

CA3CRT07 -Microprocessors and PC Hardware (Core)

Theory:3 hrs. per week

Credits:3

Unit1: (10 hrs.)

Introduction : Evolution of microprocessors. Introduction to the concept of 8085 microprocessor: Intel 8085 introduction, Architecture ,Pin diagram, Instruction cycle, Timing diagrams, Interrupts of Intel 8085.

Unit 2 : (10 hrs.)

Instruction Set of Intel 8085 : Introduction, Instruction and data format, Addressing modes, Status flags, Intel 8085 instruction set.

Unit3: (12 hrs.)

Motherboard : Components of motherboard – expansion slots, Processor socket, coprocessor, memory modules, BIOS and CMOS, chipset. Super I/O chip, ROM BIOS, System buses- Processor Buses, Memory buses, I/O Bus(ISA,PCI Local Bus, AGP, USB), Motherboard selection criteria.

Unit4: (10 hrs.)

Hard disk: Hard Disk drive, Definitions, Hard Disk operations, Disk formatting, Basic hard disk drive components, Hard disk features, Hard disk drive installation procedure, FAT Disk, VFAT, FAT 32, NTFS.

Unit5: (12 hrs.)

Types of memory: Physical Memory, Memory modules:- SIMMs, DIMMs, RIMMs, Brief study of conventional base memory, Upper memory area, High memory area, Extended memory, Expanded memory.

Book of study :

1. B Ram -Fundamentals of microprocessors and microcontrollers, Seventh revised edition, Dhanpat Rai Publications.
2. Manahar Lotia and Pradeep Nair- All about motherboard, First edition, 2005, BPB Publications..
3. Manahar Lotia and Pradeep Nair- Modern all about Hard Disk Drive , First edition, BPB publications.

References:

1. Scott Mueller - Upgrading and repairing PCs , 18 th Edition, Pearson.
2. R S. Gaonkar- Micro processor Architecture, Programming and applications with 8085, Sixth Edition, PENRAM International Publishing.

MICROPROCESSOR ARCHITECTURE INTRODUCTION

The microprocessor(s) is the central processing unit (CPU) of a computer. It is the heart of the computer.

INTEL 8085

Intel 8085 is an 8-bit, NMOS microprocessor. It is a 40 pin I.C. package fabricated on a single LSI chip. The Intel 8085 uses a single - 5Vc supply for its operation. Its clock speed is about 3 MHz. The clock cycle is 320 ns. It has 80 basic instructions and 246 opcodes.

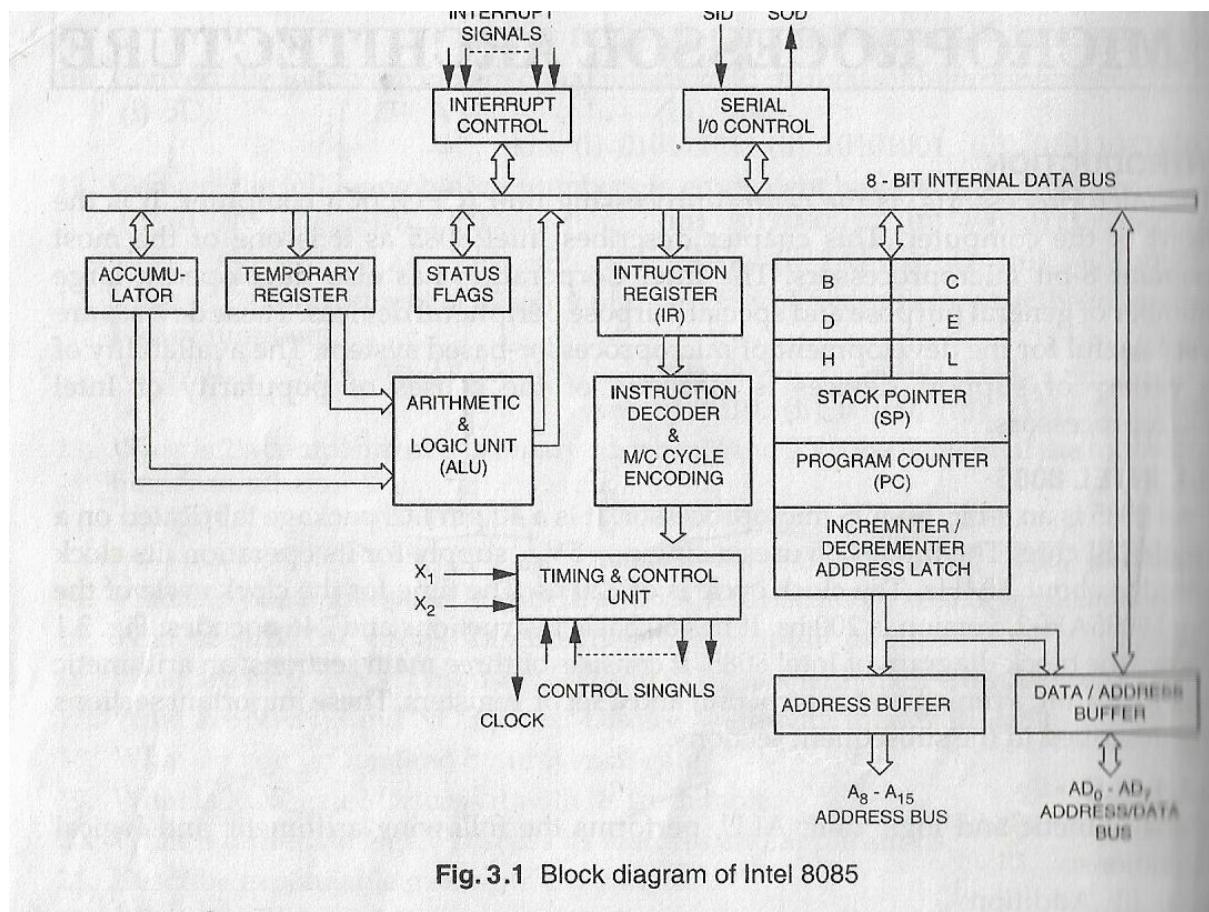


Fig. 3.1 Block diagram of Intel 8085

The above shows the block diagram of Intel 8085. It consists of three main sections: an arithmetic and logic unit, a timing and control unit and a set of registers.

ALU

The arithmetic and logic unit, ALU, performs the following arithmetic and logical operations:

- (i) Addition
- (ii) Subtraction
- (iii) Logical AND
- (iv) Logical OR
- (v) Logical EXCLUSIVE OR
- (vi) Complement (logical NOT)
- (vii) Increment (add 1)
- (viii) Decrement (subtract 1)

(ix) Left shift, Rotate left, Rotate right

(x) Clear etc.

Timing and Control Unit

The timing and control unit is a section of the CPU. 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. It controls the entire operations of the microprocessor and peripherals connected to it. Thus, the control unit of the CPU acts as the brain of the computer system.

Registers

Registers are used by the microprocessor for temporary storage and manipulation of data and instructions. Data remains in the registers till they are sent to the memory or I/O devices. Intel 8085 microprocessor has the following registers:

- (i) One 8-bit accumulator (ACC) i.e. register A
- (ii) Six 8-bit general purpose registers. These are B, C, D, E, H and L
- (iii) One 16-bit stack pointer, SP
- (iv) One 16-bit program counter, PC
- (v) Instruction register
- (vi) Temporary register

In addition to the above mentioned registers the 8085 microprocessor contains a set of five flip-flops which serve as flags (or status flags). A flag (or status flag) is a flip-flop which indicates some condition which arises after the execution of an arithmetic or logical instruction.

Accumulator (ACC).

The accumulator is an 8-bit register associated with the ALU. The register 'A' in the 8085 is an accumulator. It is used to hold one of the operands of an arithmetic or logical operation. It serves as one input to the ALU. The other operand for an arithmetic or logical operation may be stored either in memory or in one of the general-purpose registers. The final result of an arithmetic or logical operation is placed in the accumulator.

General-Purpose Registers.

The 8085 microprocessor contains six general purpose registers. They are: B, C, D, E, H and L register. To hold 16 bit data a combination of two 8-bit registers can be employed. The combination of two 8-bit registers is known as a register-pair. The valid register pairs in the 8085 are: B-C, D-L and H-L. The H-L pair is used to act as memory pointer and for this purpose it holds the 16-bit address of a memory location. The general-purpose registers and the accumulator are accessible to programmer.

Program Counter (PC).

It is a 16-bit special-purpose register. It is used to hold the memory address of the next instruction to be executed. It keeps the track of memory addresses of the instructions in a program while they are being executed. The microprocessor increments the content of the program counter during the execution of instruction so that it points to the address of the next instruction in the program at the end of the execution of an instruction.

Stack Pointer (SP).

It is a 16-bit special function register. The stack is a sequence of memory locations set aside by a programmer to store/retrieve the contents of accumulator, flags, program counter and general-purpose registers during the execution of a program. Any portion of the memory can

be used as stack. During the execution of a program sometimes it becomes necessary to save the contents of some registers which are needed for some other operations in the subsequent steps of the program. The contents of such registers are saved in the stack. Then the registers are used for some other operations. After completing the needed operations the contents which were saved in the stack are brought back to the registers. The SP holds the address of the top element of data stored in the stack.

Instruction Register.

The instruction register holds the opcode (operation code or instruction code) of the instruction which is being decoded and executed.

Temporary Register.

It is an 8-bit register associated with the ALU. It holds data during an arithmetic/logical operation. It is used by the microprocessor. It is not accessible to programmer.

Flags.

The Intel 8085 microprocessor contains five flip-flops to serve as status flags. The flip flops are set or reset according to the conditions which arise during an arithmetic or logical operation.

The five status flags of Intel 8085 are

- (i) Carry Flag(CS)
- (ii) Parity Flag (P)
- (iii) Auxiliary Carry Flag (AC)
- (iv) Zero Flag (Z)
- (v) Sign Flag (S)

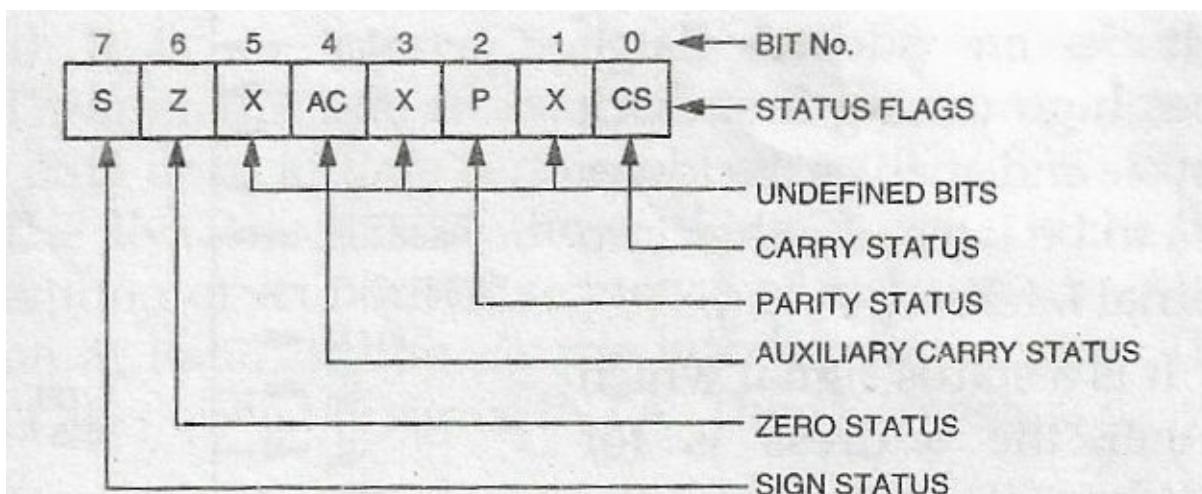


Fig. 3.2 (a) Status Flags of Intel 8085

ADD CB AND E9

$$\begin{array}{r} \text{CB} = 11001011 \\ \text{E9} = 11101001 \\ \hline \end{array}$$

10110100

RESULT IS NON - ZERO,
Z IS SET TO 0.

THERE IS CARRY,
CS IS SET TO 1.

THERE ARE 4 NUMBERS OF
1s, P IS SET TO 1.

MSB OF THE SUM IS 1,
S IS SET TO 1.

THERE IS A CARRY FROM
3rd BIT TO 4th BIT.
AC IS SET TO 1.

Fig. 3.2 (b) Status Flags for ADD operation

If a flip-flop for a particular flag is set, it indicates 1. When it is reset, it indicates 0.

Carry Flag (CS).

