

CS & IT ENGINEERING



COMPUTER ORGANIZATION AND ARCHITECTURE

Basics of COA

Lecture No.- 02

By- Vishvadeep Gothi sir



Recap of Previous Lecture



Topic

Architecture vs Organization

Topic

Numbers & Data in Computers

Topic

Components of Computer

CPU
mem.
I/O

Topic

System Buses

Topic

Types of Buses

Topics to be Covered



Topic

CPU Registers

Topic

Types of Architecture (CPU)

Topic

Program Counter

Topic

Instruction Register

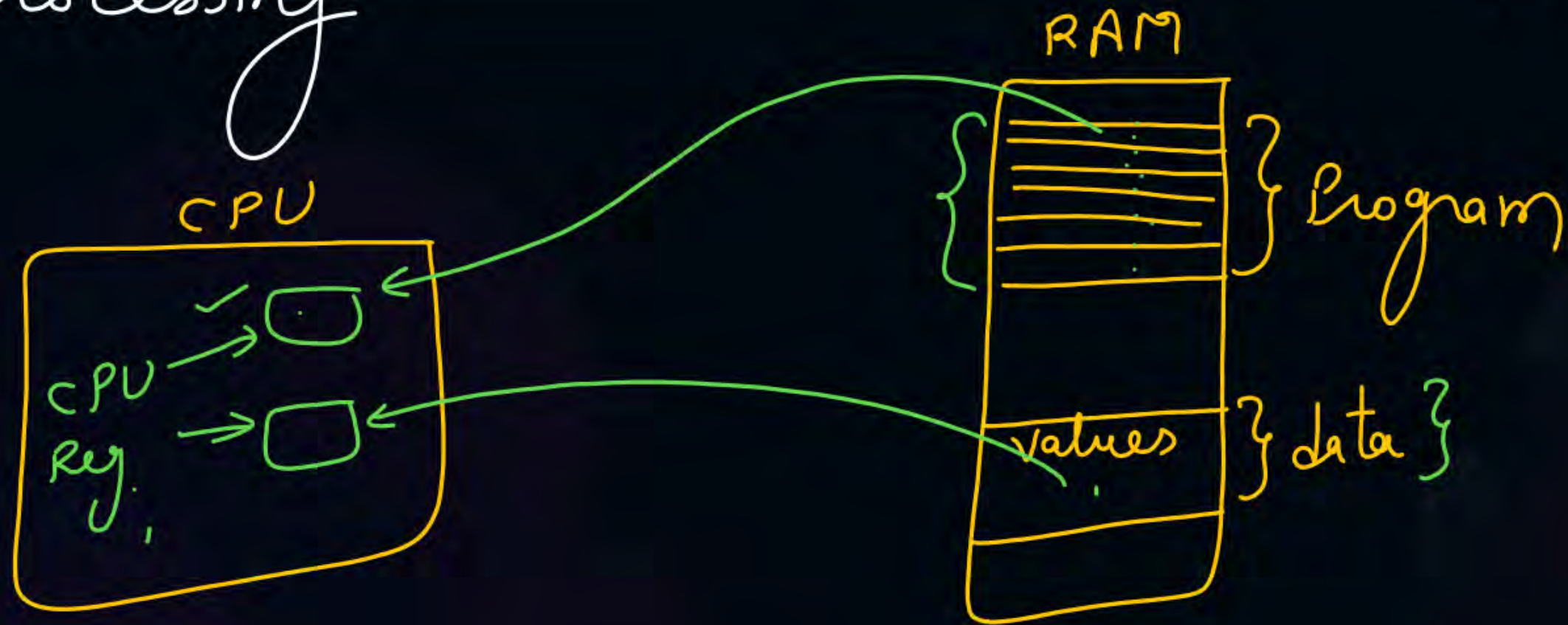
Topic

Stack Pointer



Topic : CPU Registers

small memories inside CPU to carry out processing





Topic : CPU Registers



CPU Register

- General Purpose Registers (GPRs) \Rightarrow used for any work.
(denoted by $R0, R1, R2, \dots$)
- Special Purpose Registers
 \Downarrow
used for specific purpose



