

COMPUTER SCIENCE

Computer Organization and Architecture

Cache Memory

Lecture_06



Vijay Agarwal sir





TOPICS
TO BE
COVERED

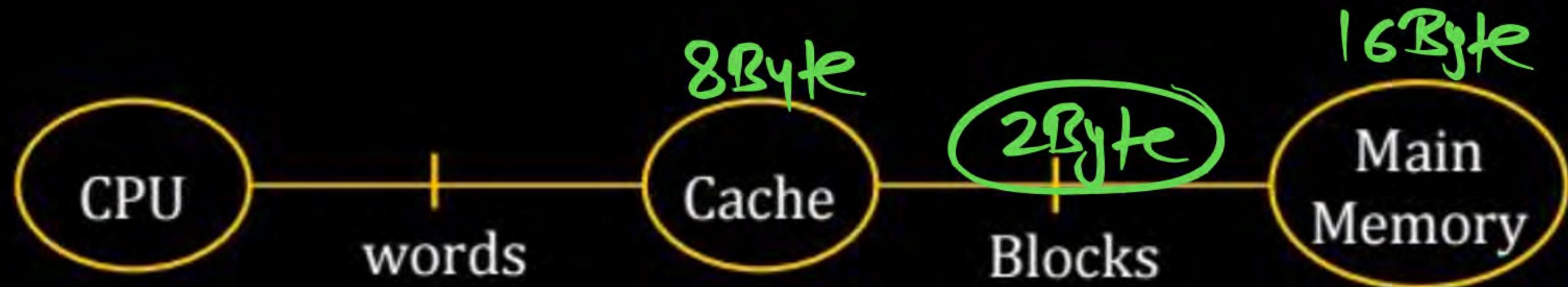
- 
- 01 Mapping Techniques
- 
- 02 Replacement Algo &
Updating Technique



Mapping Technique:

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

Memory Organization



Mapping

The process of transfer the Data from Main Memory to Cache

Memory is called mapping. There are 3 Type of Mapping

Technique

1) Direct Mapping

2) Set Associative Mapping

3) Fully Associative Mapping

Mapping Function

- ❑ Because there are fewer cache lines than main memory blocks, an algorithm, is needed for mapping main memory blocks into cache lines.
- ❑ Three techniques can be used:

①

Direct

- The simplest technique
- Maps each block of main memory into only one possible cache lines.

Associative

- permits each main memory block to be loaded into any line of the cache.
- The cache control logic interprets a memory address simply as a Tag a word field
- To determine whether a block is in the cache, the cache control logic must simultaneously examine every line's Tag for a match

②

Set Associative

- A compromise that exhibits the strength of both the direct and associative approaches while reducing their disadvantage.

① Direct Mapping

In this Direct Cache Controller interprets the CPU generated Request as follows:



$$\# \text{LINE} = \frac{\text{CM Size}}{\text{BLOCK Size}}$$

$$\text{TAG} = \text{Physical Address} - (\text{Line offset} + \text{Word offset})$$

$$\text{TAG Memory Size} = \# \text{LINE}'s \times \text{Tag bits} \text{ (Depend on the mapping technique)}$$

1) Direct Mapping

In this Technique mapping function is used to transfer the data from Main Memory to Cache Memory. The Mapping Function is

$$\text{Cache address} = \text{Main Memory request} \quad \text{MOD} \quad \# \text{ CM LINES}$$

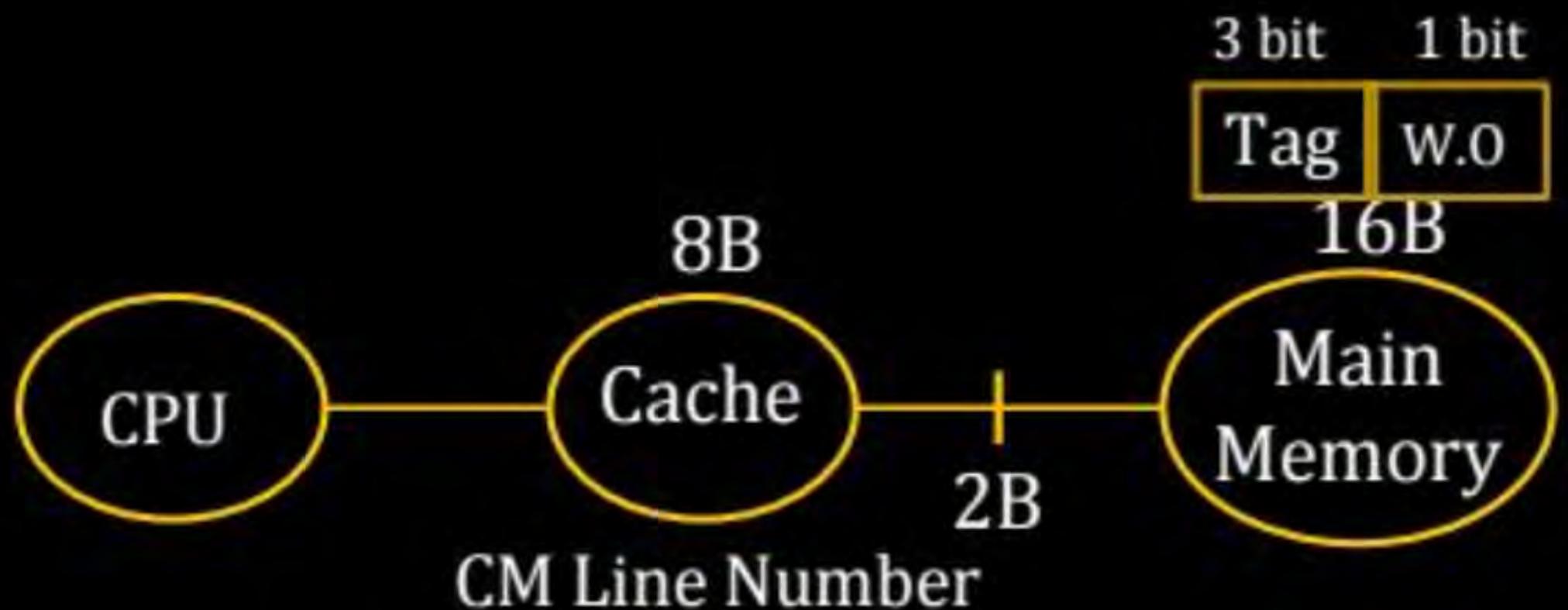
(Or)

$$K \text{ MOD } N = i$$

K: MM Block No.

N: # of Cache Line

i: CM Line Number



$$0 \bmod 4 = 0$$

$$1 \bmod 4 = 1$$

$$2 \bmod 4 = 2$$

$$3 \bmod 4 = 3$$

$$4 \bmod 4 = 0$$

$$5 \bmod 4 = 1$$

$$6 \bmod 4 = 2$$

$$7 \bmod 4 = 3$$

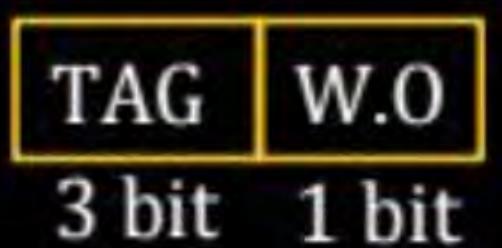
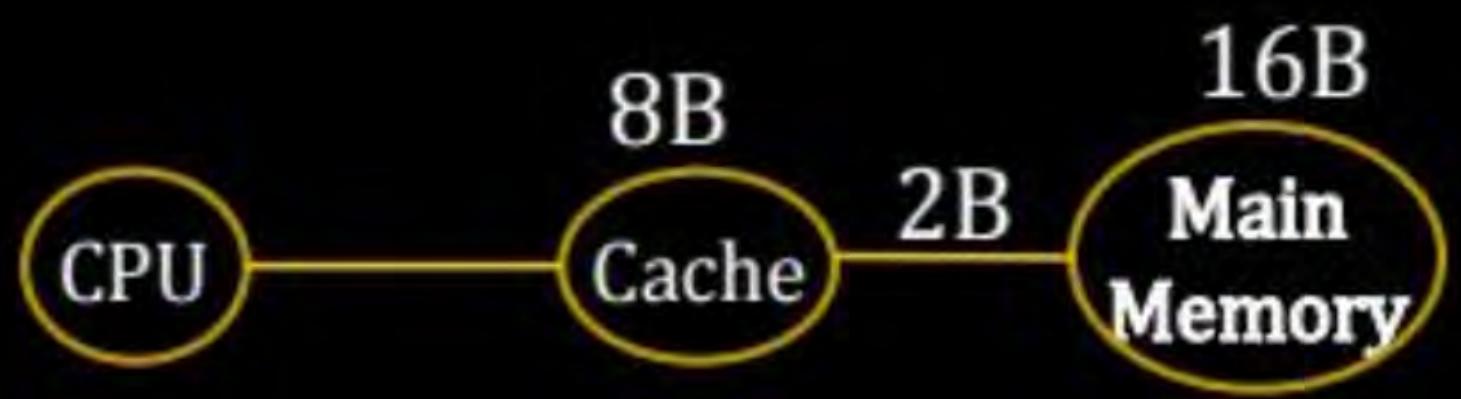
CM

B_0 & B_4 : LINE 0

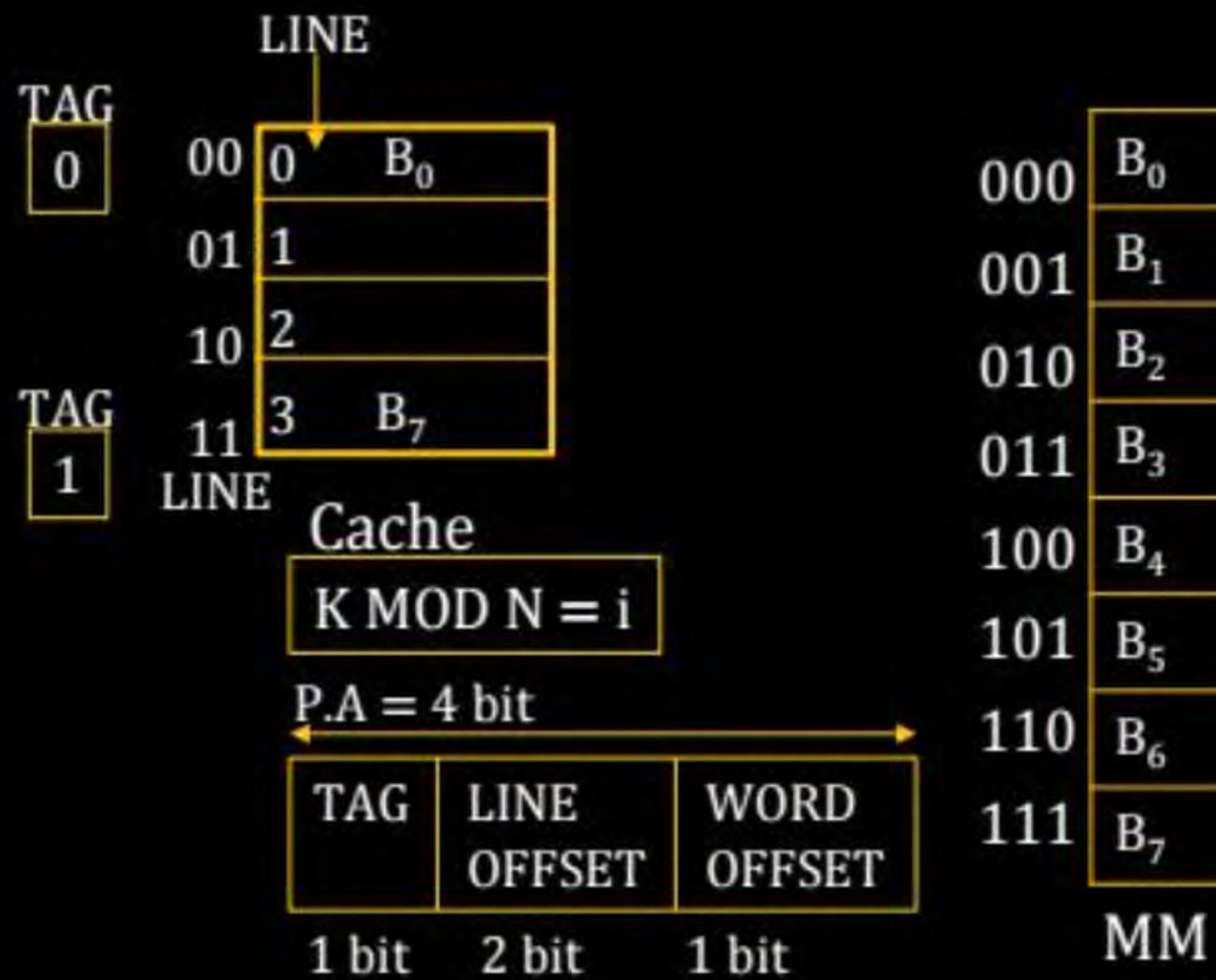
B_4 & B_5 : LINE 1

B_2 & B_6 : LINE 2

B_3 & B_7 : LINE 3



P
W



Direct Mapping

MM Block

TAG LINE

0	00
---	----

 $B_0_{[000]}$

Direct Mapping

$$\frac{K \bmod N = i}{0 \bmod 4 = '0'}$$

CM LINE

LINE '0'

TAG LINE

1	11
---	----

 $B_7_{[111]}$

$$\frac{K \bmod N = i}{7 \bmod 4 = '3'}$$

LINE '3'

Tag Memory Size = #LINE's × Tag bits

Depends On the Mapping technique

In the above example: # LINE = 4
Tag bit = 1 bit
(Direct Mapping)

