03.07.18

Chapter-1

Number System.

1. Binary Number system
2. Decimal " "
3. Octal " "
4. Hexadecimal " "
(Hex)

2. Decimal → 0, 1, 2, ... 9

$356 = 3\times100 + 5\times10 + 6\times1$

$= \underbrace{3\times10^2 + 5\times10^1 + 6\times10^0}_{\text{weights}}$

Base or radix of decimal no. system = 10

1. Binary ⇒ 0 and 1

Base of no is 2.

3. Octal ⇒ 0, 1, 2, 3, 4, 5, 6, 7

Base of system is 8

4. Hexadecimal ⇒ 0, 1, 2, .. 10, 11, 12, 13, 14, 15.
A B C D E F

Base of system is 16.

conversion

1. of decimal base 10

Binary (repeated division by 2) | octal (repeated div by 8) | Hex. (repeated ÷ by 16)

```
2 | 356
2 | 178  - 0
2 | 89   - 0
2 | 44   - 1
2 | 22   - 0
2 | 11   - 0
2 | 5    - 1
2 | 2    - 1
    1    - 0
```

```
8 | 356
8 | 44  - 4
    5   - 4
```

$(544)_8$

```
16 | 356
16 | 22  - 4
     1   - 6
```

$(164)_{16}$

Resultant

Binary value $\Rightarrow (101100100)_2$

Octal value $\Rightarrow (544)_8$

Hex value $\Rightarrow (164)_{16}$

2. 356.87

.87

To binary :-

$0.87 \times 2 = 1.74$

$0.74 \times 2 = 1.48$

$0.48 \times 2 = 0.96$

$0.96 \times 2 = 1.92$

$0.92 \times 2 = 1.84$

$0.84 \times 2 = 1.68$

1. Take only decimal value repeatedly
2. Take the integer value from top to bottom

$356.87 \Rightarrow$ binary $\Rightarrow (101100100.11011)_2 \ldots$

To octal

.87.

$0.87 \times 8 = 6.96$ upto 5 values.

$0.96 \times 8 = 7.68$

$0.68 \times 8 = 5.44$

$\Rightarrow (544.675)_8$

To hex

.87

$0.87 \times 16 = 13.92 \Rightarrow D.92$

$0.92 \times 16 = 14.72 \Rightarrow E.72$

$0.72 \times 16 = 11.52 \Rightarrow B.52$

$0.52 \times 16 = 8.32 \Rightarrow 8.32$

$\Rightarrow (164.DEB8)_{16}$

CONVERSION OF BINARY TO DECIMAL.

$(110110)_2 \Rightarrow (\ 54\ )_{10}$ [decimal]

$0 \times 2^0 = 0$

$+$

$1 \times 2^1 = 2$

$+$

$1 \times 2^2 = 4$

$+$

$0 \times 2^3 = 0$

$+$

$1 \times 2^4 = 16$

$+$

$1 \times 2^5 = 32$

54

CONVERSION OF BINARY TO OCTAL.

1.Q) $(\underline{110}\,\underline{110})_2 = (66)_8$

MSB LSB

① start from least significant bit [last value]

(or) LSB ↓ Bit

MSB ⇒ Most significant bit

② Group it in terms of 3 from right (LSB)

OCTAL IN BINARY :-

0 – 000
1 – 001
2 – 010
3 – 011
4 – 100
5 – 101
6 – 110
7 – 111

2.Q) $(\underline{001}\,\underline{101}\,\underline{111}\,\underline{101})_2 = (1575)_8$

CONVERSION OF BINARY TO HEXA DECIMAL.

8 4 2 1 → weight of binary

Group into 4

0 – 0000
1 – 0001
2 – 0010
3 – 0011
4 – 0100
5 – 0101
6 – 0110
7 – 0111
8 – 1000
9 – 1001
A – 1010
B – 1011
C – 1100
D – 1101
E – 1110
F – 1111

19) $(\underline{0}\underline{11}\,\underline{0110})_2 = (36)_{16}$

20) $0\,(\underline{001}\,\underline{1}\,\underline{0111}\,\underline{1101})_2 = (37D)_{16}$

I. convert the following numbers to DECIMAL

1. 10110.0101
2. $(16.5)_{16}$
3. $(26.24)_8$
4. $(DADA.B)_{16}$

1. 10110.0101

$0 \times 2^0 \longrightarrow 0$

$1 \times 2^1 \longrightarrow 2$

$1 \times 2^2 \longrightarrow 4$

$0 \times 2^3 \longrightarrow 0$

$1 \times 2^4 \longrightarrow 16$

$\underline{\quad 22 \quad}$

.0101

$0/2^0 \quad 0/2^1 \quad 0/2^2 \quad 1/2^3$

$\frac{1}{2} + \frac{1}{?}$

22.3125

0.0101

0×2^{-1}

$1 \times 2^{-2} \longrightarrow \frac{1}{4}$

0×2^{-3}

$1 \times 2^{-4} = \frac{1}{16}$

$22 + 1/16 + 1/4 = (22.3125)_{10}$

2. $(16.5)_{16}$ to decimal

$6 \times 16^0 = 6$

$1 \times 16^1 = 16$

$0.5 \times 16^{-1} = 5/16$

$\Rightarrow 6 + 16 + 5/16 = (22.3125)_{10}$

3. $(26.24)_8$

$$6 \times 8^0 \rightarrow 6$$
$$2 \times 8^1 \rightarrow 16$$

0.24

$$2 \times 8^{-1} = 2/8$$
$$4 \times 8^{-2} = 4/64$$

$\Rightarrow 6 + 16 + 2/8 + 4/64$

$= 22 + 20/64 = 22 + 5/16 = (22.3125)_{10}$

4. $(DADA.B)_{16}$

D A D A . B

13×16^3, 10×16^2, 13×16^1, 10×16^0, 11×16^{-1}

$\Rightarrow 53248 + 2560 + 208 + 10 + 0.6875$

$= (56026.6875)_{10}$

6.7.18

1. 0110111110110
to decimal.

11011111 0110

$0 \times 2^0 = 0$
$1 \times 2^1 = 2$
$1 \times 2^2 = 4$
$0 \times 2^3 = 0$
$1 \times 2^4 = 16$
$1 \times 2^5 = 32$
$1 \times 2^6 = 64$
$1 \times 2^7 = 128$
$0 \times 2^8 = 0$
$1 \times 2^9 = 512$
$1 \times 2^{10} = 1024$

1782

$(1782)_{10}$.

TO hexa decimal $\rightarrow (6F6)_{16}$, To octal $(3366)_8$

$\Downarrow$

0110 1111 0110
$(6F6)_{16}$

$\Downarrow$

$(3366)_8$
011 011 110 110

BINARY CODES

ASCII $\rightarrow$ American standard code for Information Interchange.