Topic : CPU Registers



CPU Register

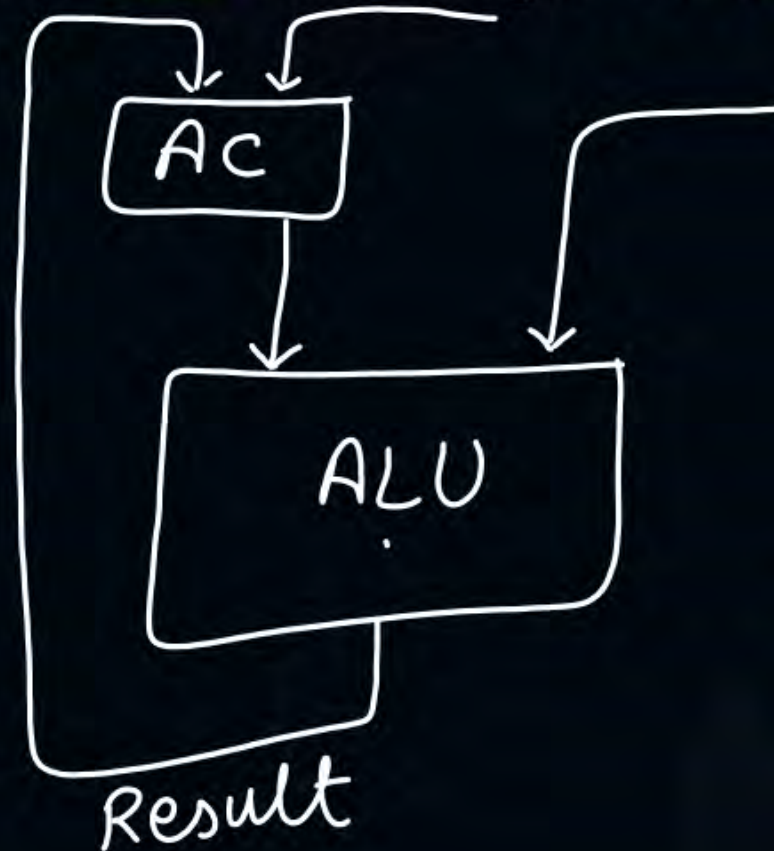
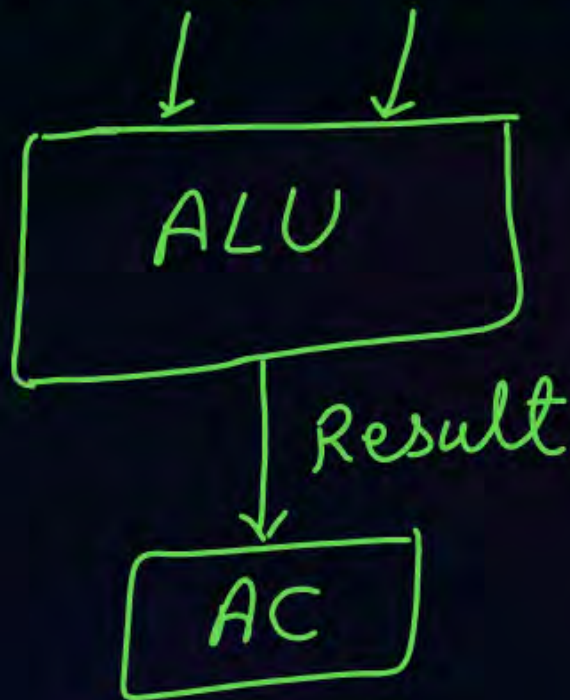
- General Purpose Registers (GPRs)
- Special Purpose Registers
 - ✓ 1. Accumulator (AC)
 - ✓ 2. Program Counter (PC)
 3. Instruction Register (IR)
 4. Stack Pointer (SP)
 5. Flag Register / Program Status Word (PSW) / *status Register*
 6. Address Register (AR) / Memory Address Register (MAR)
 7. Data Register (DR) / Memory Data Register (MDR) / MBR (*Memory Buffer Reg.*)



Topic : Accumulator

- Used to store result of ALU and sometimes ~~on~~^{one} of the operands^s for ALU too.

input operands



$a + b$

$AC \leftarrow a$

$AC \leftarrow AC + b$



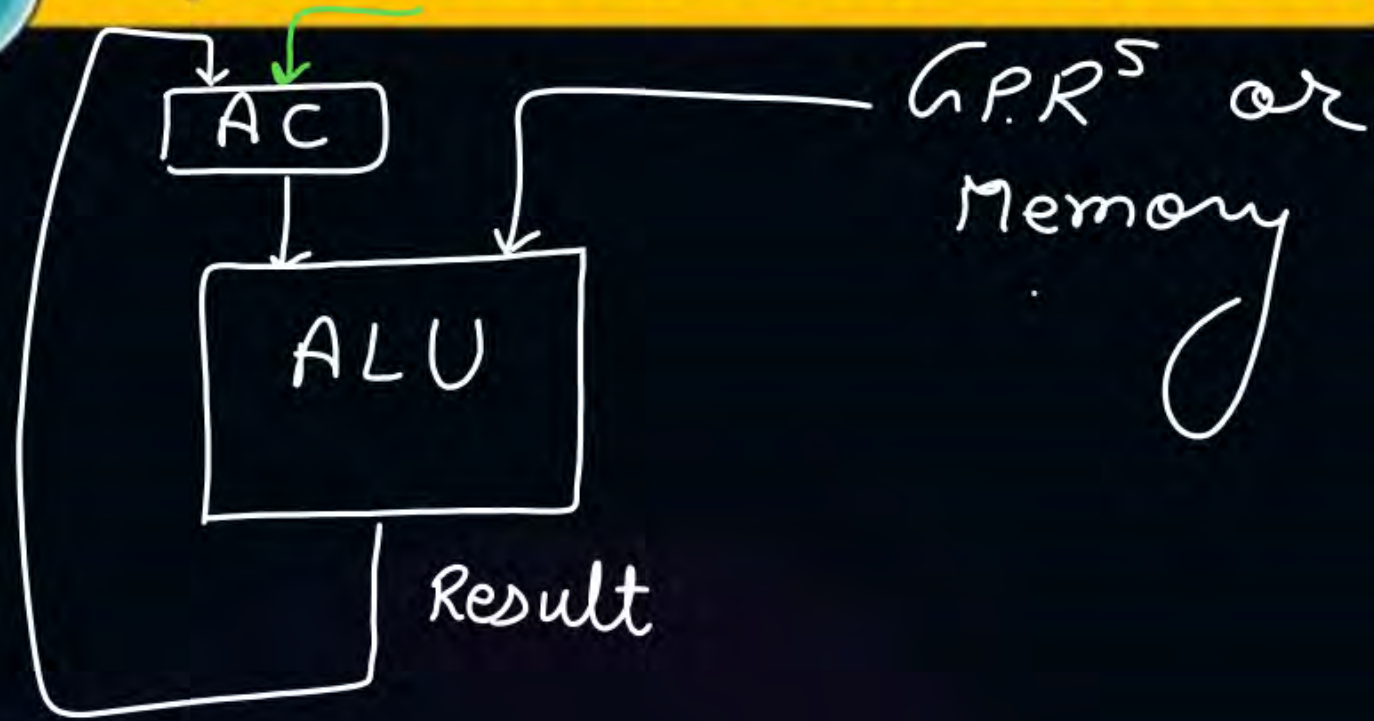
Topic : Types of Architecture

Based on ALU input:

