Unit-1 Number System

The process of assigning zero's and one's to the voltages level is known as logic levels and it's of 2 types :-

(a) +ve logic level < Here ON = 1 ; OFF = 0 >
(b) -ve logic level < Here ON = 0 ; OFF = 1 >

Decimal Numbers :- (0-9) (Base 10)

(a) Decimal to Binary :-

$(76)_{10} = (1001100)_2$

2	76
2	38 – 0
2	19 – 0
2	9 – 1
2	4 – 1
2	2 – 0
	1 – 0

LSB (Least sig. bit)
MSB (Most significant Bit)

(b) Decimal to Octal

$(76)_{10} = (114)_8$

8	76
8	9 – 4
	1 – 1

(c) Decimal to ~~octal~~ Hexadecimal

$(76)_{10} = (4\ C)_{16}$

16	76
	4 – 12

A – 10
B – 11
C – 12
D – 13
E – 14
F – 15

Binary (Base 2) ; Octal (Base 8);
Hexadecimal (Base 16)

Conversion from other no. system to decimal.

(a) Binary to decimal

$(\underset{5}{1}\ \underset{4}{0}\ \underset{3}{0}\ \underset{2}{1}\ \underset{1}{1}\ \underset{0}{1})_2 = (39)_{10}$

$(1\times 2^0) + (1\times 2^1) + (1\times 2^2) + (0\times 2^3) + (0\times 2^4) + (1\times 2)^5$
$= 1 + 2 + 4 + 32 = 39$

(b) Octal to decimal :-

$(\underset{10}{46})_8 \longrightarrow (38)_{10}$

$(6\times 8^0) + (4\times 8^1) = 6 + 32 = 38$

Hexadecimal to decimal :-

$(70)_{16} \longrightarrow (112)_{10}$

$(0 \times 16^0) + (7 \times 16^1) = 0 + 112 = 112$

Binary to octal

$(\overbrace{100}^{4}\overbrace{111}^{7})_2 \longrightarrow (47)_8$

Binary to Hexadecimal

$(\underbrace{00\,10}_{2}\,\underbrace{0111}_{7})_2 \longrightarrow (27)_{16}$

	Binary
	$0 \to 0$
	$1 \to 1$
	$2 \to 10$
	$3 \to 11$
	$4 \to 100$
	$5 \to 101$
	$6 \to 110$
	$7 \to 111$
	$8 \to 1000$
	$9 \to 1001$
A	$10 \to 1010$
B	$11 \to 1011$
C	$12 \to 1100$
D	$13 \to 1101$
E	$14 \to 1110$
F	$15 \to 1111$

Signed Numbers :-

- In this, we need ~~bit~~ 4-bits upto 7
- In signed magnitude no., an extrabit is added in MSBS (most significant bit side)

zero is added for +ve no's
1 is added for -ve no's

→ Another method to represent -ve decimal no's is ones compliment method.
eg. 3 is 0011 but in ones compliment its 1100
- because its reverse.

→ Another method is Two's compliment.
Here we add 1 in ones compliment of a no. to get two's compliment.
eg:- $1+1 = 10$ $1+1+1 = 11$

$n! = n \times (n-1) \times (n-2) ---$

$5! = 5 \times 4 \times 3 \times 2 \times 1$

$0! = 1$

Logic Gates :- divided into two categories
(a) Basic gates ; (b) Universal gates

1-august-2018

Boolean Algebra :- It contains boolean functions which have expressions having boolean variables.

1. It is used for analysing a logic circuit.
2. It is used to give its mathematical operation.
3. It is used to simplify logic circuits and logic expression

Boolean Laws :-

Law	AND law	OR law
1. Identity law	$A \cdot 1 = A$	$A + 0 = A$
2. Null Element law	$A \cdot 0 = 0$	$A + 1 = 1$
3. Indempotent law	$A \cdot A = A$	$A + A = A$
4. Commutative law	$A \cdot B = B \cdot A$	$A + B = B + A$
5. Associative law	$A \cdot (B \cdot C) = (A \cdot B) \cdot C$	$A + (B + C) = (A + B) + C$
6. Distributive law	$A + (B \cdot C) = (A + B) \cdot (A + C)$	$A \cdot (B + C) = (A \cdot B) + (A \cdot C)$
7. Absorption law	$A + A \cdot B = A$	$A \cdot (A + B) = A$
8. Double Inversion / Involution law	$\overline{\overline{A}} = A$	
9. Consensus law	$AB + \bar{A}C + BC = AB + \bar{A}C$	

Imp 10. De Morgan's Theorem :-

$\overline{A+B} = \bar{A} \cdot \bar{B}$; $\overline{A \cdot B} = \bar{A} + \bar{B}$

Some other laws :-

1. $A + \bar{A}B = A + B$
2. $\bar{A} + AB = \bar{A} + B$
3. $A + \bar{B} \cdot \bar{A} = A + \bar{B}$

Proof and Verification of Booleans law :

1. $A + (B \cdot C) = (A+B) \cdot (A+C)$

Proof. $R.H.S = (A+B) \cdot (A+C)$

$= A \cdot A + A \cdot C + B \cdot A + B \cdot C$

Acc. to Indempotent law $A \cdot A = A$

$\therefore = A + \cancel{A+} A \cdot C + B \cdot A + B \cdot C$

$= A(1+C) + A \cdot B + B \cdot C$

Acc. to Null element law

$A + 1 = 1 \quad \therefore 1 + C = 1$

$\therefore = A \cdot 1 + A \cdot B + B \cdot C$

$= A(1+B) + B \cdot C$

$= A(1) + B \cdot C$

$= A + B \cdot C = L.H.S.$ Hence Proved

3-August

2. $AB + \bar{A}C + BC = AB + \bar{A}C$

$\Rightarrow L.H.S = AB + BC + \bar{A}C$

$= A \cdot B + B \cdot C \cdot (1) + \bar{A} \cdot C$

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

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

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

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

$= AB + \bar{A}C = R.H.S.$

3. Show that $A + AB = A$

$\Rightarrow L.H.S = A + AB$

$= A(1+B)$

$= A \cdot 1 = A = R.H.S.$

De-Morgan's Theorem.

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

It means that $\bar{A} \cdot \bar{B}$ is equal to the complement of $A+B$

∴ We can write:

$$(A+B) + (\bar{A} \cdot \bar{B}) = 1$$

On applying distributive law, we get

$$(A+B+\bar{A}) \cdot (A+B+\bar{B}) = 1$$

$$(1+B) \cdot (1+A) = 1$$

$$1 \cdot 1 = 1.$$

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

L.H.S.

$$(A \cdot B) \cdot (\bar{A} + \bar{B}) = 0$$

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

$$= 0 \cdot B + A \cdot 0 = 0.$$ Proved.

2nd method. By substituting various values

Sum of Product (SOP) ; Product of Sum (POS)

SOP: Product term of a funct. are added in this form also known as Min Term & denoted by 'm'.

POS:- Sum terms of a funct. are multiplied with each other in this form. also known as Max terms, denoted by 'M'.

eg.

A	B	C	SOP	POS
0	0	0	$\bar{A} \cdot \bar{B} \cdot \bar{C}$	$A+B+C$
0	0	1	$\bar{A} \cdot \bar{B} \cdot C$	$A+B+\bar{C}$
		⋮	⋮	⋮

Net $\Sigma m = \bar{A} \cdot \bar{B} \cdot \bar{C} + \bar{A}\bar{B} \cdot C$ -- $\quad \Pi m = (A+B+C)(A+B+\bar{C}$

And Gate :.

A	B	$A \cdot B = Y$	SOP	POS
0	0	0		$A+B$
0	1	0		$A+\bar{B}$
1	0	0		$\bar{A}+B$
1	1	1	$A \cdot B$	
		Net	$A \cdot B$	$(A+B)(A+\bar{B})(\bar{A}+B)$

SOP — only Y = 1 terms will be included

POS — only Y = 0 terms will be included.

7 - August - 2018

⟹ Standard and Non-standard ~~fo~~ POS & SOP forms.
Standard form is also known as canionical form.

eg. $Y = AB + AB\bar{C} + \bar{A}BC$ → Non-stand. form SOP

$Y = AB + \bar{A}B + \bar{A}\bar{B}$ → Stand. form SOP

$Y = (\bar{A}+B) \cdot (A+B) \cdot (A+\bar{B})$ → stand. form POS

$Y = (\bar{A}+B) \cdot (A+B+C)$ → Non-stand-form POS.

⟹ Conversion from Non-stand POS & SOP to canionical/standard form.

1. SOP :-

Rules: 1. For each term find the missing literal

2. AND ing the terms with the term formed by OR ing the missing literal & its complement.

eg:- $Y = \underset{C}{AB} + \underset{B}{A\bar{C}} + \underset{A}{BC}$ ← missing.

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

$= ABC + AB\bar{C} + AB\bar{C} + A\bar{B}\bar{C} + ABC + \bar{A}BC$

2. POS :-

Rules: (a) For each term find the missing literal

(b) OR ing the each term by the term formed by AND'ing the missing literal with its complement.

eg. $Y = (A+B)(A+C)(B+\bar{C})$ ← missing C, B, A
$= (A+B+C\cdot\bar{C})(A+C+B\cdot\bar{B})(A\bar{A}+B+\bar{C})$

	A	B	C	Y	(m₀) SOP (minterm) (Σm)		(M₀) POS (Maxterm) (πM)	
0	0	0	0	0	m_0	$\bar{A}\bar{B}\bar{C}$	M_0	$A+B+C$
1	0	0	1	1	m_1	$\bar{A}\bar{B}C$	M_1	$A+B+\bar{C}$
2	0	1	0	1	m_2	$\bar{A}B\bar{C}$	M_2	$A+\bar{B}+C$
3	0	1	1	0	m_3	$\bar{A}BC$	M_3	$A+\bar{B}+\bar{C}$
4	1	0	0	0	m_4	$A\bar{B}\bar{C}$	M_4	$\bar{A}+B+C$
5	1	0	1	1	m_5	$A\bar{B}C$	M_5	$\bar{A}+B+\bar{C}$
6	1	1	0	1	m_6	$AB\bar{C}$	M_6	$\bar{A}+\bar{B}+C$
7	1	1	1	0	m_7	ABC	M_7	$\bar{A}+\bar{B}+C$

$f(Y) = \Sigma m(1,2,5,6)$; $Y = \pi m(0,3,4,7)$

$2^n = x$, n - no. of inputs, x - no. of comb.

Karnaugh - Map (K-Map):- It is the simplification technique used to reduce the Boolean equation. This technique can be used upto 6 variables & disadv. of this tech is that this tech. cannot be automated.

STEP 1:- **Filling**: in SOP

For 2 variables

A \ B	$\bar{B}$ 0	B 1
$\bar{A}$ 0	0	1
A 1	1	1

→ no need to mention

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

In case of 3 variables, the comb. is 00, 01, 11, 10. Acc. to grey code. i.e
00 → 01 → 11 → 10

01 ↓ ✗ 10 — After 01 u can't change to 10 because it ~~step~~ needs replacement of 2 bits where as 01 → 11 can be changed easily as it needs 1 bits only

For 3 variables

A \ BC	00	01	11	10
$\bar{A}$ 0	0	1	3	2
A 1	4	5	7	6

For 4 variables

AB \ CD	00	01	11	10
00	0	1	3	2
01	4	5	7	6
11	12	13	15	14
10	8	9	11	10

STEP 2: GROUPING:

* single (1) ; pair (2) ; quad (4), octa (8), Hexa (16) pair

RULES FOR GROUPING:

1. No diagonal grouping at any stage can take place
2. Only adjacent sides or adj. 1st can be group together
3. A single 1 can be shared by any no. of groups.
4. Top side with last side and left side with right side can be grouped together.

Note: 5. The size ~~of~~ the group should be as
Note: large as possible.

6. The number of groups should be as least as possible.

Make sure: Doesn't go for single grouping until or unless you have single 1 remaining

STEP 3: HAVE UR ANS :-

	AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$	00		I	II	
$\bar{A}B$	01		1	1	1
AB	11	III	1	1	
$A\bar{B}$	10	1			

I :- $\bar{C}D$ CD
$= D$
and $\bar{A}B$ AB
$= B$

$\bar{C}D$
CD
D

$\boxed{\text{Ans (I) } BD}$

II :- $CD \cdot C\bar{D}$
$= C$
and $\bar{A}B$
Ans II $= \bar{A}BC$

III :- $A \cdot \bar{C}\bar{D}$
and $A\bar{B}$
Ans III $= \bar{C}\bar{D} \cdot A\bar{B}$

Hence, final Ans. :- $BD + \bar{A}BC + A\bar{B}\bar{C}\bar{D}$ Ans.

8 - august - 2018

0-3 → 2
0-7 → 3
0-15 → 4
0-31 → 5

1 Ques. $F(A, B, C) = \Sigma m(1, 3, 5, 6, 7)$

	A	B	C	Y
m_0	0	0	0	0
m_1	0	0	1	1
m_2	0	1	0	0
m_3	0	1	1	1
m_4	1	0	0	0
m_5	1	0	1	1
m_6	1	1	0	1
m_7	1	1	1	1

A \ BC	$\bar{B}\bar{C}$ 00	$\bar{B}C$ 01	BC 11	$B\bar{C}$ 10
$\bar{A}$ 0	0 (0)	1 (1)	1 (3)	0 (2)
A 1	0 (4)	1 (5)	1 (7)	1 (6)

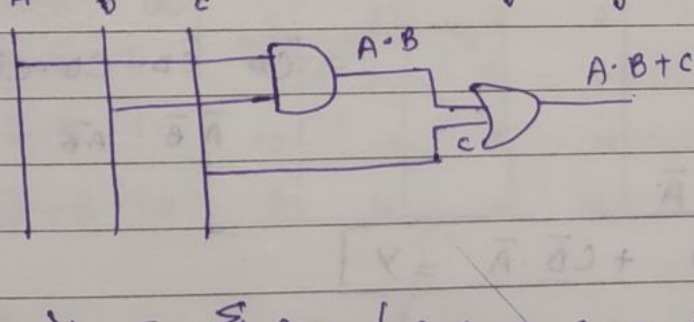
G1, G2

000 → 0
001

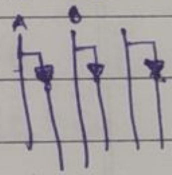
$G1 \rightarrow (\bar{B}C + BC)$
$G1 \rightarrow C$
$G2 \rightarrow B \cdot A$
Total $= G1 + G2 = \boxed{C + AB = Y}$

8421
011
101

Realisation into logic gates :-



To show NOT



2 Ques. $f\ Y = \Sigma m(0, 1, 2, 3, 6, 8, 9, 10, 11, 12, 13)$

2, 3, 4, 5, 6

$2^3=8$ $2^2=4$ $2^1=2$ $2^0=1$

A	B	C	D	Y
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	1	0	0	
1	0	0	0	
0	0	1	1	
0	1	1	0	
1	1	0	0	
1	0	1	0	
1	0	0	1	
0	1	0	1	

Y	A	B	C	D
1	0	0	0	0
1	0	0	0	1
1	0	0	1	0
1	0	0	1	1
0	0	1	0	0
0	0	1	0	1
1	0	1	1	0
0	0	1	1	1
1	1	0	0	0
1	1	0	0	1
1	1	0	1	0
1	1	0	1	1
1	1	1	0	0
1	1	1	0	1
0	1	1	1	0
0	1	1	1	1

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	1 (0)	1 (1)	1 (3)	1 (2)
$\bar{A}B$ 01	0 (4)	0 (5)	0 (7)	1 (6)
AB 11	1 (12)	1 (13)	0 (15)	0 (14)
$A\bar{B}$ 10	1 (8)	1 (9)	1 (11)	1 (10)

G1, G3, G, G2

$\bar{C}\bar{D}$ $\bar{C}D$ $CD \cdot C\bar{D}$ =
$\bar{A}\bar{B}$ $A\bar{B}$

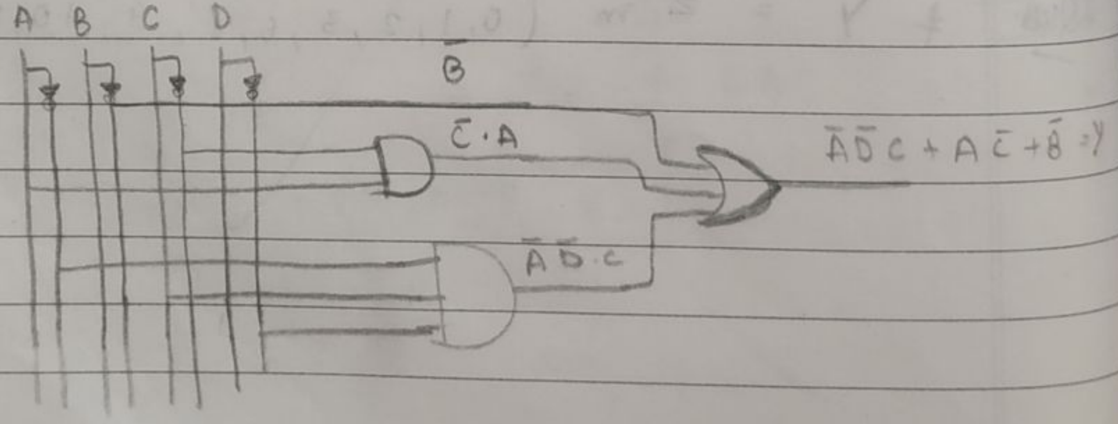
$G_1 \rightarrow \bar{B}$

$G_2 \rightarrow \bar{C} \cdot A$

$G_3 \rightarrow C\bar{D} \cdot \bar{A}$

Total = $\boxed{\bar{B} + \bar{C} \cdot A + C\bar{D} \cdot \bar{A} = Y}$

logic gate:-



Page No. 11

Date:

14/Aug/2018

$Y = \Sigma m\,(0, 1, 4, 6, 8, 11, 12, 13, 15)$

SOP: SOP.

