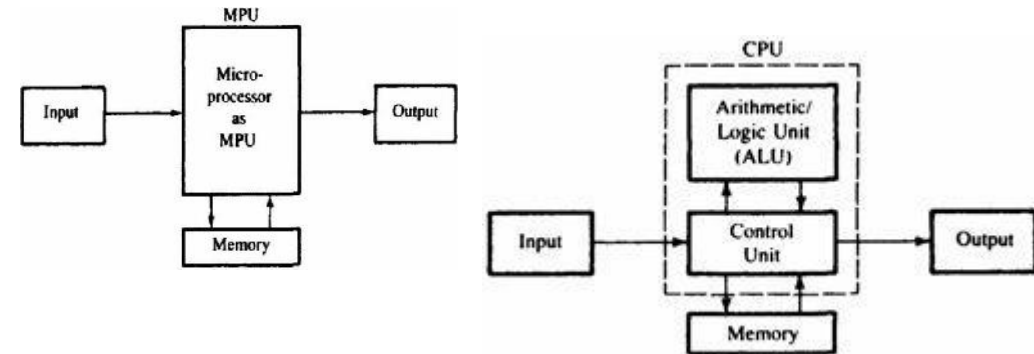


Unit – I 8085 and 8086 PROCESSOR

Introduction to microprocessor

- A microprocessor is a clock-driven semiconductor device consisting of electronic logic circuits manufactured by using either a large-scale integration (LSI) or very-large-scale integration (VLSI) technique.
- The microprocessor is capable of performing various computing functions and making decisions to change the sequence of program execution.
- In large computers, a CPU performs these computing functions. The Microprocessor resembles a CPU exactly.
- The microprocessor is in many ways similar to the CPU, but includes all the logic circuitry including the control unit, on one chip.
- The microprocessor can be divided into three segments for the sake of clarity. – They are: arithmetic/logic unit (ALU), register array, and control unit.
- A comparison between a microprocessor, and a computer is shown below:



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.

Memory: Memory stores such binary information as instructions and data, and provides that information to the microprocessor whenever necessary. 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. Read-Only memory (ROM) and Read/Write memory (R/WM), popularly known as Random- Access memory (RAM).

I/O (Input/Output): It communicates with the outside world. I/O includes two types of devices: input and output; these I/O devices are also known as peripherals.

System Bus: The system bus is a communication path between the microprocessor and peripherals: it is nothing but a group of wires to carry bits.

Arithmetic/Logic Unit: This is the area of the microprocessor where various computing functions are performed on data. The ALU unit performs such arithmetic operations as addition and subtraction, and such logic operations as AND, OR, and exclusive OR.

Register Array: This area of the microprocessor consists of various registers identified by letters such as B, C, D, E, H, and L. These registers are primarily used to store data temporarily during the execution of a program and are accessible to the user through instructions.

8085 Microprocessor

The salient features of 8085 μ p are:

- ✚ It is a 8 bit microprocessor.
- ✚ It is manufactured with N-MOS technology.
- ✚ It has 16-bit address bus and hence can address up to $2^{16} = 65536$ bytes (64KB) memory locations through $A_0 - A_{15}$.
- ✚ The first 8 lines of address bus and 8 lines of data bus are multiplexed $AD_0 - AD_7$.
- ✚ Data bus is a group of 8 lines $D_0 - D_7$.
- ✚ It supports external interrupt request.
- ✚ A 16 bit program counter (PC)
- ✚ A 16 bit stack pointer (SP)
- ✚ Six 8-bit general purpose register arranged in pairs: BC, DE, HL.
- ✚ It requires a signal +5V power supply and operates at 3.2 MHz single phase clock.
- ✚ It is enclosed with 40 pins DIP (Dual in line package).

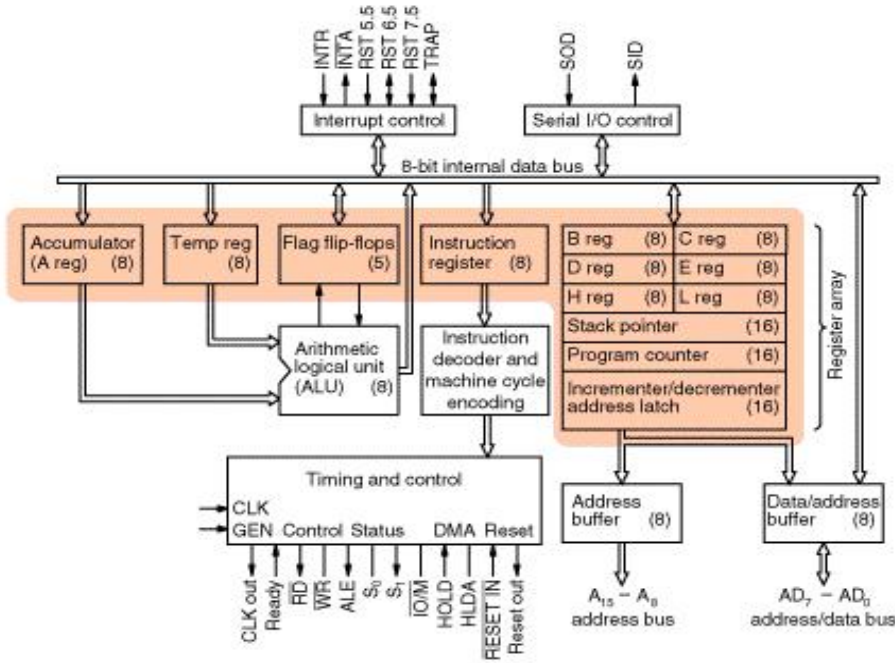


Figure 1.1 internal architecture of 8085

Arithmetic and logic unit (ALU):

The 8085A has a simple 8 bit ALU and it works in coordination with the accumulator, temporary register, five flags, and arithmetic and logic circuits. ALU has the capability of performing several mathematical and logical operations. The temporary register is used to hold the data during an arithmetic/logic operation.

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

Overview of 8085 microprocessor

Architecture of INTEL 8085

Intel 8085A is one of the most popular 8-bit microprocessors capable of addressing 64KB of memory and its architecture is simple. The architecture of 8085 includes the ALU, timing and control unit, instruction register and decoder, register array, interrupt control and serial I/O control in a package of 40 pins, requires +5V single power supply and can operate with a 3MHz single phase clock.

perform arithmetic and logical operations. The result of an operation is stored in the accumulator. The accumulator is also identified as register A.

Flag register

There are five flags in 8085, they are **sign flag (S)**, **zero flag (Z)**, **auxiliary carry flag (AC)**, **Parity flag (P)** and **carry flag (CY)** the bit position reserved for these flags in the flag register are shown below

D ₇	D ₆	D ₅	D ₄	D ₃	D ₂	D ₁	D ₀
S	Z		AC		P		CY

S – Sign Flag: After the execution of an arithmetic/logic operation, if bit D₇ of the result (usually in the accumulator) is 1, the sign flag is set. This flag is used with signed numbers. In a given byte, if D₇ is 1, the number will be viewed as a negative number; if it is 0, the number will be considered positive.

Z – Zero Flag: The Zero flag is set if the ALU operation results is 0, and the flag is reset if the result is not 0. This flag is modified by the result in the accumulator as well as in the other registers.

AC – Auxiliary Carry Flag: In an arithmetic operation, when a carry is generated by digit D₃ and passed on to digit D₄, the AC flag is set. The Flag is used only internally for the programmer to change the sequence of a program with a jump instruction.

P – Parity Flag: After an arithmetic/logic operation, if the result has an even number of 1's, the flag is set. If it has an odd number of 1's flag is reset.

CY - Carry Flag: If an arithmetic operation results in a carry. The carry flag is set; otherwise it is reset. The carry flag also serves as a borrow flag for subtraction.

TIMING AND CONTROL UNIT:

This unit synchronizes all the microprocessor operations with the clock and generates the control signals necessary for communication between the microprocessor and peripherals. The control signals \overline{RD} and \overline{WR} indicate the availability of data on the data bus.

INSTRUCTION REGISTER AND DECODER:

The Instruction register and decoder are part of the 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.

REGISTER ARRAY:

The 8085 have six general purpose registers to store eight bit data during program execution. These registers are identified as B, C, D, E, H, and L. They can be combined as register pairs BC, DE, and HL to perform 16 bit operation. The programmer can use these registers to store or copy data into the registers by using data copy instructions.

In addition to 6 general purpose register, it has two 16 bit register called stack pointer and program counter

Program Counter (PC)

This 16-bit register deals with sequencing the execution of instructions. This register is a memory pointer. Memory locations have 16-bit addresses, and that is why this is a 16-bit register. 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. When a byte (machine code) is being fetched, the program counter is incremented by one to point to the next memory location

Stack Pointer (SP)

The stack pointer is also a 16-bit register used as a memory pointer. It points to a memory location in R/W memory, called the stack. The beginning of the stack is defined by loading 16-bit address in the stack pointer.

COMMUNICATION LINES:

8085 MPU performs data transfer operation using three sets of communication lines called buses: the address bus, the data bus, and the control bus.

ADDRESS BUS:

The address bus is a group of 16 lines generally identified as A_0 to A_{15} . The address bus is unidirectional i.e. bits flow in one direction from MPU to peripheral devices. The 8085 MPU with its 16 address lines is capable of addressing $2^{16} = 65,536(64K)$ bytes memory location.

DATA BUS:

The data bus is a group of 8 lines used for data flow. These lines are bi-directional i.e. data flow in both direction between MPU and peripheral devices. The 8 data lines enable the MPU to manipulate 8bit data ranging from 00 to FF ($2^8 = 256$ numbers). The largest number that can appear on the data bus is 11111111(255_{10}).

CONTROL BUS:

The control bus comprised of various single lines that carry synchronization signals. The MPU uses such lines to provide timing signals.

Memory

Intel 8085 has three types of memory, they are Programs, data and stack memories

Program memory : Program can be a located anywhere in memory. Jump, branch and call instructions use 16-bit addresses, i.e. they can be used to jump/branch anywhere within 64 KB. All jump/branch instructions use absolute addressing.

Data memory - the processor always uses 16-bit addresses so that data can be placed anywhere

Stack memory is limited only by the size of memory. Stack grows downward.

First 64 bytes in a zero memory page should be reserved for vectors used by RST instructions.

INTERRUPTS

The processor has 5 interrupts. They are presented below in the order of their **INTR:** It has the lowest priority and is a maskable interrupt. This is also called as hand shake interrupt

RST5.5 is a maskable interrupt. When this interrupt is received the processor saves the contents of the PC register into stack and branches to 2CH (hexadecimal) address.

RST6.5 is a maskable interrupt When this interrupt is received the processor saves the contents of the PC register into stack and branches to 34H (hexadecimal) address.

RST7.5 is a maskable in saves the contents of the PC register into stack and branches to 3CH (hexadecimal) address.

TRAP is a non-maskable interrupt when this interrupt is received the processor saves the contents of the PC register into stack and branches to 24H (hexadecimal) address.

All maskable interrupts can be enabled or disabled using EI and DI instructions RST 5.5, RST6.5 and RST7.5 interrupts can be enabled or disabled individually using SIM instruction.

