Microprocessor

Unit 3:
Instruction Cycle (3 Hrs)

Course Outline

- Instruction Cycle, Machine Cycle and T-states
  - Machine Cycle of 8085 Microprocessor:
    - op-code fetch, memory read, memory write, I/O read, I/O write, interrupt
  - Fetch and Execute Operation, Timing Diagram
    - Timing Diagram of MOV, MVI, IN, OUT, LDA, STA
  - Memory Interfacing and Generation of Chip Select Signal

Timing Diagram

Timing Diagram

- 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.

Important terms related to timing diagrams:

- **Instruction cycle:**
  - this term is defined as the number of steps required by the processor to complete the entire process
    - ie. Fetching and execution of one instruction.
  - The fetch and execute cycles are carried out in synchronization with the clock.
- **Machine cycle:**
  - It is the time required by the microprocessor to complete the operation of accessing the memory devices or I/O devices.
  - In machine cycle, various operations like opcode fetch, memory read, memory write, I/O read, I/O write are performed.
- **T-state:**
  - Each clock cycle is called as T-states.

T-state

- One T-state is equal to the time period of the internal clock signal of the microprocessor

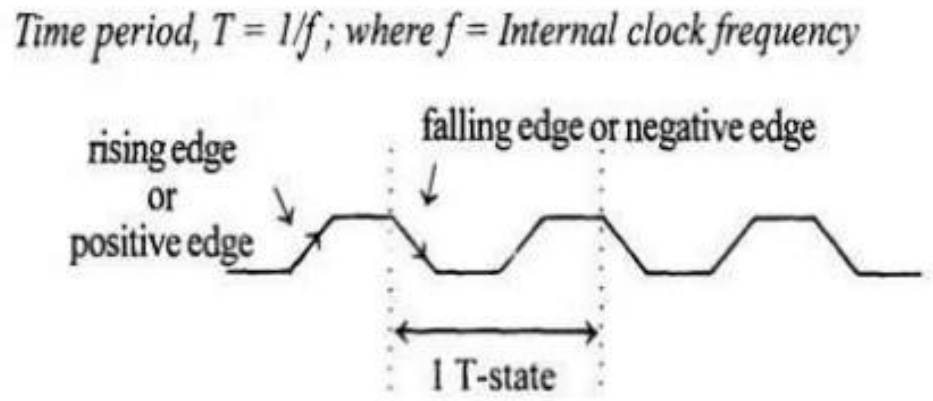


Fig 1.7 Clock Signal

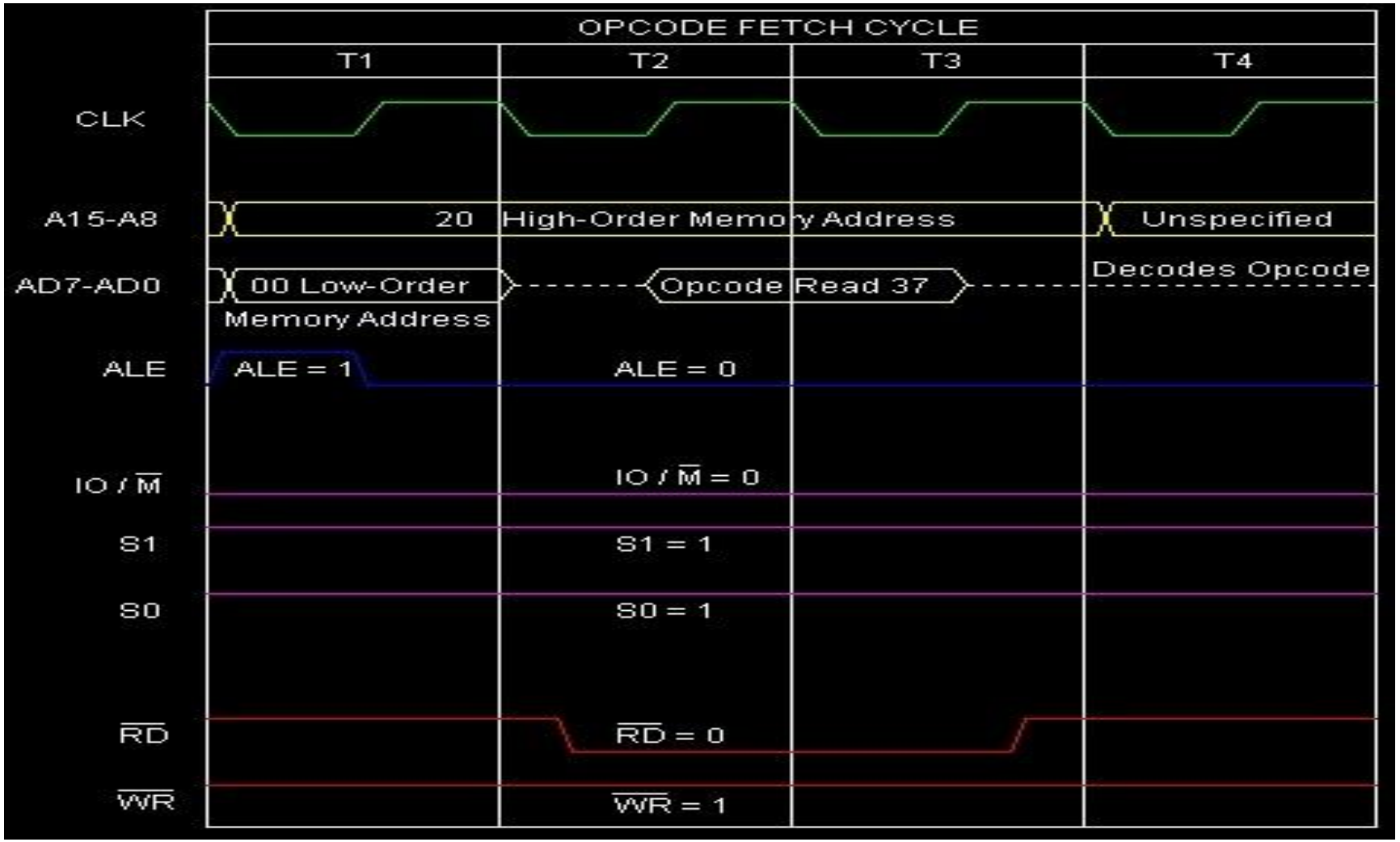
- For Ex: If the internal clock frequency of 8085 microprocessor is 3 MHZ, One T-state is equal to

  = =0.333x10-6sec=333x10-9sec. (333 nano seconds nearly)

