MUMBAI UNIVERSITY

SEMESTER 3

Digital Logic Design And Analysis

DECEMBER 2017-CBCS

Q.1(a) Convert $(1762.46)_{10}$ into octal,binary and hexadecimal. [3]

Ans :

(i) $(1762.46)_{10} = (X)_8$ i.e Decimal to octal

		Remainder
8	1762	2
	220	4
	27	3
	3	3

$\therefore (1762)_{10} = (3342)_8$

Fractional part :

0.	46	
	8	
3	68	
	8	
5	44	
	8	
3	52	

$\therefore (0.42)_{10} = (0.353)_8$

$\therefore (1762.46)_{10} = (3342.353)_8$

(ii) $(1762.46)_{10} = (Y)_2$ i.e Decimal to binary

2	1762	0
	881	1
	440	0
	220	0
	110	0
	55	1
	27	1
	13	1
	6	0
	3	1
	1	1

$\therefore (1762)_{10} = (11011100010)_8$

muquestionpapers.com

Fractional part :

0.	46
	2
0	92
	2
1	84
	2
1	68

$\therefore (0.42)_{10} = (0.011)_2$

$\therefore (1762.46)_{10} = (11011100010.011)_2$

(iii) $(1762.46)_{10} = (Z)_{16}$ i.e Decimal to hexadecimal

		Remainder
16	1762	2
	110	E
	6	6

$\therefore (1762)_{10} = (6E2)_{16}$

Fractional part :

0.	46
	16
7	36
	16
5	76
	16
C	16

$\therefore (0.42)_{10} = (0.75C)_2$

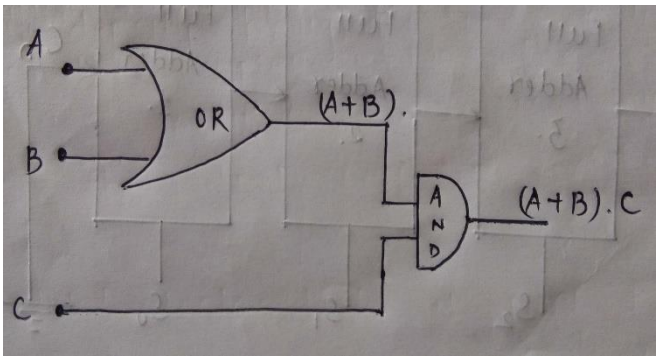
$\therefore (1762.46)_{10} = (6E2.75C)_2$

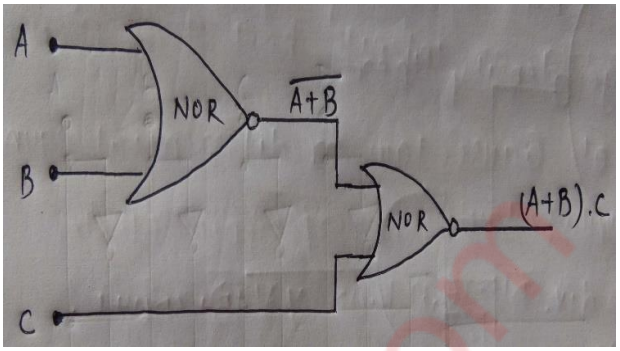
(b) Prove OR-AND configuration is equivalent to NOR-NOR configuration. **[3]**

Ans :

The expression for OR gate output is y=A+B.
The expression for AND gate output is y=A.B
The expression for NOR gate output is y=$\overline{A+B}$

muquestionpapers.com





From this we can prove that OR-AND configuration is same as NOR-NOR Configurations.

(c) Perform substration using 16's complement. **[4]**

(i) $(\mathbf{CB}10)_{16} - (971)_{16}$

(ii) $(426)_{16} - (\mathbf{DBA})_{16}$

Ans : (i) $(971)_{16}$ 15's complement is :

$(\mathbf{CB}10)_{16} - (971)_{16} = (\mathbf{CB}10)_{16} + (971)_{16} = C19F$

$(\mathbf{CB}10)_{16} - (971)_{16}$ = C19F

(ii) $(\mathbf{DBA})_{16}$ 15's complement is : 245

426 + 245 = −994 IN HEXA DECIMAL

$(426)_{16} - (\mathbf{DBA})_{16}$= -994

(d) Find 8's complement of following numbers.

(i) $(\mathbf{27})_8$ **(i)** $(\mathbf{321})_8$ **[2]**

Ans : 8's complement is 1 + 7's complement of number.

To calculate 7's complement substract each digit of number from 7

(i) $(\mathbf{27})_8$= (77-27) + 1 = 50 + 1 =51

$\therefore (\mathbf{27})_8 = (\mathbf{51})_8$

(ii) $\therefore (\mathbf{321})_8 = 777 - 321 = 456 + 1 = 457$

$\therefore (\mathbf{321})_8 = (\mathbf{457})_8$

noquestionpapers.com

(e) Perform following substraction $(52)_{10} - (65)_{10}$ using 2's complement Method. **[2]**

Ans : $(52)_{10} = (110100)_2$

$(65)_{10} = (1000001)_2$

2's complement of $(65)_{10}$ is (0111110)+1=0111111

$\therefore (52)_{10} - (65)_{10} = (110100)_2 - (0111111)_2 = -1101$

We can represent -1101 in decimal as -13 .

$$\therefore \quad (52)_{10} - (65)_{10} = (-13)_{10}$$

(f) Write hamming code for 1010. **[2]**

Ans : Given code W=(1 0 1 0)