Serial communication Signal

SID - Serial Input Data Line: The data on this line is loaded into accumulator bit 7 whenever a RIM instruction is executed

SOD – Serial Output Data Line: The SIM instruction loads the value of bit 7 of the accumulator into SOD latch if bit 6 (SOE) of the accumulator is 1.

8085 Pin description.

Properties

- Single + 5V Supply
- 4 Vectored Interrupts (One is Non Maskable)
- Serial In/Serial Out Port
- Decimal, Binary, and Double Precision Arithmetic
- Direct Addressing Capability to 64K bytes of memory

The Intel 8085A is a new generation, complete 8 bit parallel central processing unit (CPU). The 8085A uses a multiplexed data bus. The address is split between the 8bit address bus and the 8bit data bus.

Pin Description

The following describes the function of each pin:

A8 – A15 (Output 3 State)

Address Bus; The most significant 8 bits of the memory address or the 8 bits of the I/O address,3 stated during Hold and Halt modes.

AD0 - AD7 (Input/Output 3state)

Multiplexed Address/Data Bus; Lower 8 bits of the memory address (or I/O address) appear on the bus during the first clock cycle of a machine state. It then becomes the data bus during the second and third clock cycles. 3 stated during Hold and Halt modes.

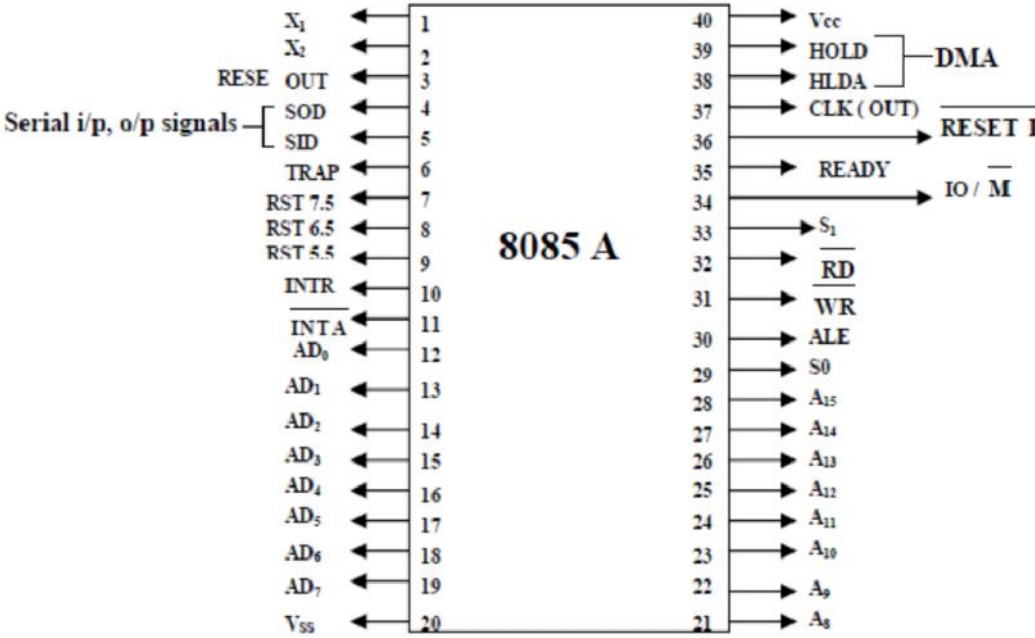


Figure 1.2 a pin diagram of 8085

ALE (Output)

Address Latch Enable: It occurs during the first clock cycle of a machine state and enables the address to get latched into the on chip latch of peripherals. The falling edge of ALE is set to guarantee setup and hold times for the address information. ALE can also be used to strobe the status information. ALE is never 3stated.

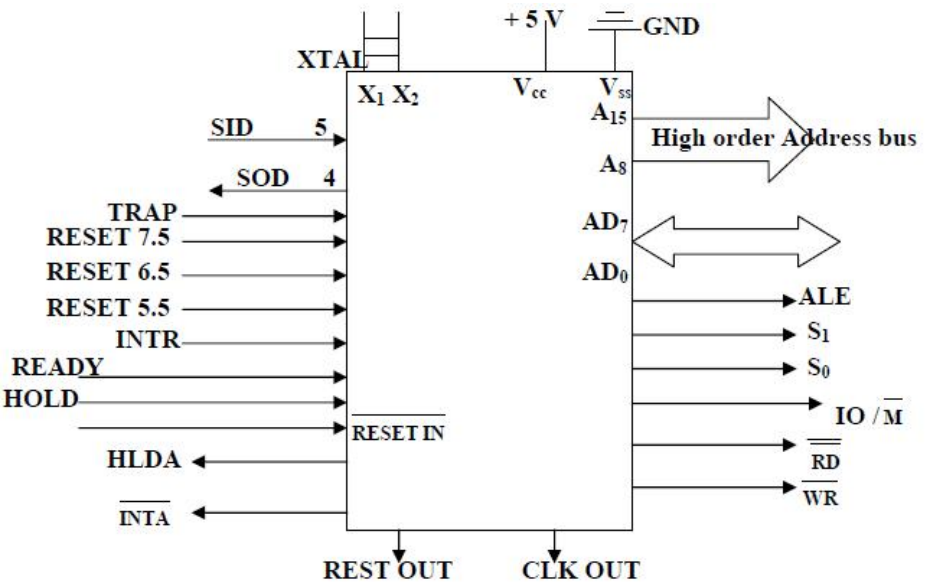


Figure 1.2 b signal diagram of 8085

SO, S1 (Output)

Data Bus Status. Encoded status of the bus cycle

S1	S0	
0	0	Halt
0	1	Write
1	0	Read
1	1	fetch

RD (Output 3state)

READ; indicates the selected memory or I/O device is to be read and that the Data Bus is available for the data transfer.

WR (Output 3state)

WRITE; indicates the data on the Data Bus is to be written into the selected memory or I/O location.

READY (Input)

If Ready is high during a read or write cycle, it indicates that the memory or peripheral is ready to send or receive data. If Ready is low, the CPU will wait for Ready to go high before completing the read or write cycle.

HOLD (Input)

HOLD; indicates that another Master is requesting the use of the Address and Data Buses. The CPU, upon receiving the Hold request. Will relinquish the use of buses as soon as the completion of the current machine cycle. Internal processing can continue. The processor can regain the buses only after the Hold is removed. When the Hold is acknowledged, the Address, Data, RD, WR, and IO/M lines are 3stated

HLDA (Output)

HOLD ACKNOWLEDGE; indicates that the CPU has received the Hold request and that it will relinquish the buses in the next clock cycle. HLDA goes low after the Hold request is removed. The CPU takes the buses one half clock cycle after HLDA goes low.

INTR (Input)

INTERRUPT REQUEST; is used as a general purpose interrupt. It is sampled only during the next to the last clock cycle of the instruction. If it is

active, the Program Counter (PC) will be inhibited from incrementing and an INTA will be issued. During this cycle a RESTART or CALL instruction can be inserted to jump to the interrupt service routine. The INTR is enabled and disabled by software. It is disabled by Reset and immediately after an interrupt is accepted.

INTA (Output)

INTERRUPT ACKNOWLEDGE; is used instead of (and has the same timing as) RD during the Instruction cycle after an INTR is accepted. It can be used to activate the 8259 Interrupt chip or some other interrupt port.

- RST 5.5
- RST 6.5 - (Inputs)
- RST 7.5

RESTART INTERRUPTS; These three inputs have the same timing as INTR except they cause an internal RESTART to be automatically inserted.

- RST 7.5 ----- Highest Priority
- RST 6.5
- RST 5.5 -----Lowest Priority

The priority of these interrupts is ordered as shown above. These interrupts have a higher priority than the INTR.

TRAP (Input)

Trap interrupt is a non maskable restart interrupt. It is recognized at the same time as INTR. It is unaffected by any mask or Interrupt Enable. It has the highest priority of any interrupt.

RESET IN (Input) Reset sets the Program Counter to zero and resets the Interrupt Enable and HLDA flip-flops. None of the other flags or registers (except the instruction register) are affected The CPU is held in the reset condition as long as Reset is applied.

RESET OUT (Output)

Indicates CPU is being reset. Can be used as a system RESET. The signal is synchronized to the processor clock.

X1, X2 (Input)

Crystal or R/C network connections to set the internal clock generator X1 can also be an external clock input instead of a crystal. The input frequency is divided by 2 to give the internal operating frequency.

CLK (Output)

Clock Output for use as a system clock when a crystal or R/ C network is used as an input to the CPU. The period of CLK is twice the X1, X2 input period.

IO/M (Output)

IO/M indicates whether the Read/Write is to memory or I/O Tristated during Hold and Halt modes.

SID (Input)

Serial input data line The data on this line is loaded into accumulator bit 7 whenever a RIM instruction is executed.

SOD (output)

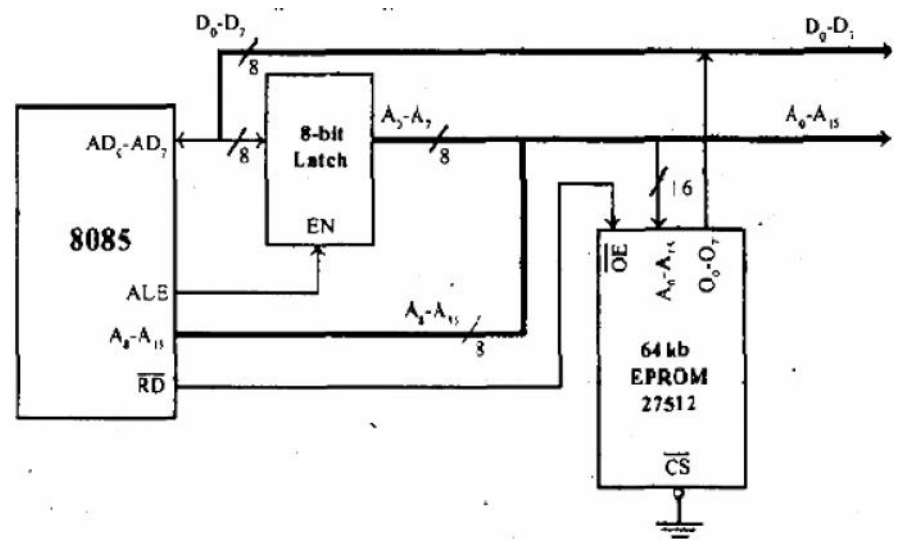
Serial output data line. The output SOD is set or reset as specified by the SIM instruction.

- Vcc
- +5 volt supply.
- Vss
- Ground Reference.

I/O and Memory interfacing examples
EXAMPLE-1

Consider a system in which the full memory space 64kb is utilized for EPROM memory. Interface the EPROM with 8085 processor.

