CHAPTER 1

Number System and Codes

Q1. Write the radix of binary, octal, decimal and hexadecimal number system. [2M]

Radix of: Binary – 2

Octal - 8

Decimal - 10

Hexadecimal -16

Q2. List the binary,octal and hexadecimal numbers for decimal no. 0 to 15. [2M]

DECIMAL	BINARY	OCTAL	HEXADECIMAL
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

Q3. Convert i) (255)10 = (?)16 = (?)8. Ii) (157)10 = (?)BCD = (?) Excess3 [4M]

$(255)_{10} = (FF)_{16}$

16 | 255	F (15)	↑
15	F	

$(255)_{10} = (377)_8$

8 | 255	7	↑
8 | 31	7	
3	3	

(ii) $(157)_{10} = (000101010111)_{BCD} = (010010001010)_{Excess3}$

$(157)_{10} = (000101010111)_{BCD}$

1 5 7
0001 0101 0111

$(000101010111)_{BCD} = (010010001010)_{Excess3}$

```
  11  111  111
 0001 0101 0111
+0011 0011 0011
 0100 1000 1010
```

Q4. Subtract using 2's compliment method (35)10 – (5)10 [6M]

Step 1 – Obtain binary equivalent of (35)10& (5)10 & then take 2's compliment of (5)10 .
i.e. (35)10 = (100011)2
(5)10 = (101)2
2's compliment of (5)10 = (000101)2 = 111010 □□1's compliment
+ 1

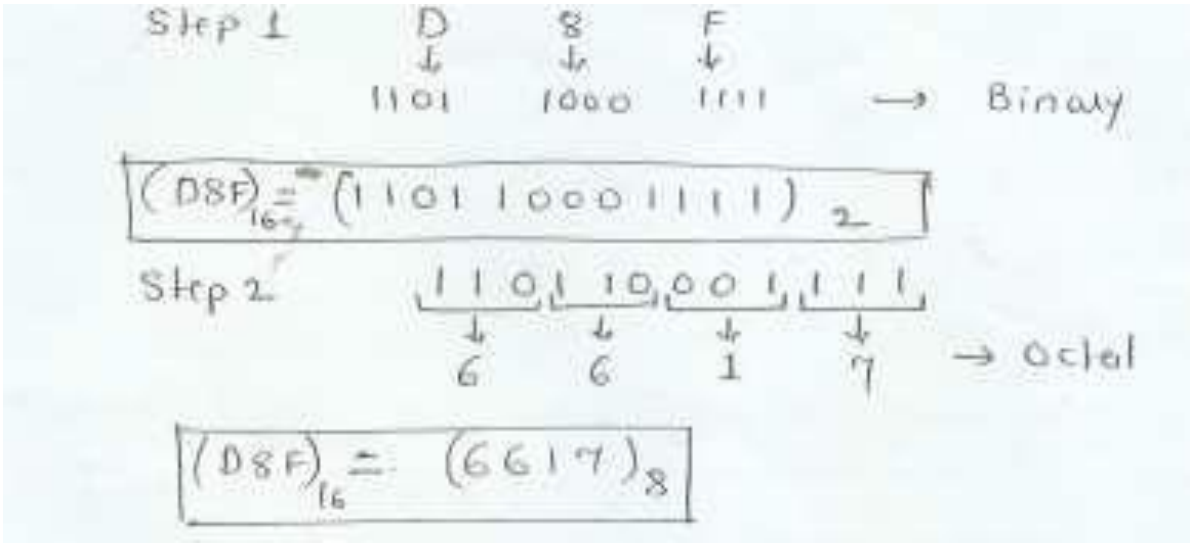
(111011)2□□2's Compliment

Step 2 - Now add (100011)2 and (111011)2

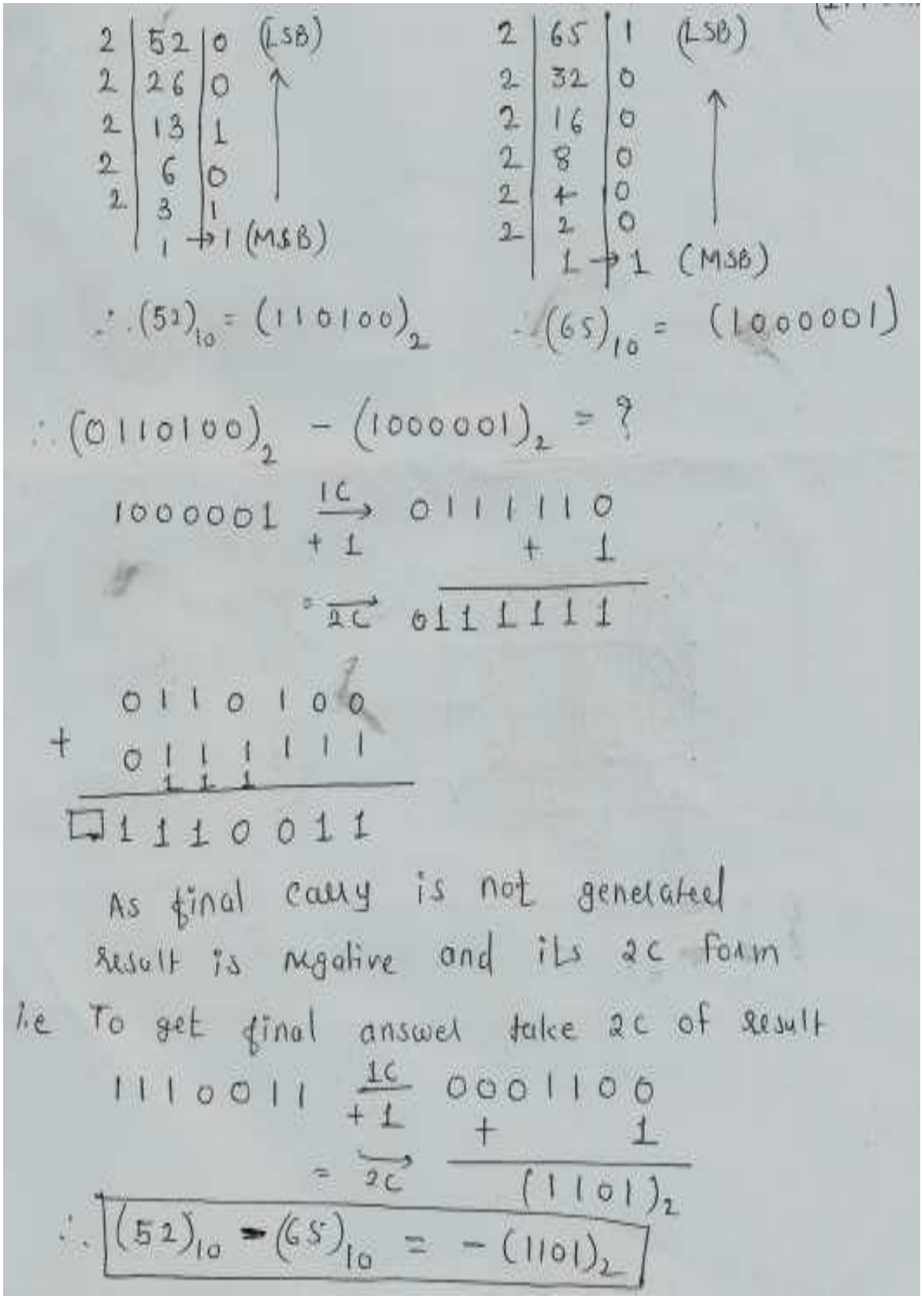
```
  100011
+ 111011
----------
 1011110
```

Carry 1 is generated so answer is in positive form, so will discard the carry generated
Therefore final answer will be (011110)2 = (30)10

Q5. Convert $(D8F)_{16}$ into binary and octal. [2M]



Q6. Perform the subtraction using 2'S Complement methods. (52)10 – (65)10 [4M]



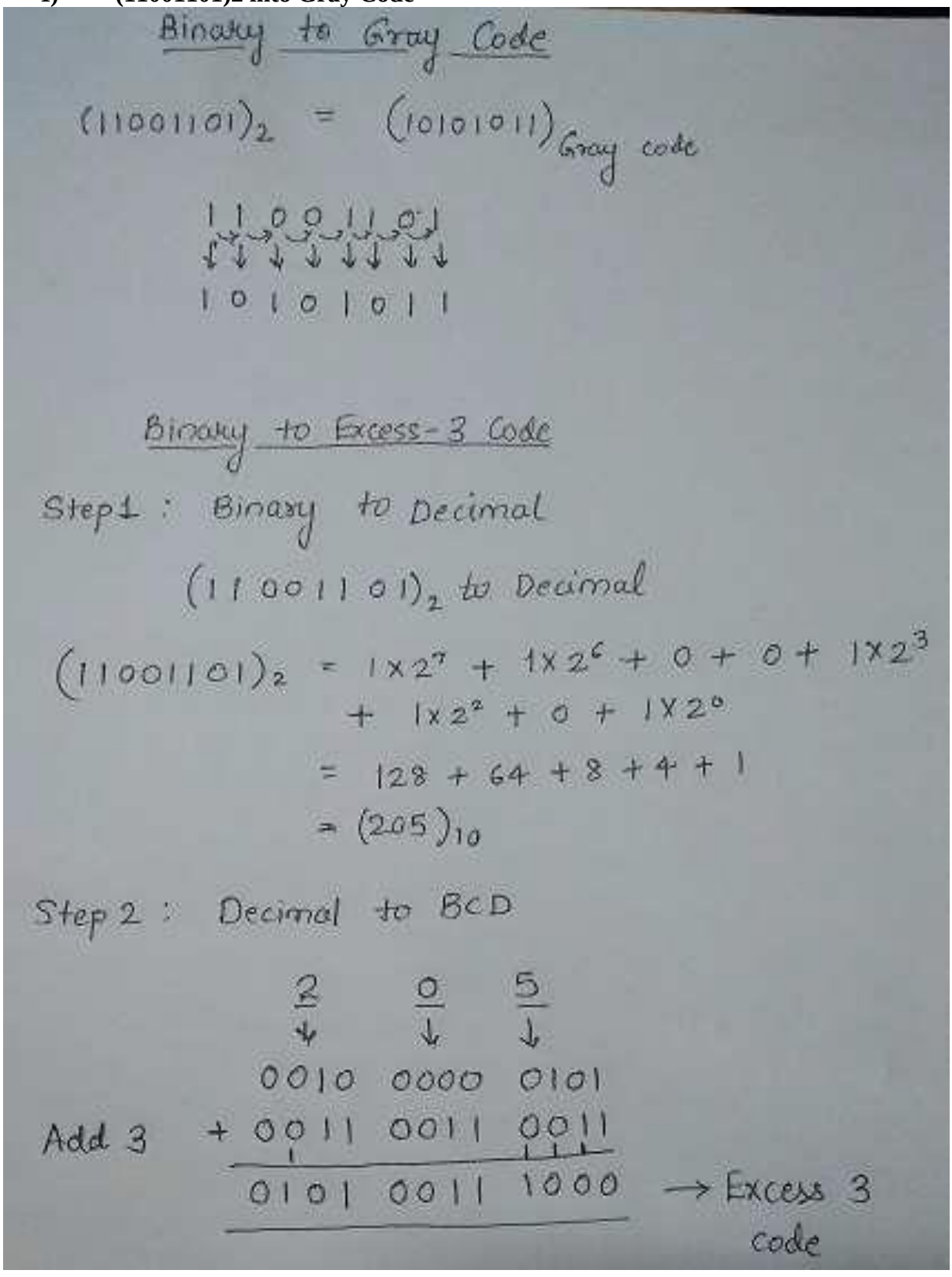
Q7. **[6M]**

I) Convert the following binary number (11001101)2 into Gray Code and Excess-3 Code.

II) Perform the BCD Addition. (17)10 + (57)10

III) Perform the binary addition. (10110 ● 110)2 + (1001 ● 10)2

I) (11001101)2 into Gray Code



II) Perform the BCD Addition. (17)10 + (57)10

$(17)_{10}$ 0001 0111
$(57)_{10}$ + 0101 0111 -------(1/2 M)
0110 1110
Valid Invalid
BCD BCD ----------(1/2 M)
ADD 0110 TO Invalid BCD

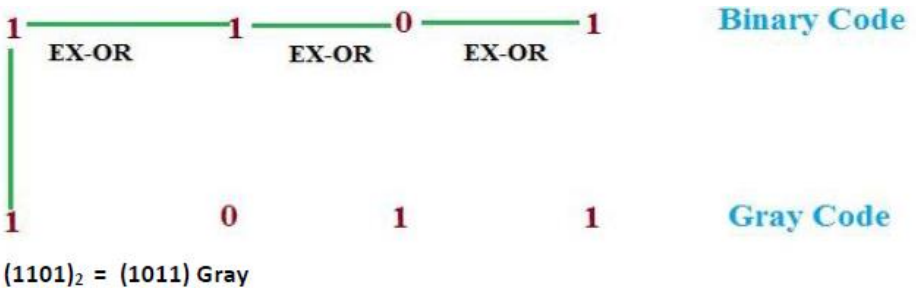
1 11
0110 1110
+ 0000 0110
01110100 ----------(1/2 M)
7 4
$= (74)_{10}$ ----------(1/2 M)

III) Perform the binary addition. (10110 ● 110)2 + (1001 ● 10)2

$10110.110)_2 - (1001.10)_2 = (100000.010)_2$

11111
10110.110
+ 1001.10
100000.010

Q8. Write the gray code to given no.(1101)2 =(?) Gray. [2M]



$(1101)_2$ = (1011) Gray

Q9. Convert: (i) (AD92.BCA)16= (?)10 = (?)8 = (?)2 [4M]

$(AD92.BCA)_{16}$
$= (10 \times 16^3) + (13 \times 16^2) + (9 \times 16^1) + (2 \times 16^0) + (11 \times 16^{-1}) + (12 \times 16^{-2}) + (10 \times 16^{-3})$
$= 40960 + 3328 + 144 + 2 + 0.6857 + 0.046875 + 0.00244$
$= (44434.7368)_{10}$

$(AD92.BCA)_{16} = (1010\ 1101\ 1001\ 0010.1011\ 1100\ 1010)_2$

$(AD92.BCA)_{16}$ = (1010 1101 1001 0010.1011 1100 1010)2
= (001 010 110 110 010 010.101 111 001 010)2
$= (126622.5712)_8$

Q10. Subtract the given number using 2's compliment method: **[4M]**
(i) $(11011)_2 - (11100)_2$
(ii) $(1010)_2$ - $(101)_2$

(i)$(11011)_2 - (11100)_2$
Now,
2's complement of (11100)2= 1's complement of (11100)2+1
1's complement of (11100)2 = (00011)2

2's complement = 00011+1 = 00100

Therefore,

$$\begin{array}{r} 1\ 1\ 0\ 1\ 1 \\ + \\ 0\ 0\ 1\ 0\ 0 \\ \hline 1\ 1\ 1\ 1\ 1 \end{array}$$

There is no carry it indicates that results is negative and in 2's complement form i.e.$(11111)_2$.

Therefore, for getting true value i.e.(+1) take 2's complement of (11111) is

1's complement + 1

= 00000 + 1

Ans= $(00001)_2$

Ans: $(11011)_2 - (11100)_2$ = 2's complement of $(11111)_2 = (-1)_{10}$

ii) Subtract $(1010)_2$ - $(101)_2$using 2's complement binary arithmetic.

2's complement of $(\mathbf{0101})_2$ = 1's complement of $(\mathbf{0101})_2$ +1

1's complement of $(\mathbf{0101})_2 = (1010)_2$

2's complement = 1010+1 = 1011

Therefore,

$$\begin{array}{r} 1\ 0\ 1\ 0 \\ + \\ 1\ 0\ 1\ 1 \\ \hline 1\ 0\ 1\ 0\ 1 \end{array}$$

(1 is carry)

There is carry ignore it, which indicates that results is positive i.e.(+5)

= $(0101)_2$

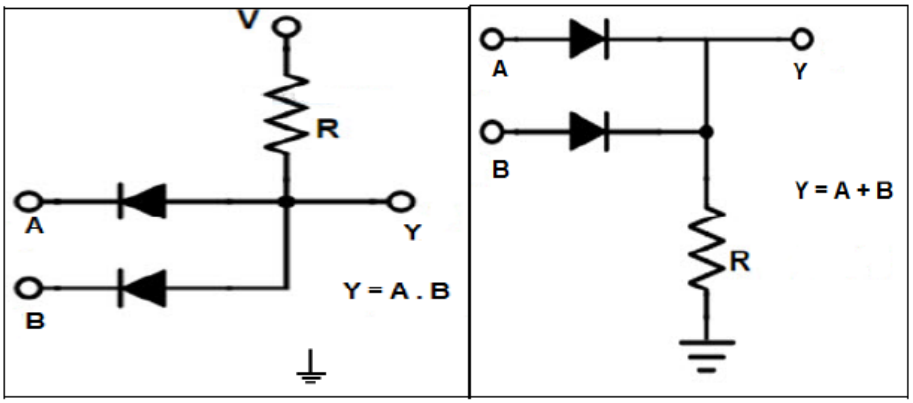
Ans: $(1010)_2$ - $(101)_2 = (0101)_2 = (+5)_{10}$

CHAPTER 2

Logic Gates and Logic Families

Q1. Draw the circuit diagram for AND and OR gates using diodes. **[2M]**

Diode AND gate :Diode OR gate :



Q2. Draw the symbol, truth table and logic expression of any one universal logic gate. Write reason why it is called universal gate. **[4M]**

Universal Gates: NAND or NOR Symbol:

Truth table:

A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

Q3. Compare TTL and CMOS logic families on the basis of following: **[4M]**

(i) Propagation delay
(ii) Power Dissipation
(iii) Fan-out Of Basic gate

Parameter	CMOS	TTL
Propagation delay	70-105 nsec/more than TTL	10 nsec/Less than CMOS
Power Dissipation	Less 0.1 mW/Less than TTL	More 10 mW/ More than CMOS
Fan-out	50/More than TTL	10/Less than CMOS
Basic gate	NAND/NOR	NAND

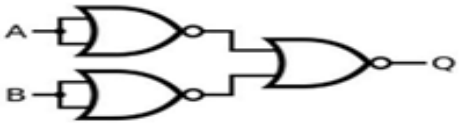
Q4. Realize the basic logic gates, NOT, OR and AND gates using NOR gate only. [4M]

(NOT GATE USING NOR GATE:1 M)

where, X = A NOR A

$x = \bar{A}$

(AND GATE USING NOR GATE:1.5 MARKS)

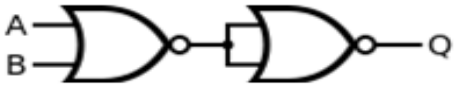


$\overline{Q}=\bar{A}+\bar{B} = \bar{A}+\bar{B}$

$=\overline{\bar{A}.\bar{B}}$

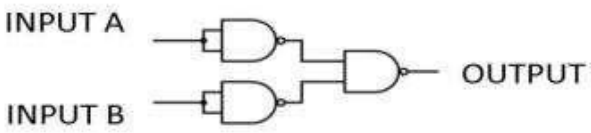
$= \mathbf{A.B}$

(OR GATE USING NOR GATE:1.5 MARKS)

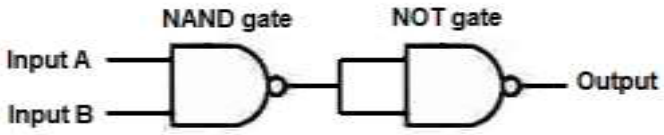


Q5. Realize the following logic expression using only NAND gates. [4M]
(i) OR (ii) AND (iii)NOT

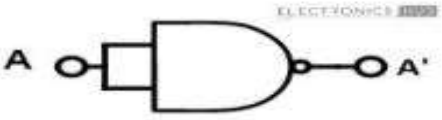
OR gate from NAND gates



AND gate



NOT Gate



Q6. State De Morgan's theorem and prove any one. **[4M]**

i) $\overline{AB} = \overline{A} + \overline{B}$

It states that compliment of product is equal to sum of their compliments.

