CLASS TEST

S.No. : 01SK_CS_ABC_010424

MADE EASY
India's Best Institute for IES, GATE & PSUs

Delhi | Bhopal | Hyderabad | Jaipur | Pune | Kolkata

Web: www.madeeasy.in | **E-mail:** info@madeeasy.in | **Ph:** 011-45124612

COMPUTER ORGANIZATION

COMPUTER SCIENCE & IT

Date of Test : 01/04/2024

ANSWER KEY ➤

1.	(a)	7.	(b)	13.	(b)	19.	(d)	25.	(d)
2.	(b)	8.	(a)	14.	(b)	20.	(d)	26.	(d)
3.	(c)	9.	(c)	15.	(b)	21.	(b)	27.	(b)
4.	(a)	10.	(b)	16.	(b)	22.	(d)	28.	(b)
5.	(d)	11.	(c)	17.	(c)	23.	(b)	29.	(a)
6.	(b)	12.	(d)	18.	(b)	24.	(b)	30.	(c)

DETAILED EXPLANATIONS

1. (a)

Since, it uses horizontal micro-programmed that requires 1 bit control / signal.
For 125 control signal, we need 125 bits.
Total number of micro-operation instruction = 215 × 6 = 1290
It requires 11 bit.

2. (b)

- **Isolated Input/Output:** This configuration uses the common bus and common address space but different control signal for both memory and input/output, so that memory address range is not affected by interface address assignment.
- **Memory Mapped Input/Output:** This configuration uses common bus and common control signals but unique address space.
- In synchronous clock is common and in asynchronous clock is different.

3. (c)

When instruction is a computation:

Memory reference : Fetch instruction
Fetch reference of the operand
Fetch operand

Total 3 memory references.

When instruction is a branch:

Memory reference : Fetch instruction
Fetch operand reference and loading program counter

Total 2 memory references.

4. (a)

- **Implied addressing mode:** Specified implicitly in the definition of instruction.
- **Immediate addressing mode:** Specified in the address field of an instruction.
- **Register addressing mode:** Registers which are in CPU.
- **Register indirect addressing mode:** Register specifies the address of the operand.

5. (d)

Instruction pipelining is a technique that implements a form of parallelism called instruction level parallelism with a single processor.
It therefore allows faster CPU thoughput.

6. (b)

Programmed I/O: Processor issues an I/O command, on behalf of a processor, to an IO module; that process then busy-waits for the operation to be completed before proceeding.
Interrupt driven I/O: The processor isues an IO command on behalf of a process, continues to execute subsequent instruction, and is interrupted by the IO module when the latter has completed its work.
Direct memory access: A DMA module controls the exchange of data between main memory and IO module.

7. (b)

For 1 second it take 10^9 byte

$$\text{So for 64 kbyte it takes} = \frac{64k}{10^9} = 64\ \mu\text{sec}$$

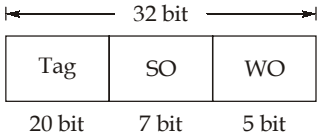
$$\text{Main memory latency} = 64\ \mu\text{sec}$$

$$\text{Total time required to fetch} = 64\ \mu\text{sec} + 64\ \mu\text{sec} = 128\ \mu\text{sec}$$

8. (a)

$$\text{Number lines} = \frac{64\text{K}}{32} \Rightarrow 2^{11}$$

$$\text{Number sets} = \frac{2^{11}}{2^4} \Rightarrow 2^7$$



$$\text{Tag memory size} = S \times P \times \#\text{ tag bits}$$
$$= 128 \times 16 \times 25 = 51200 \text{ bits}$$

9. (c)

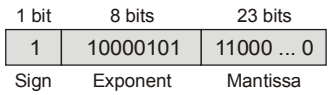
Consider each statement :

S_1 : Microprogrammed control unit uses variable logic to interrupt instruction since its uses encoded scheme for the instruction.