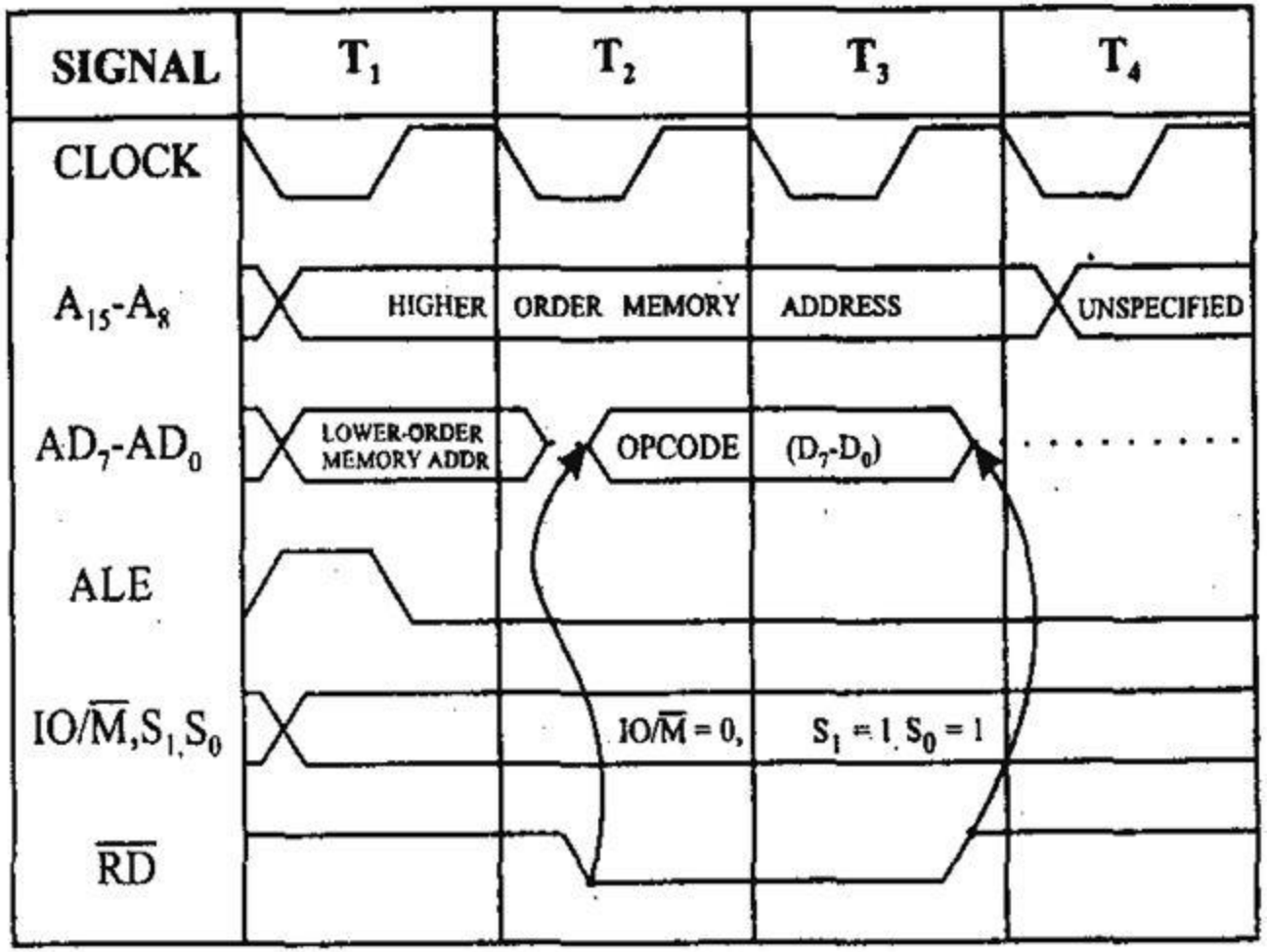
Machine cycles of 8085

- The 8085 microprocessor has 7 (seven) basic machine cycles. They are
  - Opcode fetch cycle (4T)
  - Memory read cycle (3 T)
  - Memory write cycle (3 T)
  - I/O read cycle (3 T)
  - I/O write cycle (3 T)
  - Interrupt Acknowledgement (3T)
  - Bus Idle (3T)

Opcode Fetch



Opcode Fetch Machine Cycle



Opcode Fetch

- **1st T state**
  - During the first T state, the address of the location where the opcode is stored is loaded on the address bus. In 8085, this address is stored in a 16-bit register called the program counter. Higher eight bits of the address are loaded on A8-A15, and the lower eight bits of the address are loaded into AD0-AD7 for demultiplexing.
  - Also, the ALE signal becomes active in the first T state to indicate that the data on AD0-AD7 pins are the lower address bits.
  - IO/M signal becomes low at the beginning of the first T state to indicate that the opcode will be fetched from memory (reading from memory).
  - At the beginning of the first T state, signals S1 and S0 take the value 1 and 1 respectively to indicate that it is an opcode fetch machine cycle.
- **2nd T state**
  - By the beginning of the 2nd T state or the end of 1st T state, the ALE signal goes low. By this time, 8085 expects that the lower address bits are latched, and AD0-AD7 is free to be used as a data bus.
  - At the beginning of the second T state, RD goes low, indicating that the read process has started. Meanwhile, higher address bits are present in A8-A15, and lower address bits are expected to be latched.
  - As RD goes low, the opcode (eight bits) is loaded into the data bus AD0-AD7.

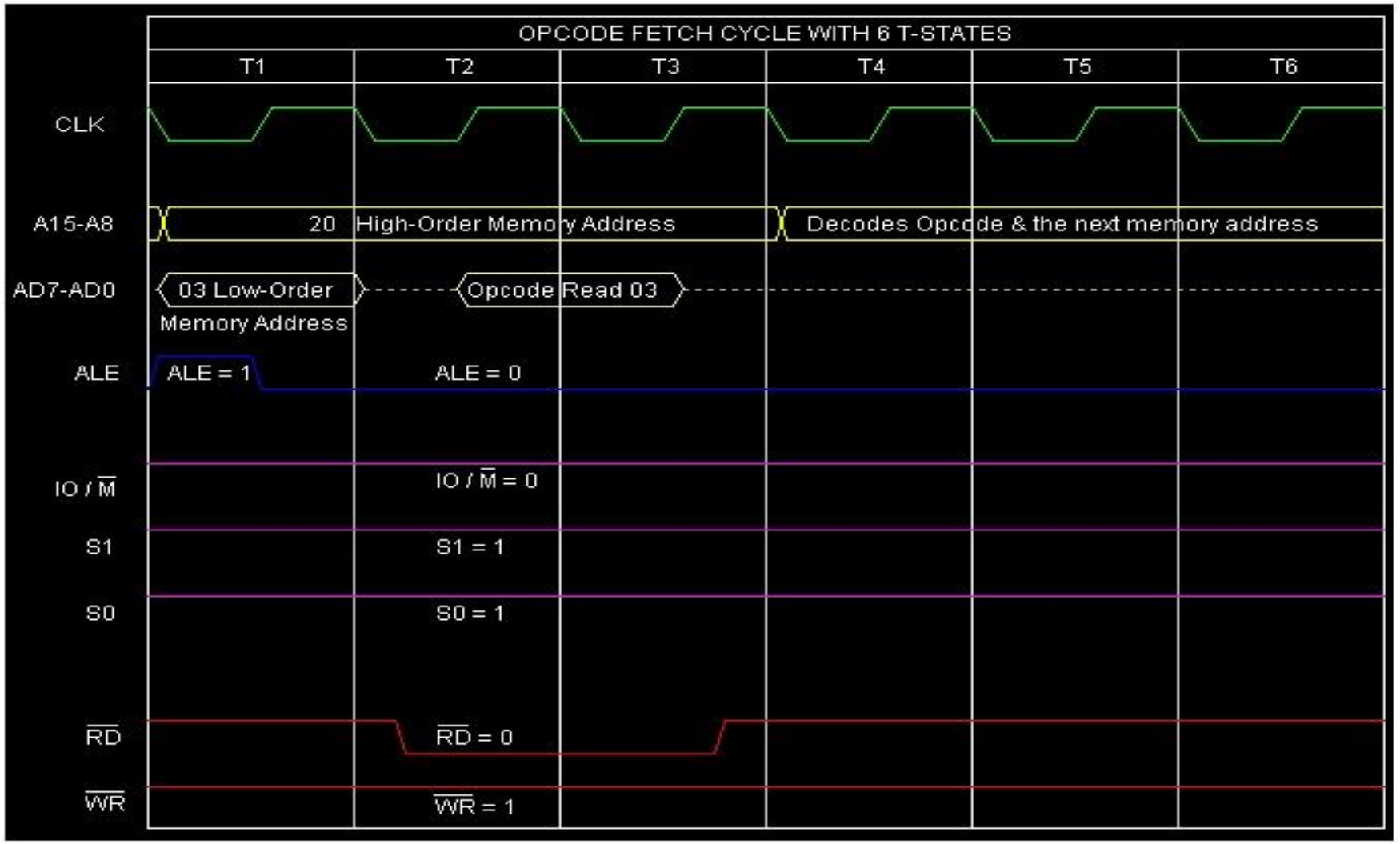
Opcode Fetch

- **3rd T state**
  - The opcode loaded on the data bus is present there until the middle of the third T state.
  - During the third T state, RD goes up, indicating that the read operation is completed and ‘the opcode is fetched’ and placed in the instruction register.
  - The data on the data bus and the higher address bits on A8-A15 exist until the middle of this T state.
- **4th T state**
  - During the fourth T state, the fetched opcode is decoded. There is nothing much to observe in the timing diagram during this process.
  - In case of some simple one-byte instructions like STC (set carry flag), execution is also completed during the fourth T state. One such instruction is MOV A, D.
  - During the fourth T state, after decoding the opcode, the microprocessor decides if it needs fifth and sixth T states, or should proceed to the next machine cycle.
  - PC is incremented by 1 here or in the sixth T state if the OFMC is extended upto sixth T state.

