8085 Microprocessor →

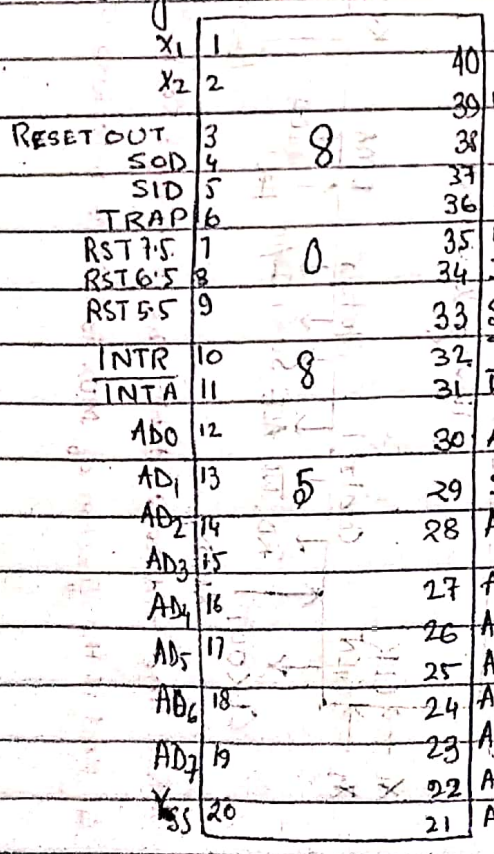
Intel 8085 is an 8 bit microprocessor introduced in 1976. Basic fabrication features →

(a) Max CPU Clock rate – 3, 5, 6 MHz
(b) Produced – 1976 to 2000.
(c) Min feature size – 3 µm.
(d) Transistor – 6500
(e) Data width – 8 bit
(f) Address width – 16 bit
(g) Pin No. – 40 DIP (Dual In Line Package).
(h) Design – Von Neumann architecture.
(i) Technology – 'N-MOS'
(j) Predecessor – Intel 8080 (k) Successor – Intel 8086

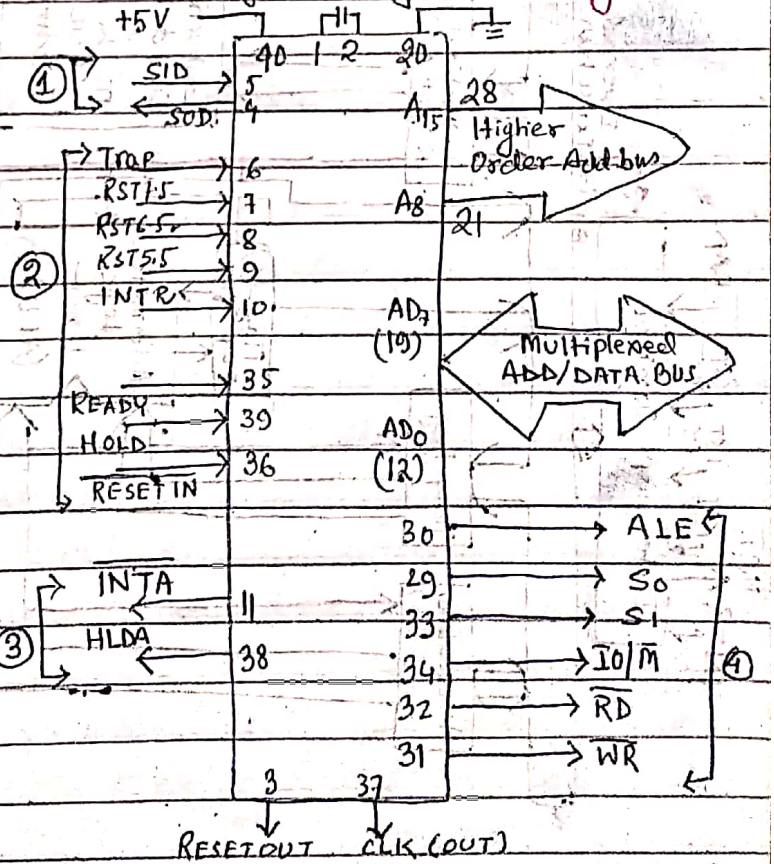
NOTE : ▪ Both 8080 & 8085 is 8 bit µp. 8085 supports the complete Ins. set of 8080 (except for the ANA/ANI operation which set 'AC' flag differently).