- AC-Based Architecture (single-Ac based architecture)
- Register Based Architecture
- Register-Memory Based Architecture
- Complex System Architecture
- Stack Based Architecture



Topic : AC-Based Architecture



a, b, c, d are in memory

$$\frac{(a+b)}{R1} * \frac{(c+d)}{AC}$$

$AC \leftarrow a$

$AC \leftarrow AC + b$

$R1 \leftarrow AC$

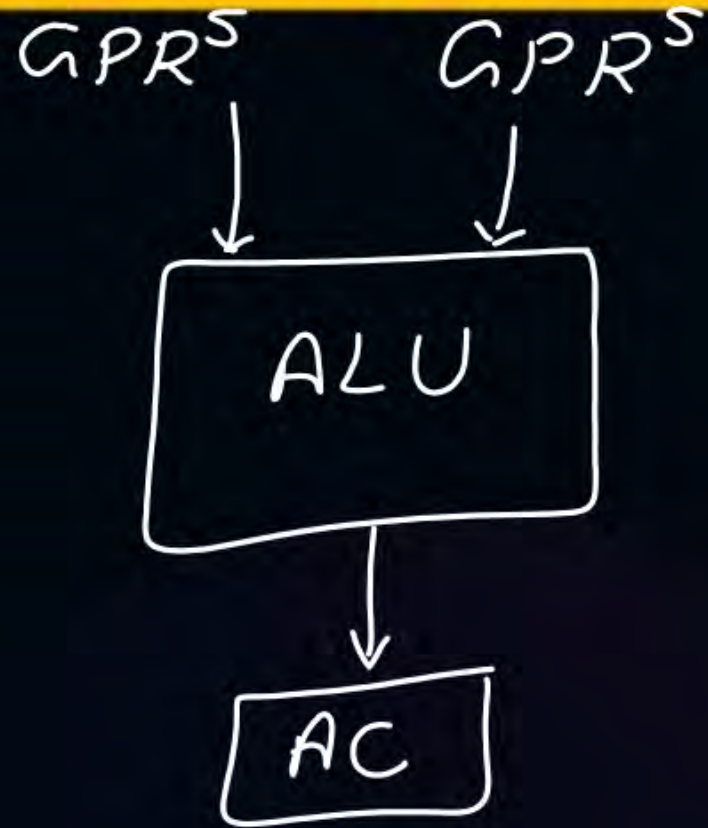
$AC \leftarrow c$

$AC \leftarrow AC + d$

$AC \leftarrow AC * R1$



Topic : Register-Based Architecture



