Microprocessor
Assignment II

Safal Piya

Q1: Explain the term: Timing diagram, instruction cycle, machine cycle and T-states. What are the control signals used in the timing diagram of 8085 microprocessors? Explain.

⇒ **Timing diagram** Visual representation of different states of components in a fully functional system across a period of time is known as Timing diagram.

In microprocessors, those different components maybe clock, registers, control signals etc.

Instruction cycle In microprocessors, it takes multiple steps for the completion of any instruction. This means each instruction cycle consist of one or more than one operation.

A complete instruction cycle includes the fetching, decoding and execution of a instruction. And the time required for the completion of all operations represents the instruction cycle.

Machine cycle As discussed above, instruction cycle represents the time required by all operations to be completed but the machine cycle represents the time required by each operation to be completed.

Those operations may include 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-states The `X1` and `X2` pins in the 8085 generates the clock signals. The clock signal passed will be periodically high and low. And the time period between two consecutive high or low signals is known as T-states.

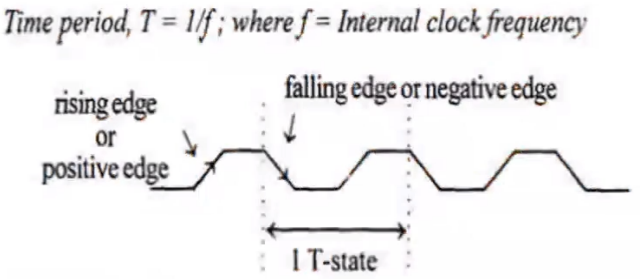

Figure 1: Clock signal

Control signals used in the timing diagram of 8085

There are three control signals used in the timing diagram of 8085. They are:

1. $\overline{\texttt{RD}}$

   This signal is used to control the `READ` operation. The overline in the `RD` represents that this signal is a active low signal i.e. $\overline{\texttt{RD}}$ signal is active during low signal.

   Thus a low signal specifies the selected memory or IO device is to be read.

2. $\overline{\texttt{WR}}$

   This signal is used to control the `WRITE` operation. The overline in the `WR` represents that this signal is a active low signal i.e. $\overline{\texttt{WR}}$ signal is active during low signal.

   Thus a low signal specifies the selected memory or IO device is to be written.

3. **ALE**
   ALE stands for "Address Latch Enable". This signal specifies whether the content in the memory location is an instruction or data.

   The lower order buffer of Program Counter (represented by $AD_7 - AD_0$) can carry both data and instruction. This data is transferred through a multiplex bus (as it carries both address and data). The contents in it is classified as data or instruction through ALE.

   Active low on ALE specifies the content is a data thus data bus is activated. And active high specifies the content is a instruction thus the lower 8-bits of the address bus is activated.

Q2: What are different machine cycles in 8085 microprocessor? Draw timing diagram of each of them and explain.

⇒ The 8085 microprocessor has 5 basic machine cycles. They are:

1. **Opcode fetch cycle (4T)**

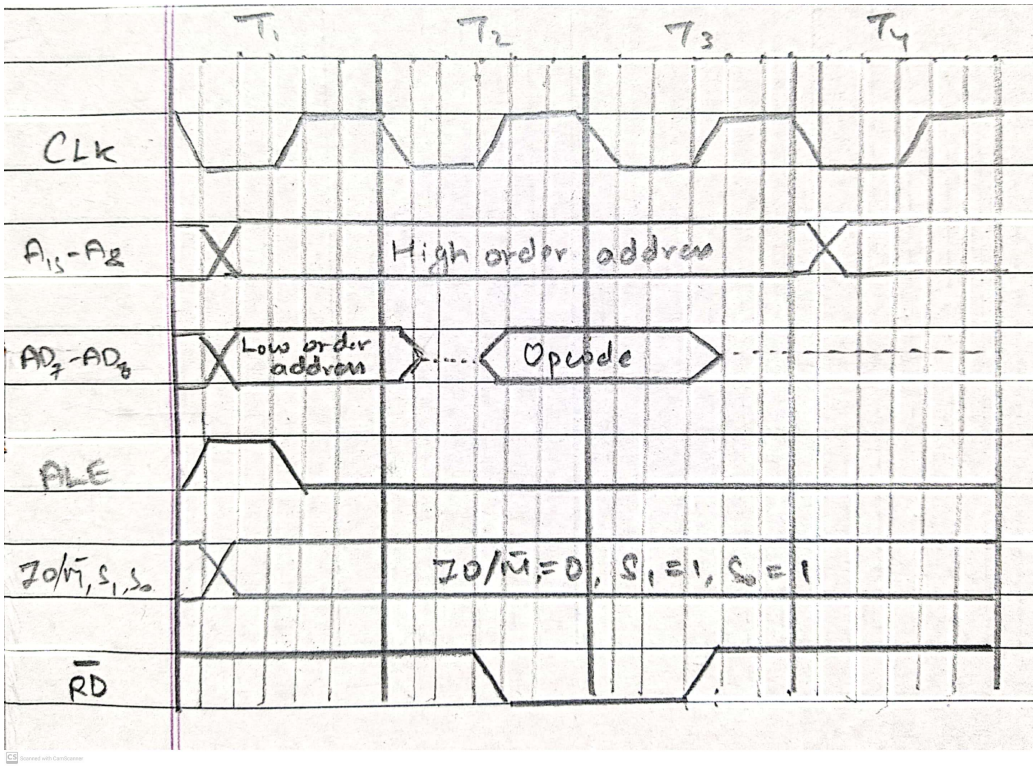

Figure 2: Opcode fetch cycle

- `X1` and `X2` sends a periodic high/low clock signal
  - The state changes with some lagging
  - Thus the change of state isn't immediate but linear
- The higher order memory address (containing the address of memory or I/O device) resides until the T_3 state
- Meanwhile the lower order memory address resides until the T_1 state only
  - When ALE signal is high, it acts as a address bus
  - But when ALE signal is low, it acts as a data bus

- ALE is high only during the T1 state
- As we have to operate on memory, the IO/$\overline{M}$ signal is low
- As we have to fetch data, both S_1 and S_0 signals are high
- Now to read the data which is available during the T2 and T3 state, the $\overline{RD}$ signal is low in those states only
- As the data fetched has to be decoded, an extra state – T4 is required.
- In the T4 state, decoding process occurs.

