I

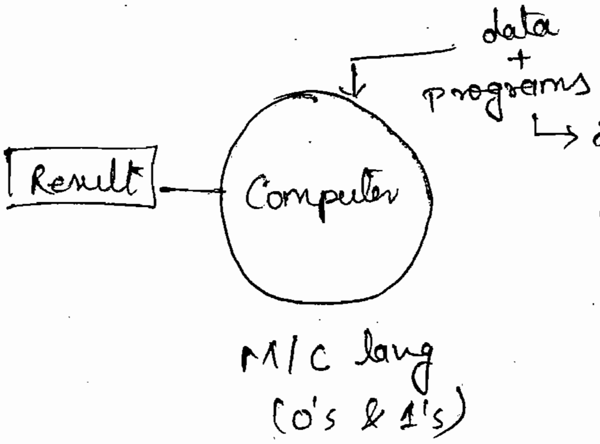
(8–11 Marks)

COMPUTER ORGANISATION

Shekar Ramesh DATE 19/11 PAGE

1. Memory Interfacing } Paul Chawdery
2. I/O " }
3. Machine Instructions – Hayes
4. Control Unit Design – Morris Mano
5. ALU Data Path – "
6. Addressing Modes – "
7. Number System – "
8. Data Representation – "
9. Pipelining – Hayes

Introduction:



↳ Set of Instructions
to
Command to perform a task

1 bit – 0 or 1 – 2

2 bit – 00, 10, 01, 11 – 4

⋮

k bits – – – – 2^k possibilties

Q.– A digital system has 986 possibilties. The $\min^m$. bits to represent – 2^{10}

10 bits.

Compute

ALU,

Comput

I/O

Compul

Memo

Register

Shekar's Victory
DATE PAGE

1 Bit - 0 or 1

8 bits - 1 byte

1024 bytes - 1 KB

1024 KB = 1 MB

1024 MB - 1 GB

Computer Architecture:

It deals with instructions, addressing modes, ALU, Pipelining etc (Internal Design).

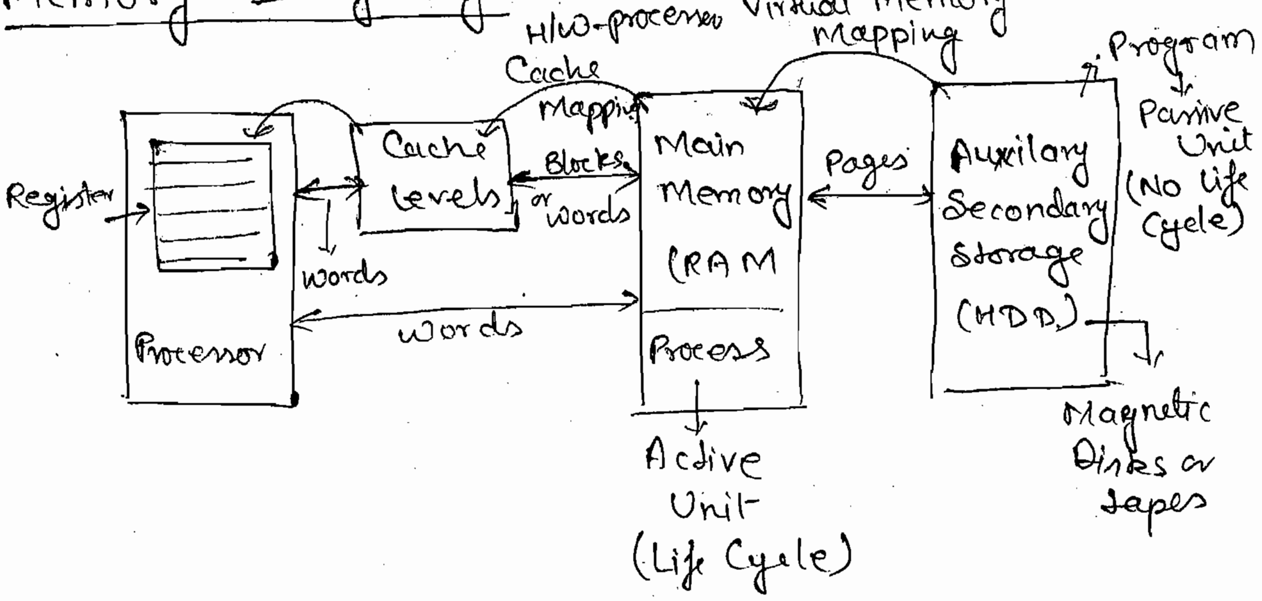
Computer Organisation:

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

Computer design:

It deals with hardware design.

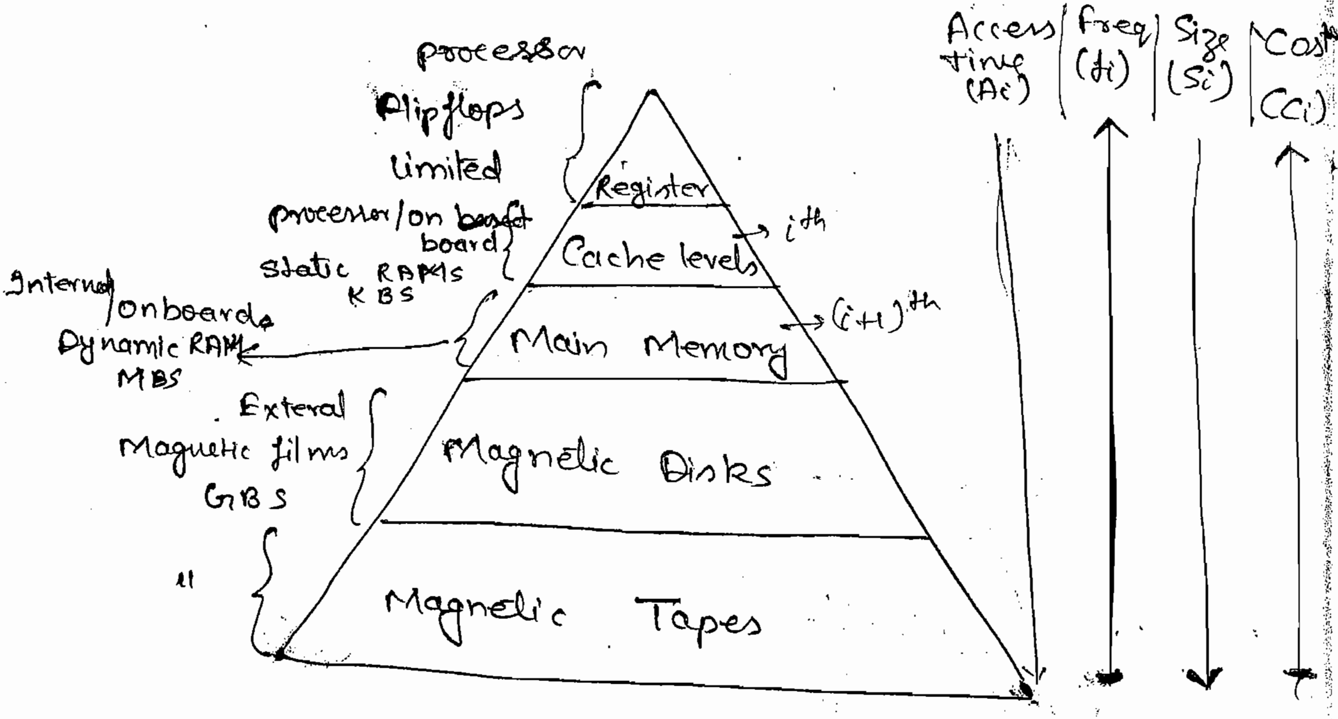
Memory Interfacing -



Gold

Memory Hierarchy:

Shekar's Victory
DATE PAGE



Cache levels
Main Memory } Random Access

Magnetic Disks → Semi Random Access

Magnetic Tapes → Sequential Access

- The purpose of memory hierarchy is to bridge the speed mismatch b/w ~~fat~~ fastest processer to slowest memory at reasonable cost.

- The goal of Memory hierarchy is to minimize average access time, of entire memory system.

- Since same info. presents at each level,
  $\boxed{i^{th} \subset (i+1)^{th}}$ memory level

- If
found
- In
Case
Case

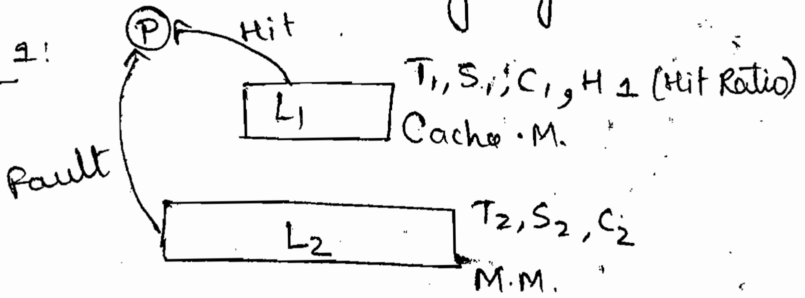
Shekar's Victory DATE PAGE

- If processor refers to i^{th} level memory, is found then 'Hit', otherwise 'Fault'.

eq) Size (S_i) | Cost/bit (C_i)

- In a 2-level memory system –

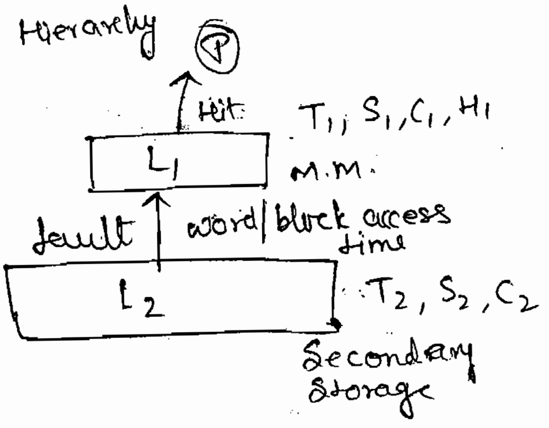
Case 1:



$$\boxed{T_{avg.} = H_1 \times T_1 + (1-H_1)\,T_2}$$

$$\boxed{C_{avg./bit} = \frac{C_1 S_1 + C_2 S_2}{S_1 + S_2}}$$

Case 2: Strict Hierarchy



idge

slowest

$$\boxed{T_{avg.} = H_1 \times T_1 + (1-H_1) \times (T_2 + T_1)}$$

nimize

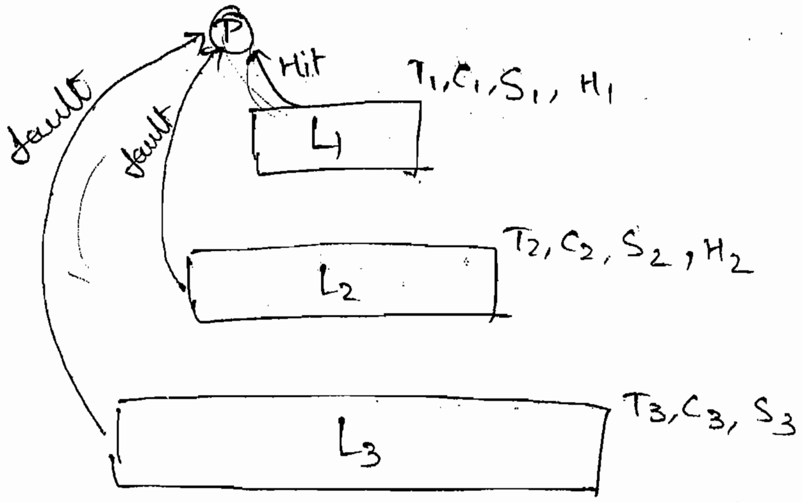
$$\boxed{C_{avg/bit} = \frac{C_1 S_1 + C_2 S_2}{S_1 + S_2}}$$

Gold

- In a 3-level Memory System

Shekar's Victory
DATE PAGE

Case 1: (default)



100 words

80% → 80 words → L_1 (Hit)

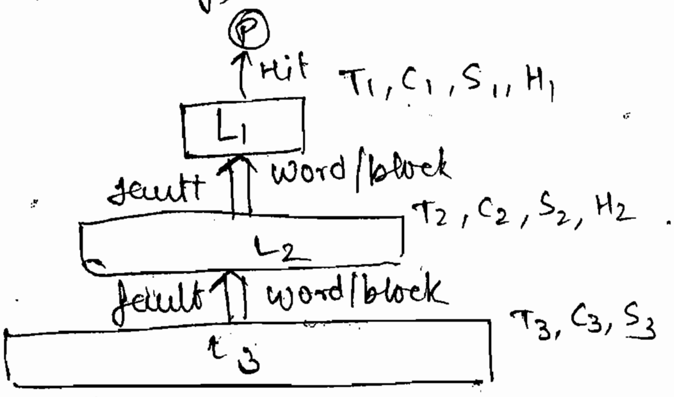
20% → 90% → L_2 ⇒ 18 words

H_2(Hit) L_3 ⇒ 2words

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

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

Case 2: (Strict Hierarchy)



Tavg

Q-

access

10 ns T_1

L_1

Q- A

access

is 3C

1)

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

$$C_{avg./bit} = \frac{C_1S_1 + C_2S_2 + C_3S_3}{S_1+S_2+S_3}$$

(Hit)

words

vords

$\boxed{\times T_3}$

Q- Consider a 2-level memory system, where the access time of level-1 & level-2 memories are $\underset{T_1}{10}$ ns & $\underset{T_2}{150}$ ns. what is the avg. access time, if the L1 Hit ratio is 90%.

$$T_{avg.} = (0.9)\times 10 + 0.1 \times 150$$
$$= 9 + 15$$
$$= 24 \text{ nS}$$

Q- A system is employing with 2-levels. The avg access time w/o level-1 is 150 ns & with level 1 is 30 ns.. The Leve 1 access time is 20 ns. what is

1) Hit Ratio

$T_2 = 150$ nS, $T_1 = 20$nS.

~~T_1+T_2~~ $T_{avg.} = 30$ nS

$$30 = H_1 \times 20 + (1-H_1)\times 150$$
$$30 = 150 - 130 H_1$$
$$H_1 = \frac{120}{130} = \frac{12}{13}$$
$$= 92.33$$

Gold

2) If Hit ratio is made to 100%, ~~what is the~~ DATE PAGE
acess time of L_1 & L_2 memories.

$T_1 = 20ns.$

$T_2 = 150 ns.$

Note:
Hit ratio doesn't influence the individual access time of L_1 & L_2. Therefore, $T_1 = 20ns$ & $T_2 = 150ns$, but the Tavg. value will change.

Q-3) If Tavg is increased by 10%, what is the % of change in Hit ratio?

$$H \propto \frac{1}{T_{avg.}}$$

$$T_{n\,avg.} = 30 + 10\% \text{ of } 30$$
$$= 33 nS$$

$$33 = H_1 \times 20 + (1 - H_1) \times 180$$

$$33 = 150 - 130 H_1$$

$$H_1 = \frac{117}{180}$$

$$= 90\% \Rightarrow 2.33\% \text{ decreased.}$$

Q- At 0.8 hit in level-1 memory, the avg. access time is increased by 20% from 60ns. ~~&~~ the L1 memory is 5 times faster than L_2. What is the % of change in Hit ratio?

A) 10% ↑ B) 20% ↑ [✓ C) 10% ↓] D) 20% ↓

Q-

time

1ns,

are (

with

Tc

Shekar's Victory
DATE PAGE

$T_{avg \cdot 1} = 60\,ns$

$T_{avg_2} = 60 + \cancel{60}\ 20\%\ \text{of}\ 60$

$= 60 + 12$

$= 72$

$60 = 0.8 \times T_1' + 0.2 \times 5T_1$

$60 = 1.8T_1$

$T_1 = \frac{60}{1.8} = \frac{\cancel{600}\ \cancel{200}\ 100}{\cancel{18}\ \cancel{6}\ 3}$

$= 33.33$

$72 = H_1 \times 33.33 + (1-H_1) \times 5 \times 33.33$

$72 = H_1 \times 33.33 + (1-H_1) \times 166.66\,ns$

$72 = 166.66 - 133.33\,H_1$

$H_1 = \frac{94.66}{133.33}$

$= 10\% \downarrow$

Q- Consider a system with 2-level cache, the access time of L1 cache, L2 cache and main memory are 1ns, 10ns & 500 ns. The Hit rate of L1 & L2 caches are 0.8 & 0.9. What is Tavg, 8 ignoring searched time within cache.

$T_{avg} = 0.8 \times 1 + 0.2 \times 0.9 \times 10 + 0.2 \times 0.1 \times 500$

$= 0.8 + 1.8 + 10$

$= 12.6$

Gold

Q- A system is employing with 3-levels. The access time of L_1, L_2 & L_3 memories is 100 ns/word, 150 ns/word & 500 ns/word. The L_2 & L_3 memories are divided into a block of 5 words. When a page fault occurs in L_1 or L_2, the processor must read from L_3 memory only. The H_1 & H_2 are 80% & 90%. What is Tavg.?

$T_1 = 100$ ns

$T_3 = 500 \times 5 = 2500$ ~~word~~ ns

$T_2 = 150 \times 5 = 750$ ns

$$T_{avg} = 0.8 \times 100 + 0.2 \times 0.9 (100+750) + 0.2 \times 0.1 (\overset{850}{100+750}+2500)$$

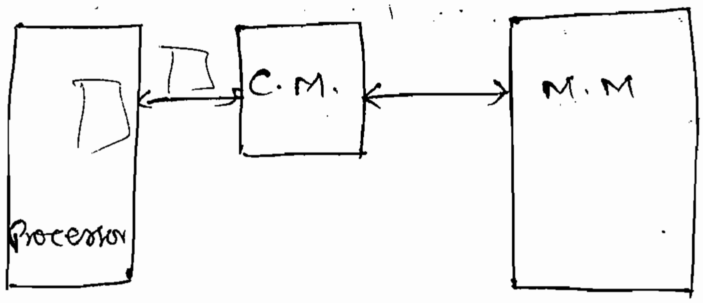
$$= 80 + 0.9 \times 170 + 0.02 \times 3350$$

$$= 80 + 153 + 67$$

$$= 233 + 67$$

$$= 300$$

Cache Memory:



- It is small and fastest memory.
- By placing most frequently used data and instructions, in a small cache, the average acces

time, can be minimized, thus improve the performance.

Shekar's Victory
DATE PAGE

• The performance of cache is known ~~with~~ hit ratio.

$$\text{Hit Ratio} = \frac{\text{No. of hits}}{\text{Total References to Memory}} \times 100$$

• When miss occurs, the processor directly obtains from M.M and a copy of it is brought into cache for future references.

$$\text{Hit Ratio} = \frac{\text{No. of hits}}{\text{No. of hits} + \text{No. of misses}} \times 100$$

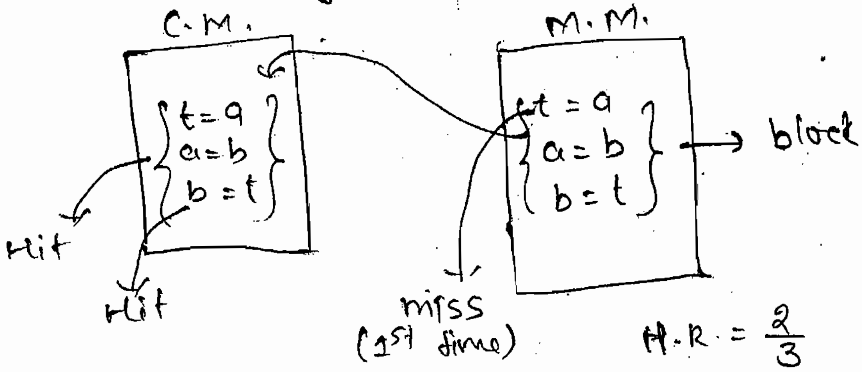
• Cache memory works on the principle called 'Locality of Reference'. i, i+1

LoR:
It states that the references to memory at any given interval of time - tends to be confined within a few localised area in memory,

It can be

1) Spatial LoR

It means that instructions in closed proximity to a recently executed instruction.



$H.R. = \frac{2}{3}$

- The increase in block size will increase Hit ratio or Performance.
- The first attempt to a word in a block is always a 'Miss' (Compulsory Miss).

Co

2) Temporal LoR:

It means that the recently accessed word (instructions or data) is likely to be needed very soon, and repeatedly.

for (i = 1; i <= 100; i++)

$i = 1 \rightarrow$ miss

$(i = 2 \rightarrow 100) \rightarrow$ hit

so $H.R. = \frac{99}{100}$

The misses due to capacity limitations are "Capacity Misses".

- The performance of Cache (Hit Ratio) depends on —

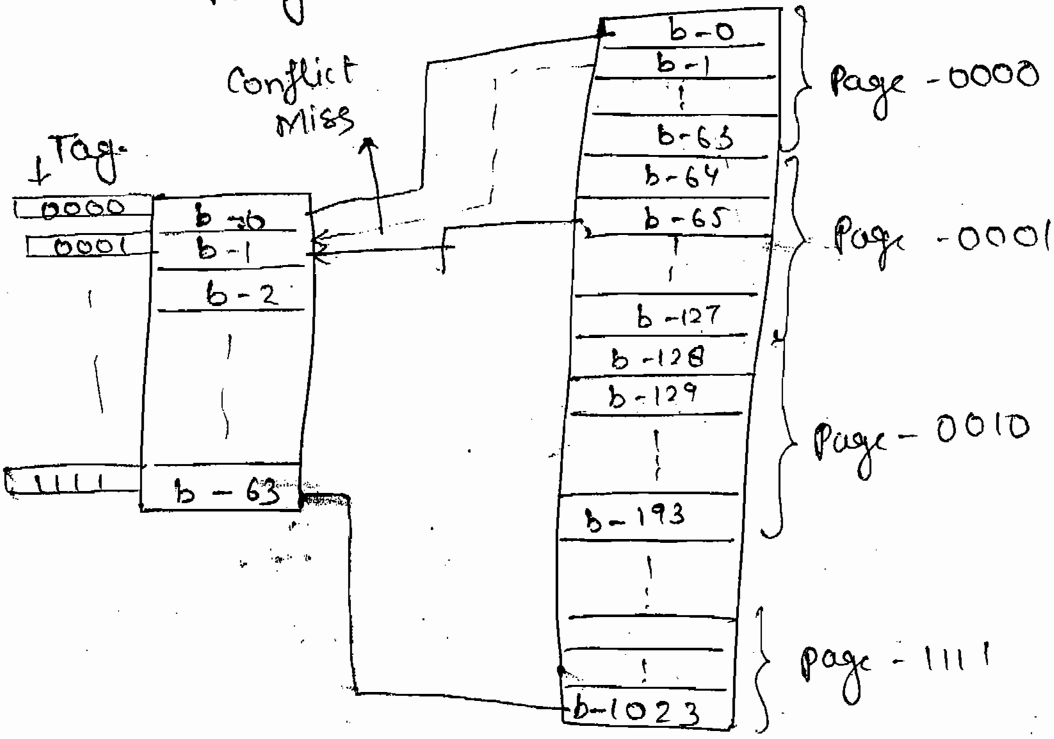
1) Cache Size (Small - KBs)
2) Cache Block Size (large enough)
3) No. of ~~o~~ levels of cache (2-level)
4) Cache Mapping Technique (8-SAM)
5) Cache Replacement Policy (LRU)
6) Cache Updation Policy

Cache Mapping Techniques:

Shekar's Victory
DATE PAGE

The main memory of ~~the system is~~ having 1024 blocks and cache is of 64 blocks. Both partitioned into a block of 16 words.

A) Direct Mapping —



① No. of bit req. to address M.M

$= 1024 * 16$ words

$= 2^{10+4}$

$= 12^{14}$

$\equiv 14$ bits required

· page = No. of blocks in M.M. as in C.M.

No. of pages $= \frac{1024}{64}$

$= 16$ pages

$\downarrow$

$2^4 \equiv 4$ bits (to represent page)

Gold ith block of page ≡ ith block of cache.

Shekar Victory DATE PAGE

- A M.M. block has fixed location in Cache. i.e. i^{th} block of a page mapped to i^{th} block of cache.
- Cache location for a memory block-

$$C.L.(M) = M \% n$$
$$= M \bmod n$$

$M \equiv$ Memory block member
$n \equiv$ No. of cache blocks.

$$C.L.(0) = 0 \% 64 = 0$$
$$C.L.(65) = 65 \% 64 = 1$$
$$C.L.(1023) = 1023 \% 64 = 63$$

- The mapping process is simple.
- The replacement is done when same cache location block is found.
- The Hit ratio is very less.
- Accessing same block from diff. pages simultaneosly is always a Miss (Conflict).
- The address is divided into

$\longleftarrow$ 14 bits $\longrightarrow$

Tag	Block	word
16 pages	64 block	16 words
4 bits	6 bits	4 bits

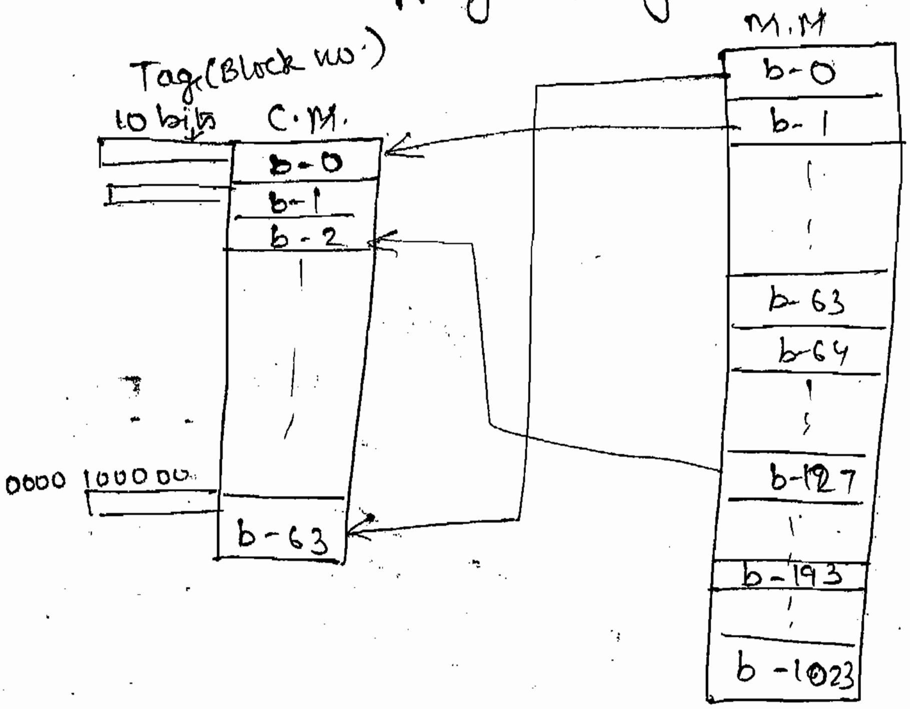
- The higher order bits of the address compared in parallel with all the tags associated with cache blocks. If a match is not found, then it is a Miss!

