CS & IT ENGINEERING

COMPUTER ORGANIZATION AND ARCHITECTURE

Cache Organization

Lecture No.- 05

By- Vishvadeep Gothi sir

Recap of Previous Lecture

Topic Cache Mapping

Topic Direct Mapping

Topic Tag

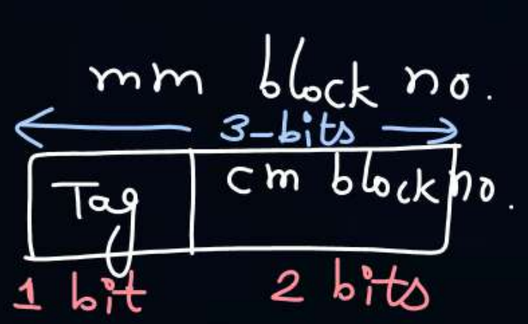
Topics to be Covered

- Topic: Cache Mapping
- Topic: Direct Mapping
- Topic: Tag
- Topic: Tag Directory

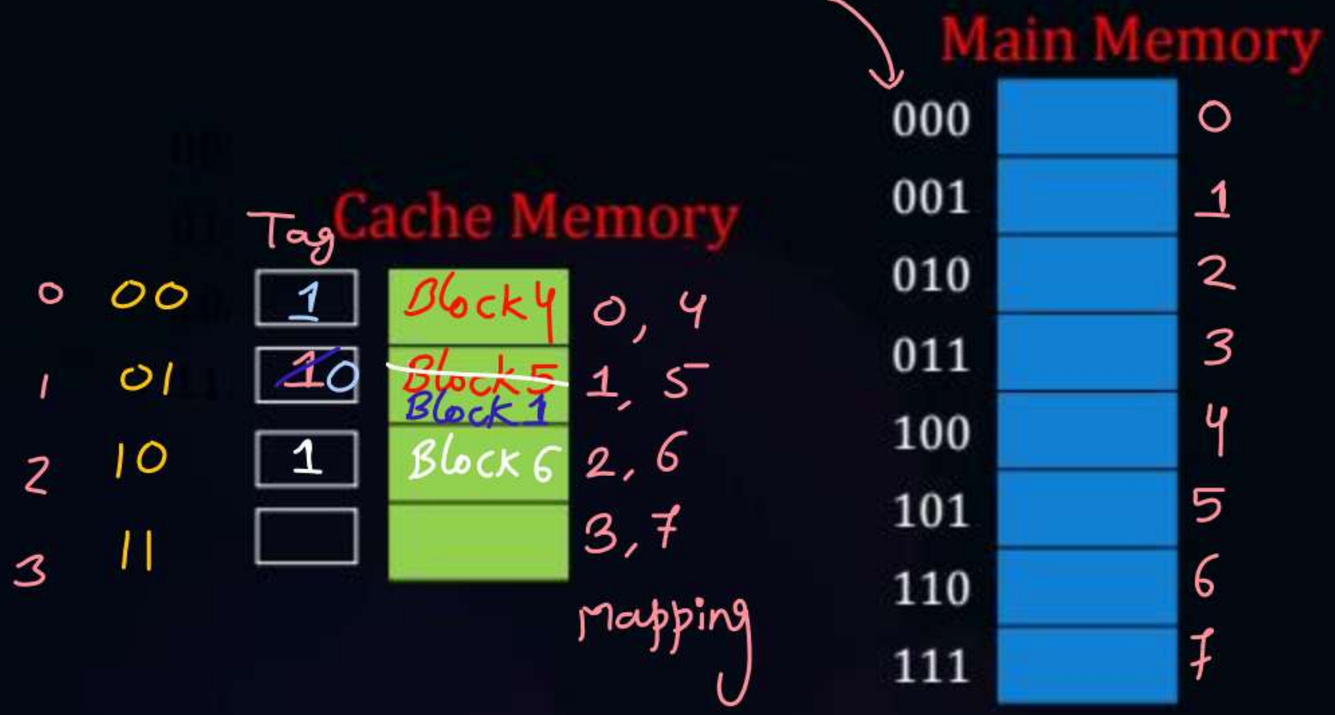
Topic : Direct Mapping

- Blocks in cache = 4 (00-11)
- Blocks in Main memory = 8 (000-111)

