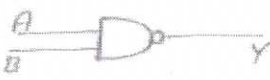


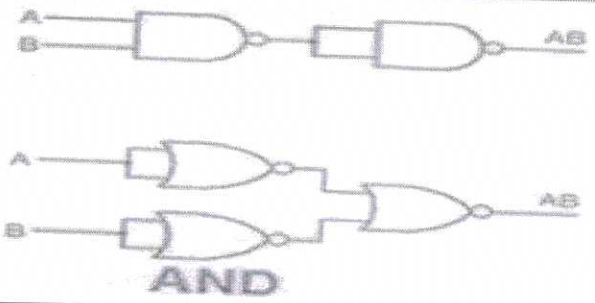
69

Page - 12  
April: 2024

ANSWER KEY SET 2		QID : 1510230047																	
DIGITAL ELECTRONICS AND MICROPROCESSORS 4032																			
I	PART A	Split up	Sub total	total															
1	Resister transistor logic ( RTL) Diode transistor logic ( DTL) Transistor transistor logic (TTL) Emitter coupled logic (ECL) Complementary metal oxide semiconductor logic (CMOS)	Any 4  0.5 x 4	2	10															
2	<p>NAND</p>  <table border="1" style="display: inline-table; vertical-align: middle;"> <tr><td>A</td><td>B</td><td>Y</td></tr> <tr><td>0</td><td>0</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </table>	A	B	Y	0	0	1	0	1	1	1	0	1	1	1	0	1 1	2	
A	B	Y																	
0	0	1																	
0	1	1																	
1	0	1																	
1	1	0																	
3	<p>Theorem-1: Complement of a sum = product of complements <math>\overline{A+B} = \overline{A} \cdot \overline{B}</math></p> <p>Theorem-2: Complement of product = sum of the complements <math>\overline{A \cdot B} = \overline{A} + \overline{B}</math></p>	1  1	2																
4	<p>i. Ramp type ADC ii. Successive approximation ADC</p>	2 x 1	2																
5	Data bus is used to transfer data between memeory and processor or between I/O device and processor.	2	2																

II	PART B	Sp lit up	Sub total	total
1	<p>1's complement : it is obtained by changing the bit each from 0 to 1 and each 1 to 0</p> <p>2's complement : it is obtained by adding 1 to 1's complement =1's complement +1</p> <p>a) 01101101 1's complement 10010010 1 + 2's complement 10010011</p> <p>b) 10011010 1's complement 01100101 1 + 2's complement 01100110</p>	2  2  2	6	

2

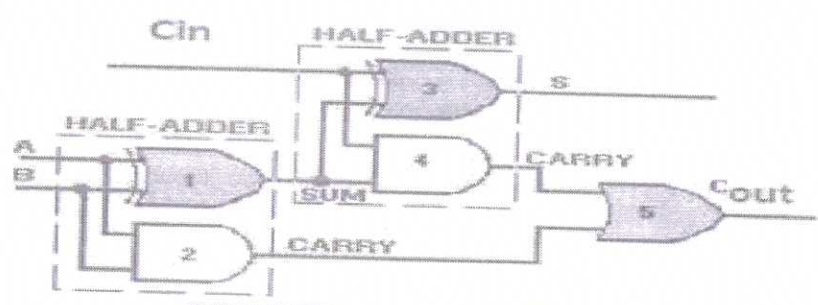


3

3

6

3



A	B	Cin	S	Cout
0	0	0	0	0
0	1	0	1	0
1	0	0	1	0
1	1	0	0	1
0	0	1	1	0
0	1	1	0	1
1	0	1	0	1
1	1	1	1	1