- The delay due to tag comparison is called Settling time or Hit latency, or ~~Hit Delay~~.

- The no. of bits in the tag is called Tag Length or Tag Size.

- The max^{m}. no. of tag comparisons = 1.

B) Associative Mapping — (Fully Associative Mapping)



- A M.M. block can be placed at any location of cache.
- The replacement is done only when the cache is full.
- Member of Conflict Miss is 0.
- Hit Ratio is very high.
- The address is divided into

← 14 bits →

Tag	Words
10 bits	4 bits

Gold

Shetty's Victory
DATE PAGE

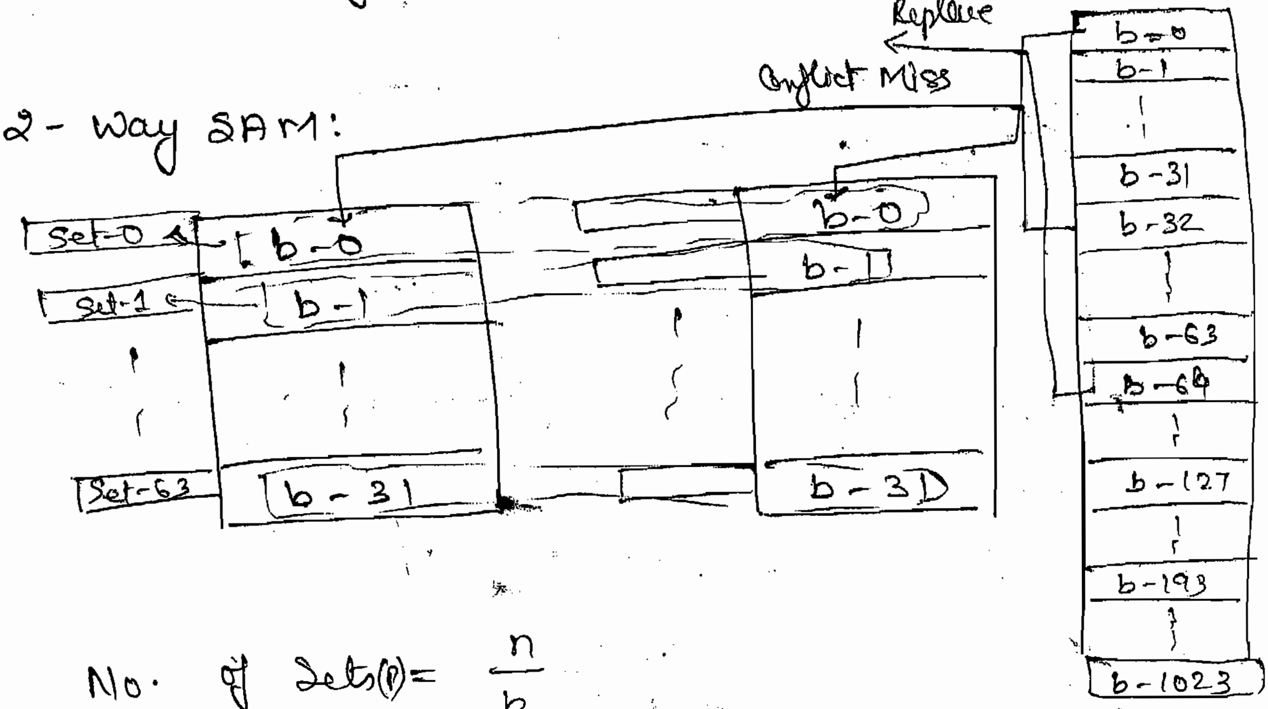
- The tag comparison is done sequentially.
- The complexity of comparison hardware is more.
- Max^{m}. no. of tag comparisons = no. of cache blocks (n)

c) Set Associative Mapping (SAM) –

K-way SAM.
|
"2" – " "
"3" – "SAM"
|
"16" " SAM"

- K is no. of blocks in a set.

2- way SAM:



No. of Sets(P) = $\frac{n}{k}$

$= \frac{64}{2} = 32$ Sets

No. of pages = $\frac{1024}{32}$

= 32 pags

= 5 bits req.

page-00000
page-00001
|
|
page 11111

Shekar's Victory
DATE PAGE

- The Set location of the M.M. block

$$S.L.(M) = M \bmod P$$

E.g.:

$$S.L.(0) = 0 \% 32$$
$$= 0$$

$$S.L.(32) = 32 \% 32$$
$$= 0$$

$$S.L.(64) = 64 \% 32$$
$$= 0$$

- A M.M. block can be placed with k. alternatives in a set.
- The replacement is done when the set is full.
- Hit Ratio is Optimal.
- The address is divided into

← 14 bits →

Tag	Set	Words
5 bits	5 bits	4 bits

5 bits ↓ 32 pages

- No. of tag comparison = k.
- The complexity of h/w is optimal.
- It is a combination of D.M. and A.M.
- It is a collection of D.Ms.
- If k = 1, SAM is DM.

Gold

Q-1 A processor refers to the Cache Memory 1000 times. Out of which 150 ~~references are~~ resulting page fault due to conflicts, 100 of them are due to capacity limitations and 100 of them are due to Compulsory page faults. What is the hit ratio in D.M. & A.M.?

		DM	AM
150 →	Conflict	150	X
100 →	Capacity	100	100
100 →	Compulsory	100	100
		350	200 miss

DM

$$H.R. = \frac{650}{1000} \times 100 = 65\%$$

AM

$$HR = \frac{800}{1000} \times 100 = 80\%$$

Q-2 Consider a cache with 256 blocks, of 16 words each. The M.M. is addressed with 16 bits. How the address is divided or what is the tag size -

1) DM

← 16 bit →

tag	block	words
4 bits	256 blk 8 bits	16 words 4 bits

Tag Size = 4

2) AM

← 16 →

tag	words
12 bits	4 bits

Tag Size = 12

3)

4) (

Not

tag

• W

•

Shekar's Victory
DATE PAGE

3) 4-SAM

tag	Set	word
6 bit	64 Set ↓ 6 bits	4 bits

$$\text{No. of Set}(P) = \frac{n}{k} = \frac{256}{4} = 64$$

Tag Size = 6

4) 8-Way SAM

~~tag~~	Set	Word
7 bits	5 bits	4 bit

$$P = \frac{256}{8} = 32$$

Tag Size = 7 bits

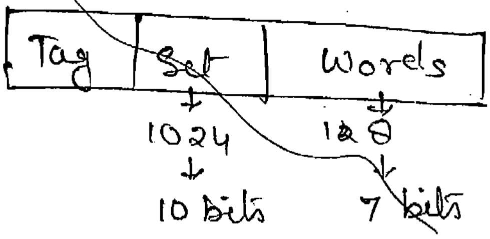
<u>Note:</u> The increase of set size (k) increases tag size.

- With respect to, Complexity → k is small.
- " " " Performance → k is large.
  (Hit Ratio)

Gold

Q- A 4-way SAM cache consists of a total of 64 blocks. The main memory contains 4096 block, each with 128 words. How the add. is divided?

$$P = \frac{4096}{4} = 1024$$



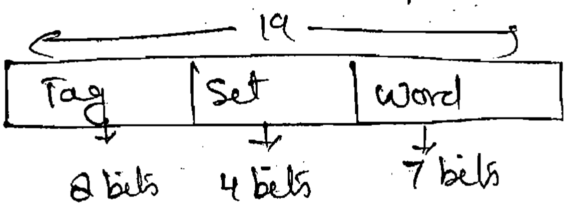
4096×128

$= 2^{12+7}$ Words

$= 2^{19}$ words

↓

19 bits req. to add. M.M.



$$P = \frac{64}{4} = 16 \text{ Sets}$$

Q- Consider a 8 Million word M.M. and 256 block cache. Both partitioned into 64 word blocks.

1) How the add. is divided?

2) What is the tag size or tag comparator size?

3) Max^{m}. no. of tag comparisons.

4) Additional memory for tags.

5) Cache Capacity or total Cache Size

For DM, AM, 4-SAM & 8-SAM.

64

each

Shekar's Victory
DATE PAGE

DM 8 M

8 M

$8 \times 2^{10} \times 2^{10} \Rightarrow 2^{23}$ words

↓

23 bits

1)

← 23 →

tag	block	word
↓	↓	↓
9 bits	256	64
	↓	↓
	8 bits	6 bits

2) 9

3) 1

4) $\boxed{\frac{\text{No. of block} \times \text{tag Size}}{8}} = \frac{256 \times 9}{8}$

5) $\boxed{\text{C.C.} = \text{Cache Size} + \text{Tage Memory} + \text{Dirty bit Memory}}$

=

A.M

1)

← 23 →

tag	Word
↓	↓
17 bits	6 bits

2) 17

3) 256

4) $\frac{256 \times 17}{8}$

5) C.C.$_1$ =

ock

?

4-SAM

1)

← 23 →

17	5	6

$P = \frac{256}{4}$

$= 64$

2) 11

3) 4

4) $\frac{256 \times 11}{8}$

5)

8-SAM

12	5	6

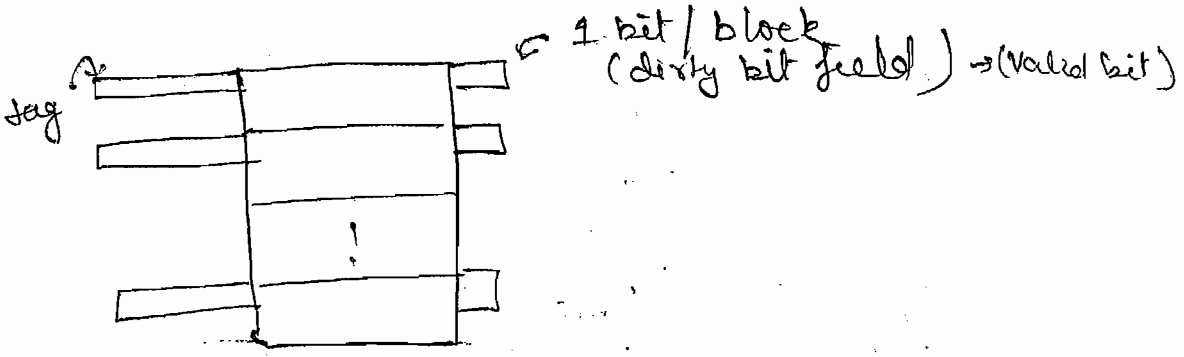
2) 12

3) 8

4) $\frac{256 \times 12}{8}$

5)

Cache Capacity



C.C. = Cache Size + Tag Memory + Dirty Bit Mapping

Q.No.1:

Case 1:

By default 1 word = 1 byte.

$C.C. = 256 \times 64 \times 1\text{ byte} + \frac{256 \times 9}{8}$

$= -$

Case 2:

Let Size of a word is 32 bits (4 byte)

$C.C. = 256 \times 64 \times 4 + \frac{256 \times 9}{8}$

* If dirty bit Included,

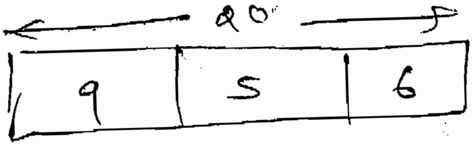
$C.C. = 256 \times 64 \times 4 + \frac{9 \times 256}{8} + \frac{256}{8}$

1 bit)

Q- Consider a 4-SAM cache, ~~with~~ with a total of 128 lines. Each line holds 64 words and the CPU generates 20 bit address of a word ~~[illegible]~~ in M.M. How the add. is divided.

An

$P = \frac{128}{4} = 32$ ~~128~~ 2



Q- A 2-SAM require 8 bit tag comparator while ~~direct~~ DM requires 6 bits tag comparator. If the delay of tag comparator is 120/k nS where k is size of tag comparator. Let there is a multiplexer of delay 3nS, what is the settling time in DM & SAM?

An 120 D.M.

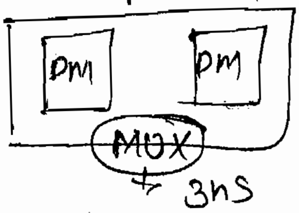
No. Multiplexer is Required so

$S.T. = \frac{120}{k} = \frac{120}{6}$

$= 20$ nS

2 - SAM

Multiplexer is req.



$S.T. = \frac{120}{8} + 3 = 18$

Gold

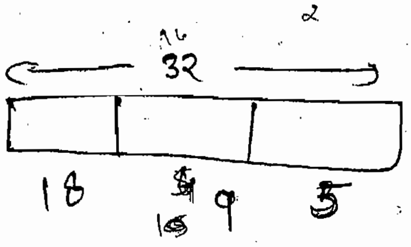
Q- Consider 2 Cache Organisations, 1 is of 32KB 2-SAM with 32 ~~bit~~ byte block size. Other is of same size but DM. The size of address is 32-bits in both cases. A 2×1 MUX has latency of 0.6nS while a k-bit tag comparator has a latency of k/10 nS. the heat latency of DM is H_1 & 2SAM is H_2.

1) The value of H_1 is ____.

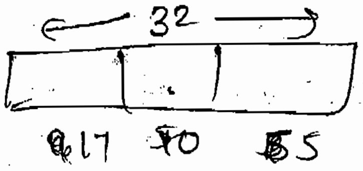
A) 2.4 B) 2.3 C) 1.8 d) 1.7

An No. of block = $\frac{32KB}{32B}$ = 1K

2-SAM



DM



$$H_2 = 0.6 + \frac{k}{10}$$
$$= 0.6 + \frac{18}{10}$$
$$= 2.4$$

$$H_1 = \frac{k}{10}$$
$$= \frac{17}{10} = 1.7$$

Cache Replacement Policy:-

- A replacement policy is required for AM & SAM but not for DM.
- The replacement Policies are aimed to minimize Miss penalty for future references.

Shankar's Victory
DATE PAGE

The Replacement Strategy can be

1) Random - No specific Criteria to replace a block.

2) FIFO - The block which enters first is the candidate for replacement.

3) LRU - The block which has no references from the longest time. (Default method), called optimal algo.

4) LFU (Least Freq. Used) - the block with fewest references (counting method).

2-bits
6nS
of
SAM

Q - Consider a direct mapped cache with 8 Cache blocks (0-7). If the memory block requests are in the order (3, 5, 2, 8, 0, 63, 9, 16, 20, 17, 25, 18, 30, 24, 2, 63, 5, 82, 17, 24), w.o.f. memory blocks will not be in the cache at the end of sequence?

A) 3 [✓ B) 18] C) 20 D) 30

Ans -

0	~~8~~, ~~0~~, ~~16~~, 24
1	~~9~~, ~~17~~, ~~25~~, 17
2	~~2~~, ~~18~~, ~~2~~, 82
3	3
4	20
5	5
6	30
7	63

3%8 = 3

5%8 = 5

Hit Ratio = 3/20

M. R. = 17/20

&SAM

Ni ze

Gold

Q- Consider a Fully Assocative Cache, with 8 Cache blocks and the following sequence of memory ~~block request~~ (4, 3, 25, 8, 19, 6, $\textcircled{25}$, $\textcircled{8}$ 16, 35, 45, 22, $\textcircled{8}$, 3, $\textcircled{16}$ $\textcircled{25}$, 7). If LRU is used, which cache block will have memory block 7?

Shekar's Victory
DATE PAGE

A) 4 ✓B) 5 C) 6 D) 7

An

0	~~4~~, 45
1	~~3~~, 22
2	~~25~~ 25
3	8
4	~~19~~, 3
5	~~6~~, $\textcircled{7}$
6	~~25~~ 16
7	35

H.R. = 5/17

M.R. = 12/17

Q- Consider a 4-SAM with 16 Cache blocks, the memory block requests are in the order (0, 255, 1, 4, 3, 8, 133, 159, 216, 129, 63 $\textcircled{8}$, 48, 32, 73, 92, 155), w.o.f. memory block, will not be in the cache, if LRU is used?

A) 3 B) 8 C) 129 ✓D) 216

An

Set 0	~~0~~, 48
	~~4~~, 32
	~~8~~, 8
	~~216~~, 92
Set 1	9
	133
	129
	73
Set 2	
Set 3	~~255~~, 155
	3
	159
	63

S.I.(M) = m % P

$P = n/k = \frac{16}{4} = 4$

S.I.(0) = 0 % 4 = 0

S.I.(255) = 255 % 4 = 3

H.R. = 1/17

M.R. = 16/17

Q- Co
If LR
reque
prese
Set
Set
Set 2
Set

Q- (
4 b
use
follo
8,
M
Set0
Set2

blocks
~~west~~-
RU is
7?

Q- Consider a 2-SAM has a total of 8 cache blocks. If LRU is used to replace for the memory block request (0,3,5,9,7,⓪,16,55). which memory block will present in the cache at the end of sequence?

Shekar's Victory

$0 \% 4 = 0$ $P = \frac{8}{2} = 4$

Set 0	0
	16
Set 1	5
	9
Set 2	
Set 3	~~3~~,55
	7

A) 0,3,5,9,16,55

B) 0,3,5,7,9,16,55

C) 0,5,7,9,16,55 ✓

D) 3,5,7,9,16,55

5),
, if

Q- Consider a small 2-SAM with a total of 4 blocks. For choosing the blocks to be replace, use LRU scheme. the no. of cache misses for the following sequence of block address is —

$$\begin{array}{ccccc} 8, & 12, & 0, & \textcircled{12}, & 8 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ M & M & M & H & M \end{array}$$ is ——

$\frac{16}{4} = 4$

$P = \frac{4}{2} \% = 2$

Set 0	~~8~~, ~~0~~, 8
	12
Set 2	

2/5

No. of miss = $\frac{4}{5}$ out of 5

Gold

Shekar's Victory DATE PAGE

Q- Consider a 2-SAM, consisting of 2^c memory blocks and 2c cache blocks. the cache location for the memory block K is ____ .

A) K mod 2C

B) K mod 2^c

C) K mod C

D) 2^c mod K

$P = \frac{2C}{2} = C$

$S.L(K) = K \% P$

$= K \bmod C$

Q.- Consider the cache has 4 blocks. For the memory references (5, 12, 13, 17, 4, 12, 13, 17, 2, 13, 19, 13, 43, 61, 19). What is the H.R. ?

i) FIFO

ii) LRU

iii) DM

iv) 2-SAM (LRU)

i) FIFO

~~5~~ ~~4~~ 43
~~12~~ ~~2~~ 61
13
~~17~~ 19

12 13 17

H.R = $\frac{6}{15}$

(ii) LRU

0	~~5~~ 4
1	12
2	13
3	17

12, 13, 17

(iii)

0	~~12~~, ~~4~~ 12
1	~~5~~, ~~13~~ ~~17~~ ~~13~~ ~~17~~ ~~13~~ 61
2	2
3	~~19~~ ~~43~~ 19

13

H R = $\frac{1}{15}$

(iv)

P = 2

Set 0	~~12~~ 2
	4
Set 1	~~5~~ ~~17~~ ~~13~~ 61
	~~13~~ ~~19~~ ~~43~~ 19

12, 13, 17, 13,

$\frac{4}{15}$

Cach

TO

1) W

and r

↳ I

↳ Alu

↳ Th

dropic

↳ E

↳ Th

2) Wr

associ

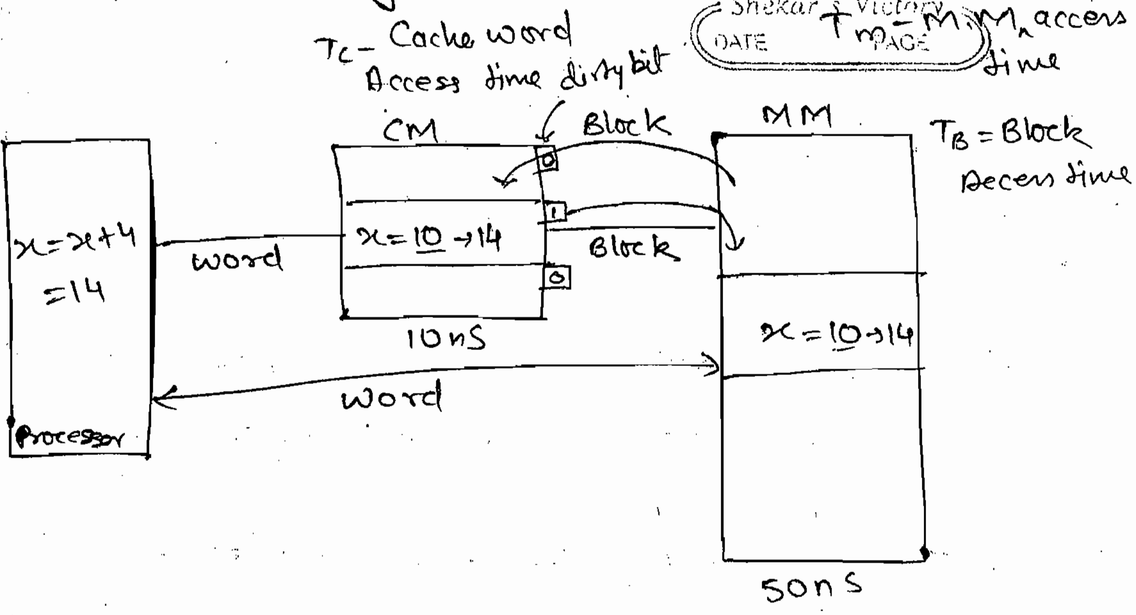
↳ Nc

↳ Eff

blocks
the
p
el c
nory
1, 19).
2, 13, 17
P = 2
3, 17, 13,

Cache Updation Policy:

Shekar's Victory
DATE PAGE



The updation policy can be

1) Write - through Policy

↳ Simultaneous updation of a word in C.M. and M.M..

↳ Inconsistency (Cache Coherence) is resolved.

↳ Always I/O has correct data.

↳ The updation is done with ~~bus~~ increased bus trapic and overhead.

↳ Effective for less updation.

↳ The updation takes Tm time.

2) Write - Back Policy

↳ The updation in M.M. is postponed untill the associated block is replaced.

↳ No additional bus trapic.

↳ Effective for more updations.

↳ A word modified in a cache block is ~~dirty~~ dirty block and the associated dirty bit is set ~~to 1.~~

Shekhar's DATE PAGE

↳ The block is transformed to M.M. and updated only when dirty bit is set to 1.

Note:

If no. of dirty blocks is 0, both Write-through & Write-back has same performance.

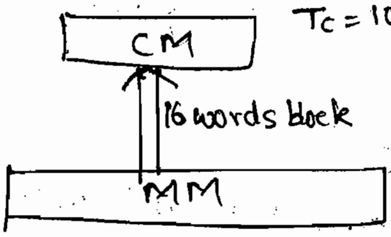
Q- A 64-word cache and ~~mo~~ M.M are divided into 16 word blocks. The access time of Cache & M.M. is 10ns ~~for~~/words & 50ns/word. The H.R. for read operation is 80% & write operation is 90%. Whenever a page fault is generated, the associated block must be brought from M.M. to Cache for both Read & Write Operations. Let there are 40% ~~diffe~~ reference for write Operations.

A) What is the average access time?

B) What is throughput?

~~A~~ If write-through updation scheme is used.

An



$T_c = 10ns/word$

$T_m = 50ns/word$

$T_B = 50 \times 16$
$= 800\ ns$

$H_R = 80\%$ $H_W = 90\%$

$f_R = 60\%$ $f_W = 40\%$

$$\boxed{T_{avg} = f_r * T_{avg(r)} + f_w * T_{avg(w)}}$$

y block

updated

through

ided

o &

I.R.

is.

do

ore

l.

Shekar's Victory
DATE PAGE

Case-II

$T_{avg(r)} = H_R * T_c + (1 - H_R)(T_B + T_c)$

$= 0.8 \times 10 + 0.2 \times 810$

$= 8 + 162$

$= 170$ nS

Simultaneous updation in Cache & MM

$T_{avg(w)} = H_w * T_M + (1 - H_w)(T_B + T_M)$

$= 0.9 \times 50 + 0.1 * 850$

$=$ ~~$49 + 81$~~ $45 + 85$

$=$ ~~90 nS~~ ~~50 nS~~ 130 nS

Case-I

$T_{avg(r)} = H_R * T_c + (1 - H_R) T_M$

$= 0.8 \times 10 + 0.2 \times 50$

$= 8 + 10$

$= 18$ nS

$T_{avg(w)} = H_w * T_M + (1 - H_w) T_m$

$= 0.9 \times 50 + 0.1 \times 50$

$= 45 + 5$

$=$ ~~14 ns~~ 50 nS

~~$T_{avg} = 170 + 18 = 188$ nS~~

~~$T_{avg} = 170 + 130 = 300$ nS~~

$T_{avg} = 0.6 \times 170 + 0.4 \times 130$

$= 102 + 52$

$= 154$ nS

$$\boxed{\text{Throughput (Performance)} = \frac{1}{T_{avg}}}$$

$= \frac{1}{154 \times 10^{-9}} = \frac{1000}{154}$ M words/sec.

$= 6.5$ MW/sec.

Gold

Q- In the above problem if write-~~Sh~~back via updation is used, what is Tavg and throughput?