1	2	3	4	5	6
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

Column 03 = column 06

i.e. $\overline{AB} = \overline{A} + \overline{B}$

Hence proved

OR

ii) $\overline{A+B} = \overline{A} \cdot \overline{B}$

It states that complement of sum is equal to product of their complements.

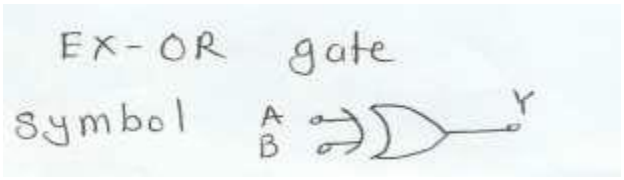
1	2	3	4	5	6
A	B	$\overline{A+B}$	$\overline{A}$	$\overline{B}$	$\overline{A} \cdot \overline{B}$
0	0	1	1	1	1
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	0

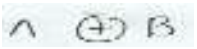
Column 03 = column 06

$\therefore \overline{A+B} = \overline{A} \cdot \overline{B}$

Hence proved.

Q7.Draw symbol, Truth table and logic equation of Ex-OR gate. **[2M]**



Logic Equation = $A\overline{+}B$ OR 

Truth Table		
A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

Q8. Simplify the following Boolean Expression and Implement using logic gate. [4M]

$AB\bar{C}\bar{D} + AB\bar{C}D + ABC\bar{D} + ABCD$

$AB\bar{C}\bar{D} + AB\bar{C}D + ABC\bar{D} + ABCD$ (2mks)

$= AB\bar{C}(\bar{D}+D) + ABC(\bar{D}+D)$ $(\because A+\bar{A}=1)$

$= AB\bar{C}\cdot 1 + ABC\cdot 1$

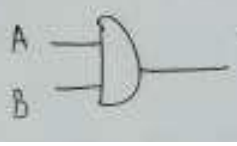
$= AB\bar{C} + ABC$ $\because (A\cdot 1 = A)$

$= AB(\bar{C}+C)$ $\because (A+\bar{A}=1)$

$= AB\cdot 1$ $\because (A\cdot 1 = A)$

$= AB$

Implementation (2mks)



Q8. Compare TTL, CMOS and ECL logic family on the following points. [6M]

(i) Basic Gates **(ii) Propogation dealy**
(iii)Fan out **(iv) Power Dissipation**
(v) Noise immunity **(vi) Speed power product**

Parameter	TTL	CMOS	ECL
Basic gates	NAND	NOR/NAND	OR/NOR
Propagation delay	10	70-105	2
Fan out	10	50	25
Power Dissipation	10mW	1.01mW	40-55mW
Noise Immunity	0.2V	5V	0.25V
Speed Power Product	100	0.7	100

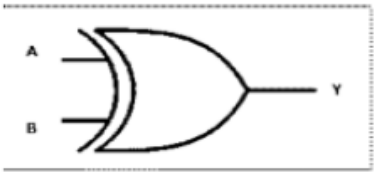
Q9. Define fan-in and fan-out of a gate. **[2M]**

Fan-in is a term that defines the maximum number of digital inputs that a single logic gate can accept. Most transistor-transistor logic (TTL) gates have one or two inputs, although some have more than two. A typical logic gate has a fan-in of 1 or 2.

Fan-out is a term that defines the maximum number of digital inputs that the output of a single logic gate can feed. Most transistor-transistor logic (TTL) gates can feed up to 10 other digital gates.

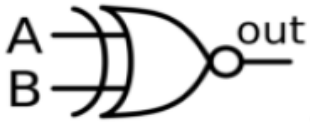
Q10. Draw the logical symbol ofEX-OR and EX-NOR gate. **[2M]**

EX-OR GATE:-



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

EX-NOR GATE:-



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

Q11. Simplify the following and realize it **[4M]**

$$Y = A + \overline{A}\overline{B}C + \overline{A}B\overline{C} + ABC + \overline{A}\overline{B}$$

$Y = A + \overline{A}\,\overline{B}C + \overline{A}\,B\,\overline{C} + ABC + \overline{A}\,\overline{B}$

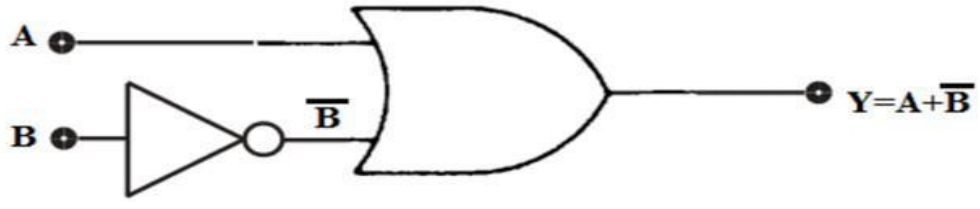
$= A(1 + BC) + \overline{A}\,B(C + \overline{C}) + \overline{A}\,\overline{B}$

$= A + \overline{A}\,B + \overline{A}\,\overline{B}$

$= A + \overline{A}\,\overline{B}$

$= (A + \overline{A}) \cdot (A + \overline{B})$

$= (A + \overline{B})$



Q12. Explain the following characteristics w.r.t. logic families: [4M]
(i) Noise margin (ii) Power dissipation
(iii) Figure of merit (iv) Speed of operation

i)Noise margin indicates the amount to noise voltage circuit can tolerate at its input for both logic 1 and logic0.

Power Dissipation: It is the amount of power dissipated in an IC.

Figure of Merit: It is defined as the product of propagation delay and power dissipated by the gate.

Speed of Operation: Speed of a logic circuit is determined by the time between the application of input and change in the output of the circuit.

CHAPTER 3

Combinational Logic Circuits

Q1. Write simple example of Boolean expression for SOP and POS. [2M]

SOP form:

$$Y = AB + BC + A\overline{C}$$

POS Form:

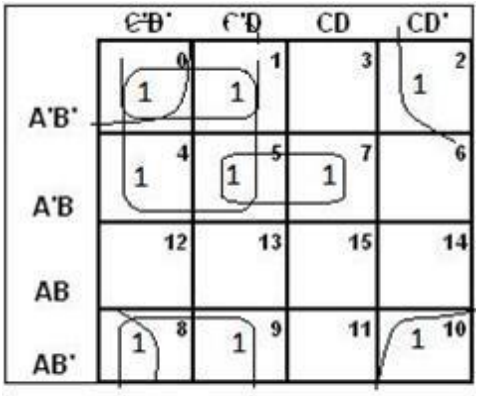
$$Y = (A + B)(B + C)(A + \overline{C})$$

Q2. State the necessity of multiplexer. [2M]

Necessity of Multiplexer:

- It reduces the number of wires required to pass data from source to destination.
- For minimizing the hardware circuit.
- For simplifying logic design.
- In most digital circuits, many signals or channels are to be transmitted, and then it becomes necessary to send the data on a single line simultaneously.
- Reduces the cost as sending many signals separately is expensive and requires more wires to send.

Q3. Minimize the following expression using K-Map. ƒ(A, B, C, D) = ∑m (0, 1, 2, 4, 5, 7, 8,9, 10) [4M]

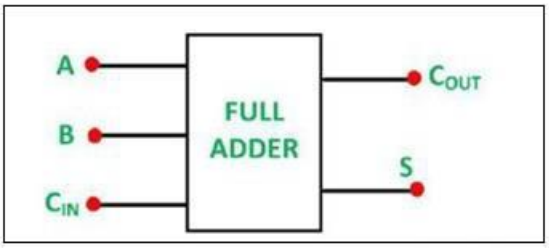


$$\overline{A}\,\overline{C} + \overline{B}\,\overline{C} + \overline{B}\,\overline{D} + \overline{A}BD$$

Q4. Describe the function of full Adder Circuit using its truth table, K-Map simplification and logic diagram. [4M]

A full adder is a combinational logic circuit that performs addition between three bits, the two input bits A and B, and carry C from the previous bit.

Block diagram :



Truth Table :

Input			Output	
A	B	Cin	Sum	Carry
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

K-Map :-

For Carry (C_{out})

A \ BC_{in}	00	01	11	10
0	0	0	1	0
1	0	1	1	1

$C_{out} = AB + A C_{in} + B C_{in}$

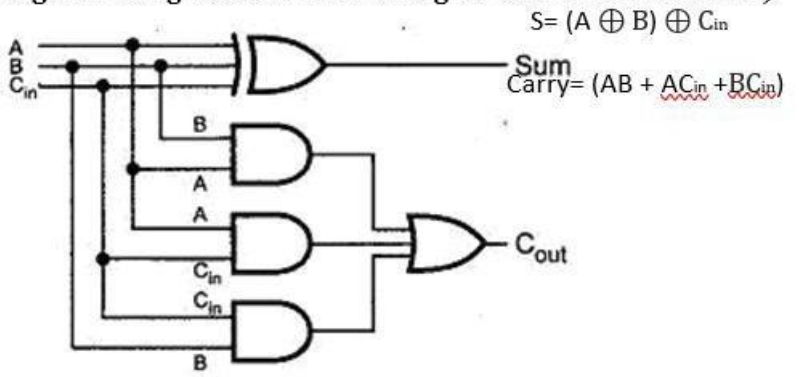
For Sum

A \ BC_{in}	00	01	11	10
0	0	1	0	1
1	1	0	1	0

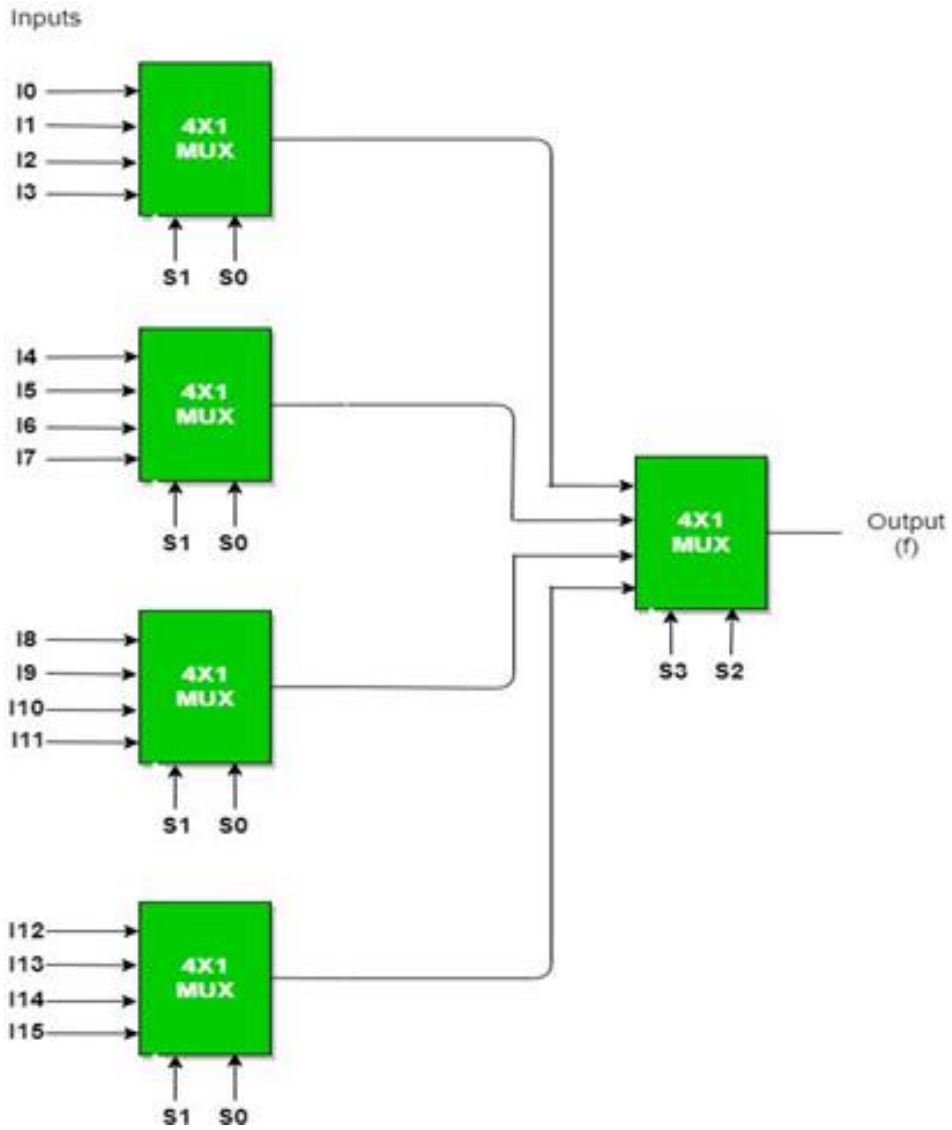
$Sum = \bar{A}\bar{B}C_{in} + \bar{A}B\bar{C}_{in} + A\bar{B}\bar{C}_{in} + ABC_{in}$

Logic Diagram:

(Note: Logic Diagram using basic or universal gate also can be consider)



Q5. Draw 16:1 MUX tree using 4:1 MUX **[4M]**

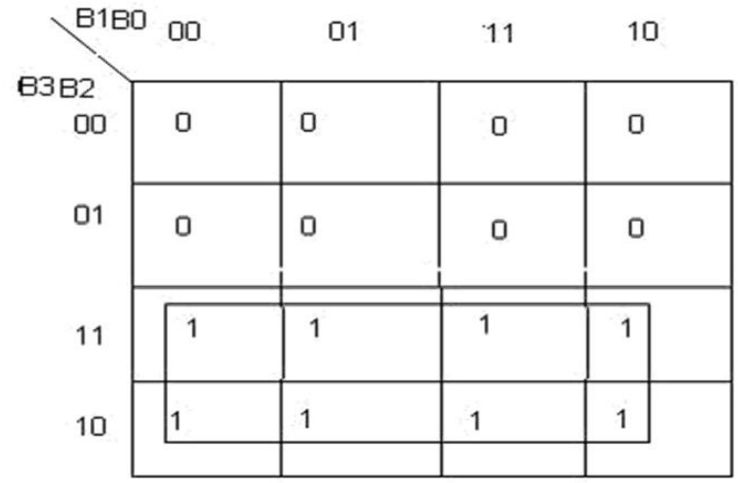


Q6. Design 4 bit Binary to Gray code converter **[6M]**

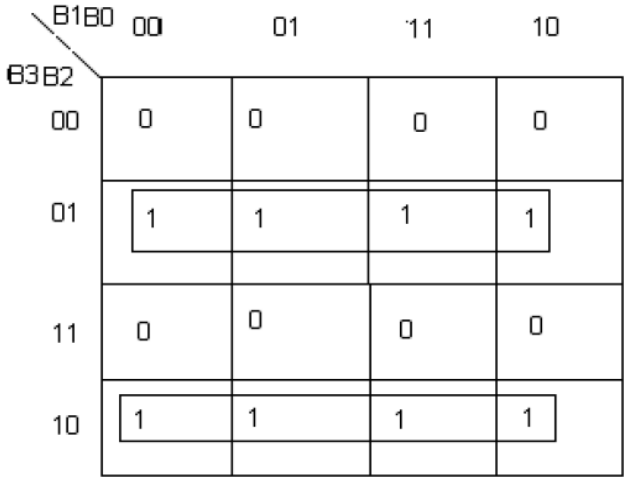
Truth Table for 4 bit Binary to Gray code converter

Binary Input				Gray output			
B_3	B_2	B_1	B_0	G_3	G_2	G_1	G_0
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	0
0	1	0	1	0	1	1	1
0	1	1	0	0	1	0	1
0	1	1	1	0	1	0	0
1	0	0	0	1	1	0	0
1	0	0	1	1	1	0	1
1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	0
1	1	0	0	1	0	1	0
1	1	0	1	1	0	1	1
1	1	1	0	1	0	0	1
1	1	1	1	1	0	0	0

K-MAP FOR G3:



G3=B3

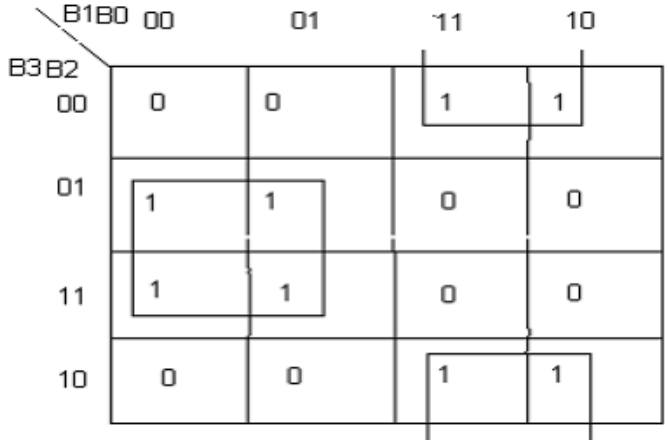


K-MAP FOR G2:

$G2=\overline{B3}\,B2+\overline{B2}\,B3$

= B3 XOR B2

K-MAP FOR G1:

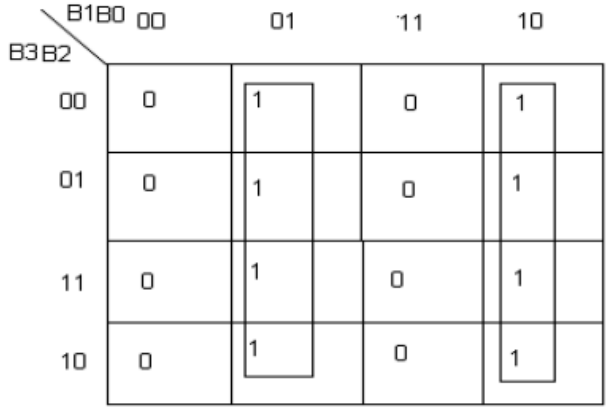


$G1=\overline{B2}\,B1 + B2\,\overline{B1}$

= B1 XOR B2

K-MAP FOR G0:

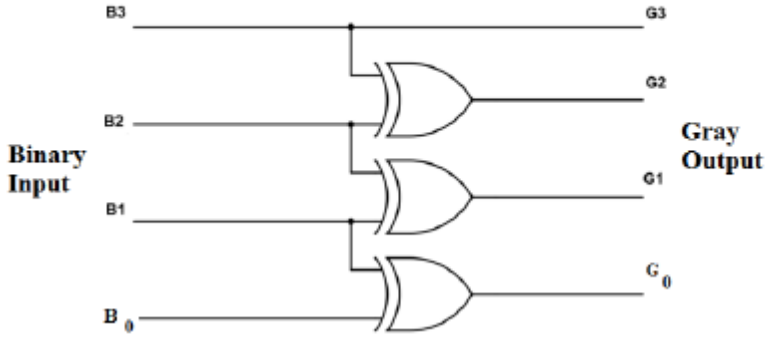
K-MAP FOR G0:



$G0=\overline{B1}B0 + B1\overline{B0}$

= B1 XOR B0

Diagram for 4 bit Binary to Gray code converter:

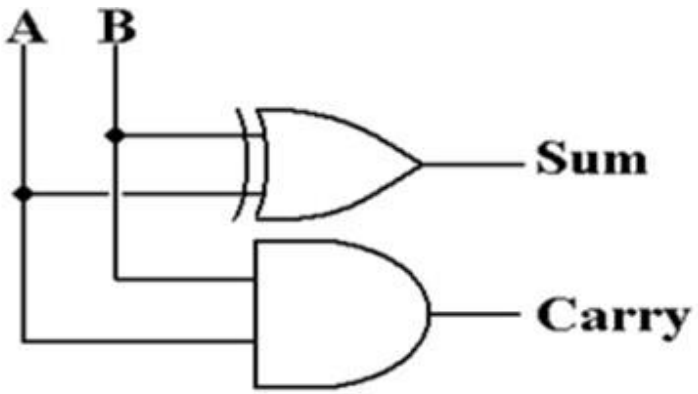


Q7. Define encoder, write the IC number of IC used as decimal to BCD encoder. [2M]

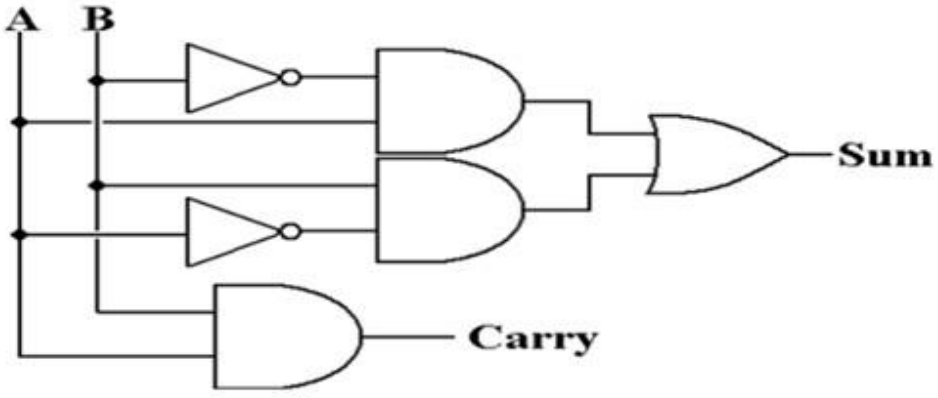
An encoder is a device or circuit that converts information from one format or code to another, for the purpose of standardization, speed or compression.

Decimal to BCD encoder IC- 74147

Q8. Draw logic diagram of half adder circuit. [2M]



OR



Note: logic diagram using NAND/NOR also can be considered.

Q9. Design BCD to seven segment decoder using IC 7447 with its truth table [6M]

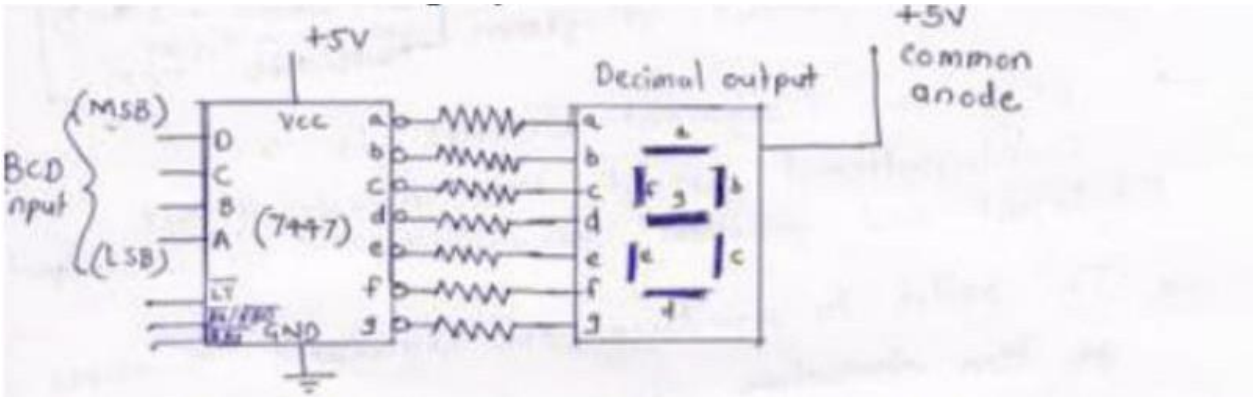
1. BCD to 7 segment decoder is a combinational circuit that accepts 4 bit BCD input and generates appropriate 7 segment output.

2. In order to produce the required numbers from 0 to 9 on the display the correct combination of LED segments need to be illuminated.

3. A standard 7 segment LED display generally has 8 input connections, one from each LED segment & one that acts as a common terminal or connection for all the internal segments

4. Therefore there are 2 types of display 1. Common Anode Display 2. Common Cathode

Display : Common Anode



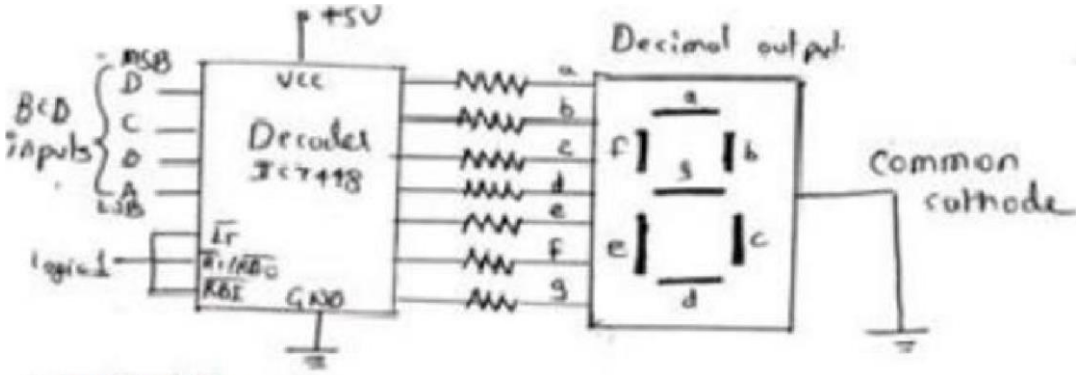
for normal Functioning $\overline{LT}$, $\overline{BI}/\overline{RBO}$ & $\overline{RBI}$
Should be connected to logic 1

Truth Table

for seven segment decoder using common anode display

BCD Inputs				7 segment coded outputs							Display outputs
D	C	B	A	$\overline{a}$	$\overline{b}$	$\overline{c}$	$\overline{d}$	$\overline{e}$	$\overline{f}$	$\overline{g}$	
0	0	0	0	0	0	0	0	0	0	1	
0	0	0	1	1	0	0	1	1	1	1	
0	0	1	0	0	0	1	0	0	1	0	
0	0	1	1	0	0	0	0	1	1	0	
0	1	0	0	1	0	0	1	1	0	0	
0	1	0	1	0	1	0	0	1	0	0	
0	1	1	0	1	1	0	0	0	0	0	
0	1	1	1	0	0	0	1	1	1	1	
1	0	0	0	0	0	0	0	0	0	0	
1	0	0	1	0	0	0	1	1	0	0	

Common Cathode Display

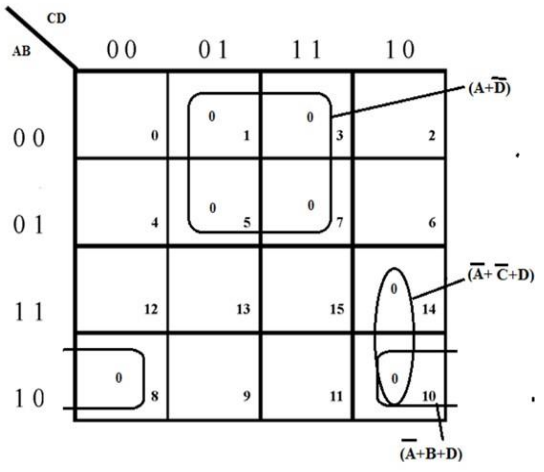


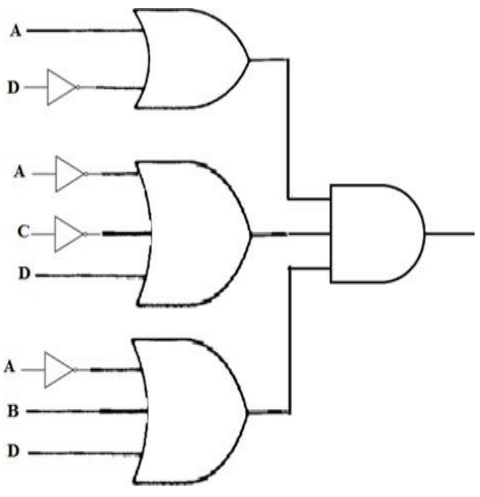
Truth Table

BCD inputs				7 Segment coded outputs							Display output
D	C	B	A	a	b	c	d	e	f	g	
0	0	0	0	1	1	1	1	1	1	0	0
0	0	0	1	0	1	1	0	0	0	0	1
0	0	1	0	1	1	0	1	1	0	1	2
0	0	1	1	1	1	1	1	0	0	1	3
0	1	0	0	0	1	1	0	0	1	1	4
0	1	0	1	1	0	1	1	0	1	1	5
0	1	1	0	0	0	1	1	1	1	1	6
0	1	1	1	1	1	1	0	0	0	0	7
1	0	0	0	1	1	1	1	1	1	1	8
1	0	0	1	1	1	1	0	0	1	1	9

Q10. Reduce the following expression using K-map and implement it

F(A,B,C,D) = M (1,3,5,7,8,10,14) [4M]





F(A,B,C,D)=(A+$\overline{D}$) ($\overline{A}$+$\overline{C}$+D) ($\overline{A}$+B+D)

Q11. Design 1:8 de multiplexer using 1:4 de multiplexer **[6M]**

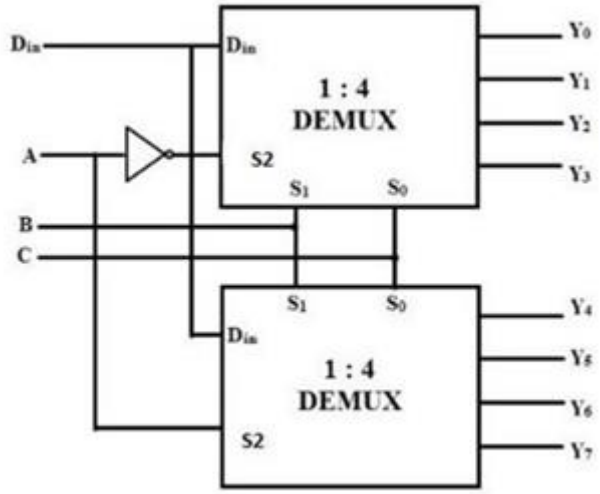


Fig:1:8 Demultiplexer using 1:4 demultiplexer

Data Input	Select Inputs			Outputs							
D	S_2	S_1	S_0	Y_7	Y_6	Y_5	Y_4	Y_3	Y_2	Y_1	Y_0
D	0	0	0	0	0	0	0	0	0	0	D
D	0	0	1	0	0	0	0	0	0	D	0
D	0	1	0	0	0	0	0	0	D	0	0
D	0	1	1	0	0	0	0	D	0	0	0
D	1	0	0	0	0	0	D	0	0	0	0
D	1	0	1	0	0	D	0	0	0	0	0
D	1	1	0	0	D	0	0	0	0	0	0
D	1	1	1	D	0	0	0	0	0	0	0

Q12. Convert the following expression into standard SOP form.

$Y = AB + A\bar{C} + BC$

$Y = AB + A\bar{C} + BC$

Total variable ABC

1st Product term = AB (C is missing)

2nd Product term = $A\bar{C}$ (B is missing)

3rd Product term = BC (A is missing)

$Y = AB \bullet 1 + A\bar{C} \bullet 1 + BC \bullet 1$

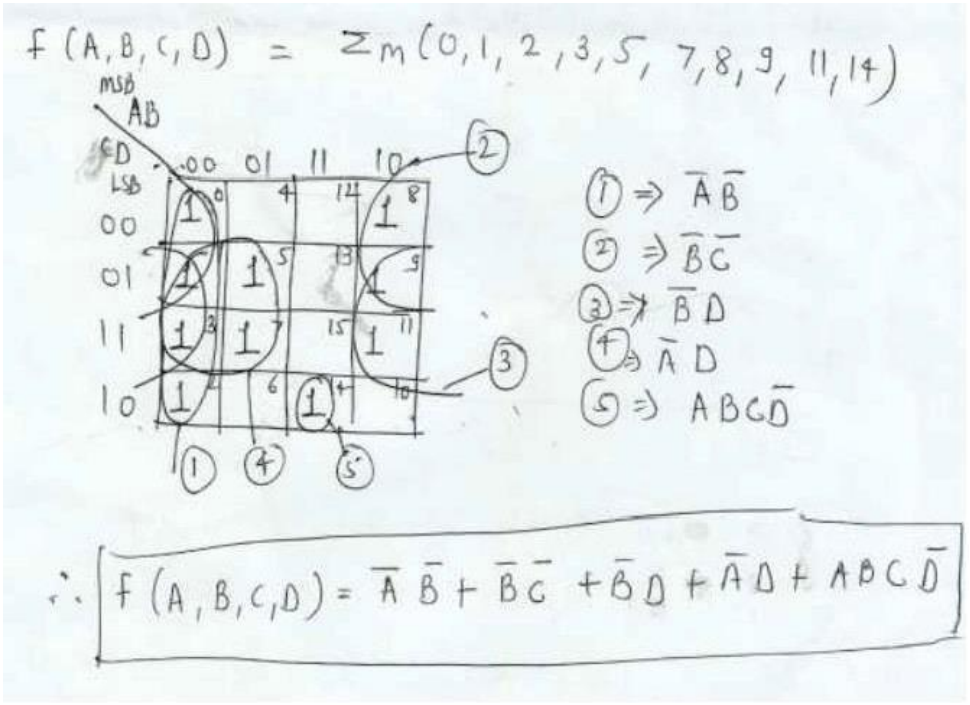
$Y = AB(C+\bar{C})\ A\bar{C}(B+\bar{B}) + BC(A+\bar{A})$

$Y = \underline{ABC} + \underline{AB\bar{C}} + \underline{AB\bar{C}} + A\bar{B}\bar{C} + \underline{ABC} + \bar{A}BC$ $(\because A+\bar{A} = A)$

$Y = ABC + AB\bar{C} + A\bar{B}\bar{C} + \bar{A}BC$ Standard SOP Form

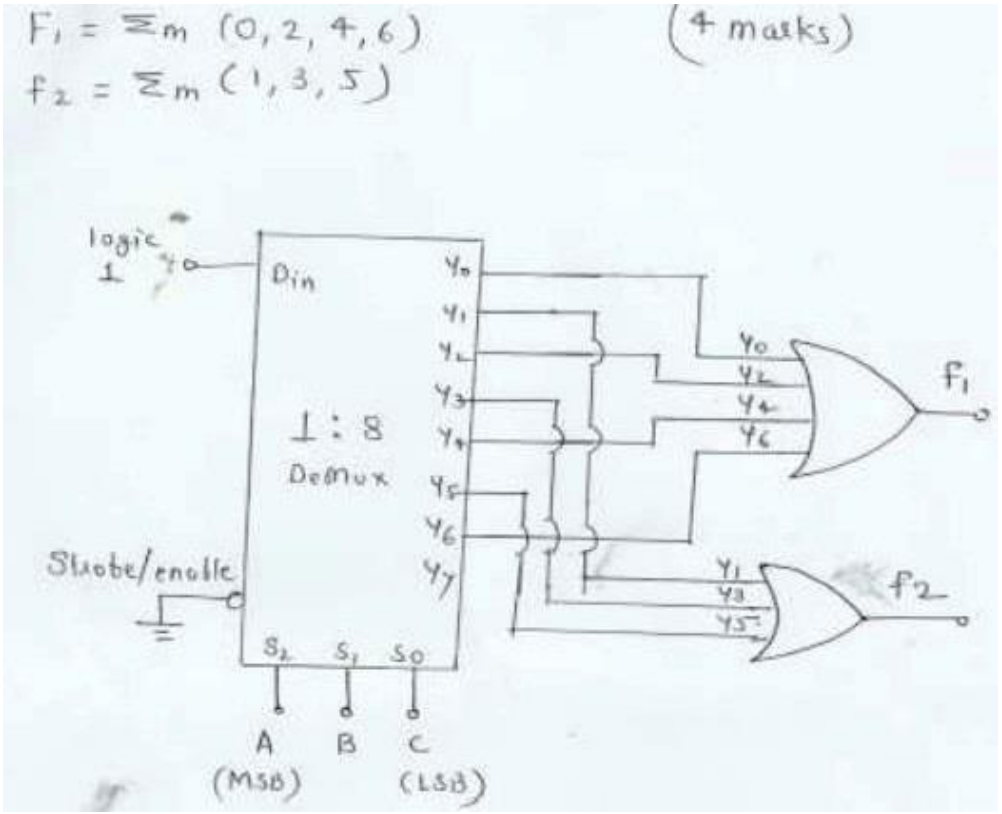
Q13. Minimize the four variable logic function using K map.

