



# CS & IT ENGINEERING

COMPUTER ORGANIZATION  
AND ARCHITECTURE

## Cache Organization

Lecture No.- 4



By- Vishvadeep Gothi sir

# Recap of Previous Lecture



**Topic**

Cache Mapping

**Topic**

Direct Mapping

**Topic**

Tag

**Topic**

Index

# Topics to be Covered



**Topic**

Set Associative Mapping

**Topic**

Fully Associative Mapping



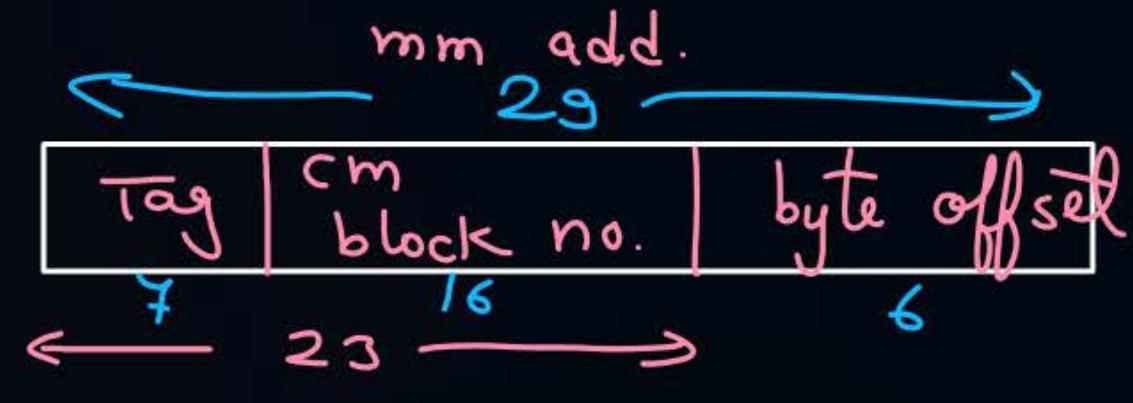
## [NAT]



- #Q. Blocks in Main memory =  $2^{23}$   
Blocks in Cache memory =  $2^{16}$

Block Size: 64 Bytes

Direct Mapping



$$\text{mm size} = 2^{23} * 64B = 2^{29} B \Rightarrow \text{mm add} = 29 \text{ bits}$$

No. of bits required for Byte Offset = ? 6

No of bits required for main memory address = ? 29

Index-bits = ? 16

Tag-bits = ? 7

Size of Tag Directory = ?  $2^{16} * 7 \text{ bits}$

[NAT]

P  
W

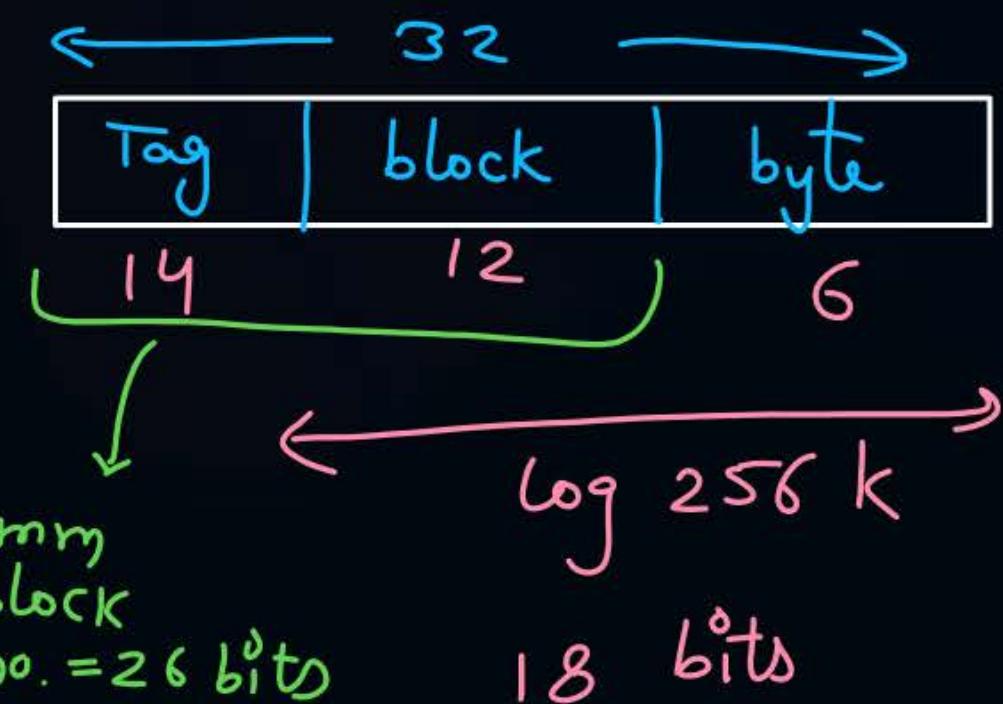
#Q. 32-bit architecture CPU

$$\text{Main Memory Size} = \underline{4\text{GB}} = 2^{32} B$$

$$\text{Cache Size} = 256\text{KB}$$

$$\text{Block Size} = 16 \text{ Words} = 16 * 4B \\ = 64B$$

Direct Mapping



No. of bits required for Byte Offset = ? 6

No of bits required for main memory address = ? 32

No of bits required for main memory block no. = ? 26

Index-bits = ? 12

Tag-bits = ? 14

Size of Tag Directory = ?  $2^{12} * 14 \text{ bits}$



## Topic : Calculating CM Block Number from MM Address



Main Memory Address Given in:

1. Binary
2. Hexadecimal
3. Decimal



# [NAT]

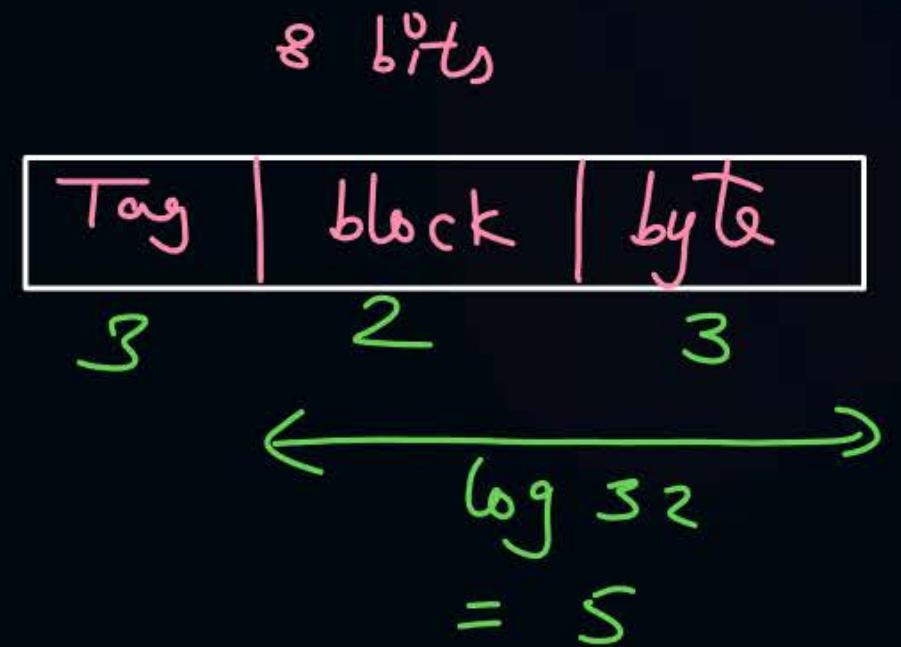


MM address = 8bits

Cache size = 32 bytes

Block size = 8 bytes

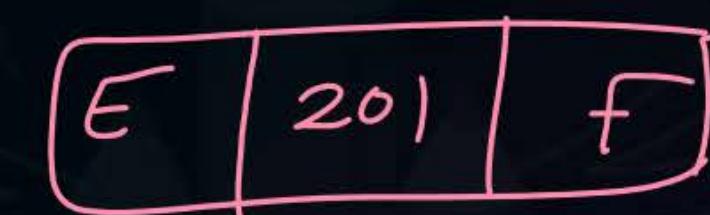
Direct Mapping



MM Address	Cache Block Number where it maps
<u>10100101</u>	$101   00   101 \Rightarrow (00)_2 = (0)_{10}$
<u>11010101</u>	$(10)_2 \Rightarrow (2)_{10}$
<u>11111101</u>	$(11)_2 \Rightarrow (3)_{10}$
<u>01001111</u>	$(01)_2 \Rightarrow (1)_{10}$
$(E2)_{16}$	<u>11100010</u> $(00)_2 = (0)_{10}$
$(B6)_{16}$	<u>10110110</u> $(10)_2 = (2)_{10}$
$(2C)_{16}$	<u>00101100</u> $(01)_2 = (1)_{10}$

#Q. 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  $(E201F)_{16}$ ?