$$\frac{(a+b)}{R1} * \frac{(c+d)}{R2}$$

a, b, c, d
are in memory

$$R1 \leftarrow a$$

$$R2 \leftarrow b$$

$$AC \leftarrow R1 + R2$$

$$R1 \leftarrow AC$$

$$R2 \leftarrow c$$

$$R3 \leftarrow d$$

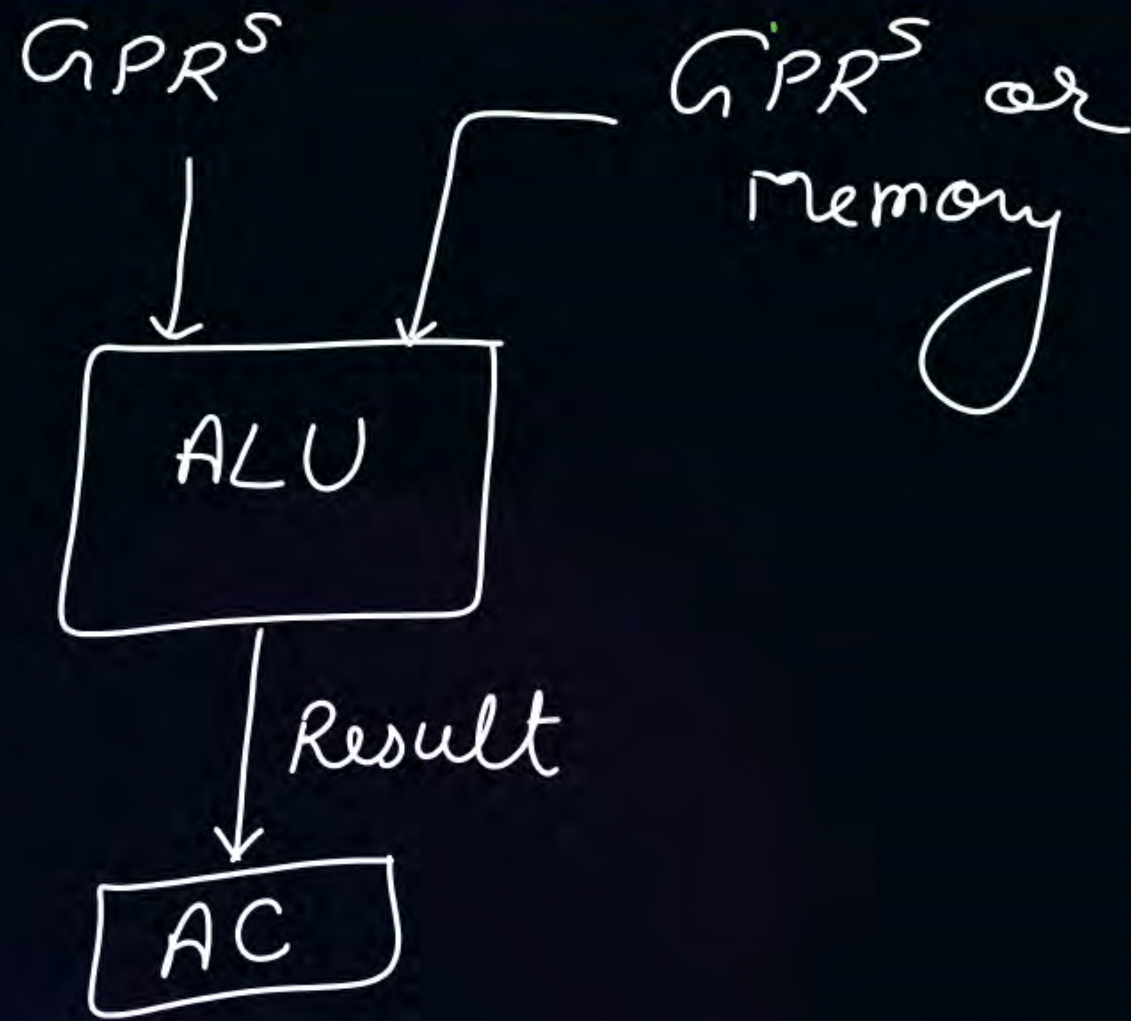
$$AC \leftarrow R2 + R3$$

$$R2 \leftarrow AC$$

$$AC \leftarrow R1 * R2$$



Topic : Register-Memory Based Architecture



$$\frac{(a+b)}{R1} * \frac{(c+d)}{R2}$$

a, b, c, d are in memory

$$R1 \leftarrow a$$

$$AC \leftarrow R1 + b$$