Ans

$$T_{avg} = f_r * T_{avg(r)} + f_w * T_{avg(w)}$$

Case-2

$$T_{avg(r)} = \{H_r * T_c + (1-H_R)(T_B+T_c)$$

$$= 170\ ns$$

Cache Updation

MM updation with x fer

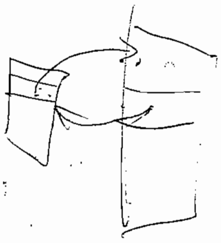
Mapping MM to CM

$$T_{avg(w)} = H_w * (T_c + T_B) + (1-H_w)(T_B + T_c + T_B)$$

$$= 0.9 \times 810 + 0.1 \times 1610$$

$$= 729 + 161$$

$$= 890\ ns$$



Case-1

$$T_{avg(r)} = H_R * T_c + (1-H_R)\, T_M$$

$$= 18\ ns$$

$$T_{avg(w)} = H_w * (T_c + T_B) + (1-H_w)\, \{T_m$$

$$= 0.9 \times 810 + 0.1 \times 50$$

$$= 729 + 5$$

$$= 734$$

$$T_{avg} = 0.6 \times 170 + 0.4 \times 890$$

$$= 102 + 356$$

$$= 458\ ns$$

$$\text{Throughput} = \frac{1}{T_{avg}}$$

$$= \frac{1}{458 \times 10^{-9}} = \frac{1000}{458}\ M\ \text{words/sec} = 2.2$$

Cach

Q- Con

occupie

and d

A) wh

Ans

Step 2

Step3

A

A

2 elemen

Al

Al

Shekar's Victory
DATE PAGE

$$\boxed{\frac{P_{CA}}{P_{CB}} = \frac{T_{avg\,cB}}{T_{avg\,cA}}}$$

Cache Memory and Arrays:-

Q- Consider an array is A[100] and each element occupies 4 words, A 32 ~~bit~~ word cache is used and divided into 8 word blocks.

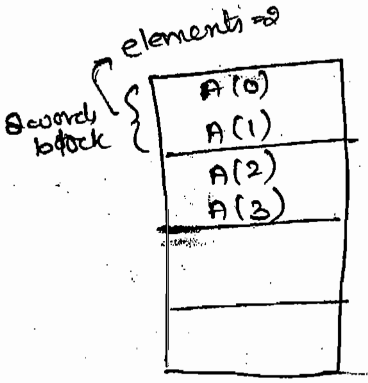
A) What is the H.R. for the statement

for (i = 0 ; i<100 ; i++)
A[i] = A[i] +10 ;

Ans Step 1
Memory Structure

a(0)	a(1)	a(2)	a(3)	a(4)	— — —

Step 2



Step3

	R	W
A(0) →	M	H
A(1) →	H	H

2 elements will shift from M.M to CM

A(2) →	M	H
A(3) →	H	H

$$H.R. \Rightarrow \frac{3}{4}$$

$$\Rightarrow 75\%$$

b) what is the H.R for

```
for (i=0; i<100; i++)
    x = A[i]+10;
```

An H.R. = $\frac{2}{4}$ = 50%

	R
A(0) →	M
A(1) →	H
A(2) →	M
A(3) →	H

c) the no. of times, block 0 modified?

~~A(0)~~	A(8)
~~A(1)~~	A(9)
~~A(2)~~	A(10)
~~A(3)~~	A(11)
A(4)	
A(5)	
~~A(6)~~	
~~A(7)~~	

0, 8, 16, 24, 32, 40, 48, 56,
64, 72, 80, 88, 96
= 13 times

Q:- Cosider an array has 100 elements & each ~~eac~~ elements occupies 4 words. A 32 ~~bit~~ word cache is used & divided into a block of 8 words
what is the H.R. –

```
A) for (i=0; i<10; i++)
       for (J=0; J<10; J++)
           A[i][J] = A[i][J]+10;
```

~~02~~ 1 elemets ← 8 Wans,

100 elemets
1 elem

Steps;
①
a)
nx
b) Color
nxn
② Ho
③ (1)R

Steps

① Memory Structure

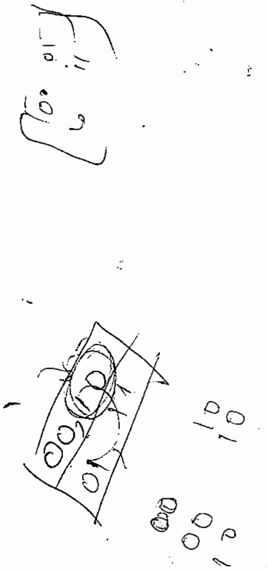
Shekar's Victory
DATE PAGE

$$\begin{bmatrix} {}^{00}7 & {}^{01}6 & {}^{02}4 \\ {}^{10}5 & {}^{11}1 & {}^{12}2 \\ {}^{20}8 & {}^{21}9 & {}^{22}3 \end{bmatrix} 3\times 3$$

a) Row Major Order (default)

7	6	4	5	1	2	8	9	3
00	01	02	10	11	12	20	21	22

nxn → 00 01 02 03 --- 10 11 12 ----



b) Coloumn Major Order

7	5	8	6	1	9	4	2	3
00	10	20	01	11	21	02	12	22

nxn → 00 10 20 30 40 --- 01 11 21 31 ---

② How many element a block can hold → 2 elements

③ (i) Row majer Order

	R	W
A(0)(0) →	M	H
A(0)(1) →	H	H
A(0)(2) →	M	H
A(0)(3) →	H	H

A(0)(0)
A(0)(1)

A(0)(2)
A(0)(3)

$$H.R. = \frac{3}{4}$$
$$= 75\%$$

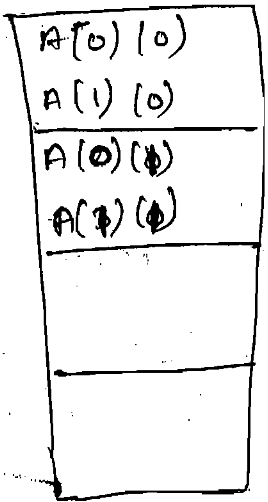
Gold

Shekar's Victory
DATE PAGE

(ii) Column Major Order

	R	W
A[0)(0) →	M	H
A(0)(1) →	M	H
¦		

H.R. = $\frac{1}{2}$
= 50%



A(0)(0)
A(1)(0)
A(0)(1)
A(1)(1)

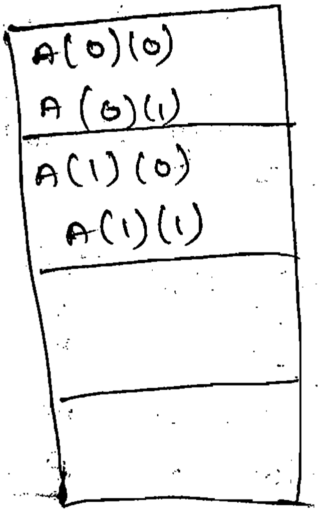
Ans b) for (i=0; i<10; i++)

```
for (i=0; i<10; i++)
    for(J=0; J<10; J++)
        A[J][i] = A[J][i]+10;
```

(i) Row Major Order

	R	W
A(0)(0) →	M	H
A(1)(0) →	M	H
¦		

H.R. = $\frac{1}{2}$
= 50%

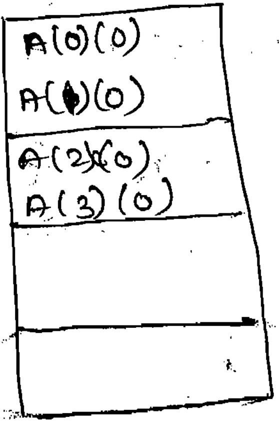


A(0)(0)
A(0)(1)
A(1)(0)
A(1)(1)

(ii) Column Major

	R	W
A(0)(0) →	M	H
A(1)(0) →	H	H
A(2)(0) →	M	H
A(3)(0) →	H	H
¦		

H.R. = $\frac{3}{4}$ = 75%



A(0)(0)
A(1)(0)
A(2)(0)
A(3)(0)

Q.
128
of si
each

for (
for

initia
of ce
M2.
(i) M
A) C

An
S
P1
Row
00

Step

Step3
A
A
A

Shekar's Victory
DATE 2-D PAGE

Q1 A CPU has 32 KB direct mapped cached with 128 byte block size. Suppose "A" is 2-D array of size 512×512 with elements, that occupying 8 bytes each. Consider the following two program segments -

P_1

```
for(i=0; i<512; i++)
  for(J=0; J<512; J++)
    x = x + A[i][J];
```

P_2

```
for(i=0; i<512; i++)
  for(j=0; j<512; j++)
    x = x + A[J][i];
```

P_1, P_2 are executed independently with same initial state, i, J & x are in registers. Let the no. of cache miss is experienced by P_1 is M_1 & P_2 is M_2.

(i) M_1 = ____

A) 0 B) 1024 C) 16384 D) 262144

An

Step 1

P_1

Row Major

| 00 | 01 | 02 | 11 | 12 | 13 | - - |

Coloumn Major

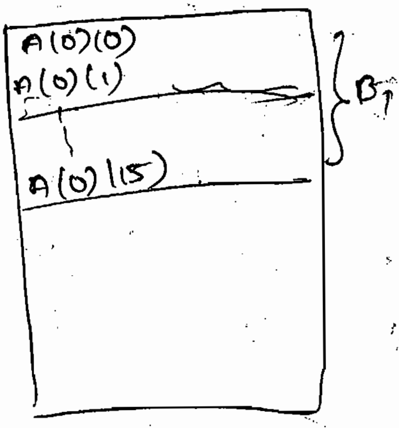
| 00 | 10 | 20 | -- | 01 | 11 | 21 | - - |

Step 2

$\frac{2^7}{2^3}$ $\frac{2^{15}}{2^7} \Rightarrow 2^{\overset{6}{4}}$ 64 blocks

$2^4 \Rightarrow 16$

Each block containing 16 element



Step3

	R
A(0)(0) →	M
A(0)(1) →	H
A(0)(15) →	H

H.R. = $\frac{15}{16}$

M.R. = $\frac{1}{16}$

No. of total misses = $\frac{512 \times 512}{16}$ = 16384

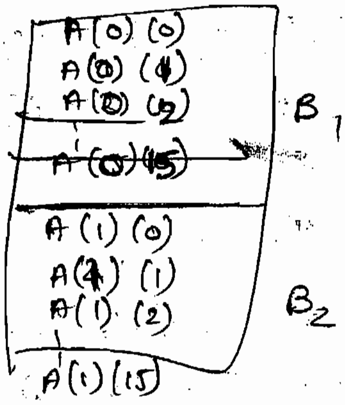
b) $\frac{M_1}{M_2} = ?$

A) 0 B) 16 C) $\frac{1}{16}$ D) 1

Ans

for P_2

	R
A(0)(0) →	M
A(1)(0) →	M
A(2)(0) →	M
⋮	
A(15)(0) →	M



$$H.R. = 0$$
$$M.R. = \frac{16}{16}$$

$$\frac{M_1}{M_2} = \frac{1/16}{16/16} = \frac{1}{16}$$

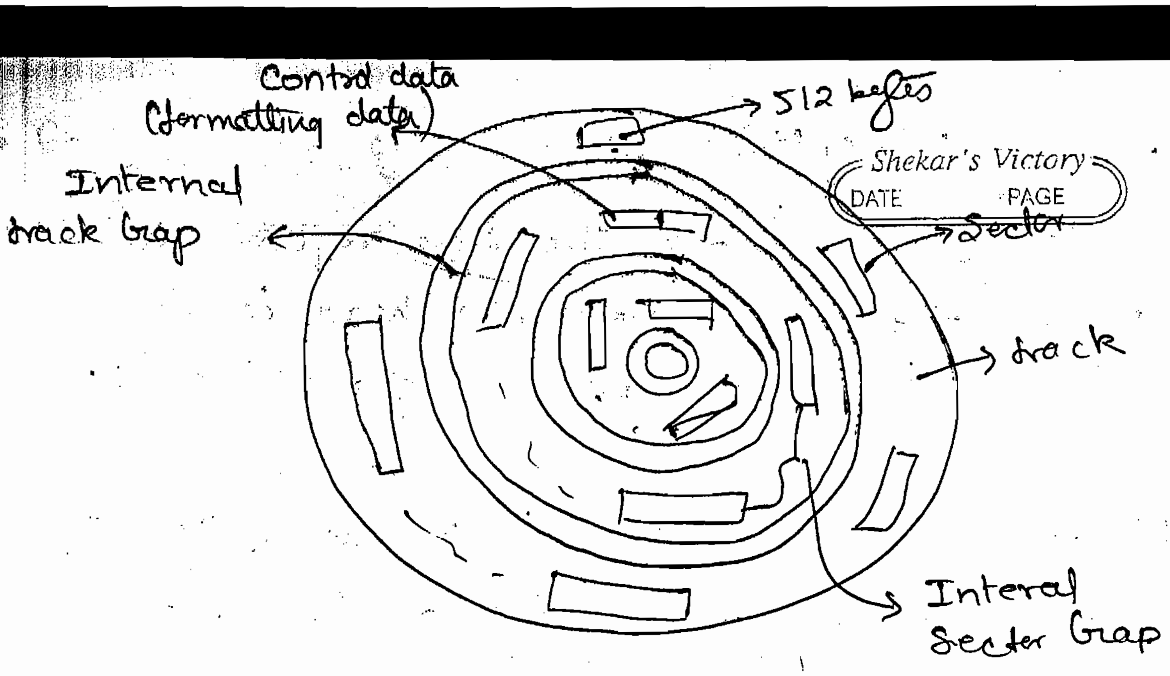
$$M_2 = \frac{512 \times 512}{16} \times 16$$
$$= 262144$$

Secondary or Auxilary Storage Devices:

① Magnetic Disks:

A magnetic disk is a thin circular metal plate, usually recorded on either side. The data on the disk organised as—

Shekar's Victory
DATE PAGE



~~Data in disk~~

- A set on concentric circle called as Track.
- Each track holds same no. of managable units, called Sectors.
- The universal size of a sector is 512 bytes.
- The disk space without format overhead is formatted disk space.
- The basic unit of transfer is a sector.
- The recording density

  e = No. of Bytes/cm.

- Max^{m}. recording density is at inner most track.
- The data tranfer rate

  D = No. of bytes/sec.

- It depends on Rotations per Minute (RPM).

Seek Time (ts) :-

The time required for the R/W head to the desired track.

Gold

Shekar's Victory
DATE PAGE

Rotational Latency (t_r)
or
Latency
or
Rotational delay → The time required to move R/W head to the beginning of desired sector.

- If position of the sector is not known, the avg. rotational latency is one-half of a rotation.

i.e. $t_r = \frac{1}{2}$ rotation time.

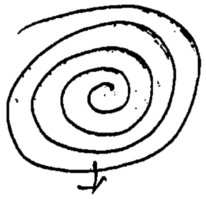
- The access time of the disk

$$\boxed{t = t_s + t_r}$$

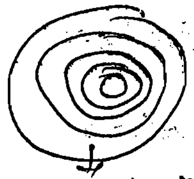
- The average access time of a disk

$$t_{avg.} = t_s + t_r + t_{data\ transfer} + \underset{\substack{\downarrow \\ \text{delay for setup with} \\ \text{controller.}}}{t_{overhead}}$$

- Magnetic disks are semi random access memories. i.e. reaching to the desired sector beginning is random and accessing bytes from it is sequential.

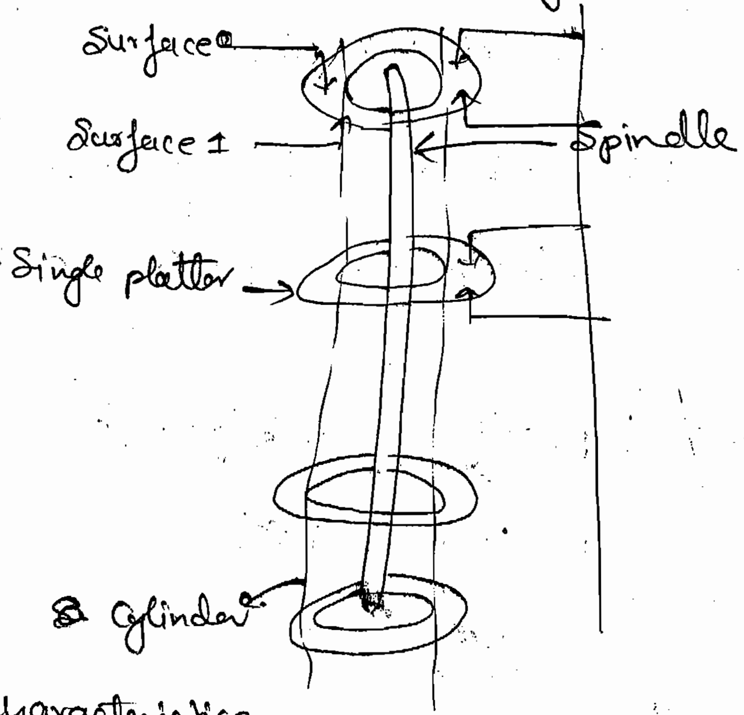


Constant Density



Constant Angular Velocity
(Same amount of data on each track).
(Variable density)

- Each rotation of the disk covers one track.
- Multiple disks can be organised as,



3 Characteristics –

1) Single | Double Sided –
↓
default

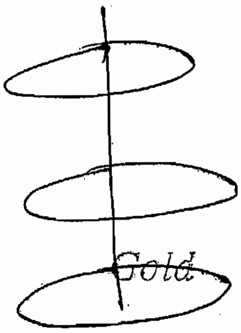
2) Fixed or Movable R/W head.

Fixed ↓ one R/W head / surface ↓ On one rotation n - tracks covered ↓ n = no. of surfaces

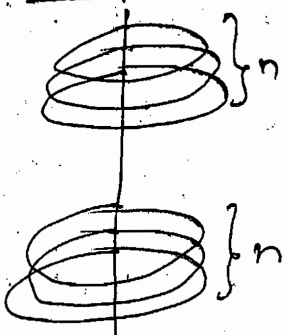
Movable ↳ one R/W head for all surfaces. ↓ One one rotation 1 - track covered.

3) Single or Multiple Platter-

Single Platter



Multiple Platter



One R/W head per platter

Shekar's Victory
DATE PAGE

- A vertical set of all tracks from same position in a disk pack forms a cylinder.

- No. of Cylinders = No. of track in a surface.

Q- Consider a disk pack with the following specification

16 Surfaces, 128 track / surface, 256 sectors / tracks, and 512 bytes / sector.

A) what is the capacity of disk pack?

$C = 16 \times 128 \times 256 \times 512$ bytes

$= 2^{4+7+8+9}$ bytes

$= 2^{28}$ B

$=$ ~~2~~ 256 MB

B) The no. of bits ~~is~~ required to address the sector ?

Total no. of sectors $= 16 \times 128 \times 256$

$= 2^{4+7+8}$

$= 2^{19}$ sectors

$\equiv$ 19 bits required.

c) In the above disk pack, the format overhead is 32 bytes / sector. what is the formatted disk space?

$16 \times 128 \times 256 \times (512-32)$

$= 2^{4+7+8} \times 480$ Bytes

$= 240$ MB

D) In the above disk pack, the format overhead is 64 bytes/sector. How much amount of memory is lost due to formatting?

Shekar's Victory

$16 \times 128 \times 256 \times 64$

$= 2^{4+7+8+6} = 2^{25}$ B

$= 32$ MB

ification

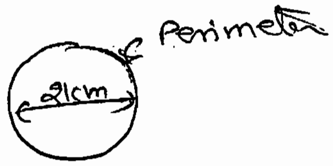
, and

E) Let the diameter of innermost track is 21 cm, what is the max^m. recording density.

$\rho = $ ~~256~~ No. of ~~Byte~~/cm

$= \frac{256 \times 2^{20}}{21}$

$= \frac{2^{28}}{21}$



Perimeter $= \pi * D$

$= \frac{22}{7} \times 21$

$= 66$ cm.

66 cm. ⟶ 1 track capacity

1 cm ⟶ ?

ad.

ace?

$\rho = \frac{1}{66} \times 256 \times 512$ Bytes/cm.

$= \frac{128}{66}$ KB/cm.

$= 1.9$ KB/cm

Gold

F) Let the diameter of innermost track is 21 cm., with 2 KB/c~~m~~m. What is 1 track capacity.

Shekar's Victory
DATE PAGE

~~2×26~~

~~42 KB~~

1 cm → 2KB

66 cm. → 2×66

= 132 KB

G) The disk is rotating at 3600 RPM, what is the data transfer rate?

$$D = \frac{\text{No. of bytes}}{\text{sec.}}$$

3600 rotations → 60×10^3 msec

1 " → $\frac{60 \times 10^3}{3600}$ msec.

= 16.66 msec.

16.66 ms ⟶ 1 track
(n * 1 track) ← fixed R/W head

16.66 ms ⟶ 16 * 256 * 512

1000 ms (1 sec) → ?

$$D = \frac{1000}{16.66} \times 16 \times 256 \times 512 \text{ Byte/sec.}$$

= 125 Mbps

H) Using a R/W head, the disk is rotating at 6000 RPM. What is the data transfer rate?

6000 rotation = 60×10^3 ms

1 " = $\frac{60000}{6000}$ = 10ms

,, with

)

10 ms → 1 track

↓ → ~~100~~ × 256 × 512

1000 ms (1 sec) → ?

$$D = \frac{1000}{10} \times 256 \times 512 \text{ bytes/sec.}$$

$$= 128 \times 100 \text{ Kbps}$$

$$= 13 \text{ mbps}$$

data

I) If the disk system has rotational speed of 3000 rpm, what is the average access time with a seek time of 11.5 msec. ?

$$T_{avg} = T_S + T_r + \cancel{T_{trans}}$$

t_r → half rotation time

$$1 \text{ rotation} \rightarrow \frac{60 \times 10^3}{3000} \text{ msec.}$$

$$20 \text{ msec.}$$

$$t_r = \frac{1}{2} \times 20 = 10 \text{msec.}$$

$$t_{avg} = 11.5 + 10$$

$$= 21.5 \text{ msec.}$$

o RPM-

J) what is the av. access time for transferring 512 bytes of data with following specification ?

Av. seek time = 5msec. , disk rotation = 6000rpm

40 KB/sec.

Controller overhead = 0.1 msec

$T_{avg} = t_s + t_r + t_{data\ tran} + t_{contr}$ ~~overhead~~

t_r

6000 r → $\cancel{3}60 \times 10^3$ msec.

1 rotation → 10 msec.

$t_r = \frac{1}{2} \times 10 = 5$ msec.

$t_{datatran}$ -

40 KB → $\cancel{10}^3$ sec. 1×10^3 msec.

512 B → $\frac{512 \times 10^3}{40 \times 2^{10}}$

$= \frac{\cancel{2}\ 10^3}{80}$

$t_{dt} = 12.5$ msec.

$t_o = 0.1$

$T_{avg} = 5 + 5 + 12.5 + 0.1$

$= 22.5 + 0.1 = 22.6$ msec.

Q- A ~~set on~~ certain moving armed disk storage with one head has the following specifications.

No. of tracks / surface = 200

disk rotation speed = 2400 rpm

tracks storage capacity = 62500 bits

average latency = P msec.

data ~~se~~ transfer rate = Q bits / sec.

What is the value of P & Q?

Ar

Q- A

each

and

storage

min^m.

(i). wo

Ar Tavg. = ~~4~~

disk rotation speed = 2400 rpm

2400 rotation → ~~2400~~ 60×10^3 msec.

1 rotation → $\frac{60 \times 10^3}{2400} = \frac{\cancel{600}\ 150}{\cancel{24}\ 6}$

⇒ 25 msec.

$P = \frac{1}{2} \times 25 = 12.5$ msec.

25 msec → 1 track
62,500 bits

1000 msec. → ?

$G = \frac{1000}{25} \times 62,500$

$= 2.5 \times 10^6$ bits/sec.

Q– A disk pack has 19 surfaces, storage area on each surface has an inner diameter of 22 cm. and outer diameter of 33 cm. The max^m. ~~recorder~~ storage density on any track is 2000 bits/cm and min^m. spacing b/w the tracks is 0.25 mm.
(i) what is the capacity of disk pack.

~~33-22 = 11 = 5.5 cm.~~
~~1 cm → 2000 bits~~
~~2000~~ ~~5.5cm → 11000 bits (1 track capacity)~~

Shekar's Victory DATE PAGE

C = n x no. of tracks x track capacity

* ~~Only in~~ Always inner surface diameter is to be taken.

$$\text{perimeter} = \pi * D$$

$$= \frac{22}{7} \times 22 = 69.14 \text{ cm.}$$

1 cm = 2000 bits

69.14 cm. = 2000 X 69.14 bits

$$\text{1 track capacity} = \frac{2000 \times 69.14}{8} \text{ bytes}$$

$$= 17.28 \text{ KB}$$

$$\text{width} = \frac{33-22}{2} = 5.5 \text{ cm.}$$

x → width of track

y → width of gap

w → width of storage

so,

$$\text{no. of track} = \frac{w}{x+y}$$

$$n_t = \frac{w}{x} \quad (\text{~~as~~ y is not given})$$

$$n_t = \frac{w}{y} \quad (x \text{ is not given})$$

$$\text{no. of track} = \frac{5.5 \text{ cm}}{0.25 \text{ mm}}$$

$$= 220 \text{ tracks.}$$

C = $\frac{19 \times 220 \times 17.28}{2^{10}}$ = ~~70.23 MB~~ 70.5 MB

B) ~~B~~

R/W

Q- A

width

addre

<

c

h

s

1st

(i) the

A) 5

(ii) The

A)(0,15

to be

B) ~~B~~ The disk is rotating at 3600 rpm using a R/W head. What is the data tranfer rate.

3600 rotation → 60×10^3 msec.

1 rotation → $\frac{60 \times 10^3}{3600}$

$= 16.66$ msec.

16.66 msec. → 1 track (17.28 KB)

1000 msec. $= \frac{17.28 \times 1000}{16.66}$

$= 1$ Mbps

Q- A hard disk has 63 sectors/track, 10 platters, each with 2 regarding surface. and 1000 cylinders. The address of a sector is given as

<c, h, s>, where

c → cylinder no.

h → surface no.

s → sector no.

thus, the 0th sector is addreseed as <0,0,0>, 1st sector as <0,0,1> and so on.

(i) the address <400, 16, 29> corresponds to sector no?

A) 505035 B) 505036 ✓C) 505037 D) 505038

(ii) The address of 1039 sector is ?

