CS & IT ENGINEERING

COMPUTER ORGANIZATION AND ARCHITECTURE

Instruction & Addressing Modes

Lecture No.- 05

By- Vishvadeep Gothi sir

Recap of Previous Lecture

Topic Effective Address

Comp. ⇒ add. of operand

Branch ⇒ Target add.

Topic Branch Instruction

Topic Instruction Cycle

Topic Fetch Cycle & Execution Cycle

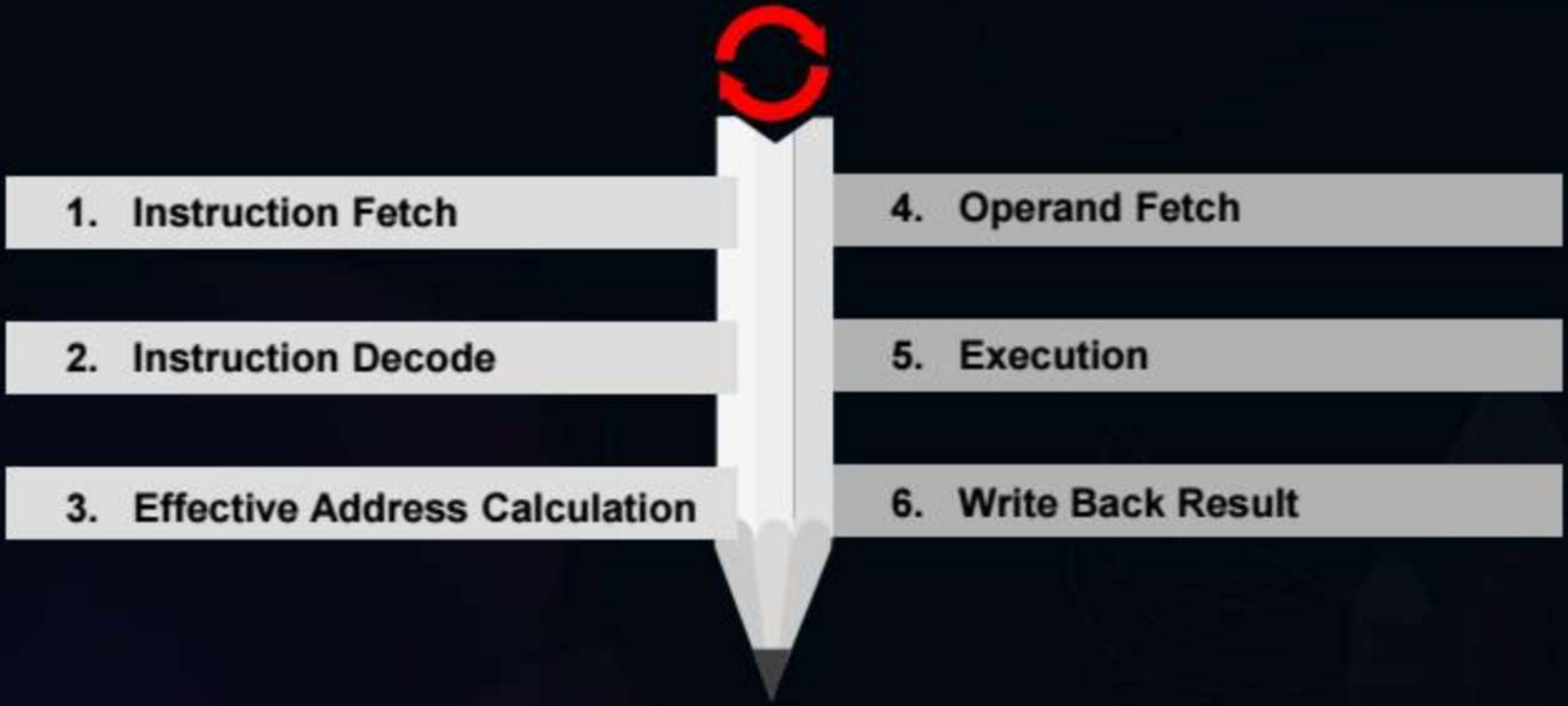
Topic Addressing Modes

Topics to be Covered

Topic Addressing Modes

Topic Types of Addressing Modes

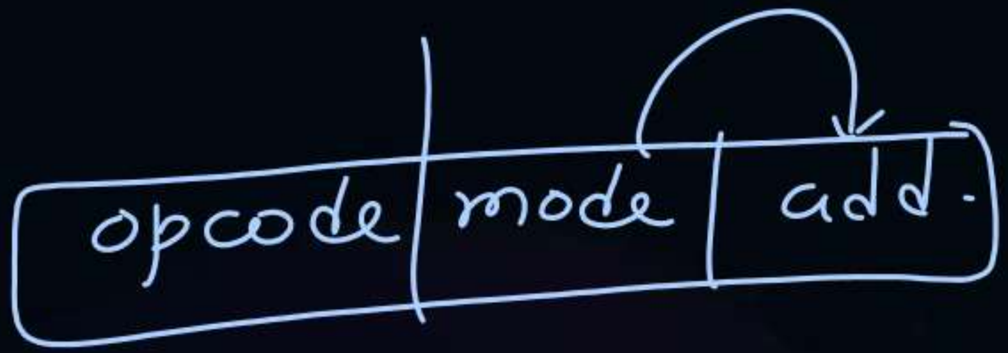
Topic : Instruction Cycle



1. Instruction Fetch
2. Instruction Decode
3. Effective Address Calculation
4. Operand Fetch
5. Execution
6. Write Back Result

Topic : Addressing Modes

- It specifies how and from where the operands are obtained for an instruction



Topic : Implied Mode

The opcode definition itself defines the operand

Opcode	Mode	~~Address~~

operation
+
operand

ex:- opcode ⇒ Increment Accumulator
(INCA)

Topic : Immediate Mode

→ This mode is used to initialize Reg^s with constant value or to use constants in inst^{ns}.