After the execution of an arithmetic instruction if a carry produced, the carry flag CS is set to 1, otherwise it is 0. The carry flag is set or reset in case of addition as well as subtraction. After the addition of two 8-bit numbers, if the sum is larger than 8 bits, a carry is produced; and the carry flag is set to 1. In case of subtraction, if borrow occurs, the carry flag is set to 1. The carry flag holds carry out of the most significant bit resulting from the execution of an arithmetic operation.

Parity Flag (P).

The parity status flag P is set to 1, if the result of an arithmetic or logical operation contains an even number of 1s. It is reset i.e. it is 0, if the result contain odd number of 1s.

Auxiliary Carry Flag (AC).

The auxiliary carry flag AC holds carry out of the bit number 3 to the bit number 4 resulting from the execution of an arithmetic operation.

Zero Flag (Z).

The zero status flag Z is set to 1, if the result of an arithmetic or logical operation is 0. If the result is not zero, the flag is set to 0.

Sign Flag (S).

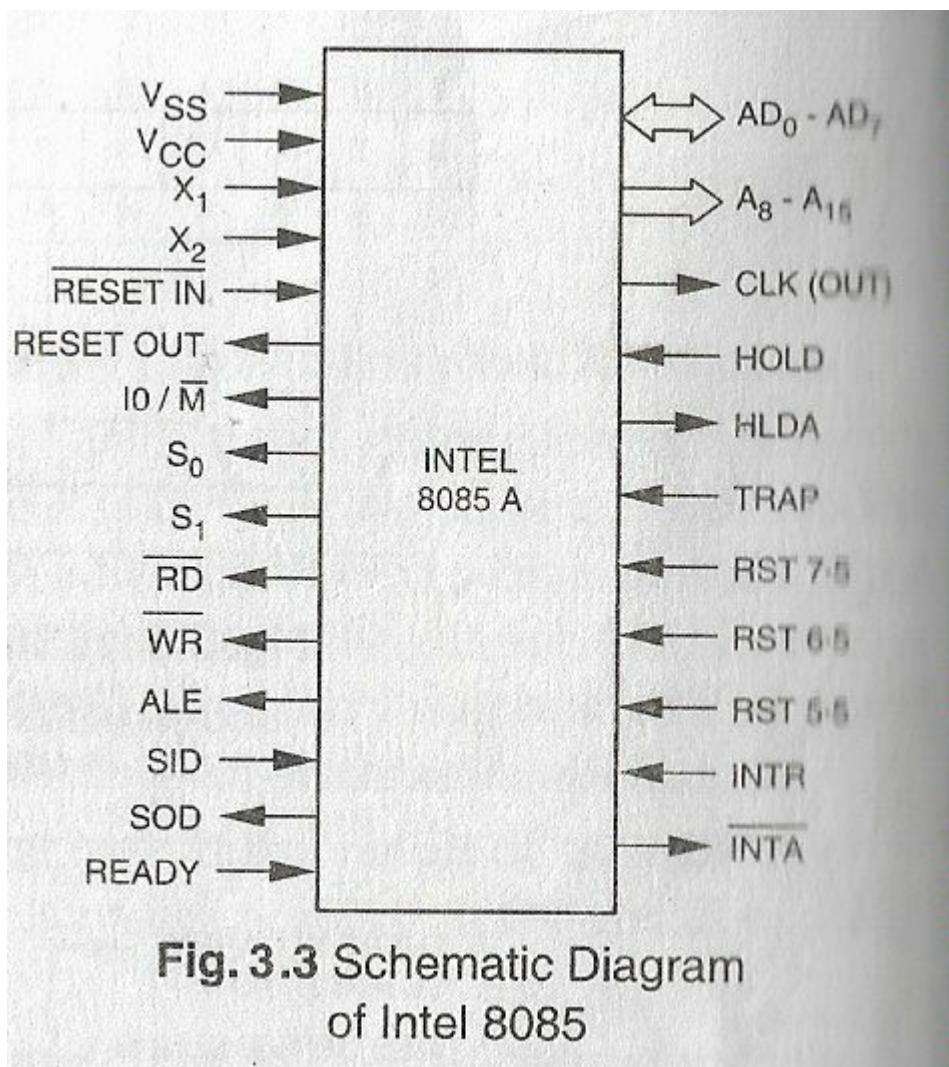
The sign flag S is set to 1, if the result of an arithmetic or logical operation is negative. If the result is positive, the sign flag is set to 0. The sign flag has its significance only when signed arithmetic operation is performed. To represent a signed number the most significant bit is reserved by the programmer to represent the sign of a number. In other words the MSB is used as a sign bit. It represents the sign of the number. If a number is negative, the sign bit is 1. For positive number, the sign bit is 0.

PSW. five bits indicate the five status flags and three bits are undefined. The combination of these 8 bits is called Program Status Word (PSW).

Data and Address Bus

The Intel 8085 is an 8-bit microprocessor. Its data bus is 8-bit wide and hence, 8 bits of data can be transmitted in parallel from or to the microprocessor. The Intel 8085 requires a 16-bit wide address bus as the memory addresses are of 16 bits. The 8 most significant bits of the address are transmitted by the address bus, A-bus (pins A8 to A15). The 8 least significant bits of the address are transmitted by address/data bus, AD bus (pins AD0-AD7). The address/data bus transmits data and address at different moments. At a particular moment it transmits either data or address. Thus, the AD bus operates in time shared mode. This technique is known as multiplexing.

Pin Configuration



**Fig. 3.3 Schematic Diagram
of Intel 8085**

The above figure shows the schematic diagram of Intel 8085.

A8 - A15 (output). These are address bus and are used for the most significant bits of the memory address or 8 bits of I/O address.

AD0 - AD7 (input/output). These are time multiplexed address/data bus i.e. they serve dual purposes. They are used for the least significant 8 bits of the memory address or I/O address during the first clock cycle of a machine cycle. Again they are used for data during second and third click cycles.

ALE (output). It is an address latch enable signal. It goes high during first clock cycle of a machine cycle and enables the lower 8 bits of the address to be latched either into memory or external latch.

I_O/M (output). It is a status signal which distinguishes whether the address is for memory or I/O. When it goes high, the address on the address bus is for an I/O device. When it goes low, the address on the address bus is for a memory location.

SO, S1 (output). These are status signals sent by the microprocessor to distinguish the various types of operation given in Table 3.1.

Table 3.1 Status Codes for Intel 8085

S_1	S_0	<i>Operations</i>
0	0	HALT
0	1	WRITE
1	0	READ
1	1	FETCH

RD (output). When microprocessor reads data or codes from a memory location or an input device, it is called READ operation. RD is a signal sent by the microprocessor to the memory /input device to control READ operation. When it goes low, the selected memory or input device is read.

WR (output). When microprocessor sends data to a memory location or an output device, it is called WRITE operation. WR is a signal sent by the microprocessor to the memory/output device to control WRITE operation. When it goes low, the data which is on the data bus, is written into the selected memory or sent to the output device.

READY (input). It is a signal sent by an input or output device to the microprocessor. This signal indicates that the input or output device is ready to send or receive data. The microprocessor examines READY signal before it performs data transfer operation. A slow input or output device is connected to the microprocessor through READY line. When READY is high, it indicates that the input or output is ready to send or receive data. When READY is low, the microprocessor waits till READY becomes high. The microprocessor examines the status of READY signal in the second clock cycle of the machine cycle.

HOLD (input). When another device of the computer system, requires address and data buses for data transfer, it sends HOLD signal to the microprocessor. After receiving the HOLD request, the microprocessor sends out a HLDA (HOLD Acknowledge) signal to the device. Then the microprocessor leaves the control over the buses as soon as the current machine cycle is completed. Internal processing may continue. The microprocessor regains the control over the buses after the HOLD signal is high, it indicates that the input or output device is removed.

HLDA (output). It is a HOLD acknowledge signal sent out by the microprocessor after receiving the HOLD signal. It is sent to the device which has issued the HOLD signal: After the removal of the HOLD signal, the HLDA goes low and thereafter the microprocessor takes over the buses.

INTR (input). It is an interrupt signal sent by an external device to the microprocessor. Through this line an external device informs the microprocessor that it is ready to transfer data or to initiate certain operations. The 8085 microprocessor has 5 interrupt lines. The INTR is one of them. When it goes high, the microprocessor suspends the execution of its normal sequence of instructions. After completing the current instruction at hand, it attends the interrupting device. The microprocessor issues an interrupt acknowledge signal INTA. Then it transfers data or takes any other action as required.

INTA (output). It is an interrupt acknowledge signal issued by the microprocessor after receiving an interrupt request from an external device. It is a low active signal.

RST 5.5, 6.5, 7.5 and TRAP (inputs). These are interrupts. When an interrupt is recognised the next instruction is executed from a fixed location in the memory as given below:

<i>Line</i>	<i>Location from which next instruction is picked up</i>
TRAP	0024
RST 5.5	002C
RST 6.5	0034
RST 7.5	003C

RST 7.5, RST 6.5 and RST 5.5 are the restart interrupts. Each of them has a programmable mask. The TRAP has the highest priority among interrupts. It is a maskable interrupt. It is unaffected by any mask or interrupt enable. The order of priority of interrupts is as follows:

TRAP (highest priority)

RST 7.5

RST 6.5

RST 5.5

INTR (lowest priority).

RESET IN (input). It resets the program counter to zero. It also resets interrupt enable and HLDA flip-flops. It does not affect any other flag or register except the instruction register. The CPU is held in reset condition as long as RESET is applied.

RESET OUT (output). It indicates that the CPU is being reset.

X1,X2(input). These are terminals to be connected to an external crystal oscillator which drives an internal circuitry of the microprocessor to produce a suitable clock for the operation of the microprocessor.

CLK (output). It is a clock output for the user, which can be used for other digital ICs. Its frequency is same at which the processor operates.

SID(input) It is data line for serial input. The data on this line is loaded into the 7th bit of the accumulator when RIM instruction is executed.

SOD (output), It is a data line for serial output. The 7th bit of the accumulator is output on SOD line when SIM instruction is executed.

Vcc +5 Volts supply

Vss ground reference

Intel 8085 Instructions

A computer receives data from the user, processes data and sends the result back to the user. The computer simply performs a given task on specified data in response to certain instructions. An instruction is a command given to the computer to perform a specified operation on given data.

MOV r1,r2. The content of register r2 is moved to register r1. For example, instruction MOV A, B moves the content of register B to register A. The instruction MOV B, A moves the content of register A to register B.

MOV M, r. The content of register r is moved to the memory location whose address is in H-L pair. For example, MOV M, A will transfer the content of accumulator to the memory location specified by H-L pair.

MVI r, data. The data given just after an instruction are moved to the register r. For example, the instruction MVI A, 05 moves 05 to register A.

LXI rp, data 16-bit. 16-bit immediate data are moved to the register pair rp. For example, the instruction LXI H, 2400H moves 2400 to H-L pair.

LDA addr. The content of the memory location whose address is specified within the Instruction itself is moved to the accumulator. For example, the instruction LDA 2400H moves the content of the memory location 2400H to the accumulator.

STA addr. The content of the accumulator is moved to the memory location whose address is specified within the instruction itself. For example, the instruction 2500H moves the content of the accumulator to memory location 2500H

ADD r. The content of the register r is added to the content of the accumulator. For example, the instruction ADD B adds the content of register B to the content of accumulator and places the sum in the accumulator.

SUB r. The content of register r is subtracted from the content of the accumulator and the result is placed in the accumulator. For example, the instruction SUB C subtracts the content of register C from the content of the accumulator and places result in the accumulator.

RAL. The content of the accumulator is rotated left one bit through carry.

INR r. The content of register r is incremented by one.

IN Port-address. This instruction transfers data from input device or port. The data available at the input port or device is moved to the accumulator.

OUT Port-address. This instruction transfers data from the processor to the output port or device. The content of the accumulator is transferred to the output port device.

Opcode and Operands

Each instruction contains two parts: operation code (opcode) and operand. The part of an instruction which specifies the task to be performed by the computer is called opcode. The second part of the instruction is the data to be operated on, and it is called operand. The operand (or data) given in the instruction may be in various forms such as 8-bit or 16-bit data, 8-bit or 16-bit address, internal registers or a register or memory location. In some instructions the operand is implicit. When operand is a register it is understood that the data is the content of the register.

Opcode and Operands

Each instruction contains two parts: operation code (opcode) and operand. The part of an instruction which specifies the task to be performed by the computer is called opcode. The second part of the instruction is the data to be operated on, and it is called operand. The operand (or data) given in the instruction may be in various forms such as 8-bit or 16-bit data, 8-bit or 16-bit address, internal registers or a register or memory location. In some instructions the operand is implicit. When operand is a register it is understood that the data is the content of the register.