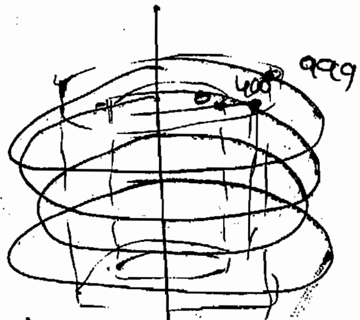
A) (0,15,31> B) <0,16,30> ✓C) <0,16,31> D) <0,17,31>

Gold

Shekar's Victory
DATE PAGE

(a) No. of surfaces = 10 x 2 = 20
No. of tracks = 1000
No. of sector / track = 63

$$\langle \underset{c}{\underset{\downarrow}{400}}, \underset{h}{\underset{\downarrow}{16}}, \underset{s}{\underset{\downarrow}{29}} \rangle$$



1 cylinder = 10 x 2 x 63

400 cyli- = 400 x 10 x 2 x 63 + 16 x 63 + 29

= 505037

(b) A) 0 + 15 x 63 + 31
B) 0 + 16 x 63 + 30
C) 0 + 16 x 63 + 31 = 1039
D) 0 + 17 x 63 + 31

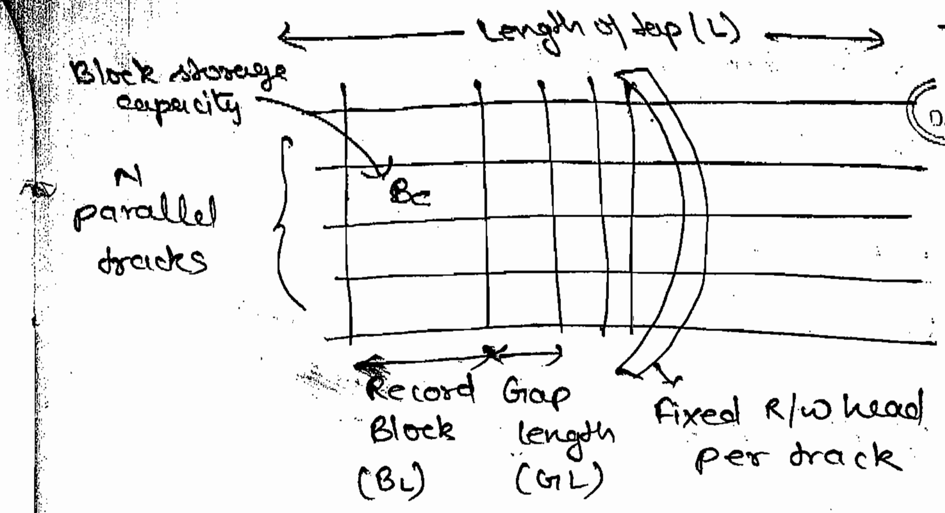
108

② Magnetic Tapes:-

- A magnetic tape is a sequentially processed storage.
- A tape is formed by depositing magnetic field on a very thin wide plastic tape.
- Iron Oxide is the magnitising field or material.
- The data on the tape is organised as

Shekar's Victory
DATE PAGE

1K = 1000



- Each track assumed to have a fixed R/w head and all the ~~tracks~~ ~~are~~ R/w heads operate simultaneously.
- The ~~rec~~ R/w operations takes place by moving the tape with a uniform velocity w.r.t. R/w head.
- The utilization factor of the tape

$$\boxed{u = \frac{B_L}{B_L + G_L}}$$

- Recording density

$$\boxed{\rho = \text{no. of bits or bytes / inch\_track}}$$

or

no. of bytes / inch.

- In tapes, ~~recco~~ ρ is constant.
- D = No. of bytes / sec.
- The data transfer rate depends on
  1) Speed of the tape
  2) Constant recording density
- ~~To~~ The max^m data x-fer rate of tape

$$\boxed{D = V * N * \rho}$$

Gold

- The effective data transfer rate -

$$D_{eff.} = u * D$$

$$= \left(\frac{B_L}{B_L + G_L}\right) \times D$$

- The capacity of the tape

$$\boxed{C = L * N * \rho}$$

= inches * n * no. of byte /inch

= No. of bytes

Q- Calculate the utilization factor of tape, if $G_L = 0.5$ inch, storage density (ρ) = 3000 Bytes/inch and Block storage capacity ~~(?)~~ = 6 KB

$$B_L = \cancel{\frac{3000}{6 \times 1000}} \text{ inch} \qquad \boxed{B_L = \frac{B_c}{\rho}}$$

$$B_L = \frac{6 \times 1000}{3000} \text{ inch}$$

$$= 2 \text{inch}$$

$$u = \frac{2}{2 + 0.5} = \frac{2}{2.5} = \frac{4}{5} = 0.8$$

Q- Suppose that data on the tape is organised into blocks, each containing 32 K bits. A gap of 0.4 inch separates the blocks from each other. The density of recording is 6250 bits/inch. How many bytes may be stored on a tape reel of 2400 feet?

An

$G_L = 0.4$ inch

$\rho = 6250$ bits/inch

$L = 2400$ ft. $= 2400 \times 12$

Capacity of tape = ?

$B_c = 32K$ bits

$$B_L = \cancel{\frac{2400 \times 12}{6250}} \text{ inch} \quad \frac{B_c}{\rho}$$

$$= \frac{32 \times 10^3}{6250}$$

$$= 5.12 \text{ inch}$$

$$\text{No. of blocks} = \frac{L}{B_L + G_L}$$

$$= \frac{2400 \times 12}{5.12 + 0.4}$$

$$= \frac{2400 \times 12}{5.562}$$

$$\simeq 5217$$

C = no. of blocks $\times B_c$

$$= 5217 \times \frac{32}{8} \text{ KB}$$

$$= 20.8 \text{ MB}$$

nch

into
inch
nlg
may

Shekar's Victory DATE PAGE

Q:- In magnetic tape memory, if a 80 track tape of linear velocity 50 inches/sec. and recording density of 110 Kb/inch-track. Max^m. data transfer rate = ?

$$D = N \times V \times \rho$$

$$= 80 \times 50 \times \frac{110}{8} \text{ KBps}$$

$$= 55 \text{ MBps}$$

Q- Consider a tape having 8 mtrs. length, moving with a velocity of 50 cm/sec. Let the linear recording density is 8K bits/track-cm. with 8 parallel track.

(i) What is the capacity of tape?

$$C = L * n * \rho$$

$$= 8 \times 100 \times 8 \times \frac{8}{8} \text{ KB}$$

$$= 6400 \text{ KB} = 6.4 \text{ MB}$$

(ii) What is block storage capacity & ~~ach~~ with a block length of 2.6 cm.

$$B_L = 2.6 \text{ cm.}$$

$$B_C = ?$$

$$\boxed{B_C = B_L * N * \rho}$$

$$= 2.6 \times 8 \times \frac{8}{8} \text{ KB}$$

$$= 20.8 \text{ KB}$$

(iii) 'te
gap of
effective

I|

Data

1. Prog
2. Inte
3. DM

1. Prog
main r
• Any
processo
readin
• Ine

tape
density
te = ?
ing with
ing density
th a

(iii) ~~Leng~~ Let, the length of a block is 2.4 cm and a gap of 0.6 cm separates from each Shekar's Victory DATE PAGE other. What is effective data tranfer rate?

$$D_{eff.} = u \times D \Rightarrow u \times V \times N \times C$$

$$= \frac{2.4}{2.4+0.6} \times 50 \times 8 \times \frac{8}{8} \text{ Kbps}$$

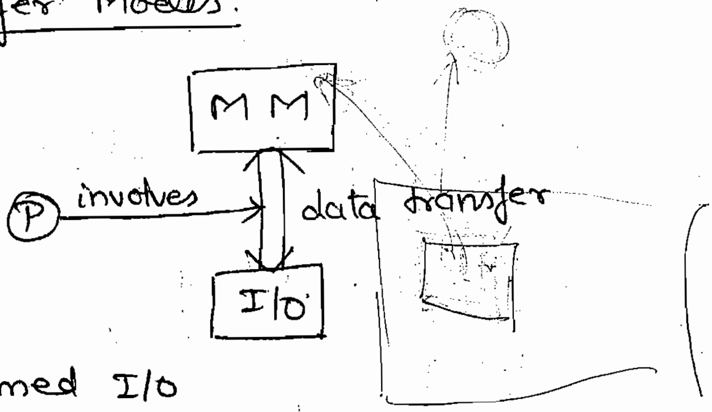
$$= 0.8 \times 50 \times 8 \text{ Kbps}$$

$$= 320 \text{ Kbps}$$

I/O Interfacing:-

(2-3marks)

Data Tranfer Modes:



1. Programmed I/O
2. Interrupt Driven I/O
3. DMA transfer

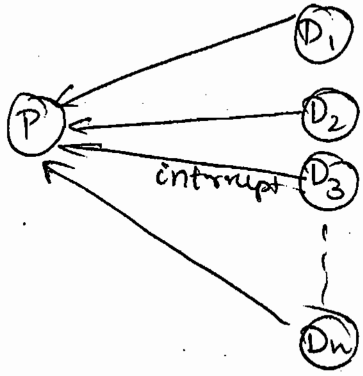
1. Programmed I/O –

- In this mode the I/O has no direct access to main memory.
- Any transfer involves execution of instructions by the processor for device status, I/O initialization, for reading and writing, for knowing status etc.
- Inefficient mode of transfer.

Gold

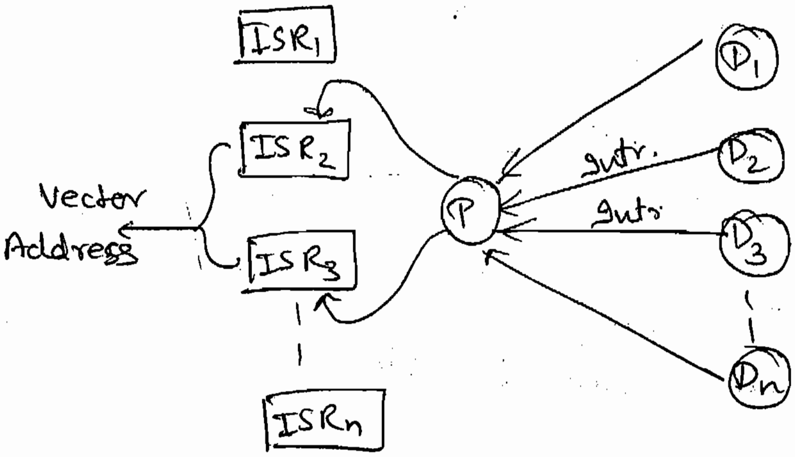
- Use to transfer few(1-2) words.

2. Interrupt Driven I/O —



- In this mode, when the device is ready, it send an interrupt for data transfer.
- Efficient than programmed I/O.
- Not suitable for large volumes of data transfer.
- If more than 1 device interrupts simultaneously, priority driven I/O is used.
- Using this the high priority device address (Vector Address) can be obtained.

A) S/w Approach or S/w Polling

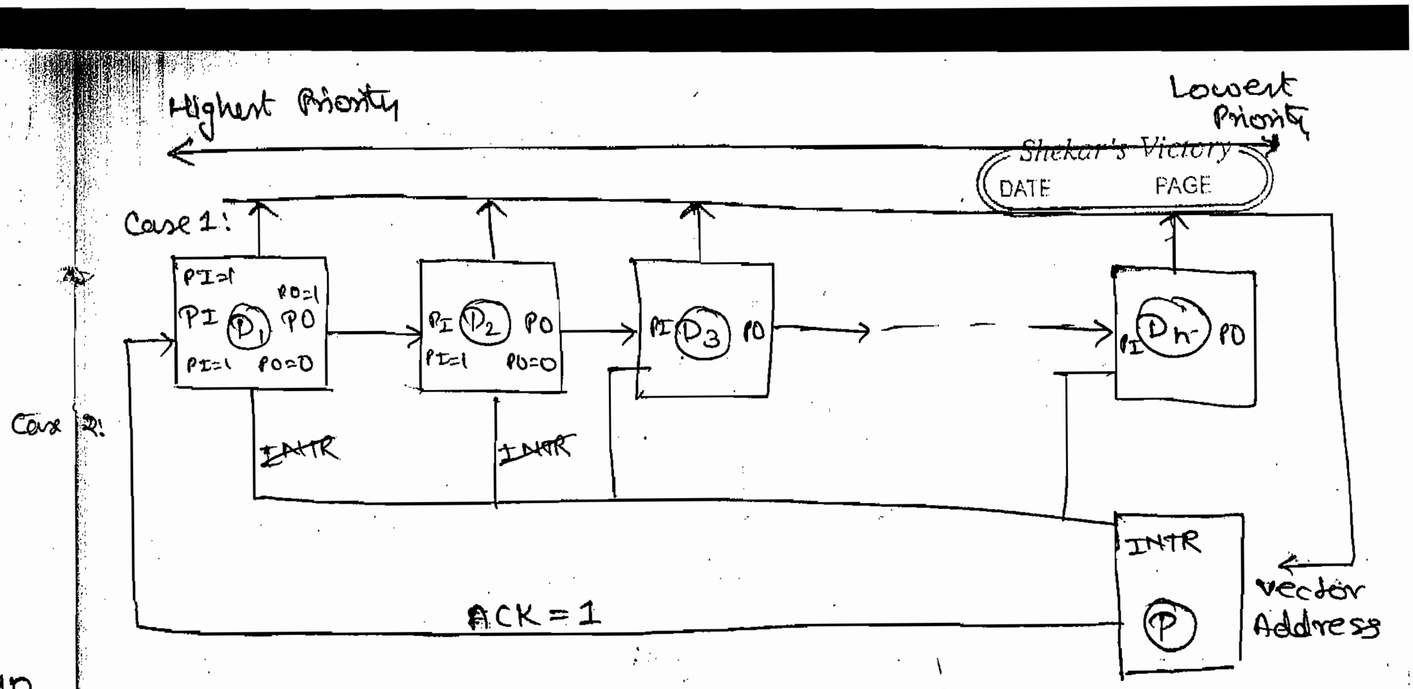


Executing a set of routines for data transfer is inefficient. Hence, not recommended.

B) Hardware Approaches

(a) Serial / Daisy Chain / H/w Polling:

Shekar's Victory
DATE PAGE



PI = Priority in

PO = Priority Out

Q- In Daisy Chain schema for connecting I/o devices w.o.f. statement is correct?

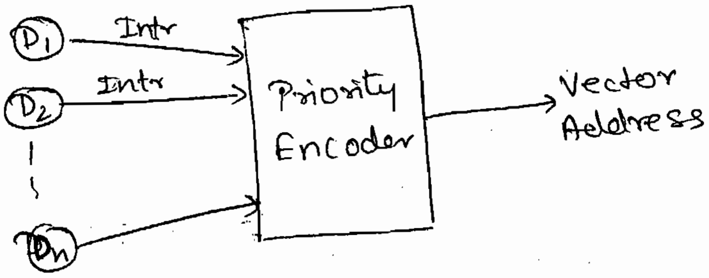
A) It gives non-uniform priority to various devices.

B) It gives uniform priority to all devices.

C) Use to connect slow device only to the processor.

D) A seperate interrupt pin on processor for each device.

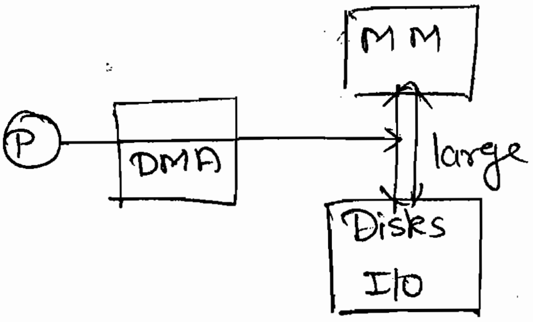
(b) Paralled Priority Driven I/O:



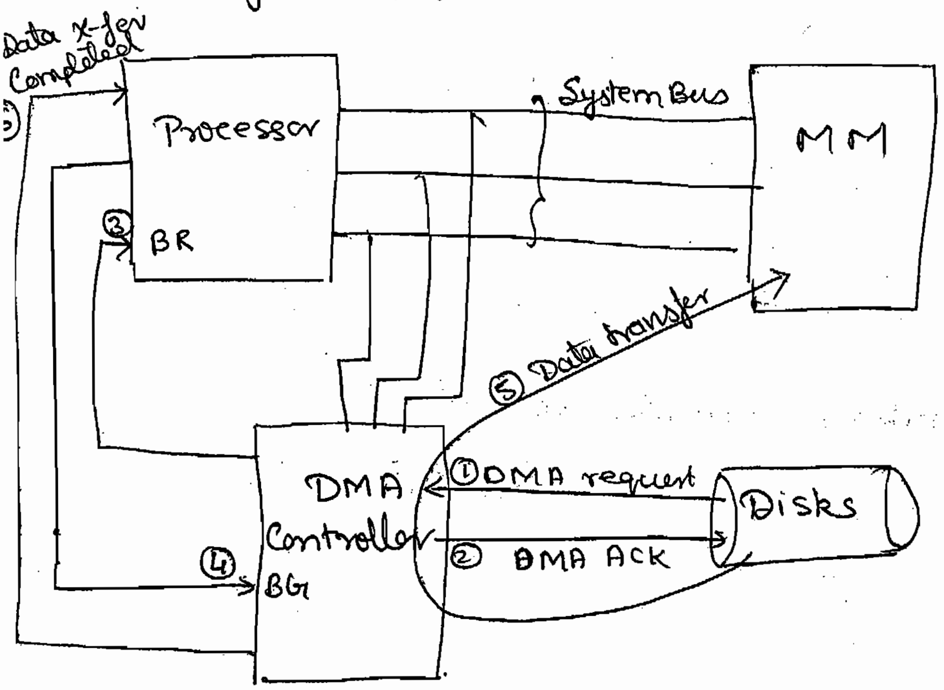
Gold

~~3~~. Direct Memory Access Transfer (DMA) —

Shekar's Victory
DATE PAGE



- During large volumes of data transfer b/w disks and main memory, the processor is completely relieved, using DMA tranfer with DMA controller.



BR: Bus Request
BG: Bus Grant

DMA Transfer Modes :-

1) Interleaved Mode
   - An alternate half cycle i.e. $\frac{1}{2}$ cycle DMA + $\frac{1}{2}$ cycle processor

2) Cycle Stealing Mode
   - DMA returns the bus after a word transfer.

3) Block Mode
   - DMA returns the bus after a block transfer.

4) Burst Mode
   - DMA returns the bus after complete data transfer.

- Some

Q- Co

1) What

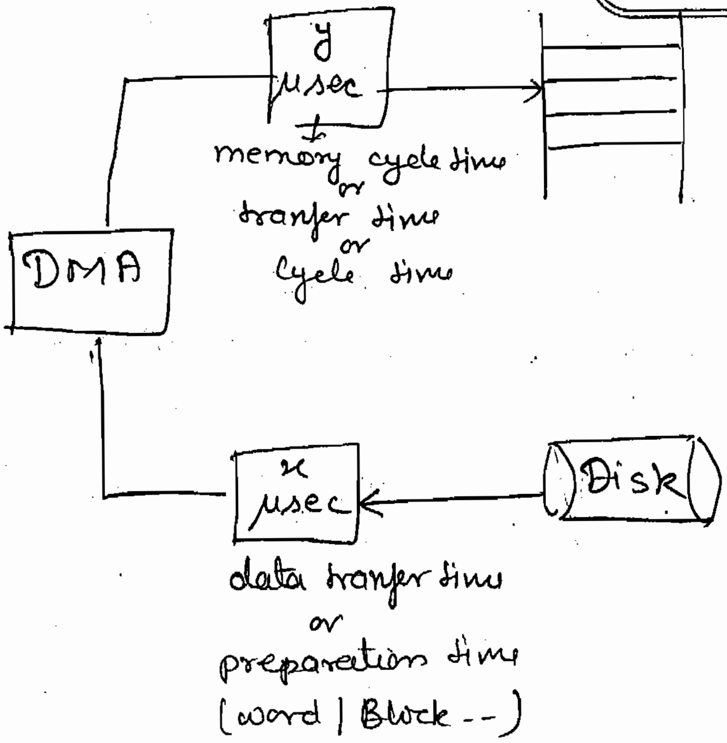
driven I

every

- Some % of performance is reduced due to DMA.

Shekar's Victory
DATE MAM, PAGE

$\boxed{1K = 1000}$



$$\% \text{ CPU Busy} = \frac{x}{x+y} * 100$$

$$\% \text{ CPU Idle} = \frac{y}{x+y} * 100$$

(% of memory cycles stolen)

Q– Consider a processor of 4 MIPS (Million Instruction/sec.)

1) What is the peak data transfer rate?, using Programm driven I/O, which requires 8 instructions overhead for every word?

$\boxed{\text{Dpeak} = \text{No. of words / sec.}}$

4×10^6 instructions ⟶ 1 sec.

1 " ⟶ 0.25 μsec. (1 instruction overhead)

1 word length → 8 instruction overhead

→ $8 * 0.25 \mu sec = 2 \mu sec.$

Gold

$2\mu s \Rightarrow 1$ word

$1\times10^6 \mu s \rightarrow \frac{10^6}{2} \times 1$

$D_{peak} = 500$ K words/sec.

2) what is the peak data transfer rate, using interrupt driven data transfer, which needs 4 instructions overhead?

~~Dpeak~~ 1 instruction overhead = $0.25\mu sec$.

1 word length → 4 ins. overhead

$\rightarrow 4 * 0.25 = 1 \mu sec$.

$1\mu s \rightarrow 1$ word

$1\times10^6 \mu s \rightarrow$ 1×10^6 word/sec.

$D_{peak} = 1\times10^6$ word/sec.

$= 1000$ K words/sec

3) How many times the performance of interrupt driven is better than programmed I/O?

$\boxed{\text{Performance} \propto D}$

$$\boxed{\frac{P_I}{P_p} = \frac{D_I}{D_p}}$$

$$P_I = \frac{1000}{500} \times P_p$$

$$\boxed{P_I = 2 P_p}$$

Q- C

steali

availa

What

by (

A

Q- Consider a device of 10 KBPS is shared memory cycle stealing mode of DMA. Whenever a 4 byte word is available, it is transferred into memory in 50 ms. What is the % of processor performance reduced by DMA?

A

$y = 50ms$

kns. 10 K byte ⟶ 1 sec.

$$4 \text{ byte} \longrightarrow \frac{1}{10 \times 1000} \times 4 \times 10^6 \ \mu sec.$$

Data transfer time (x) = $400 \mu sec.$

$x = 400 \mu sec.$

$$\underset{\text{performance reduced}}{\underset{\downarrow}{\% \text{ CPU idle}}} = \frac{y}{x+y} \times 100$$

$$= \frac{50}{400+50} \times 100$$

$$= \frac{50}{450} \times 100 = \frac{100}{9}$$

$= 11.11 \%$

Gold

Shekar's Victory
DATE PAGE

Q- Consider a disk drive of following specifications—

16 surfaces

512 track / surface

512 sectors / track

1 KB / sector &

3000 RPM with a single r/w head. Whenever a 4 byte word is available, it is transferred b/w memory and I/O with a memory cycle time of 40 nsec. What is the % of processor is blocked due to DMA?

$y = 40 \text{ nS}$

3000 rotations $\rightarrow$ 60×10^{3} msec.

1 rotation $\rightarrow$ $\frac{60 \times 10^{3}}{3000}$ mS

$\rightarrow$ 20 mS

(1 - track)

1 - track $\longrightarrow$ 20mS
$\downarrow$
(512 × 1KB)

4 bytes $\longrightarrow$?

$$x = \frac{20 \times 4}{512 \times 10^{3}} \text{ mS}$$

$$= \frac{10 \times 2^{3}}{2^{9} \times 10^{3}} \times 10^{6} \text{ nS}$$

$$= \frac{10 \times 10^{6}}{2^{6} \times 10^{3}} \text{ nS} = 156.25 \text{ nS}$$

$$\% \text{ idle} = \frac{y}{x+y} \times 100$$

$$= \frac{40}{40 + 156.25} \times 100 =$$

ever a

memory

c. what

II

Shekar's Victory
DATE 26|11 PAGE

Q. The storage area of a disk has the inner most diameter of 10cm. and outer most of 20 cm., Max^m. recording density is 1400 bits/cm., the disk rotates at 4200 rpm. The M.M. of system is having 64 bit word and 1 μsec cycle time. If Cycle stealing is used for data transfer, what is the % of memory cycles stolen for transferring 1 word.

A) 0.5 % B) 1 % [✓ C) 5 %] D) 10 %

Ans $D_1 = 10\text{cm}$, $D_2 = 20\text{cm}$, $y = 1\ \mu\text{sec}$.

$\rho = 1400$ bits/cm

$4200 \text{ rotation} \longrightarrow 60\times10^3\ \mu\text{sec}$.

$1 \text{ rotation} \longrightarrow ?$

$$\underset{(1\text{ track})}{1\text{ rotation}} = \frac{60\times10^3}{4200} = 14.28\text{ ms}.$$

$$\% = \frac{y}{x+y}\times 100.$$

Parimeter = πD

↓

1 - track space $= \frac{22}{7}\times 10$ cm $= 31.4$ cm.

$1\text{ cm.} \longrightarrow 1400$ bits

$31.4\text{ cm.} \longrightarrow ?$

1 track capacity $= 31.4\times1400 = 44$ K bits.

$\underset{(44\text{ k bits})}{1\text{ track}} \longrightarrow 14.28\text{ ms}$

$64\text{ bits} \longrightarrow ?$

$$x = \frac{64\times14.28}{44^k} = \frac{64}{44\times10^3}\times14.28\times10^3\ \mu\text{sec} = 20.7\ \mu\text{sec}$$

$$\% = \frac{y}{y+x}\times100 = \frac{1}{1+20.7}\times100 \approx \overset{4.5}{\cancel{05}} = \overset{5.0\%}{\cancel{10\%}}$$

A

Q. A hard disk with transfer rate of 10 Mbps is constantly transferring the data to memory using DMA. The processor runs at 600 MHz and it takes 300 & 900 cycles to initiate and complete DMA transfer. If the size of transfer is 20KB, what is the % of processor time consumed for transfer operation?

$$\cancel{100}\ 10 \times 10^6 \text{ bits} \longrightarrow 1 \times 10^{-6} \mu sec.$$

$$20 \times 10^6 \ '' \longrightarrow ?$$

$$x = \frac{1 \times 10^{-6} \times 20 \times 10^6}{10 \times 10^6}$$

$$x = 2000 \ \mu sec.$$

$$(\text{Time Period})_{clock} = \frac{1}{600\cancel{0} \times 10^6}$$

$$= \frac{1}{600} \ \mu sec.$$

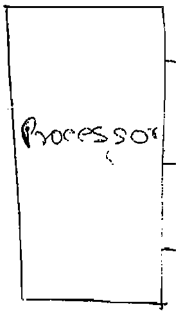
$$y = (300 + 900) \times \frac{1}{600} \ \mu sec$$

$$= 2 \mu sec.$$

$$\% = \frac{y}{y+x} \times 100 = \frac{2}{2000+2} \times 100$$

$$= 0.1\% \ \text{Ans.}$$

* I/O S



• An I

1) Periph

2) The

3) The d

memory.

