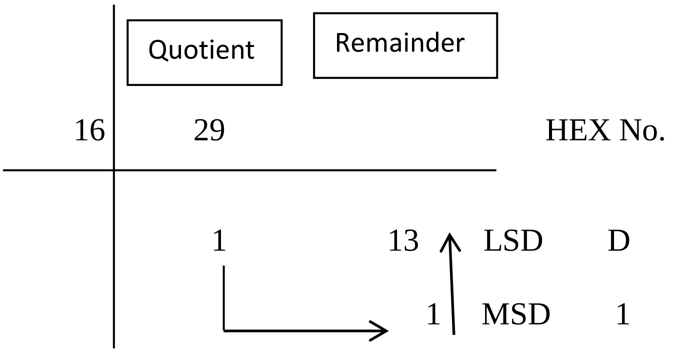
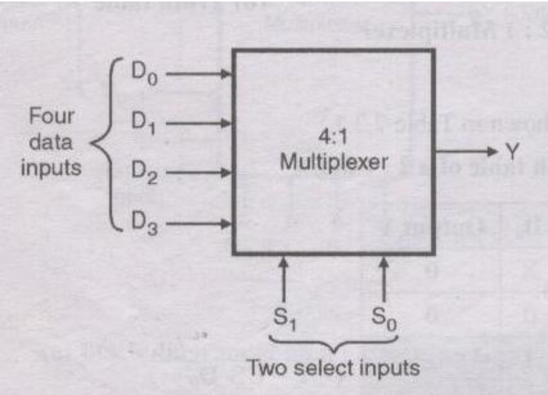
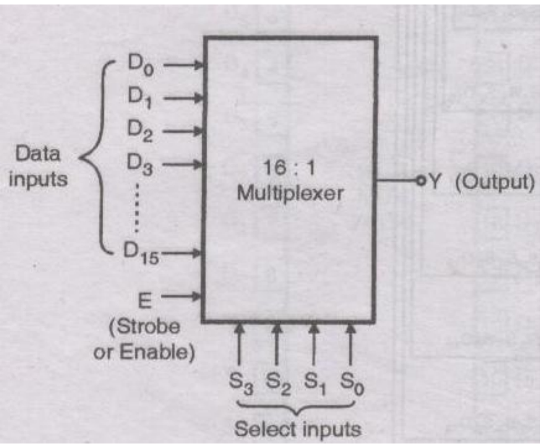
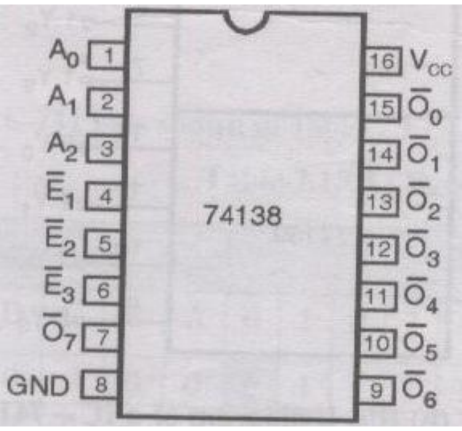
MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION
(Autonomous)
(ISO/IEC – 27001 – 2005 Certified)

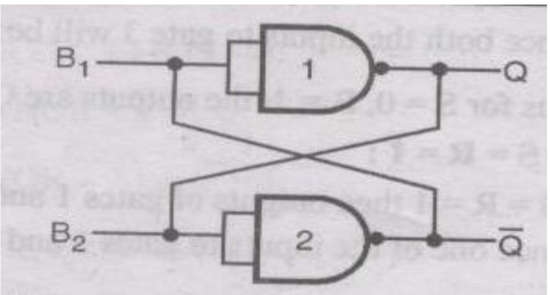
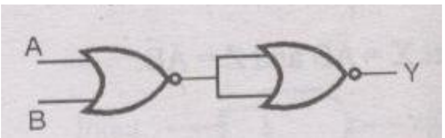
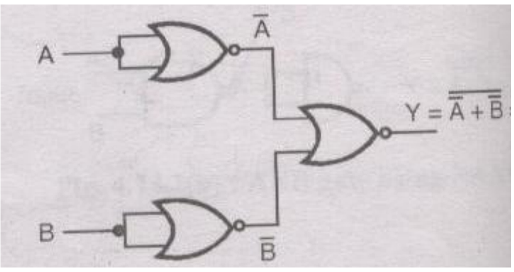
WINTER – 12 EXAMINATION

Model Answer

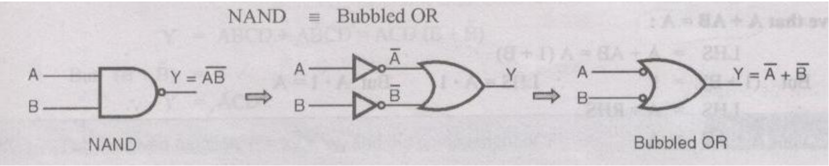
Subject Code : 12069

Q.1	**Attempt any TEN of the following**	**20 Marks**
Ans a. (i)	$101101_2 = 1 \times 2^5 + 0\times 2^4 + 1\times 2^3 + 1\times 2^2 + 0 \times 2^1 + 1\times 2^0$ $= 32+0+8+4+1$ $= (45)_{10}$ **Ans: $(101101_2) = (45)_{10}$**	½ Mark ½ Mark
(ii)	$(29)_{10} =$ $=$  Quotient Remainder 16 29 HEX No. 1 13 LSD D 1 MSD 1 **Ans: $(29_{10}) = (1D)_{16}$**	½ Mark ½ Mark
Ans b. (i)	9’s complement **of 4 =** 9 - 4 =5 10’s complement of 4 = 9’s complement **of 4 + 1** = (9 - 4)+1 = 5 + 1 = 6	½ Mark ½ Mark
(ii)	9’s complement **of 7 =** 9 - 7 =2 10’s complement of 7 = 9’s complement **of 7 +1** =(9 - 7) + 1 = 2 + 1 = 3	½ Mark ½ Mark
Ans c.	Gray code is called as unit distance code because in gray code the bit patterns for two consecutive numbers differ in only one bit position i.e only one bit changes at a time.	2 Marks
Ans d.	Commutative law of Boolean Algebra: 1st law: A . B = B. A 2nd law: A + B = B + A	1 Mark 1 Mark

Ans e. (i) (ii)	 Four data inputs D_0 D_1 D_2 D_3 4:1 Multiplexer Y S_1 S_0 Two select inputs **Fig : Symbol of 4:1 Multiplexer**  Data inputs D_0 D_1 D_2 D_3 D_{15} E (Strobe or Enable) 16 : 1 Multiplexer Y (Output) S_3 S_2 S_1 S_0 Select inputs **Fig: Symbol of 16:1 Multiplexer**	1 Mark 1 Mark
Ans f.	 A_0 1 A_1 2 A_2 3 $\overline{E}_1$ 4 $\overline{E}_2$ 5 $\overline{E}_3$ 6 $\overline{O}_7$ 7 GND 8 74138 16 V_{CC} 15 $\overline{O}_0$ 14 $\overline{O}_1$ 13 $\overline{O}_2$ 12 $\overline{O}_3$ 11 $\overline{O}_4$ 10 $\overline{O}_5$ 9 $\overline{O}_6$ **Fig: Pin out diagram of IC 74138**	2 Marks
Ans g.	Toggle flip flop used in counters reason: • Counters are used to count input clock pulses.i.e. upon application of clock pulse output count should change. • This is possible with toggle flip flop as , in toggle Flip flop upon application of clock pulse for input T=1 , output toggles(i.e. if previous value is 0, after application of Clock pulse it becomes 1 and vice versa). And for input T=0, output is No change (i.e.remains in previous stage).	2 Marks

Ans h.	 **Fig: 1 bit memory cell using NAND gates.**	2 Marks
Ans i.	Fan in : Fan in is defined as the number of input a gate has. e.g. A two input gate will have a fan in equal to 2. Fan out : Fan out is defined as the maximum number of inputs of the same IC family that a gate can drive without falling outside the specified output voltage limits. Higher the fan out ,higher the current supplying capacity of the gate. e.g. Fan out of 5 indicates that the gate can drive (supply current to) at the most 5 inputs of the same IC family.	1 Mark 1 Mark
Ans j.	 **Y= A + B** **Fig: OR gate using NOR gate only**  A $\overline{A}$ $Y = \overline{\overline{A}+\overline{B}}$ B $\overline{B}$ **Y= A. B** **Fig: AND gate using NOR gate only**	1 Mark 1 Mark

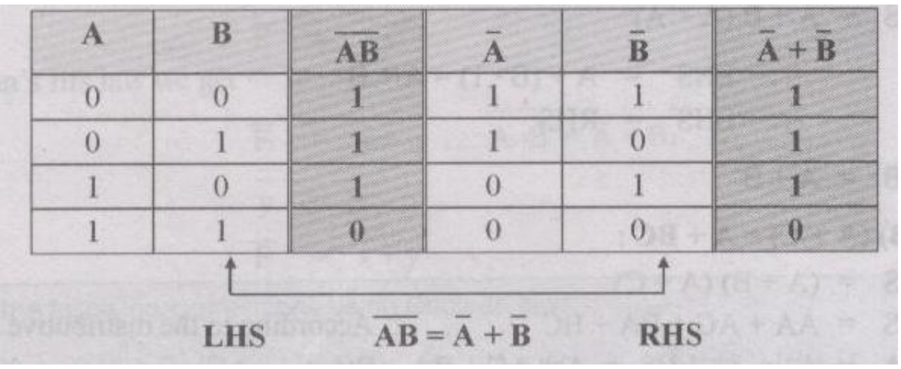
Ans k.	Advantages of TTL logic gate having open collector output : 1) Wired ANDing becomes possible:-Open-collector outputs can be tied directly together which results in the logical ANDing of the outputs. Thus the equivalent of an AND gate can be formed by simply connecting the outputs. 2) Increased current levels - Standard TTL gates with totem-pole outputs can only provide a HIGH current output of 0.4 mA and a LOW current of 1.6 mA. Many open-collector gates have increased current ratings 3) Different voltage levels - A wide variety of output HIGH voltages can be achieved using open-collector gates. This is useful in interfacing different logic families that have different voltage and current level requirements.	2 Marks for Any 2 points
Ans l.	Law of Boolean Algebra: 1) Commutative law. 2) Associative law. 3) Distributive law. 4) AND law. 5) OR law. 6) INVERSION law.	2 Marks for Any 4 laws.
Q.2	**Attempt any <u>FOUR</u> of the following**	**16 Marks**
Ans a. (i)	$\begin{array}{cccccc} & & 1 & 0 & 1 & 1 \\ + & & 1 & 1 & 0 & 1 \\ & & \boxed{1} & \boxed{1} & \boxed{1} & \\ \hline & \boxed{1} & 1 & 0 & 0 & 0 \end{array}$ →Carry $\boxed{\text{Ans:- 1011 +1101 = 11000}}$	1 Mark 1 Mark

Ans a.(ii)	**1101 -101 using 1's complement method** Solution : 1's complement of a binary number is obtained by replacing " 1" by " 0" & " 0" by " 1" . Therefore, 1's complement of 0101 =1 010 1101 -101 = 1101 + (-0101) → 1's complement of 0101 i.e. 1101 + 1010 $\boxed{1}$ 0111 → Since carry is generated add final carry to answer i.e. 0111+1=1000 **Ans: 1101-101=1000**	1 Mark
	1101 -101 using 2's complement method Solution : 2's complement of a binary number is obtained by adding " 1" to 1's complement of that number. Therefore, 2's complement of 0101 = (1's complement of 0101)+1=1 010+1=1011 1101 -101 = 1101 + (-0101) → 2's complement of 0101 i.e. 1101 + 1011 1 1 1 $\boxed{1}$ 1000 → Since carry is generated **discard it** Ans:1101-101=1000	1 Mark
Ans b.	**De- Morgan's Theorems:** `**Theorem 1 :** $\overline{AB} = \overline{A} + \overline{B}$: (NAND = Bubbled OR) - This theorem states that the, complement of a product is equal to addition of the complements.  NAND ≡ Bubbled OR A B Y = $\overline{AB}$ ⇨ A B $\overline{A}$ $\overline{B}$ Y ⇨ A B Y = $\overline{A}$ + $\overline{B}$ NAND Bubbled OR **Fig:(a) Illustration of De- Morgan's first Theorem**	**1 Mark**

Verification of the first theorem:

This theorem can be verified by writing a truth table as shown in Fig (b).