Tag	block	byte
4	12	4



A ✓ E, 201

C E, E20

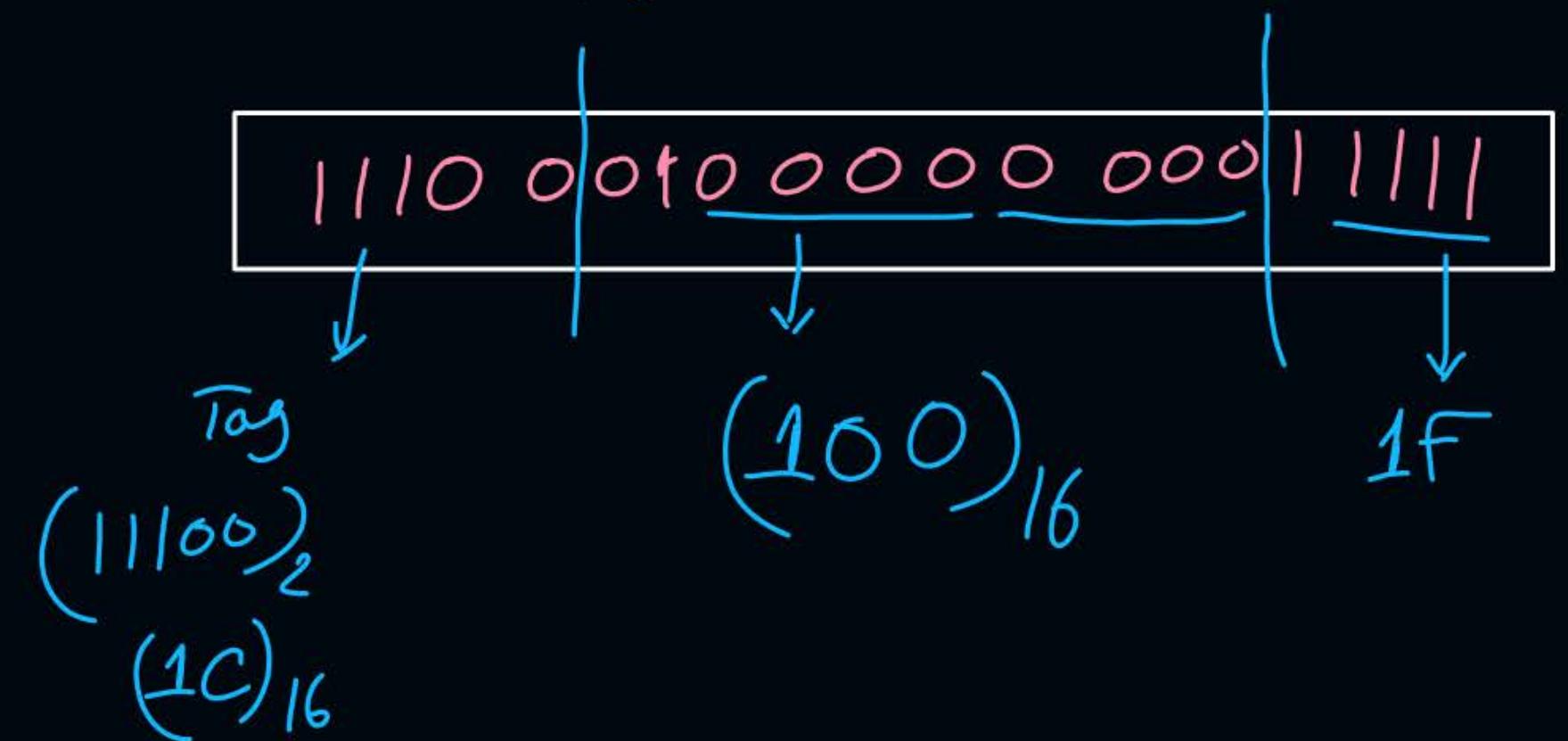
B F, 201

D 2, 01F

Ques) In prev. questi if 20 bits mm add. as :-

Tag	block	byte
5	10	5

for add.  $(E201F)_{16}$





## Topic : Calculating CM Block Number from MM Address



Main Memory Address Given in: Decimal

MM size = 128 bytes

CM size = 32 bytes

Block size = 4 bytes

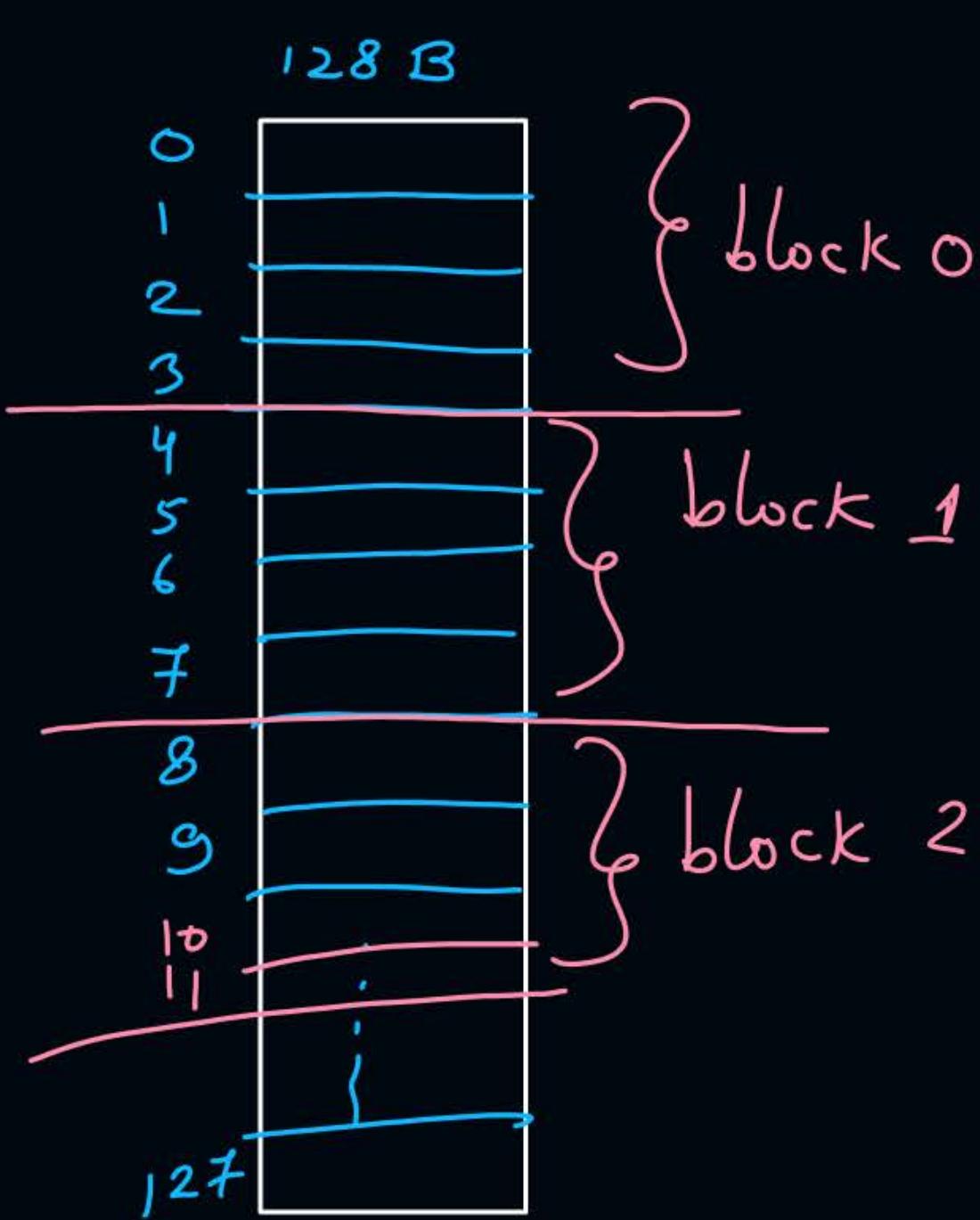
$$\text{no. of blocks in cache} = \frac{32B}{4B} = 8 \text{ blocks}$$

$$\text{mm block no.} = \left\lfloor \frac{\text{mm add.}}{\text{block size}} \right\rfloor$$

$$\text{byte offset} = (\text{mm add.}) \% \text{ block size}$$

$$\text{cm block no.} = \left( \text{mm block no.} \right) \% \text{ no. of blocks in cache}$$

$$\text{Tag} = \left\lfloor \frac{\text{mm block no.}}{\text{no. of blocks in cache}} \right\rfloor$$



$$\text{mm add.} = (99)_{10}$$

$$\text{mm block no.} = \left\lfloor \frac{99}{4} \right\rfloor = 24$$

$$\text{byte offset} = 99 \% 4 = 3$$

$$\text{cm block no.} = 24 \% 8 = 0$$

$$\text{Tag} = \left\lfloor \frac{24}{8} \right\rfloor = 3$$

3	0	3
---	---	---

mm address  $\Rightarrow (110)_{10}$

$$\text{mm block no.} = \left\lfloor \frac{110}{4} \right\rfloor = 27$$

$$\text{byte offset} = 110 \% 4 = 2$$

$$\text{cm block no.} = 27 \% 8 = 3$$

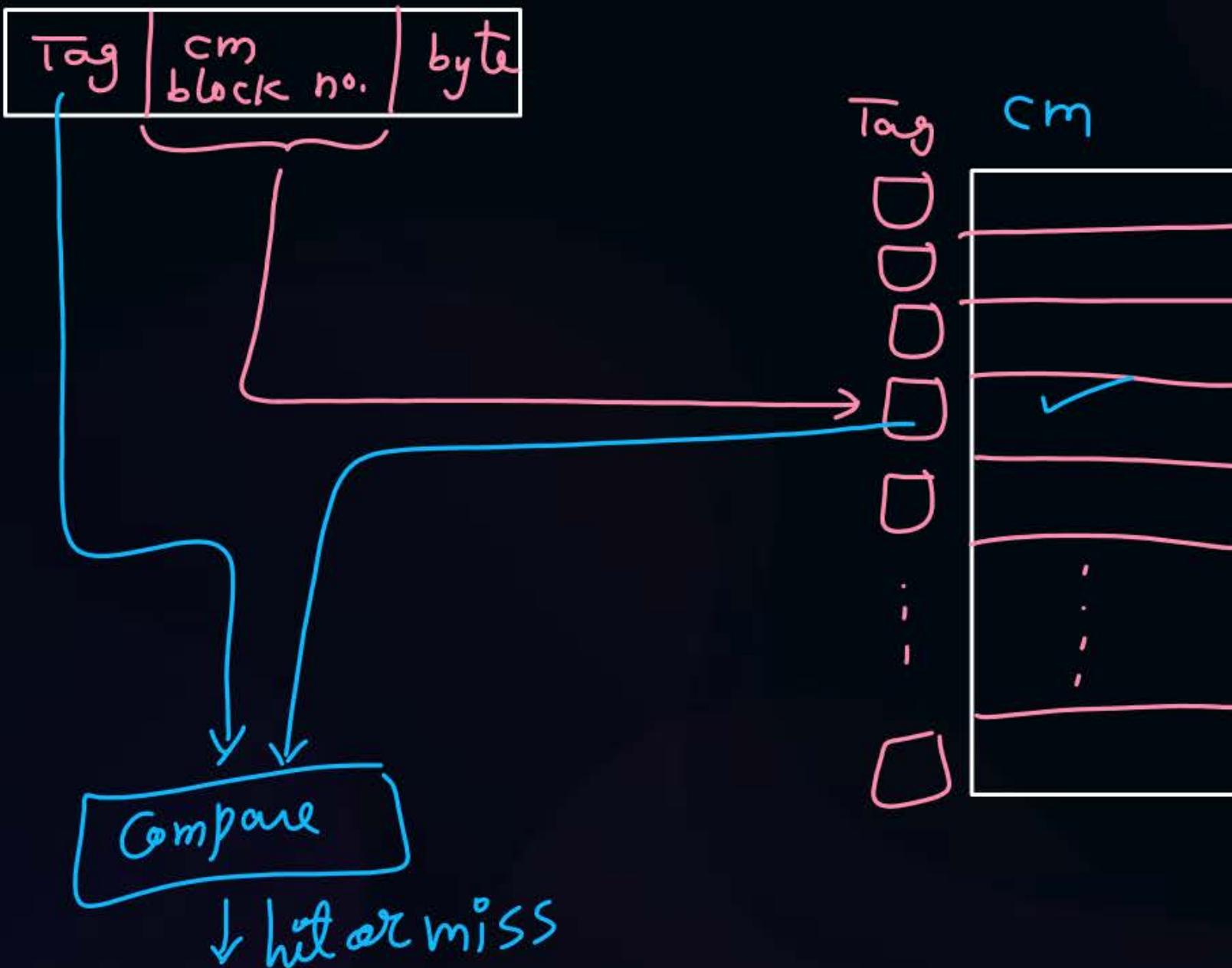
$$\text{Tag} = \left\lfloor \frac{27}{8} \right\rfloor = 3$$

