Lab 1

Name: Yash K. Patel

Roll-No: CE107

Subject: Computer System Architecture

Q-1) Study of the 8085 Architecture

Ans:

Overview:

- The architecture of the 8085 microprocessors mainly includes the timing
  & control unit, Arithmetic and logic unit, decoder, instruction register, interrupt control, a register array, serial input/output control.
- The 8085 microprocessor is capable of doing basic math operation like left shift, right shift, addition, multiplication.

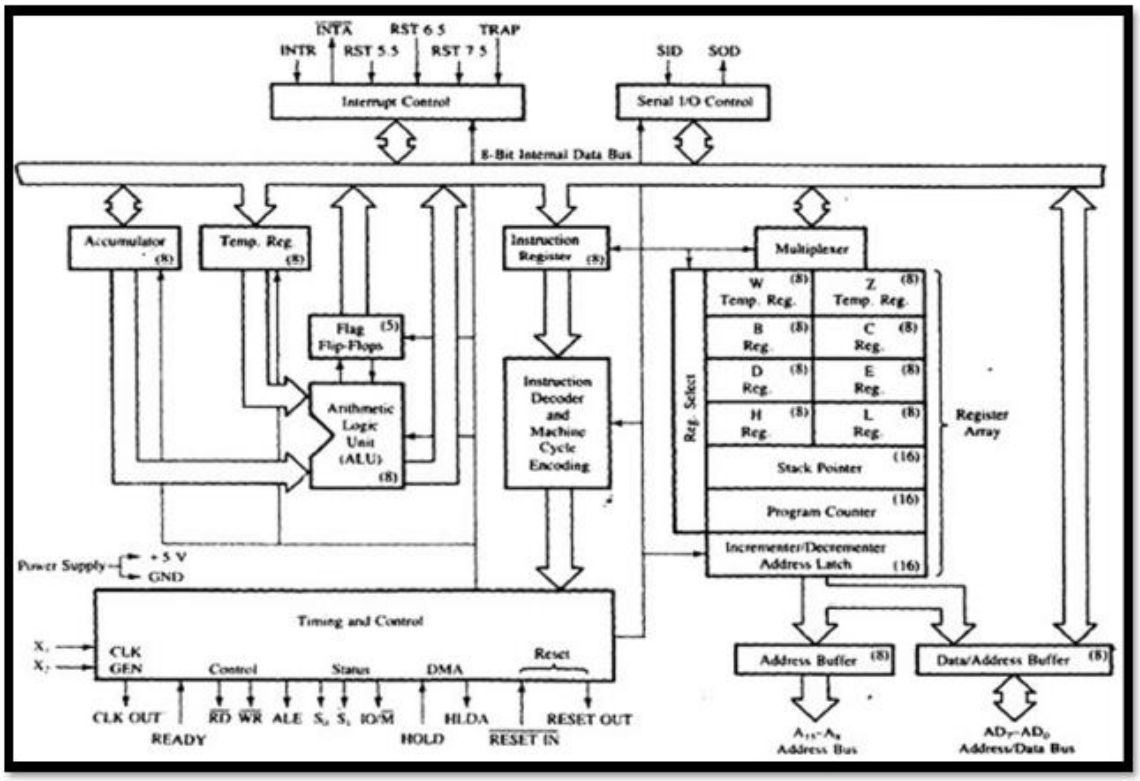
Components in 8085 microprocessors:

- Flag Registers: These registers are used to maintain flag for different kind of variables such as parity flag, Auxiliary carry flag, Zero flag, Sign flag, and rest are don't cares.

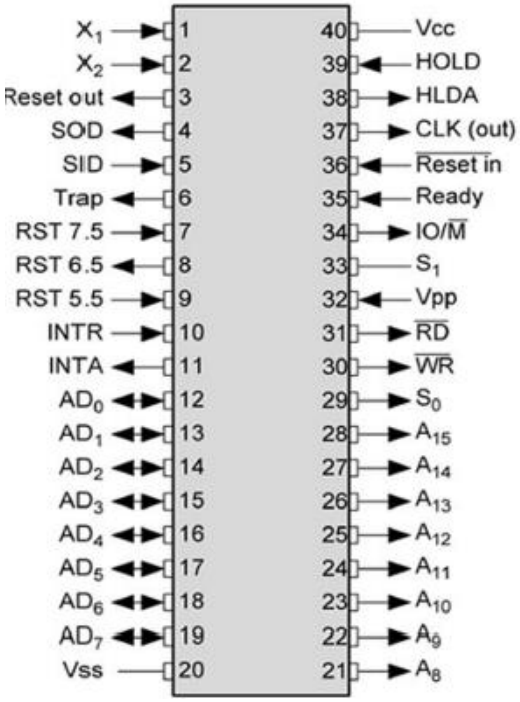
- Register Array:
  - Accumulator: An accumulator is a register for short-term, intermediate storage of arithmetic and logic data in a computer's CPU. Ex: 2+3+4 then 2+3=5 will be stored in accumulator and then will be added with 4.
  - Registers: Other registers are named from B to L these are used to store data and A is called accumulator.

