

# Computer Organization & Architecture

## Gate Notes

For more visit [www.gatenotes.in](http://www.gatenotes.in)

# INDEX

Name : Rakesh Nama.  
Class :

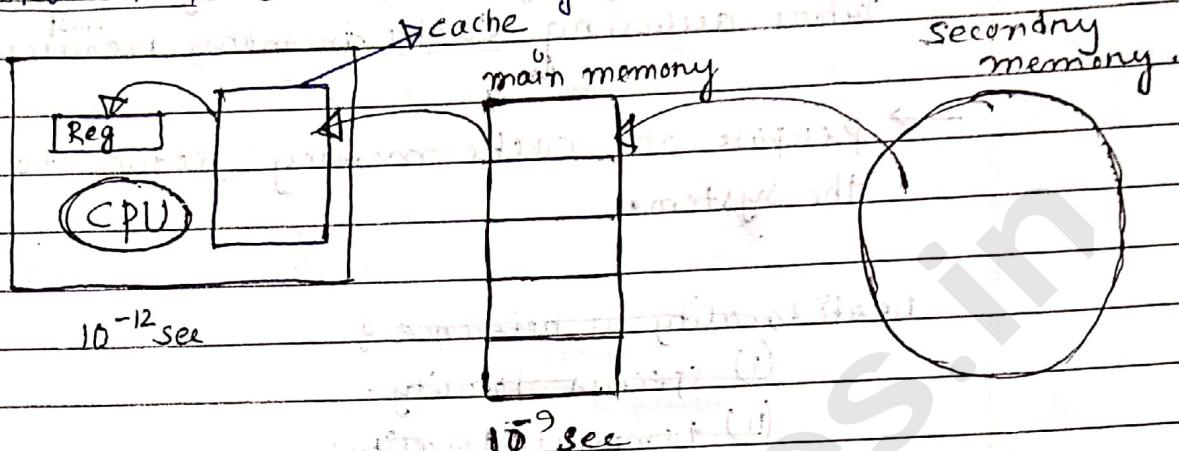
Subject : C.O.A

### Class :

Sec:

Roll No. :

- Introduction to cache memory:



→ cache memory faster than main memory.

• Cache hit: If required element present in cache, then it is called 'cache hit'. That,

• hit latency: Time taken to find out whether element present on the cache or not, that is called "hit latency".

• cache miss: If required element not present in cache, that is called 'cache miss'.

• Miss latency: Time taken to get something from main memory and then place it into the cache and then read that's called "miss latency".

• page hit: If required element present in main memory,

• Page fault: If required element not present in main memory,

• Tag directory: Tag directory says that required element present in tag or not.

→ Before accessing main memory we access page table,  
 Before accessing Cache memory we access Tags.

→ purpose of cache memory reduce to cost of  
 the system.

Types of locality of reference :-

(i) spatial locality :-

(ii) temporal locality :-

- Locality of reference :- is a term for the phenomenon in which the same values or related items from related storage locations are frequently accessed, depending on the memory access pattern.

### (Locality of reference)

#### ↓ (Temporal locality)

Recently referenced items are likely to be referenced in the near future.

#### ↓ (Spatial locality)

Items with nearby addresses tend to be referenced close together in time.

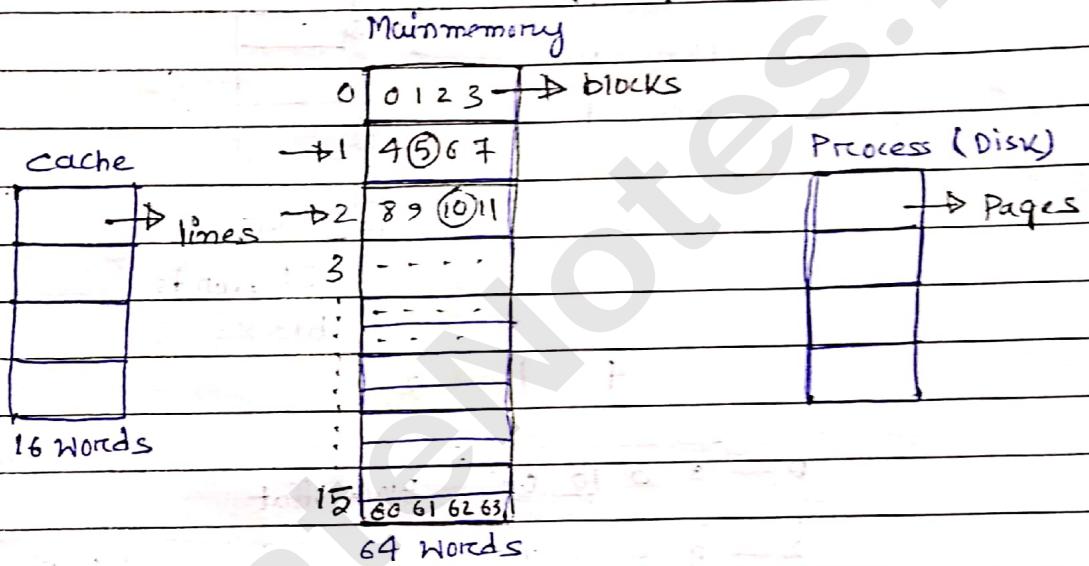
### • Introduction to Direct mapping

- Introduction to Direct mapping:

→ talking about Disk and main memory. No talking about paging.

→ talking about cache and " " " " " → BLOCKS.

→ Block size = Linesize.



→ smallest addressable unit called word.

let's

$1W = 1B$  (means our system is byte addressable)

Block size = 4 words

$$\text{No. of Block in main memory} = \frac{64}{4} = 16 \text{ BLOCK.}$$

$$\text{No. of Lines in cache} = \frac{16}{4} = 4 \text{ lines}$$

physical address contain = 6 bits



processor generate address = 

0	0	0	1	0	1
---	---	---	---	---	---

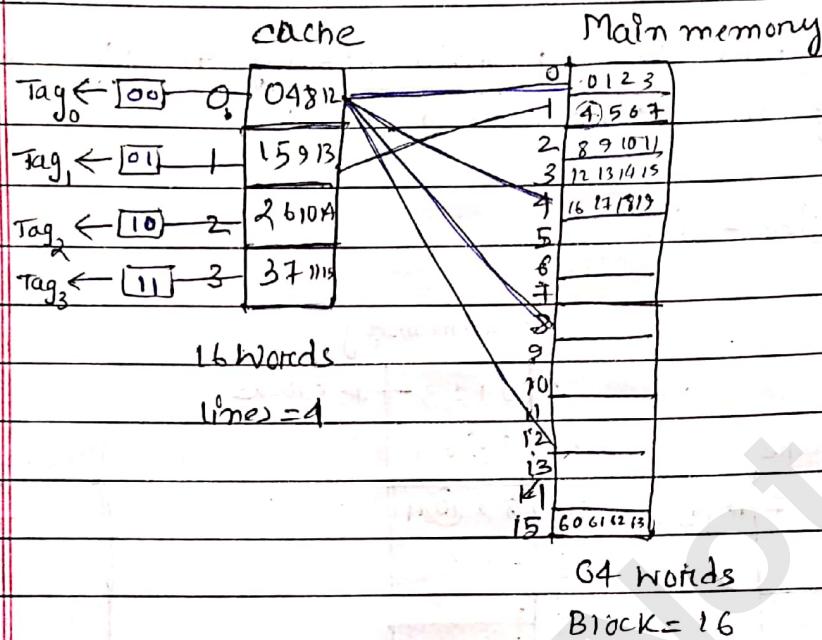
 (to word 5)  
B.I.K.      num      B.I.K. offset

= 

1	0	0	1	0	1
---	---	---	---	---	---

 (to word 10)  
B.no      B.off

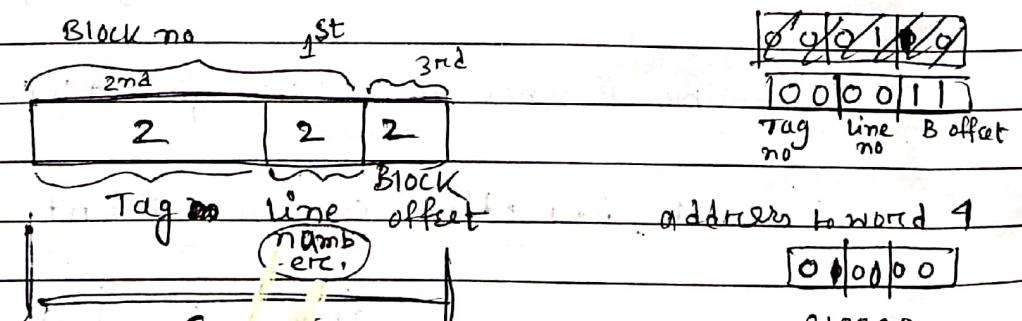
• Direct Mapping:



A table mapping cache lines to main memory words. The columns represent cache lines (0, 1, 2, 3) and the rows represent main memory words (0 to 15). The 'BLOCK offset' column shows the word number within each cache block. The last four columns show the actual memory addresses for each cache line.