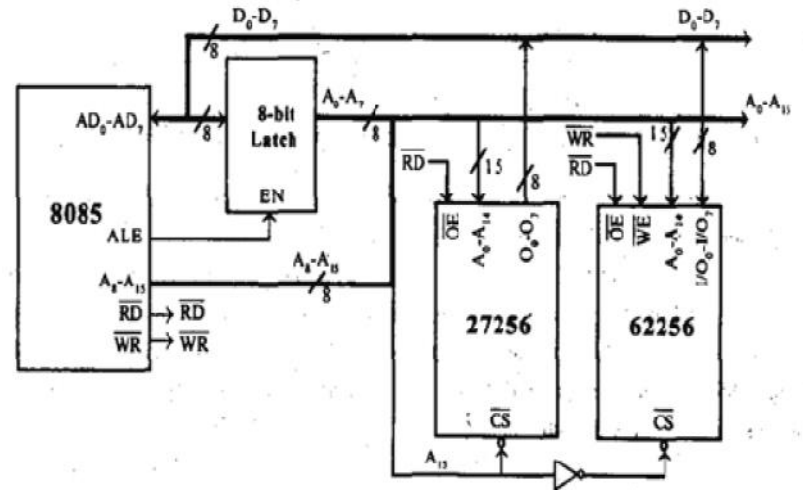
- The memory capacity is 64 Kbytes. i.e. $2^n = 64 \times 1000$ bytes where $n =$ address lines. So, $n = 16$.
- In this system the entire 16 address lines of the processor are connected to address input pins of memory IC in order to address the internal locations of memory.
- The chip select (CS) pin of EPROM is permanently tied to logic low (i.e., tied to ground).
- Since the processor is connected to EPROM, the active low RD pin is connected to active low output enable pin of EPROM.
- The range of address for EPROM is 0000H to FFFFH.



EXAMPLE-2

Consider a system in which the available 64kb memory space is equally divided between EPROM and RAM. Interface the EPROM and RAM with 8085 processor.

- Implement 32kb memory capacity of EPROM using single IC 27256.
- 32kb RAM capacity is implemented using single IC 62256.
- The 32kb memory requires 15 address lines and so the address lines A0 - A14 of the processor are connected to 15 address pins of both EPROM and RAM.
- The unused address line A15 is used as to chip select. If A15 is 1, it select RAM and If A15 is 0, it select EPROM.
- Inverter is used for selecting the memory.
- The memory used is both Ram and EPROM, so the low RD and WR pins of processor are connected to low WE and OE pins of memory respectively.
- The address range of EPROM will be 0000H to 7FFFH and that of RAM will be 8000H to FFFFH.



EXAMPLE-3

Consider a system in which 32kb memory space is implemented using four numbers of 8kb memory. Interface the EPROM and RAM with 8085 processor.

- The total memory capacity is 32Kb. So, let two number of 8kb n memory be EPROM and the remaining two numbers be RAM.
- Each 8kb memory requires 13 address lines and so the address lines A0- A12 of the processor are connected to 13 address pins of all the memory.

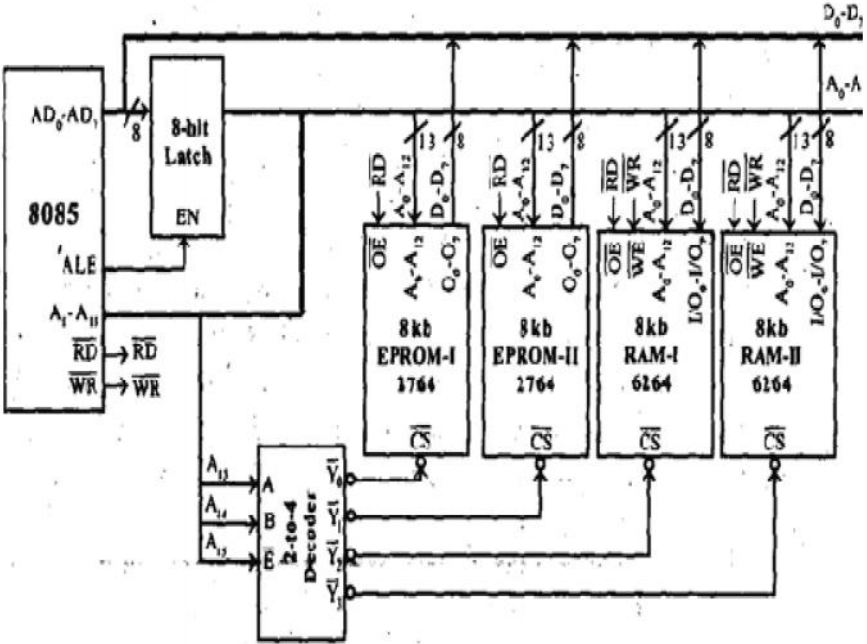


Fig - Interfacing 16Kb EPROM and 16Kb RAM with 8085

- The address lines and A13 - A14 can be decoded using a 2-to-4 decoder to generate four chip select signals.

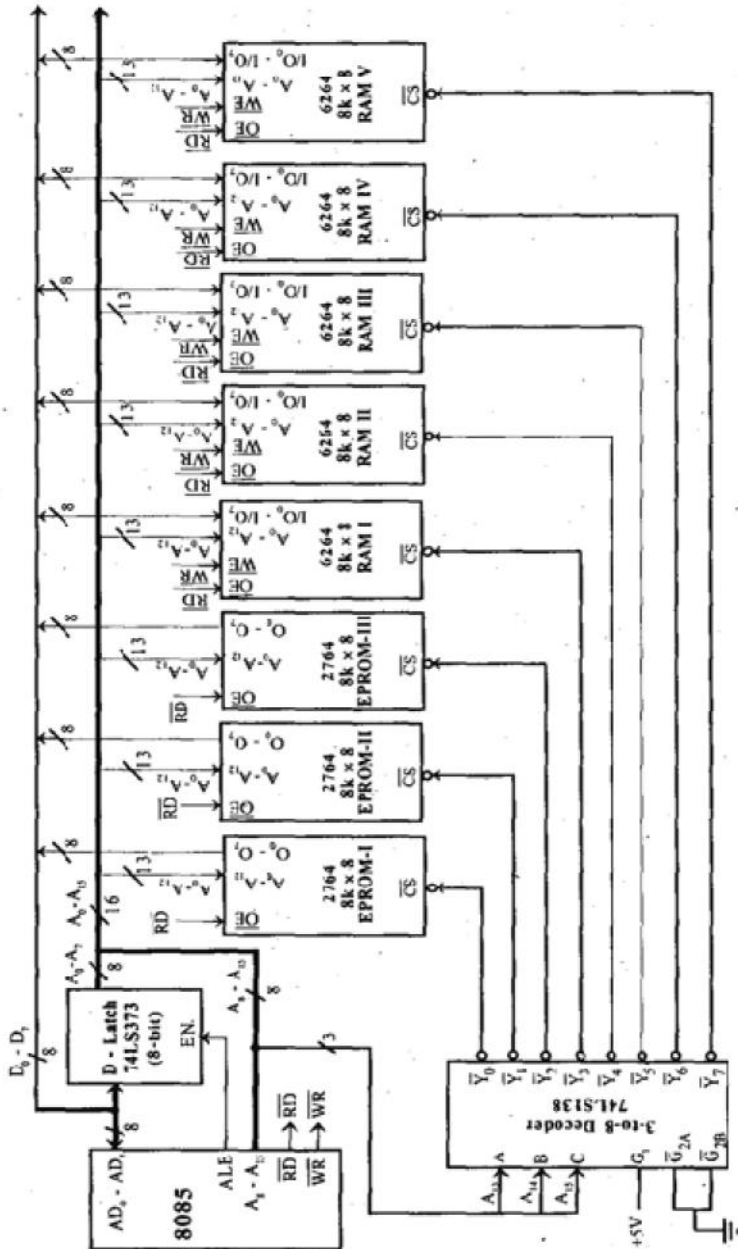
- These four chip select signals can be used to select one of the four memory IC at any one time.
- The address line A15 is used as enable for decoder.
- The simplified schematic memory organization is shown.
- The address allotted to each memory IC is shown in following table.

Device	Binary address													Hexa address			
	Decoder enable/input	Input to address pins of memory IC															
	A ₁₅ A ₁₄ A ₁₃	A ₁₂	A ₁₁ A ₁₀ A ₉ A ₈	A ₇ A ₆ A ₅ A ₄	A ₃ A ₂ A ₁ A ₀												
8kb EPROM - I	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0001
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0002
	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1FFF
8kb EPROM - II	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	2000
	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	2001
	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	2002
	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3FFF
8kb RAM - I	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4000
	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	4001
	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	4002
	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	5FFF
8kb RAM - II	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	6000
	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	6001
	0	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	6002
	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	7FFF

EXAMPLE-4

Consider a system in which the 64kb memory space is implemented using eight numbers of 8kb memory. Interface the EPROM and RAM with 8085 processor

- The total memory capacity is 64Kb. So, let 3 numbers of 8Kb EPROM and 5 numbers of 8Kb RAM
- Each 8kb memory requires 13 address lines. So the address line A0 - A12 of the processor are connected to 13 address pins of all the memory ICs.



Memory IC chip	Binary address													Hexa address			
	Decoder input			input to memory address pins													
	A ₁₅	A ₁₄	A ₁₃	A ₁₂	A ₁₁	A ₁₀	A ₉	A ₈	A ₇	A ₆	A ₅	A ₄	A ₃		A ₂	A ₁	A ₀
EPROM I	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0001
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0002

	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1FFF
EPROM II	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	2000
	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	2001
	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	2002

	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3FFF
EPROM III	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4000
	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	4001
	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	4002

	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	5FFF
RAM I	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	6000
	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	6001
	0	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	6002

	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	7FFF
RAM II	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8000
	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	8001
	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	8002

	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	9FFF
RAM III	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	A000
	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	A001
	1	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	A002

	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	BFFF
RAM IV	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	C000
	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	C001
	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	C002

	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	DFFF
RAM V	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	E000
	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	E001
	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	E002

	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	FFFF

Fig - Interfacing 4 no. 8Kb EPROM and 4 no. 8Kb RAM with 8085 The address lines A13, A14 and A15 are decoded using a 3-to-8 coder to generate eight chip select signals. These eight chip select signals can be used to select one of the eight memories at any one time.

- The memory interfacing is shown in following figure.

The address allocation for Interfacing 4 no. 8Kb EPROM and 4 no. 8Kb RAM with 8085 is,

I/O INTERFACING WITH 8085

Example 1:

A system requires 16kb EPROM and 16kb RAM. Also the system has 2 numbers of 8255, one number of 8279, one number of 8251 and one number of 8254. (8255 - Programmable peripheral interface; 8279-Keybaord/display controller, 8251 - USART and 8254 - Timer). Draw the Interface diagram. Allocate addresses to all the devices. The peripheral IC should be I/O mapped.