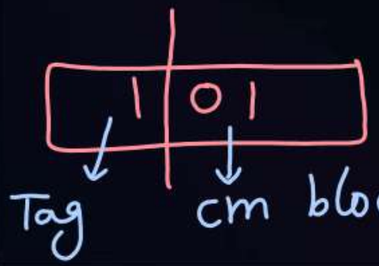
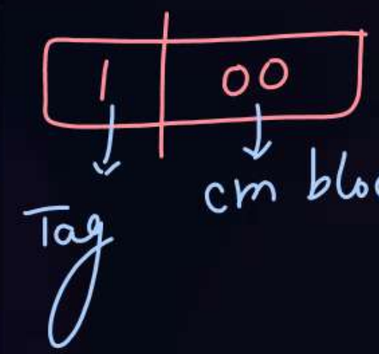
Topic : Direct Mapping



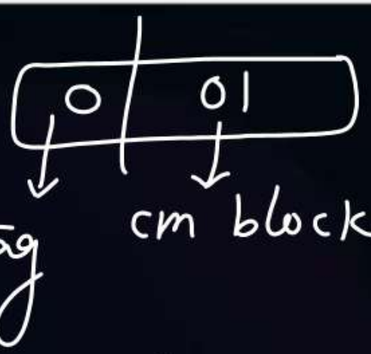
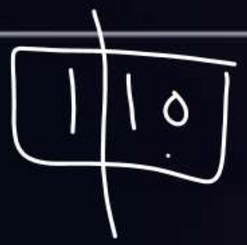
mm block no.



Topic : Direct Mapping

CPU Request (MM block)	Mapping (CM block no.)	Hit /Miss	Comments
$(5)_{10}$ $= (101)_2$	$5\ \%\ 4 = 1$ 	Miss	Bring block 5 at cm block 1 with tag 1
$(4)_{10} = (100)_2$		Miss	Bring block 4 at cm block 0 with tag 1

Topic : Direct Mapping

CPU Request (MM block)	Mapping (CM block no.)	Hit /Miss	Comments
$(1)_{10}$ $(001)_2$		Miss	Bring mm block 1 at cm block 1 by replacing block 5 and change the tag to 0.
$(110)_2$			

Topic : Direct Mapping

- Blocks in cache = 4 (00-11)
- Blocks in Main memory = 8 (000-111)
- Block Size = 2 Bytes
- Size of Cache memory = $4 * 2 = 8$ bytes
- Size of Main memory = $8 * 2 = 16$ bytes $= 2^4$ bytes
- Size of Main memory address = 4 - bits
  (byte address)

Topic : Direct Mapping

mm. address

cm block no. (Index)

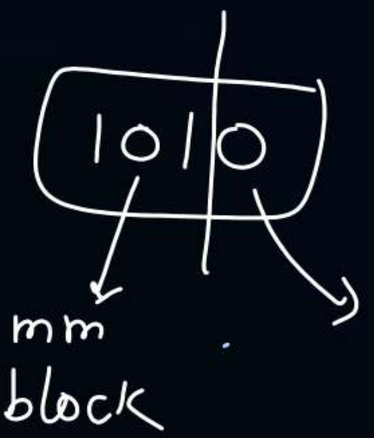
Cache Memory

00			0, 4
01	1	Block 101	1, 5
10			2, 6
11			3, 7

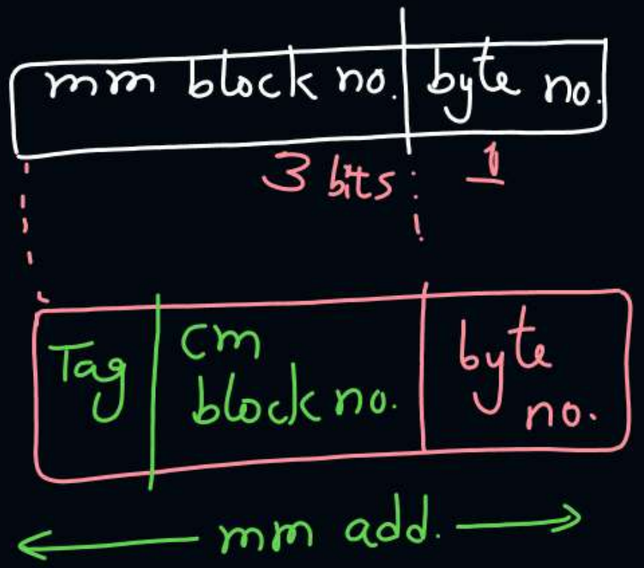
Tag blocks mapping

Main Memory

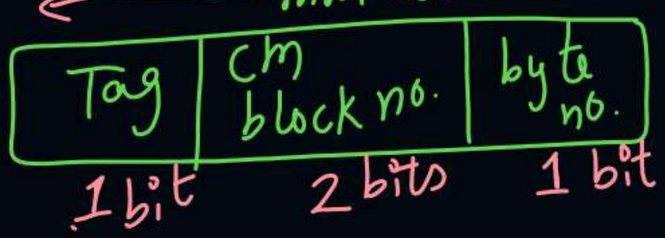
0000	block 000
0001	
0010	block 001
0011	
0100	block 010
0101	
0110	block 011
0111	
1000	block 100
1001	
1010	block 101
1011	
1100	block 110
1101	
1110	block 111
1111	



