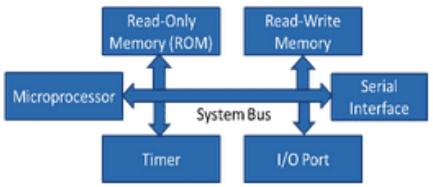
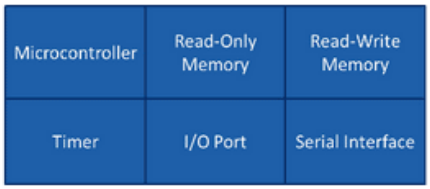
Dear Students, Don't limit your knowledge horizon, it's only a reference, and you can use other resources like class notes and book for more knowledge.
This is the solution of only important question.
Increase and decrease the content of answers according to marks.

1. **Define Microprocessor. Differentiate between Microprocessor and Microcontroller with example.**
   - Microprocessor is a multipurpose, programmable, and clock-driven, register based electronic device that reads binary instruction from the storage devices, accepts input from the input devices and processes data according to the instruction and produce result as a output. A computer with microprocessor is called as Microcomputer. Inside microprocessor there are ALU, Control unit and Register arrays (Figure of microprocessor).

Microprocessor	Micro Controller
	
Microprocessor is heart of Computer system.	Micro Controller is a heart of embedded system.
It is just a processor. Memory and I/O components have to be connected externally	Micro controller has external processor along with internal memory and i/O components
Since memory and I/O has to be connected externally, the circuit becomes large.	Since memory and I/O are present internally, the circuit is small.
Cannot be used in compact systems and hence inefficient	Can be used in compact systems and hence it is an efficient technique
Cost of the entire system increases	Cost of the entire system is low
Due to external components, the entire power consumption is high. Hence it is not suitable to used with devices running on stored power like batteries.	Since external components are low, total power consumption is less and can be used with devices running on stored power like batteries.
Most of the microprocessors do not have power saving features.	Most of the micro controllers have power saving modes like idle mode and power saving mode. This helps to reduce power consumption even further.
Since memory and I/O components are all external, each instruction will need external operation, hence it is relatively slower.	Since components are internal, most of the operations are internal instruction, hence speed is fast.
Microprocessor have less number of registers, hence more operations are memory based.	Micro controller have more number of registers, hence the programs are easier to write.
Microprocessors are based on von Neumann model/architecture where program and data are stored in same memory module	Micro controllers are based on Harvard architecture where program memory and Data memory are separate
Mainly used in personal computers	Used mainly in washing machine, MP3 players

2. Explain microprocessor as a CPU.

- Traditionally the computer block diagram is shown in the figure consisting of memory, input, output and CPU. CPU consists of ALU and the control unit. CPU consists of registers to store data, the ALU to perform arithmetic and logic operation, instruction decoders, counter and control lines .CPU components was designed with discrete components on various boards.

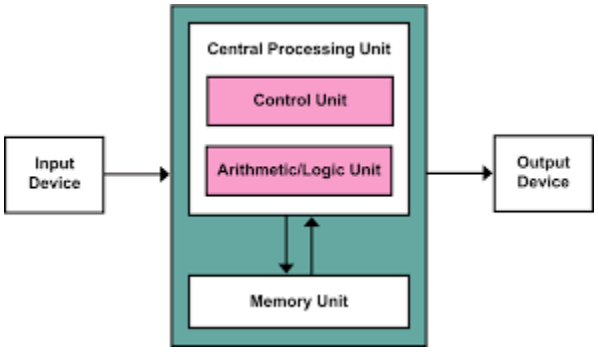


Figure 1: Block Diagram of Traditional Computer

- With the invention of integrated circuit technology, it became possible to build CPU in a single chip which is known as microprocessor. The Microprocessor as a CPU contains ALU for arithmetic and logical operation, Registers arrays for storing data and control unit for controlling overall activities. Microprocessor comes with all the blocks in a same package. Microprocessor accepts the input from the input devices processes according to the instruction provided and results the desired output through output devices as shown in figure below. The computer with microprocessor as a CPU is called as microprocessor. MPU (Microprocessor Unit) implies a complete processing unit with the necessary control signals .With the limited number of pins on microprocessor packages some of the signals such as control and multiplexed signal need to be generated by using discrete devices to make a microprocessor a complete functional unit.