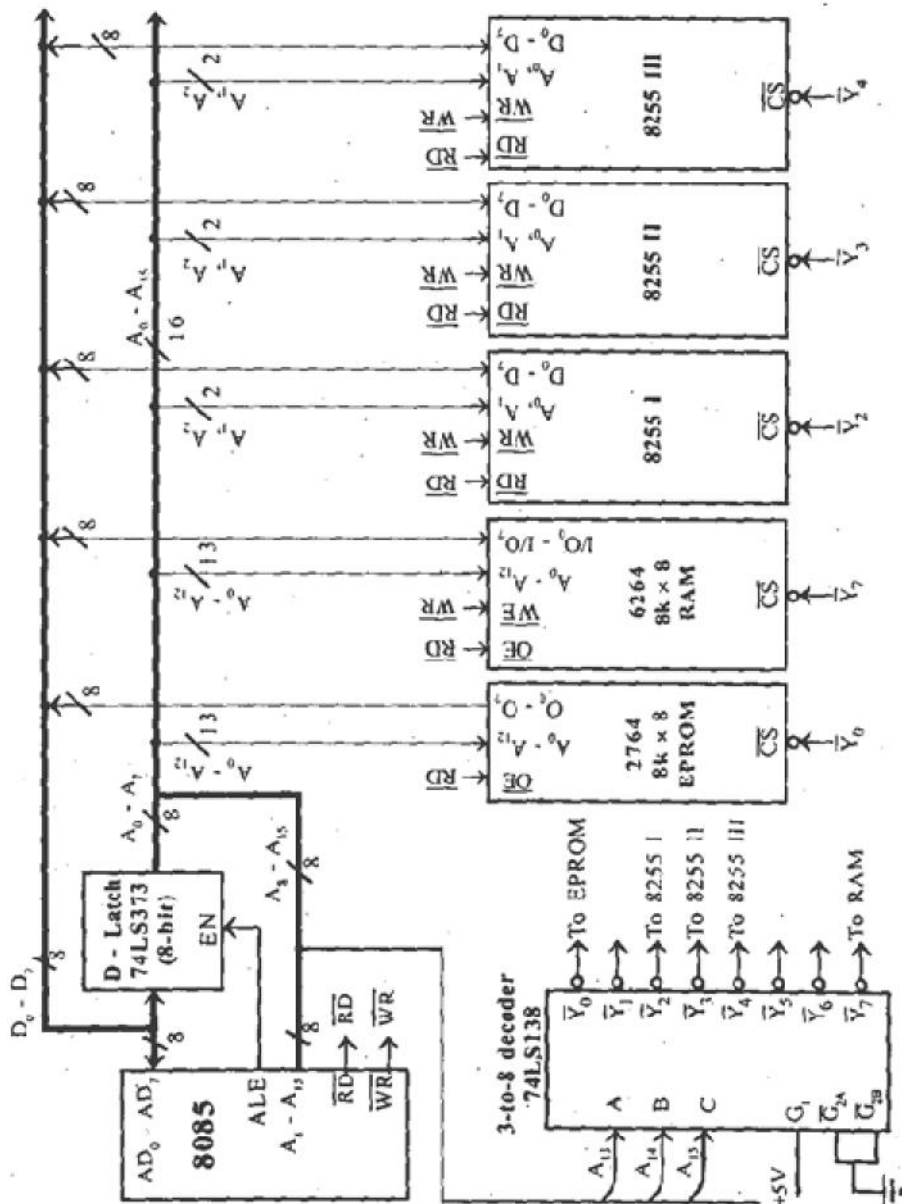
- The I/O devices in the system should be mapped by standard I/O mapping. Hence separate decoders can be used to generate chip select signals for memory IC and peripheral IC's.
- For 16kb EPROM, we can provide 2 numbers of 2764(8k x 8) EPROM.
- For 16kb RAM we can provide 2 numbers of 6264 (8k x 8) RAM.
- The 8kb memories require 13 address lines. Hence the address lines A0 - A12 are used for selecting the memory locations.
- The unused address lines A13, A14 and A15 are used as input to decoder 74LS138 (3-to-8-decoder) of memory IC. The logic low enables of this decoder are tied to IO/ M(low) of 8085, so that this decoder is enabled for memory read/write operation. The other enable pins of decoder are tied to appropriate logic levels permanently. The

4-outputs of the decoder are used to select memory ICs and the remaining 4 are kept for future expansion.

- The EPROM is mapped in the beginning of memory space from 0000H to 3FFF.
- The RAM is mapped at the end of memory space from C000 to FFFFH.

Device	Binary address												Hexa address					
	Decoder input			Input to address pins of memory/8255														
	A ₁₅	A ₁₄	A ₁₃	A ₁₂	A ₁₁	A ₁₀	A ₉	A ₈	A ₇	A ₆	A ₅	A ₄		A ₃	A ₂	A ₁	A ₀	
2764 EPROM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000	
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0001	
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0002	
	
	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1FFF	
6264 RAM	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	E000	
	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	E001	
	1	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	E002	
	
	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	FFFF	
8255 I PORT-A PORT-B PORT-C Control Register	0	1	0	X	X	X	X	X	X	X	X	X	X	X	0	0	X	4000
	0	1	0	X	X	X	X	X	X	X	X	X	X	X	0	1	X	4002
	0	1	0	X	X	X	X	X	X	X	X	X	X	X	1	0	X	4004
	0	1	0	X	X	X	X	X	X	X	X	X	X	X	1	1	X	4006
8255 III PORT-A PORT-B PORT-C Control Register	0	1	1	X	X	X	X	X	X	X	X	X	X	X	0	0	X	6000
	0	1	1	X	X	X	X	X	X	X	X	X	X	X	0	1	X	6002
	0	1	1	X	X	X	X	X	X	X	X	X	X	X	1	0	X	6004
	0	1	1	X	X	X	X	X	X	X	X	X	X	X	1	1	X	6006
8255 IIII PORT-A PORT-B PORT-C Control Register	1	0	0	X	X	X	X	X	X	X	X	X	X	X	0	0	X	8000
	1	0	0	X	X	X	X	X	X	X	X	X	X	X	0	1	X	8002
	1	0	0	X	X	X	X	X	X	X	X	X	X	X	1	0	X	8004
	1	0	0	X	X	X	X	X	X	X	X	X	X	X	1	1	X	8006

Note : The "X" indicates that the address line is not used for the particular device and they are considered as zero.



There are five peripheral IC's to be interfaced to the system.

- The chip-select signals for these IC's are given through another 3-to-8 decoder 74LS138 (I/O decoder). The input to this decoder is A11, A12 and A13

- The address lines A13, A14 and A15 are logically ORed and applied to low enable of I/O decoder.
- The logic high enable of I/O decoder is tied to IO / M(low) signal of 8085, so that this decoder is enabled for I/O read/write operation.

Fig - Internal address of 8255

Fig - Memory and I/O Port Interfacing with 8085\

Memory Mapping of I/O device	I/O Mapping of I/O device
1. 16-bit addresses are provided for I/O devices.	1. 8-bit addresses are provided for I/O devices.
2. The devices are accessed by memory read or memory write cycles.	2. The devices are accessed by I/O read or I/O write cycle. During these cycles the 8-bit address is available on both low order address lines and high order address lines.
3. The I/O ports or peripherals can be treated like memory locations and so all instructions related to memory can be used for data transfer between I/O device and the processor.	3. Only IN and OUT instructions can be used for data transfer between I/O device and the processor.
4. In memory mapped ports the data can be moved from any register to ports and vice-versa.	4. In I/O mapped ports the data transfer can take place only between the accumulator and ports.
5. When memory mapping is used for I/O devices, the full memory address space cannot be used for addressing memory. Hence memory mapping is useful only for small systems, where the memory requirement is less.	5. When I/O mapping is used for I/O devices then the full memory address space can be used for addressing memory. Hence it is suitable for systems which requires large memory capacity.
6. In memory mapped I/O devices, a large number of I/O ports can be interfaced.	6. In I/O mapping only 256 ports ($2^8 = 256$) can be interfaced.
7. For accessing the memory mapped devices, the processor executes memory read or write cycle. During this cycle IO/\bar{M} is asserted low ($IO/\bar{M} = 0$).	7. For accessing the I/O mapped devices, the processor executes I/O read or write cycle. During this cycle IO/\bar{M} is asserted high ($IO/\bar{M} = 1$).

INTERRUPT STRUCTURE

- Interrupt is signals send by an external device to the processor, to request the processor to perform a particular task or work.
- Mainly in the microprocessor based system the interrupts are used for data transfer between the peripheral and the microprocessor.
- The processor will check the interrupts always at the 2nd T-state of last machine cycle.
- If there is any interrupt it accept the interrupt and 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.
- It returned to main program by RET instruction.

Types of Interrupts:

It supports two types of interrupts.

1. Hardware interrupts
2. Software interrupts

Software interrupts:

- The software interrupts are program instructions. These instructions are inserted at desired locations in a program.

- The 8085 has eight software interrupts from RST 0 to RST 7. The vector address for these interrupts can be calculated as follows.

The Table shows the vector addresses of all interrupts.

Interrupt	Vector address
RST 0	0000 _H
RST 1	0008 _H
RST 2	0010 _H
RST 3	0018 _H
RST 4	0020 _H
RST 5	0028 _H
RST 6	0030 _H
RST 7	0038 _H

Hardware interrupts:

- An external device initiates the hardware interrupts and placing an appropriate signal at the interrupt pin of the processor.
- If the interrupt is accepted then the processor executes an interrupt service routine.

The 8085 has five hardware interrupts

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

Interrupt	Vector address
RST 7.5	003C _H
RST 6.5	0034 _H
RST 5.5	002C _H
TRAP	0024 _H

TRAP:

- This interrupt is a non-maskable interrupt. It is unaffected by any mask or interrupt enable.
- TRAP has the highest priority and vectored interrupt.
- TRAP interrupt is edge and level triggered. This means that the TRAP must go high and remain high until it is acknowledged.
- In sudden power failure, it executes an ISR and sends the data from main memory to backup memory.
- The signal, which overrides the TRAP, is HOLD signal. (i.e., If the processor receives HOLD and TRAP at the same time then HOLD is recognized first and then TRAP is recognized).

There are two ways to clear TRAP interrupt.

1. By resetting microprocessor (External signal)
2. By giving a high TRAP ACKNOWLEDGE (Internal signal)

RST 7.5:

- The RST 7.5 interrupt is a maskable interrupt.
- It has the second highest priority.
- It is edge sensitive. i.e. Input goes to high and no need to maintain high state until it is recognized.
- Maskable interrupt. It is disabled by,
 1. DI instruction
 2. System or processor reset.
 3. After reorganization of interrupt.
- Enabled by EI instruction.

RST 6.5 and 5.5:

- The RST 6.5 and RST 5.5 both are level triggered. i.e. Input goes to high and stays high until it is recognized.
- Maskable interrupt. It is disabled by,
 1. DI, SIM instruction
 2. System or processor reset.
 3. After reorganization of interrupt.
- Enabled by EI instruction.
- The RST 6.5 has the third priority whereas RST 5.5 has the fourth priority.