mm address



for our example

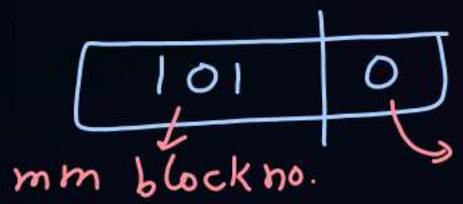
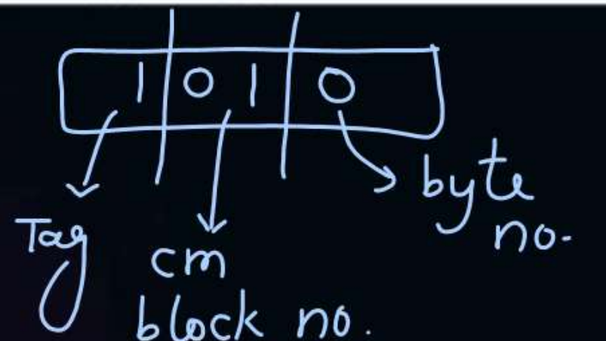
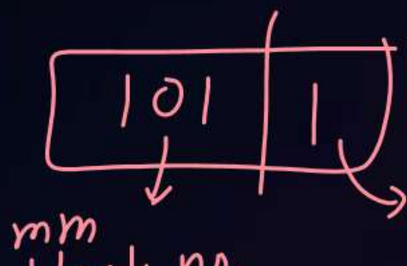
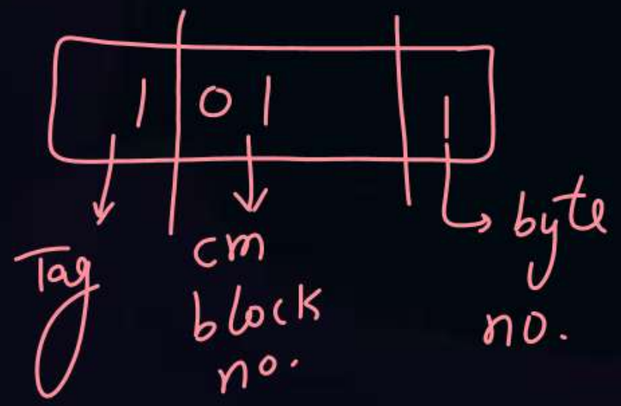


$$\text{no. of bits for byte no.} = \log_2(\text{block size})$$

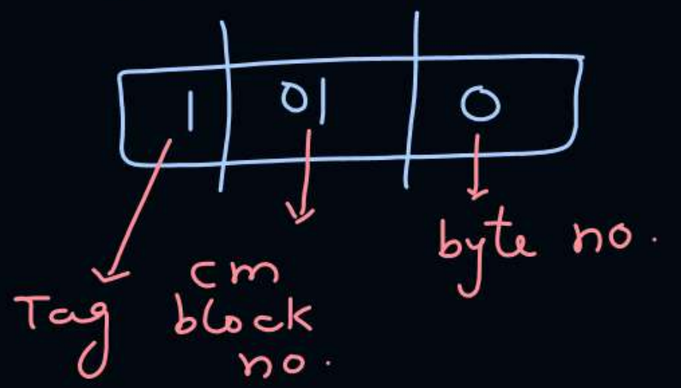
$$\text{no. of bits for cm block no.} = \log_2(\text{no. of blocks in cache})$$

$$\text{no. of blocks in cache} = \frac{\text{cache size}}{\text{block size}}$$

Topic : Direct Mapping

CPU Request (MM add.)	Mapping(CM block no.)	Hit/Miss	Comments
$(1010)_2$ 		Miss	Bring mm block 101 in cm at block 01 with tag 1
$(1011)_2$ 			

CPU generates add. 1010



↳ goto cm block 01 ⇒ no any block present there

⇓

MISS

Topic : Direct Mapping

CPU generates add. 1011

CPU accesses byte no. 1 of this block

byte no.