$$S = A \oplus B \oplus C_{in}$$

$$C_{out} = (A \oplus B)C_{in} + AB$$

3

6

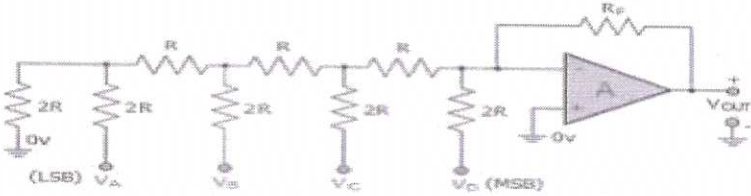
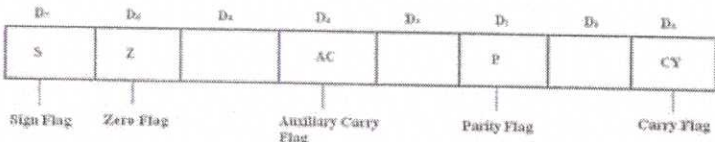
3

4

Sl. no.	Asynchronous counter	Synchronous counter
1	All the flip-flops are not clocked simultaneously.	All the flip-flops are clocked simultaneously.
2	Output of first flip-flop drives the clock for second flip-flop, the output of second drives the third and so on.	there is no interconnection between output of one flip-flop and clock of next flip-flop,
3	The settling time is cumulative sum of individual flip-flops.	The settling time is equal to highest settling time of all flip-flops.
4	Asynchronous counter is known as serial counter	Synchronous counter is known as parallel counter.
5	Its design and implementation is very simple.	Design and implementation becomes tedious and complex as the number of states increases.
5	slow in speed	faster in speed

6

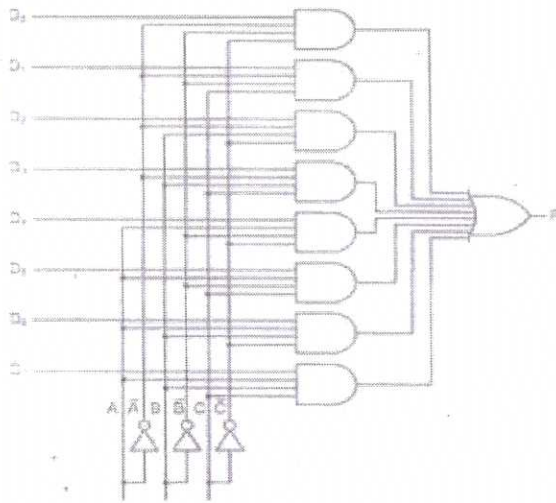
6

5	 <p>As its name implies, the “ladder” description comes from the ladder-like configuration of the resistors used within the network. A R-2R resistive ladder network provides a simple means of converting digital voltage signals into an equivalent analogue output. Input voltages are applied to the ladder network at various points along its length and the more input points the better the resolution of the R-2R ladder. The output signal as a result of all these input voltage points is taken from the end of the ladder which is used to drive the inverting input of an operational amplifier</p>	4	6	
6	<p>An <b>instruction</b> is a binary pattern designed inside a microprocessor to perform a specific function. The entire group of instructions, called the <b>instruction set</b>, determines what functions the microprocessor can perform.</p> <p>These instructions can be classified into the following five functional categories: 1. data transfer (copy) operations, 2. Arithmetic operations, 3. Logical operations. 4. Branching operations, and 5. Machine-control operations.</p> <p>The data transfer instructions move data between registers or between memory and registers. The arithmetic instructions add, subtract, increment, or decrement data in registers or memory. The logical (Boolean) operations on data in registers and memory and on condition flags. The logical AND, OR, and Exclusive OR instructions enable you to set specific bits in the accumulator ON or OFF. The branching instructions alter normal sequential program flow, either unconditionally or conditionally. I/O instruction, stack instruction and machine instructions are also included in the set.</p>	6	6	
7	 <ul style="list-style-type: none"> <li>- Zero flag (Z): If a result of an instruction has a value zero, this flag is set. Otherwise it is reset</li> <li>- Sign flag (S): If the MSB of the result of an operation has a value 1, this flag is set, otherwise reset</li> <li>- Carry flag (C): After an operation if a carry is generated a carry flag will set</li> <li>- Auxiliary carry flag (AC): During the arithmetic operation, if a carry is moved from the 3<sup>rd</sup> bit to 4<sup>th</sup> bit, then this flag is set, otherwise it is reset</li> <li>- Parity flag (P): The flag is set only when the result of operation contains even number of ones. If it is odd, the flag will reset.</li> </ul>	6	6	





