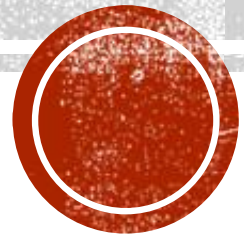
INSTRUCTION AND DATA FLOW



- 8086 allows the user to define memory areas for storing program and data
- The program memory can be accessed by using CS register and the data memory can be accessed by using DS, ES and SS registers
- The program instructions are stored in program memory which is an external device
- To execute a program in 8086, the base address and offset address of the first instruction of the program should be loaded in CS register and IP respectively

- The 8086 computes the 20-bit physical address of the program instruction by multiplying the contents of CS register by 16_{10} and adding it to the content of IP.
- The 20 bit physical address is given out on the address bus
- Then RD' is asserted low
- Also other control signals necessary for program memory read operation are asserted
- The IP is incremented by two to point next instruction or next word of the same instruction

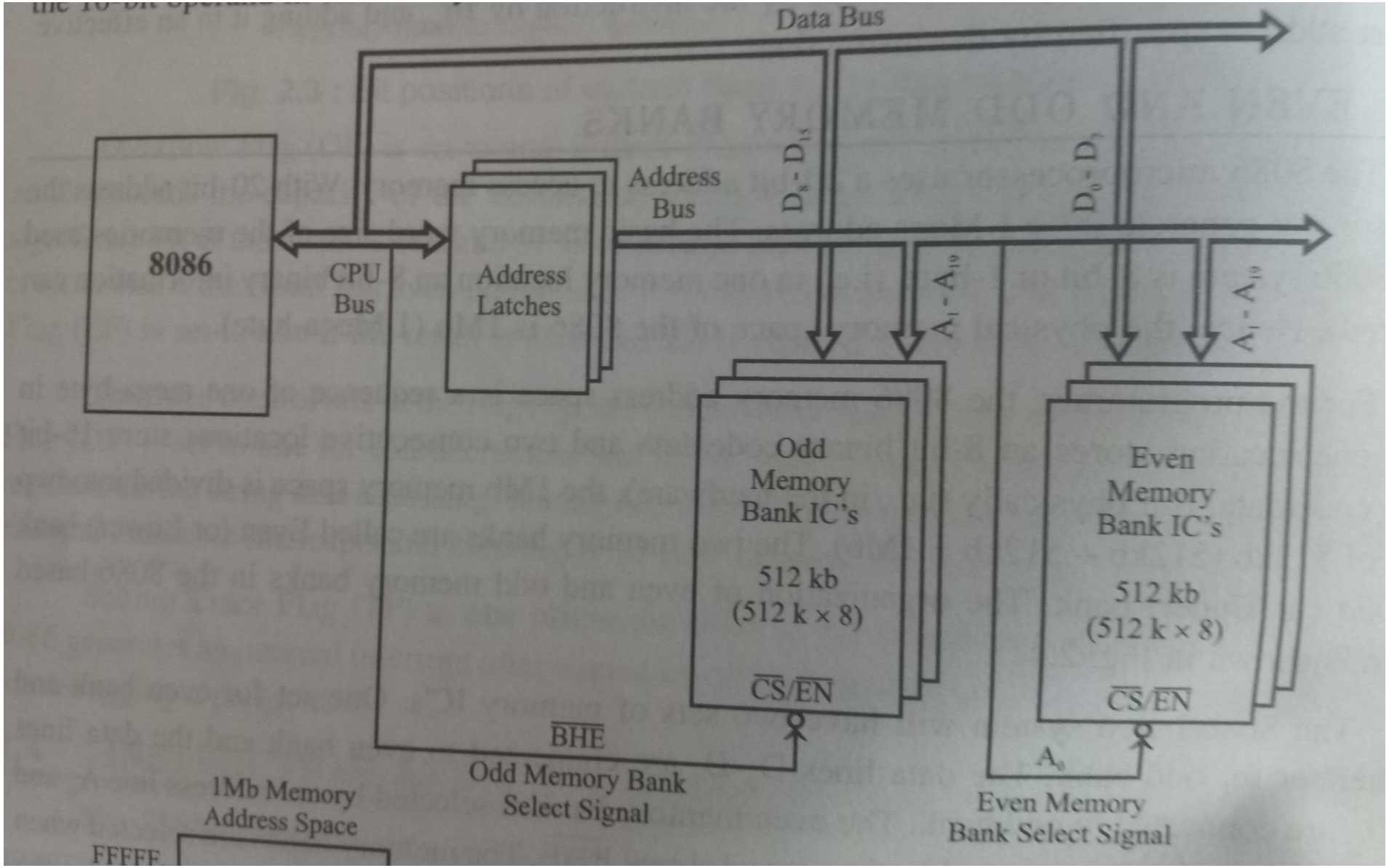
- The address and control signals enable the memory to output one word of program memory on the data bus.
- After a predefined time, the RD' is asserted high and this instant the content of the data bus is latched into the two empty locations of instruction queue.
- Then BIU starts fetching the next word of the program code as explained above.
- The BIU keeps on fetching the program codes, word by word from consecutive memory locations whenever two locations of queue are empty.