Instruction Word Size

A digital computer understands instructions written in binary codes (machine codes). The machine codes of all instructions are not of the same length. According to the word size the Intel 8085 instructions are classified into the following three types:

- (1) 1-byte instruction
- (2) 2-byte instruction
- (3) 3-byte instruction

One-Byte instruction.

Examples of one-byte instructions are:

MOV A, B. Move the content of register B to register A. 78H is the opcode for MOV A, B. The opcode 78H can be written in the binary form as 01111000. The 1st two bits, i.e., 01 are for MOV operation; the next three bits 111 are the binary code for register A and the last three bits 000 are the binary code for register B.

ADD B. Add the content of register B to the content of the accumulator. 80H is the opcode for the instruction ADD B. In this instruction one of the operands is register B (its content) which is indicated in the instruction itself. In this type of instruction (arithmetic group of instruction) it is assumed that the other operand is in the accumulator. The opcode 80H in the binary form is 10000000. The first five bits, i.e. 10000 specify the operation to be performed, i.e. ADD operation. The last three bits 000 are the code for register B for 8085 microprocessor.

All one-byte instructions contain information regarding operands in the opcode itself.

Two-Byte Instruction.

In a two-byte instruction the 1st byte of the instruction is its opcode and the 2nd byte is either data or address. Examples are:

- (i) MVI B, 05; Move 05 to register B.
06, 05; MVI B, 05 in the code form.

The 1st byte 06 is the opcode for MVI B and the 2nd byte 05 is the data which is to be moved to register B.

- (ii) IN 01 : Read data at port B.

DB, 01; IN 01 in the code form.

DB is the opcode for the instruction IN and 01 is the address of a port. A two-byte instruction is stored in two consecutive memory locations.

Three-Byte Instruction.

In a three byte instruction the 1st byte of the instruction is opcode and the 2nd and 3rd bytes are either 16-bit data or 16-bit address. Examples are:

- (i) LXI H, 2400H : Load H-L pair with 2400H.
21, 00, 24 ;LXI H, 2400H in the code form.

The 1st byte 21 is the opcode for the instruction LXI H. The 2nd byte 00 is 8 LSBs of the data (2400H), which is loaded into register L. The 3rd byte 24 is 8 MSBs of the data (2400H), which is loaded into register H.

(1) LDA 2500H ; get the content of the memory location 2500H into the accumulator.
3A, 00, 25; LDA 2500H in the code form.

The 1st byte 3A is the opcode for the instruction LDA. The 2nd byte 00 is 8 LSBs of the address of the memory location-2500H. The 3rd byte 25 is 8 MSBs of the address of the memory location 2500H. In this instruction the data is the content of the memory

3.2 INSTRUCTION CYCLE

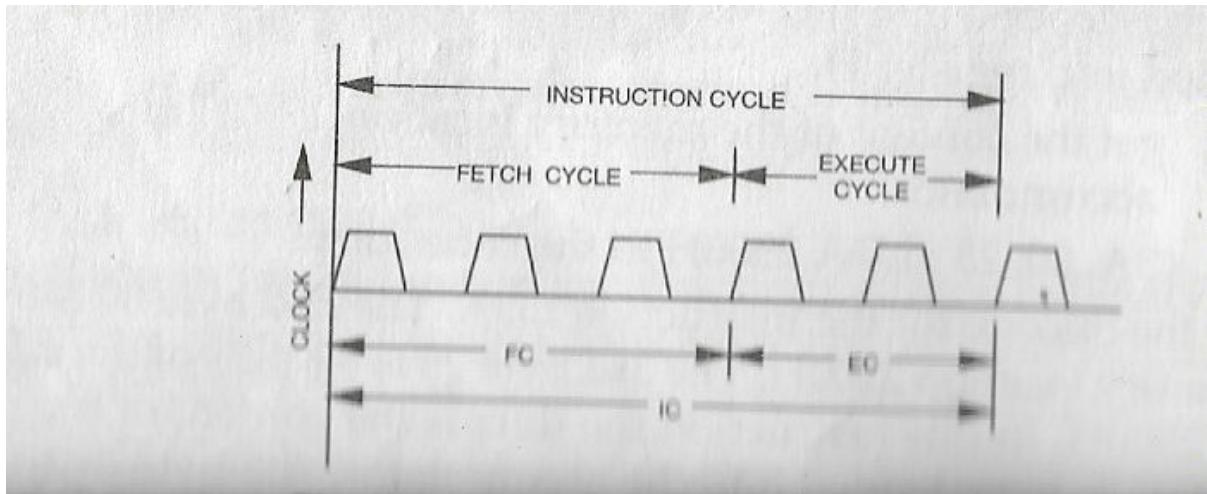
An instruction is a command given to the computer to perform a specified operation on given data. To perform a particular task a programmer writes a sequence of instructions, called a program. Program and data are stored in the memory. The CPU fetches one instruction from the memory at a time and executes it. It executes all the instructions of a program one by one to produce to the final result. The necessary steps that a CPU carries out to fetch an instruction and necessary data from the memory, and to execute it, constitute an instruction cycle. An instruction cycle consists of a fetch cycle and execute cycle, In fetch cycle a CPU fetches opcode (the machine code of an instruction) from the memory. The necessary steps which are carried out to fetch an opcode from the memory, constitute a fetch cycle. The necessary steps which are carried out to get data, if any, from the memory and to perform the specific operation specified in an instruction, constitute an execute cycle. The time required to fetch an opcode (FC) is a fixed slot of time while the time required to execute an instruction (EC) is variable which depends on the type of instruction to be executed. The total time required to execute an instruction is given by

$$IC = FC + EC$$

3.2.1 Fetch Operation

The 1st byte of an instruction is its opcode. An instruction may be more than one byte long. The other bytes are data or operand address. The program counter (PC) keeps the memory address of the next instruction to be executed. In the beginning of a fetch cycle the content of the program counter, which is the address of the memory location where opcode is available, is sent to the memory. The memory places the opcode on the data bus so as to transfer it to the CPU. The entire operation of fetching an opcode takes three clock cycles. A slow memory may take more time. In case of a slow memory the CPU has to wait till the memory sends the opcode. The clock cycle for which the CPU waits is

called wait cycle.



3.2.2 Execute Operation

The opcode fetched from the memory goes to the data register, DR (data/address buffer in Intel 8085) and then to instruction register, IR. From the instruction register it goes to the decoder circuitry which decodes the instruction. The decoder circuitry is within the microprocessor. After the instruction is decoded, execution begins. If the operand is in the general purpose registers, execution is immediately performed. The time taken in decoding and execution is one clock cycle. If an instruction contains data or operand address which are still in the memory, the CPU has to perform some read operations to get the desired data. After receiving the data it performs execute operation. A read cycle is similar to a fetch cycle. In case of a read cycle the quantity received from the memory are data or operand address instead of an opcode. In some instructions write operation is performed. In write cycle data are sent from the CPU to the memory or an output device. Thus, in some cases an execute cycle may involve one or more read or write cycles or both. Figure shows fetch cycle.

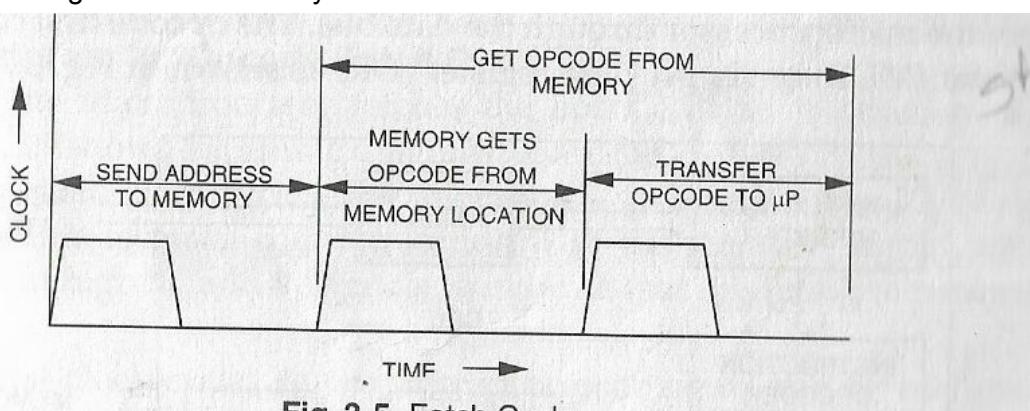


Fig. 3.5 Fetch Cycle

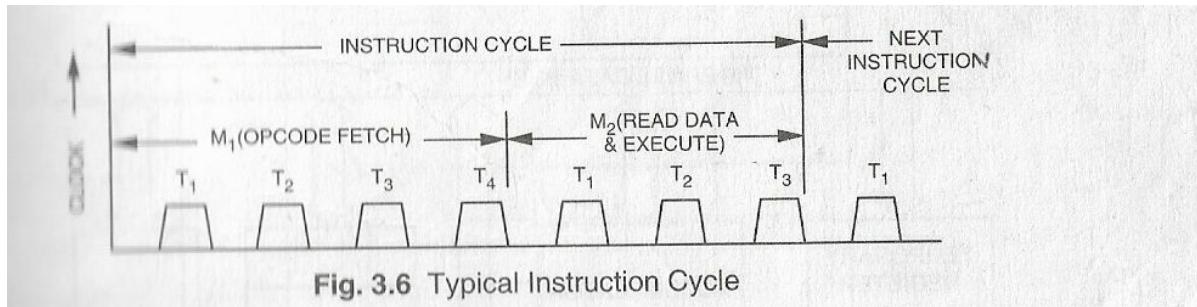
Machine Cycle and State

The necessary steps carried out to perform the operation of accessing either memory or I/O device, constitute a machine cycle. In other words necessary steps carried out to perform a fetch, a read or a write operation constitute a Machine cycle.

In a machine cycle one basic operation such as opcode fetch, memory read, memory write, I/O read or I/O write is performed. An instruction cycle consists of several machine cycles.

The opcode of an instruction is fetched in the first machine cycle of an instruction cycle. Most

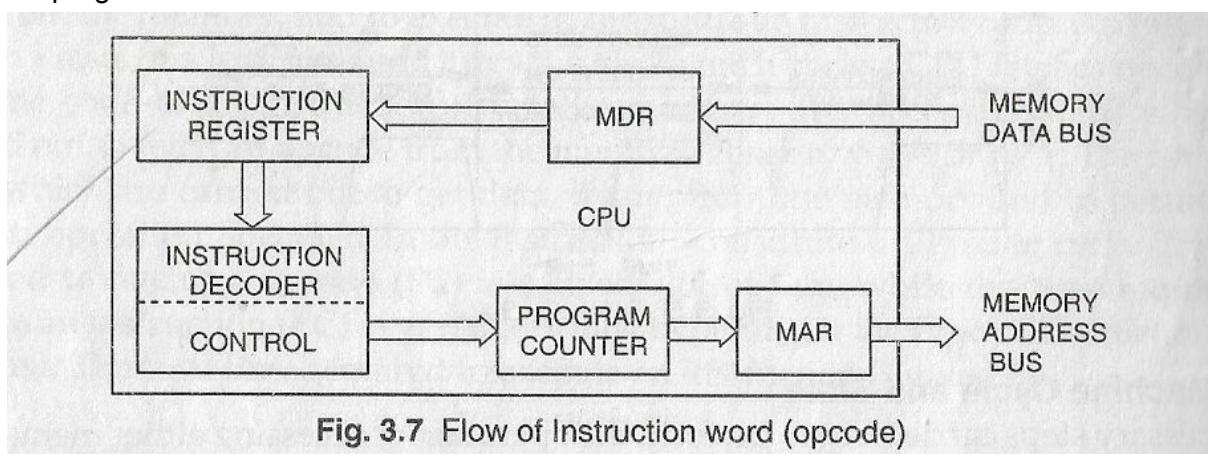
of the single-byte instructions require only one machine cycle to fetch the opcode and execute the instruction. Two-byte and three-byte instructions require more than one machine cycle. Additional machine cycles are needed to read data from or to write data into the memory or I/O devices. Figure shows instruction cycle for MVI r, data. It has two machine cycles: one for fetching opcode, and the other for reading data from the memory and to execute the instruction.



One subdivision of an operation performed in one clock cycle is called a state or T-state. The subdivisions are internal states synchronised with the system clock. So, One clock cycle of the system clock is referred to as a state.

