

Chapter 6

Memory System Hierarchy: Role of Memory System (cont.)



Four Questions Related to Cache Memory

Q1: Where can a MM block be placed in cache memory lines? (Mapping techniques)

Q2: How is a MM block found if it is in cache (i.e. upper level memory)? (Block identification)

Q3: Which MM block should be replaced on a cache miss? (Block replacement)

Q4: What happens on a cache write? (Write strategy)



Q1: Where can a MM block be placed in cache?

(Mapping Techniques)

What is the meaning of Mapping techniques?

Because there are fewer cache lines than MM blocks (i.e. $C \ll B$), an algorithm is needed for mapping MM blocks into cache lines. Such that, when one MM block is read in cache, **another one should be moved out from cache.**

Features of Mapping Techniques:

- * The **mapping techniques** gives the correspondence between MM blocks and cache lines.
- * **Mapping functions** minimize the probability that when a MM block is moved-out from a cache line, it will be referenced again in the near future.



In general, the CPU executes a program which consists of a set of instructions and data related to some of these instructions. The CPU will execute one instruction in a time. This is done in the following manner:

When the CPU needs a word (**word: may be the next instruction to be executed or may be the data needed to complete the execution of current instruction**), it will generate the address of this word (**n-bit address**) and send it by the address bus in order to fetch this word from the nearest memory which is the cache memory.

- If the word is found in the cache, the word is transferred by data bus to CPU. (**this operation is called cache hit and the CPU references the cache.**)
- Else a **cache miss occur** and the CPU will reference the lower level memory (i.e. **MM**) by sending the **n-bit address** of the word to MM. In this case, the word will be fetched to CPU and the **MM block** that contains this word is transferred to one of the cache lines.



Three Mapping techniques are used to store a MM block in the cache memory lines:

1) Direct Mapped Cache:

- The simplest technique
- Maps each block of MM into only a specific and determined cache line

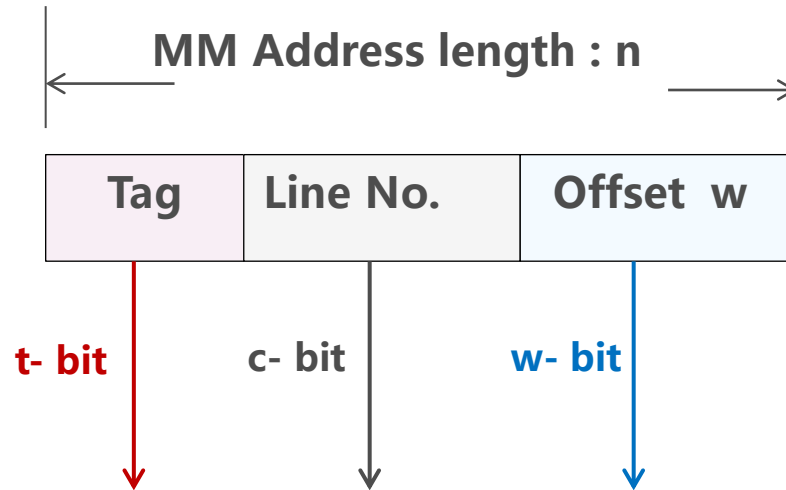
2) Fully Associative Cache: Permits each MM block to be loaded into any available line of the cache

3) Set Associative Cache: Is a compromise that exhibits the strengths of both the direct and associative approaches while reducing their disadvantages



1) Direct Mapped (DM) Cache

Direct mapping (DM) cache interprets a MM address (n-bit) as a three distinct fields:



Direct Mapped Cache Address

The MM address length (bits):

$$n = (t + c + w) \text{ bits}$$

In DM cache each MM block is assigned to a specific line in the cache according to the formula:

$$i = j \text{ mod } C$$

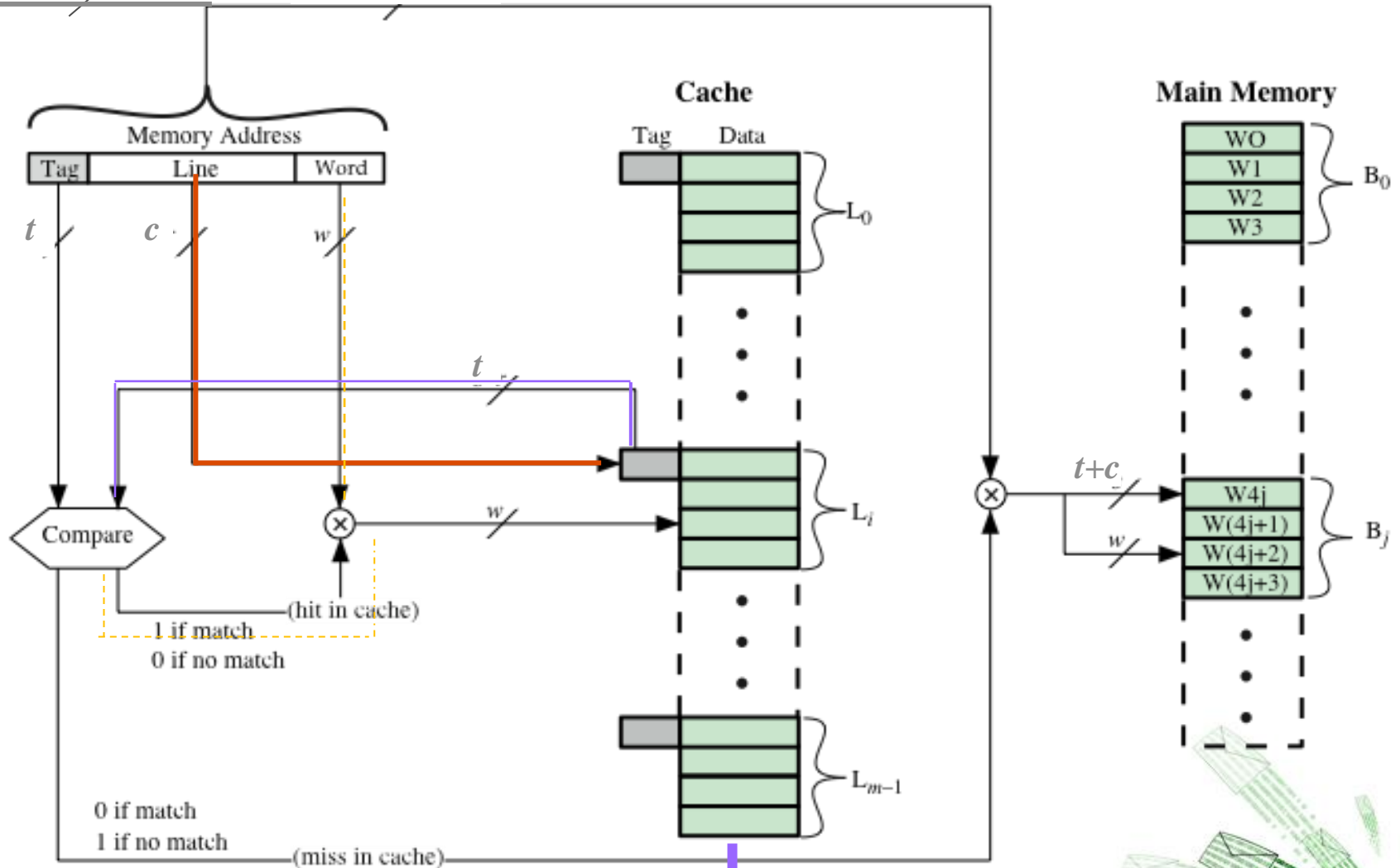
where **i**: is the cache line number assigned to MM block **j**,
C: is the total number of lines in the cache.



Address generated by CPU

n

$$n = t + c + w$$



Fetch the requested word to CPU



Since the cache at start is always empty. Using the mapping formula:

$$i = j \bmod C$$

Cache line	MM block that held in a cache line
0	0, C, 2C, 3C,.....
1	1, C+1, 2C+1, ...
2	2, C+2, 2C+2,
3	3, C+3, 2C+3, 3C+3, ...
:	:
:	:
C-1	C-1, 2C-1, 3C-1, ...