$$R1 \leftarrow AC$$

$$R2 \leftarrow c$$

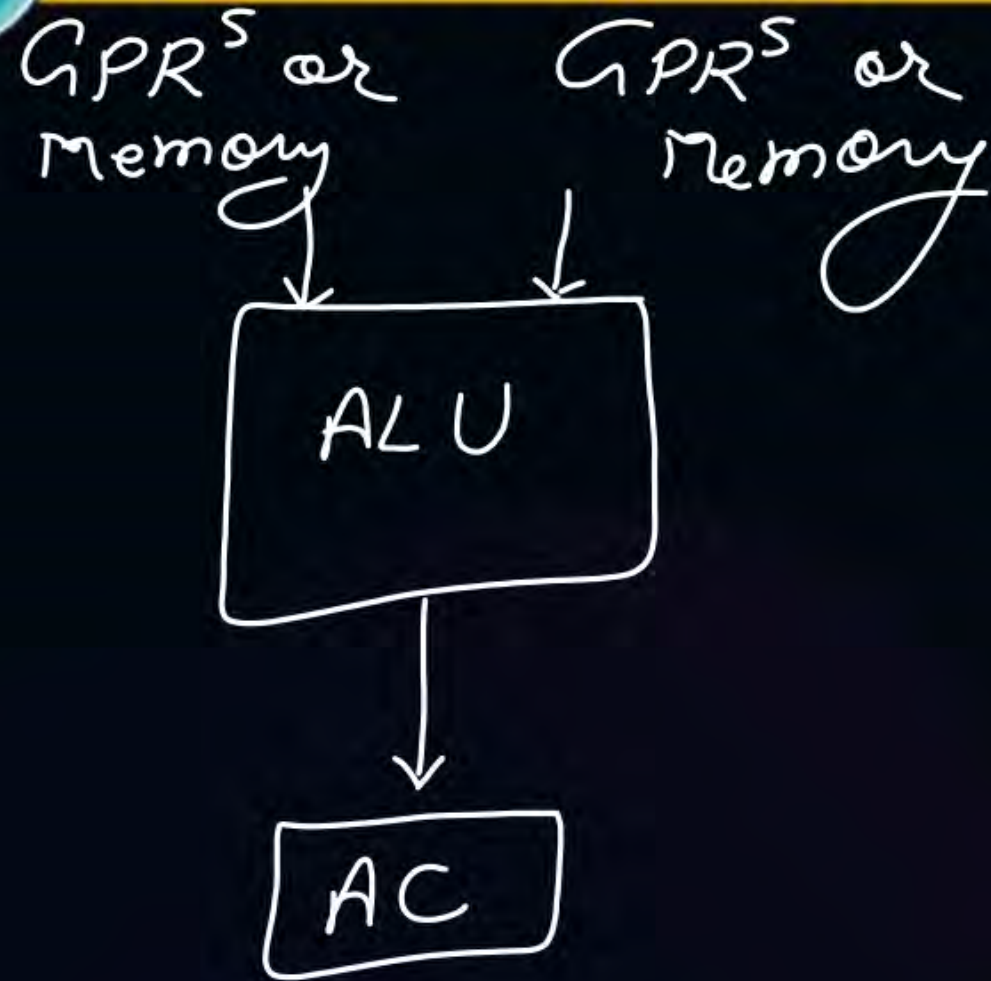
$$AC \leftarrow R2 + d$$

$$R2 \leftarrow AC$$

$$AC \leftarrow R1 * R2$$



Topic : Complex System Architecture



$$(a+b) * (c+d)$$

$$AC \leftarrow a + b$$

$$R1 \leftarrow AC$$

$$AC \leftarrow c + d$$

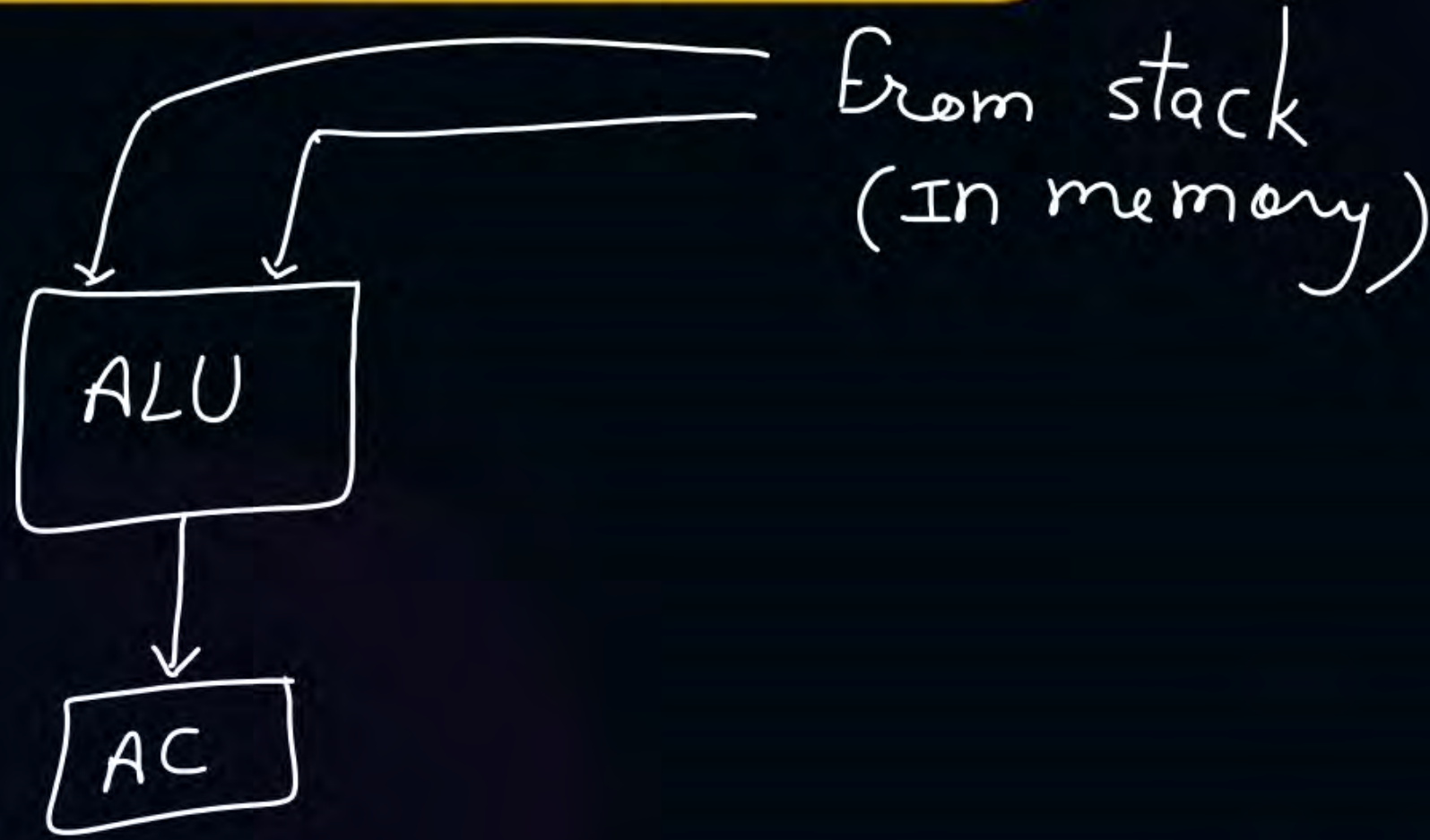
$$R2 \leftarrow AC$$

$$AC \leftarrow R1 * R2$$



Topic : Stack-Based Architecture

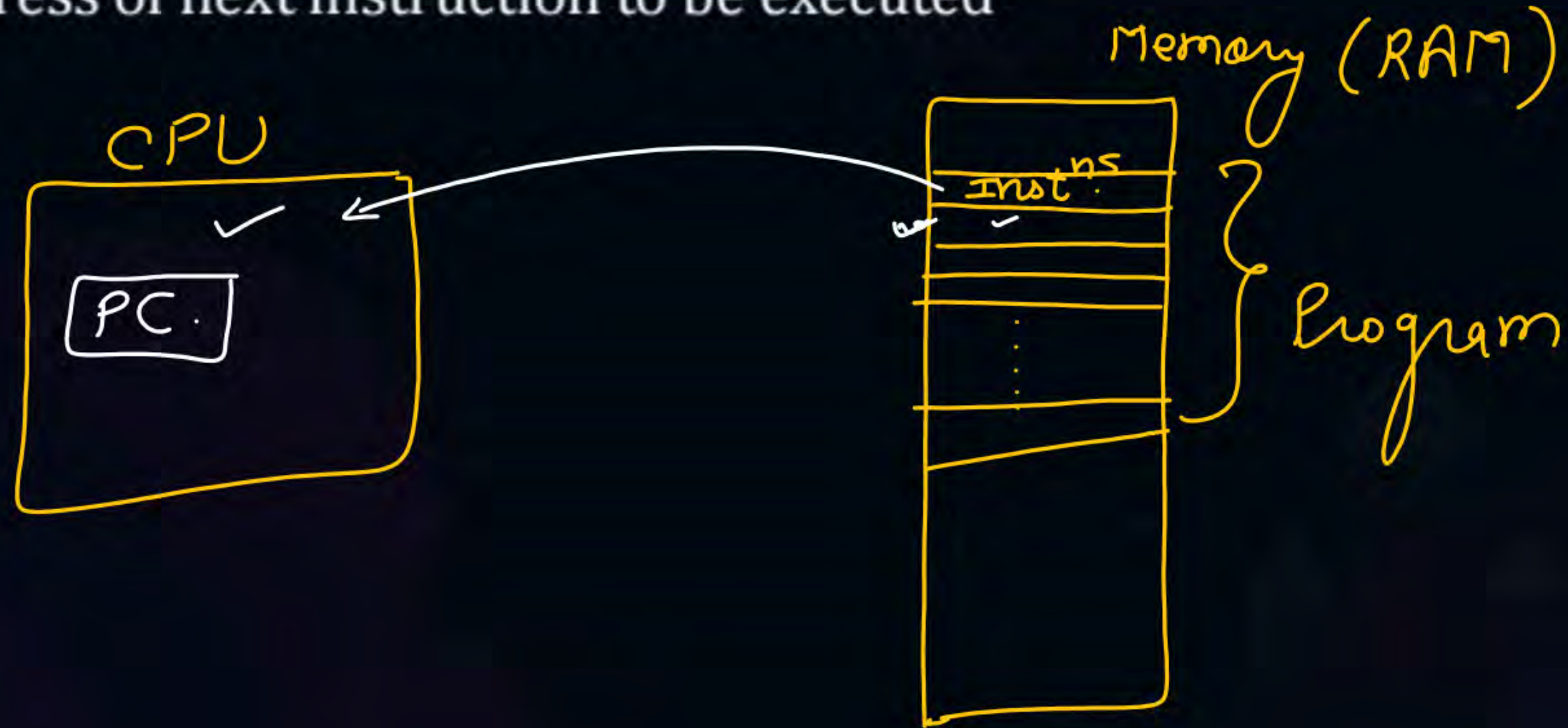
(not used now a days)





