**8085 Microprocessor :**

It is a programmable electronics chip ( Integrated Circuit (IC) ).  A single IC has computing and decision making capabilities similar to central processing unit of a computer. It is used in almost all types of electronics devices like mobile phones, printers, washing machines, etc. and also used in advanced applications like radars, satellites and flights.

* It has 8 bit data bus and 16 bit address bus, thus it is capable of addressing 64 KB of memory.
* It has 8 bit ALU 8 bit ALU that can perform 8 bit operations.
* Lower order address bus is multiplexed with data bus to minimize the chip size.

 **Bit**: A bit is a single binary digit.

 **Word:** A word refers to the basic data size or bit size that can be processed by the arithmetic and logic unit of the processor. A 16-bit binary number is called a word in a 16-bit processor.

 **Memory Word**: The number of bits that can be stored in a register or memory element is called a memory word.

**Bus:** A bus is a group of wires (lines) that carry similar information. **System Bus**: The system bus is a group of wires used for communication between the microprocessor and peripherals.

* **Address Bus**: It carries the address, which is a unique binary pattern used to identify a memory location or an I/O port.
* **Data Bus**: The data bus is used to transfer data between memory and processor or between I/O device and processor.
* **Control Bus**: The control bus carry control signals, which consists of signals for selection of memory or I/O device from the given address, direction of data transfer and synchronization of data transfer in case of slow devices.
* **The 8085 microprocessor** is an 8-bit processor available as a 40-pin IC package (shown the figure below) and uses +5 V for power. It can run at a maximum frequency of 3 MHz.



* The 8085 has extensions to support new interrupts, with three maskable interrupts (RST 7.5, RST 6.5 and RST 5.5), one non-maskable interrupt (TRAP), and one externally serviced interrupt (INTR).
* Three control signals are available on chip: (i) RD : it is a active low signal. Which indicate that the selected IO or Memory device is to be read and data is available on the data bus. (ii) WR : it is a active low signal which indicate that the data on the data bus are to be written into a selected memory or IO location. (iii) ALE : it is a +ve going pulse generated every time the 8085 begins an operation (machine cycle), which indicate that the bits on AD7-AD0 are address bits.
* Three status signals are available on chip: (i) IO/M : this is a status signal used to differentiate between IO and Memory operations. If it is high then IO operation and If it is low then Memory operation. (ii) S1 and S0 : status signals similar to IO/M, can identify various operations that are rarely used in the systems.

**Internal Architecture of 8085 Microprocessor:** The architecture of 8085 consists of three main sections, ALU (Arithmetic and Logical Unit), timing and control unit and Registers (shown in the following figure).



**Arithmetic and Logic Unit (ALU):**The ALU performs the actual numerical and logical operations.

* The ALU performs the following arithmetic and logical operations.
	+ Addition, Subtraction
	+ Logical AND, Logical OR, Logical Ex - OR
	+ Complement (logical NOT)
	+ Increment, Decrement
	+ Left shift, Right shift
	+ Clear, etc.
* ALU includes the accumulator, the temporary register, the arithmetic and logic circuits and flags. It always stores result of operations in Accumulator.

**Timing & Control Unit:** It generates timing and control signals, which are necessary for the execution of instructions.

* It controls data flow between CPU and peripherals (including memory).
* It provides status, control and timing signals, which are required for the operation of memory and I/O devices.
* **8085 System Bus:** Microprocessor communicates with memory and other devices (input and output) using three buses: Address Bus, Data Bus and Control Bus.
* **Address Bus:** The Address bus consists of 16 wires. The size of the address bus determines the size of memory, which can be used. To communicate with memory the microprocessor sends an address on the address bus to the memory. Address bus is unidirectional, *i.e.,*numbers only sent from microprocessor to memory.
* **Data Bus:** Bus is bidirectional. Size of the data bus determines what arithmetic can be done. Data bus also carries instructions from memory to the microprocessor.
* Memory size = 2A x D where, A denotes the address lines, and D denotes the data lines.
* **Control Bus:** Control bus are various lines which have specific functions for coordinating and controlling μP operations. The control bus carries control signals partly unidirectional, partly bidirectional. Control signals are things like read or write.

**Registers:** 8085 has six general purpose registers to store 8 bit data, these are identified as B, C, D, E, H and L . They can be combined as register pairs BC, DE and HL to perform some 16 bit operations.

* **Accumulator:** The accumulator is an 8 bit register included as a part of Arithmetic Logic Unit (ALU). This register is used to store 8 bit data and to perform arithmetic and logical operations. The result of an operation is stored in the accumulator.
* **Flag Register:** The ALU includes five flip-flops. They are called Zero (Z), Carry (CY), Sign (S), Parity (P) and Auxiliary Carry (AC) flags. The microprocessor uses these flags to test data conditions. The conditions (set or reset) of the flags are tested through the software instructions.  The combination of the flag register and the accumulator is called Program Status Word (PSW) and PSW is the 16-bit unit for stack operation.

