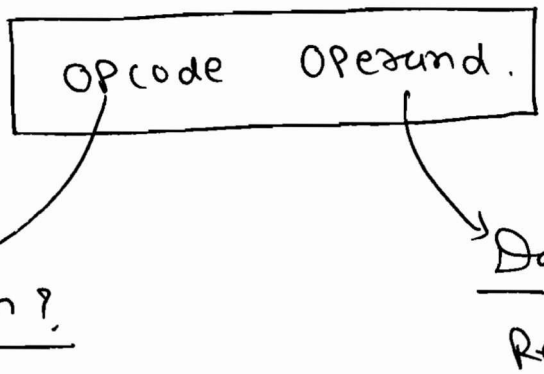
☆ Addressing Modes:

⇒ It is the method of specifying the operand in an instruction.

⇒ Instruction: 

Eg: MOV C, B
opcode operand

* 5 Types of addressing modes:

1. Register
2. Direct
3. Register Indirect
4. Immidiate
5. Impiicit / Impiied.

(*)

① Register Addressing Mode:

⇒ Regs / R_p

Eg. (i) MOV C, A

→ A → C

(ii) ADD R

→ A + R → R

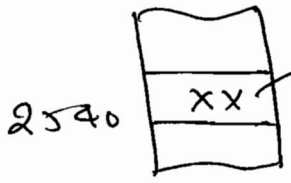
(iii) DAD R_p

→ HL + R_p → HL.

② Direct Addressing Mode:

→ It involes memory excess.

e.g: (i) `LDA 2540` ⇒



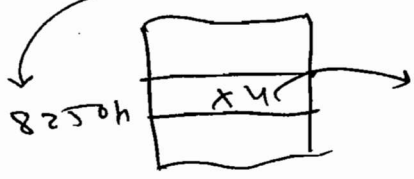
A ⇒ Direct.

memory
→ ↓ Address is given directly.

③ Register Indirect Addressing mode:

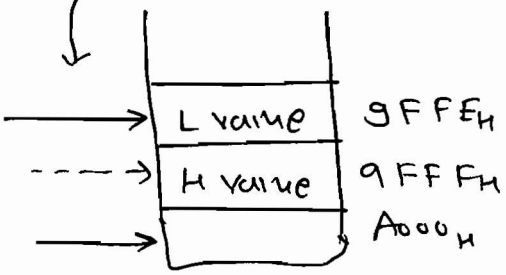
→ memory Address is specified by R_p & R.

e.g.: (i) `LDAX B` ⇒ BC = 8250H



So, Reg indirect.

(ii) Push H ⇒ SP = A000H. ⇒ Reg. Indirect



④ Immediate Addressing mode:

⇒ Source operand is alway 8 bit | 16 bit value.

E.g. : (i) MVI C, 32 ⇒ 32 → C

(ii) ADI 45 ⇒ A + 45 → A.

(iii) LXI B, 1020H ⇒ 1020H → BC

(iv) XRI 30H ⇒ A ⊕ 30 → A.

⑤ Implicit | Implied Addressing mode:

⇒ Source and destination are predefine.

E.g. RAL, RAR, CMA, CMC, STC, PCHL, XCHG, PCHL, XTHL.

Ex-1 Determine the addressing modes of the following instructions:

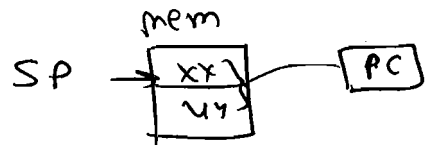
1) LHLD 9A00. → Direct

2) MOV C, M → Reg. Indirect

3) NOP/HLT → ~~Implicit & Implied~~ NO Addr. mode.

5) MVI M, 32. => Reg. Indirect and also Immediate.

6) RET. => Reg. Indirect.



=> Reg. Indirect.

7) CALL 3800 => Reg. Indirect / Immediate.

8) DAA => Implied.