2. **Memory read cycle (3T)**

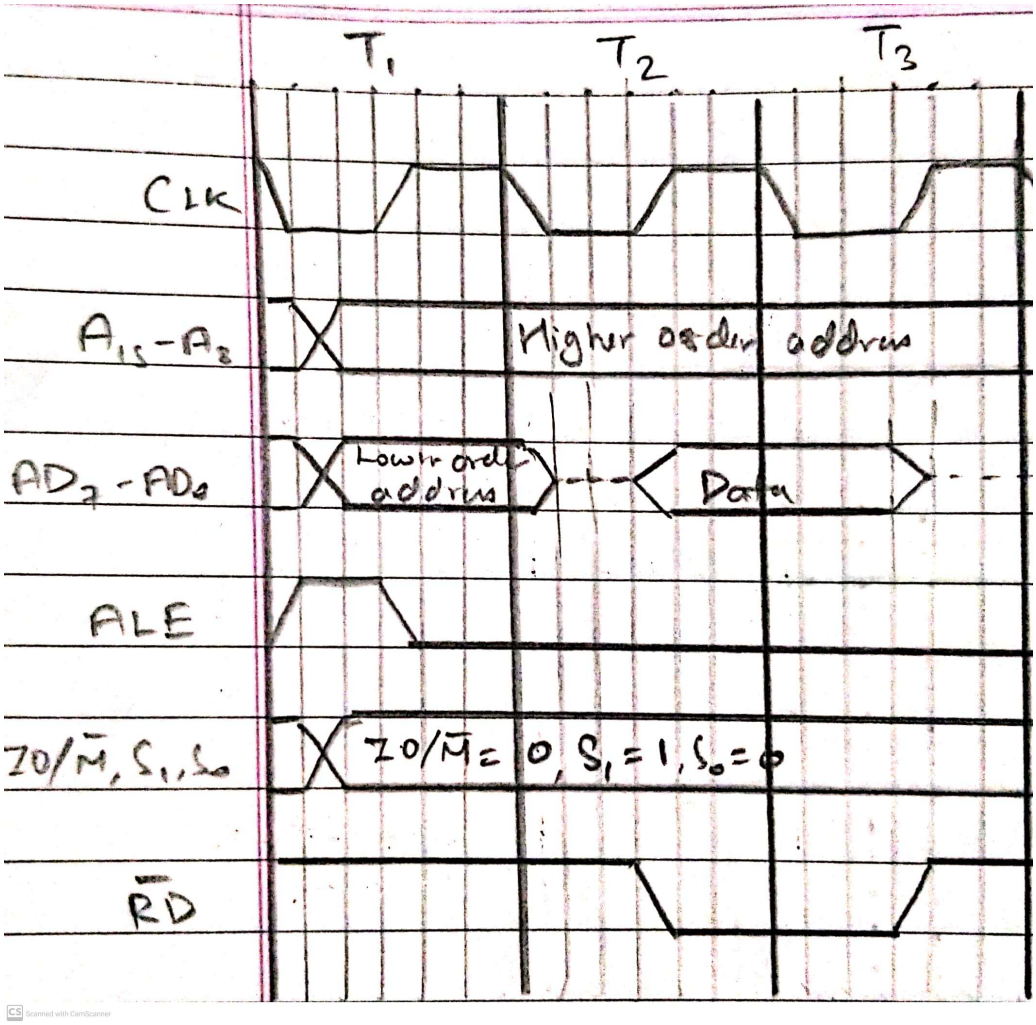

Figure 3: Memory read cycle

- X1 and X2 sends a periodic high/low clock signal
  - The state changes with some lagging
  - Thus the change of state isn't immediate but linear
- The higher order memory address (containing the address of memory or I/O device) resides until the T_3 state
- Meanwhile the lower order memory address resides until the T_1 state only
  - When ALE signal is high, it acts as a address bus
  - But when ALE signal is low, it acts as a data bus

- ALE is high only during the T1 state
- As we have to operate on memory, the IO/$\overline{M}$ signal is low
- As we have to read data, S_1 signal is high whereas S_0 is low.
- Now to read the data which is available during the T2 and T3 state, the $\overline{RD}$ signal is low in those states only
- As the data is only read (not fetched), decoding process is not required and thus no T4 state

3. **Memory write cycle (3T)**

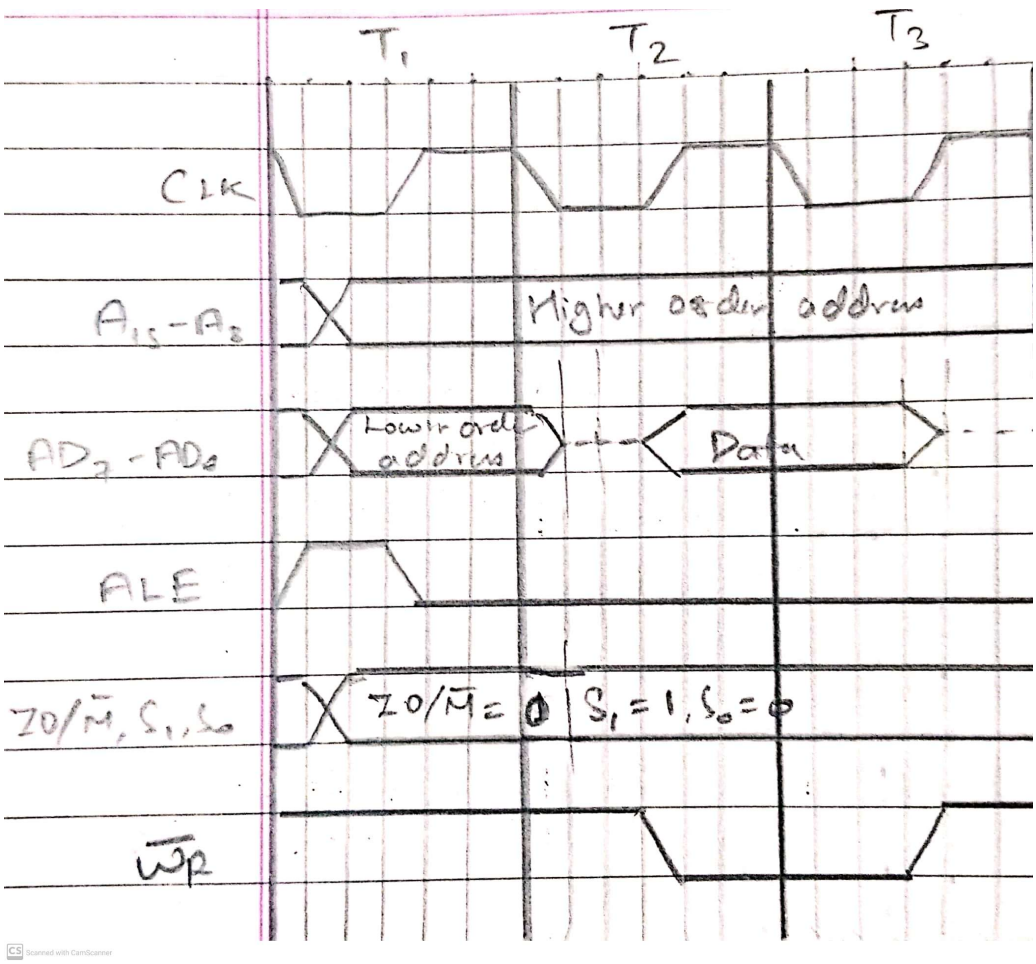

Figure 4: Memory write cycle

- X1 and X2 sends a periodic high/low clock signal
  - The state changes with some lagging
  - Thus the change of state isn't immediate but linear
- The higher order memory address (containing the address of memory or I/O device) resides until the T_3 state
- Meanwhile the lower order memory address resides until the T_1 state only
  - When ALE signal is high, it acts as a address bus
  - But when ALE signal is low, it acts as a data bus
- ALE is high only during the T1 state
- As we have to operate on memory, the IO/$\overline{M}$ signal is low