3	3	2
---	---	---



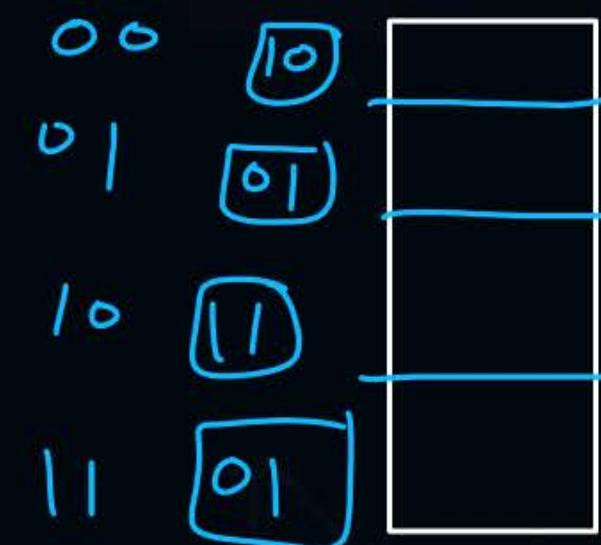
# Topic : Checking Hit/Miss in Direct Mapped Cache

mm add.



#Q. Consider a 64 bytes direct mapped cache with a block size of 16 bytes. Main memory size is 256bytes. Currently in the cache, the blocks are having tags as follows:

Block	Tag
00	10
01	01
10	11
11	01

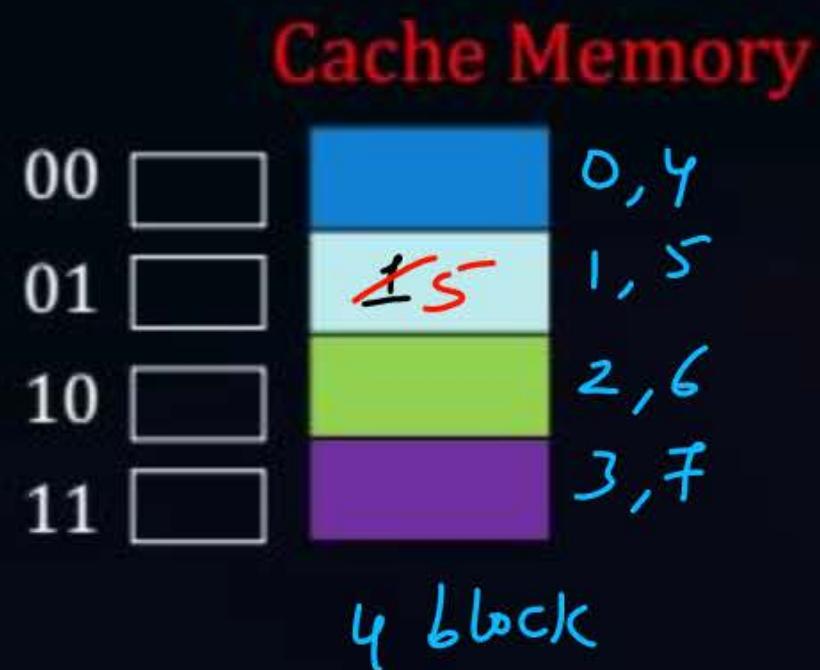


Identify the correct statement with respect to the availability of the main memory data into cache?

- a) Main memory byte number 243 present in cache
- b) Main memory byte number 143 present in cache
- c) Main memory byte number 43 present in cache
- d) Main memory byte number 119 present in cache



## Topic : Problem With Direct Mapping



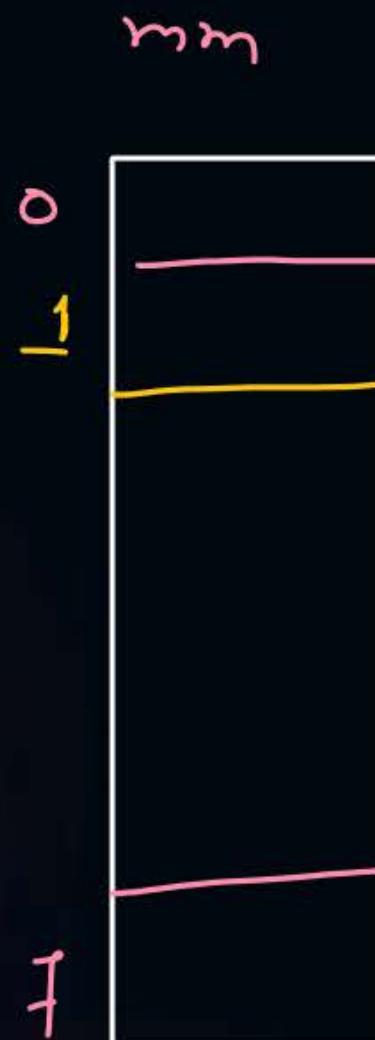
CPU requests mm block no. :-

1, 5, 1, 5, 1, 5, 1, 5  
miss  
miss  
miss



## Topic : Set Associative Mapping

2-way set associative



CPU Requests mm blocks :-





## Topic : Set Associative Mapping

P  
W

cm set no. = (mm block no.) % no. of sets in cache

no. of sets in cache =  $\frac{\text{no. of blocks in cache}}{\text{associativity}}$

mm add.



for  
set offset

no. of bits in set offset

$$= \log_2 (\text{no. of sets in cache})$$

Tag directory size = no. of blocks in cache \* ( Tag + extra bits )