Hamming code is given by ,

X=W.G

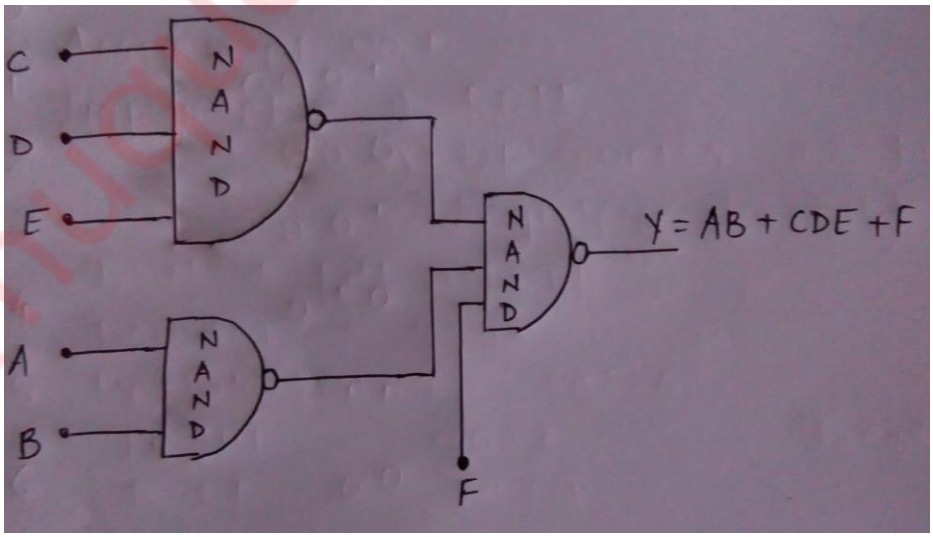
$$X=(\,1 \quad 0 \quad 1 \quad 0\,) \cdot \begin{bmatrix} 1 & 0 & 00 & 1 & 10 \\ 0 & 1 & 00 & 1 & 01 \\ 0 & 0 & 10 & 0 & 1\ 1 \\ 0 & 0 & 01 & 1 & 1\ 1 \end{bmatrix}$$

X = (1 0 1 0 0 0 1)

(g) Implement the following Boolean equation using NAND gates only.

Y= AB+CDE+F **[2]**

Ans : In this y can be implemented using 3 input nand and 2 input nand gates.



muquestionpapers.com

(h) Explain the term prime implicant. **[2]**

Ans :

i) A group of related 1's (implicant) on a Karnaugh map which is not subsumed by any other implicant in the same map. Equivalently (in terms of Boolean algebra), a product term which is a "minimal" implicant in the sense that removing any of its literals will yield a product term which is not an implicant (on a Karnaugh map it would appear "maximal").

ii) A group of related 0's (implicant) on a Karnaugh map which is not subsumed by any other implicant (of 0's) in the same map.

Q.2(a) Design a 4-bit ripple adder. **[10]**

Ans :

(i) Multiple full adder circuits can be cascaded in parallel to add an N-bit number. For an N- bit A parallel A adder, there must be N number of full adder circuits.

(ii) A ripple carry adder is a logic circuit in which the carry-out of each full adder is the carry in of the succeeding next most significant full adder. It is called a ripple carry adder because each carry bit gets rippled into the next Â stage.

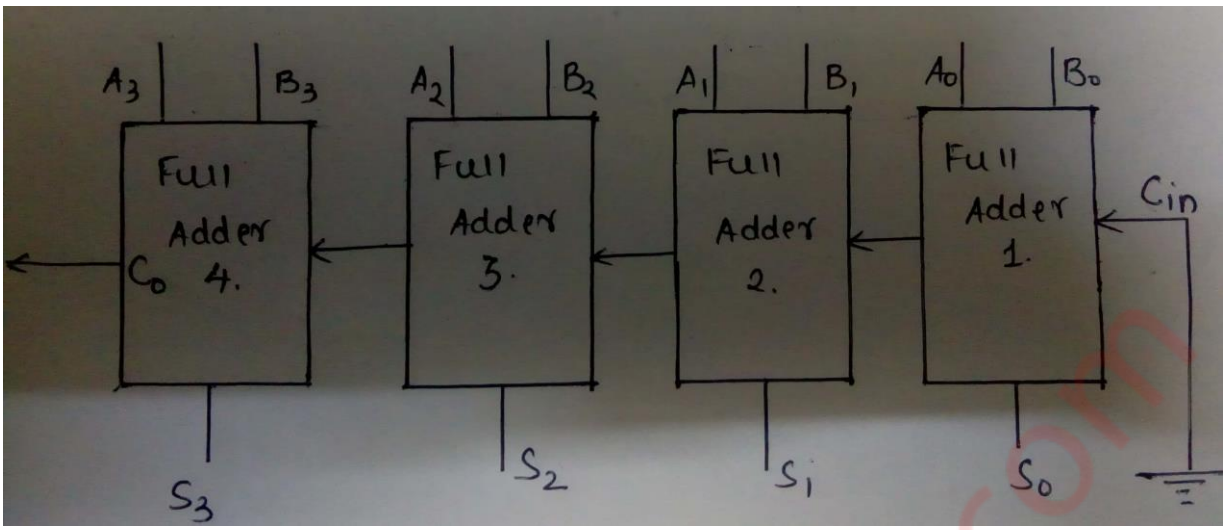
(iii)In a ripple carry adder the sum and carry out bits of any half adder stage is not valid until the carry in of that stage occurs.

(iv)Propagation delays inside the logic circuitry is the reason behind this. Propagation delay is time elapsed between the application of an input and occurance of the corresponding output.

(v)Consider a NOT gate, When the input is “0” the output will be “1” and vice versa. The time taken for the NOT gate’s output to become “0” after the application of logic “1” to the NOT gate’s input is the propagation delay here.

(v)Similarly the carry propagation delay is the time elapsed between the application of the carry in signal and the occurance of the carry out (Cout) signal.

muquestionpapers.com



(vi) Sum out S0 and carry out Cout of the Full Adder 1 is valid only after the propagation delay of Full Adder 1.

(vii) In the same way, Sum out S3 of the Full Adder 4 is valid only after the joint propagation delays of Full Adder 1 to Full Adder 4. In simple words, the final result of the ripple carry adder is valid only after the joint propogation delays of all full adder circuits inside it.

(b) Obtain the minimal expression using QuineMc-Cluskey method.

