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

Topics to be Covered

- **Topic** Cache Mapping
- **Topic** Direct Mapping
- **Topic** Tag & Index

Topic : Direct Mapping

Cache	0	1	2	3	4	5	6	7	8	9

Main Memory	00	01	02	03	04	05	06	07	08	09
	10	11	12	13	14	15	16	17	18	19
	20	21	22	23	24	25	26	27	28	29
	30	31	32	33	34	35	36	37	38	39
	:	:	:	:	:	:	:	:	:	:
	90	91	92	93	94	95	96	97	98	99

Topic : Direct Mapping

mm block no.

Tag	cm block no.

mm block no.
← 3 bits →

Tag	cm block no.
1	2

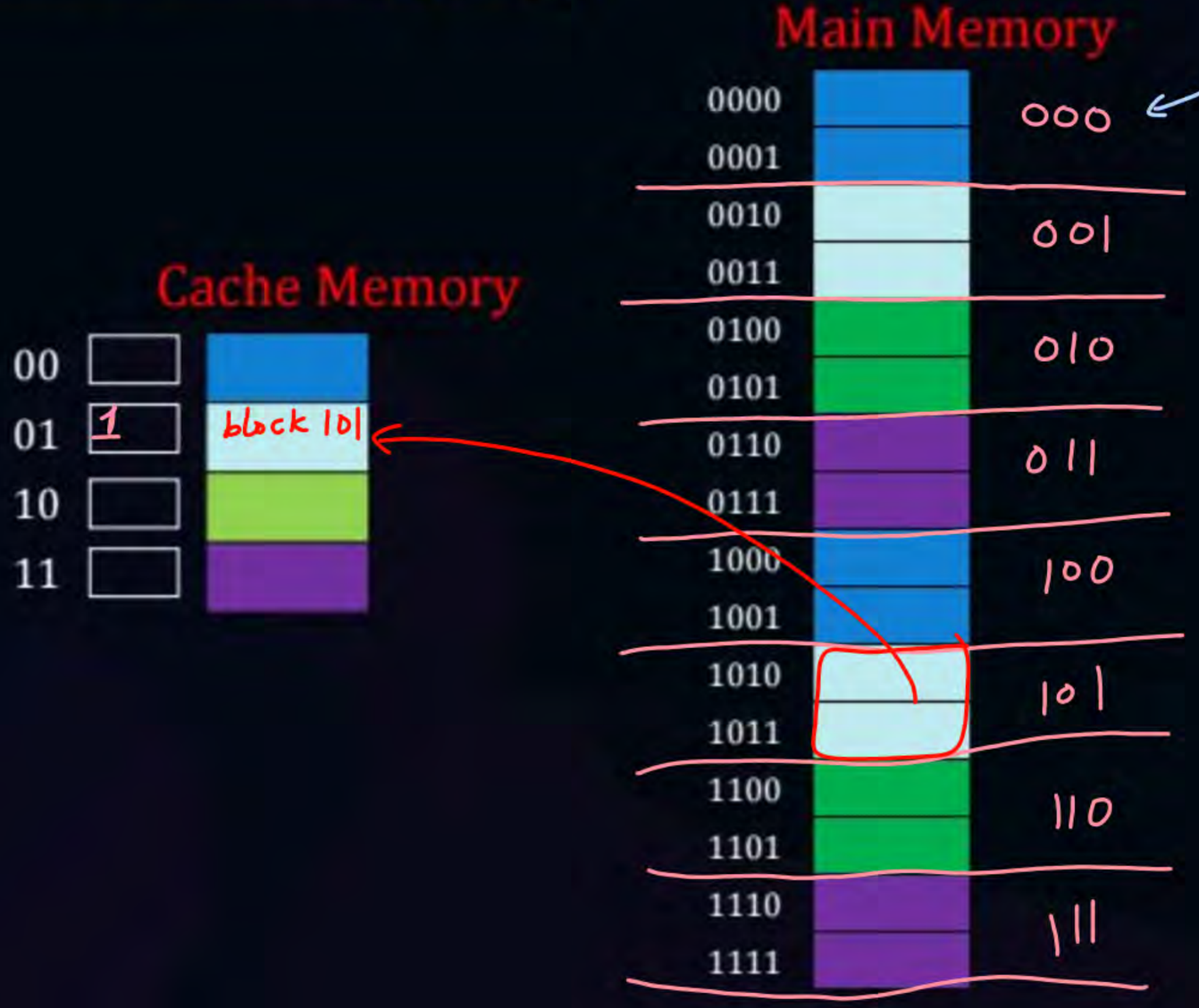
→ cm line no.

