5

TIMING DIAGRAM OF 8085

5.1 INTRODUCTION

Timing diagram is the display of initiation of read/write and transfer of data operations under the control of 3-status signals $IO/\overline{M}$, S_1, and S_0. As the heartbeat is required for the survival of the human being, the CLK is required for the proper operation of different sections of the microprocessors. All actions in the microprocessor is controlled by either leading or trailing edge of the clock. If I ask a man to bring 6-bags of wheat, each weighing 100 kg, he may take 6-times to perform this task in going and bringing it. A stronger man might perform the same task in 3-times only. Thus, it depends on the strength of the man to finish the job quickly or slowly. Here, we can assume both weaker and strong men as machine. The weaker man has taken 6-machine cycle (6-times going and coming with one bag each time) to execute the job where as the stronger man has taken only 3-machine cycle for the same job. Similarly, a machine may execute one instruction in as many as 3-machine cycles while the other machine can take only one machine cycle to execute the same instruction. Thus, the machine that has taken only one machine cycle is efficient than the one taking 3-machine cycle. Each machine cycle is composed of many clock cycle. Since, the data and instructions, both are stored in the memory, the μP performs fetch operation to read the instruction or data and then execute the instruction. The μP in doing so may take several cycles to perform fetch and execute operation. The 3-status signals : $IO/\overline{M}$, S_1, and S_0 are generated at the beginning of each machine cycle. The unique combination of these 3-status signals identify read or write operation and remain valid for the duration of the cycle. Table-5.1(*a*) shows details of the unique combination of these status signals to identify different machine cycles. Thus, time taken by any μP to execute one instruction is calculated in terms of the clock period.

The execution of instruction always requires read and writes operations to transfer data to or from the μP and memory or I/O devices. Each read/ write operation constitutes one machine cycle (MC_1) as indicated in Fig. 5.1 (*a*). Each machine cycle consists of many clock periods/ cycles, called **T-states**. The heartbeat of the microprocessor is the clock period. Each and every operation inside the microprocessor is under the control of the clock cycle. The clock signal determines the time taken by the microprocessor to execute any instruction. The clock cycle shown in Fig. 5.1 (*a*) has two edges (leading and trailing or lagging). State is defined as the time interval between 2-trailing or leading edges of the clock. Machine cycle is the time required to transfer data to or from memory or I/O devices.

Table 5.1(*a*) Machine cycle status and control signals

Machine cycle	*Status*			*Controls*		
	IO / $\overline{M}$	S_1	S_0	$\overline{RD}$	$\overline{WR}$	$\overline{INTA}$
Opcode Fetch (OF)	0	1	1	0	1	1
Memory Read	0	1	0	0	1	1
Memory Write	0	0	1	1	0	1
I/O Read (I/OR)	1	1	0	0	1	1
I/O Write (I/OW)	1	0	1	1	0	1
Acknowledge of INTR (INTA)	1	1	1	1	1	0
BUS Idle (BI) : DAD	0	1	0	1	1	1
ACK of RST, TRAP	1	1	1	1	1	1
HALT	Z	0	0	Z	Z	1
HOLD	Z	X	X	Z	Z	1
X ⇒ Unspecified, and Z ⇒ High impedance state						

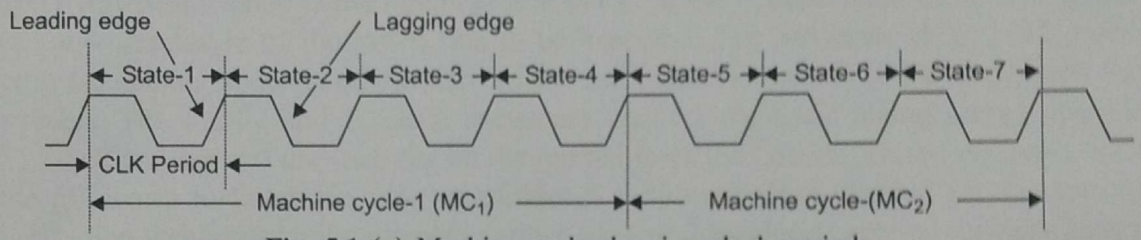

Fig. 5.1 (*a*) Machine cycle showing clock periods

5.2 PROCESSOR CYCLE

The function of the microprocessor is divided into fetch and execute cycle of any instruction of a program. The program is nothing but number of instructions stored in the memory in sequence. In the normal process of operation, the microprocessor fetches (receives or reads) and executes one instruction at a time in the sequence until it executes the halt (HLT) instruction. Thus, an instruction cycle is defined as the time required to fetch and execute an instruction. For executing any program, basically 2-steps are followed sequentially with the help of clocks

- Fetch, and
- Execute.