$\boxed{8421}$ $\rightarrow$ Binary code $\Rightarrow$ 0 to 15 $\rightarrow$ hexadecimal

- ✓ 8421
- 84-2-1
- ✓ 2421
- ✓ BCD
- ✓ Gray
- ✓ Biquinary
- Excess-3

Binary coded Decimal   WEIGHTED CODES
~~Binary coded decimal~~

	• BCD (~~8421~~) 8 4 2 1	84-2-1	~~BCD~~ 2421 ~~84-2-1 ⇒ 8~~	(Non Weighted) • Excess 3 +3 to BCD	Biquinary Bi 50
0	0000	0000	0000	0011	0 01
1	0001	0111	0001	0100	1 01
2	0010	0110	0010	0101	2 01
3	0011	0101	0011	0110	3 01
4	0100	0100	0100	0111	4 01
5	0101	1011	1011	1000	5 10
6	0110	1010	1100	1001	6 10
7	0111	1001	1101	1010	7 10
8	1000	1000	1110	1011	8 10
9	1001	1111	1111	1100	9 10

• Gray code [Non weighted codes]

Reflected code (or) Unit distance code (or) cyclic

(0 to 15)

	Gray code
0	0000
1	0001
2	0011
3	0010
4	0110
5	0111
6	0101
7	0100
8	1100
9	1101
10	1111
11	1110
12	1010
13	1011
14	1001
15	1000

reflected o/p

Excess-3

0011
~~0010~~ 0100
1011
1100
1101
1110
1111

$\boxed{0}$10

1 1P

Sign magn f

+13 (0) sig

−13 (1) no

13 00

+13 00

−13 10

MSB a

Scanned by CamScanner

(Non Weighted)
- Excess 3
+3 to BCD

0011
0100
0101
0110
0111
1000
1001
1010
1011
1100

Biquinary (Weighted)

	Bi 50	quinary 43210
0	01	00001
1	01	00010
2	01	00100
3	01	01000
4	01	10000
5	10	00001
6	10	00010
7	10	00100
8	10	01000
9	10	10000

Finding of ones parity.

$\boxed{0}1011000$ → no of 1s is 3
odd parity

1 1011000 → no of 1s is 4
even parity.

Signed Numbers / Integers.

Decimal → 9's (Diminished radix), 10's (Radix)

Sign magnitude form

+13 (0), −13 (1) → signed nos

One's complement (Diminished Radix comp)

+ → same
− → take complement
1→0, 0→1

Two's complement (Radix)

− → take complement
1→0, 0→1
add +1.

```
 11110010
       +1
 --------
 11110011
```

	Sign magnitude	One's complement	Two's complement
13	00001101	00001101	00001101
+13	00001101	00001101	00001101
−13	10001101	11110010	11110010

MSB add (1)

[can be of 8 (or) 5 bit]

[In computers subtraction is done in 2's complement]

BINARY ADDITION

A	B	CY (→carry)
0	0	00
0	1	00
1	0	00
1	1	10

TRUTH TABLE

10+1 => 11

1Q)

```
   1 1  1
   1 0 1 1
   1 1 1 1
  ---------
  1 1 0 1 0
```

summation is more than the required bit it is called as **CARRY**

2Q)

```
   1
   1 1 0 1
   1 1 0 1
   1 1 0 1
  ---------
 1 0 0 1 1 1
```

```
2 | 25
2 | 12 - 1
2 |  6 - 0
2 |  3 - 0
     1 - 1
```

8 4 2 1

1 1 1 1

1-Q)

	sign Mag.	1's comp	2's comp
25	00011001		
+25	00011001	00011001	00011001
-25	10011001	~~00011010~~ 11100110 +1	11100111

BINARY SUBTRACTION

$$\begin{array}{r} \overset{0}{\cancel{1}}\,\overset{10}{\cancel{0}}\,1\,1 \\ (-)\;1\,1\,0\,1 \\ \hline 1\,1\,0\,1 \\ \hline \end{array}$$

By 2's comp method.

13 → 1101 → 1101

11 → − 1011 → 0101

$\underbrace{1\,00}\,1\,0$ → 10010

Steps

1. first no. take as it is
2. 2nd no take 2's comp
3. add both

1's comp

$$\begin{array}{r} 0100 \\ +\quad 1 \\ \hline 0101 \\ \hline \end{array}$$

2's comp

11 − 13: 1011 − 1101

$$\begin{array}{r} 1011 \\ 0011 \\ \hline 11\,1\,0 \\ \hline \end{array}$$

→ 2's

~~0010~~

$$\begin{array}{r} 0001 \\ +\quad 1 \\ \hline 0010 \\ \hline \end{array}$$

LOGIC GATES

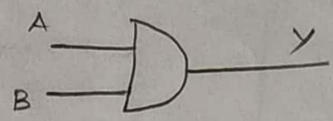
AND	$Y = A \cdot B$	Basic gates
OR	$Y = A + B$ [diff from binary addition]	
NOT	$Y = \bar{A}$	
NAND	$Y = \overline{A \cdot B}$ NOT + AND	Universal gates
NOR	$Y = \overline{A + B}$ NOT + OR	
XOR	$Y = A \oplus B \Rightarrow \bar{A}B + B\bar{A}$	Both are complementary to each other.
X NOR	$Y = A \odot B \Rightarrow AB + \bar{A}\,\bar{B}$	

$$\overline{A \oplus B} = A \odot B$$

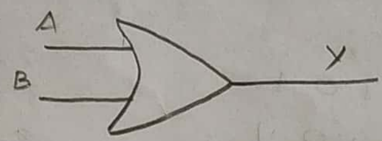
$$\overline{A \odot B} = A \oplus B$$

Symbols

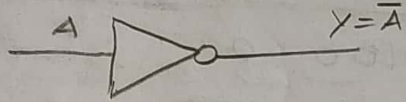
AND



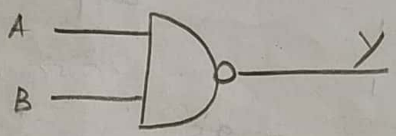
OR



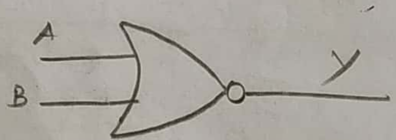
NOT



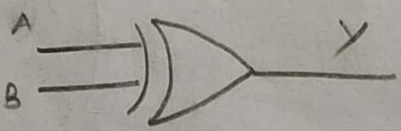
NAND



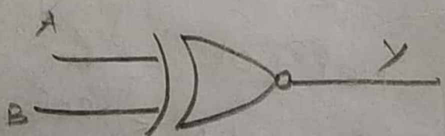
NOR



X OR



XNOR



10.7.18

Logic gates.

AND = Y = A.B

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

XOR

A	B	$Y = A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

OR

A	B	Y=A+B
0	0	0
0	1	1
1	0	1
1	1	1

XNOR

A	B	$Y = A \odot B$
0	0	1
0	1	0
1	0	0
1	1	1

Inverter (NOT)

A	~~B~~	$Y = \overline{A}$
0		1
1		0

NAND

A	B	$Y = \overline{A.B}$
0	0	1
0	1	1
1	0	1
1	1	0

NOR

A	B	$Y = \overline{A+B}$
0	0	1
0	1	0
1	0	0
1	1	0

AND laws:-

$A \cdot 0 = 0$

$A \cdot 1 = A$

$A \cdot A = A$

$A \cdot \bar{A} = 0$

OR laws:-

$A + 0 = A$

$A + 1 = 1$

$A + A = A$

$A + \bar{A} = 1$

Double Inversion law:-

$(A')'$ $\bar{\bar{A}} = A$

Commutative laws:-

$AB = BA$

$A + B = B + A$

Associative laws:-

$A(BC) = (AB)C$

$A + (B + C) = (A + B) + C$

Distributive laws:-

$A(B + C) = AB + AC$

$A + BC = (A + B)(A + C)$

Absorption law:

$A(A+B) = A$

$A + AB = A$

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

De Morgan's law:

$(A+B)' = A'B'$

$(AB)' = A'+B'$

Boolean expression.