1 Mark



A	B	$\overline{AB}$	$\overline{A}$	$\overline{B}$	$\overline{A}+\overline{B}$
0	0	1	1	1	1
0	1	1	1	0	1
1	0	1	0	1	1
1	1	0	0	0	0

LHS $\overline{AB} = \overline{A} + \overline{B}$ RHS

Fig (b) Verification of theorem $\overline{AB} = \overline{A} + \overline{B}$

Theorem 2: $\overline{A + B} = \overline{A} . \overline{B}$: NOR = Bubble AND

1 Mark

- This theorem states that the, complement of a sum is equal to product of complements. This theorem is illustrated in Fig ©

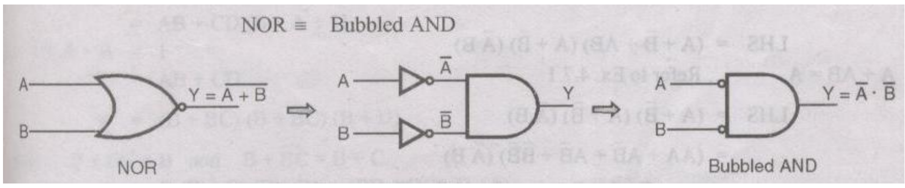
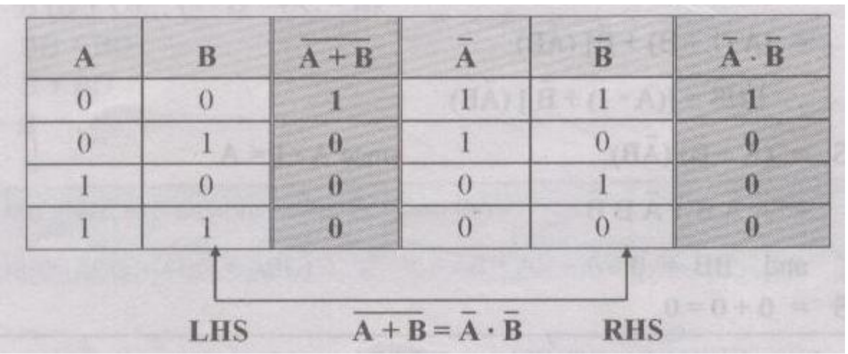


Fig (c) Illustration of De- Morgan's Second Theorem

Verification of the second theorem :

1 Mark



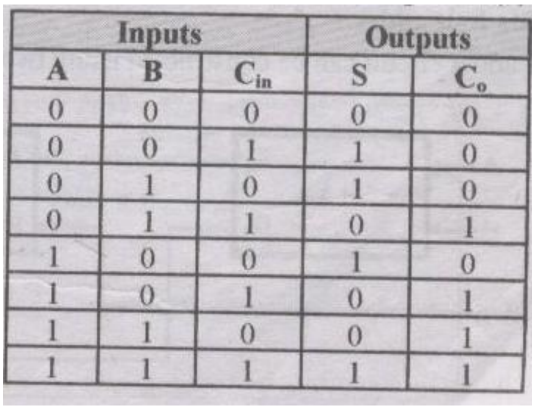
A	B	$\overline{A+B}$	$\overline{A}$	$\overline{B}$	$\overline{A} \cdot \overline{B}$
0	0	1	1	1	1
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	0

LHS $\overline{A+B} = \overline{A} \cdot \overline{B}$ RHS

Fig (d) Truth table to verify De- Morgan's second theorem

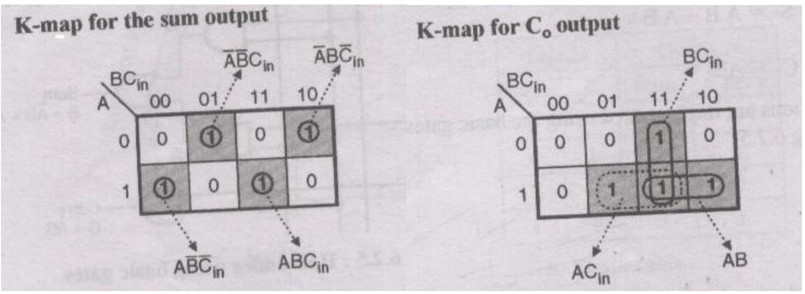
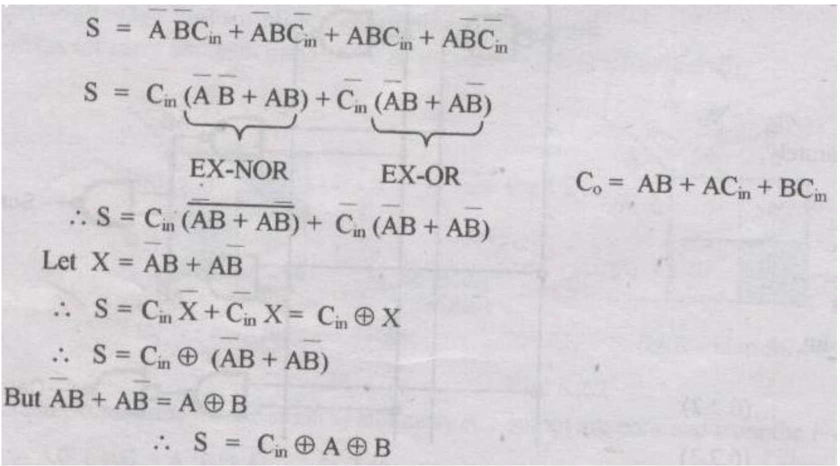
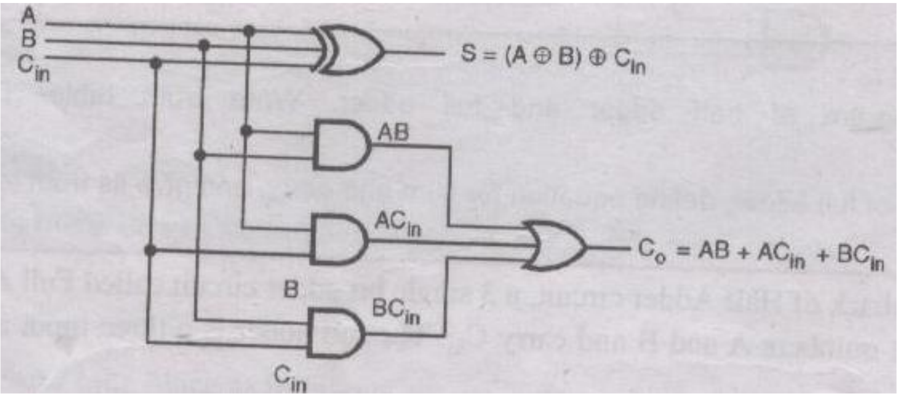
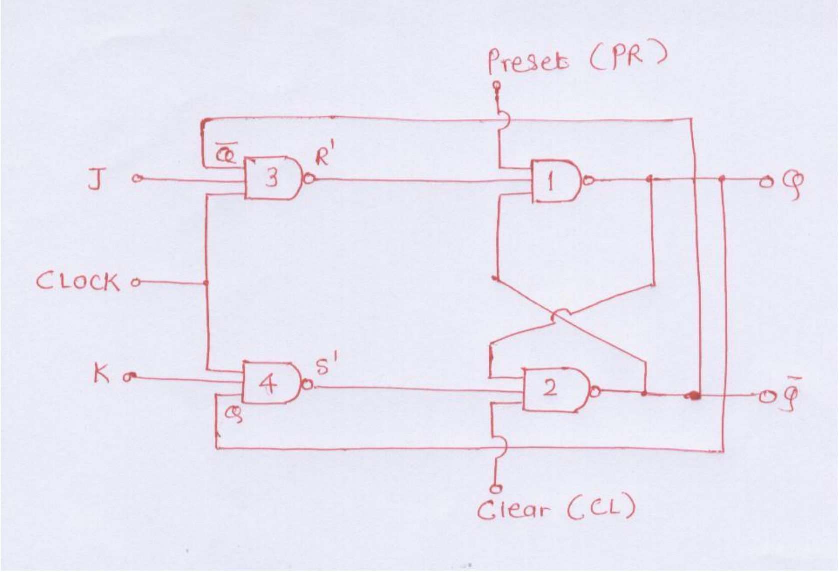
Ans c.

1 Mark



Inputs			Outputs	
A	B	C_{in}	S	C_o
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Fig: Truth Table of full adder

	 K-map for the sum output BC_{in} A 00 01 11 10 0 1 $\bar{A}\bar{B}C_{in}$ $\bar{A}B\bar{C}_{in}$ $A\bar{B}\bar{C}_{in}$ ABC_{in} K-map for C_o output BC_{in} AC_{in} AB **Fig: K- Map to realize full adder.**	**1 Mark**
	 $S = \bar{A}\bar{B}C_{in} + \bar{A}B\bar{C}_{in} + A\bar{B}\bar{C}_{in} + ABC_{in}$ $S = C_{in}\underbrace{(\bar{A}\bar{B} + AB)}_{\text{EX-NOR}} + \bar{C}_{in}\underbrace{(\bar{A}B + A\bar{B})}_{\text{EX-OR}}$ $C_o = AB + AC_{in} + BC_{in}$ $\therefore S = C_{in}\overline{(\bar{A}B + A\bar{B})} + \bar{C}_{in}(\bar{A}B + A\bar{B})$ Let $X = \bar{A}B + A\bar{B}$ $\therefore\ S = C_{in}\bar{X} + \bar{C}_{in}X = C_{in} \oplus X$ $\therefore\ S = C_{in} \oplus (\bar{A}B + A\bar{B})$ But $\bar{A}B + A\bar{B} = A \oplus B$ $\therefore\ S = C_{in} \oplus A \oplus B$	**1 Mark**
	 A B C_{in} $S = (A \oplus B) \oplus C_{in}$ AB AC_{in} BC_{in} $C_o = AB + AC_{in} + BC_{in}$ B C_{in} **Fig: Full adder**	**1 Mark**
Ans d.	 Preset (PR) J $\bar{Q}$ 3 R' 1 Q CLOCK K 4 S' Q 2 $\bar{Q}$ Clear (CL) **Fig: Clocked J- K Flip Flop with Preset and Clear**	2 Marks

Preset & Clear:

- In the flip flops when the power switch is turned on, the state of output Q and $\overline{Q}$ is uncertain.
- The outputs can be Q = 0, $\overline{Q}$ = 1 to the other way round.
- But this uncertainity cannot be tolerated in certain applications. In some applications it is necessary to initially set or reset the flip flops.
- This can be practically achieved by adding two or more inputs to a flip flop, called preset (PR) and clear (CLR) inputs.

Function of Preset:
Preset is used to set the flip flop i.e. to make output Q =1.

1 Mark

Function of Clear:
Clear is used to reset the flip flop i.e. to make output Q =0.

1 Mark

Ans e.