- When a branch instruction is encountered, the queue is emptied and then filled with program codes from the new address loaded in CS and IP by the branch instruction
- The EU reads the program instructions from queue, decodes and executes them one by one.
- If the execution of an instruction requires data from memory, then BIU is interrupted to read data in memory.
- When BIU is interrupted, it completes the fetching of current instruction word and then starts reading/writing the data generating a 20 bit data memory address

EVEN AND ODD MEMORY BANKS

- 8086 uses 20 bit address, so it can generate 2^{20} = Mega address
- Hence physical memory space is 1MB and one location stores an 8 bit binary code/data, two consecutive locations store 16 bit binary code/data
- Physically 1MB space is divided into two banks of 512 kb
- The two memory banks are called Even (or lower) bank and odd (or upper) bank

ORGANISATION OF BANKS



- We have two sets of memory ICs, Even and ODD
- The data lines D_0 - D_7 are connected to even bank and a data lines D_8 – D_{15} are connected to Odd bank
- The even memory bank is selected by the address line A_0 and the odd memory bank is selected by the control signal BHE'
- Memory banks are selected when these signals are low

The status of A_0 and BHE for byte and word

TABLE - 2.8 : STATUS OF A_0 AND $\overline{BHE}$ DURING MEMORY ACCESS

Memory bank	Operand type	Status of		Data lines used for memory access	No. of bus cycle
		A_0	$\overline{BHE}$		
Even	Byte	0	1	$D_0 - D_7$	One
Odd	Byte	1	0	$D_8 - D_{15}$	One
Even	Word	0	0	$D_0 - D_{15}$	One
Odd	Word	1	0	$D_8 - D_{15}$	First cycle
		0	1	$D_0 - D_7$	Second cycle

BUS CYCLE AND TIMING DIAGRAM

- The basic operations performed by the CPU bus are called bus cycles
- Bus cycles can be classified as follows
  - Memory read cycle (Four T states)
  - Memory write cycle (Four T states)
  - IO read cycle (Four T states)
  - IO Write cycle (Four T states)
  - Interrupt Acknowledge Cycle (EightT states)

- The processor takes a definite time to perform a bus cycle
- The time taken to perform a bus cycle are specified in terms of T States
- The time duration of one T-state is equal to one time period of the internal clock of the processor
- The T state starts in the middle of falling edge of the clock signal