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	1 ₀	1 ₁	₃	₂
$\bar{A}B$ 01	1 ₄	₅	₇	1 ₆
AB 11	1 ₁₂	1 ₁₃	1 ₁₅	₁₄
$A\bar{B}$ 10	1 ₈	₉	1 ₁₁	₁₀

G1, G2, G3, G4, G5

$G_1 \rightarrow \bar{A}\bar{B}\ \ \bar{A}B\ \ AB\ \ A\bar{B}\ ,\ \bar{C}\bar{D} \Rightarrow \bar{C}\bar{D}$

$G_2 \rightarrow \bar{C}\,\bar{A}\,\bar{B}$

$G_3 \rightarrow D\,AB$

$G_4 \rightarrow ACD$

$G_5 \rightarrow \bar{D}\,\bar{A}\,B$

$Y = G_1 + G_2 + G_3 + G_4 + G_5$

$Y = \bar{C}\bar{D} + \bar{A}\bar{B}\bar{C} + ABD + ACD + \bar{A}B\bar{D}$ Ans.

POS :- $\pi M\,(2, 5, 7, 10, 14)$.

eg.

A+B \ C+D	C+D 00	C+$\bar{D}$ 01	$\bar{C}$+$\bar{D}$ 11	$\bar{C}$+D 10
A+B 00	₀	₁	₃	₂
A+$\bar{B}$ 01	₄	0 ₅	0 ₇	₆
$\bar{A}$+$\bar{B}$ 11	₁₂	0 ₁₃	0 ₁₅	₁₄
$\bar{A}$+B 10	₈	₉	₁₁	₁₀

$C + \bar{D} + \bar{C} + \bar{D} = \bar{D}$

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

$G_1 = \bar{B} + \bar{D}$

(d) Don't Care Conditions :

$Y = \Sigma m\,(0, 1, 4, 6, 8, 11, 12, 13, 15) \pm d\,(2, 3, 7)$

neither 0 or 1

take it 0 or 1 acc. to the we need.

eg.

AB \ CD	00	01	11	10
00	1 ₀	1 ₁	₃	X ₂
01	1 ₄	X ₅	X ₇	1 ₆
11	1 ₁₂	1 ₁₃	1 ₁₅	₁₄
10	1 ₈	₉	1 ₁₁	₁₀

Special Case of K-Map :

AB \ CD	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	1			1
$\bar{A}B$				
AB				
$A\bar{B}$	1			1

$\bar{C}\bar{D} \cdot C\bar{D} = \bar{D}$

$\bar{A}\bar{B} \cdot A\bar{B} = \bar{B}$

$Y = \bar{B}\bar{D}$

Ques. $Y = \Sigma m\,(0, 2, 5, 7, 8, 10, 13, 15)$

AB \ CD	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	1 (0)	(1)	(3)	1 (2)
$\bar{A}B$	(4)	1 (5)	1 (7)	(6)
AB	(12)	1 (13)	1 (15)	(14)
$A\bar{B}$	1 (8)	(9)	(11)	1 (10)

G_1

$G_1 = \bar{C}D, CD = D$

$\bar{A}B, AB = B$

$G_1 = BD$

$G_2 = \bar{B}\bar{D}$

$Y = BD + \bar{B}\bar{D}$

5 - Variable K- Map :-

	A	B	C	D	E	Y
m_0	0	0	0	0	0	1
m_1	0	0	0	0	1	0
m_2	0	0	0	1	0	1
m_3	0	0	0	1	1	
m_4	0	0	1	0	0	1
m_5	0	0	1	0	1	
m_6	0	0	1	1	0	1
m_7	0	0	1	1	1	
m_8	0	1	0	0	0	
m_9	0	1	0	0	1	1
m_{10}	0	1	0	1	0	
m_{11}	0	1	0	1	1	1
m_{12}	0	1	1	0	0	
m_{13}	0	1	1	0	1	1

A = 0

BC \ DE				
	0	1	3	2
	4	5	7	6
	12	13	15	14
	8	9	11	10

A = 1

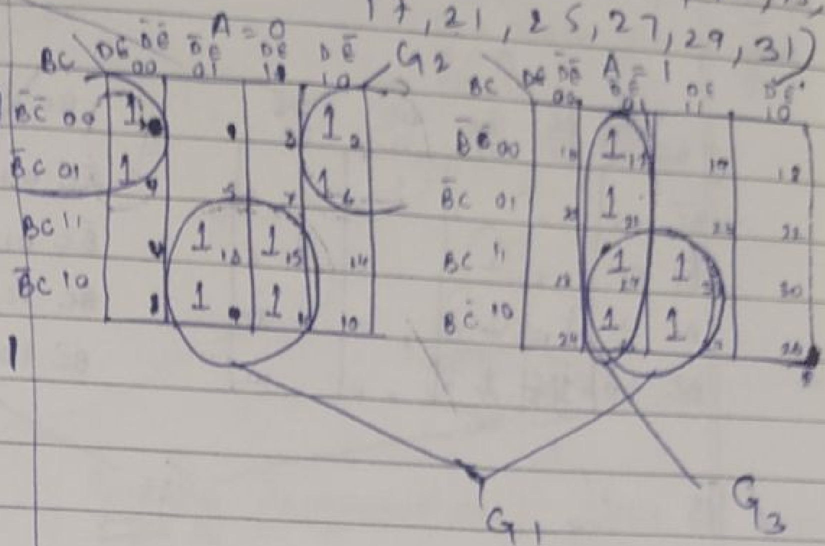
BC \ DE				
	16	17	19	18
	20	21	23	22
	28	29	31	30
	24	25	27	26

Only adjacent form grouping

13 ⟺ 29 gp.

	A	B	C	D	E	Y
m_{14}	0	1	1	1	0	1
m_{15}	0	1	1	1	1	1
m_{16}	1	0	0	0	0	
m_{17}	1	0	0	0	1	1
m_{18}	1	0	0	1	0	
m_{19}	1	0	0	1	1	
m_{20}	1	0	1	0	0	
m_{21}	1	0	1	0	1	1
m_{22}	1	0	1	1	0	
m_{23}	1	0	1	1	1	
m_{24}	1	1	0	0	0	
m_{25}	1	1	0	0	1	1
m_{26}	1	1	0	1	0	
m_{27}	1	1	0	1	1	1
m_{28}	1	1	1	0	0	
m_{29}	1	1	1	0	1	1
m_{30}	1	1	1	1	0	
m_{31}	1	1	1	1	1	1

Ques. $Y = \Sigma m(0, 2, 4, 6, 9, 11, 13, 15, 17, 21, 25, 27, 29, 31)$

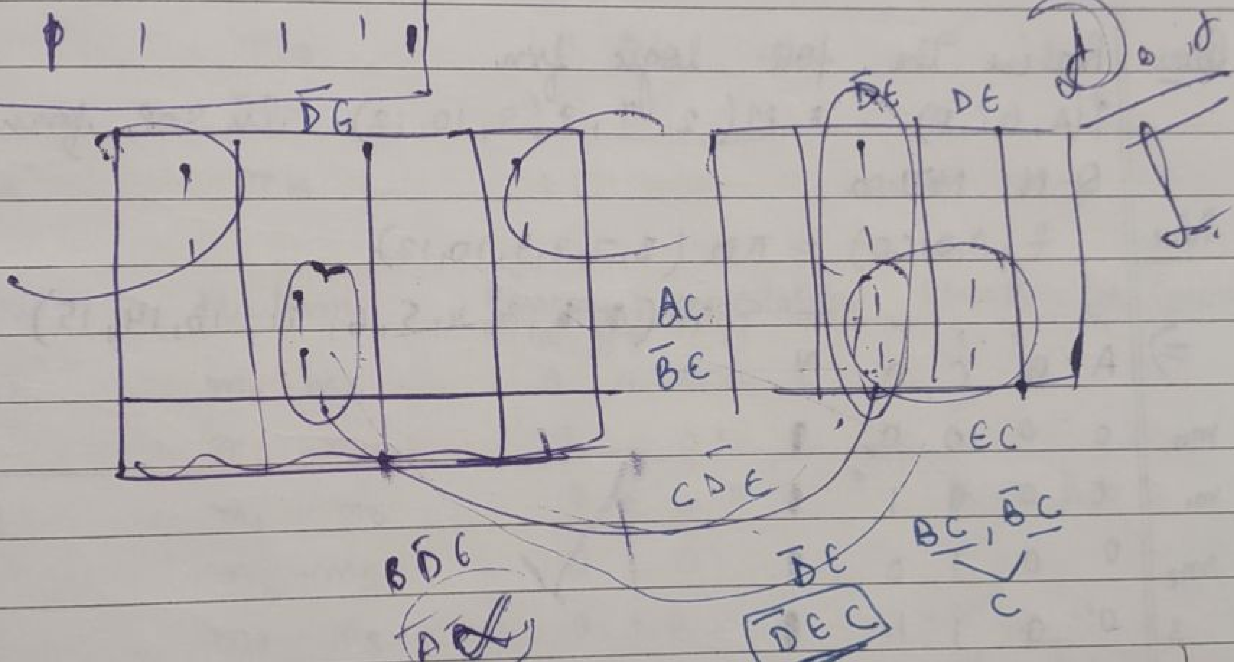


$G_1 = BE$

$G_2 = \bar{A}\bar{B}\bar{E}$

$G_3 = A\bar{D}E$

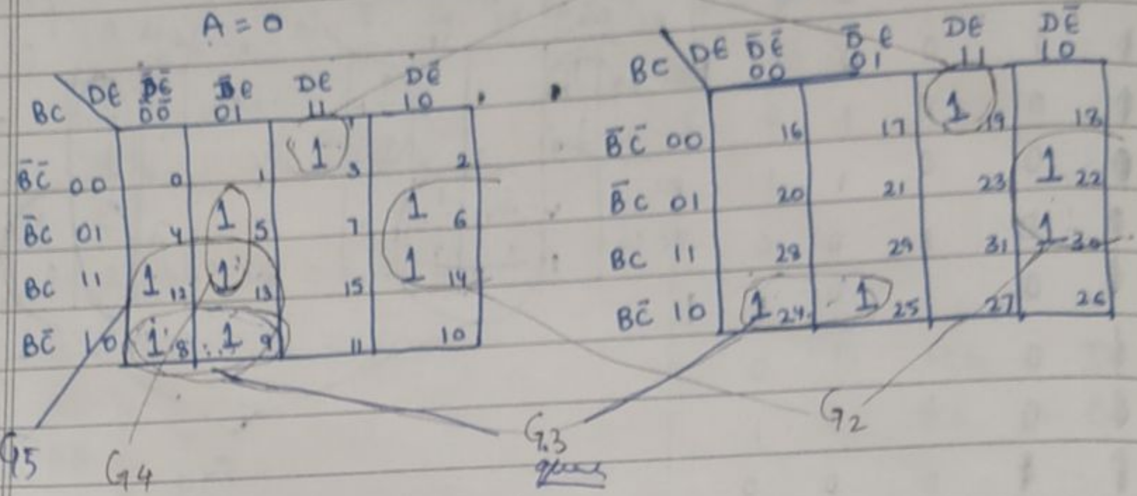
$Y = BE + \bar{A}\bar{B}\bar{E} + A\bar{D}E$



$Y = \Sigma m(3, 5, 6, 8, 9, 12, 13, 14, 19, 22, 24, 25, 30)$

Ques. $Y = \Sigma m(3, 5, 6, 8, 9, 12, 13, 14, 19, 22, 24, 25, 30)$

Ans



$G_1 = DE \cdot \bar{B}\bar{C}$; $G_2 = CD\bar{E}$; $G_4 = \bar{A}C\bar{D}E$

$G_3 = \bar{D}B\bar{C}$, $G_5 = \bar{A}B\bar{D}$

21-August-2018

Imp. Quine - Mc - Cluskey (QM) Method

Ques. Realise the foll. logic fxn
$f(A,B,C,D) = \pi M(2, 7, 8, 9, 10, 12)$ in SOP form using Q-M Method

Solve. $f(A,B,C,D) = \pi M(2, 7, 8, 9, 10, 12)$

$= \Sigma m(0, 1, 3, 4, 5, 6, 11, 13, 14, 15)$

$\Rightarrow$

	A	B	C	D	Y
m_0	0	0	0	0	1
m_1	0	0	0	1	1
m_2	0	0	1	0	0
3	0	0	1	1	1
4	0	1	0	0	1
5	0	1	0	1	1
6	0	1	1	0	1
7	0	1	1	1	0

8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	1	0	0
11	1	0	1	1	1
12	1	1	0	0	0
13	1	1	0	1	1
14	1	1	1	0	1
15	1	1	1	1	1

Change of 1 bit only can make grouping possible.

⇒

Group	Minterm	Binary Representation (A B C D)	Check for Group.
0	m_0	0 0 0 0	✓
1	m_1	0 0 0 1	✓
	m_4	0 1 0 0	✓
2	m_3	0 0 1 1	✓
	m_5	0 1 0 1	✓
	m_6	0 1 1 0	✓
3	m_{11}	1 0 1 1	✓
	m_{13}	1 1 0 1	✓
	m_{14}	1 1 1 0	✓
~~4~~	m_{15}	1 1 1 1	✓

⇒

Group	Minterm	Binary Representation (A B C D)	Check for Group.
0	$m_0 - m_1$	0 0 0 _	✓
	$m_0 - m_4$	0 _ 0 1	✓
1	$m_1 - m_3$	0 0 _ 1	* $\bar{A}\bar{B}D$
	$m_1 - m_5$	0 _ 0 1	✓
	$m_4 - m_5$	0 1 0 _	✓
	$m_4 - m_6$	0 1 _ 0	* $\bar{A}B\bar{D}$
2	$m_3 - m_{11}$	_ 0 1 1	* $\bar{B}CD$
	$m_5 - m_{13}$	_ 1 0 1	* $B\bar{C}D$
	$m_6 - m_{14}$	_ 1 1 0	* $BC\bar{D}$

Date:

3	$m_{11}-m_{15}$	1 _ 1 1	* ACD
	$m_{13}-m_{15}$	1 1 _ 1	* ABD
	$m_{14}-m_{15}$	1 1 1 _	* ABC

⇒

Group	Minterm	Binary Representation A B C D	check for Gp.
0	$m_0-m_1-m_4-m_5$	0 _ 0 _	$\bar{A}\bar{C}$
	$m_0-m_4-m_1-m_5$	0 _ 0 _	

Prime implicants - all those no. which were not gp. in 2nd last table & last table in which no further grouping is possible - all these values are known together give prime implicants

$$\Rightarrow Y = \bar{A}\bar{B}D + \bar{A}\bar{B}\bar{D} + \bar{B}CD + B\bar{C}D + BC\bar{D} + ACD + ABD + ABC + \bar{A}\bar{C}$$

⇒ Prime Implicant Table - To have Essential Prime implicant we need to draw it / make it

PI	Minterm No. Decimal equivalence	0	1	3	4	5	6	11	13	14	15
$\bar{A}\bar{B}D$	1, 3		X	X							
$\bar{A}\bar{B}\bar{D}$	4, 6				X		X				
$\bar{B}CD$	3, 11			X				X			
$B\bar{C}D$	5, 13					X			X		
$BC\bar{D}$	6, 14						X			X	
ACD	11, 15							X			X
ABD	13, 15								X		X
ABC	14, 15									X	X
$\bar{A}\bar{C}$	0, 1, 4, 5	X	X		X	X					

EPI

$y = \bar{B}CD + BC\bar{D} + ABD + \bar{A}\bar{C}$

representation

Ques: Using K map solve.
$f(A,B,C,D) = \Sigma m(0,1,3,4,5,6,11,13,14,15)$

Solve:-

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	1 ₀	1 ₁	1 ₃	₂
$\bar{A}B$ 01	1 ₄	1 ₅	₇	1 ₆
AB 11	₁₂	1 ₁₃	1 ₁₅	1 ₁₄
$A\bar{B}$ 10	₈	₉	1 ₁₁	₁₀

G_1, G_2, G_3, G_4

$G_1 = \bar{A}\bar{C}$, $G_2 = \bar{B}CD$, $G_3 = C\bar{D}B$,
$G_4 = ABD$, $G_5 =$
$y = \bar{A}\bar{C} + \bar{B}CD + BC\bar{D} + ABD$

Why Q-M Map? Coding is possible in case of Q-M Map whereas it is not so possible in case of K-Map.