F(A,B,C,D)=∑m(1,5,6,12,13,14) + **d**(2,4) **[10]**

Ans :

Step 1: Write down the binary equivalent of the given expressions (including the don't care):

	A	B	C	D
1	0	0	0	1
2	0	0	1	0
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0

muquestionpapers.com

Step 2:
Create the following table in which the binary number with equal number of 1's are grouped together. For e.g. the first group contains binary numbers with only one number of 1's, second group contains binary numbers with two 1's and so on.

GROUP	MINTERM	BINARY REPRESENTATION				
		A	B	C	D	
1	m1	0	0	0	1	√
	d2	0	0	1	0	√
	d4	0	1	0	0	√
2	m5	0	1	0	1	√
	m6	0	1	1	0	√
	m12	1	1	0	0	√
3	m13	1	1	0	1	√
	m14	1	1	1	0	√

Step 3: In the next step compare every element of group 1 with elements in group 2, elements of group 2 with 3, etc. In general compare elements of nth group with elements of n+1th group. If only 1 element is changing in the comparison the mark it with an underscore. Put a tick mark against both the elements if the elements are present in the comparison. For e.g. in m1 and m5 the A, C, D bits are same but only B bit changes, so an underscore is put against it.

GROUP	MATCHED PAIRS	BINARY REPRESENTATION				
		A	B	C	D	
1	m1 m5	0	_	0	1	
	d2 m6	0	_	1	0	
	d4 m5	0	1	0	_	√

muquestionpapers.com

	d4 m6	0	1	_	0	√
	d4 m12	_	1	0	0	√
2	m5 m13	_	1	0	1	√
	m6 m14	_	1	1	0	√
	m12 m13	1	1	0	_	√
	m12 m14	1	1	_	0	√

Step 4: In this step again match the values of nth group with n+1th group and mark tick against the ones that gets matched, followed by underscore to the single bits that change. Since m1 m5, d2 m6 are not getting matched with any other no tick marks is put against them.

GROUP	MATCHED PAIRS	BINARY REPRESENTATION			
		A	B	C	D
1	d4 m5 m12 m13	_	1	0	_
	d4 m6 m12 m13	_	1	_	0

Step 5: Now make the last table which contains the prime implicant and the min terms involved (Prime implicants are the ones that do not get matched). Write them in their alphabet form with 0 bit represented as bars. In this step write all the numbers which were given in the question (not the don't care ones). Put a cross below the numbers which contain the minterms.

Prime Implicant	Minterms Involved	1	5	6	12	13	14
BC	4, 5, 12, 13		×		×	×	
BD	4, 6, 12, 14			×	×		×
ACD	1, 5	×	×				
ACD	2, 6			×			

mquestionpapers.com

If the column of the numbers contain only one cross then the prime implicant belonging to the single cross belonging row is considered. In this case the columns of 1, 13 and 14 contain only 1 crosses so the row belonging to them is considered which are BC, BD, ACD. These are the reduced answer.

Therefore the final answer is:

$$Y = BC + BD + ACD$$

$$\therefore F(A,B,C,D) = BC + BD + ACD$$

Q.3(a) Implement full adder using 8:1 multiplexer. [10]

Ans :

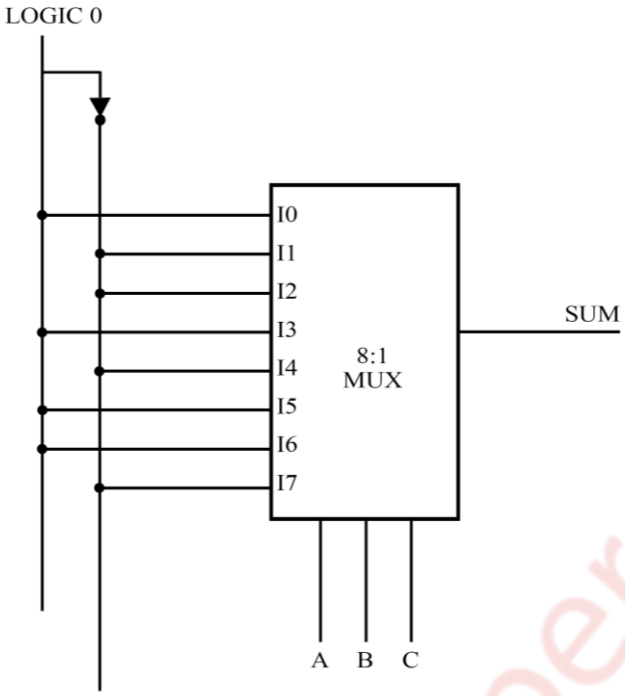
3 bit binary adder is adder circuit which performs bit by bit addition and gives output as sum and carry .The truth table for a full adder :

A	B	C	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

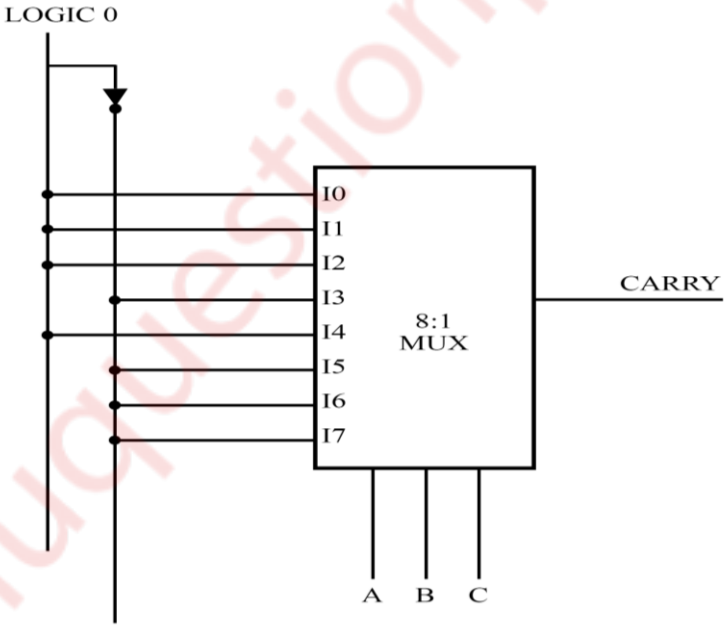
We will require two 8:1 multiplexer to implement a full adder. One for sum and one for carry.

muquestionpapers.com

In 8:1 MUX we have 3 selection lines, assigning them A, B, C.For sum :



For carry :



muquestionpapers.com

(b) Implement the following functions using demultiplexer.