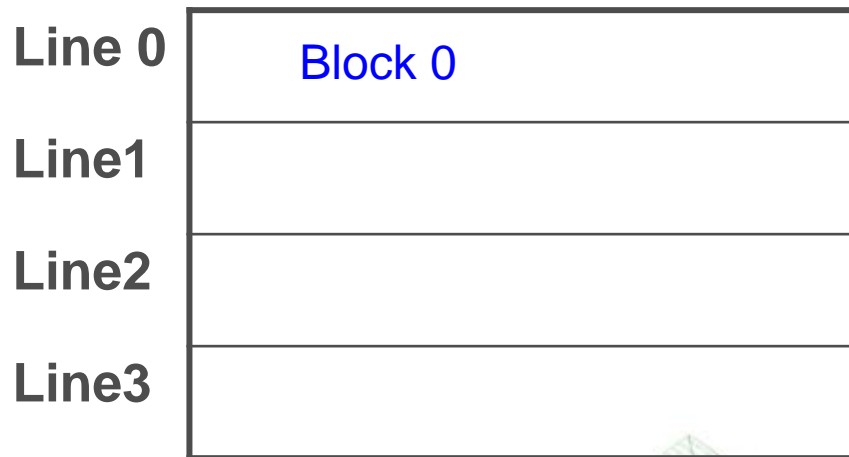
Example: Consider a DM cache contains 4 (1-word) lines. Find the number of misses in the cache if the following sequence of MM block accesses is generated : 0, 8, 0, 6, 8, 9, 6

Solution:

• Since the cache at start is always empty. Using the mapping formula:

$$i = j \text{ mod } C$$

MM Block	Hit	Miss
Block 0		miss



Cache



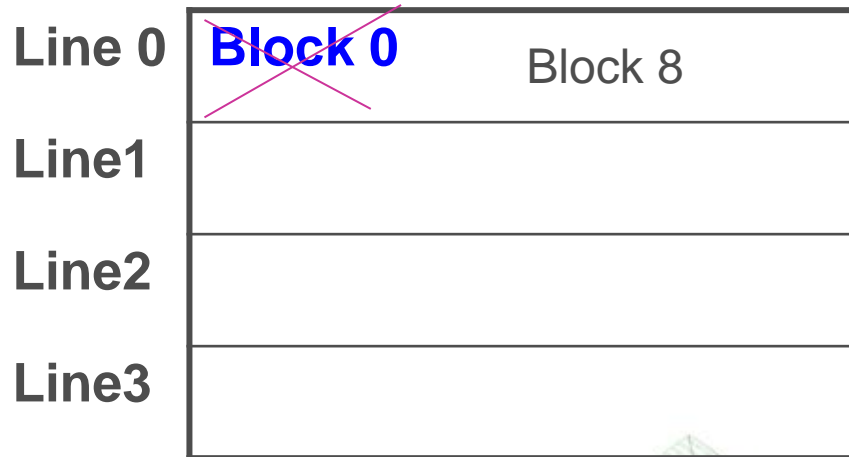
Example: Consider a DM cache contains 4 (1-word) lines. Find the number of misses in the cache if the following sequence of MM block accesses is generated : 0, 8, 0, 6, 8, 9, 6

Solution:

• Since the cache at start is always empty. Using the mapping formula:

$$i = j \text{ mod } C$$

MM Block	Hit	Miss
Block 0		miss
Block 8		miss



Cache



Example: Consider a DM cache contains 4 (1-word) lines. Find the number of misses in the cache if the following sequence of MM block accesses is generated : 0, 8, 0, 6, 8, 9, 6

Solution:

• Since the cache at start is always empty. Using the mapping formula:

$$i = j \text{ mod } C$$

MM Block	Hit	Miss
Block 0		miss
Block 8		miss
Block 0		miss

Line 0	Block 8	Block 0
Line1		
Line2		
Line3		

Cache



Example: Consider a DM cache contains 4 (1-word) lines. Find the number of misses in the cache if the following sequence of MM block accesses is generated : 0, 8, 0, 6, 8, 9, 6

Solution:

- Since the cache at start is always empty. Using the mapping formula:

$$i = j \text{ mod } C$$

MM Block	Hit	Miss
Block 0		miss
Block 8		miss
Block 0		miss
Block 6		miss

Line 0	Block 0
Line1	
Line2	Block 6
Line3	

Cache



Example: Consider a DM cache contains 4 (1-word) lines. Find the number of misses in the cache if the following sequence of MM block accesses is generated : 0, 8, 0, 6, 8, 9, 6

Solution:

• Since the cache at start is always empty. Using the mapping formula:

$$i = j \text{ mod } C$$

MM Block	Hit	Miss
Block 0		miss
Block 8		miss
Block 0		miss
Block 6		miss
Block 8		miss

Line 0	Block 0	Block 8
Line1		
Line2	Block 6	
Line3		

Cache



Example: Consider a DM cache contains 4 (1-word) lines. Find the number of misses in the cache if the following sequence of MM block accesses is generated : 0, 8, 0, 6, 8, 9, 6

Solution:

• Since the cache at start is always empty. Using the mapping formula:

$$i = j \text{ mod } C$$

MM Block	Hit	Miss
Block 0		miss
Block 8		miss
Block 0		miss
Block 6		miss
Block 8		miss
Block 9		miss

Line 0	Block 8
Line1	Block 9
Line2	Block 6
Line3	

Cache



Example: Consider a DM cache contains 4 (1-word) lines. Find the number of misses in the cache if the following sequence of MM block accesses is generated : 0, 8, 0, 6, 8, 9, 6

Solution:

• Since the cache at start is always empty. Using the mapping formula:

$$i = j \text{ mod } C$$

MM Block	Hit	Miss
0		miss
8		miss
0		miss
6		miss
8		miss
9		miss
6	hit	

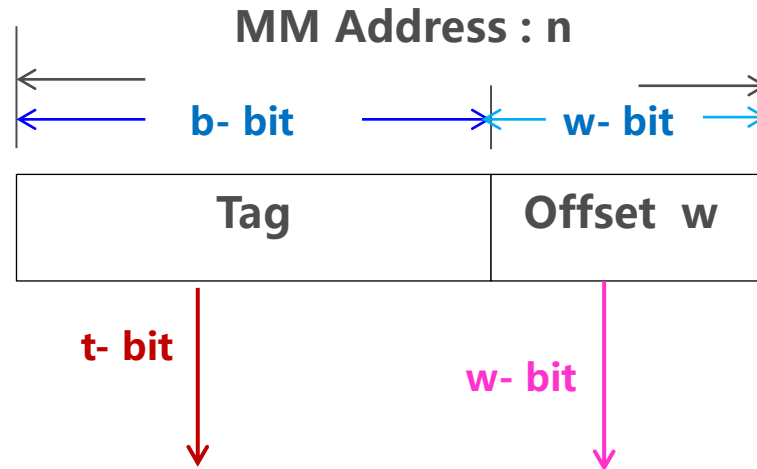
Line 0	Block 8
Line1	Block 9
Line2	Block 6
Line3	

Cache



2) Fully Associative Cache Access

Fully associative cache allows each block of MM to be stored in any line of the cache. Main memory address ($n = b + w$) bits is divided into two distinct fields: tag (t - bits) and word offset (w - bits).