Ques) mm add. = 28 bits

Cm size = 4KB

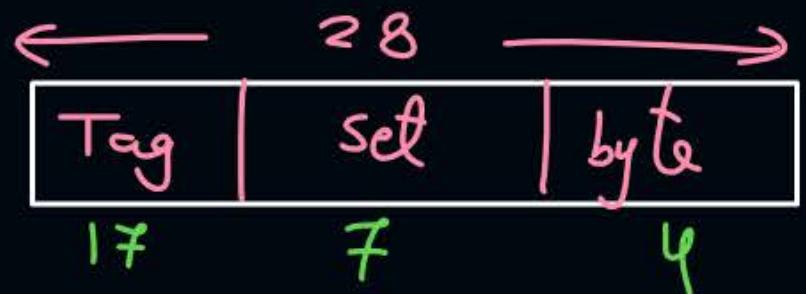
block = 16B

2-way set ass. Cache

$$\text{no. of blocks in cache} = \frac{4\text{KB}}{16\text{B}} = \frac{2^{12}}{2^4} = 2^8$$

$$\text{no. of sets in cache} = \frac{2^8}{2} = 2^7$$

set offset = 7 bits

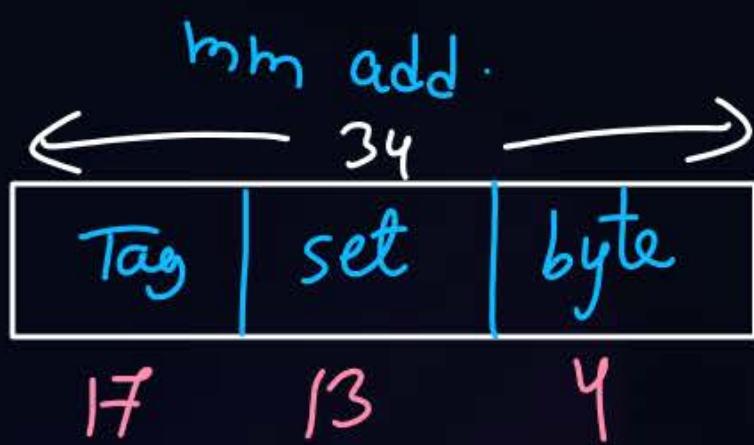


Tag directory size

$$= 2^8 * 17 \text{ bits}$$

#Q. A computer has a 512Kbyte, 4-way set associative, write back data cache with block size of 16 Bytes. The processor sends 34 bit addresses to the cache controller. Each cache tag directory entry contains, in addition to address tag, 2 valid bits, 1 modified bit and 1 replacement bit

1. The number of bits in the tag field of an address is 17 bits
2. The size of the cache tag directory is  $2^{15} * (17 + 2 + 1 + 1) \text{ bits} = 2^{15} * 21 \text{ bits}$



$$\text{no. of blocks in cache} = \frac{512 \text{ KB}}{16 \text{ B}} = \frac{2^{19}}{2^4} = 2^{15}$$

$$\text{no. of sets in cache} = \frac{2^{15}}{4} = \frac{2^{15}}{2^2} = 2^{13} \Rightarrow \text{set offset} = 13 \text{ bits}$$