no. of bits required for cm block no. $= \log_2$ (no. of blocks in cm)

Topic : Direct Mapping

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

Topic : Direct Mapping

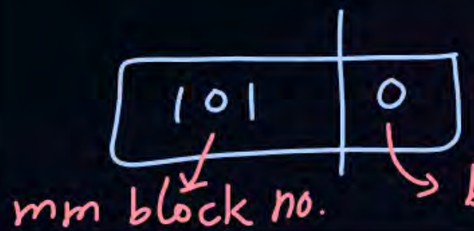
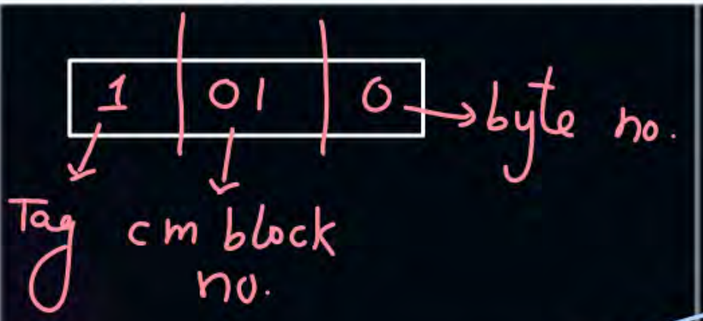
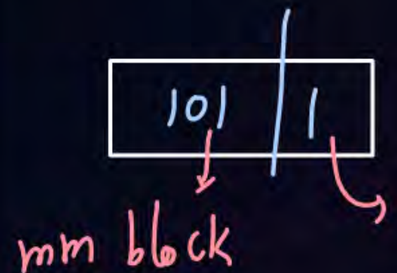
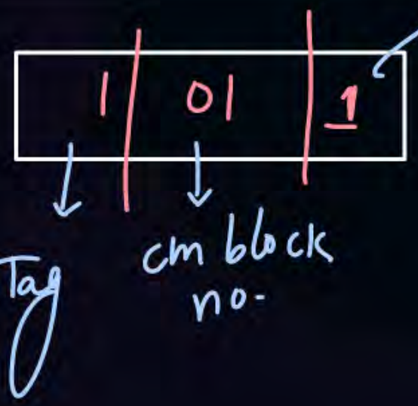


← mm add. →

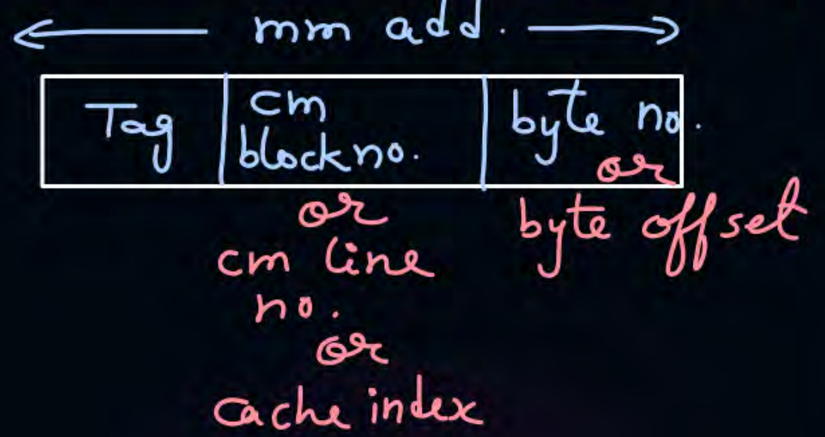
mm block no.	byte no. of the block

Tag	CM block no.	byte no.