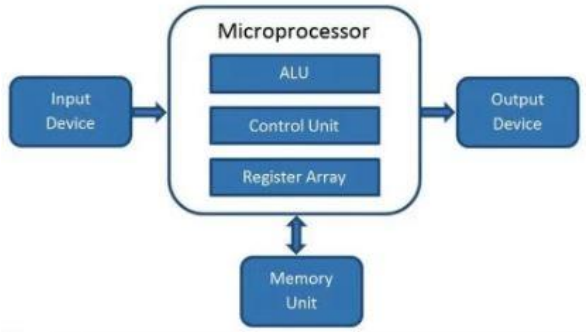


Figure 2: Block diagram of Microprocessor as CPU

3. Explain the organization of microprocessor based system with block diagram.

- The microprocessor is a semiconductor device (Integrated Circuit) manufactured by the VLSI (Very Large Scale Integration) technique. It includes the ALU, register arrays and control circuit on a single chip. A system designed using a microprocessor as its CPU is called a microcomputer. The Microprocessor based system (single board microcomputer) consists of microprocessor as CPU, semiconductor memories like EPROM and RAM, input device, output device and interfacing devices. The memories, input device, output device and interfacing devices are called peripherals. The block of Microprocessor based system are described as below:

ALU (Arithmetic/Logic Unit) – It performs arithmetic operations such as addition and subtraction, and logic operations as AND, OR, and XOR. Results are stored either in registers or in memory.

Register Array – It consists of various registers sometimes identified by letter such as B, C, D, E, H, L, IX, and IY. These registers are used to store data and addresses temporarily during the execution of a program.

Control Unit – The control unit provides the necessary timing and control signals to all the operations in the microcomputer. It controls the flow of data between the microprocessor and memory and peripherals. It controls overall activities of a computer.

Input – The input section transfers data and instructions in binary from the outside world to the microprocessor. It includes such devices as a keyboard, switches, a scanner, and an analog-to-digital converter.

Output – The output section transfers data from the microprocessor to such output devices as LED, CRT, printer, magnetic tape, or another computer.

Memory – It stores such binary information as instructions and data, and provides that information to the microprocessor. To execute programs, the microprocessor reads instructions and data from memory and performs the computing operations in its ALU section. Results are either transferred to the output section for display or stored in memory for later use. Memory may be ROM i.e. used to store program that do not need alterations and (Read Write Memory) R/WM i.e. used to store user program and data.

System bus – It is a communication path between the microprocessor and peripherals. The microprocessor communicates with only one peripheral at a time. The timing is provided by the control unit of the microprocessor. The system bus may contain Address Bus, Data bus and Control Bus.

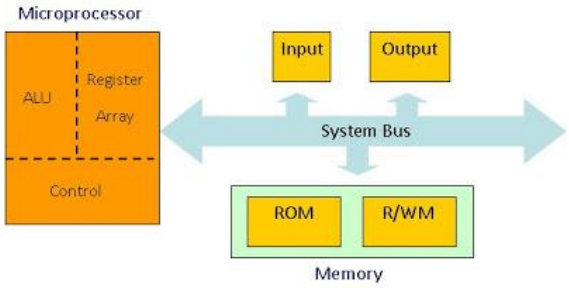


Figure 3 : Block Diagram Of Microprocessor Based system

4. **Explain the microprocessor architecture and its operations.**
   - The microprocessor is a programmable digital device, designed with registers, flip-flops and timing elements. The microprocessor has a set of instruction, designed internally to manipulate data and communicate with peripherals. This process of data manipulation and communication is determined by the logic design of the microprocessor, called the architecture.
   - The microprocessor can be programmed to perform functions on given data by selecting necessary instruction from its sets. These instructions are given to the microprocessor by writing them in the memory. Writing instruction and data is done through an input device such as keyboard. The instructions are stored in memory. The Microprocessor sequentially fetches the instruction from the memory, decodes it and executes the instruction. The sequence of fetch, decode and execute is continued until the microprocessor comes across a instruction to stop. The microprocessor reads or transfers one instruction at a time, matches it with its instruction sets and performs the data manipulation indicated by the instruction. The result can be stored in a memory or sent to such output devices as LED or a CRT terminal. Microprocessor can respond to external signal. It can be interrupted reset or asked to wait to synchronize with slower peripherals.
   - The function of microprocessor has following categories:
     a) Microprocessor-initiated operations
     b) Internal operations
     c) Peripherals Operations

     To perform these functions, microprocessor requires a group of logic circuit and a set of signal called as control signal.

5. **Explain the bus architecture of 8085 microprocessor.**
   - The microprocessor MPU performs various operations with peripheral devices or a memory location by using three sets of communication lines called buses: the address bus, the data bus and the control bus. And these three combined lines is called as system bus.

   Address bus:

   The address bus is a group of 16 lines generally called as A0 – A15 to carry a 16-bit address of memory location. In a computer system, each peripheral or memory location is identified by a

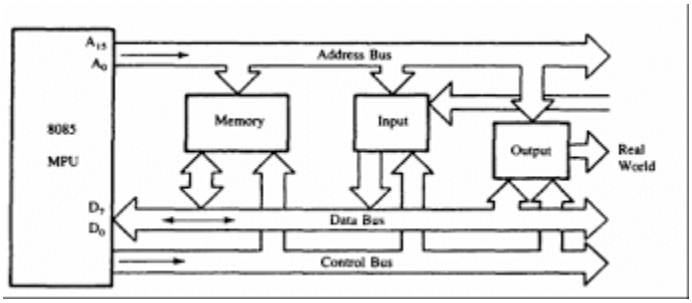
binary number called an address. The address bus is unidirectional, that means bit flow in only one direction from MPU to peripheral.
8085 MPU carries 16-bit address i.e. 216 = 65,536 or 64K memory locations.