- Program Counter: The program counter holds the memory address of the next instruction to be performed. This is used to conserve time.
- Stack Pointer: The SP or stack pointer is a 16-bit register and functions similar to a stack, which is constantly increased or decreased with two throughout the push and pop processes.
- Increment or Decrement Register: The 8-bit register contents or else a memory position can be increased or decreased with one. The 16-bit register is useful for incrementing or decrementing program counters as well as stack pointer register content with one. This operation can be performed on any memory position or any kind of register.
- Address bus and data bus: The data bus carry's information that is to be stocked up. Data bus is bidirectional. However, address bus is unidirectional as only stores the data at the given address.
- Timing and control unit: The timing & control unit can be used to supply the signal to the 8085 microprocessor architecture for achieving the particular processes. The timing and control units are used to control the internal as well as external circuits. These are classified into four types namely control units like RD' ALE, READY, WR', status units like S0, S1, and IO/M', DM like HLDA, and HOLD unit, RESET units like RST-IN and RST-OUT.

Architecture of 8085 Microprocessor:



Pin Diagram:



Q-2) To study the basic data transfer instructions, the opcodes and operands.

Ans:

Opcode	Operand	Meaning	Explanation
MOV	Rd, Sc M, Sc Dt, M	Copy from the source (Sc) to the destination(Dt)	This instruction copies the contents of the source register into the destination register without any alteration. **Example** − MOV K, L
MVI	Rd, data M, data	Move immediate 8-bit	The 8-bit data is stored in the destination register or memory. **Example** − MVI K, 55L
LDA	16-bit address	Load the accumulator	The contents of a memory location, specified by a 16-bit address in the operand, are copied to the accumulator. **Example** − LDA 2034K
LDAX	B/D Reg. pair	Load the accumulator indirect	The contents of the designated register pair point to a memory location. This instruction copies the contents of that memory location into the accumulator. **Example** − LDAX K
LXI	Reg. pair, 16-bit data	Load the register pair immediate	The instruction loads 16-bit data in the register pair designated in the register or the memory. **Example** − LXI K, 3225L
LHLD	16-bit address	Load H and L registers direct	The instruction copies the contents of the memory location pointed out by the address into register L and copies the

			contents of the next memory location into register H. **Example** − LHLD 3225K
STA	16-bit address	16-bit address	The contents of the accumulator are copied into the memory location specified by the operand. This is a 3-byte instruction, the second byte specifies the low-order address and the third byte specifies the high-order address. **Example** − STA 325K
STAX	16-bit address	Store the accumulator indirect	The contents of the accumulator are copied into the memory location specified by the contents of the operand. **Example** − STAX K
SHLD	16-bit address	Store H and L registers direct	The contents of register L are stored in the memory location specified by the 16-bit address in the operand and the contents of H register are stored into the next memory location by incrementing the operand. This is a 3-byte instruction, the second byte specifies the low-order address and the third byte specifies the high-order address. **Example** − SHLD 3225K
XCHG	None	Exchange H and L with D and E	The contents of register H are exchanged with the contents of register D, and the contents of register L are exchanged with the contents of register E. **Example** − XCHG

SPHL	None	Copy H and L registers to the stack pointer	The instruction loads the contents of the H and L registers into the stack pointer register. The contents of the H register provide the high-order address and the contents of the L register provide the low-order address. **Example** − SPHL
XTHL	None	Exchange H and L with top of stack	The contents of the L register are exchanged with the stack location pointed out by the contents of the stack pointer register. The contents of the H register are exchanged with the next stack location (SP+1). **Example** − XTHL
PUSH	Reg. pair	Push the register pair onto the stack	The contents of the register pair designated in the operand are copied onto the stack in the following sequence. The stack pointer register is decremented and the contents of the high order register (B, D, H, A) are copied into that location. The stack pointer register is decremented again and the contents of the low-order register (C, E, L, flags) are copied to that location. **Example** − PUSH K
POP	Reg. pair	Pop off stack to the register pair	The contents of the memory location pointed out by the stack pointer register are copied to the low-order register (C, E, L, status flags) of the operand. The stack pointer is incremented by 1 and the contents of that memory

			location are copied to the high-order register (B, D, H, A) of the operand. The stack pointer register is again incremented by 1. **Example** − POPK
OUT	8-bit port address	Output the data from the accumulator to a port with 8bit address	The contents of the accumulator are copied into the I/O port specified by the operand. **Example** − OUT K9L
IN	8-bit port address	Input data to accumulator from a port with 8-bit address	The contents of the input port designated in the operand are read and loaded into the accumulator. **Example** − IN5KL

Q-3) To write a simple program of addition of two numbers and to check the status of each of the flag.

Ans:

main.asm

```
1 MVI B,4H
2 MVI A,10H
3 ADD A
4 HLT
```

Registers

A/PSW	0x 00 02
BC	0x 00 00
DE	0x 00 00
HL	0x 00 00
SP	0x 00 00
PC	0x 08 00

Flags

Z	☐
S	☐
P	☐
C	☐
AC	☐

4

Assembler Output

1	6 64	MVI B,40H
2	3E 128	MVI A,80H
3	87	ADD A
4	76	HLT