INTR:

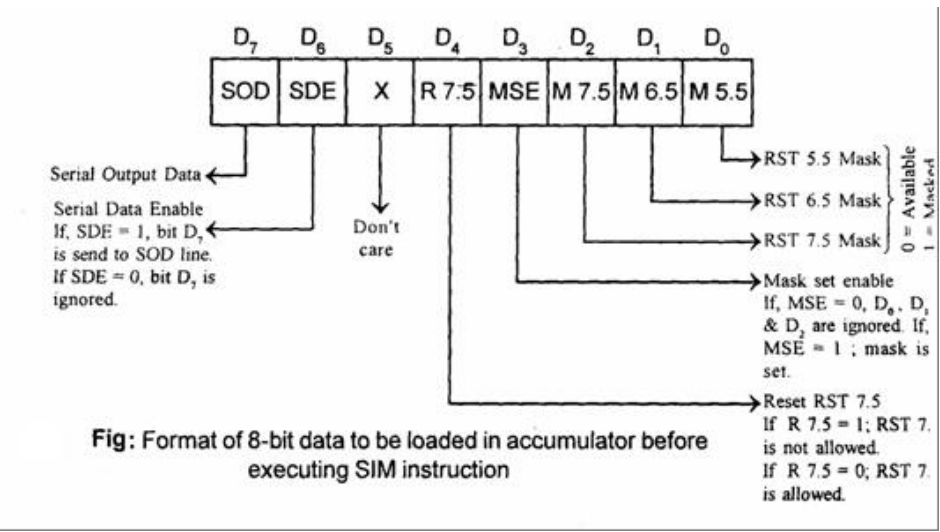
- INTR is a maskable interrupt. It is disabled by,
 1. DI, SIM instruction
 2. System or processor reset.
 3. After reorganization of interrupt.
- Enabled by EI instruction.
- Non-vectored interrupt. After receiving INTA (active low) signal, it has to supply the address of ISR.
- It has the lowest priority.
- It is a level sensitive interrupt. i.e. Input goes to high and it is necessary to maintain high state until it is recognized.
- 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. In response to the acknowledge signal, external logic places an instruction OPCODE on the data bus. In the case of multibyte instruction, additional interrupt acknowledge machine cycles are generated by the 8085 to transfer the additional bytes into the microprocessor.
4. On receiving the instruction, the 8085 save the address of next instruction on stack and execute received instruction.

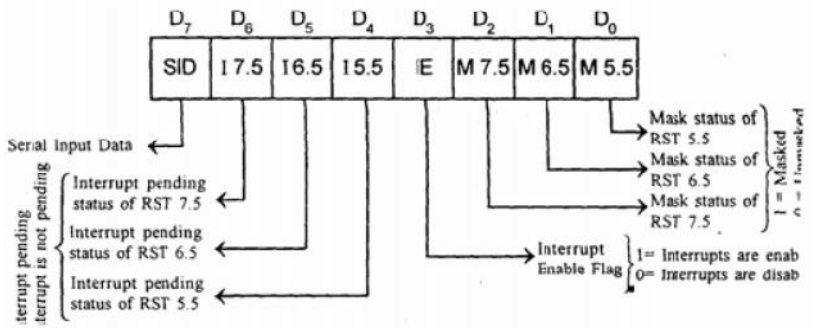
SIM and RIM for interrupts:

- The 8085 provide additional masking facility for RST 7.5, RST 6.5 and RST 5.5 using SIM instruction.

The format of the 8-bit data is shown below.



- The status of these interrupts can be read by executing RIM instruction.
- The masking or unmasking of RST 7.5, RST 6.5 and RST 5.5 interrupts can be performed by moving an 8-bit data to accumulator and then executing SIM instruction.
- The status of pending interrupts can be read from accumulator after executing RIM instruction.
- When RIM instruction is executed an 8-bit data is loaded in accumulator, which can be interpreted as shown in fig.



Interrupt type	Trigger	Priority	Maskable	Vector address
TRAP	Edge and Level	1 st	No	0024H
RST 7.5	Edge	2 nd	Yes	003CH
RST 6.5	Level	3 rd	Yes	0034H
RST 5.5	Level	4 th	Yes	002CH
INTR	Level	5 th	Yes	-

Timing Diagram of 8085 Microprocessor

Timing Diagram is a graphical representation. It represents the execution time taken by each instruction in a graphical format. The execution time is represented in T-states.

Instruction Cycle:

The time required to execute an instruction is called instruction cycle.

Machine Cycle:

The time required to access the memory or input/output devices is called machine cycle.

T-State:

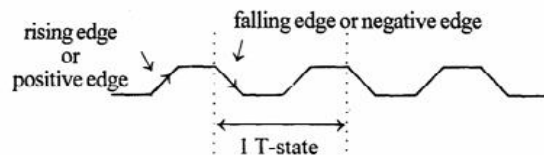
- The machine cycle and instruction cycle takes multiple clock periods.
- A portion of an operation carried out in one system clock period is called as T-state.

MACHINE CYCLES OF 8085:

The 8085 microprocessor has 5 (seven) basic machine cycles. They are

1. Opcode fetch cycle (4T)
2. Memory read cycle (3 T)
3. Memory write cycle (3 T)
4. I/O read cycle (3 T)
5. I/O write cycle (3 T)

Note : Time period, $T = 1/f$; where f = Internal clock frequency



- Each instruction of the 8085 processor consists of one to five machine cycles, i.e., when the 8085 processor executes an instruction, it will execute some of the machine cycles in a specific order.
- The processor takes a definite time to execute the machine cycles. The time taken by the processor to execute a machine cycle is expressed in T-states.
- One T-state is equal to the time period of the internal clock signal of the processor.
- The T-state starts at the falling edge of a clock.

Opcode fetch machine cycle of 8085 :

- Each instruction of the processor has one byte opcode.

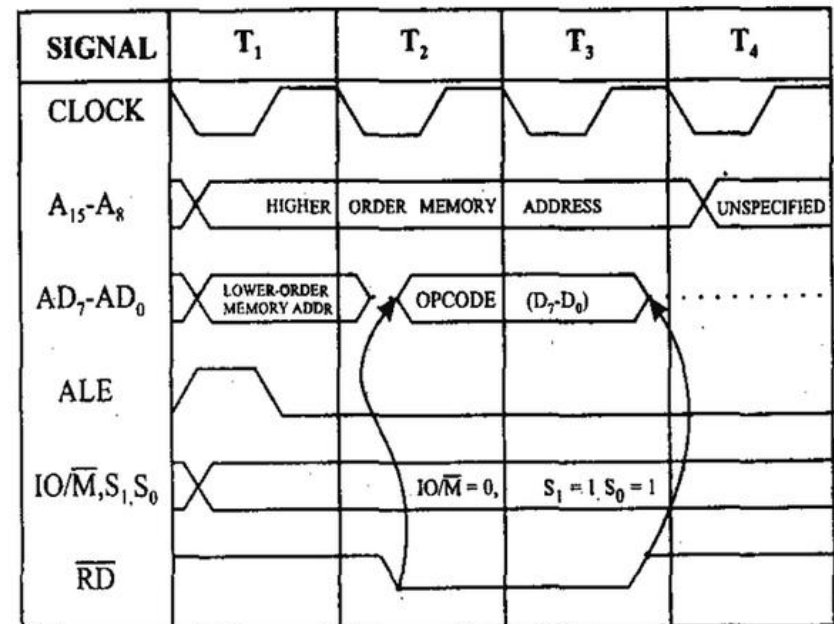


Fig - Timing Diagram for Opcode Fetch Machine Cycle

- The opcodes are stored in memory. So, the processor executes the opcode fetch machine cycle to fetch the opcode from memory.

- Hence, every instruction starts with opcode fetch machine cycle.
- The time taken by the processor to execute the opcode fetch cycle is 4T.
- In this time, the first, 3 T-states are used for fetching the opcode from memory and the remaining T-states are used for internal operations by the processor.

Memory Read Machine Cycle of 8085:

- The memory read machine cycle is executed by the processor to read a data byte from memory.
- The processor takes 3T states to execute this cycle.
- The instructions which have more than one byte word size will use the machine cycle after the opcode fetch machine cycle

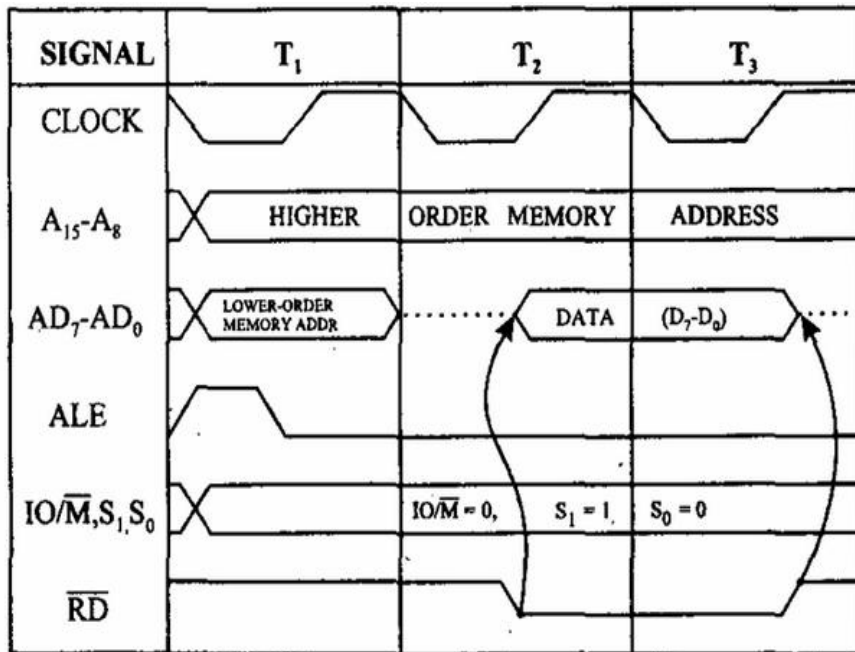
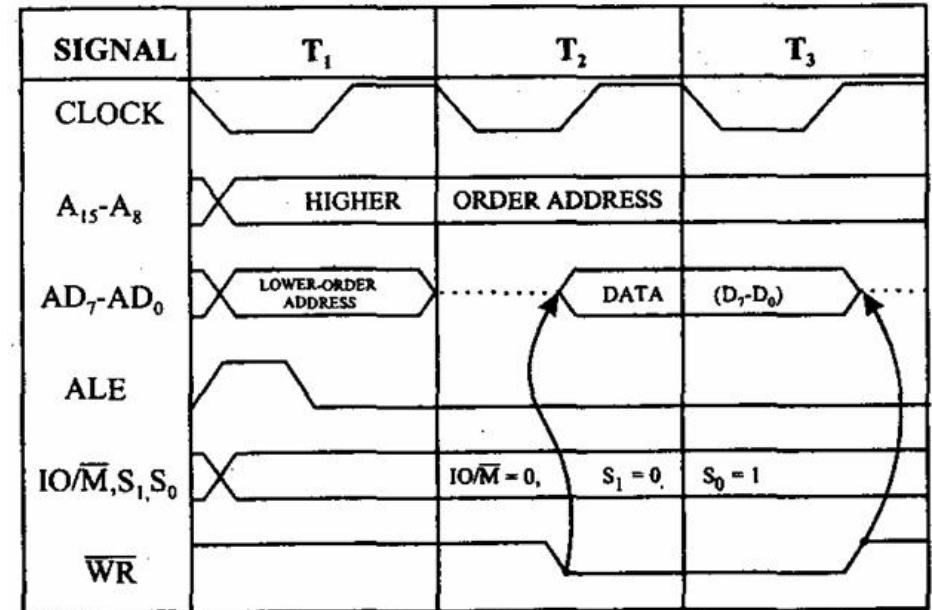


Fig - Timing Diagram for Memory Read Machine Cycle

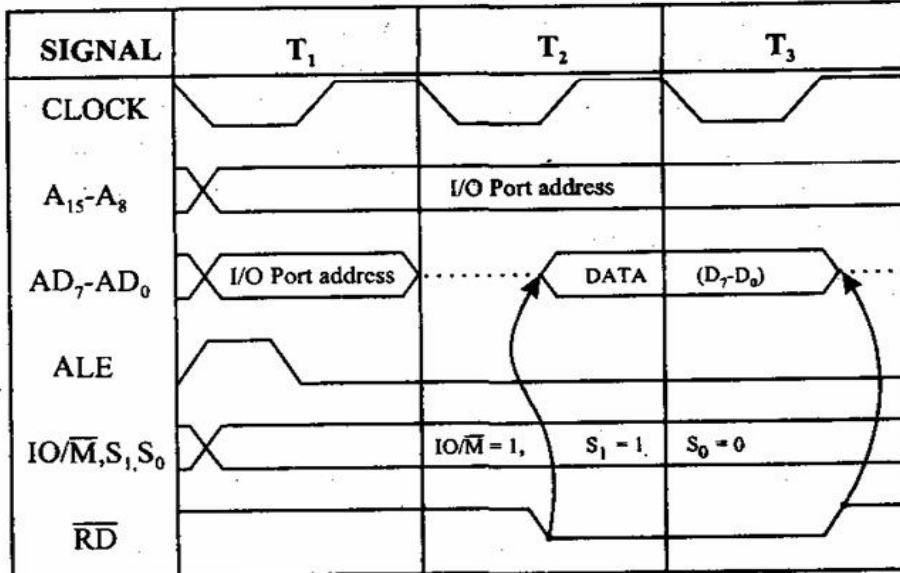
Memory Write Machine Cycle of 8085:

- The memory write machine cycle is executed by the processor to write a data byte in a memory location.
- The processor takes,3T states to execute this machine cycle.