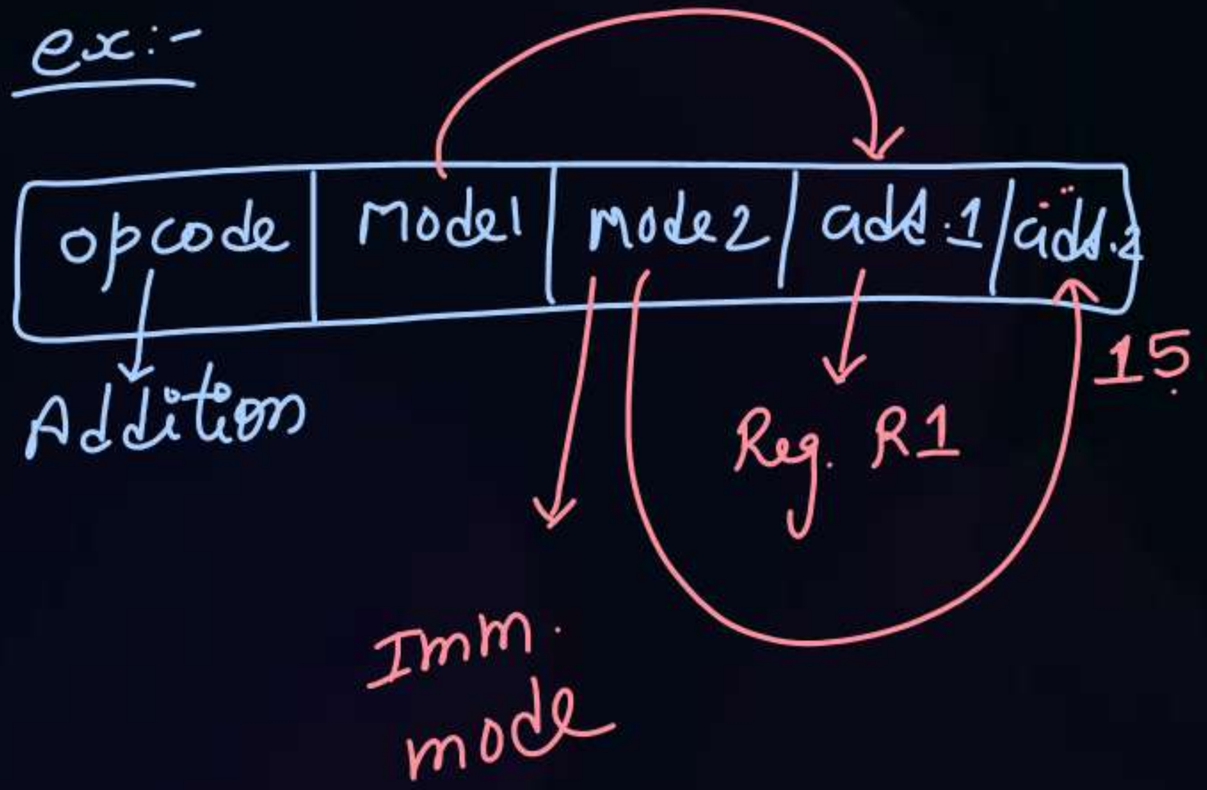
The address field of instruction specifies the operand value