Tag

cm block no.

goto cm block no. 01 ⇒ Tag there ⇒ 1

Tag in CPU's request ⇒ 1

match

hit

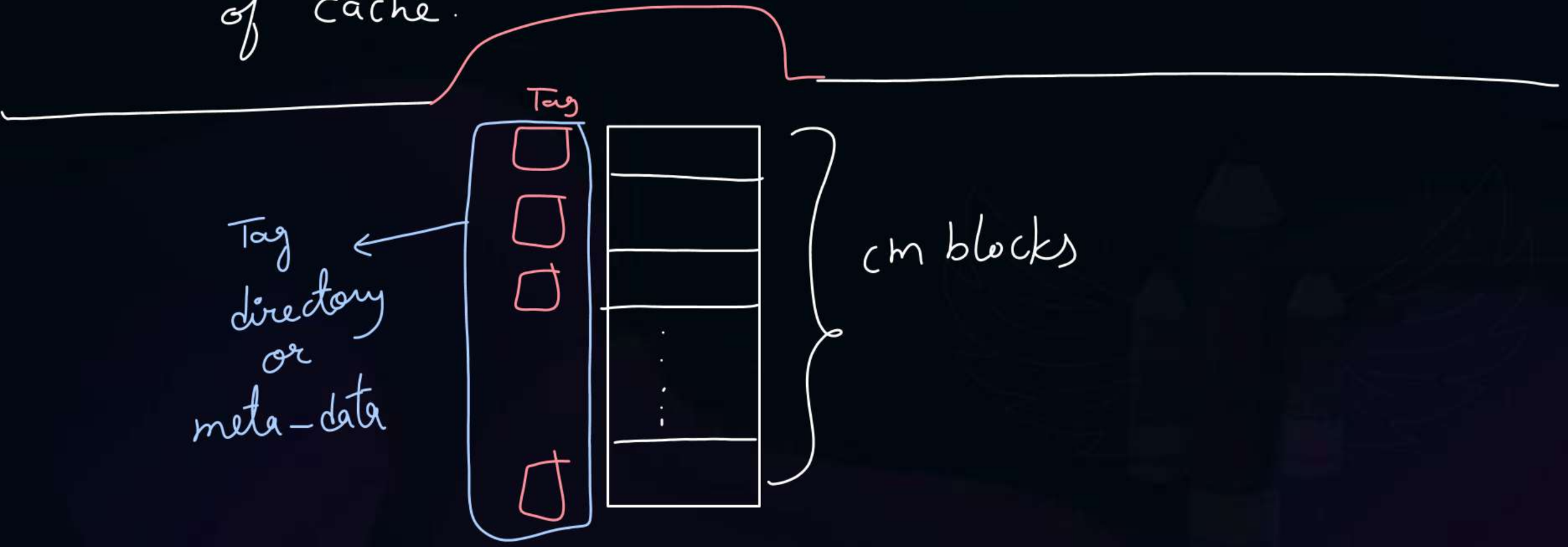
Topic : Indexing in Direct Mapping

cm block no. is known as index in direct mapping

cm block $\Leftrightarrow$ cm line

Topic : Cache Controller

Cache controller maintains one tag information for each block of cache.