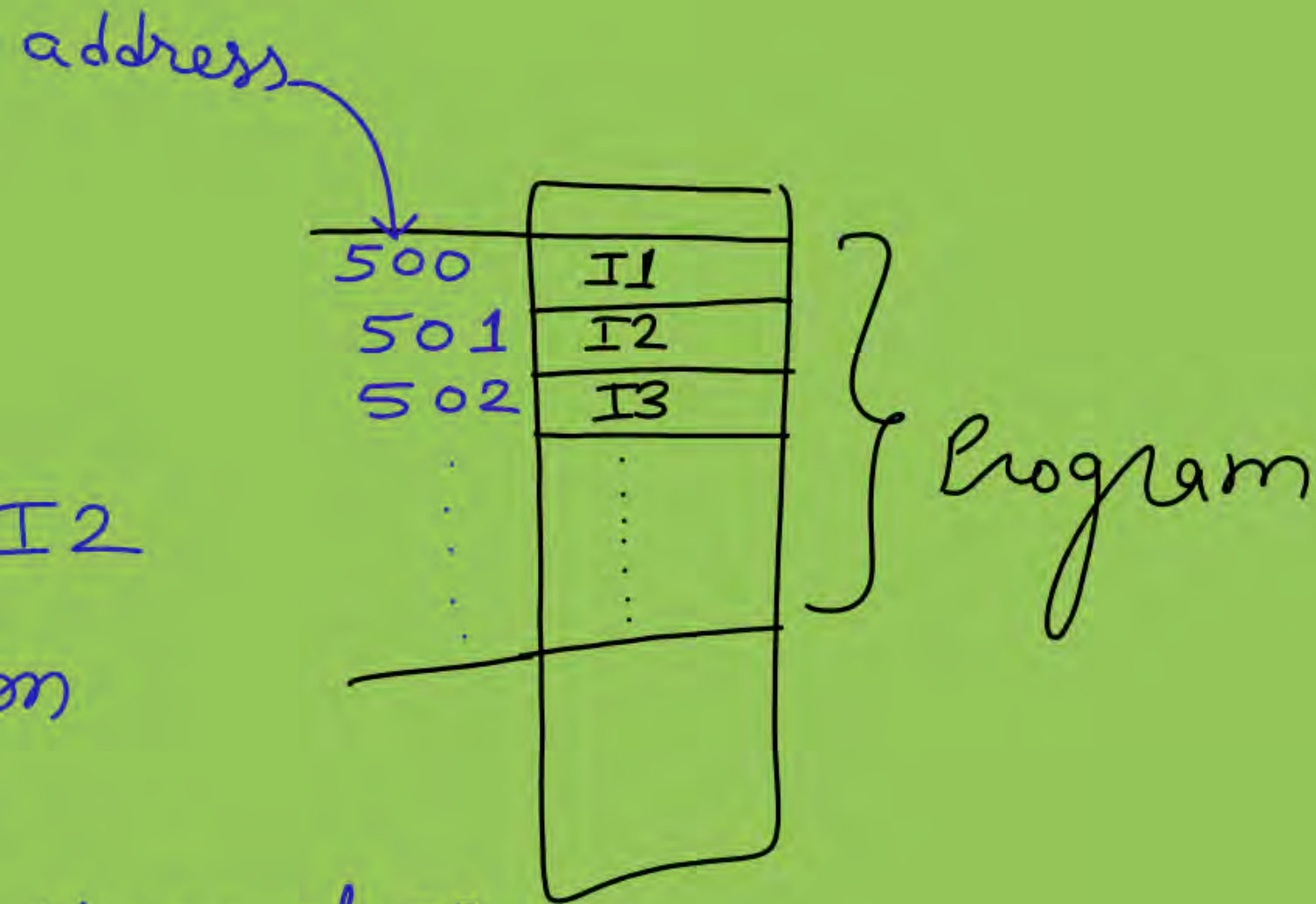
Topic : Program Counter

- Stores address of next instruction to be executed



if instruction I2
is in execution
in CPU, then