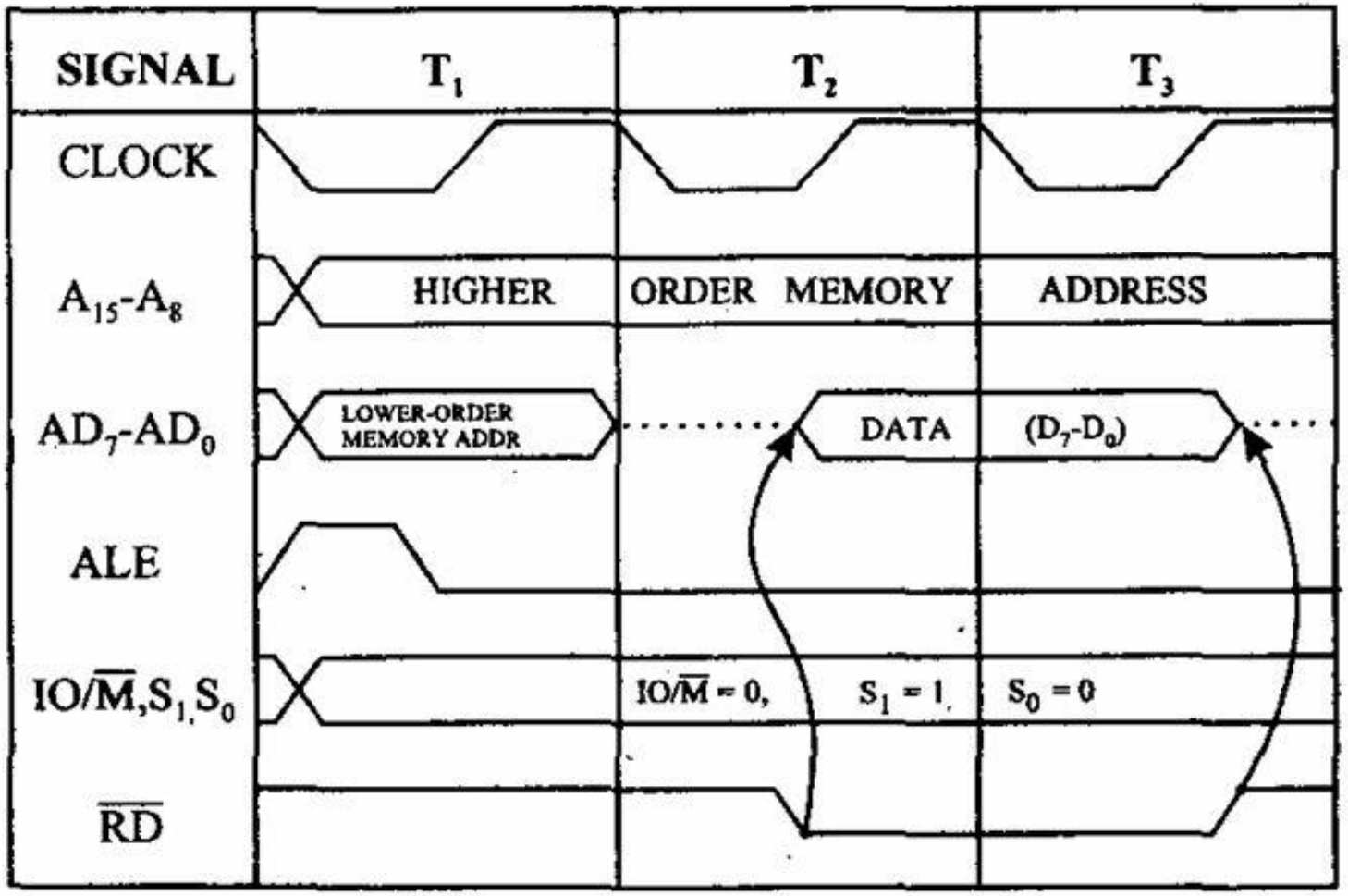
Opcode Fetch

- **5th and 6th T state**
  - In case of one-byte instructions that operate on 16-bit data and some other instructions, OFMC may extend up to six T states.
  - During the fifth and sixth T states, execution of these instructions takes place.
  - Since these instructions are simple, they get executed in the OFMC itself.
  - Examples of such instructions are DCX, INX, PCHL, SPHL, CALL, RSTN and conditional RET.

INX rp


OPCODE FETCH CYCLE WITH 6 T-STATES
T1
T2
T3
T4
T5
T6
CLK
A15-A8
20
High-Order Memory Address
Decodes Opcode & the next memory address
AD7-AD0
03 Low-Order
Memory Address
Opcode Read 03
ALE
ALE = 1
ALE = 0
IO / M
IO / M = 0
S1
S1 = 1
S0
S0 = 1
RD
RD = 0
WR
WR = 1