F(A,B,C,D) = ∑m(0,1,2,3,5,7,8,9,11,14) **[4M]**



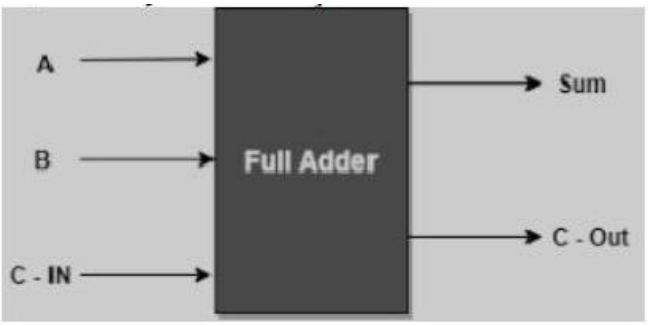
Q14. Implement the following function using demultiplexer **[4M]**

f1 = ∑m(0,2,4,6) **f2 = ∑m(1,3,5)**



Q15. Design a full Adder using Truth Table and K-map. **[4M]**

A full adder is a combinational logic circuit that performs addition between three bits, the two input bits A and B, and carry C from the previous bit.



Truth Table:

Input			Output	
A	B	Cin	Sum	Carry
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

K-map simplification for carry and sum

For Carry (C_{out})

A \ BC_{in}	00	01	11	10
0	0	0	1	0
1	0	1	1	1

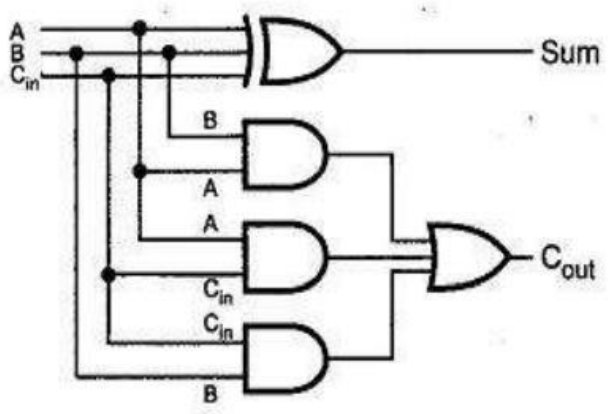
$C_{out} = AB + A C_{in} + B C_{in}$

For Sum

A \ BC_{in}	00	01	11	10
0	0	1	0	1
1	1	0	1	0

$Sum = \bar{A}\,\bar{B}C_{in} + \bar{A}B\bar{C}_{in} + A\bar{B}\,\bar{C}_{in} + ABC_{in}$

Logical diagram:



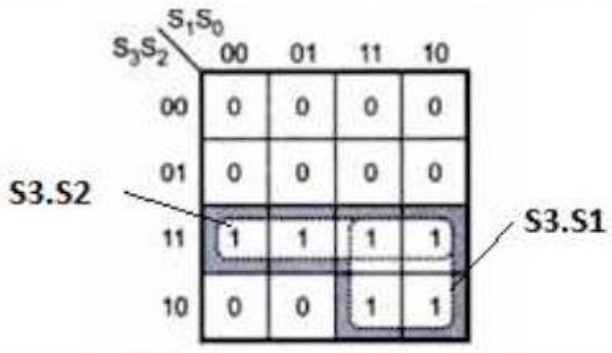
Q16. Design a BCD adder using IC 7483. **[6M]**

1) To implement BCD adder we require:

- 4-bit binary adder for initial addition
- Logic circuit to detect sum greater than 9
- One more 4-bit adder to add 0110201102 in the sum if sum is greater than 9 or carry is 1

2) The logic circuit to detect sum greater than 9 can be determined by simplifying the Boolean expression of given truth Table.

Inputs				Output
S_3	S_2	S_1	S_0	Y
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

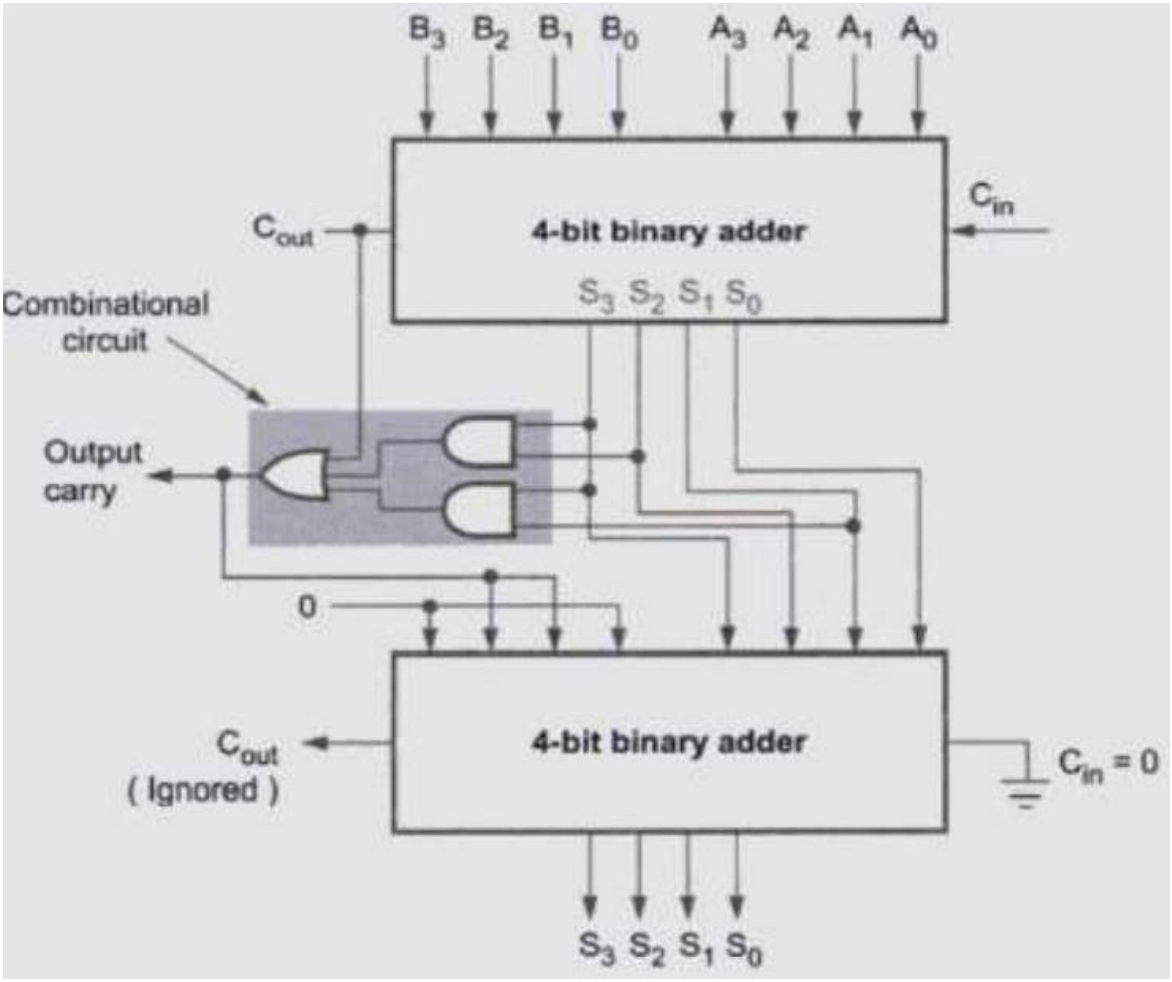


= S3.S2+S3.S1

3) Y=1 indicates sum is greater than 9. We can put one more term, C_out in the above expression to check whether carry is one.

4) If any one condition is satisfied we add 6(0110) in the sum.

5) With this design information we can draw the block diagram of BCD adder, as shown in figure below.



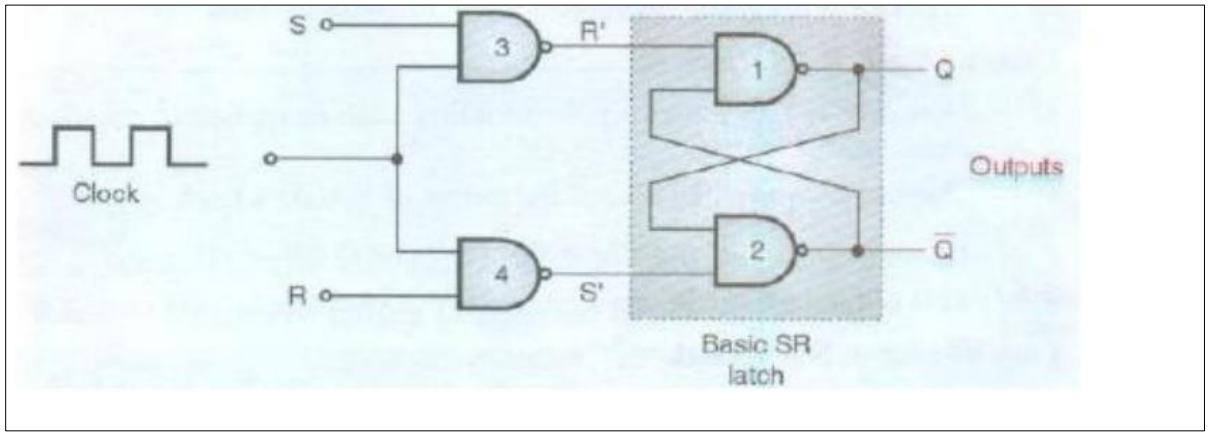
CHAPTER 4

Sequential Logic Circuits

Q1. Compare between synchronous and asynchronous counter (any two points). [2M]

Synchronous Counter	Asynchronous Counter
All flip flops are triggered with same clock.	Different clock is applied to different flip flops.
It is faster.	It is lower
Design is complex.	I Design is relatively easy.
Decoding errors not present.	Decoding errors present.
Any required sequence can be designed	Only fixed sequence can be designed.

Q2. Describe the operation of R-S flip flop using NAND gates only. [4M]



When clock = 0, the outputs of NAND gates 3 and 4 will be forced to be 1 irrespective of the values of S and R. That means R'= S' = 1. Hence the outputs of basic SR/F/F i.e. Q n+1 and will not change. Thus, if clock = 0, then there is no change in the output of the clocked SR flip-flop.

Case I : S = R = 0, clock = 1: No change If S=R=0 then outputs of NAND gate 3 and 4 are forced to become 1. Hence R' and S' both will be equal to 1. Since R' and S' are the inputs of the basic S – R flipflop using NAND gates. There will be no change in the state of outputs.

Case II : S =1, R = 0, clock = 1: Set Now S=0, R=1 and a positive going edge is applied to the clock Output of NAND 3 i.e. R' = 0 and output of NAND 4 i.e. S' = 1. Hence output of SR flip-flop is Q n+1 = 1 and = 0. This is the set condition.

Case III : S =0, R = 1, clock = 1: Reset Now S=0, R=1 and a positive edge is applied to the clock input. Since S=0, output of NAND – 3 i.e. R´= 1. And as R' = 1 and clock = 1 the output of NAND-4 i.e. S´ = 0. Hence output of SR flip-flop is Q n+1 = 0 and = 1. This is the reset condition.

Case IV : S =1, R = 1, clock = 1: Undefined/ forbidden As S=1, R=1 and clock = 1, the outputs of NAND gates 3 and 4 both are 0 i.e. S' = R'=0. So both the outputs Q n+1 $\overline{Qn+1}$ and

Hence output is Undefined/ forbidden.

Q3. State the applications of shift register. **[4M]**

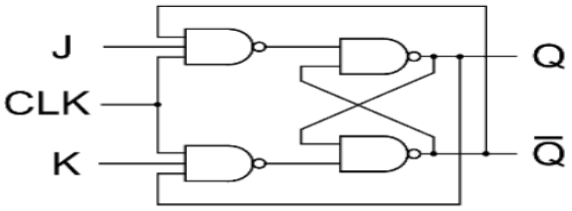
1] Shift register is used as Parallel to serial converter, which converts the parallel data into serial data. It is utilized at the transmitter section after Analog to Digital Converter (ADC) block.

2] Shift register is used as Serial to parallel converter, which converts the serial data into parallel data. It is utilized at the receiver section before Digital to Analog Converter (DAC) block.

3] Shift register along with some additional gate(s) generate the sequence of zeros and ones. Hence, it is used as sequence generator.

4] Shift registers are also used as counters. There are two types of counters based on the type of output from right most D flip-flop is connected to the serial input. Those are Ring counter and Johnson Ring counter.

Q4. Describe the working of J-K flip-flop and state the race around condition. **[4M]**



Inputs			Outputs		
J	K	CLK	Q	$\overline{Q}$	Comments
0	0	↑	Q_0	$\overline{Q}_0$	No change
0	1	↑	0	1	RESET
1	0	↑	1	0	SET
1	1	↑	$\overline{Q}_0$	Q_0	Toggle

The clock signal is applied to CLK input.

IF CLK =0 than F/F is disabled and O/P Q and do not change

If CLK= 1 and J=K=O then the output Q and will not change their state.

If J=0 and K= 1 then JK flip flop will reset and Q= 0 & =1

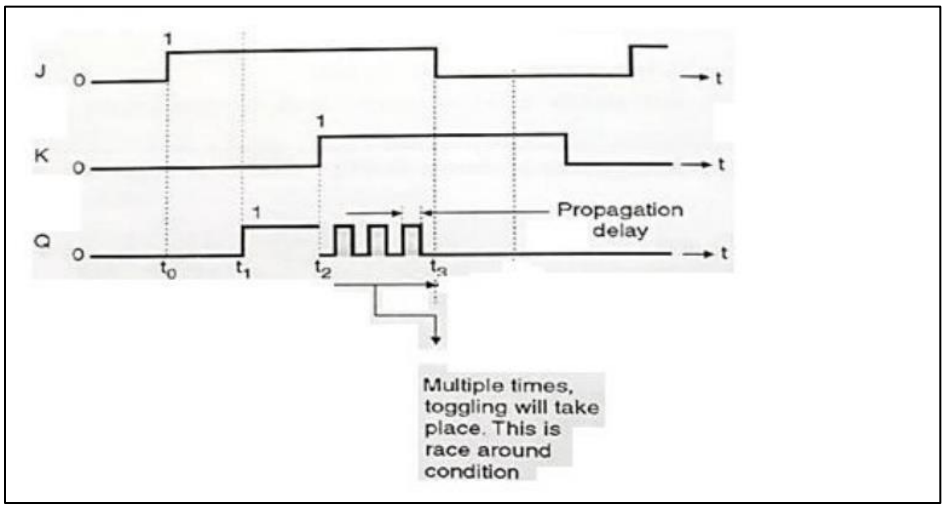
If J=1 and K=0 then output will be set and Q=1 & =0

If J= K=1 then Q & outputs are inverted and FF will toggle

Race Around condition:

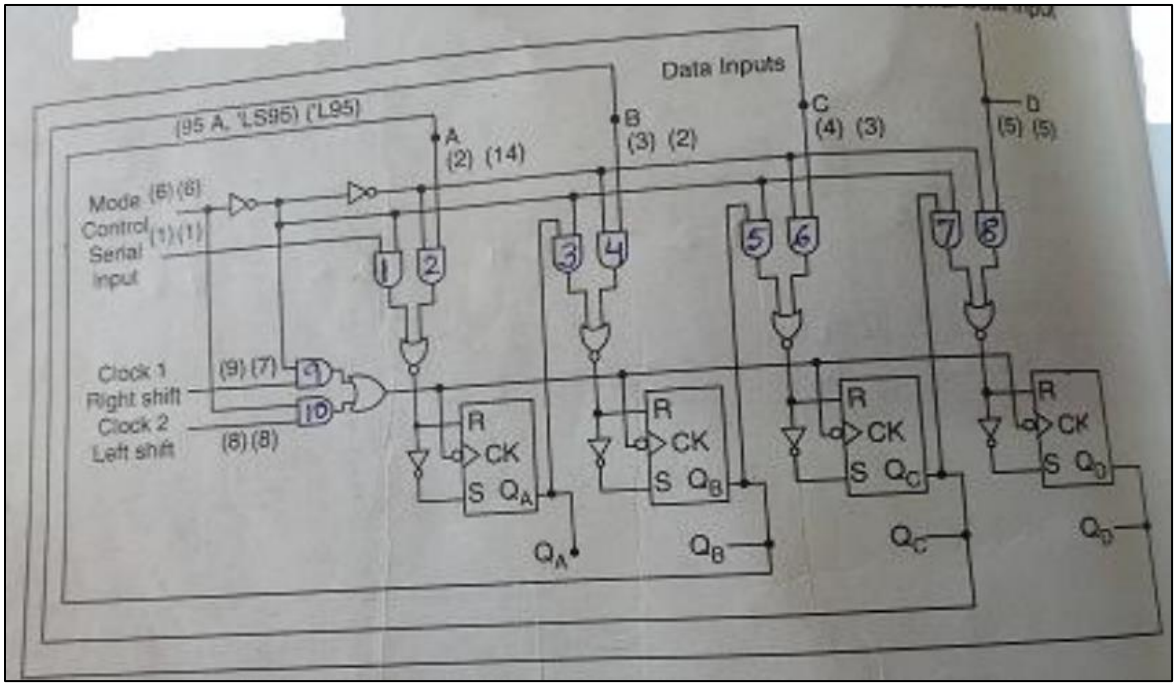
Race around condition occurs in J K Flip-flop only when J=K=1 and clock/enable is high (logic

1) as shown below



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.