- As we have to write data, S_1 signal is low whereas S_0 is high.
- Now to write the data which is available during the T2 and T3 state, the $\overline{WR}$ signal is low in those states only
- As the data is only written (not fetched), decoding process is not required and thus no T4 state

4. **I/O read cycle (3T)**

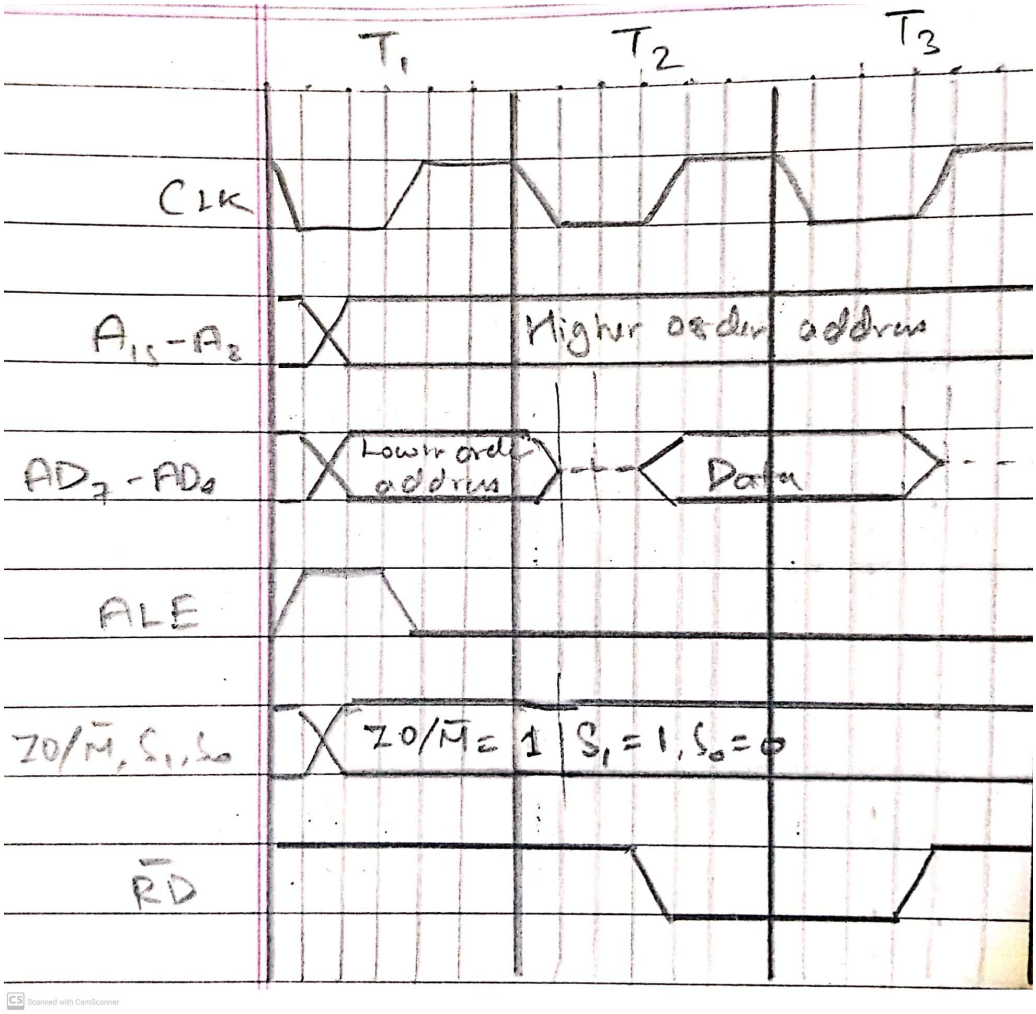

Figure 5: I/O read cycle

- `X1` and `X2` sends a periodic high/low clock signal
  - The state changes with some lagging
  - Thus the change of state isn't immediate but linear
- The higher order memory address (containing the address of memory or I/O device) resides until the T_3 state
- Meanwhile the lower order memory address resides until the T_1 state only
  - When ALE signal is high, it acts as a address bus
  - But when ALE signal is low, it acts as a data bus
- ALE is high only during the T1 state
- As we have to operate on I/O device, the $\texttt{IO/}\overline{\texttt{M}}$ signal is high

- As we have to write data, S_1 signal is high whereas S_0 is low.
- Now to read the data which is available during the T2 and T3 state, the $\overline{\texttt{RD}}$ signal is low in those states only
- As the data is only read (not fetched), decoding process is not required and thus no T4 state