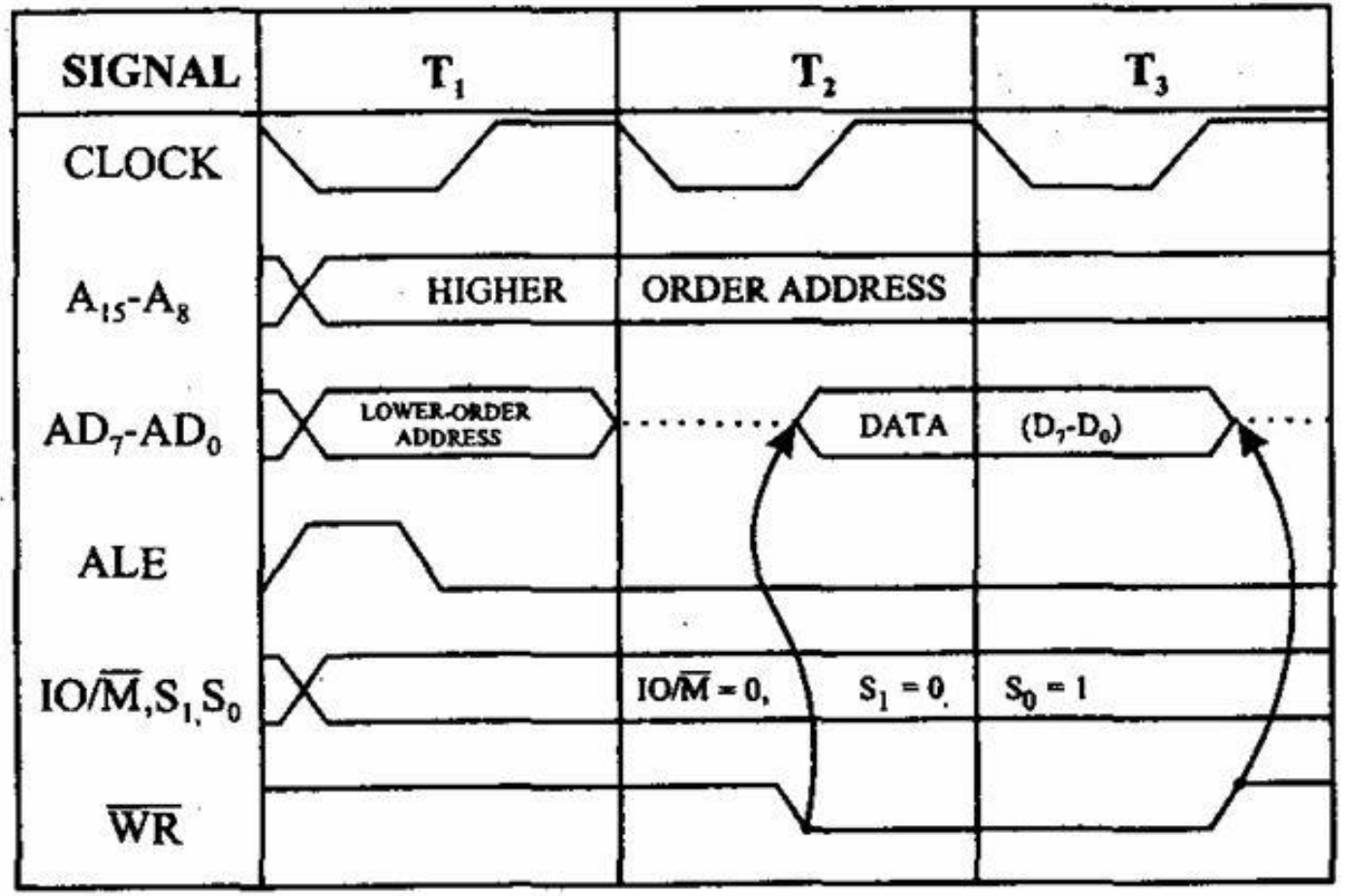
Memory Read Machine Cycle



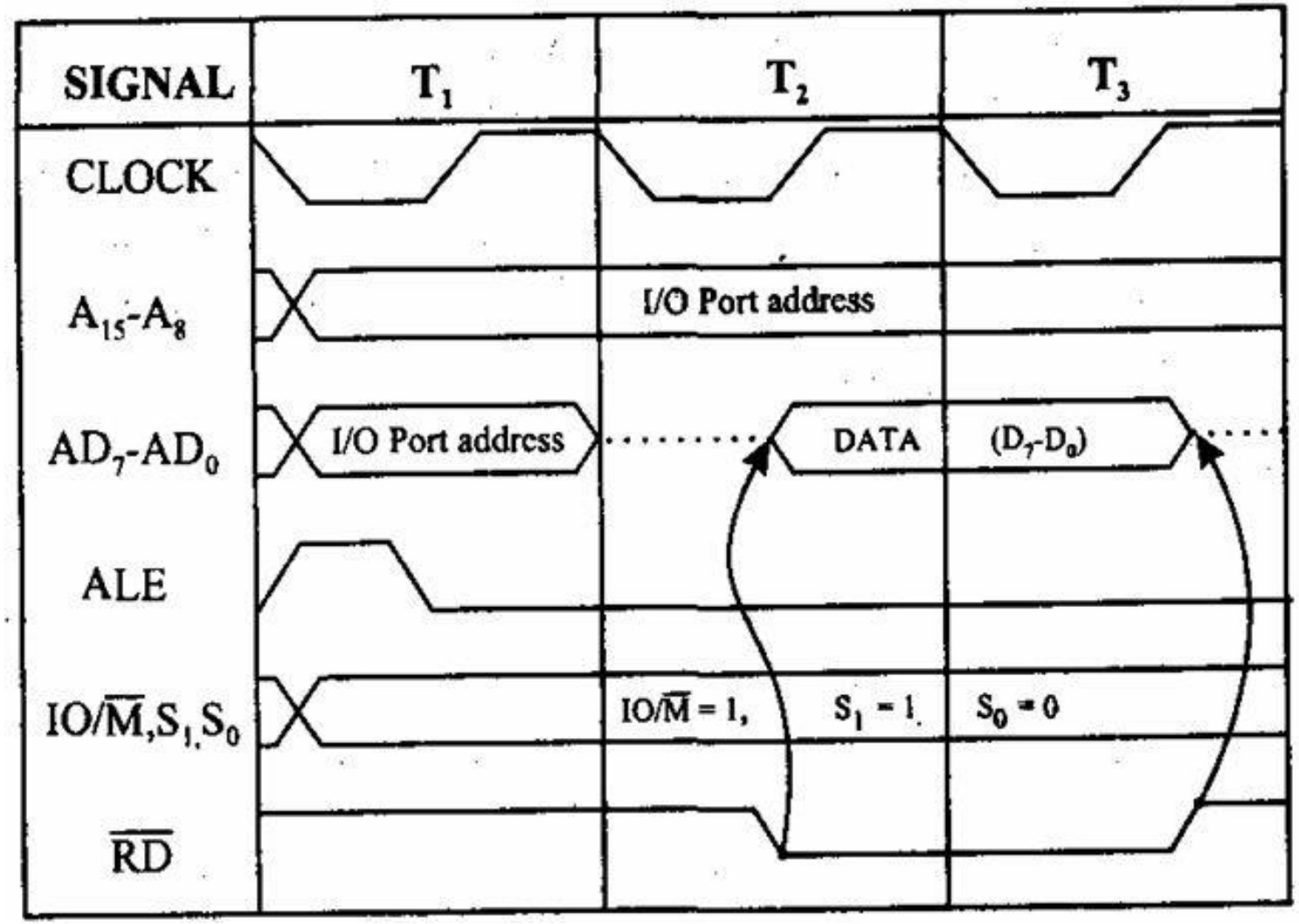
Memory Read

- **1st T state**
  - Higher address bits loaded into A8-A15.
  - Lower address bits loaded into AD0-AD7.
  - ALE signal goes high in the beginning to indicate that AD0-AD7 contains lower address bits.
  - IO/M goes low since it is a memory operation.
  - S1 and S0 become 1 and 0 respectively, indicating Memory Read Machine Cycle.
  - ALE goes low by the end of the first T state. Lower address bits are expected to be latched by this time.
- **2nd and 3rd T states**
  - RD goes low, indicating the initiation of the read operation.
  - Data is read from the memory location and is loaded into the data bus AD0-AD7. The data is loaded into the data bus at the beginning of the 2nd T state and exists until the end of the third T state.
  - By the end of the third T state, RD goes high, indicating the end of the read operation.