Q5. Describe the working of 4 bit universal shift register. **[6M]**



Working:

1. PARALLEL LOAD: When mode control (M) is connected to logic 1, AND gates 2, 4, 6, 8 will be enables and AND gates 1, 3,5,7, will be disabled . The 4-bit binary data will be loaded parallel. The clock-2 input will be applied to the flip-flops , since M= 1, AND gates -10 is enabled and gate-9 is disabled. Input will transfer parallel data to QA to QD outputs.

2. SHIFT RIGHT: When mode control (M) is connected to logic 0, AND gates 1,3,5,7 will be enabled and gates 2, 4,,6, 8,will be disabled. The data will be shifted serially. The clock -1, input will be applied to the flip-flops, Since M = 0, AND gates - 9 is enabled, and gates -10 is disabled. The data is shifted serially to right from QA to QD.

3. SHIFT LEFT: When mode control (M) is connected to logic 1, AND gates 2,4,6,8 will be enabled. This mode permits parallel loading of the resister and shift -left operation. . The shift - left operation can be accomplished by connecting the output of each flip flop to the parallel input of the previous flip- flop and serial input is applied at the input.

Q6. Design a mod-6 Asynchronous counter with truth-table and logic. **[6M]**

MOD 6 asynchronous counter will require 3 flip flops and will count from 000 to 101. Rest of the states are invalid. To design the combinational circuit of valid states, following truth table and K-map is drawn:

Q_C	Q_B	Q_A	Reset Logic
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

From the above truth table, we draw the K-maps and get the expression for the MOD 6 asynchronous counter.

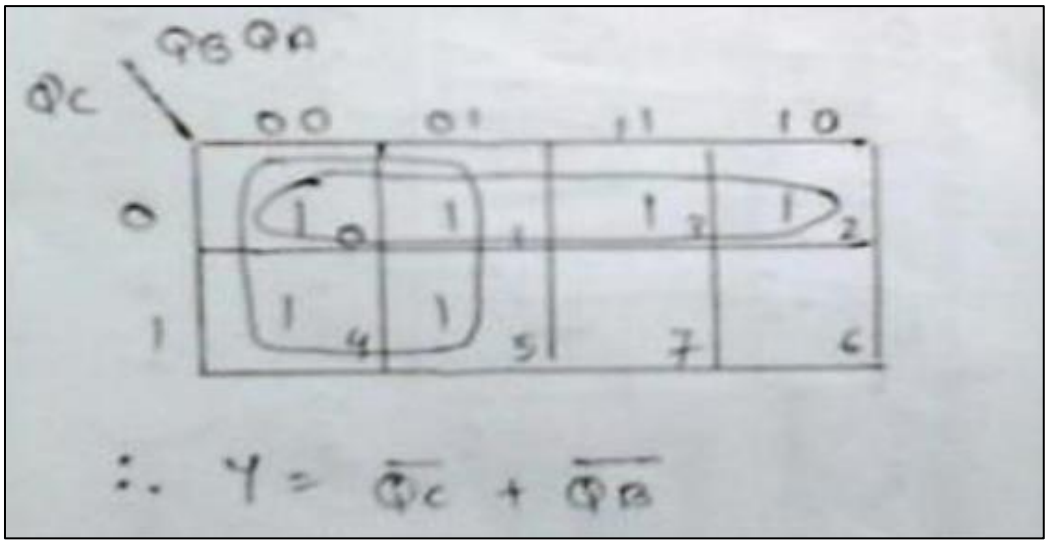


Fig: K-map for above truth table

Thus reset logic is OR of complemented forms of QC and QB. This will be given to the reset inputs of the counter so that as soon as count 110 reaches, the counter will reset. Thus the counter will count from 000 to 101. The implementation of the designed MOD 6 asynchronous counter is shown below:

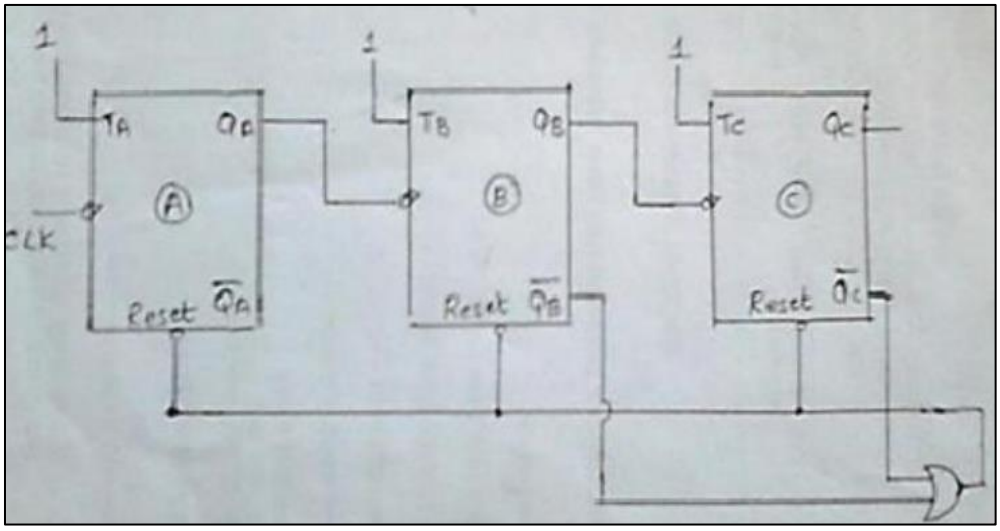
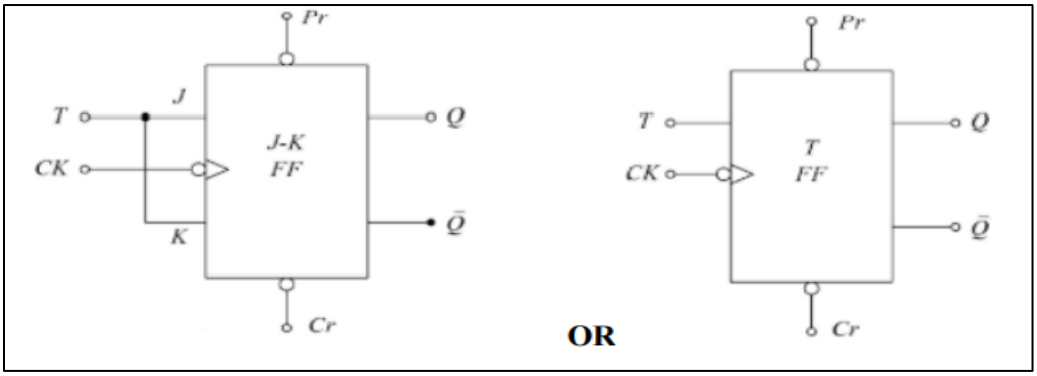


Fig: Circuit diagram of MOD 6 asynchronous counter

Q7. Draw logic diagram of T flip-flop and give its truth table. [2M]

Logic Diagram -



Truth Table –

Input T_n	Output Q_{n+1}	Operation Performed
0	Q_n	No change
1	$\bar{Q}_n$	Toggle

Q8. Define modulus of a counter. Write the numbers of flip flops required for Mod-6 counter. [2M]

Modulus of counter is defined as number of states/clock the counter countes. The numbers of flip flops required for Mod-6 counter is 3.

Q9. State function of preset and clear in flip flop. [2M]

In the flip flop , when the power is switched on, the state of the circuit is uncertain i.e. may be Q = 1 or Q = 0.

Hence, the function of preset is to set a flip flop i.e. Q = 1andthe function of clear is to clear a flip flop i.e. Q = 0.

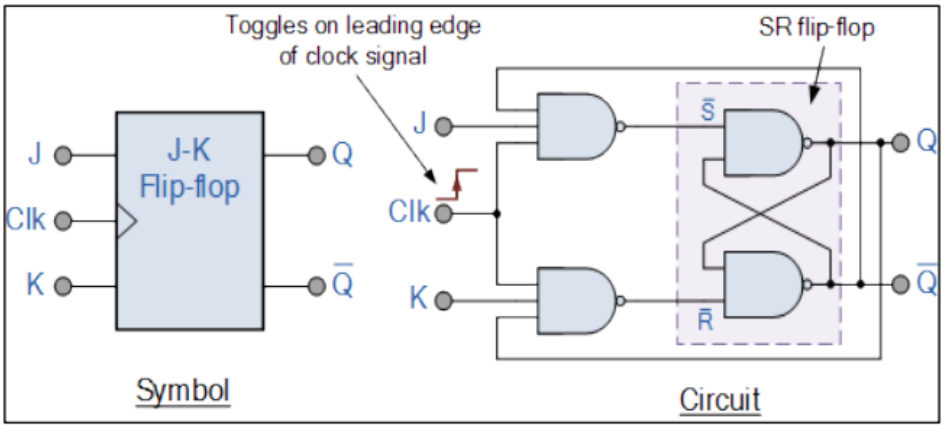
Inputs			Output	Operation performed
CK	*Cr*	*Pr*	*Q*	
1	1	1	Q_{n+1}(Table 7.1)	Normal FLIP-FLOP
0	0	1	0	Clear
0	1	0	1	Preset

Q10. Describe the working of JK flip-flop with its truth table and logic diagram. [4M]

Truth Table –

J	K	CLK	Q
0	0	↑	Q_0 (no change)
1	0	↑	1
0	1	↑	0
1	1	↑	$\overline{Q}_0$ (toggles)

Diagram –



Working –

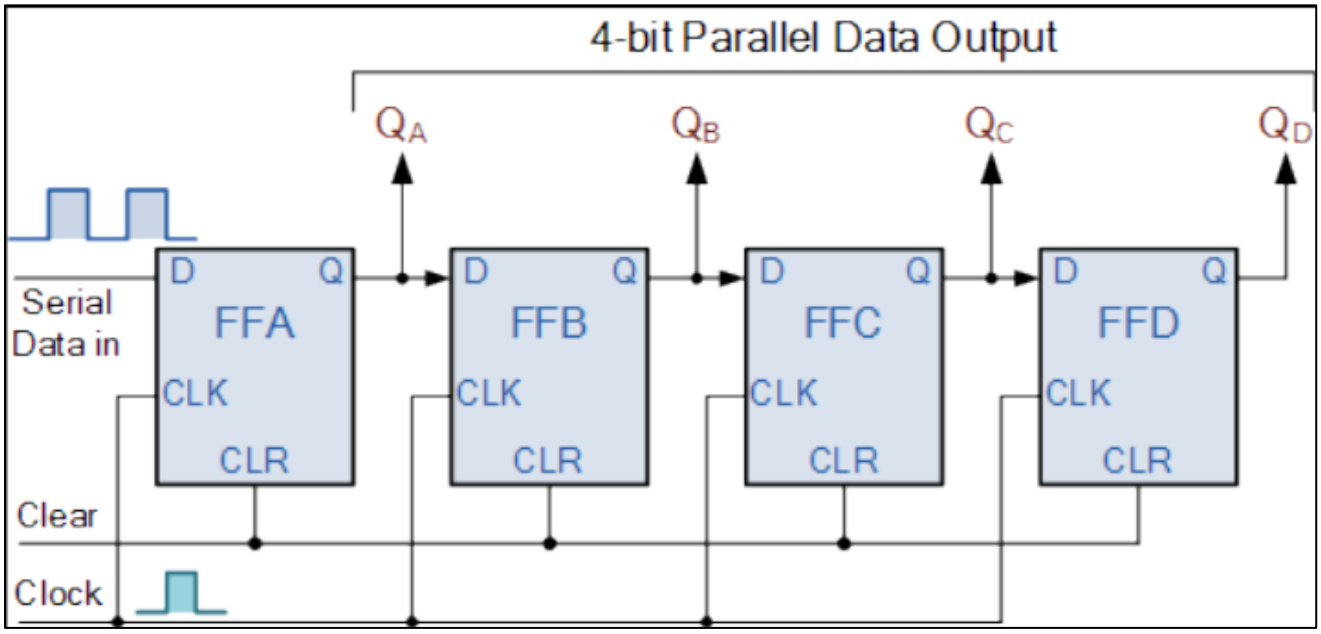
The JK flip flop is basically a gated SR flip-flop with the addition of a clock input circuitry that prevents the illegal or invalid output condition that can occur when both inputs S and R are equal to logic level “1”. Due to this additional clocked input, a JK flip-flop has four possible input combinations, “logic 1”, “logic 0”, “no change” and “toggle”.

Both the S and the R inputs of the previous SR bistable have now been replaced by two inputs called the J and K inputs, respectively after its inventor Jack Kilby. Then this equates to: J = S and K = R.

The two 2-input AND gates of the gated SR bistable have now been replaced by two 3-input NAND gates with the third input of each gate connected to the outputs at Q and Q. This cross coupling of the SR flip-flop allows the previously invalid condition of S = “1” and R = “1” state to be used to produce a “toggle action” as the two inputs are now interlocked.

If the circuit is now “SET” the J input is inhibited by the “0” status of Q through the lower NAND gate. If the circuit is “RESET” the K input is inhibited by the “0” status of Q through the upper NAND gate. As Q and Q are always different, we can use them to control the input. When both inputs J and K are equal to logic “1”, the JK flip flop toggles

Q11. Draw and explain working of 4 bit serial Input parallel Output shift register. [4M]



Explanation –

If a logic “1” is connected to the DATA input pin of FFA then on the first clock pulse the output of FFA and therefore the resulting QA will be set HIGH to logic “1” with all the other outputs still remaining LOW at logic “0”. Assume now that the DATA input pin of FFA has returned LOW again to logic “0” giving us one data pulse or 0-1-0.

The second clock pulse will change the output of FFA to logic “0” and the output of FFBand QB HIGH to logic “1” as its input D has the logic “1” level on it from QA. The logic “1” has now moved or been “shifted” one place along the register to the right as it is now at QA.

When the third clock pulse arrives this logic “1” value moves to the output of FFC (QC) and so on until the arrival of the fifth clock pulse which sets all the outputs QA to QD back again to logic level “0” because the input to FFA has remained constant at logic level “0”.

The effect of each clock pulse is to shift the data contents of each stage one place to the right, and this is shown in the following table until the complete data value of 0-0-0-1 is stored in the register. This data value can now be read directly from the outputs of QA to QD.

Then the data has been converted from a serial data input signal to a parallel data output. The truth table and following waveforms show the propagation of the logic “1” through the register from left to right as follows.