5. **I/O write cycle (3T)**

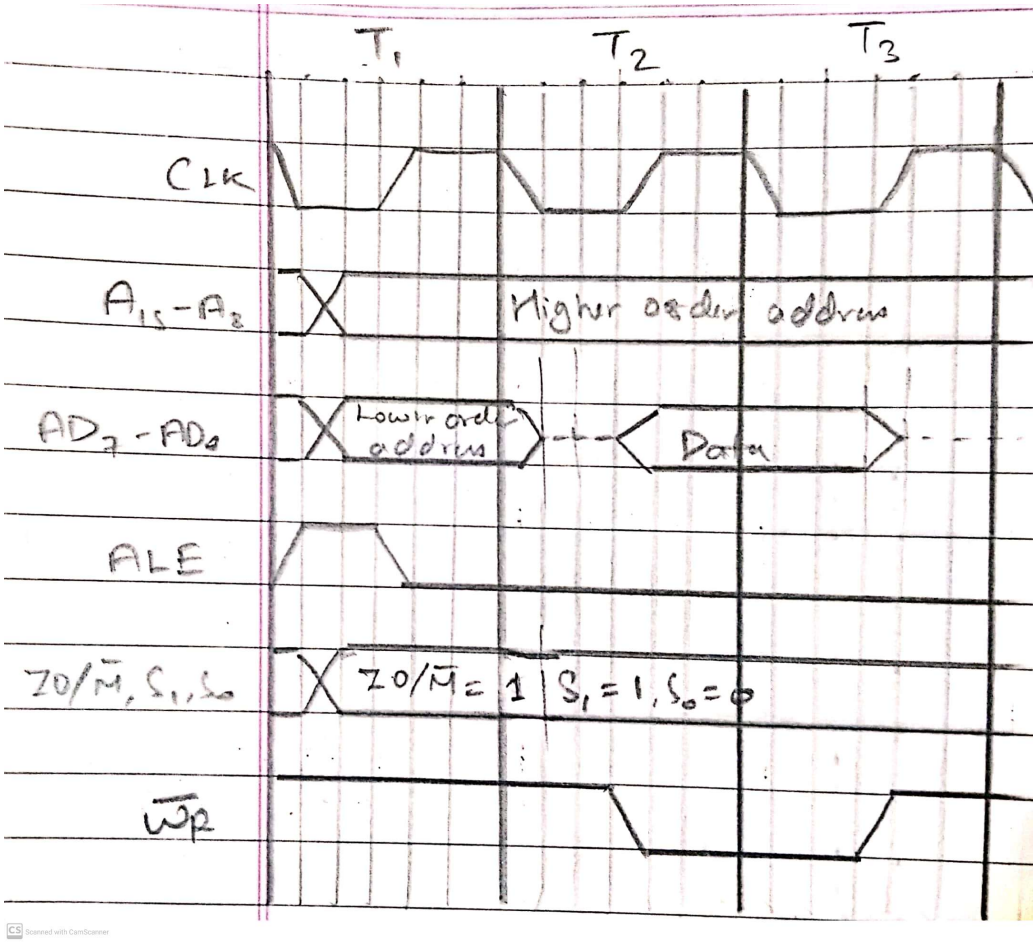

Figure 6: I/O write cycle

- X1 and X2 sends a periodic high/low clock signal
  - The state changes with some lagging
  - Thus the change of state isn't immediate but linear
- The higher order memory address (containing the address of memory or I/O device) resides until the T_3 state
- Meanwhile the lower order memory address resides until the T_1 state only
  - When ALE signal is high, it acts as a address bus
  - But when ALE signal is low, it acts as a data bus
- ALE is high only during the T1 state
- As we have to operate on I/O device, the $\texttt{IO}/\overline{\texttt{M}}$ signal is high
- As we have to write data, S_1 signal is low whereas S_0 is high.

– Now to write the data which is available during the T2 and T3 state, the $\overline{WR}$ signal is low in those states only
– As the data is only written (not fetched), decoding process is not required and thus no T4 state

Q3: Draw the timing diagram of instruction `MVI A, 01H` stored in memory location 2030H and explain it.

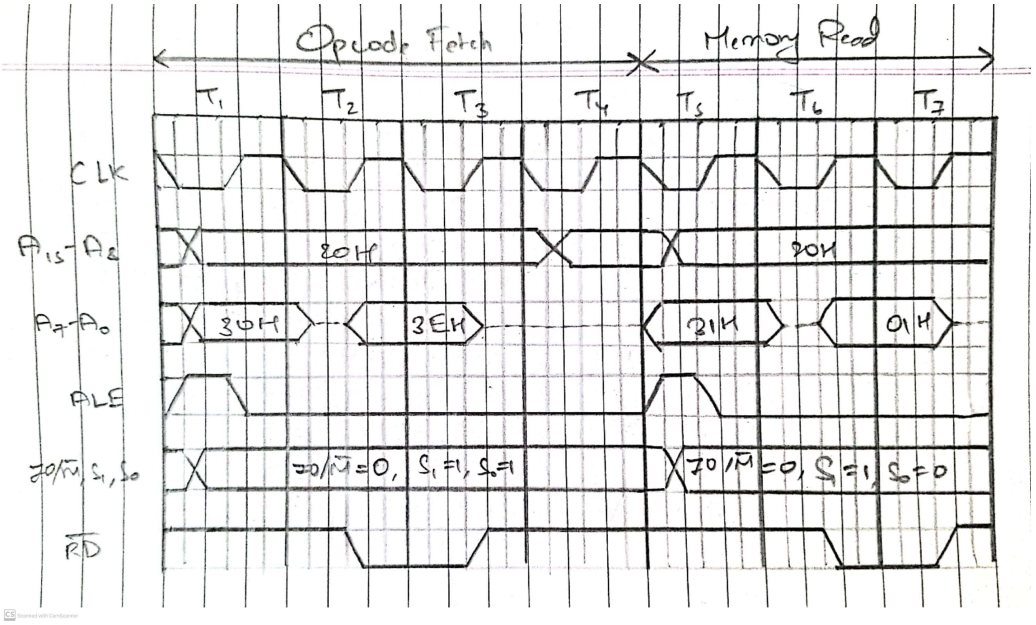

Figure 7: Timing diagram of `MVI A, 01H`

⇒ `MVI A, 01H` instruction inserts the data `01H` into the `A` register. This instruction consists of two machine cycles: Fetching of opcode and Memory read.

Fetching of opcode: It is done in 4 T states in which the first 3 T states includes getting the hex representation of the opcode and the last decodes the hex into the opcode.

Clock: X1, X2 sends a periodic high/low clock signal

Address bus: The instruction is stored in the memory location `2030H`. Thus the data on higher order address (A_{15}–A_8) is `20` and the data on lower order address (A_7–A_0) is `30`.

Data bus: Hex representation of the opcode `MVI A` is `3E`. Thus the data on D_7–D_0 is `3E`.

ALE: ALE is enabled on the first T state to enable latch and store the lower order address.

IO/$\overline{M}$: As we have to operate on memory, the IO/$\overline{M}$ signal is low.

S_1, S_0: As we have to fetch data, both S_1 and S_0 signals are high.

$\overline{RD}$: Now to read the data which is available during the T2 and T3 state, the $\overline{RD}$ signal is low in those states only

Memory read: It is done in 3 T states.

Clock: X1, X2 sends a periodic high/low clock signal

Address bus: The data is stored in the memory location `2031H` (One address higher than the opcode). Thus the data on higher order address (A_{15}–A_8) is `20` and the data on lower order address (A_7–A_0) is `31`.

Data bus: As the data to store is `01H`, data on D_7–D_0 is `01`.

ALE: ALE is enabled on the first T state to enable latch and store the lower order address.

IO/$\overline{M}$: As we have to operate on memory, the IO/$\overline{M}$ signal is low.

S_1, S_0: As we have to read data, S_1 signal is high whereas S_0 is low.

$\overline{RD}$: Now to read the data which is available during the T2 and T3 state, the $\overline{RD}$ signal is low in those states only.

Q4: Draw the timing diagram of instruction `MOV A, B` and explain it.