4) The

• An I

1) Bufferi

2) Convert

3) Perfor

4) Impro

5) Capable

• An I/(

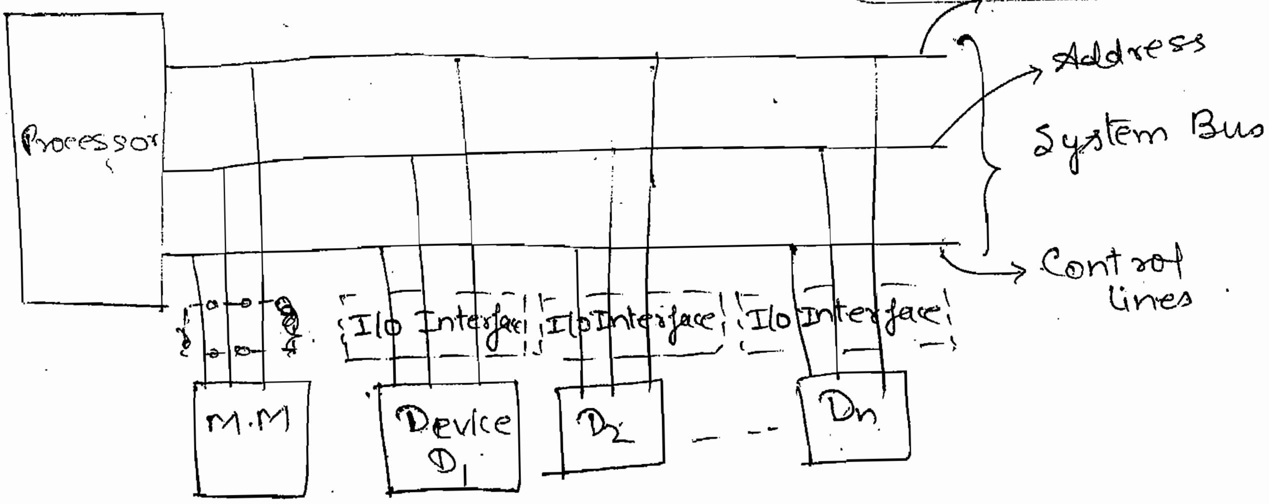
multiple

channel.

is consdan-
the
& 900
If the
processor

I/O System :-

Shekar's Victory
DATE PAGE



- An I/o interface is required because -

1) Peripherals are mechanical in nature.

2) The data transfer rate differs from CPU and memory

3) The data format and codes differs from CPU and memory.

4) The operating mode of each device varies from other.

- An I/o interface performs -

1) Buffering info.

2) Converts serial data to parallel and vice-versa,

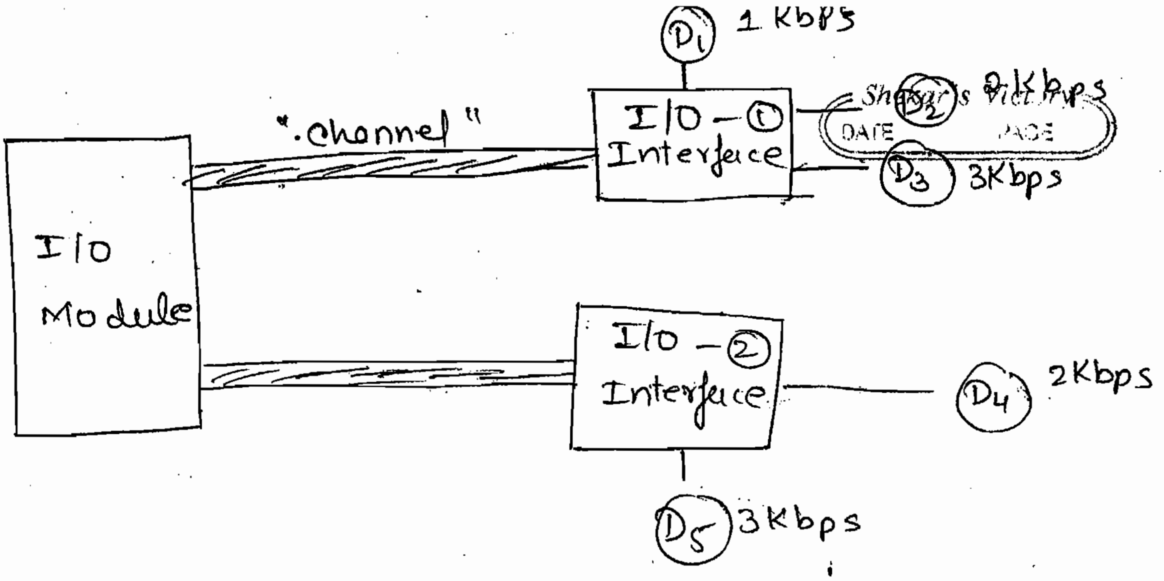
3) Performs error-control.

4) Improves data transfer rate.

5) Capable of executing I/O instructions, called I/O processor.

- An I/O-interface capable ~~to~~ manage several devices, multiple interfaces are connected to a module using channel.

Gold



I/O System

- A channel can be —

1) Selector Channel:

$$\text{Data transfer } (D_s) = \max_{i=1}^{n} \{D_i\}$$

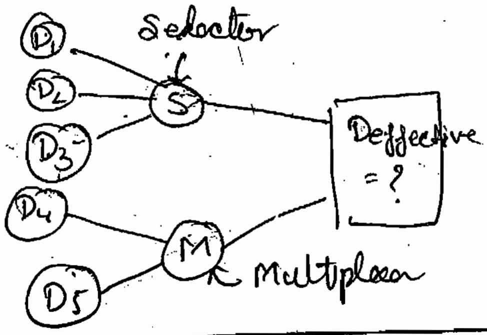
e.g., here $D_s = 3\text{Kbps}$

2) Multiplexer Channel:

$$D_m = \text{Sum}_{i=1}^{n} \{D_i\}$$

E.g. $D_m = 1+2+3+2+3 = 11 \text{ Kbps}$

Q.



$$\boxed{D_{effective} = \max_{i=1}^{3} \{D_i\} + \text{Sum}_{i=4}^{5} \{D_i\}}$$

Q- An
attached
of 16 K
10 line
effective

De

I/O Tr

Se

* In
genera

* In
asychro

* Synchro

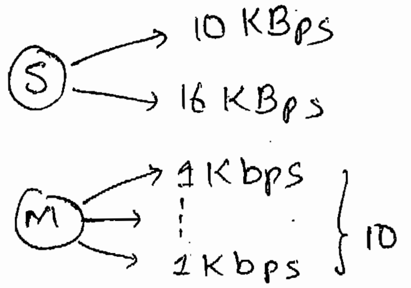
kbps

s

2Kbps

Q- An I/o System is consisting of a selector channel attached with a ~~tape~~ drive of 10 KBps and a disk drive of 16 KBps and a multiplexer channel attached with 10 line printers, each of 1 Kbps rate. What is the effective data transfer rate?

Shekar's Victory

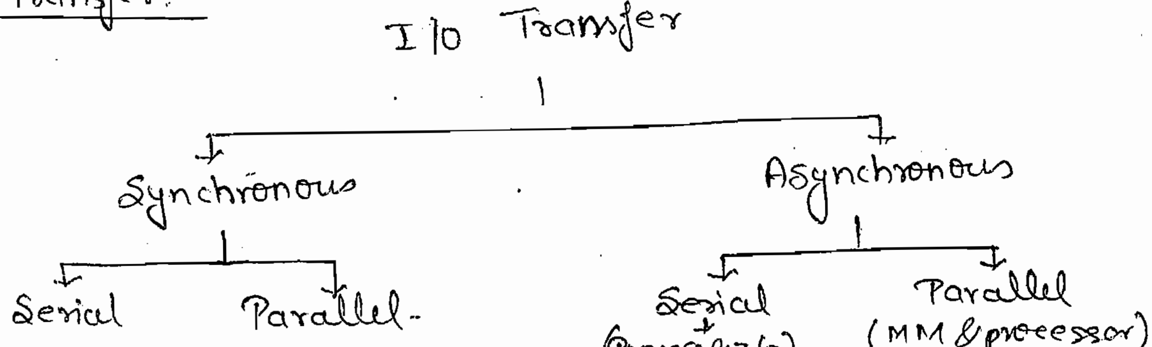


$$D_{eff.} = \max_{i=1}^{2} \{D_i\} + \sum_{i=3}^{10} \{D_i\}$$

$$= 16 \text{ KBps} + \frac{10 \times 1}{8} \text{ KBps}$$

$$= 17.25 \text{ KBps}$$

I/O Transfer:-



* In Computer N/w's transfer is serial synchronous generally.

* In Single Comp. System transfer is generally asychronous.

* Synchronous → Serial →

SYN	SYN	STX	DATA	ETX	CRC	SYN

Synchronization Characters

Must be excluded from data transfer ~~rate~~

* Asynchronous → Serial →

Start Bits	Parity Bits	Char	Stop / End Bits
0/1/2	0/1/2	8 bits	0/1/2

Shekar's Victory
DATE PAGE

Synchronization Bits ↓ Are included in data transfer rate.

- Data transfer rate

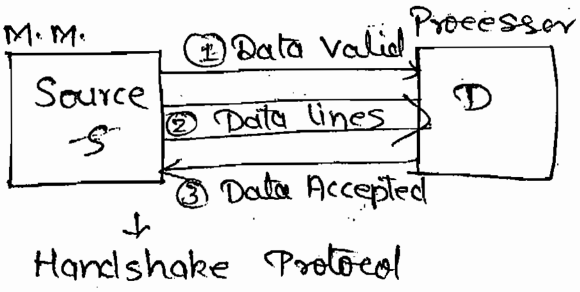
D = No. of character transfered per sec
= Chars/sec.

- Baud Rate / Bit Rate

$$\eta = \frac{\text{No. of Bits transferred}}{\text{Sec.}}$$

- The Baud Rate is the rate at which serial information is transferred.

* Asynchronous → Parallel →



Q- Assume each character code consist of 8 bits. the no. of characters, that can be transmitted per sec. through asynchronous serial lines with a baud rate of 2400 and two stop bits, is ____ ?

Baud Rate η = 2400 bits/sec

$$D = \frac{2400}{8+2} = 240 \text{ chars./sec.}$$

Q. In
parity
to susta

Q. A se
1 stop
synchror
followed
is 1200
rates of
A) 100
B) 80
C) 100
D) 80

Soln.

fer

nsfer rate.

nformation

ser

ts. the through 400 and

Q. In serial communication, employing 8 data bits, a parity bits and 2 stop bits. The min^m baud rate required to sustain a transfer rate of 300 char./sec. is —— ?

$$D = 300$$

$$\eta = 300 \times (8+1+2)$$
$$= 300 * 11 = 3300 \text{ bits/sec.}$$

Q- A serial transmission T_1 uses 8 info bits, 2 start bit 1 stop bits & 1 parity bit, for each character. A synchronous transmition, T_2 uses 3 8-bit sync. chars, followed by 30 8-bit info. char. If the bit rate is 1200 bits/sec. in both cases, what are the transfer rates of T_1 & T_2 ?

A) 100 ch/sec, 153 char/sec.

B) 80 " , 136 "

[✓ C)] 100 " , 135 "

D) 80 " , 153 "

Sol^n- $\eta = 1200$ bits/sec.

$$D_{T_1} = \frac{1200}{8+2+1+1} = 100 \text{ chars/sec.}$$

for 240 info bits ⟶ 24 bits sync. bits required

1200 ⟶ ?

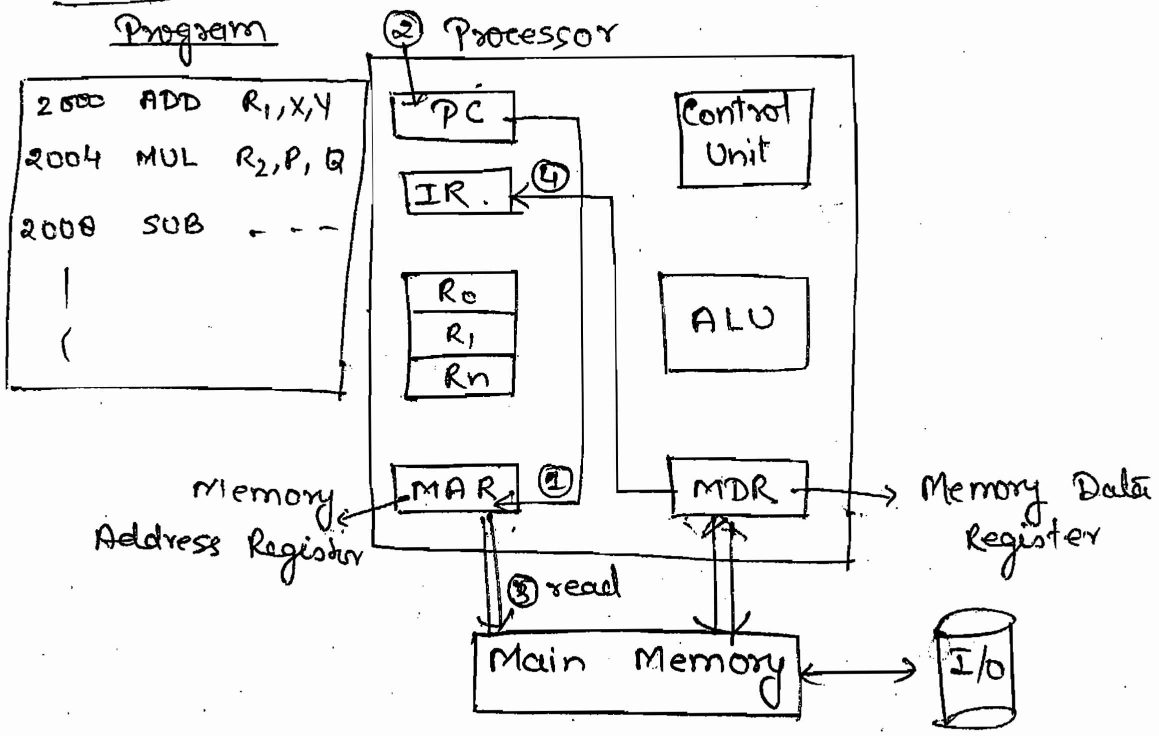
$$= \frac{1200 \times 24}{240} = 120 \text{ Sync. Bits}$$

$$T_2 = \frac{1200-120}{8} = 135 \text{ char/sec.}$$

MACHINE INSTRUCTIONS:

Shekar's Victory
DATE PAGE

Basic Operational Concept –



MAR (Memory Address Register):
It holds address of instructions or data to be read or written with memory.

MDR (Memory Data Register):
It holds data or instructions to be read or written with memory.

IR:
It holds current instruction code.

PC:
It holds address of instruction to be executed.

Instruction cycle –

1. Instruction Fetch (IF)
   a) MAR ⟵ PC   (e.g. MAR ⟵ 2000)
   b) PC ⟵ PC+1   (e.g. next Instruction → 2004)
   c) Read (MR)

Shekar's Victory
DATE PAGE

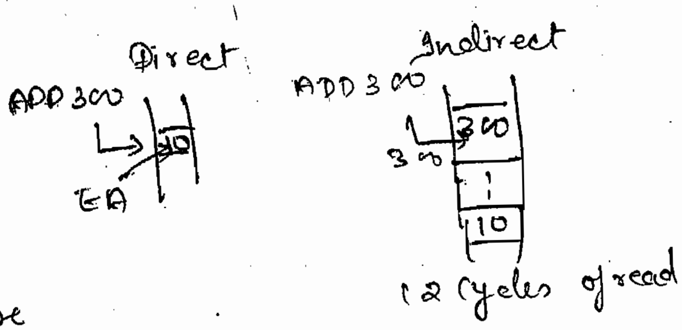
d) IR ← MDR (Part of MDR)

2. Instruction Decode (ID)
IR to Control Unit

3. Operand Fetch (O F)

4. Execution & Store (Ex)
↳ Write-back (final value will be written back)

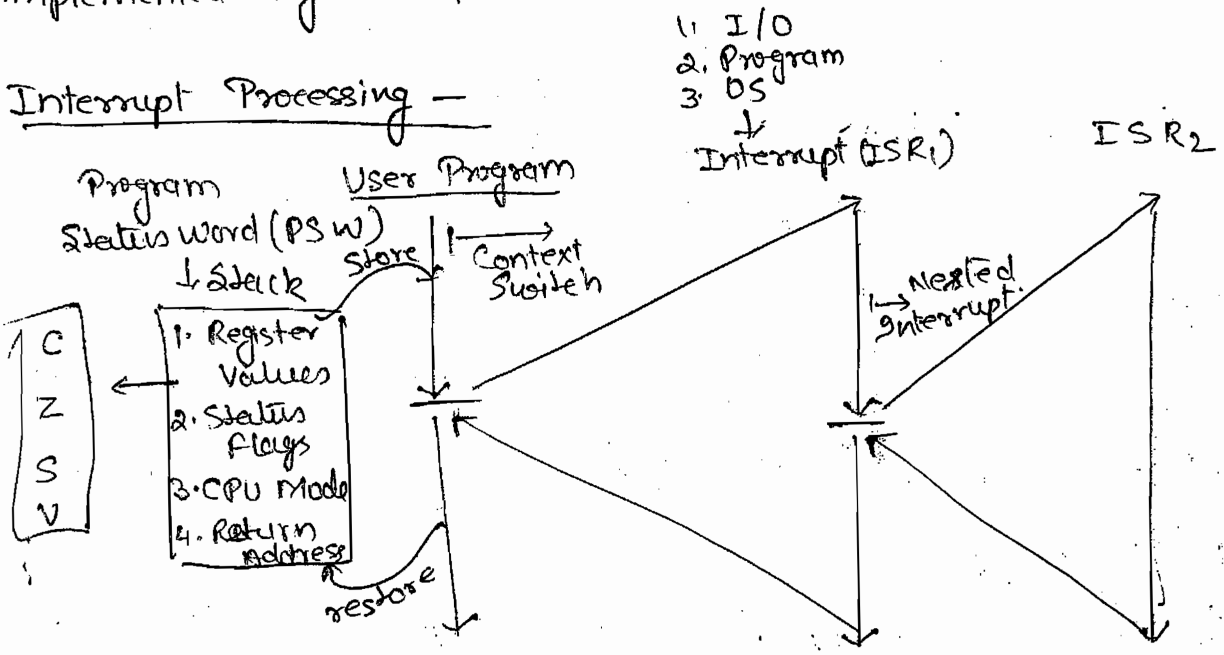
5. Instruction Cycle
If Indirect addl. mode is used.



6. Interrupt Cycle / Phase

- Register transfers are micro operations, each instruction execution involves a sequence of micro operations to be implemented by the processor.

Interrupt Processing –



Gold

No. of PSWs = No. of Interrupts processed

Shekar's Victory
DATE PAGE

- In general the CPU is operated in two modes—

1. System | Supervisory | Priviliged Mode:
   Executes operating system programs to obtain system services.

2. User / Non-Priviliged Mode:
   Executes user application.

- The interrupt can be —

1. External / H/w Interrupts:
   ~~Ar~~ Raised due to timing & I/O devices.

2. Internal Interrupts:
   Raised due to erroneous use of instructions and data.
   e.g. * Invalid Opcode
   * Register Overflow
   * Division by Zero.

3. Software Interrupts:
   Arise due to switching from user mode to system or vice-versa.

Q- A processor needs software Interrupt to ______.

A) Test the interrupt system.

B) To implement Co-routines.

[✓ C) To Obtain System Services.]

D) To return from subroutine.

Q. A
In orde

Q. A CP
an inte
A) As
B) By
of fetch
[✓ C) By]
executio
D) By

PC

Instr

Q. A CPU has two modes, Priviliged and Non-priviliged In order to change from one to another required Software interrupt

Sarkar's Victory DATE PAGE

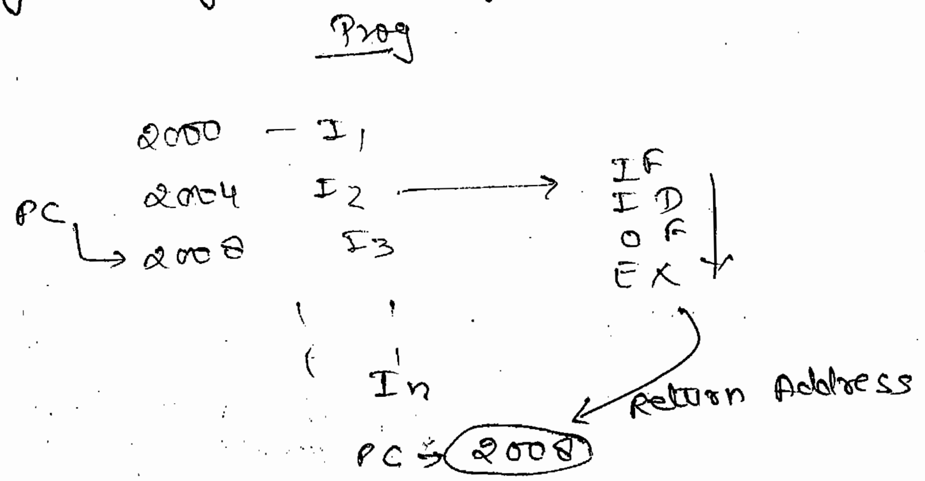
Q. A CPU generally handles an interrupt by executing an interrupt service routine ____.

A) As soon as an interrupt is Raised.

B) By checking the interrupt register, at the end of fetch cycle. B

✓ C) By checking the int. register, after finishing the execution of current instruction.

D) By checking the int. register, at fixed time interval.



Instruction Formats –

Opcode	Operand Address Part

Opcode	Mode	Address Part of Operand

Mode → ~~Adding~~ Addressing Mode

Gold

Instruction can be

1.

Opcode	A_1	A_2	A_3	A_4

(Registes / Memory) Operands ← → Next inst'n address

* PC not required

4 – addr. Inst'n

2.

Opcode	A_1	A_2	A_3

← Operands → (Register / Memory)

* PC is required.

3 – addr. Inst'n
↓
CISC inst'n

- Requires more than one word refferred from memory.

3.

Op code	A_1	A_2

← → (Register / Memory) operands

2 – addr. Inst'n

3/4.

Op code	A_1

← → Memory reference only of operand

1 – addr. Inst'n
↓
~~A~~ 1 operand must be implicitly accumulator.
* Simple & easily decoded.

5.

Opcode

0 – addr. Inst'n
↓
Both the operands must be from top of the stack.

E.g. ADD

6.

Opcode	A_1	A_2	A_3	– – –	A_n

← Registers only. →

RISC – Inst'n.
↓
Provides faster execution.

Problems
Q. Consi
having the
after the

MOV
LOA
AD
ST

A) M[10
D)

Method

1. R_1, R_2
2. X, Y,
3. MOV
Ir
4. R_i
5. (R_i)
6. 100 (R
↓
Displace or offset

Problems Related to Inst'n Set:

Shekar's Victory DATE PAGE

Q. Consider the memory locations 1000, 1001 & 1020 are having the values 18, 1 & 16. Identify the correct statement, after the following program is executed.

1000	1001	1020
↓	↓	↓
18	1	16

		sol^n:	
MOVI	R_s , 1	$R_s \leftarrow 1$	$R_s = 1$
LOAD	R_d, 1000 (R_s)	$R_d \leftarrow M[(R_s)+1000]$	$R_d = M[1+1000]$ $= M[1001]$ $= 1$
ADD I	R_d, 1000	$R_d \leftarrow R_d + 1000$	$R_d = 1 + 1000$ $= 1001$
STOR I	$0(R_d)$, 20	$M[(R_d)+0] \leftarrow 20$	$M[0+1001] = 20$ $M[1001] = 20$

A) $M[1000] = 20$ B) $M[1020] = 20$ C) $M[1021] = 20$

~~D)~~ $M[1001] = 20$

Method of Solving Such Probs –

1. $R_1, R_2, R_3, - - R_n$ → Register References
2. X, Y, A, - - - , 3000 , 400 ⟶ Memory References
3. MOV I R_s , 1 ↳ Value
   ↓
   Immediate
4. $R_i \longrightarrow R_i$ Operand ↳ Value (Register Mode)
5. $(R_i) \rightarrow R_i$ ↳ Memory Address $M[(R_i)]$ (Register Indirect)
6. $100\ (R_i) \longrightarrow M[(R_i)+100]$ (Scaled Register Indirect)
   ↓
   Displacement
   or
   offset

Gold

Shekar's Victory DATE PAGE

Q- Let the size of four instr are given as 2 & 4 words. For all the instruction fetch will ~~take 2 clocks~~ 2 clocks/words and the execution of memory related instructions consumes 4 clocks & all others takes 1 clock. For the following program segment —

Inst'n		Operation
MOVI	R_s, 1	$R_s \leftarrow 1$
LOAD	R_d, 1000(R_s)	$R_d \leftarrow M[(R_s)+1000]$
ADDI	R_d, 1000	$R_d \leftarrow R_d + 1000$
~~ST~~ TORI	0(R_d), 20	$M[(R_d)+0] \leftarrow 20$

1) The no. of clock cycles required to complete the above program.

F → 2
Ex → M — 4
Others → 1

Inst'n	Operation	Size	IF + Ex
I_1		2	2×2+ 1 = 5
I_2	Memory based	4	4×2 + 4 = 12
I_3		2	2×2 +1 = 5
I_4	Memory based	4	4×2+4 = 12
			34 Clocks

2) Let the size of a word is 64 bits. The no. of bytes required to store above program.

