UNIT-5

PAGE NO. // //
DATE // //

Evalution of Microprocessor

The first microprocessor Introduced In 1971 was a 4-bit microprocessor with 4m5KB memory and had a set of 45 Instructions. In the past 5 decades microprocessor speed has doubled every two years, as predicted by Gordon Moore, Intel co-founder. Current microprocessors can access 64GB memory. Depending on width of data microprocessors can process, they are of these categories-

- 8-bit
- 16-bit
- 32-bit
- 64-bit

Size of Instructions set is another Important consideration while categorizing microprocessor. Initially, microprocessors had very small Instructions sets because complex hardware was expensive as well as difficult to build.

As technology developed to overcome these issues, more and more complex instruction were added to increase functionality of the microprocessor. However, soon it was realized that having large instructions sets was counterproductive as many instructions that were rarely used & sat idle on ~~pro~~ precious memory space. So the old school of thought that supported smaller instruction sets gained popularity.

Name	Year	Transistors	Data width	Clock speed.
8080	1974	6,000	8 bits	2MHz
8085	1976	6,500	8 bits	5MHz
8086	1978	24,000	16 bits	5MHz
8088	1979	29,000	8 bits	5MHz
80286	1982	134,000	16 bits	6MHz
80386	1985	275000	32 bits	16MHz
80486	1989	1,200,000	32 bits	25MHz
PENTIUM	1993	3,100,000	32/64 bits	60MHz
PENTIUM II	1997	7,500,000	64 bits	233MHz
PENTIUM III	1999	9,500,000	64 bits	450MHz
PENTIUM IV	2000	42,000,000	64 bits.	1.5GHz

Microprocessor- 8085 :-

8085 is pronounced as "eighty-eighty-five" microprocessor. It is an 8-bit microprocessor ~~by~~ designed by Intel in 1977 using NMOS technology.

- 8-bit data Bus
- 16-bit address bus, which can address upto 64Kb.
- A 16-bit program counter
- A 16-bit Stack pointer
- Six 8-bit registers arranged in pairs:
- BC, DE, HL.
- Requires +5V Supply to operate at 3.2MHz Single phase clock. It is used in washing machines, microwaves ovens, mobile phone, etc.

8085 Microprocessor-Functional Units :-

1- Accumulator :- It is an 8-bit register used to perform arithmetic, logical, I/O and Load / STORE operations. It is connected to internal data bus and ALU.

2- Arithmetic and Logic unit :-
As the name suggests, it performs arithmetic and logical operations like Addition, Subtraction, AND, OR, etc. on 8-bit data.

3- General purpose register :- There are 6 general purpose registers in 8085 processor i.e. B, C, D, E, H and L. Each register can hold 8-bit data. These registers can work in pair to hold 16-bit data and their pairing combination is like B-C, D-E and H-L.

4- Program Counter :- It is a 16-bit register used to store the memory address location of the next instruction to be executed. Microprocessor increments the program whenever an instruction is being executed, so that the program counter points to the memory address of the next instruction that is going to be executed.

5- Stack pointer :- It is also a 16-bit register works like stack, which is always incremented / decremented by 2 during push and pop operations.

6- Temporary register :- It is 8-bit register which hold the temporary data of arithmetic and logical operations.

7- Flag register - It is an 8-bit register having five 1-bit flip-flop which hold either 0 or 1 depending upon the result stored in the accumulator.
i) Sign (S).
ii) Zero (Z).
iii) Auxiliary carry (AC)
iv) Parity (P)
v) Carry (C).
Its bit position is shown in the following table-

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

8- Instruction register and decoder -
It is provides timing and control signal to the microprocessor to perform

PAGE NO. // //
DATE // //

operations. Following are the timing and control signals, which control external and internal circuits - control signals - READY, RD', WR', ALE
Status signals - S0, S1, IO/M'
DMA signals - HOLD, HLDA
RESET signals - RESET IN, RESET OUT.

