

Sessional Test II–April, 2023

ID No: .....

Semester II

Total No. of Pages: 02

### **Title of the Course: Computer System Architecture**

**Time: 90 minutes**

Course Code: CS118

**Max. Marks: 40**

### Instructions:

**Instructions:**

- There is one question having five parts. Each part is having four distinct options out of which only one choice will be correct. There is no negative marking for incorrect answers.

For Section B

- There are 6 Questions of 2 marks each. There is a choice to attempt 5 questions out of 6.

### For Section C

- There are 4 Questions of 5 marks each. There is a choice to attempt 3 questions out of 4.

### For Section D

- There are 2 Questions of 10 marks each. There is a choice to attempt 1 question out of 2.

## **Section-A**

(All Questions are Compulsory. Each question carries 01 mark)

1.

a) Which one of the following interrupts can be caused by an executing program?

- |       |          |      |          |
|-------|----------|------|----------|
| (i)   | Internal | (ii) | External |
| (iii) | Hardware | (iv) | Software |

b) In which addressing mode, the address of the operand is not required in the instruction itself?

- |                                 |                               |
|---------------------------------|-------------------------------|
| (i) Implied Addressing mode     | (ii) Register Addressing mode |
| (iii) Immediate Addressing mode | (iv) Direct Addressing mode   |

c) Shared memory multiprocessor fit best into which one of the following Flynn's classification of computers?

- (i) SISD      (ii) SIMD  
(iii) MISD      (iv) MIMD

d) For transferring data from keyboard to the attached computer, which of the following would be the preferred mode of transfer?

- |       |                               |      |                               |
|-------|-------------------------------|------|-------------------------------|
| (i)   | Direct memory access (DMA)    | (ii) | Programmed I/O                |
| (iii) | Hardware interrupt driven I/O | (iv) | Software interrupt driven I/O |

e) In FIFO stack organization, after PUSH instruction is executed, the contents of stack pointer gets \_\_\_\_\_ and after POP the contents of SP gets \_\_\_\_\_.

- |       |                          |      |                          |
|-------|--------------------------|------|--------------------------|
| (i)   | Incremented, decremented | (ii) | Decremented, incremented |
| (iii) | Incremented, incremented | (iv) | Decremented, decremented |

### Section-B

(Attempt any 5 questions, each question carries 02 marks)

2. The contents of flag/status register gets updated after execution of any instruction. Specify the various status bits of a general computer system.
3. Direct memory access is a technique of data transfer between processor and memory. Specify the two ways of data Transfer in DMA.
4. Mention with brief description the various types of interrupts that can cause a break in the normal execution of a program?
5. List down the differences that exist between the RISC and CISC.
6. PUSH and POP are stack related instructions. Write the sequence of micro-operations for PUSH and POP instructions.
7. Draw the space time diagram for pipeline of 4 segments and 6 tasks.

### Section-C

(Attempt any 3 questions, each question carries 5 marks, subparts (if any) carry equal weightage)

8. Draw the block diagram of a bus organization for seven CPU register. Also, generate the 14 bit Control word of CPU when R3 is to be selected using SELA, R5 is to be selected using SELB, output of ALU is to be stored in R6 and OPR bits are 10100 (1 is M<sub>1</sub> and 0 is LSB).
9. Define addressing mode and explain at least five with an example.
10. The Computer Instructions are classified into three major categories. Write their names and also explain the need of Data transfer instructions with at least five examples of it.
11. Pipelining is a process to speed up the execution of a program by segmentation. Demonstrate the pipeline organization on the below given example. Also compute the total clock cycles required to complete the execution.

$$A_i * B_i + C_i \quad \text{for } i = 1, 2, \dots, 7$$

### Section-D

(Attempt any one question, each question carries 10 marks, subparts (if any) carry equal weightage)

12. Write a program in assembly language that evaluates  $X = (A+B) * (C+D)$  using the following instruction formats. (2.5 each)
  - a) Three-Address Instructions
  - b) Two-Address Instructions
  - c) One-Address Instructions
  - d) Zero-Address Instructions
13.
  - a) With the help of a neat block diagram, explain the classification done by M. J. Flynn which is based on the organization of a computer system by the number of instructions and data items that are manipulated simultaneously. (6 marks)
  - b) A non-pipeline system takes 60 ns to process a task. The same task can be processed in a six-segment pipeline with a clock cycle of 20 ns. Determine the speedup ratio of the pipeline for 100 tasks. What is the maximum speedup that can be achieved? (4 marks)