Tag Memory Size = $4 \times 1 = 4$ bits



Consider the following program

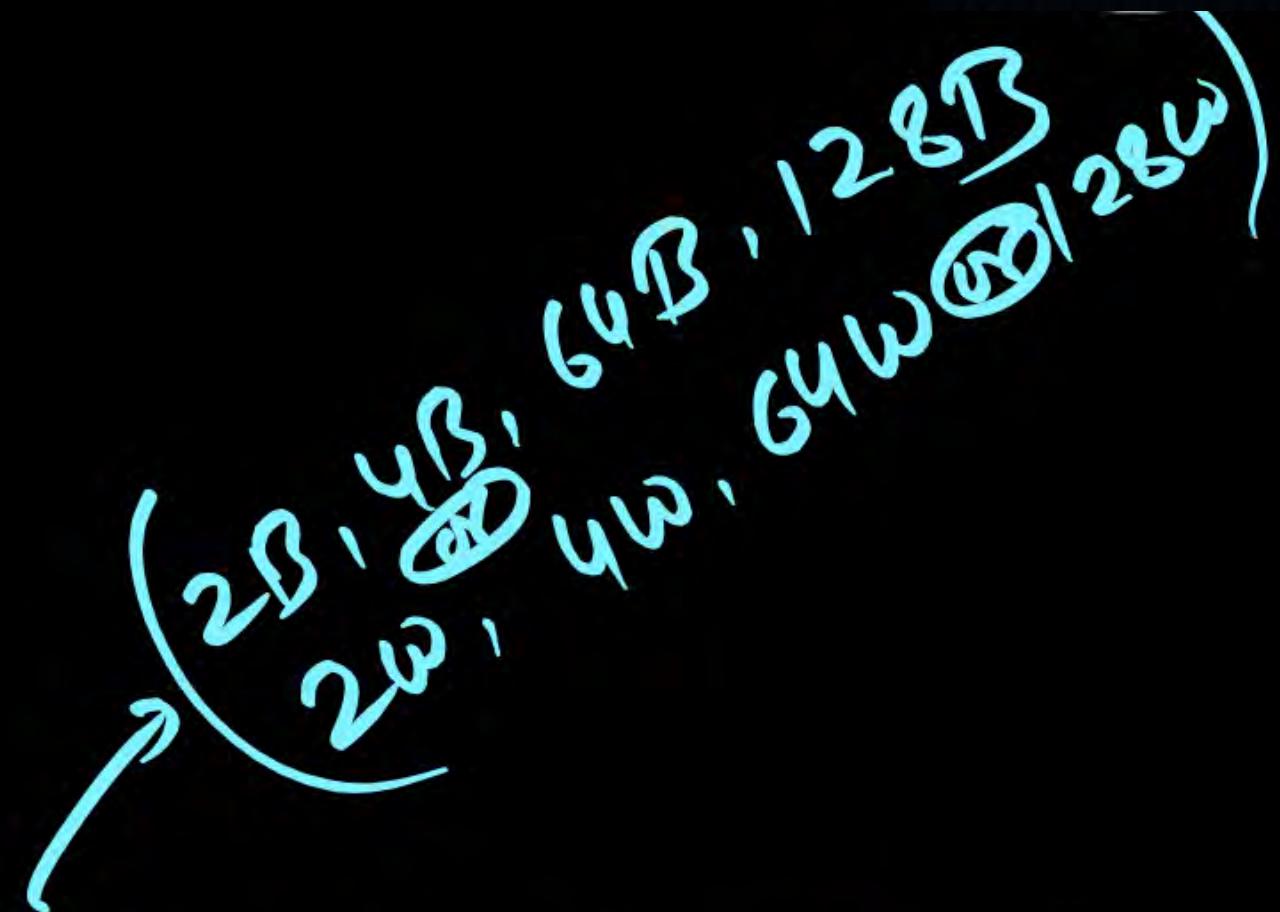
I₁: MOV r₀ [0000]

I₂: MOV r₁ [1000]

I₃: ADD r₀r₁

Cache Hit & Cache Miss.

Note In a Mapping Process 1 Complete Block is transferred from MM to CM But Based On Word offset that Repetitive word/Byte given from CM to CPU.



LOR [Locality of Reference]

Consider the following program

I₁: MOV r₀ [0000]

I₂: MOV r₁ [0001]

I₃: MOV r₂ [1000]

I₄: MOV r₃ [1001]

already in Morning class.

LOR [Locality of Reference]

Consider the following program

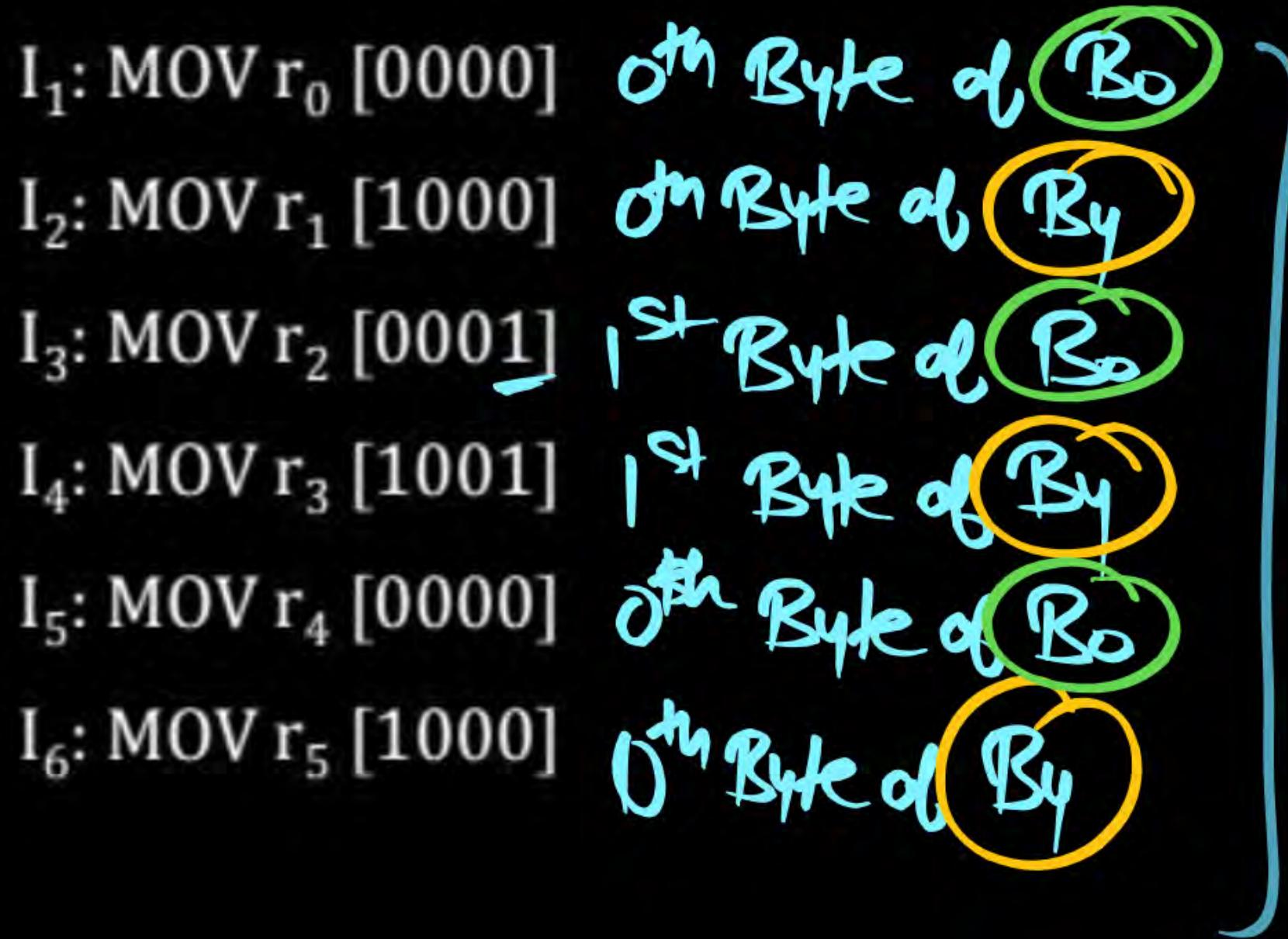
I₁: MOV r₀ [0000]

I₂: MOV r₁ [0001]

I₃: MOV r₂ [1000]

I₄: MOV r₃ [1001]

Disadvantage of Direct Mapping



Disadvantage of Direct Mapping

I₁: MOV r₀ [0000]

I₂: MOV r₁ [1000]

I₃: MOV r₂ [0001]

I₄: MOV r₃ [1001]

I₅: MOV r₄ [0000]

I₆: MOV r₅ [1000]

Set Associative Mapping function

Cache Set = MM Request MOD #SET is in Cache Address

(OR)

$$K \bmod S = i$$

k: MM Block Number

s: # Cache Set

i: Cache Set Number

$$k \bmod s = i$$

Q

What is meaning of

Q

Tb # LINES = 16

- (i) 2 way Set Associative
- (ii) 4 way Set Associative
- (iii) 8 Way Set Associative
- (iv) N-Way Set Associative.

Example1: # Line's = 16 & 2 way set Associative

$$\# \text{SET} = \frac{\# \text{LINES}}{\text{N-way}} = \frac{16}{2} = 8 \quad \boxed{\# \text{SET} = 8} \quad S = 8$$

$$K \bmod 8 = i$$

✓	✓	✓	✓	✓	✓	✓	✓
---	---	---	---	---	---	---	---

SET 0

SET 1

2

3

4

5

SET 6

SET 7

$$K \bmod 8 = i \quad 0 - 7$$

Example2:

#LINE = 16 & 4 way set Associative

$$\# \text{SET} = \frac{16}{4} \Rightarrow 4$$

S = 4

K MOD 4 = i

K MOD S = j

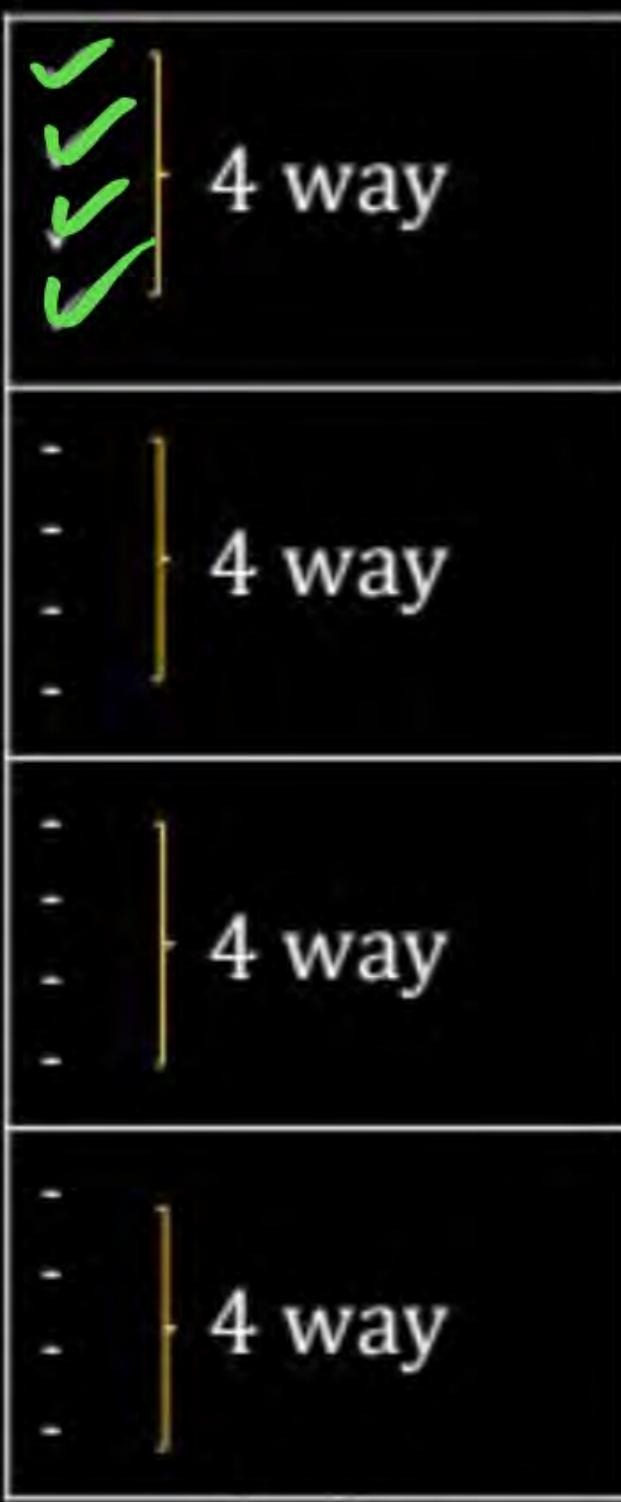
SET 0

SET 1

SET 2

SET 3

Cache



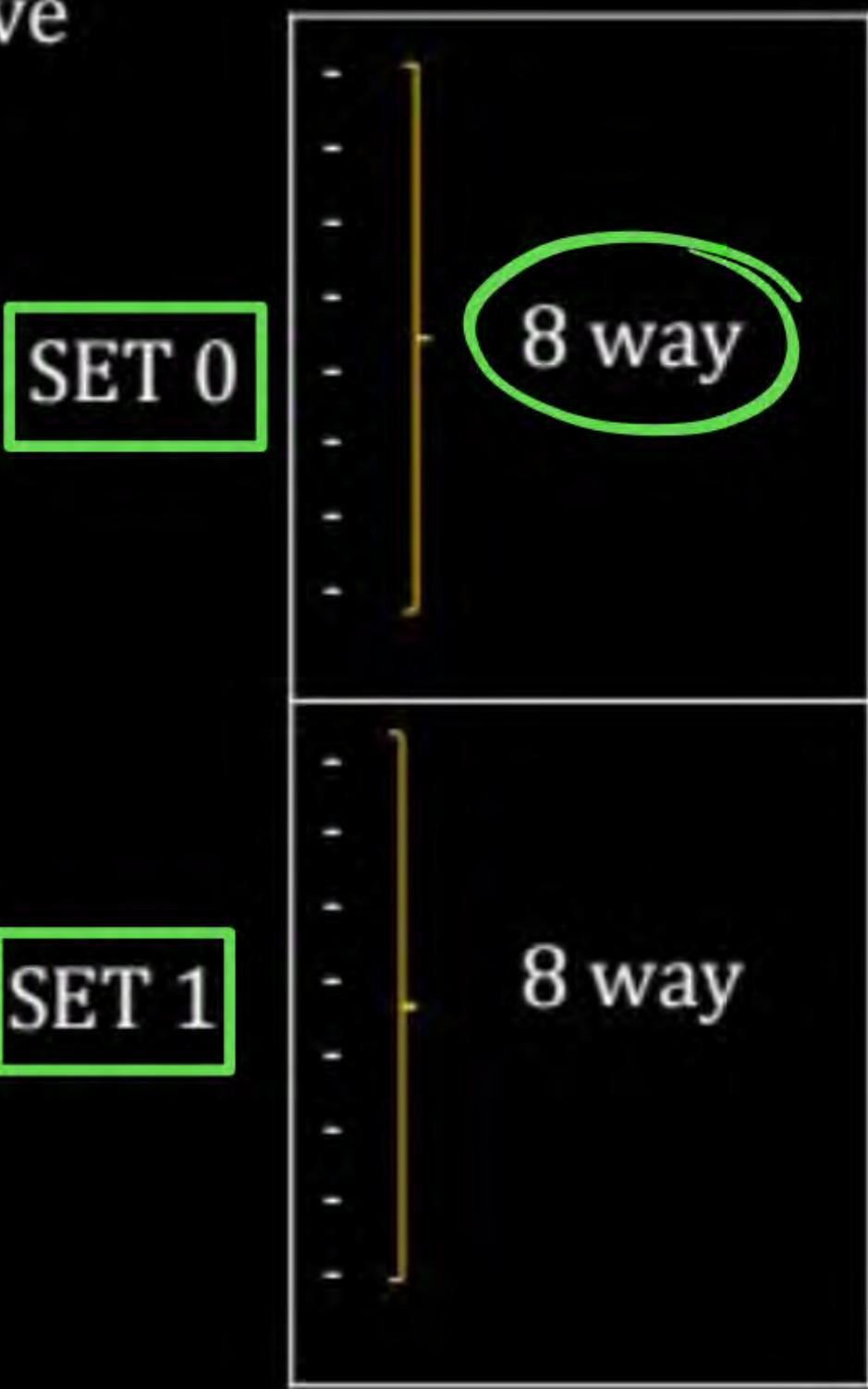
Example3:

#LINE = 16 & 8 way set Associative

$$\# \text{ SET} = \frac{16}{8} \Rightarrow 2 \quad S = 2$$

$$K \bmod 2 = i$$

$$k \bmod S = j$$



Example4:

#LINE = 16 & 16 way set Associative

$$\# \text{ SET} = \frac{16}{16} \Rightarrow 1$$

$$S = 1$$

$$K \bmod 1 = i$$



Fully
Associative
(Whole cache as a set)

Set Associative Mapping function

Cache

Set = MM Request MOD #SET is in Cache
Address

(OR)

K MOD S = i

k: MM Block Number

s: # Cache Set

i: Cache Set Number

Set Associative Mapping function

Cache

Set = MM Request MOD #SET is in Cache

Address

(OR)

K MOD S = i

k: MM Block Number

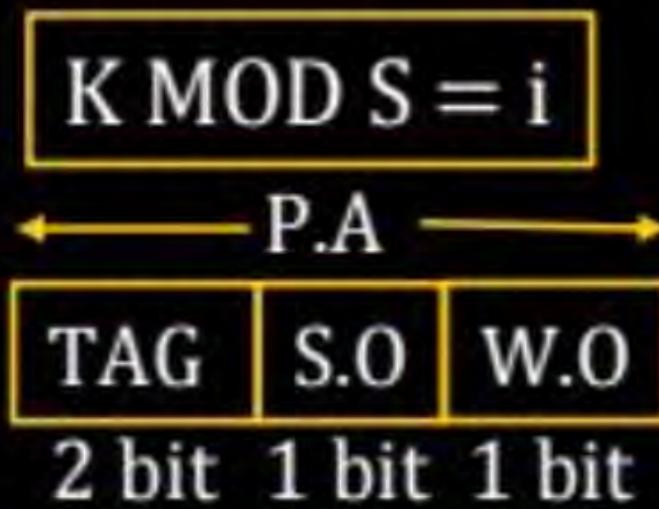
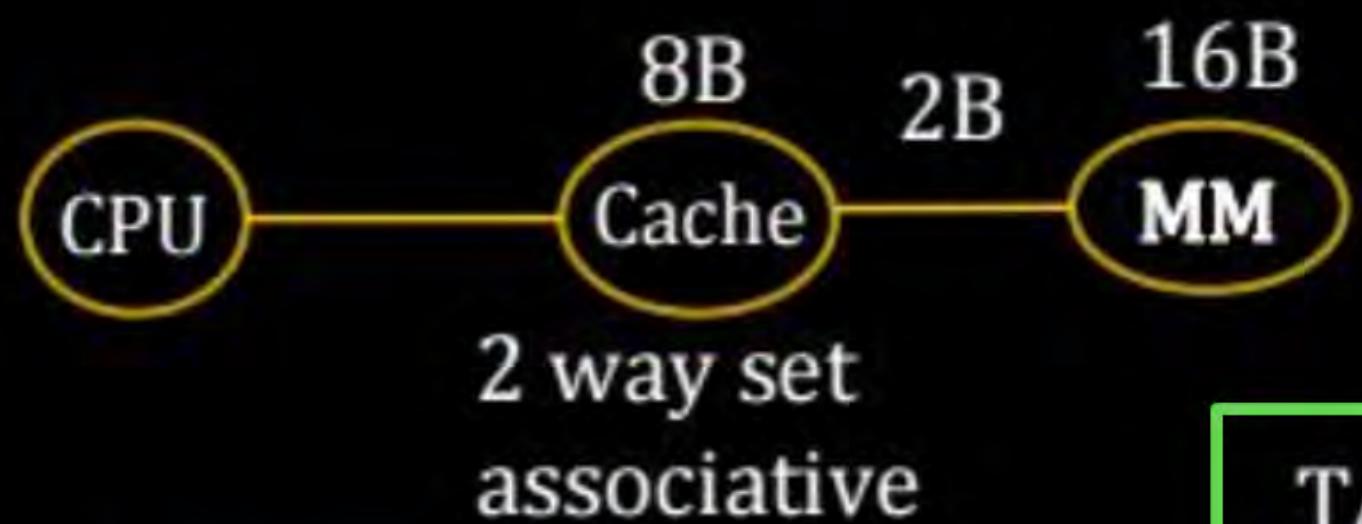
s: # Cache Set

i: Cache Set Number

MM BLOCK Mapping Tech. CM SET

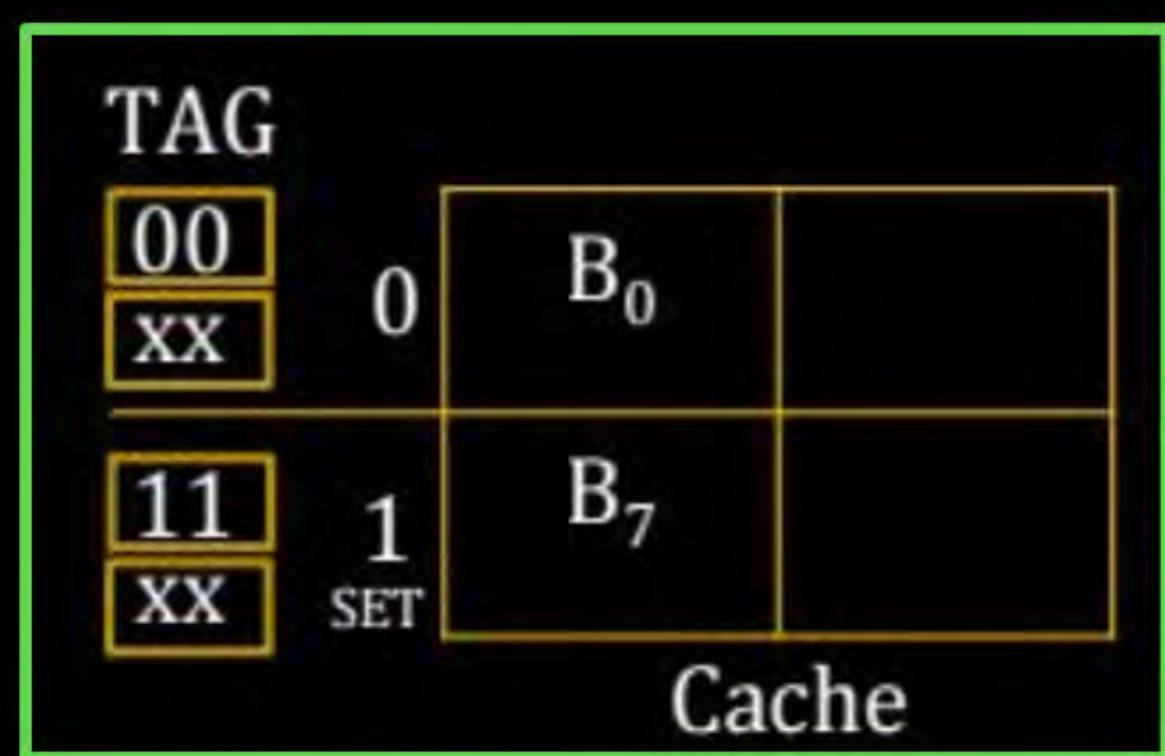
$$\begin{array}{l} K \bmod S = i \\ \hline B_0[000] \quad \xrightarrow{\hspace{1cm}} \text{SET '0'} \\ 0 \bmod 2 = '0' \end{array}$$

$$\begin{array}{l} K \bmod S = i \\ \hline B_7[111] \quad \xrightarrow{\hspace{1cm}} \text{SET '1'} \\ 7 \bmod 2 = '1' \end{array}$$



$B_0[000] \rightarrow$ TAG S.O
2 bit 1 bit → SET '0'

$B_7[111] \rightarrow$ TAG S.O
2 bit 1 bit → SET '1'



P W

000	B_0
001	B_1
010	B_2
011	B_3
100	B_4
101	B_5
110	B_6
111	B_7

Q) How to Identify Cache Hit & Miss in Set Associative Mapping.

Practically ?

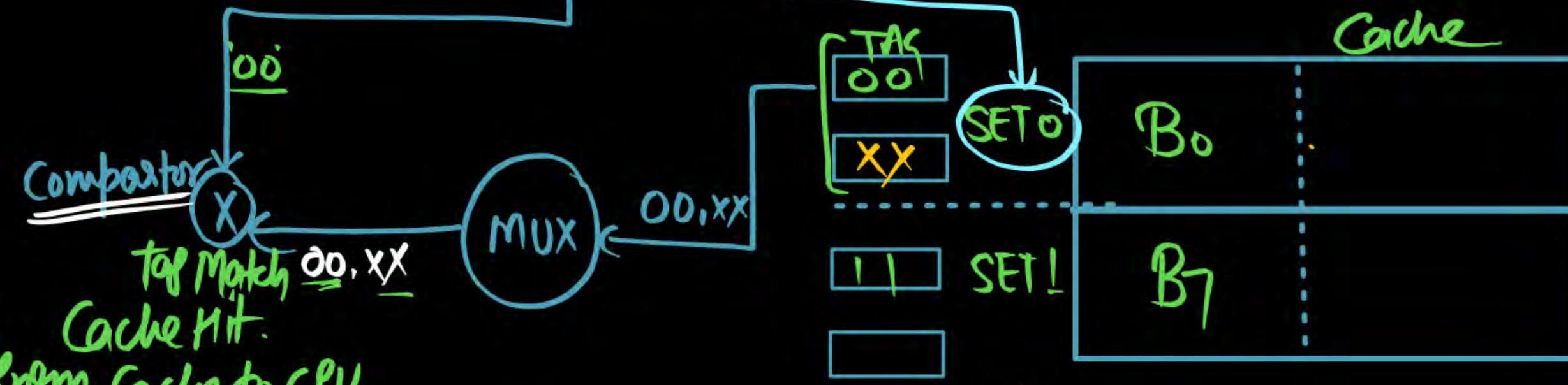
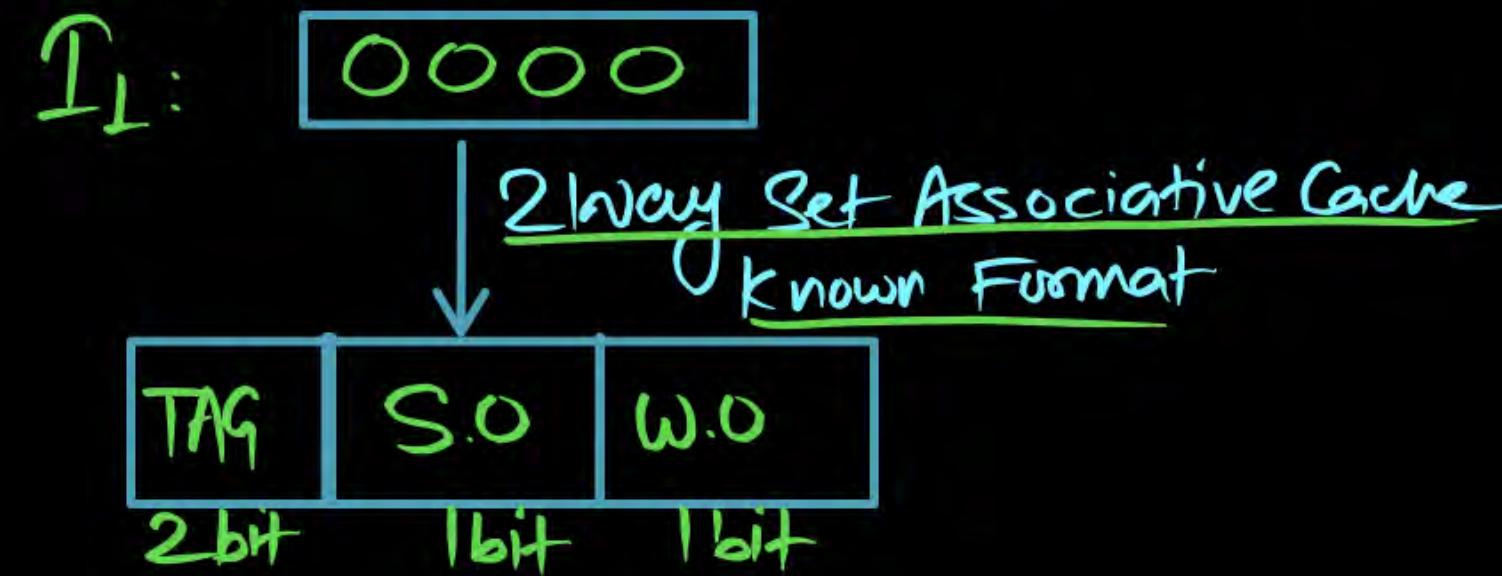
Consider the following program Π : CPU generated Request ^{first} Refer to Cache

$I_1: \text{MOV } r_0 [0000]$

$I_2: \text{MOV } r_1 [1000]$

$I_3: \text{ADD } r_0 r_1$

Π Any of the tag Match then Cache Hit.