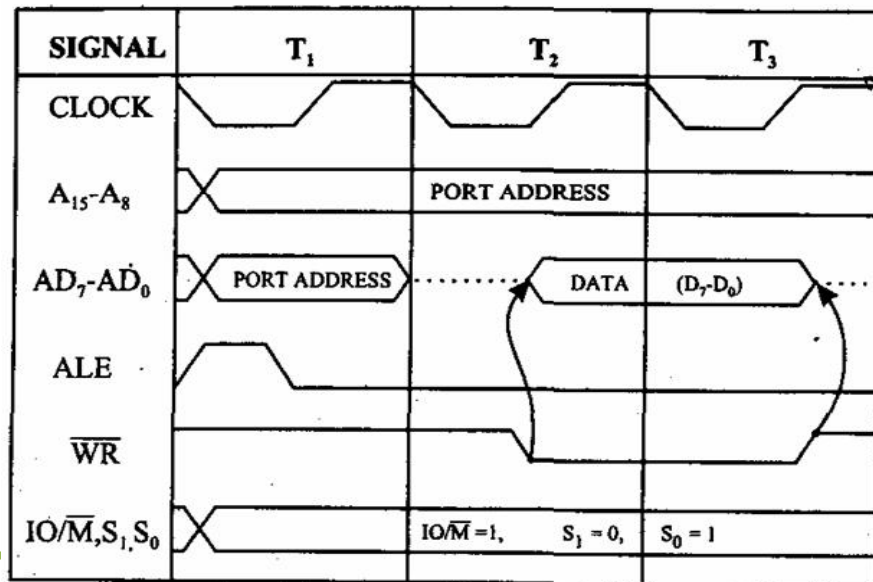
I/O Read Cycle of 8085:

- The I/O Read cycle is executed by the processor to read a data byte from I/O port or from the peripheral, which is I/O, mapped in the system.
- The processor takes 3T states to execute this machine cycle.
- The IN instruction uses this machine cycle during the execution



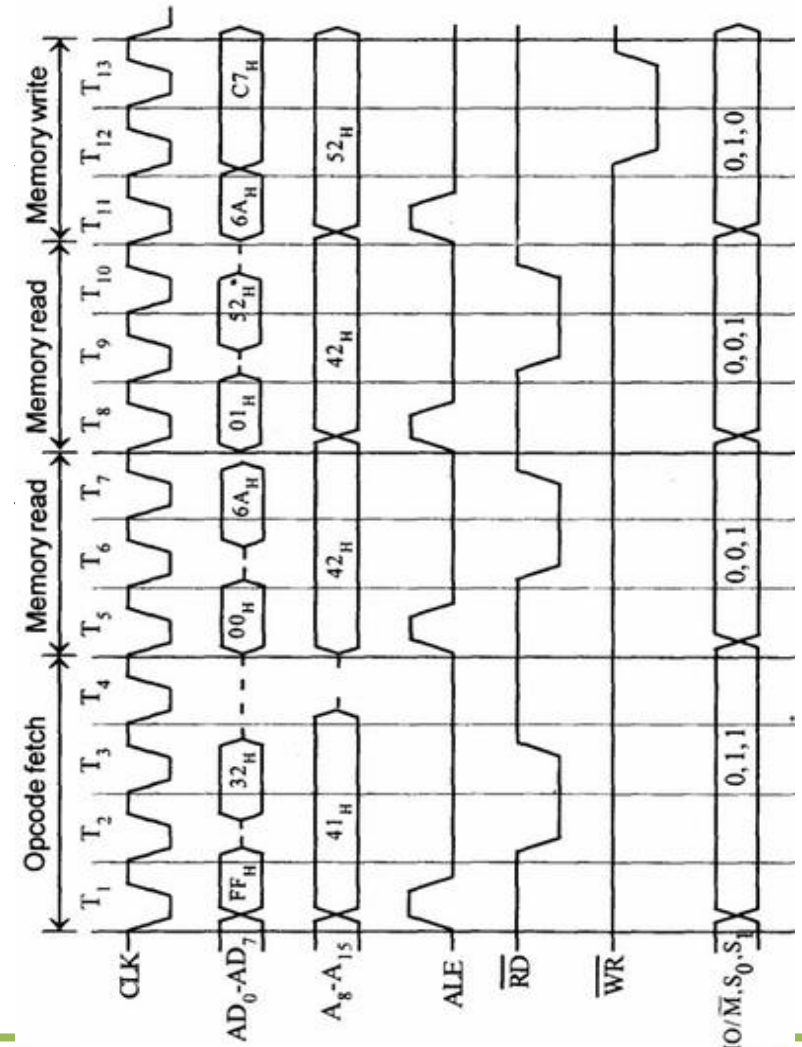
I/O Write Cycle of 8085:

- The I/O write machine cycle is executed by the processor to write a data byte in the I/O port or to a peripheral, which is I/O, mapped in the system.
- The processor takes, 3T states to execute this machine cycle



Timing diagram for STA 526AH.

- STA means Store Accumulator -The contents of the accumulator is stored in the specified address(526A).
- The opcode of the STA instruction is said to be 32H. It is fetched from the memory 41FFH(see fig). - *OF machine cycle*
- Then the lower order memory address is read(6A). - *Memory R*



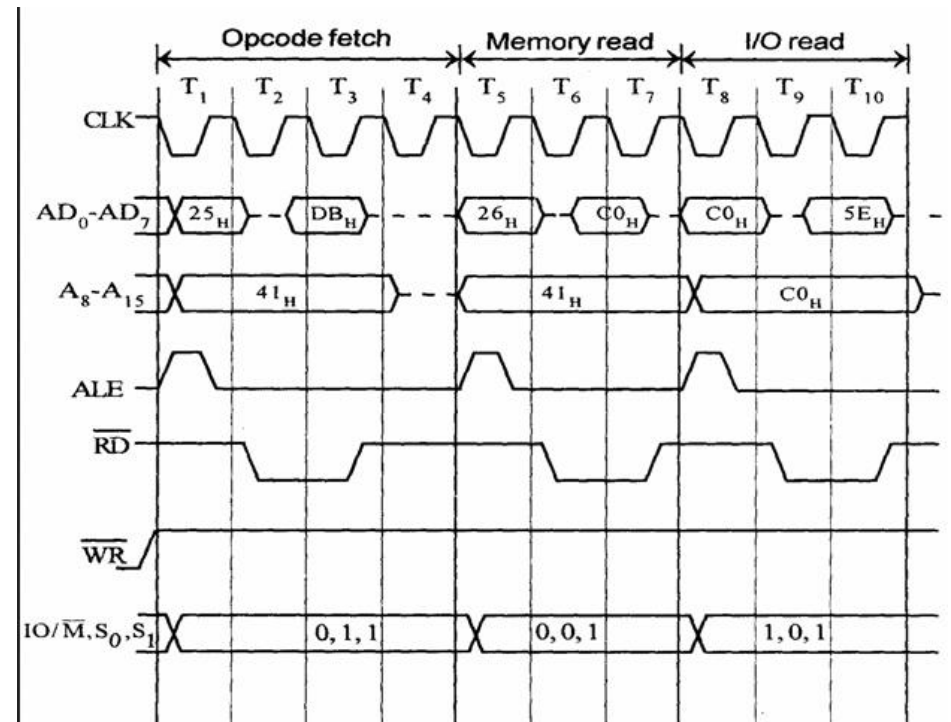
- Read the higher order memory address (52).- *Memory Read Machine Cycle*
- The combination of both the addresses are considered and the content from accumulator is written in 526A. – *Memory Write Machine Cycle*
- Assume the memory address for the instruction and let the content of accumulator is C7H. So, C7H from accumulator is now stored in 526A

Address	Mnemonics	Op code
41FF	STA 526AH	32H
4200		6AH
4201		52H

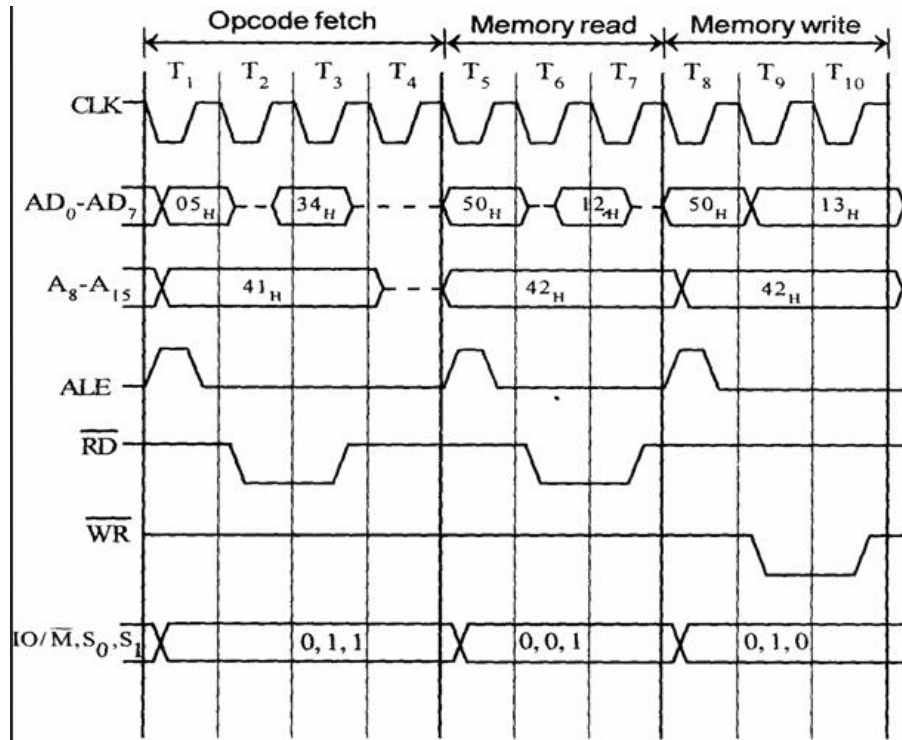
Timing diagram for IN C0H.