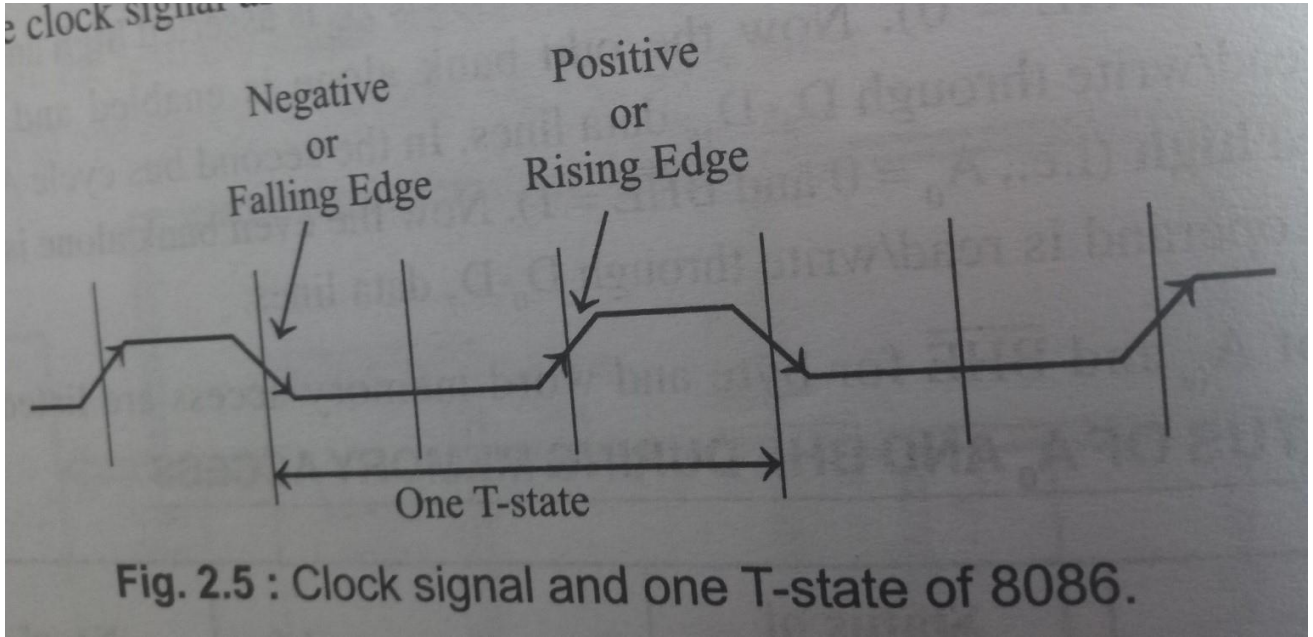