**Flag registers:**



The flag register is a special purpose register and it is completely different from other registers in microprocessor. It consists of 8 bits and only 5 of them are useful. The other three are left vacant and are used in the future Intel versions.These 5 flags are set or reset (when value of flag is 1, then it is said to be set and when value is 0, then it is said to be reset) after an operation according to data condition of the result in the accumulator and other registers. The 5 flag registers are:

1. **Sign Flag:** It occupies the seventh bit of the flag register, which is also known as the most significant bit. It helps the programmer to know whether the number stored in the accumulator is positive or negative. If the sign flag is set, it means that number stored in the accumulator is negative, and if reset, then the number is positive.
2. **Zero Flag:**: It occupies the sixth bit of the flag register. It is set, when the operation performed in the ALU results in zero(all 8 bits are zero), otherwise it is reset. It helps in determining if two numbers are equal or not.
3. **Auxillary Carry Flag:** It occupies the fourth bit of the flag register. In an arithmetic operation, when a carry flag is generated by the third bit and passed on to the fourth bit, then Auxillary Carry flag is set. If not flag is reset. This flag is used internally for BCD(Binary-Coded decimal Number) operations.

**Note –** This is the only flag register in 8085 which is not accessible by user.

1. **Parity Flag:** It occupies the second bit of the flag register. This flag tests for number of 1’s in the accumulator. If the accumulator holds even number of 1’s, then this flag is set and it is said to even parity. On the other hand if the number of 1’s is odd, then it is reset and it is said to be odd parity.
2. **Carry Flag:** It occupies the zeroth bit of the flag register. If the arithmetic operation results in a carry(if result is more than 8 bit), then Carry Flag is set; otherwise it is reset
* **Interrupts In 8085**
* What is Interrupt?
* Interrupt is a mechanism by which an I/O or an instruction can suspend the normal execution of processor and get itself serviced. Generally, a particular task is assigned to that interrupt signal.
* In the microprocessor based system the interrupts are used for data transfer between the peripheral devices and the microprocessor.
* Interrupt Service Routine (ISR): A small program or a routine that when executed services the corresponding interrupting source is called as an ISR.
* Maskable/Non-Maskable Interrupt
* An interrupt that can be disabled by writing some instruction is known as Maskable Interrupt otherwise it is called Non-Maskable Interrupt.
* There are 6 pins available in 8085 for interrupt:
* 1. TRAP
* 2. RST 7.5
* 3. RST6.5
* 4. RST5.5
* 5. INTR
* 6. INTA
* Execution of Interrupts
* When there is an interrupt requests to the Microprocessor then after accepting the interrupts Microprocessor send the INTA (active low) signal to the peripheral.
* The vectored address of particular interrupt is stored in program counter.
* The processor executes an interrupt service routine (ISR) addressed in program counter.
* There are two types of interrupts used in 8085 Microprocessor:
* 1. Hardware Interrupts
* 2. Software Interrupts
* **Software Interrupts**
* A software interrupts is a particular instructions that can be inserted into the desired location in the rpogram. There are eight Software interrupts in 8085 Microprocessor.
* From RST0 to RST7.
* 1. RST0
* 2. RST1
* 3. RST2
* 4. RST3
* 5. RST4
* 6. RST5
* 7. RST6
* 8. RST7
* They allow the microprocessor to transfer program control from the main program to the subroutine program. After completing the subroutine program, the program control returns back to the main program.
* **Hardware Interrupt**
* As it have already discussed that there are 6 interrupt pins in the microprocessor used as
* Hardware Interrrupts given below:
* 1. TRAP
* 2. RST7.5
* 3. RST6.5
* 4. RST5.5
* 5. INTR

Note:

* INTA is not an interrupt. INTA is used by the Microprocessor for sending the acknowledgement.
* TRAP has highest priority and RST7.5 has second highest priority and so on.
* The Vector address of these interrupts is given below:
* Interrupt Vector Address
* RST7.5 003CH
* RST6.5 0034H
* RST5.5 002CH
* TRAP 0024H
* **TRAP**
* It is non maskable edge and level triggered interrupt. TRAP has the highest priority and vectores interrupt. Edge and level triggered means that the TRAP must go high and remain high until it is acknowledged. In case of sudden power failure, it executes a ISR and send the data from main memory to backup memory.
* As we know that TRAP can not be masked but it can be delayed using HOLD signal.
* This interrupt transfers the microprocessor's control to location 0024H.
* TRAP interrupts can only be masked by resetting the microprocessor. There is no other way to mask it.
* RST7.5
* It has the second highest priority. It is maskable and edge level triggered interrupt. The vector address of this interrupt is 003CH. Edge sensitive means input goes high and no need to maintain high state until it is recognized.
* It can also be reset or masked by resetting microprocessor. It can also be resettled by DI instruction.
* **RST6.5 and RST5.5**
* These are level triggered and maskable interrupts. When RST6.5 pin is at logic 1, INTE flip-flop is set.
* RST 6.5 has third highest priority and RST 5.5 has fourth highest priority.
* It can be masked by giving DI and SIM instructions or by resetting microprocessor.
* **INTR**
* It is level triggered and maskable interrupt.