F1(A,B,C)=∑m(0,3,7) F2(A,B,C)=∑m(1,2,5) [5]

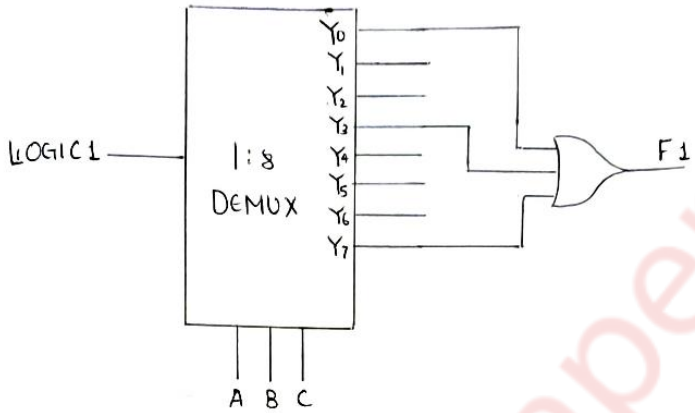
Ans :

(1) F1 (A, B, C) = Σm(0, 3, 7)

Number of variables = 3

Number of select lines = 3

Number of lines = 2^3 = 8

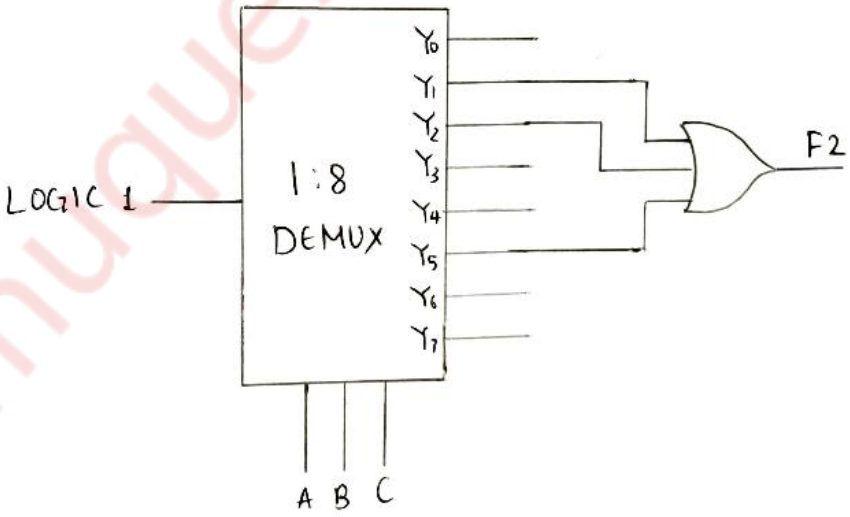


(2) F2 (A, B, C) = Σm(1, 2, 5)

Number of Variables = 3

Number of Select Lines = 3

Number of Lines = 2^3 = 8



muquestionpapers.com

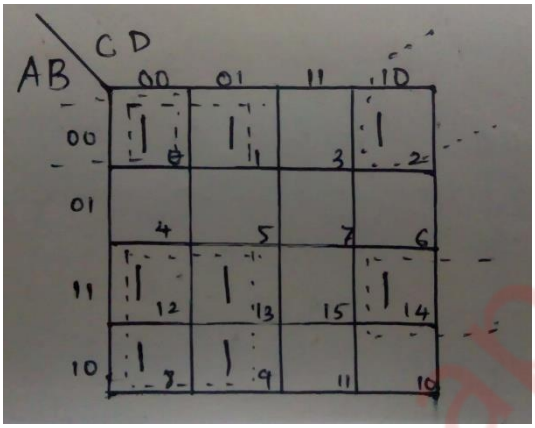
(c) Simplify F(A,B,C,D)=∏M(3,4,5,6,7,10,11,15) and implement using minimum number of gates. **[5]**

Ans :

We can write same function using sum of product way ,

F(A,B,C,D)=∏m(0,1,2,8,9,12,13,14)

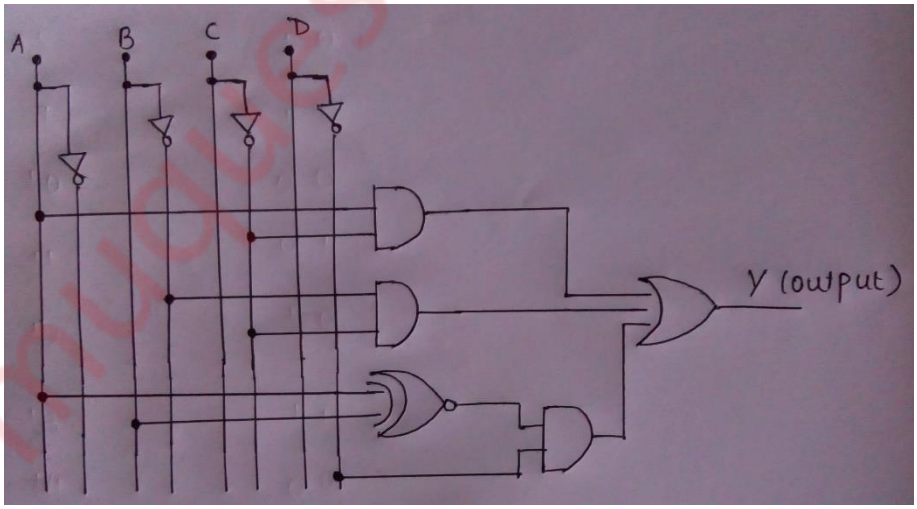
K-map is given by



The expression after grouping :

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

Circuit Diagram :



muquestionpapers.com

Q.4(a) Compare TTL and CMOS logic withn respect to fan in,fan out Propogation delay,power consumption,noise margin,current and Voltage parametes. [5]