Memory Write Machine Cycle



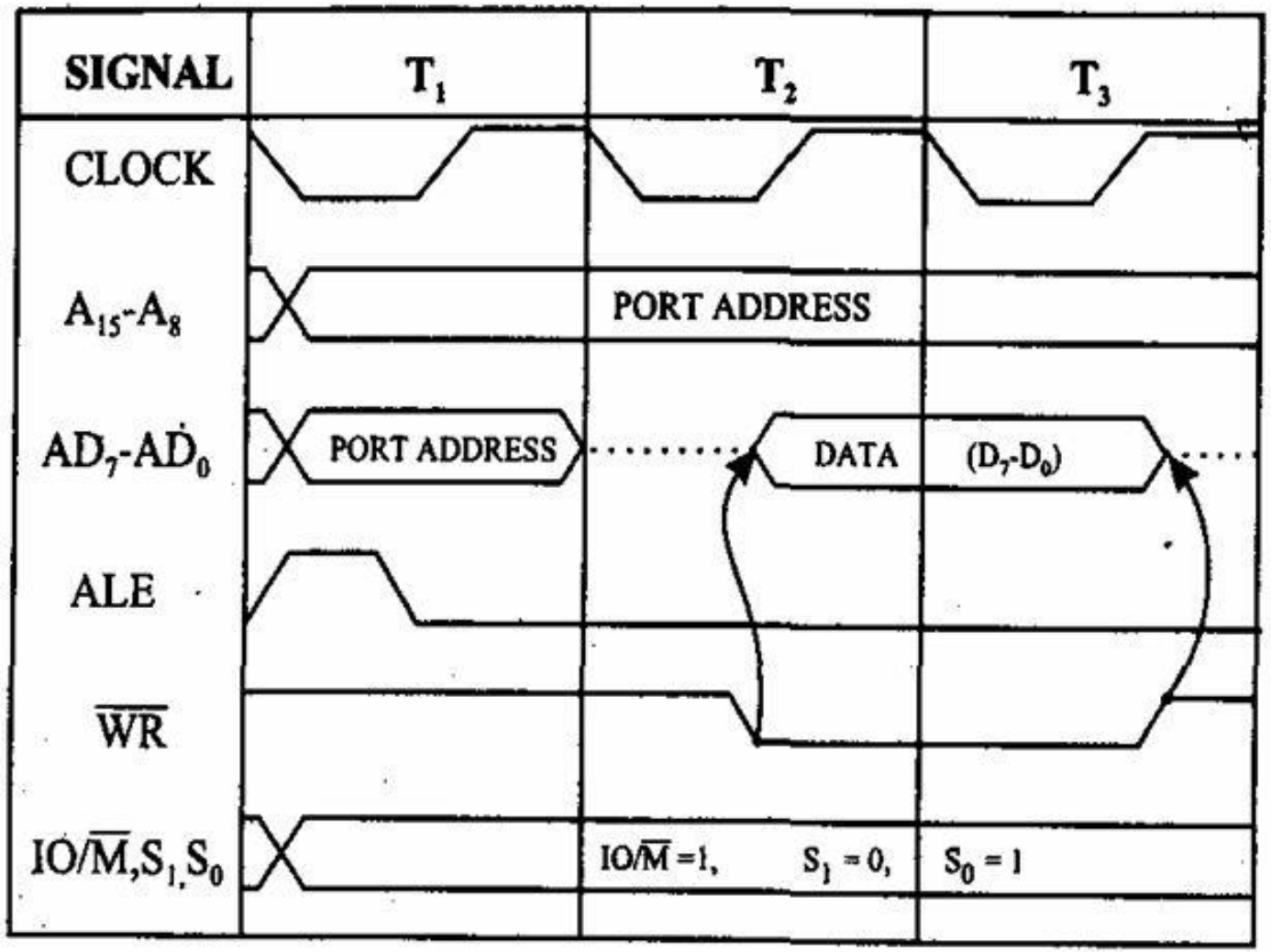
IO Read Machine Cycle



IO Read

- **1st T state**
  - 8-bit address is loaded into A8-A15.
  - The same 8-bit address is loaded into AD0-AD7.
  - ALE signal goes high in the beginning to indicate that AD0-AD7 contains address bits and not data bits.
  - IO/M goes high since the microprocessor is dealing with an IO device.
  - S1 and S0 become 1 and 0 respectively, indicating a ‘read’ machine cycle.
  - ALE goes low by the end of the first T state. Address bits in AD0-AD7 are expected to be latched by this time.
- **2nd and 3rd T states**
  - RD goes low, indicating the initiation of the read operation.
  - Data is read and is loaded on the data bus (AD0-D7) at the beginning of the second T state and exists until the end of the third T state.
  - In the third T state, the data is transferred from the data bus to the accumulator.
  - By the end of the third T state, RD goes high, indicating the end of the read operation.

IO Write Machine Cycle



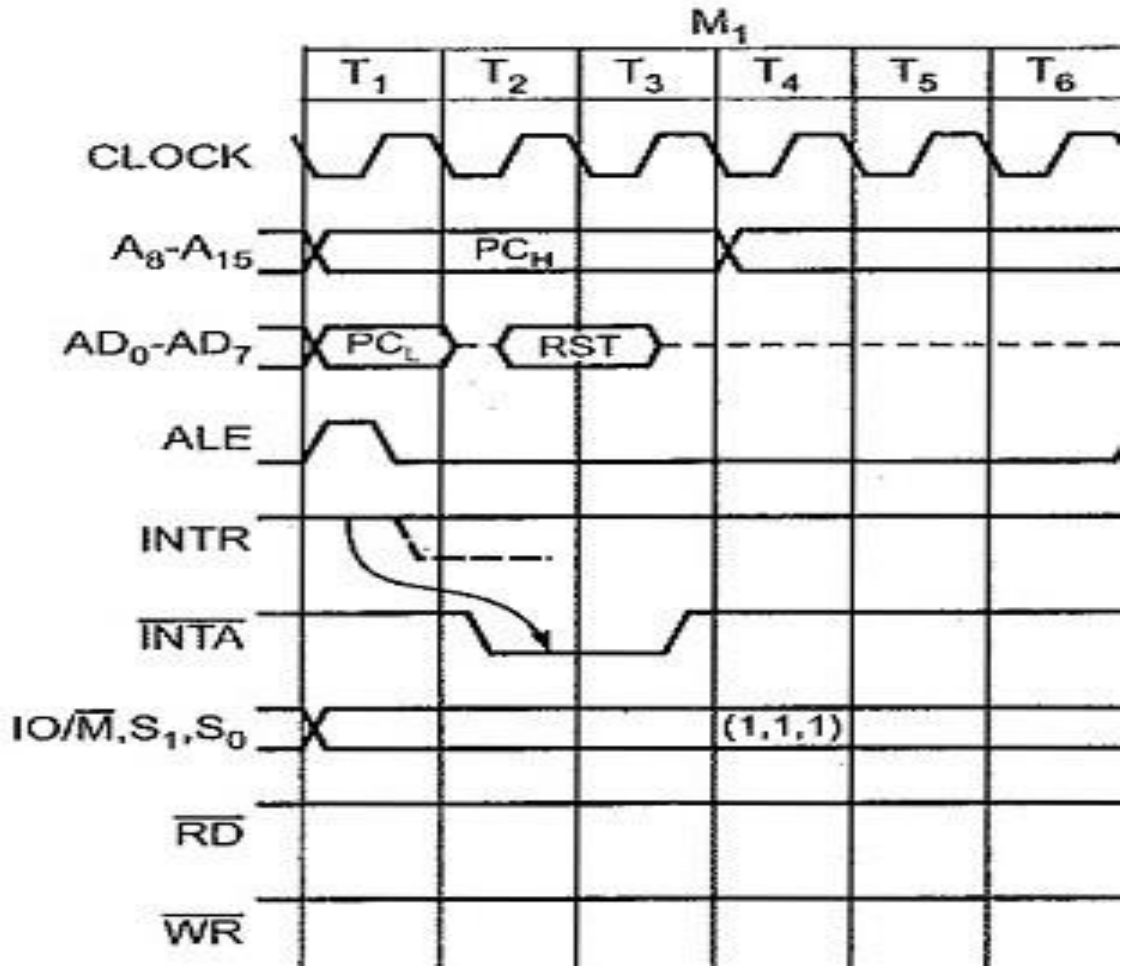
Interrupt acknowledge machine cycle

- Before proceeding on to talking about the interrupt mechanism in 8085.
- An external IO device issues an interrupt signal (INTR), to tell the microprocessor that it wants a certain task done.
- When a microprocessor receives such a signal, it responds by making the $\overline{INTA}$ signal (interrupt acknowledgement signal) low for some time to tell the device that it has received the request and will take the necessary actions.
- Responding to an external device with $\overline{INTA}$ signal after receiving an interrupt request – this whole process is carried out in a machine cycle called “Interrupt acknowledge machine cycle.”

Interrupt acknowledge machine cycle

- To interrupt the 8085 microprocessor, we usually execute one of the two instructions – RST or CALL.
- The RST instruction has only one interrupt acknowledge cycle of 6 T-states.
- Whereas the CALL instruction has three interrupt acknowledge cycles (First -> 6 T-states; Second and Third -> 3 T-states.) Why? Because RST is a one-byte instruction and CALL is a three-byte instruction.
- Note that the addresses for the Interrupt Subroutine (ISR) will be provided by the instructions themselves. Hence, the PC won't be incremented.