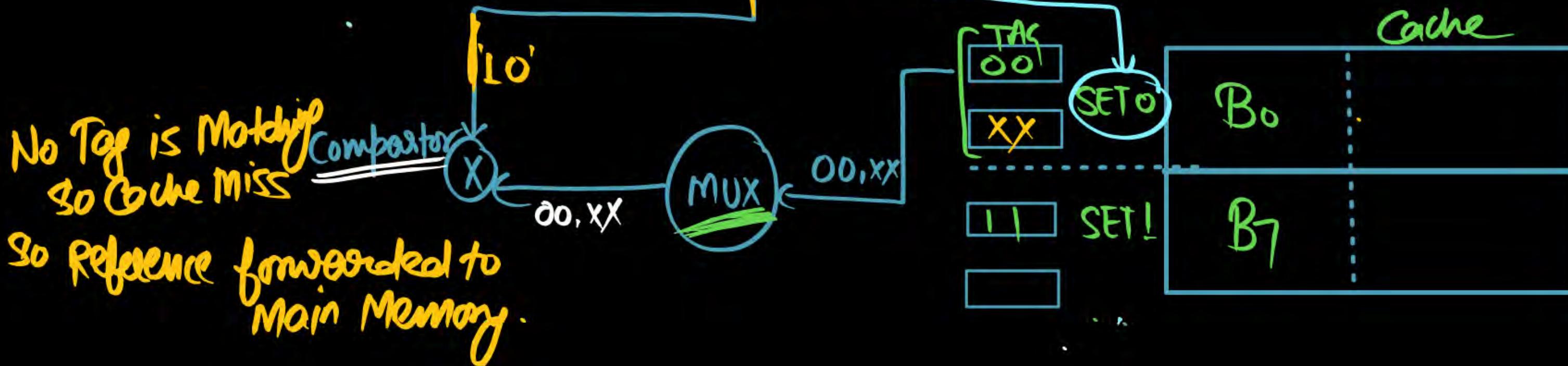
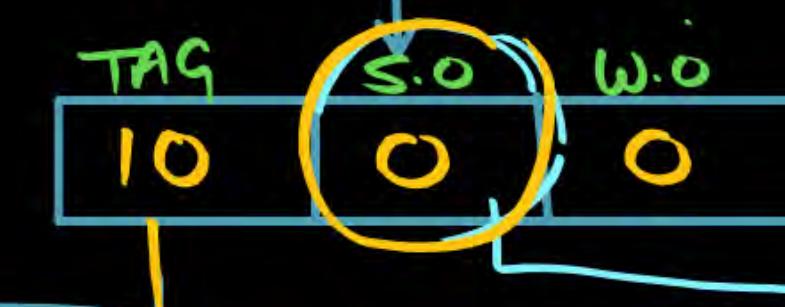
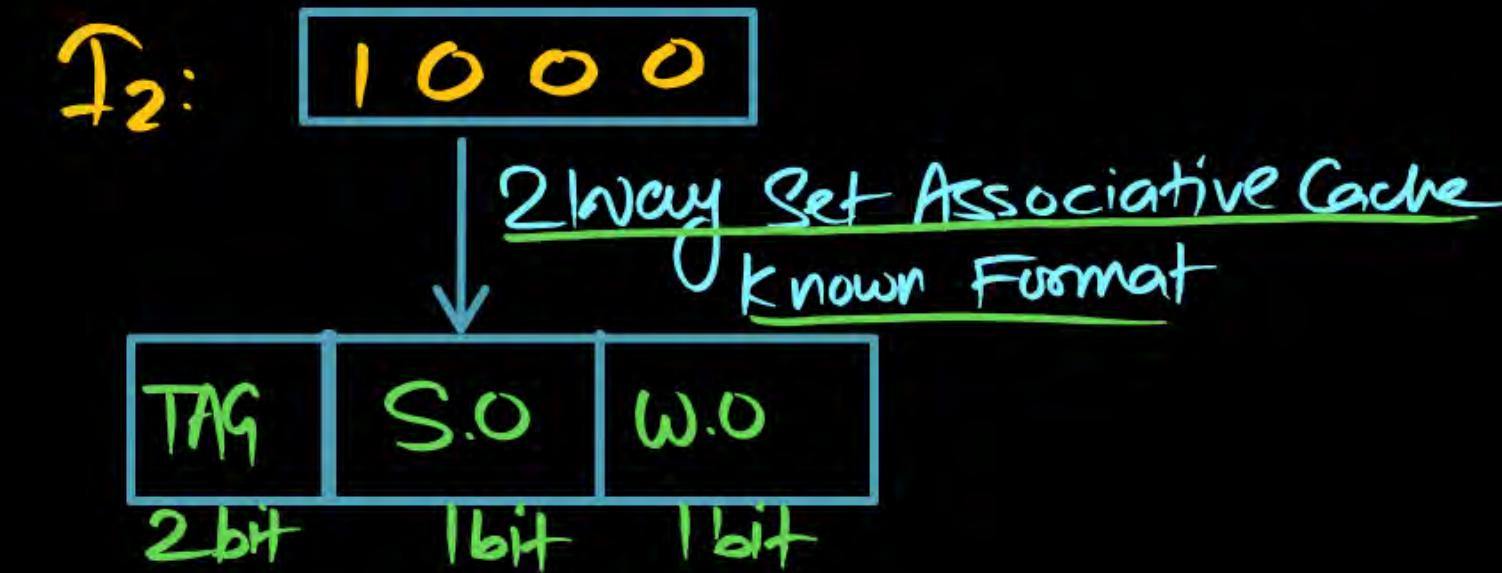
Respective Data given from Cache to CPU.
Cache Hit.

Consider the following program

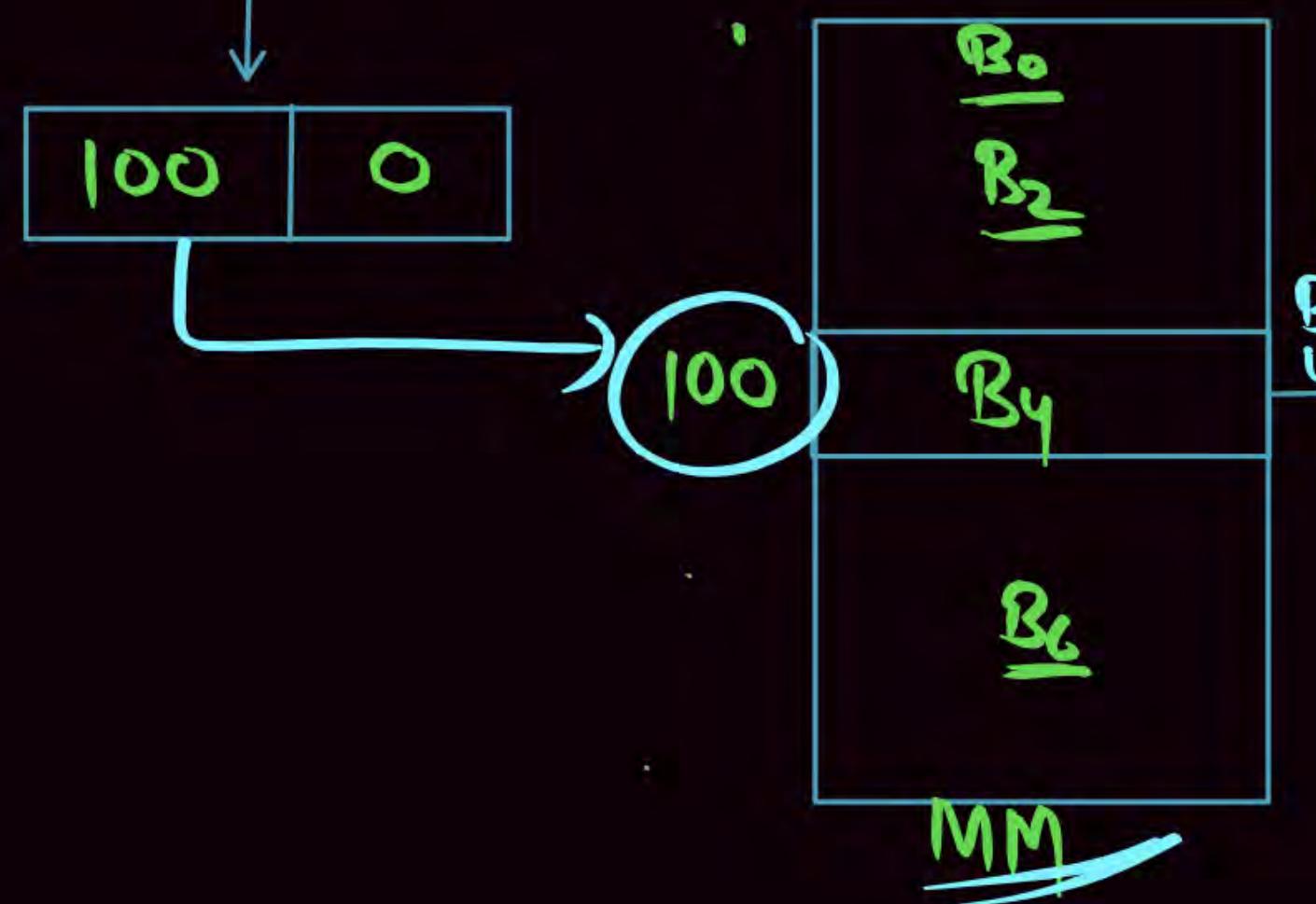
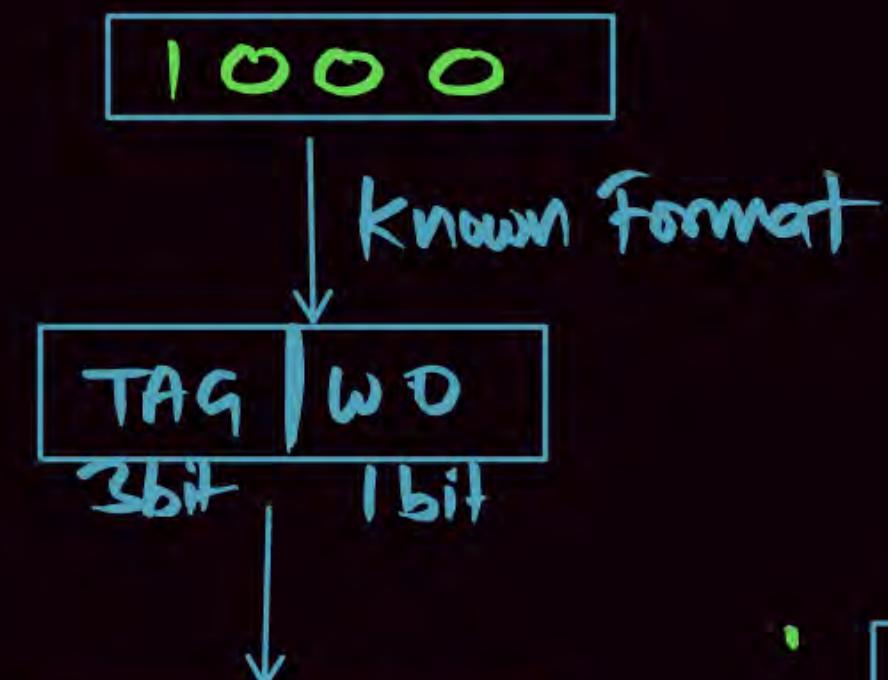
I₁: MOV r₀ [0000]

I₂: MOV r₁ [1000]

I₃: ADD r₀r₁



Reference forwarded to Main Memory



B₄ [100]

10	10
TAG.	S.O.

TAG S.O (W.O)
2bit 1bit 1bit

Now Cache Set Full
Replacement Algo
Required Along with
Mapping function.



$$\frac{\text{Tag Memory Size}}{\text{#SETS} \times \frac{\text{\#LINES In a each set}}{\text{Tag bits}}}$$

Example: # SET = 2 $\frac{\text{\# LINE in Each set}}{= 2}$ TAG = 2 bit

$$\frac{\text{Tag Memory size}}{= 2 \times 2 \times 2 \\ = 8 \text{ bits}}$$

1) In direct Mapping

Hit Latency = Latency of Tag comparator

2) In Set Associative Mapping

Hit Latency = Latency of Tag comparator + Latency of Multiplexer

Set Associative Mapping:

$$\#SET = \frac{\#UNES}{N\text{-Way}}$$

Direct Mapping :

Number of Comparator = 1

Size of Comparator = #bits in Tag (Tag bits)

Set Associative Mapping : (N-way, 2-way 4-way 8-way)

• Number of Comparator = N-way

• Size of Comparator = #bits in Tag (Tag bits).

Important Points About Set Associative

#LINE = L & N way set Associative

$$\# \text{SET}[S] = \frac{L}{N}$$

$$\# \text{SET} = \frac{\# \text{LINES}}{N\text{-way}}$$

- (i) If N=1; Direct Mapping
- (ii) If N=L ; Associative Mapping; (S=1)ie Only 1 Set

$$K \bmod S = i$$

k: MM Block Number
s: # Cache Set
i: Cache Set Number

Important Points About Set Associative

#LINE = L & N way set Associative

$$\# \text{SET}[S] = \frac{L}{N}$$

If N=1; Direct Mapping

If N=L; Associative Mapping; (S=1)ie Only 1 Set;

Set Associative Mapping Follows Direct Mapping as well as Set Associative

Mapping. Using Mapping Function as:

$$K \bmod S = i$$

within the Set MM Block Can be placed anywhere.