- Fetching the Opcode DBH from the memory 4125H.
- Read the port address C0H from 4126H.
- Read the content of port C0H and send it to the accumulator.
- Let the content of port is 5EH.

Address	Mnemonics	Op code
4125	IN C0H	DBH
4126		C0H



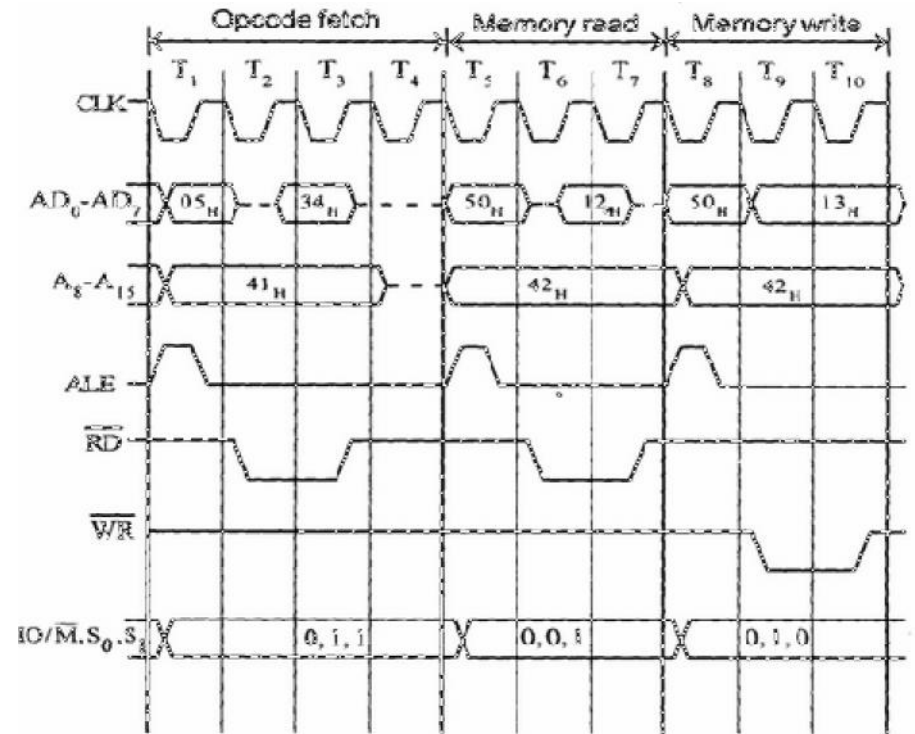
Timing diagram for MVI B, 43H.



Address	Mnemonics	Op code
2000	MVI B, 43H	06H
2001		43H

- Fetching the Opcode 06H from the memory 2000H. (OF machine cycle)
- Read (move) the data 43H from memory 2001H. (memory read)

Timing diagram for INR M

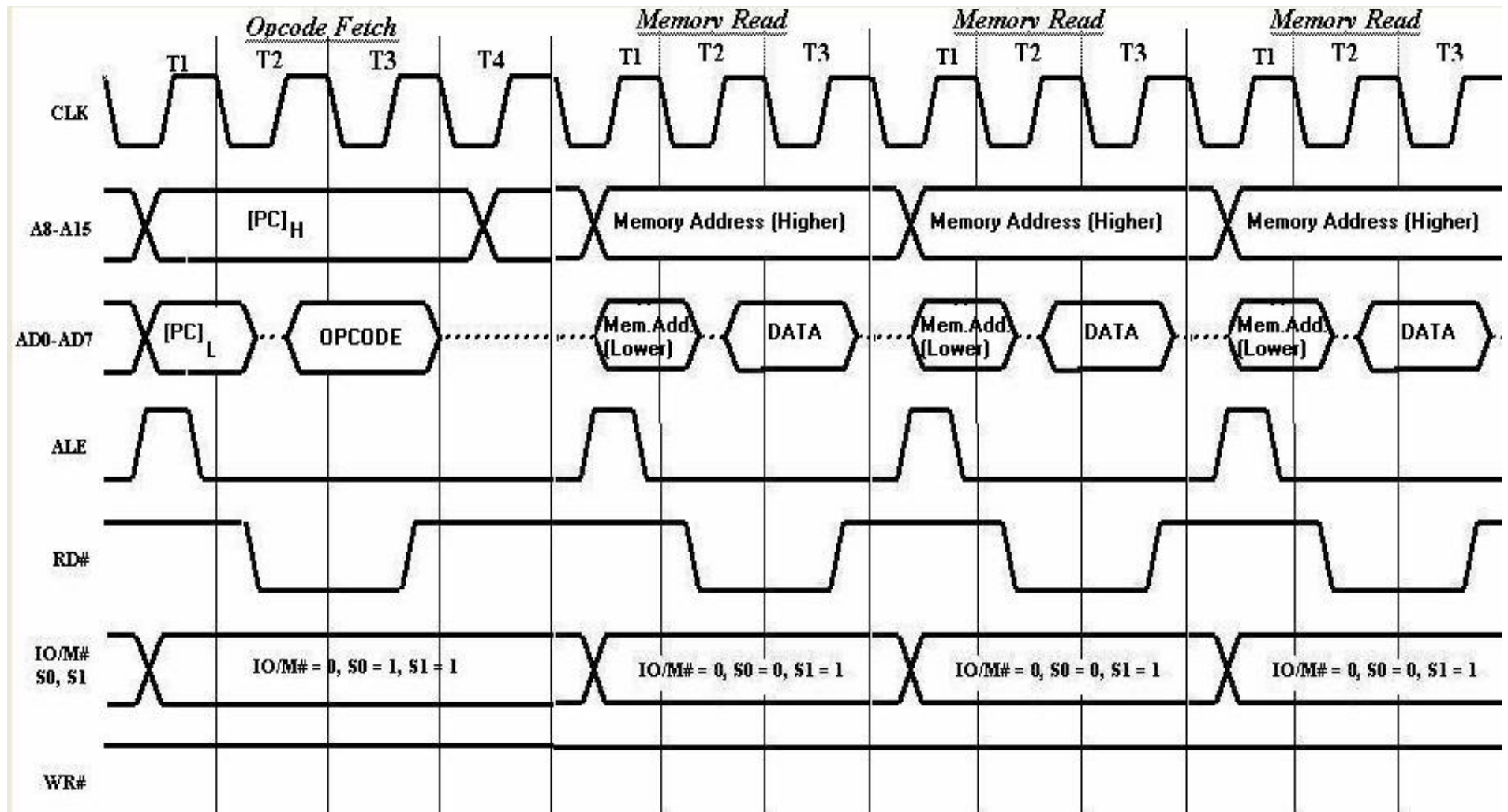


- Fetching the Opcode 34H from the memory 4105H. (OF cycle)

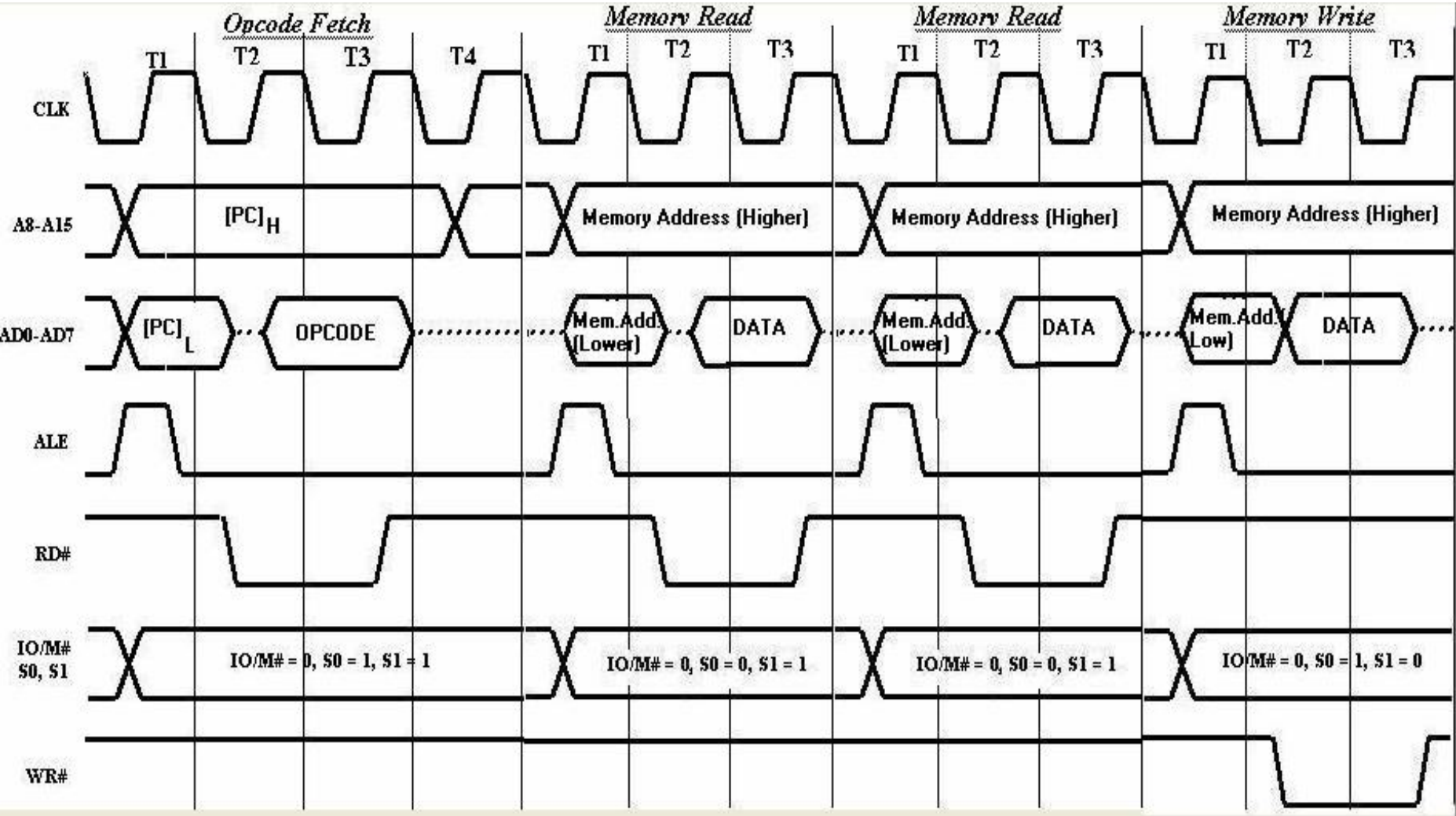
Address	Mnemonics	Op code
4105	INR M	34H

- Let the memory address (M) be 4250H. (MR cycle -To read Memory address and data)
- Let the content of that memory is 12H.
- Increment the memory content from 12H to 13H. (MW machine cycle)

