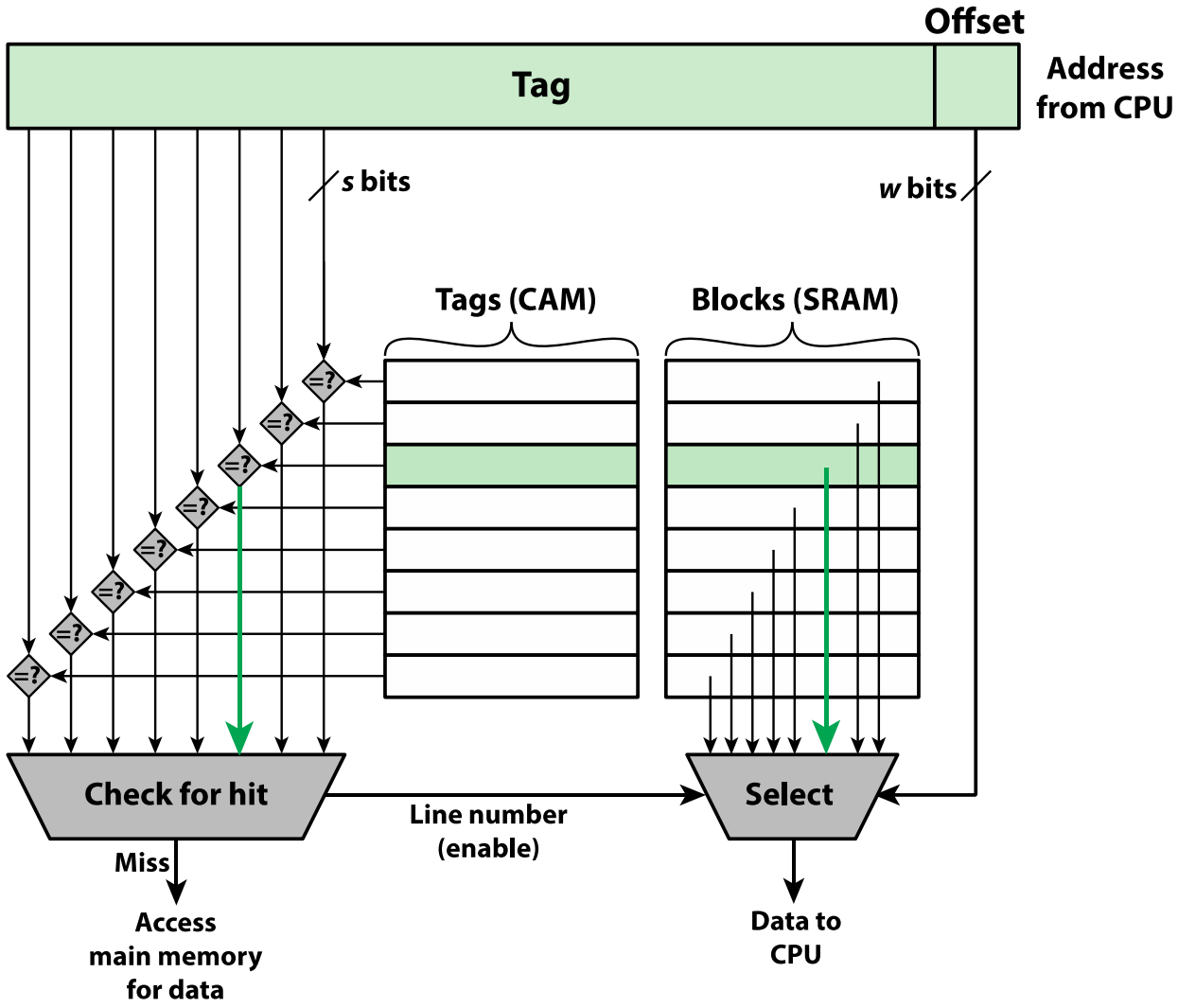
  - How many comparisons to check if a block exists?