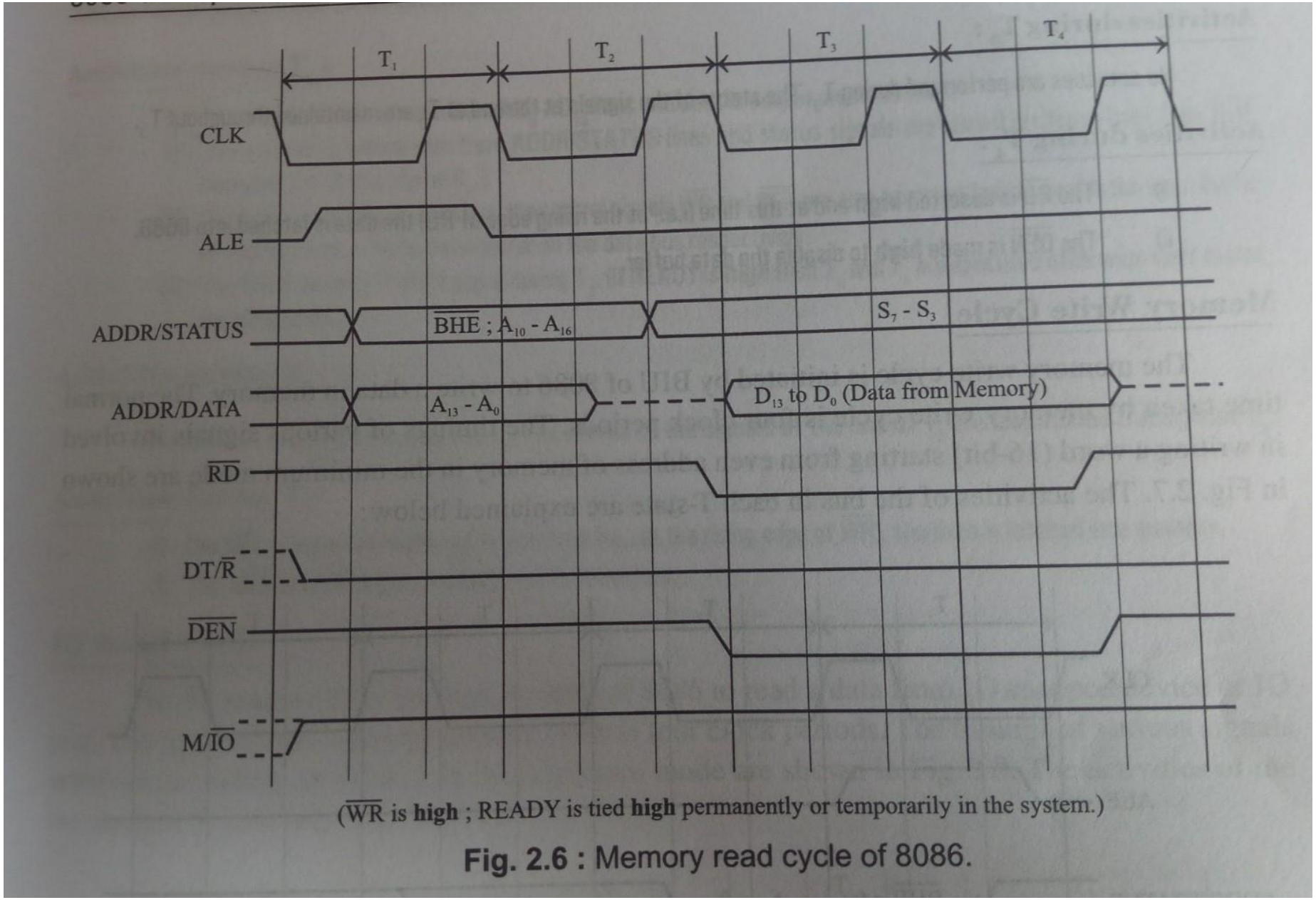