Ques) mm add = 30 bits

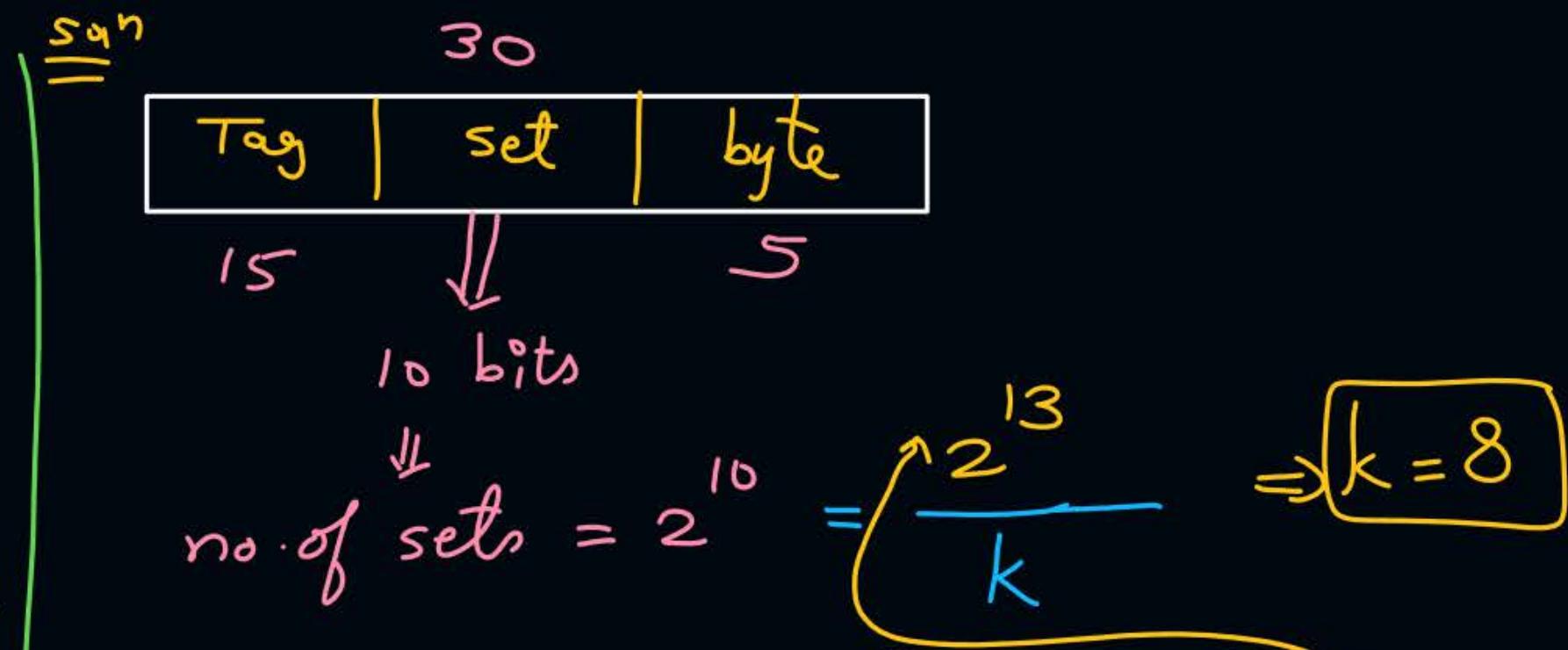
cm size = 256 KB

block size = 32 B

k-way set ass. Cache

Tag = 15 bits

$$k = \frac{8}{8}$$



$$\text{no. of blocks in cache} = \frac{256 \text{ KB}}{32 \text{ B}}$$

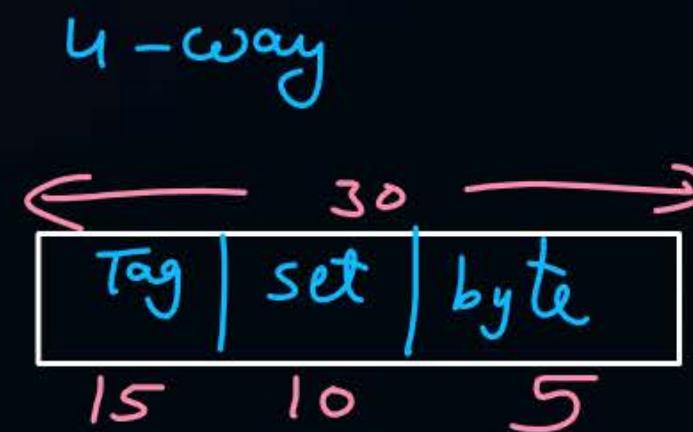
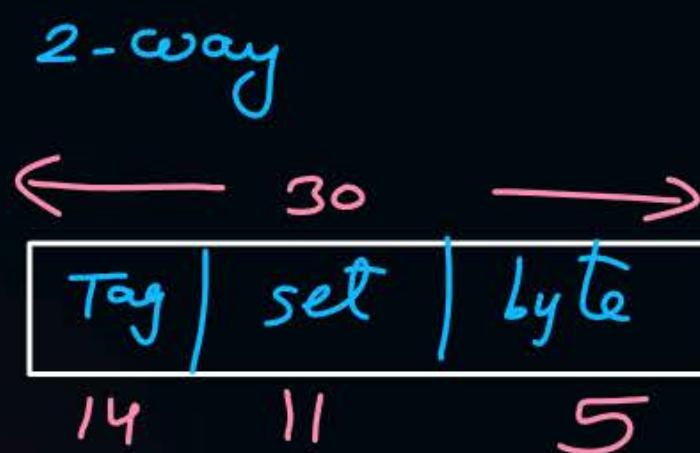
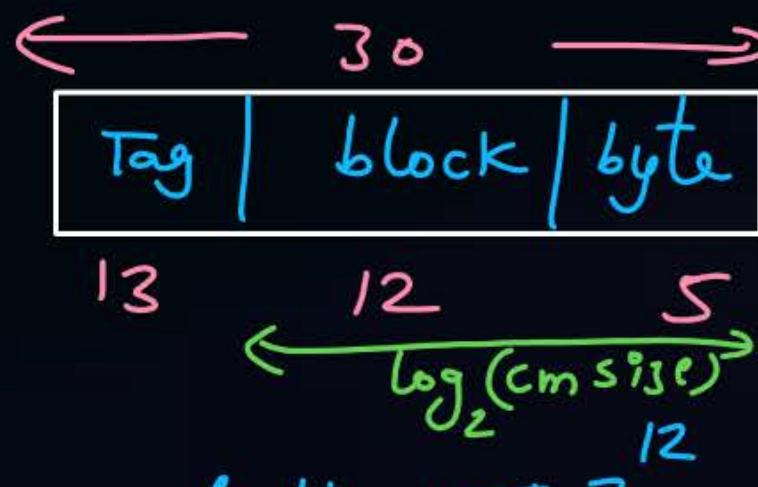
$$= \frac{2^{18}}{2^5} = 2^{13}$$



## Topic : Tag and Index in Associative Mapping

ex:-

Direct:-



$$\text{no. of sets} = \frac{2^{12}}{2} = 2^{11}$$

$$\text{no. of sets} = \frac{2^{12}}{4} = 2^{10}$$

Note:-

Everytime the associativity is doubled then Index is decreased by 1 and Tag is increased by 1.

for k-way set ass. cache

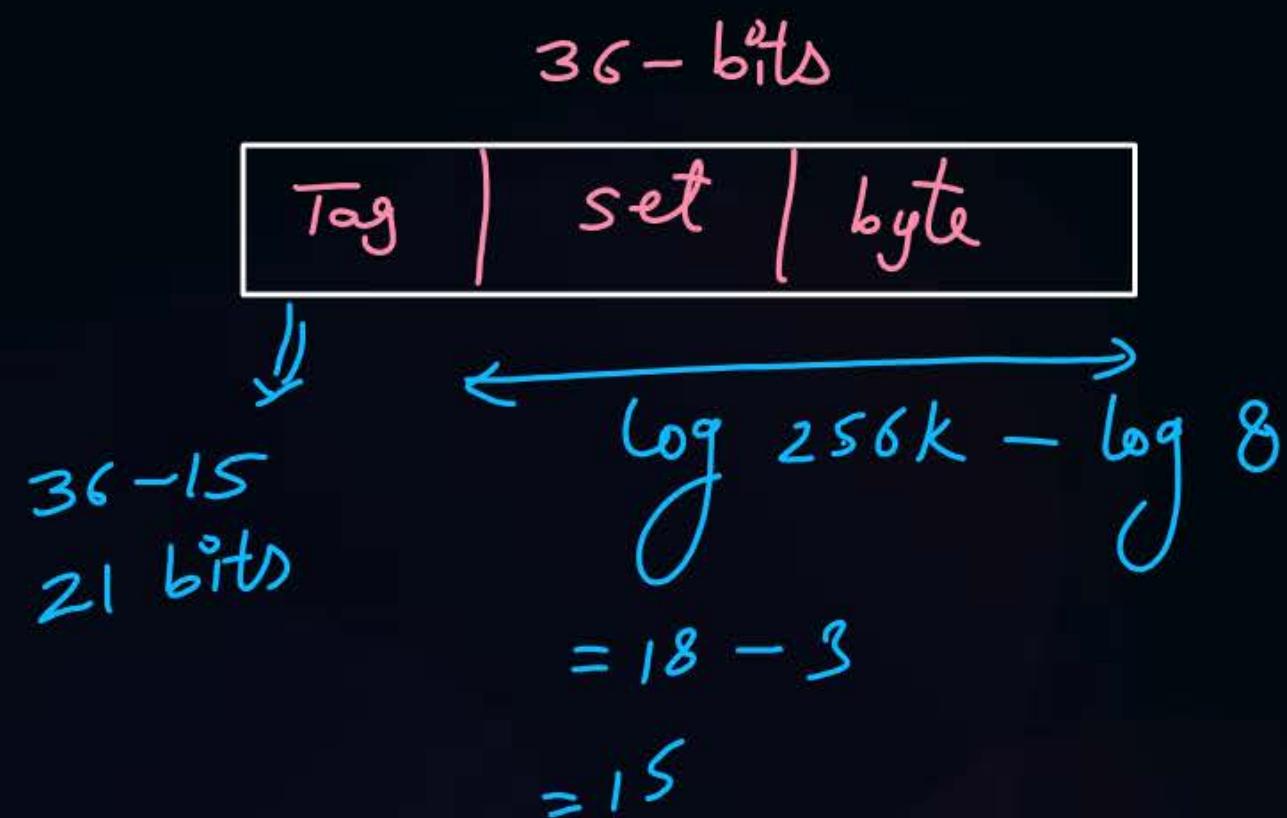
Tag	set	byte
-----	-----	------

$$\xleftarrow{\log_2(\text{cm size}) - \log_2 k}$$

$$\text{Tag bits} = \text{mm add.} - \left[ \log_2(\text{cm size}) - \log_2 k \right]$$

$$\text{Tag bits} = \text{mm add.} - \log_2(\text{cm size}) + \log_2 k$$

#Q. The width of the physical address on a machine is 36 bits. The width of the tag field in a 256 KB 8-way set associative cache is 21 bits ?