Opcode	Mode	Address

Immediate

→ operand value

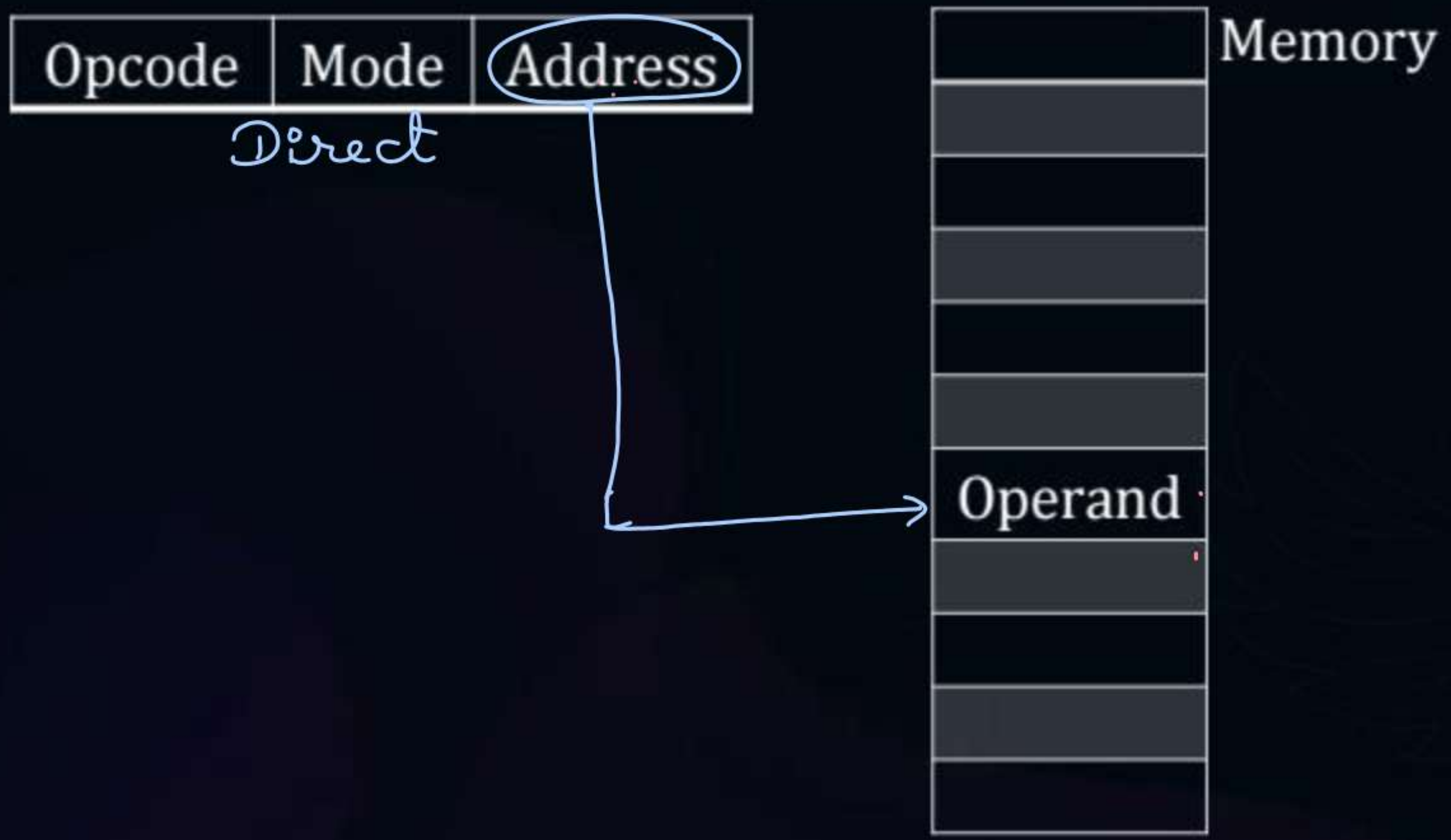
ex:-



$R1 \leftarrow R1 + \underline{15}$

Topic : Direct Mode /absolute

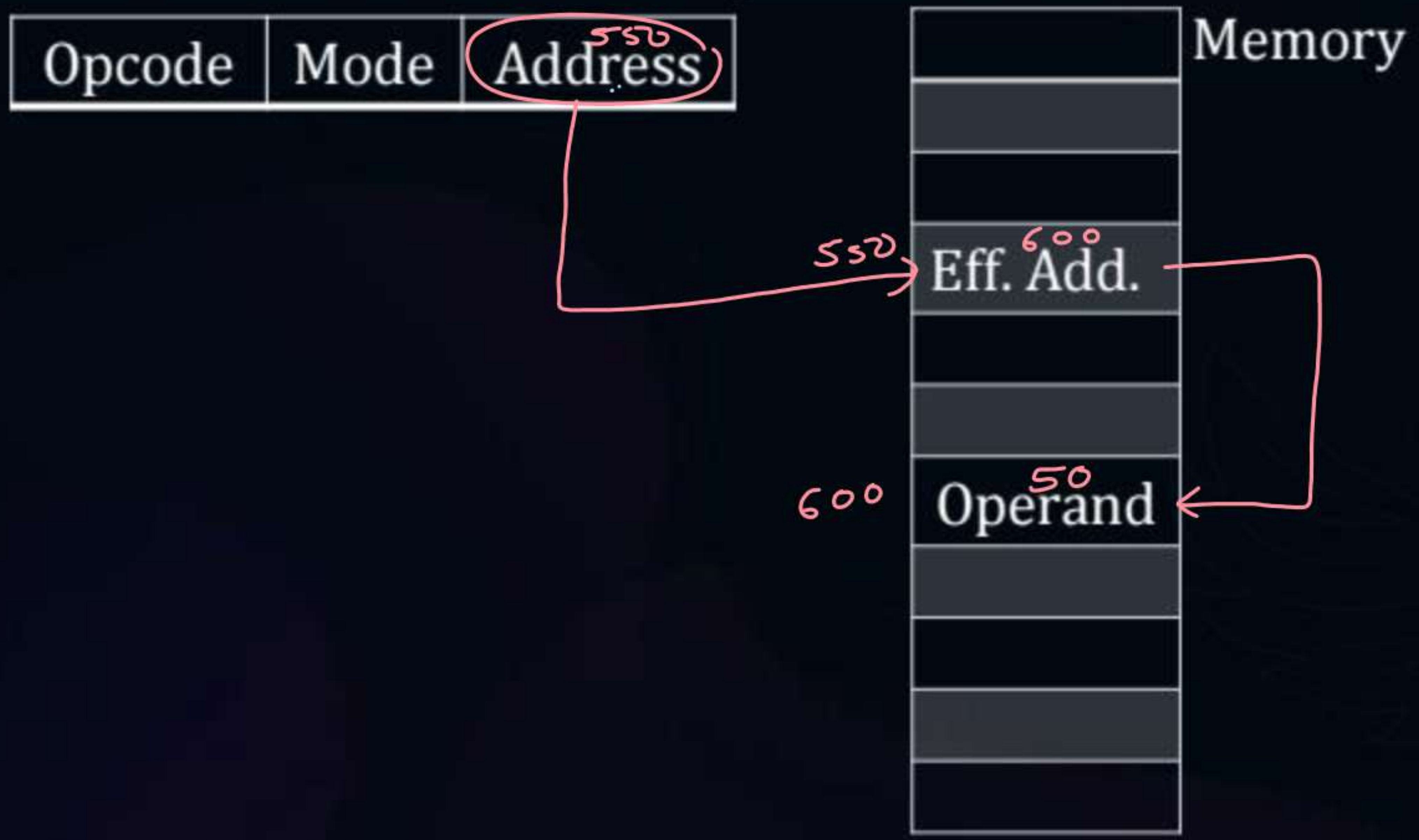
The address field of instruction specifies the effective address