Topic : Tag Directory

Tag directory size = no. of blocks in cache * Tag bits

ex:-

mm add. = 12-bits

cm size = 64 bytes = 2^6 B

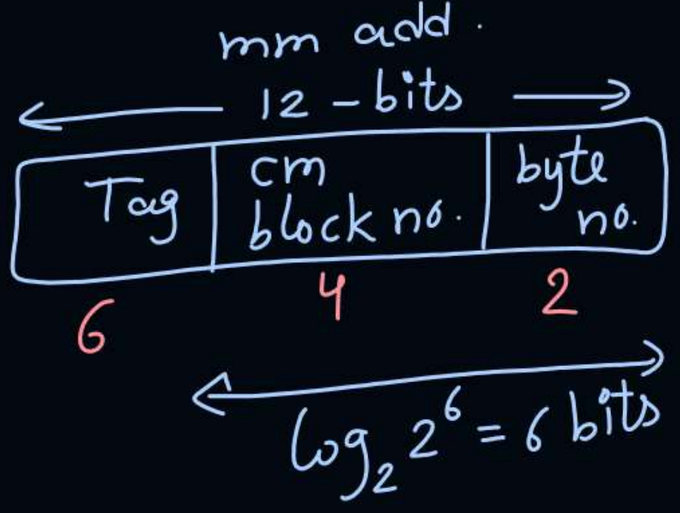
block size = 4 bytes

Direct mapping

block size = 4 B = 2^2 B ⇒ 2 bits

no. of block in cm = $\frac{64\ B}{4B} = 16 = 2^4$

cm block no. = 4 bits



Tag directory size = 16 * 6

= 96 bits

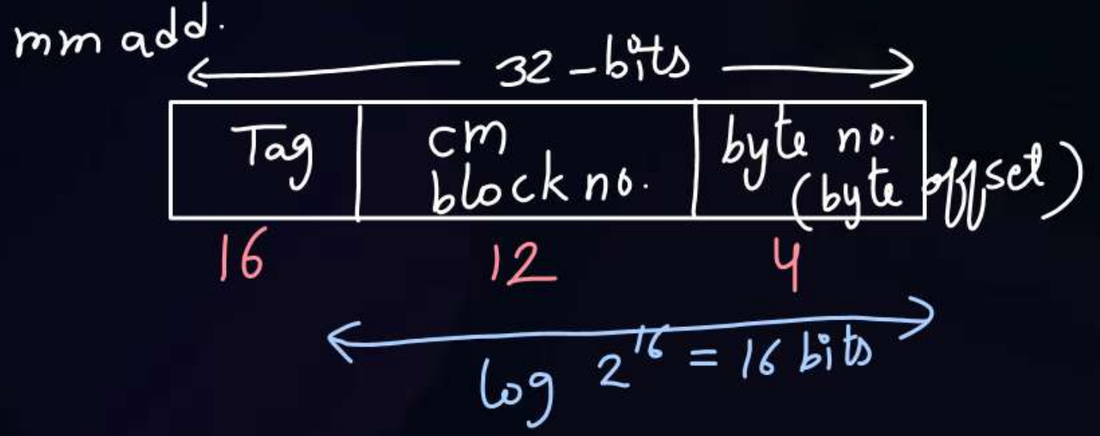
[NAT]

2^{16} B

2^{4}

#Q. Consider a direct mapped cache of size 64KB with block size 16 Bytes. The CPU generates 32-bits addresses.

1. Number of bits for byte offset? 4 bits
2. Number of blocks in cache? 2^{12} or 4k
3. The number of bits needed for cache indexing? 12 bits
4. The number of tag bits? 16 bits
5. Tag Directory size? $2^{12} * 16$ bits = 64 k bits = 8k bytes



$$\text{no. of blocks in cache} = \frac{64\text{kB}}{16\text{B}} = 4\text{k} = 2^{12} \Rightarrow \text{cm block no.} = 12 \text{ bits}$$