Interrupt acknowledge machine cycle



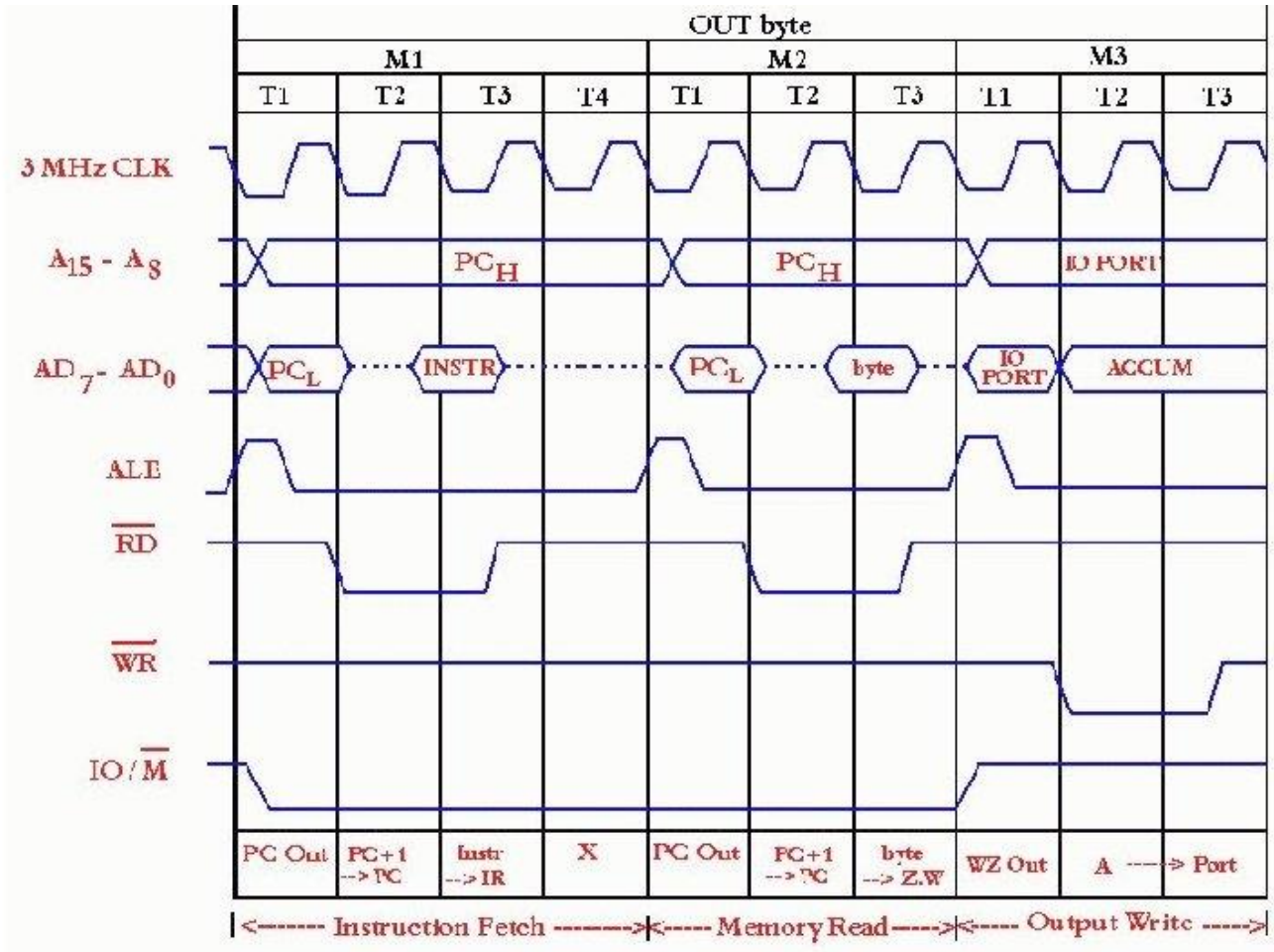
Interrupt acknowledge machine cycle

- **1st T state**
  - The first T state of all the machine cycles involving data transfer is for the demultiplexing of AD0-AD7. The same is the case here. A8-A15 contains higher address bits. ALE signal goes high. AD0-AD7 contains address for that interval of time.
  - The difference here is that the INTA signal is high. RD and WR both are low. IO/M = 1 and S1S0 = 1
- **2nd and 3rd T state**
  - These are also similar to the 2nd and 3rd T state of OFMC.
  - ALE goes low. AD0-AD7 is ready for data transfer and now contains the data until the middle of the third T state.
  - Remember, we discussed that some OFMC are of 6 T states? Similar to that, an IAMC is also of 6 T states.
- **4th, 5th, and 6th T state**
  - During the last three T states, instruction is decoded, and according to the decoded instructions, further machine cycles are executed to complete that instruction.
  - The decoded instruction is either CALL instruction or RST instruction. The execution of decoded instruction is completed in the subsequent machine cycles.

Number and Type of machine cycles for different instructions

S No	Instruction	No: of machine cycles	Machine cycle - 1	Machine cycle - 2	Machine cycle - 3	Machine cycle - 4
1	MOV A,B	1	OF	-	-	-
2	MVI A, 50H	2	OF	MR	-	-
3	LDA 5000H	4	OF	MR	MR	MR
4	STA 5000H	4	OF	MR	MR	MW
5	IN 80H	3	OF	MR	IOR	-
6	OUT 80H	3	OF	MR	IOW	-