Topic : Indirect Mode

→ It is used to implement pointers

The address field of instruction specifies the add. of effective address

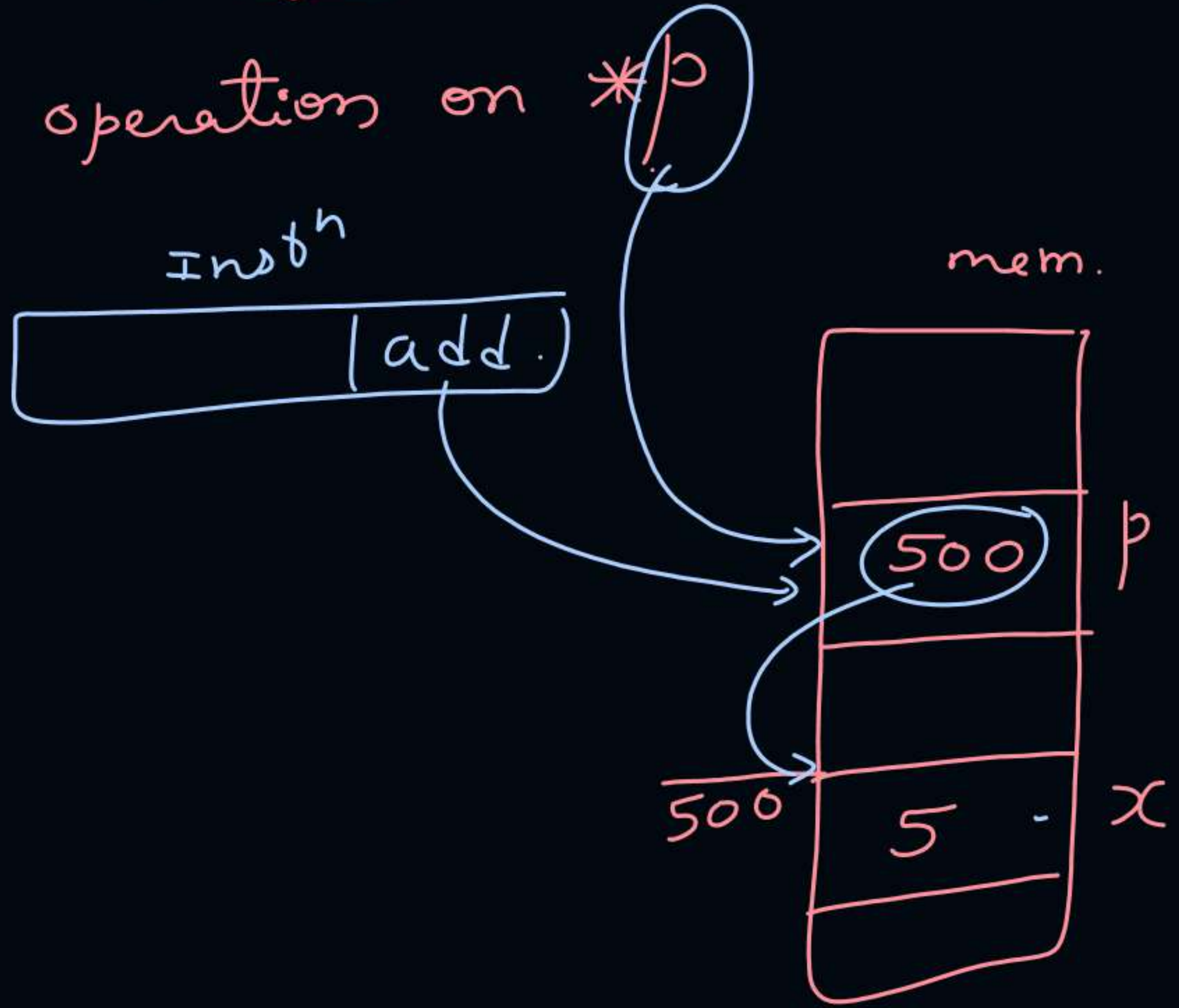


Time to get operand

= 2 * mem. access time

In c-prog.:-

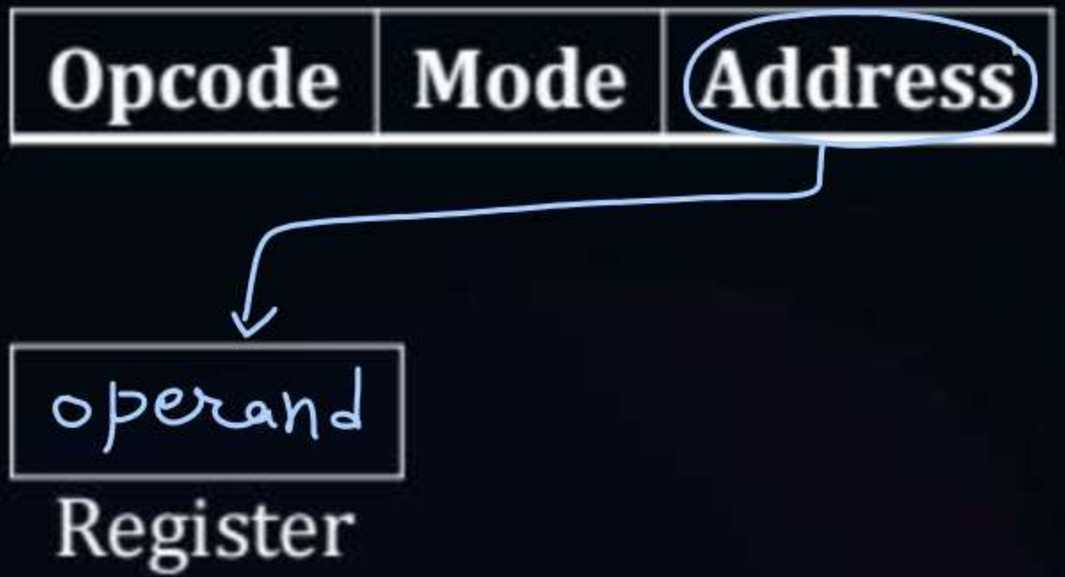
operation on *p



```
int x = 5;
int *p;
p = &x;
```