$Y = A'B + BCD$

individuals are called as literals

Reduction of Boolean Expression.

1. Boolean laws (simple)
2. Karnaugh Map. (complicated)
3. Tabulation method. (V. complicated)

I. Use Boolean laws & reduce the expression.

1. $xy + xy'$
2. $xyz + x'y + xyz'$
3. $(a+b+c')(a'b'+c)$

Answers.

1. $xy + xy'$

$x(y+y') = x \cdot 1$
$= x$

2. $xyz + x'y + xyz'$

$= xy(z+z') + x'y$

$= xy + x'y$

$= y(x+x')$

$= y \cdot 1 = y$

3. $(a+b+c')(a'b'+c)$

$= aa'b' + a'b'b + a'b'c + ac + bc + \underbrace{c'c}_{0}$

$= 0 + 0 + a'b'c' + ac + bc$

$= a'b'c' + ac + bc$ //

4. $(BC' + A'D)(AB' + CD')$

$= \underbrace{AB'\cancel{B}C'}_{0} + \underbrace{BC'\cancel{C}D'}_{0} + \underbrace{AA'\cancel{D}B'}_{0} + \underbrace{A'D\cancel{C}D'}_{0}$

$= 0$

5. $A'B(D' + C'D) + B(A + A'CD)$

$= A'BD' + \underline{A}'BC'\underline{D} + BA + B\underline{A}'C\underline{D}$

$= \cancel{A'D(BC'+}$

$= A'BD(C' + C) + A'BD' + BA$

$= A'BD + A'BD' + BA$

$= A'B(D + D') + BA$

$= A'B + BA$

$= B(A + A')$

$= B$

6. $(A'+C)(A'+C')(A+B+C'D)$ $\quad A'(B+C'D)$

4 lil

$= (A'A' + A'C' + CA' + \cancel{CC'})(A + B + C'D)$

$= A'(A' + C' + C)(A + B + C'D)$

$= A'(\cancel{A'A} + A'B + A'C'D + C'A + C'B + C'C'D +$
$AC + CB + \cancel{CC'}D)$

$= A'(A'B + \underline{A'C'D} + \underline{C'A} + C'B + C'C'D + AC + CB)$

$= A'[B(C + C') + A(C + C') + C'C'D + A'C'D + A'B$

$= A'[B + A + C'C'D + A'C'D + A'B]$

$(A' + C'$

$= A'(A +$

$= A'(B +$

Impleme

$Y = xy$

NAND

$Y = \overline{\overline{Y}} =$

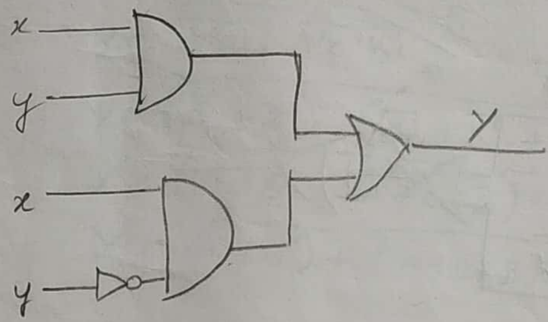
$(A'+C')(A+B+C'D)$

$= A'(A+B+C'D) = \cancel{A'A} + A'B + A'C'D$

$= A'(B+C'D)$

Implementation using Logic gates.

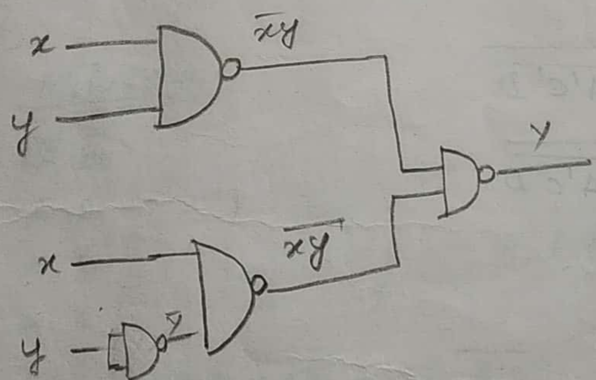
$Y = xy + xy'$



NAND

$Y = \overline{\overline{Y}} = \overline{\overline{xy + xy'}}$ NAND ⟺ NOR

$= \cancel{x} = \overline{\overline{xy}\cdot\overline{xy'}}$



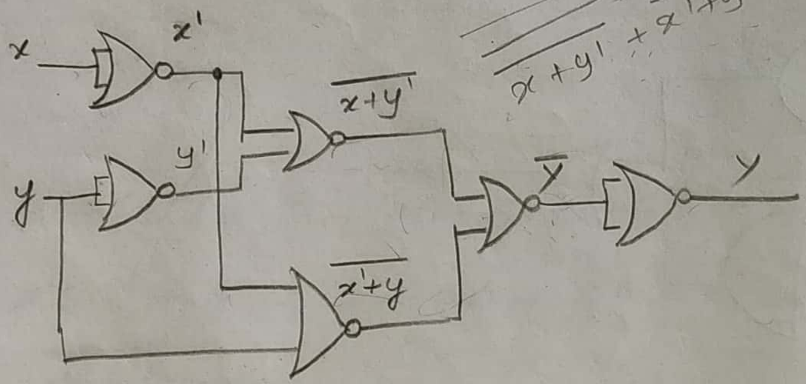
[avoid multiplication]

NOR.

$$Y = \overline{\overline{xy} \cdot \overline{xy'}}$$

$$= \overline{(x'+y')(x'+y)}$$

$$= \overline{(x'+y')} + \overline{(x'+y)}$$



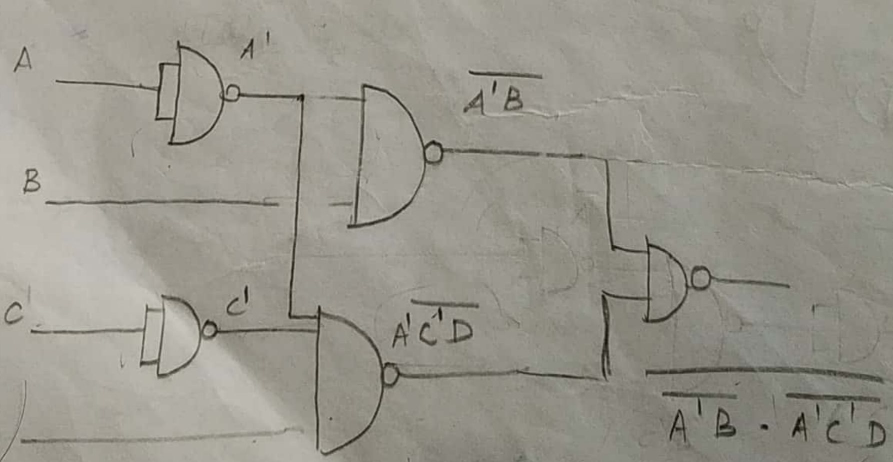
$A'(B+C'D)$ only NAND and only NOR.