Topic : Direct Mapping

CPU Request (MM add.)	Mapping(CM block no.)	Hit/Miss	Comments
mm add. = 1010 		Miss	bring mm block no. 101 into cache block 01 with tag 1
mm add. = 1011 		Hit	byte 1 of the block is sent to CPU for access.

Topic : Direct Mapping



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

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

Topic : Indexing in Direct Mapping

cm block no. ⟹ index

cm

00

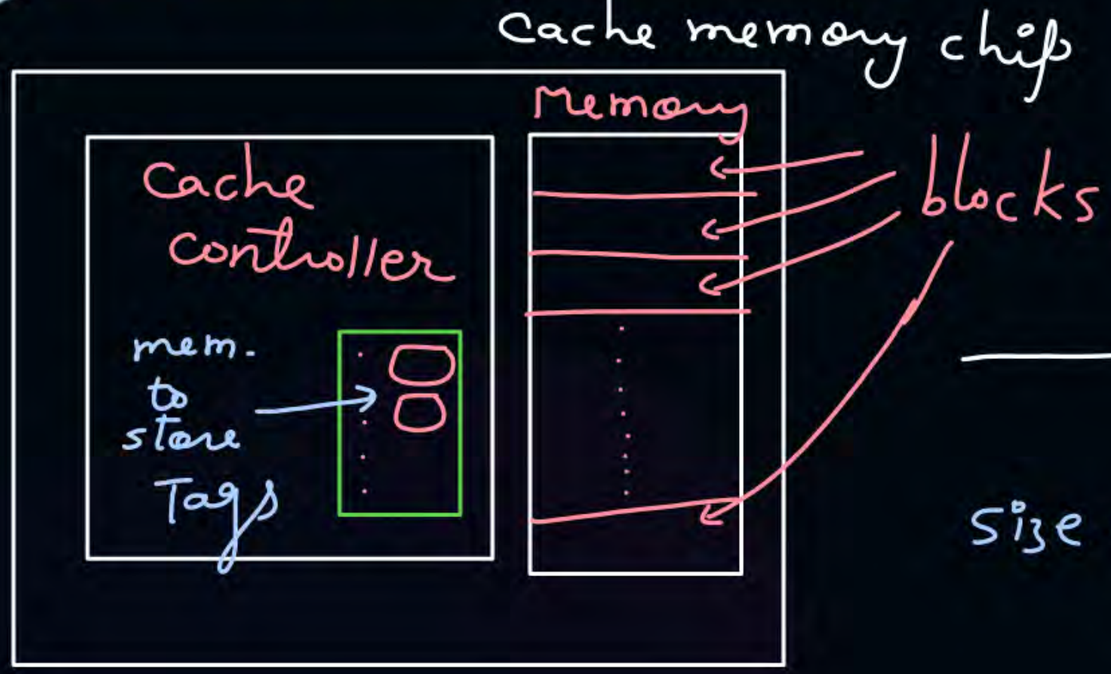
01

10

11

4 blocks

Topic : Cache Controller



Cache controller stores one tag info^{n} per block of cache

Size of tag directory = no. of blocks in cache * Tag-bits

or

Size of meta data

#Q.)