Data Bus:

The data bus is a group of eight bidirectional lines used for data flow in both the directions between MPU and peripheral devices. The 8 data lines are manipulating 8-bit data ranging from 00 to FF i.e. (28 = 256) numbers from 0000 0000 -1111 1111.

Control bus:

Control bus is having various single lines used for sending control signals in the form of pulse to the memory and I/O devices. The MPU generates specific control signals to perform a particular operations. Some of these control signals of 8085 are Address Latch Enable (ALE), RD, WR.



6. Explain the 8085 microprocessor signals with block diagram.

1. Address Bus and Data Bus:

The address bus is a group of sixteen lines i.e. A0-A15. The address bus is unidirectional, i.e., bits flow in one direction from the microprocessor unit to the peripheral devices and uses the high order address bus.

2. Control and Status Signals:

- **ALE –** It is an Address Latch Enable signal. It goes high during first T state of a machine cycle and enables the lower 8-bits of the address, if its value is 1 otherwise data bus is activated.
- **IO/M' –** It is a status signal which determines whether the address is for input-output or memory. When it is high (1) the address on the address bus is for input-output devices. When it is low (0) the address on the address bus is for the memory.
- **SO, S1 –** These are status signals. They distinguish the various types of operations such as halt, reading, and instruction fetching or writing.

IO/M'	S1	S0	DATA BUS STATUS
0	1	1	Opcode fetch
0	1	0	Memory read
0	0	1	Memory write
1	1	0	I/O read
1	0	1	I/O write
1	1	1	Interrupt acknowledge
0	0	0	Halt

- **RD' –** It is a signal to control READ operation. When it is low the selected memory or input-output device is read.
- **WR' –** It is a signal to control WRITE operation. When it goes low the data on the data bus is written into the selected memory or I/O location.
- **READY –** It senses whether a peripheral is ready to transfer data or not. If READY is high(1) the peripheral is ready. If it is low(0) the microprocessor waits till it goes high. It is useful for interfacing low speed devices.

3. Power Supply and Clock Frequency:

- **Vcc –** +5v power supply
- **Vss –** Ground Reference
- **XI, X2 –** A crystal is connected at these two pins. The frequency is internally divided by two, therefore, to operate a system at 3MHZ the crystal should have frequency of 6MHZ.
- **CLK (OUT) –** This signal can be used as the system clock for other devices.

4. Interrupts and Peripheral Initiated Signals:

The 8085 microprocessor has five interrupt inputs. They are TRAP, RST 7.5, RST 6.5, RST 5.5, and INTR. These interrupts have a fixed priority of interrupt service. If two or more interrupts go high at the same time, the 8085 will service them on priority basis. The TRAP has the highest priority followed by RST 7.5, RST 6.5, RST 5.5. The priority of interrupts in 8085 is shown in the table.

TRAP - 1
RST 7.5 - 2
RST 6.5 - 3
RST 5.5 - 4
INTR – 5

The microprocessor acknowledges Interrupt Request by INTA signal. In addition to Interrupts, there are three externally initiated signals namely RESET, HOLD and READY. To respond to HOLD request, it has one signal called HLDA.

- **INTR –** It is an interrupt request signal.
- **INTA' –** It is an interrupt acknowledgment sent by the microprocessor after INTR is received.

5. Reset Signals:

- **RESET IN' –** When the signal on this pin is low (0), the program-counter is set to zero, the buses are tristated and the microprocessor unit is reset.
- **RESET OUT –** This signal indicates that the MPU is being reset. The signal can be used to reset other devices.

6. DMA Signals:

- **HOLD –** It indicates that another device is requesting the use of the address and data bus. Having received HOLD request the microprocessor relinquishes the use of the buses as soon as the current machine cycle is completed. Internal processing may continue. After the removal of the HOLD signal the processor regains the bus.
- **HLDA –** It is a signal which indicates that the hold request has been received after the removal of a HOLD request, the HLDA goes low.

7. Serial I/O Ports:

Serial transmission in 8085 is implemented by the two signals,

- **SID and SOD –** SID is a data line for serial input where as SOD is a data line for serial output.