S_2 : Horizontal microprogramming control unit does not requires an additional hardware (like a decoder) because a fixed logic is associated with the instructions.

S_3 : The performance of a system depends on the direct proportion of memory accesses satisfied by cache.

10. (b)



$$\text{Bias exponent value} = 10000101$$

$$\text{Actual exponent} = 10000101 - 127 \quad [\because \text{ 127 is bias}]$$
$$= 133 - 127 = 6$$

$$\text{Normalized mantissa bits} = 11000000000000000000000$$

$$\text{Actual value} = 1.11000000000000000000000$$

$$\text{Decimal number} = 1.11000 \times 2^6$$
$$= -(1110000)_2$$
$$= -(112)_{10}$$

11. (c)

Rate of transfer to or from any one disk = 30 MBps.

Maximum memory transfer rate = $\frac{4\text{ B}}{10\times10^{-9}} = 400\times10^{6}\text{Bps} = 400\text{ MBps}$

Since rate of data transfer = 30 MBps

Here number of disk transfer = $\left\lceil\frac{400}{30}\right\rceil = 13$

Therefore, 13 disks can simultaneously transfer data to or from the main memory.

12. (d)

	1	2	3	4	5	6	7	8	9	10	11
I_1	IF	ID	EX	MM	WB						
I_2		IF	ID	ID	EX	MM	WB				
I_3			IF	IF	ID	EX	MM	WB			
I_4					IF	ID	EX	MM	WB		
I_5						IF	ID	ID	EX	MM	WB

Total 11 cycles are required.

13. (b)

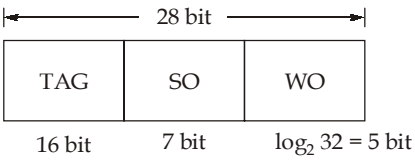
Memory mapped I/O uses the same address bus to address both memory and I/O devices the memory and registers of the I/O devices are mapped to address values.
So, when an address is accessed by the CPU, it may refer to a portion of physical RAM, but it can also refer to memory of I/O device.

14. (b)

$$\text{Number of lines} = \frac{16\text{ KB}}{32\text{ B}} = \frac{2^{14}}{2^{5}} = 2^{9}$$

$$\text{Number of sets} = \frac{2^{9}}{2^{2}} = 2^{7}$$

Physical address size = 256 MB = 28 bits



Total set will be from 0 to 127 (using 7 bits).
In option (b)

$(FB3D64C)_{16}$ = F B 3 D 0 1 1 0 0 1 0 0 C

16 bit TAG | 7 bit set | 5 bit WO

It's decimal value is 50

Hence option (b) is correct.

15. (b)

Considering each statement :

S_1 : Delayed control transfer, also known as delayed branching, is an attempt to cope with control hazards.

S_2 : The branch target stores the previous target address for the current branch, other algorithms for branch prediction also exist.

S_3 : For any given instruction set architecture implemented on a *N*-stage pipelined processor, *N* registers probably is not enough registers to completely prevent structural hazards involving a shortage of register hardware.

16. (b)

$$P_1 \text{ CPU time} = \frac{[1 \times 0.1 + 2 \times 0.1 + 3 \times 0.5 + 4 \times 0.3]}{1.5 \times 10^9}$$

$$= 2 \times 10^{-9} \text{ sec} = 2 \text{ nsec}$$

$$P_2 \text{ CPU time} = \frac{[2 \times 0.1 + 2 \times 0.1 + 2 \times 0.5 + 2 \times 0.3]}{2.5 \times 10^9}$$

$$= 0.8 \times 10^{-9} \text{ sec} = 0.8 \text{ nsec}$$

$\therefore$ P_2 is faster than P_1 processor.

17. (c)

2.5 memory reference per instruction $\Rightarrow \frac{1000}{2.5}$ instruction per 1000 reference.

$\Rightarrow$ 400 instructions.