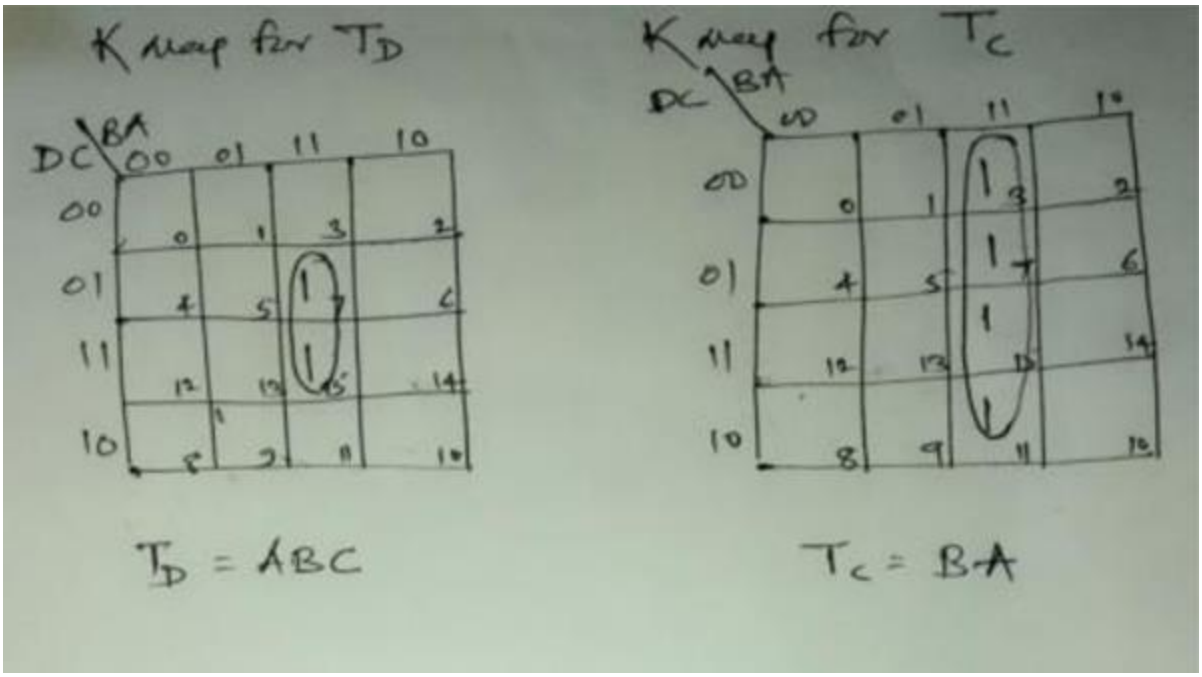
Basic Data Movement through A Shift Register

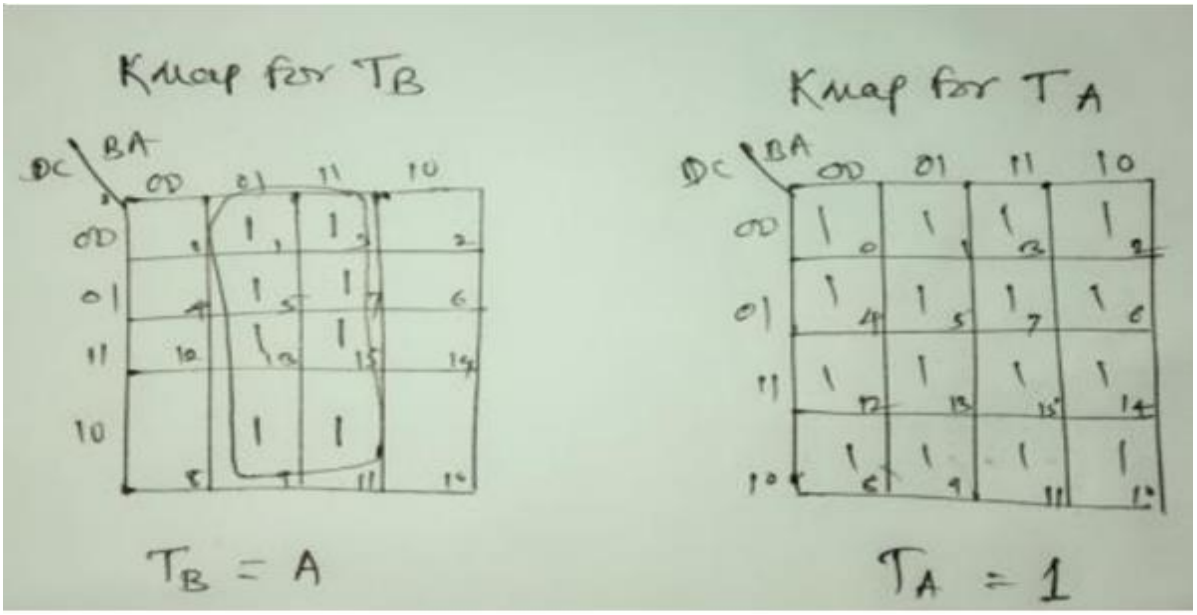
Clock Pulse No	QA	QB	QC	QD
0	0	0	0	0
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1
5	0	0	0	0

Q12. Design a 4-bit synchronous counter and draw its logic diagram. **[6M]**

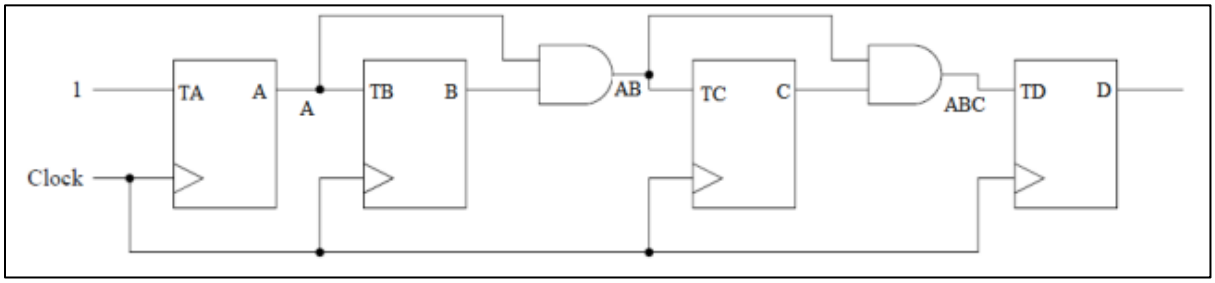
State Table –

Present state				Next state				Flip flop inputs			
D	C	B	A	D^+	C^+	B^+	A^+	T_D	T_C	T_B	T_A
0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	1	0	0	1	0	0	0	1	1
0	0	1	0	0	0	1	1	0	0	0	1
0	0	1	1	0	1	0	0	0	1	1	1
0	1	0	0	0	1	0	1	0	0	0	1
0	1	0	1	0	1	1	0	0	0	1	1
0	1	1	0	0	1	1	1	0	0	0	1
0	1	1	1	1	0	0	0	1	1	1	1
1	0	0	0	1	0	0	1	0	0	0	1
1	0	0	1	1	0	1	0	0	0	1	1
1	0	1	0	1	0	1	1	0	0	0	1
1	0	1	1	1	1	0	0	0	1	1	1
1	1	0	0	1	1	0	1	0	0	0	1
1	1	0	1	1	1	1	0	0	0	1	1
1	1	1	0	1	1	1	1	0	0	0	1
1	1	1	1	0	0	0	0	1	1	1	1

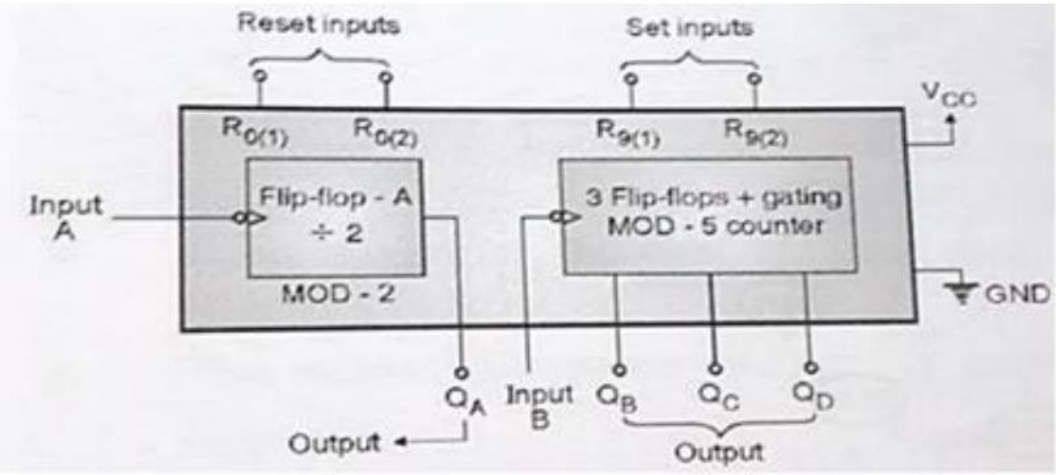




Logic Diagram –



Q13. Give block schematic of decade counter IC 7490. Design Mod-7 counter using it [6M].
Block schematic of decade counter IC 7490 –



Mod-7 means states are from 0,1,2,3,4,5,6,0

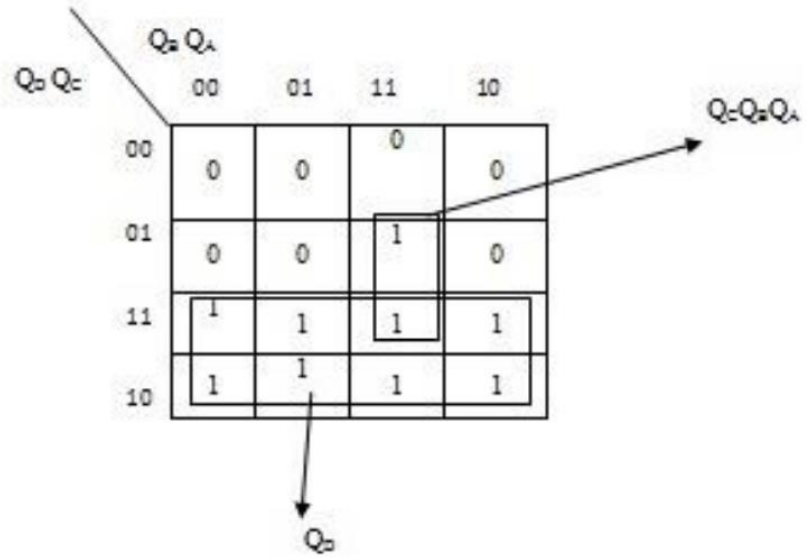
Therefore we have to reset counter IC 7490 when QD, QC, QB, QA=0111

Design reset logic: Output of reset circuit should be HIGH because R0(1) and R0(2) are active high inputs. Therefore, reset logic output should be low for states 0 to 6. Output should be HIGH for states 7 onwards.

Truth table & K-map:

Q_D	Q_C	Q_B	Q_A	Y	
0	0	0	0	0	
0	0	0	1	0	
0	0	1	0	0	
0	0	1	1	0	
0	1	0	0	0	
0	1	0	1	0	
0	1	1	0	0	
0	1	1	1	1	Invalid State
1	0	0	0	1	
1	0	0	1	1	

For Y

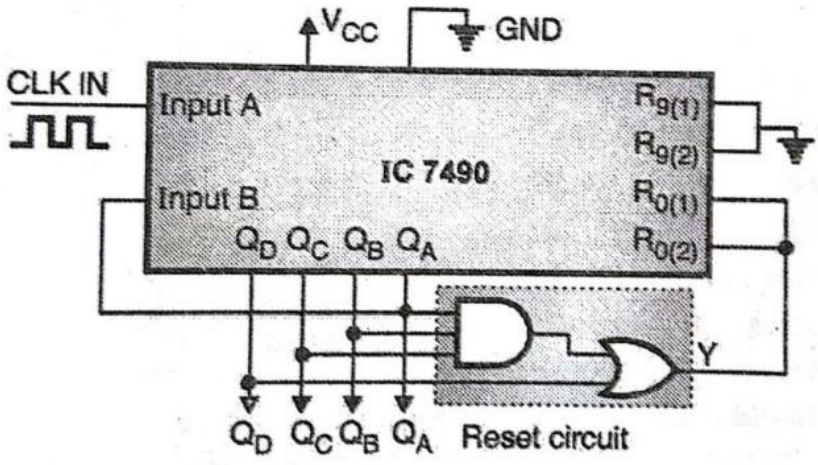


Expression for Y:

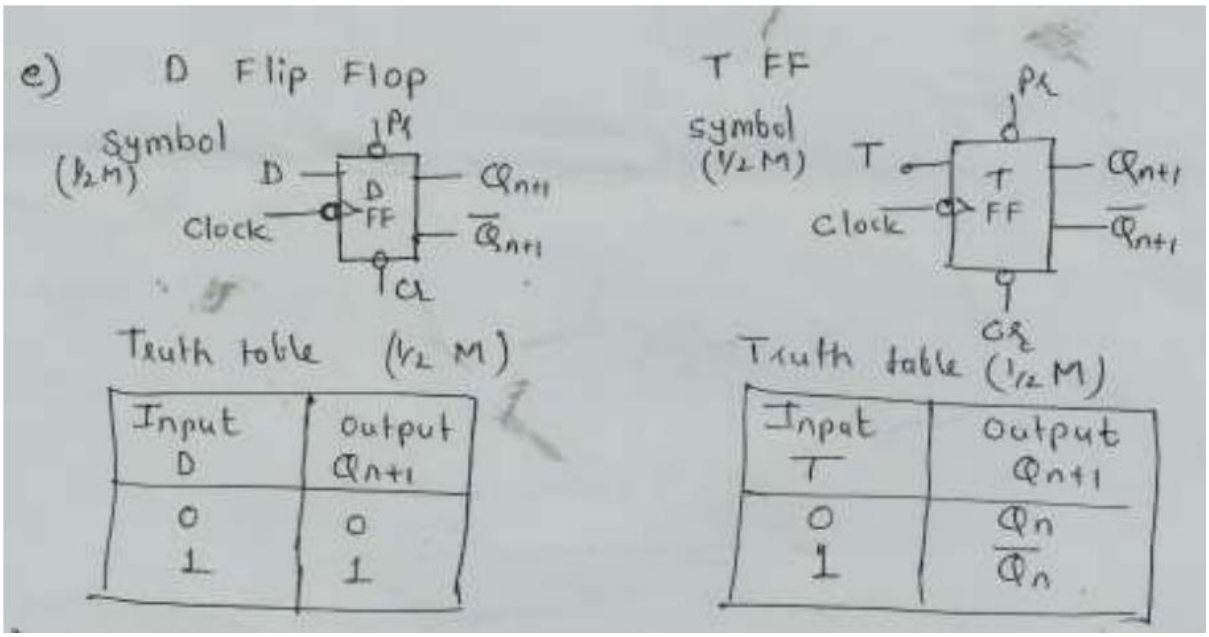
Y= QC QB QA + QD

Circuit is-

Logic Diagram:



Q14. Draw symbol and write truth table of D and T Flip Flop. **[2M]**



Q15. Write down number of flip flops are required to count 16 clock pulses. **[2M]**

No of states= no. of clock pulses = 16

2 n =m

n = no.of flip flops requried

m= no.of states

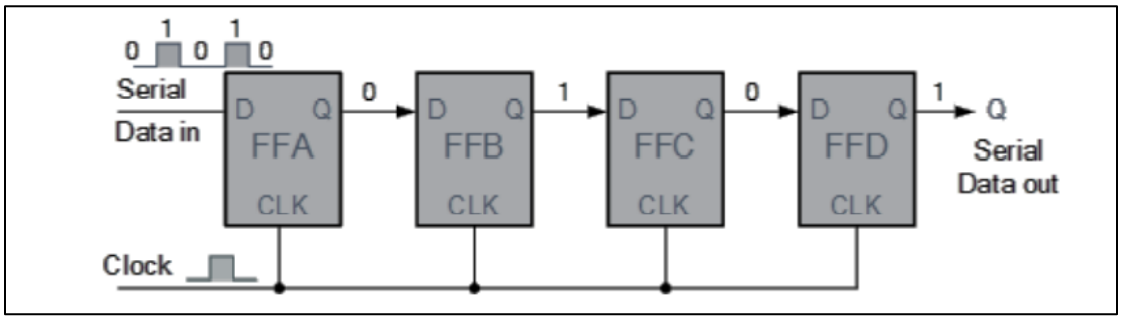
$2^n = 16$

n = 4

4 flip flops are required to count 16 clock pulse.

Q16. Describe the working of 4 bit SISO (serial in serial out) shift register with diagram and waveform if input is 01101. [4M]

Diagram: (use SR or JK or D type flip flop)

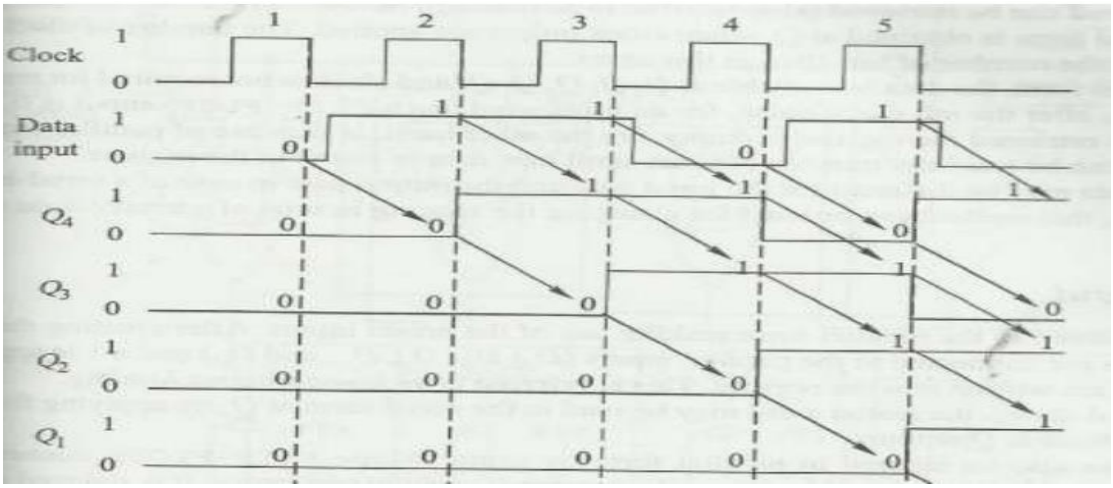


Working:

The DATA leaves the shift register one bit at a time in a serial pattern, hence the name Serial-in to Serial-Out Shift Register or SISO.