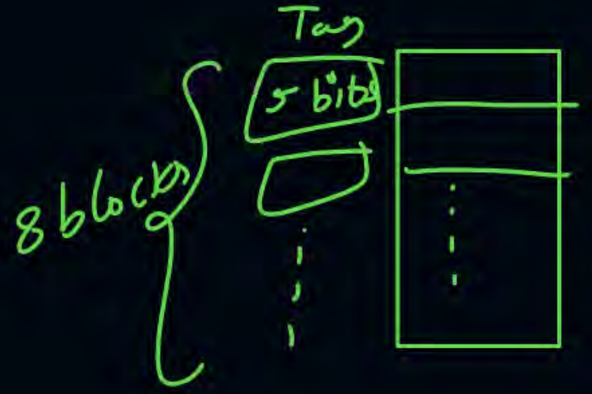
Cache memory = 128 bytes

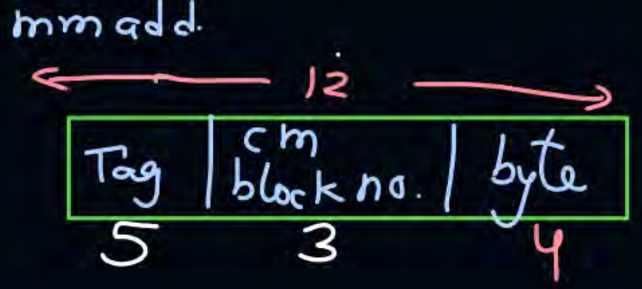
block size = 16 bytes = 2^4 B

mm add. = 12 bits

Direct mapping

byte no. = $\log_2 2^4$ = 4 bits





no. of blocks in cache = $\frac{128\,B}{16\,B} = \frac{2^7}{2^4} = 2^3$

cm block no. = 3 bits

Tag directory size = 2^3 * 5 bits

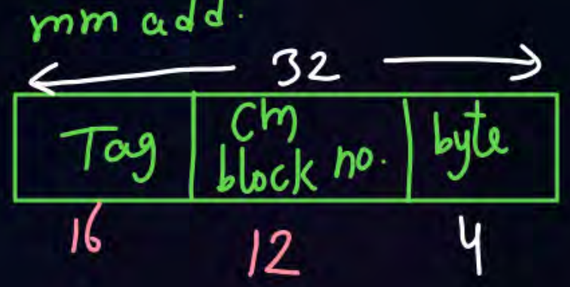
= 40 bits

[NAT]

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}
3. The number of bits needed for cache indexing? 12 bits
4. The number of tag bits? 16 bits
5. Tag Directory size? $= 2^{12} * 16 \text{ bits} = 2^{10} \cdot 2^2 * 16 \text{ bits} = 64 \text{ k bits} = 8 \text{ k bytes}$



no. of blocks in cm $= \frac{64\text{k}\cancel{\text{B}}}{16\cancel{\text{B}}}$

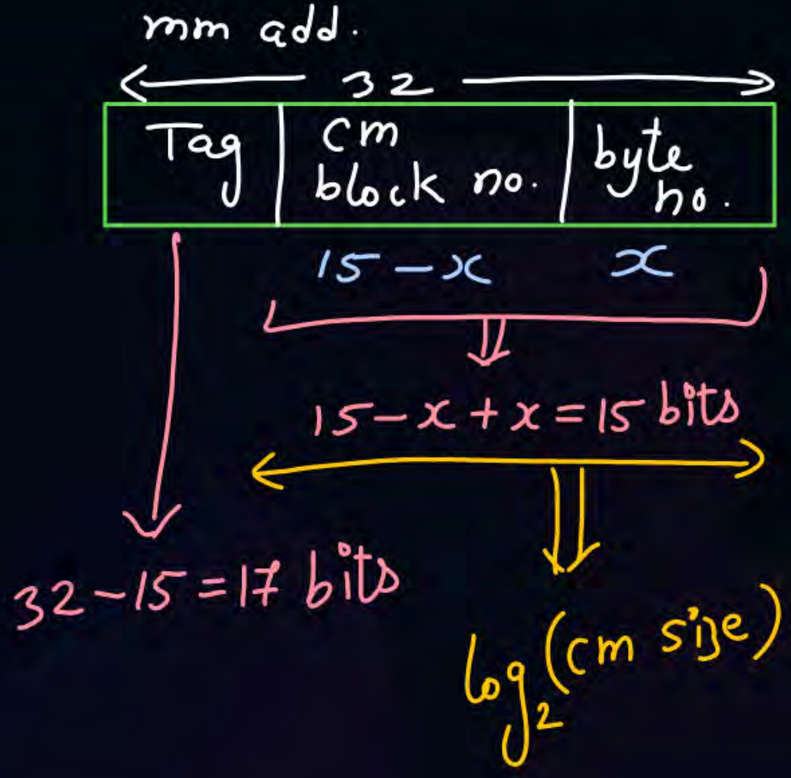
$= \frac{2^6 \cdot 2^{10}}{2^4} = 2^{12}$

[NAT]

Ans = 17

#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?