Total no. of words = 2+4 +2+4 = 12

Total no. of bytes = $\frac{12 \times 64}{8}$

= 96 bytes

3) Assu
in decim
return (
occurs (

I_1

I

Inst'n
I_1
I_2
I_3
I_4

4) Let th
is the re
interrupt
Inst'n n
I_5

* Dur

* No inte

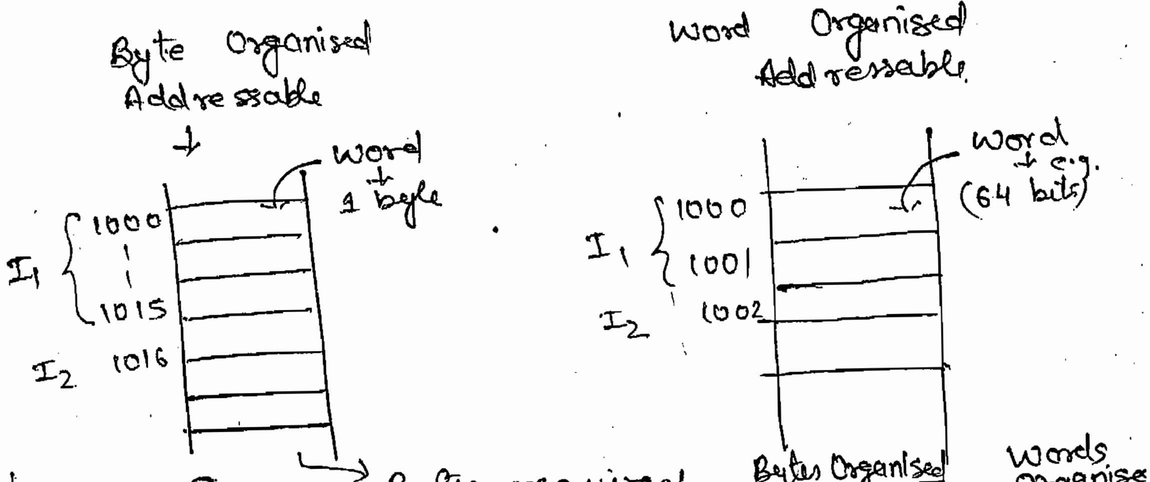
HN = 2 & 4
GE
ocks/words
consumes
following

the

5

2
clocks
bytes

3) Assume the program is loaded from location 1000 in decimal, in a byte organised memory. What is the return address pushed on to stack, when an interrupt occurs during ADD I instruction.



Instr'n	Size	Bytes required	Bytes Organised	Words Organised
I_1	2	$\frac{2 \times 64}{8} = 16$	1000 - 1015	1000 - 100
I_2	4	$\frac{4 \times 64}{8} = 32$	1016 - 1047	1002 - 100
I_3	2	$\frac{2 \times 64}{8} = 16$	1048 - 1063	1006 - 1007
I_4	4	$\frac{4 \times 64}{8} = 32$	1064 - 1095	1008 - 101

Return Address

4) Let the last inst'n in program is HALT (1 word). What is the return address saved onto stack, if there is an interrupt during HALT instruction.

Instr'n	Size	Bytes Req.	Bytes Organised	word Organised
I_5	1	$\frac{1 \times 64}{8} = 8$	1096 - 1103	1012 ~~1013~~

Return add.

* During HALT the return address is the same instruction

* No interrupt will arise during HALT instruction.

Gold

Shekar's Victory
DATE PAGE

Q1. Consider the following program segment for a CPU having 3 registers.

Inst'n		Operation	(words) Inst'n Size
MOV	R_1,5000	$R_1 \leftarrow M[5000]$	2
MOV	R_2, (R_1)	$R_2 \leftarrow M[(R_1)]$	1
ADD	R_2,R_3	$R_2 \leftarrow R_2 + R_3$	1
MOV	6000, R_2	$M[6000] \leftarrow R_2$	2
HALT		Stop	1

1) Consider the memory is byte addressable with a size of 32 bits and the program has been loaded starting from location 1000. If an interrupt occurs, while the CPU has been haulted, after executing HALT instruction. The return address saved in the stack will be — .

Inst'n	Size	Bytes Req.	Byte Addressable
I_1	2	$\frac{2\times32}{8} = 8$	1000 – 1007
I_2	1	$\frac{1\times32}{8} = 4$	1008 – 1011
I_3	1	$\frac{1\times32}{8} = 4$	1012 – 1015
I_4	2	$\frac{2\times32}{8} = 8$	1016 – 1023
I_5	1	$\frac{1\times32}{8} = 4$	1024 – 1027

1024 ↓ Return Address

2) Let the clock cycles required for various operations are —

(i) Register to/from Memory transfer — 3 Clocks

(ii) ~~Add~~ ADD in both operands in Registers → 1 Clock

(iii) Inst'n fetch & Decod — 2 ~~6ot~~ clocks/word

The no. of clock cycles required to complete the program —

Inst'n

I_1

I_2

I_3

I_4

I_5

Q.3 Consi

Inst'n

MOV R_1, 3

LOOP:

MOV R_2,(

ADD R_2,

MOV $(R_3$

INC R

DEC R

BNZ LO

HALT

Assum

3000 is

1000.

1) Assu

of memo

executing

A) 10

CPU

a

;, while

nstruction.

—

Addressable

1007

1011

-1015

-1023

-1027

ions are-

R

te the

Inst'n	~~Size~~ Operation	Size	F&D + Ex ~~ExA [illegible] F~~
I_1	Memory	2	2x2 + 3 = 7
I_2	Memory	1	1x2 + 3 = 5
I_3		1	1x2+1 = 3
I_4	Memory	2	2x2 + 3 = 7
I_5		1	1x2 + — = 2
			24

Shekar's Victory
DATE PAGE

Q.~~2~~ Consider the following program segment —

Inst'n	Operation	(words) Inst'n Size
MOV R_1, 3000	$R_1 \leftarrow M[3000]$	2
LOOP:		
MOV R_2, (R_3)	$R_2 \leftarrow M[(R_3)]$	1
ADD R_2, R_1	$R_2 \leftarrow R_1 + R_2$	1
MOV $(R_3), R_2$	$M[(R_3)] \leftarrow R_2$	1
INC R_3	$R_3 \leftarrow R_3 + 1$	1
DEC R_1	$R_1 \leftarrow R_1 - 1$	1
BNZ LOOP;	→ Branch if not zero	2
HALT		

Assume that the ~~const~~ content of memory location 3000 is 10, and it is loaded from memory location 1000.

1) Assume the memory is word addressable, the no. of memory references for ~~ex~~ accessing the data is executing the program completely is —?

A) 10 B) 11 C) 20 [✓ D) 21]

Gold

Shekar's Victory
DATE PAGE

Instn	Operation	Size	Bytes Required	Word Addressable
I_1	Memory	2		1000 - 01
I_2				1002
I_3	Memory	1		1003
I_4		1		1004
I_5	Memory	1		1005
I_6		1		1006
I_7		1		1007 - 08
I_8		2		
I_9				

given $M[3000] = 10$

$\boxed{R_1 = 10}$

* All Branch operations will work on Last ALU operation. result.

$I_3 \rightarrow 10 \times 1$

$I_5 \rightarrow 10 \times 1$

$I_1 \rightarrow \frac{1}{21}$

* Everytime value of R_1 will decrement by 1, 10 times.

2). Let the size of a word is 32 bit, if the interrupt occurs during execution of INC ~~==~~ R_3. What written address push on to the stack?

~~A)~~ 1005 B) 1020 C) 1024 D) 1040

Actual Return add. → 1006

Addres

• Th

execution

• Addr

1) Poin

Program

2) Less

3) Provi

Ins

The c

1) Imp

of instr

Eg:

2) Imme

Eg:

word Addressable
1000 - 01
002
003
004
005
006
07 - 08

Addressing Modes (AM):

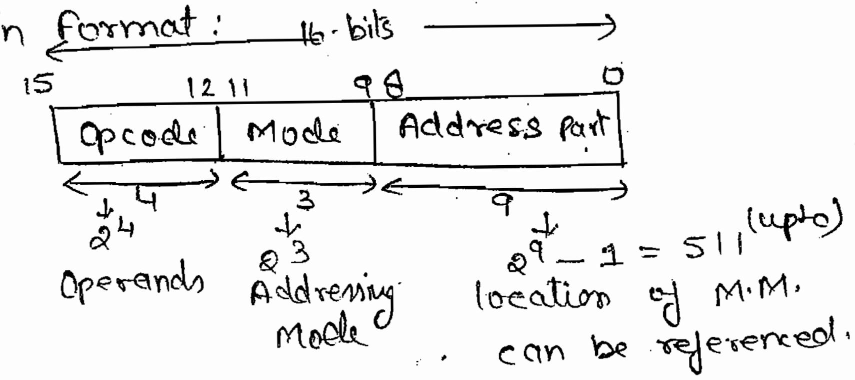
Operand	Address

→ Address
→ value
→ Register (directly / Indirectly)

- The way the operand is choosen during program execution depends on addressing mode.
- Addressing modes (AM) provides —

1) Pointer to Memory, Indexing a table, Controlling a loop program realloeation etc.
2) Less no. of bits in address part of operand.
3) Provides faster execution.

Inst'n format: 16-bits



2^4 Operands

2^3 Addressing Mode

$2^9 - 1 = 511$ (upto) location of M.M. can be referenced.

The addressing modes can be —

5 times.

1) Implied Mode:
The operand location is knaon from the definition of instruction itself.

Eg: 1. Complement Accumulator (CMA)
2. All zero Address Instructions (ADD, SUB, etc.)

Interrupt item

40

2) Immediate mode:
The operand value is from the instruction itself.

Eg:
MOV I $R_1, 10$
~~Ga~~ADI $R_2, 20$

- Relatively faster than all other modes.
- Used to initialize registers to a constant value.
- The range of values initialised is limited by address part. E.g. Max^m: 511 values initialised.

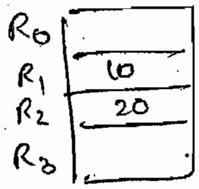
Shekar's Victory DATE PAGE

3) Register Mode:

The operands reside in CPU registers.

e.g. ADD R_1, R_2

$\hookrightarrow R_1 \leftarrow R_1 + R_2$



- Faster than memory addressing.
- Less no. of bits in address field.

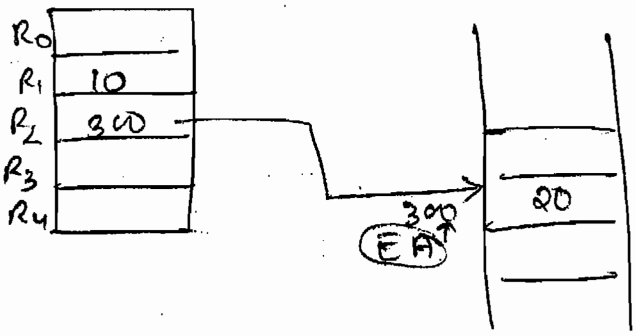
$\underline{16} = 2^{\boxed{4}} \rightarrow$ 4 bits required

4) Register Indirect Mode:

Address part specifies a register which contains effective address of an operand.

ADD $R_1, (R_2)$

$R_1 \leftarrow R_1 + M[(R_2)]$



5) Auto Increament / Decreament Mode:

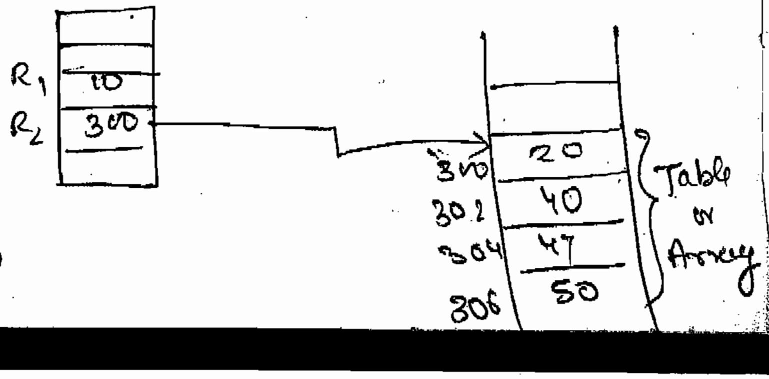
- It is similar to register indirect, except that the contents of register is incremented or decreamented after the value at location is accessed.

ADD $R_1, (R_2)$

$R_1 \leftarrow R_1 + M[(R_2)]$

0 to <n

$\boxed{x = x + A[i]}$;



- Impler
- Allows
- The

6) Absolu

- effective

- Use
- Used

Eg

- Refe address

7) Indir

- Addre address c
- Allows two me

2.)

y address

contains

that
eamented

0

0
0
1
0
Table or Array

- Implements loop control mechanism.
- Allows to process complete table or array

Shekar's Victory
DATE PAGE

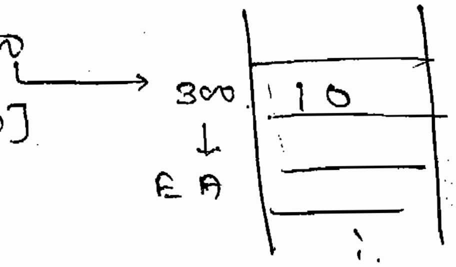
- The size of INC or DEC depends on size of eleme

6) Absolute or Direct Addressing :

- The address part specifies the ~~operand value~~ effective address of an operand.

MOV R_1, 300

$R_1 \leftarrow M[300]$



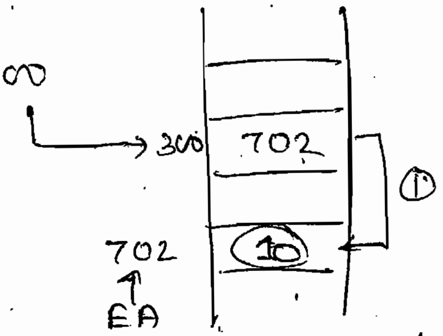
- Used to declare global variables in a program.
- Used for branch instruction.

Eg. BNZ 302

- Refers very small address space, limited by address part.

7) Indirect Addressing Mode:

MOV R_1, 300



- Address part specifies a location, contains effective address of an operand.
- Allows to refer large address space, but requires two memory cycles.

Gold

8) Displacement Addressing Mode:

Shekar's Victory
DATE PAGE

E A = Address Part + ~~Register~~ Value

Effective Addr. of Operand

↓

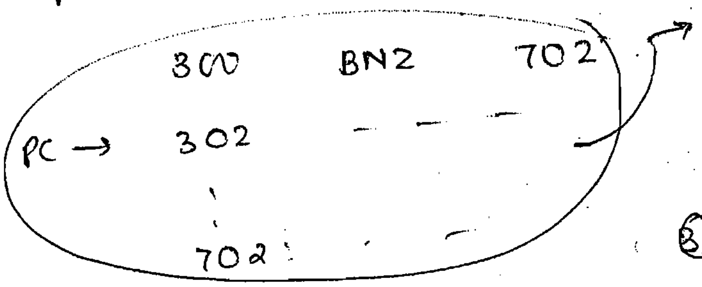
(i) (if) Address + (then) Offset / Displacement

(ii) (if) Offset + (then) Address

a) Relative Addr. Mode:

$EA = \text{offset} + \underset{(\text{address})}{PC}$

Only offset part is needed to provide, PC will automatically have the address so less no. of bits will require.



① Direct addr. ✗

② Indirect add ✓
↓
2 memory cycle require

③ Relative addr.

$EA = 400 + \overset{(PC)}{302} = 702$

300 BNZ 400

↓

① 1 memory reference

② 1 ALU opratio

- fewer bits in address part.
- Relatively faster than ~~address~~ memory addressing.
- Provides program rellocation.

b) Indexed Addressing Mode!

$EA = \underset{\text{of Array}}{\text{Base address}} + \underset{(\text{Index Register})}{\text{Offset}}$

c) Base Register Addressing:

$EA = \text{Offset} + \underset{(\text{Base Register})}{\text{Base address of Segment}}$

• Allo
• Use

Q. ⓔ
a) I
b)
c) A

Q. (i)
(ii)
(iii)

Q.
(i) B
(ii)
(iii) S
(iv) I

- Allows to implement position independent code.
- Used to implement records or structures

Shekar's Victory
DATE PAGE

Q. Match the following –

a) In direct	(i) loops (c)
b) Immediate	(ii) pointers (a)
c) Auto Decreoment	(iii) Constants (b)

Q.

(i) Indireect	a) Array (ii)
(ii) Indexed	b) Relocalable code (iii)
(iii) Base Register	c) Passing array as peremeter (i)

Q.

(i) Base addressing	a) Reentrancy (iii)
(ii) Indexed Addr.	b) Accumulator (iv)
(iii) Stack	c) Array (ii)
(iv) Implied	d) Position Independent (i)

Value

lacement

will

ails will

X

✓

cycle requin

= 702

memory reference

ALU operati

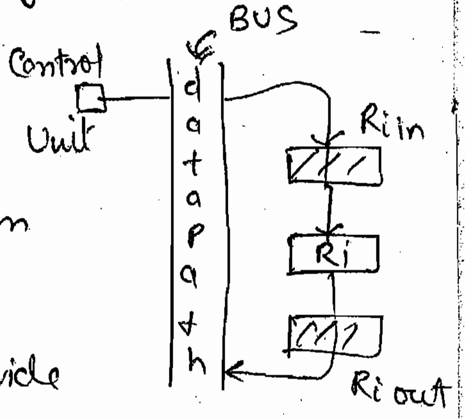
ssing.

Gold

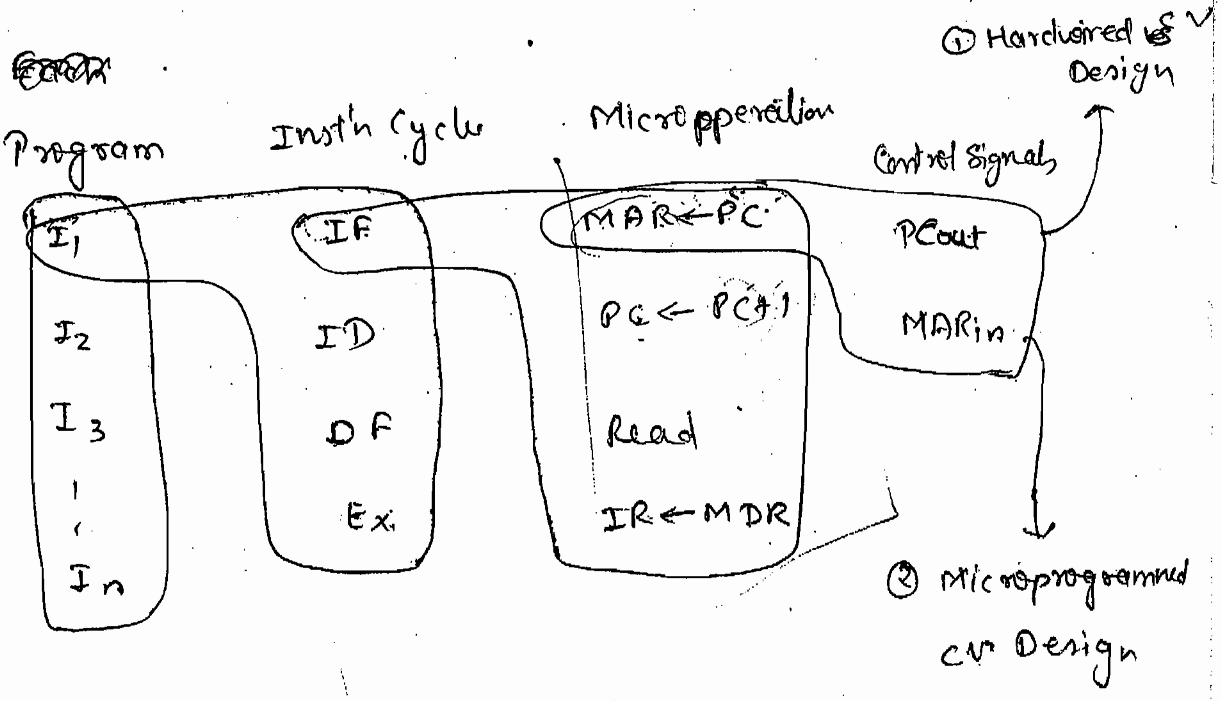
Shekar's Victory
DATE 27/11 PAGE

Control Unit Design:-

In processor some general purpose & special purpose registers are available. All registers are connected to a common path called data path (BUS). Every Register has a switch Rin & Riout.



- If Riin is set to '1', the contents of the bus loaded into Ri.
- If Riout is set to '1', the contents from Ri will be placed on bus.
- The purpose of Control unit is to provide appropriate Timing and control signals.



Hardwired Control Unit Design:

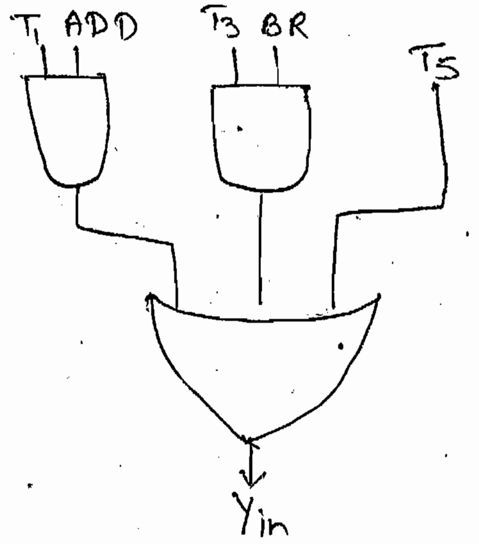
- The control unit uses fixed logic circuits to interpret instructions and generate control signals from there.
- Every control signal is expressed as SOP (Sum of products) expression and realised using digital h/w.

Ex: $Y_{in} = T_1 \cdot ADD + T_3 \cdot BR + T_5$ → will enable for all instructions,

Here Yin is enable during T_1 for ADD instruction, during T_3 for branch instruction, during T_5 for all instructions and so on.

- It is realised using digital hardware as

Y_{in} ⇒



↳ Control functions are implemented in h/w.

↳ Relatively control signals are generated fast than microprogram.

↳ Allows to enable simaltaneously control signals.

↳ Relatively less flexible because any changes in microoperations ~~are~~ or control signals requires re-design or re-wiring.

↳ Relatively less flexible for large no. of ~~in~~ control signals & instructions.

↳ so, Implemented ~~of~~ in RISC operation.

Q- Consider the following hypothetical machine which uses 3 data registers A, B & C and supports 3 inst's I_1, I_2 & I_3. Obtain the logic function that will generate the hardwired control for the signal Ain & Bout with the following data.

	I_1	I_2	I_3
T_1	Ain, Bout	Ain, Cin, Bout	Bin, Bout
T_2	Bin, Cin, Aout	Ain, Aout	Ain, Bin, Cout
T_3	Bin, Bout	Bin, Bout	Bin, Bout
T_4	Cin, Aout	Bin, Aout	Ain, Aout
T_5	End	End	End

A-

~~First~~ Step 1: Search where the control signals Ain & Bout are present.

Step 2: Options are in I.T format or T.I format.

Step 3: For any particular time interval, Is the control signal presents for all the instruction?

$$A_{in} = T_1 I_1 + (T_1 + T_2) I_2 + (T_2 + T_4) I_3$$

$$B_{out} = T_1 + T_3$$

$\downarrow$

Bout is present for all instn's during T_1 & T_3.

which uses
PAGE
I_1, I_2 & I_3-
the
with the

signals

or

Is the

Q- A hardwired CPU uses 10 control signals S_1, -- S_{10}, in various time ~~steps~~ stamp T_1 to T_5 to implement 4 instn's I_1 to I_4 as follows

Shekar's Victory DATE PAGE

	T_1	T_2	T_3	T_4	T_5
I_1	S_1, S_3, S_5	S_2, S_4, S_6	S_1, S_7	S_{10}	S_3, S_8
I_2	S_1, S_3, S_5	S_8, S_9, S_{10}	S_5, S_6, S_7	S_6	S_{10}
I_3	S_1, S_3, S_5	S_7, S_8, S_{10}	S_2, S_6, S_9	S_{10}	S_1, S_3
I_4	S_1, S_3, S_5	S_2, S_6, S_7	S_5, S_{10}	S_6, S_9	S_{10}

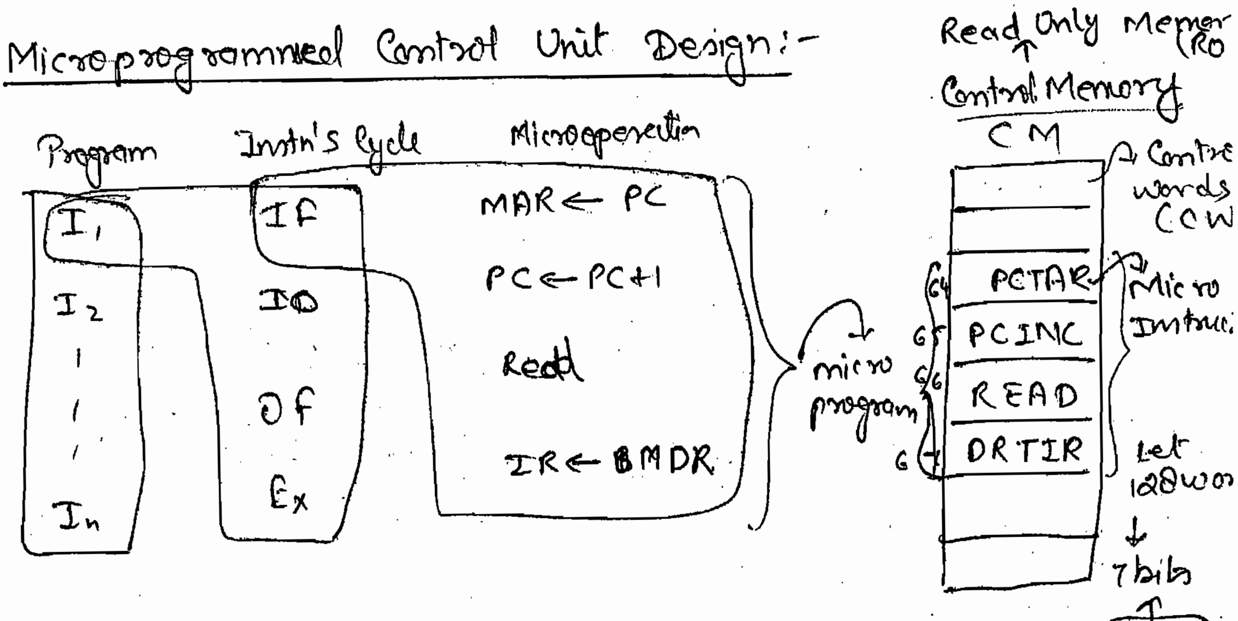
Obtain the logic function to generate S_5 and S_{10}.