Va



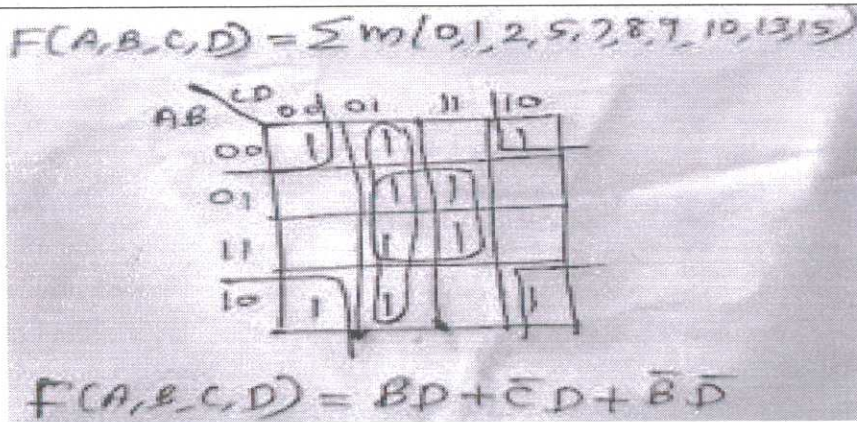
Select Data inputs			Output
S <sub>2</sub>	S <sub>1</sub>	S <sub>0</sub>	Y
0	0	0	D <sub>0</sub>
0	0	1	D <sub>1</sub>
0	1	0	D <sub>2</sub>
0	1	1	D <sub>3</sub>
1	0	0	D <sub>0</sub>
1	0	1	D <sub>1</sub>
1	1	0	D <sub>2</sub>
1	1	1	D <sub>3</sub>

4

7

3

Vb



4

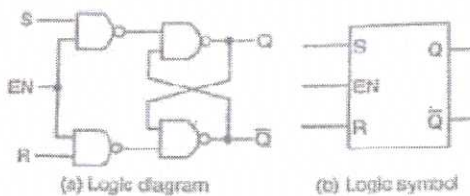
8

4

Vi

a

**The gated S-R latch:** In the latches described earlier, the output can change state any time the input conditions are changed. So, they are called *asynchronous* latches. A gated S-R latch requires an ENABLE (EN) input. Its S and R inputs will control the state of the flip-flop only when the ENABLE is HIGH. When the ENABLE is LOW, the inputs become ineffective and no change of state can take place. The ENABLE input may be a clock. So, a gated S-R latch is also called a *clocked S-R latch* or *synchronous S-R latch*. Since this type of flip-flop responds to the changes in inputs only as long as the clock is HIGH, these types of flip-flops are also called *level triggered flip-flops*. The sequential circuits (machines) controlled by a clock are called *synchronous sequential circuits (machines)*. The logic diagram, the logic symbol and the truth table for a gated S-R latch are shown in Figure 10.7. In this circuit, the invalid state occurs when both S and R are simultaneously HIGH.