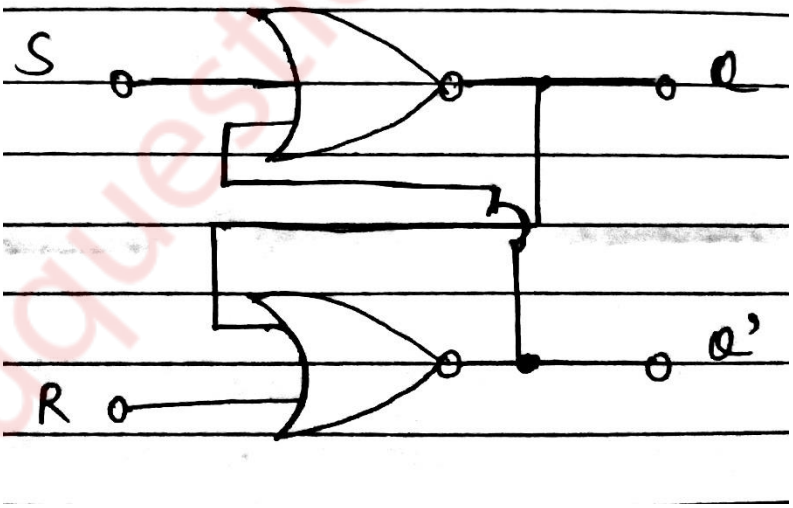
Ans :

Parameter	TTL	CMOS
Propagation Delay	1 to 200nsec	1.5 to 33nsec
Power consumption	Relatively High	Depends on Vcc
Noise Margin	0.3-0.5	0.3Vcc
Current	High (0.2-2mA)	Low (1uA)
Voltage	5V	3-18V
Fan in	High fan in	Low fan in
Fan out	Approx. 10	>50
Transistor used	BJT	FET
Type	Current controlled Current devices	Voltage controlled current device

(b) Draw the circuit of SR flip flop using two NOR gates and write the Architecture body for the same using structural modelling. [5]

Ans :

SR flip flop using two NOR gate :



SR FLIP FLOP

muquestionpapers.com

Architectural Body :

```
begin
PROCESS(CLOCK)
 variable tmp: std_logic;
 begin
 if(CLOCK='1' and CLOCK'EVENT) then
 if(S='0' and R='0')then
 tmp:=tmp;
 elsif(S='1' and R='1')then
 tmp:='Z';
 elsif(S='0' and R='1')then
 tmp:='0';
 else
     tmp:='1';
 end if;
 end if;
 Q <= tmp;
 QBAR <= not tmp;
end PROCESS;
end behavioral;
```

muquestionpapers.com

(c) Explain 1-digit BCD adder. **[10]**

Ans :

i) BCD adder adds two BCD digits and produces output as a BCD digit. A BCD or Binary Coded Decimal digit cannot be greater than 9.

ii)The two BCD digits are to be added using the rules of binary addition. If sum is less than or equal to 9 and carry is 0, then no correction is needed. The sum is correct and in true BCD form.

iii)But if sum is greater than 9 or carry =1, the result is wrong and correction must be done.

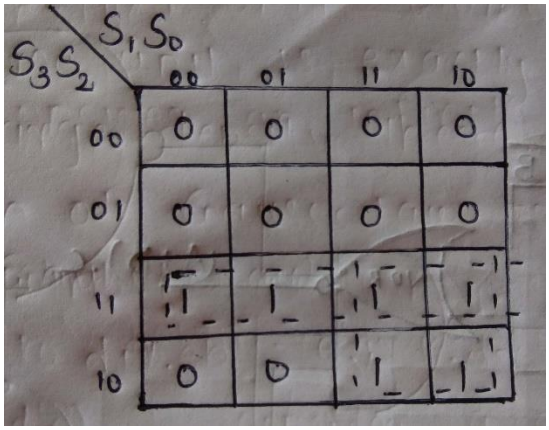
iv)The wrong result can be corrected adding six (0110) to it.

v)For implementing a BCD adder using a binary adder circuit IC 7483, additional combinational circuit will be required, where the Sum output S3−S0S3−S0 is checked for invalid values from 10 to 15.

vi) Truth Table :

S3	S2	S1	S0	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

muquestionpapers.com



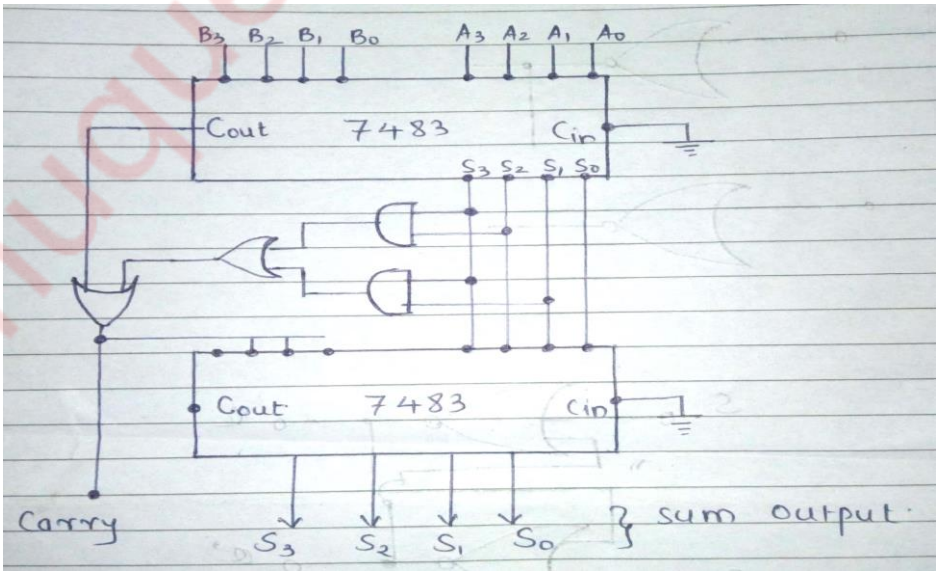
vi) The Boolean expression is, Y=S3S2+S3S1Y=S3S2+S3S1

vii)The BCD adder is shown below. The output of the combinational circuit should be 1 if Cout of adder-1 is high. Therefore Y is ORed with Cout of adder 1.