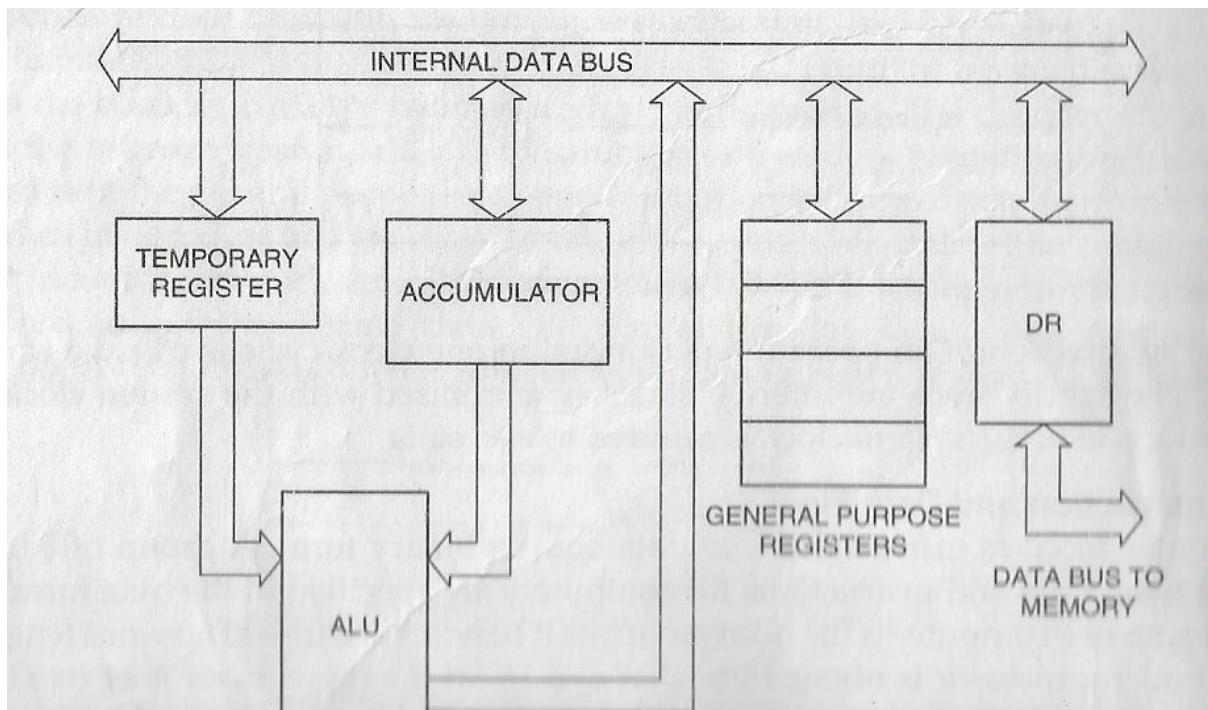
A Instruction and Data Flow

A computer receives instructions and data both in binary form. Two kinds of words namely instruction word (instruction code or opcode) and data word are processed during an instruction cycle. In the beginning of a fetch cycle the content of the program counter is transferred to a special register known as memory address register, MAR or simply address register, AR; (address buffer in Intel 8085). The content of MAR is transferred to the memory through the address bus. By sending certain control signals to the memory the microprocessor also indicates that it wants to read the content of the memory. The decoder circuitry in the memory activated and the memory understands what is to be done. Then the memory sends opcode to the microprocessor through the data bus. The opcode first comes in memory data register (MDR) or simply data register (DR). In case of Intel 8085, there is a data buffer for this purpose. The operation code is then placed in the instruction register (IR). The instruction is decoded by instruction decoder and it is executed. Finally, the content of the program counter is incremented.



The execution of an instruction requires the flow of data word in the most of the instructions. The flow of a data word is shown in Fig. A data word is received either from the memory or

input device. The data word flows to the processor through the data bus and is placed in the accumulator or any other general purpose register depending upon the instruction. After the execution of an instruction the data is placed in a register or a memory location. After the execution of a program the result (data) is placed in the memory or sent to an output device. When a data word is written into the memory, it is also held in MDR (or DR or Data Buffer) until the write operation is Complete.

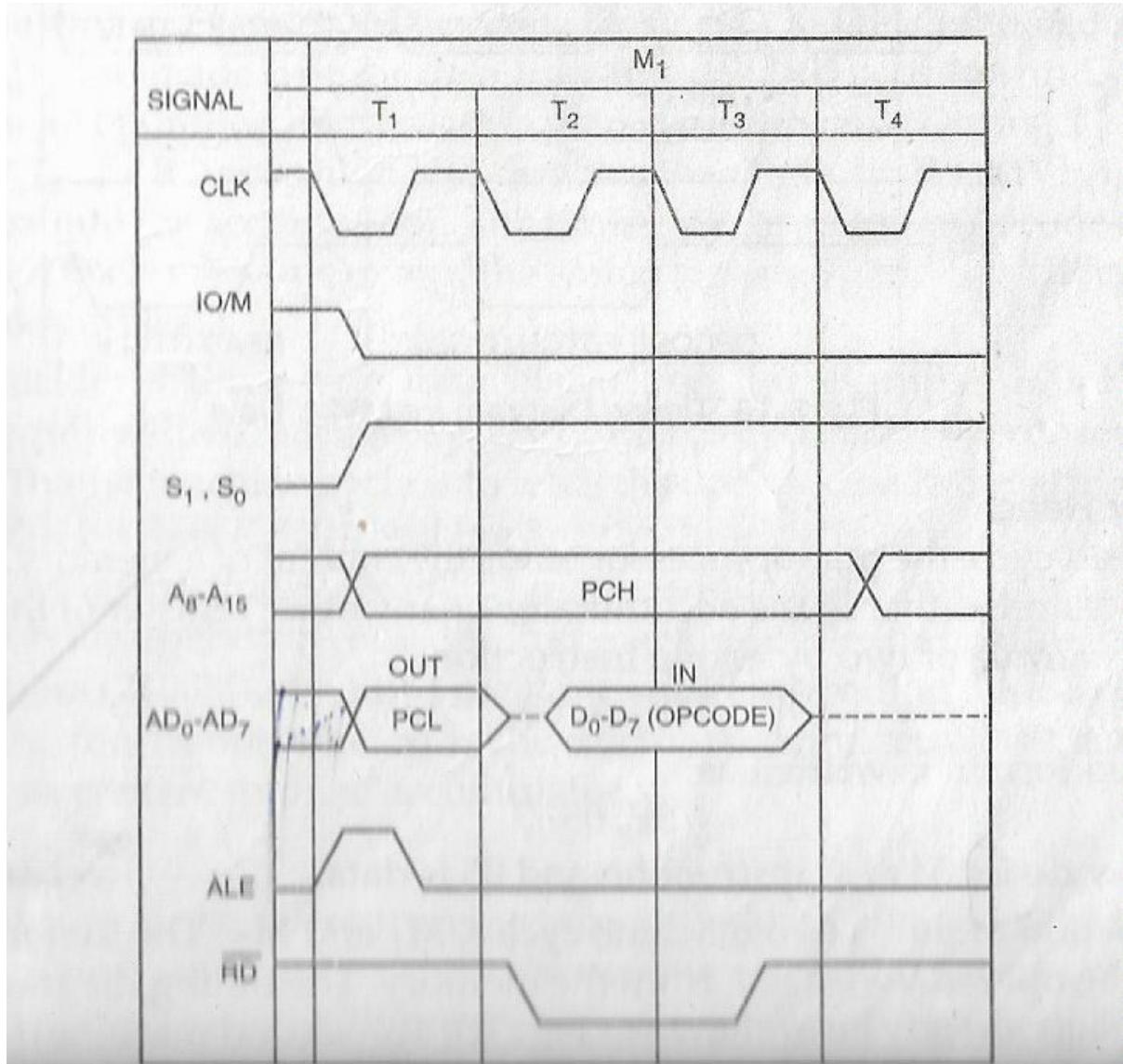


TIMING DIAGRAM

The necessary steps which are carried out in a machine cycle can be represented graphically. Such a graphical representation is called timing diagram.

Timing Diagram for Opcode Fetch Cycle

In à feich cycle the microprocessor fetches the opcode of an instruction from the memory.



Figures shows the timing diagram for an opcode fetch cycle. T₁, T₂, T₃, and T₄ are consecutive four clock cycles. The microprocessor issues a low IO/M signal to indicate that it wants to make communication with the memory. Again the micro-processor sends out high S₀, and S₁ signals to indicate that it is going to perform fetch (Falion.)

During the first clock cycle, T₁, the microprocessor sends out the address of the memory location where the opcode is available. The 16-bit memory address is sent through the address bus A and address/data bus AD. The 8 MSBs of the memory address are sent over the A bus, and 8 LSBs of the memory address over AD bus. Since, the AD bus is needed to transfer data during subsequent clock cycles, it is used in time multiplexed mode. Therefore, it has to be made available to carry data during T₂ and T₃. To accomplish this the microprocessor sends an address latch enable signal ALE to latch the 8 LSBs of the memory address either in the memory or an external latch.

During T₂, AD bus becomes ready to carry data. In T₂ the microprocessor makes RD low. Now, memory gets the opcode from the specified location and places it on the data bus. During T₃, the opcode is placed in the instruction register, IR which is within the

microprocessor. The memory is disabled when RD goes high during T3. The fetch cycle is completed by T3. The opcode is decoded in T4.

If an instruction is two or three bytes long, it requires more machine cycles. The first machine cycle M1, is for fetching the opcode from the memory. Subsequent machine cycles M2, M3 etc. are required either to read data or address from the memory or I/O device or to write data into the memory or I/O devices. Figure shows the timing diagram for a two byte instruction MVT r, data. The first machine cycle M1, shows fetch cycle, and M2 a data read cycle.