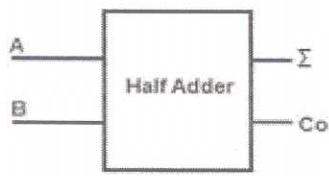
EN	S	R	Q <sub>n</sub>	Q <sub>n+1</sub>	State
1	0	0	0	0	No change (NC)
1	0	0	1	1	
1	0	1	0	0	Reset
1	0	1	1	0	
1	1	0	0	1	Set
1	1	0	1	1	
1	1	1	0	x	Indeterminate (invalid)
1	1	1	1	x	
0	x	x	0	0	No Change (NC)
0	x	x	1	1	

3

7

4

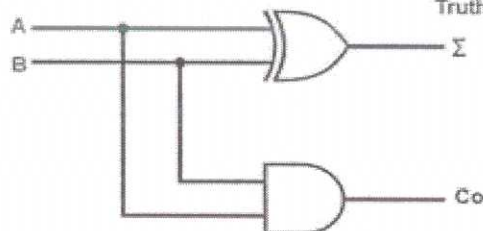
Vi  
b



Block Diagram

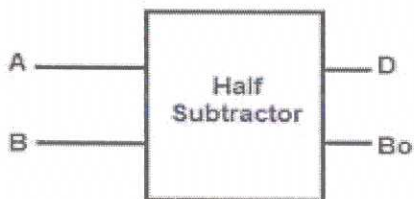
A	B	Sum ( $\Sigma$ )	Carry Out (Co)
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Truth Table