OUT instruction



INR M

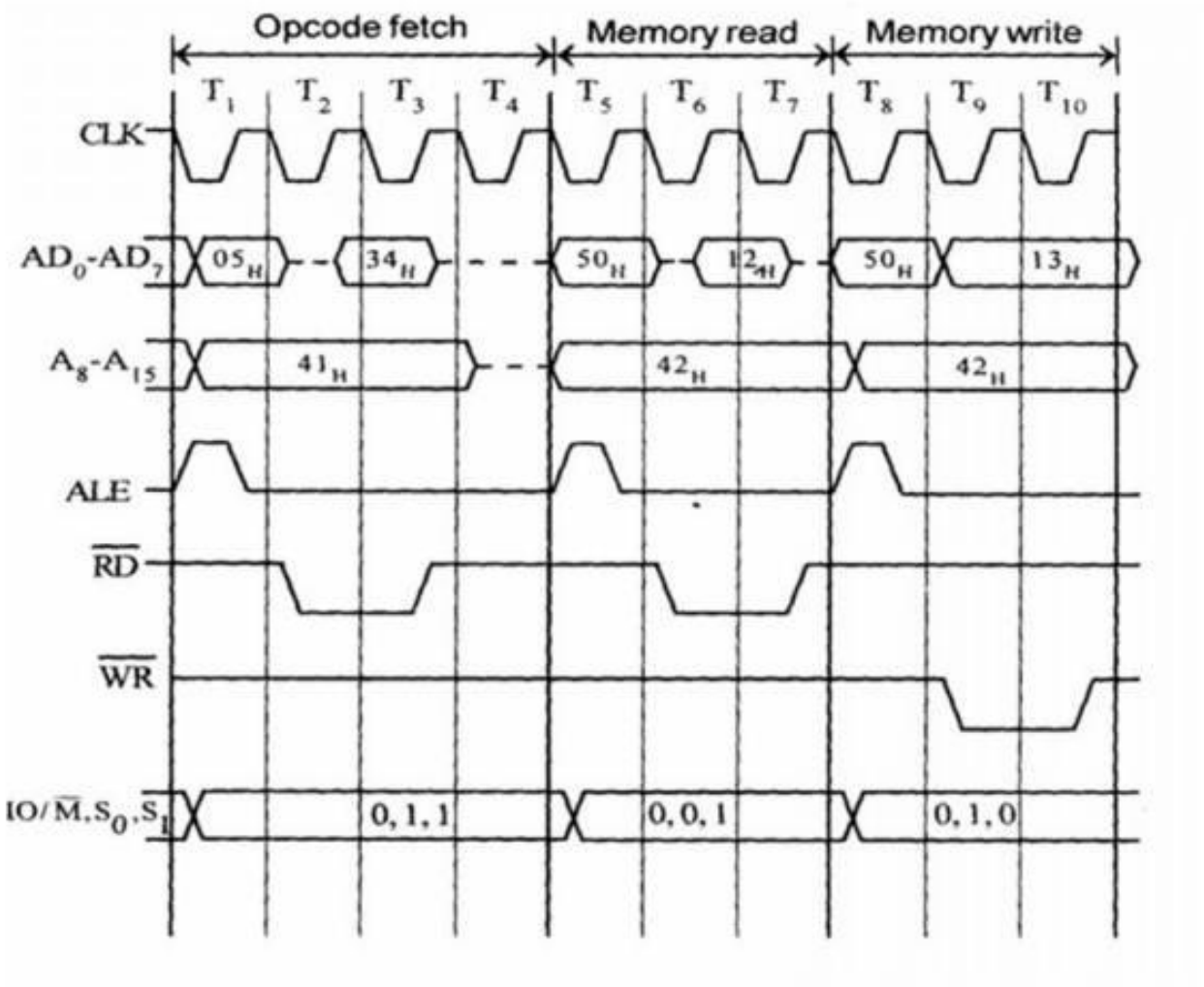


Fig 1.13 Timing Diagram for INR M

Memory Interfacing

Interfacing

- Interface is the path for communication between two components.
- Interfacing a microprocessor is to connect it with various peripherals to perform various operations to obtain a desired output.
- Types of interfacing:
  - **Memory Interfacing** is used when the microprocessor needs to access memory for reading and writing data stored in the memory . It is used when reading/writing to a specific register of a memory chip.
  - **I/O Interfacing** is achieved by connecting keyboard(input) and display monitors(output) with the microprocessor.

Memory Interfacing

- 8085 can support a memory chip of size up to 64 kB.
- We can interface a memory chip of size less than that too.
- Also, we can interface several memory chips to a single 8085 microprocessor, until and unless their combined size does not exceed 64 kB.

Memory Chips: Types

- Memory chips come in a variety of types and with different storage capacities.
- A broad classification of memory chips based on their read and write capability is:
  - **RAM (Random Access Memory):** We can read as well as write data on this type of memory. The chip of this type has pins for both memory read and memory write signals.
  - **ROM (Read Only Memory):** As the name suggests, we can only write data on this type of memory chip. The chip of this type has a pin only for memory read signal.
- **ROM** is used to store programs, while **RAM** is used to store the data.

Memory Chips: Logic Diagram

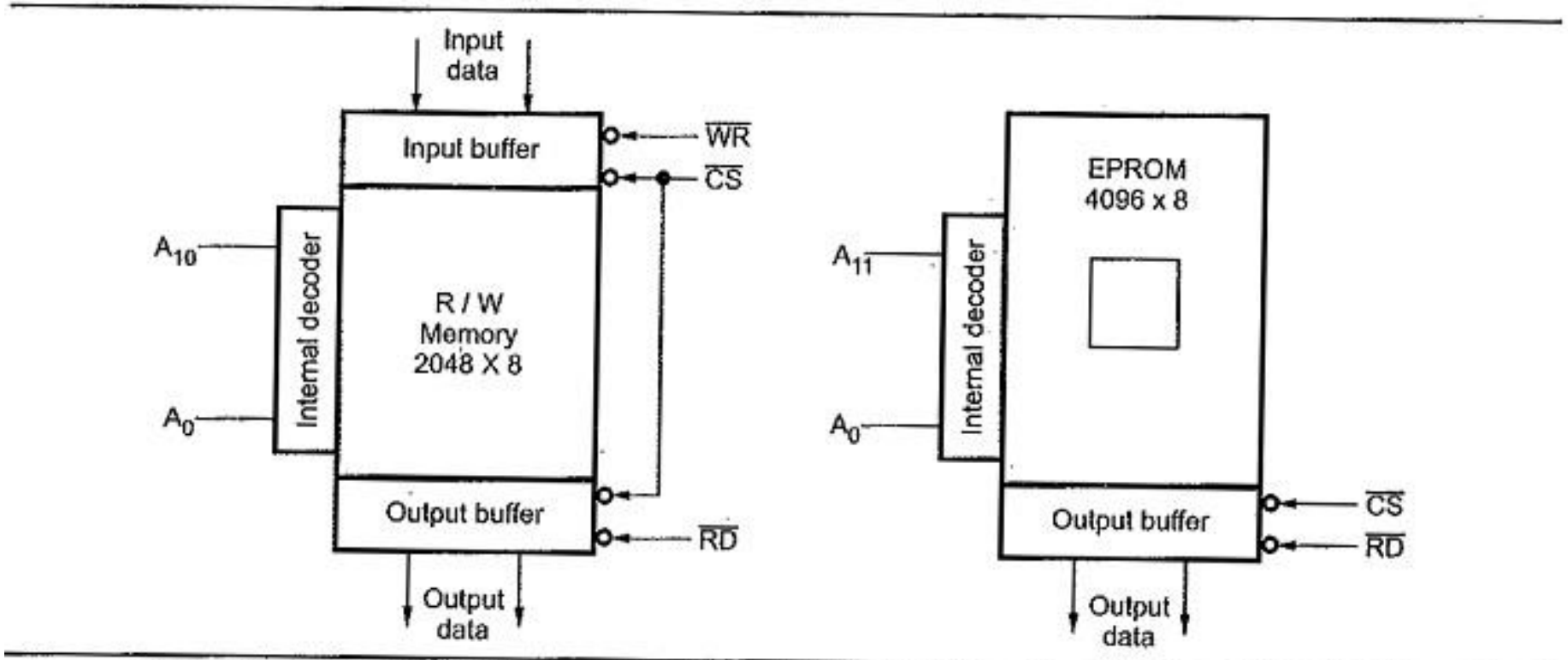
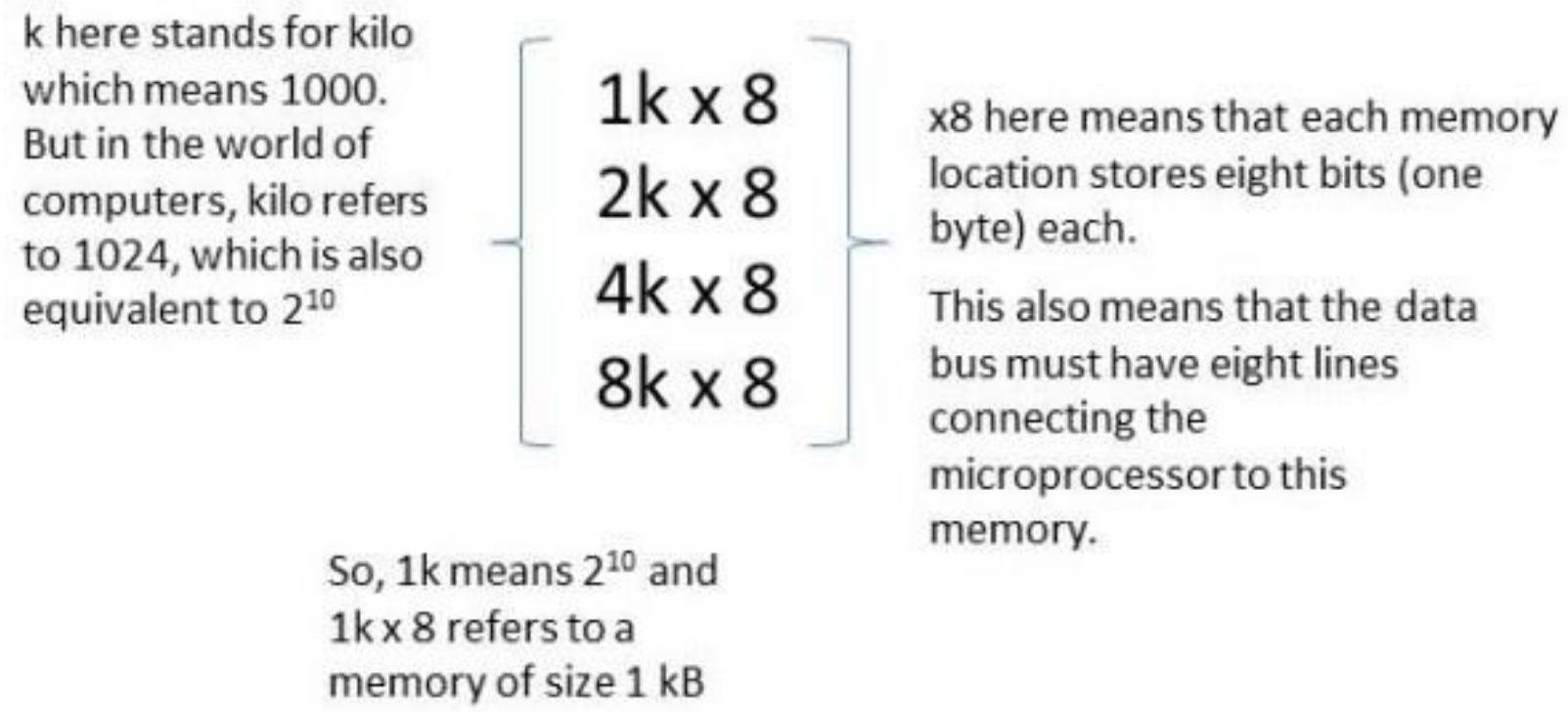