Ques. $= \Sigma m(0,1,2,3,5,7,8,9,\ 11,14)$ Solve using Q-M-Method

Solve. $f(A,B,C,D) = \Sigma m(0,1,2,3,5,7,8,9,11,14)$

	A	B	C	D	Y
m_0	0	0	0	0	1
m_1	0	0	0	1	1
m_2	0	0	1	0	1
m_3	0	0	1	1	1
m_4	0	1	0	0	0
m_5	0	1	0	1	1
m_6	0	1	1	0	0
m_7	0	1	1	1	1
m_8	1	0	0	0	1

	A	B	C	D	Y
m_9	1	0	0	1	1
m_{10}	1	0	1	0	0
m_{11}	1	0	1	1	1
m_{12}	1	1	0	0	0
m_{13}	1	1	0	1	0
m_{14}	1	1	1	0	1
m_{15}	1	1	1	1	0

⇒

Group	Minterm	Binary Rep. A B C D	Checks for Gp.
0	m_0	0 0 0 0	✓
1	m_1	0 0 0 1	✓
	m_2	0 0 1 0	✓
	m_8	1 0 0 0	✓
2	m_3	0 0 1 1	✓
	m_5	0 1 0 1	✓
	m_9	1 0 0 1	✓
3	m_7	0 1 1 1	✓
	m_{11}	1 0 1 1	✓
	m_{14}	1 1 1 0	* $AB C\bar{D}$

⇒

Group	Minterms	(B.R.)(A B C D)	Check for Gp.
0	$m_0 - m_1$	0 0 0 –	✓ ✓
	$m_0 - m_2$	0 0 – 0	✓ ✓
	$m_0 - m_8$	– 0 0 0	✓ ✓
1	$m_1 - m_3$	0 0 – 1	✓ ✓
	$m_1 - m_5$	0 – 0 1	✓ ✓
	$m_1 - m_9$	– 0 0 1	✓ ✓
2	$m_2 - m_3$	0 0 1 –	* ✓
	$m_8 - m_9$	1 0 0 –	* ✓
2	$m_3 - m_7$	0 – 1 1	✓ ✓
	$m_3 - m_{11}$	– 0 1 1	✓ ✓
	$m_5 - m_7$	0 1 – 1	✓ ✓
	$m_9 - m_{11}$	1 0 – 1	✓ ✓

⇒

Group.	Minterms	A B C D	Check for Gp.
0	$m_0 - m_2 - m_1 - m_3$	0 0 – –	
	$m_0 - m_8 - m_1 - m_9$	– 0 0 –	
1	$m_1 - m_3 - m_5 - m_7$	0 – – 1	
	$m_1 - m_3 - m_9 - m_{11}$	– 0 – 1	

0 - 01

~~$m_1 - m_5 - m_3 - m_7$~~	~~0 _ _ 1~~
~~$m_1 - m_9 - m_3 - m_{11}$~~	~~_ 0 _ 1~~
~~$m_2 - m_3$~~	

⇒

Group	Minterm	ABCD	Check for Gp.
0	$m_0 - m_1 - m_2 - m_3$	0 0 _ _	
	$m_0 - m_1 - m_8 - m_9$	_ 0 0 _	$\bar{A}\bar{B}$
	$m_0 - m_2 - m_1 - m_3$	0 0 _ _	
	$m_0 - m_8 - m_1 - m_9$	_ 0 0 _	$\bar{B}\bar{C}$
1	$m_1 - m_3 - m_5 - m_7$	0 _ _ 1	
	$m_1 - m_3 - m_9 - m_{11}$	_ 0 _ 1	
	~~$m_1 - m_3 - m_5 - m_7$~~	~~0 _ _ 1~~	$\bar{A}D$
	$m_1 - m_9 - m_3 - m_{11}$	_ 0 _ 1	$\bar{B}D$
	$m_1 - m_5 - m_3 - m_7$	0 _ _ 1	

$$Y = \bar{A}\bar{B} + \bar{B}\bar{C} + \bar{A}D + \bar{B}D + ABC\bar{D}$$

Do it again

⇒ Prime implicant Table

P.I	Decimal eq. Minterm No	0	1	2	3	5	7	8	9	11	14
		Given Minterms									
$\bar{A}\bar{B}$	0, 1, 2, 3	x	x	x	x						
$\bar{B}\bar{C}$	0, 1, 8, 9	x	x					x	x		
$\bar{A}D$	1, 3, 5, 7		x		x	x	x				
$\bar{B}D$	1, 3, 9, 11		x		x				x	x	
$ABC\bar{D}$	14										*

Using K-Map solve $\Sigma m(0,1,2,3,5,7,8,9,11,14)$

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	1 (0)	1 (1)	1 (3)	1 (2)
$\bar{A}B$ 01	(4)	1 (5)	1 (7)	(6)
AB 11	(12)	(13)	(15)	1 (14)
$A\bar{B}$ 10	1 (8)	1 (9)	1 (11)	(10)

G_1, G_2, G_3, G_4, G_5

$G_1 \rightarrow \bar{A}D$; $G_2 \rightarrow \bar{A}\bar{B}$; $G_3 \rightarrow A\bar{B}D$; $G_4 \rightarrow \bar{B}\bar{C}\bar{D}$

$G_5 \rightarrow ABC\bar{D}$

$Y = \bar{A}D + \bar{A}\bar{B} + A\bar{B}D + \bar{B}\bar{C}\bar{D} + ABC\bar{D}$

1001 → 1101

22-aug-2018

Code Conversion

I. Gray code

(Not to cons. about carry)

	A	B	C	D (Binary)	E	F	G	H (Gray)
0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	1
2	0	0	1	0	0	0	1	1
3	0	0	1	1	0	0	1	0
4	0	1	0	0	0	1	1	0
5	0	1	0	1	0	1	1	1
6	0	1	1	0	0	1	0	1
7	0	1	1	1	0	1	0	0
8	1	0	0	0	1	1	0	0
9	1	0	0	1	1	1	0	1
10	1	0	1	0	1	1	1	1
11	1	0	1	1	1	1	1	0
12	1	1	0	0	1	0	1	0
13	1	1	0	1	1	0	1	1

00	00
01	01
11	10
10	11

6 0110 → 0101

1110 → 1011

Binary → Gray: add previous bit to the next bit

1011 → 1101

Binary to Gray

1001
↓
1101

14	1	1	1	0	1	0	0	1
15	1	1	1	1	1	0	0	0

Gray to Binary

For E

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	0_0	0_1	0_3	0_2
$\bar{A}B$ 01	0_4	0_5	0_7	0_6
AB 11	1_{12}	1_{13}	1_{15}	1_{14}
$A\bar{B}$ 10	1_8	1_9	1_{11}	1_{10}

G_1

For F

AB \ CD	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$ 00	0_0	0_1	0_3	0_2
$\bar{A}B$ 01	1_4	1_5	1_7	1_6
AB 11	0_{12}	0_{13}	0_{15}	0_{14}
$A\bar{B}$ 10	1_8	1_9	1_{11}	1_{10}

G_1, G_2

For G

AB \ CD	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0_0	0_1	1_3	1_2
$\bar{A}B$	1_4	1_5	0_7	0_6
AB	1_{12}	1_{13}	0_{15}	0_{14}
$A\bar{B}$	0_8	0_9	1_{10}	1_{11}

G_1, G_2

For H

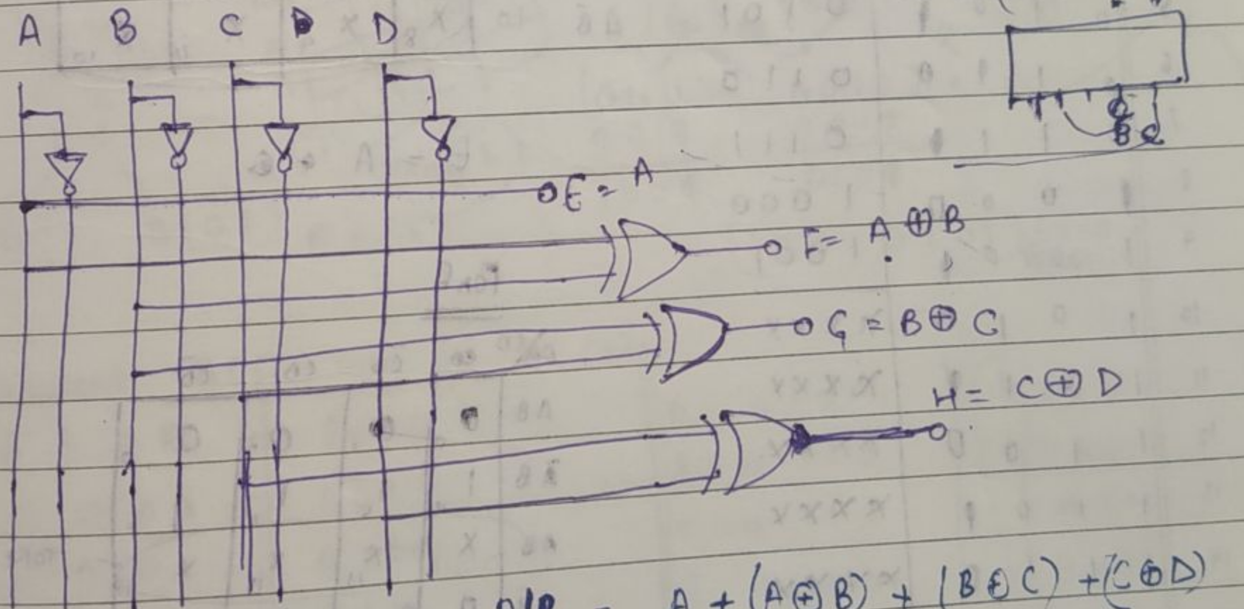
AB \ CD	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0_0	1_1	0_3	1_2
$\bar{A}B$	0_4	1_5	0_7	1_6
AB	0_{12}	1_{13}	0_{15}	1_{14}
$A\bar{B}$	0_8	1_9	0_{11}	1_{10}

G_1, G_2

For $E = A$; For $F = \bar{A}B + A\bar{B} = A \oplus B$ EXOR

For $G = B\bar{C} + \bar{B}C$; For $H = \bar{C}D + C\bar{D}$

$= B \oplus C$ EXOR ; $= C \oplus D$ EXOR



$O/P = A + (A \oplus B) + (B \oplus C) + (C \oplus D)$

24-august-2018

II **Binary Coded Decimal (BCD) Code / 8421 Code**

BCD

Decimal to Bin

0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

(a) $(23)_{10} \longrightarrow (10111)_2$

$\downarrow$

(BCD) — 23 → 2, 3

2 → 0010, 3 → 0011

$(23)_{10} = (0010\,0011)_{BCD}$

(b) 146 → 1, 4, 6

1 → 0001, 4 → 0100, 6 → 0110

$(146)_{10} = (0001\,0100\,0110)_{BCD}$

Convert decimal no → 4 bit BCD code

Ques. Design a four-bit binary to 4-bit BCD code

	A	B	C	D	BCD E F G H
0	0	0	0	0	0 0 0 0
1	0	0	0	1	0 0 0 1
2	0	0	1	0	0 0 1 0
3	0	0	1	1	0 0 1 1
4	0	1	0	0	0 1 0 0
5	0	1	0	1	0 1 0 1
6	0	1	1	0	0 1 1 0
7	0	1	1	1	0 1 1 1
8	1	0	0	0	1 0 0 0
9	1	0	0	1	1 0 0 1
10	1	0	1	0	X X X X
11	1	0	1	1	X X X X
12	1	1	0	0	X X X X
13	1	1	0	1	X X X X
14	1	1	1	0	X X X X
15	1	1	1	1	X X X X

K-Map.

For E

AB \ CD	$\bar{C}\bar{D}$ 00	$\bar{C}D$ 01	CD 11	$C\bar{D}$ 10
$\bar{A}\bar{B}$ 00	0 (0)	0 (1)	0 (3)	0 (2)
01	0 (4)	0 (5)	0 (7)	0 (6)
AB 11	X (12)	X (13)	X (15)	X (14)
$A\bar{B}$ 10	X (8)	X (9)	X (11)	X (10)

Take as 1

$E = A + C$

For F

AB \ CD	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	0 (0)	0 (1)	0 (3)	0 (2)
$\bar{A}B$	1 (4)	1 (5)	1 (7)	1 (6)
AB	X (12)	X (13)	X (15)	X (14)
$A\bar{B}$	0 (8)	0 (9)	X (11)	X (10)

Take as 1

$F = B$

For G

AB \ CD	$\bar{C}\bar{D}$		CD	$C\bar{D}$
$\bar{A}\bar{B}$	0 (0)	0 (1)	1 (3)	1 (2)
$\bar{A}B$	0 (4)	0 (5)	1 (7)	1 (6)
AB	X (12)	X (13)	X (15)	X (14)
$A\bar{B}$	0 (8)	0 (9)	X (11)	X (10)

take as 1

$G = C$

For H

AB \ CD				
	0 (0)	1 (1)	1 (3)	0 (2)
	0 (4)	1 (5)	1 (7)	0 (6)
	X (12)	X (13)	X (15)	X (14)
	0 (8)	1 (9)	X (11)	X (10)

Take as 1

$H = D$

III $\boxed{\text{Ex-3 code}}$ derived from BCD code

(No.) → BCD $\xrightarrow{+}$ 3 → req. (Ex-3 code)
to every bit

eg. $(8)_{10}$ → 1000
+ 0011
(1011) Ex-3

$(42)_{10}$
0100 0010
+0010 + 0010
0110 0101 (sic)

eg. $(42)_{10}$ = (0111 0101) Ex-3
+3 +3
7 5
(0111 0101) Ex-3

$P\bar{Q}S$

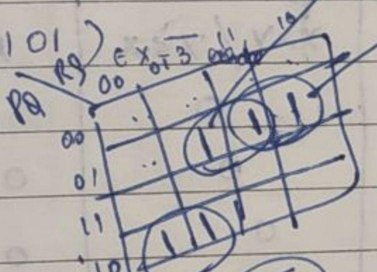
$\bar{P}QR$

$\bar{P}Q(\bar{R}+\bar{S}) + \bar{P}QR + PQ\bar{R}$ (BCD)

$P\bar{Q}(R+\bar{S}) + P\bar{Q}\bar{S} + P\bar{Q}S$

$\cancel{PQR}$

$Q \oplus R + \bar{P}\bar{Q}S$



$P\bar{Q}\bar{R}$

eg. $(62)_{10}$
+3 +3
9 5
(1001 0101) Ex-3

$(62)_{10}$
(0110 0010) BCD
0011 0011
1001 0101
(10010101) Ex-3.

R ⊕ Q

⇒ Convert BCD → Ex-3 code

	A B C D	+3. Ex-3
0	0 0 0 0	0011
1	0 0 0 1	0100
2	0 0 1 0	0101
3	0 0 1 1	0110
4	0 1 0 0	0111
6	0 1 0 1	1000
6	0 1 1 0	1001
7	0 1 1 1	1011
8	1 0 0 0	1011
9	1 0 0 1	1100
10		x x x x
:		
15		x x x x

74
99

for F

for G

AB \ CD				
	0	0	0	0
	0	1	1	1
	x	x	x	x
	1	1	x	x

G3

G1

G2

E = A + BC + BD

for G

for H

input → var. - K-map
know
output, → how many K-map.

<u>28/Aug/18.</u> Magnitude comparitive

Two - bit magnitude comparitive

	X (AB)	Y (CD)	x<y / y>x	x>y / y<x	x=y
m_0	00	00	0	0	1
m_1	00	01	1	0	0
m_2	00	10	1	0	0
m_3	00	11	1	0	0
4	01	00	0	1	0
5	01	01	0	0	1
6	01	10	1	0	0
7	01	11	1	0	0
8	10	00	0	1	0
9	10	01	0	1	0
10	10	10	0	0	1
11	10	11	1	0	0

			Y > X	X > Y	X = Y
	11	00	0	1	0
12	11	01	0	1	0
13	11	10	0	1	0
14	11	11	0	0	1
15					

For X < Y

AB \ CD	$\bar C\bar D$ 00	$\bar C D$ 01	CD 11	$C\bar D$ 10
$\bar A\bar B$ 00	0	1_{1}	1_{3}	1_{2}
$\bar A B$ 01	4	5	1_{7}	1_{6}
AB 11	12	13	15	14
$A\bar B$ 10	8	9	1_{11}	10

G2 G3 G1 G3

$$Y = C\cdot\bar A + \bar A\bar B D + \bar B C D$$

3 And, 1 OR gates

For X > Y

AB \ CD	$\bar C\bar D$	$\bar C D$	CD	$C\bar D$
$\bar A\bar B$	0	1	3	2
$\bar A B$	1_{4}	5	7	6
AB	1_{12}	1_{13}	15	1_{14}
$A\bar B$	1_{8}	1_{9}	11	10

G2 G3 G1

$$Y = A\bar C + B\bar C\bar D + AB\bar D$$

3 AND, 1 OR Gates

diagonal direction

↙ Ex-OR

↘ Ex-NOR

$\oplus$ $\bar A B + A\bar B$ = X-OR

$\odot$ $AB + \bar A\bar B$ = X-NOR

For X = Y

AB \ CD	$\bar C\bar D$	$\bar C D$	CD	$C\bar D$
$\bar A\bar B$	1_{0}	1	3	2
$\bar A B$	4	1_{5}	7	6
AB	12	13	1_{15}	14
$A\bar B$	8	9	11	1_{10}

$$Y = \bar A\bar B\bar C\bar D + \bar A B\bar C D + ABCD + A\bar B C\bar D$$

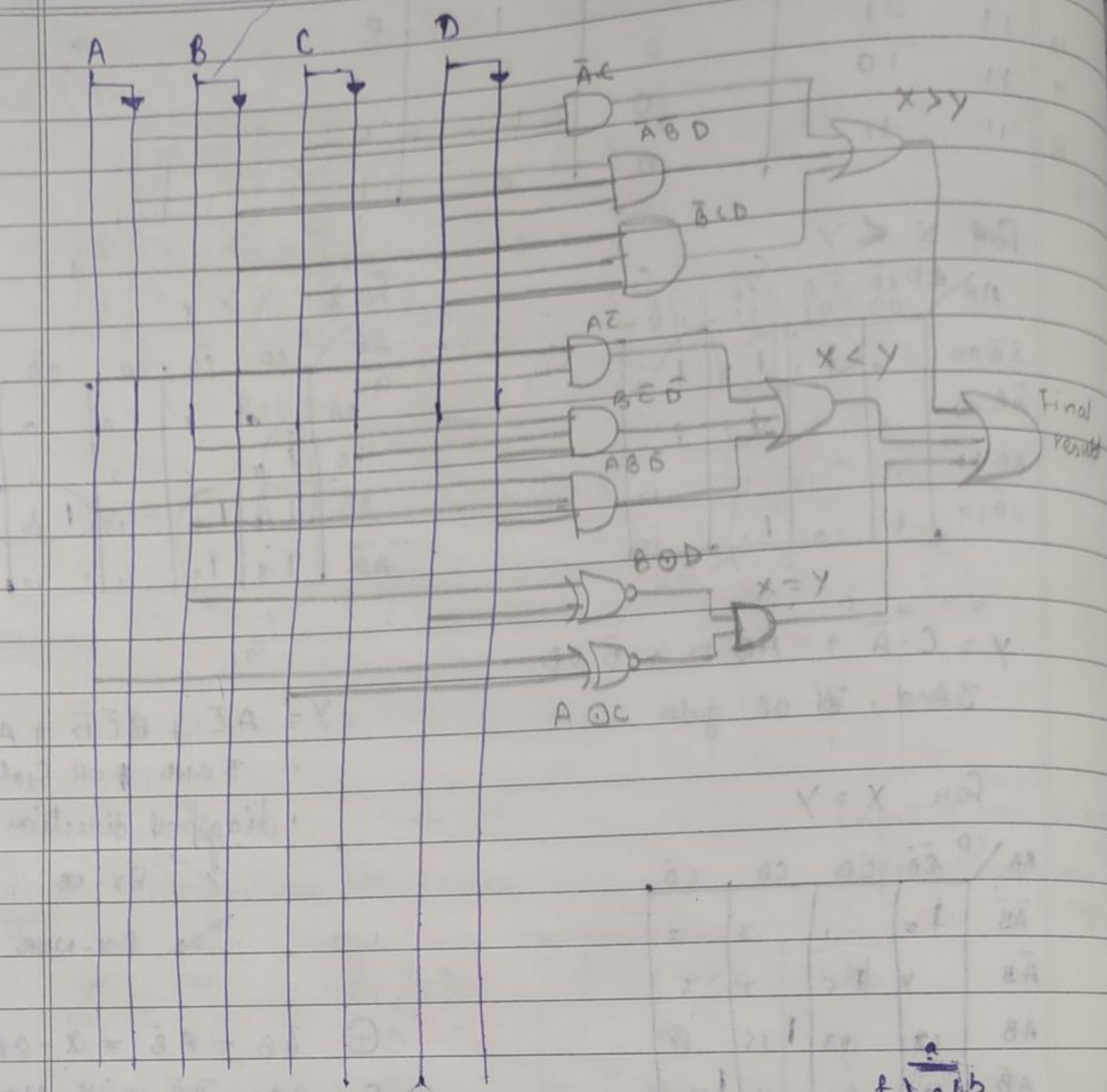
$$= \bar A\bar C(\bar B\bar D + BD) + AC(BD + \bar B\bar D)$$

$$= (\bar B\bar D + BD)(AC + \bar A\bar C)$$

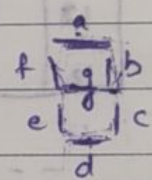
$$= (B \odot D)(A \odot C)$$

→ 2 Ex-NOR, 1 And gate

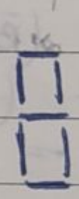
Final Result :- $(\bar A C + \bar A\bar B D + \bar B C D) + (A\bar C + B\bar C\bar D + AB\bar D) + [(B\odot D)(A\odot C)]$