Topic : Register Mode / Reg. – Direct Mode

The address field of instruction specifies a register which holds operand



Opcode	Mode	Address

operand

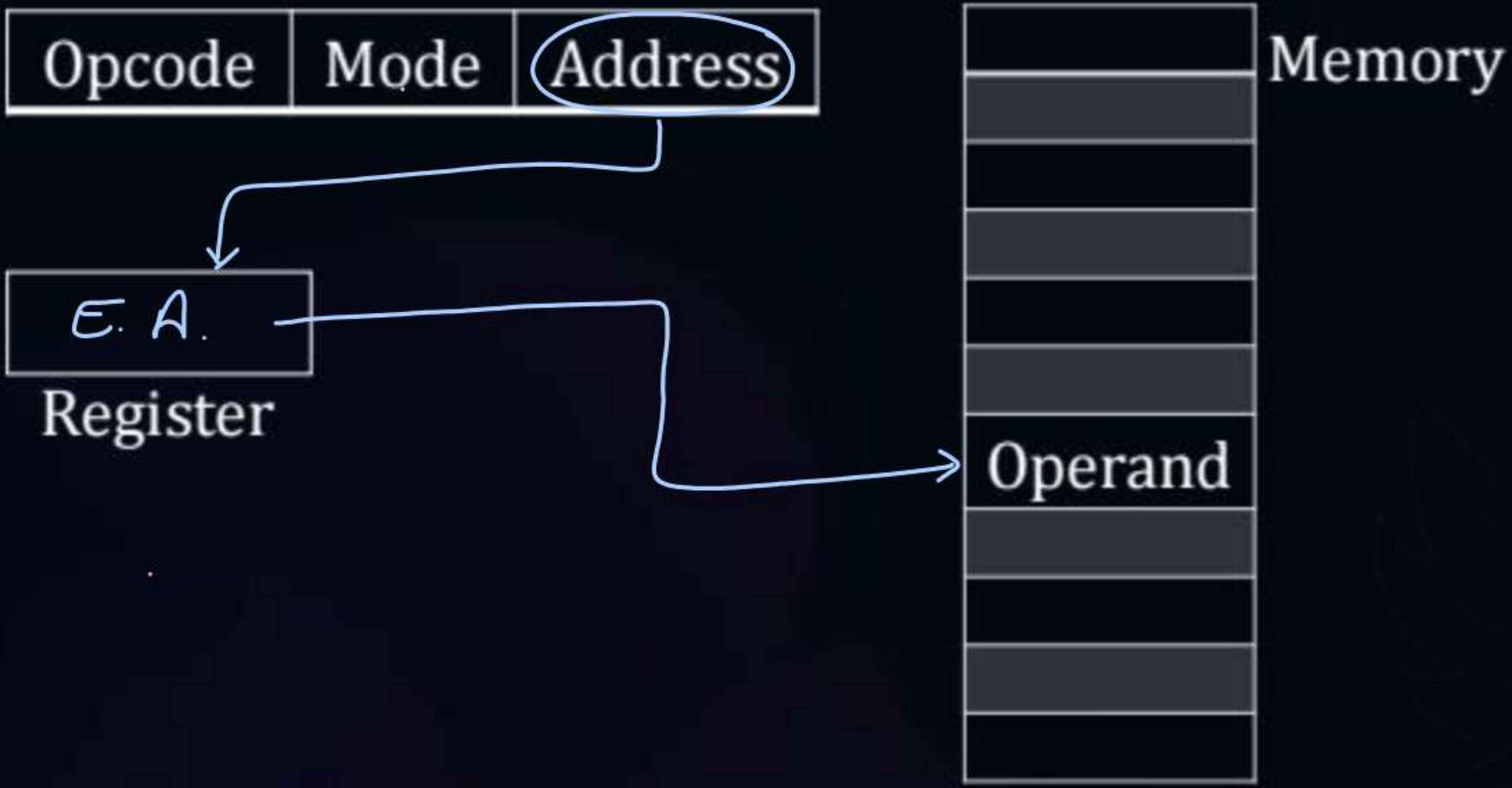
Register

Topic : Register Indirect Mode

→ It is used to shorten instʰ length.

effective add.

The address field of instruction specifies a register which holds ~~operand~~



Direct mode → Time to obtain operand = 1 mem. access time

opcode	mode	add.
x	y	33

$\Rightarrow (x + y + 33)$ bits

Reg. Indirect mode

opcode	mode	add.
x	y	7

$(x + y + 7)$ bits

→ Time to obtain operand = 1 Reg. Access time + 1 mem. access time

ex:- Assume a Computer

RAM = 8GB

= 8G X 1B

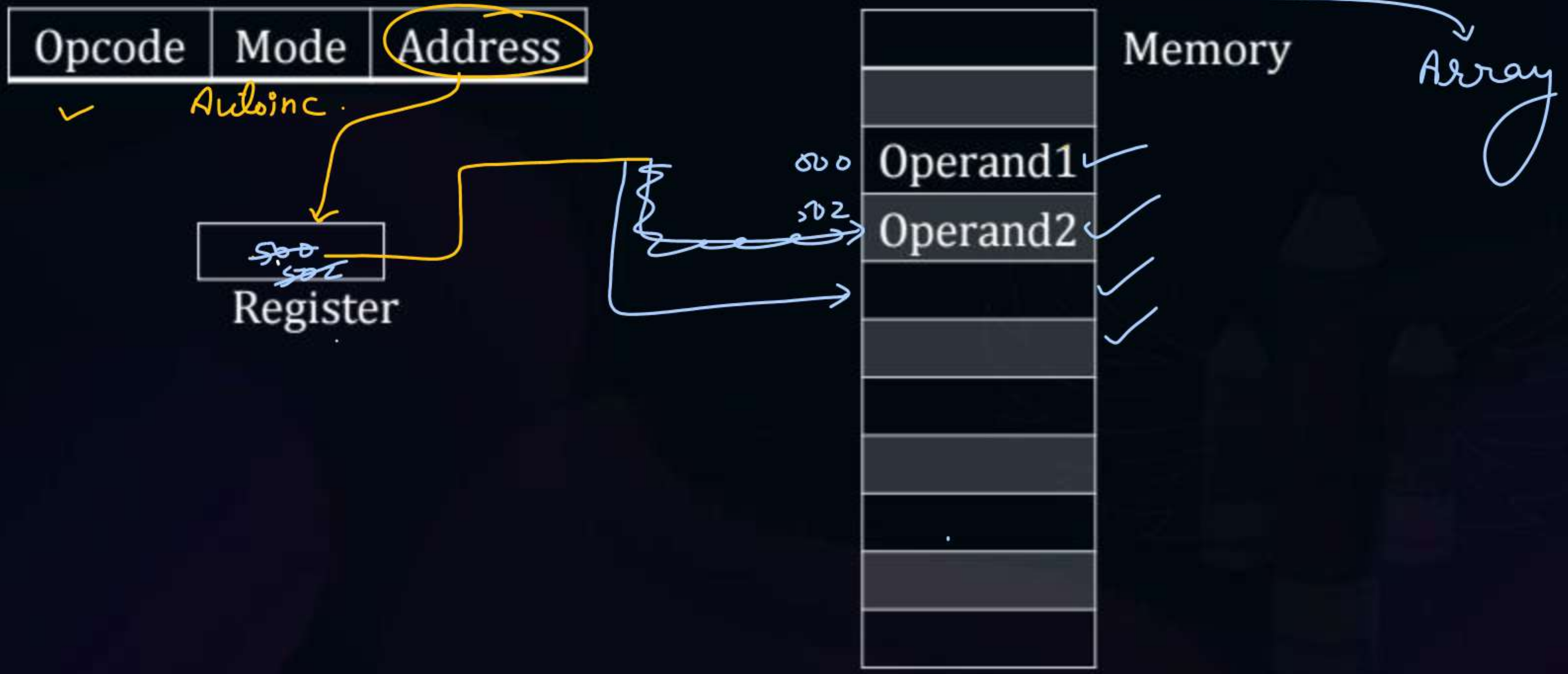
mem. add. = 33 bits

GPR^S = 128

Reg. in inst^n = 7-bits

Topic : Autoincrement/Autodecrement Mode

Variant of register indirect mode, in which content of register is automatically incremented or decremented to access a table of content sequentially.



Opcode	Mode	Address

Register

Memory

Operand1

Operand2

for micro-processor designs :- (default)