Symbol of Ex- OR gate: -

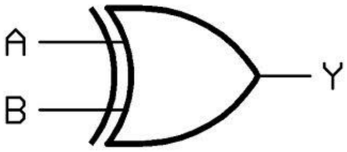


Fig: Symbol of 2 input Ex- OR gate

1 Mark

Input		Output
A	**B**	**Y = A Ex- OR B**
0	0	0
0	1	1
1	0	1
1	1	0

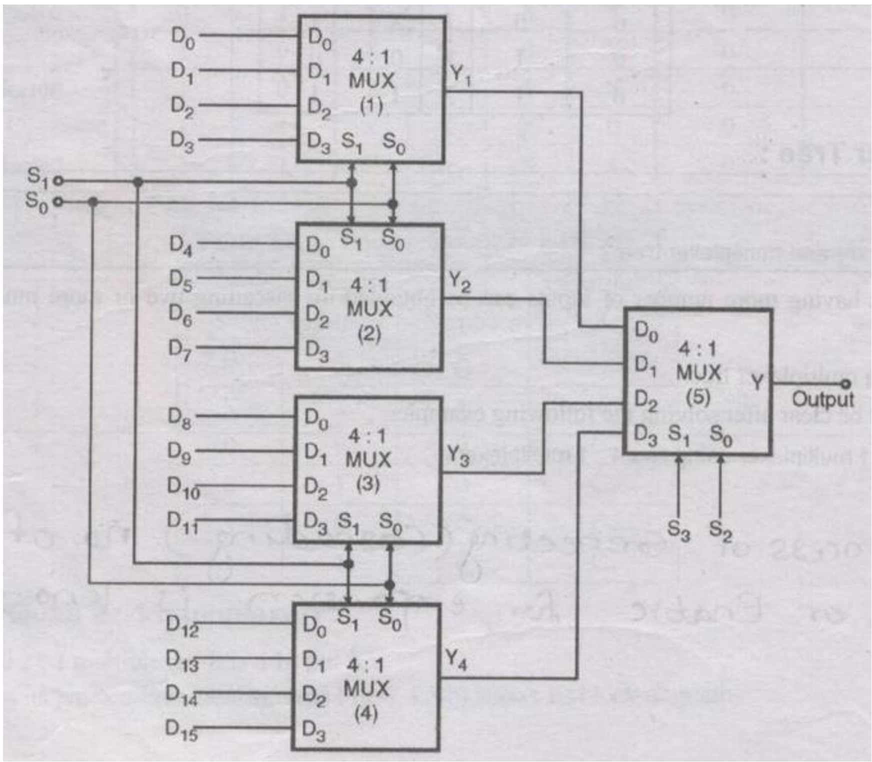
Table : Truth table of 2 input Ex- OR gate

1 Mark

Truth table for logic diagram:

Input			Output
A	B	C	Y = A+ B+ C
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

2 Marks

Ans f.	 **Fig: Multiplexer tree for 16:1 multiplexer using 4:1**	4 Marks

<table>
<tr><td>Q.3</td><td>Attempt any <b><u>FOUR</u></b> of the following:</td><td>16 Marks</td></tr>
<tr><td>Ans a. (i)

(ii)</td><td>4:1 MUX is the combinational circuit since its output depends only on the present inputs.

Decade counter is sequential logic circuit because it uses flip-flops , whose outputs depend on present inputs as well as past inputs/outputs (feedback).</td><td>2 Marks

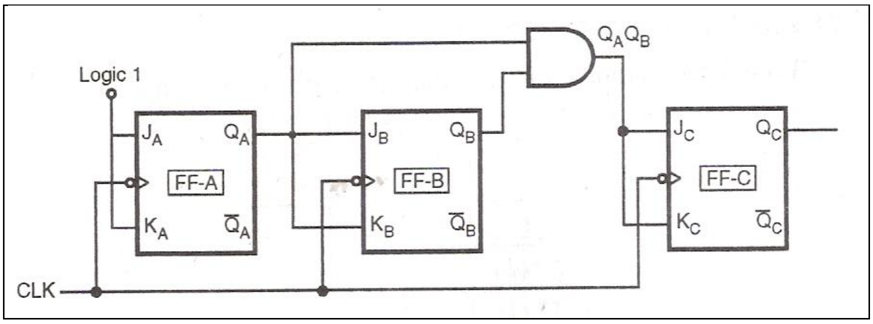
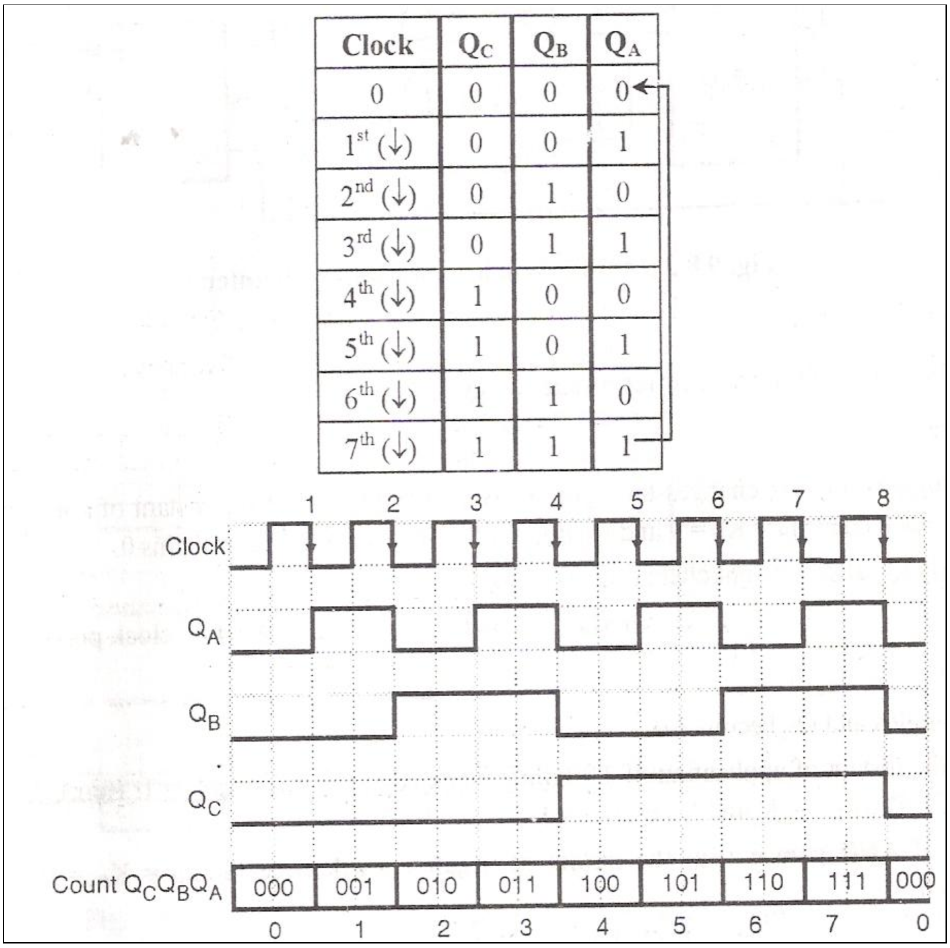
2 Marks</td></tr>
<tr><td>Ans b.</td><td>


Fig: 3 bit synchronous counter.

Truth Table and Waveforms are as shown



<table>
<tr><th>Clock</th><th>Q_C</th><th>Q_B</th><th>Q_A</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0 ←</td></tr>
<tr><td>1st (↓)</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>2nd (↓)</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td>3rd (↓)</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>4th (↓)</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>5th (↓)</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>6th (↓)</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>7th (↓)</td><td>1</td><td>1</td><td>1</td></tr>
</table>