P) Which of the following Instn is getting executed given that following:

Address : 4040_H

Data : 44_H

Memmapped IO ← IO/$\overline{M}$: 1

X $\overline{RD}$: 1

writing ← $\overline{WR}$: 0.

(a) IN 40_H

(b) IN 44_H.

(c) OUT 40_H

(d) OUT 44_H

P) Which of the following will store the value of accumulator into a output device, whose address is $F250_H$ → mem. mapped IO

i) LXI H, F250
mov A, M

(ii) LXI H, F250
OUT M

(iii) LXI H, F250
OUT F250

(iv) LXI H, F250.
mov M, A.
→ A ⟶ M

* Delay Program:

	T-States
1) MVI B, FF	7T
LOOP: DCR B	4T (False)
JNZ LOOP.	7T / 10T ← (True)
	≈10T

$\Rightarrow T_D = T_0 + T_L$;

T_0 = Delay outside Loop.

$T_0 = 7T = (7 \times 320ns)$

T_L = Delay within the loop.

$= (Count_{10} \times$ No. of T-states in loop $\times T)$

$T_L = (255_{10} \times 14 \times 320ns)$

$\therefore T_D = (7 \times 320ns) + (255 \times 14 \times 320ns) - (3 \times 320ns)$

↑ correction.

2) LXI B, FFFF$_H$	10T
LOOP: DCX B	6T
MOV A,C	4T
ORA B	4T
JNZ LOOP	10T / 7T

$\Rightarrow \quad T_0 = 10T = [10 \times 320ns]$

$T_L = [Count_{10} \times 24T]$

$= [65,535 \times 24 \times 320ns]$

$\therefore \quad T_D = T_0 + T_L$

$T_D = [10 \times 320ns] + [65,535 \times 24 \times 320ns].$

* Nested loop: (Loop within loop).

$\Rightarrow$

		T-states.
	MVI B, Count 2	7T
Loop 2:	MVI C, Count 1	7T
Loop 1:	DCR C	4T
	JNZ Loop 1	$\approx 10T$ (T_{L1})
	DCR B	4T
	JNZ Loop 2	$\approx 10T$

$\Downarrow$

$$T_D = [21T + T_{L_1}]\ Count2_{10} + [7 \times 320ns] - [Count2 \times 3T] - [3T].$$

Where $T_{L_1} = [Count1_{10} \times 14 \times 320ns]$

$$* \quad T_D = (21T + T_{L_1})\ Count2_{10} + (7 \times 320ns) - [Count2 + 1]\ 3T.$$

→ JNZ Loop 1 is false Count 2 times.

⇒ Subtract [count 2 × 3T].

JNZ loop 2 is false only one time.

⇒ Subtract 3T.

* General Points:

① ⓐ 1-Byte unConditional CALL Instruction = RSTn

ⓑ 1-Byte unconditional JMP Instruction = PCHL

② The DMA input HOLD is having the preference over non-maskable interrupt TRAP.

③ µp Checks the ready pin in T_2 state of each machine cycle. If it finds it as 'zero' it inserts wait T-states betn T_2 & T_3 until the ready becomes one.

④ How to shift 16 bit data towards left by 1 bit?

⇒ i) Store 16 bit no. into HL pair.

ii) Use DAD H instruction.

$$HL = 0000\ 0000\ 0000\ 0100_2 = 4_{10}$$

$$+HL = 0000\ 0000\ 0000\ 1000_2 = 8_{10} \quad (\times 2)$$

$$\overline{HL}$$

⑤ In I/O mapped I/O mode can we give

=> Yes, because they can be recognized based on their control signals. Hence, in this mode we can connect 256 input devices and 256 output devices in total 512.

Addr = F4H Addr = F4H

8085 µp

① Address => $A_7 - A_0 = F4_H$ | $A_{15} - A_8 = F4_H$

② Control signal => $IO/\overline{M} = 1$, $\overline{WR} = 0$ → output device.

③ Data