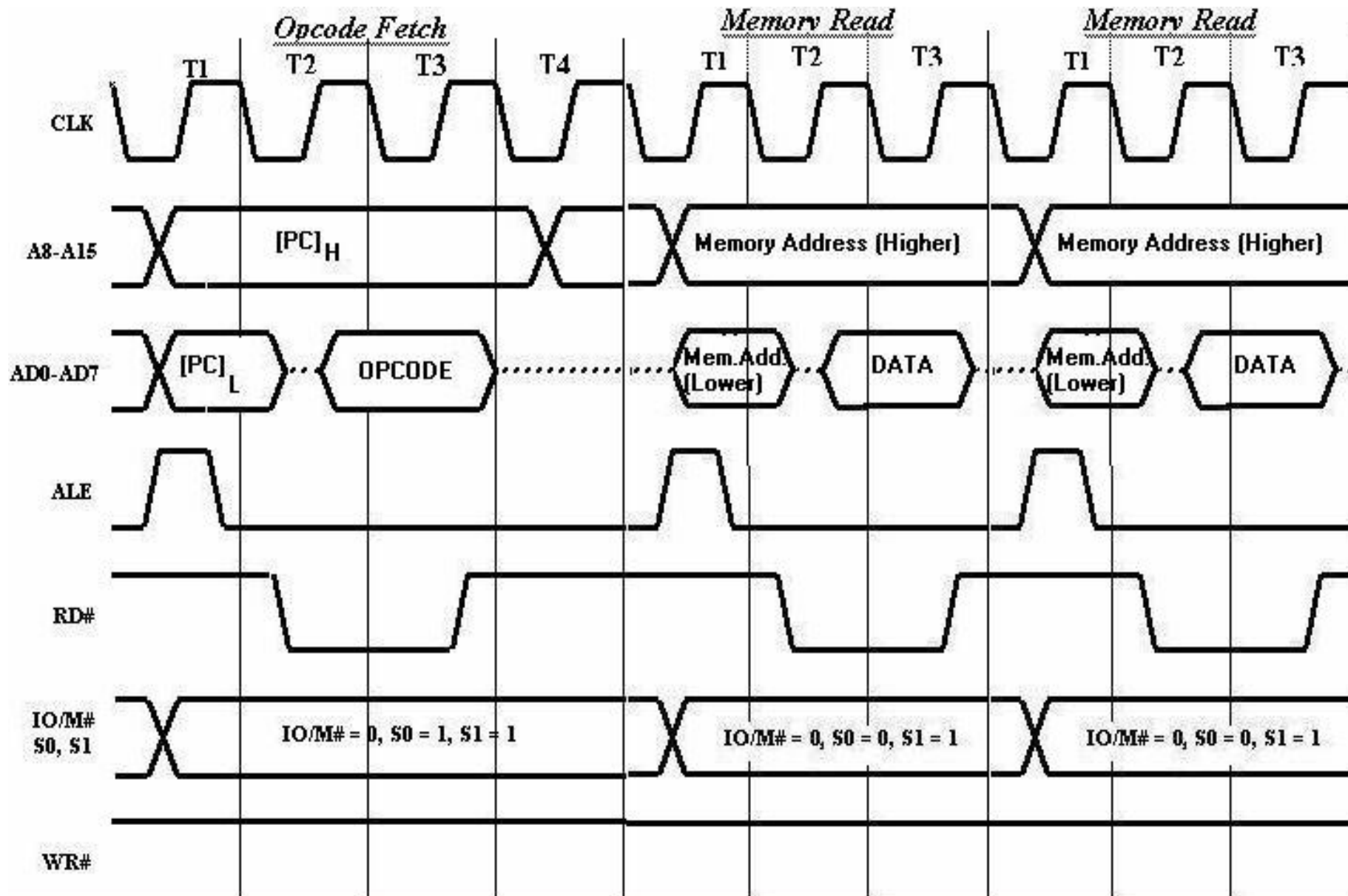
LDA 16bit Addr



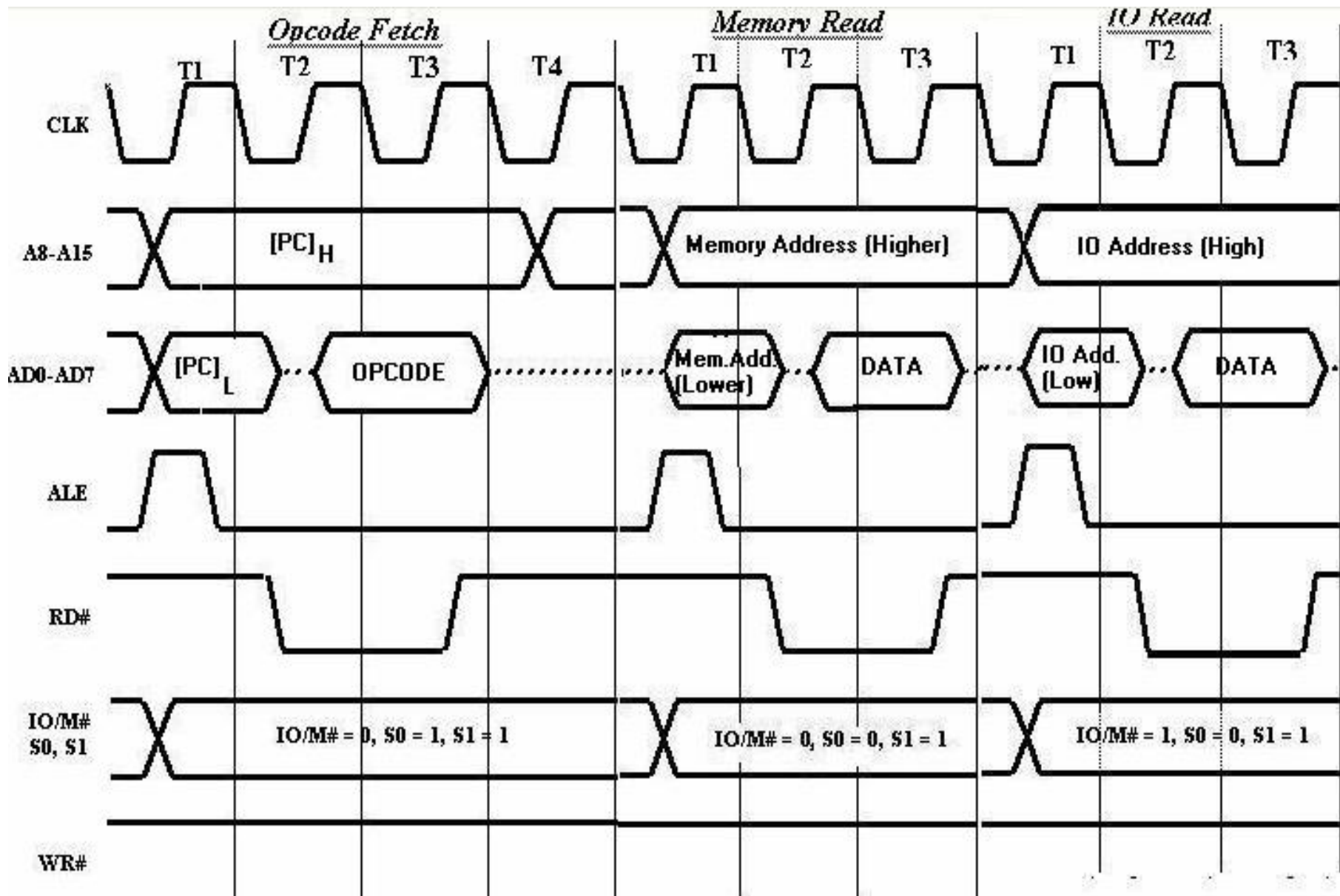
STA 16bit Addr

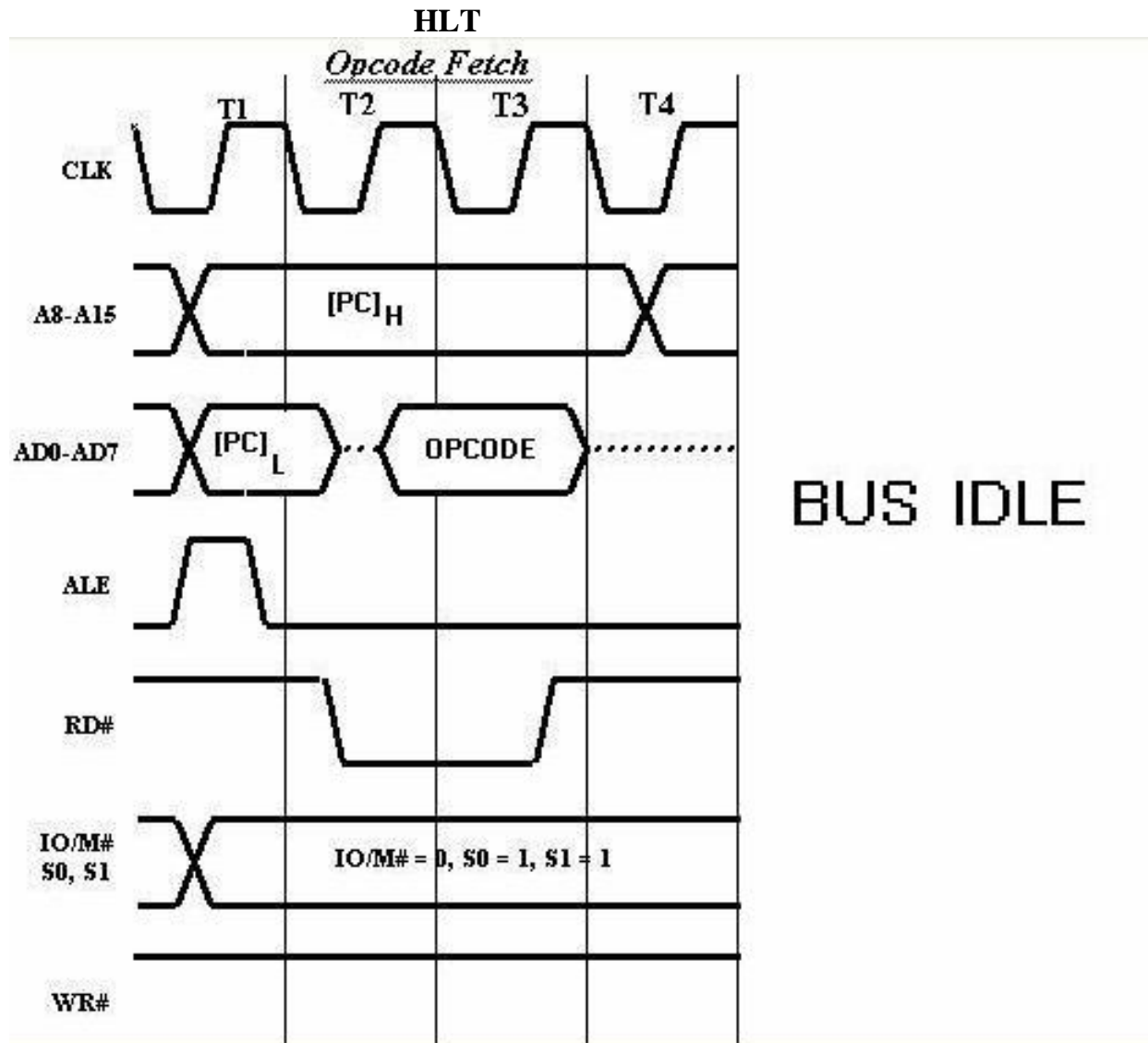


JMP 16bit Addr



IN 8bit PortAddr





SHLD 16bit Addr