We need to check the tag of all cache lines, so

CBN = 8KB (2^{13}) / BS (2^5) = (2^8) = 256

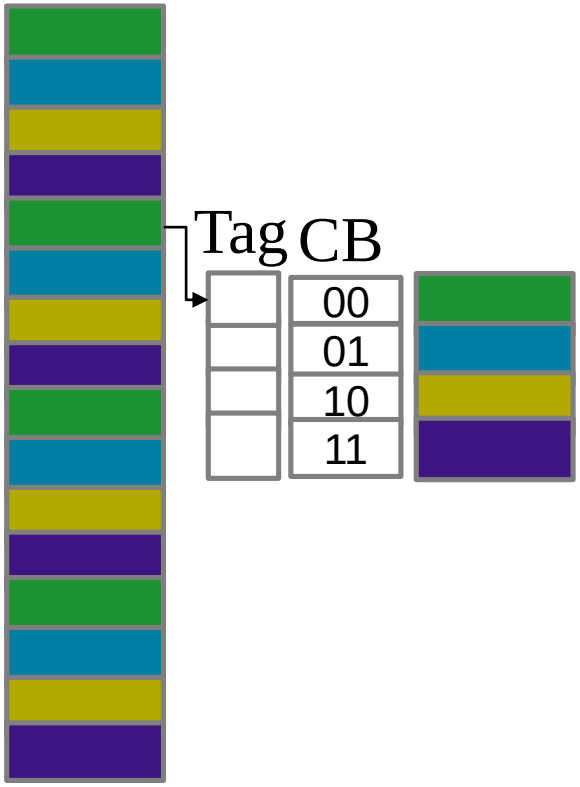
TOO MANY? Use CAM

Figure 5.10
Fully Associative Cache Organization

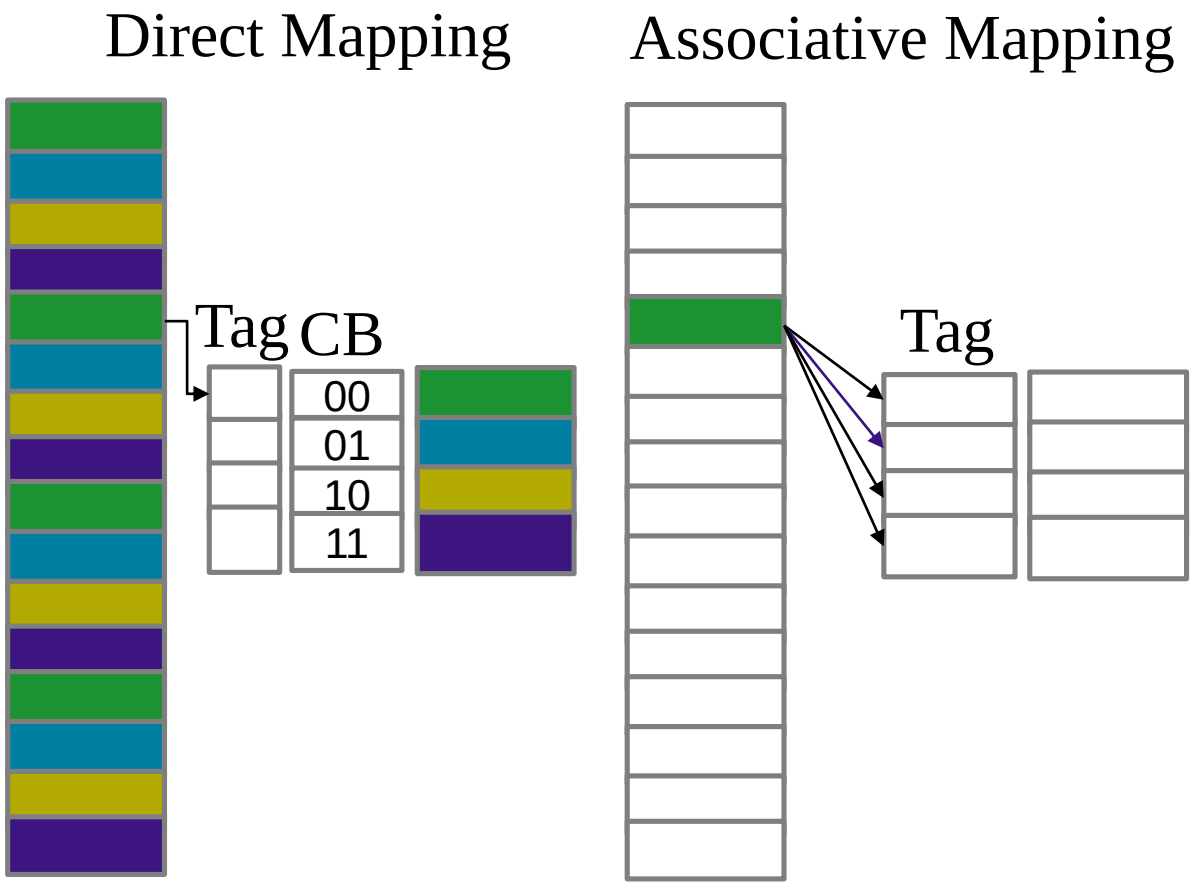