</td><td>2 Marks

1 Mark

1 Mark</td></tr>
</table>

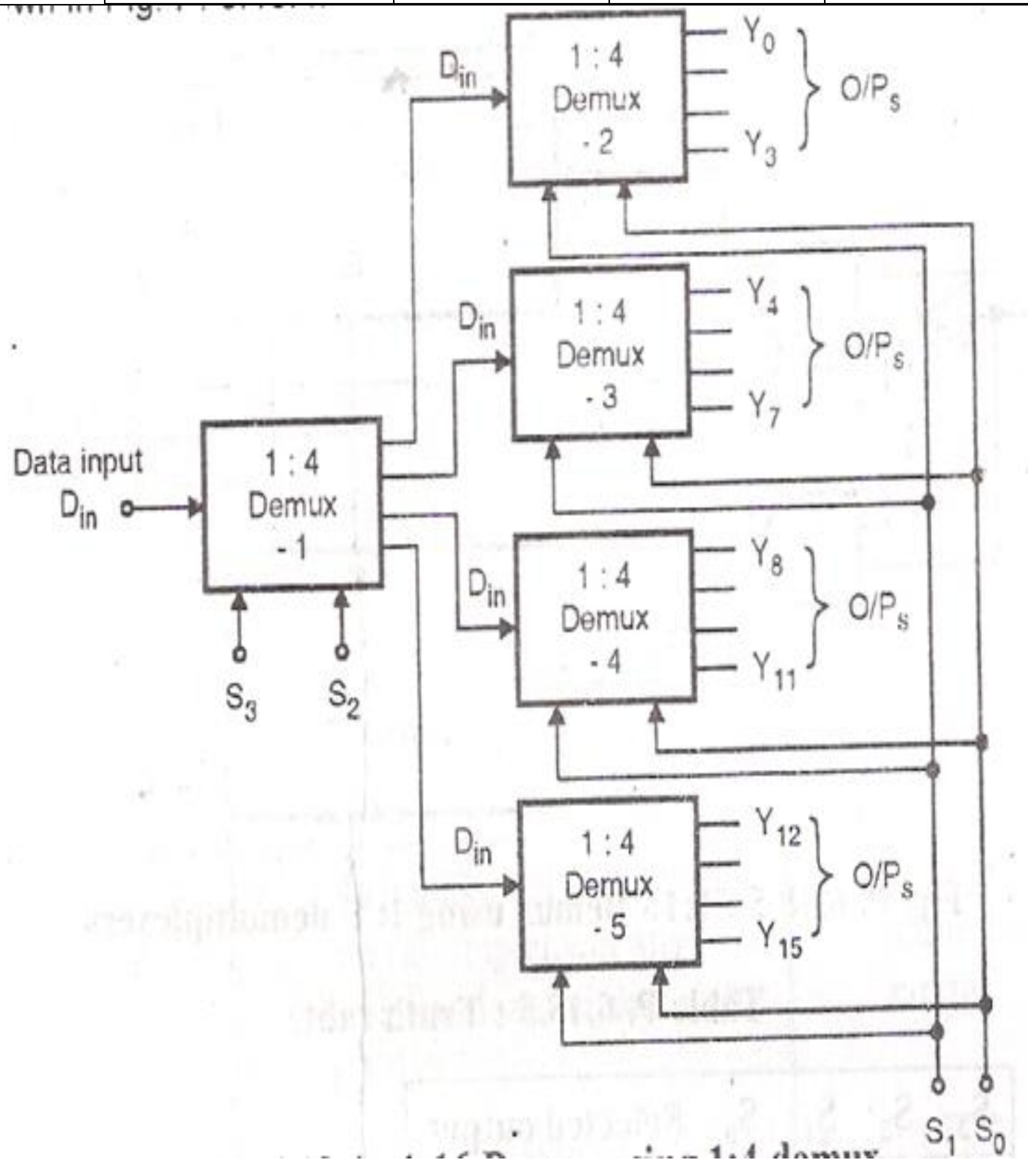
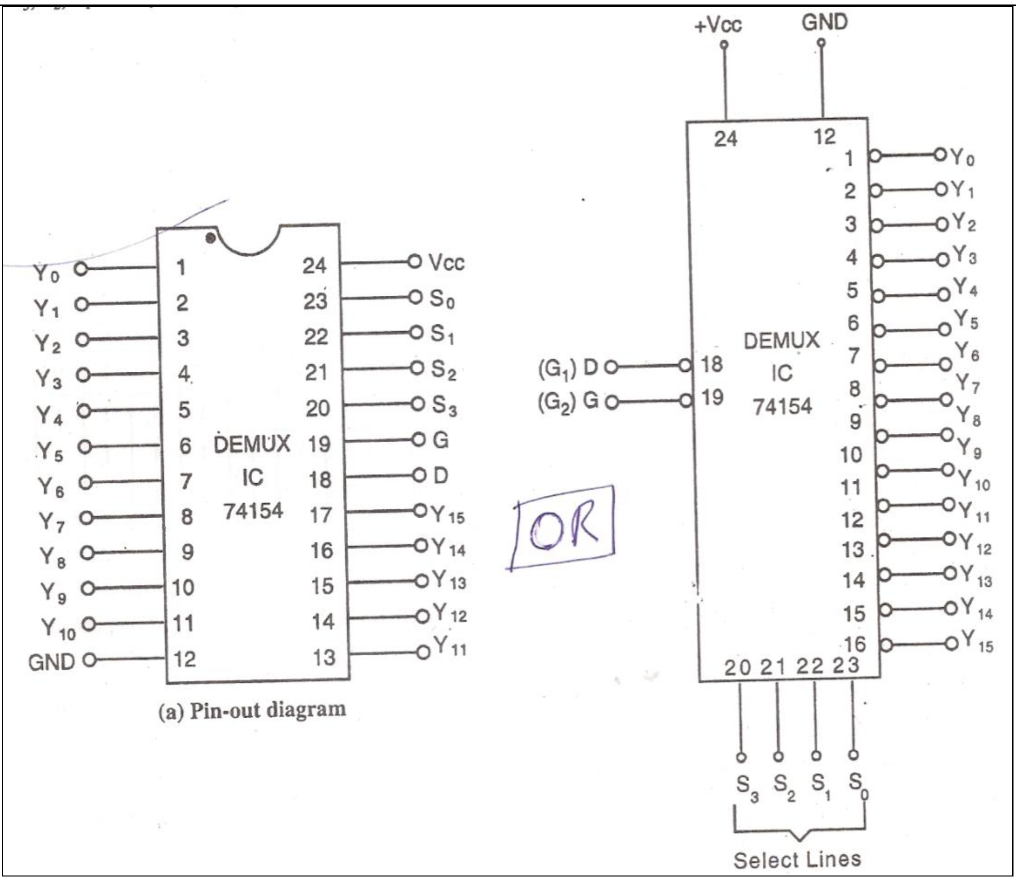
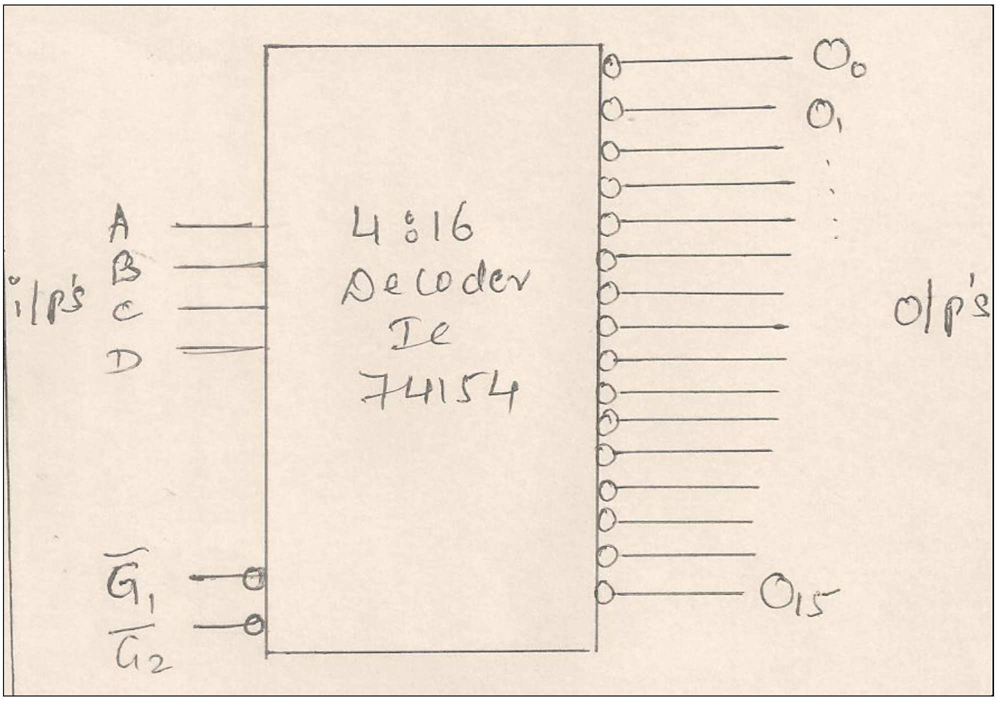
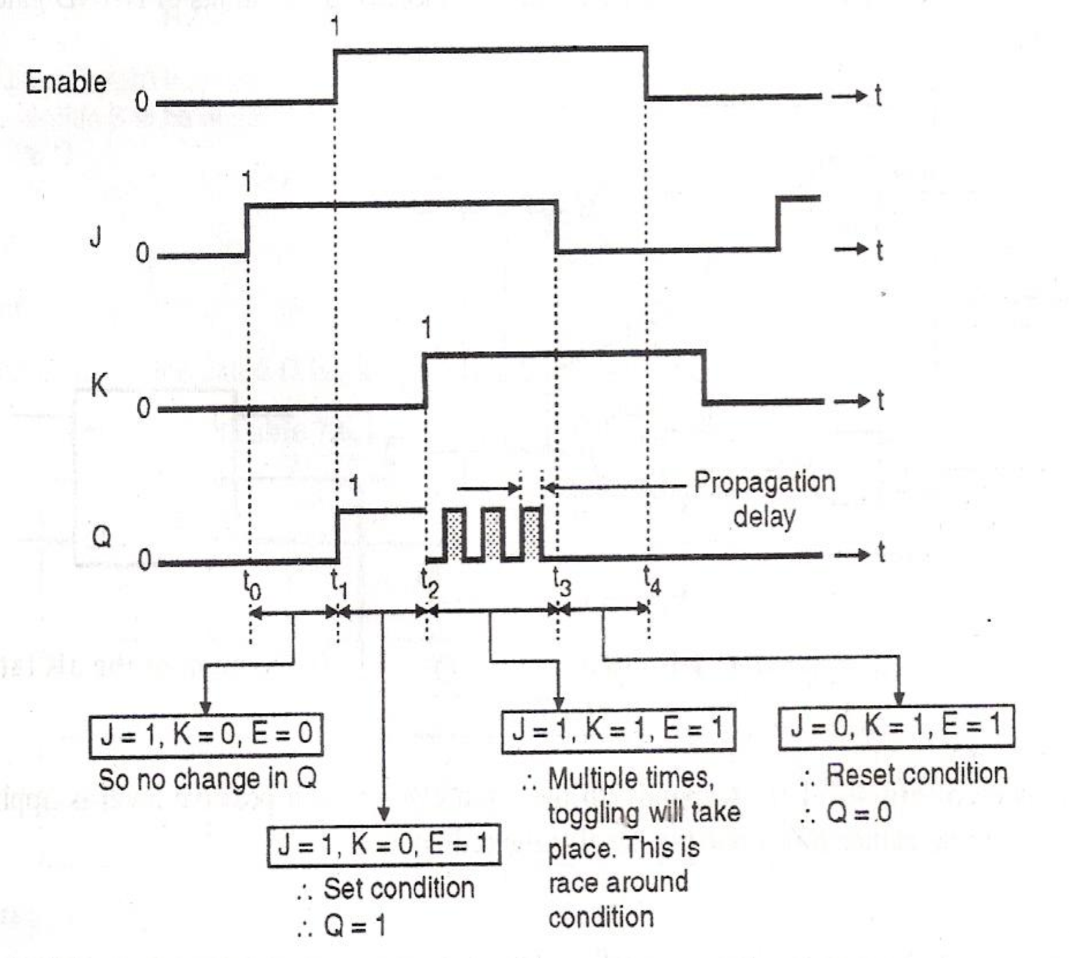
<table>
<tr>
<td>Ans c</td>
<td>
<table>
<tr><td>PARAMETER</td><td>TTL</td><td>ECL</td><td>CMOS</td></tr>
<tr><td>Propagation delay</td><td>35ns</td><td>2ns</td><td>102ns</td></tr>
<tr><td>Fan-out</td><td>10</td><td>25</td><td>>50</td></tr>
</table>
</td>
<td>2 Marks
2 Marks</td>
</tr>
<tr>
<td>Ans d.</td>
<td>


Fig: Realization of 1:16 demux using 1:4 demux
</td>
<td>4 Marks</td>
</tr>
</table>

Ans e.	 Y0 Y1 Y2 Y3 Y4 Y5 Y6 Y7 Y8 Y9 Y10 GND Vcc S0 S1 S2 S3 G D Y15 Y14 Y13 Y12 Y11 DEMUX IC 74154 (a) Pin-out diagram OR +Vcc GND (G1) D (G2) G S3 S2 S1 S0 Select Lines Fig: Pin out diagram of IC 74154	2 Marks
	 i/p's A B C D 4:16 Decoder IC 74154 G1 G2 O0 O1 O15 o/p's Fig: IC 74154 can be used as decoder (4:16 decoder)	2 Marks

Ans f.

Race around condition occurs in J K Flipflop only when J=K=1 and clock/enable is high (logic 1) as shown below-



2 Marks

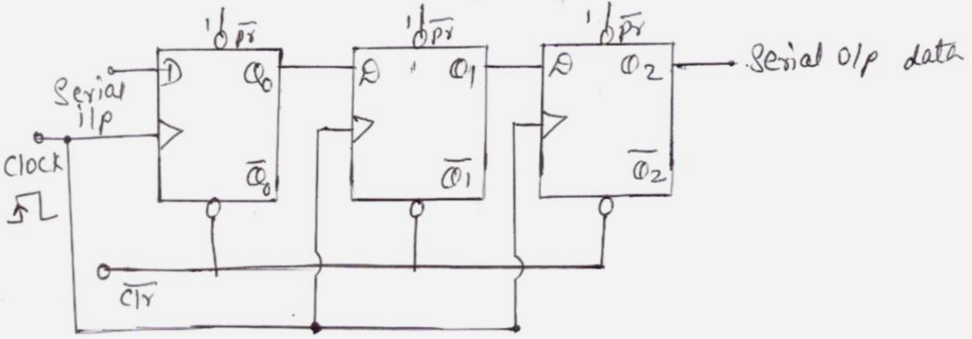
Explanation:-In JK Flip-flop when J=K=1 and when clock goes high, output should toggle (change to opposite state) ,but due to multiple feedback output changes/toggles many times till the clock/enable is high. Thus toggling takes place more than once, called as racing or race around condition. Thus to avoid RAC following methods can be used-

1. Design the clock with time less than toggling time(this method is not economical)
2. Use edge triggering.
3. Use Master Slave J K Flip-flop.