viii)The output of combinational circuit is connected to B1B2 inputs of adder-2 and B3=B1+0B3=B1+0 as they are connected to ground permanently. This makes B3B2B1B0B3B2B1B0 = 0110 if Y' = 1.

ix)The sum outputs of adder-1 are applied to A3A2A1A0A3A2A1A0 of adder-2.

x)The output of combinational circuit is to be used as final output carry and the carry output of adder-2 is to be ignored.



muquestionpapers.com

Q.5 (a) Convert JK flip flop to SR flip flop and D filp flop. **[10]**

*Ans: **(A) JK Flip flop to SR flip flop :***

1.Truth table of SR FF:

S	R	Q	$\bar{Q}$
0	0	X	X
0	1	0	1
1	0	1	0
1	1	-	-

2.Excitation Table of JK FF:

Qn	Q_{n+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

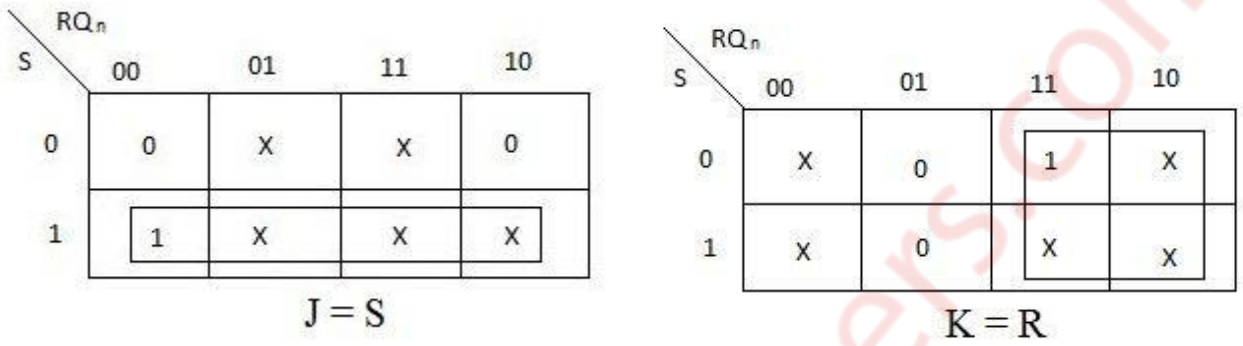
3.Conversion Table:

S	R	Q_n	Q_{n+1}	J	K
0	0	0	0	0	X
0	1	1	1	X	0
0	0	0	0	0	X
0	1	1	0	X	1
1	0	0	1	1	X
1	1	1	1	X	0
1	0	-	-	X	X
1	1	-	-	X	X

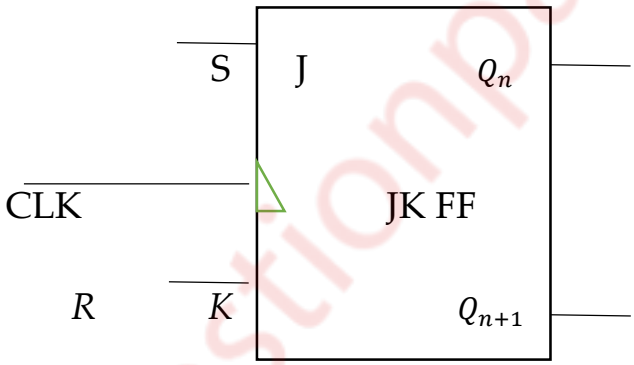
muquestionpapers.com

4. *Logical expressions for the inputs :*

Simplify the logical expressions for the inputs of the given flip-flop (J and K) in terms of the inputs of the desired flip-flop (S and R) and the flip-flop's present-state, Q_n.This can be done by following any logical simplification technique like that of the K-map.



5. *Circuit Diagram :*



(B) JK TO D FF :

1.Truth Table of D flip flop :

D	Q	$\bar{Q}$
0	0	0
0	1	0
1	0	1
1	1	1

muquestionpapers.com

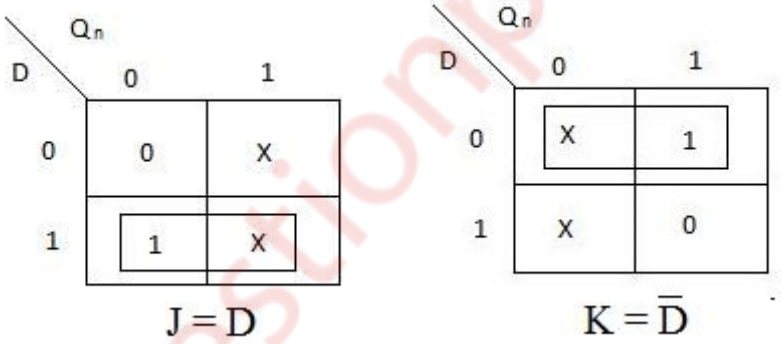
2.Excitation Table of JK FF:

Qn	Q_{n+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

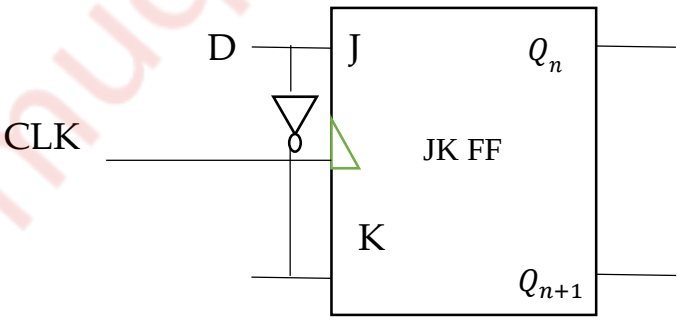
3.Conversion Table :

D	Q_n	Q_{n+1}	J	K
0	0	0	0	X
0	1	1	0	1
1	0	0	1	X
1	1	0	1	0

4.Use a K-map to obtain the logical expressions for the inputs J and K in terms of D and Qn.



