

Department of Computer Engineering Techniques (Stage: 4)

Advance Computer Technologies

Dr.: Hussein Ali Ameen

hussein_awadh@mustaqbal-college.edu.iq

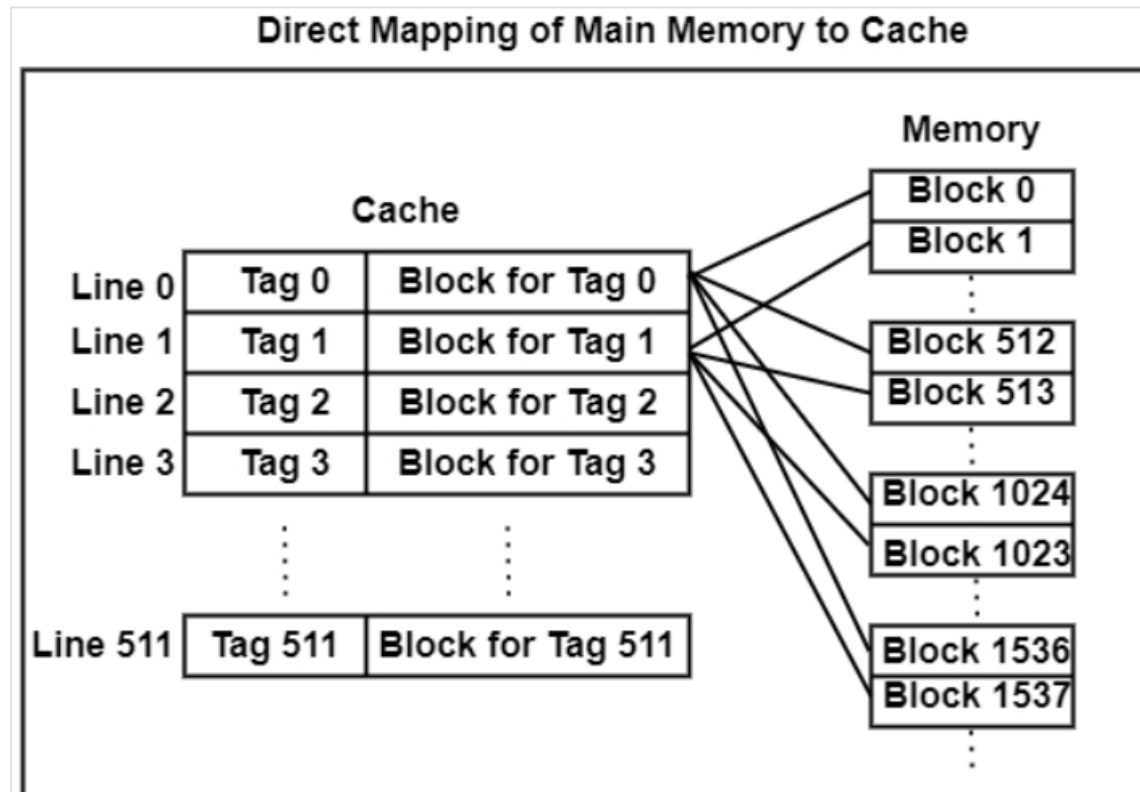
CACHE ORGANIZATION

There are three types of cache organization:

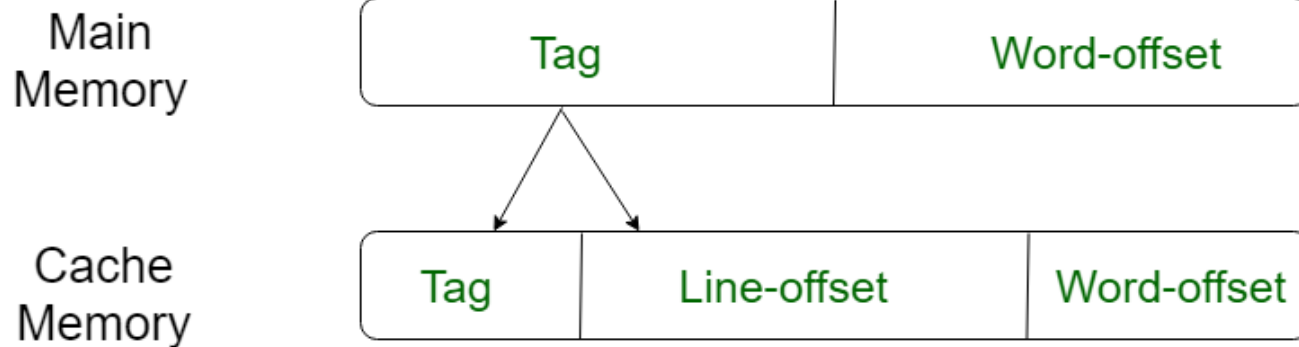
- 1- Direct Mapping
- 2- Fully Associative Mapping
- 3- Set Associative Mapping

1- Direct Mapping

- Direct mapping is a procedure used to assign each memory block in the main memory to a particular line in the cache. If a line is already filled with a memory block and a new block needs to be loaded, then the old block is discarded from the cache.



1- Direct Mapping

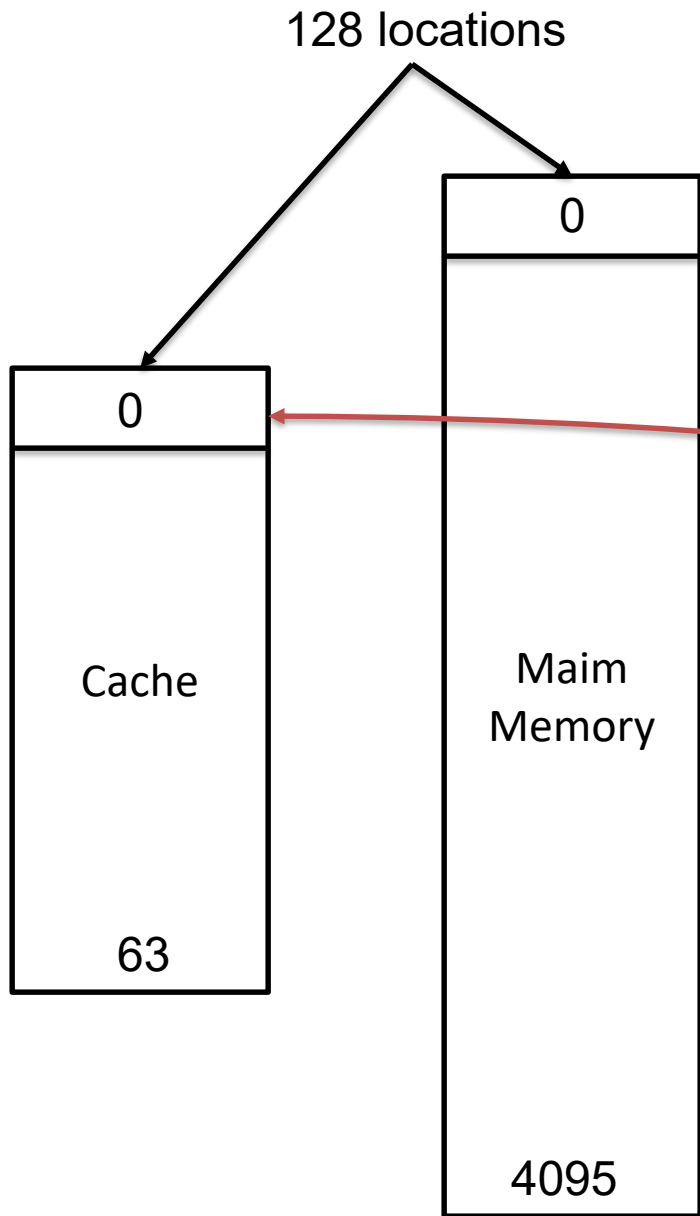


A direct cache mapping process used with 64 lines to access a memory with 4096 lines (pages) and each page / line contain 128 locations