	4	2		
0	0 0 0 [0 0] ✓		00 - 04812	
1	0 0 0 0 1		01 - 15913	
2	0 0 1 0		10 - 2610A	
3	0 0 1 1		11 - 37115	
4	0 1 10 0 ✓			
5	0 1 0 1			
6	0 1 1 0			
7	0 1 1 1			
8	1 0 [0 0] ✓			
9	1 0 0 1			
10	1 0 1 0			
11	1 0 1 1			
12	1 1 [0 0] ✓			
13	1 1 0 1			
14	1 1 1 0			
15	1 1 1 1			

32 16 8 4 2 1  
0 0 0 1 0 0  
\* [0 0] (1 0 0 1) - 3  
offset L.N. BKA  
address to word 3



1 Bit ~~and~~  
Byte = 8 bit  
 $KB = 2^{10} B$   
 $MB = 2^{10} KB$   
 $GB = 10^9 MB$

atlantis

Date \_\_\_\_\_

Page \_\_\_\_\_

5

## Direct Mapping Problem - 1

MM size

Cache size

Block size

Tag bits

Tag directory size

	MM size	Cache size	Block size	Tag bits	Tag directory size
✓ Q1	128 KB	16 KB	256 B	3 bit	$(3 \times 2^6)$ bit
✓ Q2	32 GB	32 KB	1 KB	10 bit	$(10 \times 2^5)$ "
✓ Q3	<del>64 MB</del> $2^{16} B$	512 KB	1 KB	7	$(7 \times 2^9)$ "
✓ Q4	16 GB	<del>512 KB</del> $2^{14} B$	1 KB	10	$(10 \times 2^{14})$ "
✓ Q5	64 MB	$2^{16} B$	can't guess	10	can't guess
✓ Q6	$2^{26} B$	512 KB	can't guess	7	can't guess

Assuming that memory is Byte addressable.

Q1:

$$\text{Main memory size} = 128 KB$$

$$= 2^{10} \times 128 B$$

$$= 2^{17} B$$

$$\text{Block size} = 256 B$$

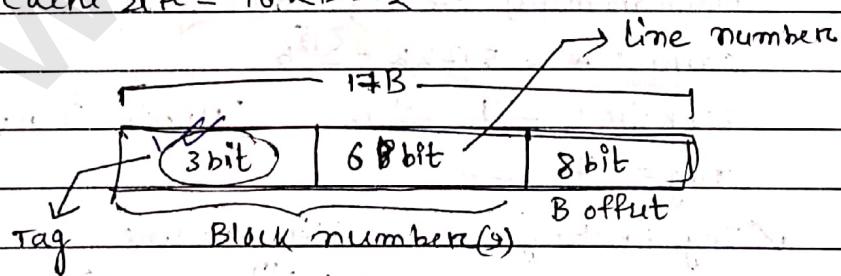
$$= 2^8 B$$

$$\text{no. of blocks} = \frac{2^{17}}{2^8} = 2^9 - \text{bit}$$

$$\text{Cache size} = 16 KB = 2^{14} B$$

$$2^{14}$$

$$2^8$$



$$\text{no. of lines} = \frac{2^{14}}{2^8} = 2^6 - \text{bit}$$

Line = Block size

$$\begin{aligned} \text{Tag Directory size} &= (\text{Tag size} \times \text{no. of lines}) \\ &= (3 \times 2^6) \text{ bit} \end{aligned}$$

Q2:

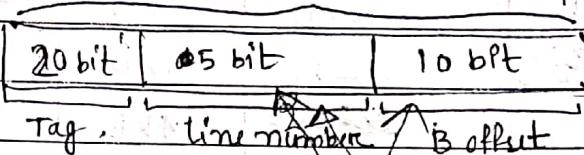
$$\text{MM size} = 32 \text{ GB}$$

$$= 2^{30} \times 2^{10} \times 32 \text{ MB}$$

$$= 2^{10} \times 2^{10} \times 32 \text{ KB}$$

$$= 2^{10} \times 2^{10} \times 2^{10} \times 32 \text{ B}$$

$$= 2^{30+5} \text{ B} = 2^{35} \text{ B}$$

 $\rightarrow 35 \text{ bit}$ 

$$\text{Block size} = 1 \text{ KB}$$

$$= 2^{10} \text{ B}$$

(10 bit for Block B offset)

$$\text{Cache size} = 32 \text{ KB}$$

$$= 2^{10} \times 2^5 \text{ B}$$

$$= 2^{15} \text{ B}$$

$$\text{no. of line} = \frac{\text{C size}}{\text{line size}}$$

$$= \frac{2^{15}}{2^{10}}$$

$$= 2^5$$

$$\text{Tag directory size} = 10 \times 2^{10} \text{ Tag} \times \text{no. of lines.}$$

$$= (10 \times 2^5)$$

$$2^6 \text{ B}$$

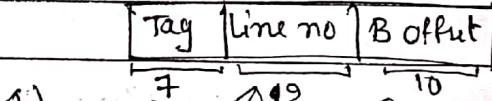
Q3:

$$\text{Main memory size} = ? \quad (64)$$

$$\text{Cache size} = 512 \text{ KB} = 2^{19} \text{ B}$$

$$\text{Block size} = 1 \text{ KB} = 2^{10} \text{ B}$$

$$\text{Tag} = 7$$



$$\checkmark \text{ Tag directory size} = 7 \times 2^{19} \text{ no. of line}$$

$$= 7 \times \frac{2^{19}}{2^{10}} = (7 \times 2^9) \text{ B bit}$$

$$\text{no. of line, } \frac{2^{19}}{2^{10}} = 2^9$$

~~no. of block & block size~~

$$\text{Address size} = 7 + 9 + 10 = 26$$

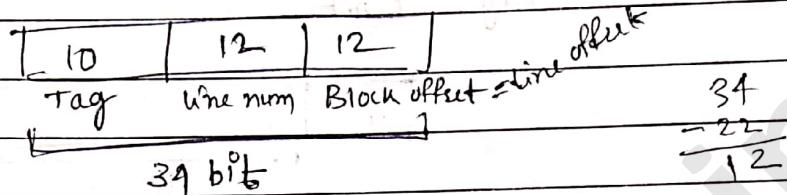
$$\text{Main memory size} = 2^{26} \text{ B}$$

$$= 2^{32} \text{ B} = 2^6 \text{ MB} = 64 \text{ MB}$$

$$= 2^{30} \times 2^2 \text{ B} = 4 \text{ GB}$$

~~Q24~~

$$\text{MMU size} = 16 \text{ GB} = 2^{30} \times 2^4 \text{ B} = 2^{34} \text{ B}$$



Cache size = ?

$$\text{Block size} = 4 \text{ KB} = 2^{10} \times 2^4 \text{ B} = 2^{12} \text{ B}$$

Tag = 10

Tag = Directory.

$$2^{12} \times 2^{12} = 2^{24} \text{ B}$$

cache size =  ~~$2^{24}$~~   $\approx 2^{24}$  B ~~blocks~~  $\approx 16 \text{ MB}$ 

$$\text{no of lines} = \frac{\text{c size}}{\text{line size}} = \frac{2^{24}}{2^{12}} = 2^{12}$$

$$\text{Tag directory} = \frac{10}{\cancel{10}} \times 2^{12}$$

$$= 10 \text{ bits} \cdot (10 \times 2^{12}) \text{ bit.}$$

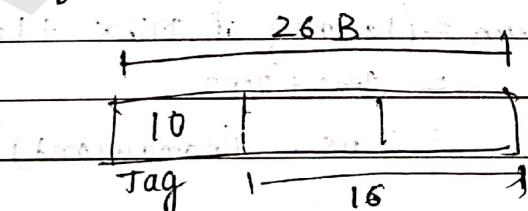
~~Q25~~

$$\text{MMU size} = 64 \text{ MB} = 2^{20} \times 2^6 \text{ B} = 2^{26} \text{ B (PA)}$$

cache size = ?

BLOCK size = ?