Now
$$200 = \left(\frac{260}{400}\right)x + \left(\frac{120}{400}\right)2x$$

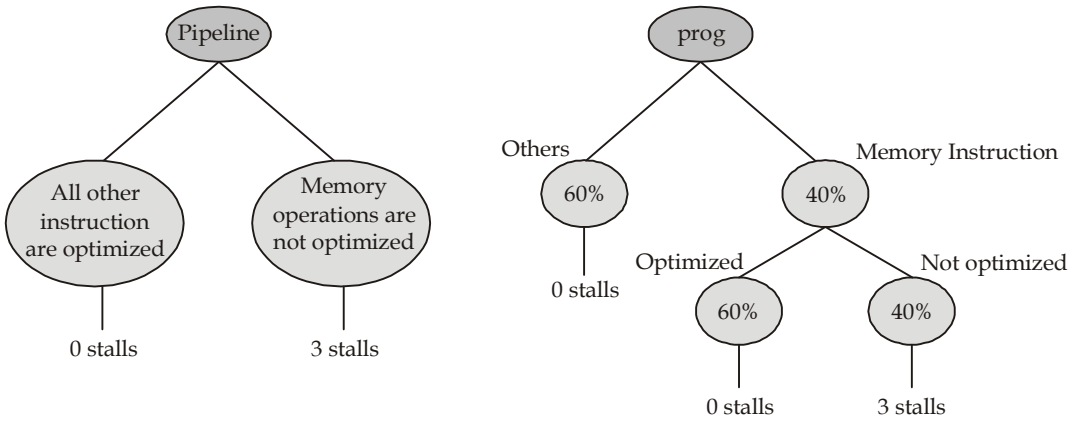
$$x = \frac{400 \times 200}{500}$$

$$x = \frac{80000}{500}$$

$$x = 160$$

$$2x = 320$$

18. (b)



Number of stalls/Instruction = (0.4 × 0.4 × 3) = 0.48

Average instruction ET = (1 + # stalls / Instruction) × Cycle time

= (1 + 0.48) × 8 ns = 11.84 ns

19. (d)

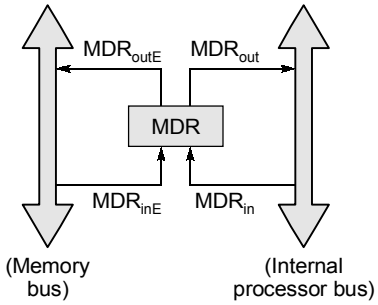
Increasing the cache line size brings in more from memory when a miss occurs. If accessing a certain byte suggests that nearby bytes are likely to be accessed soon (locality), then increasing the cache line essentially prefetches those other bytes. This, in turn, forestalls a later cache miss on those other bytes. If misses occur because the cache is too small, then the designers should increase the size!

Conflict misses occur when multiple memory locations are repeatedly accessed but map to the same cache location. Consequently, when they are accessed, they keep kicking one another out of the cache. Increasing the associativity implies that each chunk of the cache is effectively doubled so that more than one memory item can rest in the same cache chunk.

20. (d)

Considering each statement :

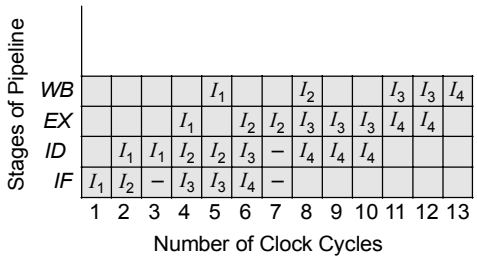
S_1 : 4-control signals are needed for each data register.



MDR is directly connected to data lines of the processor. It has 2 input and 2 output. Data may be loaded into MDR either from memory or from internal bus. Data present in MDR may be placed on either bus are memory. It requires total 4 control signals.