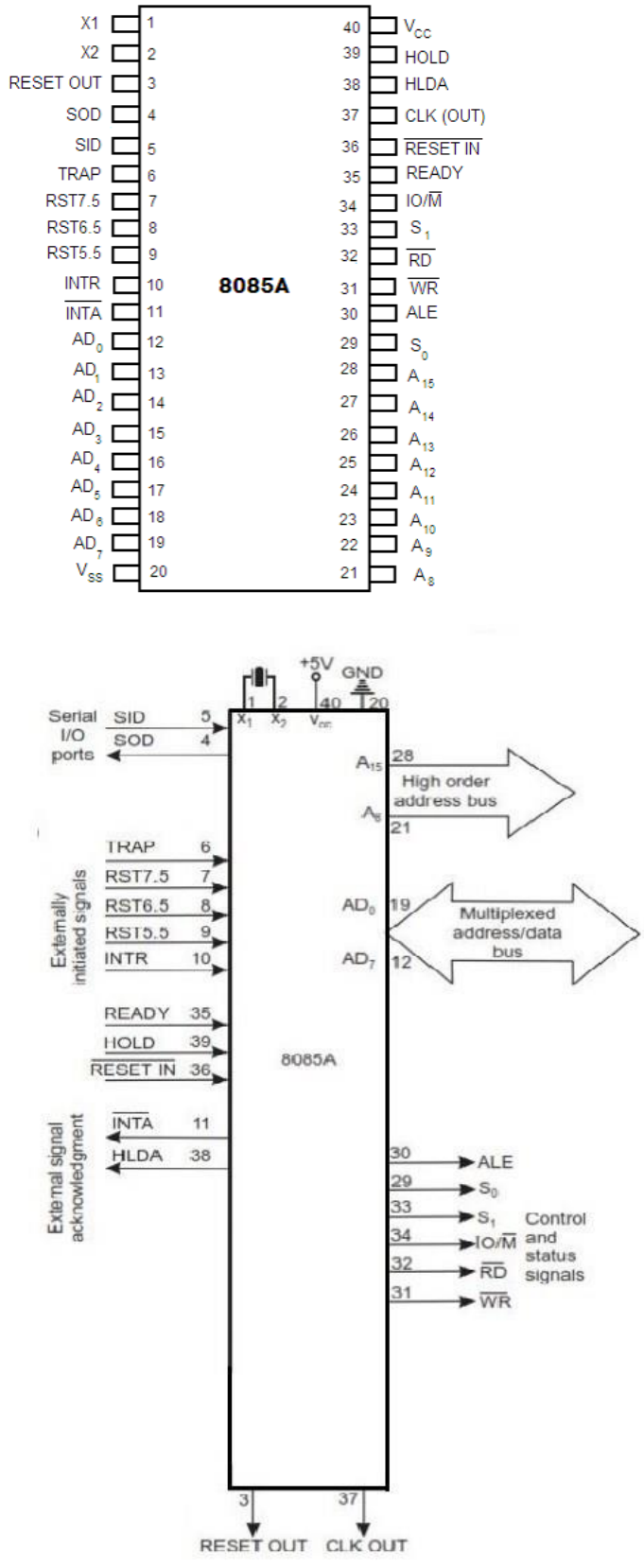


Figure 4: Pin Diagram of 8085 Microprocessor

7. **Explain the 8085 microprocessor with its functional diagram**.

- 8085 consists of the following functional units –

1. ALU

The ALU performs the actual arithmetic and logic operation such as 'add', 'subtract', 'AND', 'OR' e.t.c. It uses data from memory and from Accumulator to perform arithmetic operation and always stores result of operation in Accumulator. The ALU consists of accumulator, flag register and temporary register.

a. Accumulator: The accumulator is an 8-bit register that is 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. The accumulator is also identified as register A.

b. Flag register: 8085 has 8-bit flag register. There are only 5 active flags. Flags are flip-flops which are used to indicate the status of the accumulator and other register after the completion of operation. These flip-flops are set or reset according to the data condition of the result in the accumulator and other registers.

i. Sign flag(S):

Sign flag indicates whether the result of a mathematical or logical operation is negative or positive. If the result is negative, this flag will be set (i.e. S=1) and if the result is positive, the flag will be reset (i.e. S=0).

ii. Zero flag (Z):

Zero flag indicates whether the result of a mathematical or logical operation is zero or not. If the result of current operation is zero, the flag will be set (i.e. Z=1) otherwise the flag will be reset (Z=0).This flag will be modified by the result in the accumulator as well as in the other register.

iii. Auxiliary carry flag (AC):

In operation when a carry is generated by bit D3 and passes on to bit D4, the AC flag will be set otherwise AC flag will be reset. This flag is used only internally for BCD operation and is not available for the programmer to change the sequence of program with the jump instruction.

iv. Parity flag (P):

This flag indicates whether the current result is of even parity (no. of 1's is even) or odd parity (no. of 1's is odd).If even parity, P flag will be set otherwise reset.

v. Carry flag (CY):

This flag indicates whether during an addition or subtraction operation carry or borrow is generated or not. If carry or borrow is generated, the flag will be set otherwise reset.

2. Timing and control unit

This unit produces all the timing and control signal for all the operation. This unit synchronizes all the MP operations with the clock and generates the control signals necessary for communication between the MP and peripherals.

3. Instruction register and decoder

The instruction register and decoder are part of ALU. When an instruction is fetched from memory, it is loaded in the instruction register. The decoder decodes the instruction and establishes the sequence of events to follow. The IR is not programmable and cannot be accessed through any instruction.

4. Register array

The register unit of 8085 consists of

-Six general-purpose data registers B,C,D,E,H,L

-Two internal registers W and Z

-Two 16-bit address registers PC (program counter) and SP (stack pointer)

-One increment/decrement counter register

-And, one multiplexer (MUX)

The six general-purpose registers are used to store 8-bit data. They can be combined as register pairs BC, DE, and HL to perform some 16-bit operations.