The following sequence of events occurs

* when INTR signal goes high:
* 1. The 8085 checks the status of INTR signal during execution of each instruction.
* 2. If INTR signal is high, then 8085 complete its current instruction and sends active low interrupt acknowledge signal, if the interrupt is enabled.
* 3. On receiving the instruction, the 8085 save the address of next instruction on stack and execute received instruction
* **Program Counter (PC):** This 16 bit register deals with sequencing the execution of instruction. The microprocessor uses this register to sequence the execution of the instructions. The function of the program counter is to point to the memory address from which the next byte is to be fetched.
* **Stack Pointer (SP):** The stack pointer is also a 16 bit register used as a memory pointer. It points to a memory location in read-write memory, called the stack.
* **Instruction Register/Decoder:** Temporary store for the current instructions of a program. Latest instruction sent here from memory prior to execution. Decoder then takes instruction and decodes or interprets the instruction. Decoded instruction then passed to next stage.
* **Memory Address Register:** Holds address, received from PC of next program instruction.
* **Control Generator:** It generates signal within μP to carry out the instructions which have been decoded.
* **Register Selector:** This block controls the use of the register stack
* **General Purpose Registers:**μP requires extra registers for versatility. It can be used to store additional data during a program.

**Operations of Microprocessor:**The microprocessor performs the following four operations using address bus, data bus and control bus:

* Memory Read: Reads data (or instruction) from memory.
* Memory Write: Writes data (or instruction) into memory.
* I/O Read: Accepts data from input device.
* I/O Write: Sends data to output device.

**The 8085 Instruction Format:** An instruction is a command to the microprocessor to perform a given task on a specified data. Each instruction has two parts, one is task to be performed, called the operation code (opcode), and the second is the data to be operated on called the operand. The 8085 instruction set is classified according to word size.

* **One-Byte Instructions:** A 1-byte instruction includes the opcode and operand in the same byte. Operands are internal registers and are coded into the instruction.
* **Two-Byte Instructions:** In a two-byte instruction, the first byte specifies the operation code and the second byte specifies the operand. Source operand is a data byte immediately following the opcode.
* **Three-Byte Instructions:** In a three byte instruction, the first byte specifies the opcode and the following two bytes specify the 16-bit address. Note that, the second byte is the low-order address and the third byte is the high-order address.

**The 8085 Addressing Modes:** The various formats for specifying operands are called the addressing modes. For 8085, they are

* **Immediate Addressing:**
	+ Data is provided in the instruction.
	+ Load the immediate data to the destination provided.
	+ Example: MVI A, 12 H
* **Register Addressing:**
	+ Data is provided through the registers.
	+ Example: MOV B, C
* **Direct Addressing:**
	+ Used to accept data from outside devices to store in the accumulator or send the data stored in the accumulator to the outside device.
	+ Example: MOV A, [1000]
* **Indirect Addressing:**
	+ Effective address is calculated by the processor and the contents of the address is used to form a second address. The second address is where the data is stored.
	+ Example: MOV A, [[1000]]
* **Implicit addressing:**
	+ In this addressing mode the data itself specifies the data to be operated upon.
	+ Example: CMA ; Complement the contents of accumulator

**8085 Instruction set** An instruction is a binary pattern designed inside a microprocessor to perform a specific function. Each instruction is represented by 8 bit binary value. Instruction set can be categorised int0 5 types:

1. Data transfer instructions:
	* These instructions are used to transfer data from one register to another register, from memory to register or register to memory.
	* When an instruction of data transfer group is executed, data is transferred from the source to the destination without altering the contents of the source.
	* Examples: MOV, MVI, LXI, LDA, STA, etc.
2. Arithmetic instructions:
	* These instructions are used to perform arithmetic operations such as addition, subtraction, increment or decrement of the content of a register or memory.
	* Examples: ADD, ADC, ADI, DAD, SUB, INR, DCR, etc.
3. Logical instructions:
	* These instructions are used to perform logical operations such as AND, OR, compare, rotate etc.
	* Examples: ANA, ANI, ORA, ORI, XRA, CMA, CMC , STC, CMP, RLC, RAL, RAR, etc.
4. Branching Instructions:
	* These instructions are used to perform conditional and unconditional jump, subroutine call and return, and restart.
	* Examples: JZ, JNZ, JC, JNC, JP, JM, JPE, JPO, CALL, RET, RST, etc.
5. Machine Control Instructions:
	* These instructions control machine functions such as Halt, Interrupt, or do nothing.
	* The microprocessor operations related to data manipulation can be summarized in four functions: copying data, performing arithmetic operations, performing logical operations, testing for a given condition and alerting the program sequence.
	* Example: PUSH, POP, HLT, XTHL, NOP, EI, DI, etc.