⇒ only NAND

$$Y = A'B + A'C'D$$

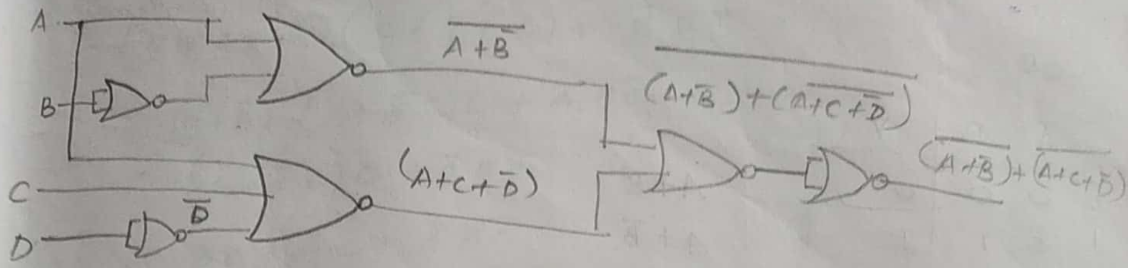
$$Y = \overline{\overline{Y}} = \overline{\overline{A'B + A'C'D}}$$

$$= \overline{\overline{A'B} \cdot \overline{A'C'D}}$$



ations)

only NOR



$Y = \overline{\overline{A'B} \cdot \overline{A'C'D}}$

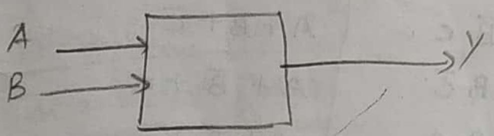
$Y = \overline{(\overline{\overline{A}} + \overline{B}) \cdot (\overline{A'} + \overline{C'} + \overline{D})}$

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

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

13.07.18

Maxterms and Minterms.



OR

A	B	Y	
0	0	0	0
0	1	1	1
1	0	1	2
1	1	1	3

Min term:

$\overline{A}\ \overline{B}$

$\overline{A}\ B$

$A\ \overline{B}$

$A\ B$

0 → complemented

→ $\overline{A}$ (or) $\overline{B}$

canonical form → A (or) B

$Y = \overline{A}B + A\overline{B} + AB$ (sum of pdts)

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

$= B + A\overline{B}$

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

$= A+B$ → standard form.

→ consider only ones.

Maxterms:

Sum of literals/variables

0 → A uncomplemented

1 → $\bar{A}$ complemented (or) $\bar{B}$

OR

	A	B	Y	
0	0	0	0	$A+B$
1	0	1	1	$A+\bar{B}$
2	1	0	1	$\bar{A}+B$
3	1	1	1	$\bar{A}+\bar{B}$

$Y = A+B$ (product of sum)

→ consider zeros.

Q)

	A	B	C	Y	Min	Max
0	0	0	0	0	$\bar{A}\bar{B}\bar{C}$	$A+B+C$
1	0	0	1	1	$\bar{A}\bar{B}C$	$A+B+\bar{C}$
2	0	1	0	1	$\bar{A}B\bar{C}$	$A+\bar{B}+C$
3	0	1	1	1	$\bar{A}BC$	$A+\bar{B}+\bar{C}$
4	1	0	0	0	$A\bar{B}\bar{C}$	$\bar{A}+B+C$
5	1	0	1	0	$A\bar{B}C$	$\bar{A}+B+\bar{C}$
6	1	1	0	0	$AB\bar{C}$	$\bar{A}+\bar{B}+C$
7	1	1	1	1	ABC	$\bar{A}+\bar{B}+\bar{C}$

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

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

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

$= \cancel{BC + \bar{A}(\bar{B}C}$

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

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

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

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

Max:

$(A+B+$

$=(B+\bar{A}$

$= (B+$

$= ($

$= B$

$=$

$=$

$=$

Max:

$(A+B$

1. canonic

2. standa

sum

produc

3.

$Y=(A$

$= A$

$=$

$=$

Scanned by CamScanner

Max:

$$\cancel{(A+B+C)(\bar{A}+B+C)(\bar{A}+\bar{B}+\bar{C})(\bar{A}+\bar{B}+C)}$$

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

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

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

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

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

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

$$\boxed{=BC+\bar{A}B+\bar{A}C.} \rightarrow \text{std form}$$

sum of pdt

Max:

$$\boxed{(A+B+C)(\bar{A}+B+C)(\bar{A}+B+\bar{C})(\bar{A}+\bar{B}+C)} \rightarrow \text{canonical}$$

[pdt of max terms]

1. canonical form:
   - → sum of minterms. [all the variables are present]
   - → Product of max terms.
2. standard form
   - sum of products
   - product of sums.

Q.

$$y=(A+A\bar{B}) \rightarrow \text{sum of pdt [std form]}$$

missing terms B means take $\rightarrow B+\bar{B}$

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

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

$$=AB+A\bar{B} \rightarrow \text{sum of min terms}$$

[canonical form]

$$=\Sigma(2,3)$$

$(A+B)(A+\bar{B})$

2. $y = BC + \bar{A}B + \bar{A}C \rightarrow$ sum of pdt

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

$= BCA + \underbrace{BC\bar{A} + \bar{A}BC} + \underline{\bar{A}B\bar{C}} + \bar{A}CB + \underline{\bar{A}C\bar{B}}$

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

$= \Sigma m(1,2,3,7)$

$\Pi m = [(A+B+C)(\bar{A}+B+C)(\bar{A}+B+\bar{C})(\bar{A}+C+\bar{B})]$

4 5 6

3. $y = (\bar{A}+\bar{B})(A+B+\bar{C})(\bar{B}+C) \rightarrow$ pdt of sum

missing variable $\rightarrow C\bar{C} \Rightarrow 0$

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

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

6 7 1 2 6

$= \Pi M(1,2,6,7)$

Q.4. $y = (c'+d)(b+c')$

5. $y = (b+cd)(c+bd)$

4. $y = (c'+d)(b+c')$

$= \cancel{b\bar{b}+c}$

$= (\bar{c}+d+b\bar{b})(b+\bar{c}+d\bar{d})$

$= (\bar{c}+d+b)(\bar{c}+d+\bar{b})(b+\bar{c}+d)(b+\bar{c}+\bar{d})$

$= \underline{(b+\bar{c}+d)}(\bar{b}+\bar{c}+d)\underline{(b+\bar{c}+d)}(b+\bar{c}+\bar{d})$

