

## \* Arithmetic, Logic & Shift Micro Operations :-

In Computer Architecture, Micro-Operations are fundamental operations performed by the control unit of CPU on data stored in registers or memory.

### ① Arithmetic Micro-Operations :-

Arithmetic Micro-Operations involve the mathematical calculations on data.

Common Arithmetic Operations include :-

- |                  |                     |             |
|------------------|---------------------|-------------|
| → Addition       | basic<br>operations | → increment |
| → Subtraction    |                     | → decrement |
| → Multiplication |                     | → Shift     |
| → division       |                     |             |

→ Arithmetic Micro-Operations are performed on binary numbers within the CPU's arithmetic Logic Unit (ALU).

→ Example (a) ADD R1, R2, R3

This instruction adds the values in registers R2 and R3 and stores the result in registers R1.

Instruction	Operation
ADD R1, R2, R3	$R1 \leftarrow R2 + R3$

(b) Subtraction:-

This instruction subtracts the value in register R3 from the value in Register R2 and stores the result in register R1.

Instruction	Operation	Example
SUB R1, R2, R3	$R1 \leftarrow R2 - R3$	SUB R1, R2, R3

(c) Multiplication

JP Notes

Example:- MUL R1, R2, R3

This instruction multiplies the values in registers R2 and R3 and stores the result in register R1.

(d) Division:-

Example:- DIV R1, R2, R3

This instruction divides the value in register R2 by the value in register R3 and stores the quotient in register R1.

Increment

$R1 \leftarrow R1 + 1$  :- Increment the contents of R1 by One.

Decrement

$R1 \leftarrow R1 - 1$  :- Decrement the contents of R1 by One.

## ② Logic Micro Operation :-

→ Logic Micro - operations deal with bitwise manipulation of data.

→ Common Logic Operations include :-

→ AND

→ OR

→ NOT

→ XOR

and bit-shifting Operations.

→ They are used for tasks like data comparison, clearing specific bits and bitwise operations on binary data.

JPNotes

(a) Bitwise AND Operation:-

Consider two binary numbers,  $A = 10101011$  and  $B = 11001100$ . We want to perform bitwise AND operation on these numbers.

$$A: 10101011$$

$$\text{AND}$$

$$B: 11001100$$

$$\text{Result: } 10001000$$
(b) Bitwise OR Operation:-

The bitwise OR operation takes two binary numbers and produce a result where each bit is the logical OR of the corresponding bits in the operands.

Example:-  $A: 10101011$

$$\text{OR}$$

$$B: 11001100$$

$$\text{Result: } 11101111$$
(c) Bitwise NOT Operation:-

The bitwise NOT operation, also called bitwise complement, inverts each bit of a binary number.

Example:-  
A: 10101011  
NOT

Result:- 01010100

(d) Bitwise XOR Operation:-

The bitwise XOR operation takes two binary numbers and produces a result where each bit is the logical XOR of the corresponding bits in the operands.

Example:-

A: 10101011  
XOR:  
B: 11001100

Result:- 01101111

Notes by  
@JP Notes