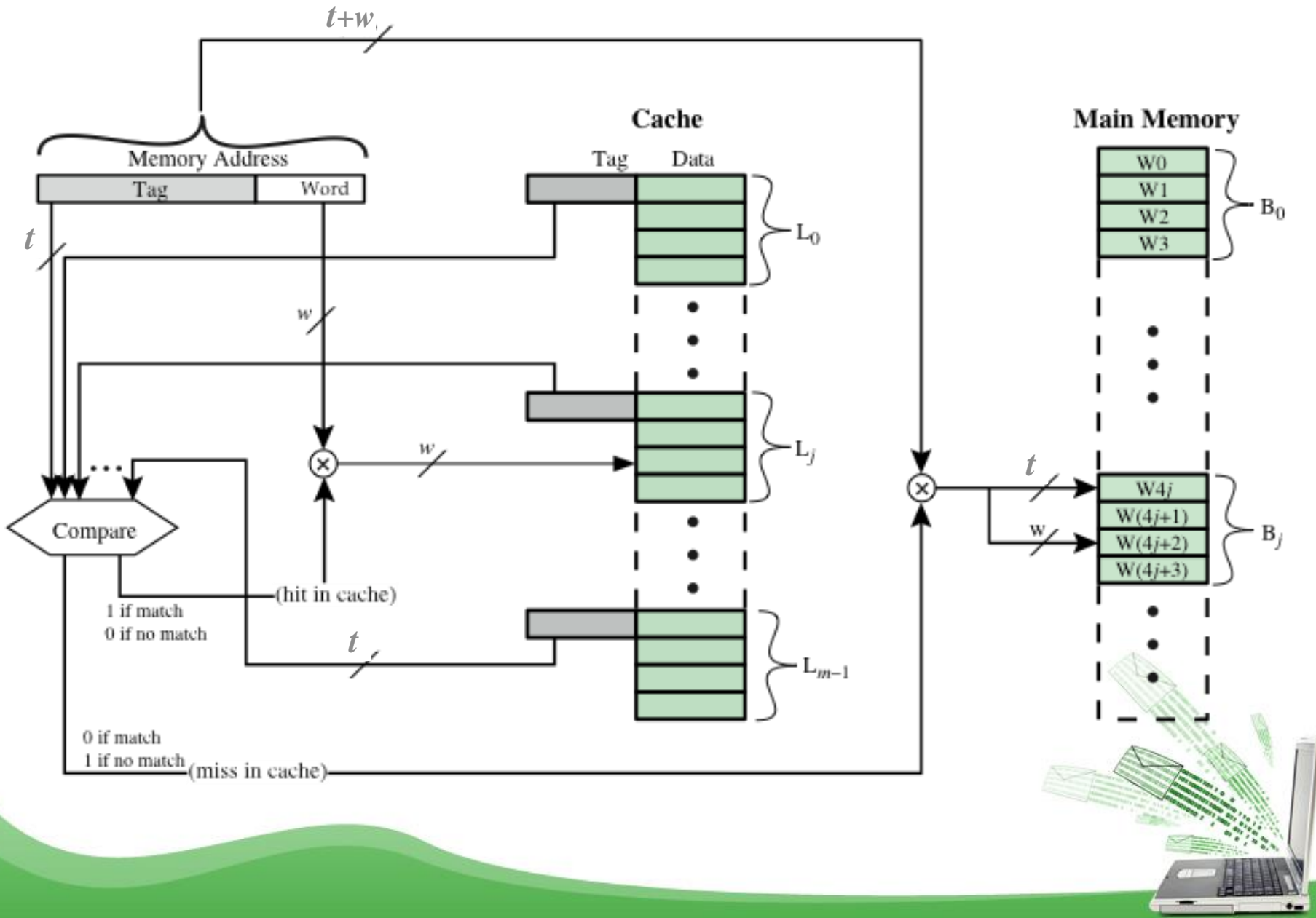


**Fully Associative
Cache Address Fields**

*Fully associative mapping is very flexible and overcomes direct mapping's main weakness. The fully associative cache has the best hit ratio because any line in the cache can hold any address that needs to be cached. However this cache suffers from problems involving searching the cache.



2) Fully Associative Cache Organization



Example: Consider a fully associative cache contains 4 (1-word) lines. Find the number of misses in the cache if the following sequence of MM block accesses is generated : 0, 8, 0, 6, 8, 9, 6

Solution: The MM block is placed in any available cache line

MM Block	Hit	Miss
Block 0		miss

Line 0	Block 0
Line1	
Line2	
Line3	

Cache



Example: Consider a fully associative cache contains 4 (1-word) lines. Find the number of misses in the cache if the following sequence of MM block accesses is generated : 0, 8, 0, 6, 8, 9, 6

Solution: The MM block is placed in any available cache line

MM Block	Hit	Miss
Block 0		miss
Block 8		miss

Line 0	Block 0
Line1	Block 8
Line2	
Line3	

Cache



Example: Consider a fully associative cache contains 4 (1-word) lines. Find the number of misses in the cache if the following sequence of MM block accesses is generated : 0, 8, 0, 6, 8, 9, 6