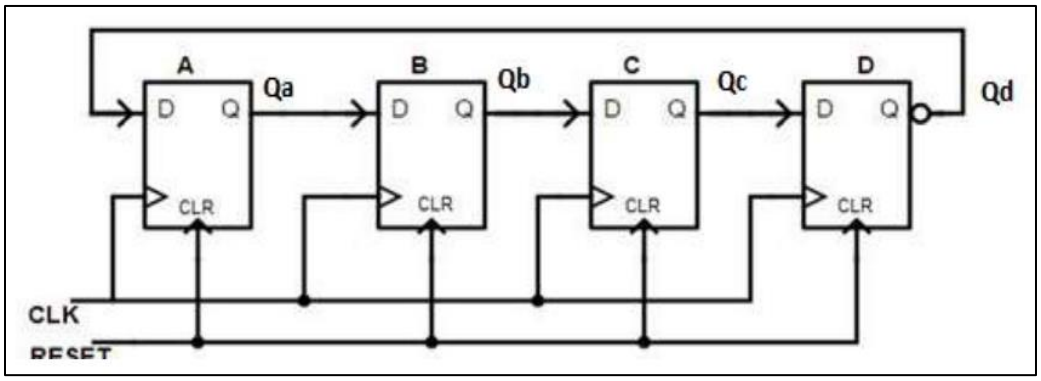
The SISO shift register is one of the simplest of the four configurations as it has only three connections, the serial input (SI) which determines what enters the left hand flip-flop, the serial output (SO) which is taken from the output of the right hand flip-flop and the sequencing clock signal (Clk). The logic circuit diagram below shows a generalized serialin serial-out shift register, Output of FFA is Q4,FFB Q3,FFC Q2 and FFD is Q1

Waveform:(Input is 01101)

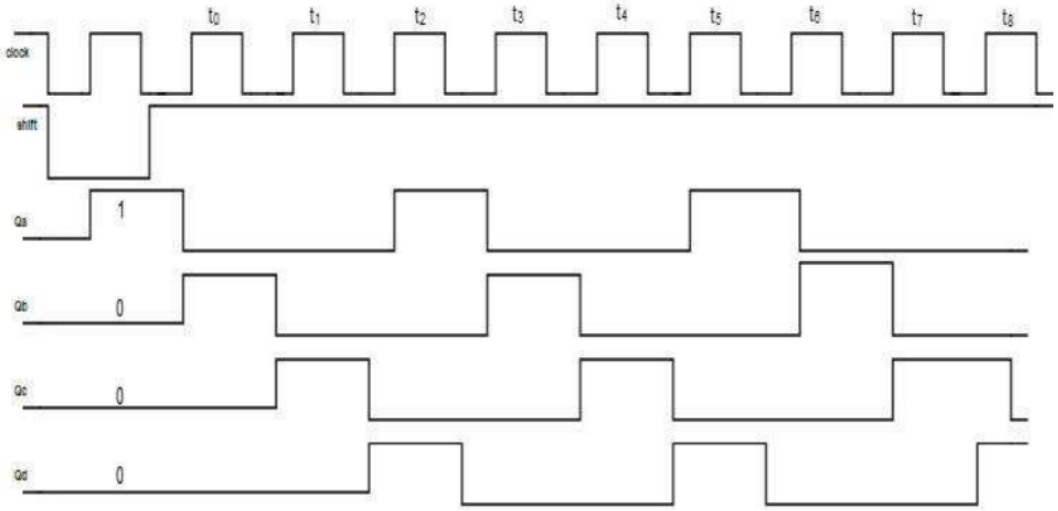


Q17. Describe the working of ring counter using D flip flop with diagram and waveforms. [4M]

Diagram –



Waveforms:



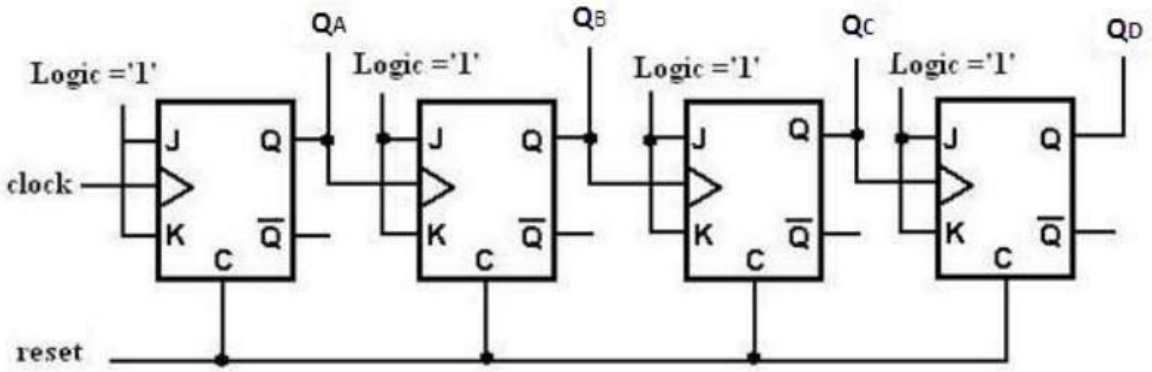
Working:

The ring counter is a cascaded connection of flip flops, in which the output of last flip flop is connected to input of first flip flop. In ring counter if the output of any stage is 1, then its reminder is 0. The Ring counters transfers the same output throughout the circuit.

That means if the output of the first flip flop is 1, then this is transferred to its next stage i.e. 2nd flip flop. By transferring the output to its next stage, the output of first flip flop becomes 0. And this process continues for all the stages of a ring counter. If we use n flip flops in the ring counter, the „1“ is circulated for every n clock cycles.

Q18. Design a 4bit ripple counter using JK flip flop, with truth table and waveforms. [6M]

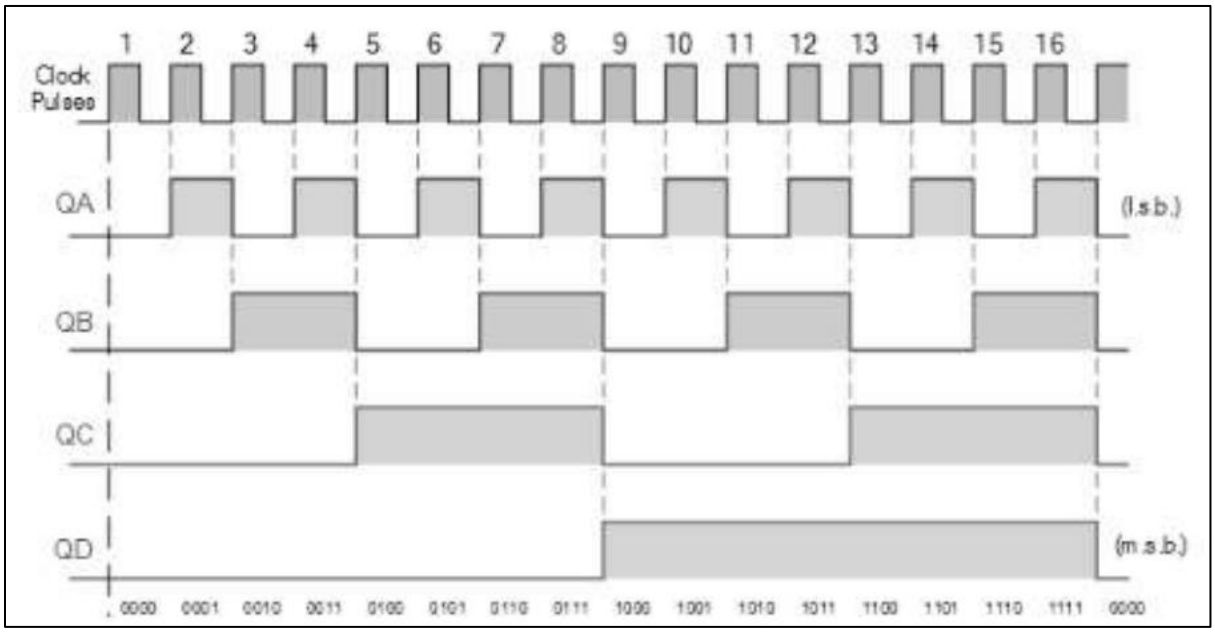
Circuit Diagram:



Truth Table:

State	Q_D	Q_C	Q_B	Q_A
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1
0	0	0	0	0

Timing Diagram / Waveforms:

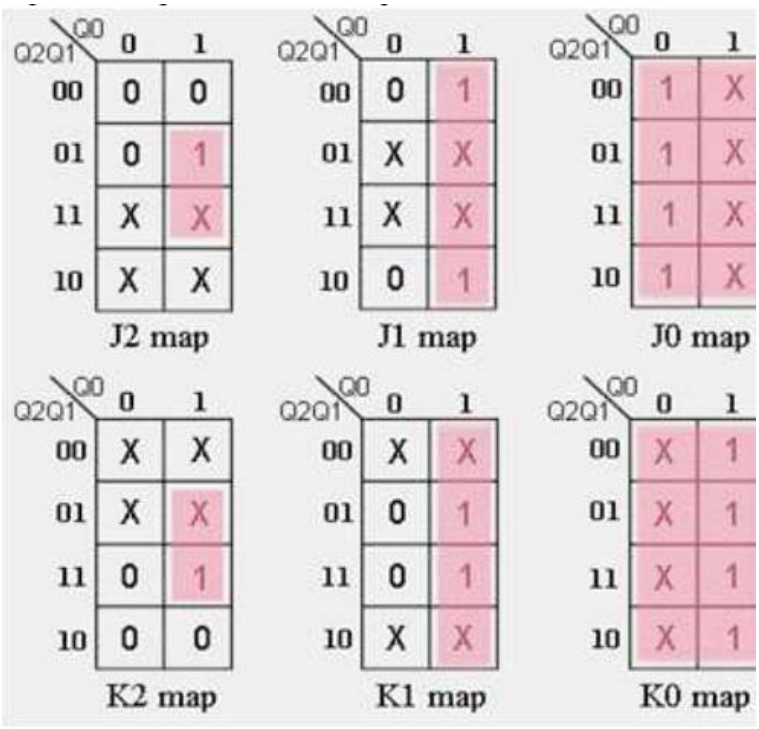


Q19) Design a 3 bit synchronous counter using JK Flip Flop.

1) Step1: Construct JK state table with corresponding excitation table

Output State Transitions		Flip-flop inputs		
Present State Q2 Q1 Q0	Next state Q2 Q1 Q0	J2 K2	J1 K1	J0 K0
0 0 0	0 0 1	0 X	0 X	1 X
0 0 1	0 1 0	0 X	1 X	X 1
0 1 0	0 1 1	0 X	X 0	1 X
0 1 1	1 0 0	1 X	X 1	X 1
1 0 0	1 0 1	X 0	0 X	1 X
1 0 1	1 1 0	X 0	1 X	X 1
1 1 0	1 1 1	X 0	X 0	1 X
1 1 1	0 0 0	X 1	X 1	X 1

Step 2: Build Karnaugh Map or Kmap for each JK inputs:



Q2Q1 \ Q0	0	1
00	0	0
01	0	1
11	X	X
10	X	X

J2 map

Q2Q1 \ Q0	0	1
00	0	1
01	X	X
11	X	X
10	0	1

J1 map

Q2Q1 \ Q0	0	1
00	1	X
01	1	X
11	1	X
10	1	X

J0 map

Q2Q1 \ Q0	0	1
00	X	X
01	X	X
11	0	1
10	0	0

K2 map

Q2Q1 \ Q0	0	1
00	X	X
01	0	1
11	0	1
10	X	X

K1 map

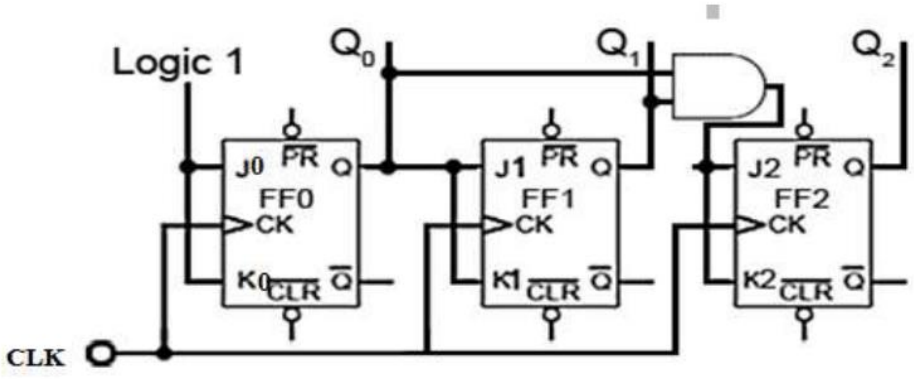
Q2Q1 \ Q0	0	1
00	X	1
01	X	1
11	X	1
10	X	1

K0 map

J2=Q1.Q0 J1=Q0 J0=1

K2=Q1.Q0 K1=Q0 K=1

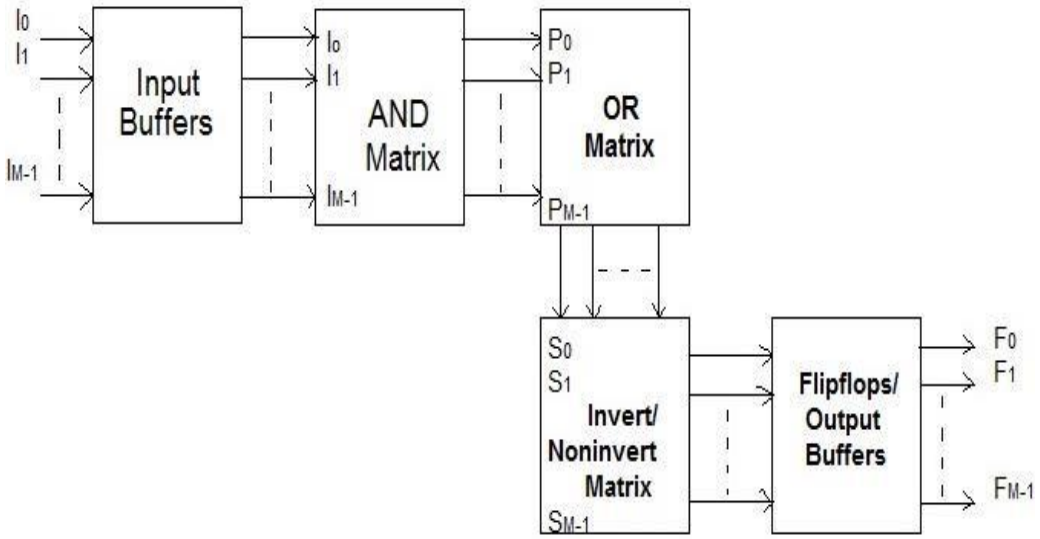
Step3: Draw the complete design as below:



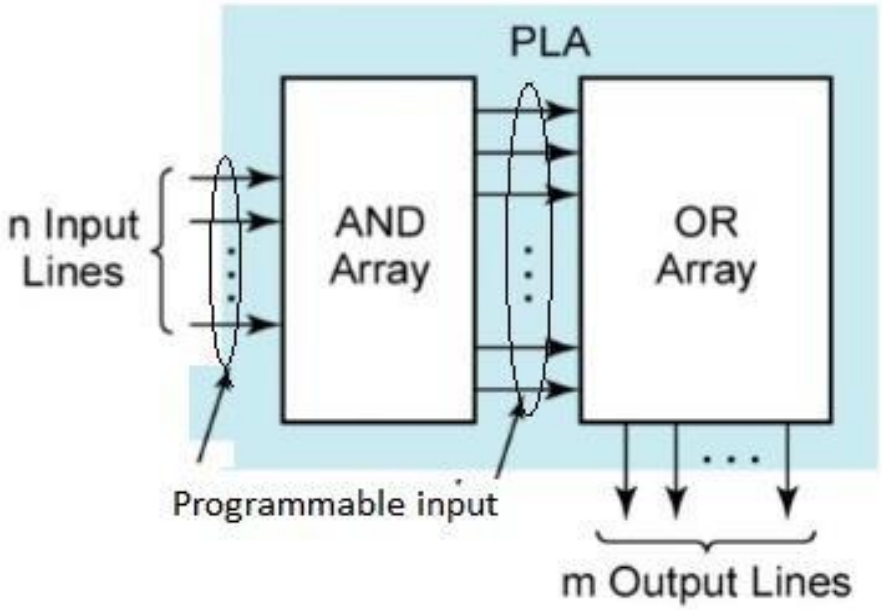
Chapter 5

Data Converters and PLDS

Q1. Draw the block diagram of Programmable Logic Array. **[4M]**



OR



\

Q2.Calculate analog output of 4 bit DAC for digital input 1101. Assume VFS = 5V. [4M]

Formula :-

VR = VFS

$$V_o = V_R [d_1 2^{-1} + d_2 2^{-2} + \ldots + d_n 2^{-n}]$$

$= 5(1x2^{-1} + 1x2^{-2}+0x2^{-3}+1x2^{-4})$
$= 5(0.5+O.25+0+0.0625)$
$= 4.0625$ Volts

OR

$$V_{FS} = V_R \cdot \left(\frac{b3}{2} + \frac{b2}{4} + \frac{b1}{8} + \frac{b0}{16}\right)$$

Note – (Since V_R is not given find V_R)

Full Scale o/p mean

b3 b2 b1 b0 = 1111

$V_{FS} = 5V$

$$5 = V_R \cdot \left(\frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{16}\right)$$

$V_R = 5.33$

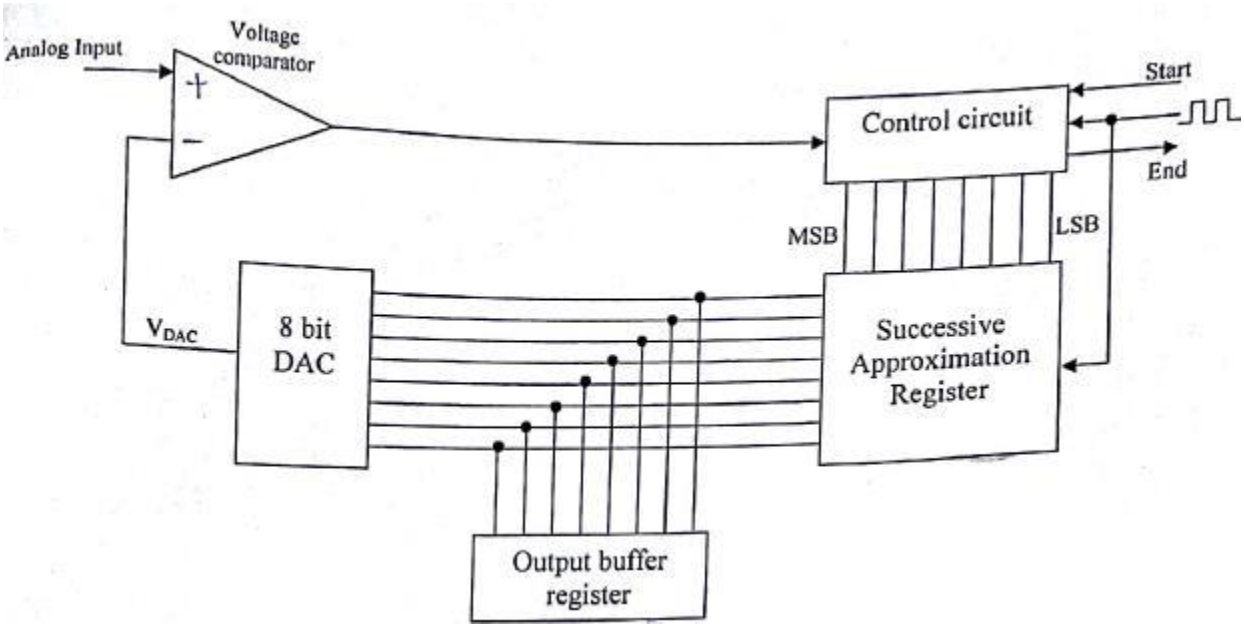
For digital i/p b3 b2 b1 b0 = 1101

$$V_0 = 5.33 \left(\frac{1}{2} + \frac{1}{4} + \frac{0}{8} + \frac{1}{16}\right)|$$

V0 = 4.33V

Q3. Describe the working of Successive Approximation ADC. Define Resolution and conversion time associate with ADC. [6M]

Circuit diagram:



When the start signal goes low the successive approximation register SAR is cleared and output voltage of DAC will be 0V. When start goes high the conversion starts. After starts, during first clock pulse the control circuit set MSB bit so SAR output will be 1000 0000. This is connected as input to DAC so output of DAC is (analog output) compared with Vin input voltage. If VDAC is more than Vin the comparator output –Vsat, if VDAC is less than Vin, the comparator output is +Vsat.