5. Circuit Diagram:



muquestionpapers.com

(b) Design 3 bit synchronous counter using T flip flops. [10]

Ans : (i) A synchronous counter, in contrast to an asynchronous counter, is one whose output bits change state simultaneously, with no ripple.

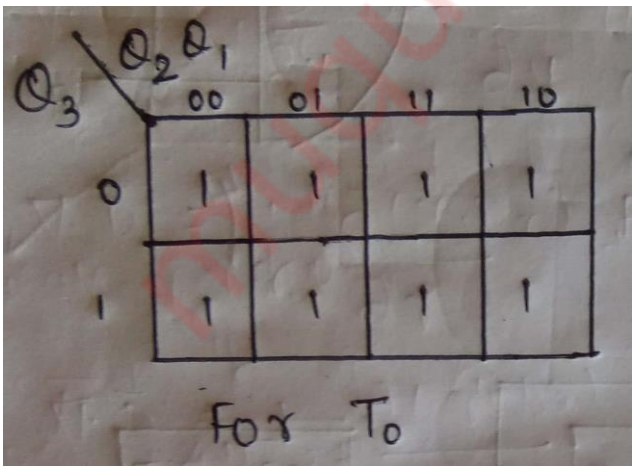
(ii) The only way we can build such a counter circuit from T flip-flops is to connect all the clock inputs together, so that each and every flip-flop receives the exact same clock pulse at the exact same time.

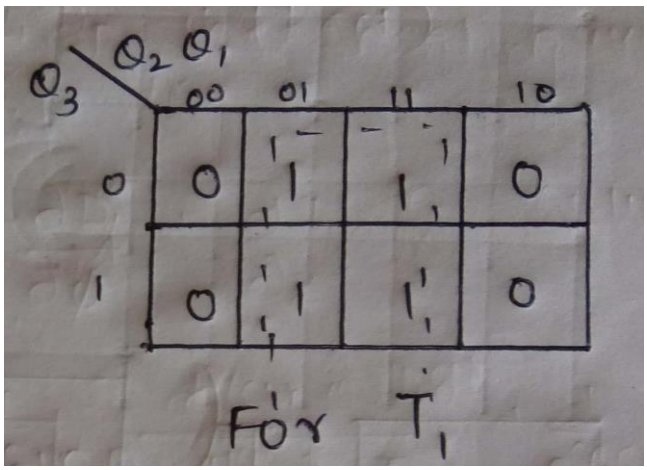
(iii) For an up counter that means it will count from value 0 to 7 for interval difference of 1.

(iv) Truth table for up counter is given by ,

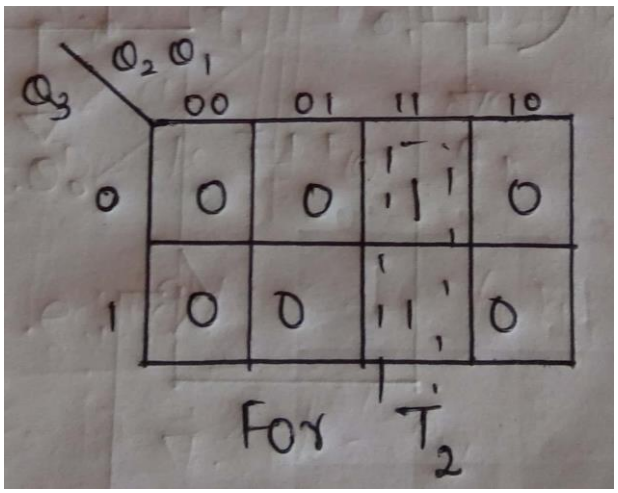
Present State	Next State	T2	T1	T0
000	001	0	0	1
001	010	0	1	1
010	011	0	0	1
011	100	1	1	1
100	101	0	0	1
101	110	0	1	1
110	111	0	0	1
111	000	1	1	1

(v) From this truth table :





mquestionpapers.com

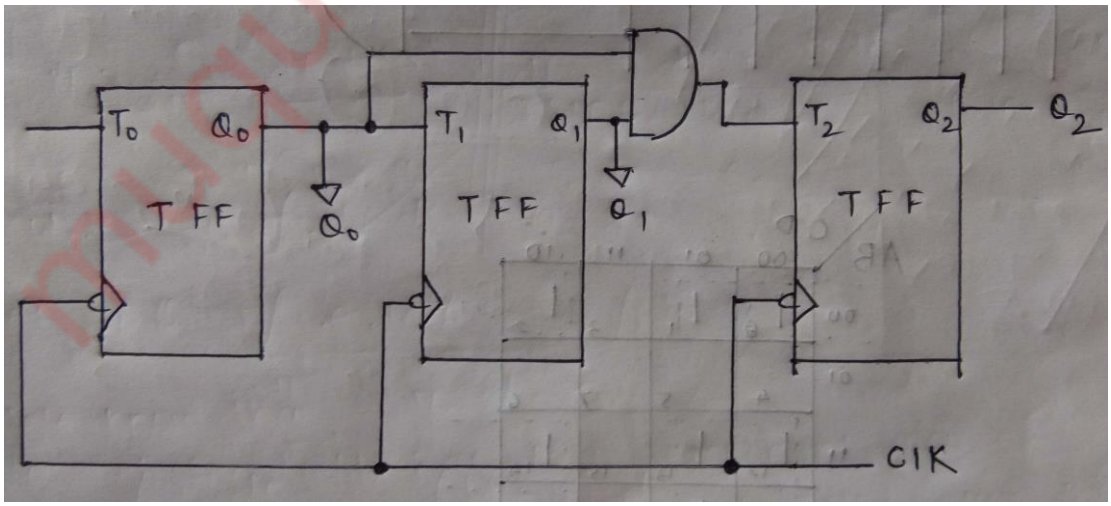


(vii)The 3-bit Synchronous binary up counter contains three T flip-flops & one 2-input AND gate. All these flip-flops are negative edge triggered and the outputs of flip-flops change (affect) synchronously. The T inputs of first, second and third flip-flops are 1, Q0Q0 & Q1Q0Q1Q0 respectively.