2 6 3

$= \Pi m(2,3,6)$

$= \Sigma m = (0,1,4,5,7)$

5. $y = (b+c$

$y = (\cancel{bc}$

$= (b+c)$

$= (b+c$

$= (b+c+\bar{c}$

$(\bar{b}+$

$= \pi($

$= \Sigma(3$

6. convert

6. convert

i) $f(x,y,z)$

(ii) $F(A,$

5. $y = (b+cd)(c+bd)$

$y = \cancel{\left(b(c+\bar{c})(d+\bar{d}) + cd(b+\bar{b})\right)}.$

$\cancel{\left((b+\bar{b})c(d+\bar{d}) + (c+\bar{c})bd\right)}$

$= (b+c)(b+d)(c+b)(c+d)$

$= (b+c+d\bar{d})(b+d+c\bar{c})(c+b+d\bar{d})(c+d+b\bar{b})$

$= \underline{(b+c+d)}(b+c+\bar{d})\cancel{(b+c+d)}(b+\bar{c}+d)$

$\underline{(b+c+d)}\,\underline{(b+c+\bar{d})}\,\cancel{(b+c+d)}(\bar{b}+c+d)$

$= \pi(0,1,2,4)$

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

6. ~~convert~~

6. convert to other canonical form. 0 – 8

(i) $f(x,y,z) = \Sigma(1,3,5) \Rightarrow \pi(0,2,4,6,7)$

$f(xyz) = (\bar{x}\bar{y}z + \bar{x}yz + x\bar{y}z)$

1 3 5

(ii) $F(A,B,C,D) = \pi(3,5,8,11)$

$= \Sigma(0,1,2,4,6,7,9,10,12,13,14,15)$

$\pi \Leftrightarrow \Sigma$

complemental to each other

~~F(A,B,C,D) = Π (3,5,8,11)~~

Karnaugh Map (K-Map)

Variables–

A → MSB
B → LSB.

1) 2

A \ B	$\bar{B}$ (0)	B (1)
(0) $\bar{A}$	$\bar{A}\bar{B}$ 0	$\bar{A}B$ 1
(1) A	$A\bar{B}$ 2	AB 3

0 0
0 1
1 0
1 1

⇒ 2 variable K map

Q)

Truth table:

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

A \ B	0	1
0		1
1	1	1

min term ②

①

STEPS:

[consider adjacent one's]

⇒ only in vertical (or) horizontal way

⇒ consider in powers of 2
(ie) 1, $\underbrace{2, 4, 8, 16}$ ....

⇒ group the higher no of adjacent ones.

① → A
② → B

Y = A+B

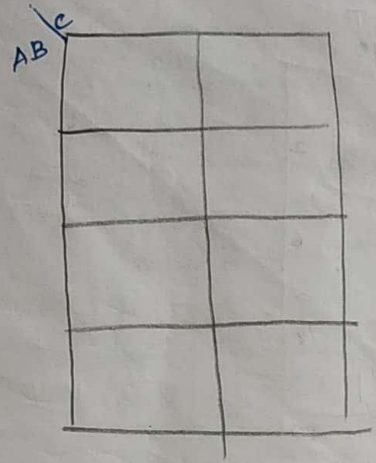
1			1
1		1	1

due to folding & combination are possible

$F = \bar{C} + AB$

3 Variable form.

[adjacent box → 1 bit variation]



(or)

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

Q)

	A	B	C	Y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	1

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

$$\boxed{Y = \bar{A}C + \bar{A}B + BC}$$

1. $F(x,y,z) = \Sigma(3,4,5,7)$

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

$Y = BC + \overset{AC}{\cancel{AC}} + A\bar{B} + \cancel{AC}$

including or excluding gives the same value

5	7

↓ redundant term

[not necessary]

4 Variable Map — 16 square boxes.

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

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

17.07.18

Reduce the boolean expression using k-map.

1. $F(A,B,C,D) = \Sigma(0,1,3,5,7,8,9,11,13,14,15)$

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

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

boxes.

2. $F(A,B,C,D) = \Pi(5,6,7,9,10,11,13,15)$

$= \Sigma(0,1,2,3,4,8,12,14)$

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

0000	0
0001	1
0010	2
0011	3
0100	4
0101	5
0110	6
0111	7
1000	8
1001	9
1010	10
1011	11
1100	12
1101	13
1110	14
1111	15

$\cancel{= \bar{A}\bar{B} + \bar{C}\bar{D} + AB\bar{D}}$

$\cancel{= BD + AD + A\bar{B}C + \bar{A}BC}$

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

3. $F(w,x,y,z) = w\bar{x}y + \bar{x}\bar{y}\bar{z} + y\bar{z} + \bar{w}\bar{z}$

wx \ yz	$\bar{y}\bar{z}$ 00	$\bar{y}z$ 01	yz 11	$y\bar{z}$ 10
$\bar{w}\bar{x}$ 00	1 (0)	(1)	(3)	1 (2)
$\bar{w}x$ 01	1 (4)	(5)	(7)	1 (6)
wx 11	(12)	(13)	(15)	1 (14)
$w\bar{x}$ 10	1 (8)	(9)	1 (11)	1 (10)

$\cancel{y = \bar{w}}$

$y = \cancel{y\bar{z} + \bar{w}\bar{z}} + w\bar{x}y$

$y = y\bar{z} + w\bar{x}y + \bar{w}\bar{z} + \bar{x}\bar{z}$

$= w\bar{x}y(z+\bar{z}) + \bar{x}\bar{y}\bar{z}(w+\bar{w}) + (x+\bar{x})(w+\bar{w})y\bar{z} + \bar{w}\bar{z}(x+\bar{x})(y+\bar{y})$

$= w\bar{x}yz + w\bar{x}y\bar{z} + \bar{x}\bar{y}\bar{z}w + \bar{x}\bar{y}\bar{z}\bar{w} + (xw + x\bar{w} + \bar{x}w + \bar{x}\bar{w})y\bar{z} + \bar{w}\bar{z}(xy + x\bar{y} + \bar{x}y + \bar{x}\bar{y})$

$= w\bar{x}yz + w\bar{x}y\bar{z} + \bar{x}\bar{y}\bar{z}w + \bar{x}\bar{y}\bar{z}\bar{w} + xwy\bar{z} + x\bar{w}y\bar{z} + \bar{x}wy\bar{z} + \bar{x}\bar{w}y\bar{z} + \bar{w}\bar{z}xy + \bar{w}\bar{z}x\bar{y} + \bar{w}\bar{z}\bar{x}y + \bar{w}\bar{z}\bar{x}\bar{y}$

11, 5, 8, 0, 14, 6, 10, 2, 6, 4, 2, 0

(0, 2, 4, 6, 8, 10, 11, 14)

Scanned by CamScanner

4. $F = \Sigma(1,2,3,5,6,7,9,11,12,13,15)$

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	1 (5)	1 (7)	1 (6)
AB 11	1 (12)	1 (13)	1 (15)	14
$A\bar{B}$ 10	8	1 (9)	1 (11)	10

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

20/08/18.

K-map with dont care. "X".

1. $F(\quad) = \Sigma m(2,5,7,9,11,12,14) + \Sigma d(3,4,6)$

dont care 'X'

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

$X \rightarrow$ may be 1 (or) 0

$F = \bar{A}B + \bar{A}C + B\bar{D} + A\bar{B}D$

2.Q)

$F = \Sigma \bar{x}\, y\, z$ ...

3Q. F(ABC...

D(ABC...

2Q). $F = \bar{x}$ ...

3Q)

$Y = A$ ...