$32KB = 2^{15}B$



assume block size $= 2^x$ bytes

no. of bits needed for byte no. $= x$ bits

$$\text{no. of blocks in cm} = \frac{32kB}{2^x B}$$

$$= \frac{2^{15}\,\cancel{B}}{2^x\,\cancel{B}} = 2^{15-x}$$

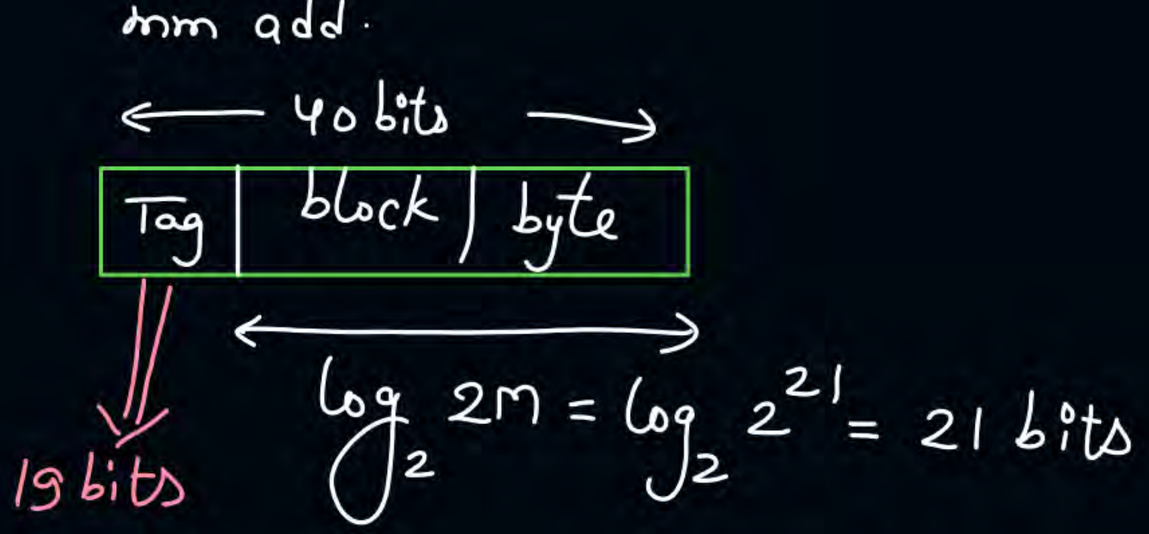
cm block no. $= (15-x)$ bits

Ques) CM size = 2MB

mm add. = 40 bits

Tag = 19 bits

Direct mapping



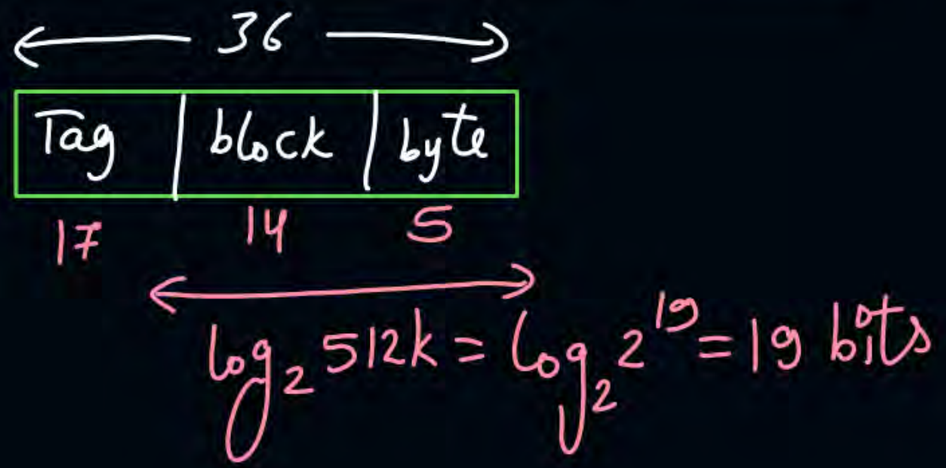
Ques) CM size = 512kB

block size = 32 B = 2^5 B

Direct mapping

mm add. = 36 bits

Tag directory size = $2^{14} * 17$ bits



Topic : Cache Initialization

Topic : Cache Initialization

valid/invalid ⇒ to show block & Tag are valid or not

0	Invalid
1	valid

when system turns on all V/I bits are initialized to zero.

Topic : Performance Improvement of Write Back Cache

only modified blocks must

Topic : Performance Improvement of Write Back Cache

- 0 → not dirty
- 1 → dirty

only modified blocks are written back to mm when replaced from cache

Dirty bit stores for each block in tag directory only for write back cache

↳ never for write through cache

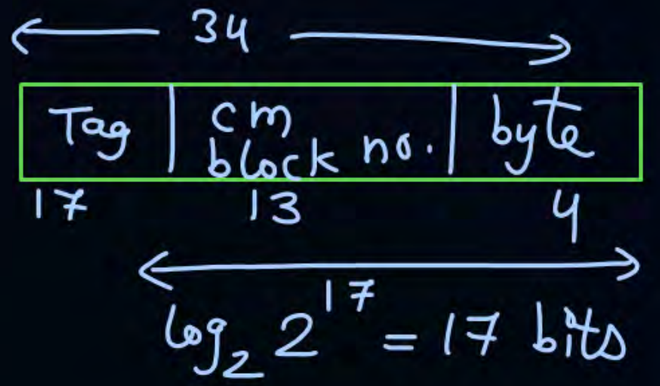
Tag directory size = no. of blocks in cm * (Tag + extra bits)

[NAT]

#Q. Block Size = 16 bytes