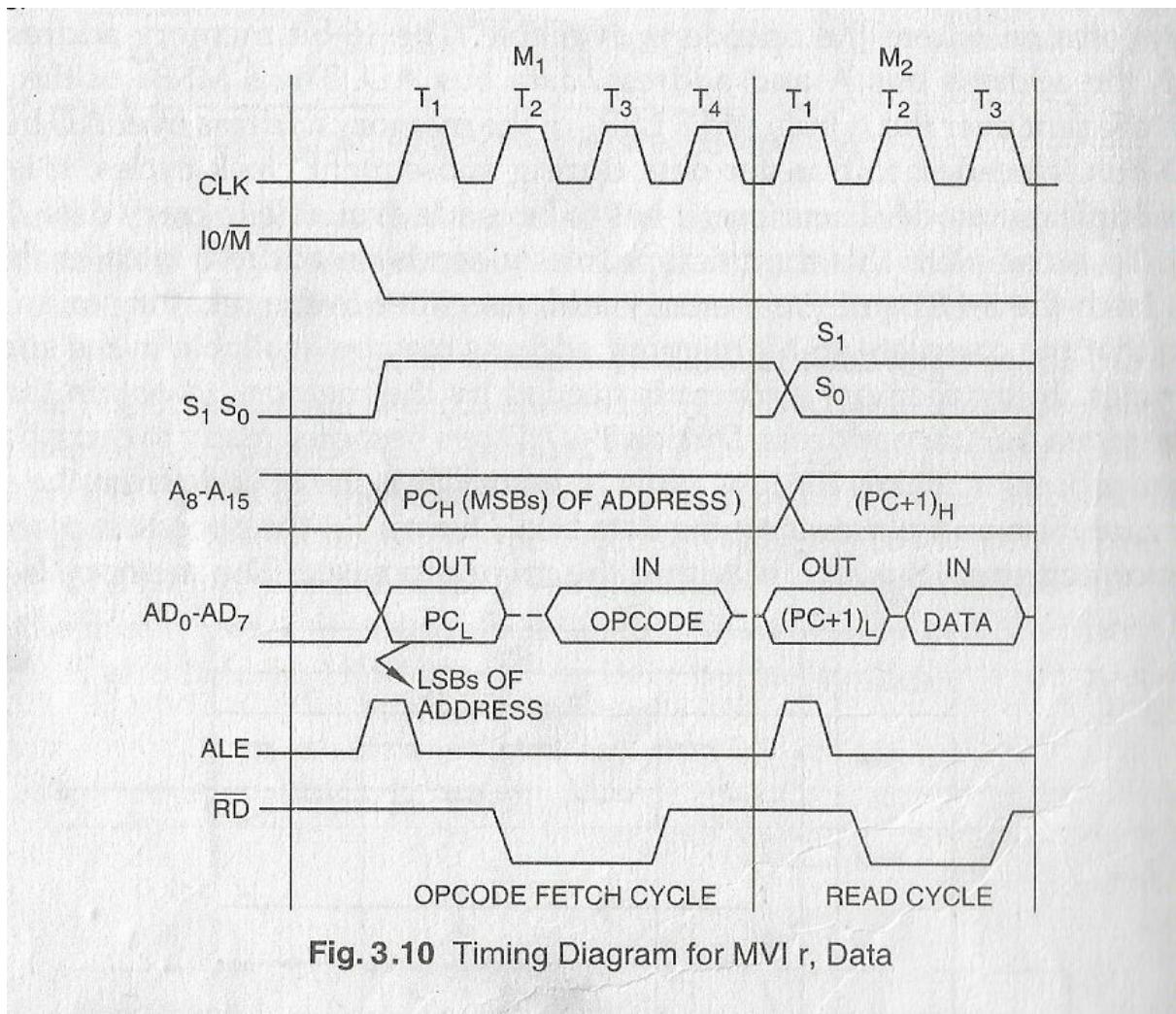


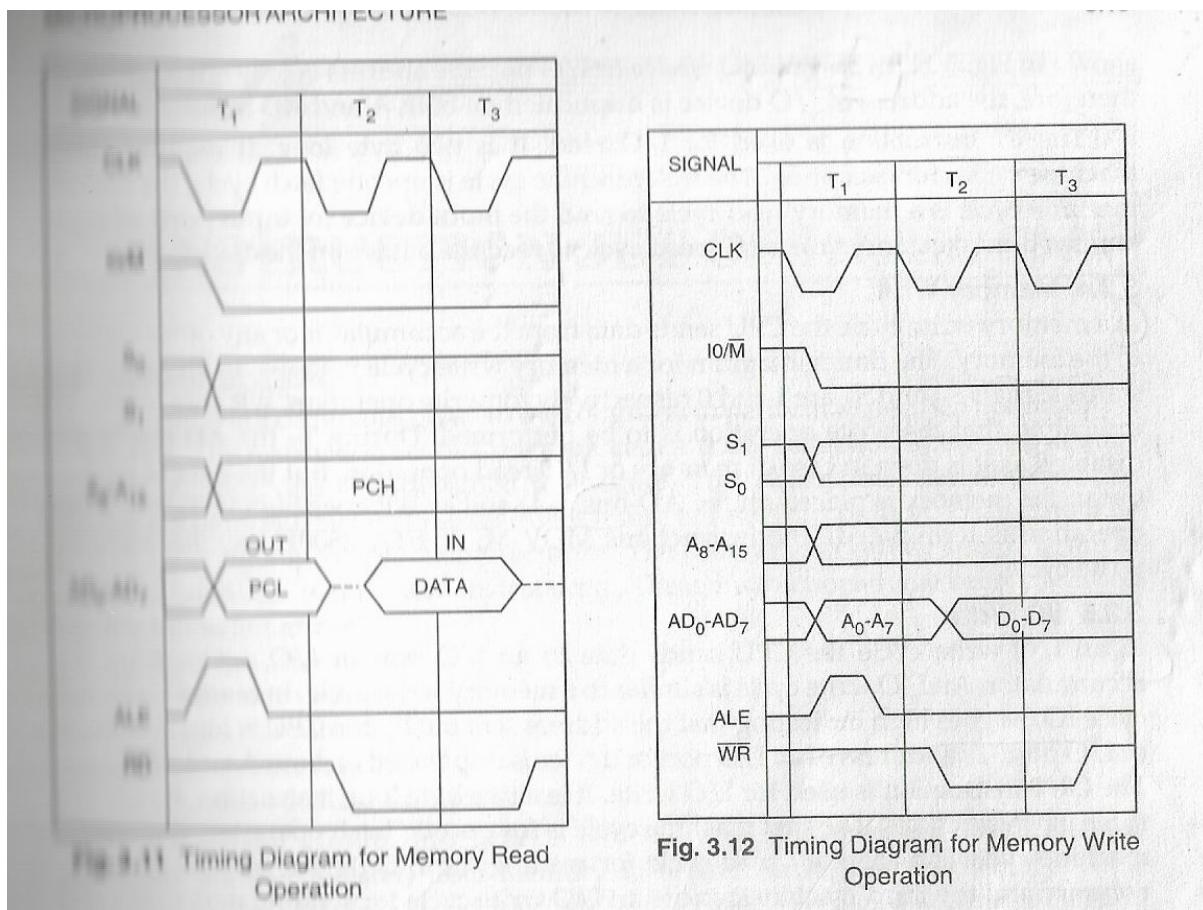
Fig. 3.10 Timing Diagram for MVI r, Data

Memory Read

In a memory read cycle the microprocessor reads the content of a memory location. The content is then placed either in the accumulator or any other register of the CPU.

Consider an example of two byte long instruction. MVI A, 05. In the coded form it is written as 3E, 05 where 3E is opcode for MVI A instruction and 05 is data.

This instruction requires two machine cycles, M1 and M2. The first machine cycle M1 is to fetch the operation code 3E from the memory. The second machine cycle M2 is for reading the data(05) from the memory. It is a memory read cycle.



Here IO/M goes low indicating that the address is for memory. S₁ and S₀ are set to 1 and 0 respectively for read operation. On the address lines A₈ - A₁₅, the 8 MSBs of the memory address of the data (05) are sent. During T₁, 8 LSBs of the memory address of the data are sent on AD₀ - AD₇. During T₂, 8 LSBs of the address is latched and AD₀ - AD₇ are made free for data transmission. RD goes low in T₁ to enable the memory for read operation. Now, data is placed on data bus. During T₃ the data enters into CPU. In T₃, RD goes high and disables the memory. For MVI A, 05 instruction data enters into the accumulator.

I/O Read

In an I/O read cycle the microprocessor reads the data available at an input port or input device. The data is placed in the accumulator. An I/O read cycle is similar to memory read cycle. The only difference between a memory ready cycle and I/O read cycle is that signal IO/M goes high in case of I/O read. It indicates that the address on the A-bus is for an input device. In case of I/O device or I/O port the address is only 8-bits long and therefore, the address of I/O device is duplicated on both A and AD buses.

The IN instruction is used for I/O read. It is two byte long. It requires three machine cycles for execution. The first machine cycle is opcode fetch cycle, the second machine cycle is a memory read cycle to read the input device (or input port) address and the third machine cycle is I/O read cycle to read the data from the device or port.

Memory Write

In a memory write cycle the CPU sends data from the accumulator or any other register to the memory. The status signals S₀ and S₁ are 1 and 0 respectively for write operation. WR goes low in T₂ indicating that the write operation is to be performed. During T₂ the

AD-bus is not disabled as it is done in case of memory or I/O read operation. But the data to be sent out to the memory is placed on the AD-bus. As soon as WR goes high in T3, the write operation is terminated. The instructions MOV M, A ; STA 2500IH etc. use memory write cycle.

I/O Write

In an I/O write cycle the CPU sends data to an I/O port or I/O device from the accumulator. An I/O write cycle is similar to a memory write cycle. In case of I/O write cycle IO/M goes high indicating that the address sent out by the CPU is for I/O device or I/O port. The address of an I/O port or device is duplicated on both A and AD buses. The OUT instruction is used for I/O write. It is a two byte long instruction. It requires three machine cycles. The first machine cycle is for opcode fetch operation, the second machine cycle is a memory read cycle for reading the I/O device address from the memory and the third machine cycle is an I/O write cycle for sending data to the I/O device.

PROBLEMS

1. What are the various registers of 8085? Discuss their function.
2. Discuss the function of ALU of 8085.
3. What are various status flags provided in 8085? Discuss their roles.
4. Discuss the function of the following signals of 8085:
IO/M, INTR, INTA, HOLD, HLDA and READY.
5. Discuss the function of the following signals of 8085:
RD, WR, ALE, So and S1.
6. Discuss instruction cycle, machine cycle and state.
7. Discuss fetch operation and execute operation.
8. Draw and explain the timing diagram for fetch operation.
9. Draw and explain the timing diagram for memory read operation.
10. Draw and explain the timing diagram of memory write operation.
11. Draw and explain the timing diagram for the instruction MVI r, data.
12. Draw and explain the timing diagram for I/O read operation.
13. Draw and explain the timing diagram for I/O write operation.
14. How many machine cycles are required by the following instructions of Intel 8085?

Justify your answers.

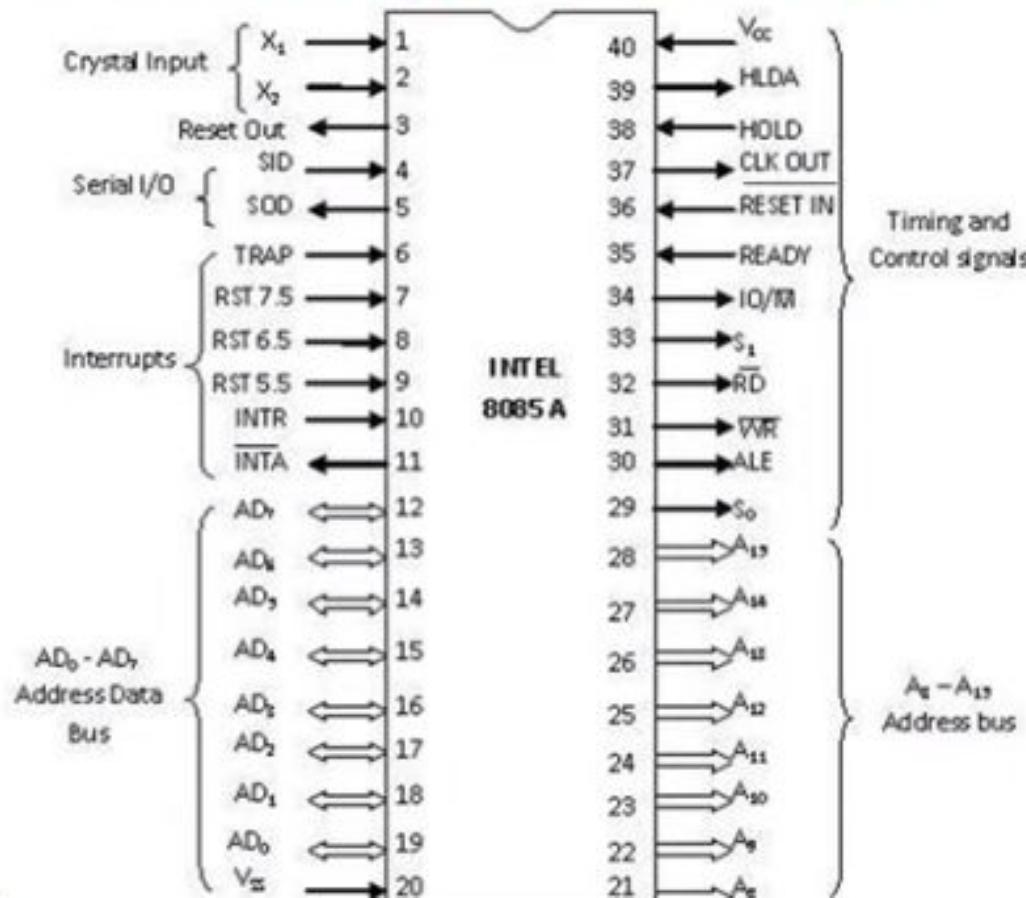
MOV r1,r2; MVI r, data; MOV r, M; LXI rp, data; LDA addr; and IN addr.

15. Explain the requirement of a program counter, stack pointer and status flags in the architecture of Intel 8085 microprocessor.

Pin diagram of 8085



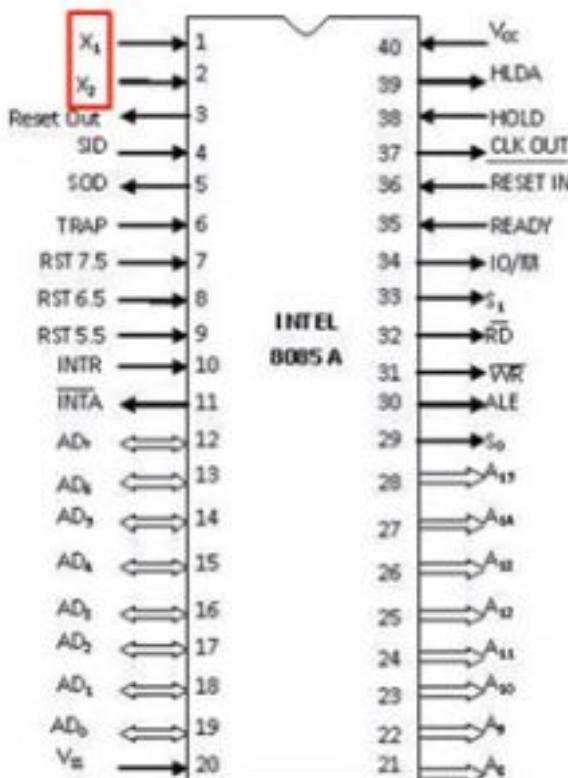
Pin Diagram of 8085



X₁ & X₂

Pin 1 and Pin 2 (Input)

- These are also called Crystal Input Pins.
- 8085 can generate clock signals internally.
- To generate clock signals internally, 8085 requires external inputs from X₁ and X₂.