Note

In the Set Associative Mapping Replacement Algorithm is required along with the Mapping function when cache set is Full.

Important Points About Set Associative



#LINE = L & N way set Associative

$$\# \text{SET}[S] = \frac{L}{N}$$

If N=1; Direct Mapping

If N=L ; Associative Mapping; (S=1)ie Only 1 Set.

Tag Bits in N way Set Associative Mapping:

$$\text{Tag bits in } N^{\text{way}} \text{ Set Associative} = \text{Tag bits in Direct Mapping} + \log_2 N$$

N: N Way Set Associative

Only In Direct Mapping

$$\text{Tag} = \frac{\text{MM Size}}{\text{CM Size}}$$

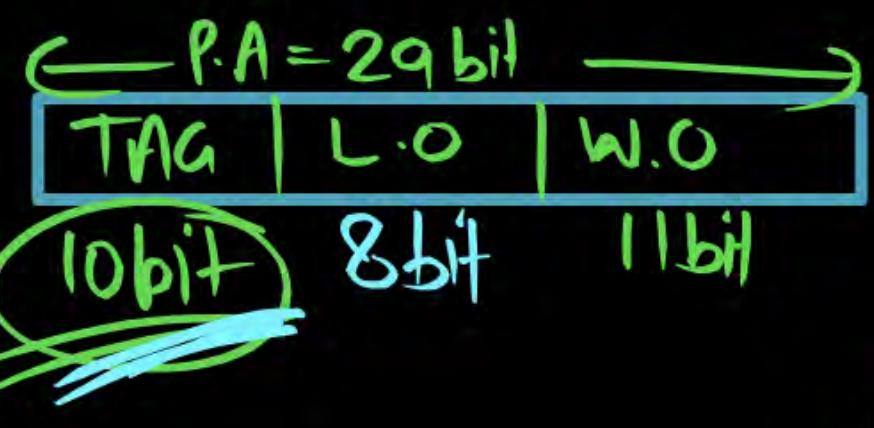
$$\text{Tag bit} = \lceil \log_2 \# \text{Tag} \rceil$$

Important Points About Set Associative

Tag bits in N Set Associative = Tag bits in Direct Mapping + $\log_2 N$

Eg. Consider a **Direct Mapping** if the size of cache memory is 512KB & Main Memory 512MB & Cache line size is 2KB then calculate the Number of bit Required for TAG?

$$\# \text{LINES} = \frac{\text{CM Size}}{\text{Line(Block)Size}} \Rightarrow \frac{512\text{kB}}{2\text{kB}} \Rightarrow 256 [2^8]$$



L.O = 8 bit

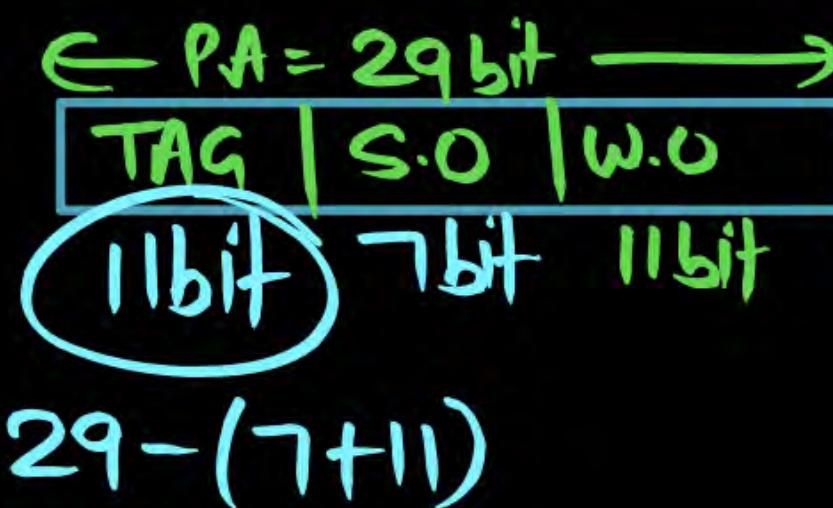
OR

$$\text{TAG} = \frac{\text{MM Size}}{\text{CM Size}} \Rightarrow \frac{512\text{MB}}{512\text{kB}} = 10 \text{ bit}$$

Important Points About Set Associative

Tag bits in N Set Associative = Tag bits in Direct Mapping + $\log_2 N$

Eg. Consider a **2-way set associative** if the size of cache memory is 512KB & Main Memory 512MB & Cache line size is 2KB then calculate the Number of bit Required for TAG?



$$\boxed{\text{TAG} = 11 \text{bit}}$$

$$\begin{aligned}\# \text{LINES} &= 2^8 \\ \# \text{SET} &= \frac{2^8}{2} = 2^7 \\ \text{S.O} &= 7 \text{bit}\end{aligned}$$

2 way Set Associative

$$\begin{aligned}\text{Tag bit} &= \text{Tag bit in Direct Map} + [\log_2 N\text{-Way}] \\ &\Rightarrow 10 + [\log_2 2] \\ &= 10 + 1\end{aligned}$$

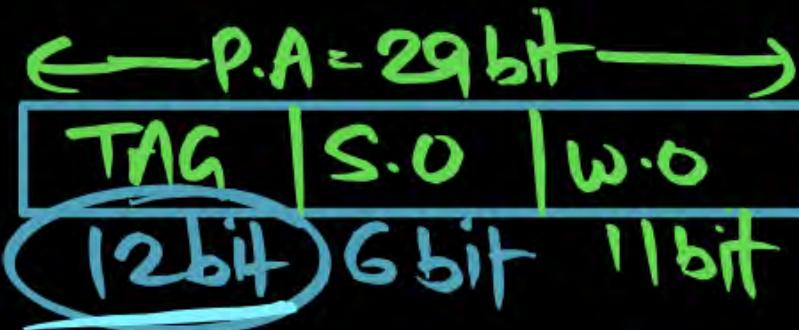
$$\boxed{\text{Tag} = 11 \text{bit}}$$

Important Points About Set Associative

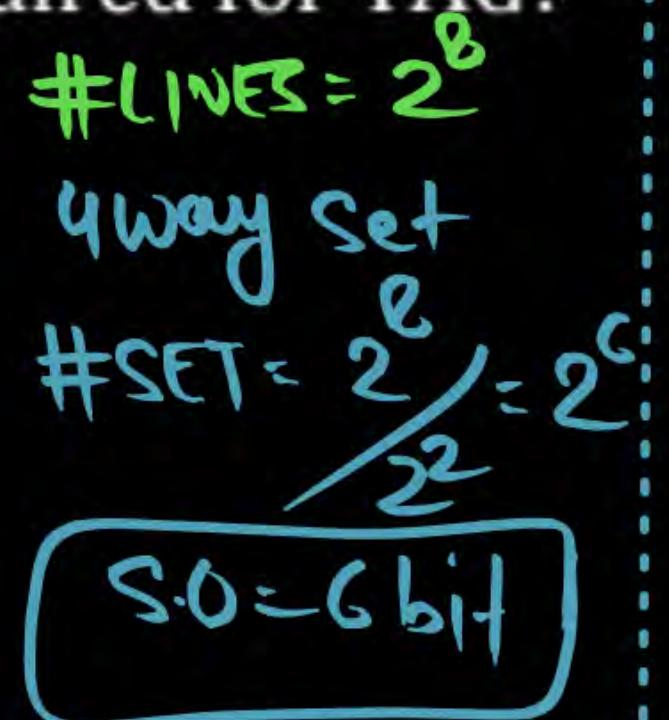


Tag bits in N Set Associative = Tag bits in Direct Mapping + $\log_2 N$

Eg. Consider a **4-way set associative** if the size of cache memory is 512KB & Main Memory 512MB & Cache line size is 2KB then calculate the Number of bit Required for TAG?



$$\begin{aligned} \text{TAG} &= 29 - (6+11) \\ &= 12 \text{ bit} \end{aligned}$$



4 Way Set Associative.

$$\begin{aligned} \text{Tag bit} &= 10 + \lceil \log_2 4 \rceil \\ &= 10 + 2 \end{aligned}$$

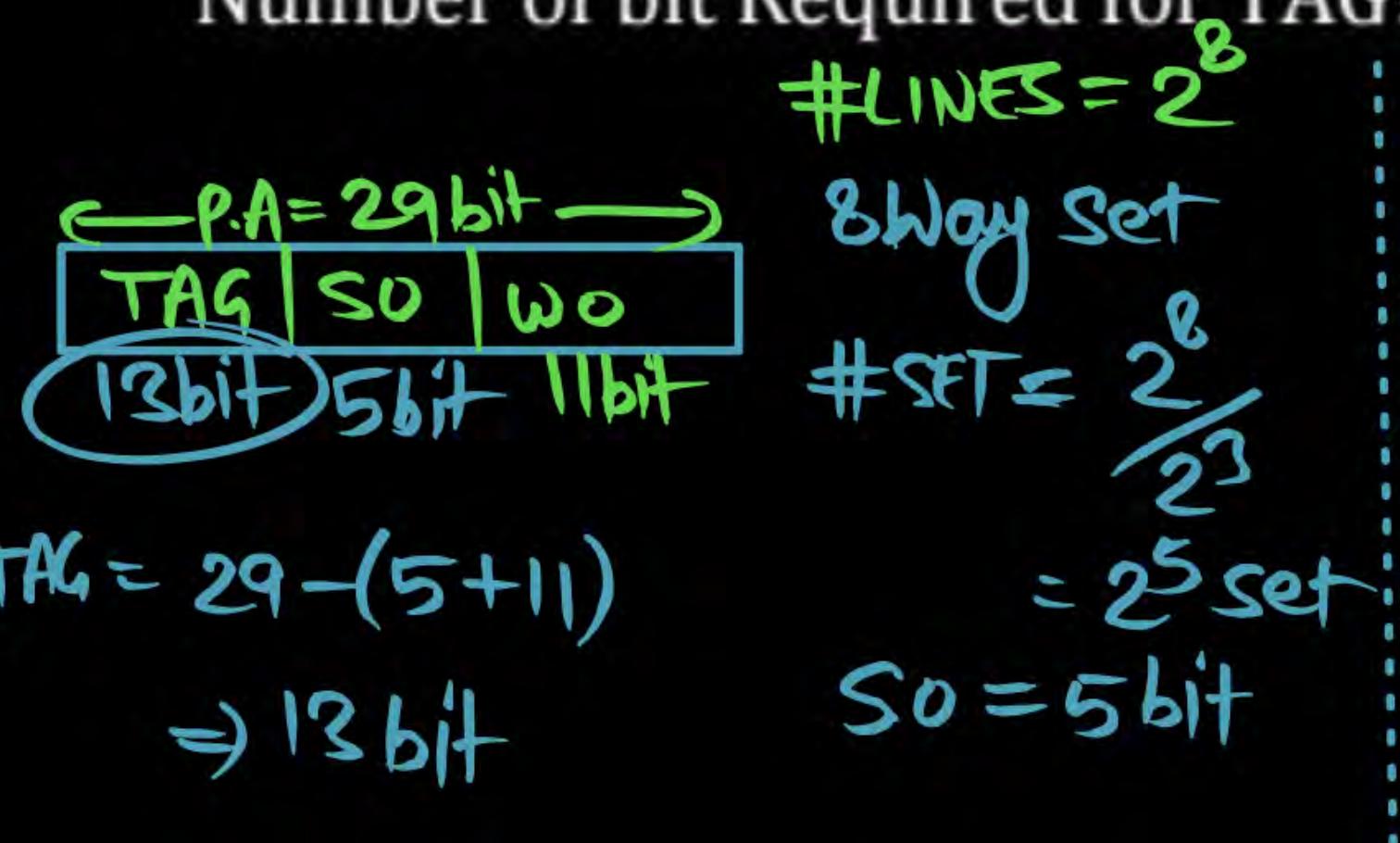
Tag = 12 bit

Avg

Important Points About Set Associative

Tag bits in N Set Associative = Tag bits in Direct Mapping + $\log_2 N$

Eg. Consider a **8-way set associative** if the size of cache memory is 512KB & Main Memory 512MB & Cache line size is 2KB then calculate the Number of bit Required for TAG?