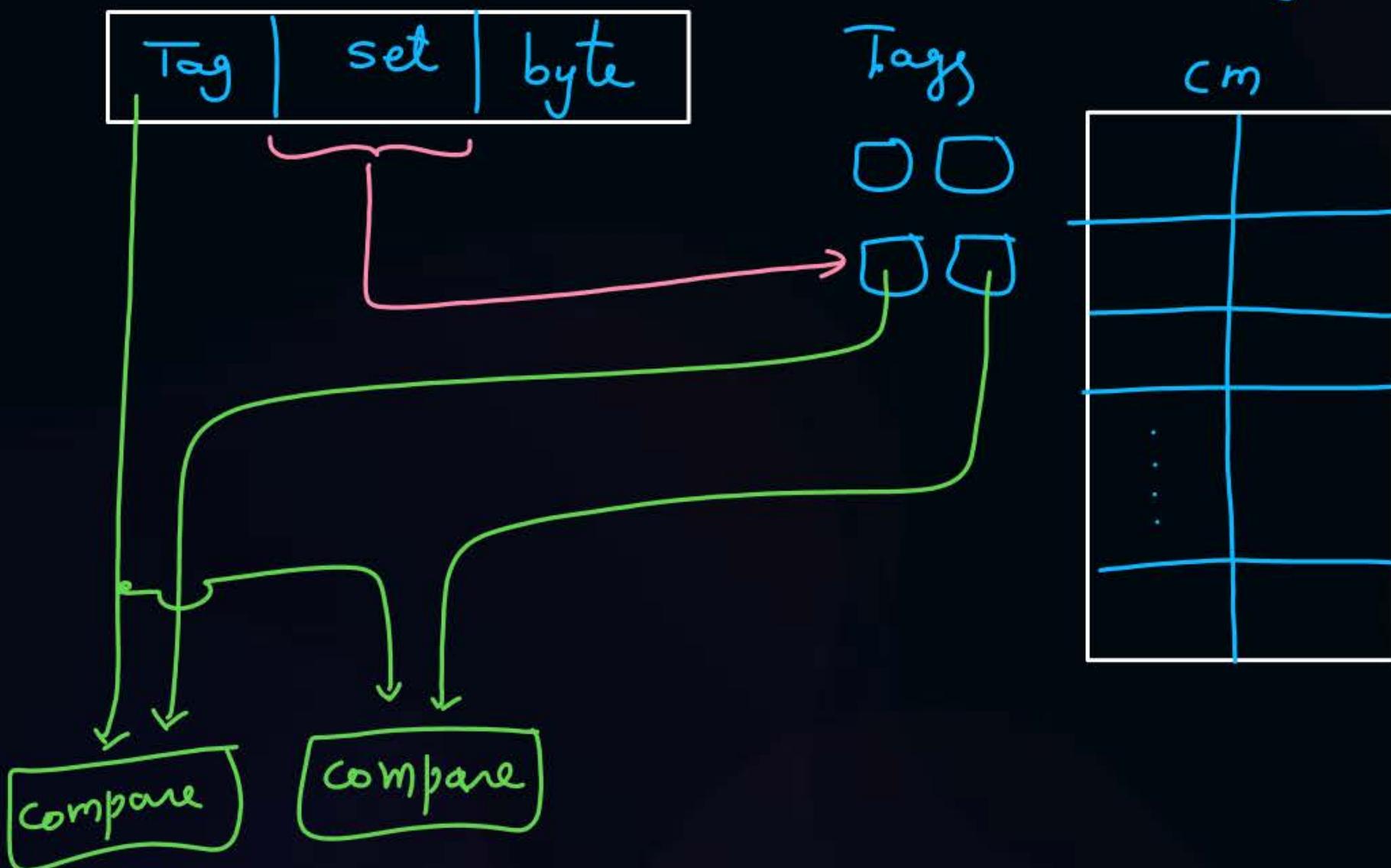


# Topic : Checking Hit/Miss in Set Associative Mapping



mm add.

2-way set



Finding cache set no - from mm address :-

1. Binary
2. Hexadecimal
3. Decimal



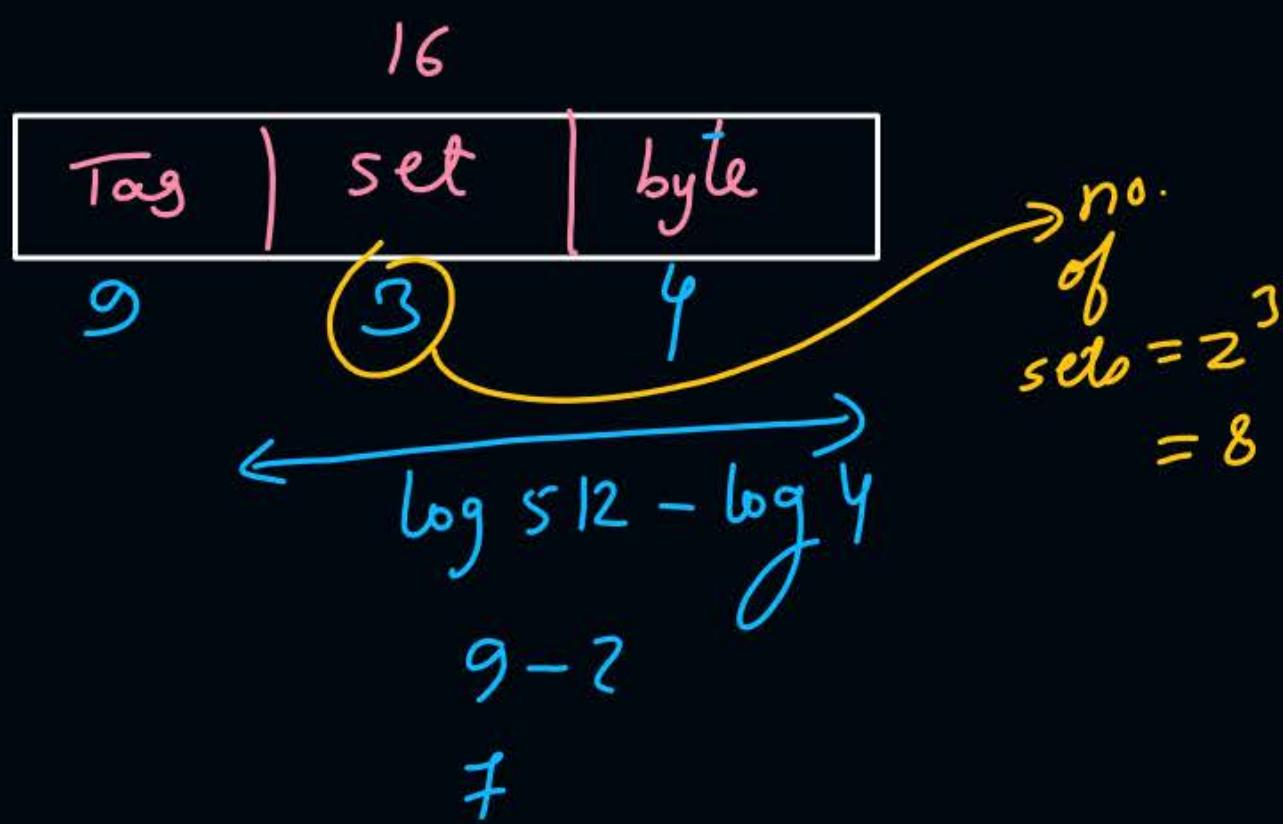
ex:-

$$cm\ size = 512 B$$

$$block = 16 B$$

4-way set ass.