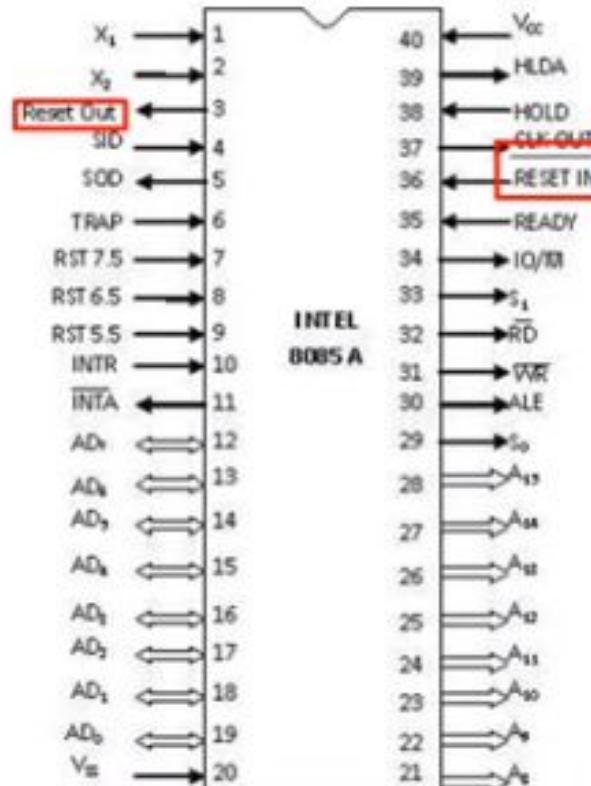


RESET IN and RESET OUT

Pin 36 (Input) and Pin 3 (Output)

- **RESET IN:**

- It is used to reset the microprocessor.
- It is active low signal.
- When the signal on this pin is low for at least 3 clocking cycles, it forces the microprocessor to reset itself.

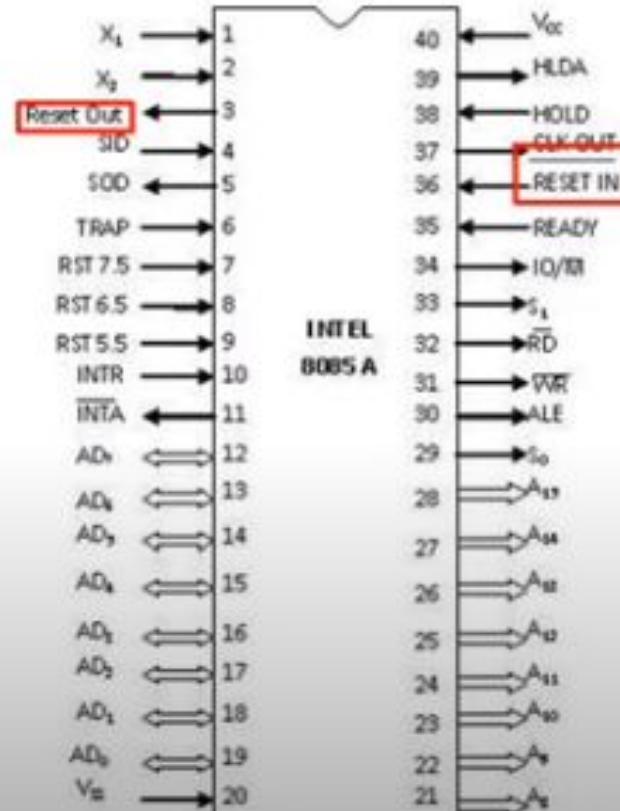


RESET IN and RESET OUT

Pin 36 (Input) and Pin 3 (Output)

- Resetting the microprocessor means:

- Clearing the PC and IR.
- Disabling all interrupts (except TRAP).
- Disabling the SOD pin.
- All the buses (data, address, control) are **tri-stated**.
- Gives HIGH output to RESET OUT pin.

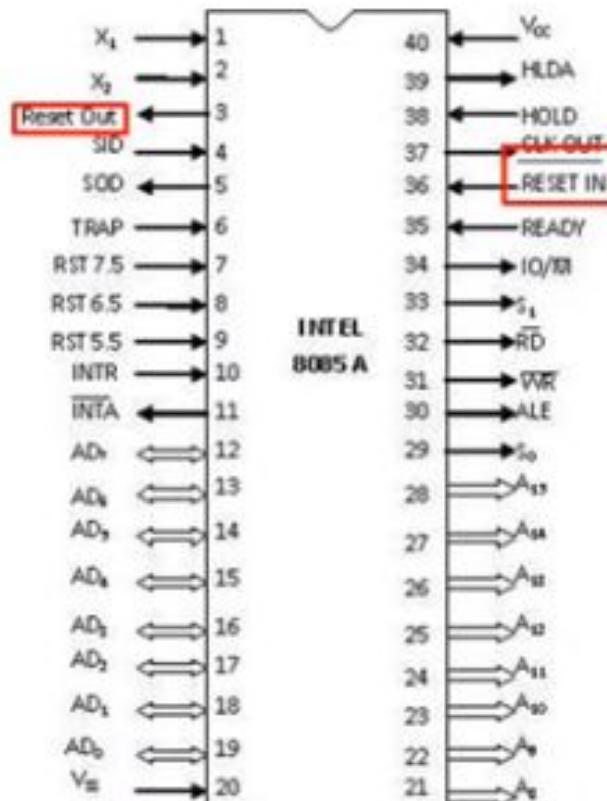


RESET IN and RESET OUT

Pin 36 (Input) and Pin 3 (Output)

- **RESET OUT:**

- It is used to reset the peripheral devices and other ICs on the circuit.
- It is an output signal.
- It is an active high signal.
- The output on this pin goes high whenever RESET IN is given low signal.
- The output remains high as long as RESET IN is kept low.

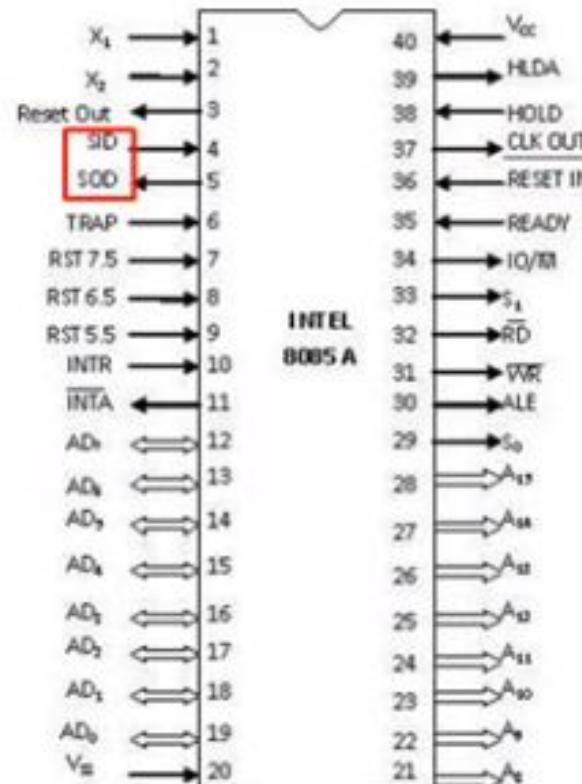


SID and SOD

Pin 4 (Input) and Pin 5 (Output)

- **SID (Serial Input Data):**

- It takes 1 bit input from serial port of 8085.
- Stores the bit at the 8th position (MSB) of the Accumulator.
- RIM (Read Interrupt Mask) instruction is used to transfer the bit.

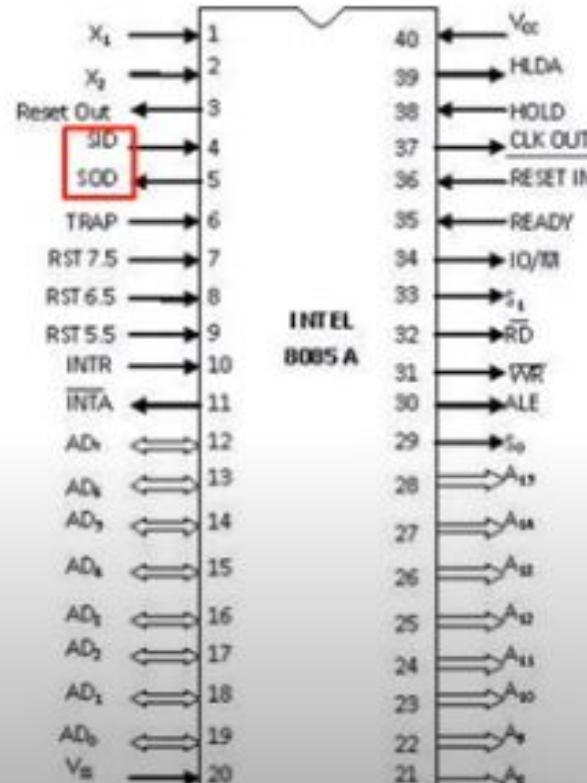


SID and SOD

Pin 4 (Input) and Pin 5 (Output)

- **SOD (Serial Output Data):**

- It takes 1 bit from Accumulator to serial port of 8085.
- Takes the bit from the 8th position (MSB) of the Accumulator.
- SIM (Set Interrupt Mask) instruction is used to transfer the bit.



Interrupt Pins

- **Interrupt:**
 - It means *interrupting* the normal execution of the microprocessor.
 - When microprocessor receives interrupt signal, it discontinues whatever it was executing.
 - It starts executing new program indicated by the interrupt signal.
 - Interrupt signals are generated by external peripheral devices.
 - After execution of the new program, microprocessor goes back to the previous program.

Sequence of Steps Whenever There is an Interrupt

- Microprocessor completes execution of current instruction of the program.
- PC contents are stored in stack.
- PC is loaded with address of the new program.
- After executing the new program, the microprocessor returns back to the previous program.
- It goes to the previous program by reading the top value of stack.

Five Hardware Interrupts in 8085

- TRAP
- RST 7.5
- RST 6.5
- RST 5.5
- INTR

Non-Maskable Interrupts

- The interrupts which are always in *enabled* mode are called non-maskable interrupts.
- These interrupts can never be disabled by any software instruction.
- TRAP is a non-maskable interrupt.

Maskable Interrupts

- List of Maskable Interrupts:
 - RST 7.5
 - RST 6.5
 - RST 5.5
 - INTR

Priority Based Interrupts

- Whenever there exists a simultaneous request at two or more pins then the pin with higher priority is selected by the microprocessor.
- Priority is considered only when there are simultaneous requests.

- Priority of interrupts:

Interrupt	Priority
TRAP	1
RST 7.5	2
RST 6.5	3
RST 5.5	4
INTR	5

Non-Vectored Interrupts

- The interrupts which don't have fixed memory location for transfer of control from normal execution.
- The address of the memory location is sent along with the interrupt.
- INTR is a non-vectored interrupt.

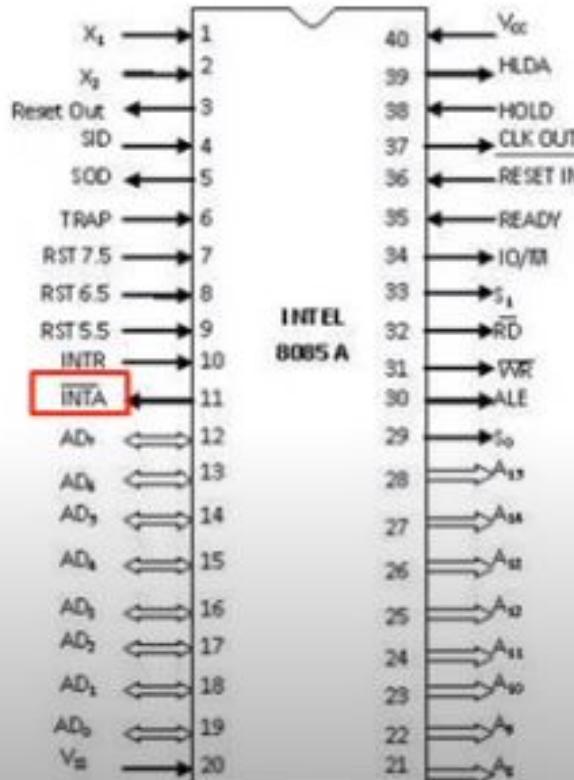
Vectored Interrupts

- List of vectored interrupts:
 - RST 7.5
 - RST 6.5
 - RST 5.5
 - TRAP

INTA

Pin 11 (Output)

- It stands for interrupt acknowledge.
- It is an outgoing signal.
- It is an active low signal.
- Low output on this pin indicates that microprocessor has acknowledged the INTR request.