1- Show the format of the main memory

2- Show the format of cache memory

3- How can the following blocks in the main memory maintained in the cache memory 0, 1, 2, 62, 63, 64, 65, 127, 128

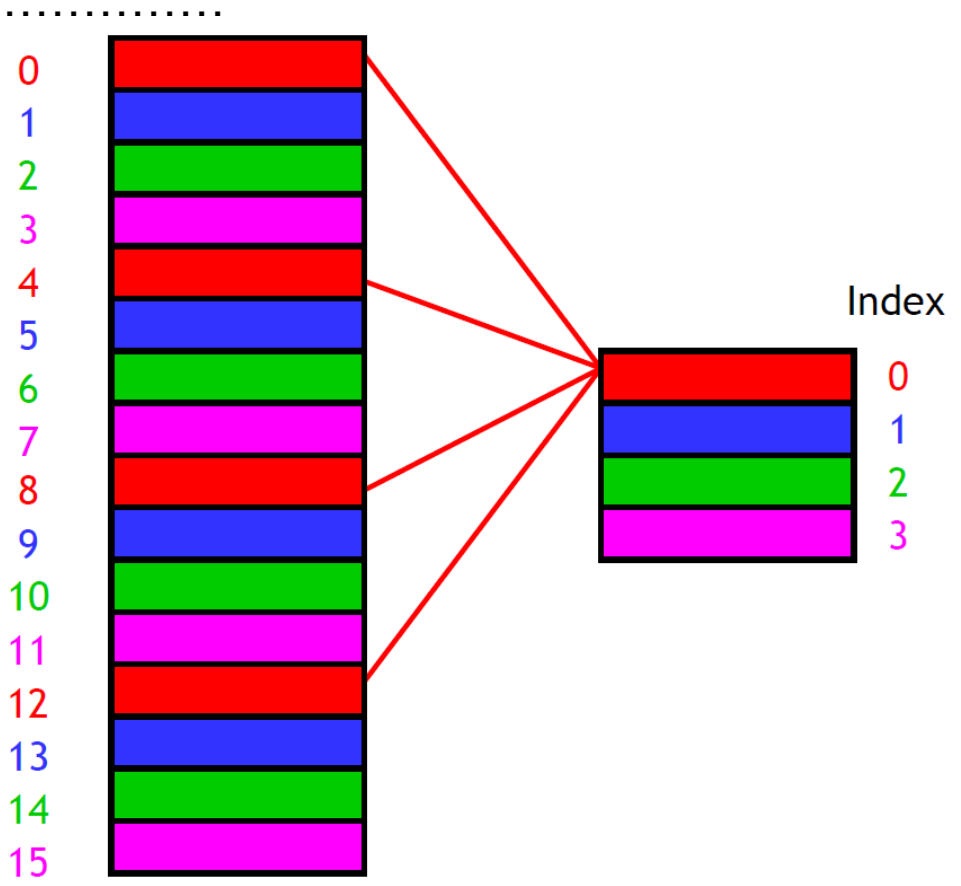
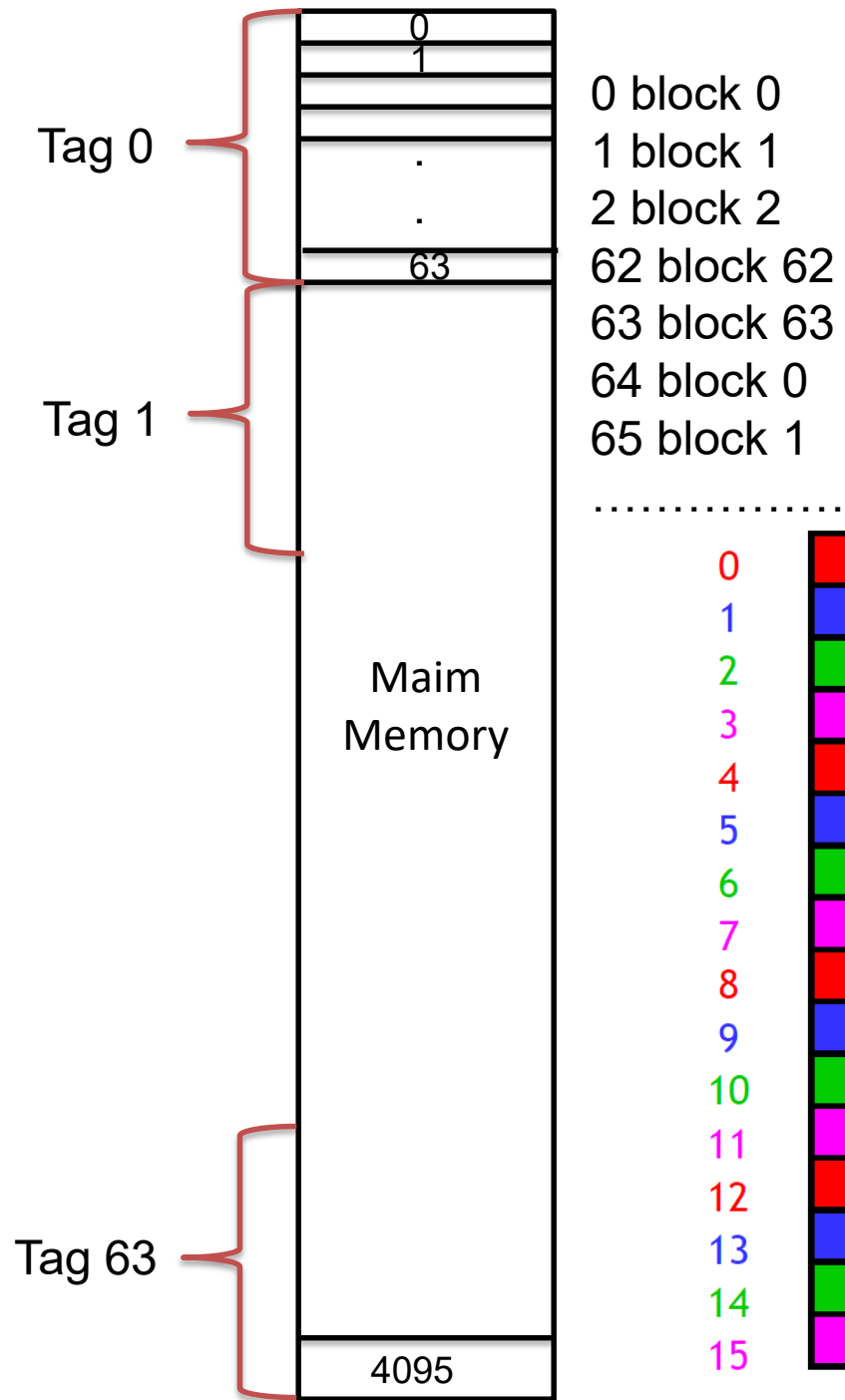


Size of the memory $4096 \times 128 = 2^{12} \times 2^7 = 2^{19}$

Tag	Blocks	Word / location
	64	128

Tag = 6 bit?