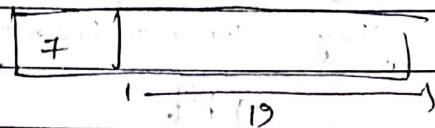
tag = 10

cache size =  $2^{16} \text{ B}$ Cache size =  $2^{16} \text{ B}$ Cache size =  $2^{16} \text{ B}$ Cache size =  $2^{16} \text{ B}$

~~B6:~~

$$\text{Cache size} = 512 \text{ KB} = 2^{19} \text{ B}$$

Tag = 7 bit



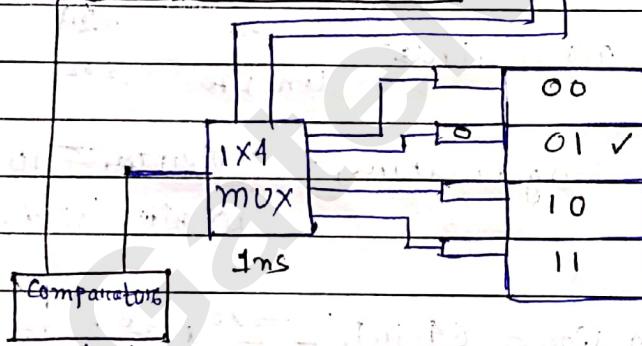
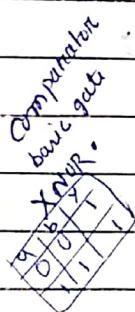
$$19 + 7 = 26$$

$$\checkmark \text{ MM size} = 2^{26} \text{ B}$$

### • Direct Mapping HW Implementation =

CPU generated address  $\rightarrow$ 

Tag	LN	BD
0	01	



1 - hit

0 - miss

$$\begin{aligned} \text{Total time taken} &= (\text{latency of MUX} + \text{latency of comparator}) \\ &= 1\text{ms} + 1\text{ms} \\ &= 2\text{ms. (nanoseconds)} \end{aligned}$$

$\rightarrow$  no of MUX and comparators depends on ~~no. of~~ no. of Tag bits.

Direct  
mapping  $\rightarrow$

$$\text{Tag size} = K \text{ bit}$$

$$\text{no. of MUX required} = K$$

$$\text{no. of Comparators} = 1 * (K \text{ bit compare})$$

→ always (in Direct map)

Q1:

MM size = 1 GB

(hence latency  
Mux negligible)

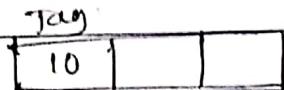
Cache size = 1 MB

Comparator size = 10 Kms.

1-bit latency = ?



Hit latency



MM size = 1 GB

\*\*\* Tag = Main M size  
Cache size

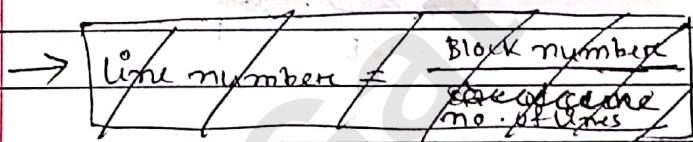
$$= \frac{1 \text{ MB}}{1 \text{ MB}} \Rightarrow 2^{10} \text{ KB } 1 \text{ KB} \Rightarrow 2^{10} \text{ B}$$

K = 10

Hit latency = ~~10 \* K~~ 10 Kms

$$= 10 \times 10 \text{ ns}$$

$$= 100 \text{ ns}$$



line number = (B number) %  
(no. of lines)

K = m % n

• Disadvantage of direct mapping =

→ Conflict miss problem.

ex:-

Block req by CPU -

0	4	8, 12, 5, 6, 9, 1, 15, 20
1	5	9
2	6	
3	15	→ page was empty for long time, but Line no. 0 used heavily, this no. of lines = 4 is conflict miss problem.

$$L-N = (B.N) \% (N.L)$$

Ex:

Block request by CPU

0	4	8	12	16	20
1		2			
2			3		
3					

cache.

4 8 12 16 20 address

$$K = m/n$$

~~n = num of lines in cache~~

$$4 \leq 16 \quad 4 \% 4 = 0$$

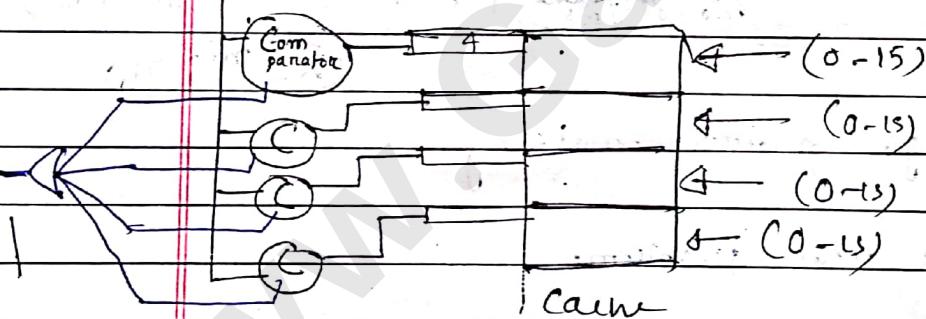
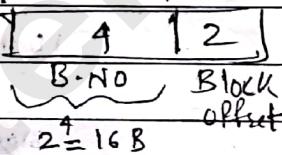
$$8 \leq 16 \quad 8 \% 4 = 0$$

Line no. 1, 2, 3, empty but they line 0 are heavily used  
it is conflict miss problem.

### Associative mapping = (intro)

can solve

→ By Associative mapping A Conflict miss problem.



1 - Cache hit

0 - Cache miss

→ 4 lines - 4 comparators need, so hardware cost increase along with freedom.

→ No of Comparators required = No of cache lines,

→ We get freedom to place Block anywhere in cache, but requirement of Comparators increase.

(Q1)

Main memory size = 32 GB

Block size = 32 KB

✓ Tag(K) = ?

Propagation delay (PD) of comp = 10 ns

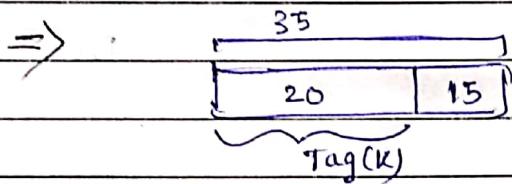
PD of OR Gate = 10 ns.

✓ Hit latency = ?

$$MM = 32 \text{ GB}$$

$$= 2^{30+5} \text{ B}$$

$$= 2^{35} \text{ B}$$



$$BS = 32 \text{ KB}$$

$$= 2^{15} \text{ KB}$$

✓ hit latency = PD of comparator + PD of OR gate  
 $= 10 \text{ ns} + 10 \text{ ns}$   
 $= (10 + 10) \text{ ns} + 10 \text{ ns}$   
 $= 30 \text{ ns.}$

(Q2)

	MM size	cache size	Block size	Tag size	Tag directory size
① -	128 KB	16 KB	256 B	9 bits	$(2^9 \times 2^6)$ bits
② -	32 GB	32 KB	1 KB	25 bits	$(2^5 \times 32)$ bits
③ -	128 MB	512 KB	1 KB	17	$(17 \times 2^9)$ bits
④ -	16 GB	?	4 KB	22 bits	(can't guess)
⑤ -	64 MB	?	64 KB	10	(can't guess)
⑥ -	not possible	512 KB	not possible	?	not possible.

$$\textcircled{1} \quad \text{MM size} = 128 \text{ KB}$$

$$\text{CS} = \text{Cache size} = 16 \text{ KB}$$

$$\text{no of lines} = \frac{\text{CS}}{\text{BS}}$$

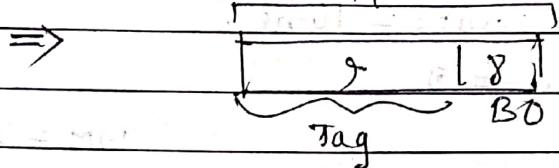
$$\text{BLOCK size (BS)} = 256 \text{ B}$$