April, 2023

Computer System Architecture

### Section A

- Q1  
a) (iv)  
b) (i)  
c) (iv)  
d) (iii)  
e) (i)

### Section B

Q2  
 $\boxed{V|Z|S|C} \rightarrow \text{Flag / Status Register}$

Overflow  $V = 1$ , if overflow  
 $= 0$ , if no overflow  
 $Z = 1$ , if result is zero  
 $= 0$ , if result is not zero.  
 $S = 1$ , if result is negative  
 $= 0$ , if result is positive  
 $C = 1$ , if carry is generated  
 $= 0$ , if no carry

Q3 DMA  $\rightarrow$  direct memory access

- ① Burst Mode  $\rightarrow$  Data at once  
② Cycle Stealing Mode  $\rightarrow$  in between

Q4 External interrupts  $\rightarrow$  Caused by I/O devices

Internal interrupts  $\rightarrow$  Caused by internal errors in the program.

Software interrupt  $\rightarrow$  Caused deliberately by the programmer

Q5 RISC → Reduced instruction set computer  
 CISC → Complex instruction set computer

- ① less no. of instructions
- ② 5-20 addressing modes.
- ① few instructions
- ② few addressing modes.

## ⑥ Push & Pop

PUSH →  $SP \leftarrow SP + 1$   
 $M(SP) \leftarrow DR$   
 If ( $SP = 0$ ) then ( $FULL \leftarrow 1$ )  
 $EMPTY \leftarrow 0$ .

Pop →  $EMPTY \leftarrow 0$   
 $DR \leftarrow M(SP)$   
 $SP \leftarrow SP - 1$   
 If ( $SP = 0$ ) then ( $EMPTY \leftarrow 1$ )  
 $FULL \leftarrow 0$ .

Q7 Segments ( $K$ ) = 4.

Tasks ( $n$ ) = 6

No. of clock cycles.