PC stores \Rightarrow address of I3
 \Downarrow
502





Topic : Instruction Register

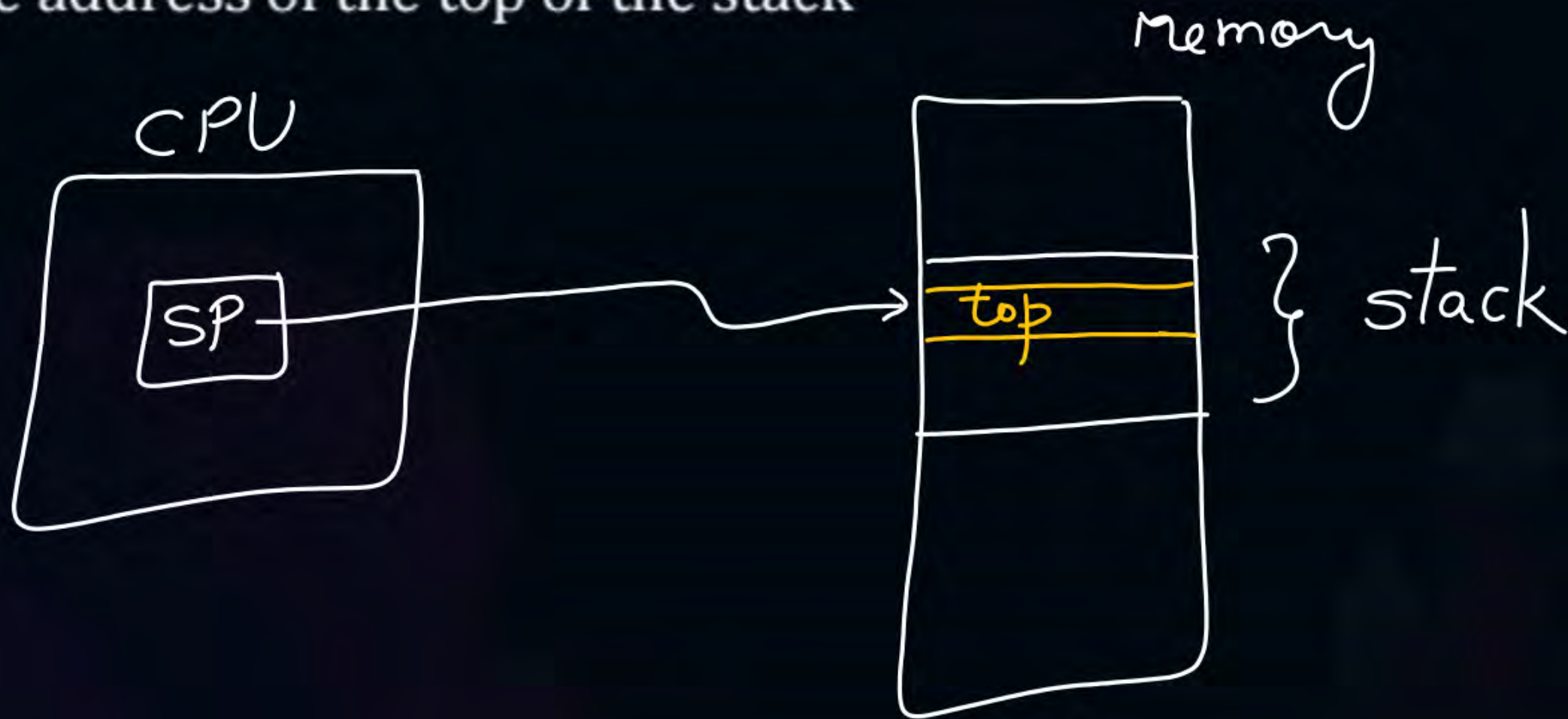
- Stores the current instruction to be executed