Logic Diagram

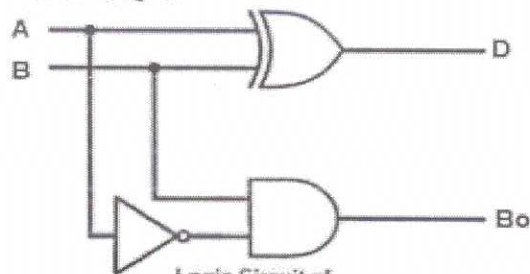
4



Block Diagram

A	B	D	B <sub>o</sub>
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

Truth Table

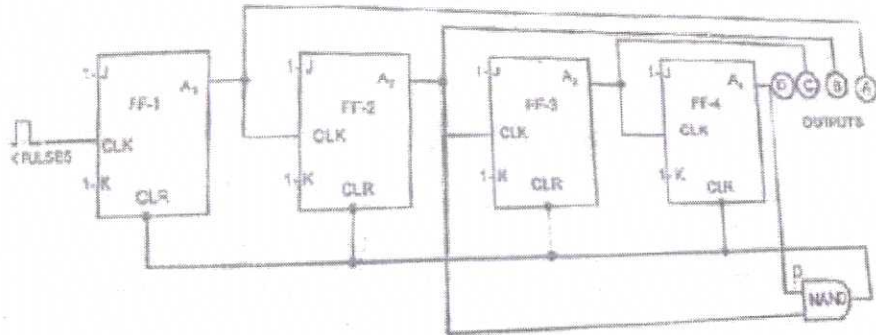


Logic Circuit of Half Subtractor

4

8

Vii  
a

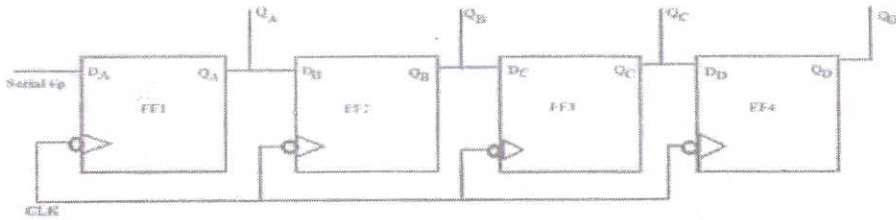


7

7

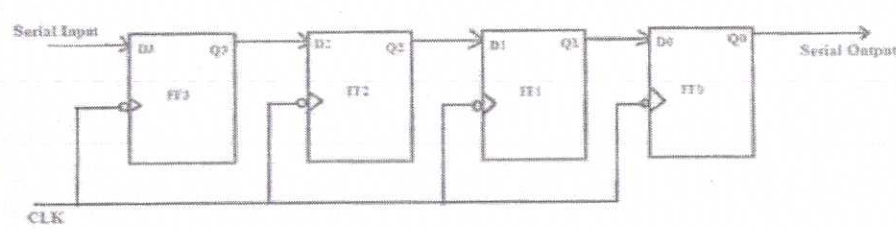
Vii  
b

*SIPO*



4

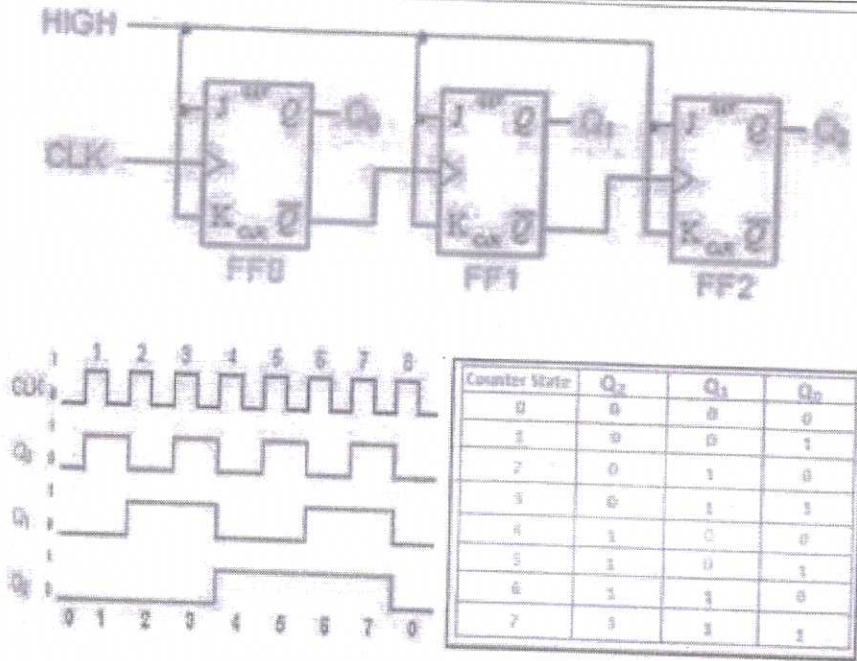
*SISO*



4

8

Viii  
a



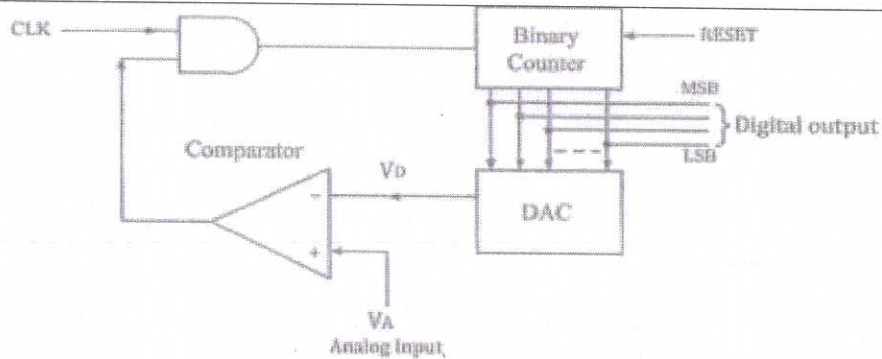
Three flip flops are used to count eight values. high value is given to j and k so that flip flop will toggle at the application of each clock pulse. output of flip flop is connected to clock of next. Q0 is LSB Q2 is MSB

