ESE

MADE EASY
WORKbOOK 2025

Detailed Explanations of
Try Yourself *Questions*

Electronics Engineering

Microprocessors

MADE EASY PUBLICATIONS

1 Intel 8085 and Intel 8086

T1. Sol.

Flags

The ALU includes five flip-flops, which are set or reset after an operation according to data conditions of the result in the accumulator and other registers. They are called Zero(Z), Carry (CY), Sign (S), Parity (P), and Auxiliary Carry (AC) flags; their bit positions in the flag register are shown in the Figure below. The most commonly used flags are Zero, Carry, and Sign. The microprocessor uses these flags to test data conditions.

Flag Register

S	Z	X	AC	X	P	X	CY

For example, after an addition of two numbers, if the sum in the accumulator id larger than eight bits, the flip-flop uses to indicate a carry — called the Carry flag (CY) — is set to one. When an arithmetic operation results in zero, the flip-flop called the Zero(Z) flag is set to one. The first Figure shows an 8-bit register, called the flag register, adjacent to the accumulator. However, it is not used as a register; five bit positions out of eight are used to store the outputs of the five flip-flops. The flags are stored in the 8-bit register so that the programmer can examine these flags (data conditions) by accessing the register through an instruction. These flags have critical importance in the decision-making process of the micro- processor. The conditions (set or reset) of the flags are tested through the software instructions. For example, the instruction JC (Jump on Carry) is implemented to change the sequence of a program when CY flag is set. The thorough understanding of flag is essential in writing assembly language programs.

T2. Sol.

Direct Memory Access (DMA)

During any given bus cycle, one of the system components connected to the system bus is given control of the bus. This component is said to be the master during that cycle and the component it is communicating with is said to be the slave. The CPU with its bus control logic is normally the master, but other specially designed components can gain control of the bus by sending a bus request to the CPU. After the current bus cycle is completed the CPU will return a bus grant signal and the component sending the request will become the master.

Taking control of the bus for a bus cycle is called cycle stealing. Just like the bus control logic, a master must be capable of placing addresses on the address bus and directing the bus activity during a bus cycle. The components capable of becoming masters are processors (and their bus control logic) and DMA controllers. Sometimes a DMA controller is associated with a single interface, but they are often designed to accommodate more than one interface.

The 8085 microprocessor receives bus requests through its HOLD pin and issues grants from the hold acknowledge (HLDA) pin. A request is made when a potential master sends a 1 to the HOLD pin. Normally, after the current bus cycle is complete the 8085 will respond by putting a

1 on the HLDA pin. When the requesting device receives this grant signal it becomes the master. It will remain master until it drops the signal to the HOLD pin, at which time the 8085 will drop the grant on the HLDA pin. One exception to the normal sequence is that if a word, which begins at an odd address is being accessed, then two bus cycles are required to complete the transfer and a grant will not be issued until after the second bus cycle.

When a DMA controller becomes master it places an address on the address bus and sends the interface the necessary signals to cause it to put data on, or receive data from, the data bus. Since the DMA controller determines when the bus request is dropped, it can return control to the CPU after each data byte is transferred and then request control again when the next data byte is ready, or it can retain control until the entire block is moved. The former is the usual case because this allows the CPU to continue its work until the next data byte is available.

During a block input byte transfer, the following sequence occurs as the data byte is sent from the interface to the memory: The interface sends the DMA controller a request for DMA service. A Bus request is made to the HOLD pin (active High) on the 8086 microprocessor and the controller gains control of the bus. A Bus grant is returned to the DMA controller from the Hold Acknowledge (HLDA) pin (active High) on the 8086 microprocessor. The DMA controller places contents of the address register onto the address bus. The controller sends the interface a DMA acknowledgment, which tells the interface to put data on the data bus. (For an output it signals the interface to latch the next data placed on the bus.)

The data byte is transferred to the memory location indicated by the address bus.

The interface latches the data.The Bus request is dropped, the HOLD pin goes Low, and the controller relinquishes the bus. The Bus grant from the 8085 microprocessor is dropped and the HLDA pin goes Low. The address register is incremented by 1. The byte count is decremented by 1. If the byte count is non-zero, return to step 1, otherwise stop. Direct Memory Access Controller (DMAC) options for data transfer. The DMA Controller has several options available for the transfer of data. They are:

(i) **Cycle Steal :** A read or write signal is generated by the DMAC, and the I/O device either generates or latches the data. The DMAC effectively steals cycles from the processor in order to transfer the byte, so single byte transfer is also known as cycle stealing.