Auto inc. mode ⇒ Post increment

Auto dec. mode ⇒ Pre decrement

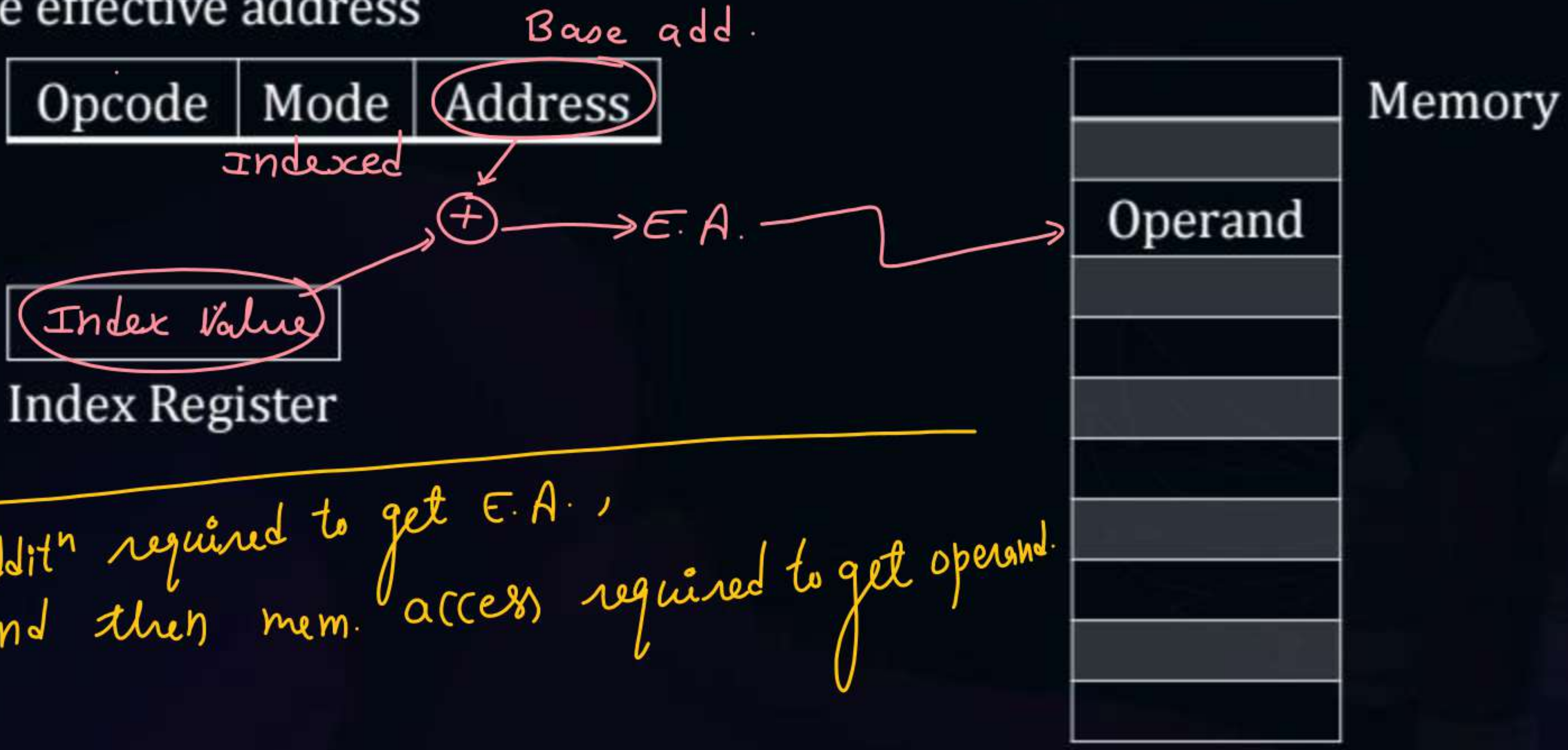
Amount of increment or decrement depends on size of data value accessed.

Topic : Indexed Mode

→ used to access an element of array

or Index-Reg. mode

Address part of instruction (base address) is added to index register value to get the effective address

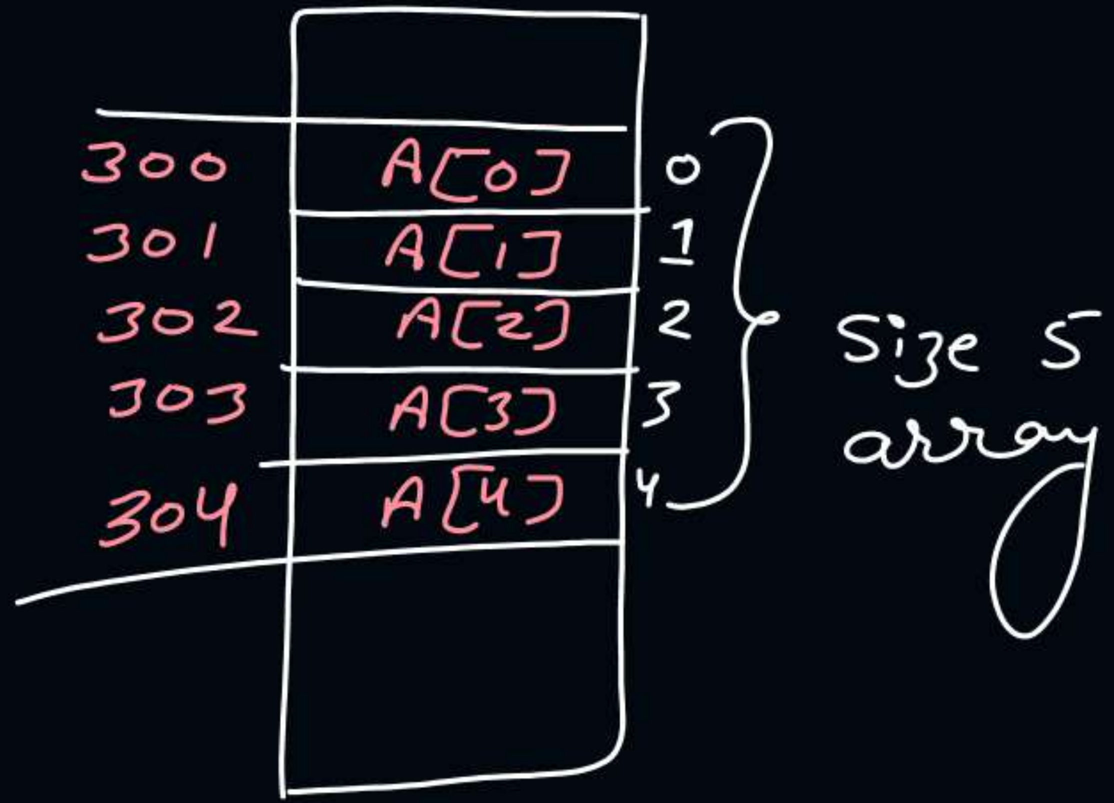


Additn required to get E.A.,
and then mem. access required to get operand.

ex:-

array in mem.

char A[5];



x = A[4];

add. of A[3] = 300 + 1 * 3

= 300 + 3

= 303

(A + 3) in C-programming

add. of A[i] => A + i * size

base add.

index of element

size of each element in inst^n

index value of addressing mode

E.A. (add. of an element of array) = Base address + index value

Base address ↓ from add. part of inst^n

index value ↓ from index Reg.

formula to get effective add. in Indexed mode :-

$$\boxed{\text{E.A.} = \text{Address part of inst}^h + \text{Index Register value}}$$

Implementation of Indexed mode

Index Reg. is a special purpose Reg.

opcode	mode	Add.