$$= \frac{2^{14}}{2^8}$$

$$\text{Tag size} = ?$$

$$= 2^6$$

$$\text{Tag directory size} = ?$$



$$\text{BB} = 128$$

$$= 2^7 \times 2^{10} \text{ B}$$

$$= 2^{17} \text{ B}$$

$$\text{BS} = 256 \text{ B} = 2^8 \text{ B}$$

$$\checkmark \text{ Tag} = 9$$

$$\checkmark \text{ Tag directory size} = \text{Tag size} * \text{no of lines}$$

$$= (9 * 2^6) \text{ bits}$$

$$\text{no of com neg} = 2^6 = 64$$

\textcircled{2}

$$\text{MM} = 2^{35} \text{ B}$$

$$\text{CS} = 2^{15} \text{ B}$$

$$\text{BS} = 2^{10} \text{ B}$$

$$\text{no. of Line} = \frac{2^{15}}{2^{10}}$$

$$= 2^5$$

$$\checkmark \text{ tag directory} = \text{tag} * \text{no of lines}$$

$$= 25 * 2^5$$

\textcircled{3}

$$\text{MM} = ?$$

$$\text{CS} = 2^{19} \text{ B}$$

$$\text{BS} = 2^{10} \text{ B}$$

$$\text{TS} = 17$$

$$\text{tag directory} = ?$$

$$\checkmark \text{ MM size} = 2^{10+4}$$

$$= 2^{27} \text{ B} = 8 \text{ MB}$$

$$= 128 \text{ MB}$$

$$\text{no. of lines} = \frac{2^{19}}{2^{10}} = 2^9$$

$$\leftarrow \text{Tag D.S} = (17 * 2^9) \text{ bits.}$$

$$(4) MM = 2^{31} B$$

$$CS \Rightarrow CS = ? X$$

$$BS = 2^{12} B$$

$$\sqrt{TS} = ?$$

$$T.DS = ? X$$

$$\begin{aligned} \text{Tag size} &= (34 - 12) \\ &= (22) \end{aligned}$$

$$(5) MM = 2^{26} B$$

$$CS = ?$$

$$BS = ? (2^{16})$$

$$\text{Tag} = 10$$

$$\text{Tag D-S} = ?$$

$$BS = 2^{26-10}$$

$$= 2^{16} = 64 KB$$

$$(6) MM = ? X$$

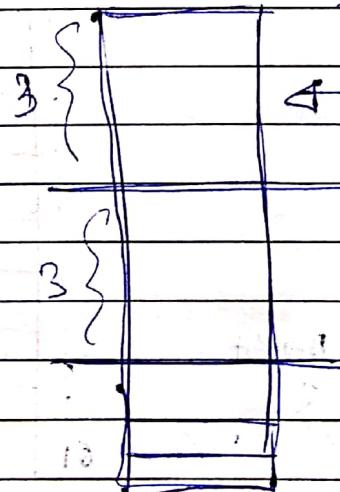
$$CS = 2^{19} B$$

$$TS = ?$$

$$BS = ? X$$

$$T.DS = ? X \quad \text{not possible.}$$

3-way set



cache

memory

- set Associative mapping = (adv no. of comparisons reduced)

## Ex<sup>o</sup>(how set associative work)

MM size = 64 B

$$\text{cache s} = 32 \text{ B}$$

Block S = 4 B

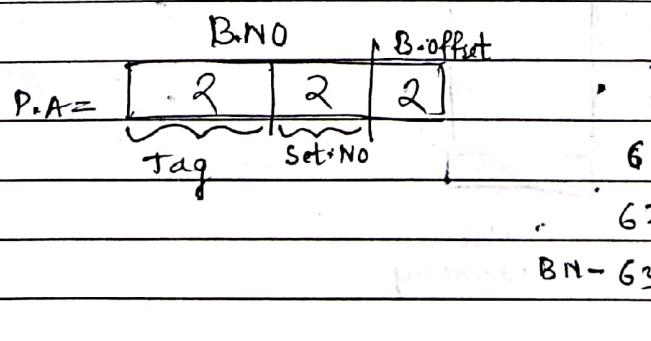
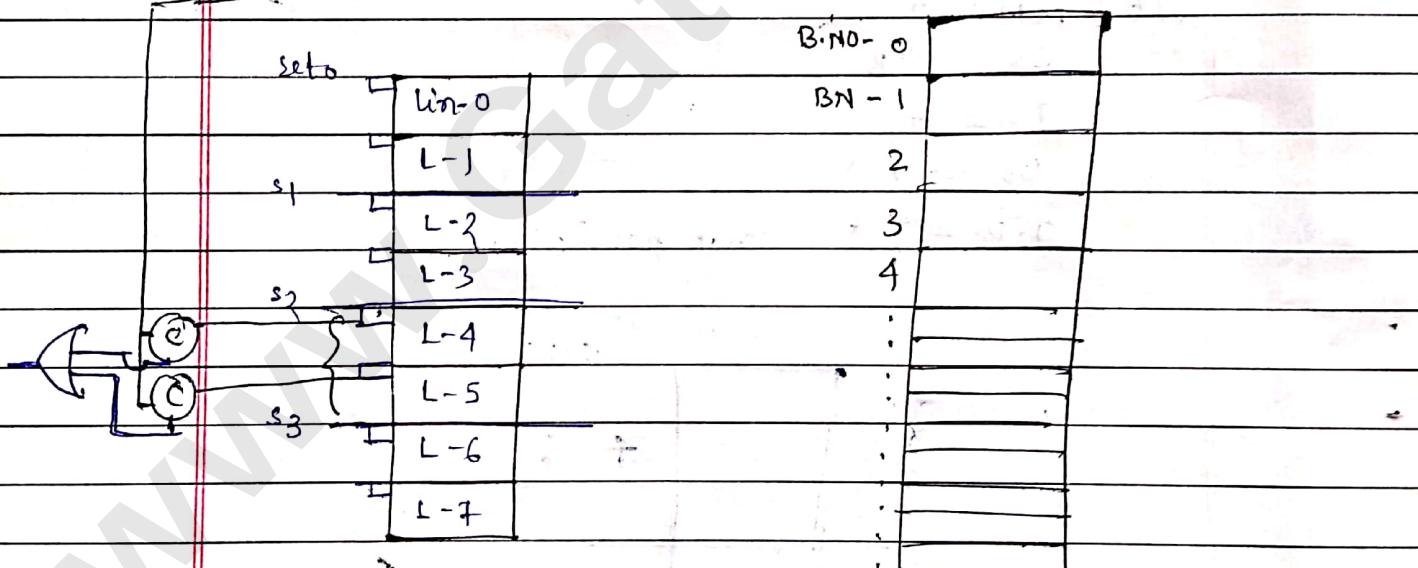
~~✓ set size = 2 Blocks (lines)~~

✓ (2-way set associative)

$$\text{lines} = \frac{CS}{BS} = \frac{32}{4} = 8 \text{ lines}$$

$$\text{sets} = \frac{\text{lines}}{\text{sets}} = \frac{8}{2} = 4 \text{ sets}$$

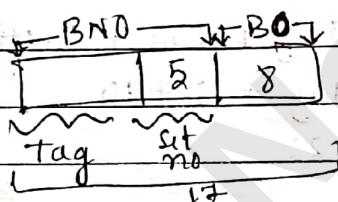
$$\text{give address (P-A)} = \boxed{01} \boxed{10} \boxed{11} \\ \text{SNO}$$



(Q1)

	MM size	cache size	Block size	Tag bits	Tag directory size	set associative
①	128 KB	16 KB	256 KB	4	(4 * 2 <sup>6</sup> ) bits	2-way set
②	32 GB	32 KB	1 KB	22	(22 * 2 <sup>5</sup> ) bits	4
③	2 <sup>3</sup> MB	512 KB	1 KB	7	(7 * 2 <sup>9</sup> ) bits	8
④	16 GB	64 MB	4 KB	10	(10 * 2 <sup>14</sup> ) bit	4
⑤	64 MB	256 KB	X	10	X	4
⑥	88 MB	512 KB	X	7	X	8

(Q1)



$$MM\ size = 2^{17} B$$

$$CS = 2^{19} B$$

$$BS = 2^{18} B$$

$$\text{no. of lines} = \frac{CS}{BS} = \frac{2^{19}}{2^{18}} = 2^1 \text{ lines.}$$

$$\text{set no. of sets} = \frac{\text{no. of lines}}{\text{set size}}$$

$$= \frac{2^1}{2} = 2^0 = 2^5 = (32 \text{ sets})$$

$$PA = \text{Tag} + \text{set no} + \text{B.offset}$$

$$\Rightarrow 17 = \text{Tag} + 5 + 8$$

$$\Rightarrow \text{Tag} = 9$$

$$\begin{aligned} \text{Tag directory} &= \text{Tagsize} * \text{no. of lines} \\ &= (4 * 2^6) \text{ bits.} \end{aligned}$$