The two internal registers W and Z are used to hold 8-bit data during the execution of some instructions, CALL and XCHG instructions.

SP is 16-bit registers used to point the address of data stored in the stack memory. It always indicates the top of the stack.

PC is 16-bit register used to point the address of the next instruction to be fetched and executed stored in the memory.

5. System bus

a. Data bus

It carries 'data', in binary form, between MP and other external units, such as memory. Typical size is 8 or 16 bits.

b. Address bus

It carries address of operand in binary form. Typical size is 16-bit.

c. Control Bus

Control Bus is various lines which have specific functions for coordinating and controlling MP operations.

E.g.: Read/Write control line.

6. Interrupt Control

Interrupt is a signal, which suspends the routine what the MP is doing, brings the control to perform the subroutine, completes it and returns to main routine. May be hardware or software interrupts. Some interrupts may be ignored (maskable), some cannot (non-maskable).

E.g. INTR, TRAP, RST 7.5, RST 6.5, RST 5.5

7. Serial I/O Control

The MP performs serial data input or output (one bit at a time). In serial transmission, data bits are sent over a single line, one bit at a time. The 8085 has two signals to implement the serial transmission: SID (serial input data) and SOD (serial output data).

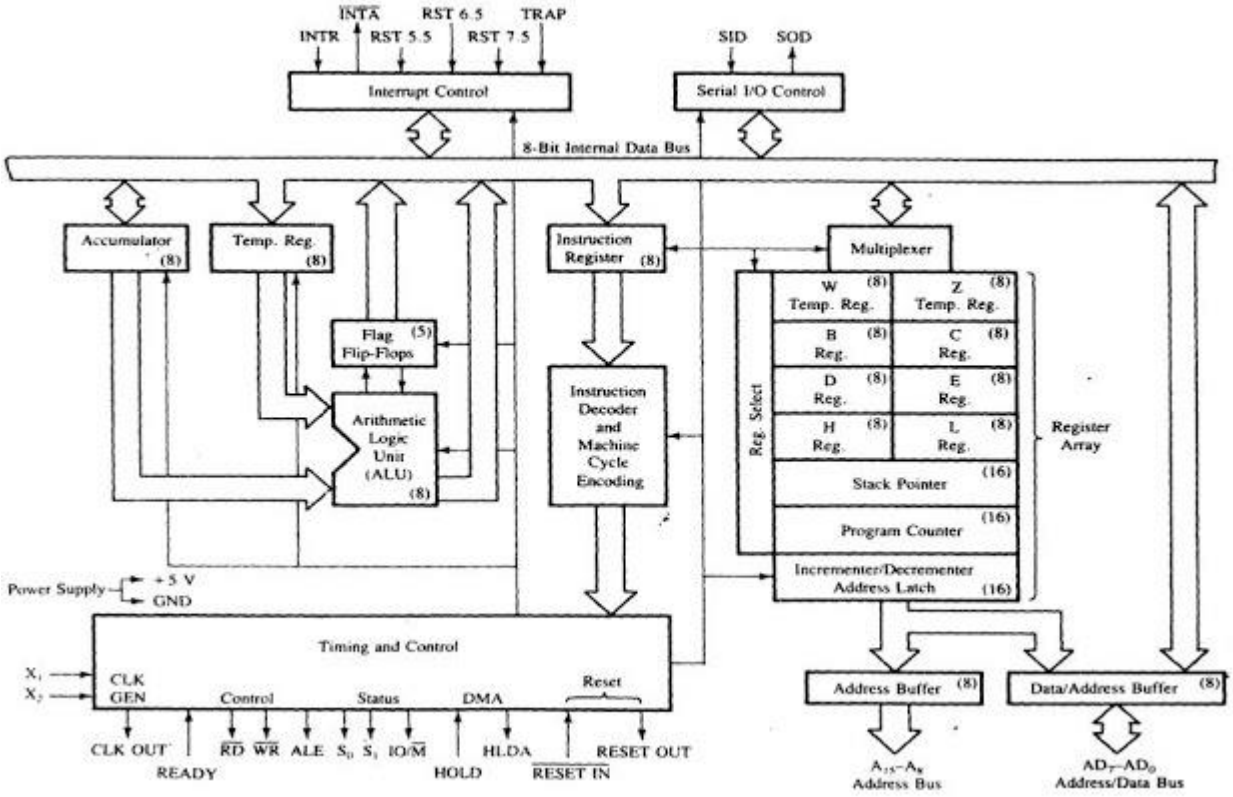


Figure 5: Block Diagram of 8085 Microprocessor

8. Explain the opcode fetch and memory read machine cycles for MVI A, 32H with timing for execution diagram.

Algorithm –

- Decide what the opcode is and what the data is. Here, opcode is 'MVI B' and data is 32.
- Assume the memory address of the opcode and the data. For example:

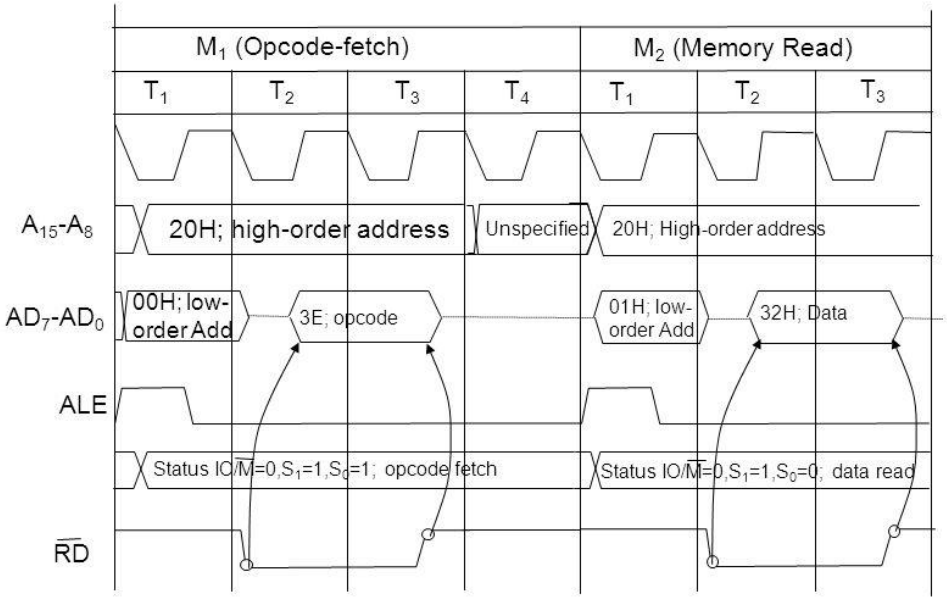
```
MVI B, 32
2000: Opcode
2001: 32
```

- The opcode fetch will be same in all the instructions.
- Only the read instruction of the opcode needs to be added in the successive T states.

- For the opcode read the IO/M (low active) = 0, S1 = 1 and S0 = 0. Also, only 3 T states will be required.

MVI A,32H Instruction

2000H 3EH ;MVI A, 32H
2001H 32H



In Opcode fetch (t1-t4 T states) –

- 00 – lower bit of address where opcode is stored, i.e., 00
- 20 – Higher bit of address where opcode is stored, i.e., 20.
- ALE – Provides signal for multiplexed address and data bus. Only in t1 it used as address bus to fetch lower bit of address otherwise it will be used as data bus.
  RD (low active) – Signal is 1 in t1, t2 & t4, no data is read by microprocessor. Signal is 0 in t3, data is read by microprocessor.
  WR (low active) – Signal is 1 throughout; no data is written by microprocessor.
  IO/M (low active), S0 and S1 – Signal is 1 in throughout; operation is performing on input/output.

In Opcode read (t5-t7 T states) –

- 00 – lower bit of address where opcode is stored, i.e, 01
- 20 – higher bit of address where opcode is stored, i.e, 20.
- ALE – Provides signal for multiplexed address and data bus. Only in t5 it used as address bus to fetch lower bit of address otherwise it will be used as data bus.
  RD (low active) – Signal is 1 in t1, t2 & t4, no data is read by microprocessor. Signal is 0 in t3, data is read by microprocessor.
  WR (low active) – Signal is 1 throughout, no data is written by microprocessor.
  IO/M (low active) and S1 – Signal is 1 in throughout, operation is performing on input/output.
  S0 – Signal is 0 throughout, operation is performing on memory.

9. Explain the 8085 microprocessor addressing modes with example.

- The way of specifying data to be operated by an instruction is called addressing mode.
- In 8085 microprocessor there are 5 types of addressing modes:

Immediate Addressing Mode –If the data is present within the instruction itself, then it is called immediate addressing mode. If the data is 8-bit, then the instruction will be of 2 bytes, if the data is of 16-bit then the instruction will be of 3 bytes. Examples: MVI B, 45 LXI H 3050 JMP address

Register Addressing Mode –If the data is present in the register and the register are specified in the instruction itself it is called as register addressing mode. Examples: MOV A, B ; ADD B ; INR A

Direct Addressing Mode – If the data to be operated is available inside a memory location and that memory location is directly specified as an operand is called as direct addressing mode. The operand is directly available in the instruction itself. Examples: LDA 2050 LHLD 2050 IN 35

Register Indirect Addressing Mode –If the data to be operated is available inside a memory location and that memory location is indirectly specified by a register pair is called as register indirect addressing mode. Examples: MOV A, M LDAX B LXI H, 9570

Implied/Implicit Addressing Mode –If the instruction operand is hidden or doesn't have any operand then it is called as implied addressing mode. Examples: CMA RRC RLC

11. **Write short notes on: Control and Status Signals, Flags, Instruction Cycle, Machine Cycle, T-States.**