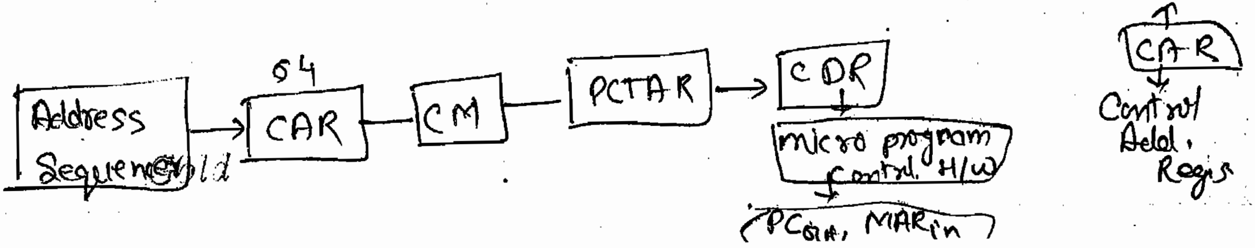
~~$S_5 = T_1 + I_2 T_3 + I_4 T$~~

$$S_5 = T_1 + (I_2 + I_4) T_3$$

$$S_{10} = (I_2 + I_3) T_2 + I_4 T_3 + (I_1 + I_3) T_4 + \cancel{I_2 T_5} \; (I_2 + I_4) T_5$$

Microprogrammed Control Unit Design:-





Shekar's Victory
DATE PAGE

↳ Control functions are implemented in S/w.

↳ Relatively control signals are generated slow.

↳ Each micro program consists of a set of micro instructions.

3↳ Each micro instruction is completed in one clock cycle.

↳ Relatively flexible because any changes in micro operation sequence needs to change only in Control Memory (CM - ROM).

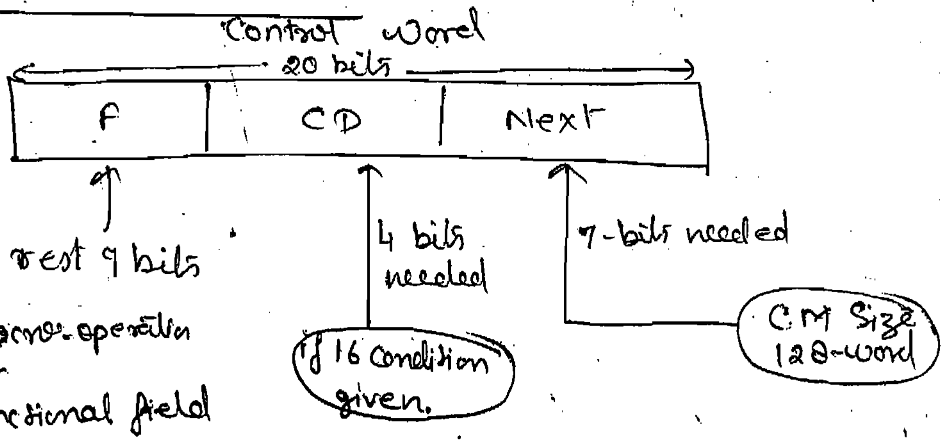
↳ Effective for large no. of control signals & instructions.

↳ So, Implemented in CISC processors.

↳ The address sequencer will update next CAR by -

1) Increamenting CAR
2) Conditional or Unconditional Brenching.
3) By Micro program routine call.
4) Mapping Opcode bits to control memory address.

3/ Micro Instruction Format:



F: Micro-operation or Functional field

CD: Conditional field

Next: Next micro inst'n Addr. field

Q. Th
processor
is devid
'Next' ad
field (
μPC,
the sig

• The

Hori

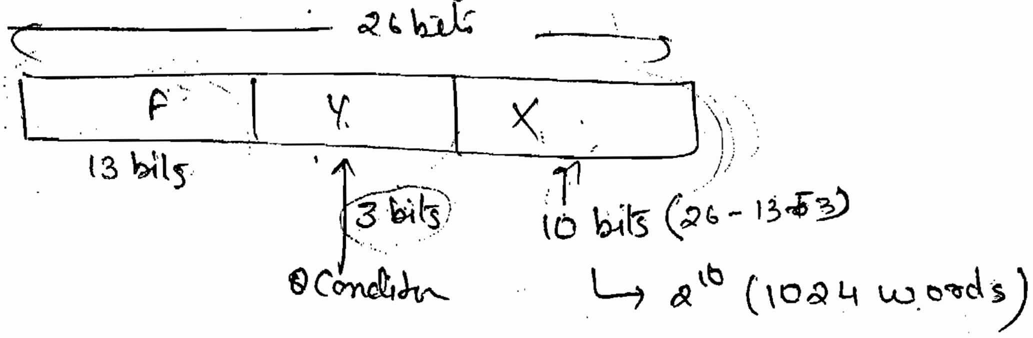
1) 1
each bi
control
e.g. 9

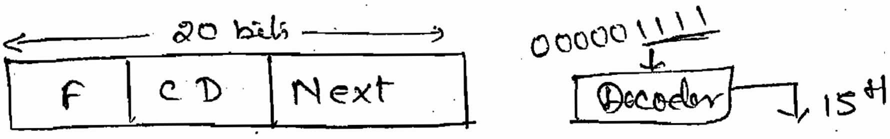
2) No addit

3) Relative
feslly.
4) Max^m.
E.g. 9

Shekar's Victory
DATE PAGE

Q. The micro instructions stored in ~~control memory~~ of a processor have a width of 26 bits. Each micro instruction is devided into 3 fields. A micro instruction field (F) of 13 bits, 'Next' address field ~~(Next)~~ (X) and a 'conditional' select field (Y). Let the processor has 8 condition ~~from~~ for µPC, how many bits in X, Y field and what is the size of control memory in word?





- The µPC can be—

Horizontal µPC	Vertical µPC
1) ~~One~~ 1 bit/Control Signal i.e. each bit will required for one control signal. e.g: 9 Control Signals	1) The control signal ~~are~~ encoded for k-bits as 2^k signals. e.g. 2^9 → 512 signals. (all bits will represent 1 control signal)
2) No additional H/w required	2) A decoder circuit is required
3) Relatively control signal are generated fastly.	3) Relatively Slow.
4) Max^m. degree of parallelism E.g. 9	4) Max^m. degree of parallelism is always 1.

5) For large no. of instruction and control signals, CW & CM is large.

) Average access time to enable control signals,

$$\boxed{T_{avg.} = T_{CM} + T_{\mu PC\text{-}H/w}}$$

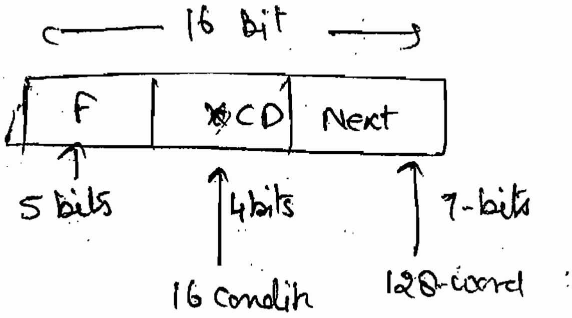
5) CW & CM is small

Shekar's Victory
DATE PAGE

6) The average time to enable control signal

$$\boxed{T_{avg.} = T_{CM} + T_{decoder} + T_{\mu PC\text{-}H/w}}$$

Q. A 16-bit microinstruction supports, 16 conditions and stored in 128 word CM. What ~~ws~~ will the no. of control signal generate in HμPC & VμPC.

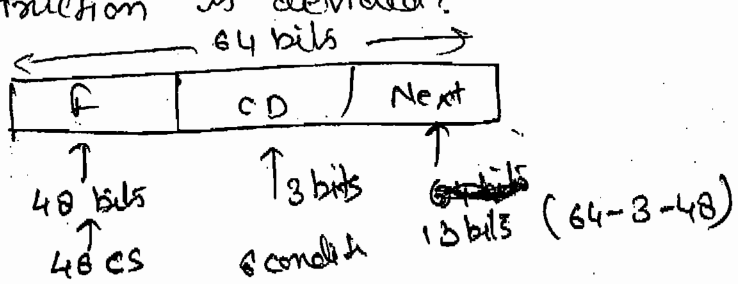


- For HμPC, with 5 bits, can represent 5 control signal
- for VμPC, with 5 bits can represen $2^5 = 32$ Control Signals

Q- Consider a control unit design in which 48 control signals are to be generated and the system is supporting 8-Flag conditions. If the ~~co~~ 64-bit control word is stored in CM, then ~~for~~

① for HμPC

(Q-1) How the instruction is devided?



(ii) Wha

S

(iii) Mo

② for

(i) for

size

(ii) what

(iii) Max

(ii) what is the size of control memory in bytes ?

2^{13} → words

Each words is having 64 bits.

So $\frac{2^{13} \times 64}{8}$ Bytes.

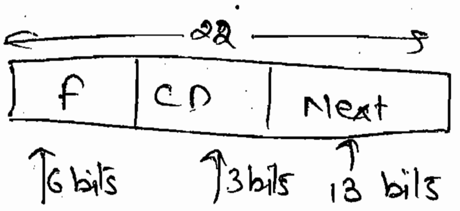
= 64 KB

(iii) Max^{m}. degree of parallelism is 48.

② for VμPC

(i) for same no. of locations as HμPC, what is the size of control word in VμPC.

No. of locations → 2^{13}



48 → 64
2^{c} ← 2^{6}

48 Signals 8 Condition

6 + 3 + 13 = 22 bits

(ii) what is the ~~signal~~ size of ~~an~~ CM in bytes ?

$\frac{2^{13} \times 22}{8}$ bytes = 22 K B

(iii) Max^{m}. degree of parallelism → 1.

{To overcome the shortcomings of HμPC & VμPC.

Soft-VμPC:

Shekar's Victory
DATE PAGE

F	CD	Next
9	4	7

2	3	4		
F_1	F_2	F_3	CD	Next

28 Control Signals = 2^2 2^3 2^4

3 Control Signal = 1 1 1
(3 degree of parallelism)

Max^m. degree of parallelism = 3

↳ Here, Control Signals are devided into groups of mutually exclusive signals, each group enables 1 control signal simulltaneously.

Q. A ~~a~~ μ-inst'n supports 5 mutualley exclusive group of control signal.

G_1 : 20 G_2 : 10 G_3 : 2 G_4 : 42 G_5 : 17 CS

(i) How many bits are saved using VμPC over HμPC.

G_1^{20}	G_2^{10}	G_3^{2}	G_4^{42}	G_5^{17}	CD	Next
21 bits = 5 bits	4	1	6	5		

HμPC = 20+10+2+42+17
= 91 bits

VμPC = 21 bits

No. of bits saved = 91 − 21
= 70 bits

(ii) Ma

Q- Con

7 clock

There a

Control .

1 - addr

& CAR

A- E

7 c

we stu

11

TH

S

(ii) Max^{m}. degree of parallelism = ~~5~~ 5 Groups

Q- Consider a CPU where all instructions take 7 clock cycles to complete the execution of each. There are 140 instructions in inst'n set and 125 Control signals are ~~mee~~ needed to be generated using 1-address HμPC. What is the min^{m}. size of CW & CAR?

A- ⊛ Control Signals → 125
So, No. of bits in F is → 125 bits

7 clock ~~clo~~ cycle is needed for each inst'n. But, we studied that 1 μinst'n needed 1 clock cycle. So,

1 inst'n ⟶ 7 cycle
i.e. 7 μinst'n or CW

140 inst'n ⟶ 140 × 7 CW's
= 980 CW's
⇓
≃ 10 bits, So CAR = 10

*

F	Next
125 bits	10 bits

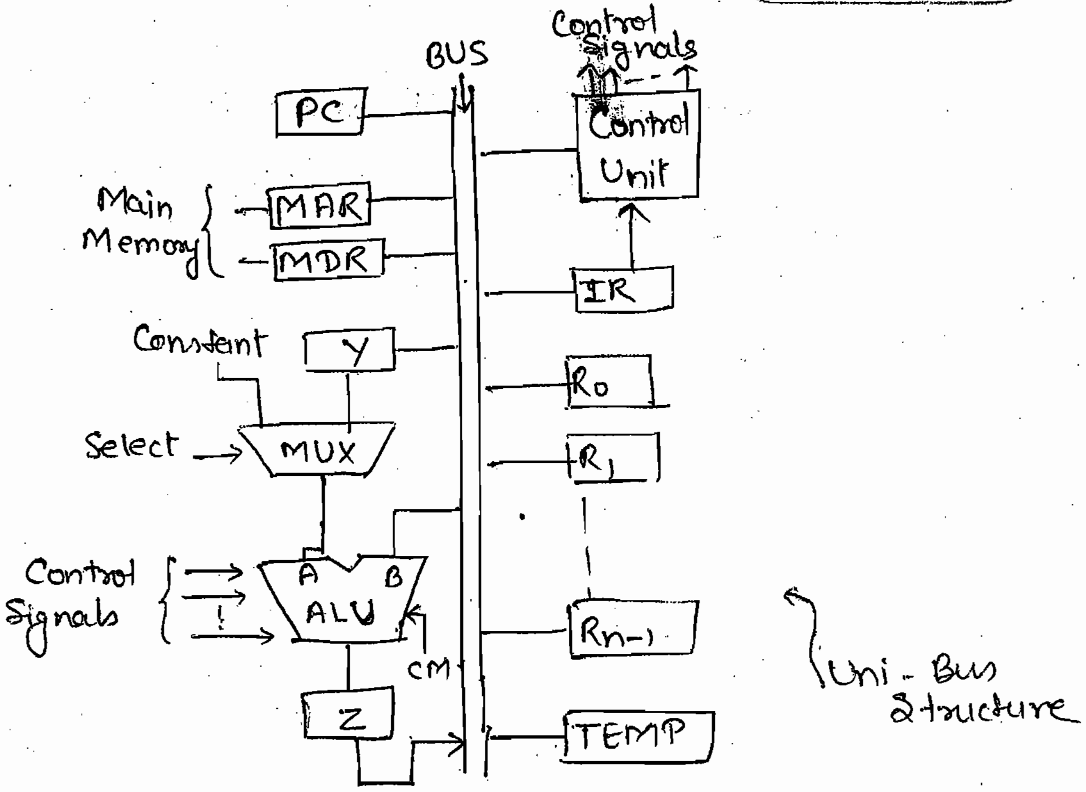
CW = 125+10 = $\underline{\underline{135}}$ bits.

$T_{H/W} < T_{H\mu PC} < T_{SoftV\mu PC} < T_{V\mu PC}$ ⟶ Response Time

$S_{\mu PC} < S_{SoftV\mu PC} < S_{H\mu PC}$ ⟶ Size of CM.

ALU & Data Path :-

Shekar's Victory
DATE PAGE



- The registers, ALU and interconnecting BUS is called ALU Data path.
- Constant is used to increament PC.
- If Select = 0 ⇒ MUX output = Constant

  If Select = 1 ⇒ MUX output = Y
- The operations performed can be —

[No. of ~~cd~~ clocks = No. of Steps]

1. Register Transfer:

   e.g. $R_2 \leftarrow R_1$

   Step 1:

   R_1out, R_2in

   Min^m. No. of clocks = 1.

[No more than 1 Out operation can be in one Clock cycle]

2. ALU Operation: $(R_3 \leftarrow R_1 + R_2)$

   Step 1: R_1out, Yin

   Step 2: R_2out, Select = 1, ADD

   Step 3: Zout, R_3in

   Min^m. no. of clocks = 3

⇒ If

Step ①

Step 2:

• Ir

will de

3. Mer

• If

• If

Step 1:

Step 2:

Step 3:

4. Memo

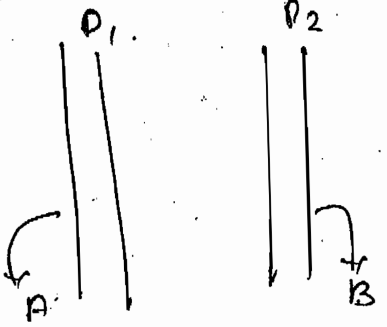
Step 1:

Step 2:

Step 3:

Shekar's Victory
DATE PAGE

⇒ If two data paths are used,



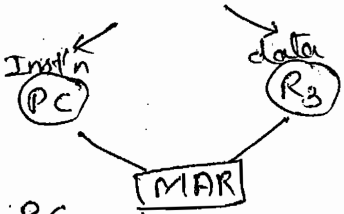
Step ① R_1out, Y_{in}, R_2out, Select =1, Add

Step 2: Z_{out}, R_3in

Min^m no. of clocks = 2.

• In a processor, the increase in no. of data paths will decreases clock cycles required.

3. <u>Memory Read</u>: $(R_1 \leftarrow M[(R_3)])$



• If instruction then put in PC, & .

• If data, then put in R_3.

Step 1: R_3out, MARin, Read

Step 2: WFMC (Wait for Memory Function to Complete)

Step 3: MDRout, R_1in

Min^m. no. of clocks = 3 (depends on WFMC)

4. <u>Memory Write</u>: $(M[(R_3)] \leftarrow R_1)$

Step 1: R_3out, MARin

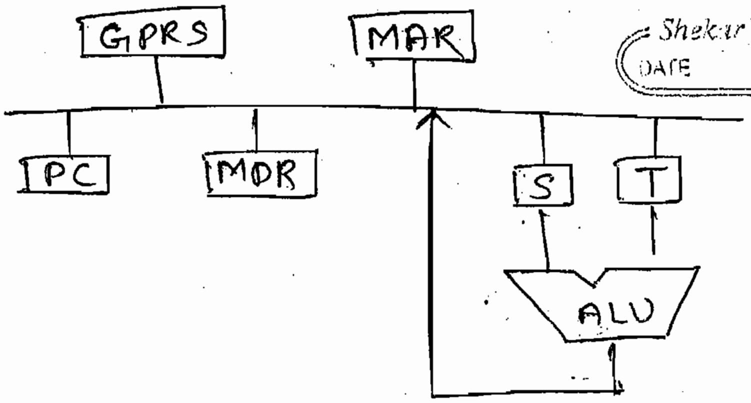
Step 2: R_1out, MDRin, Write

Step 3: WFMC

Min^m no. of clocks = 3

Q-

Shekar's Victory
DATE PAGE



The min^m no. of clock cycles required to perform

$\boxed{R_0 \leftarrow R_1 + R_2}$ A) 2 B) 3 C) 4 D) 5

An

~~Step 1: R_1out , Sin~~

~~Step 2 : R_2 out , ~~Add~~ Select = 1, Add~~

~~Step 3 : Zout , R_0in~~

Step 1 : ~~Sey~~ R_1out , Sin

Step 2 : R_2out , Tin , Add , Roin

Min^m no. of clocks = 2

NUMBER SYSTEM :-

	Radix/Base	Digits	Min^m Bits
Binary	2	0, 1	1
Octal	8	0 to 7	3
Decimal	10	0 to 9	4
Hexadecimal	16	0 to 9 A to F	4

Conver

① D

(2

② Binar

(10

$2^4 \times 1 + 2^3$

16 + 4

=

③ Hexa

(B

$16^3 \times B + 16^2$

④ Direc

Octal :

Binary :

Hexa :

Shekar's Victory
DATE PAGE

Conversions:-

① Decimal to Binary

$(23.875)_{10} = (?)_2$

$$\begin{array}{r|l l}
2 & 23 & 1 \\
2 & 11 & 1 \\
2 & 5 & 1 \\
2 & 2 & 0 \\
2 & 1 & 1 \\
 & 0 &
\end{array}$$

0.875×2

$\leftarrow 1.750$

0.750×2

$\leftarrow 1.500$

0.5×2

$\leftarrow 1$

$\Rightarrow (10111.111)_2$

② Binary to Decimal

$(10111.111)_2 = (?)_{10}$

~~$+2^2\times1+2^1\times1+2^0\times1$~~

$2^4\times1+2^3\times0+2^2\times1+2^1\times1+2^0\times1+2^{-1}\times1+2^{-2}\times1+2^{-3}\times1$

$16+4+2+1+\frac{1}{2}+\frac{1}{4}+\frac{1}{8}$

$=(23.875)_{10}$

③ Hexa to Decimal

$(B027)_{16} = (?)_{10}$

$16^3\times B+16^2\times0+16^1\times2+16^0\times7$

④ Direct Relationship amont Octal, Binary & Hexadecimal

Octal: $(6 \quad 6 \quad 6 \quad 6)_8$

Binary: 110110110110

Hexa: $(D \quad B \quad 6)_{16}$

Gold

Q- $(123456)_8 = (?)_{16}$ & $(?)_4$.

Shekar's Victory
DATE PAGE

1000	
1001	
1010	A
1011	B
1100	C
1101	D
1110	E

Convert to Binary.

$(123456)_8 \rightarrow 001010011100101110$

Hexa: $(A\ 7\ 2\ E)_{16}$ ←

Binary: 001010011100101110 →

Radix4: $(2\ 2\ 1\ 3\ 0\ 2\ 3\ 2)_4$

Base 4 means, It is 4 digit system so 2 bits required to represent.

Q- The no. of 1's in binary representation of the expression

$16^3 \times 11 + 16 \times 4 + 3 =$ ____

$(16^3 \times 11 + 16^2 \times 0 + 16^1 \times 4 + 16^0 \times 3)_{10}$

↓

$(B\ 0\ 4\ 3)_{16}$

↓

$(1011\ 0000\ 0100\ 0011)_2$

No. of 1s = 6

Q- The no. of 1's in binary representation of the expression

$4096 \times 9 + 256 \times 6 + 16 \times 7 + 2 =$ — —

$(16^3 \times 9 + 16^2 \times 6 + 16^1 \times 7 + 16^0 \times 2)_{10}$

↓ Binary

$(1001\ 0110\ 0111\ 0010)_2$

No. of 1's = 8

1000.
1001
1010 A
1011 B
1100 C
1101 D
1110 E

Shekar's Victory DATE PAGE

Q- A decimal no. has 46 digits, the no. of bits to represent in binary is ?

Sol

3 digit Decimal no.

Let 999

No. of digits ← $\boxed{10^3 - 1}$

↓ radix

3 digit Binary No.

Let 7

$\boxed{2^3 - 1}$

Conversion.

k → no. of digits in binary.

$$\boxed{10^3 - 1 = 2^k - 1}$$

bits

So,

$$10^{46} \cancel{-1} = 2^k \cancel{-1}$$

the

$$k = 46 \log_2 10$$

Q- A binary no. has 96 digits, the no. of ~~bits dig~~ bil to represent in decimal = is ?

$$2^{96} - 1 = 10^k - 1$$

$$\boxed{k = 96 \log_{10} 2}$$

the

Thus, for any 2 systems -

$$\boxed{r_1^{k_1} = r_2^{k_2}}$$

when ~~[illegible]~~ r_1 & r_2 → Radix / Base

k_1, k_2 → No. of digits

Gold

Shekar's Victory
DATE PAGE

~~Q~~ (a) Data Representation:-

- Fixed Point Representation
  - (1) Integer
  - (2) Small range of values
- Floating Point Representation
  - (1) Real or float data.
  - (2) Suitable for large range of values.

A) Fixed Point Data Representation:

3 approaches -

1) Signed Magnitude form:

b7	-	-	-	b2	b1	b0

b_7 (sign bit)

$b_7 = 0 \Rightarrow$ +ve magnitude
$b_7 = 1 \Rightarrow$ −ve magnitude

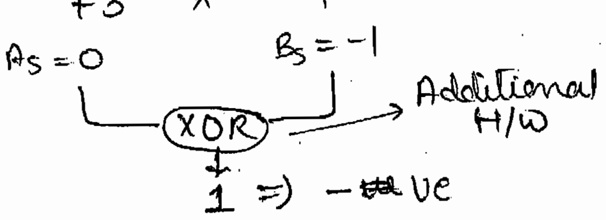
E.g.

$+14 \rightarrow 00001110$
$-14 \rightarrow 10001110$

Limitations:

a) Sign bit considered explicitly, requires additional hardware for resultant sign.

e.g. $+3 * -4$ $A_s, B_s \rightarrow$ Sign bit



b) Addition & Substraction are performed in separate hardware.

~~3)~~ c) Zero has two representation
+0 ⇒ 00000000
−0 ⇒ 10000000
Hence, difficult to test for zero.

2) 1

a) Sign
h/w

b) Zero
+0
−0

3) 2's

a) Sign
b) Add
only.

[

c) Zero

2) 1's Complement form :

+14 ⇒ 00001110

−14 ⇒ 1's complement of (+14)

⇓

11110001

a) Sign bit is not considered explicitly, no additional h/w required.

b) Zero has two representations

+0 ⇒ 00000000

−0 ⇒ 1's Complement → 11111111

3) 2's Complement Form :

+14 ⇒ 00001110

−14 ⇒ 2's complement of (+14)

⇓

1's complement + 1

$$\begin{array}{r} 11110001 \\ +1 \\ \hline 11110010 \end{array}$$

a) Sign bit not considered explicitly.

b) Addition & Substraction are performed using Adder only. i.e.

+B ⇒ → 1's complement

$-B = \overline{B} + 1$

$$\boxed{A - B = A + \overline{B} + 1}$$

c) Zero has only one representation.

+0 ⇒ 00000000

$$-0 \Rightarrow \begin{array}{r} 11111111 \\ +1 \\ \hline 00000000 \end{array}$$

Cold ① discarded.

8) Floating Point Representation:-

Shekar's Victory
DATE PAGE

$$\pm\, m * r^{\pm e}$$

m: mantissa/Significant

r: Radix | Base

e: Exponent

E.g. $\underset{m}{\underset{\downarrow}{+32.625}} \times \underset{r}{\underset{\downarrow}{10}}^{+3 \to e}$

$\underset{m}{\underset{\downarrow}{+1011.101}} \times \underset{r}{\underset{\downarrow}{2}}^{+101 \to e}$

↳ 32.625×10^{3}

left shift of decimal point $\swarrow$ 3.2625×10^{4}

right shift of decimal point $\searrow$ 326.25×10^{2}

$\downarrow$ $\underset{m}{\underset{\downarrow}{0.32625}} \times 10^{5}$

↳ ~~10H~~

e.g.② $\underset{m}{\underset{\downarrow}{0.1011101}} \times 2^{1001}$

~~No~~ the MST of mantissa should be non-zero, that is called Normalised floating point no.