Size of Cache memory = 128KB

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

Bits in byte offset? 4

Bits in cache block number? 13

Bits in tag? 17

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

[NAT]

H.W.

#Q. Blocks in Main memory = 2^{23}

Blocks in Cache memory = 2^{16}

Block Size: 64 Bytes

Direct Mapping

No. of bits required for Byte Offset = ?

No of bits required for main memory address = ?

Index-bits = ?

Tag-bits = ?

Size of Tag Directory = ?

[NAT]

H.W.

#Q. 32-bit architecture CPU

Main Memory Size = 4GB

Cache Size = 256KB

Block Size = 16 Words

Direct Mapping

No. of bits required for Byte Offset = ?

No of bits required for main memory address = ?

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

Index-bits = ?

Tag-bits = ?

Size of Tag Directory = ?

Topic : Calculating CM Block Number from MM Address

1. mm add given in binary

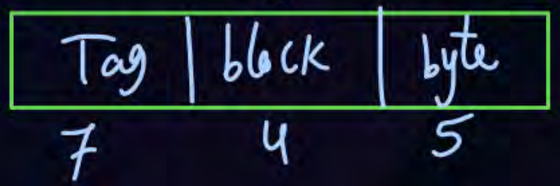
2. mm add. given in hexadecimal ⇒ Convert to binary

ex:- mm add. = 16 bits

cm size = 512 bytes = 2^9

block size = 32 bytes

Direct mapping



mm add.	cm block no.
1010101010101010 →	1010101 | 0101 | 01010
1100111011110000 →	1100111 | 0111 | 10000