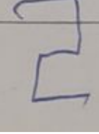


BCD to Seven segment display encoder



	A	B	C	D	a	b	c	d	e	f	g
0	0	0	0	0	1	1	1	1	1	1	0
1	0	0	0	1	0	1	1	0	0	0	0
2	0	0	1	0	1	1	0	1	1	0	1
3	0	0	1	1	1	1	1	1	0	0	1
4	0	1	0	0	0	1	1	0	0	1	1
5	0	1	0	1	1	0	1	1	0	1	1
6	0	1	1	0	1	0	1	1	1	1	1
7	0	1	1	1	1	1	1	0	0	0	0
8	1	0	0	0	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	1	0	1	1

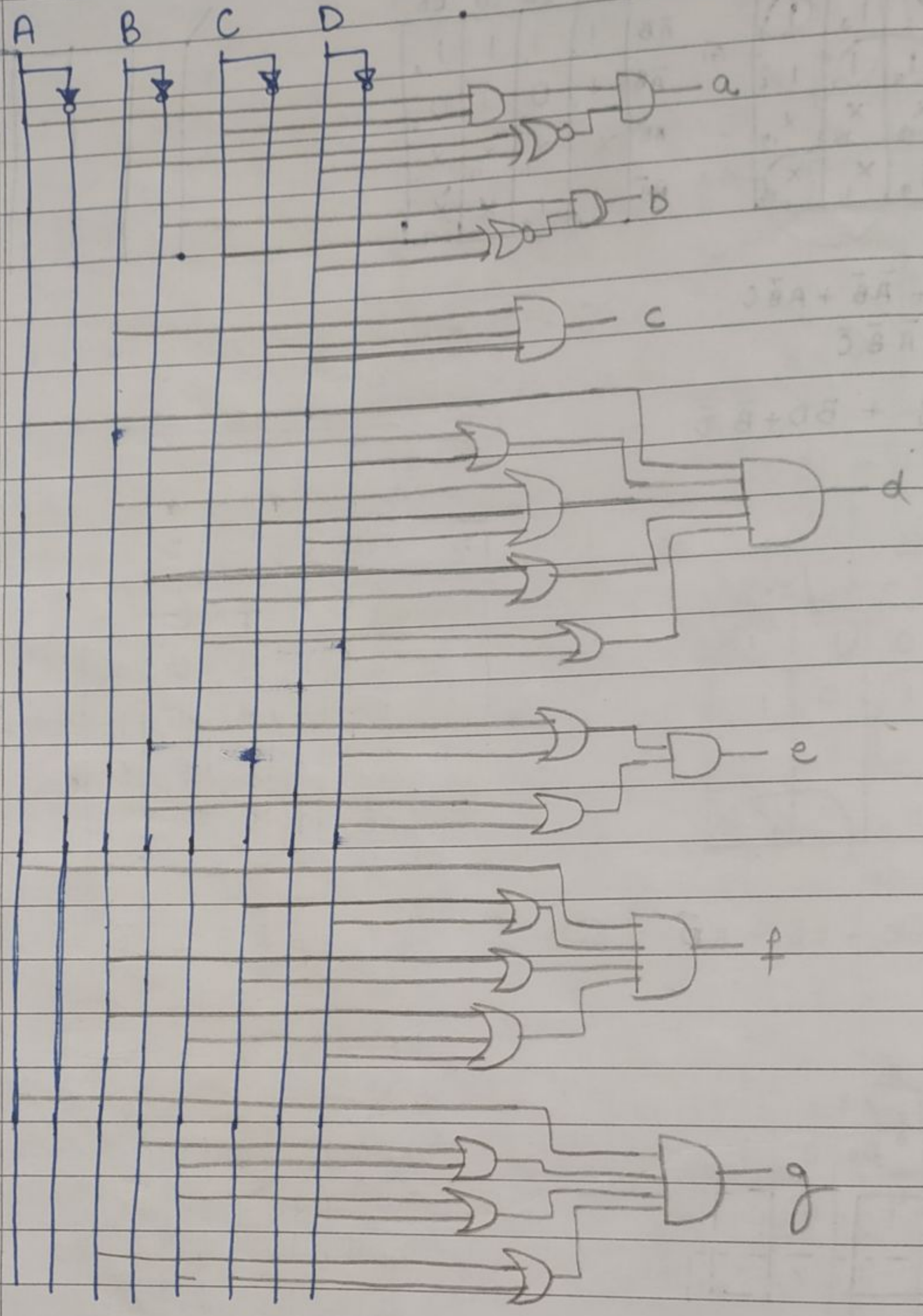


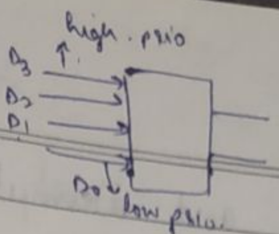


Page No. ______
Date: 28.

OR gate and

AND



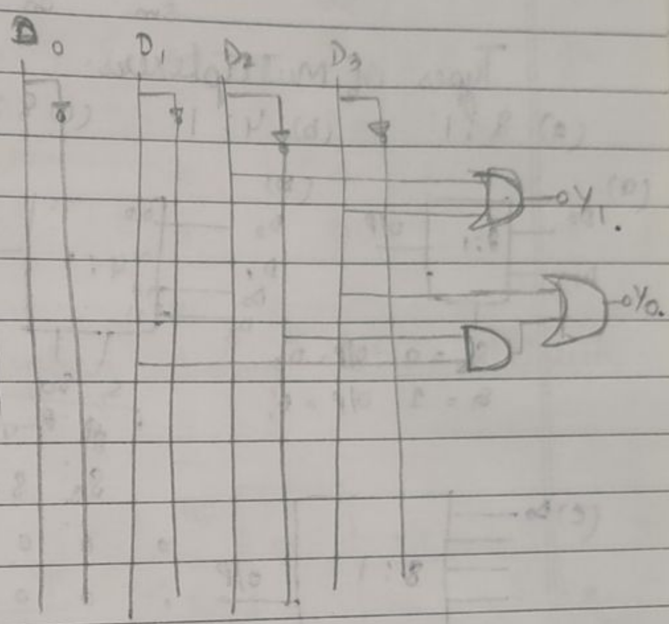


Priority Encoder :-

A	B	C	D		MSB	LSB
D_3	D_2	D_1	D_0		Y_1	Y_0
0	0	0	0	0	X	X
1	0	0	0	1	0	0
2	0	0	1	0	0	1
3	0	0	1	1	0	1
4	0	1	0	0	1	0
5	0	1	0	1	1	0
6	0	1	1	0	1	0
7	0	1	1	1	1	0
8	1	0	0	0	1	1
9	1	0	0	1	1	1
10	1	0	1	0	1	1
11	1	0	1	1	1	1
12	1	1	0	0	1	1
13	1	1	0	1	1	1
14	1	1	1	0	1	1
15	1	1	1	1	1	1

$\Longleftrightarrow$

D_3	D_2	D_1	D_0	Y_1	Y_0
0	0	0	0	x	x
0	0	0	1	0	0
0	0	1	x	0	1
0	1	x	x	1	0
1	x	x	x	1	1



for Y_1

AB \ CD	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	X 0	0 1	0 3	0 2
$\bar{A}B$	1 4	1 5	1 7	1 6
AB	1 12	1 13	1 15	1 14
$A\bar{B}$	1 8	1 9	1 11	1 10

G_2, G_1

$G_1 = A$, $G_2 = B$

$Y_1 = D_3 + D_2$

for Y_0

AB \ CD	$\bar{C}\bar{D}$	$\bar{C}D$	CD	$C\bar{D}$
$\bar{A}\bar{B}$	X 0	0 1	1 3	1 2
$\bar{A}B$	0 4	0 5	0 7	0 6
AB	1 12	1 13	1 15	1 14
$A\bar{B}$	1 8	1 9	1 11	1 10

G_2, G_1

$G_1 = A$, $G_2 = \bar{B}C$

$Y_0 = D_3 + \bar{D_2} D_1$

$2^4 = 16$

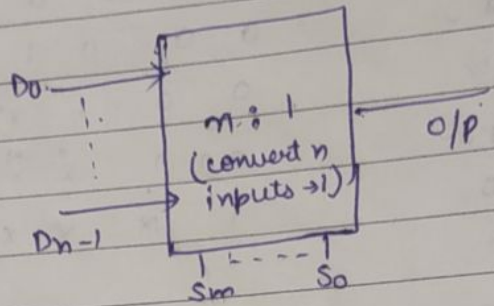
$\frac{16:1}{n:1}$

$2^m = n$

– Multiplexers and Demultiplexers :-

? Multiplexers :- to control O/P data or to tell / command which data line to transfer – Select line

$2^m = n$ no. of inputs



Types of Multiplexers.

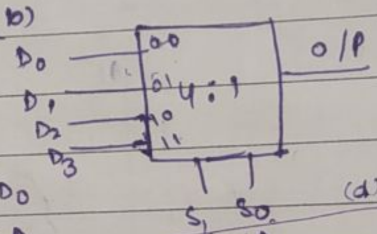
(a) 2 : 1 (b) 4 : 1 (c) 8 : 1 (d) 16 : 1

(a)

$S_0 = 0$ O/P = D_0

$S_1 = 1$ O/P = D_1

(b)



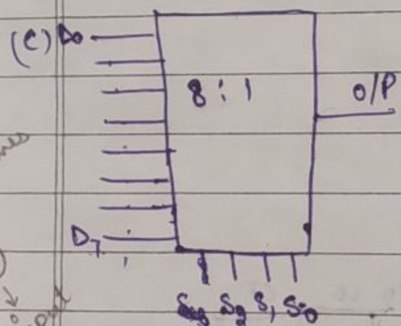
S_1	S_0	O/P
0	0	D_0
0	1	D_1
1	0	D_2
1	1	D_3

(d) (c)

8 4 2 1

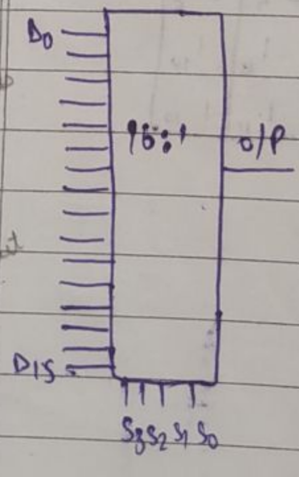
	S_3	S_2	S_1	S_0	O/P (c)	O/P (d)
0	0	0	0	0	D_0	D_0
1	0	0	0	1	D_1	D_1
2	0	0	1	0	D_2	D_2
3	0	0	1	1	D_3	D_3
4	0	1	0	0	D_4	D_4
5	0	1	0	1	D_5	D_5
6	0	1	1	0	D_6	D_6
7	0	1	1	1	D_7	D_7
8	1	0	0	0		D_8
9	1	0	0	1		D_9
10	1	0	1	0		D_{10}
11	1	0	1	1		D_{11}
12	1	1	0	0		D_{12}
13	1	1	0	1		D_{13}
14	1	1	1	0		D_{14}
15	1	1	1	1		A_{15}

(c)



2^{3} → select lines = 8 → input

(d)



2^{4} → select lines = 16 → input

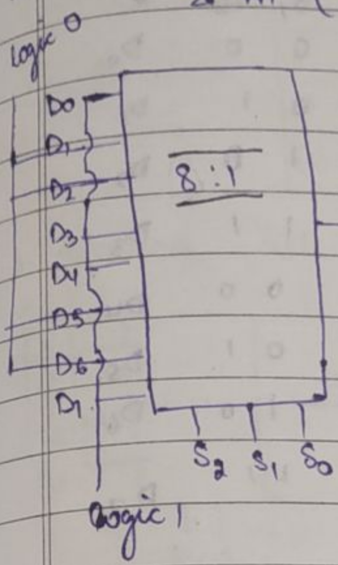
74150 16:1 (MUX)

DM74150N

Implementation of Multiplexers :-

Q. Implement using Multiplexer.

$\Sigma m (0,3,4,7)$

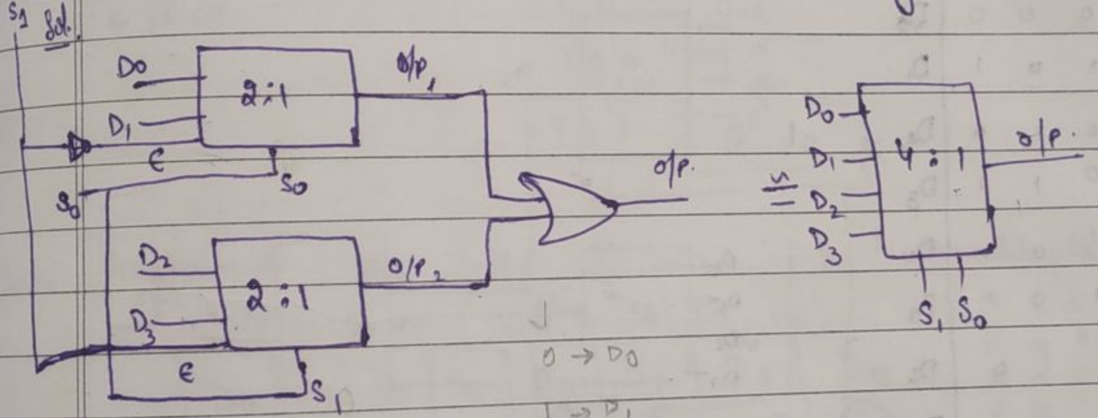


	S_2	S_1	S_0	S (Sum)	Cout (Carry)	o/p
0	0	0	0	1	0	1
1	0	0	1	1	0	
2	0	1	0	1	0	
3	0	1	1	0	1	1
4	1	0	0	1	0	1
5	1	0	1	0	1	
6	1	1	0	0	1	
7	1	1	1	0	1	1

⇒ MULTIPLEXER - TREE

Ques. Implement 4 : 1 multiplexer using two 2 : 1 multiplexer

Sol.

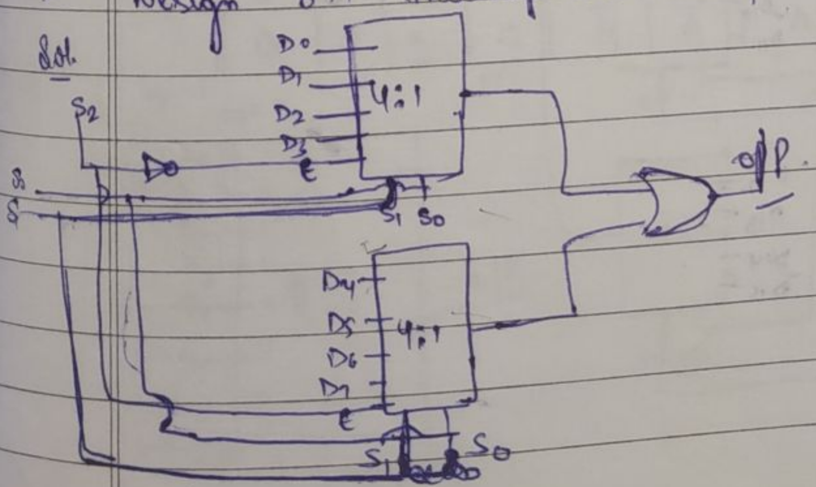


S_1	S_0	
0	0	D_0
0	1	D_1
1	0	D_2
1	1	D_3

4-Sept

Design 8:1 multiplexer using two 4:1 multiplexer

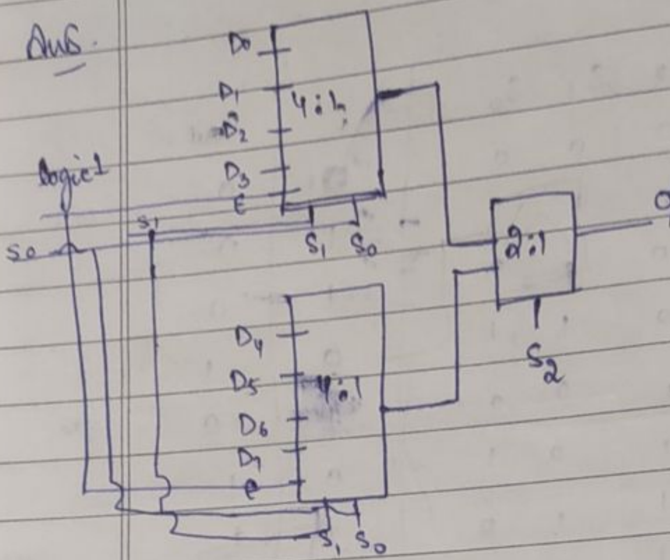
Sol.



S_2	S_1	S_0	
0	0	0	D_0
0	0	1	D_1
0	1	0	D_2
0	1	1	D_3
1	0	0	D_4
1	0	1	D_5
1	1	0	D_6
1	1	1	D_7

Ques. Using only multiplexers i.e without any basic gates derive 8:1 multiplexer.

Ans.