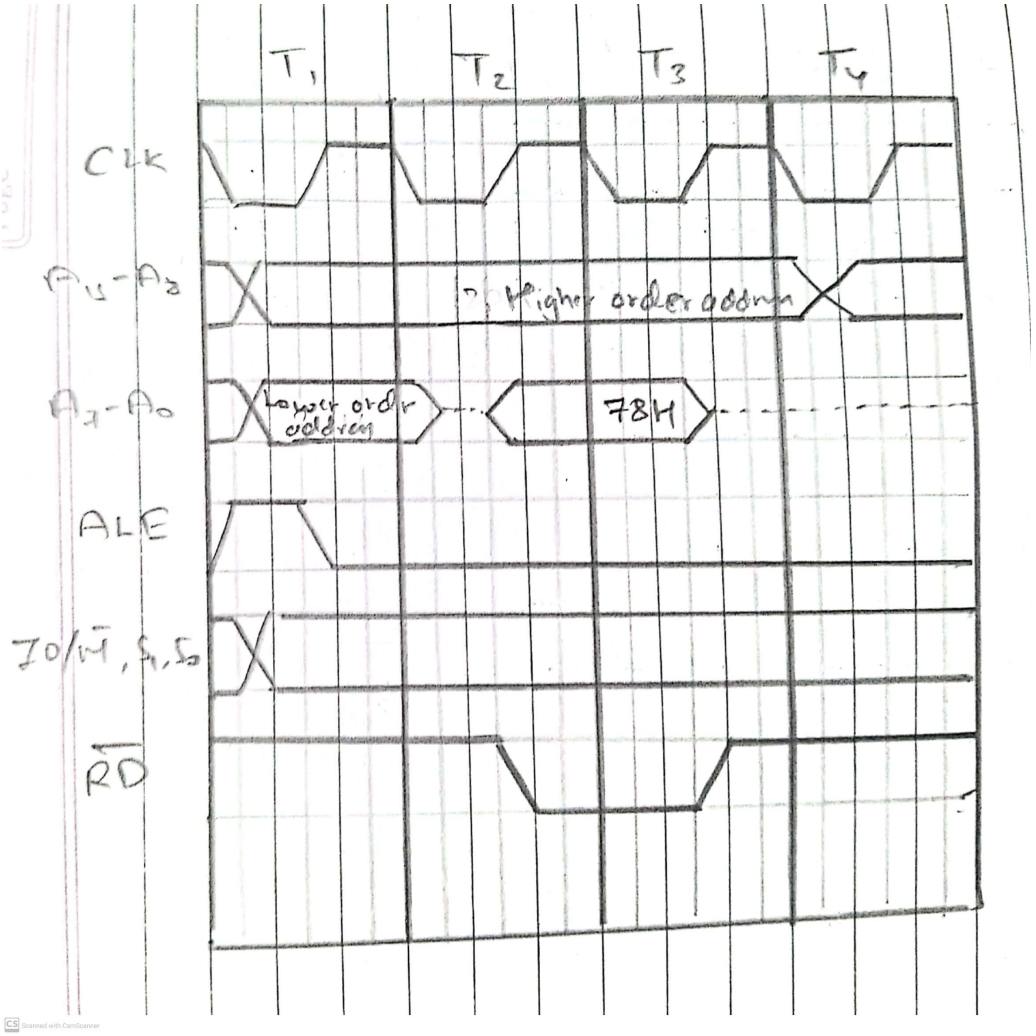

Figure 8: Timing diagram of `MOV A, B`

⇒ `MOV A, B` instruction inserts the data of `B` register into the `A` register. This instruction consists of only one machine cycle: Fetching of opcode.

Fetching of opcode: It is done in 4 T states in which the first 3 T states includes getting the hex representation of the opcode and the last decodes the hex into the opcode.

Clock: X1, X2 sends a periodic high/low clock signal

Address bus: The instruction to be fetched is actually stored in a memory location. Here no specific memory location is given but for example if the instruction is stored in 2030H, the data on higher order address (A_{15}–A_8) is 20 and the data on lower order address (A_7–A_0) is 30.

Data bus: Hex representation of the opcode MOV A, B is 78. Thus the data on D_7–D_0 is 78.

ALE: ALE is enabled on the first T state to enable latch and store the lower order address.

IO/$\overline{M}$: As we have to operate on memory, the IO/$\overline{M}$ signal is low.

S_1, S_0: As we have to fetch data, both S_1 and S_0 signals are high.

$\overline{RD}$: Now to read the data which is available during the T2 and T3 state, the $\overline{RD}$ signal is low in those states only.

Q5: Draw the timing diagram of instruction STA 1001H and explain it.

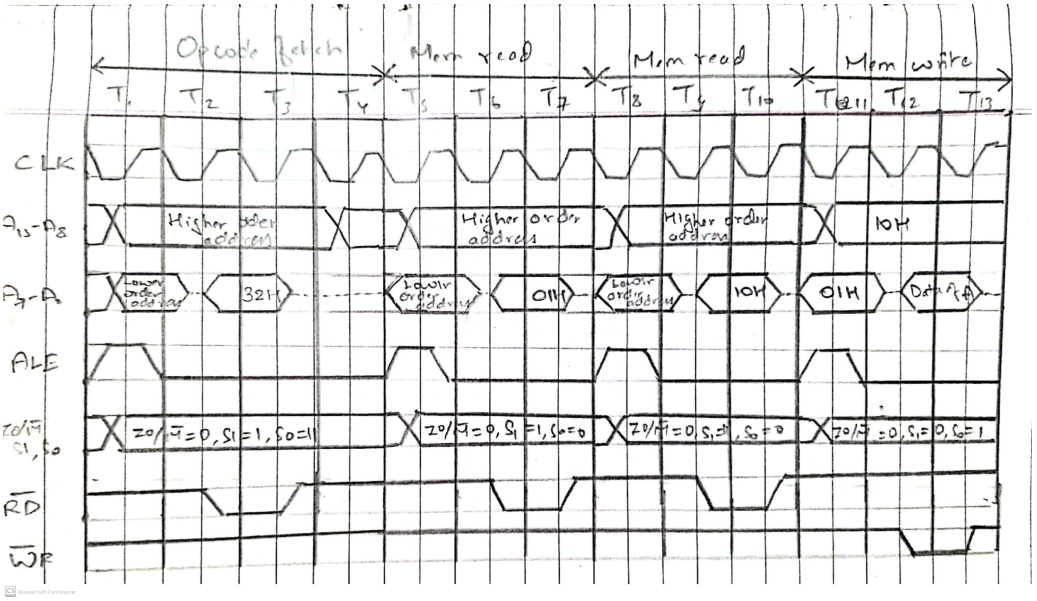

Figure 9: Timing diagram of STA 1001H

⇒ STA 1001H instruction stores the data of A register into the memory address 1001H. This instruction consists of four machine cycles: Fetching of opcode, Memory read 2 times and Memory write.

Fetching of opcode: It is done in 4 T states in which the first 3 T states includes getting the hex representation of the opcode and the last decodes the hex into the opcode.

Clock: X1, X2 sends a periodic high/low clock signal

Address bus: The instruction to be fetched is actually stored in a memory location. Here no specific memory location is given but for example if the instruction is stored in 2030H, the data on higher order address (A_{15}–A_8) is 20 and the data on lower order address (A_7–A_0) is 30.