Address and Data Pins

- **Address Bus:**

- The address bus is used to send address to memory.
- It selects one of the many locations in memory.
- Its size is 16-bit.

Address and Data Pins

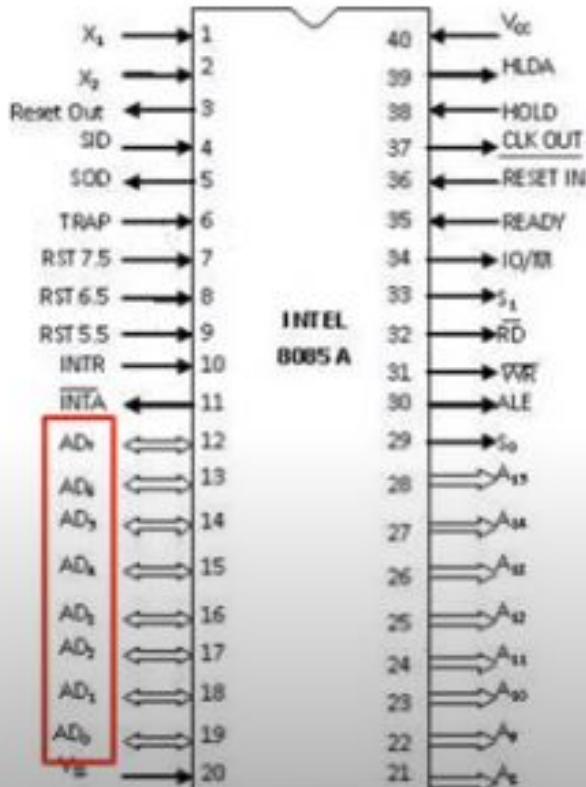
- **Data Bus:**

- It is used to transfer data between microprocessor and memory.
- Data bus is of 8-bit.

AD₀ – AD₇

Pin 19-12 (Bidirectional)

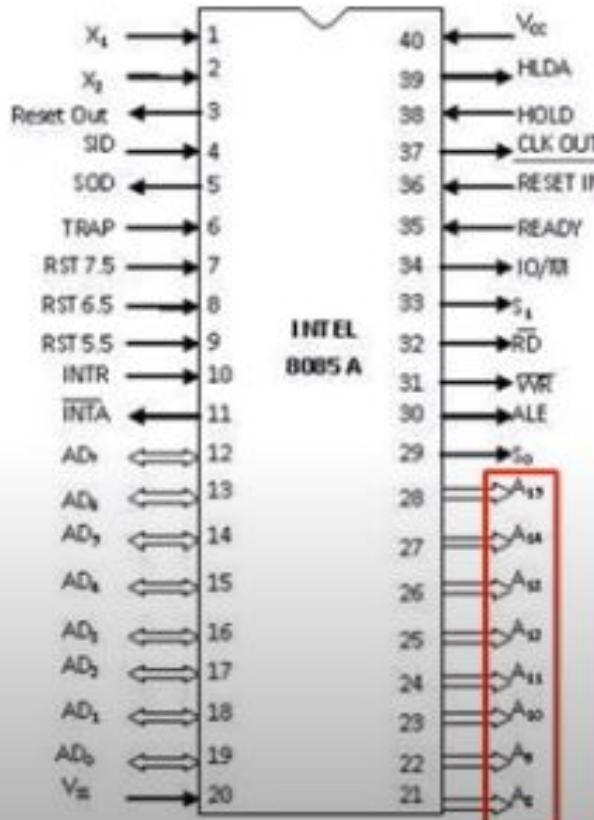
- These pins serve the dual purpose of transmitting lower order address and data byte.
- During 1st clock cycle, these pins act as lower half of address.
- In remaining clock cycles, these pins act as data bus.
- The separation of lower order address and data is done by address latch.



$A_8 - A_{15}$

Pin 21-28 (Unidirectional)

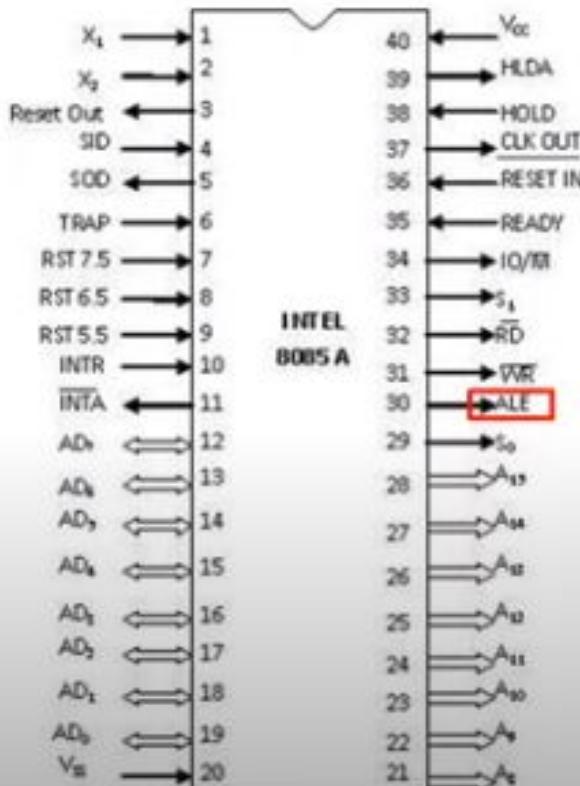
- These pins carry the higher order of address bus.
- The address is sent from microprocessor to memory.
- These 8 pins are switched to high impedance state during HOLD and RESET mode.



ALE

Pin 30 (Output)

- It is used to enable Address Latch.
- It indicates whether bus functions as address bus or data bus.
- If ALE = 1 then
 - Bus functions as address bus.
- If ALE = 0 then
 - Bus functions as data bus.



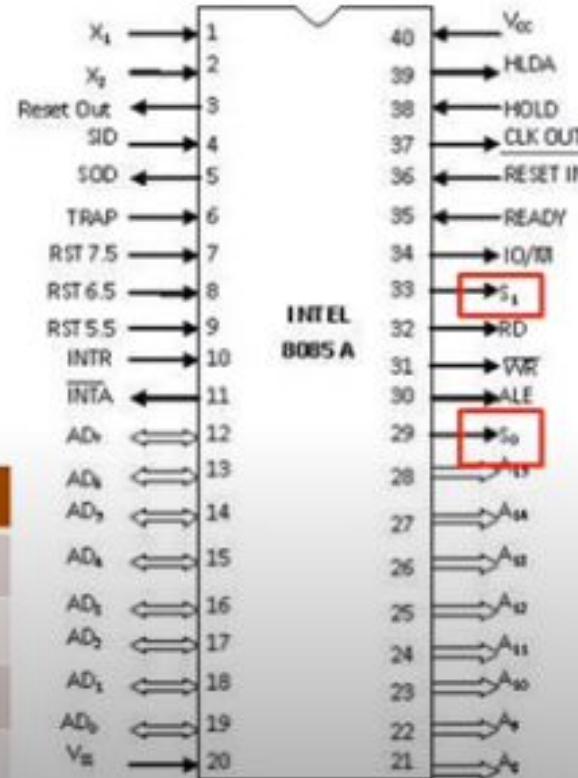
S₀ and S₁

Pin 29 (Output) and Pin 33 (Output)

- S₀ and S₁ are called Status Pins.

- They tell the current operation which is in progress in 8085.

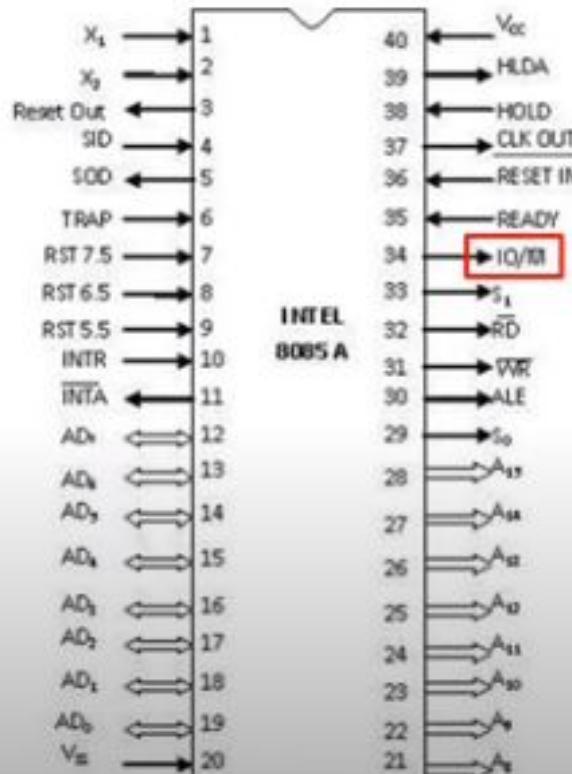
S ₀	S ₁	Operation
0	0	Halt
0	1	Write
1	0	Read
1	1	Opcode Fetch



IO/M

Pin 34 (Output)

- This pin tells whether I/O or memory operation is being performed.
- If $\text{IO/M} = 1$ then
 - I/O operation is being performed.
- If $\text{IO/M} = 0$ then
 - Memory operation is being performed.



IO/ \bar{M}

Pin 34 (Output)

- The operation being performed is indicated by S_0 and S_1 .
- If $S_0 = 0$ and $S_1 = 1$ then
 - It indicates WRITE operation.
- If $IO/\bar{M} = 0$ then
 - It indicates Memory operation.
- Combining these two we get **Memory Write Operation**.

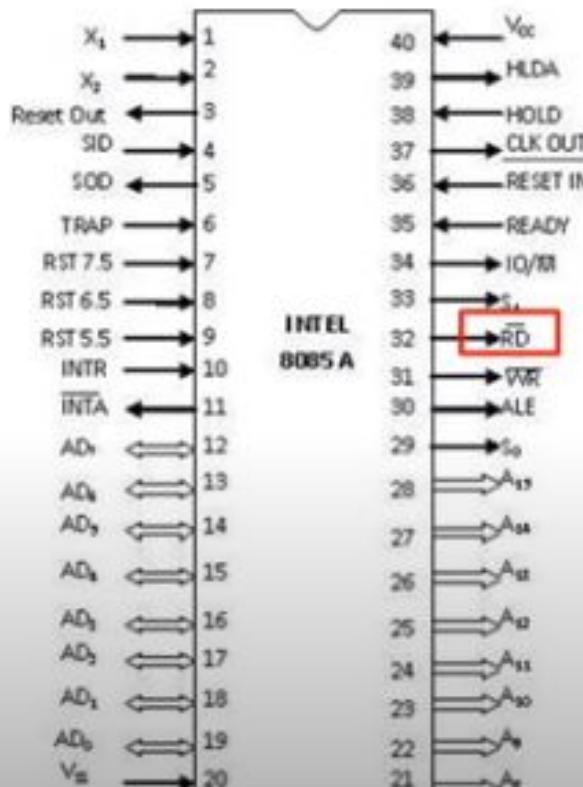
Table Showing IO/M, S₀, S₁ and Corresponding Operations

Operations	IO/M	S ₀	S ₁
Opcode Fetch	0	1	1
Memory Read	0	1	0
Memory Write	0	0	1
I/O Read	1	1	0
I/O Write	1	0	1
Interrupt Ack.	1	1	1
Halt	High Impedance	0	0

RD

Pin 32 (Output)

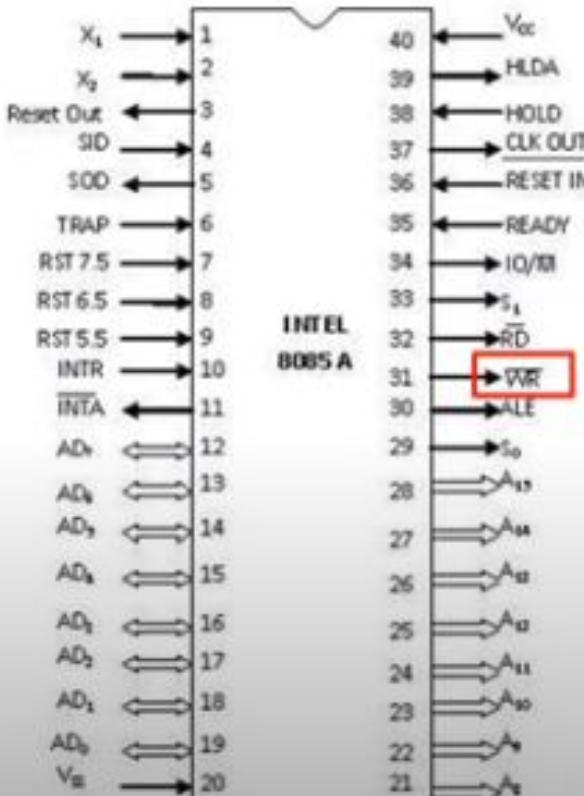
- RD stands for Read.
- It is an active low signal.
- It is a control signal used for Read operation either from memory or from Input device.
- A low signal indicates that data on the data bus must be placed either from selected memory location or from input device.