(2)

$$MM = 32 \text{ GB} = 2^{35} \text{ B}$$

$$CS = 32 \text{ KB} = 2^{15} \text{ B}$$

$$BS = 1 \text{ KB} = 2^{10} \text{ B}$$

$$T = ? \quad (22)$$

$$T.D = ?$$

Set-arr = 4-way

$$\text{no. of set} = \frac{\text{no. of line}}{\text{set size}} = \frac{2^5}{2^2} = 2^3$$

$$\rightarrow P.A = Tag + S.no + B.off$$

$$\rightarrow 35 = Tag + 3 + 10$$

$$\checkmark \rightarrow Tag = 22$$

$$\checkmark T.D = (22 * 2^5)$$

(3)

$$MM = ?$$

$$CS = 512 \text{ KB} = 2^{19} \text{ B}$$

$$BS = 1 \text{ KB} = 2^{10} \text{ B}$$

$$T = 7$$

$$T.D = ?$$

Set-arr = 8-way

~~see 3~~

Tag	6	10
-----	---	----

= 20 bits & 23 bits

& (MRW) =

$$\text{no. of line} = \frac{2^{19}}{2^{10}} = 2^9$$

$$\checkmark MM \cdot \text{size} = 2^{29} = 2^3 \text{ MB}$$

$$\text{no. of sets} = \frac{\text{no. of line}}{\text{size of sets}}$$

$$\checkmark T.D = T * \text{no. of lines}$$

$$= \frac{2^7}{2^3} = 2^6$$

$$= (7 * 2^9) \text{ bits}$$

④  $MM = 16 \text{ GB} = 2^{34} \text{ B}$

$CS = ?$

$BS = 4 \text{ KB} = 2^{12} \text{ B}$

$T = 10$

$T.D = ?$

set. am = 4-Way set.

↓ 34 ↓

10	12	12
----	----	----

Tag set no B.offset

no. of sets =  $2^{12}$

no. of sets =  $\frac{\text{no. of lines}}{\text{set size}}$

no. of lines =  $\text{set size} \times \text{no. of sets}$

$T.D = ? \quad T * \text{no. of lines}$

$= 10 * 2^{12}$

$= 160 \text{ KB}$

$CS = \text{no. of lines} * \text{line size}$

$= 2^{12} * 2^{12}$

$= 2^{26} \text{ B}$

$= 2^{26} \text{ MB}$

$= 2^{26} (64 \text{ MB})$

$= 4 * 2^{12}$

$= 2^{12}$

⑤  $MM = 64 \text{ MB} = 2^{26} \text{ B}$

$CS = ?$  (2<sup>18</sup> B)

$BS = ?$

$Tag = 10$

$Tag.D = ?$

set. am = 4 way

↓ 26 ↓

10	?	?
----	---	---

Tag set no B.offset

↓ 26 ↓

10	X	Y
----	---	---

16

⑥

MM = ?

CS = 512 KB

BS = ?

Tag = ?

Tag.D = ?

set. am = 8 way

~~Cache~~

Tag

Cache size = no. sets \* cache block size

\* lines per set

\* line size (BS)

$= 2^x * 2^y * 2^z$

$= 4 * 2^{x+y}$

$= 4 * 2^{16}$

$= 2^{18} = 2^8 \text{ KB} = 256 \text{ KB}$

(Q2)

MM size	Cache size	Block size	Tag size	Tag directory	set associative
64MB	-	-	10	-	4-way
-	512KB	-	7	-	8-way

$$\textcircled{6} \quad MM = ? - (2^{23} \text{ B})$$

$$CS = 512\text{ KB} = 2^{19} \text{ B}$$

$$BS = -$$

$$TS = 7$$

$$TDS = -$$

set ass = 8-way

Tag set Block

7 2 3 y

cache size = no. of sets \* line's per set

\* BLOCK size

$$2^{19} = 2^x * 2^3 * 2^y$$

$$\Rightarrow 2^{16} = 2^{n+y}$$

$$\Rightarrow n+y = 16$$

$$P.A = 7 + (n+y)$$

$$= 7 + 16 = \textcircled{23}$$

$$MM = 2^{23} = 8 \text{ MB}$$

- Comparing all the mappings

$$\text{MM size} = 4\text{GB} = 2^{32} \text{B}$$

$$\text{Cache} = 4\text{MB} = 2^{22} \text{B}$$

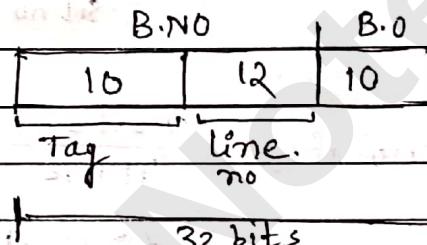
$$\text{BLOCK size} = 1\text{KB} = 2^{10} \text{B}$$

$$P.A = 32 \text{ Bits}$$

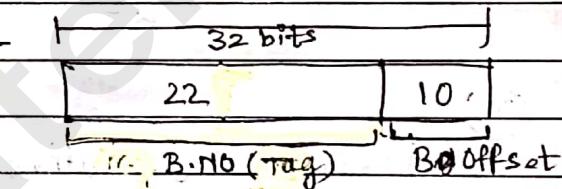
$$\text{no. of lines} = \frac{CS}{BS} = \frac{2^{22}}{2^{10}} = 2^{12} \text{ lines}$$

$$\text{no. of lines} = \frac{\text{no. of bits}}{\text{size of UF}} = \frac{2^{12}}{2^3} = 2^{10}$$

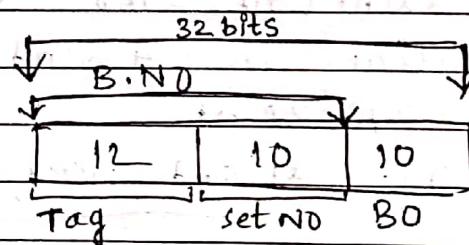
Direct mapping :-



Associative Mapping :-



4-Way Set Assoc :-



Relationship between

	Tag	Comparator registers	size of comparators (Tag directory)	Lines	size of comparator
Direct	10	1	$10 \times 2^{12}$	$2^{12}$	10
Associative	22	$2^{12}$	$22 \times 2^{12}$	$2^{12}$	22
4-Way	12	4	$12 \times 2^{12}$	$2^{12}$	12

✓ Questions - gate

Q-1995

(Q1)

Cache size = 1 K Words =  $2^{12}$  words

Block size = 64 words. =  $2^6$  words

Set size = 4 blocks.

The number of bits in "set" and "word" field of MM address are -

 $\rightarrow$ 

	1	4	6
--	---	---	---

Set no      B.O or (word)

$$\text{no. of sets} = \frac{\text{no. of lines}}{\text{Set size}}$$

$$\text{no. of lines} = \frac{2^{12}}{2^6} = 2^6$$

$$= \frac{2^5}{2^2} = 2^4 \text{ (sets)}$$

(Q2)

4-Way set associative.

cache lines = 128

lines size = 64 words

P.A. = 20 bits

Tag, set and word fields are ?

 $\rightarrow$ 

9	5	6
---	---	---

Tag      set no      B.O or (word)

20

$$\text{no. of lines} = \frac{\text{Cache size}}{\text{lines size}}$$

Cache size =  $2^6 * 2^7 = 2^{13}$  words.

$$\text{sets} = \frac{\text{lines}}{\text{Set size}} \Rightarrow \frac{2^7}{2^2} \Rightarrow 2^5$$

Tag = 9, set = 5, words (B.O) = 6.

(Q3) Blocks in cache = 128

4-way set associative

MM contains  $2^{14}$  blocks.

Block size is 256 eight bitwords.

(i) How many bits are required for addressing MM? (2)

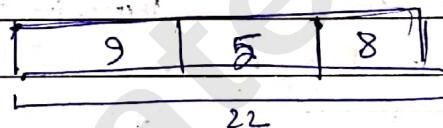
(ii) How many bits are needed to represent the tag, set and word fields. (9) (3) (8)

$$\rightarrow \text{cachelines} = 2^7$$

$$\text{set size} = 4$$

$$\text{MM Blocks} = 2^{14}$$

$$\text{Block size} = 2^8 \text{ bitword.}$$



$$\text{sets} = \frac{2^7}{2^2} = 2^5$$

no. of lines

$$\text{MM size} = 2^{14} \times 2^8 \text{ words} \\ = 2^{22} \text{ words}$$

(Q4)

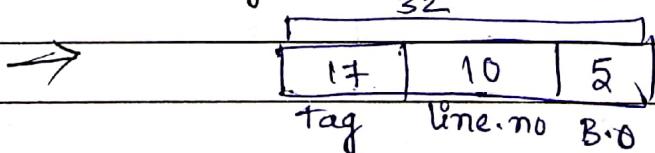
Direct mapped cache

$$\text{cache size} = 32 \text{ KB} = 2^{15} \text{ B}$$

$$\text{Block size} = 32 \text{ B} = 2^5 \text{ B}$$

$$\text{PA} = 32 \text{ bits}$$

The number of bits needed for cache indexing and tag bits are respectively.



$$\text{no. of lines} = 2^{15} = 2^{10}$$

$$2^5$$

$$\text{Cache indexing} = 10 \text{ bit}$$

$$\text{Tag bit} = 17 \text{ bits } \text{Ans. } (10, 17)$$

9-2006 05

1-20-1-21

Considerate three certain categories

① first one

② 2<sup>nd</sup> one

## sets associative

G. Reuter

$$\text{Carby size} = 32 \text{ } \mu\text{B} = 2^{15} \text{ B}$$

Cache life same

## 2-Way set association

## Direkt mappe

$$\text{Block size} = 32B = 2^5B$$

$$P.A = 32 \text{ bits}$$

2-to-1 MUX latency = 0.6 ns.