10- Interrupt control - As the name suggests it controls the interrupts during a process. When a microprocessor is executing a main program an interrupt occurs, the microprocessor shifts the control from the main program to process the incoming requests. After the request is completed, the control goes back to the main program.
There are 5 interrupt signals in 8085 microprocessor - INTR, RST 7.5, RST 6.5, RST 5.5, TRAP.

11- Serial Input/Output control - It controls the serial data communication by using these two instructions -

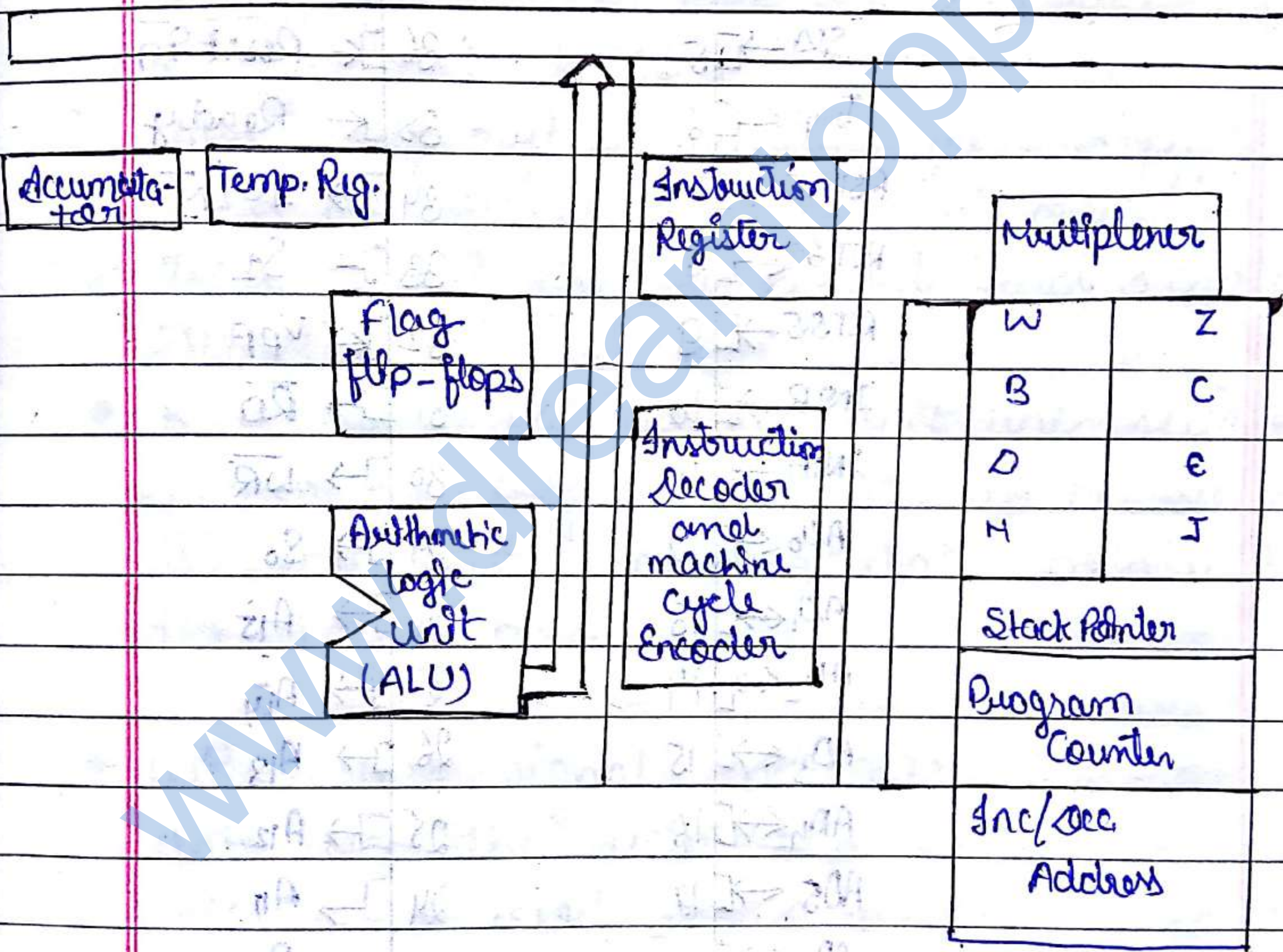
PAGE NO. // //
DATE // //

(i) SID (Serial input data)
(ii) SOD (Serial output data).

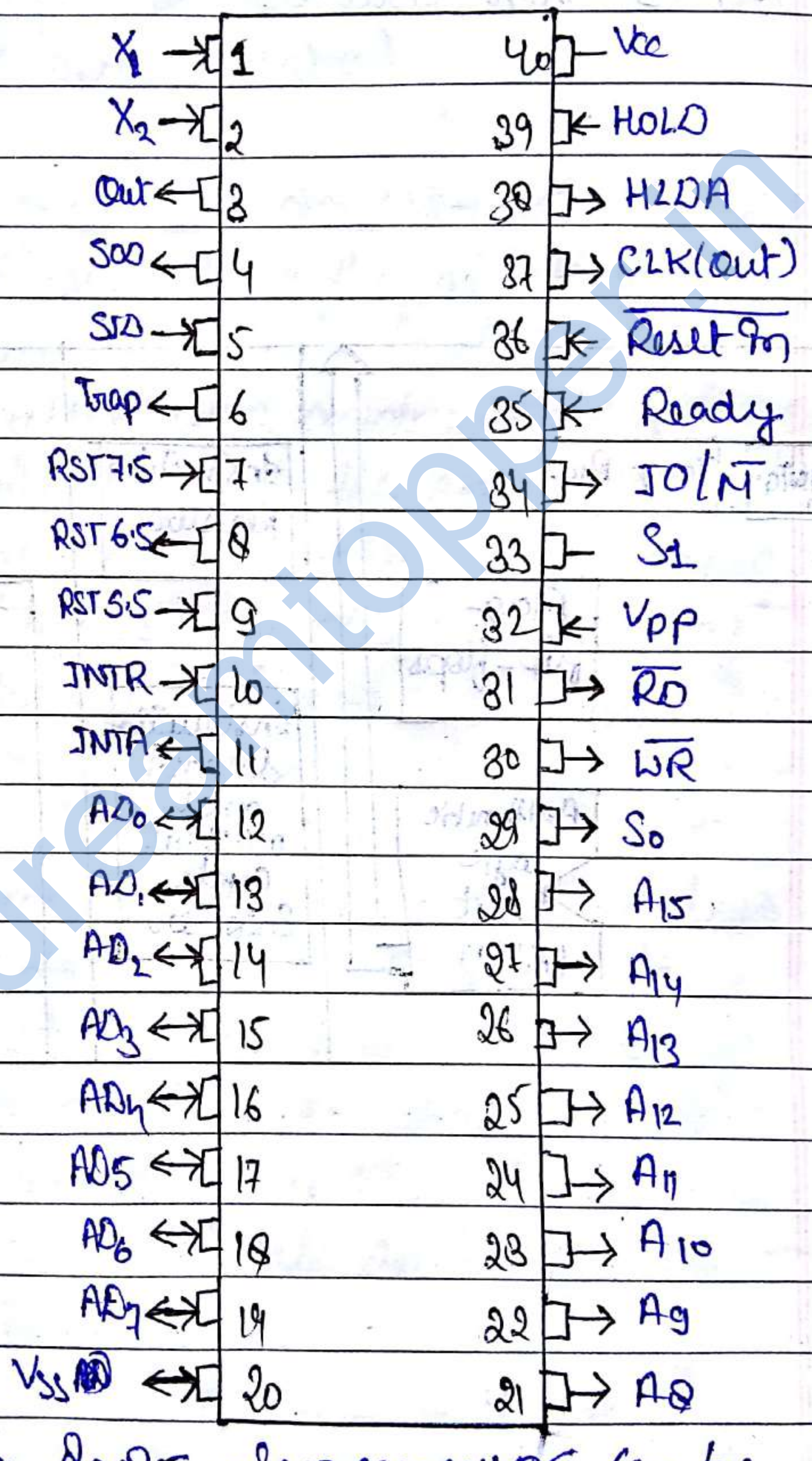
12- Address buffer and address-data buffer - The content stored in the stack pointer and program counter is loaded into the address buffer and address-data buffer to communicate with the connected CPU. The memory and I/O chips are connected to these buses; the CPU can exchange the desired data with the memory and I/O chips.