Set Associative Mapping

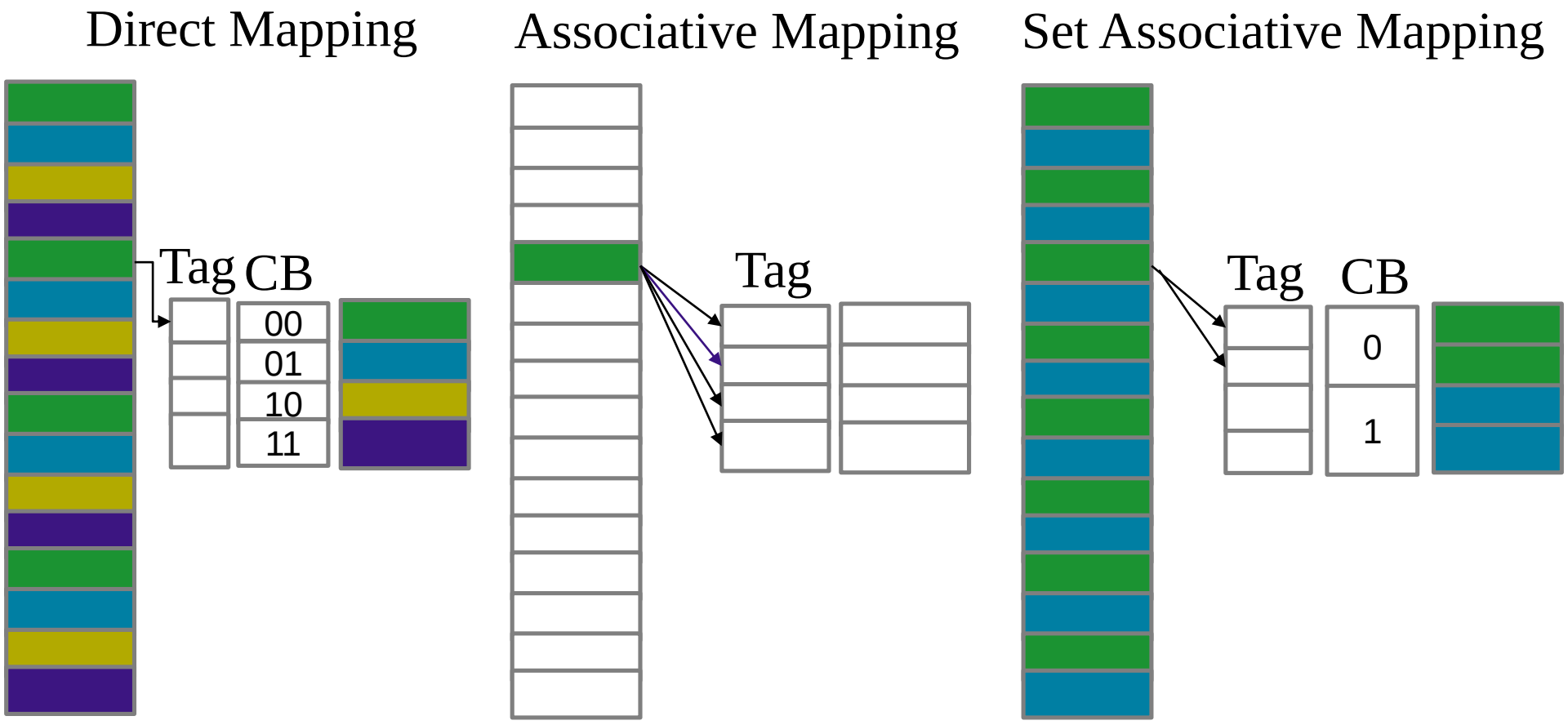
Direct Mapping



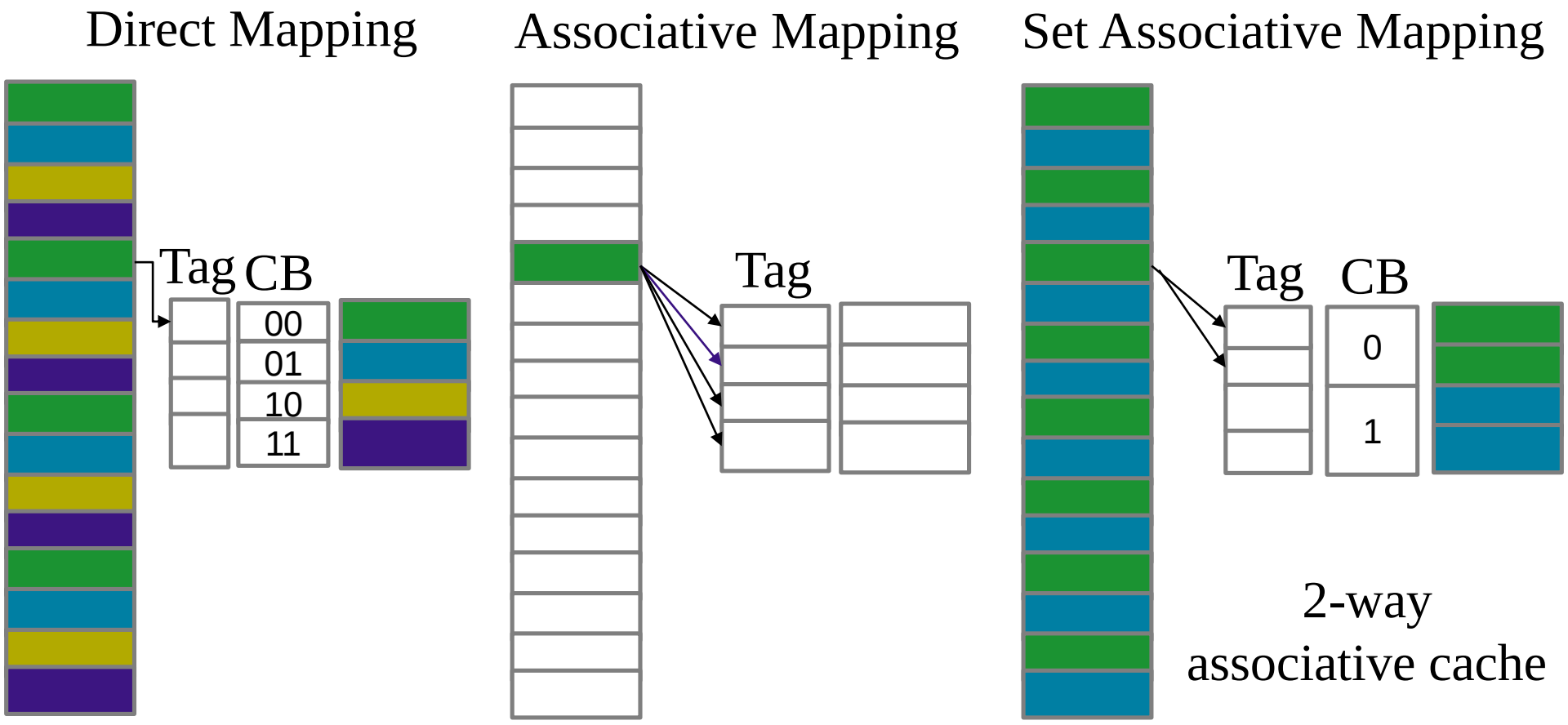
Set Associative Mapping



Set Associative Mapping



Set Associative Mapping



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

Set Associate Mapping Example

- Assume you want to place block number 6195 in a set associative cache that has 8 blocks, where a block size is 16 bytes. Where will this block be saved in (a) 1-way, (b) 2-way, and (c) 4-way associative caches.