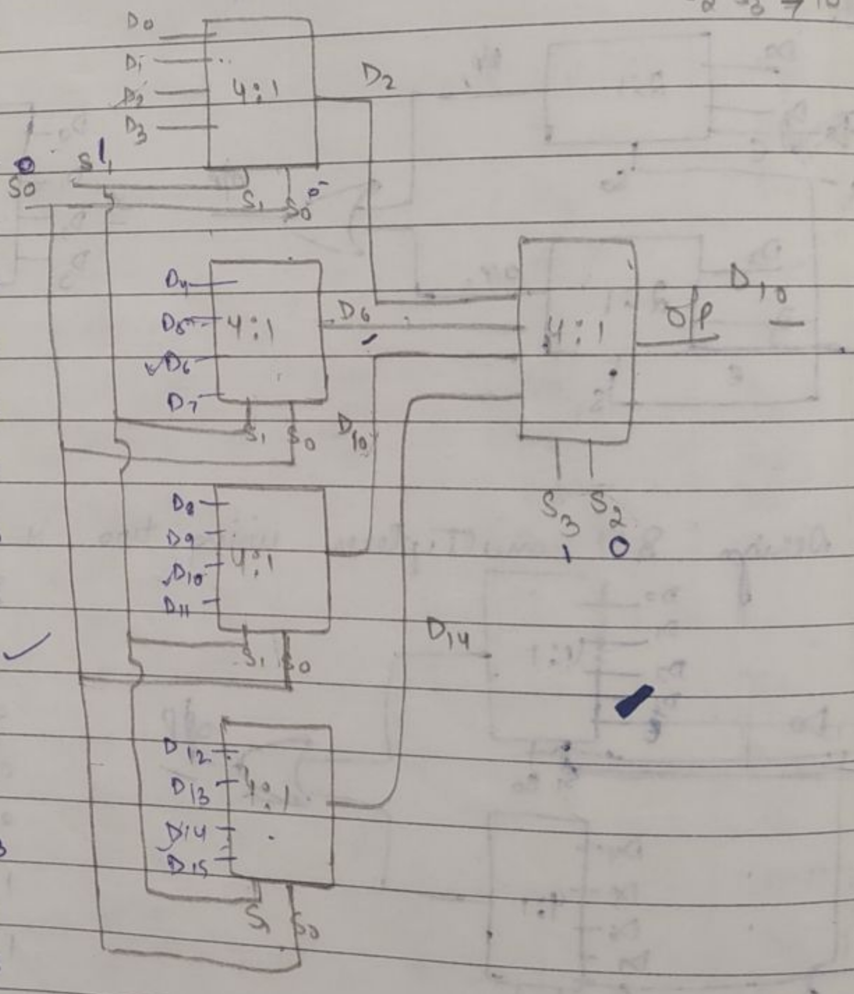
S_2	S_1	S_0	D
0	0	0	D_0
0	0	1	D_1
0	1	0	D_2
0	1	1	D_3
1	0	0	D_4
1	0	1	D_5
1	1	0	D_6
1	1	1	D_7

Ques. Draw / Implement the 16:1 multiplexer using only 4:1 multiplexer

Let $S_0\ S_1 \rightarrow 10$

$S_2\ S_3 \rightarrow 10$

S_3	S_2	S_1	S_0	
0	0	0	0	D_0
0	0	0	1	D_1
0	0	1	0	D_2
0	0	1	1	D_3
0	1	0	0	D_4
0	1	0	1	D_5
0	1	1	0	D_6
0	1	1	1	D_7
1	0	0	0	D_8
1	0	0	1	D_9
1	0	1	0	D_{10} ✓
1	0	1	1	D_{11}
1	1	0	0	D_{12}
1	1	0	1	D_{13}
1	1	1	0	D_{14}
1	1	1	1	D_{15}



If both are included → 1
If no one is include → 0
other corresponding → ✓

left MSB.

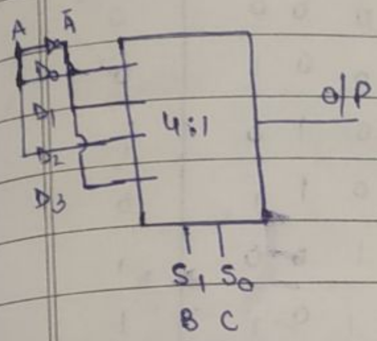
4/sept.

Ques. Implement the logic function using 4:1 multiplexer
$Y = \Sigma m(1,3,4,6)$

Sol. 1. Apply two variables B and C to the select lines
2. Write a design table for leftover Most significant A.

	D_0	D_1	D_2	D_3
0 ($\bar{A}$)	0	1	2	3
(1) A	4	5	6	7

3. Encircle all the given terms in table.



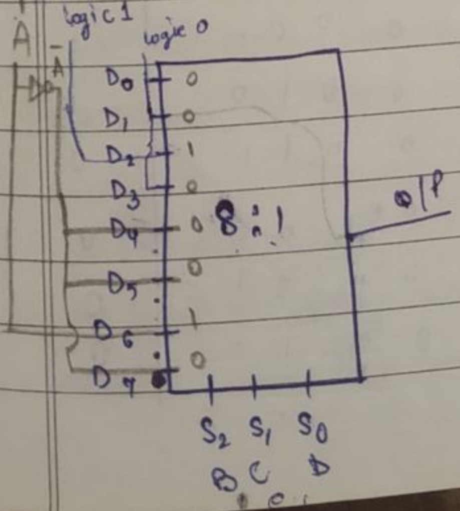
A	B	C	Y
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

	D_0	D_1	D_2	D_3
(0) $\bar{A}$	0	1	2	3
(1) A	4	5	6	7
	A	$\bar{A}$	A	$\bar{A}$

Ques. Implement the foll. function 8:1 multiplexers
$Y = \Sigma m(2,4,5,7,10,14)$

	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7
$\bar{A}$	0	1	2	3	4	5	6	7
A	8	9	10	11	12	13	14	15
	0	0	1	0	$\bar{A}$	$\bar{A}$	A	$\bar{A}$

A	B	C	D	Y
0	0	0	0	0
0	0	0	1	0
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	1
1	1	1	1	0



(2) Minimization of logic fxn. OR, AND, NOT, NOR, NAND, EX-OR, EX-NOR, Basic theorem of Boolean Algebra, SOP & POS, Canonical form, Minigni K-Map, Q-M Map.

(3) Combinational Circuits (CC) - Intro, C.C Design, Encoders, decoders, Adders, Subtractors & Code Converters, Parity checker, Seven segment display, Mag. comparator, MUX, DeMUX, Implementation of C.C's using MUX

(SC) Sequential Circuits :- Intro, flip flops, clocked flip flops, SR, JK, D, T and edge triggered flip flop, Excitation tables of F.F, Shift Registers, Type of Shift Reg. ; Counter - types, design with state eq. & state diag.

(4.) D/A & A/D converters - Intro, weighted D/A converter, Binary ladder D/A converter, steady state accuracy test, D/A accuracy & resolution, || A/D convertor, Counter type A/D Converter Successive approx. A/D converter, Single & dual slope A/D converter, A/D accuracy & resolution

Logic Families :- RTL, DCTL, DTL, TTL, ECL, CMOS & its various types, Comp. of logic families.

Combinational ~~sequential~~ switching circuits are those whose o/p levels at any instant of time are depended only on the levels present at the inputs at that time. Any prior I/P level conditions have no effect on the present output because they ~~h~~ have no memory. A circuit consisting of only logic gates is a combinational circuit.

Page No. ______
Date:

Maxterms are known as False minterms.
PI obtained by using the maxterms – False PI
The FPI which contains atleast one 0 which cannot be covered by any other FPI is called Essential FPI (EFPI)

- A Non-PI is a minterm which does not have any adjacent minterms i.e cant form a part of a bigger sq.
- A Non-EPI is a selective PI. It is neither essential nor redundant. It may or may not appear in the minimal expression.

Limitations of K-Map

→ This method is convenient as long as the no. of variables doesn't exceed 6. As no. of variables exceed it is impossible/difficult to make judgements about the combinations form.
→ It is a manual technique and simplification process is heavily dependent on the human abilities.
→ It cannot be programmed.

Date:

Syllabus

(1) Number System & Binary Code :-
Intro ; Binary , Octal & Hexadecimal No. system Conversion, Signed & Unsigned no. , Binary Addition & subtraction ; Sub. using 1's & 2's complement, ASCII code , Excess 3 Code , Grey code , BCD code & BCD addition .

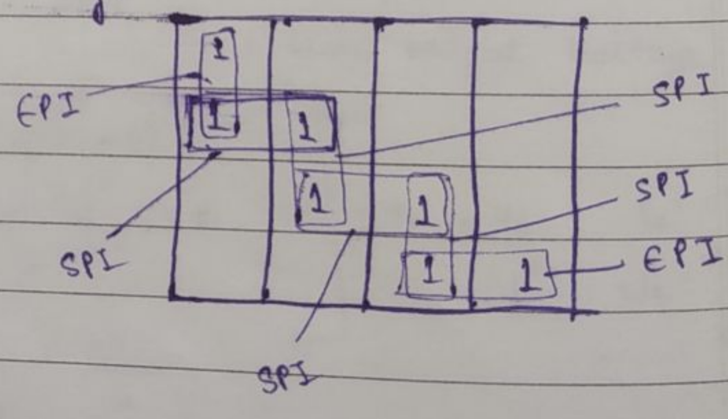
K-Map.

Prime Implicants , Essential P.I , Redundant P.I, Selective P.I

- Each sq. or rect. made up of the bunch of adjacent minterms is called a subcube.
- Each of these subcubes is called a prime implicant (PI)
- The PI which contains atleast one 1 which cannot be covered by any other prime implicant is called an essential prime implicant (EPI).
- The PI whose each 1 is covered by one EPI is called a redundant prime implicant (RPI)
- A PI which is neither a EPI nor a RPI is called a Selective Prime Implicant (SPI)

eg:-

eg.



Page No. ______
Date:

No. of step = $\frac{V_{in}}{\text{Resolution}}$

=

A-D Convertors.

2-Nov

Types :-

(i) Parallel comparator ADC

(ii)

(iii)

$2^{n}-1$ = No. of comp. req. (n = no. of bit)

Now. If ~~ded.~~ another data is 011

$V_2\ V_1\ V_0 = 011$

we have logic $0 = 0V$

logic $1 = +7V$

$$V_A = \frac{V_0/R_0 + V_1/(R_0/2) + V_2/(R_0/4)}{1/R_0 + 1/(R_0/2) + 1/(R_0/4)}$$

$$= \frac{7/R_0 + 7/(R_0/2) + 0}{1/R_0 + 2/R_0 + 4/R_0}$$

$$= \frac{7/R_0 + 14/R_0}{7/R_0} = \frac{7+14}{7} = \frac{21}{7} = 3 \text{ volts}$$

So analog o/p is analog +3 V

Drawbacks :- 1. Each resistor in the network has different values.

2. More resistors are used thus much greater currents are needed.

We R-2R ladder.

Parameters

→ Resolution = step size = $\frac{V_R}{2^n}$ ← bit

→ % ~~Resistance~~ Resolution = $\frac{\text{step size}}{fso} \times 100$

↳ full scale output voltage.

$fso \Rightarrow \text{step size} \times (2^N - 1)$

$\Rightarrow V_R - \text{weight of LSB} = V_R - \frac{V_R}{2^n}$

Let $n=3$

$V_R - \frac{V_R}{2^3}$

$= \frac{7V_R}{8}$

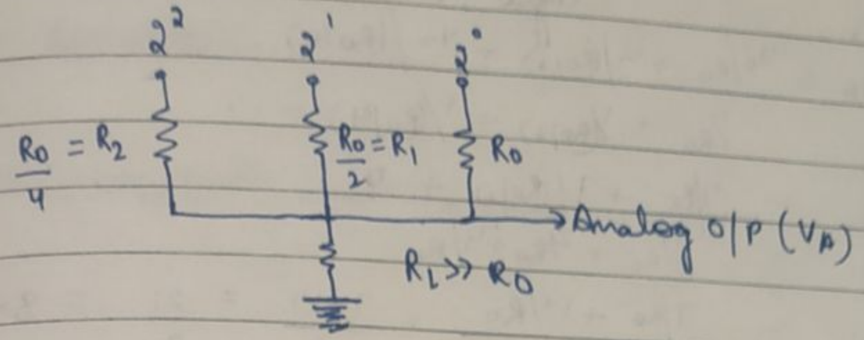
$$\frac{\text{Step size}}{(2^n - 1)} \times \frac{100}{\text{Step size}} = \frac{100}{2^N - 1}$$

eg. $N = 2$

$\frac{100}{2^2-1} = \frac{100}{3} = 33.\overline{3}$

Page No. ___ Date:

2. These ~~the~~ three voltages represent the digital bits of they must be summed together to form the analog voltage.



This ckt can be solved using Milliman theorem.
Acc. to Milliman theorem

$$V = \frac{V_0/R_0 + V_1/R_1 + V_2/R_2 + V_3/R_3 + \dots}{1/R_0 + 1/R_1 + 1/R_2 + 1/R_3 + \dots}$$

$\boxed{V = IR}$ here All resistances are in Parallel

$$\therefore V = \frac{I}{R}$$

As in general $I = V/R$

$\therefore$ Req. eq. ~~of~~ Milliman's Theorem.

eg. If digital I/P is 001 i.e $V_2V_1V_0 = 001$
Let us assume that logic 0 = 0 V
& " " " " logic 1 = +7 V

Acc. to Milliman's Theorem

$$V_A = \frac{V_0/R_0 + V_1/(R_0/2) + V_2/(R_0/4)}{1/R_0 + 1/(R_0/2) + 1/(R_0/4)}$$

$$= \frac{7/R_0 + 0 + 0}{1/R_0 + 2/R_0 + 4/R_0} = \frac{7/R_0}{7/R_0} = +1\text{ V}$$

So ~~at~~ analog O/P is analog +1 V.

Comparision of logic families

	Propagation delay time (ns)	Power dissipation per gate (mW)	Noise Margin (v)	Fan in	Fan out	Cost
TTL	9	10	~~4323~~ 0.4	8	10	low
ECL	1	50	0.25	5	10	High
MOS	50	0.1	1.5		10	low
CMOS	<50	0.01	5	10	50	low
IIL	1	0.1	0.34	5	8	Very low

Binary Weighted Register (Resistance) D-A converter.

Digital I/P	Analog O/P
000	0V
001	+1V
010	+2V
011	+3V
100	+4V
101	+5V
110	+6V
111	+7V

Resistive divider ckt do two things to change digital input into an equivalent analog O/P.

1. The 2^0 bit must be changed to +1 volts &
   2^1 bit must be changed to +2V &
   2^2 bit must be changed to +4V.

for Test-3.

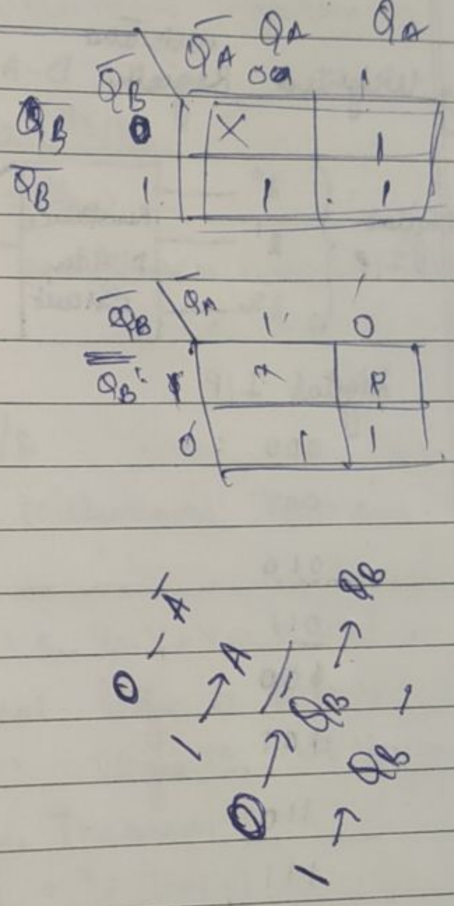
Page No. ____
Date: 29/10/18

counters & design
logic families, there different parameters & comparision

Logic families

S

Q_B	Q_A	$\overline{Q_B}$	$\overline{Q_A}$	dr 2 -- dto
0	0	1	1	1
0	1	1	0	1
1	0	0	1	1
1	1	0	0	0

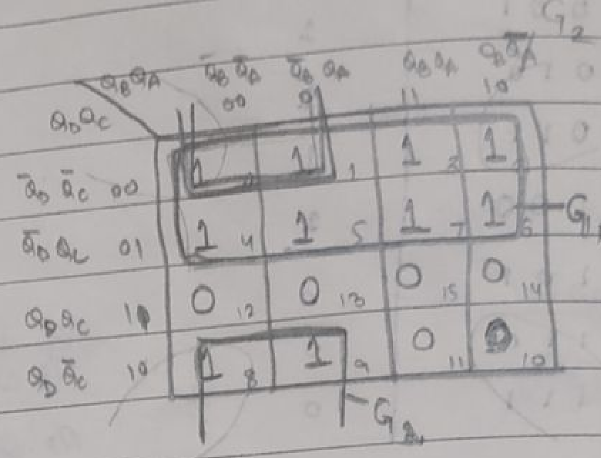


Date:

HW
Imp → Make Mode -10 counter. (Decade Counter)

bubbled

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



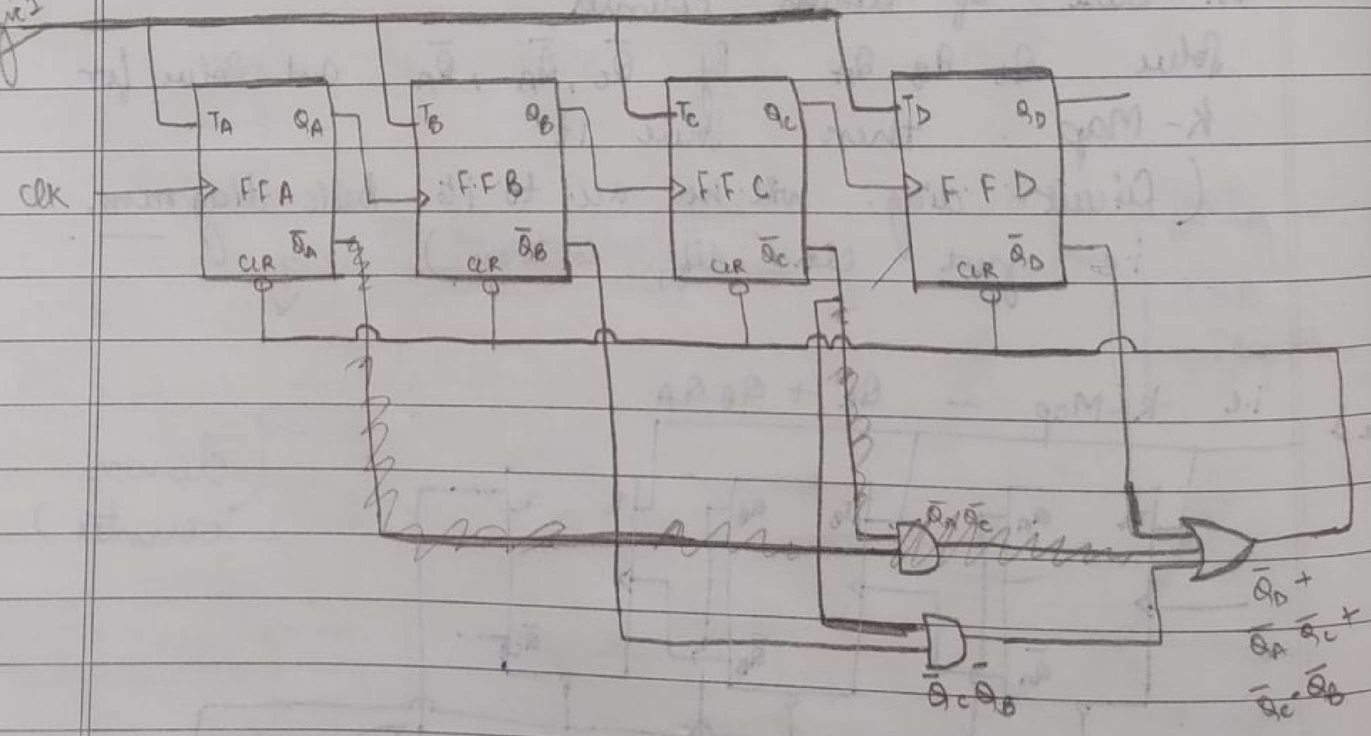
G_1 :- $\bar{Q}_D$

~~G_2 :- $\bar{Q}_C * \bar{Q}_A$~~

G_2 :- $\bar{Q}_C * \bar{Q}_B$

$Y = \bar{Q}_D + \cancel{\bar{Q}_A\bar{Q}_C} + \bar{Q}_C \cdot \bar{Q}_B$

logic



$\bar{Q}_C \bar{Q}_B \bar{Q}_A$
0 0 0

Asynchronous Mode 5

	$Q_C Q_B Q_A$	Y (Bubble)
0	0 0 0	1
1	0 0 1	1
2	0 1 0	1
3	0 1 1	1
4	1 0 0	1
5	1 0 1	0
6	1 1 0	0
7	1 1 1	0

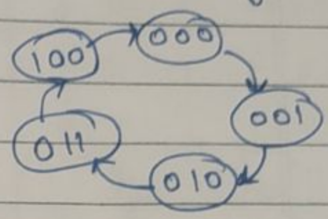
$Q_C \backslash Q_B Q_A$	00	01	11	10
0	1	1	1	1
1	1	0	0	0

$\bar{Q}_C + \bar{Q}_B \bar{Q}_A$

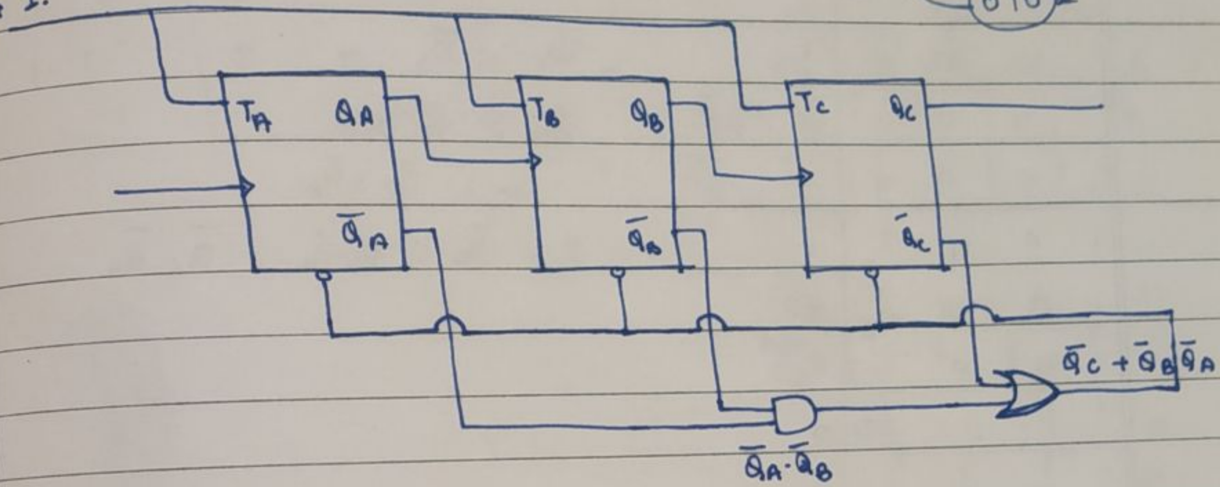
$\bar{Q}_B \cdot \bar{Q}_A$

$Q_B \cdot Q_A$

State diagram



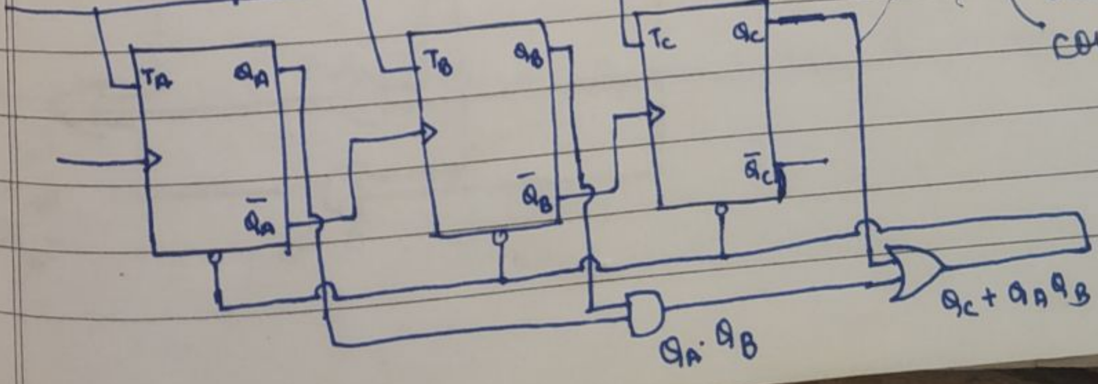
Logic 1.



In case of down counter
Solve $Q_C\ Q_B\ Q_A$ By $\bar{Q}_C, \bar{Q}_B, \bar{Q}_A$ and Solve for K-Map. then Solve it.
(Circuit diag. will be acc. to its basic diagram.
i.e just clk diff. is there). i.e

i.e K-Map — $Q_C + Q_B Q_A$

Logic 1



(down counter)

Mode

$2^{(\text{no. of flip-flops})}$

eg. in 2 bit. $2^2 = 4$ Mode $(0-3)$
in 3 bit $2^3 = 8$ mode $(0-7)$
$2^4 = 16$ Mode $(0-15)$

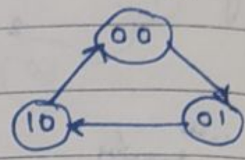
$2^3\ 2^2\ 2^1\ 2^0$
8 4 2 1

Ques. Design a mode 3 - asynchronous 2 bit ripple counter.

using Bubble ✓ preferred

	Q_B	Q_A	Y	Y
0	0	0	0	1
1	0	1	0	1
2	1	0	0	1
3	1	1	1	0

state diagram



$Q_B \backslash Q_A$	0	1
0	0	0
1	0	1

$= Q_A \cdot Q_B$. (AND)

bubble reverses the result/ input

$Q_B \backslash Q_A$	0	1
0	1	1
1	1	0

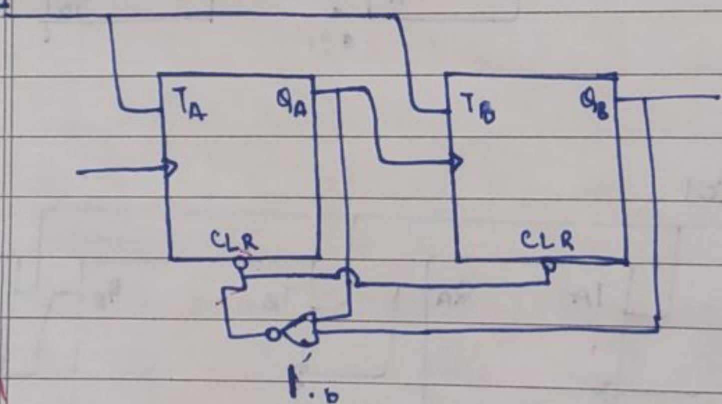
$\overline{Q_A} + \overline{Q_B}$

$Q_B \backslash Q_A$	0	1
0	1	1
1	1	0

$= \overline{Q_A} + \overline{Q_B}$ (NAND)

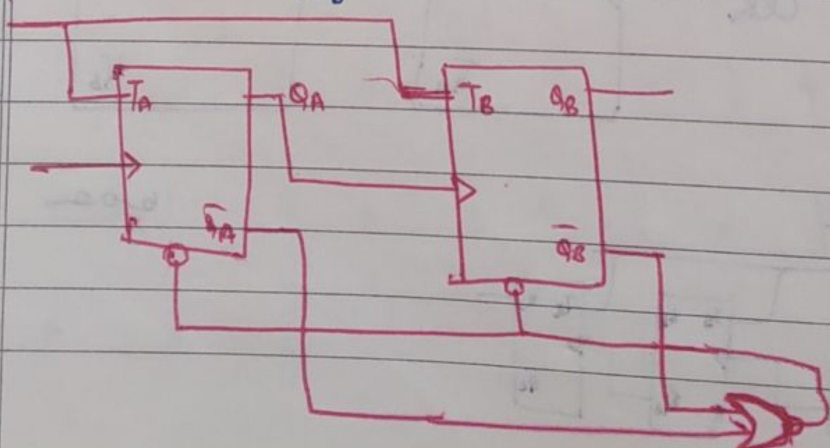
give to CLR

Logic 1



[1st log use LSB Flip Flop]

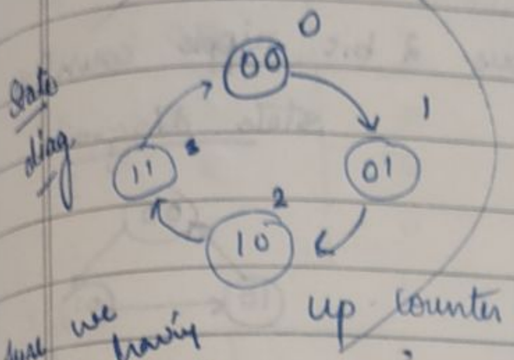
logic 1



$Q_B \; Q_A = 00$

Q_B	Q_A
0	0
0	1
1	0
1	1
0	0

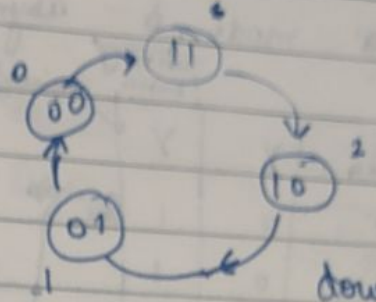
State diag.



up counter

here we will having Q_n as the clk to next

23-oct-2019

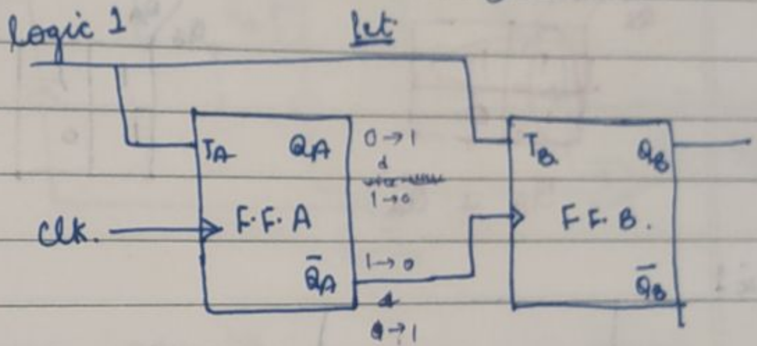


down counter.

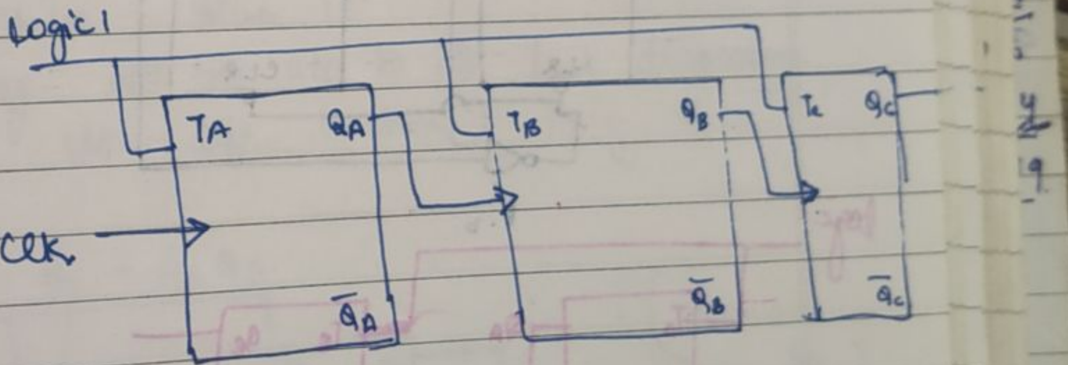
here we will be having $\overline{Q_n}$ as the clk to next.

down counter.

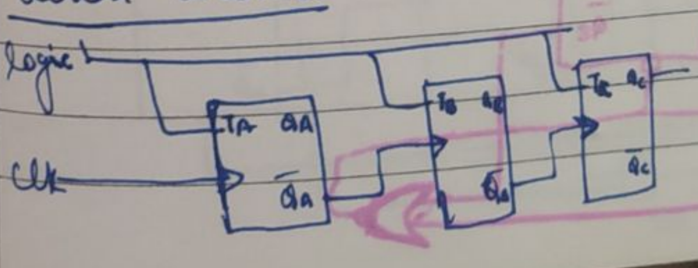
Q_B	Q_A
1	1
1	0
0	1
0	0



For 3-bit. upcounter.



down counter



16/oct/18

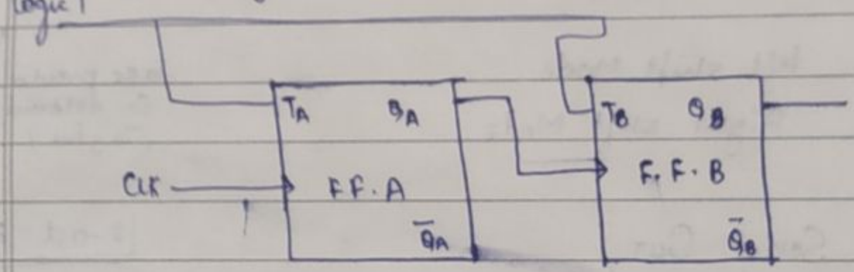
Counters : The digital circuit used for counting pulses is called Counter. It is sequential circuit and made up of group of flip flops. It is the widest application of flip-flops. Mostly the counters are used for counting pulses but with some modification they are often used to measure frequency or time period.

Types of counter :-

- Basically, 2 types of counters
1. Asynchronous Counter — don't provide clk together, doesn't work at same time
2. Synchronous Counter — provide clk equally at a same time
   - ↳ Ring Counter ↳ Johnson Counter

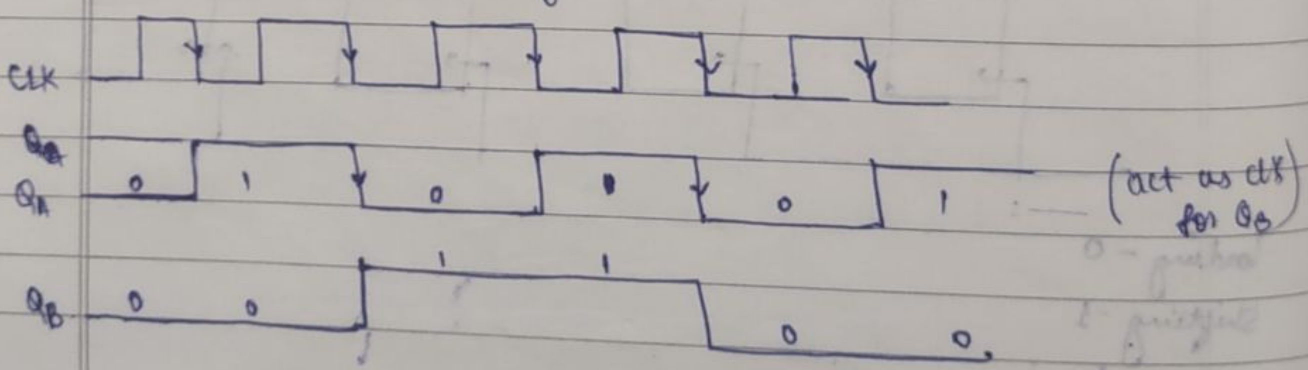
- Asynchronous Counter (Ripple Counters)

2-Bit Asynchronous Counter (Ripple Counter)



Let initially both the F.F are at rest state. Let the Outputs Q_B, Q_A be 0,0 and also assume that Both the F.F work on -ve edge triggering.

On the I^{st} -ve edge.