If output of DAC i.e. VDAC is + Vsat (i.e unknown analog input voltage Vin> VDAC) then MSB bit is kept set, otherwise it is reset. Consider MSB is set so SAR will contain 1000 0000. The next clock pulse will set next bit i.e D6 a digital output of 1100 0000. The output voltage of DAC i.e VDAC is compared with Vin, if it is + Vsatthe D6 bit is kept as it is, but if it is –Vsat the D6 bit reset.

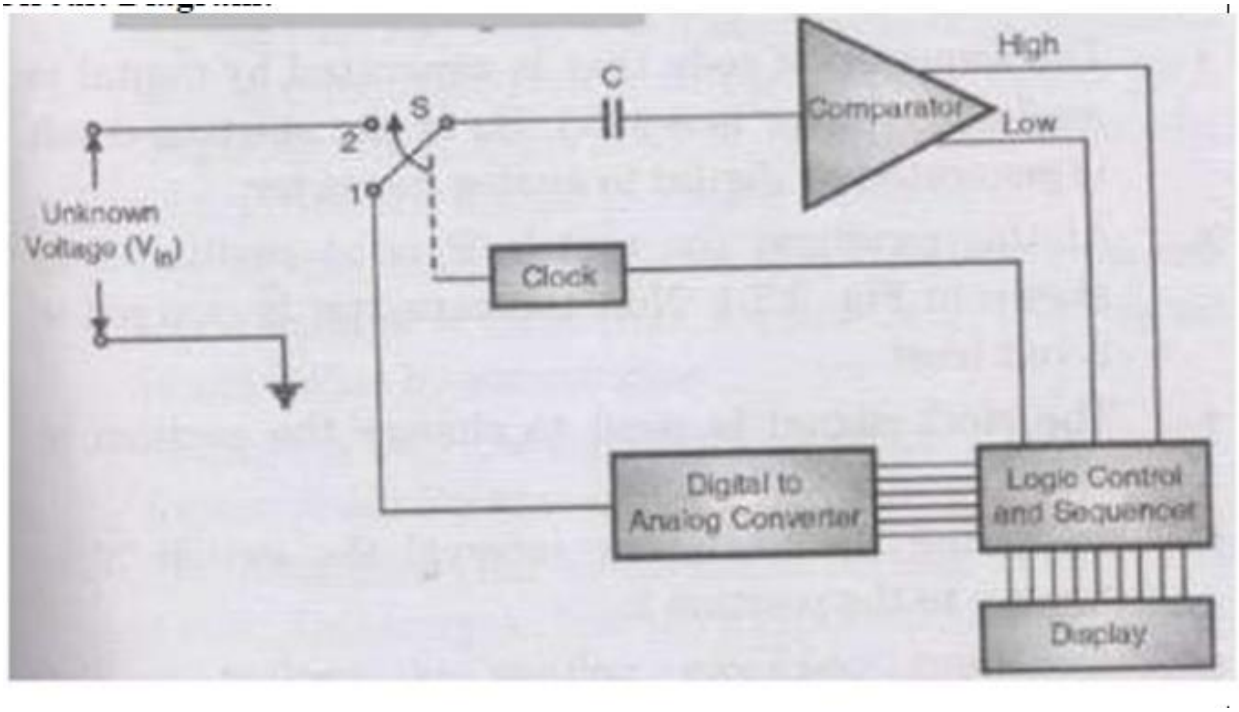
The process of checking and taking decision to keep bit set or to reset is continued upto D0. Then the DAC input will be digital data equal to analog input.

When the conversation if finished the control circuits sends out an end of conversion signal and data is locked in buffer register

Resolution: The voltage input change necessary for a one bit change in the output is called resolution. Conversion Time: The conversion time is the time required for conversion from an analog input voltage to the stable digital output

OR

Circuit Diagram:



Explanation:
DAC= Digital to Analog converter
EOC= End of conversion
SAR =Succesive approximation register
S/H= Sample and hold circuit
Vin= input voltage
Vref= reference voltage
The successive approximation Analog to Digital converter circuit typically consisting of four sub circuits-
1. A sample and hold circuit to acquire the input voltage Vin.
2. An analog voltage comparator that compares Vin to the output of internal
DAC and outputs the result of comparison to successive approximation register(SAR).
3. SAR sub circuits designed to supply an approximate digital code of Vin to the internal DAC.
4. An internal reference DAC that supplies the comparator with an analog voltage equivalent of digital code output of SAR for comparison with Vin. The successive approximation register is initialized so that most significant bit (MSB) is equal to digital 1. This code is fed into DAC which the supplies the analog equivalent of this digital code Vref/2 into the comparator circuit for the comparison with sampled input voltage. If this analog voltage exceeds Vin the comparator causes the SAR to reset the bit, otherwise a bit is left as 1. Then the next bit is set to 1 and the same test is done continuing this binary search until every bit in the SAR has been tested. The resulting code is the digital approximation of the sampled input voltage and is finally output by DAC at end of the conversion (EOC).

Resolution and conversion time associate with ADC

Resolution:

It is the maximum number of digital output codes.

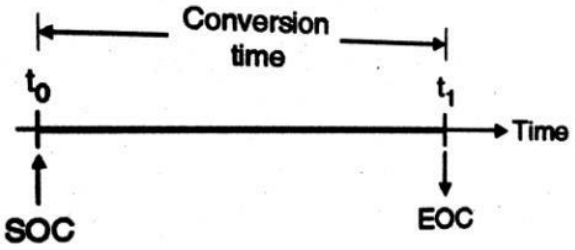
Resolution= 2^n

(OR)

It is defined as the ratio of change in the value of input analog voltage required to change the digital output by 1 LSB.

Conversion time:

The time difference between two instants i.e. 'to' where SOC signal is given as input to the ADC and 't1' where EOC signal we get as output from ADC. It should be small as possible.



Q4. Compare the following (Any three points)

i) Volatile with Non-volatile memory ii) SRAM with DRAM memory [6M]

Parameter	Volatile memory	Non-Volatile memory
definition	Memory required electrical power to keep information stored is called volatile memory	Memory that will keep storing its information without the need of electrical power is called nonvolatile memory.
classification	All RAMs	ROMs, EPROM, magnetic memories
Effect of power	Stored information is retained only as long as power is on.	No effect of power on stored information
applications	For temporary storage	For permanent storage of information

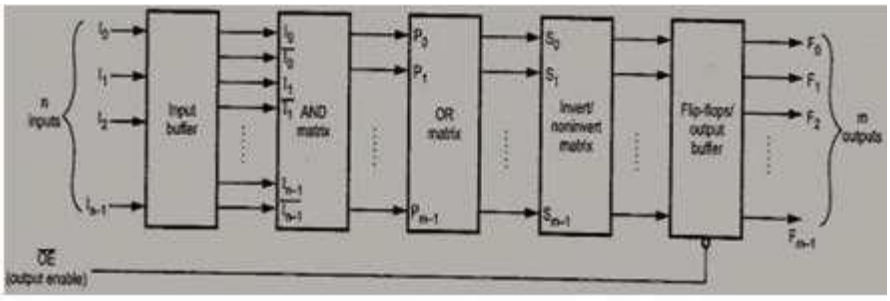
2. SRAM with DRAM memory

Parameter	SRAM	DRAM
Circuit configuration	Each SRAM cell is a flip flop	Each cell is one MOSFET & a capacitor
Bits stored	In the form of voltage	In the form of charges
No of components per cell	More	Less
Storage capacity	Less	More
Refreshing	It does not require refreshing	It require refreshing.
Cost	It is expensive	It is cheaper
Speed	It is faster	It is slower comparatively

Q5. List the types of DAC **[2M]**

1) Binary weighted DAC
2) R –2R ladder network DAC

Q6. Draw block diagram of programmable logic Array. **[2M]**



\

Q7. Compare the following:
(i) Volatile with Non Volatile. **(ii) EPROM with EEPROM.** **[6M]**

(i)Volatile with Non Volatile.

Parameter	Volatile memory	Non-Volatile memory
definition	Memory required electrical power to keep information stored is called volatile memory	Memory that will keep storing its information without the need of electrical power is called nonvolatile memory.
classification	All RAMs	ROMs, EPROM, magnetic memories
Effect of power	Stored information is retained only as long as power is on.	No effect of power on stored information
applications	For temporary storage	For permanent storage of information

ii)EPROM with EEPROM.

Parameter	EPROM	EEPROM.
Stands for	Erasable Programable Read-Only Memory.	Electrically Erasable Programmable Read-Only Memory.
Basic	Ultraviolet Light is used to erase the content of EPROM.	EEPROM contents are erased using electrical signal.
Appearance	EPROM has a transparent quartz crystal window at the top.	EEPROM are totally encased in an opaque plastic case.
Technology	EPROM is modern version of PROM.	EEPROM is the modern version of EPROM.

Q8. Calculate the analog output for 4 bit weighted register type DAC for inputs
(i) 1011
(ii) 1001 Assume (Vfs) full scale range of voltage is 5V [6M]

Given:
VR = Vfs = 5V
Formula Used:
Vo = - VR (B1.2-1 + B2.2-2 + B3.2-3 + B4.2-4)
1. 1011
Vo = - VR (B1.2-1 + B2.2-2 + B3.2-3 + B4.2-4)

= - 5 (1*1/2 + 0 + 1*1/23 + 1 *1/24)

= - 5 (1*1/2 + 1*1/8 + 1 *1/16)
= - 5 (0.5 + 0.125 + 0.0625) = 3.4375V
Vo = 3.4375 V
2. 1001
Vo = - VR (B1.2-1 + B2.2-2 + B3.2-3 + B4.2-4)
= - 10 (1*1/2 + 0 + 0 + 1 *1/24)
= - 10 (1*1/2 + 0 + 0 + 1 *1/16)
= - 10 (0.5 + 0.0625) = 2.8125V
Vo = 2.8125 V

Q9. State two specification of DAC. [2M]

1.Resolution:
Resolution is defined as the ratio of change in analog output voltage resulting from a change of 1 LSB at the digital input VFS is defined as the full scale analog output voltage i.e. the analog output voltage when all the digital input with all digits 1.
Resolution = VFS /(2n −1)
2. Accuracy:
Accuracy indicates how close the analog output voltage is to its theoretical value. It indicates the deviation of actual output from the theoretical value. Accuracy depends on the accuracy of the resistors used in the ladder, and the precision of the reference voltage used. Accuracy is always specified in terms of percentage of the full scale output that means maximum output voltage
3. Linearity:
The relation between the digital input and analog output should be linear.
However practically it is not so due to the error in the values of resistors used for the resistive networks.
4. Temperature sensitivity:
The analog output voltage of D to A converter should not change due to changes in temperature. But practically the output is a function of temperature. It is so because the resistance values and OPAMP parameters change with changes in temperature.

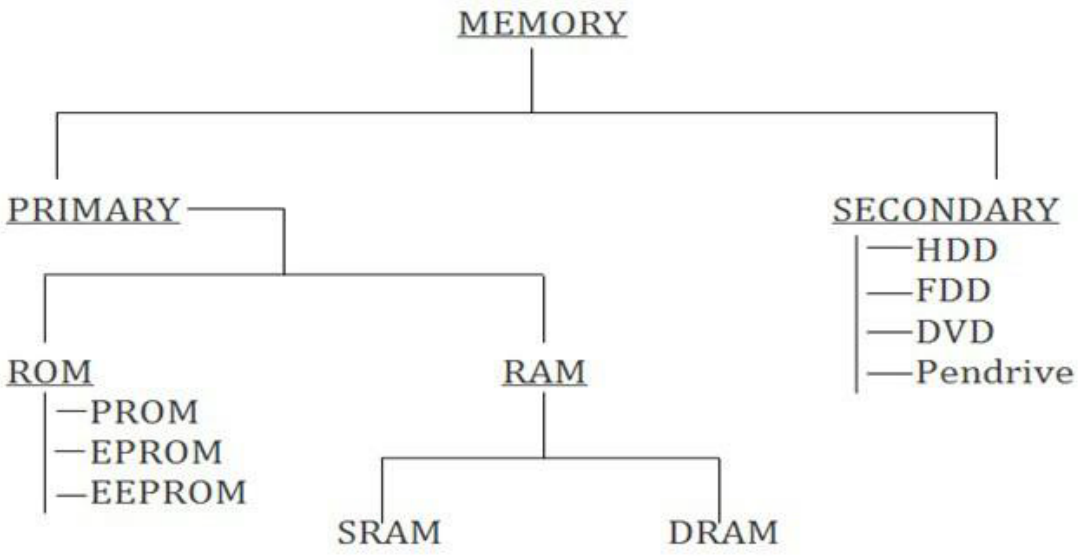
5. Settling time:
The time required to settle the analog output within the final value, after the change in digital input is called as settling time.
The settling time should be as short as possible.

6. Long term drift
Long term drift are mainly due to resistor and semiconductor aging and can affect all the characteristics.
Characteristics mainly affected are linearity, speed etc.

7. Supply rejection
Supply rejection indicates the ability of DAC to maintain scale, linearity and other important characteristics when the supply voltage is varied.
Supply rejection is usually specified as percentage of full scale change at or near full scale voltage at 25oe

8. Speed:
It is defined as the time needed to perform a conversion from digital to analog. It is also defined as the number of conversions that can be performed per second.

Q10. Give classification of memory and compare RAM and ROM (any four points) [4M]



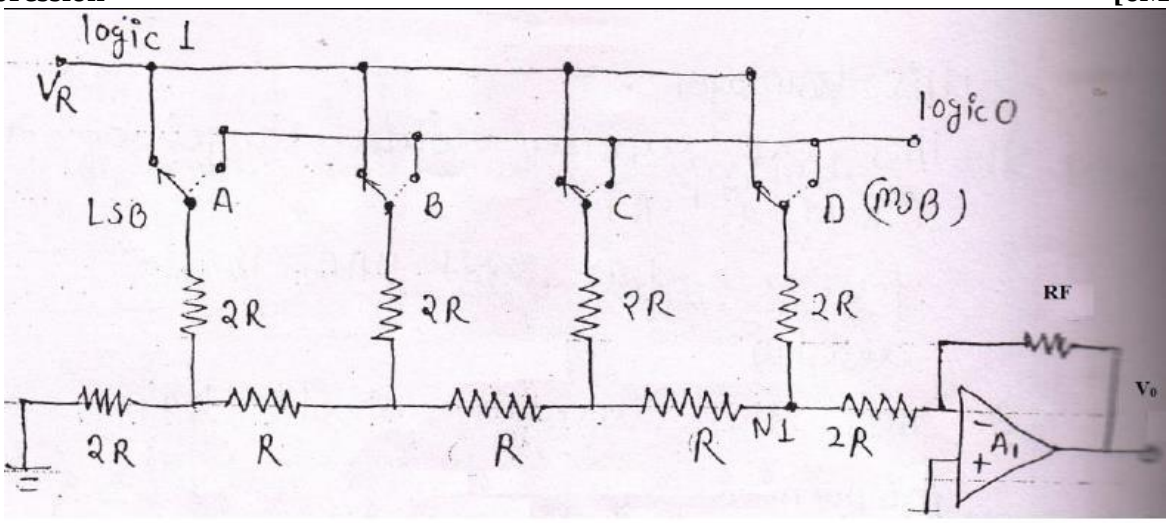
RAM	ROM
Temporary storage	Permanent storage
Store data in Mbs	Store data in Gbs
Volatile	Non Volatile
Writing is faster	Writing is slower

Q11. Compare between PLA and PAL. **[4M]**

\

PLA	PAL
1) Both AND and OR arrays are programmable	1) OR array is fixed and AND array is programmable.
2) Costliest and complex than PAL	2) Cheaper and simpler
3) AND array can be programmed to get desired minterms.	3) AND array can be programmed to get desired minterm.
4) Large number of functions can be implemented.	4) Provides the limited number of functions.
5) Provides more programming flexibility.	5) Offers less flexibility, but more likely used.

Q12. Draw the circuit diagram of 4 bit R-2R ladder DAC and obtain its output voltage expression **[6M]**



Therefore output analog voltage V0 is given by,

$$V_0 = -\left(\frac{R_f}{3R}\cdot\frac{V_R}{2^4}b_0 + \frac{R_f}{3R}\cdot\frac{V_R}{2^3}b_1 + \frac{R_f}{3R}\cdot\frac{V_R}{2^2}b_2 + \frac{R_f}{3R}\cdot\frac{V_R}{2^1}b_3\right)$$

$$V_0 = -\left(\frac{R_f}{3R}\right)\left(\frac{V_R}{2^4}\right)\left[8b_3 + 4b_2 + 2b_1 + b_0\right]$$