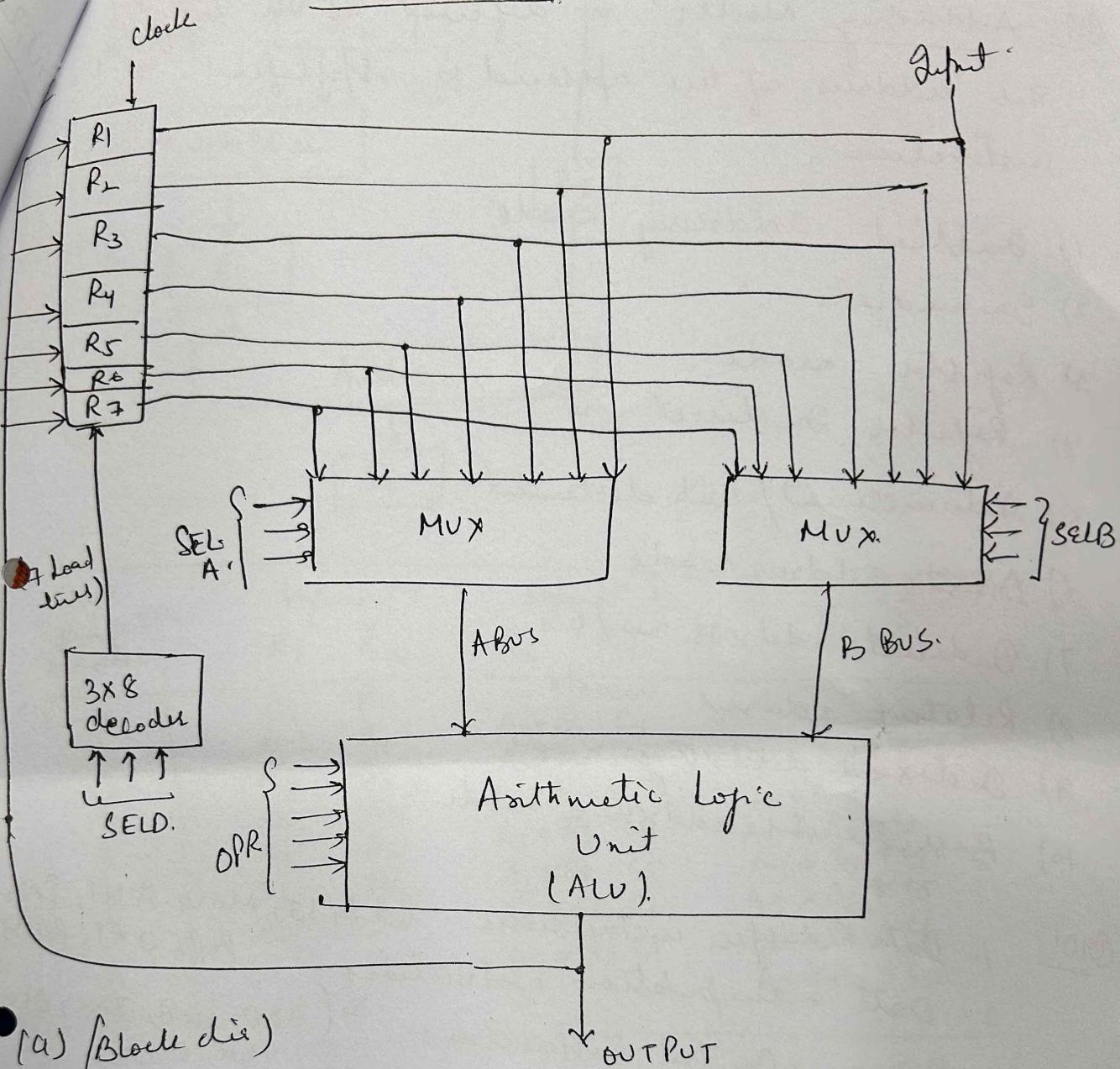
$$= k + (n-1)$$

$$= 4 + (6-1) = 9 \text{ clock cycles}$$

Space time diagram:

Segments ↓	clock cycles →								
	1	2	3	4	5	6	7	8	9
1	T <sub>1</sub>	T <sub>2</sub>	T <sub>3</sub>	T <sub>4</sub>	T <sub>5</sub>	T <sub>6</sub>			
2		T <sub>1</sub>	T <sub>2</sub>	T <sub>3</sub>	T <sub>4</sub>	T <sub>5</sub>	T <sub>6</sub>		
3			T <sub>1</sub>	T <sub>2</sub>	T <sub>3</sub>	T <sub>4</sub>	T <sub>5</sub>	T <sub>6</sub>	
4				T <sub>1</sub>	T <sub>2</sub>	T <sub>3</sub>	T <sub>4</sub>	T <sub>5</sub>	T <sub>6</sub>

Section C



(a) Block dia)

(b) Control word

3	3	3	5
SEL A	SEL B	SEL D	OPR

R<sub>3</sub> → 011 101 100 10100  
 ↑      ↑  
 R<sub>5</sub>    R<sub>6</sub>

⇒ [ 01110111010100 ] → Control word

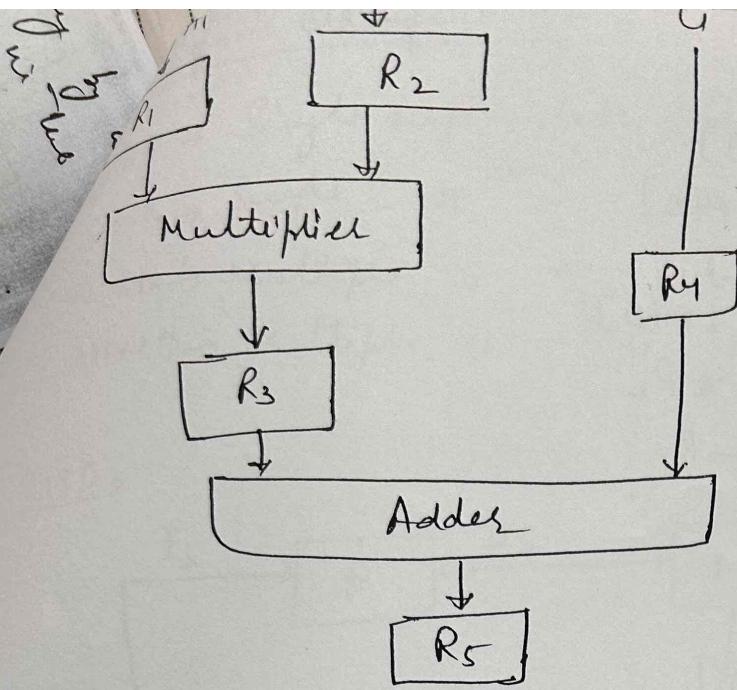
Q9) Addressing mode is defined as the way by which the address of the operand is specified in the instruction.