Direct mapping :-

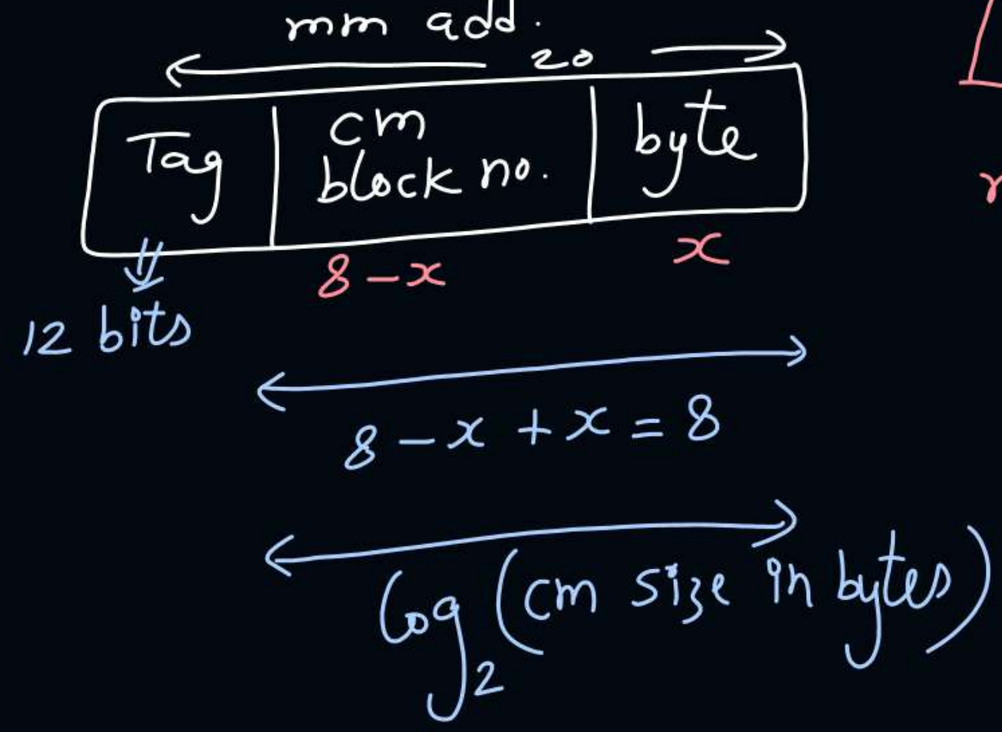
ex :- mm add. = 20 bits

cm size = 256 bytes = 2^8 B

assume block size = 2^x bytes

↓

byte no. = x bits



no. of blocks in cache = $\frac{2^8 B}{2^x B} = 2^{8-x}$

⇓

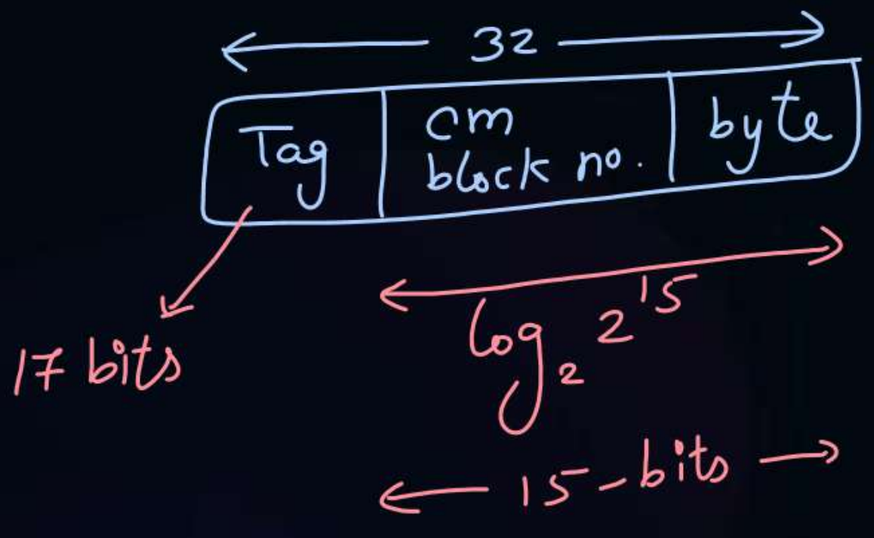
cm block no. = (8 − x) bits

[NAT]

2^{15} B

#Q. Consider a direct mapped cache of size 32KB. The CPU generates 32-bits addresses. The number of tag bits in main memory address are?