2.Q)

$F = \bar{x}yz + x\bar{y}\bar{z} + x\bar{y}z$

3Q. $F(ABCD) = \Sigma(0,6,8,13,14)$;
$D(ABCD) = \Sigma(2,4,10)$

2Q). $F = \bar{x}yz + x\bar{y}\bar{z} + x\bar{y}z$

$x \backslash yz$	$\bar{y}\bar{z}$ 00	$\bar{y}z$ 01	yz 11	$y\bar{z}$ 10
$\bar{x}$ 0	(0)	(1)	1 (3)	(2)
x 1	1 (4)	1 (5)	(7)	(6)

$Y = x\bar{y} + \bar{x}yz$

3Q)

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

No need to map all x.

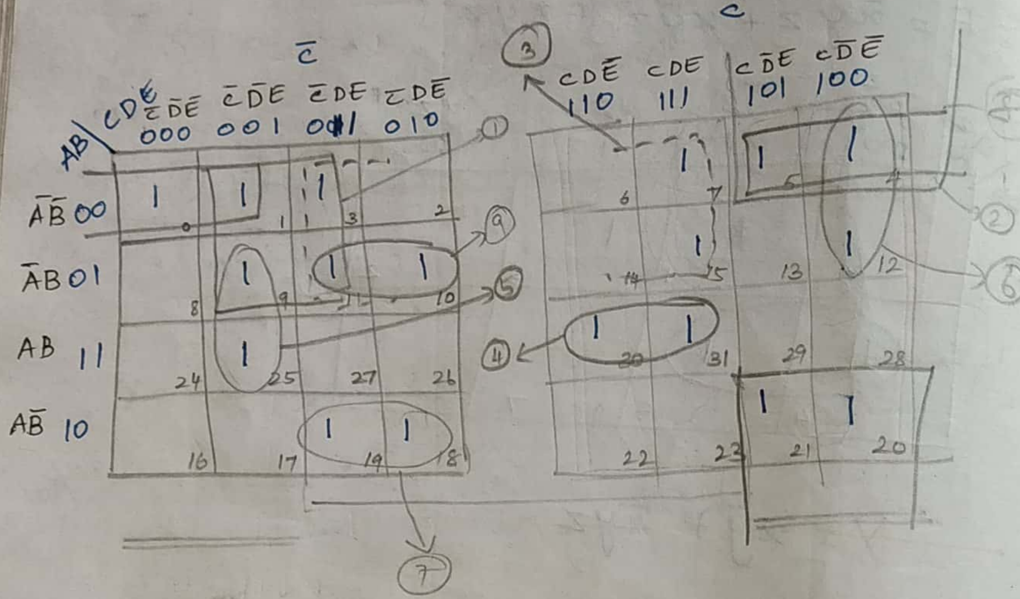
~~$Y = \bar{A}\bar{C}\bar{D} + A.B\bar{C}D + \bar{B}\bar{C}\bar{D}\ \bar{B}\bar{D} + \bar{A}\bar{D} + C\bar{D}$~~

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

Scanned by CamScanner

5 VARIABLE MAP. (0-31)

Q. $F = \Sigma(0,1,3,4,5,7,9,10,11,12,15,18,19,20,21,25,30,31)$



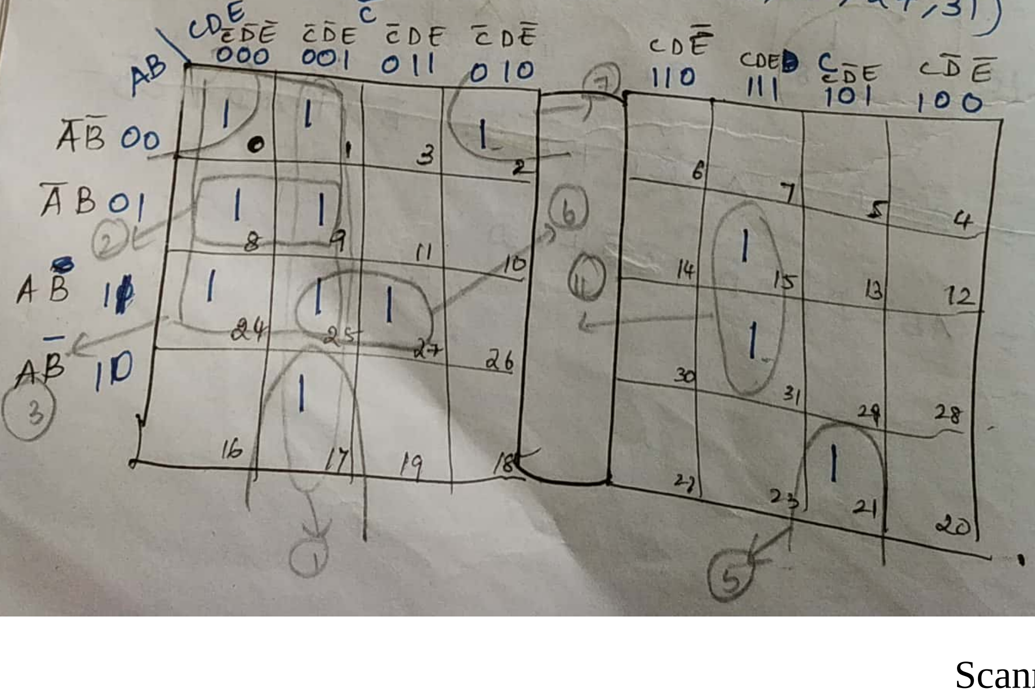
$$F = \bar{A}\bar{C}E + \bar{A}\bar{B}\bar{D} + \bar{A}DE + ABCD + B\bar{C}\bar{D}E + \bar{A}C\bar{D}\bar{E} + A\bar{B}\bar{C}D + \bar{B}C\bar{D} + \bar{A}B\bar{C}D$$

TABULATION METHOD

(Quine McKluskey Method)

5 Variable

Q). $F = \Sigma(0,1,2,8,9,15,17,21,24,25,27,31)$



30, 31)

$F = \cancel{\bar{C}\bar{D}E} + \bar{A}\bar{C}\bar{D} + B\bar{C}\bar{D} + BCDE + A\bar{B}\bar{D}E + AB\bar{C}E + \bar{A}\bar{B}\bar{C}\bar{E}$

TABULATION METHOD

(Quine McKluskey Method)

[consider the highest number
→ convert into binary
→ group with no of one
(—) → dont care symbol

Q.

Total 6 groups

	A	B	C	D	E	
0	0	0	0	0	0	✓
1	0	0	0	0	1	✓
2	0	0	0	1	0	✓
8	0	1	0	0	0	✓
9	0	1	0	0	1	✓
17	1	0	0	0	1	✓
24	1	1	0	0	0	✓
21	1	0	1	0	1	
25	1	1	0	0	1	✓
15	0	1	1	1	1	✓
27	1	1	0	1	1	✓
31	1	1	1	1	1	✓

Iteration -1

(0,1) 0000 –
(0,2) 000 – 0
(0,8) 0 – 000
(1,9) 0 – 001
(1,17) – 0001
(8,9) 0100 –
(8,24) – 1000
(9,25) – 1001
(17,21) 10 – 01
(17,25) 1 – 001
(24,25) 1100 –
(25,27) 110 – 1
(15,31) – 1111
(27,31) 11 – 11

Iteration -2

Iteration - 1.