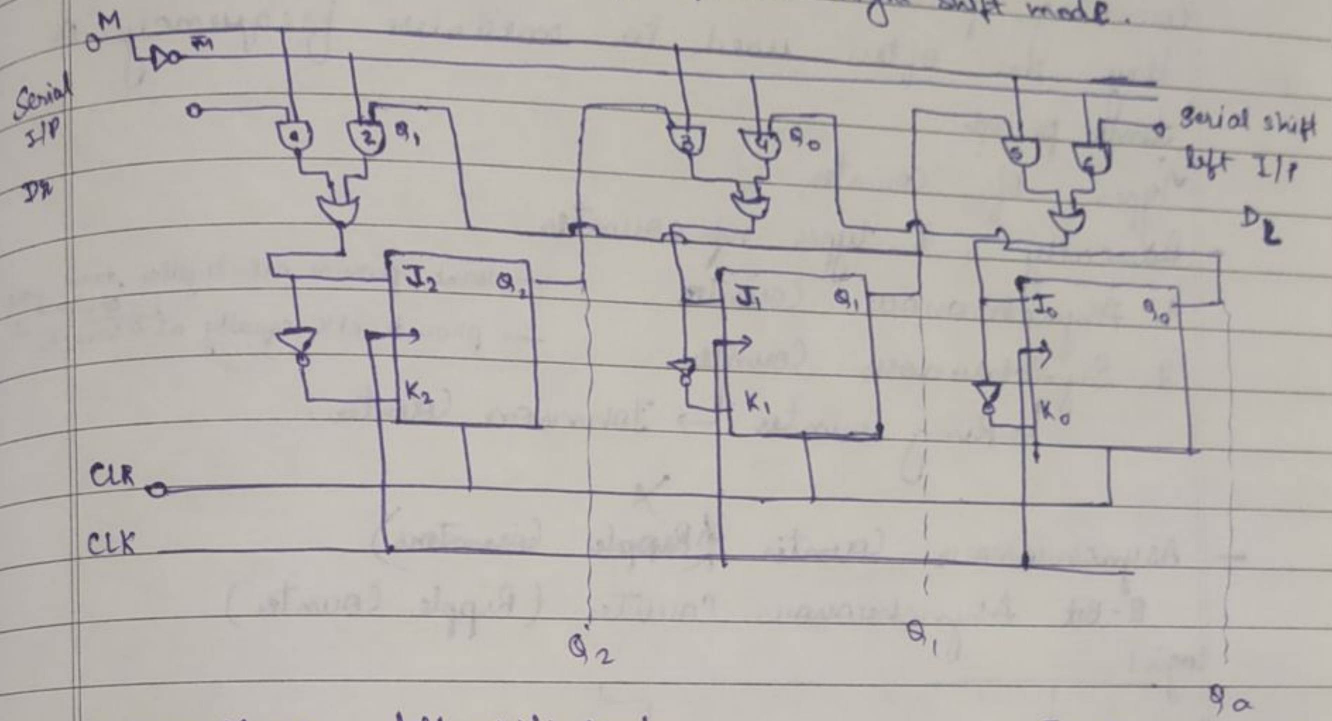
for Test3 :-

Part 4. Counters
A-D , D-A Convertor. ; logic families ;

10-oct-2018.

Serial In Parallel Out (Bi-directional Shift register)

Left Shift Mode   Right shift mode.

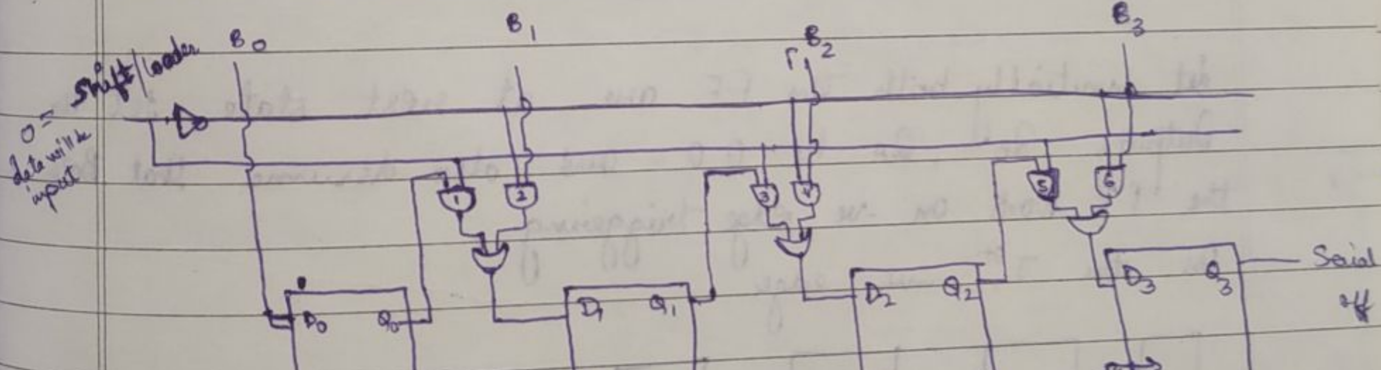


When M = 0   left shift Mode
When M = 1   Right shift Mode.

Take previous value for determination (To give)

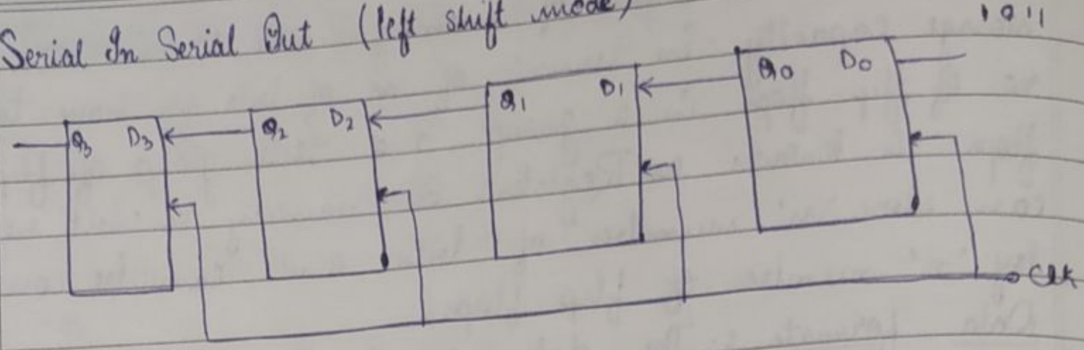
12-oct-2018

Parallel In Serial Out



loading - 0
Shifting - 1

Serial In Serial Out (left shift mode)

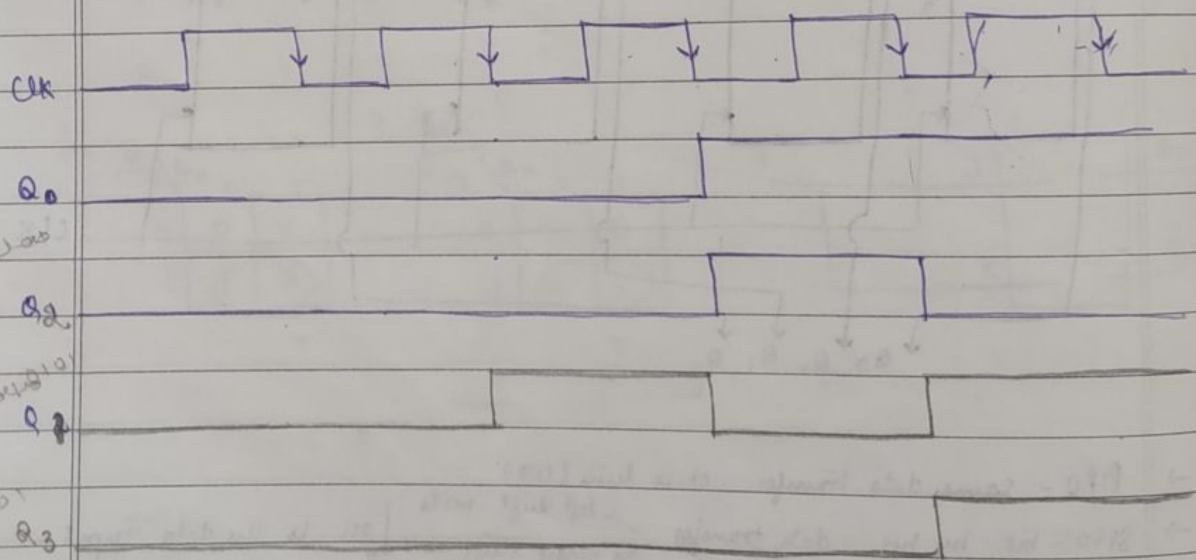


Before applying a clk let the o/p of ff Q_3, Q_2, Q_1, Q_0 be 0000

Firstly Apply MSB to it

			Clk
	$Q_3\ Q_2\ Q_1\ Q_0$	0000	x
So at the first clk	→	0001	↓
2nd clk		0010	↓
3rd clk		0101	↓
4th clk		1011	↓

	Clk.	Q_3	Q_2	Q_1	Q_0
Ist clk.	x	0	0	0	0
IInd clk	↓	0	0	1	0
IIIrd clk	↓	0	1	0	1
IVth clk	↓	1	0	1	1



Parity Encoder
in syllabus.

Page No.
Date: 5 | 10 | 18

Registers :-

Flip Flop can store 1 bit of memory and to increase the storage capacity in terms of no of bits we have to use no. of flip-flops in a group and this group of flip-flops is known as Register. So, namely 'n' bit register can store 'n' number of bits and can be constructed by 'n' number of flip-flops.

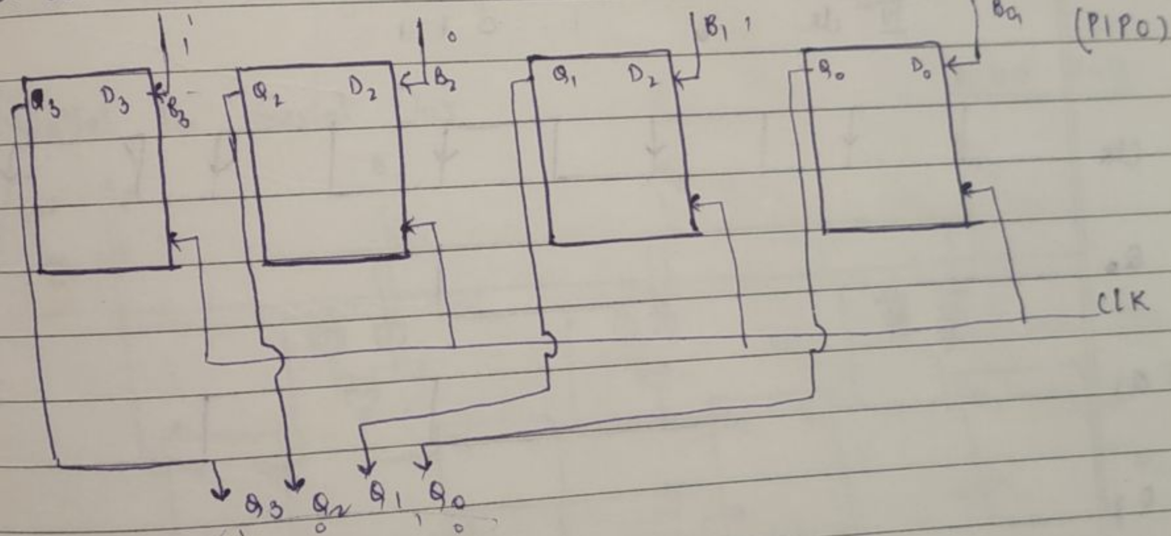
Data formats :- The data can be loaded serially or in parallel (all the bits at a time)

Classification :- Classified on the basis of mode of operation

1. Serial In and Serial Out (SISO)
2. Serial In and Parrallel Out (SIPO)
3. Parallel In and Serial Out (PISO)
4. Parallel In and Parallel Out (PIPO)

Buffer Register :- (a) 4 Bit BR :-

simplest type of register & constructed by using 4 'D' flip flops & we can construct 'n' Bit BR.



→ PIPO - same data transfer at a time (CLK)

→ SISO - bit by bit data transfer → left shift mode / → Right shift mode] acc. to the data input.

- Convert SR flip flop to D flip flop

D	Q_n	Q_{n+1}	S	R
0	0	0	0	X
1	0	1	1	0
0	1	0	0	1
1	1	1	X	0

for S

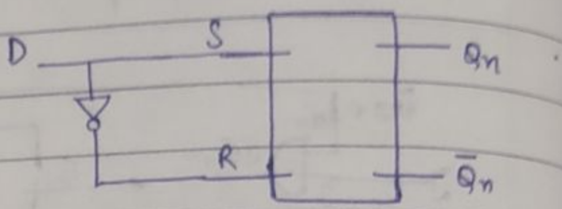
D \ Q_n	0	1
0	0	0
1	1	X

$= D$

for R

D \ Q_n	0	1
0	X	1
1	0	0

$= \overline{D}$



- Convert JK flip flop to T flip flop.

T	Q_n	Q_{n+1}	J	K
0	0	0	0	x
1	0	1	1	x
1	1	0	x	1
0	1	1	x	0

for J

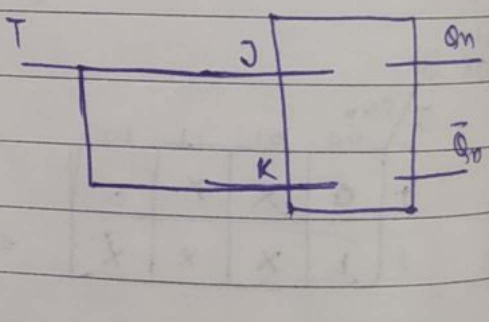
T \ Q_n	0	1
0	0	x
1	1	X

T

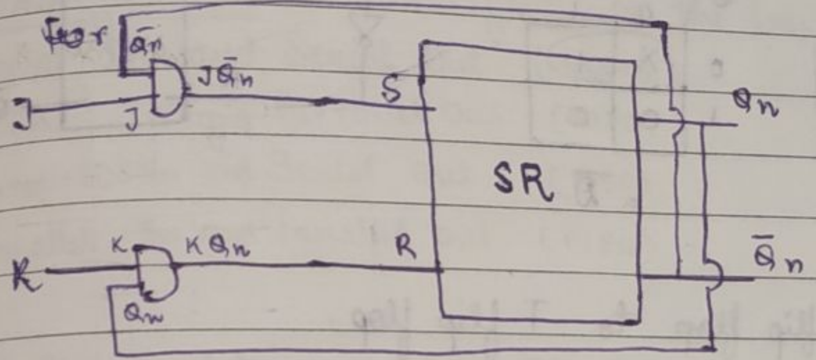
for K

T \ Q_n	0	1
0	X	0
1	X	1

T



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



- Convert JK flip flop to SR flip flop.

SR	Q_n	Q_{n+1}	JK
0 x	0	0	0 x
1 0	0	1	1 x
0 1	1	0	x 1
x 0	1	1	x 0

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

for J

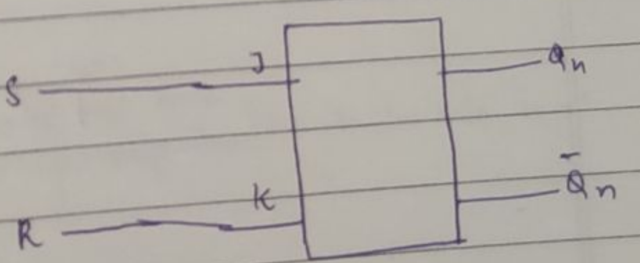
S \ RQ_n	00	01	11	10
0	0	x	x	0
1	1	x	x	x

= S

for K

S \ RQ_n	00	01	11 (RQ_n)	10 ($R\bar{Q}_n$)
0	x	0	1	x
1	x	0	x	x

= R



- Conversion of J-K-flip-flop to T-flip-flip

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

<u>for J</u>

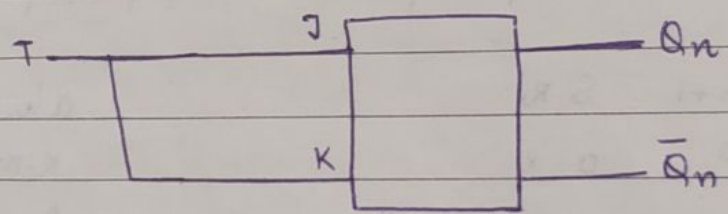
T \ Q_n	0	1
0	0	X
1	1	X

$= T$

for K

T \ Q_n	0	1
0	X	0
1	X	1

$= T$



- Convert SR flip flip to JK flip flop.

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

<u>For S</u>

J \ KQ_n	$\bar{K}\bar{Q}_n$ 00	$\bar{K}Q_n$ 01	KQ_n 11	$K\bar{Q}_n$ 10
$\bar{J}$ 0	0	X	0	0
J 1	1	X	0	1

$= J\bar{Q}_n$

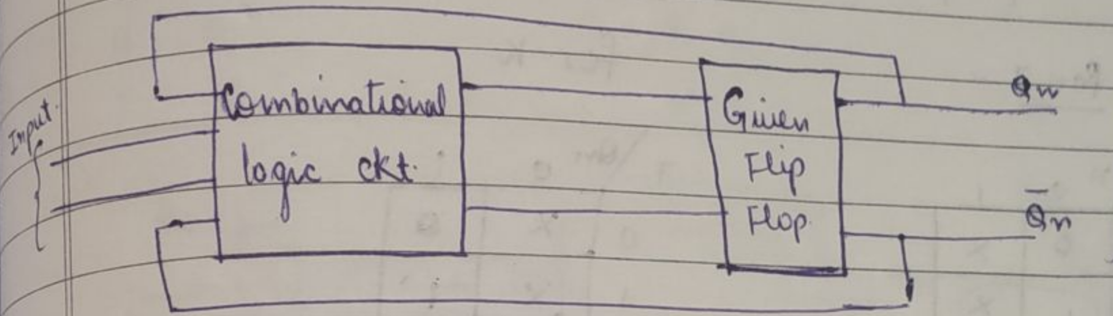
<u>For R</u>

J \ KQ_n	$\bar{K}\bar{Q}_n$ 00	$\bar{K}Q_n$ 01	KQ_n 11	$K\bar{Q}_n$ 10
$\bar{J}$ 0	X	0	1	X
J 1	0	0	1	0

$= KQ_n$

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

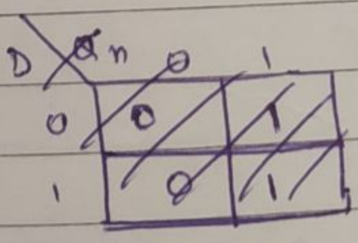
3-Oct-2018



Conversion from S-R flip flop to D-flip flop

D	Q_n	Q_{n+1}	S	R
0	0	0	0	X
1	0	1	1	0
0	1	0	0	1
1	1	1	X	0

always made K-Map with Q_n
Input 1 + Q_n = 2 var
Input 2 + Q_n = 3 var
and so on



for S

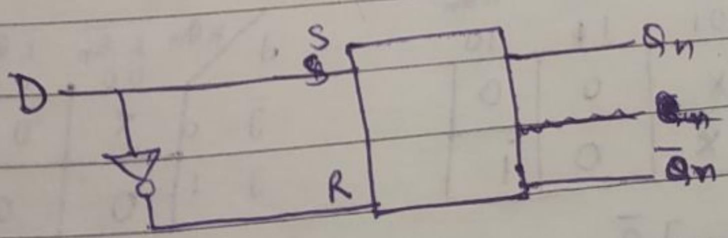
D \ Q_n	0	1
0	0	0
1	1	X

= D

for R

D \ Q_n	0	1
0	X	1
1	0	0

= $\bar{D}$



JK flip flop

characteristics

J	K	Q_n	Q_{n+1}	
0	0	0	0	No change
0	0	1	1	
0	1	0	0	Reset
0	1	1	0	
1	0	0	1	set
1	0	1	1	
1	1	0	1	Toggle
1	1	1	0	

Take care:-

J	K	
0	0	N.C
0	1	Reset
1	0	Set
1	1	Toggle

Exictation Table

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

T- flip flop.