4

8

4

viii  
b



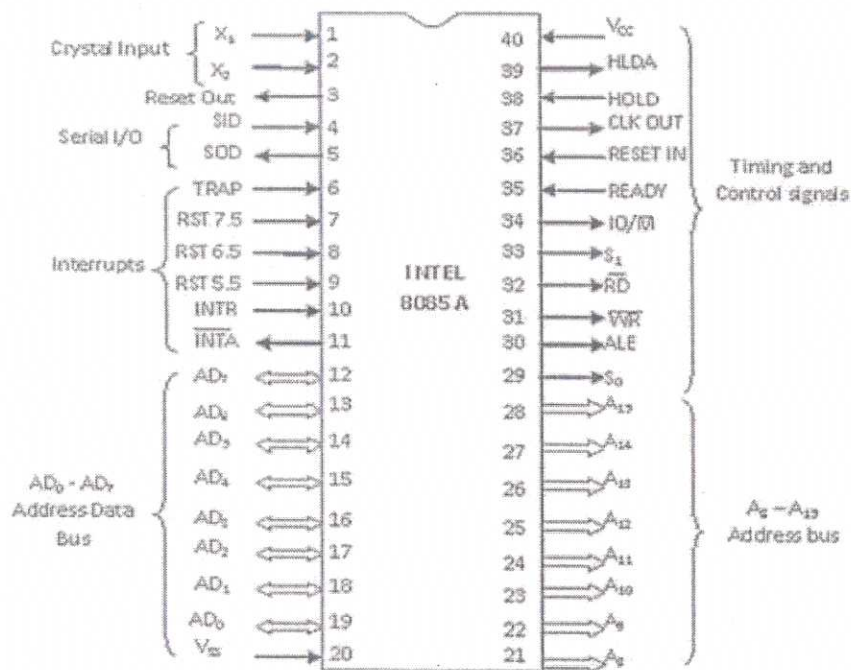
A start pulse is applied to reset the counter to zero and AND gate is enabled and clock pulses get through to counter. since  $V_A > V_D$  comparator output will be high. as the counter advances, the DAC output  $V_D$  increases one step at a time and this continues until  $V_D > V_A$ .

4

3

7

<p>ix a</p>	<p>The diagram illustrates the internal architecture of the 8085 microprocessor. At the top, the Interrupt Control block handles signals for INTR, RST 5.5, RST 6.5, RST 7.5, and TRAP. The Serial I/O Control block manages the SID and SOD signals. These are connected to an 8-bit internal data bus. Key components include the Accumulator (8-bit), Temporary Register (8-bit), Instruction Register (8-bit), and Arithmetic Logic Unit (ALU) (8-bit). The ALU is connected to the Register Array, which contains registers B, C, D, E, H, L, Program Counter (PC), Stack Pointer (SP), and Increment/Decrement Address Latch. The Register Array also includes a Multiplexer and a Register Select input. The Instruction Decoder and Machine Cycle Encoder are connected to the Instruction Register and the Register Array. The Timing and Control block manages the clock (CLK GEN), control signals (RD, WR, ALE, <math>\overline{S}_1, \overline{S}_2, \overline{RD/ST}</math>), DMA, HOLD, and RESET signals. It also provides the Address Buffer (8-bit) and Data/Address Buffer (8-bit) for external communication. The power supply is connected to +5V and GND.</p>	<p>9</p>	<p>9</p>	
<p>ix b</p>	<ol style="list-style-type: none"> <li>1. It is an 8-bit microprocessor i.e. it can accept, process, or provide 8-bit data simultaneously.</li> <li>2. It operates on a single +5V power supply connected at <math>V_{cc}</math>; power supply ground is connected to <math>V_{ss}</math>.</li> <li>3. 8085 microprocessor requires two-phase, 50% duty cycle, TTL clock. These clock signals are generated by an internal clock generator.</li> <li>4. The maximum clock frequency of 8085 microprocessors is 3MHz whereas minimum clock frequency is 500 KHz.</li> <li>5. 8085 microprocessor provides 16 address lines, therefore it can access <math>2^{16} = 64K</math> bytes of memory.</li> <li>6. It provides 8 bit I/O addresses to access <math>(2^8)</math> 256 I/O ports.</li> <li>7. In 8085, the lower 8-bit address bus (<math>A_0 - A_7</math>) and data bus (<math>D_0 - D_7</math>) are Multiplexed to reduce number of external pins. But due to this, external hardware (latch) is required to separate address lines and data lines.</li> <li>8. 8085 microprocessor has five hardware interrupts: TRAP, RST 5.5, RST 6.5, RST 7.5, INTR.</li> <li>9. The hardware interrupt capability of 8085 microprocessor can be increased by providing external hardware..</li> <li>10. It supports 74 instructions.</li> </ol>	<p>6 x 1</p>	<p>6</p>	<p>Any 6</p>
<p>X a</p>				