S_2 : The main disadvantage of direct mapping is that cache hit ratio decreases sharply if two or more frequently used blocks map on the same region.

21. (b)



22. (d)

IF	ID	OF	PD and WB
I_1 : 1 memory reference	2 memory reference	1 memory reference	–
I_2 : 1 memory reference	3 memory reference	2 memory reference	–
I_3 : 1 memory reference	1 memory reference	–	1 ALU operation
I_4 : 1 memory reference	3 memory reference	2 memory reference	–
I_5 : 1 memory reference	–	–	1 ALU operation

Total Execution Time = (Number of memory reference × 3 cycles + Number of ALU operation × 1 cycles) × cycle time

$$= (19 \times 3 + 2 \times 1) \times \frac{1}{5\ \text{GHz}} = \frac{59}{5}\ \text{ns} = 11.8\ \text{ns}$$

23. (b)

The required probability = $^{10}C_3 (0.35)^3 (0.65)^7 = 0.252$

24. (b)

Time taken by I/O device = $\frac{16\,\text{MB}}{128\,\text{kB}} = 128\,\text{sec}$

Percentage time CPU is busy = $\frac{128}{128+28} \times 100 = 82.05$

25. (d)

(i) is true, it is logically done.

(ii) is false, structural dependency resolved using re-naming.

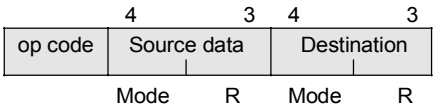
(iii) Delayed branch re-arranges code to reduce control dependency.

26. (d)

$$\begin{array}{r} 10001110 \\ 10000000 \\ \text{Sum} = \overline{100001110} \end{array}$$

Z = 0, C = 1, O =1, S = 0

27. (b)



Register = 5 = 3 bits

Modes = 14 =4 bits

Type	Single operand or No operand	Double operand
Arithmetic (10)	2	8
Logic (15)	9	6
Data moving (20)	12	8
Branch (10)	5	5

Total 27 double operands = 5 bits

Size of instruction word = 5 + 7 + 7 = 19

28. (b)

The device generates $8 \times 1024 = 8192$ bytes/sec

i.e. 1 second 8192 bytes

$$\text{for 1 byte} \Rightarrow \frac{1}{8192}\text{sec}$$

$$\Rightarrow 122\,\mu s$$

Given that each interrupt consumes 100 μs.

$\therefore$ Fraction of processor time consumed in $= \frac{100}{122}$ (for every byte) $= 0.82$

29. (a)

3 memory reference $\rightarrow$ 1 instruction

900 memory reference $\rightarrow$? instruction

$$\text{Number of instruction} = \frac{900}{3} = 300$$

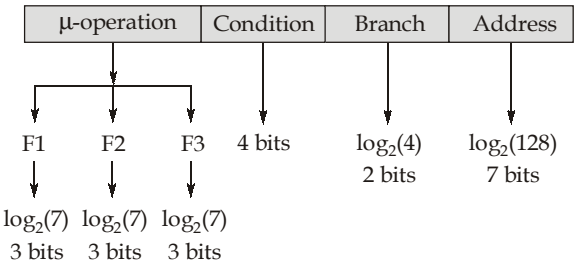
$$\text{Number of memory stalls/instruction} = \left[\frac{\text{Number of miss } L_1}{\text{Number of instruction}} \times \text{Hit } L_2\right] + \left[\frac{\text{Number of miss } L_2}{\text{Number of instruction}} \times \text{Miss panelty } L_2\right]$$

$$= \left[\frac{200}{300} \times 100\right] + \left[\frac{80}{300} \times 300\right]$$

$$= \left[\frac{200}{3} + 80\right] \simeq 146.66 \text{ cycles}$$

30. (c)

μ-instruction format:



Size of one micro-instruction = (3 + 3 + 3) + 4 + 2 + 7 = 22 bits

So, option (c) is correct answer.

■■■■