Topic : Stack Pointer



- Stores the address of the top of the stack





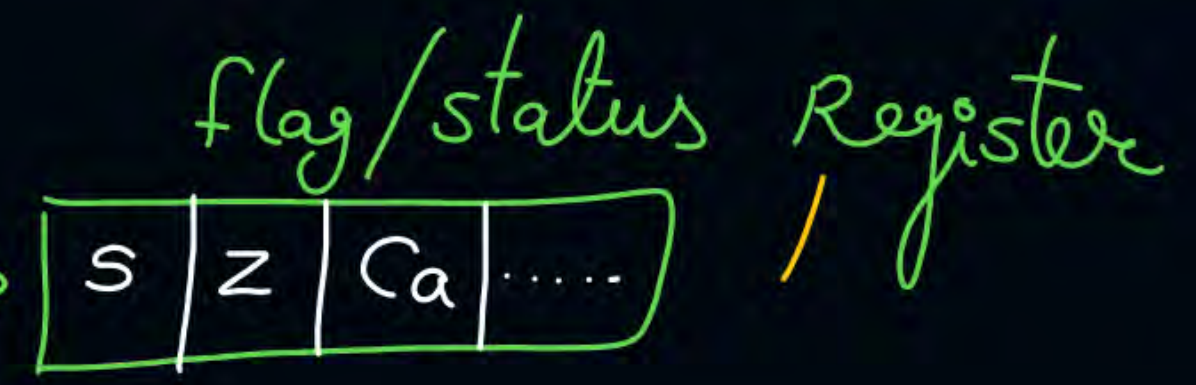
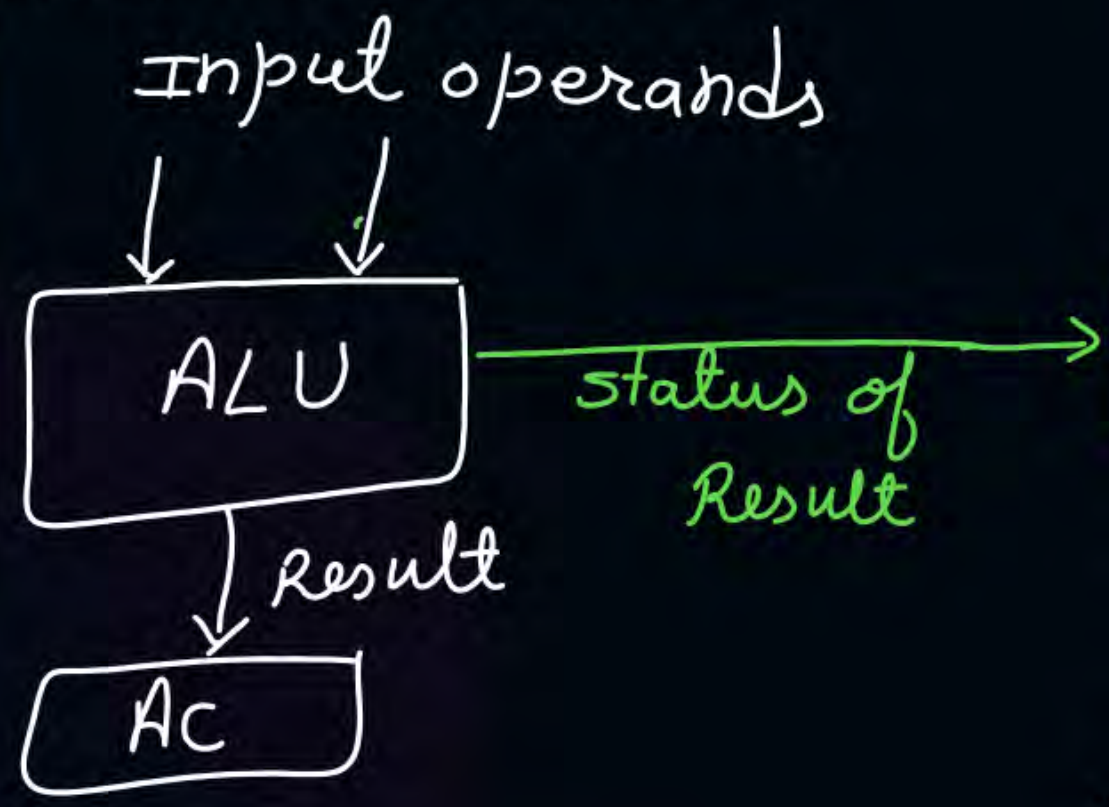
Topic : Flag or Status Register

$$a > b \Rightarrow \frac{a-b}{\downarrow}$$

result is +ve and non-zero

- Stores the status of the ALU result

it is used for conditions checking



S \Rightarrow Sign
Z \Rightarrow Zero { 1 \Rightarrow result is zero
 0 \Rightarrow result is non-zero
Ca \Rightarrow Carry



Topic : Address Register or MAR

- Used to send address to memory



Topic : Data Register or MDR

- Used to send data to memory
- And to receive data from memory

#Q. A CPU has 4 bytes instructions. A program (Instructions I_1 to I_{200}) starts at address 200 (in decimal). Find the address of following instructions:

1. I_1
2. I_5
3. I_{120}

#Q. A CPU has 4 bytes instructions. A program (Instructions I_1 to I_{200}) starts at address 500 (in decimal). What should be the PC value when instruction I_6 will be executing in CPU?

#Q. A CPU has 4 bytes instructions. A program (Instructions I_1 to I_{200}) starts at address 500 (in decimal). What should be the PC value when instruction i will be executing in CPU?



2 mins Summary

Topic

CPU Registers

Topic

Types of Architecture

Topic

Program Counter

Topic

Instruction Register

Topic

Stack Pointer

Tomorrow

AR, }
DR } ✓

[Memory,
memory address,
PC,]



Happy Learning

THANK - YOU