The time taken by the μP in performing the fetch and execute operations are called fetch and execute cycle. Thus, sum of the fetch and execute cycle is called the instruction cycle as indicated in Fig. 5.2 (*a*).

Instruction Cycle (IC) = Fetch cycle (FC) + Execute Cycle (EC)

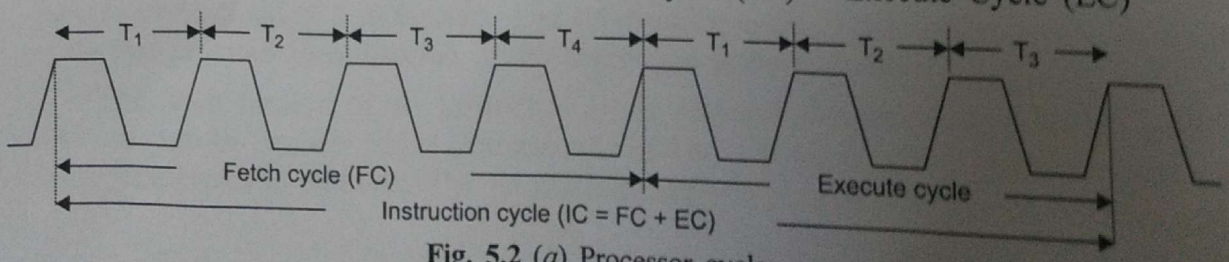

Fig. 5.2 (*a*) Processor cycle

fetch. The 8-bits obtained during an opcode fetch are always interpreted as the Opcode of an instruction. The machine cycle including wait states is shown in Fig. 5.2 (*c*).

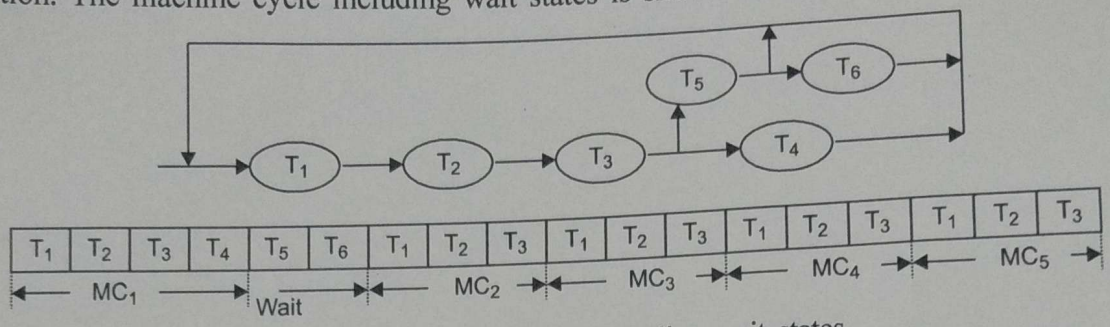

Fig. 5.2 (*c*) Machine cycle including wait states

Note : Some instructions do not require any machine cycle other than that necessary to fetch the instruction. Other instructions, however, require additional machine cycles to write or read data to or from memory or I/O devices.

A typical fetch cycle is explained in Fig. 5.2 (*d*). In Fig. 5.2 (*d*) only two clock cycles have been shown as the requirement to read the instruction. Since the access time of the memory may vary and it may require more than 2-clock cycles, the microprocessor has to wait for more than 2-clocks duration before it receives the opcode instruction. Hence, most of the microprocessors have the provisions of introducing wait cycle within the fetch cycle to cope up with the slow memories or I/O devices.

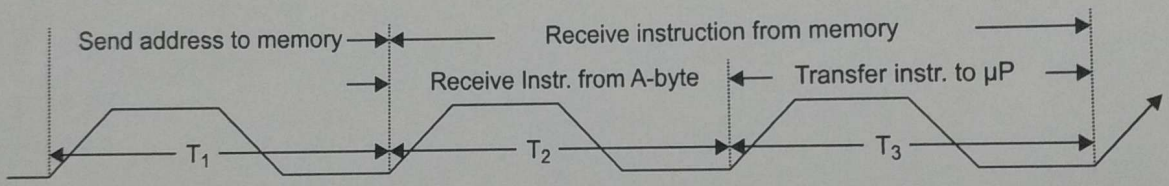

Fig. 5.2 (*d*) Fetch cycle

Opcode Fetch

A microprocessor either reads or writes to the memory or I/O devices. The time taken to read or write for any instruction must be known in terms of the µP clock. The 1st step in communicating between the microprocessor and memory is reading from the memory. This reading process is called opcode fetch. The process of opcode fetch operation requires minimum 4-clock cycles T_1, T_2, T_3, and T_4 and is the 1st machine cycle (M_1) of every instruction.