Data bus: Hex representation of the opcode STA is 32. Thus the data on D_7–D_0 is 32.

ALE: ALE is enabled on the first T state to enable latch and store the lower order address.

IO/$\overline{M}$: As we have to operate on memory, the IO/$\overline{M}$ signal is low.

S_1, S_0: As we have to fetch data, both S_1 and S_0 signals are high.

$\overline{RD}$: Now to read the data which is available during the T2 and T3 state, the $\overline{RD}$ signal is low in those states only.

Memory read #1: It is done in 3 T states.

Clock: X1, X2 sends a periodic high/low clock signal

Address bus: Suppose the opcode is stored in the memory location 2030H, then the data is stored in the memory location 2031H (One address higher than the opcode). Thus the data on higher order address (A_{15}–A_8) is 20 and the data on lower order address (A_7–A_0) is 31.

Data bus: Here, the memory location to store is taken as data (i.e. 1001H is taken as data). The data is of 2 bytes (16-bits). But the data bus can carry only 1 byte (8-bit). Thus two machine cycles for memory read are required among which the first memory read machine cycle will carry the lower data bit. Meaning the value of D_7–D_0 is 01.

ALE: ALE is enabled on the first T state to enable latch and store the lower order address.

IO/$\overline{M}$: As we have to operate on memory, the IO/$\overline{M}$ signal is low.

S_1, S_0: As we have to read data, S_1 signal is high whereas S_0 is low.

$\overline{RD}$: Now to read the data which is available during the T2 and T3 state, the $\overline{RD}$ signal is low in those states only.

Memory read #2: It is done in 3 T states.

Clock: X1, X2 sends a periodic high/low clock signal

Address bus: Suppose the opcode is stored in the memory location 2030H, then the data is stored in the memory location 2031H (One address higher than the opcode). Thus the data on higher order address (A_{15}–A_8) is 20 and the data on lower order address (A_7–A_0) is 31.

Data bus: Here, the second memory read machine cycle will carry the higher data bit. Meaning the value of D_7–D_0 is 10.

ALE: ALE is enabled on the first T state to enable latch and store the lower order address.

IO/$\overline{M}$: As we have to operate on memory, the IO/$\overline{M}$ signal is low.

S_1, S_0: As we have to read data, S_1 signal is high whereas S_0 is low.

$\overline{RD}$: Now to read the data which is available during the T2 and T3 state, the $\overline{RD}$ signal is low in those states only.

Memory write: It is done in 3 T states.

Clock: X1, X2 sends a periodic high/low clock signal

Address bus: The data of A register is to be saved in memory location 1001H. Thus the data on higher order address (A_{15}–A_8) is 10 and the data on lower order address (A_7–A_0) is 01.

Data bus: The data of A register will be carried in the data bus.

ALE: ALE is enabled on the first T state to enable latch and store the lower order address.

IO/$\overline{M}$: As we have to operate on memory, the IO/$\overline{M}$ signal is low.

S_1, S_0: As we have to write data, S_1 signal is low whereas S_0 is high.

$\overline{RD}$: Now to read the data which is available during the T2 and T3 state, the $\overline{RD}$ signal is low in those states only.

Q6: Draw the timing diagram of instruction `LDA 0101H` and explain it.

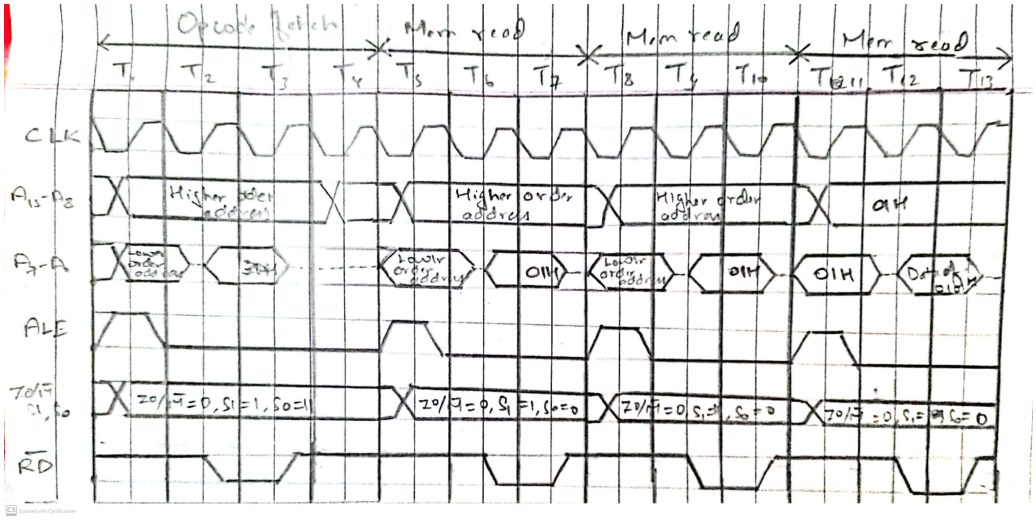

Figure 10: Timing diagram of `LDA 0101H`

⇒ `LDA 0101H` instruction stores the data on memory location `0101H` into the `A` register. This instruction consists of four machine cycles: Fetching of opcode, Memory read 3 times.

Fetching of opcode: It is done in 4 T states in which the first 3 T states includes getting the hex representation of the opcode and the last decodes the hex into the opcode.

Clock: X1, X2 sends a periodic high/low clock signal

Address bus: The instruction to be fetched is actually stored in a memory location. Here no specific memory location is given but for example if the instruction is stored in `2030H`, the data on higher order address (A_{15}–A_8) is `20` and the data on lower order address (A_7–A_0) is `30`.

Data bus: Hex representation of the opcode `LDA` is `3A`. Thus the data on D_7–D_0 is `3A`.

ALE: ALE is enabled on the first T state to enable latch and store the lower order address.

IO/$\overline{M}$: As we have to operate on memory, the IO/$\overline{M}$ signal is low.

S_1, S_0: As we have to fetch data, both S_1 and S_0 signals are high.

$\overline{RD}$: Now to read the data which is available during the T2 and T3 state, the $\overline{RD}$ signal is low in those states only.

Memory read #1: It is done in 3 T states.

Clock: X1, X2 sends a periodic high/low clock signal

Address bus: Suppose the opcode is stored in the memory location 2030H, then the data is stored in the memory location 2031H (One address higher than the opcode). Thus the data on higher order address (A_{15}–A_8) is 20 and the data on lower order address (A_7–A_0) is 31.

Data bus: Here, the memory location to get the data from is taken as data (i.e. 0101H is taken as data). The data is of 2 bytes (16-bits). But the data bus can carry only 1 byte (8-bit). Thus two machine cycles for memory read are required among which the first memory read machine cycle will carry the lower data bit. Meaning the value of D_7–D_0 is 01.