▪ Extra features of 8085 w.r.to 8080 is 8085 supports different interrupts and serial I/O functions.

Pin Diagram (Fig 3)



Signal Diagram. (Fig 4)

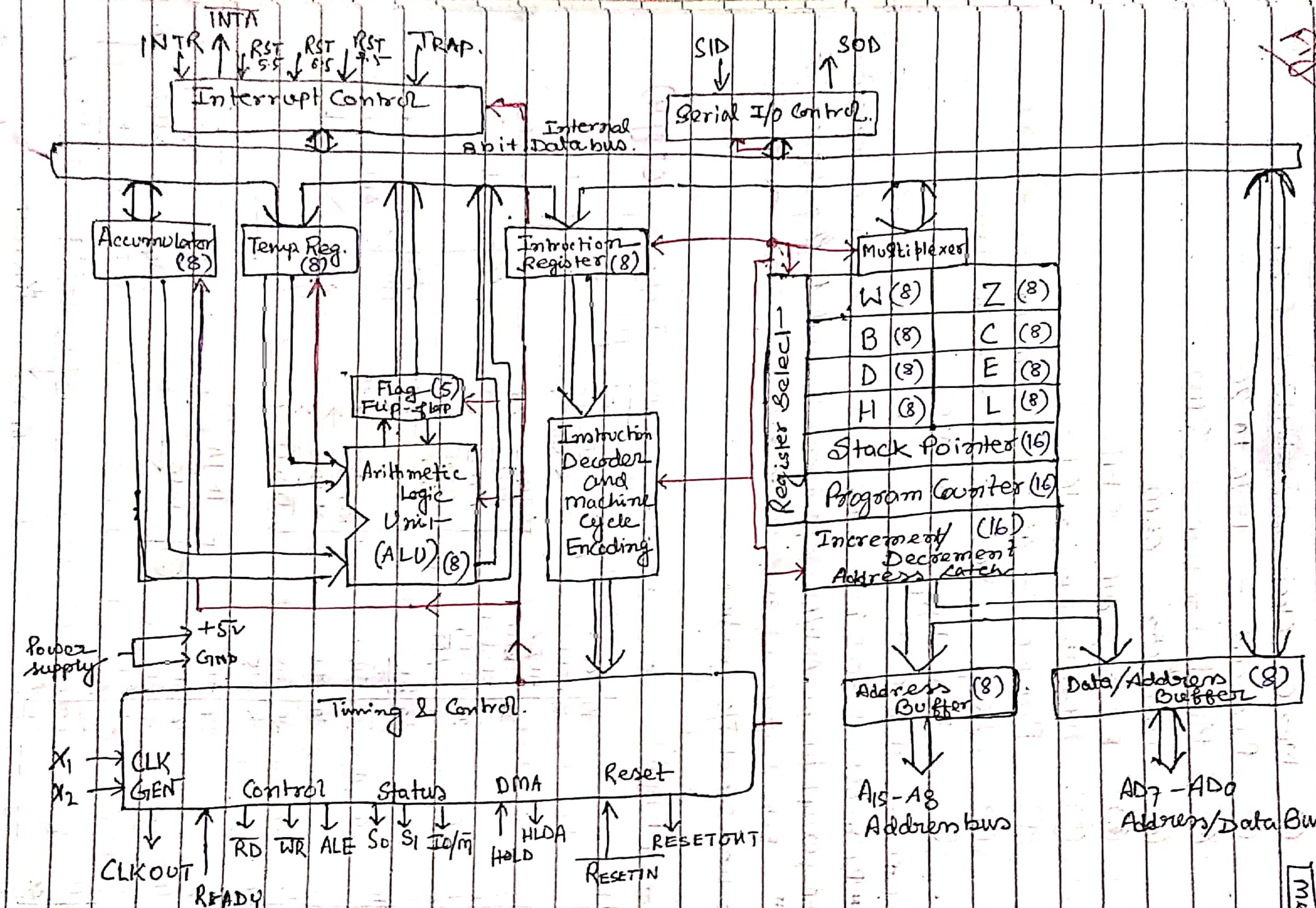


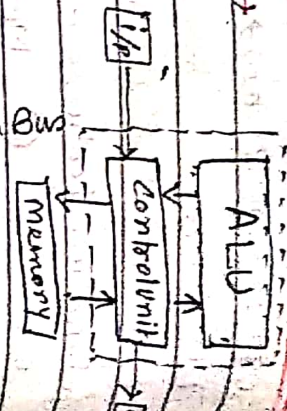
① Serial I/O Ports
② Externally Initiated Signals
③ External Signal Acknowledgement.
④ Control & Status Signals.

① PUSH PSW (AC+Flag) ↰ Only these two instructions use the combined 16 bit [Accumulator (8 bit) + Flag (8 bit)] for storing AC, Flag into Stack
② POP PSW (AC+Flag) ↲

8085 Microprocessor Functional Block Diagram →

Fig 5





μP ≈ CPU

* Red line indicates that the "Timing & Control" block controls the activity of all other blocks.

Q: What is the function of MUX in this architecture??

a) Interrupt Control → 8085 up can handle hardware and software interrupts. It has total four hardware interrupts - Trap, RST 7.5, RST 6.5, RST 5.5 & total eight software interrupts RST0 - RST1 - RST2 - RST3 - RST4 - RST5 - RST6 - RST7. Besides, 8085 up has one hardware INTR (Interrupt Request) Pin & one INTA (interrupt acknowledgment (Active Low)) Pin.

b) Serial I/O Control → 8085 up has the feature of serial data i/p & data o/p. It support only serial communication. It has specific instructions for handling serial communication.

c) Accumulator & Temporary Register → In most of the arithmetic & logical operations, the result is stored in accumulator. The temporary register is used to hold the data during arithmetic & logical operations. ~~It is used for temporary register purpose.~~ The temporary register is not accessible by the user/Programmer.