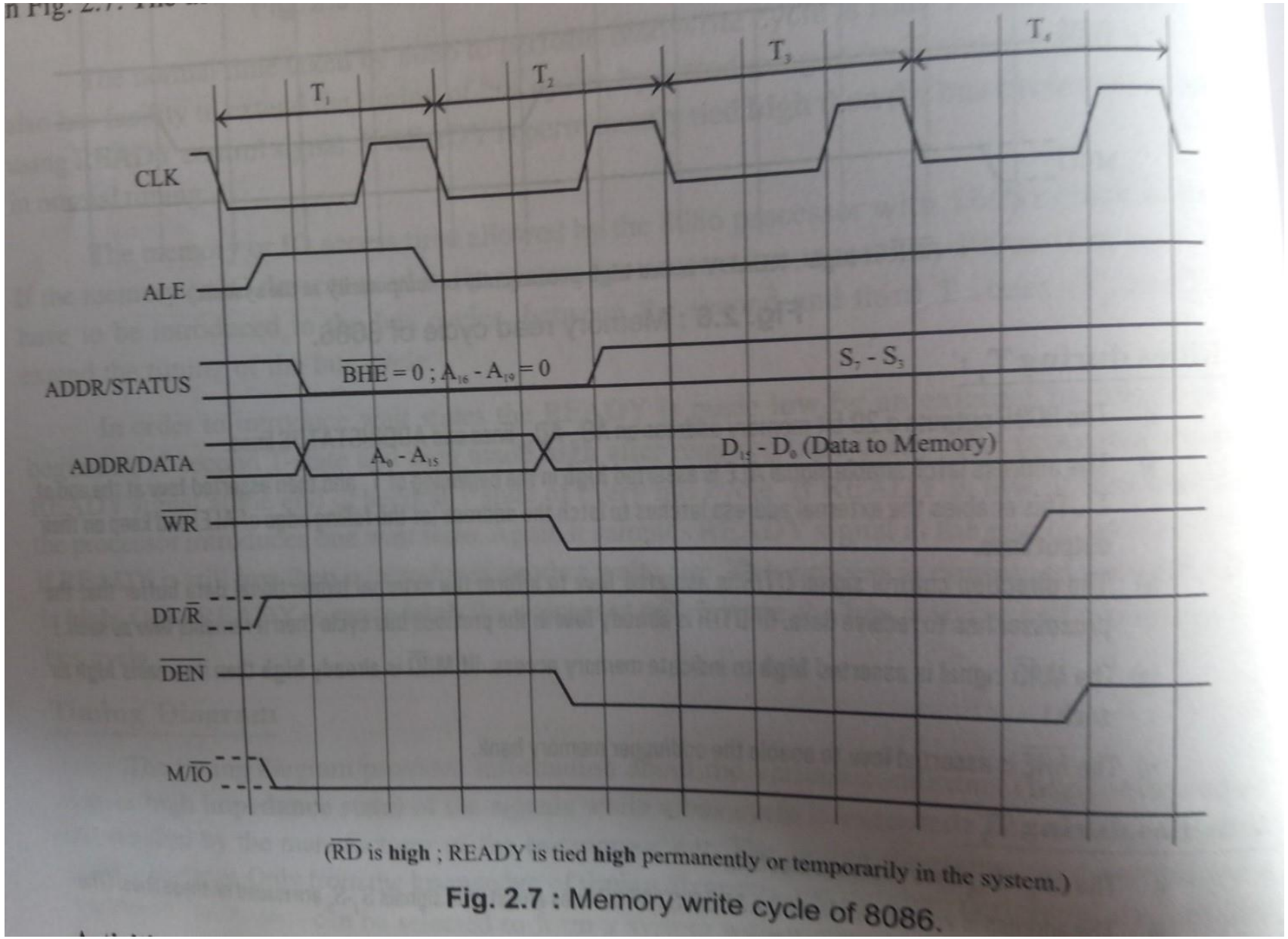
Fig. 2.5 : Clock signal and one T-state of 8086.