ALE: ALE is enabled on the first T state to enable latch and store the lower order address.

IO/$\overline{M}$: As we have to operate on memory, the IO/$\overline{M}$ signal is low.

S_1, S_0: As we have to read data, S_1 signal is high whereas S_0 is low.

$\overline{RD}$: Now to read the data which is available during the T2 and T3 state, the $\overline{RD}$ signal is low in those states only.

Memory read #2: It is done in 3 T states.

Clock: X1, X2 sends a periodic high/low clock signal

Address bus: Suppose the opcode is stored in the memory location 2030H, then the data is stored in the memory location 2031H (One address higher than the opcode). Thus the data on higher order address (A_{15}–A_8) is 20 and the data on lower order address (A_7–A_0) is 31.

Data bus: Here, the second memory read machine cycle will carry the higher data bit. Meaning the value of D_7–D_0 is 01.

ALE: ALE is enabled on the first T state to enable latch and store the lower order address.

IO/$\overline{M}$: As we have to operate on memory, the IO/$\overline{M}$ signal is low.

S_1, S_0: As we have to read data, S_1 signal is high whereas S_0 is low.

$\overline{RD}$: Now to read the data which is available during the T2 and T3 state, the $\overline{RD}$ signal is low in those states only.

Memory read #3: It is done in 3 T states.

Clock: X1, X2 sends a periodic high/low clock signal

Address bus: The data on A register is to be saved from memory location 0101H. Thus the data on higher order address (A_{15}–A_8) is 01 and the data on lower order address (A_7–A_0) is 01.

Data bus: The data on the memory location 0101H will be carried in the data bus.

ALE: ALE is enabled on the first T state to enable latch and store the lower order address.

IO/$\overline{M}$: As we have to operate on memory, the IO/$\overline{M}$ signal is low.

S_1, S_0: As we have to read data, S_1 signal is high whereas S_0 is low.

$\overline{RD}$: Now to read the data which is available during the T2 and T3 state, the $\overline{RD}$ signal is low in those states only.

Q7: Draw the timing diagram of instruction `ADD B` and explain it.

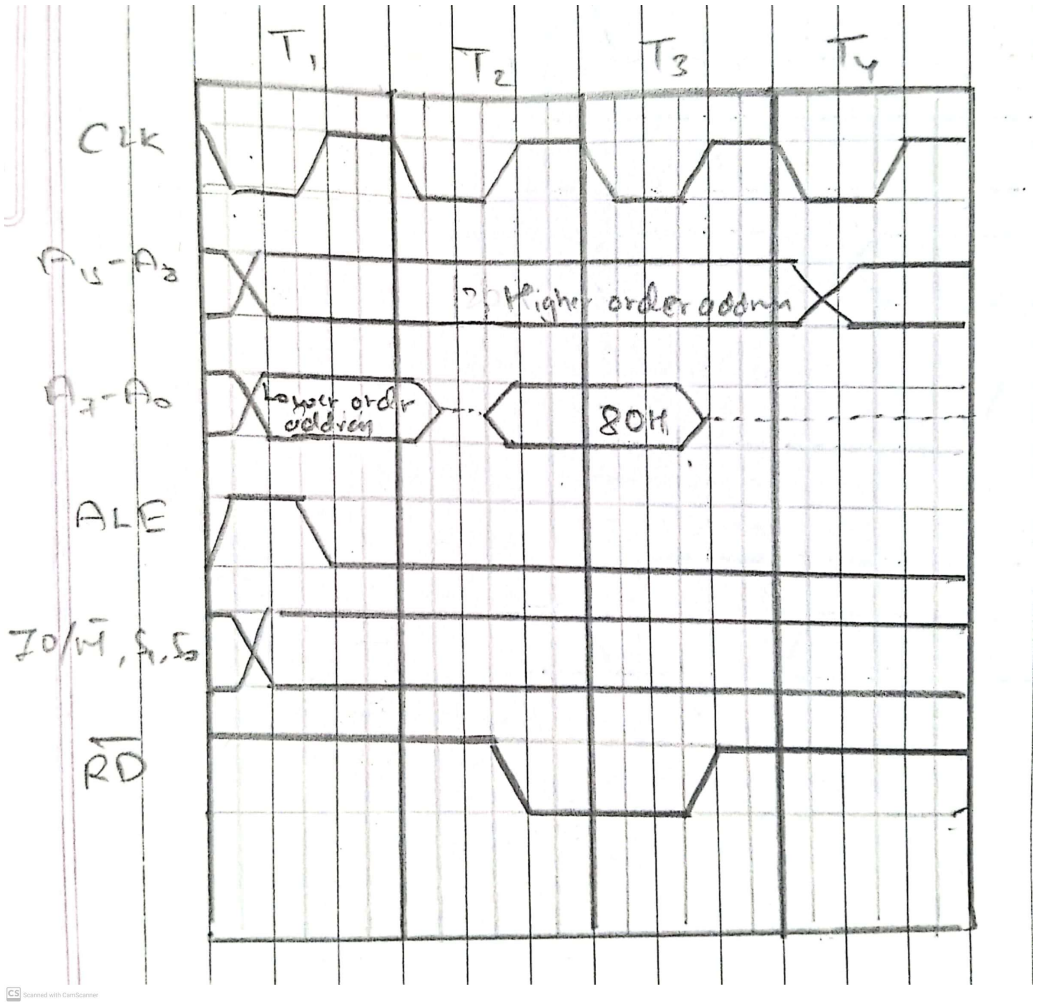

Figure 11: Timing diagram of `ADD B`

⇒ `ADD B` instruction adds the data of `A` register with the data of `B` register and store it back in `A`. This instruction consists of only one machine cycle: Fetching of opcode.

Fetching of opcode: It is done in 4 T states in which the first 3 T states includes getting the hex representation of the opcode and the last decodes the hex into the opcode.

Clock: X1, X2 sends a periodic high/low clock signal

Address bus: The instruction to be fetched is actually stored in a memory location. Here no specific memory location is given but for example if the instruction is stored in `2030H`, the data on higher order address (A_{15}–A_8) is `20` and the data on lower order address (A_7–A_0) is `30`.

Data bus: Hex representation of the opcode `ADD B` is `80`. Thus the data on D_7–D_0 is `80`.

ALE: ALE is enabled on the first T state to enable latch and store the lower order address.

IO/$\overline{M}$: As we have to operate on memory, the IO/$\overline{M}$ signal is low.

S_1, S_0: As we have to fetch data, both S_1 and S_0 signals are high.

$\overline{RD}$: Now to read the data which is available during the T2 and T3 state, the $\overline{RD}$ signal is low in those states only.