8 Way Set Associative

$$\begin{aligned} \text{Tag bit} &= 10 + \lceil \log_2 8 \rceil \\ &= 10 + 3 \end{aligned}$$

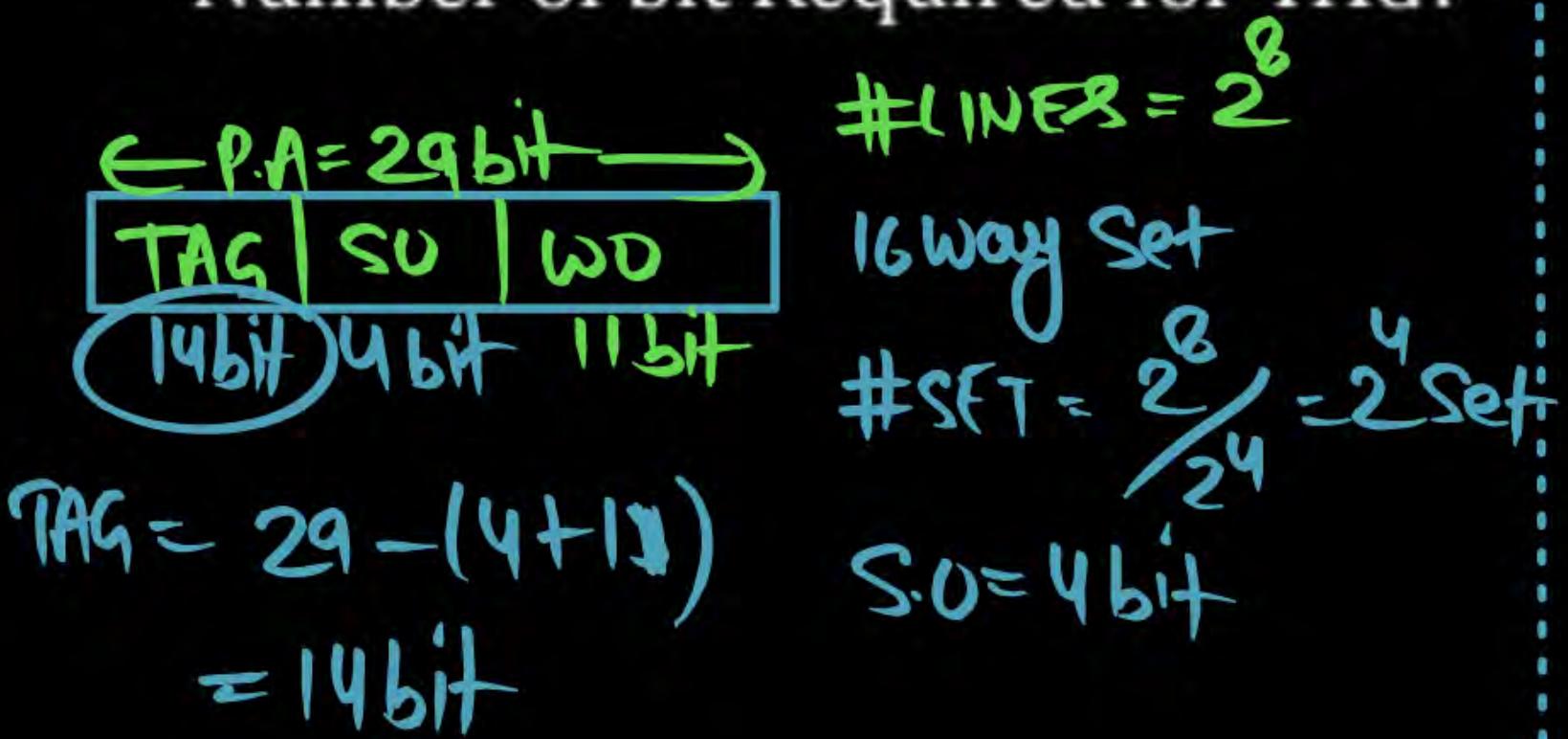
Tag bit = 13 Ans

Important Points About Set Associative



Tag bits in N Set Associative = Tag bits in Direct Mapping + $\log_2 N$

Eg. Consider a **16-way set associative** if the size of cache memory is 512KB & Main Memory 512MB & Cache line size is 2KB then calculate the Number of bit Required for TAG?



16 Way Set Associative

$$\begin{aligned} \text{Tag bit} &= 10 + [\log_2 16] \\ &= 10 + 4 \end{aligned}$$

Tag bit = 14 bit Ans

Till Now :

83 Pipeline
Cache

83 GATE P4Q

+ 29 Today GATE P4Q

(111+) GATE P4Q till Cache
memory.

Fully Associative / Associative Mapping

- No Mapping function is used to transfer the Data from MM to CM.

Note

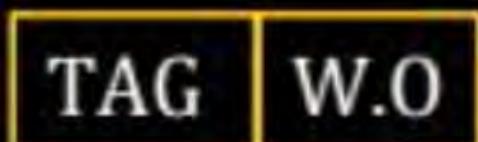
In Associative Number of Comparator = Number of CM Blocks (#CM Lines)

Note

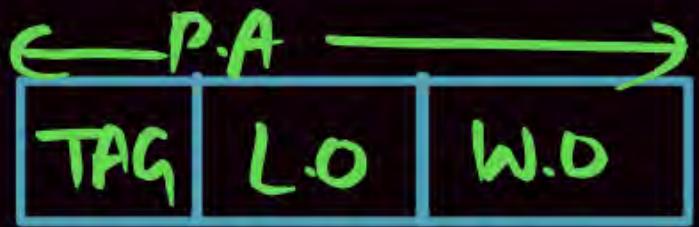
Size of Comparator = # Tag bits.

3) Associative Mapping

- ❑ In this no mapping function is used to transfer the data from MM to CM.
- ❑ Any Block of Main Memory can be placed Anywhere in Cache Memory.
- ❑ No Conflict Miss;
- ❑ This Cache is designed without address called as Content Addressable Memory.
- ❑ In Associative Mapping More Tag bits is required & More Tag Memory size.
- ❑ In Associative Mapping More Hardware Required, Expensive(N Tag Comparator, Here N is Number of Lines., For Each line Comparator Required)
- ❑ In Associative Cache Design, Counter Sequence is used to map the Data, means any Main Memory Block any cache memory line in a sequence.
- ❑ In Associative Mapping Physical Address is Interpreted as:

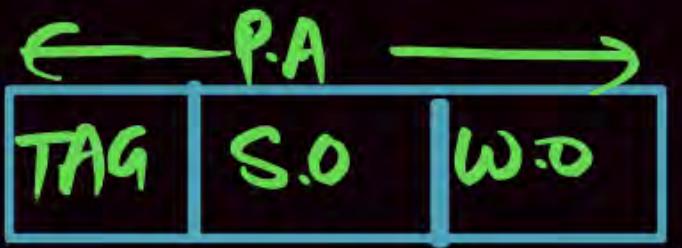


Direct Mapping



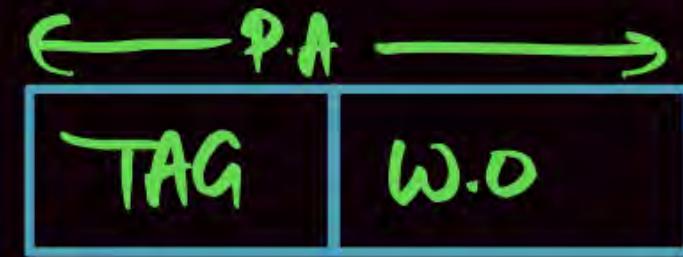
$$K \bmod N = i$$

Set Associative Mapping



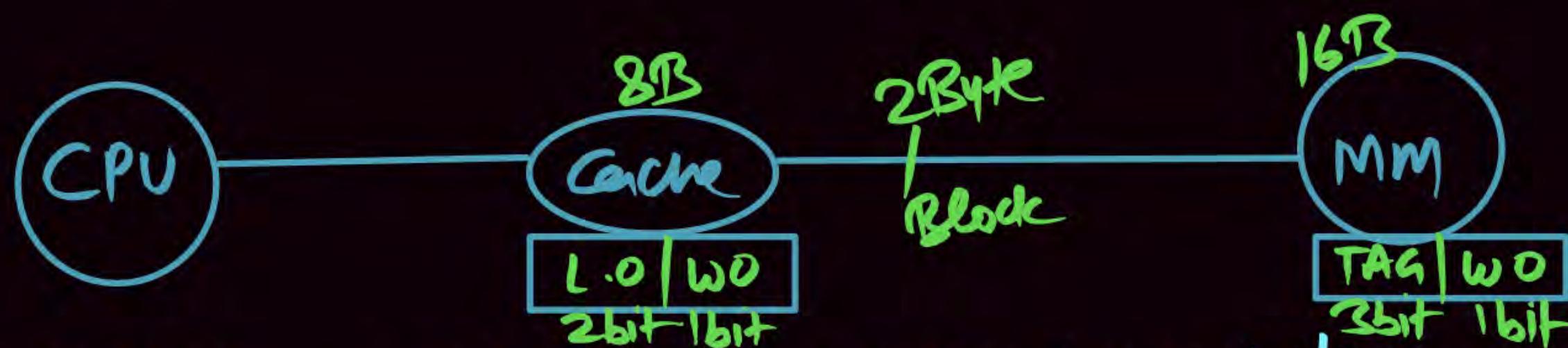
$$K \bmod S = i$$

Associative/Fully Associative mapping

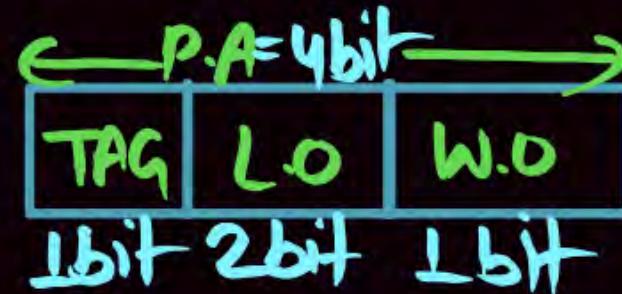


No Mapping function

(e)



Direct mapping

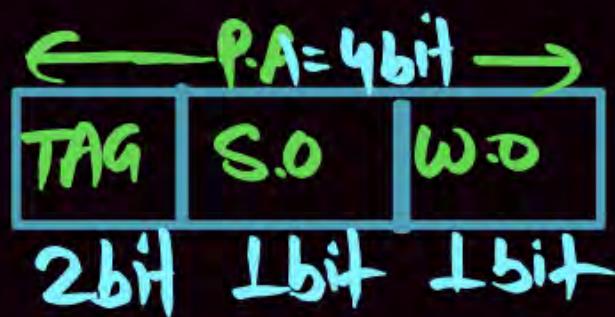


$$K \bmod N = i$$

$$K \bmod 4 = i$$

$$\text{Tag Memory} = \# \text{Lines} \times \text{Tag bit}$$

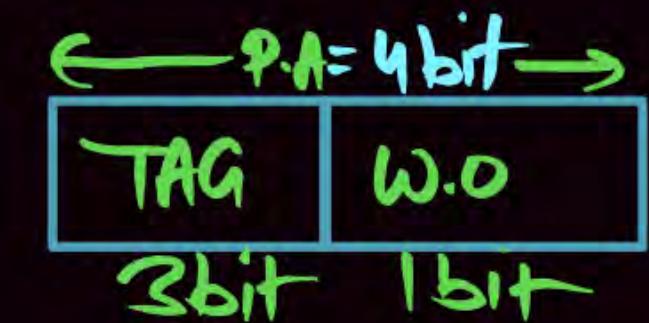
$$\Rightarrow 4 \times 1 \\ = 4 \text{ bits}$$



$$K \bmod S = i$$

$$K \bmod 2 = i$$

$$\text{Tag Memory} = 4 \times 2 \\ = 8 \text{ bit}$$



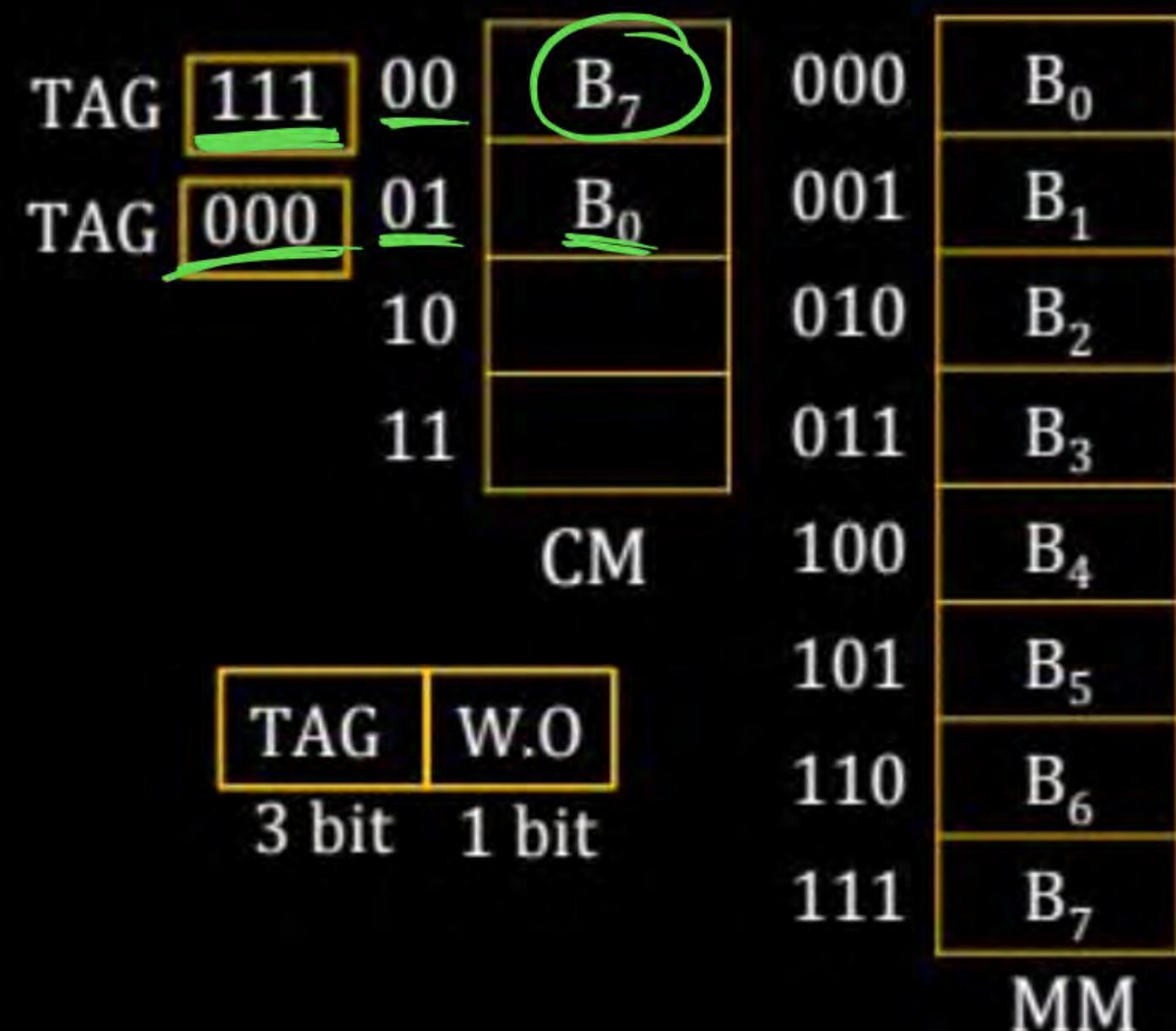
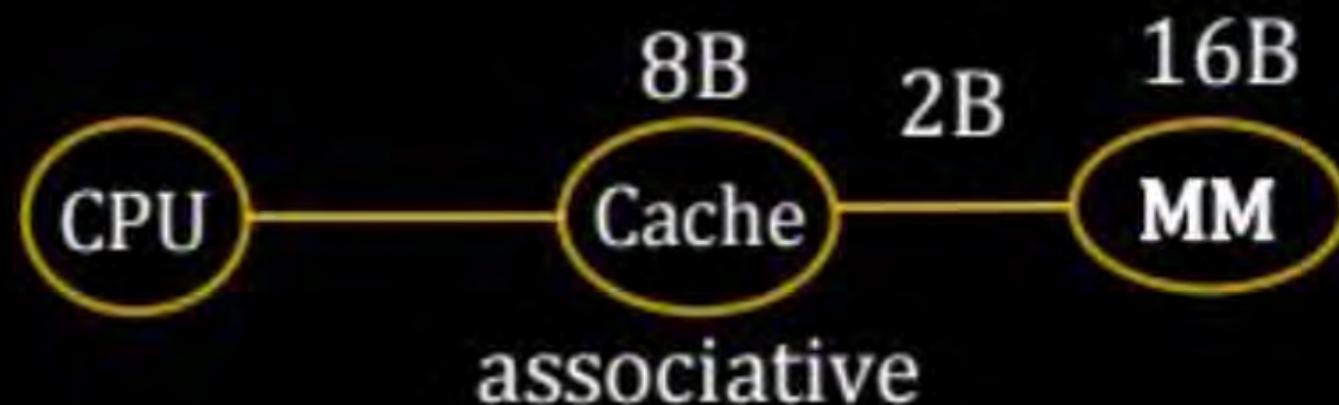
No Mapping function

$$\text{Tag Memory} = 4 \times 3$$

$$= 12 \text{ bits}$$

3) Associative Mapping

- In this no mapping function is used to transfer the data from MM to CM.
- No Conflict Miss;



MM Block Associative
 Mapping

B₇[111]

No mapping
Function

Any line

B₀[000]

No mapping
Function

Any line

$$\frac{\text{Tag Memory}}{\text{Size}} = \# \text{ LINES} \times \text{Tag bits}$$

Example: #LINE = 4 & Tag bits = 3

$$\frac{\text{Tag Memory}}{\text{Size}} = 4 \times 3 = 12 \text{ bits}$$

Q.