- The normal time taken by 8086 to perform read/write cycle is four T states
- The processor also has facility to extend the timing of bus cycles by introducing extra T states called wait states using READY control signal
- **TIMING DIAGRAM**
- Provides information about various conditions of signals while a bus cycle is executed
- It is supplied by the manufacturer and are essential for a system designer.
- From this knowledge, the matched peripheral devices like memory, ports etc can be selected



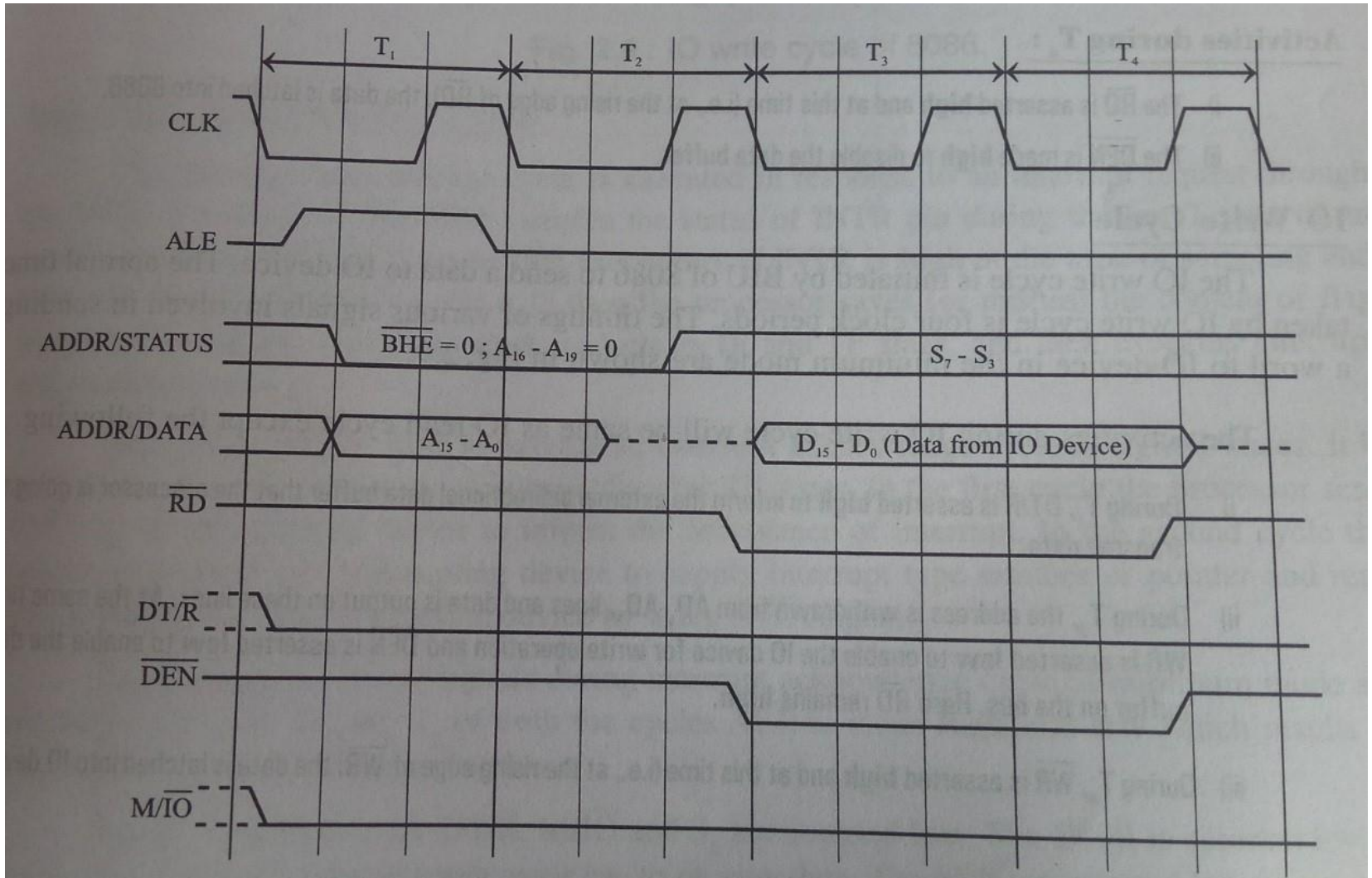
($\overline{WR}$ is **high** ; READY is tied **high** permanently or temporarily in the system.)

Fig. 2.6 : Memory read cycle of 8086.