↳ 17 bits



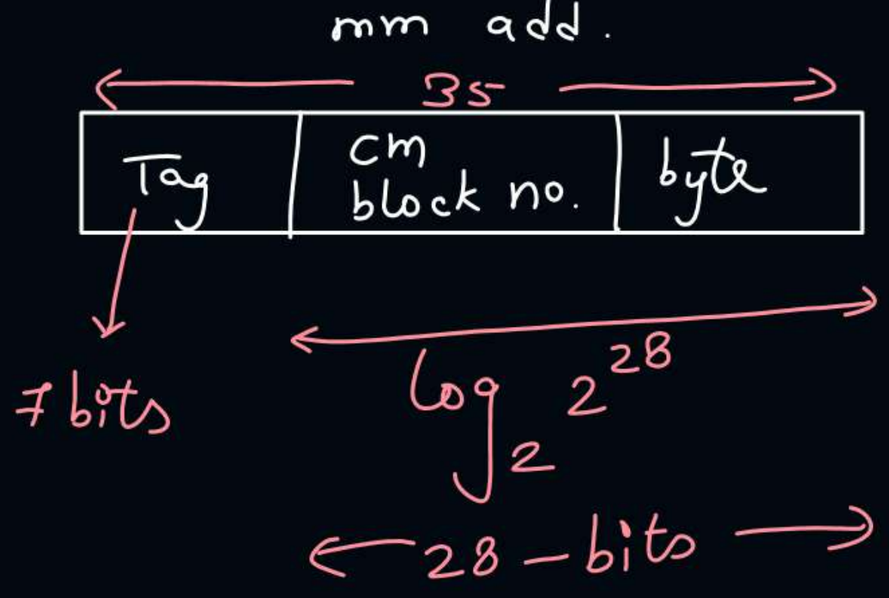
#Q) cm size = 256 MB = 2^{28} B

mm add. = 35 bits

Tag = ____ bits

Direct mapping

Ans = 7

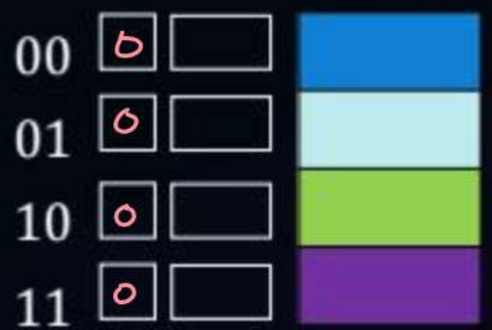


Topic : Cache Initialization

Topic : Cache Initialization

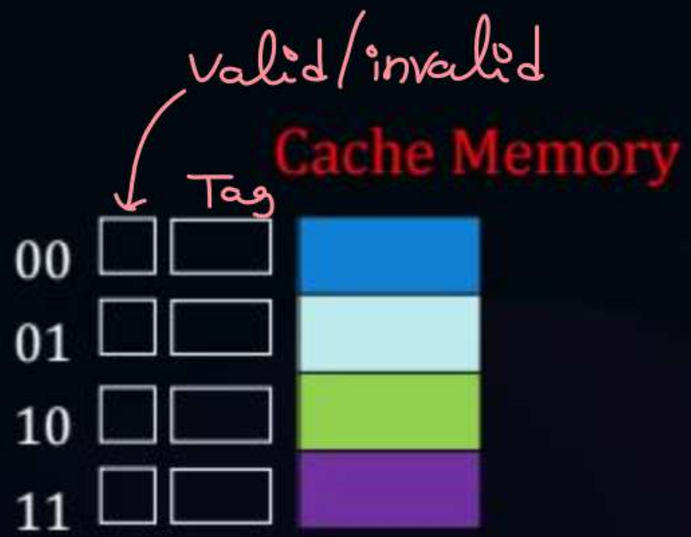
valid/invalid bit ⟶ 0 ⇒ invalid block & Tag

1 ⇒ valid block & Tag

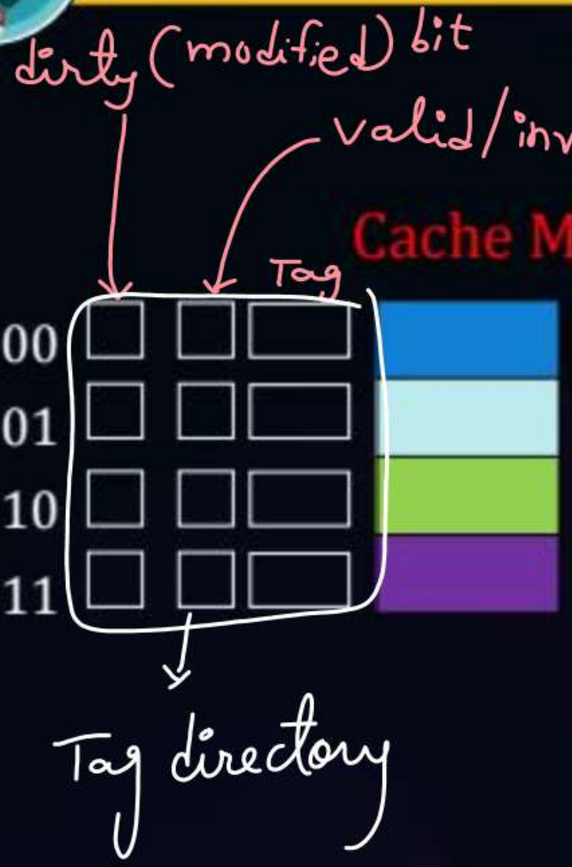


When computer starts all valid/invalid bits are initialized with 0.

Topic : Performance Improvement of Write Back Cache



Topic : Performance Improvement of Write Back Cache



Dirty bit

0

CPU has performed only read in that block

1

CPU has performed write in that block of cache

(if such blocks are replaced from cache then write back required for these blocks)

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

[NAT]

#Q. Block Size = 16 bytes $= 2^4 B$

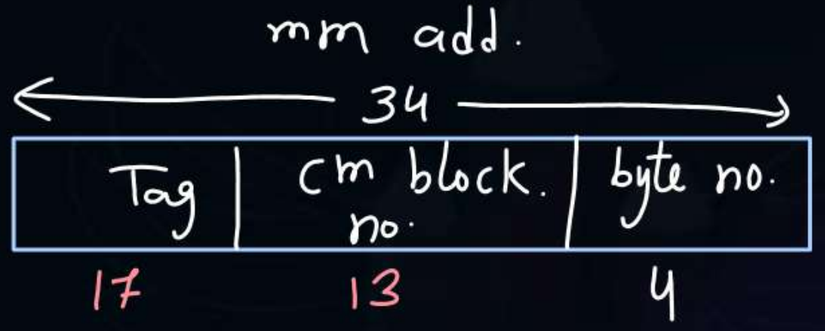
Size of Cache memory = 128KB $= 2^{17} B$