Solution: The MM block is placed in any available cache line

MM Block	Hit	Miss
Block 0		miss
Block 8		miss
Block 0	hit	

Line 0	Block 0
Line1	Block 8
Line2	
Line3	

Cache



Example: Consider a fully associative cache contains 4 (1-word) lines. Find the number of misses in the cache if the following sequence of MM block accesses is generated : 0, 8, 0, 6, 8, 9, 6

Solution: The MM block is placed in any available cache line

MM Block	Hit	Miss
Block 0		miss
Block 8		miss
Block 0	hit	
Block 6		miss

Line 0	Block 0
Line1	Block 8
Line2	Block 6
Line3	

Cache



Example: Consider a fully associative cache contains 4 (1-word) lines. Find the number of misses in the cache if the following sequence of MM block accesses is generated : 0, 8, 0, 6, 8, 9, 6

Solution: The MM block is placed in any available cache line

MM Block	Hit	Miss
Block 0		miss
Block 8		miss
Block 0	hit	
Block 6		miss
Block 9		miss

Line 0	Block 0
Line1	Block 8
Line2	Block 6
Line3	Block 9

Cache



Example: Consider a fully associative cache contains 4 (1-word) lines. Find the number of misses in the cache if the following sequence of MM block accesses is generated : 0, 8, 0, 6, 8, 9, 6

Solution: The MM block is placed in any available cache line

MM Block	Hit	Miss
0		miss
8		miss
0	hit	
6		miss
8	hit	
9		miss
6	hit	

Line 0 **Block 0**

Line1 **Block 8**

Line2 **Block 6**

Line3 **Block 9**

Cache



3) Set Associative Cache Memory

This mapping technique represents a good compromise between DM and fully associative mapping. It builds on the strengths of both: (namely, the **easy control of the DM cache** and **the more flexible mapping of the fully associative cache**).

In set associative mapping:

- * Cache consists of a number of sets (S)
- * Each set contains a number of lines (X)
- * A given MM block maps (stored) at any line within a given set

The cache is then described in the number of lines each set contains. If a set can hold X lines, the cache is referred to as an X-way set associative cache . A MM block can be stored in any one of the X- lines in a set such that:

cache set number (i):

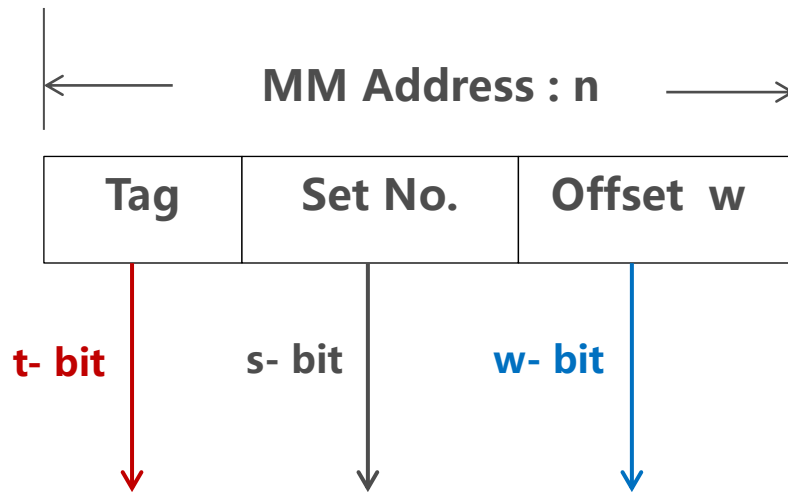
$$i = j \text{ mod } S$$

where j: is the MM block number



Set associative cache interprets a MM address (n) to be comprises of 3 distinct fields:

Tag (t -bit), Set number (s -bit), and word number or offset (w -bit)