0, 1, 2, 62, 63, 64, 65, 127, 128



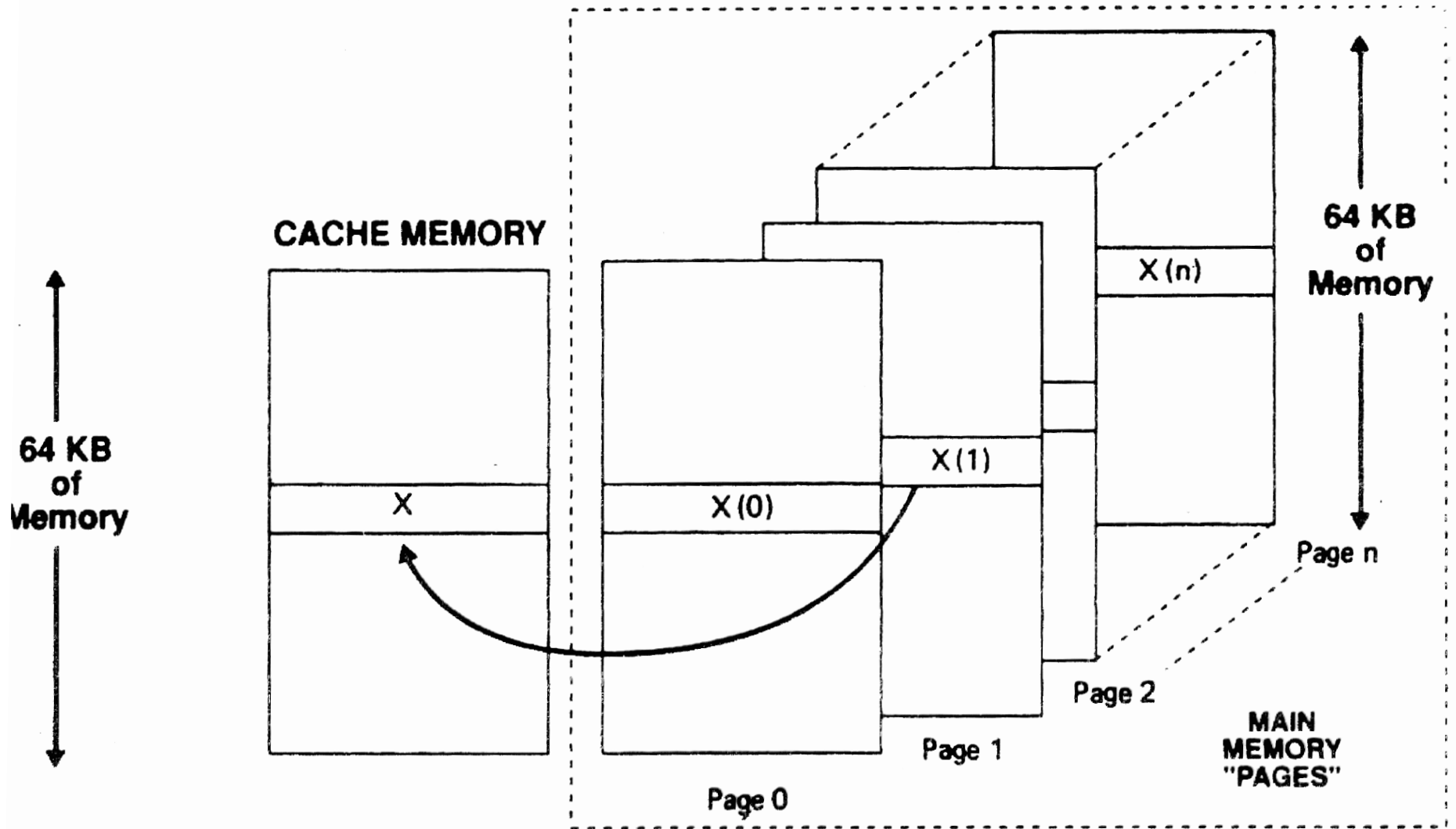
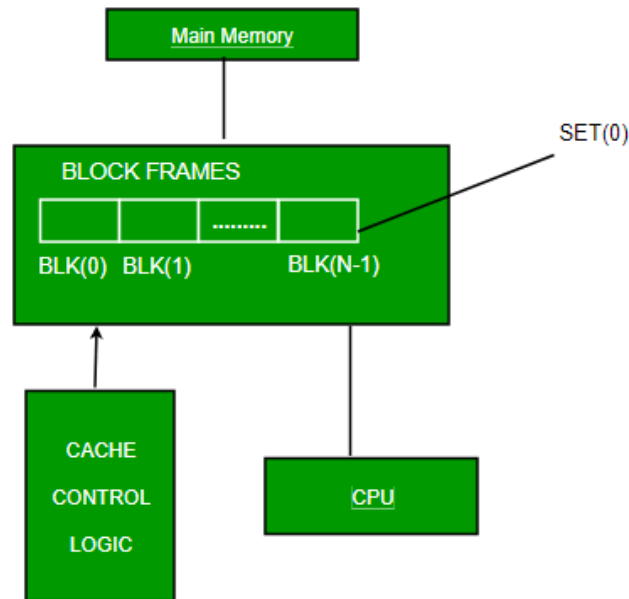


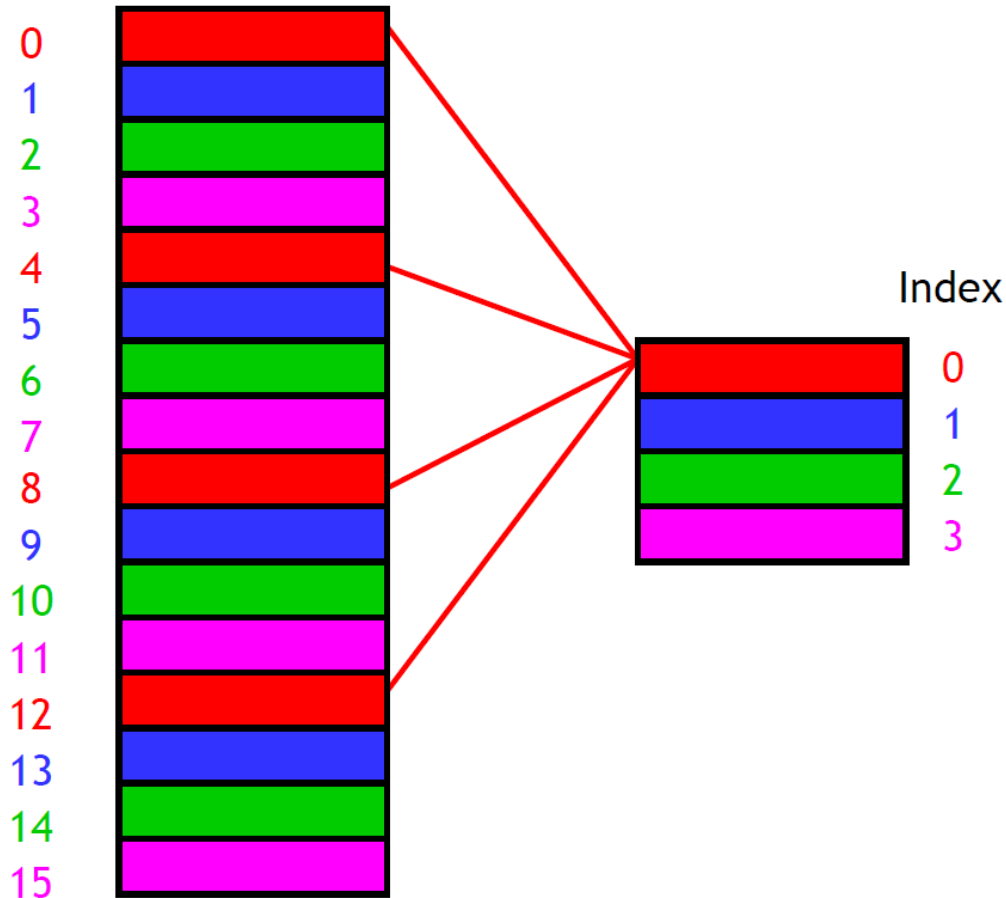
Figure 10.42 Organization of a direct-mapped memory subsystem. (Reprinted by permission of Intel Corp. Copyright/Intel Corp. 1990)

2- Associative Mapping

- In this type of mapping, the associative memory is used to store content and addresses of the memory word. Any block can go into any line of the cache. This means that the word id bits are used to identify which word in the block is needed, but the tag becomes all of the remaining bits. This enables the placement of any word at any place in the cache memory. It is considered to be the fastest and the most flexible mapping form.