Q8: Draw the timing diagram of instruction `INR C` and explain it

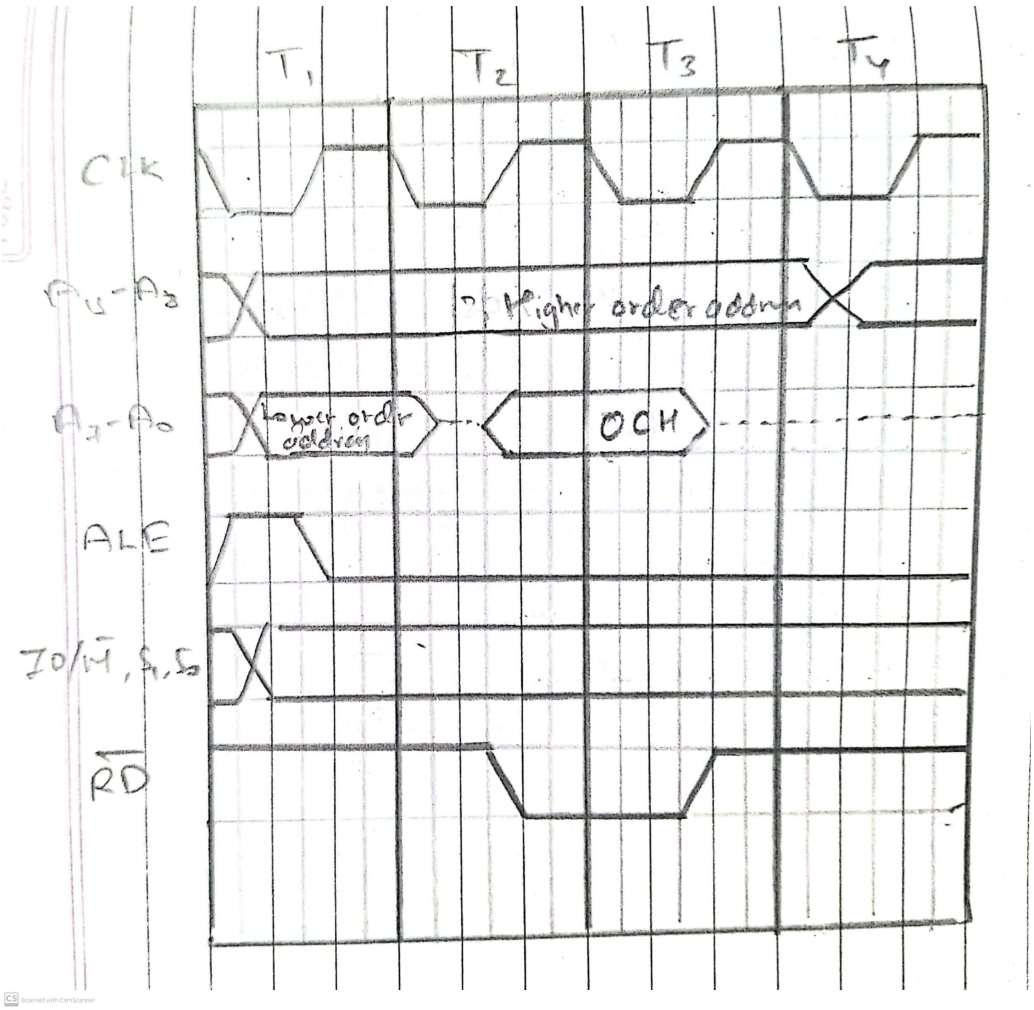

Figure 12: Timing diagram of `INR C`

⇒ `INR C` instruction increases the data of `C` register by one. This instruction consists of only one machine cycle: Fetching of opcode.

Fetching of opcode: It is done in 4 T states in which the first 3 T states includes getting the hex representation of the opcode and the last decodes the hex into the opcode.

Clock: X1, X2 sends a periodic high/low clock signal

Address bus: The instruction to be fetched is actually stored in a memory location. Here no specific memory location is given but for example if the instruction is stored in `2030H`, the data on higher order address (A_{15}–A_8) is `20` and the data on lower order address (A_7–A_0) is `30`. `20` and the data on lower order address (A_7–A_0) is `31`.

Data bus: Hex representation of the opcode `INR C` is `0C`. Thus the data on D_7–D_0 is `0C`.

ALE: ALE is enabled on the first T state to enable latch and store the lower order address.

IO/$\overline{M}$: As we have to operate on memory, the IO/$\overline{M}$ signal is low.

S_1, S_0: As we have to fetch data, both S_1 and S_0 signals are high.

$\overline{RD}$: Now to read the data which is available during the T2 and T3 state, the $\overline{RD}$ signal is low in those states only

Q10: As we know that the address bit on line AD_7–AD_0 remains only for machine cycle T1 but we need it till machine cycle T3. During machine cycle T2 and T3, line AD_7–AD_0 acts as data bus. How does the lower address bit on the line AD_7–AD_0 made available till T3? What is the mechanism behind it? Explain it.

$\Rightarrow$ The lower address bit on the line AD_7–AD_0 are made available till T3 through the use of 8-bit latch.

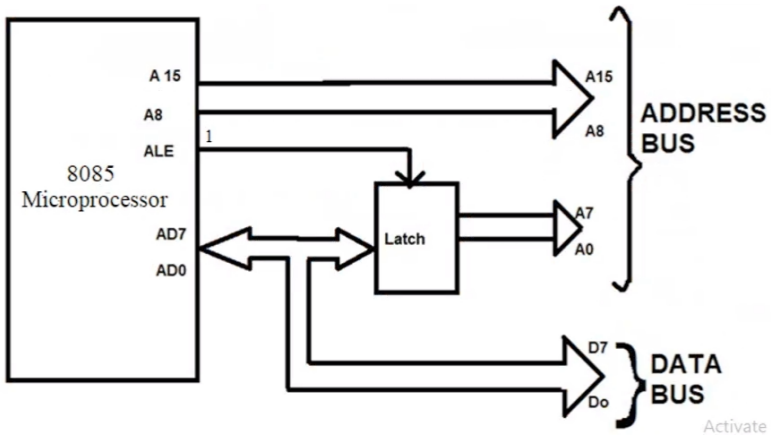

Figure 13: Multiplexing and Demultiplexing of Address-Data bus in 8085

8085 saves pins by multiplexing address bus lines A_7 – A_0 and data bus D_7 – D_0 onto its pins AD_7 – AD_0. But external memories and circuitry will need the data and address as separate signals so the 8085 has an Address Latch Enable (ALE) pin to control a latch that will demultiplex them.

In T1 state, the ALE is high and AD_7 – AD_0 drive out address lines A_7 – A_0. Also the latch is enabled in which the address bits from the address lines are flown into. After T1, the ALE will go low, latch is disabled and AD_7 – AD_0 is now used as a bidirectional data bus.

Thus the lower address bits are stored in the latch even though the lines is acting as a data bus and the data is made available till T3 state.