d) Flag (flip-flop) →

D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0
S	Z		AC		P		CY

(CY) - Carry flag. to show the result's condition whether carry is generated or not.
Set CY = 1 when carry generated
Reset CY = 0 when Carry not generated.

(P) - Parity flag. to show even no. of '1' in the result.
Set P = 1 - When even no. of '1' in the result
Reset P = 0 - When odd no. of '1' in the result.

→ Note : If the result is '00'H that is '0000 0000'$_2$ then Parity flag will be set (i.e. P = = 1).

(AC) - In arithmetic operation, when a carry is generated by digit D_3 and passed on to digit D_4, the AC flag is set (=1).
otherwise Reset (=0).

→ NOTE : The instruction set does not include any conditional jump based on 'AC' flag.

(Z) - Zero flag. To show the result is Zero or not Zero.
If result is Zero then Z=1, else Z=0.

(S) - If D_7 bit is 1 the Sign flag (S) - Set (=1)
else Sign flag (S) - Reset (=0)
} Applicable for Signed No. Calculations.

For unsigned no calculations 'S' will ignored.

Q: Suppose 'Ac' content is '00'H & register 'B' content is '00'H. Now using 'Add B' instruction these two contents are added. Then what will be the status of flag register & 'Ac' register?

Ans: AC ← 00H

B ← 00H

ADD B Means AC ⇐ AC + B

AC ⇐ 00H + 00H

AC ⇐ 00H.

Now after addition the flag registers' value will be changed according to the result. Now, final result is stored at 'Ac'. So, result is '00'H that means zero flag is set.

Parity flag is Set. & AC P CY

∴ Flag register content →

S	Z		AC		P		CY
0	1	0	0	0	1	0	0

⇒ 44H.