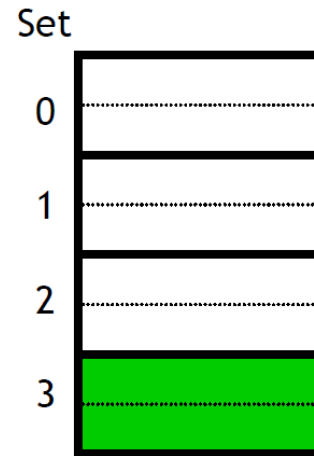
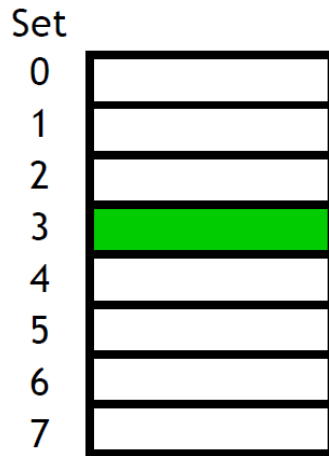


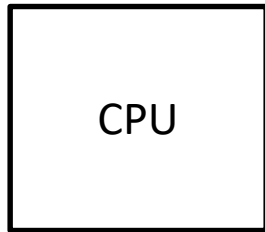
2- Associative Mapping



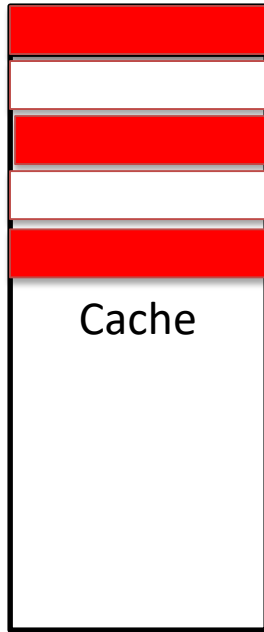
**Remember:
Direct Mapping**

2- Associative Mapping

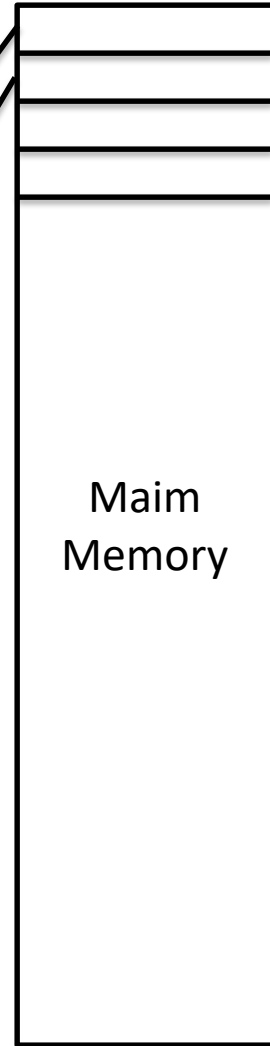




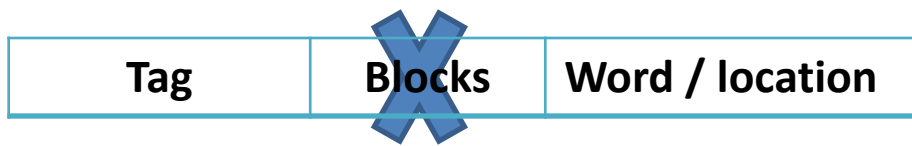
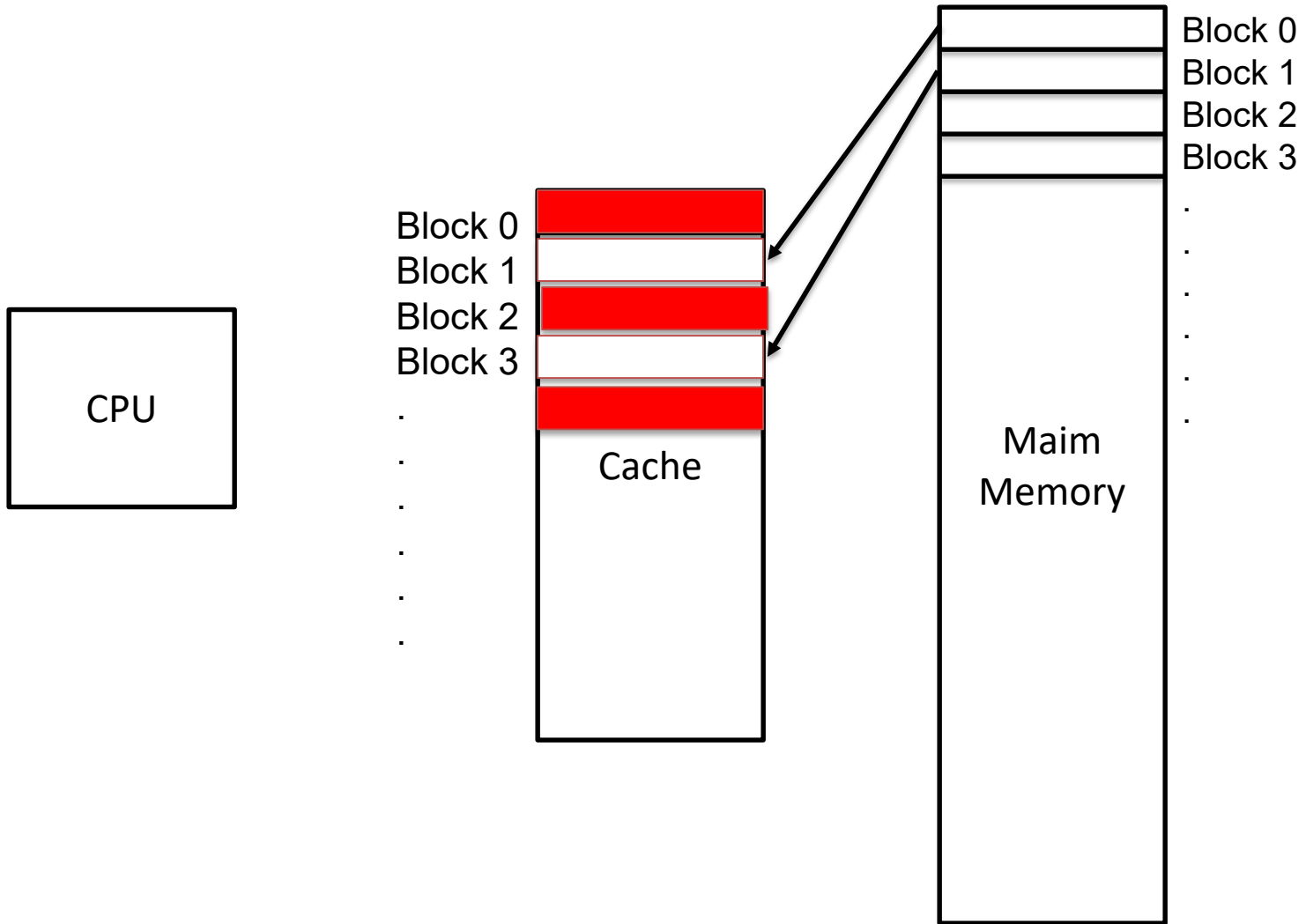
Block 0
Block 1
Block 2
Block 3
.
.
.
.
.
.