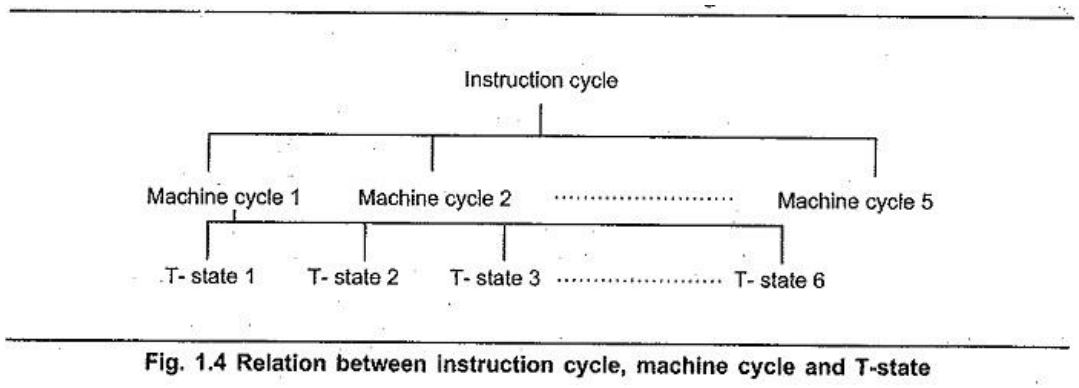


Fig. 1.4 Relation between instruction cycle, machine cycle and T-state

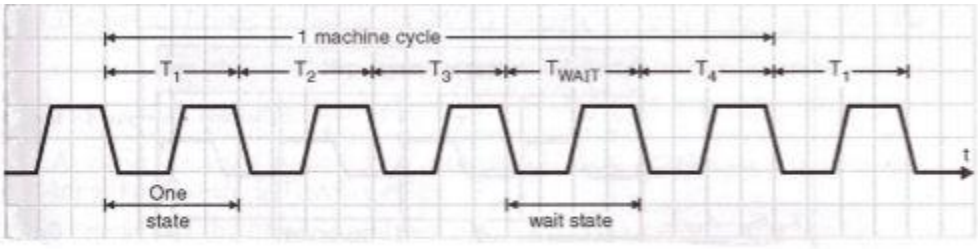
- During normal operation, the microprocessor sequentially fetches, decodes and executes one instruction after another until a halt instruction (HLT) is executed. The fetching, decoding and execution of a single instruction constitutes an **instruction cycle**, which consists of one to five read or write operations between processor and memory or input/output devices. Each memory or I/O operation requires a particular time period, called **machine cycle**. In other words, to move byte of data in or out of the microprocessor, a machine cycle is required. Each machine cycle consists of 3 to 6 clock periods/cycles, referred to as T-states. Therefore we can say that, one instruction cycle consists of one to five machine cycles and one machine cycle consists of three to six T-states i.e. three to six clock periods, as shown in the Figure above.

Instruction cycle:

The time a microprocessor needs to fetch and execute one entire instruction is known as an instruction cycle. There are typically four stages of an instruction cycle that the CPU carries out-

1. **Fetching the instruction:** The next instruction is fetched from the memory address that is currently stored in the program counter (PC) and stored in the instruction register (IR). At the end of the fetch operation, the PC points to the next instruction that will be read at the next cycle.
2. **Decode the instruction:** During this cycle the encoded instruction present in the IR (instruction register) is interpreted by the decoder.
3. **Read the effective address:** In case of a memory instruction (direct or indirect) the execution phase will be in the next clock pulse. If the instruction has an indirect address, the effective address is read from main memory and any required data is fetched from main memory to be processed and then placed into data registers (Clock Pulse: T3). If the instruction is direct, nothing is done at this clock pulse. If this is an I/O instruction or a Register instruction, the operation is performed (executed) at clock Pulse.
4. **Execute the instruction:** The control unit of the CPU passes the decoded information as a sequence of control signals to the relevant function units of the CPU to perform the actions required by the instruction such as reading values from registers, passing them to the ALU to perform mathematical or logic functions on them and writing the result back to a register. If the ALU is involved, it sends a condition signal back to the CU. The result generated by the operation is stored in the main memory or sent to an output device. Based on the condition of any feedback from the ALU, Program Counter may be updated to a different address from which the next instruction will be fetched.

Machine cycle:



Clock, T-state, machine cycle etc.

Machine cycle is defined as the time required completing one operation of accessing memory, I/O or acknowledging an external request. In other words The basic microprocessor operation such as reading a byte from I/O port or writing a byte to memory is called as machine cycle. A machine cycle consists of several T-states. In 8085 the cycle may consists of three to six T-states. The time TCY in the above figure is called as the machine cycle. Thus a machine cycle consists of several T-states.

T-state: T-state is defined as the one subdivision of the operation performed in one clock cycle period. These subdividion are internal states synchronized with the system clock and T-state is precisely equal to one clock period.One complete cycle of clock is called as T-state as shown in the above figure. The time intervals T1T1 orT2T2 are the examples of T-state. A T-state is measured from the falling edge of one clock pulse to the falling edge of the next clock pulse.