2 Marks

Q4.	Attempt any Four of the following	16Marks
Ans a.	3 bit SISO shift register is as shown below:-	

3 bit SISO shift Register



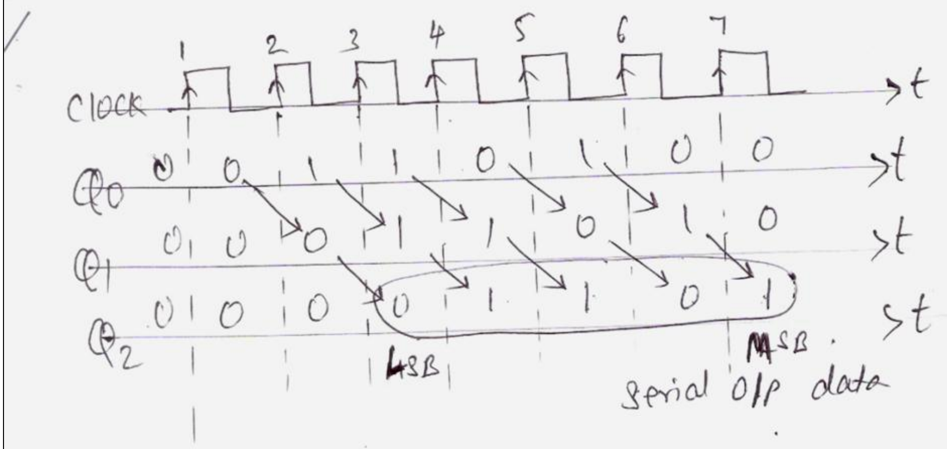
Data = 10110
↳MSB ↳LSB

Truth table.

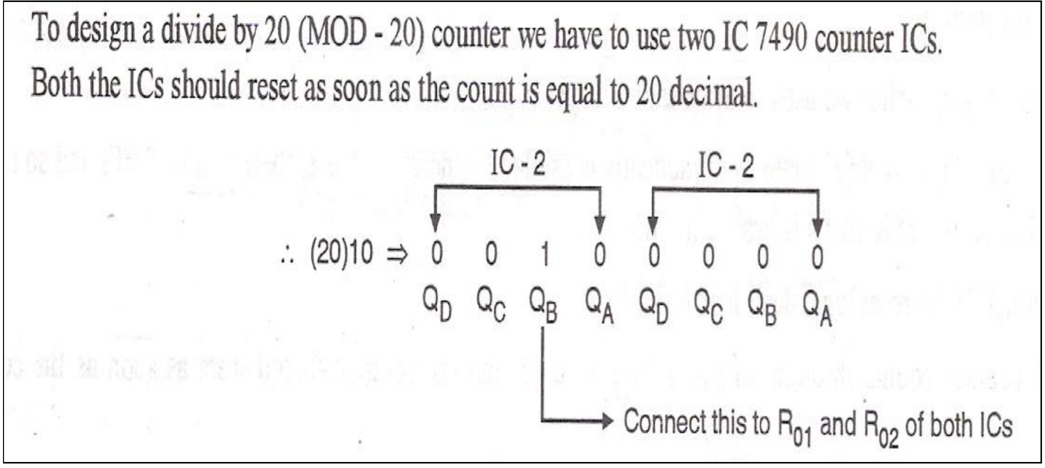
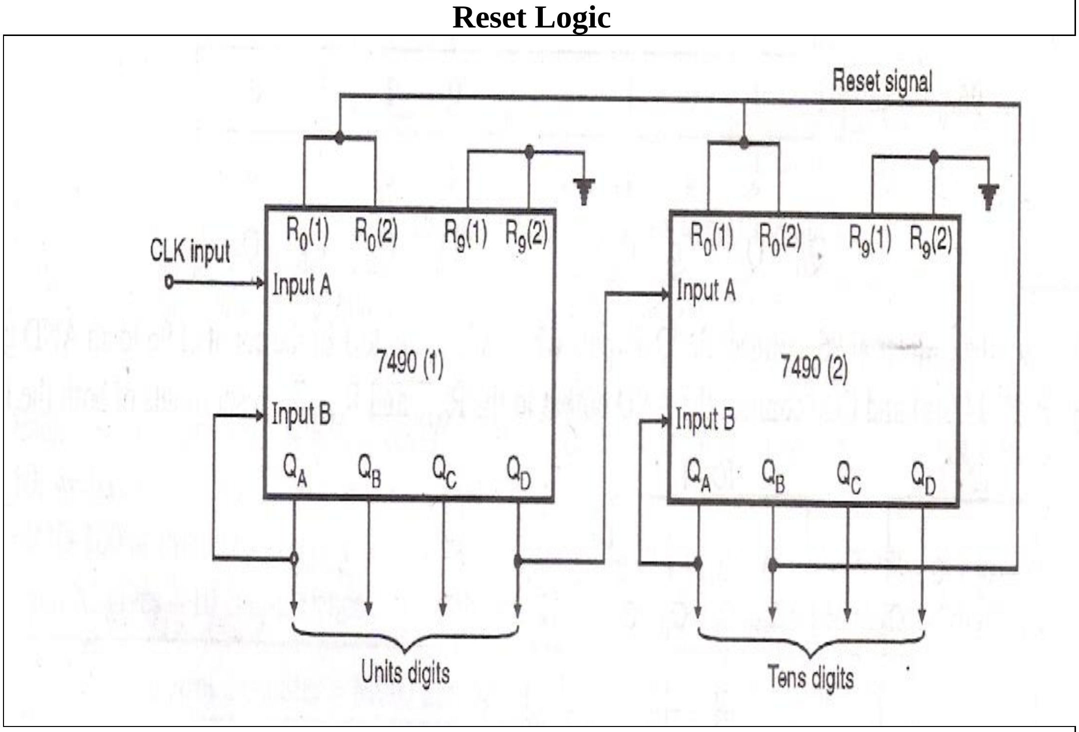
CLK	Q0	Q1	Q2	
0	0	0	0	
1	0	0	0	
2	1	0	0	
3	1	1	0	LSB
4	0	1	1	
5	1	0	1	Serial o/p.
6	0	1	0	
7	0	0	1	MSB

2 Marks

waveforms



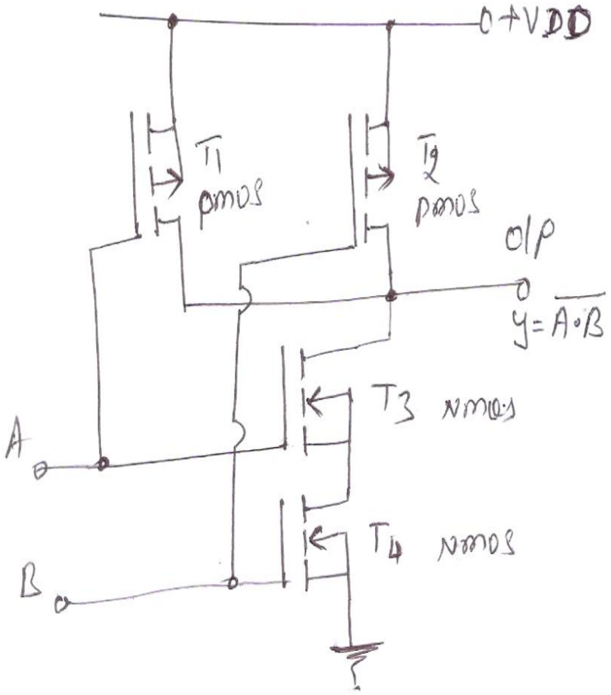
2 Marks

Ans b.	Procedure for design is as shown-  To design a divide by 20 (MOD - 20) counter we have to use two IC 7490 counter ICs. Both the ICs should reset as soon as the count is equal to 20 decimal. IC - 2 IC - 2 ∴ (20)10 ⇒ 0 0 1 0 0 0 0 0 Q_D Q_C Q_B Q_A Q_D Q_C Q_B Q_A → Connect this to R_{01} and R_{02} of both ICs **Reset Logic**	2 Marks
	 Reset signal CLK input $R_0(1)$ $R_0(2)$ $R_9(1)$ $R_9(2)$ Input A 7490 (1) Input B Q_A Q_B Q_C Q_D $R_0(1)$ $R_0(2)$ $R_9(1)$ $R_9(2)$ Input A 7490 (2) Input B Q_A Q_B Q_C Q_D Units digits Tens digits Fig : **Logic diagram of MOD-20 counter using IC 7490.**	2 Marks

Ans c.

CMOS NAND gate

CMOS NAND Gate



Truth Table

A · B	$Y=\overline{A \cdot B}$
0 0	1
0 1	1
1 0	1
1 1	0

2 Marks

Operation:-

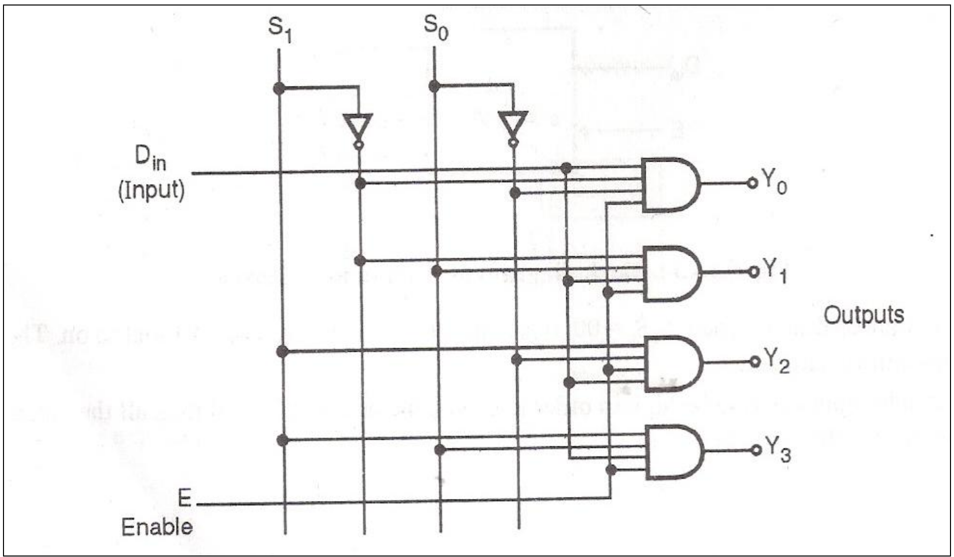
Case 1:-A=B=0
Under this condition, T1 andT2 are in on state while T3 and T4 are in cut off state, so output is at logic 1 level.

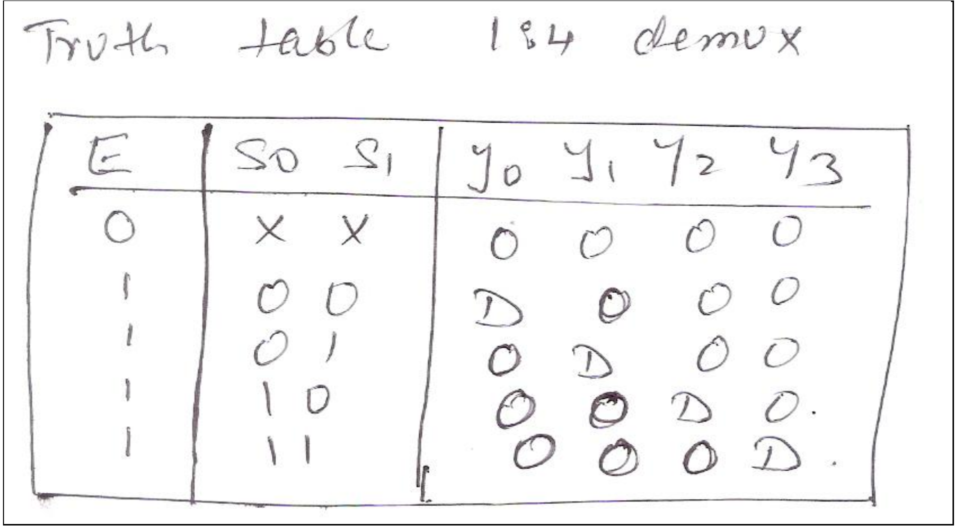
Case 2 and Case 3:- A=0,B=1 OR A=1,B=0
Under this condition either of T1 or T2 will be in on state, while both T3 and T4 will remain in off state, so output will be at logic 1 level.

Case 4:- A=B=1
Under this condition, T1 andT2 are in off state while T3 andT4 are in on off state, so output is at logic 0 level.