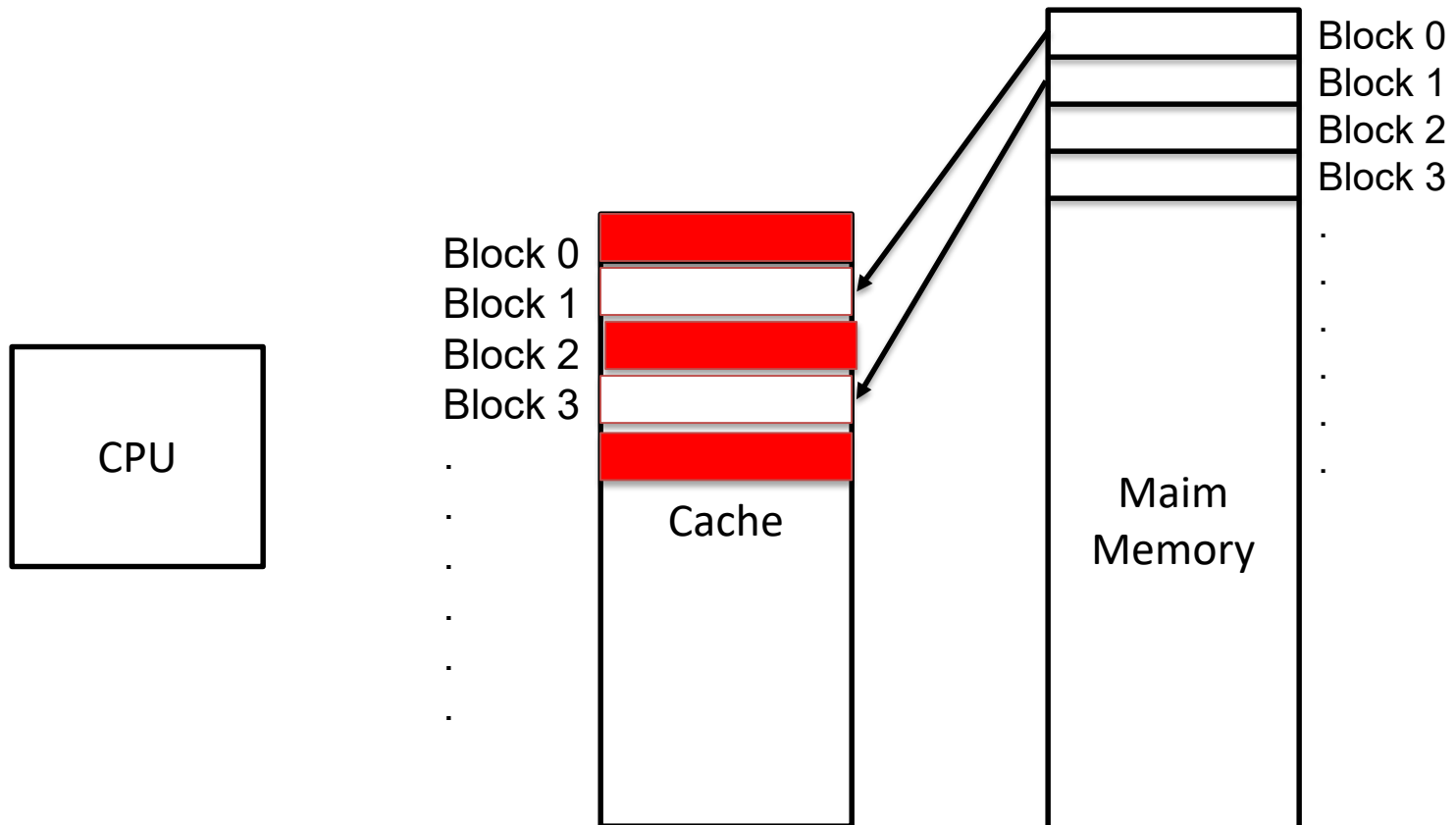


اینما يوجد مكان فارغ



Block 0
Block 1
Block 2
Block 3
.
.
.
.
.
.





Size of the memory

Tag	Word / location
-----	-----------------

Tag = No. of Blocks in main memory

An associative cache mapping process used a memory 2^{16} and block size 8, cache memory 32 lines

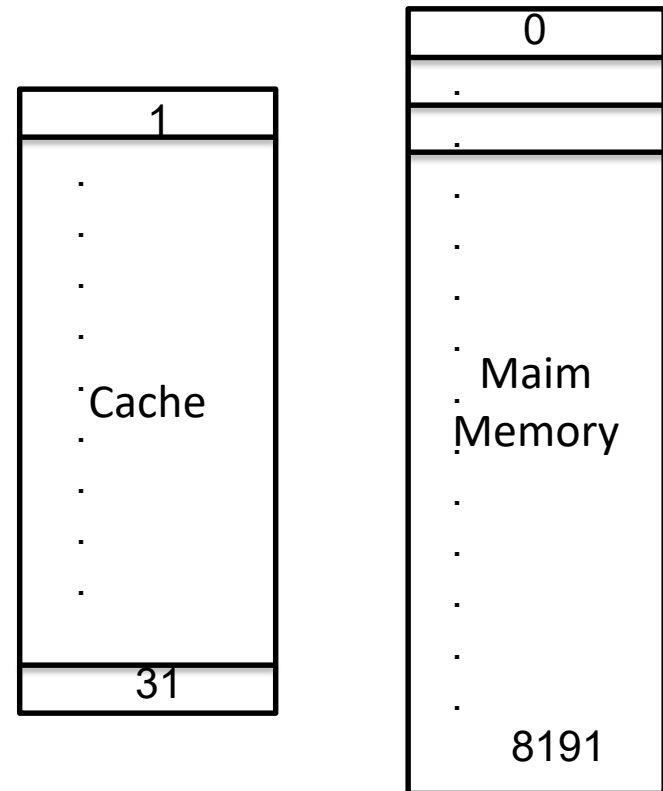
- 1- Show the format of the main memory
- 2- Find the block, tag and word for the address

0010 0000 0111 1011
0111 0110 0110 1000 1001

Size of the memory $2^{16} = 2^{10} \times 2^6 =$
 1024×64
 $= 64 \text{ KB}$

$2^{16} / 2^3 = 2^{13} = 2^{10} \times 2^3 = 1024 \times 8$
 $= 8192 \text{ block size}$