Consider a 64KB Direct Mapped Cache organized into a 64 word blocks. Word length of the CPU is 32bits. Main Memory 4GB. In the Cache Controller is comprising of 1 Valid bit & 1 Update bits. calculate the number of bit required for

- (i) P.A
- (ii) TAG
- (iii) L.O
- (iv) W.O
- (v) #LINES
- (vi) TAG Memory Size.**

Direct Mapped Cache
CacheSize = 64KB

Block Size = 64Word

Word Length = 32 bits \Rightarrow 4Byte

MM = 4GBYTE \Rightarrow 2^{32} Byte \Rightarrow P.A = 32bit

$$\#LINE = \frac{\text{CacheSize}}{\text{B.S}} \Rightarrow \frac{64\text{KB}}{256\text{B}} = \frac{2^{16}}{2^8} = 2^8 \text{ Lines}$$

Block Size = 64Word

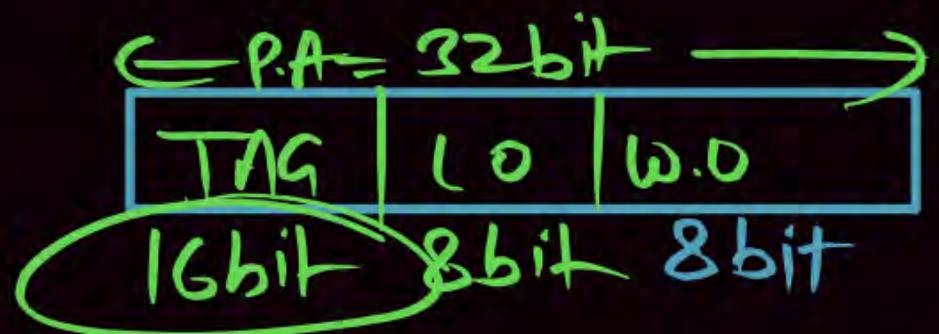
$\Rightarrow 64 \times 4\text{Byte}$

$\Rightarrow 256\text{Byte}$

2^8Byte

2^8Lines

L.O = 8bit



$$\text{Tag entry} = \text{Tag bit} + \text{extra bit} = 16 + 1 + 1 = 18 \text{ bits}$$

$$\text{Tag Memory Size} = \#LINE \times \text{Tag bit}$$

$$\Rightarrow 2^8 \times 18$$

$$= 256 \times 18$$

$$= 4608 \text{ bits} \quad \underline{\text{Ans}}$$

Q.

Consider fully associative cache consists 8 Block & MM contain 128 Block & Request made by the CPU:

119, 84, 37, 0, 16, 0, 84, 120, 121, 93, 37, 0, 43, 39, 47, 48.

Calculate # of compulsory & Capacity miss?



0	119 43
1	84 39
2	37 47
3	0 48
4	16
5	120
6	121
7	93

Hit: 0, 84, 37, 0

#Hits: 4

Total Miss = 12

③ Replacement Algorithm

When Cache is Full, then replacement algorithm are required to replace the exist cache block with new block.

In the CM design 3 type of replacement algorithm is used.

- 1) Random Algorithm
- 2) FIFO Replacement
- 3) LRU Replacement

In the random algorithm, any cache block can be replaced based on the random selection.

3 Type of Miss.

Types of Misses

In the CM design 3 types of misses are present.

- 1) Compulsory miss - (Cold start miss / first reference miss)

This miss will occur when the very first reference to the cache itself
a miss.

- 2) Capacity Miss - This miss will occur when cache is full.

- 3) Conflict Miss (Collision miss / reference miss)

This miss will occur when the too many blocks are placed into same
cache line or same cache SET.

Q.1

Consider 4 block cache memory (initially empty) with the following MM block references.

7, 8, 10, 15, 7, 8, 16, 7, 8, 10

Identify the Hit Ratio using

(i) **FIFO**

(iii) **Direct Mapped cache**

(ii) **LRU**

(iv) **2 - way Set Associative with LRU**

Q.1

Consider 4 block cache memory (initially empty) with the following MM block references.

7, 8, 10, 15, 7, 8, 16, 7, 8, 10

Identify the Hit Ratio using

(i) FIFO

Q.1

Consider 4 block cache memory (initially empty) with the following MM block references.

7, 8, 10, 15, 7, 8, 16, 7, 8, 10

Identify the Hit Ratio using

(ii) LRU

Q.1 Consider 4 block cache memory (initially empty) with the following MM block references.

7, 8, 10, 15, 7, 8, 16, 7, 8, 10

Identify the Hit Ratio using $k \bmod N = i$

(iii) Direct Mapped cache

CM Line No.

$$k \bmod 4 = i$$

$$7 \bmod 4 = 3$$

$$8 \bmod 4 = 0$$

$$10 \bmod 4 = 2$$

$$15 \bmod 4 = 3$$

$$7 \bmod 4 = 3$$

$$8 \bmod 4 = 0 \Rightarrow \text{HIT}$$

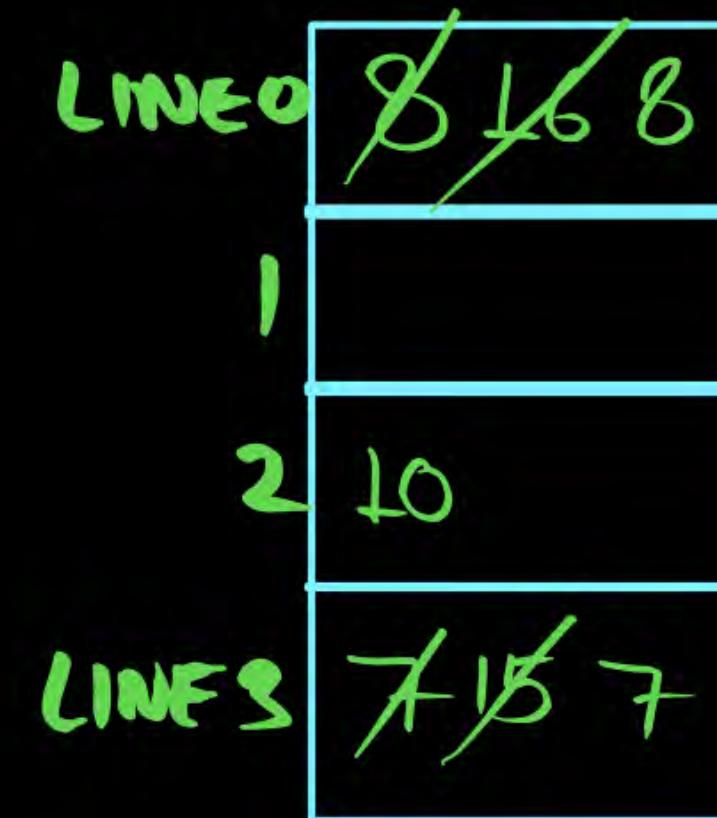
$$16 \bmod 4 = 0$$

$$7 \bmod 4 = 3 \Rightarrow \text{HIT}$$

$$8 \bmod 4 = 0$$

$$10 \bmod 4 = 2 \Rightarrow \text{HIT}$$

$$\begin{aligned} \text{Hit Ratio} &= \frac{3}{10} \\ &= 0.3 \end{aligned}$$



Q.1 Consider 4 block cache memory (initially empty) with the following MM block references.

7, 8, 10, 15, 7, 8, 16, 7, 8, 10

Identify the Hit Ratio using

(iv) 2 - way Set Associative with LRU

$$k \bmod S \Rightarrow k \bmod 2 = i$$

$$7 \bmod 2 = 1$$

$$8 \bmod 2 = 0$$

$$\cancel{10} \bmod 2 = 0$$

$$15 \bmod 2 = 1$$

$$7 \bmod 2 = 1 \Rightarrow \text{HIT}$$

$$\uparrow 8 \bmod 2 = 0 \Rightarrow \text{HIT}$$

$$\uparrow \cancel{16} \bmod 2 = 0 \Rightarrow \text{MISS}$$

$$\uparrow 7 \bmod 2 = 1 \Rightarrow \text{HIT}$$

$$\uparrow 8 \bmod 2 = 0 \Rightarrow \text{HIT}$$

$$\uparrow 10 \bmod 2 = 0 \Rightarrow \text{MISS}$$

$$\text{Hit Ratio} = \frac{4}{10} = 0.4$$

$$\begin{aligned} \# \text{LINE} &= 4 \\ \text{2 Way Set} \\ \# \text{SET} &= \frac{4}{2} = 2 \end{aligned}$$

SET 0	8 10 16 10
SET 1	7 15

Q.

Consider a small two-way set-associative cache memory, consisting of 4 blocks. For choosing the block to be replaced, use the least recently used (LRU) scheme. The number of cache misses for the following sequence of block addresses is 8, 12, 0, 12, 8

- (a) 2
- (b) 3
- (c) 4
- (d) 5

[GATE - 2004]

P
W

Types of Misses

In the CM design 3 types of misses are present.

- 1) Compulsory miss - (Cold start miss / first reference miss)

This miss will occur when the very first reference to the cache itself a miss.

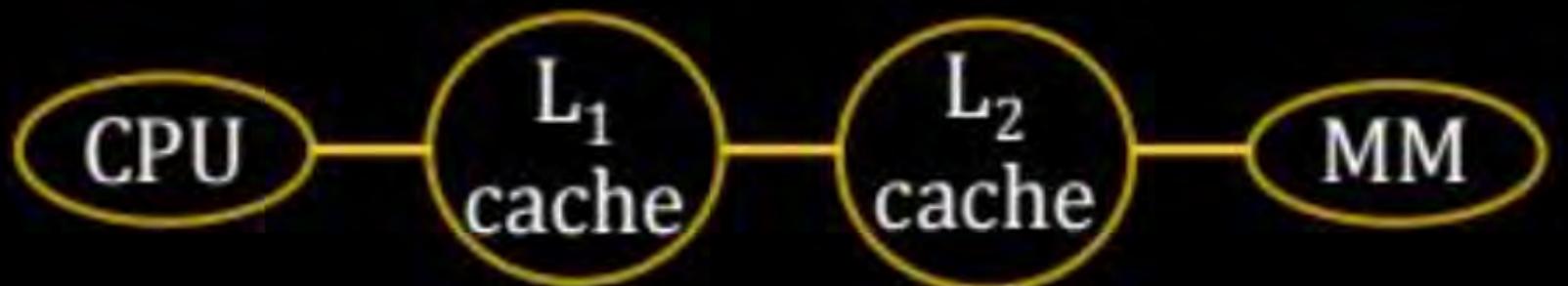
- 2) Capacity Miss - This miss will occur when cache is full.

- 3) Conflict Miss (Collision miss / reference miss)

This miss will occur when the too many blocks are placed into same cache line or same cache SET.

Multi level cache

- ❑ To reduce the miss penalty multi-level caches are used in the system design.
- ❑ The number of cycles required to transfer the data from higher levels to L_1 due to miss operation is called as miss penalty



Local Miss Rate [LMR]

Global Miss Rate [GMR]

$$\text{Local Miss Rate} = \frac{\text{\#misses in the cache}}{\text{\# accesses to that cache}}$$

$$\text{Global Miss Rate} = \frac{\text{\#misses in the cache}}{\text{Total \#CPU generated reference}}$$

Average access time of the memory is always calculated in terms of Hit time, miss rate and miss penalty is as follows:

$$T_{avg} = \text{Hit time } L_1 + (\text{Miss Rate } L_1 * \text{Miss penalty } L_1)$$

$$\text{Miss penalty } L_1 = \text{Hit time } L_2 + (\text{Miss rate } L_2 * \text{Miss penalty } L_2)$$

$$\text{Miss penalty } L_2 = \text{MM Access Time}$$

Types of Misses

In the CM design 3 types of misses are present.

- 1) Compulsory miss - (Cold start miss / first reference miss)

This miss will occur when the very first reference to the cache itself a miss.

- 2) Capacity Miss - This miss will occur when cache is full.

- 3) Conflict Miss (Collision miss / reference miss)

This miss will occur when the too many blocks are placed into same cache line or same cache SET.