Characteristics

T	Q_n	Q_{n+1}	
0	0	0	No change
0	1	1	
1	0	1	Toggle
1	1	0	

T	
0	N.C
1	Toggle

Exictation Table

Q_n	Q_{n+1}	T
0	0	0
0	1	1
1	0	1
1	1	0

Characteristics of S-R flip flop.

S	R	Q_n	Q_{n+1}	
0	0	0	0	] No change
0	0	1	1	
0	1	0	0	] Reset
0	1	1	0	
1	0	0	1	] Set
1	0	1	1	
1	1	X	X	] Invalid

Previous → Present
Present → Next

$Q_n, \bar{Q}_n \rightarrow Q_{n+1}, \bar{Q}_{n+1}$

Truth Table

S	R	
0	0	N.C
0	1	Reset
1	0	Set
1	1	Race/Invalid

same value = X

Exitation Table :

Present Q_n	Next Q_{n+1}	S	R
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0

D - flip flop.

Characteristics

D	Q_n	Q_{n+1}	
0	0	0	] Reset
0	1	0	
1	0	1	] Set
1	1	1	

Truth Table

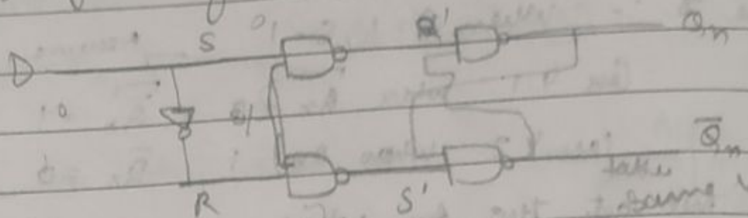
D	Q_n	
0	0	Reset
1	1	Set

Exictation Table :

Q_n	Q_{n+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

-ve level triggering gated D' flip flop

Delay flip flop — D-flip flop



D	Q_n	Q_{n+1}
0	Q_n	0
1	Q_n	1

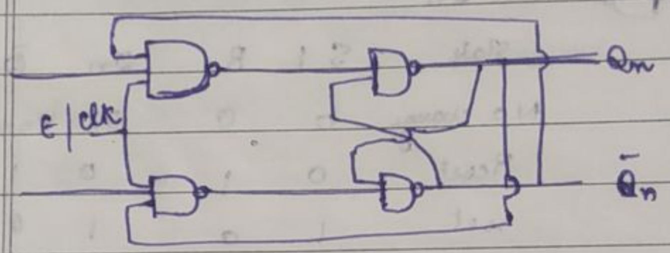
take same values

[Race cond - unavoid cond]
Race around - discussed later

26 | sept | 2018

→ JK- flip flop

Discuss all the cases in detail with diff. diagram.



E	J	K	Q_{n+1}
0	x	x	N.C(Q_n)
1	0	0	N.C. (Q_n)
1	0	1	Reset
1	1	0	Set
1	1	1	Toggle ($\bar{Q}_n$)

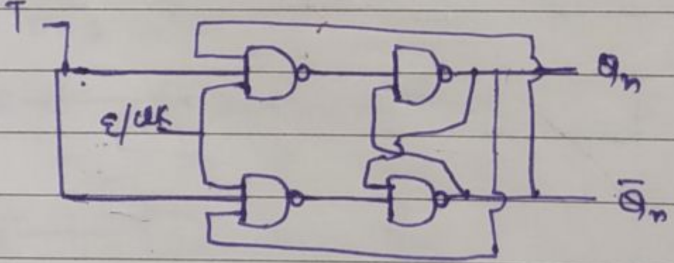
Toggle

(Race - around cond.)

Q_n	→ Q_{n+1}
0	→ 1
1	→ 0

same value
diff - value

→ T- flip-flop.



E	T	Q_{n+1}
0	X	NC.
1	0	N.C
1	1	Toggle

28 | sept | 2018

$\boxed{\text{Conversion of Flip Flops.}}$

Case 3 :- When $R = 1$, $S = 0$

$Q_n = 0$, $\bar{Q}_n = 1$ Reset state

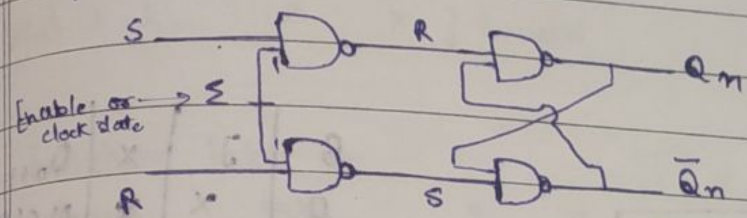
Case 4 :- When $R = 1$, $S = 1$

Case 4.1 when $Q_n = 0$, $\bar{Q}_n = 1$ Assume

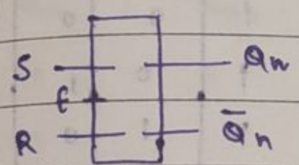
Case 4.2 when $Q_n = 1$, $\bar{Q}_n = 0$

We get the Ans which we Assume so No change state

Gated S-R latch



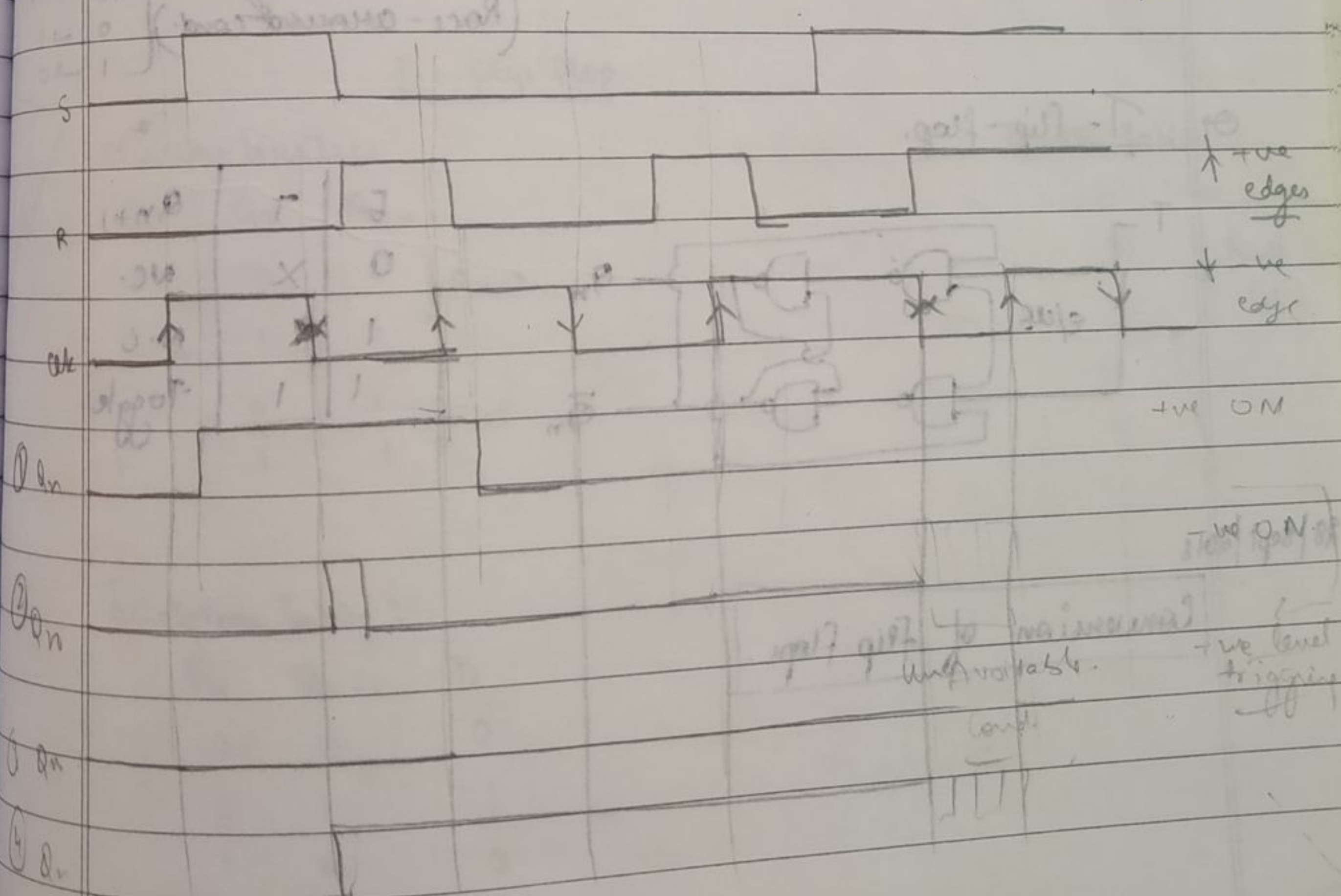
$\bar{S}$	$\bar{R}$	Q_n	$\bar{Q}_n$	Q_{n+1}	$\bar{Q}_{n+1}$
0	0	Q_n	$\bar{Q}_n$	Q_n	$\bar{Q}_n$
0	1	Q_n	$\bar{Q}_n$	0	1
1	0	Q_n	$\bar{Q}_n$	1	0
1	1	Q_n	$\bar{Q}_n$	x	x



State	S	R	Q_n	$\bar{Q}_n$
No change	0	0	$Q_n = Q_{n+1}$	
Reset	0	1	0	1
set	1	0	1	0
Invalid	1	1	1	1

① +ve level trigging ② -ve level ③ +ve edge trigger

21-sept-2018

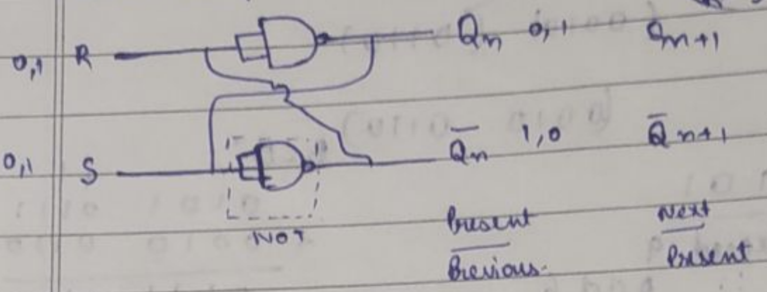


Combinational : o/p not depend on previous o/p

18-Sept-2018

Sequential Circuits - whose output depends on the previous o/p & ...

Cross-Coupled Nand Gate - 1 bit memory cell.
(R-S latch/latch)



A	B	AND	NAND
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

Cases:-

(i) When $R=0$, $S=0$
invalid state $\quad q_n = 1$, $\bar{q}_n = 1$
Prohibited state

latch - no clock pulse is provided

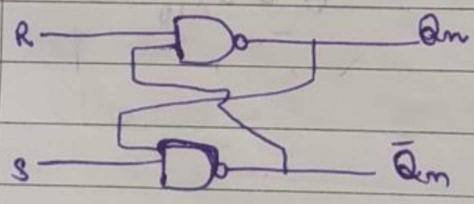
(ii) When $R=0$, $S=1$
$Q_n = 1$, $\bar{Q}_n = 0$

Flip-flops - clock pulse is provided

(iii) When $R=1$, $S=0$
$Q_n = 0$, $\bar{Q}_n = 1$

(iv) When $R=1$, $S=1$
$Q_n = 0$, $\bar{Q}_n = 0$

R-S latch.



State	R	S	Q_n	$\bar{Q}_n$
Invalid	0	0	1	1
Set	0	1	1	0
Reset	1	0	0	1
Nochange	1	1	$Q_n = Q_{n+1}$	

Invalid cond.

Case 1. When $R, S = 0$
$Q_n = 1$, $\bar{Q}_n = 1$ invalid state / Prohibited state

Case II When $R = 0$, $S = 1$
$Q_n = 1$ $\quad \bar{Q}_n = 0$ Set state.

Q. Add $(57)_{10}$ & $(26)_{10}$ using BCD addition

$(57)_{10}$ → (0101) (0111)

$(26)_{10}$ → (0010) (0110)

$\therefore (01010111)_{BCD}$ $(0010\ 0110)_{BCD}$

```
  0101 0111
+ 0010 0110
-----------
  0111 1101
```

0111 1101 — Exceed 9 $\therefore$ Add 6

```
  0111 1101
+ 0000 0110
-----------
  1000 0011
```

$(1000\ 0011)_{BCD}$

$(83)_{10}$

Q. Perform $(57)_{10} + (46)_{10}$ using BCD addition

$(5\ 7)_{10}$ 46

$(0101\ 0111)_{BCD}$ $(0100\ 0110)_{BCD}$

```
  0101 0111
+ 0100 0110
-----------
  1001 1101
+ 0110 0110
-----------
 10000 0011
```

carry, i.e. result is now > 9 $\therefore$ Add 6 to it

$= [0001\ 0000\ 0011]_{BCD}$

$(103)_{10}$

12-Sept-2018

BCD addition

eg. Add $(2)_{10}$ & $(6)_{10}$ using BCD addition

$(2)_{10} = 0010$ + $(6)_{10} = 0110$

$$\begin{array}{r} 0010 \\ +\ 0110 \\ \hline 1000 \end{array}$$

$\rightarrow = 1000 = (8)_{10}$

$(2)_{10} + (6)_{10} = (8)_{10}$

eg. Add $(7)_{10}$ & $(6)_{10}$

$(7)_{10} = (0111)$ & $(6)_{10} = 0110$

$$\begin{array}{r} 0111 \\ +\ 0110 \\ \hline 1101 \end{array}$$

Take care:

$(7)_{10} + (6)_{10} = (1101) = (13)_{10}$

Rule: whenever the result is greater than 9 then add 6 to the result always.

Now in above eg. we will have

$1101 + 0110 = \boxed{1}\ 0011$ ↓ carry

$$\begin{array}{r} 1101 \\ 0110 \\ \hline \boxed{1}0011 \end{array}$$

$= (0001\ \ 0011)_{BCD}$

$(1\ \ 3)_{10}$

$15 - 9$ diff $= 6$

$(4)_{10} = (0100)_2$
$+ (0111)_2$
$(1011)_2$

As the final carry is zero. so the result is in -ve form of in 2's complement representation

$\boxed{0}\ (1011)_2$
-1
$(1010)_2$ — 1's complement
$(0101)_2$ — (Again compl.) - neg. Ans.
$=(5)_{10}$.

Hence. $(4)_{10} - (9)_{10} = (5)_{10}$

Case(iii) when A & B both are -ve

eg. Perform $(-4)_{10} - (6)_{10}$ using 2's complement

STEP 1. obtain 2's complement of $(6)_{10}$

$(6)_{10} = (0110)_2$
$= (1001)_2$ — 1's complement
$+1 = (1010)_2$ — 2nd complement

STEP 2. $(-4)_{10} = (4)_{10} = (0100)_2$ $(-4)_{10} = (1011)_2$ —

Add $(-4)_{10}$ to 2's complement of $(6)_{10}$

$(6)_{10} = (0110)_2 \rightarrow (1001)_2$ - 1st comp
$+1$
$(1010)_2$ - 2st complement

$(4)_{10} = (0100)_2 \rightarrow (1011)_2$
$+1$
$(1100)_2 \rightarrow$ 2st comp.

$(1010)_2 + (1100)_2$ $\quad$ 1 $(0110)_2 \rightarrow$ True form

1010
1100
1 0110

Date:

General Rules for 2's complement

STEP 1. Add $(A)_2$ to 2's complement of $(B)_2$

STEP 2. If final carry is generated then the result is +ve and is in true form.

If the carry generated is zero or no carry then the result is in its -ve form and in 2's complement representation

Cases:-

(i) When A > B (ii) When A < B (iii) A & B = -ve (iv) A = B.

Case (i) When A > B & A & B are +ve

eg. Perform $(9)_{10} - (5)_{10}$ using 2's complement

STEP 1. Obtain 2's comp. of $(5)_{10}$

8421 $(5)_{10} = (0101)_2$

$= (1010)_2$ - 1's complement

+1

$(1011)_2$ - 2's complement

STEP 2. Add $(9)_{10}$ to 2's complement of $(5)_{10}$

$(9)_{10} = (1001)_2$

$(-5)_{10} = (1011)_2$ - 2's comp.

$\boxed{1}\,0100$ — True form.

Hence $(9)_{10} - (5)_{10} = (0100)_2$ - Ans.

Case (ii) When A < B & A & B are +ve

eg. Perform $(4)_{10} - (9)_{10}$ using 2's complement form.

STEP 1. Obtain 2's complement of $(9)_{10}$

$(9)_{10} = (1001)_2$

$(0110)_2$ — 1's comp.

+1

$(0111)_2$ — 2's comp

STEP 2. Add $(4)_{10}$ to 2's comp. of $(9)_{10}$