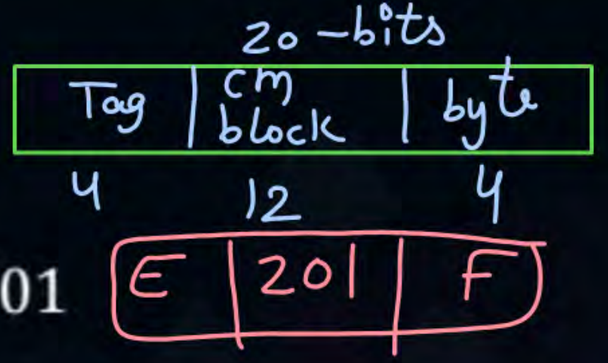
cm block no = $(0101)_2 = (5)_{10}$

cm block no = $(0111)_2 = (7)_{10}$

[MCQ]

mm add. = 20 bits

#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 $(E201F)_{16}$?

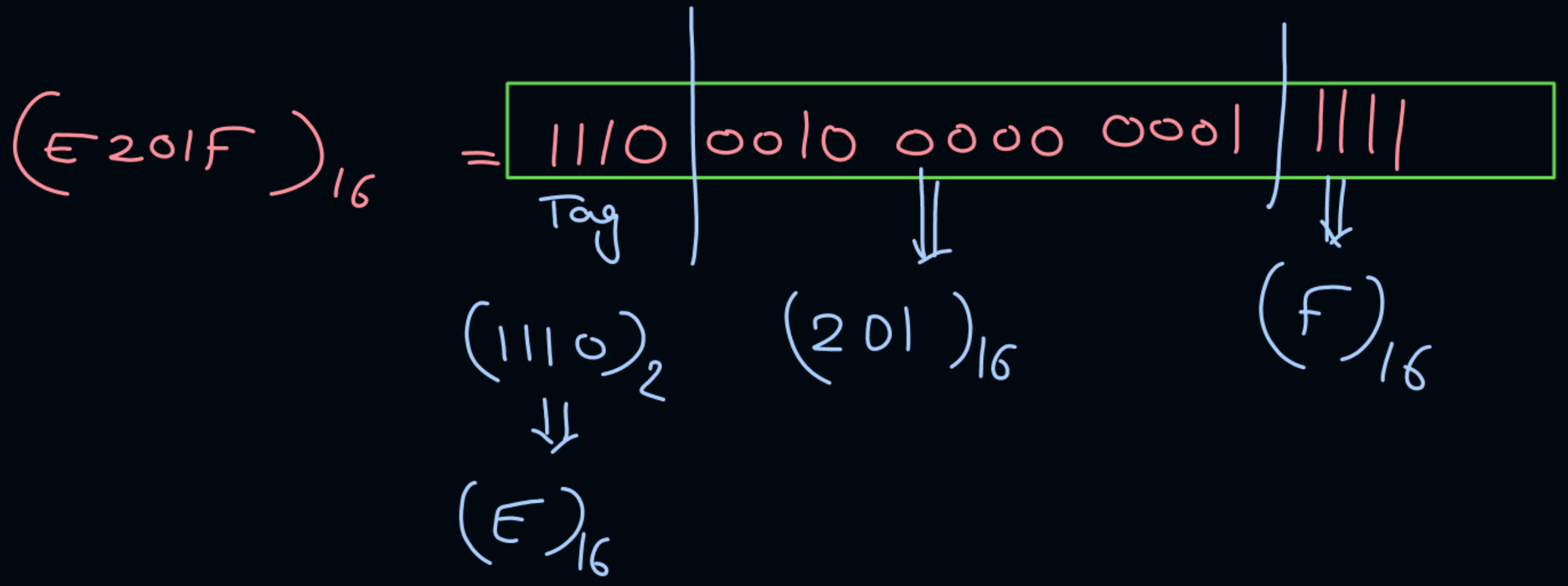


A ✓ E, 201

B F, 201

C E, E20

D 2, 01F


$(E201F)_{16}$
=
1110
0010 0000 0001
1111
Tag
$(1110)_2$
$(E)_{16}$
$(201)_{16}$
$(F)_{16}$

2 mins Summary

Topic Cache Mapping

Topic Direct Mapping

Topic Tag & Index

Happy Learning

THANK - YOU