PYQ's & Home Work Questions

Home Work Questions

- Q.1** Consider a Direct Mapping if the size of Cache memory is 512KB & Main Memory 512 KB & Cache line size (Block) is 64KB the calculate the number of bit required for
- (i) P.A
 - (ii) TAG
 - (iii) B.O
 - (iv) W.O
 - (v) #LINES
 - (vi) TAG Memory Size.

Q.2

Consider a Direct Mapping, Cache size = 64 byte, Line Size = 8

Byte. MM = 256 Byte then #bits for P.A, TAG, L.O, W.O Tag
memory size?

Q.3 Consider a Direct Mapping, Cache size = 128 KB, Line Size = 64

Byte. Main Memory is 1MB then what is the line number of physical address $(ABCDE)_{16}$?

Q.4

Consider a 2-way set associative if the size of cache memory is 512KB & Main Memory 512MB & Cache line size is 64KB then calculate the Number of bit Required for

Q.5

Consider a 2-way set associative Cache Size = 256 KB, Line size = $\frac{P}{W}$,
32 Byte, MM = 1MB, then what is the set number of Physical
address $(ABCDE)_{16}$?



GATE PYQ's

An 8-way set associative cache of size 64 KB (1 KB = 1024 bytes) is used in a system with 32-bit address. The address is sub-divided into TAG, INDEX, and BLOCK OFFSET.

The number of bits in the TAG is _____.

[GATE-2023-CS: 2M]

Consider a computer system with a byte-addressable primary memory of size 2^{32} bytes. Assume the computer system has a direct-mapped cache of size 32 KB ($1 \text{ KB} = 2^{10}$ bytes), and each cache block is of size 64 bytes.

The size of the tag field is _____ bits.

[GATE-2021(Set-1)-CS: 1M]

Consider a machine with a byte addressable main memory of 2^{32} bytes divided into blocks of size 32 bytes. Assume that a direct mapped cache having 512 cache lines is used with this machine. The size of the tag field in bits is ____.

[GATE-2017(Set-1)-CS: 2M]

Consider a machine with a byte addressable main memory of 2^{20} bytes, block size of 16 bytes and a direct mapped cache having 2^{12} cache lines. Let the addresses of two consecutive bytes in main memory be $(E201F)_{16}$ and $(E2020)_{16}$. What are the tag and cache line address (in hex) for main memory address $(E201F)_{16}$?

[GATE-2015(Set-3)-CS: 1M]

A E, 201

B F, 201

C E, E20

D 2, 01F

Q.5

[Common Data for this and next question]

A computer has a 256 Kbyte, 4-way set associative. Write back data cache with block size of 32 Bytes. The processor sends 32 bit address to the cache controller. Each cache tag directory entry contains, in addition to address tag, 2 valid bits, 1 modified bit and 2 replacement bit.

The number of bits in the tag field of an address is

- (a) 11
- (b) 14
- (c) 16
- (d) 27

[GATE - 2012: 2 Marks]

Q.6

[Common Data from previous question]

A computer has a 256 Kbyte, 4-way set associative. Write back data cache with block size of 32 Bytes. The processor sends 32 bit address to the cache controller. Each cache tag directory entry contains, in addition to address tag, 2 valid bits, 1 modified bit and 2 replacement bit.

[GATE - 2012: 2 Marks]

The size of the cache tag directory is

- (a) 160 Kbits
- (b) 136 Kbits
- (c) 40 Kbits
- (d) 32 Kbits

MCQ**Q.7**

An 8KB direct-mapped write back cache is organized as multiple blocks, each of size 32 bytes. The processor generates 32-bit addresses. The cache controller maintains the tag information for each cache block comprising of the following.

1Valid bit

1Modified bit

As many bits as the minimum needed to identify the memory block mapped in the cache.

What is the total size of memory needed at the cache controller to store meta-data (tags) for the cache? [GATE-2011-CS: 2M]

A 4864 bits

B 6144 bits

C 6656 bits

D 5376 bits

Q.8

Q.9



Common Data for next two questions:

Consider a computer with a 4-ways set-associative mapped cache of the following characteristics: a total of 1 MB of main memory, a word size of 1 byte, a block size of 128 words and a cache size of 8 KB.

While accessing the memory location 0C795H by the CPU, the contents of the TAG field of the corresponding cache line is

[GATE-2008-CS: 2M]

- A 000011000
- C 00011000

- B 110001111
- D 110010101

The number of bits in the TAG, SET and WORD fields, respectively are:

[GATE-2008-CS: 2M]

A 7, 6, 7

B 8, 5, 7

C 8, 6, 6

D 9, 4, 7

Q.10

Consider a 4-way set associative cache consisting of 128 lines with a line size of 64 words. The CPU generates a 20-bit address of a word in main memory. The number of bits in the TAG, SET and WORD fields are respectively.

- (a) 9, 6, 5
- (b) 7, 7, 6
- (c) 7, 5, 8
- (d) 9, 5, 6

[GATE - 2007]

P
W

A certain processor uses a fully associative cache of size 16 kB. The cache block size is 16 bytes. Assume that the main memory is byte addressable and uses a 32-bit address. How many bits are required for the Tag and the Index fields respectively in the addresses generated by the processor?

[GATE-2019-CS: 1M]

A 24-bits and 0-bits

B 28-bits and 4-bits

C 24-bits and 4-bits

D 28-bits and 0-bits

The width of the physical address on a machine is 40 bits. The width of the tag field In a 512 KB 8-way set associative cache is _____ bits.

[GATE-2016(Set-2)-CS: 2M]

A 4-way set-associative cache memory unit with a capacity of 16 KB is built using a block size of 8 words. The word length is 32 bits. The size of the physical address space is 4 GB. The number of bits for the TAG field is ____.

[GATE-2014(Set-2)-CS: 1M]

A cache memory unit with capacity of N words and block size of B Words is to be designed. If it is designed as a direct mapped cache, the length of the TAG field is 10 bits. If the cache unit is now designed as a 16-way set-associative cache, the length of the TAG field is _____ bits.

[GATE-2017(Set-1)-CS: 2M]

Q.15

The main memory of a computer has 2^m blocks while the cache has 2^c blocks. If the cache uses the set associative mapping scheme with 2 blocks per set, then block k of the main memory maps to the set.

[GATE - 1999]

- (a) $(k \bmod m)$ of the cache
- (b) $(k \bmod c)$ of the cache
- (c) $(k \bmod 2^c)$ of the cache
- (d) $(k \bmod 2^m)$ of the cache

The size of the physical address space of a processor is 2^P bytes. The word length is 2^W bytes. The capacity of cache memory is 2^N bytes. The size of each cache block is 2^M words. For a K-way set-associative cache memory, the length (in number of bits) of the tag field is

[GATE-2018-CS: 2M]

A $P - N - \log_2 K$

B $P - N + \log_2 K$

C $P - N - M - W - \log_2 K$

D $P - N - M - W + \log_2 K$

A computer system with a word length of 32 bits has a 16 MB byte-addressable main memory and a 64 KB, 4-way set associative cache memory with a block size of 256 bytes. Consider the following four physical addresses represented in hexadecimal notation.

$A_1 = 0 \times 42C8A4, A_2 = 0 \times 546888, A_3 = 0 \times 6A289C, A_4 = 0 \times 5E4880$

Which one of the following is TRUE?

[GATE-2020-CS: 2M]

- A A₁ and A₃ are mapped to the same cache set.
- B A₂ and A₃ are mapped to the same cache set.
- C A₃ and A₄ are mapped to the same cache set.
- D A₁ and A₄ are mapped to different cache sets.

Consider a set-associative cache of size 2 kb ($1\text{ KB} = 2^{10}$ bytes) with cache block size of 64 bytes. Assume that the cache is byte - addressable and a 32-bit address is used for accessing the cache. If the width of the tag field is 22 bits, the associativity of the cache is ____.

[GATE-2021(set-2)-CS: 1M]

Consider a 4-way set associative cache (initially empty) with total 16 cache blocks. The main memory consists of 256 blocks and the request for memory blocks is in the following order:

0, 255, 1, 4, 3, 8, 133, 159, 216, 129, 63, 8, 48, 32, 73, 92, 155

Which one of the following memory block will NOT be in cache if LRU replacement policy is used?

[GATE-2009-CS: 2M]

A 3

B 8

C 129

D 216

Consider a 2-way set associative cache with 256 blocks and uses LRU replacement. Initially the cache is empty. Conflict misses are those misses which occur due to contention of multiple blocks for the same cache set. Compulsory misses occur due to first time access to the block. The following sequence of accesses to memory blocks (0, 128, 256, 128, 0, 128, 256, 128, 1, 129, 257, 129, 1, 129, 257, 129) is repeated 10 times. The number of conflict misses experienced by the cache is _____.

[GATE-2017(Set-1)-CS: 2M]

If the associativity of a processor cache is doubled while keeping the capacity and block size unchanged, which one of the following is guaranteed to be NOT affected?

[GATE-2014(Set-2)-CS: 2M]

- A Width of tag comparator
- B Width of set index decoder
- C Width of way selection multiplexer
- D Width of processor to main memory data bus

Consider a two-level cache hierarchy with L_1 and L_2 caches. An application incurs 1.4 memory accesses per instruction on average. For this application, the miss rate of L_1 cache is 0.1; the L_2 cache experiences on average, 7 misses per 1000 instructions. The miss rate of L_2 expressed correct to two decimal places is _____. [GATE-2017(Set-1)-CS: 1M]

In a two-level cache system, the access times of L_1 and L_2 caches are 1 and 8 clock cycles, respectively. The miss penalty from the L_2 cache to main memory is 18 clock cycles. The miss rate of L_1 cache is twice that of L_2 . The average memory access time (AMAT) of this cache system is 2 cycles. The miss rates of L_1 and L_2 respectively are:

[GATE-2017(Set-2)-CS: 2M]

- A 0.111 and 0.056
- B 0.056 and 0.111
- C 0.0892 and 0.1784
- D 0.1784 and 0.0892

Q.24



Common Data for next two questions:

Consider a machine a 2-way set associative data cache of size 64Kbytes and block size 16 bytes. The cache is managed using 32 bit virtual addresses and the page size is 4 Kbytes. A program to be run on this machine begins as follows:

Double ARR [1024] [1024]

Int i, j;

```
/* Initialize array ARR to 0.0 */  
for (i = 0; i < 1024; i++)  
    for (j = 0; j < 1024; j++)  
        ARR [i] [j] = 0.0;
```

The size of double 8 bytes. Array ARR is in memory starting at the beginning of virtual page 0xFF000 and stored in row major order. The cache is initially empty and no pre-fetching is done. The only data memory references made by the program are those to array ARR.

MCQ

The total size of the tags in the cache directory is

[GATE-2008-CS: 2M]

- A 32 kbits
- B 34 kbits
- C 64 kbits
- D 68 kbits

Q.25

[Common Data for this and next question]

Consider two cache organization. The first one is 32 KB 2-way set associative with 32-byte block size. The second one is of the same size but direct mapped. The size of an address is 32 bits in both cases. A 2-to-1 multiplexer has latency of 0.6 ns while a k-bit comparator has a latency of $k/10$ ns. The hit latency of the set associative organization is h_1 while that of the direct mapped one is h_2 . The value of h_1 is

- (a) 2.4 ns
- (b) 2.3 ns
- (c) 1.8 ns
- (d) 1.7 ns

[GATE - 2006: 2 Marks]

Q.26

[Common Data from previous question]

Consider two cache organization. The first one is 32 KB 2-way set associative with 32-byte block size. The second one is of the same size but direct mapped. The size of an address is 32 bits in both cases. A 2-to-1 multiplexer has latency of 0.6 ns while a k-bit comparator has a latency of $k/10$ ns. The hit latency of the set associative organization is h_1 while that of the direct mapped one is h_2 . The value of h_2 is

- (a) 2.4 ns
- (b) 2.3 ns
- (c) 1.8 ns
- (d) 1.7 ns

[GATE - 2006: 2 Marks]

Q.27

A computer system has a level - 1 instruction cache (1-cache), a level-1 data cache(D-cache) and a level-2 cache(L2-cache) with the following specifications.

P
W

	Capacity	Mapping method	Block size
1-cache	4K words	Direct mapping	4 words
D-cache	4K words	2-way set associative mapping	4 words
L2-cache	64K words	4-way set associative mapping	16 words

Capacity mapping method block size 1-cache 4K words direct mapping 4 words D-cache 4 k words 2 way set associative mapping 4 words L2-cache 64K words 4-way set associative mapping 16 words. The length of the physical address of a word in the main memory is 30 bits. The capacity of the tag memory in the 1-cache, D-cache & L2-cache is. Respectively,

(a) $1K \times 18\text{-bit}$, $1K \times 19\text{-bit}$, $4K \times 16\text{-bit}$ (b) $1K \times 16\text{-bit}$, $1K \times 19\text{-bit}$, $4K \times 18\text{-bit}$
(c) $1K \times 16\text{-bit}$, $512 \times 18\text{-bit}$, $1K \times 16\text{-bit}$ (d) $1K \times 16\text{-bit}$, $512 \times 18\text{-bit}$, $1K \times 18\text{-bit}$

[GATE - 2006: 2 Marks]

Q.28 Consider a small two-way set-associative cache memory, consisting of 4 blocks. For choosing the block to be replaced, use the least recently used (LRU) scheme. The number of cache misses for the following sequence of block addresses is 8, 12, 0, 12, 8

- (a) 2
- (b) 3
- (c) 4
- (d) 5

[GATE - 2004]

Q.29

Consider a system with 2 KB direct mapped data cache with a block size of 64 bytes. The system has a physical address space of 64 KB and a word length of 16 bits. During the execution of a program, four data words P, Q, R and S are accessed in that order 10 times (i.e., PQRSPQRS....) Hence, there are 40 accesses to data cache altogether. Assume that the data cache is initially empty and no other data words are accessed by the program. The addresses of the first bytes of P, Q, R and S are 0xA248, 0xC28A, 0xCA8A and 0xA262, respectively. For the execution of the above program, which of the following statements is/are TRUE with respect to the data cache? [2022: MSQ 2M]

- A Every access to S is a hit.
- B Once P is brought to the cache it is never evicted.
- C At the end of the execution only R and S reside in the cache.
- D Every access to R evicts Q from the cache.

**THANK
YOU!**