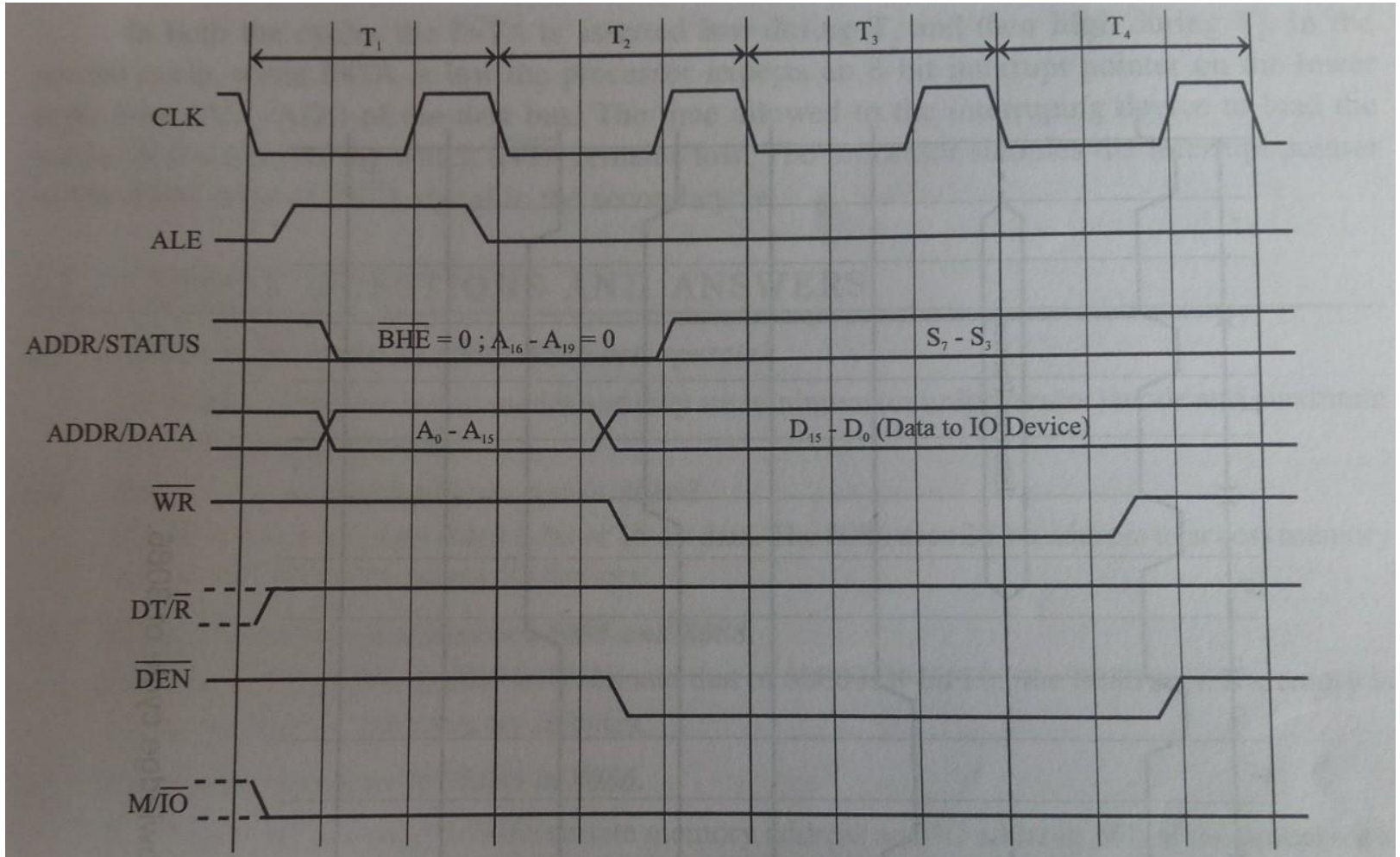
(RD is **high** ; READY is tied **high** permanently or temporarily in the system.)

Fig. 2.7 : Memory write cycle of 8086.



($\overline{WR}$ is **high** ; READY is tied **high** permanently or temporarily in the system.)

Fig. 2.8 : IO read cycle of 8086.



(RD is **high** ; READY is tied **high** permanently or temporarily in the system.)

Fig. 2.9 : IO write cycle of 8086.