Tag=Block= 13 bit
 Word= 3 bit



0010 0000 0111 1011

2- fully associative

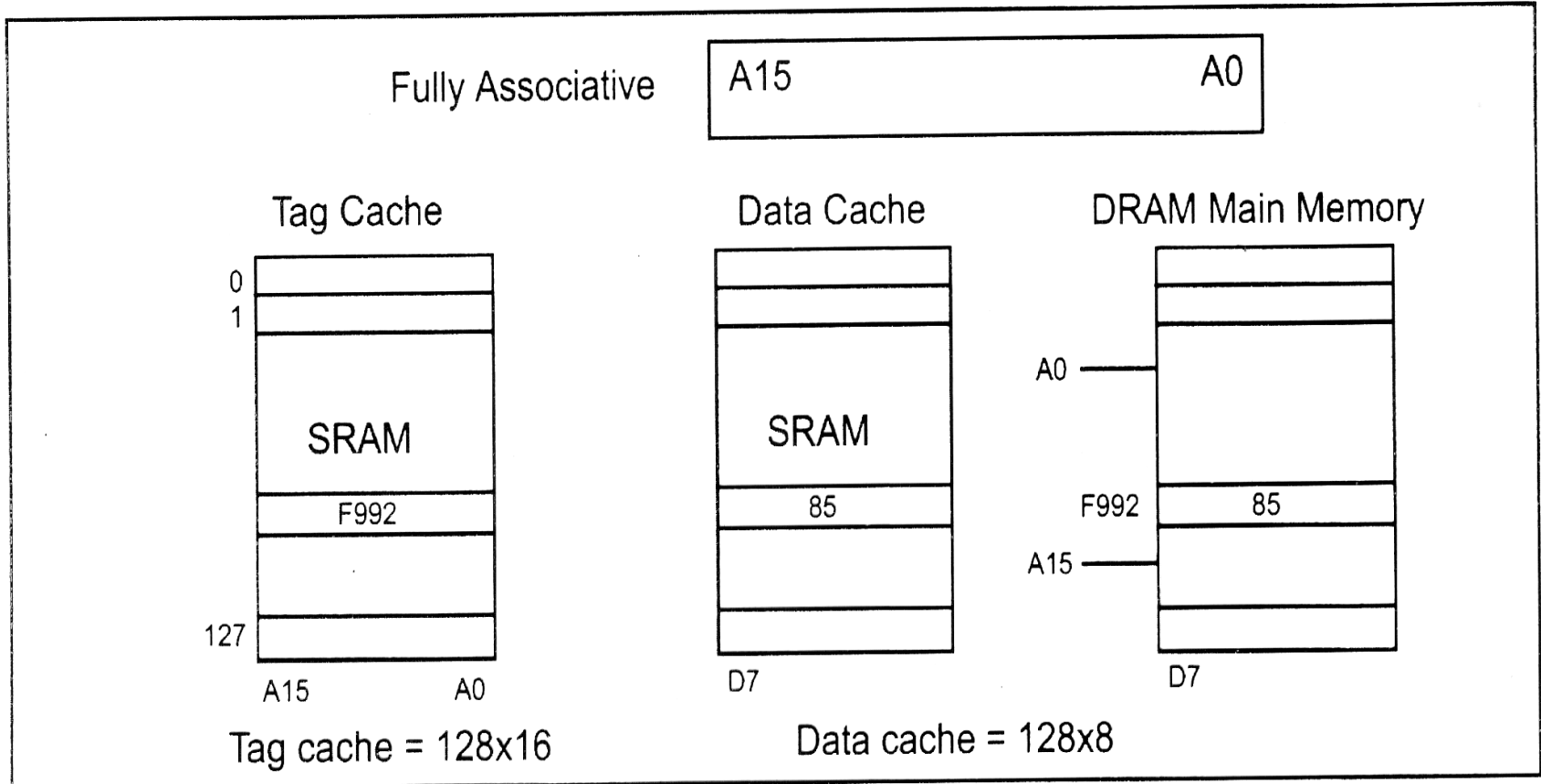


Figure 22-5. Fully Associative Cache

3- set associative

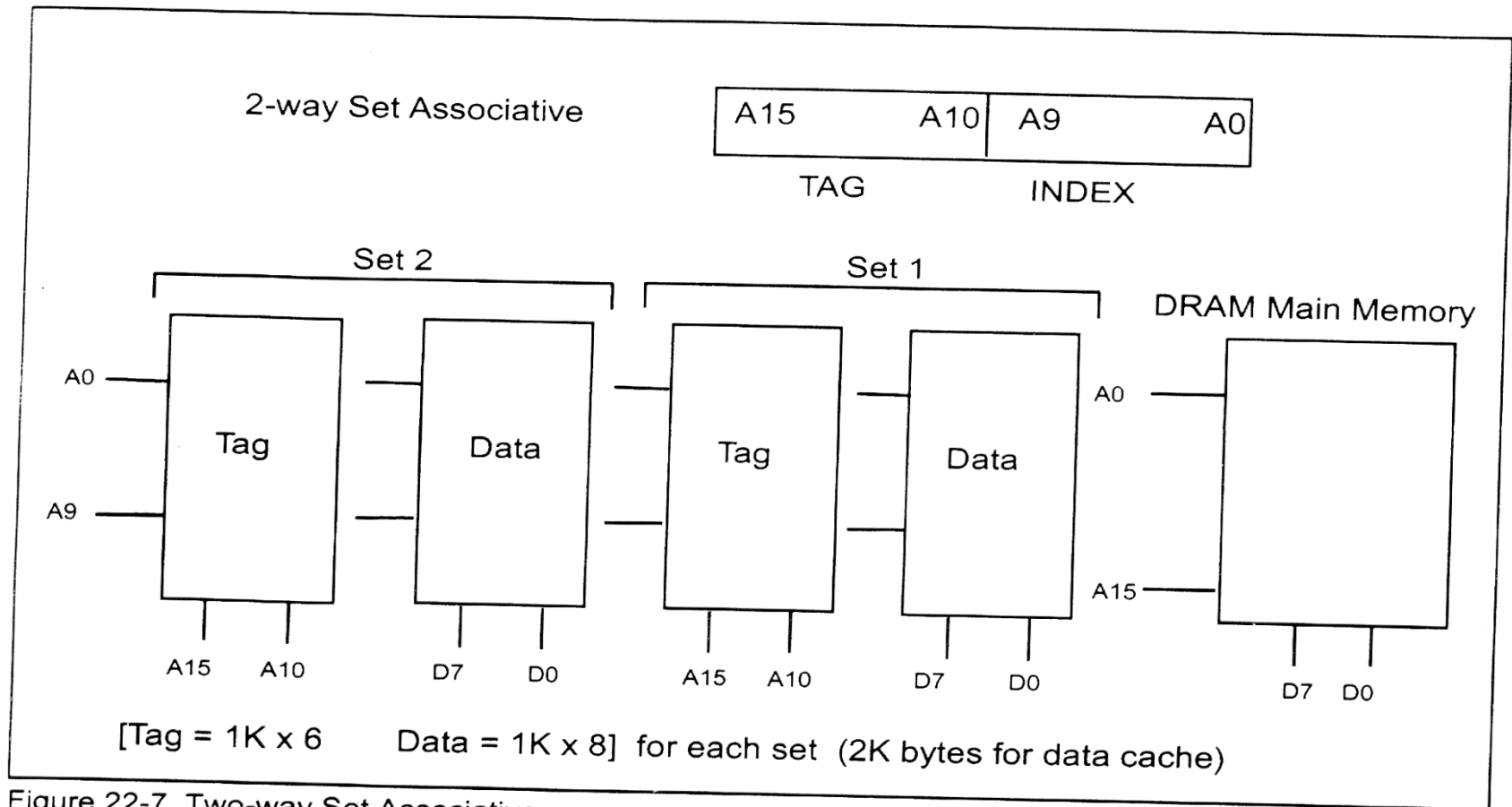


Figure 22-7. Two-way Set Associative

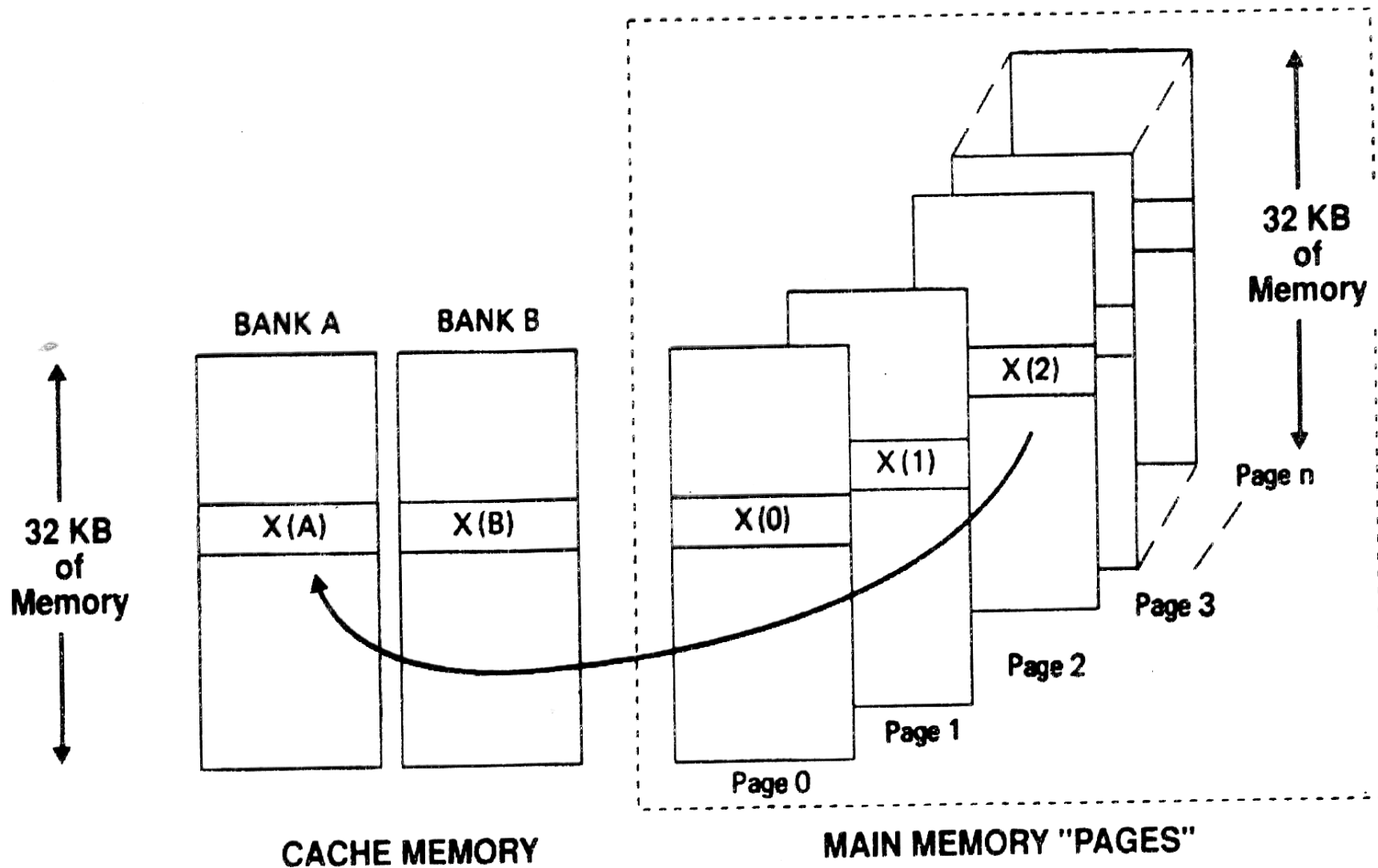