Pin diagram of 8085 microprocessor

9

9

#### 8085 Pin description

- Higher Order Address pins-  $A_{15} - A_8$ 
  - The address bus has 8 signal lines  $A_8 - A_{15}$  which are unidirectional.
- Lower Order Address/ Data Pins-  $AD_7 - AD_0$ 
  - These are time multiplexed pins and are de-multiplexed using the pin ALE
  - So, the bits  $AD_0 - AD_7$  are bi-directional and serve as  $A_0 - A_7$  and  $D_0 - D_7$  at the same time.
    - During the execution of the instruction, these lines carry the address bits during the early part, then during the late parts of the execution, they carry the 8 data bits.
    - In order to separate the address from the data, we can use a latch to save the value before the function of the bits changes.
- Control Pins - RD, WR
  - These are active low Read & Write pins
- Status Pins - ALE, IO/M (active low),  $S_1, S_0$ 
  - ALE (Address Latch Enable)-Used to de-multiplex  $AD_7 - AD_0$
  - IO/M - Used to select I/O or Memory operation
  - $S_1, S_0$  - Denote the status of data on data bus
- Interrupt Pins - TRAP, RST 7.5, RST 6.5, RST 5.5, INTR, INTA
  - These are hardware interrupts used to initiate an interrupt service routine stored at predefined locations of the system memory.
- Serial I/O pins - SID (Serial Input Data), SOD (Serial Output Data)
  - These pins are used to interface 8085 with a serial device.
- Clock Pins-  $X_1, X_2, CLK(OUT)$ 
  - $X_1, X_2$  - These are clock input pins. A crystal is connected between these pins such that  $f_{crystal} = 2f_{8085}$  where  $f_{crystal}$  = crystal frequency &  $f_{8085}$  = operating frequency of 8085
  - CLK(OUT) - This is an auxiliary clock output source

3

1

1

X b	<p>The way of specifying data to be operated by an instruction is called addressing mode.</p> <p><b>Types of addressing modes –</b></p> <p><b>Immediate addressing mode:</b> In immediate addressing mode the source operand is always data. Here data to be used is immediately given in the instruction itself. If the data is 8-bit, then the instruction will be of 2 bytes, if the data is of 16-bit then the instruction will be of 3 bytes. Eg:- MVI B,05 (move the data 05 immediately to register B)</p> <p><b>Register addressing mode:</b> here data/operand to be operated is in the general purpose register. therefore the operation is performed within various registers. Eg:- MOV A,B (move the content of register B to register A)</p> <p><b>Direct Addressing Mode:</b> In direct addressing mode, the data to be operated is available inside a memory location and that memory location is directly specified as an operand. The operand is directly available in the instruction itself. Eg:- LDA F500 (load the content of memory location to the accumulator)</p> <p><b>Indirect addressing mode:</b> data to be operated is available inside a memory location and that memory location is indirectly specified by a register pair. Eg:- LDAX B (move content of B-C register to accumulator)</p> <p><b>Implied/Implicit/Inherent Addressing Mode:</b> In implied/implicit addressing mode the operand is hidden and the data to be operated is available in the instruction itself. Eg:- CMA (finds and stores the 1's complement of the contents of accumulator, A in A) RRC (rotate accumulator A right by one bit)</p>	6	6	
-----	--	---	---	--

12

\*\*\*\*\*

*[Handwritten signature]*  
 Lect in EE  
 GPC Panch.