2 Marks

<table>
<tr><td>Ans d.</td><td>The four characteristics of ECL logic family are
1. The fan-out capacity is large (about 25) because of its low output impedance.
2. The supply voltage is -5.2 V
3. The power dissipation per gate is-large, about 40 mW.
4. The propagation delay time is 2ns.
5. It has two outputs i.e. normal output and its complement also. Therefore, this avoids the need for inverters.
6. The transistors used are not allowed to get saturated. Therefore, the speed is Highest.</td><td>1 mark each for any four points</td></tr>
<tr><td>Ans e.</td><td>The internal logic diagram of 1:4 demultiplexer and write it's truth table is as shown-

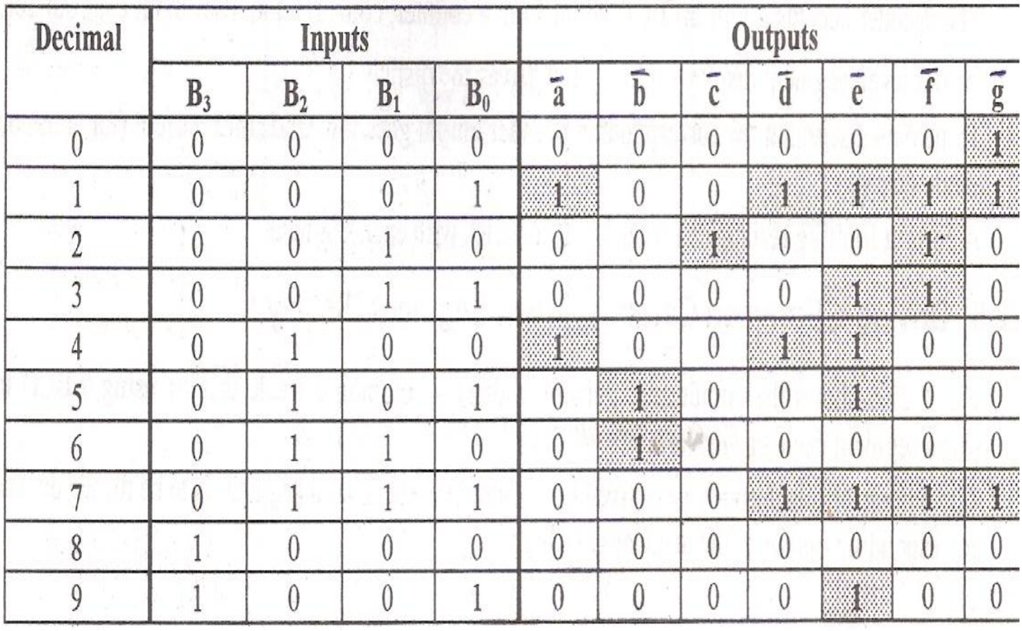



Truth table 1:4 demux

<table>
<tr><th>E</th><th>S0</th><th>S1</th><th>y0</th><th>Y1</th><th>Y2</th><th>Y3</th></tr>
<tr><td>0</td><td>X</td><td>X</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>D</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td>D</td><td>0</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>D</td><td>0.</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>D.</td></tr>
</table>
</td><td>2 Marks
2 Marks</td></tr>
</table>

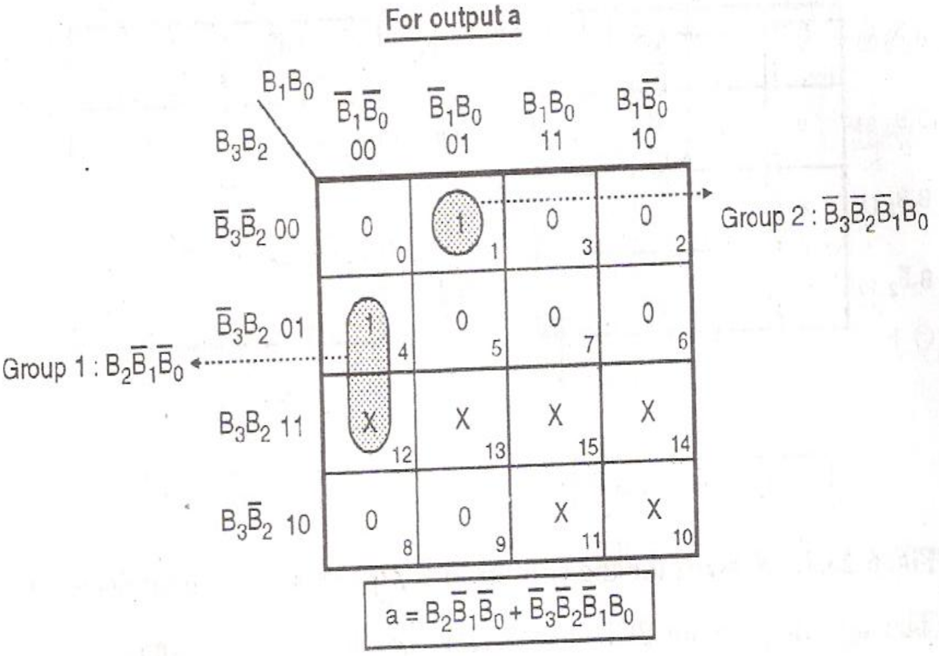
Ans f.

Truth table for BCD to 7 segment code converter is as shown below where only segment
‘a’is to be solved for equation using K map as shown below-



Decimal	Inputs				Outputs						
	B_3	B_2	B_1	B_0	$\bar{a}$	$\bar{b}$	$\bar{c}$	$\bar{d}$	$\bar{e}$	$\bar{f}$	$\bar{g}$
0	0	0	0	0	0	0	0	0	0	0	1
1	0	0	0	1	1	0	0	1	1	1	1
2	0	0	1	0	0	0	1	0	0	1	0
3	0	0	1	1	0	0	0	0	1	1	0
4	0	1	0	0	1	0	0	1	1	0	0
5	0	1	0	1	0	1	0	0	1	0	0
6	0	1	1	0	0	1	0	0	0	0	0
7	0	1	1	1	0	0	0	1	1	1	1
8	1	0	0	0	0	0	0	0	0	0	0
9	1	0	0	1	0	0	0	0	1	0	0

2 Marks



For output a

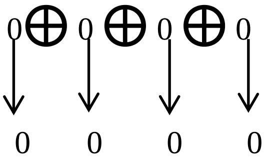
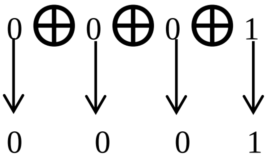
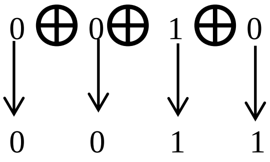
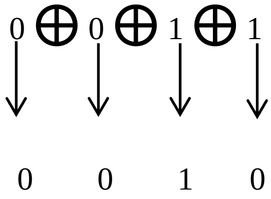
B_3B_2 \ B_1B_0	$\bar{B}_1\bar{B}_0$ 00	$\bar{B}_1B_0$ 01	B_1B_0 11	$B_1\bar{B}_0$ 10
$\bar{B}_3\bar{B}_2$ 00	0 (0)	1 (1)	0 (3)	0 (2)
$\bar{B}_3B_2$ 01	1 (4)	0 (5)	0 (7)	0 (6)
B_3B_2 11	X (12)	X (13)	X (15)	X (14)
$B_3\bar{B}_2$ 10	0 (8)	0 (9)	X (11)	X (10)

Group 2 : $\bar{B}_3\bar{B}_2\bar{B}_1B_0$

Group 1 : $B_2\bar{B}_1\bar{B}_0$

$$\boxed{a = B_2\bar{B}_1\bar{B}_0 + \bar{B}_3\bar{B}_2\bar{B}_1B_0}$$

2 Marks

Q5.	**Attempt any <u>FOUR</u> of the following:**	**16 Marks**
Ans. a	Excess-3 code is derived from the BCD code by adding $(0011)_2$ or	
	$(3)_{10}$ to each code word.	½ Mark
	e.g. (i) Excess-3 code of (0000)BCD = (0000 + 0011)	
	= $(0011)_{XS-3}$	½ Mark
	(ii) Excess-3 code of (0001)BCD = (0001 + 0011)	
	= $(0100)_{XS-3}$	½ Mark
	(iii) Excess-3 code of (0010)BCD = (0010 + 0011)	
	= $(0101)_{XS-3}$	½ Mark
	(iv) Excess-3 code of (0011)BCD = (0011 + 0011)	
	= $(0110)_{XS-3}$	
	Gray code is derived from the BCD code by Ex-ORing bit with next	
	Steps:	
	(i) Write MSB as it is.	
	(ii) Then EX-ORing each Gray bit with next bit till we reach to LSB.	
	(i) Gray code of (0000)BCD = (0000)carry	½ Mark
		
	(ii) Gray code of (0001)BCD = $(0001)_{Gray}$	½ Mark
		
	(iii) Gray code of (0010)BCD = $(0011)_{Gray}$	½ Mark
		
	(iv) Gray code of (0011)BCD = $(0010)_{Gray}$	½ Mark
		

Ans. b

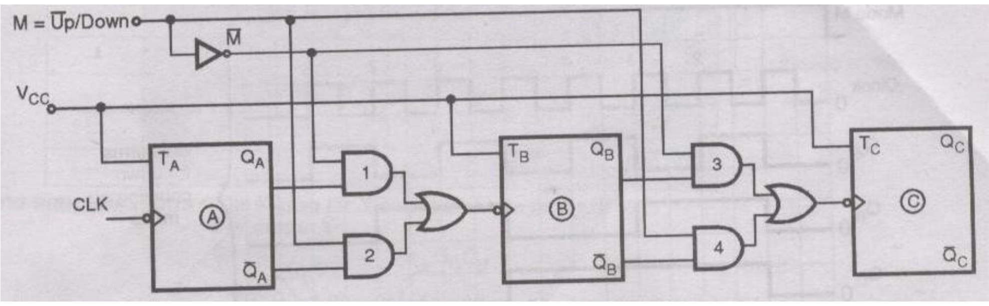


Fig: Logic diagram of 3 bit up-down counter.

2 Marks

Operating Principle:

1. With M=0 (Up counting mode):
   - If M=0 & M=1, then the AND gates 1 and 3 in fig. p. 10.5.1 (a) will be enabled whereas the AND gates 2 and 4 will be disabled.
   - Hence Q_A gets connected to the clock input of FF-B and Q_B gets connected to the clock input of FF-C.
   - These connections are same as those for the normal up counter. Thus, with M=0 the circuit works as an up counter.

2. With M=1 (Down counting mode):-
   - If M=1, then AND gates 2 & 4 in fig. p. 10.5.1 (a) will be enabled whereas the AND gates 1 & 3 are disabled.
   - Hence, Q_A gets connected to the clock input of FF-B and Q_B gets connected to the clock input of FF-C.
   - As discussed earlier, these connections will produce a down counter. Thus, with M=1 the circuit works as down counter.