In order to differentiate between the data byte pertaining to an opcode or an address, the machine cycle takes help of the status signal $IO/\overline{M}$, S_1, and S_0. The $IO/\overline{M} = 0$ indicates memory operation and $S_1 = S_0 = 1$ indicates Opcode fetch operation.

The opcode fetch machine cycle M_1 consists of 4-states (T_1, T_2, T_3, and T_4). The 1st 3-states are used for fetching (transferring) the byte from the memory and the 4th-state is used to decode it.

Thus, thorough understanding about the communication between memory and microprocessor can be achieved only after knowing the processes involved in reading or writing into the memory by the microprocessor and time taken w.r.t. its clock period. This can be explained by examples.

fetch. The 8-bits obtained during an opcode fetch are always interpreted as the Opcode of an instruction. The machine cycle including wait states is shown in Fig. 5.2 (*c*).

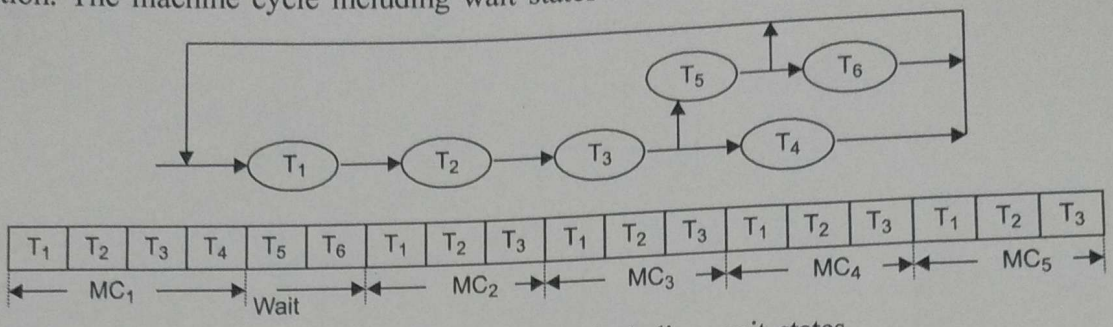

Fig. 5.2 (*c*) Machine cycle including wait states

Note : Some instructions do not require any machine cycle other than that necessary to fetch the instruction. Other instructions, however, require additional machine cycles to write or read data to or from memory or I/O devices.

A typical fetch cycle is explained in Fig. 5.2 (*d*). In Fig. 5.2 (*d*) only two clock cycles have been shown as the requirement to read the instruction. Since the access time of the memory may vary and it may require more than 2-clock cycles, the microprocessor has to wait for more than 2-clocks duration before it receives the opcode instruction. Hence, most of the microprocessors have the provisions of introducing wait cycle within the fetch cycle to cope up with the slow memories or I/O devices.

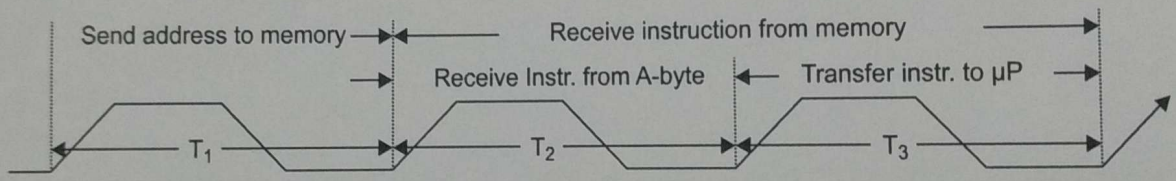

Fig. 5.2 (*d*) Fetch cycle

Opcode Fetch

A microprocessor either reads or writes to the memory or I/O devices. The time taken to read or write for any instruction must be known in terms of the µP clock. The 1st step in communicating between the microprocessor and memory is reading from the memory. This reading process is called opcode fetch. The process of opcode fetch operation requires minimum 4-clock cycles T_1, T_2, T_3, and T_4 and is the 1st machine cycle (M_1) of every instruction.

In order to differentiate between the data byte pertaining to an opcode or an address, the machine cycle takes help of the status signal $IO/\overline{M}$, S_1, and S_0. The $IO/\overline{M} = 0$ indicates memory operation and $S_1 = S_0 = 1$ indicates Opcode fetch operation.

The opcode fetch machine cycle M_1 consists of 4-states (T_1, T_2, T_3, and T_4). The 1st 3-states are used for fetching (transferring) the byte from the memory and the 4th-state is used to decode it.

Thus, thorough understanding about the communication between memory and microprocessor can be achieved only after knowing the processes involved in reading or writing into the memory by the microprocessor and time taken w.r.t. its clock period. This can be explained by examples.