- 1) Implied addressing mode
- 2) Immediate " "
- 3) Register mode
- 4) Register Indirect "
- 5) Auto increment / auto decrement
- 6) Direct address mode
- 7) Indirect address mode
- 8) Relative address mode
- 9) Indexed addressing mode
- 10) Base register addressing mode

Q10)  
1. Data Transfer instructions ( LD, ST, MOVE X4H, PN )  
2. Data manipulation instructions ( Pop, PVT, PUSH )  
3. Program Control instructions ( ADD, SUB, INC/DEC  
  MUL, DIV )  
↓  
( BR, JMP, SLP, CALL, RET, CMP, TST )

Q11       $A_i \rightarrow B_i + C_i$   
                for  $i = 1, 2, 3 \dots \rightarrow$

$R_1 \leftarrow A, R_2 \leftarrow B$       Input A, B  
 $R_3 \leftarrow R_1 + R_2, R_4 \leftarrow C$       multiply and input C  
 $R_5 \leftarrow R_3 + R_4.$       Add C, to product



Clock Pulse Number	Segment 1		Segment 2		Segment R5
	R1	R2	R3	R4	
1	A <sub>1</sub>	B <sub>1</sub>	—	C <sub>1</sub>	—
2	A <sub>2</sub>	B <sub>2</sub>	A <sub>1</sub> *B <sub>1</sub>	C <sub>2</sub>	A <sub>1</sub> *B <sub>1</sub> +C <sub>1</sub>
3	A <sub>3</sub>	B <sub>3</sub>	A <sub>2</sub> *B <sub>2</sub>	C <sub>3</sub>	A <sub>2</sub> *B <sub>2</sub> +C <sub>2</sub>
4	A <sub>4</sub>	B <sub>4</sub>	A <sub>3</sub> *B <sub>3</sub>	C <sub>4</sub>	A <sub>3</sub> *B <sub>3</sub> +C <sub>3</sub>
5	A <sub>5</sub>	B <sub>5</sub>	A <sub>4</sub> *B <sub>4</sub>	C <sub>5</sub>	A <sub>4</sub> *B <sub>4</sub> +C <sub>4</sub>
6	A <sub>6</sub>	B <sub>6</sub>	A <sub>5</sub> *B <sub>5</sub>	C <sub>6</sub>	A <sub>5</sub> *B <sub>5</sub> +C <sub>5</sub>
7	A <sub>7</sub>	B <sub>7</sub>	A <sub>6</sub> *B <sub>6</sub>	C <sub>7</sub>	A <sub>6</sub> *B <sub>6</sub> +C <sub>6</sub>
8	—	—	A <sub>7</sub> *B <sub>7</sub>	—	A <sub>7</sub> *B <sub>7</sub> +C <sub>7</sub>
9	—	—	—	—	—

### Section D

Q12)  $X = (A+B) \star (C+D)$

a) Three address

ADD R1, A, B

$R_1 \leftarrow M[A] + M[B]$

ADD R2, C, D

$R_2 \leftarrow M[C] + M[D]$

MUL X, R1, R2

$M[X] \leftarrow R_1 * R_2$

b) Two address

MOV R1, A	$R_1 \leftarrow M[A]$
ADD R1, B	$R_1 \leftarrow R_1 + M[B]$
MOV R2, C	$R_2 \leftarrow M[C]$
ADD R2, D	$R_2 \leftarrow R_2 + M[D]$
MUL R1, R2	$R_1 \leftarrow R_1 * R_2$
MOV X, R1	$M[X] \leftarrow R_1$

c) One Address

LOAD A	$AC \leftarrow M[A]$
ADD B	$AC \leftarrow AC + M[B]$
STORE T	$M[T] \leftarrow AC$
LOAD C	$AC \leftarrow M[C]$
ADD D	$AC \leftarrow AC + M[D]$
MUL T	$AC \leftarrow AC * M[T]$
STORE X	$M[X] \leftarrow AC$

d) Zero Address

PUSH A	$TOS \leftarrow A$
PUSH B	$TOS \leftarrow B$
ADD	$TOS \leftarrow (A + B)$
PUSH C	$TOS \leftarrow C$
PUSH D	$TOS \leftarrow D$
ADD	$TOS \leftarrow (C + D)$
MUL	$TOS \leftarrow (C + D) * (A + B)$
Pop X	$M[X] \leftarrow TOS$

## Fujimoto's classification

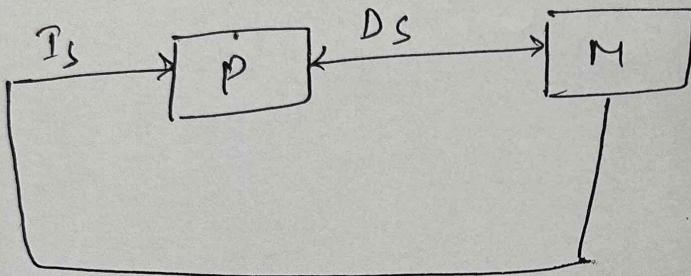
SISD  $\rightarrow$  Single instruction stream, single data stream