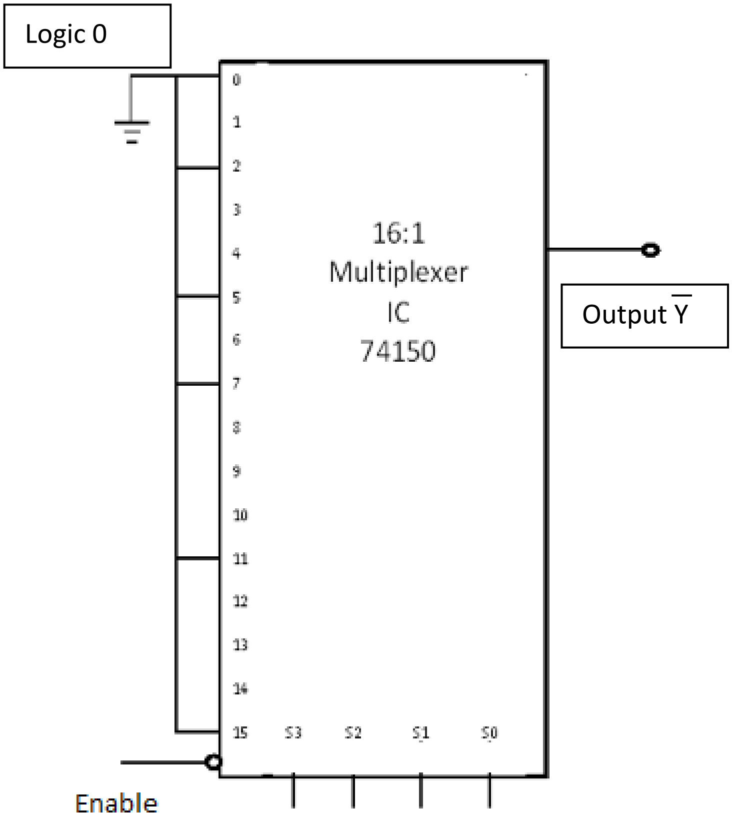
2 Marks

Ans. c

Realization of function
f = **Σ** (0,2,5,7,11,15) using IC 74150:

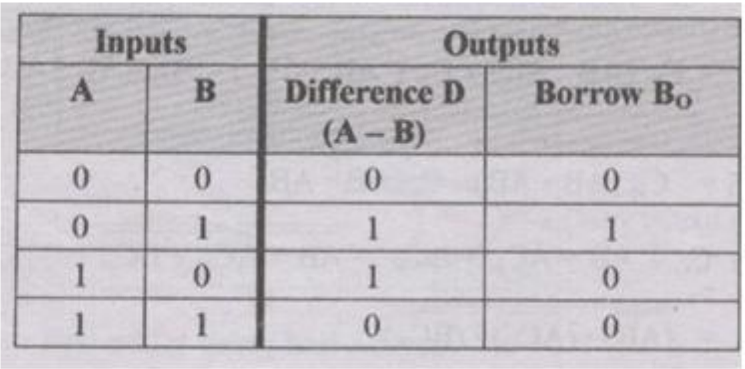
- Given function is in sum of product (SOP) form i.e. output for given minterm is 1 e.g. when 0th line is selected o/p is logic 1.
- Maximum given minterm is 15, Hence, 16:1 multiplexer i.e. (IC 74150) should be used.
- IC 74150 has complemented output (Y), means the output will be equal to the complement of the selected data input. Therefore, connect given minterms to logic 0 i.e. GND (ground). As shown in fig. So at output it will produce Logic 1 for selected data input.

1 Mark for Logic

	 Logic 0 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 S3 S2 S1 S0 16:1 Multiplexer IC 74150 Output $\overline{Y}$ Enable **<u>Note:</u>** Marks are to be given for any correct alternate logic e.g. at I/P instead of logic 0, connect **logic 1**. Hence output $\overline{Y}$ =0 so connect **NOT gate** at O/P to get logic HIGH output.	3 Marks for Logic Diagram

Ans d.

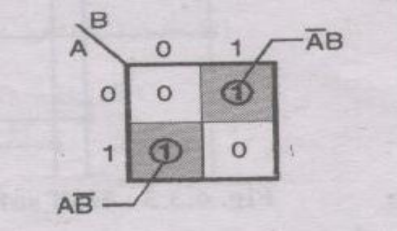
Half subtractor is a combinational logic ckt.it produces difference between two binary bits and produces difference and borrow at the output.



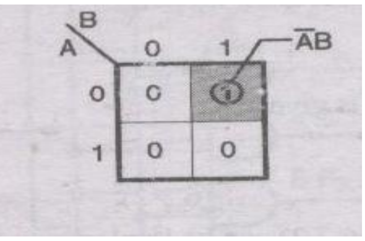
Inputs		Outputs	
A	B	Difference D (A – B)	Borrow B_O
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

Truth table of Half subtractor

1 Mark



K-map for difference



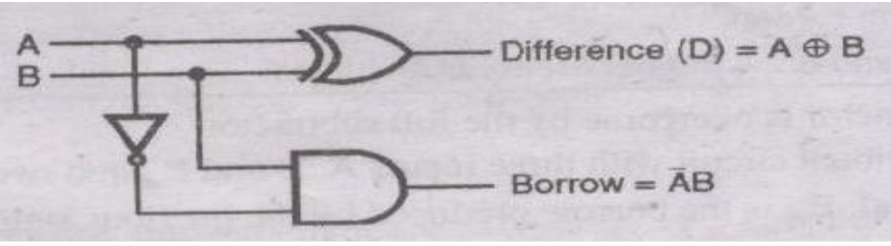
K – map for borrow

2 Marks

Difference D= $\bar{A}$. B +A . $\bar{B}$

= A Ex-OR B

Borrow= $\bar{A}$. B



Half Subtractor Circuit

1 Mark

Ans . e (i)

(17+58)

(17)= 0001 0111 ------ BCD

+ (58)= + 0101 1000 ----- BCD

1

(75) 0110 1111----- Invalid BCD

+

0110 hence add $(6)_{10}$=0110_2

1 1 1

0111 0101

7 5

1 Mark

Ans: (17) + (58) = (75)

1 Mark

(ii)

2-9

Solution: BCD subtraction is performed using one of the following methods.

(a) Nine’s complement method
(b) Ten’s complement method

1 Mark

a) BCD subtraction using 9’s complement method.
Step 1:- obtain 9’s complement of (9)10: 9’s complement of (9)10 us (9-9) = 0.

Step 2:- Add 2 and nine’s complement or (9):

```
  2
+0-------nine’s complement of 9
___
0 2
└──→ Final carry is 0. Hence, result is negative hence take 9’s complement of
the result.
```

Step 3:-
Take 9’s complement of the result

```
 9---1001
-2---0010
      11   ------result obtained in step 2
__    _____
 7   0111------final answer
```

1 Mark

(2)– (9)= (-7)

OR

OR

b) BCD subtraction using 10’s complement method.
Step 1:- obtain 10’s complement of (9): 9’s complement of (9)+1= (9-9) = 0+1=1.

1 Mark

Step 2:- Add 2 and tens’s complement of(9):

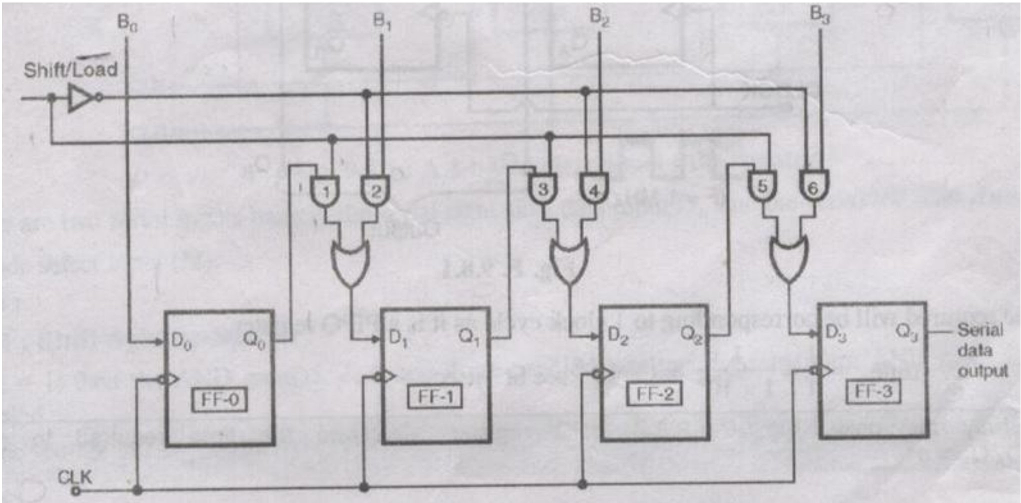
```
  2
+1-------tens’s complement of 9
___
0 3
└──→ Final carry is 0. Hence, result is negative hence take 10’s complement of
the result.
```

Step 3:-
Take 10’s complement of the result =9’s complement of result+1
(9-3)+1= 6+1=7

1 Mark

(2)– (9)= (-7)

Ans. f



2Marks

Fig: Parallel in series out shift register

Working Principle:

1Mark

Load mode:

- When the shift/Load =0; AND gates 2,4,& 6 become active.
- They will pass B_1, B_2, B_3 bits to the corresponding flip flop.
- On the low going edge of clock, the binary bits B_0, B1, B_2, B_3 will get loaded into the corresponding flip flop.
- Thus Parallel loading takes place.

Shift mode:

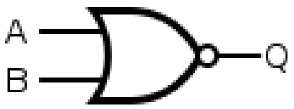
1 Mark

- When the shift/Load =1; AND gates 2,4,& 6 become inactive. Hence Parallel loading of data becomes impossible.
- AND gates 1,3 & 5 become active.
- Therefore the shifting of data takes place from left to right bit by bit on application of clock pulse.

Q6. **Attempt any FOUR of the following:** **16 Marks**

Ans. (a)

- Symbol of the 2 input NOR gate.



1½ Mark

- Truth table

Input		**Output**
A	**B**	**$Q = \overline{A+B}$**
0	0	1
0	1	0
1	0	0
1	1	0

1½ Mark

- Logical Output Equation:

$Q = \overline{A+B}$

1 Mark

Ans. (b)

(i) It is mod-6 (divide by 6) asynchronous (Ripple) up counter.

2 Marks

ii) Truth table of divide by 6 asynchronous up counter.

CLK	Q_C	Q_B	Q_A	Y output	Count	
Initially	0	0	0	1	0	
1	0	0	1	1	1	
2	0	1	0	1	2	
3	0	1	1	1	3	
4	1	0	0	1	4	
5	1	0	1	1	5	
6	1	1	0	0	0	

2 Marks

Ans. (c)

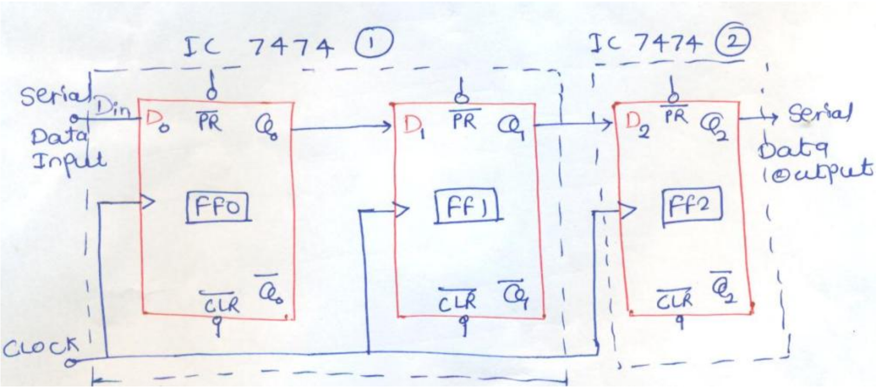
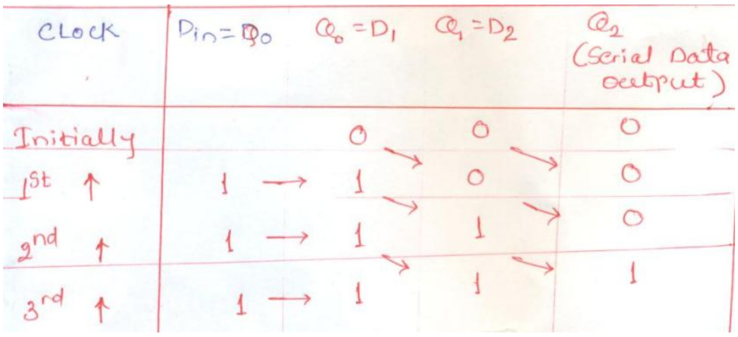


Fig: 3-bit SISO shift register using IC 7474

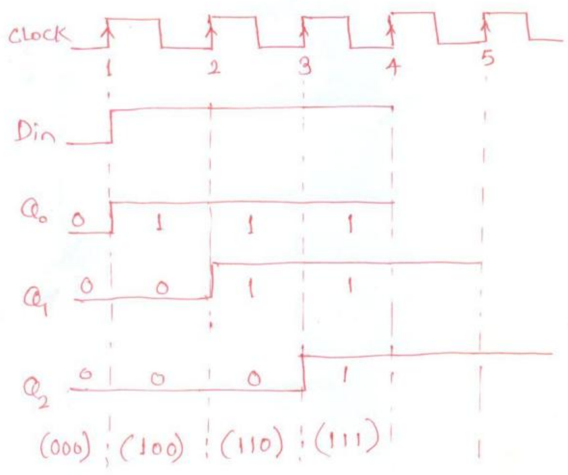
1 1/2 Marks



Clock	$D_{in} = Q_0$	$Q_0 = D_1$	$Q_1 = D_2$	Q_2 (Serial Data output)
Initially		0	0	0
1st ↑	1 →	1	0	0
2nd ↑	1 →	1	1	0
3rd ↑	1 →	1	1	1

Truth table of SISO shift register using IC 7474

1 Mark



Waveform for SISO shift register using IC 7474

1 $^{1/2}$ Marks

Ans (d)

Truth table.