$$mm\ add. = 16 bits$$



mm add.	cm set no.
101011100101010	$(010)_2 = (2)_{10}$
0101001100 <u>11</u> 0001	$(011)_2 = (3)_{10}$
$(C6B2)_{16}$	$(011)_2 = (3)_{10}$
$(374A)_{16}$	$(100)_2 = (4)_{10}$
$(10239)_{10}$	$639 \% 8 = (7)_{10}$
mm block no. = $\left[ \frac{10239}{16} \right]$ = 639	

If mm add. given in decimal:-

$$\text{mm block no.} = \left\lfloor \frac{\text{mm add.}}{\text{block size}} \right\rfloor$$

$$\text{byte offset} = \text{mm add. \% block size}$$

$$\text{cm set no.} = (\text{mm block no.}) \% \text{ no. of sets in cache}$$

$$\text{Tag} = \left\lfloor \frac{\text{mm block no.}}{\text{no. of sets in cache}} \right\rfloor$$



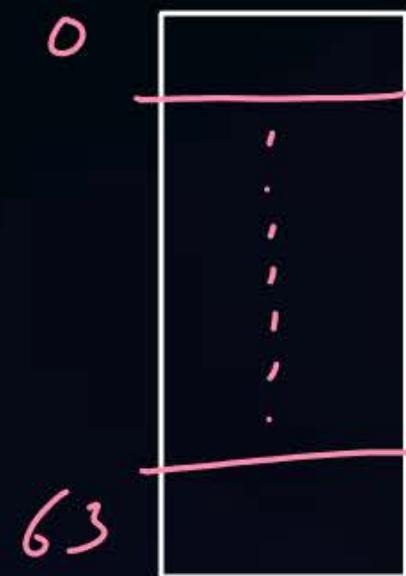
## Topic : Fully Associative Mapping

P  
W

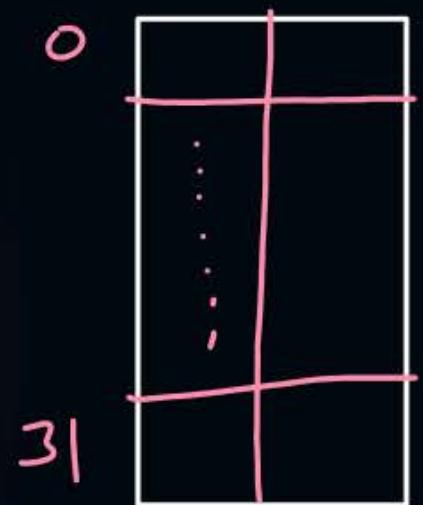
$$\log_2^{-1} = \circ$$

assume cache contains 64 blocks

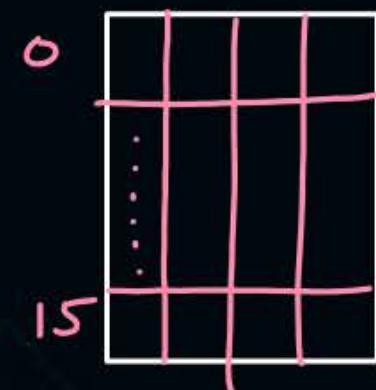
Direct :-



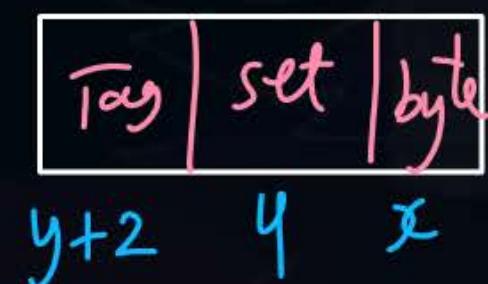
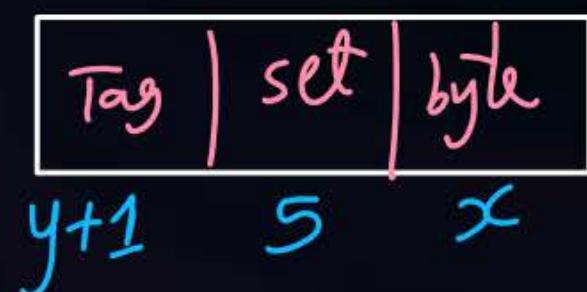
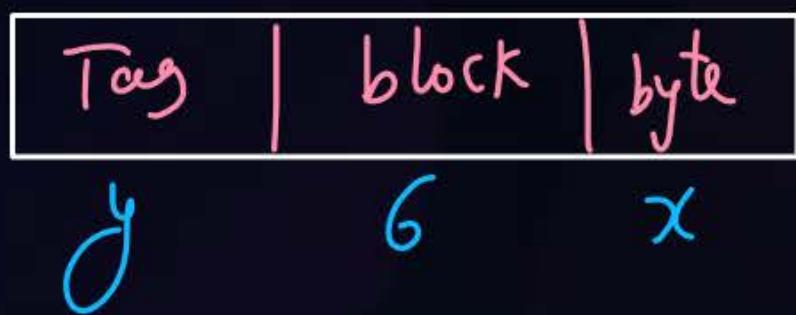
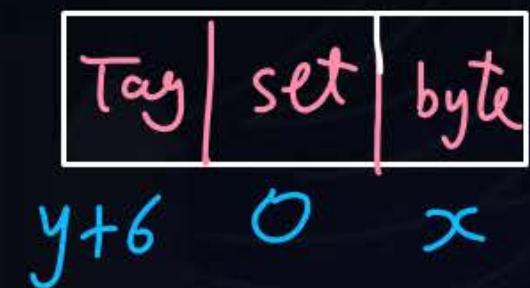
2-way



4-way



64-way





## Topic : Fully Associative Mapping

→ all blocks of cm come under  
single set.

P  
W

mm add.

Index  $\Rightarrow$  0 bit



Tag directory size = no. of blocks in cache \* (Tag + extra bits)



## 2 mins Summary



**Topic**

Set Associative Mapping

**Topic**

Fully Associative Mapping



# Happy Learning

## THANK - YOU