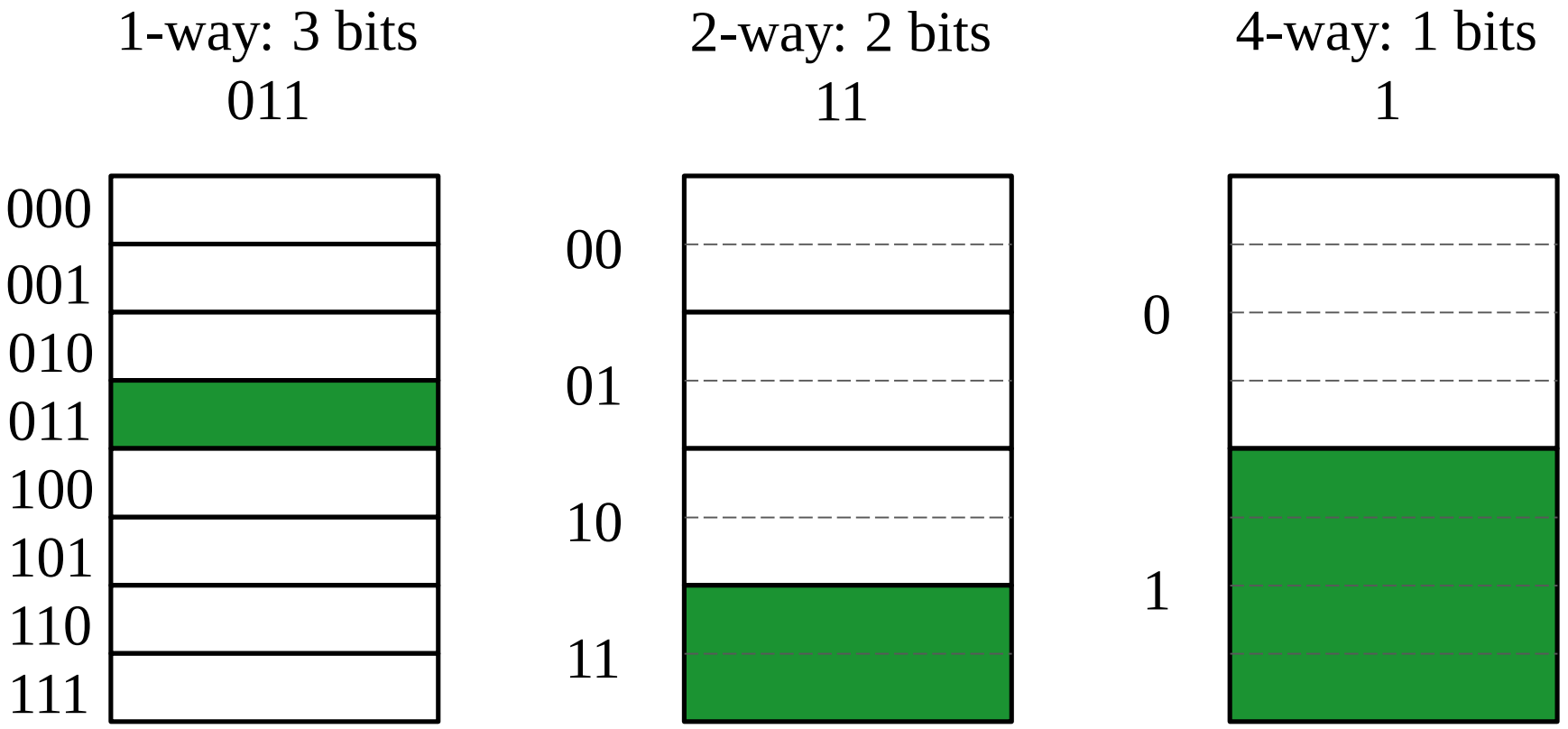
Set Associate Mapping Example

- Assume you want to place block number 6195 in a set associative cache that has 8 blocks, where a block size is 16 bytes. Where will this block be saved in (a) 1-way, (b) 2-way, and (c) 4-way associative caches.

Block address “6195” = 1100000110011 binary
Block size is 16 bytes (2^4) so offset = 4 bits
1-way: 8 sets (blocks) so Index (Cache Block) = 3 bits
2-way: 4 sets so Index = 2 bits
4-way: 2 sets so Index = 1 bits
So address: 110000**011**0011 has 4 bits offset, 3 bits set index, and the rest tag bits (doesn’t matter how many)

Set Associate Mapping Example

110000 **011** 0011



Copyright