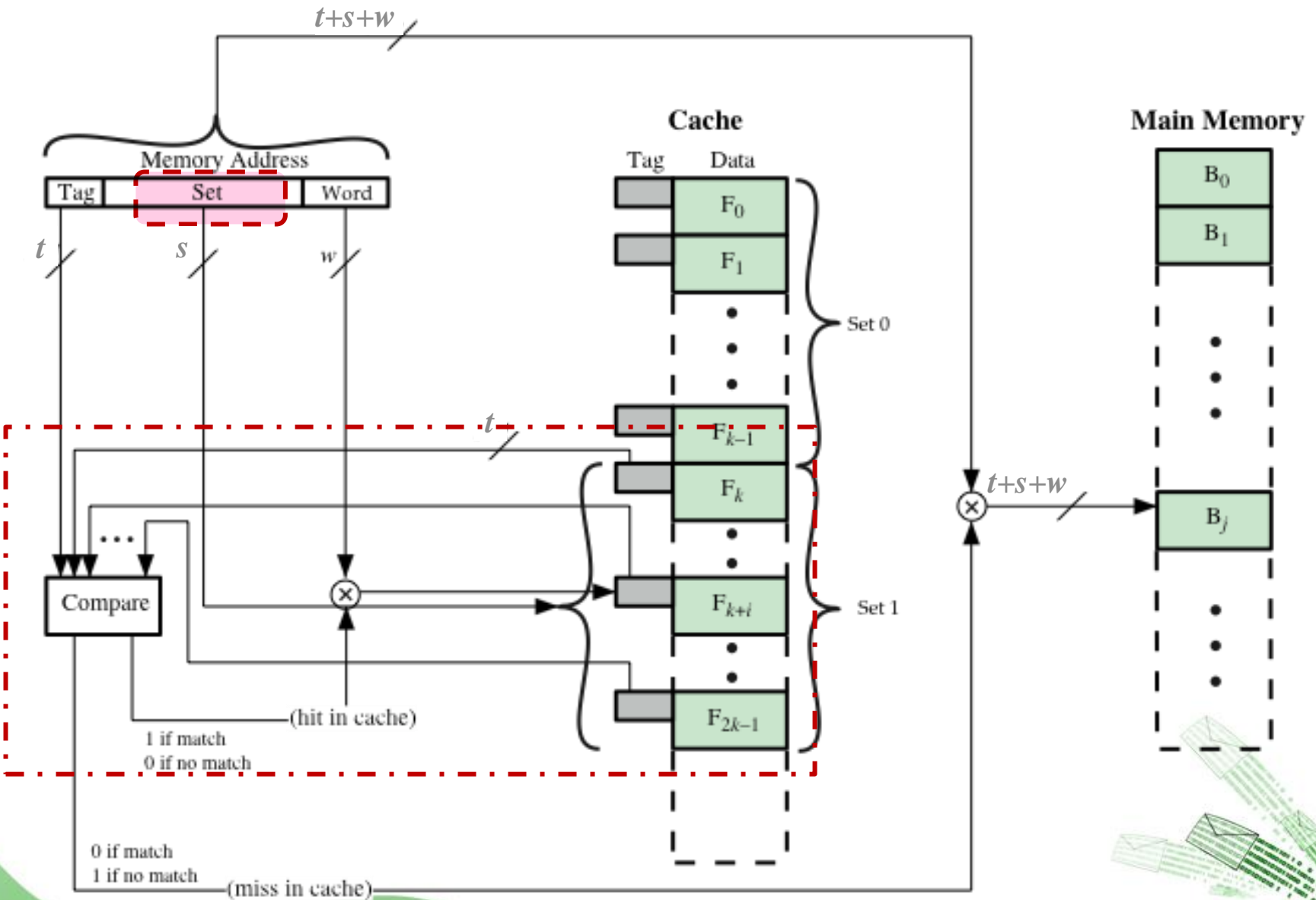


Set Associative Cache
Address

MM Address length (bits): $n = t + s + w$



3) X-Way Set Associative Cache Memory Organization



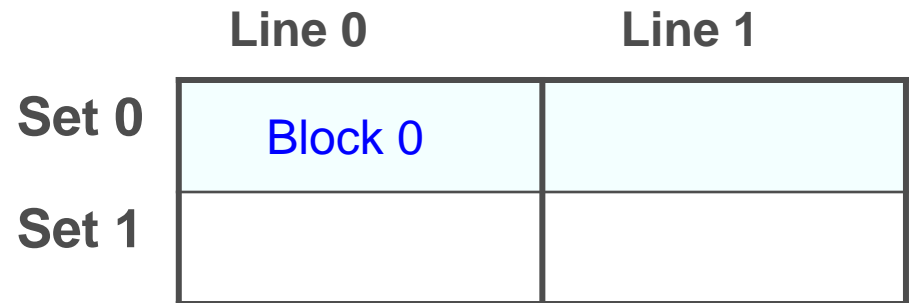
Example:

2-way Set-Associative cache contains 2 sets, 2 one-word blocks each. Find the number of misses in the cache if the following sequence of MM block accesses is generated : 0, 8, 0, 6, 8, 9, 6

Solution: Since the cache at start is always empty , thus this MM block is not in cache.

According to the formula : Mapping: $i = j \bmod S$

MM Block	Hit	Miss
0		miss



Cache



Example:

2-way Set-Associative cache contains 2 sets, 2 one-word blocks each. Find the number of misses in the cache if the following sequence of MM block accesses is generated : 0, 8, 0, 6, 8, 9, 6

Solution: Since the cache at start is always empty , thus this MM block is not in cache.

According to the formula : Mapping: $i = j \bmod S$

MM Block	Hit	Miss
0		miss
8		miss

	Line 0	Line 1
Set 0	Block 0	Block 8
Set 1		

Cache



Example:

2-way Set-Associative cache contains 2 sets, 2 one-word blocks each. Find the number of misses in the cache if the following sequence of MM block accesses is generated : 0, 8, 0, 6, 8, 9, 6

Solution: Since the cache at start is always empty , thus this MM block is not in cache.

According to the formula : Mapping: $i = j \bmod S$

MM Block	Hit	Miss
0		miss
8		miss
0	hit	

	Line 0	Line 1
Set 0	Block 0	Block 8
Set 1		

Cache



Example:

2-way Set-Associative cache contains 2 sets, 2 one-word blocks each. Find the number of misses in the cache if the following sequence of MM block accesses is generated : 0, 8, 0, 6, 8, 9, 6

Solution: Since the cache at start is always empty , thus this MM block is not in cache.

According to the formula : Mapping: $i = j \bmod S$

MM Block	Hit	Miss
0		miss
8		miss
0	hit	
6		miss

	Line 0	Line 1
Set 0	Block 6	Block 8
Set 1		

Cache



Example:

2-way Set-Associative cache contains 2 sets, 2 one-word blocks each. Find the number of misses in the cache if the following sequence of MM block accesses is generated : 0, 8, 0, 6, 8, 9, 6

Solution: Since the cache at start is always empty , thus this MM block is not in cache.

According to the formula : Mapping: $i = j \bmod S$

MM Block	Hit	Miss
0		miss
8		miss
0	hit	
6		miss
8	hit	

	Line 0	Line 1
Set 0	Block 6	Block 8
Set 1		

Cache



Example:

2-way Set-Associative cache contains 2 sets, 2 one-word blocks each. Find the number of misses in the cache if the following sequence of MM block accesses is generated : 0, 8, 0, 6, 8, 9, 6

Solution: Since the cache at start is always empty , thus this MM block is not in cache.

According to the formula : Mapping: $i = j \bmod S$

MM Block	Hit	Miss
0		miss
8		miss
0	hit	
6		miss
8	hit	
9		miss

	Line 0	Line 1
Set 0	Block 6	Block 8
Set 1	Block 9	

Cache



Example:

2-way Set-Associative cache contains 2 sets, 2 one-word blocks each. Find the number of misses in the cache if the following sequence of MM block accesses is generated : 0, 8, 0, 6, 8, 9, 6

Solution: Since the cache at start is always empty , thus this MM block is not in cache.

According to the formula : Mapping: $i = j \text{ mod } S$

MM Block	Hit	Miss
0		miss
8		miss
0	hit	
6		miss
8	hit	
9		miss
6	hit	

	Line 0	Line 1
Set 0	Block 6	Block 8
Set 1	Block 9	

Cache