I/P	Select lines		Output of demux (Inverted)	Final output
	A	B	Y3 Y2 Y1 Y0	Y
1	0	0	1 1 1 0	1
1	0	1	1 1 0 1	1
1	1	0	1 0 1 1	1
1	1	1	0 1 1 1	1

2 Marks

NAND logic: If one of the input to NAND gate is logic ‘0’ then output is ‘1’

The given is diagram is 1:4 De multiplexer with the input Logic 1 and outputs are inverted. And outputs are connected as inputs to Four input NAND gate.
Therefore as shown in truth table
For logic High input ,Select lines A= B=0 ; $\overline{Y0}$ is selected and we will get Logic 0 on it as the output is inverted.
Hence the output status is $\overline{Y3}$ $\overline{Y2}$ $\overline{Y1}$ $\overline{Y0}$ = 1 1 1 0.
Which is applied to NAND gate, Hence the output will be at Logic High .
Similarly for the Select lines AB= 0 1 , 1 0 , 1 1 one of the outputs ($\overline{Y1}$ $\overline{Y2}$ $\overline{Y3}$) is remaining 0. Hence according to logic of NAND gates we are getting output Y always 1.

2 Marks

Ans. (e)

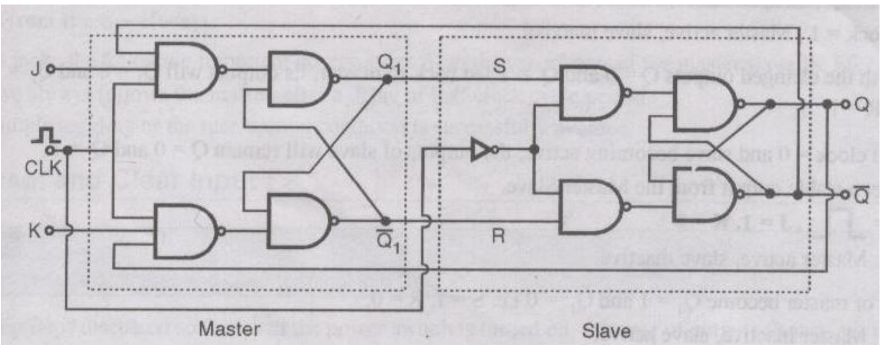
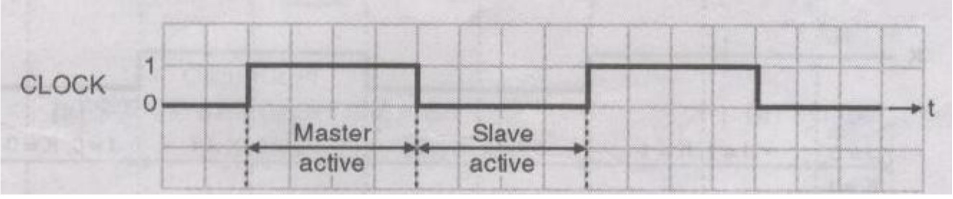


Fig: logic diagram of master Slave J K flip Flop

2 Marks

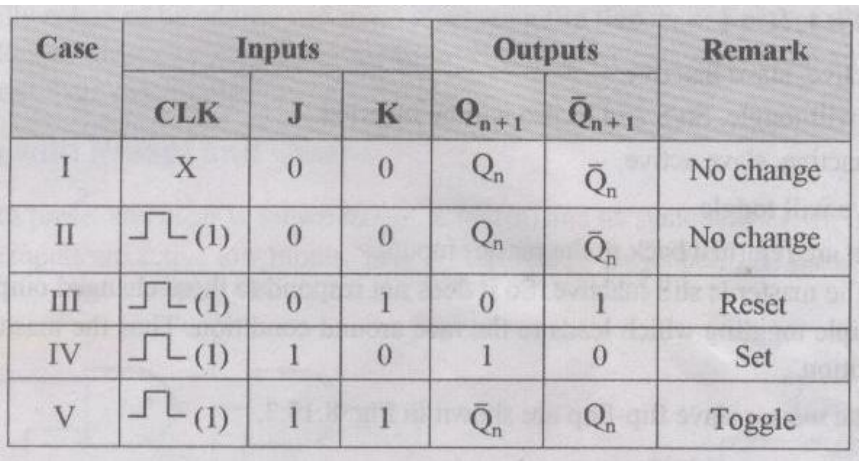
Operation :
In the positive half cycle of the clock , the master is active and in the negative half cycle the slave is active as shown in figure below

2 Marks



Refer the truth table below:

Case I : Clock = x, J=K=0
1. For **clock = 1** the master is active and slave is inactive.
As J=K=0 , the outputs of master Q and $\overline{Q}$ will not change. Hence the S and R inputs to the slave will remain unchanged.
2. As soon as **clock = 0** , slave becomes active and master is inactive but since S and R have not changed the slave outputs will also remain unchanged,
Hence the out put **will not change** if J = k = 0



Case	Inputs			Outputs		Remark
	CLK	J	K	Q_{n+1}	$\overline{Q}_{n+1}$	
I	X	0	0	Q_n	$\overline{Q}_n$	No change
II	⎍ (1)	0	0	Q_n	$\overline{Q}_n$	No change
III	⎍ (1)	0	1	0	1	Reset
IV	⎍ (1)	1	0	1	0	Set
V	⎍ (1)	1	1	$\overline{Q}_n$	Q_n	Toggle

Truth table of master slave JK flip flop

Case II: Clock=1 , J=K=0---- same as case .I

Case III: Clock= ⎍
J=0 and K= 1.

Clock =1: Master active , slave inactive.
Hence outputs of the master become Q= 0 , $\overline{Q}$ = 1

That means S = 0 and R=1 ----**Reset state**

Clock =0: Slave Active, master Inactive

Hence outputs if the slave becomes Q=0, $\overline{Q}$ = 1.

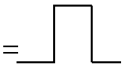
Again if **clock = 1:** Master active; slave inactive.

i.e. with changed outputs Q= 0 , $\overline{Q}$ = 1 fed back to master, its outputs will Q= 0 , $\overline{Q}$ = 1. That means S=0 and R=1.

Hence with clock=0 and slave become active. The outputs of slave will remain Q= 0 , $\overline{Q}$ = 1 ----**Reset state**

Case IV: with positive half cycle of clk and J=1, K=0 Master become active. **i.e.** Q= 1 , $\overline{Q}$ = 0.------ **Set state**

For Negative half cycle of clock Slave become active. And copies the status of master i.e. Q= 1 , $\overline{Q}$ = 0------- **Set state**

Case V: Clock = 

J= K= 1

Clock =1: Master active slave inactive.

Therefore outputs of Master will toggle.

So S & R also will be inverted

Clock=0: Master Inactive , Slave Active.

Therefore outputs of Slave will toggle.

These changed outputs are returned back to the master inputs.

But since clock=0, the master is still inactive. So it does not respond to these changed outputs.

This avoids the multiple toggling which leads to the race around condition.

Thus Master –slave flip flop will avoid race around condition.

Ans. f

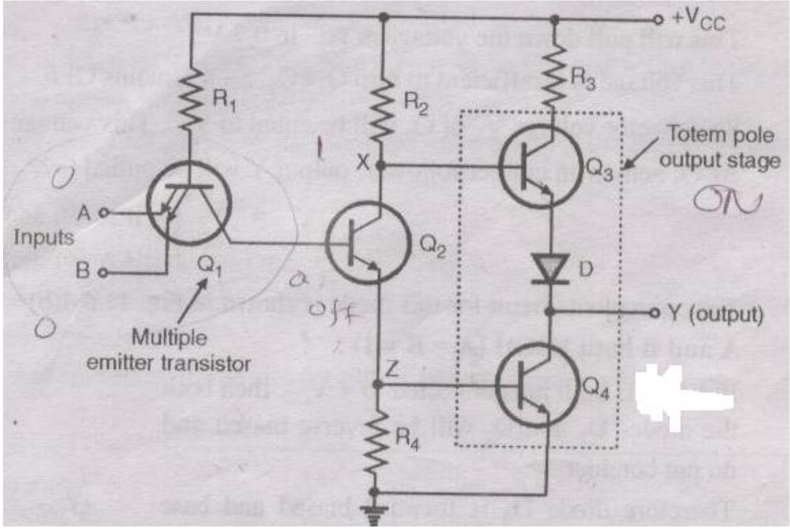


Fig: Operating principle of TTL NAND gate

2 Marks

Input		**Output**
A	**B**	$\mathbf{Y = \overline{A.B}}$
0	0	1
0	1	1
1	0	1
1	1	0

Operating Principle:

2 Marks

1. **A and B both LOW (A = B =0):**

- If A and B both are connected to ground, then both the B- E junctions of transistors Q_1 are forward biased.
- Hence diodes D_1 and D_2 will conduct to force the voltage at point C to 0.7 V.
- This voltage is insufficient to forward bias base emitter junction of Q_2. Hence **Q_2 will remain OFF.**
- Therefore its collector voltage V_x rises to $V_{cc.}$
- As transistor Q_3 is operating in the emitter follower mode, output Y will be pulled up to high voltage.
  Therefore, Y =1 (HIGH) ….. For A =B=) (LOW)

2. **Either A or B LOW (A =0, B=1 or A=1, B=0):**

- If any one input (A or B) is connected to ground with the other terminal left open or connected to +Vcc, then the corresponding diode (D1 & D2) will conduct.
- This will pull down the voltage at "C" to 0.7 V.
- This voltage is insufficient to turn ON **Q2. So it remains OFF.**
- So collector voltage Vx of Q2 will be equal to Vcc. This voltage acts as base voltage for Q3.
- As Q3 acts as an emitter follower, output Y will be pulled to Vcc.
  Y= 1 ---- if A=0 and B=1
  -----if A=1 and B=0

3. **A and B both HIGH (A=B=1):-**

- If A and B both are connected to +Vcc, then both the diodes D1 & D2 will be reverse biased and do not conduct.
- Therefore diode D3 is forward biased and base current is supplied to transistor Q2 via R1 and D3.
- As Q2 conducts, the voltage at X will drop down and **Q3 will be OFF,** whereas voltage at Z (across R3) will increase **to turn ON Q4.**
- As Q4 goes into saturation, the output voltage Y will be pulled down to a low voltage, **Y=0.**