(ii) **Burst Transfer :** To achieve block transfers, some DMAC's incorporate an automatic sequencing of the value presented on the address bus. A register is used as a byte count, being decremented for each byte transfer, and upon the byte count reaching zero, the DMAC will release the bus. When the DMAC operates in burst mode, the CPU is halted for the duration of the data transfer.

■■■■

T1. **(1)**

XRA A $\rightarrow$ A = 00 H   CY = 0

DCR A decrease the content of accumulator and jump instruction check for zero flag. Thus loop will be executed for 50 H times or 80 times. Once zero flag is set content of B is incremented and carry flag is checked. Since carry flag is clear the control of program will not go to loop again.

$\Rightarrow$ INR B is executed only once.

T2. **Sol.**

Infinite.

T3. **(65279)**

First of all DCR L will be executed for 255 times till content of L becomes 0.

After that for every decrement of H it will be executed

256 times (first when L is decremented by 1 and become FF_H and then $FF_H = 255_D$ times) and this will be happen til content of H become 1 i.e. 254 times.

So overall it will be executed

$$= 255 + 254 \times 256$$
$$= 65279$$

T4. **Sol.**

(i)   AX | 20 | H $\longrightarrow$ SP $\longrightarrow$ | 20 H |

(ii)   BX | 34 | H $\longrightarrow$ SP $\longrightarrow$ | 34 H | 20 H |

AX | 34 | H $\longrightarrow$ SP $\longleftarrow$ | 34 H | 20 H |

(iii)   34 H + 34 H = 68 H

CX | 20 | H $\longrightarrow$ SP $\longleftarrow$ | 20 H |

Contents of registers.

AX = 68 H
BX = 34 H
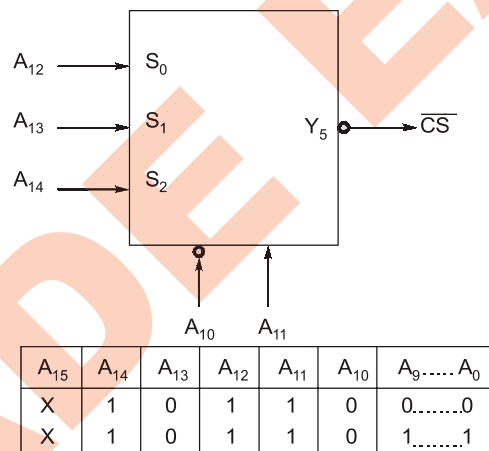CX = 20 H

T5. **(d)**

■■■■

T1. Sol.

Total Memory of 8085 = 2^{16} bytes

Size of one page = 256 = 2^8 bytes

Number of pages = $\dfrac{2^{16}}{2^8}$ = 2^8 = 256 pages

T2. (d)

1 KB memory is interfaced

A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	$A_9 \ldots A_0$
X	1	0	1	1	0	0........0
X	1	0	1	1	0	1........1

If $A_{15} = 0 \Rightarrow$ 5800 H to 5 BFFH

If $A_{15} = 1, \Rightarrow$ D800H to DBFFH

T3. Sol.

8085 is an 8 bit microprocessor with 8 bit address lines for I/O devices. So, in I/O mapped I/O mode, 8085 can have at most 2^8 = 256 input devices and 256 output devices. Their addresses will lie in the range from 00000000 to 11111111.

T4. (c)

The chip select is active high, hence, $CS = A_{15} A_{14} A_{13} = 111$ chip select of 1 kB RAM, $CS = A_{12} \overline{A}_{11} \overline{A}_{10} = 100$

The address range,

A_{15}	A_{14}	A_{13}	A_{12}	A_{11}	A_{10}	A_9	$\cdots$	A_0
1	1	1	1	0	0	0	$\cdots$	0
			$\vdots$				$\vdots$	
1	1	1	1	0	0	1	$\cdots$	1

$\therefore$ The address range, F000H to F3FFH.

■■■■