(0,1)	0000 -
(0, 2)	000 - 0
(0, 8)	0 - 000
(1, 9)	0 - 001
(1, 17)	- 0001
(8, 9)	0100 -
(8, 24)	- 1000
(9, 25)	- 1001
(17, 21)	10 - 01
(17, 25)	1 - 001
(24, 25)	1100 -
(25, 27)	110 - 1
(15, 31)	- 1111
(27, 31)	11 - 11

A B C D E

Iteration 2

(0, 1, 8, 9)	0 - 00 -
~~(0, 8, 1, 9)~~	~~0 - 00 -~~
(1, 9, 17, 25)	- - 001
~~(1, 17, 9, 25)~~	~~- - 001~~
(8, 9, 24, 25)	- 100 -
~~(8, 24, 9, 25)~~	~~- 100 -~~

Prime Implicants)

(0, 2)	$\bar{A}\bar{B}\bar{C}\bar{E}$
(17, 21)	$A\bar{B}\bar{D}E$
(25, 27)	$A\bar{B}\bar{C}E$
(15, 31)	$BCDE$
(27, 31)	$ABDE$
(0, 1, 8, 9)	$\bar{A}\bar{C}\bar{D}$
(1, 9, 17, 25)	$\bar{C}\bar{D}E$
(8, 9, 24, 25)	$B\bar{C}\bar{D}$

steps

1. single entry from each column

P. Implicants	Minterm	0 ✓	1 ?	2 ✓	8 ✓	9 ✓	15 ✓	17 ✓	21 ✓	24 ✓	25 ✓	27 ?	31 ✓
$\bar{A}\bar{B}\bar{C}\bar{E}$	(0,2)	X	*	X									
$A\bar{B}\bar{D}E$	(17,21)							X	X				
$A\bar{B}\bar{C}E$	(25,27)										X	X	
$BCDE$	(15,31)						X						X
$ABDE$	(27,31)											X	X
$\bar{A}\bar{C}\bar{D}$	(0,1,8,9)	X	X		X	X							
$\bar{C}\bar{D}E$	(1,9,17,25)		X			X		X			X		
$B\bar{C}\bar{D}$	(8,9,24,25)				X	X				X	X		

<u>steps</u>

1. single entry from each column should be selected as essential P.I
2. They have to be marked & put a tick mark
3. their min terms are selected and ticked [on the top]
4. The left over values are denoted as (?) [ie] 1, 27

$$P = \bar{A}\bar{B}\bar{C}\bar{E} + A\bar{B}\bar{D}E + BCDE + A\bar{B}\bar{C}E + \bar{A}\bar{C}\bar{D} + B\bar{C}\bar{D}$$

2. $F = \Sigma(1, 3, 4, 5, 10, 11, 12, 13, 14, 15)$

	A	B	C	D
1	0	0	0	1
3	0	0	1	1
4	0	1	0	0
5	0	1	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

[separate according to no of 1's]

	A	B	C	D
✓1	0	0	0	1
✓4	0	1	0	0
✓3	0	0	1	1
✓5	0	1	0	1
✓10	1	0	1	0
✓12	1	1	0	0
✓11	1	0	1	1
✓13	1	1	0	1
✓14	1	1	1	0
✓15	1	1	1	1

Iteration-I (A B C D)

	ABCD	
(1,3)	00–1	
(1,5)	0–01	
(4,5)	010–	✓
(4,12)	–100	✓
(3,11)	–011	
(5,13)	–101	✓
(10,11)	101–	✓
(10,14)	1–10	✓
(12,13)	110–	✓
(12,14)	11–0	✓
(11,15)	1–11	✓
(13,15)	11–1	✓
(14,15)	111–	✓

Iteration-2 (A B C D)

	ABCD
(4,5,12,13)	–10–
(4,12,5,13)	–10–
(10,11,14,15)	1–1–
(10,14,11,15)	1–1–
(12,13,14,15)	11––
(12,14,13,15)	11––

PRIME impli

(4,5,12,1

(10,11,14,

(12,13,14

(1,3) –

(1,5) –

(3,11)

P.I	Mint
$\bar{A}\bar{B}D$	(1
$\bar{A}\bar{C}D$	(1
$\bar{B}CD$	(
$B\bar{C}$	(
AC	(
AB	(

PRIME implicants.-

$(4,5,12,13) \rightarrow B\bar{C}$

$(10,11,14,15) \rightarrow AC$

$(12,13,14,15) \rightarrow AB$

$(1,3) \longrightarrow \bar{A}\bar{B}D$

$(1,5) \longrightarrow \bar{A}\bar{C}D$

$(3,11) \longrightarrow \bar{B}CD$

1
3
4
5
10
11
12
13
14
15

P.I	Min term.	1 ?	3 ?	4 ✓	5 ✓	10 ✓	11 ✓	12 ✓	13 ✓	14 ✓	15 ✓
$\bar{A}\bar{B}D$	(1, 3)	x	x								
$\bar{A}\bar{C}D$	(1, 5)	x			x						
$\bar{B}CD$	(3, 11)		x				x				
$\boxed{B\bar{C}}$	(4, 5, 12, 13)			x	x			x	x		
$\boxed{AC}$	(10, 11, 14, 15)					x	x			x	x
AB	(12, 13, 14, 15)							x	x	x	x

$\Rightarrow \bar{A}\bar{B}D + B\bar{C} + AC$

Scanned by CamScanner

27.7.18 HW

Q. $F = (A,B,C,D) = \Sigma m\,(0,6,8,13,14)$

$d\,(A,B,C,D) = \Sigma d\,(2,4,10)$

Follow the same procedure for both Σ

Exclude don't care in PI table.

		Iteration - I		Iteration - 2	
0	0000	(0,2)	00-0 ✓	(0,2,4,6)	0--0
2	0010	(0,4)	0-00 ✓	(0,2,8,10)	-0-0
4	0100	(0,8)	-000 ✓	~~(0,4,2,6)~~	~~0--0~~
8	1000	(2,6)	0-10 ✓		
6	0110	(2,10)	-010 ✓	~~(0,8,2,10)~~	~~-0-0~~
10	1010	(4,6)	01-0 ✓	(2,6,10,14)	--10
13	1101	(8,10)	10-0 ✓	~~(2,10,6,14)~~	~~--10~~
14	1110	(6,14)	-110 ✓		
		(10,14)	1-10 ✓		

(Neglect the don't care terms in the table)

P.I

$(0,2,4,6) \longrightarrow \bar{A}\bar{D}$

$(0,2,8,10) \longrightarrow \bar{B}\bar{D}$

$(2,6,10,14) \longrightarrow C\bar{D}$

0, ~~2~~, ~~4~~, 6, 8, ~~10~~, 14

P.I	Minterms	0 ✓	6 ✓	8 ✓	14 ✓	13 ✓
$\bar{A}\bar{D}$	(0, 2, 4, 6)	X	X			
$\bar{B}\bar{D}$	(0, 2, 8, 10)	X		X		
$C\bar{D}$	(2, 6, 10, 14)		X		X	
$AB\bar{C}D$	13					X

$AB\bar{C}D + \bar{A}\bar{D} + \bar{B}\bar{D} + C\bar{D}$

7.7.18

Tutorial. on Unit-1.

I.