13- Address bus and data bus - Data bus carries the data to be stored. It is bidirectional, whereas address bus carries the location to where it should be stored and it is unidirectional. It is used to transfer the data and address I/O devices.

★ 8085 Architecture–



The following figure depicts the pin diagram of 8085 microprocessor –



The pin of a 8085 microprocessor can be classified into seven groups –

1- Address bus- $A_{15} - A_8$, It carries the most significant 8-bits of memory/IO address.

2- Data bus- ~~AD~~ AD7 - AD0, It carries the least significant 8-bit address and data bus.

3- Control and status signals- These signals are used to identify the nature of operation. There are 3 control signal and 3 status signals.

- RD- This signal indicates that the selected IO or memory device is to be read and is ready for accepting data available on the data bus.

- WR- This signal indicates that the data on the data bus is to be written into a selected memory or IO location.

- ALE - It is a positive going pulse generated when a new operation is starting in the microprocessor. When the pulse goes high, it indicates address. When the pulse goes down it indicates data.

PAGE NO. // //
DATE // //

Three status signals are IO/M, S0 & S1.

1- IO/M – This signal is used to differentiate between IO and memory operation, i.e. when it is high it indicates IO operation and when it is low then it indicates memory operation.

2- S1 & S0 – These signals are used to identify the type of current operation.

3- Power Supply – There are two power supply signals – VCC & VSS indicates +5V power supply and it indicates ground signal.

Clock signals – There are 3 clock signals – i.e. X_1, X_2, CLK OUT.

- X_1, X_2 – A crystal (RC, LC N/W) is connected at these two pins and is used to set frequency of the internal clock generator. This frequency is internally divided by 2.

PAGE NO. // //
DATE // //

- CLK Out :- This signal is used as the system clock for devices connected with the microprocessor.

★ Interrupts & externally Initiated signals :- Interrupt are the signals generated by external devices to request the microprocessor to perform a task. There are 5 Interrupt signals, i.e TRAP, RST 7.5, RST 6.5, RST 5.5 and INTR.

- INTA - It is an interrupt acknowledgement signal.
- RESET IN - This signal is used to reset the microprocessor by setting the program counter to zero.
- RESET OUT - This signal is used to reset all the connected devices when the microprocessor ~~is~~ is reset.
- READY - This signal indicates that the device is ready to send or receive data. If READY is low, then the CPU has to wait for READY to go high.
- HOLD - This signal indicates that another master is requesting the use of address and data buses.

- HLDA (Hold Acknowledgement) - It indicates that the CPU has received the HOLD request and it will relinquish the bus in the next clock cycle. HLDA is set to low after the HOLD signal is removed.

7. Serial I/O Signals :- There are 2 serial signals, i.e. SID and SOD and these signals are used for serial communication.

- SOD (serial output data line) :- The output SOD is set / reset as specified by the SIM instruction.

- SID (Serial Input data line) ;- The data on this line is loaded into accumulator whenever a RIM instruction is executed.