Various versions of 8086 have maximum clock frequency from 5MHz to 10MHz. Hence the minimum time for one T-state is between 100 to 200 n sec.

10. List the features of 8086 microprocessor with its block diagram.

The most prominent features of a 8086 microprocessor are as follows −

i. It has an instruction queue, which is capable of storing six instruction bytes from the memory resulting in faster processing.
ii. It was the first 16-bit processor having 16-bit ALU, 16-bit registers, internal data bus, and 16-bit external data bus resulting in faster processing.
iii. It is available in 3 versions based on the frequency of operation −

8086 → 5MHz
8086-2 → 8MHz
8086-1 → 10 MHz

iv. It uses two stages of pipelining, i.e. Fetch Stage and Execute Stage, which improves performance.
v. Fetch stage can prefetch up to 6 bytes of instructions and stores them in the queue.
vi. Execute stage executes these instructions.
vii. It has 256 vectored interrupts.
viii. It consists of 29,000 transistors.

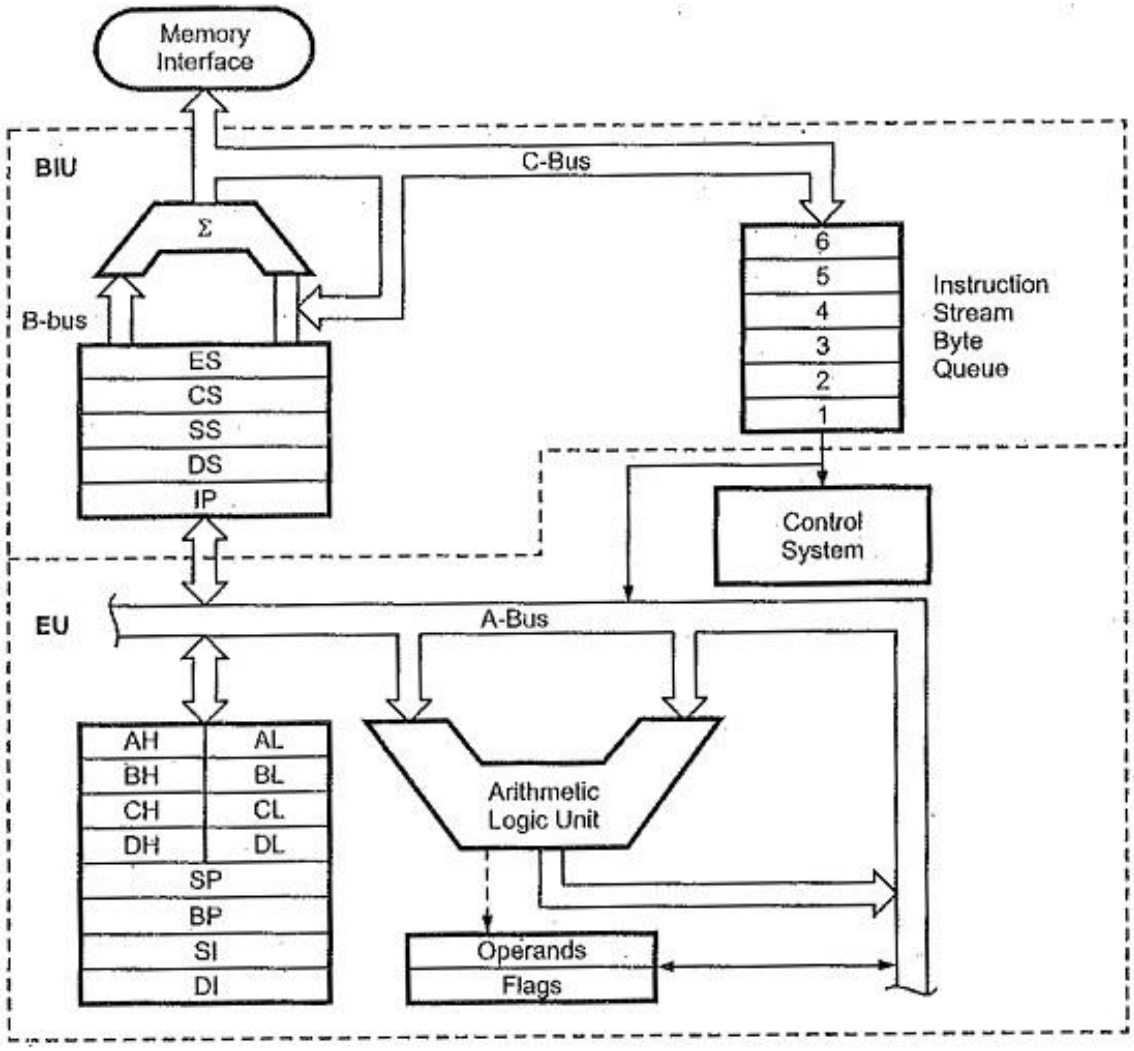


Fig. 6.2 8086 Internal block diagram