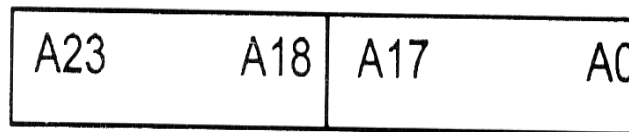


Figure 10.43 Organization of a two-way set associative memory subsystem. (Reprinted by permission of Intel Corp. Copyright/Intel Corp. 1990)

Example 22-14

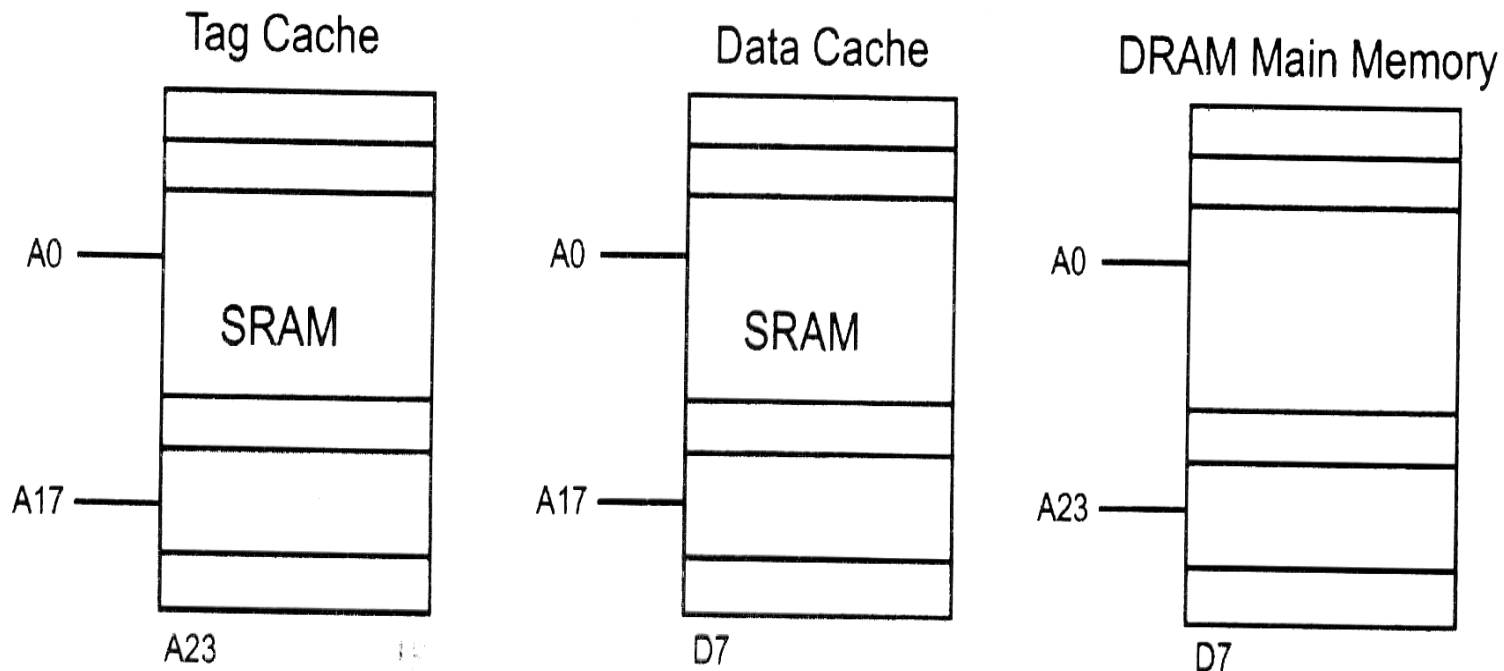
This example shows direct-mapped cache for 16M main memory.