K bit comparators has latency =  $K/10 \text{ ns}$ .

The hit latency of set associative organization is  $\lceil \log_2 n \rceil$  and " " " Direct mapped is  $n$ .

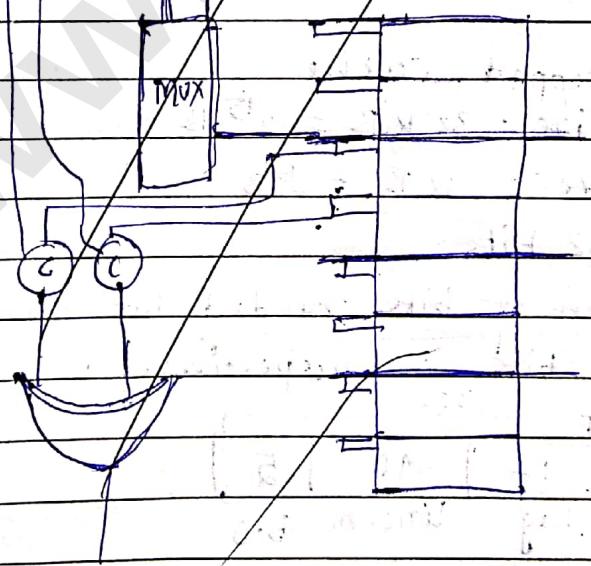
find  $n_1$  and  $n_2$ ?

$\rightarrow PA =$   32 bits

$$\text{line } m_0 \neq \frac{2^{15}}{2^5} = 2^{10}$$

$$\text{no. of set} = \frac{2^{10}}{2^1} = 2^9$$

$$\text{No of computers required} = 2^t$$

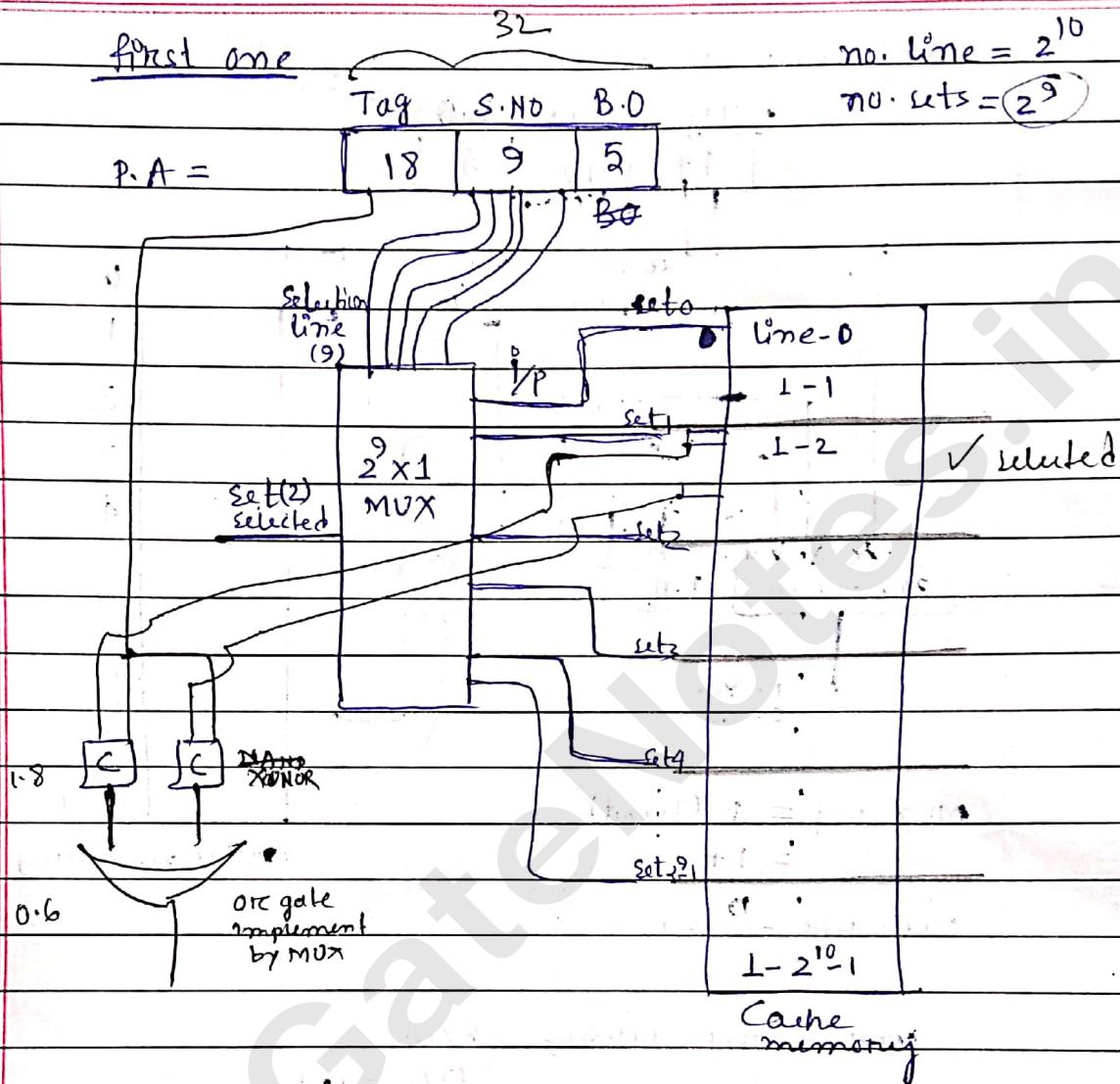


~~here, MUX required~~

$\approx 2 \times$  Tag bits

$$= K \times \text{Tag bits}$$

hit latency =



$$\text{MUX req} = \text{Tag bit} \times 2$$

K-way set associative  $\Rightarrow (\text{MUX req}) = T \times K$

size of each MUX =  $2^S \text{ to } 1$

comparator need = K

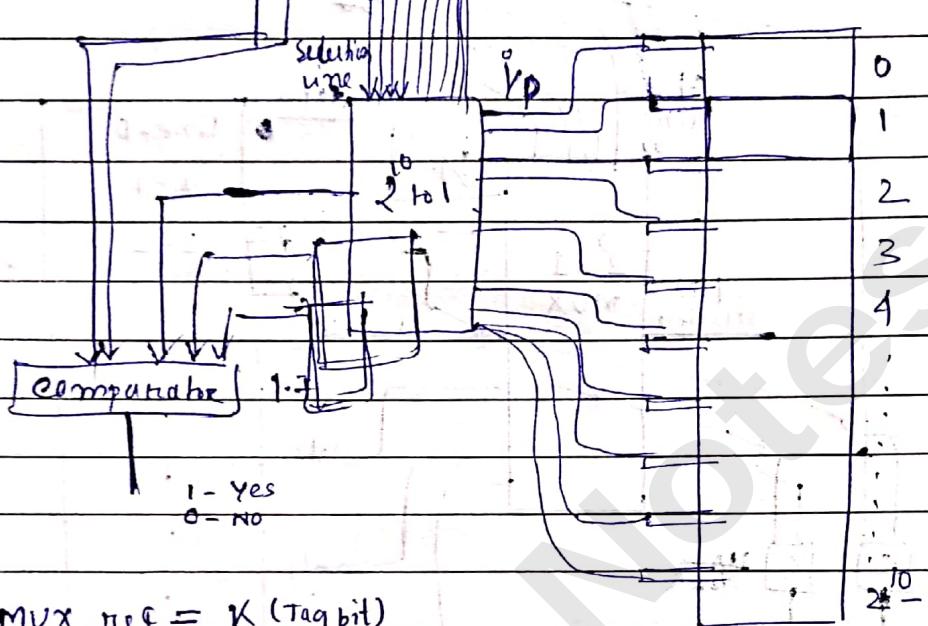
$$\text{comparator latency} = K/10 = 18/10 = 1.8 \text{ ns.}$$

$$t_1 = 1.8 + 0.6 \\ = 2.4 \text{ ns}$$

~~Second one -~~

Tag L.H.O B.O - 32 bit

17	10	5
----	----	---



$$\text{MUX ref} = K (\text{Tag bit})$$

$$= 17$$

cache memory

$$\text{each size} = 2^{10} \text{ to } 1$$

$$\text{computation latency} = K/10$$

$$= 17/10 = 1.7$$

$$\text{Total } t_2 = 1.7 \text{ ms}$$

g-2011

(Q6) Direct mapping

cache size = 8KB

BS = 32 byte.