SIMD  $\rightarrow$  Single " " " , multiple " "

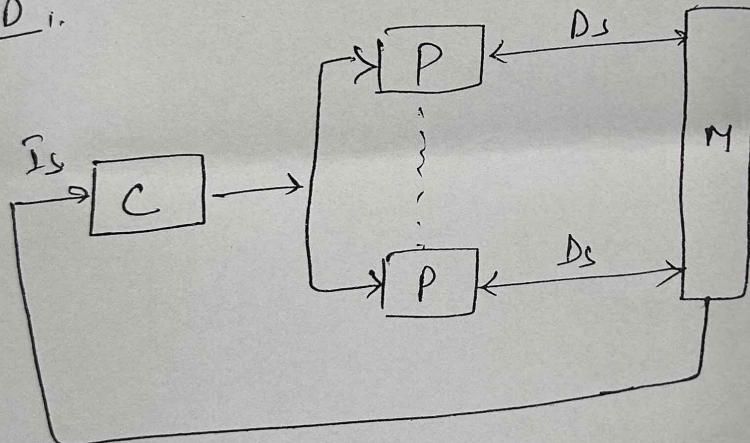
MISD  $\rightarrow$  Multiple " " " , single " "

MIMD  $\rightarrow$  Multiple " " " , multiple " "

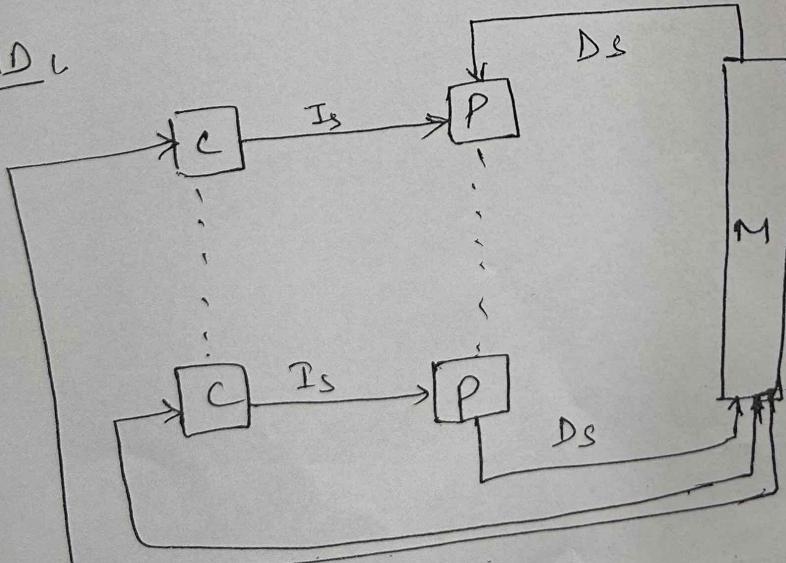
### SISD



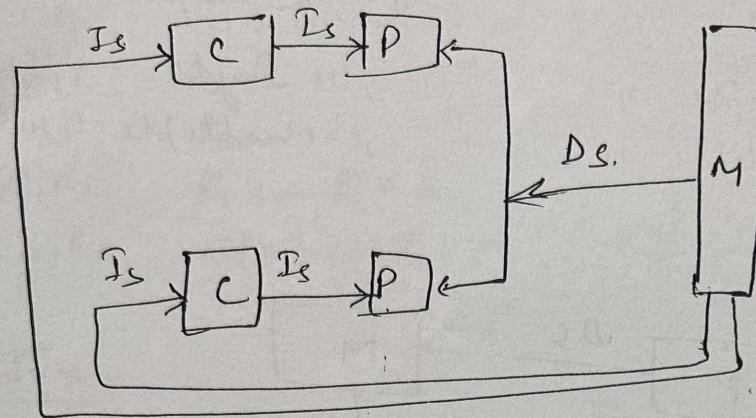
### SIMD



### MIMD



MISD



(b)  $S = n t_n / (k + n - 1) t_p$

$n = 1w$

Total time taken for  $1w$  task =  $n t_n$   
 $= 1w \times 60 = 6000 \text{ ns}$

For polypropene  
total time required =  $(k + n - 1) \cdot t_p$

Speedup ratio

$$= S = \frac{6000}{(k + n - 1)}$$