★. Addressing Modes in 8085 :-
These are the instructions used to transfer the data from one register to another register, from the memory to the register, and from the register to

PAGE NO. // //
DATE // //

the memory without any alteration in the content. Addressing modes in 8085 is classified into 5 groups –

1- Immediate addressing mode – In this mode, the 8/16 bit data is specified in the instruction itself as one of its operand. For example:- MVI K, 20F: means 20F is copied into register K.

2- Register addressing mode – In this mode, the data is copied from one register to another. For ex:- MOV K, B: means data in register B is copied to register K.

3- Direct addressing mode – In this mode, the data is directly copied from the given address to the register.
For ex – LDB 5000K : means the data at address 5000K is copied to register B.

4- Indirect addressing mode – In this mode, the data is transferred from one register to another by using the address pointed

by the register.
For ex - MOV K, B: means data is transfered from the memory address pointed by the register to the register K.

5- Implied addressing mode - This mode does not ~~req~~ require any operand, the data is specified by the opcode itself.
For ex - CMP.

★. Interrupts in 8085:-
Interrupts are the signals generated by the external devices to request the microprocessor to perform a task. There are 5 interrupt signals, i.e. TRAP, RST 6.5, RST 7.5, RST5.5 and INTR.
Interrupt are classified into following groups based on their parameter -

1- Vector interrupt - In this type of interrupt the interrupt address is known to the processor.
For ex:- RST 5.5, RST 6.5, RST 7.5, TRAP.

PAGE NO. // //
DATE // //

2- Non-Vector Interrupt - In this type of Interrupt, the interrupt is not known to the processor so, the interrupt address needs to be sent externally by the device to perform interrupts. For ex - INTR.

3- Maskable Interrupt :- In this type of Interrupt, we can disable the interrupt by writing some instructions into the program. For example: RST7.5, RST6.5, RST 5.5.

4- Non-Maskable Interrupt :- In this type of Interrupt we cannot disable the interrupt by writing some instructions into the program. For example: TRAP.

5- Software Interrupt :- In this type of Interrupt, the programmer has to add the instructions into the program to execute the interrupt. There are 8 software interru in 8085, i.e. RST0, RST1, RST2, RST3, RST4, RST5, RST6, and RST7.

6- • Hardware Interrupt :- There are 5 interrupts pins in 8085 used as hardware interrupt, i.e., TRAP, RST 7.5, RST 6.5, RST 5.5, INTA.

NOTE :- INTA is not an interrupt, it is used by the microprocessor for sending acknowledgement. TRAP has the highest priority, then RST 7.5 and so on.

• Interrupt Service Routine (ISR) :-
A small program or a routine that when executed, services the corresponding interrupting source is called an ISR.

★. TRAP :-
It is a non-maskable interrupt, having the highest priority among all interrupts. By default, it is enabled until it gets acknowledged. In case of failure, it executes as ISR and sends the data to backup memory. This interrupt transfer the control to the location 0024H.

RST 7.5:- It is a Maskable Interrupt, having the second highest priority among all interrupts. When this interrupt is executed, the processor saves the content of the PC register into the stack and branches to 003CH address.

RST 6.5:- It is a maskable interrupt, having the third highest priority among all interrupts. When this interrupt is executed the processor saves the content of the PC register into the stack branches to 0034H address.

RST 5.5:- It is a maskable interrupt. When this interrupt is executed, the processor saves the content of the PC register into the stack and branches to 002CH address.

INTR:- It is a maskable interrupt, having the lowest priority among all interrupts. It can be disabled by resetting the microprocessor.

- When INTR signal goes high, the following event can occur-

PAGE NO. // //

DATE // //

- The microprocessor checks the status of INTR signal during the execution of each instruction
- ~~When~~ When the INTR signal is high, then the microprocessor ~~saves the address of the next~~ completes its current instructions and sends active low interrupt acknowledge signal.
- When instructions are received, then the microprocessor saves the address of the next instruction on stack and executes the received instruction.