PA = 32 bits.

The cache controller maintains tag information for each cache block comprising of following.

1 valid bit, 1 modified bit and a ~~and many~~ bits as the minimum needed to identify block mapped in 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?

$$\rightarrow \text{Lines} = \frac{CS}{BS} = \frac{2^{13}}{2^3} = 2^8$$

19	8	5
$(19+1+1) = 21$		

$$\begin{aligned} T\text{-Directory} &= (21 * 2^8) \text{ bits} \\ &= 21 * 256 \\ &= 5376 \text{ bits.} \end{aligned}$$

P-2<sup>29</sup>, A-1  
10-2<sup>10</sup>, S-2<sup>10</sup> (Q7)  
Cache size = 256 KB =  $2^{18}$  B  
set size = 4

$$B.S = 32 B = 2^5 B$$

$$P.A = 32 \text{ bits}$$

16	11	5
Tag	Set	BO

$$\begin{aligned} \text{no. of lines} &= \frac{BS \times CS}{BS} \\ &= 2^{13} \end{aligned}$$

- ① The number of bits in the tag field of an address - (16).

$$\begin{aligned} \text{sets} &= \frac{2^8 \text{ lines}}{2^2} \\ &= \frac{2^{13}}{2^2} = 2^{11} \end{aligned}$$

- ② The size of the cache tag directory is - Lines \* tag

$$\begin{aligned} 2^{13} * (16 + 2 + 11) &= 2^{13} * 16 \\ &= 2^{13} * 20 \\ &= 2^3 * 20 \text{ KB} \\ &= 160 \text{ KB.} \end{aligned} \quad \begin{aligned} &= 2^{13} * 16 \\ &= 2^{13} * 4 \\ &= 2^{17} \\ &= 2^7 \text{ KB} \\ &= 128 \text{ KB} \end{aligned}$$

g-2019  
(88)

4-way set associativity

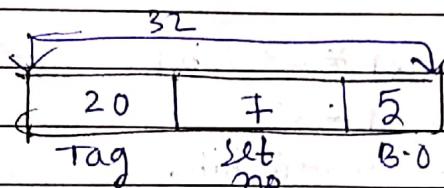
$$\text{Cache size} = 16 \text{ KB} = 2^{14} \text{ B}$$

$$\text{Block size} = 8 \text{ words} = (8 \times 32) \text{ bits} = \frac{2^8}{2^3} \cdot B_0 = 2^5 \text{ B}$$

$$\text{Word size} = 32 \text{ bits}$$

$$PAS = 4 \text{ sets} = 2^{32} \text{ B}$$

$$\text{tag bits} = ?$$



$$\text{Tag bits} = 20$$

$$\text{Lines} = \frac{CS}{BS}$$

$$\text{set-lines} = \frac{2^9}{2^2} = 2^7 = 128$$

$$= 2^9$$

$$= 2^7$$



Set  
no.  
0  
1  
2  
3  
4  
5  
6  
7

Block  
no.  
0  
1  
2  
3  
0  
1  
2  
3  
0  
1  
2  
3  
0  
1  
2  
3

Address bits 11 10 9 8 7 6 5 4 3 2 1 0

Set no. bits 3 2 1 0

Block no. bits 3 2 1 0

Tag bits 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

atlantis

Date \_\_\_\_\_

Page \_\_\_\_\_

27

atlantis

Date \_\_\_\_\_  
Page 28

## Computer Architecture:

It deals with instructions, addressing modes, ALU, pipelining etc (Internal design)

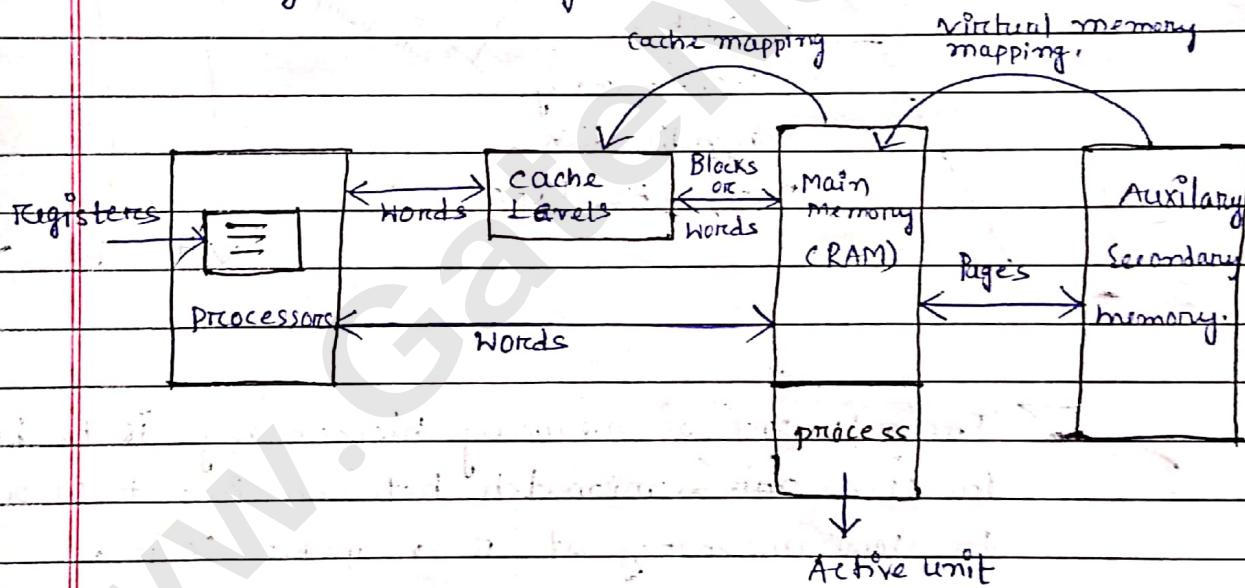
## Computer organization:

It deals with how various memory and I/O interact with a system.

## Computer design:

It deals with hardware design.

## Memory Interfacing -

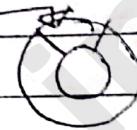


- Memory Hierarchy :

Cache levels }  
 Main memory } Random Access.

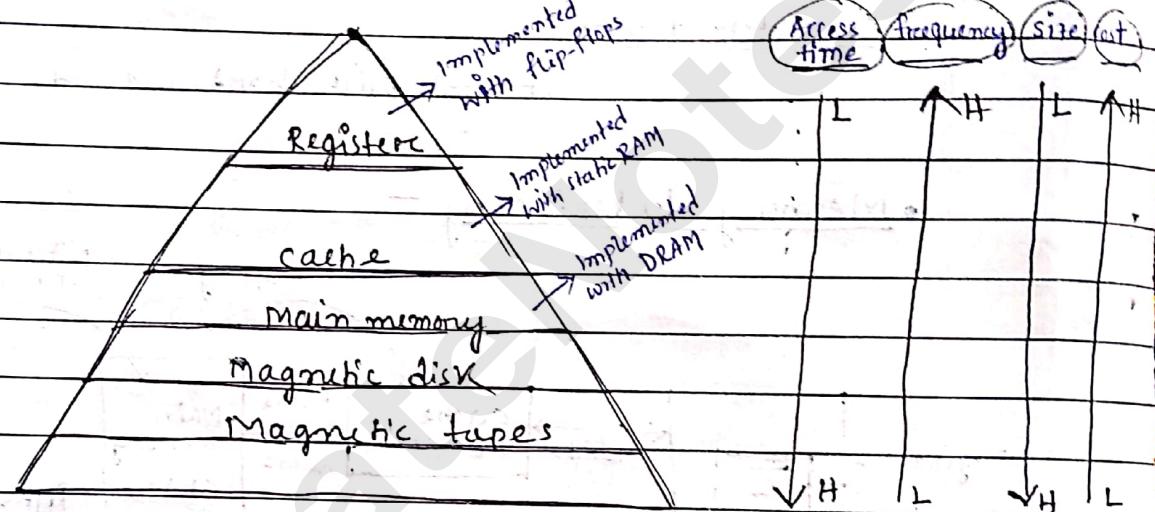


Magnetic disk → semi Random Access.