Size of Main memory address = 34-bits

Direct Mapping

For each block apart from tag 1 valid bit and 1 modified bit are stored in cache

no. of blocks in cache $= \frac{2^{17} B}{2^4 B} = 2^{13}$



Bits in byte offset? 4 bits

Bits in cache block number? 13 bits

Bits in tag? 17 bits

Tag Directory size? $2^{13} * (17 + 1 + 1) = 2^{13} * 19$ bits $= 152$ k bits

$\log 2^{17} = 17$ bits

[NAT]

mm Size = $2^{23} * 2^{6} B = 2^{29} B$

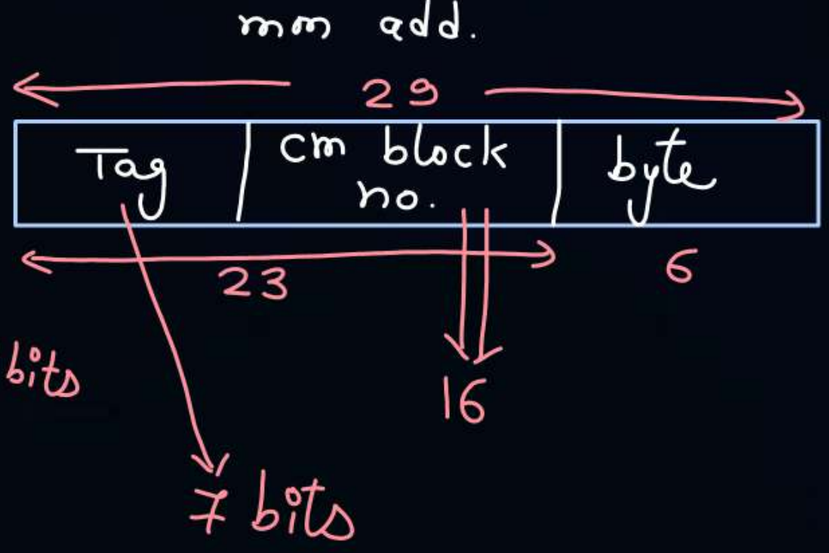
↳ mm add. = 29 bits

#Q. Blocks in Main memory = 2^{23} → mm block no. = 23 bits

Blocks in Cache memory = 2^{16}

Block Size: 64 Bytes

Direct Mapping



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 = ? $2^{16} * 7$ bits

[NAT]

1 word = 32 bits = 4 bytes

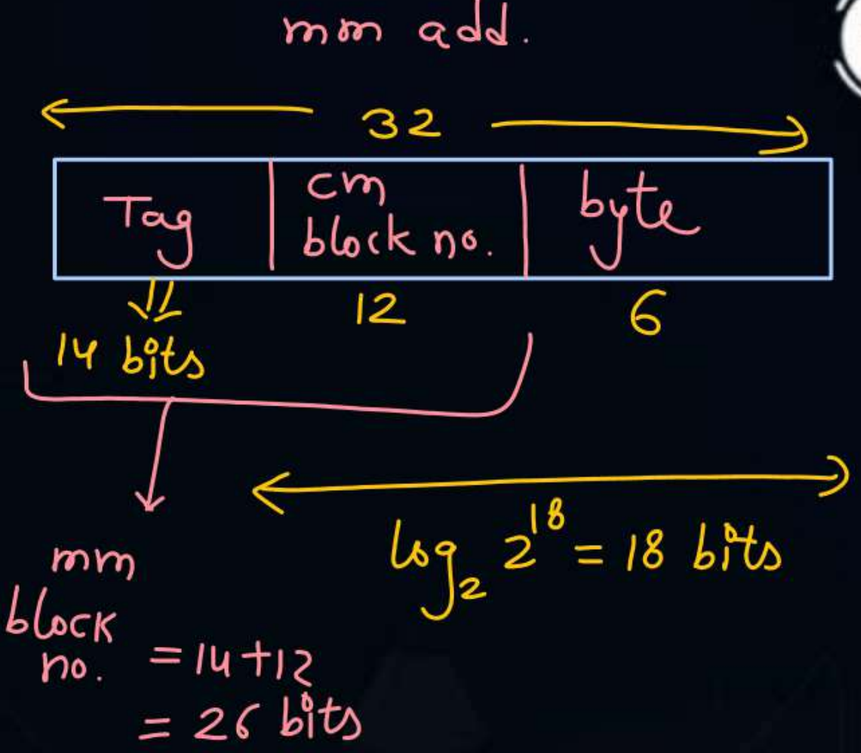
#Q. 32-bit architecture CPU

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

Cache Size = 256KB $= 2^{18}$ B

Block Size = 16 Words $= 16 * 4B = 64B = 2^{6}$ B

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 = ? $(2^{12} * 14$ bits$)$

2 mins Summary

Topic Cache Mapping

Topic Direct Mapping

Topic Tag

Topic Tag Directory

Happy Learning

THANK - YOU