WR

Pin 31 (Output)

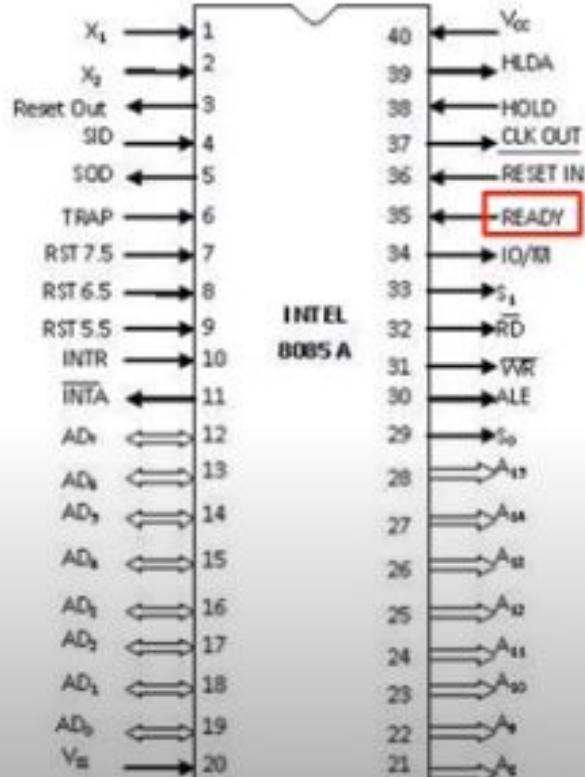
- WR stands for Write.
- It is also active low signal.
- It is a control signal used for Write operation either into memory or into output device.
- A low signal indicates that data on the data bus must be written into selected memory location or into output device.



READY

Pin 35 (Input)

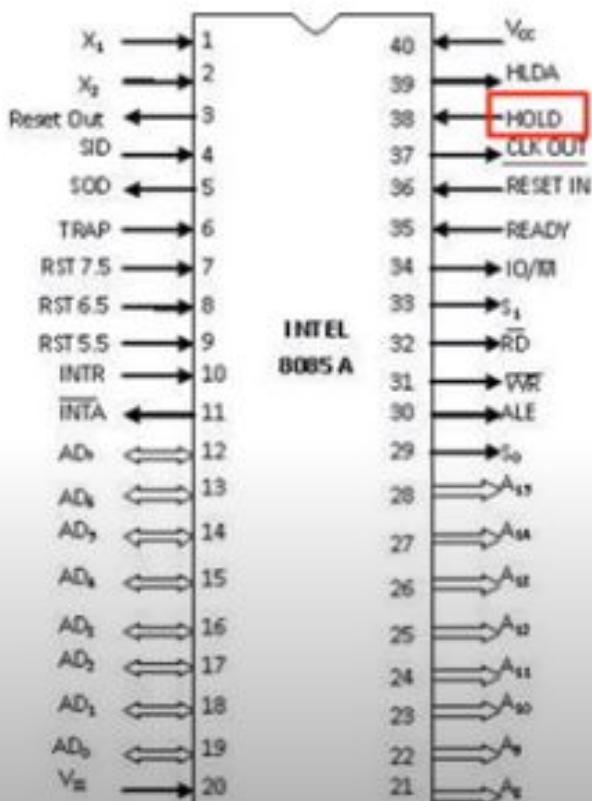
- This pin is used to synchronize slower peripheral devices with fast microprocessor.
- A low value causes the microprocessor to enter into **wait state**.
- The microprocessor remains in wait state until the input at this pin goes high.



HOLD

Pin 38 (Input)

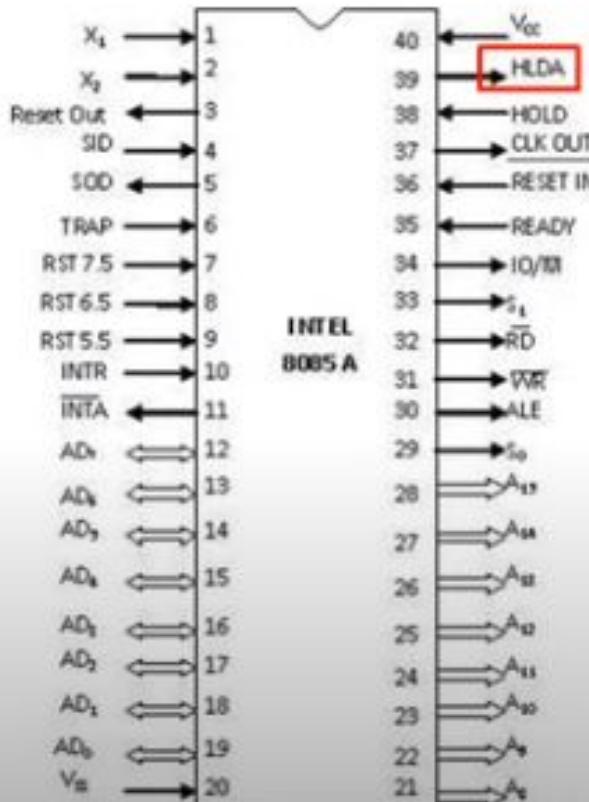
- HOLD pin is used to request the microprocessor for DMA transfer.
- A high signal on this pin is a request to microprocessor to relinquish the hold on buses.
- This request is sent by DMA controller.
- Intel 8257 and Intel 8237 are two DMA controllers.



HLDA

Pin 39 (Output)

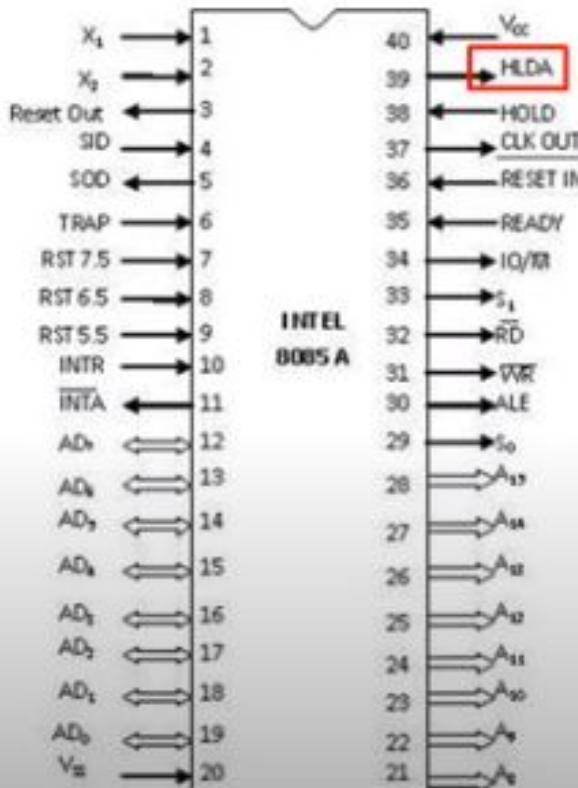
- HLDA stands for Hold Acknowledge.
- The microprocessor uses this pin to acknowledge the receipt of HOLD signal.
- When HLDA signal goes high, address bus, data bus, RD, WR, IO/M pins are **tri-stated**.
- This means they are cut-off from external environment.



HLDA

Pin 39 (Output)

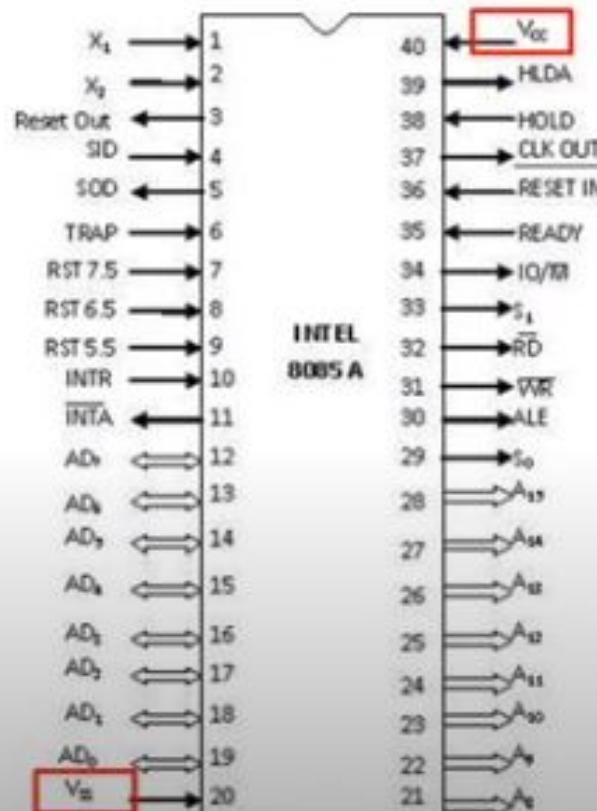
- The control of these buses goes to DMA Controller.
- Control remains at DMA Controller until HOLD is held high.
- When HOLD goes low, HLDA also goes low and the microprocessor takes control of the buses.



V_{SS} and V_{CC}

Pin 20 (Input) and Pin 40 (Input)

- +5V power supply is connected to V_{CC} .
- Ground signal is connected to V_{SS} .



Types of instructions

Opcode and Operands

Each instruction contains two parts: operation code (opcode) and operand.

The part of an instruction which specifies the task to be performed by the computer is called opcode.

The second part of the instruction is the data to be operated on, and it is called operand.

The operand may be 8-bit or 16-bit data, 8-bit or 16-bit address, internal registers or a register or memory location.

In some instructions the operand is implicit.

Instruction Word Size

According to the word size the Intel 8085 instructions are classified into the following three types:

1-byte instruction

2-byte instruction

3-byte instruction

Examples of one-byte instructions are:

MOV A, B. Move the content of register B to register A. 78H is the opcode for MOV A, B. The opcode 78H can be written in the binary form as 01111000. The 1st two bits, i.e., 01 are for MOV operation; the next three bits 111 are the binary code for register A and the last three bits 000 are the binary code for register B.

ADD B. Add the content of register B to the content of the accumulator. 80H is the opcode for the instruction ADD B. In this instruction one of the operands is register B (its content) and the other operand is in the accumulator. The opcode 80H in the binary form is 10000000. The first five bits, i.e. 10000 specify the operation to be performed, i.e. ADD operation. The last three bits 000 are the code for register B for 8085 microprocessor.

Two-Byte Instruction.

In a two-byte instruction the 1st byte of the instruction is its opcode and the 2nd byte is either data or address. Examples are:

(i) MVI B, 05; Move 05 to register B.

06, 05; MVI B, 05 in the code form.

The 1st byte 06 is the opcode for MVI B and the 2nd byte 05 is the data which is to be moved to register B.

ii) IN 01 : Read data at port B.

DB, 01; IN 01 in the code form.

DB is the opcode for the instruction IN and 01 is the address of a port

Three-Byte Instruction.

In a three byte instruction the 1st byte of the instruction is opcode and the 2nd and 3rd bytes are either 16-bit data or 16-bit address. Examples are:

(i) LXI H, 2400H : Load H-L pair with 2400H.

21, 00, 24 ;LXI H, 2400H in the code form.

The 1st byte 21 is the opcode for the instruction LXI H. The 2nd byte 00 is 8 LSBs of the data (2400H), which is loaded into register L. The 3rd byte 24 is 8 MSBs of the data (2400H), which is loaded into register H.