Magnetic tapes → sequential Access.

Access time      frequency      size (cost)



→ The purpose of memory hierarchy is to bridge the speed mismatch between fastest processor to slow memory at reasonable cost.

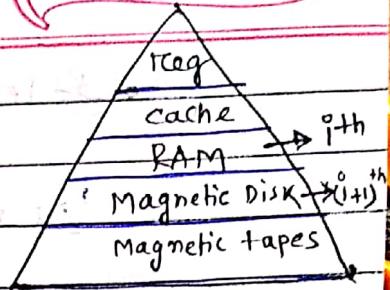
→ The goal of memory hierarchy is to minimize average access time of entire memory system.

- Level memory -

## 2 level memory -

Information in

$i^{\text{th}}$   $(i+1)^{\text{th}}$  level



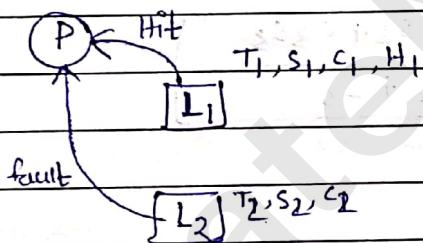
→ If processor refers to  $i^{\text{th}}$  level memory is found them "Hit" otherwise "Miss (or) fault".

→ There are two way in which the processor is connected to various levels of memory

Case 1°

$$T_1 < T_2$$

Cases



$$\text{Hit rate} = \frac{x}{100}$$

$$\text{Miss rate} = (1 - \frac{x}{100})$$

$T_1 \rightarrow$  Time to access

$S_1 \rightarrow$  size of level 1 memory.

$c_1 \rightarrow$  cost per bit.

$H_1 \rightarrow$  Hit rate.

$$T_{\text{avg}} = H_1 T_1 + (1 - H_1) T_2$$

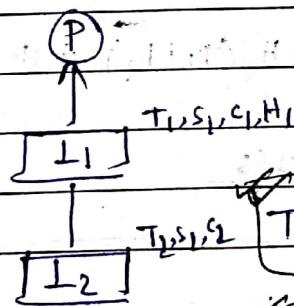
$$\text{Cost}_{\text{avg}} = \frac{c_1 s_1 + c_2 s_2}{s_1 + s_2}$$

$$100 \rightarrow x$$

$$T_{\text{avg}} = \frac{x T_1 + (100-x) T_2}{100}$$

$$= H_1 T_1 + (1 - H_1) T_2$$

Case 2°

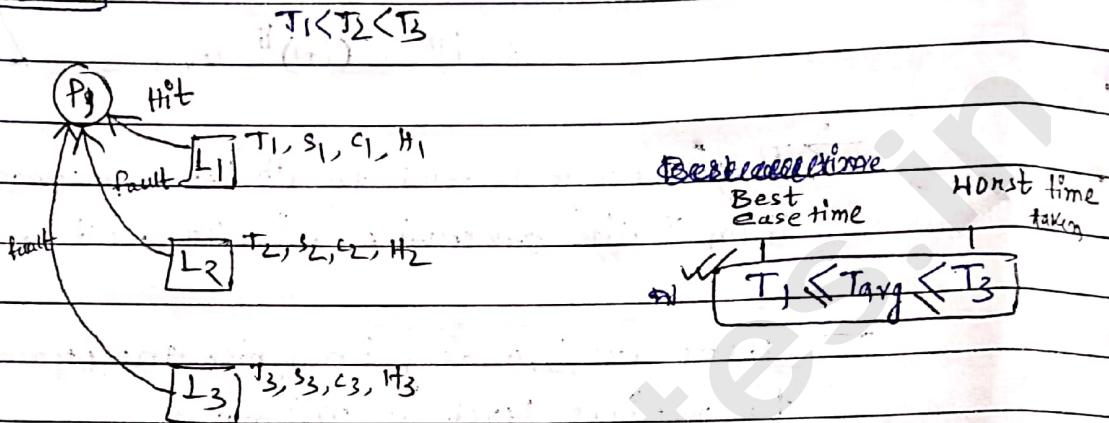


$$T_{\text{avg}} = H_1 T_1 + (1 - H_1)(T_1 + T_2)$$

~~$$\text{Cost}_{\text{avg}} = \frac{s_1 c_1 + s_2 c_2}{s_1 + s_2}$$~~

• 3-level memory -

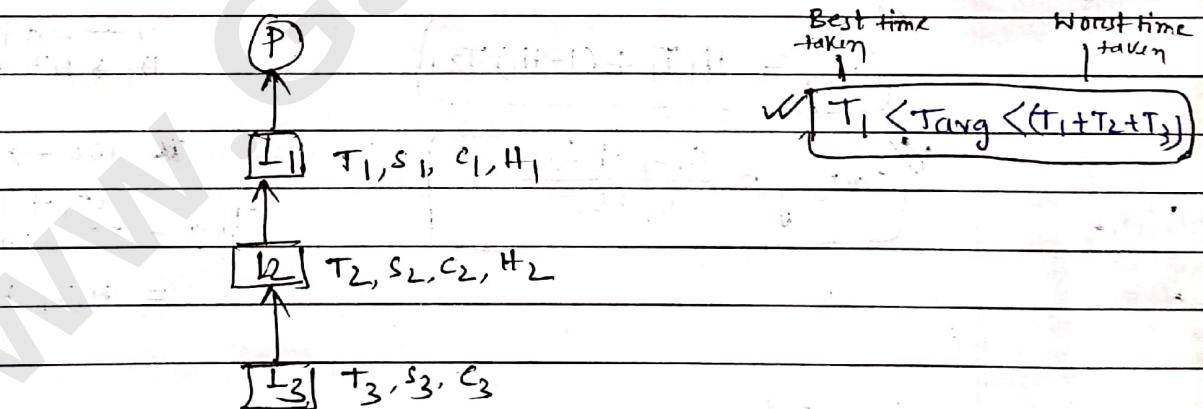
case -1



$$T_{avg} = H_1 T_1 + (1-H_1) H_2 T_2 + (1-H_1)(1-H_2) T_3$$

$$C_{avg} = \frac{C_1 S_1 + C_2 S_2 + C_3 S_3}{S_1 + S_2 + S_3}$$

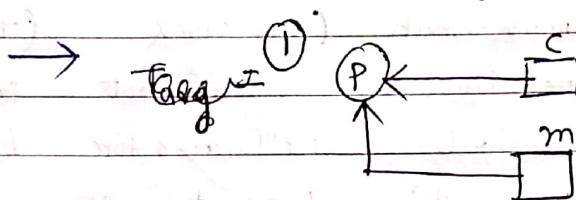
Case -2



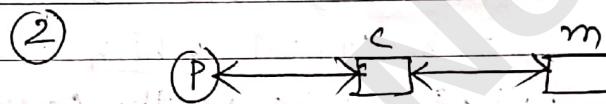
$$T_{avg} = H_1 T_1 + (1-H_1) H_2 (T_1 + T_2) + (1-H_1)(1-H_2)(T_1 + T_2 + T_3)$$

$$C_{avg} = \frac{S_1 C_1 + S_2 C_2 + S_3 C_3}{S_1 + S_2 + S_3}$$

Q: The average memory access time for a machine with a cache hit rate of 80% where the cache access time is 5ns and memory access time is 100ns is?



$$\begin{aligned}
 T_{avg} &= H_1 T_c + (1-H_1) T_m \\
 &= (0.8)(5\text{ns}) + (1-0.8)(100\text{ns}) \\
 &= [24 \text{ ns}]
 \end{aligned}$$



$$\begin{aligned}
 T_{avg} &= H_1 T_c + (1-H_1) (T_c + T_m) \\
 &= (0.8)(5\text{ns}) + (1-0.8) (5\text{ns} + 100\text{ns}) \\
 &= 4\text{ns} + (0.2) (105\text{ns}) \\
 &= 4\text{ns} + 21.0\text{ns} \\
 &= [25 \text{ ns}].
 \end{aligned}$$

- Cache replacement policy :-

Replacement policy is required for associative mapping and set associative mapping but not for direct mapping.

Replacement policies are aimed to minimize miss penalty for ~~referencess~~ future references.

To Get Full Content

Click of the below link given  
in this page



[www.GateNotes.in](http://www.GateNotes.in)

or

visit:[www.gatenotes.in](http://www.gatenotes.in)