Direct Mapped



TAG

INDEX



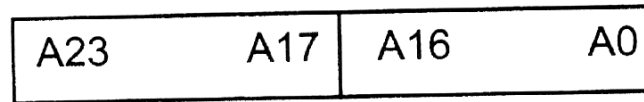
$$\text{Tag cache} = (2^{18} \times 6) / 8 = 192\text{K bytes}$$

$$\text{Data cache} = (2^{18} \times 8) / 8 = 256\text{K bytes}$$

Example 22-15

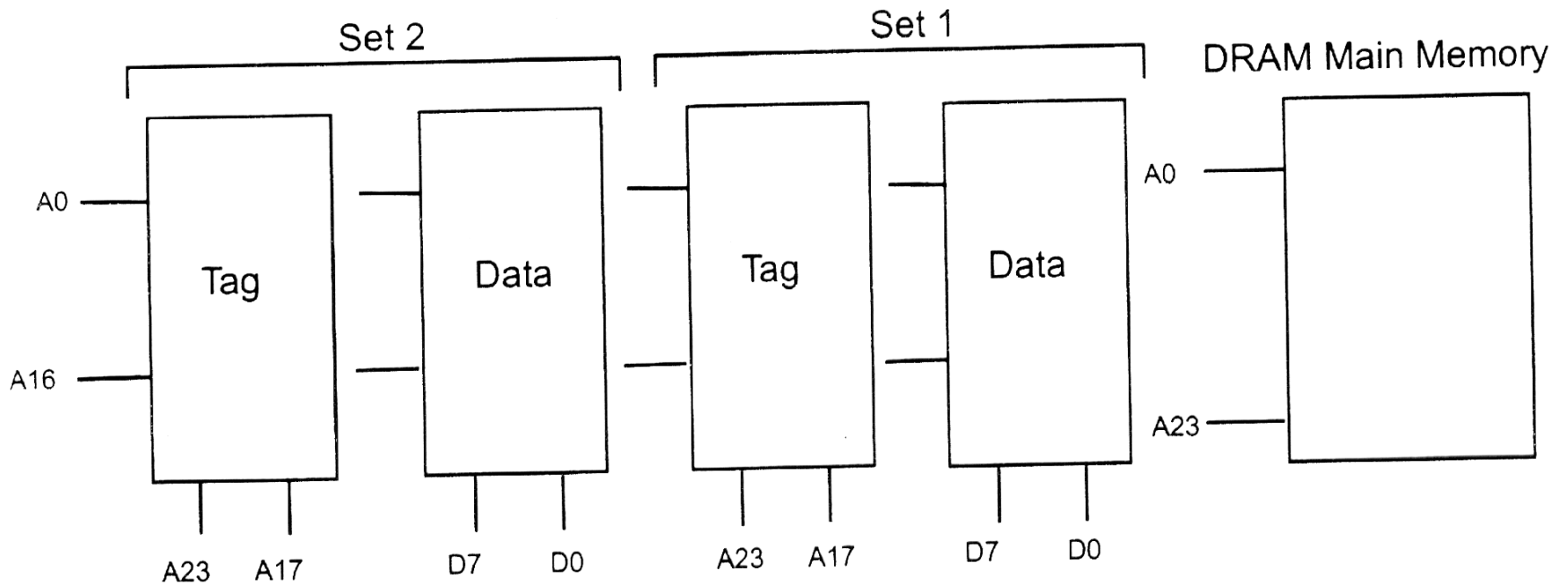
This example shows 2-way set associative cache for 16 MB main memory.

2-Way Set Associative



TAG

INDEX



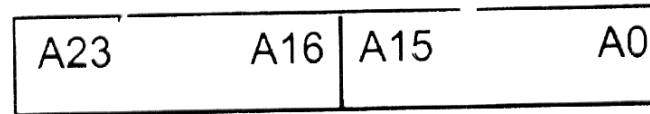
$$\text{Tag} = 2[(2^{17} \times 7)/8] = 224\text{K bytes}$$

$$\text{Data} = 2[(2^{17} \times 8)/8] = 256\text{K bytes}$$

Example 22-16

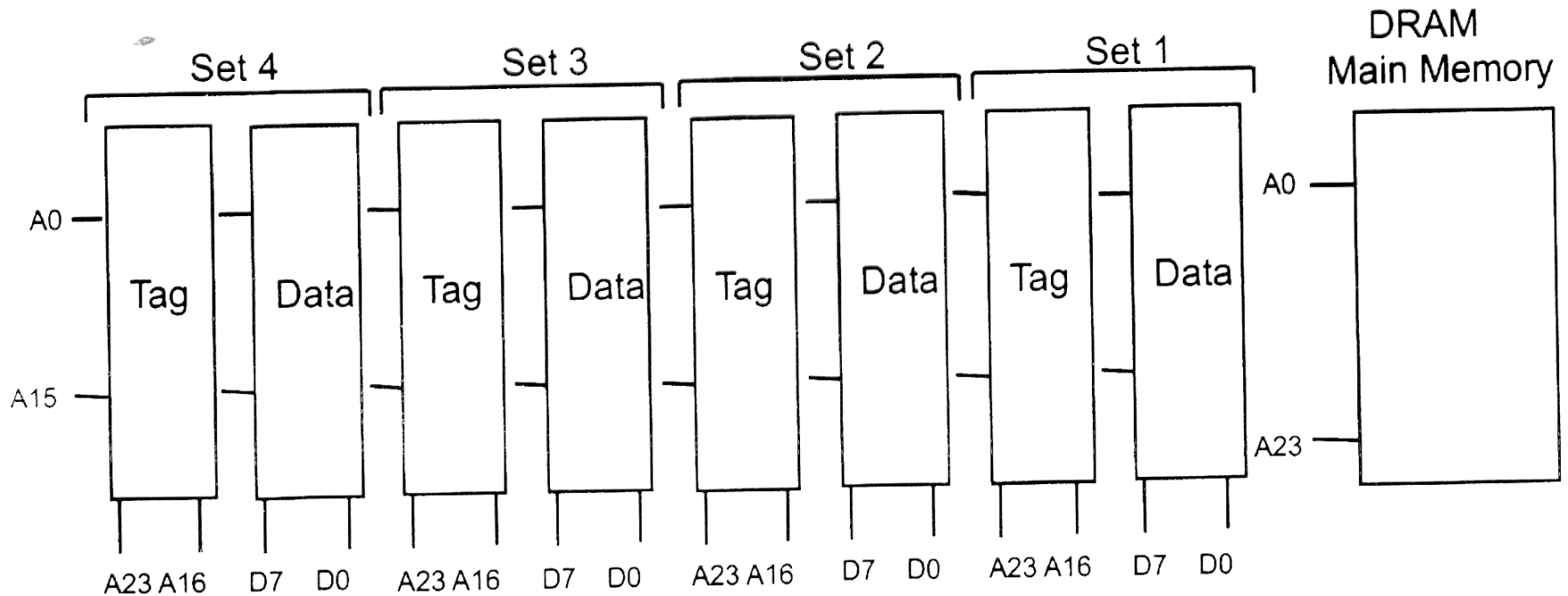
This example shows 4-way set associative cache for 16 MB main memory.

4-Way Set Associative



TAG

INDEX



$$\text{Tag} = 4[(2^{16} \times 8)/8] = 256\text{K bytes}$$

$$\text{Data} = 4[(2^{16} \times 8)/8] = 256\text{K bytes}$$

40. Calculate the tag and data cache sizes needed for each of the following cases if the memory requesting address to main memory is 20 bits ($A_{19} - A_0$). Assume a data bus of 8 bits. Draw a block diagram for each case.
- (a) fully associative of 1024 depth
 - (b) direct mapped where $A_{15} - A_0$ is for the index
 - (c) 2-way set associative where $A_{14} - A_0$ is for the index
 - (d) 4-way set associative
 - (e) 8-way set associative
41. In Problem 40, compare the size of data cache and tag cache parts (b), (c), (d), and (e). What is your conclusion?
42. Calculate the tag and data cache sizes needed for each of the following cases if the memory requesting address to main memory is 24 bits ($A_{23} - A_0$). Assume a data bus of 8 bits. Draw a block diagram for each case.
- (a) fully associative of 1024 depth
 - (b) direct mapped where $A_{19} - A_0$ is for the index
 - (c) 2-way set associative where $A_{18} - A_0$ is for the index
 - (d) 4-way set associative
 - (e) 8-way set associative
43. In Problem 42, compare the size of data cache and tag cache for (b), (c), (d), and (e). What is your conclusion based on this comparison?