• Any

1) N

Ra

2) ±

• Any

ted as

Single

1) S

sign of mantissa

2) E' =

E' is

Es valu

e.g. E =

E' =

E = 3

E' =

3) The

sign (1)

0/1

0/1

ory
3E

• Any floating point number represented as

Shekar's Victory
DATE PAGE

1) Normalised Form

$$\pm 0.1bbb--b \times 2^{\pm e}$$

Radix, Decimal point assumed.

2) $\pm \underbrace{1.bbb---b}_{m} * 2^{\pm e}$

radix, decimal point & 1 are assumed.

• Any floating point data in IEEE 754 can be represented as

Single Precision

1) ← 32 bits →

S	E'	M
1	8	23
sign of mantissa	Baised Exponent	Mandissa

2) $E' = E + 127$

E' is taken to make Es value always positive.

e.g. $E = -3$

$E' = -3 + 127 = 124$

$E = 3$

$E' = 3 + 127 = 130$

Double Precision

1) ← 64 bits →

S	E'	M
1	11	52

2) $E' = E + 1023$

For k bits bias value is $\Rightarrow 2^{k-1} - 1$

↓

for Single & Double precision both

3) The value represented by Single Precision –

Sign (1)	Exponent (8)	Mandissa (23)	Value
0/1	Other than all 0's & all 1's	xxx--x	$(-1)^s (1.m) * 2^{E'-127}$
0/1	all 0's	other than all zero's	$(-1)^s (0.m) * 2^{-126}$

zero,

Gold

Note: $1.011\ 000 \cdots 0$

$1 + 2^{-1}\times 0 + 2^{-2}\times 1 + 2^{-3}\times 1$

For Double Precision

Value

$(-1)^s\ (1.m)\ 2^{E'-1023}$

$(-1)^S\ (0.M)\ 2^{-1022}$

Q- what is the value denoted by 23 Single precision.

$\overset{1}{0}\ \overset{8}{\overleftrightarrow{10000011}}\ \overset{23}{\overleftrightarrow{1100\cdots 0}}$

$S=0 \qquad E'=131$

$V = (-1)^s\ (1.M)\ 2^{E'-127}$ ← 2nd form

$= (-1)^0\ [1+(1100\cdots0)]\ 2^{131-127}$

$= \cancel{(-1)^0}\left(1 + 2^{-1}\times 1 + 2^{-2}\times 1 + 2^{-3}\times 0 \cdots\right)\times 2^4$

$= \left(1+\frac{1}{2}+\frac{1}{4}\right)\times 2^4 \Rightarrow 16+8+4 = 28$

Q- Identify the value denoted by

~~$V=(-1)^1$~~ $1 \quad 00101111 \quad 100\cdots0$

$S=1 \qquad E'=47$

$V = (-1)^1\ (1+2^{-1}\times 1)\ 2^{47-127} \Rightarrow -(1.5) * 2^{-80}$

Q- Represent -23.875 in single precision →?

Sol:-

↓ Binary

$-(10111.111)_2$

↓

$1.0111111 \times 2^{+4}$

$M = 0111111$

$E' = \cancel{4}\ E + 127 = 4+127 = 131$

$S = 1$

1	8 bits	23 bits
S	E′	M
1	10000011	0111111 00---0

9 PIPELIN

Eg:

A_i

R_1

Segment 1 or Stage 1

Stage 2

Stage

1

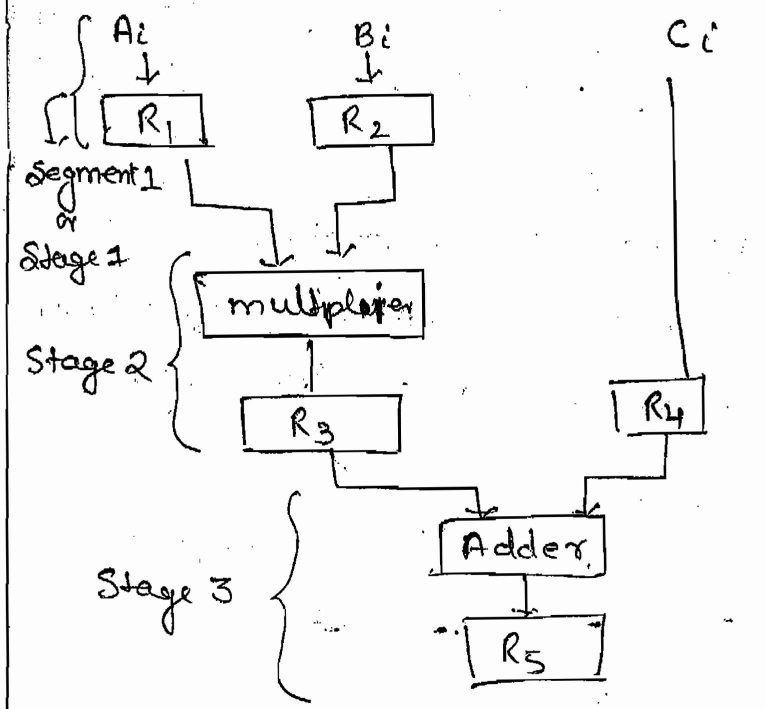
Let

- Ass
- the
- Using

Shekar's Victory
DATE PAGE

(9) PIPELINING:

Eg: $D_i \leftarrow A_i * B_i + C_i$
$i = 1$ to 4



1 segment ≡ 1 clock cycle.

Let n → no. of tasks / inst'n
k → no. of stages / segment

- Assume each stage completes in 1 clock cycles.
- the No. of clocks in sequential or non-pipelined

No. of clocks = n * k

e.g. → 4*3 = 12 clock cycles.

- Using pipelined execution

Clocks ↓	Seg 1 $_{R_1, R_2}$	Seg 2 $_{R_3, R_4}$	Seg 3 $_{R_5}$
1	A_1, B_1	—	—
2	A_2, B_2	$A_1 * B_1, C_1$	—
3	A_3, B_3	$A_2 * B_2, C_2$	$A_1 * B_1 + C_1$
4	A_4, B_4	$A_3 * B_3, C_3$	$A_2 * B_2 + C_2$
5	—	$A_4 * B_4, C_4$	$A_3 * B_3 + C_3$
⑥	—	—	$A_4 * B_4 + C_4$
7			
8			

No. of clocks = 6

Gold

$\boxed{\text{No. of clocks} = k + (n-1)}$

k ⇒ cycle taken by 1st task

(n-1) ⇒ ~~cycle taken~~ by rest of tasks.

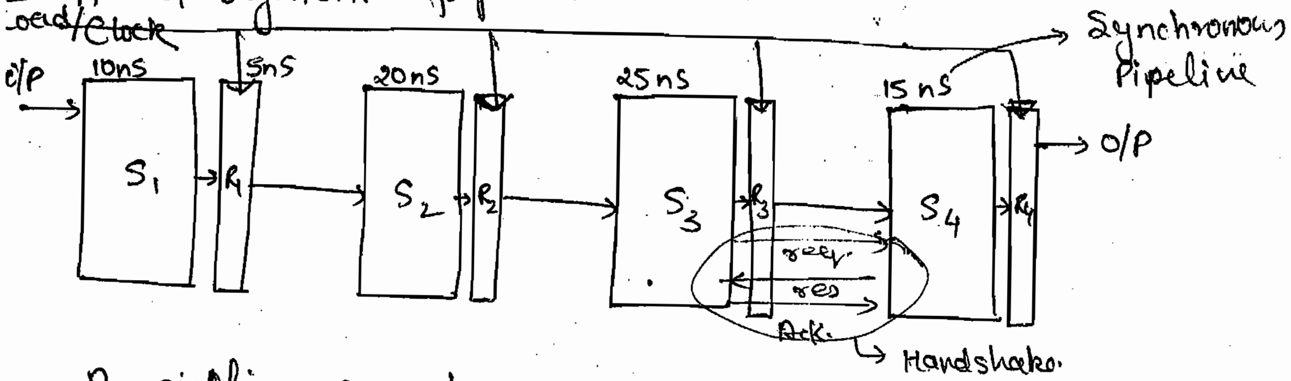
Shekar's Victory DATE PAGE

e.g.

$3 + (4-1) = 6$

- Pipelining allows to impliment parallel processing in a uniprocessor system.
- Pipelining improves the performance or throughput
- Pipelining divides the given problem into sub operations called segment, all segments operate simultaneously.

Q. A 4-segment pipeline is



A pipeline can be -

1) Asynchronous Pipeline.

The data flow along the pipeline stages is controlled with hand-shake protocol.

2) Synchronous Pipeline

- On a common load all the registers transfer data to the next level or stages, simultaneously.
- By default all the pipelines are synchronous.

A task is the total operation performed by going through all stages of a pipeline.

ken by
~~stask~~
~~aken~~ by
asks,

g in a

operations
ously.

ynchronous
Pipeline
o/p

ansfer

us.

oing

Shekar's Victory
DATE PAGE

Stage Delay (t_i) :-

The processing time in each stage or segment, it varies from stage to stage based on type of operation.

Interstage Delay (t_d) –

Interstage transfer of data (buffer overhead), it is same for all stages.

- The time period for the clock cycle in pipelined execution

$$\boxed{t_p = \overset{k}{\underset{i=1}{\max^{m}}} \{t_i\} + t_d}$$

↳ adding buffer overhead

$\max^{m}\{t_i\} >>> t_d$, so

$$\boxed{t_p = \overset{k}{\underset{i=1}{\max}} \{t_i\}}$$

- The frequency of pipeline

$$\boxed{f_p = \frac{1}{t_p}}$$

- The time required in pipelined execution for n, k –

$$\boxed{T_p = \{k + (n-1)\} * t_p}$$

Speed Up :

$$\boxed{S = \frac{\text{time without pipeline}}{\text{time with pipeline}}}$$

Gold

Shekar's Victory
DATE PAGE

- Let t_n is the time required to complete a task in non-pipeline

$$\boxed{S = \frac{n * t_n}{\{k+(n-1)\} * t_p}} \quad \text{———} \quad ①$$

~~$t_n \simeq ktp$~~

1 task time in non-pipeline ← $t_n \simeq k * t_p$

t_p → 1 clock cycle time.

k → No. of cycles

Substitute in ①

$$\boxed{S = \frac{n\,k}{k+(n-1)}} \quad \text{———} \quad ②$$

- For large no. of tasks ~~k~~

$$k+(n-1) \longrightarrow n$$

Substitute in ①

$$S = \frac{n \times t_n}{n * t_p}$$

$$\boxed{S = \frac{t_n}{t_p}}$$

Substitute in ②

$$S = \frac{n\,k}{n}$$

$$\boxed{S = k}$$

- The max^m speed up that can be achieved using pipelined processor is always equal to No. of stages.

$$\boxed{S_{max} = S_{ideal} = k}$$

- The efficiency of a pipeline,

$$\boxed{E_k = \frac{S}{k}}$$

$$= \frac{\frac{nk}{k+(n-1)}}{k}$$

$$\boxed{E_k = \frac{n}{k+(n-1)}}$$

- The throughput of a pipeline,

H_k = No. of jobs performed in unit time

$$H_k = \frac{n}{\{k+(n-1)\} * t_p}$$

$$H_k = E_k * \frac{1}{t_p}$$

$$\boxed{H_k = f_p * E_k}$$

Q- The no. of clock cycles needed in a pipeline to execute 100 tasks in six segments, is —?

$$k + (n-1)$$
$$= 6 + (100-1)$$
$$= 6 + 99 = 105 \text{ clocks.}$$

Q- For a given six-segment pipeline, what is the speed up for 100 tasks.

$$S = \frac{nk}{k+(n-1)}$$

$$= \frac{100 \times 6}{105}$$

$$= 5.71$$

Gold

Q- During floating point Arithmetic the time delays for 4-stages of a pipeline are 50ns, 80ns, 90ns & 80 ns, with an interstage delay of 10ns.

(i) What is the speed up achieved ?

$t_p = 90 + 10$

$t_p = 100 \, nS$

~~Sx~~ $t_n = 50 + 80 + 90 + 80 + 10$ ↳ final result will go to resister only.

$= 300 + 10 = 310$

$S = \frac{t_n}{t_p}$

$S = \frac{310}{100}$

$S = 3.1$

(ii) Max^m. Speed up achieved is ?

$S_{max} = k$

$= 4$

Q- Consider a 4-stage pipeline, which is operated with 1 MHz clock. What is the ave. time required for 10 instructions ?

$n = 10, \quad k = 4, \quad t_p = 1 \, MHz$

$t_p = \frac{1}{10^6} = 1 \, \mu sec.$

~~Tavg~~ =

$T_p = (k + (n-1)) * t_p$

$= (4 + 9) \times 1 \, \mu sec.$

$= 13 \, \mu sec.$

Q- Consid

a uniform

5ns.

(i) Wha

(ii) What

S

Q- Consi

of S_3 i

S_1 is ho

of S_1 i

t_p

t_n

S

days for tory SE ~~20~~ ns,

t will sister only.

id with

er co

Shekar's Victory DATE PAGE

Q. Consider a 4-stage pipeline where each stage takes a uniform delay of 20 nS and has buffer overhead of 5nS.

(i) What is the freq. of pipeline?

$$t_p = 20 + 5$$
$$= 25 \text{ nS}$$

$$f_p = \frac{1}{25}$$
$$= 0.04 \times 10^9$$
$$= 40 \text{ MHz}$$

(ii) What is the speed up achieved?

$$S = \frac{t_n}{t_p}$$

$$S = \frac{20 \times 4 + 5}{25}$$

$$S = \frac{85}{25} = 3.4$$

Q- Consider a 5-stage instruction pipeline, where delay of S_3 is twice to that of S_2 & half to that of S_5. S_1 is having same delay as that of S_3 & S_4. The delay of S_1 is 10nS, what is the speed up achieved?

	S_1	S_2	S_3	S_4	S_5
	t_1	$t_1/2$	t_1	t_1	$2t_1$
∴	10	5	10	10	20

$$t_p = 20$$
$$t_n = 55$$
$$S = \frac{55}{20} = 2.75$$

Gold

Shekar's Victory DATE PAGE

Q- ~~Sr is~~ Consider two pipelines A & B where A is having 5 stages with delays 3, 1, 4, 1 & 2 ns. B is having 9 stages with a uniform delay of 2 ns. How much time is saved using design B over design A for 100 instructions?

$\boxed{T_p = \{k + (n-1)\} * t_p}$

$n = 100$

$t_{ap} = 4$ ~~ns~~

$t_{bp} = 2$ ~~x9~~ $=$ ~~18~~ 2

~~B~~ $T_a = \{5 + 99\} \times 4$
$= 104 \times 4$
$= 416$

$T_b = (9 + 99) \times 2$
$= 108 \times 2$
$= 216$

~~Ex~~ Time saved $= T_a \sim T_b$
$= 200$ ns

Q- What is the efficiency of a six segment pipeline for 100 tasks?

$$E_k = \frac{n}{k + (n-1)}$$
$$= \frac{100}{6 + 99}$$
$$= 0.95$$

3) Instruction Pipeline:

Each instruction execution involves the phases - IF, ID, OF, Ex etc. Overlapping of these phases for multiple instructions is instruction pipeline.

- The behaviour of a pipeline can be visualized using space - time diagram.
- For 4 - instructions in 4 stages

A is
B is
ns. How
sign A
2
x 2
eline
s -
cases
l using

Space-time Diagram

Shekar's Victory
DATE PAGE

	1	2	3	4	5	6	7
I_1	IF	ID	OF	Ex			
I_2		IF	ID.	OF	Ex		
I_3			IF	ID	OF	Ex	
I_4				IF	ID	OF	Ex

All segments are fully overlapped.

Q. Consider a 4-stage pipeline where the clock cycles required for each stage of 4 instructions given as

	F	D	E	S		
I_1	2	1	2	2	⇒ 7	~~9~~
I_2	1	3	3	2	⇒ 9	~~8~~
I_3	2	2	2	2	⇒ 8	~~4~~
I_4	1	3	1	1	⇒ 6	~~2~~

(i) The no. of clock cycles required in sequential or non-pipelined execution is ?

$7+9+8+6 = 30$ clocks.

(ii) No. of clock cycles req in pipelined execution

~~2+3+3+2+3+3+2=20~~

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
I_1	F	F	D	E	E	S	S								
I_2		—	F	D	D	D	E	E	E	S	S				
I_3			—	F	F	D-	D	D	E	E	E	S	S		
I_4				—	—	F	—	—	D	D	D	E	~~S~~	S	

Gold

14 Clocks

Shekar's Victory
DATE PAGE

Stall Cycle:

During which no meaningful op~~erations per~~formed, for an instruction. It arise —

1) Uneven stage clock cycles for inst's
2) Data and control dependency
3) The effective speed up

$$S_{eff.} = \frac{\cancel{Stalls}\ S_{ideal}}{1+(Stalls\ freq. * Stall\ Sycles)}$$

$$\boxed{S_{eff.} = \frac{k}{1+(Stall\ freq. * Stall\ cycles)}}$$

Q- Suppose there are 5 stages that have 1 stall cycle per memory dependency. What is the effective speed-up with 20% ~~at~~ memory reference ?

$$S_{eff} = \frac{5}{1+\frac{20}{100} * 1}$$

$$= \frac{5}{1.2} \simeq 4.16$$

Q- Consider a pipelined processor with 4-stages to execute the loop

```
for(i=1 ; i<=100 ; i++)
{
    I1 ;
    I2 ;
    I3 ;
    I4 ;
}
```

The
instruction

I_1

I_2

I_3

I_4

(i) The

$i=1$ I_1 I_2 I_3 I_4 1 S_1

$i=2$ I_1.

(ii) The stages

I_1

I_2

I_3

I_4

formed,

The clock cycles needed by the stages for each instruction are -

	S_1	S_2	S_3	S_4
I_1	1	2	1	2
I_2	2	1	2	1
I_3	1	1	2	1
I_4	2	1	2	1

(i) The O/P of I_1 for $i = 2$ is available after ____ clocks.

stall

nce ?

		1	2	3	4	5	6	7	8	9	10	11	12	13
$i=1$	I_1	S_1	S_2	S_2	S_3	S_4	S_4							
	I_2		S_1	S_1	S_2	S_3	S_3	S_4						
	I_3			—	S_1	S_2	$\overline{S_3}$	S_3	$S_{\cancel{4}3}$	S_4				
	I_4				—	S_1	S_1	S_2	$\overline{S_3}$	S_3	$S_{\cancel{4}3}$	S_4		
$i=2$	I_1						—	—	S_1	S_2	S_2	$\overline{S_3}$	$S_{\cancel{4}3}$	S_4

No. of clocks = 13.

(ii) The no. of cycles needed by 4 instructions in 4 stages as

to

	S_1	S_2	S_3	S_4
I_1	2	1	1	1
I_2	1	3	2	2
I_3	2	1	1	3
I_4	!	2	2	2

	1	2	3	4	5	6	7	8	9	10	11		
I_1	S_1	S_1	S_2	S_3	S_4								
I_2		—	S_1	S_2	S_2	S_2	S_3	S_3	S_4	S_4			
I_3				S_1	S_1	—	S_2	—	S_3	—	S_4	S_4	S_4
I_4				—	—	S_1	—	S_2	S_2	S_3	S_3	S_4	S_4

No. of clocks = 13

Q- The no. of clocks required to execute the loop

- 23 clocks

```
for (i=1 ; i≤2 ; i++)
{
    I1 ;
    I2 ;
    I3 ;
    I4 ;
}
```

2 iteration → 8 instructions.

Pipeline Conflicts / Hazards / Dependencies / Difficulties :-

i) Data Dependency

E.g. I_1 : $R_2 \longleftarrow R_0 + R_1$

I_2 : $R_3 \longleftarrow R_2 * R_4$

I_3 : $R_5 \leftarrow R_6 - R_7$

	1	2	3	4	5	6	7
I_1	IF	ID	OF	EX			
I_2		IF	ID	OF	EX		
I_3			IF	ID	OF	EX	
I_4				IF	ID	OF	EX

- One instruction depends one results of its previous, leads to data dependency. (like R_2 in I_1 & I_2)

- Solutions → H/W (pipeline)
  → S/W (Compiler)

i) Dyn

B

parallel

ii) Del

I

I

I

But

iii) Oper

I

• By Def

Or Con

Fetch

the F,

complete

E stag

3 - clock

for thi

require

No. of clocks = 13

op 23 clocks

ies :-

6	7
Ex	
OF	Ex

evious,

Shekar's Victory
DATE PAGE

i) Dynamic Scheduling (H/w or S/w)

Rearrangement of instructions dynamically.

I_1 I_2 I_2 → I_1 I_3 I_2

But the instructions after dependency must be parallel.

ii) Delayed Load (S/w)

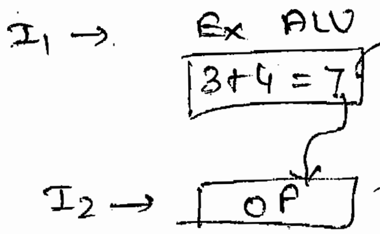
Will use NOP (No- operation)
↓
executed by OS.

I_1 I_2 I_3 → I_1 NOP I_2

NOP → will execute all the stages but not give any result.

But increases no. of clock cycles.

iii) Operand Forwarding (H/w)



In addition of ~~Instead~~ ~~of~~ storing value in R_2, the copy of value is also directly fetched by I_2.

- By Default Operand forwarding will be used in ques.

Q:- Consider a pipelined processer with 4-stages Fetch (F), Decode (D), Execute (E) and Write-back (W). The F, D & W stages requires 1-clock cycle to complete the operation for every instruction. The ~~each~~ E stage requires 1-clock for addition & substraction, 3-clocks for multiplication. If Operand Forwarding is used, for the following program, the no. of clock cycles required ~~Told~~ complete that ?

Shekar's Victory
DATE PAGE

I_1: ADD R_2, R_1, R_0

I_2: MUL R_4, R_3, R_2

I_3: SUB R_6, R_5, R_4

	F	D	E	W
I_1	1	1	1	1
I_2	1	1	3	1
I_3	1	1	1	1

	1	2	3	4	5	6	7	8
I_1	F	D	E	W				
I_2		F	D	E	E	E	W	
I_3			F	D	—	—	E	W

2) Control Dependency

E.g.

I_1

I_2

I_3 BNZ I_7

I_4

|

I_7

	1	2	3	4	5	6	7
I_1							
I_2							
I_3			IF	ID	OF	Ex	
I_7/I_4				—	—	—	IF

- Arise due to branching.
  ↳ I_3 BNZ I_7
  Untill I_3 will not complete, can't initiate I_4 or I_7.

~~1) Prefetched Target address.~~

1) Pref
I
address

2) Dela

that I

3) Branch

S.

Pre
ch

① I

Branch r
takes
(Target = I

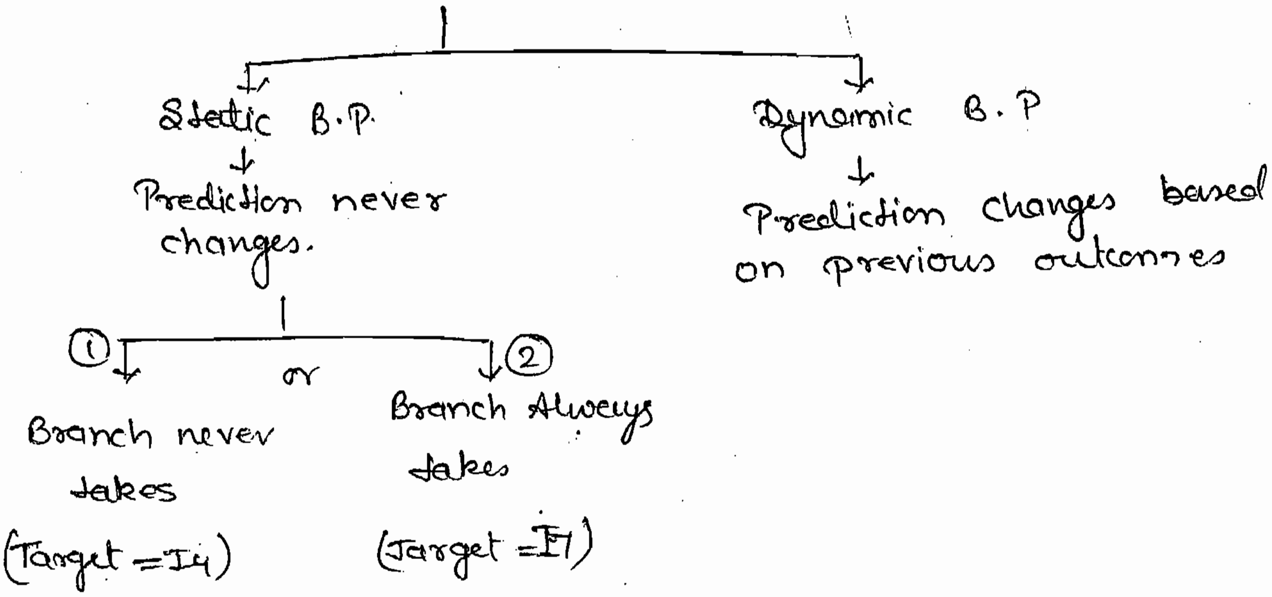
* If
penalty

Shekar's Victory DATE PAGE

1) Prefetch target Address (H|W) :
If branch is found, it will prefetch the ~~bre~~ target address and accordingly I_4 or I_7 will start.

2) Delayed Load (S/W):
Will execute NOP untill I_3 completes after that I_4 or I_7 executes, but increases clock cycles.

3) Branch Prediction (Default) :



Static B.P.
Prediction never changes.

Dynamic B.P
Prediction changes based on previous outcomes

① Branch never takes (Target = I_4)

or

② Branch Always takes (Target = I_7)

6 7

* If prediction goes wrong, leads to branch penalty.

Ex

IF

I_7.

Gold