∴ Ans : (i) 'Ac' = 00H

(ii) 'Flag' = 44H.

e) Arithmetic & Logic Unit → All arithmetic & logical operations are carried out here. Two i/p are coming from "Accumulator" & "temporary register" and one o/p is directly going to the 8 bit data bus. A Block c-d are connected with it.

f) Instruction Register & Instruction Decoder - Machine cycle Encoding → This block is part of the ALU. When an instruction is fetched from memory, it is loaded in Instruction register. The decoder decodes the instruction & establishes the sequence of further operations/flow. The instruction register is not programmable/accessable by the user.

g) Timing and control unit → This unit does two important tasks.

(i) It synchronises all the microprocessor operations with the internal clock.

(11) Generates the control signals necessary for communication between the microprocessor and peripherals, memory.

Control signals for µP ↔ Peripheral Devices == $\overline{IOR}$, $\overline{IOW}$

$\overline{IOR}$ ($IO/\overline{M}=1$, $\overline{RD}=0$, $\overline{WR}=1$) $\overline{IOW}$ ($IO/\overline{M}=1$, $\overline{WR}=0$, $\overline{RD}=1$)

µP ↔ Memory == $\overline{MEMR}$, $\overline{MEMW}$ ($\overline{MEMR}$: $IO/\overline{M}=0$, $\overline{RD}=0$, $\overline{WR}=1$) ($\overline{MEMW}$: $IO/\overline{M}=0$, $\overline{WR}=0$, $\overline{RD}=1$)

8085 Machine Cycle Status & Control Signals →

Sl. No.	Machine cycle	Status $IO/\overline{M}$	S_1	S_0	Control Signals
1.	Opcode Fetch	0	1	1	$\overline{RD}=0$
2.	Memory Read	0	1	0	$\overline{RD}=0$
3.	Memory Write	0	0	1	$\overline{WR}=0$
4.	I/O Read	1	1	0	$\overline{RD}=0$
5.	I/O Write	1	0	1	$\overline{WR}=0$
6.	Interrupt Acknowledge	1	1	1	$\overline{INTA}=0$
7.	HALT	Z	0	0	$\overline{RD}$, $\overline{WR}$ = Z and $\overline{INTA}$=1
8.	HOLD	Z	X	X	
9.	RESET	Z	X	X	

Z = Tri State (High Impedence)

X = Unspecified.

h) Register Array → It contains programmable and non programmable registers. The programable registers are B-C-D-E-H-L-SP-PC & non-programmable registers are W, Z. The W, Z registers are used to hold 8 bit data during execution of some instructions.** However, because they are used internally, they are not available to the programmer. Increment/decrement add. latch is also part of reg. array which is used to increase/decrease address.

i) Address buffer & Data/Address Buffer → These two buffer circuit used to generate higher order address bus (A_{15}-A_8) & lower order multiplexed address data bus (AD_7-AD_0).