Fig. 4.13 (a) Logic diagram for RAM Fig. 4.13 (b) Logic diagram for EPROM

Memory Chips: Sizes

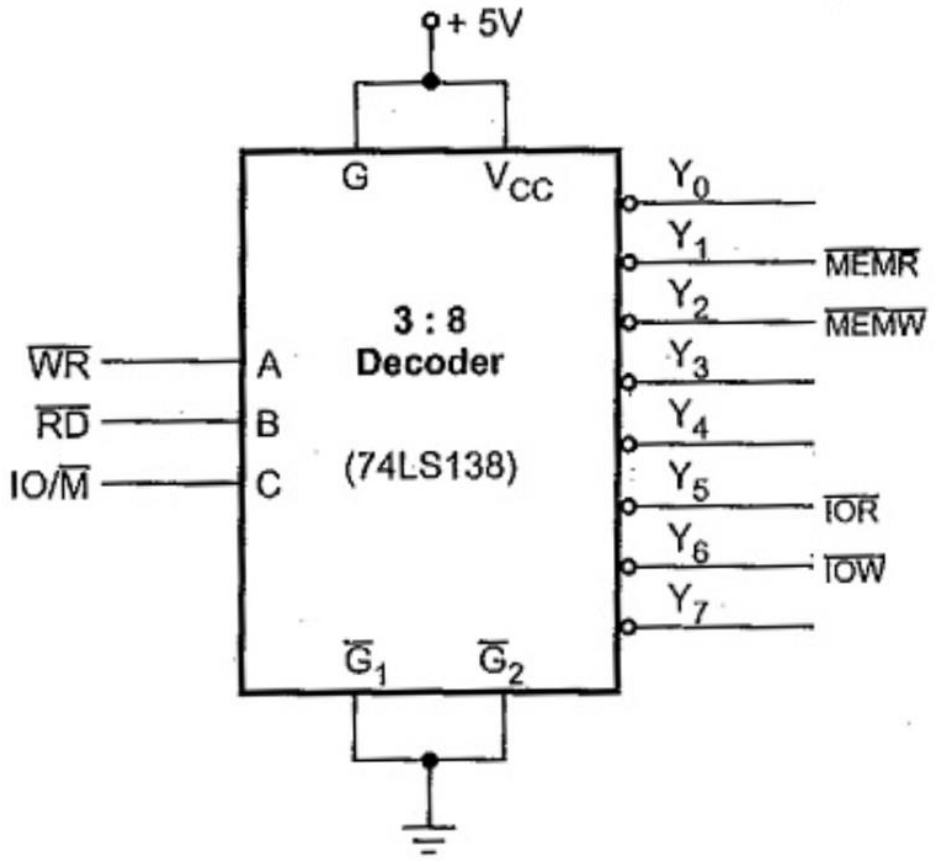
- Memory chips come in different sizes.

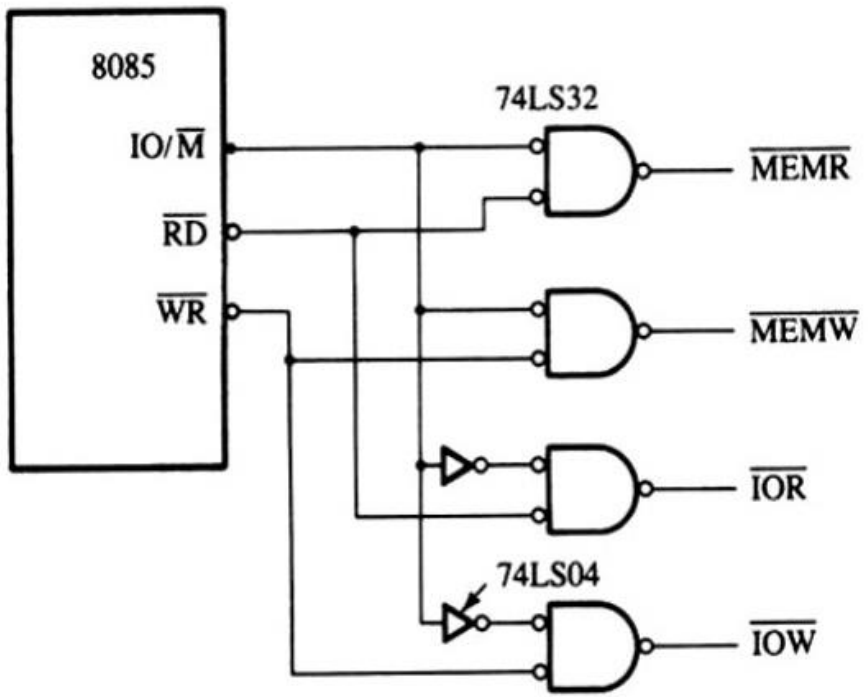


Step in Interfacing

- **Creating and Connecting Control Signals**
  - OE (Output Enable) and WR (Memory Write)**.**
- **Data Bus interfacing**
  - Connect corresponding lines (D0-D7 from 8085) to the corresponding pins (D0-D7 of the memory chip)
- **Address bus Interfacing**
  - Connecting address bus to memory and separating chip select lines
- **Generating the chip select signal**
  - Based upon chip select lines, chip select signal is generated

Creating and Connecting Control Signals





Data Bus interfacing

- There are eight lines comprising the data bus of both 8085 and the memory chips.
- The interfacing of the data bus is the simplest part.
- We just connect corresponding lines (D0-D7 from 8085) to the corresponding pins (D0-D7 of the memory chip).

Address bus Interfacing

- Lets, we have a 1kB RAM with 10 address lines.
- So, the first 10 lines of the address bus of 8085 will be connected to the corresponding address lines of the 1kB RAM.
- The remaining address lines will be used to generate the chip select (CS) signal.

Generating the chip select signal

- Using NAND Gates
- Using 3x8 decoder

The Final Circuit