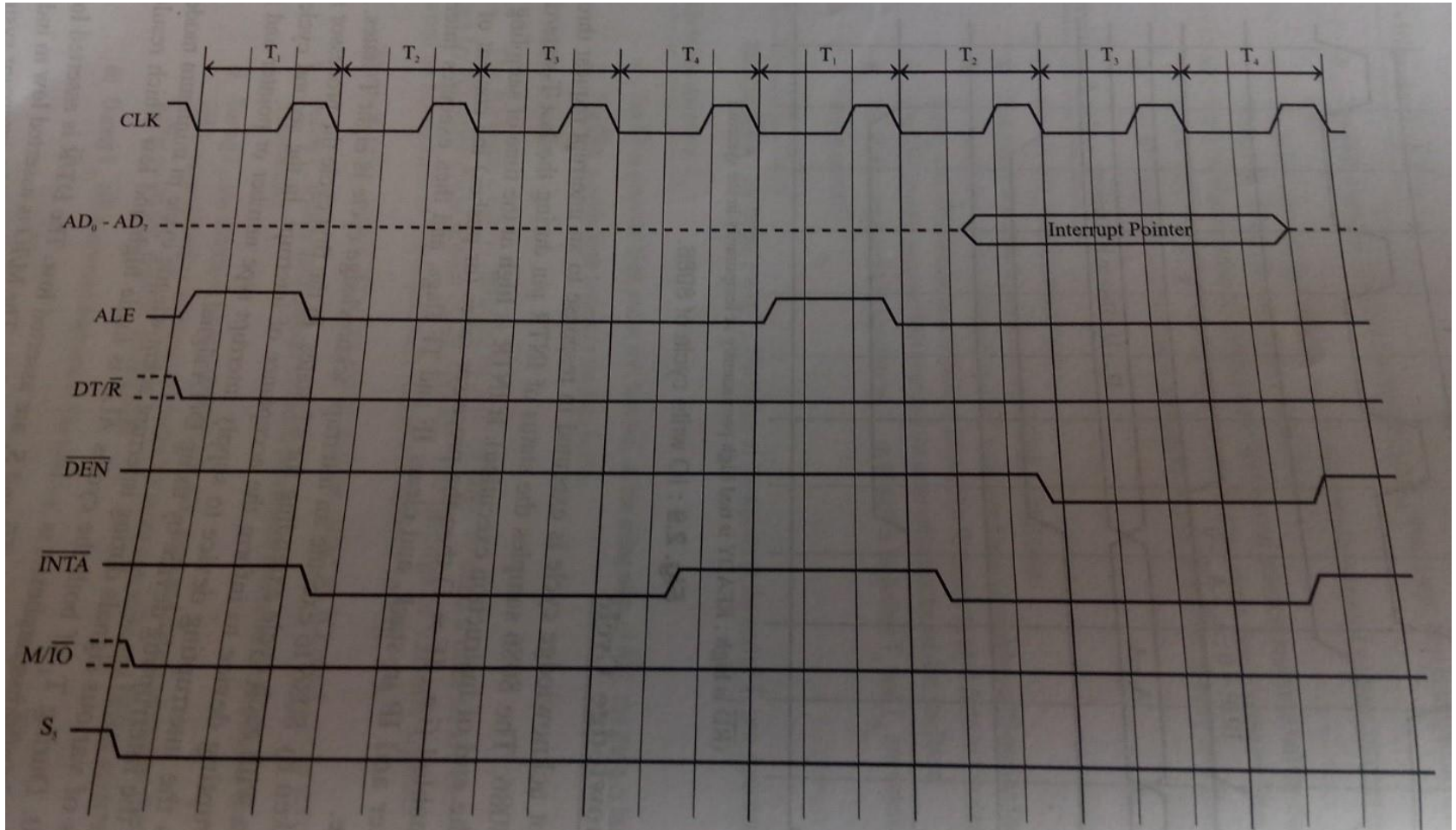


Fig. 2.10 : Interrupt acknowledge cycle of 8086.