(viii)The output of first T flip-flop **toggles** for every negative edge of clock signal.

(ix)The output of second T flip-flop toggles for every negative edge of clock signal if Q0Q0 is 1.

(x)The output of third T flip-flop toggles for every negative edge of clock signal if both Q0Q0 & Q1Q1 are 1.
(Here we can write Q3Q2Q1=Q2Q1Q0).



muquestionpapers.com

Q.6 Write short note on (any four) **[20]**

(a) State table :

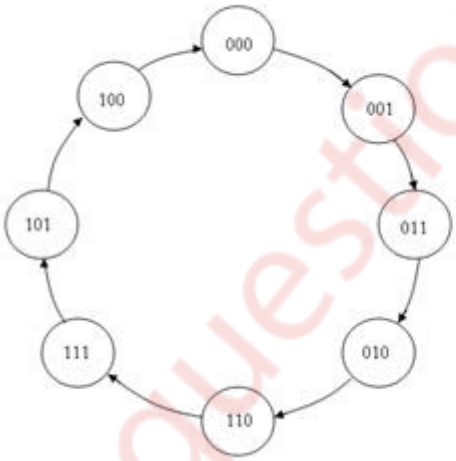
Ans :

i) The state table representation of a sequential circuit consists of three sections labeled *present state*, *next state* and *output*.

Ii)The present state designates the state of flip-flops before the occurrence of a clock pulse.

iii) The next state shows the states of flip-flops after the clock pulse, and the output section lists the value of the output variables during the present state.

iv) In addition to graphical symbols, tables or equations, flip-flops can also be represented graphically by a state diagram.
v) In this diagram, a state is represented by a circle, and the transition between states is indicated by directed lines (or arcs) connecting the circles.



Present State			Next State		
Q2	Q1	Q0	Q2	Q1	Q0
0	0	0	0	0	1
0	0	1	0	1	1
0	1	1	0	1	0
0	1	0	1	1	0
1	1	0	1	1	1
1	1	1	1	0	1
1	0	1	1	0	0
1	0	0	0	0	0

(b) ALU IC 74181

Ans :

i)The **74181** is a bit slice arithmetic logic unit (ALU), implemented as a 7400 series TTL integrated circuit.

ii)The first complete ALU on a single chip, it was used as the arithmetic/logic core in the CPUs.

muquestionpapers.com

iii) The 74181 is a 7400 series medium-scale integration (MSI) TTL integrated circuit, containing the equivalent of 75 logic gates and most commonly packaged as a 24-pin DIP.

iv) The 4-bit wide ALU can perform all the traditional add / subtract / decrement operations with or without carry, as well as AND / NAND, OR / NOR, XOR, and shift.

v) Many variations of these basic functions are available, for a total of 16 arithmetic and 16 logical operations on two four-bit words. Multiply and divide functions are not provided but can be performed in multiple steps using the shift and add or subtract functions.

vi)Shift is not an explicit function but can be derived from several available functions; e.g., selecting function "A plus A" with carry (M=0) will give an arithemetic left shift of the A input.

(c) Sequence generator :

Ans :

i)There are counters which pass through a definite number of states in a pre-determined order. For example, a 3-bit up-counter counts from 0 to 7 while the same order is reversed in the case of 3-bit down counter.

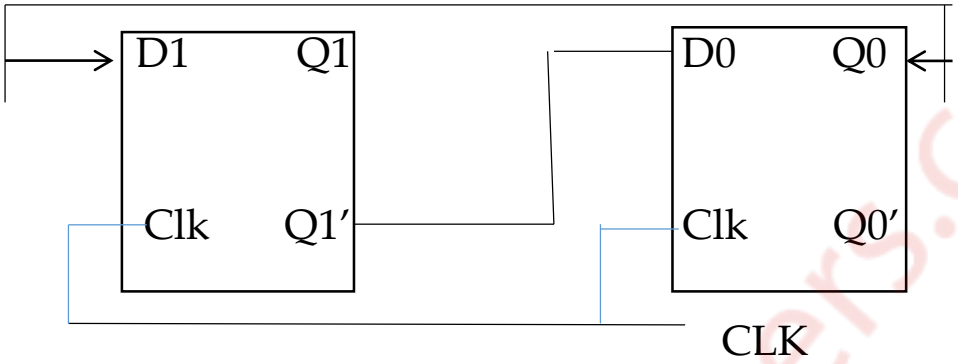
ii) These circuits when suitably manipulated can be made to count till an intermediate level also. This means that instead of counting till 7, we can terminate the process by resetting the counter just at, say, 5. Such counters are then known as mod-N counters.

iii)We can design sequence generator using some D flip flops.
It will generate sequence from some given input binary digit.

iv) Sequence generator :

muquestionpapers.com

Q_1	Q_0	Q_1*	Q_0*	D_1	D_0
0	0	0	1	0	1
0	1	1	1	1	1
1	1	1	0	1	0
1	0	0	0	0	0



Sequence Generator

(d) Data flow modelling
Ans :

i)The process of identifying, modeling and documenting how data moves around an information system. Data flow modeling examines processes (activities that transform data from one form to another), data stores (the holding areas for data), external entities (what sends data into a system or receives data from a system, and data flows (routes by which data can flow).

ii)Data flow modeling is one of the foundations of the Structured Systems Analysis and Design Method.

iii)Data flow modeling also is called data flow diagramming.

muquestionpapers.com

(e) 4 bit ring counter

Ans :

i) A ring counter is a Shift Register (a cascade connection of flip-flops) with the output of the last flip flop connected to the input of the first.

ii)It is initialised such that only one of the flip flop output is 1 while the remander is 0.

iii)The 1 bit is circulated so the state repeats every n clock cycles if n flip-flops are used. The MOD of a counter is the number of unique states. The MOD of the n flip flop ring counter is n.

iv)It can be implemented using D-type flip-flops (or JK-type flip-flops).

v) 4-bit Ring Counter :

Truth Table :

CLK	Q_3	Q_2	Q_1	Q_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

muquestionpapers.com