** Some instructions → CALL

Demultiplexing the bus AD7 - AD0

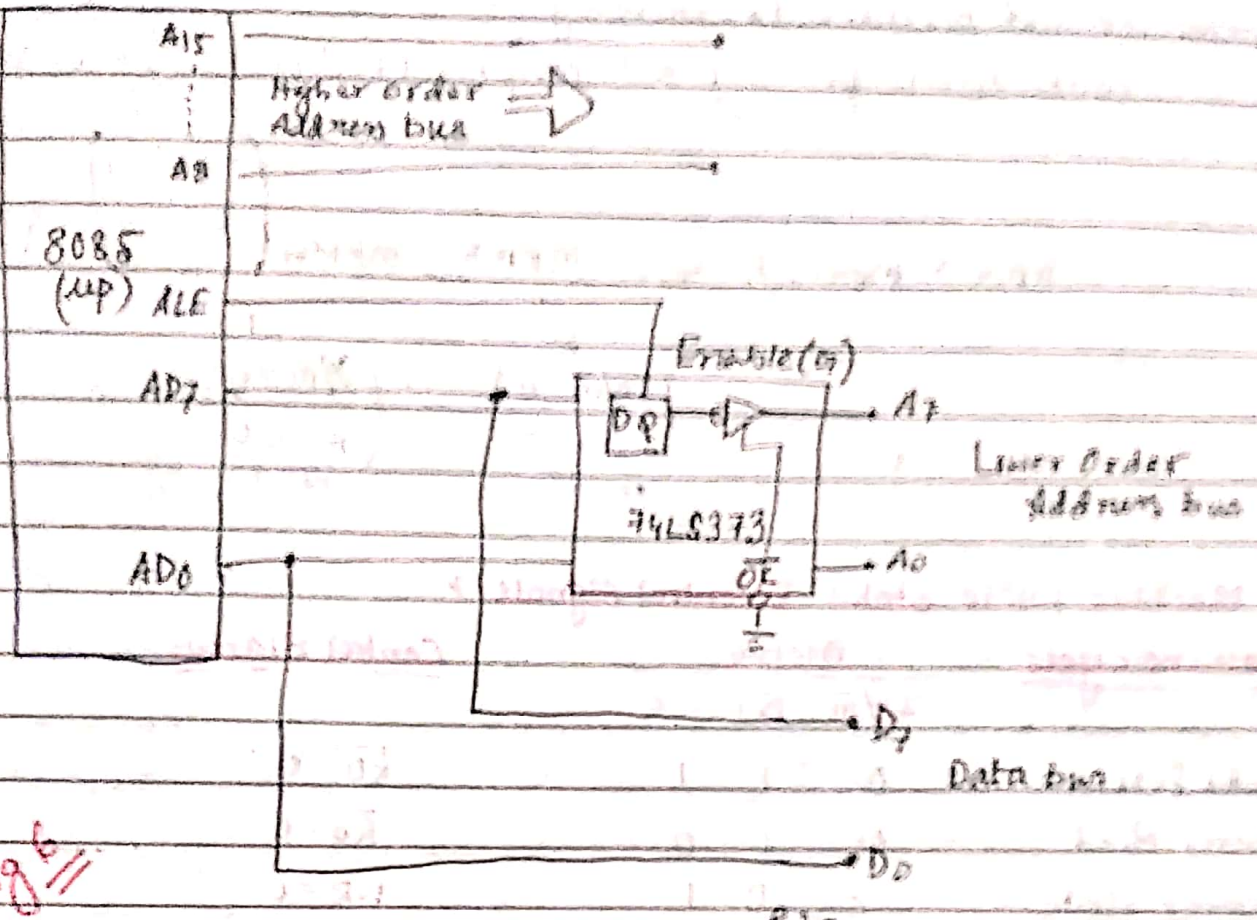


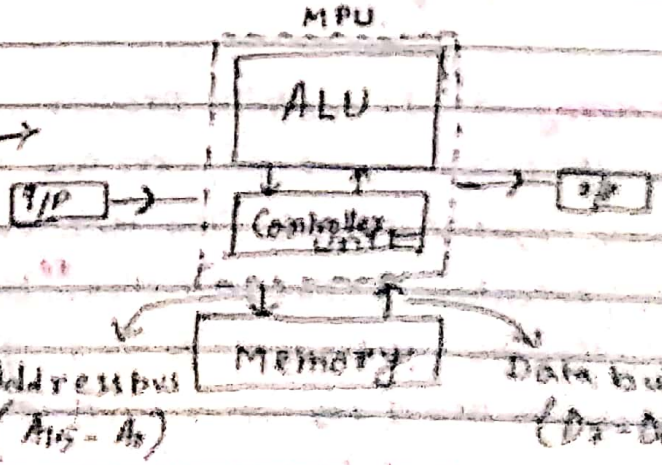
fig 6 //

fig

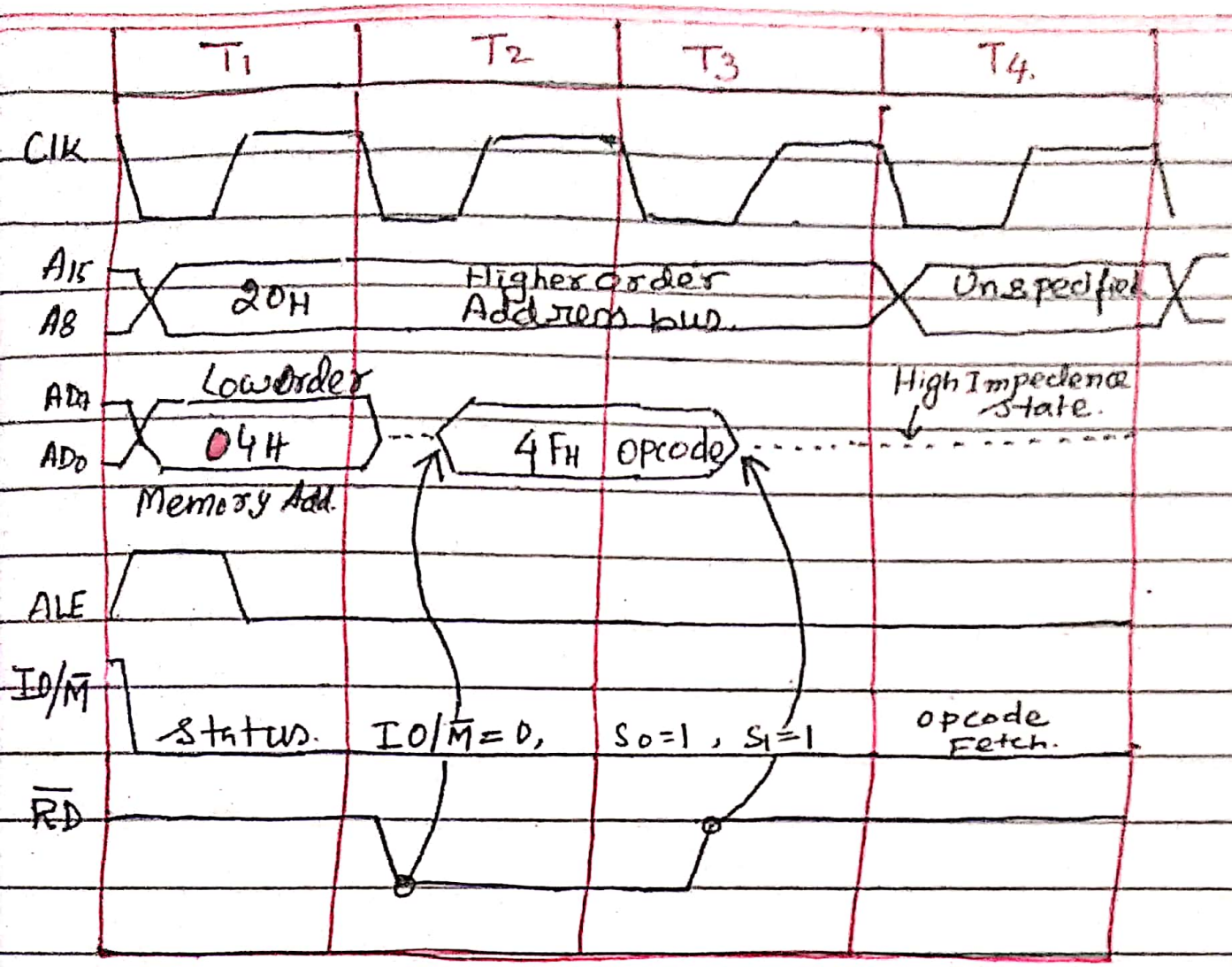
The above figure shows that the ALE & latch circuit (74LS373) are used to demultiplexed AD7 - AD0 bus. Address Latch Enable (ALE) is connected to the enable pin (G) of the latch, and the output Control ($\overline{OC}$) signal pin is grounded. When ALE = High (Logic 1), then Enable pin of D F/F is active and data AD7 - AD0 latched to A7 - A0

When, ALE goes low (Logic 0) the previous data is latched until the next high ALE signal.

** Block diagram of MPU →



Now 8. from previous figure ⑥, we can understand how 8085 μP communicate with Memory block which is externally connected with 8085 μP



Timing Diagram : Transfer of Byte from Memory to MPU

or

Timing Diagram of MOV C, A Instruction

* MVI B, 06H. - Instruction.