Then is no any special purpose Index Reg.

opcode	mode	Index Reg.	Add.

Identifier

⇓

it specifies a GPR, which is going to be used as index Reg.

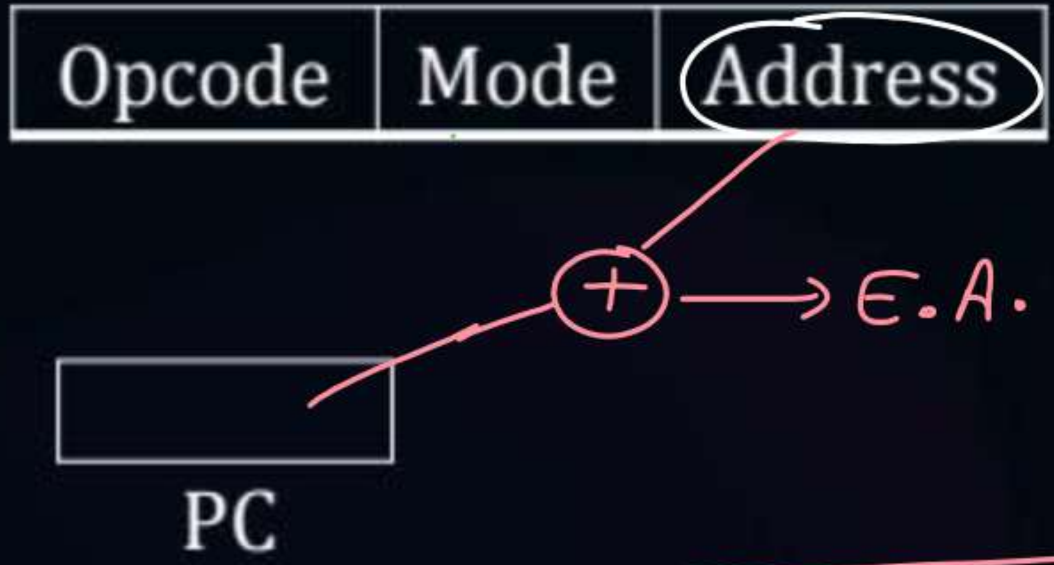
Topic : PC-Relative Mode

or (Position Independent Mode)

↳ This mode is used for branch type of inst^{ns} → intra-segment jumping

Address part of instruction (offset) is added to PC register value to get the effective address

Relative add. to skip (offset)



E.A. = Add. part of inst^{n} + PC value

Assuming inst^n I2 is executing in CPU,

PC = 204

CPU decodes I2 as branch inst^n,

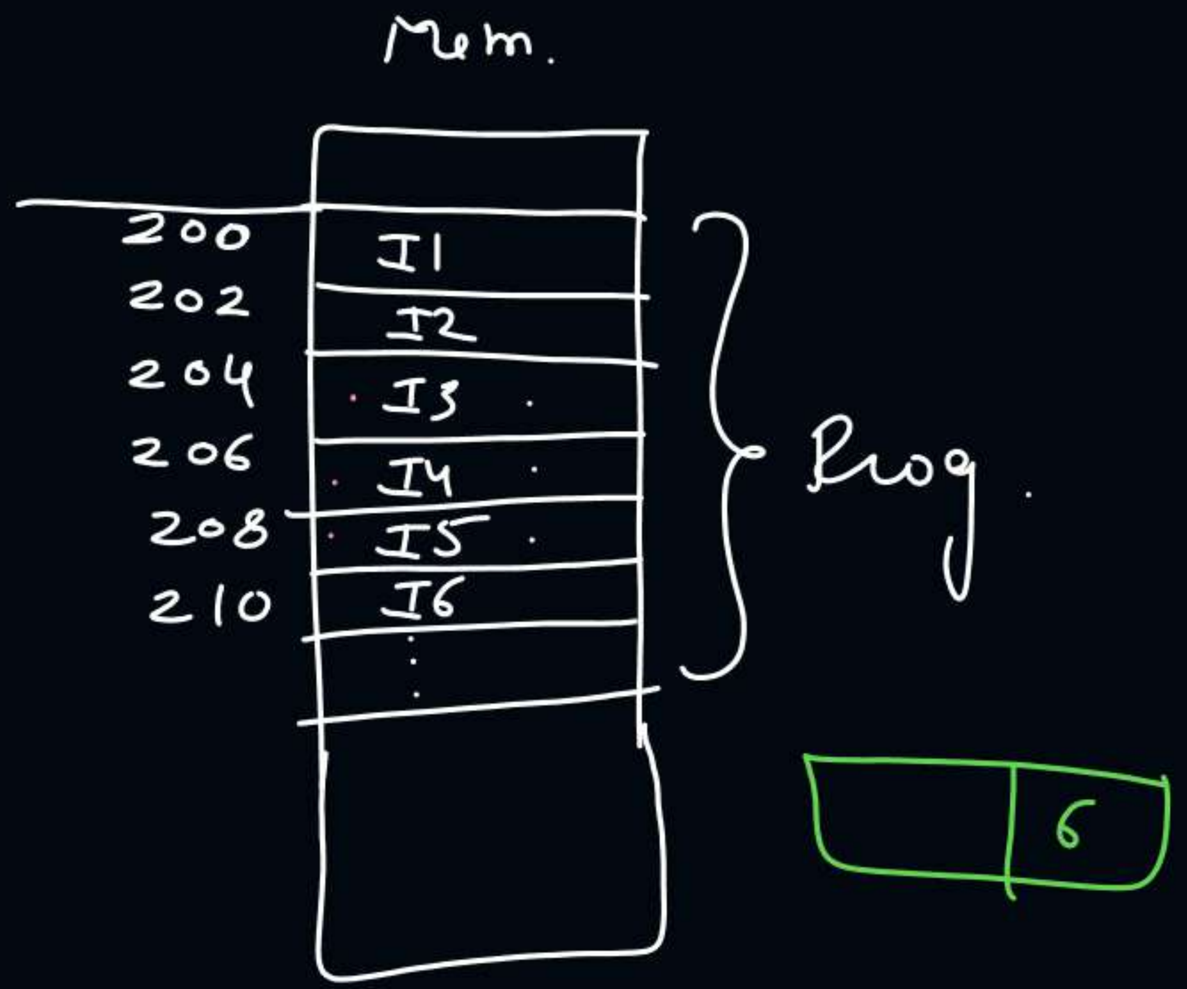
& assuming Target of I2 is I6 inst^n

⇓

3 inst^{ns} (I3 to I5) to be skipped,
which are stored on 6 addresses.

Target add. = 204 + 6

= 210

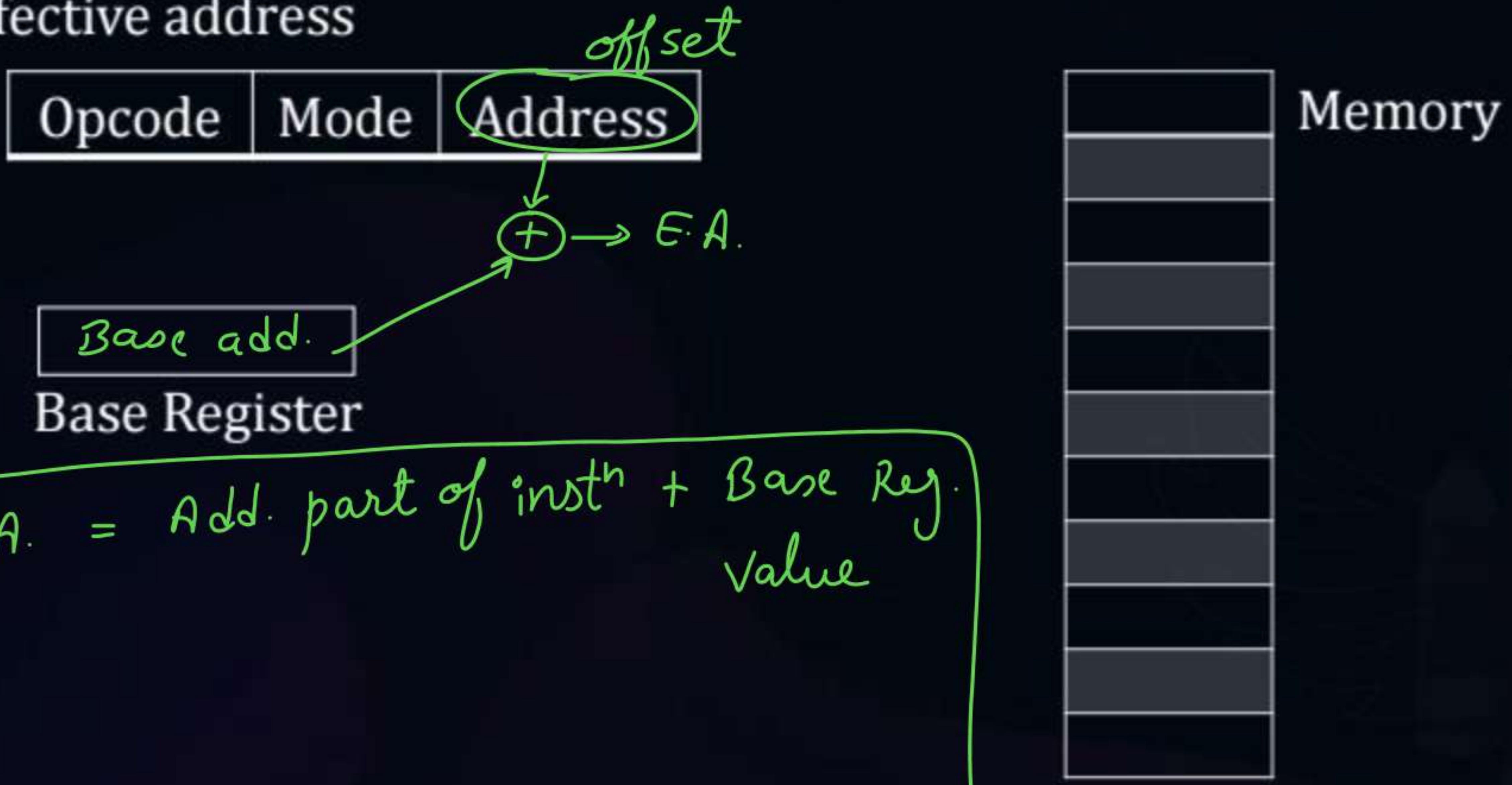


	Offset
for forw jump	+ve
for back jump	−ve

Topic : Base Register Mode

→ It is used inter-segment branching

Address part of instruction (offset) is added to Base register value to get the effective address



E.A. = Add. part of inst^n + Base Reg. Value

Topic : Example

	Memory	
200	*Opcode*	Mode
201	Address = 500	
202	Next Instruction	
399	450	
400	700	
500	800	
600	900	
702	-----------------	
800	300	

PC =200

R500 = 400

XR = 100

AC

Mode	Effective Address	Operan d
1. Immediate Mode		
2. Direct Mode		
3. Indirect Mode		
4. Register Mode		
5. Register Indirect Mode		
6. Autodecrement Mode		
7. Indexed Mode		
8. PC- Relative Mode		

[MCQ]

#Q. An instruction is stored at Location 300 with its address field at location 301. The address field has the value 400. A processor register contains the number 150. Evaluate the effective address, if addressing mode is:

A Direct

B Immediate

C Relative

D Register Indirect

[MCQ]

#Q. In case the code is position independent, the most suitable addressing mode is

A Direct mode

B Indirect mode

C Relative mode

D Indexed mode

[MCQ]

#Q. The addressing mode that permits relocation, without any change whatsoever in the code, is

A Indirect addressing

B Base register addressing

C Indexed addressing

D PC relative addressing

[MCQ]

#Q. A relative branch mode type instruction is stored in memory at address 300. The branch is made to an address 450.

1. What should be the value of relative address field of the instruction?
2. Determine the value of PC before instruction fetch, after the fetch and after execution phase?

[MCQ]

#Q. Consider a hypothetical processor with an instruction of type LW R1, 20(R2); which during execution reads a 32-bit word from memory and stores it in a 32-bit register R1. The effective address of the memory location is obtained by the addition of a constant 20 and the contents of register R2. Which of the following best reflects the addressing mode implemented by this instruction for operand in memory?

A Immediate Addressing

B Register Addressing

C Register indirect scaled addressing

D Base indexed addressing

[NAT]

#Q. Consider a three word machine instruction

ADD A[RØ], @B

The first operand (destination) “A[R0]” uses indexed addressing mode with R0 as the index register. The second operand (Source) “@B” uses indirect addressing mode. A and B are memory addresses residing at the second and the third words, respectively. The first word of the instruction specifies the opcode, the index register designation and the source and destination addressing modes. During execution of ADD instruction, the two operands are added and stored in the destination (first operand).

The number of memory cycles needed during the execution cycle of the instruction is_____.

2 mins Summary

Topic Addressing Modes ✓

Topic Types of Addressing Modes

Happy Learning

THANK - YOU