1. Do the following conversion:

A. convert $(27.315)_{10}$ to binary

B. $(C3DF)_{16}$ to binary

C. $(26.24)_8$ to decimal.

D. $(DADA.B)_{16}$ to decimal.

2. Obtain 1's and 2's complement of the number.

A. 11011010

B. 1010.1101

3. Perform subtraction using 2's complement of the subtrahend

A. 1001 − 110101

B. 101000 − 10101

4. ~~Convert~~ Perform the binary equivalent of 49 and 29 (of base 10) using signed 2's complement representation. Then perform

(i) $(-29) + (+49)$

(ii) $(-29) + (-49)$

convert the answer back to decimal and verify the result.

5. convert the binary 110111О to gray code.

6. Reduce the boolean expression using boolean law.

A. $(A'+C)(A'+C')(A+B+C'D)$

For the reduced eqn draw circuit using universal gates.

Scanned by CamScanner

7. Express the function $(cd + b'c + bd')(b+d)$ sum of min terms & pdt of max terms (canonical form)

8. ~~Express~~ convert A. $F(x,y,z) = \Sigma(1,3,5)$ to other canonical form.

B. $F(A,B,C,D) = \pi(3,5,8,11)$

9. Convert :-

A. $(u + xw)(x + u'v)$ into sum of pdt and pdt of sum (standard form)

10. Reduce using k-map

A. $F(w,x,y,z) = \Sigma(2,3,12,13,14,15)$

B. $F = w'z + xz + x'y + wx'z$

C.

11. Using Tabulation method find the reduced expression

A. $F = \Sigma(1,3,4,5,10,11,12,13,14,15)$

B. $F = \Sigma(5,6,7,12,14,15)$

$d = \Sigma(3,9,11,15)$

Verify the results using k-map.

ANSWERS. for Tutorial on Unit-1

1\. A. $(27.315)_{10} \rightarrow (11011.01010)_2$

2	27
2	13 − 1
2	6 − 1
2	3 − 0
	1 − 1

$.315 \times 2 = 0.63$

$.63 \times 2 = 1.26$

$.26 \times 2 = 0.52$

$.52 \times 2 = 1.04$

$.04 \times 2 = 0.08$

$(27.315)_{10} \Rightarrow (11011.01010)_2$

B. $(C3DF)_{16}$ to binary.

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

~~12 3 13 15~~

~~$15 \times 16^0 = 15$~~

~~$13 \times 16^1 = 240$~~

~~$3\,15 \times 16^2 = 3840$~~

~~$10 \times 16^3 =$~~

12 3 13 15

$15 \times 16^0 = 15$

$13 \times 16^1 = 208$

$3 \times 16^2 = 768$

$12 \times 16^3 = 49,152$

$(50143)_{10}$

(or) $(1100\ 0011\ 1101\ 1111)_2$

C. $(26.24)_8$ to decimal:

26

$6 \times 8^0 \Rightarrow 6$

$2 \times 8^1 \Rightarrow 16$

22

.24

$4 \times 8^{-1} = 0.5$

$2 \times 8^{-2} = 0.03125$

0.53125

$(26.24)_8 = (22.53125)_{10}$

D. $(DADA.B)_{16}$ to decimal:

D A D A . B

$10 \times 16^0 = 10$

11×16^{-1}

0.6875

$13 \times 16^1 = 208$

$10 \times 16^2 = 2560$

$13 \times 16^3 = 53,248$

56,027.

$(DADA.B)_{16} = (56027.6875)_{10}$

9-53

1's & 2's com

A. 1101101

1's →

2's →

B. 1010.11

1's →

2's

3. Subtrac

A. 1001

0

(+) 0

0

B. 10100

(+) 0

1

2. 1's & 2's complement.

A. 11011010

1's → 00100101

2's → + 1

00100110

B. 1010.1101

1's → 0101.0010

+ 1 + 1

2's

0110 .0010

0101.0010

+ 1

0101.0011

3. Subtraction

Answer

A. 1001 − 0110101 ⇒ 2's → 1001011

9 53

(+) 0001001

1010100

9 − 53 = −44

001001

(+) 0010001

010100

0001001

1001011

1010100

2's 0101011

+ 1

0101100

32+8+4 = 44

B. 101000 − 010101 ⇒ 2's → 101010

+ 1

101011

10100

(+) 01011

11111

101000

(+) 101011

100011

101000

− 10101

010011

(without 2's complement)

Note:

1's complement

$$\boxed{r^n - r^{-m} - N}$$

For 2's complement

$$\boxed{r^n - N}$$

Using 2's complement:

```
       101000
 2's → 101011
      ---------
      1010011
```

4. Binary equivalence of 49 & 29.

```
2 | 49              2 | 29
2 | 24 - 1          2 | 14 - 1
2 | 12 - 0          2 |  7 - 0
2 |  6 - 0          2 |  3 - 1
2 |  3 - 0              1 - 1
     1 - 1
```

$(49)_{10} \rightarrow (110001)_2$ $\quad$ $(29)_{10} \rightarrow (11101)_2$

Sign Magnitude Form.

	Sign Mag.	1's	2's
49	00110001		
+49	00110001	00110001	00110001
-49	10110001	11001110	11001111
29	00011101		
+29	00011101	00011101	00011101
-29	10011101	11100010	11100011

(i) (−29) + (+49)

$$\begin{array}{rr} -29 \rightarrow & 11100011 \\ +49 \rightarrow & 00110001 \\ \hline & 100010100 \end{array} \quad (+)$$

$$\begin{array}{r} 49 \\ -29 \\ \hline 20 \end{array}$$

(ii) (−29) + (−49)

$$\begin{array}{rr} -29 \rightarrow & 11100011 \\ -49 \rightarrow & 11001111 \\ \hline & 110110010 \end{array}$$

Verification

110110010

1's 00100 1101

+1

2's 01001110

(−29) + (+49) ⇒ 20 [To verify (i)]

00010100

$0 \times 2^0 = 0$

$0 \times 2^1 = 0$

$1 \times 2^2 = 4$

$0 \times 2^3 = 0$

$1 \times 2^4 = 16$

$\boxed{20 \longrightarrow \text{(i) is Verified}}$

(−29) + (−49) ⇒ 01001110

0×2^0

$1 \times 2^1 \Rightarrow 2$

$2^2 \Rightarrow 4$

$2^3 \Rightarrow 8$

0×2^4

0×2^5

$64 \Leftarrow 1 \times 2^6$

$$\begin{array}{r} 29 \\ +49 \\ \hline 78 \end{array}$$

$\boxed{2+4+8+64 = 78 \longrightarrow \text{(ii) is verified.}}$

Note: (2m)

BINARY TO GRAY

1 1 1 0

1 0 0 1

→ 1st no as it is

→ next nos take X-OR

GRAY TO BINARY

1 1 1 0

1 0 1 1

→ next ~~no~~ ans value take X-OR.

5. Convert binary 1101110 ⇒ gray code

1 1 0 1 1 1 0

↓

1 0 1 1 0 0 1 ⇒ gray code.

6. Reduce the boolean expression using boolean law.

A. $(A'+C)(A'+C')(A+B+C'D)$

Using distribution law

$(A'+C)(A'+C')$ ⇓ $(A'+CC')$

$(A'+\cancel{CC'})(A+B+C'D)$

$= A'(A+B+C'D)$

$= \cancel{A'A}+A'B+A'C'D$

$= A'B+A'C'D$

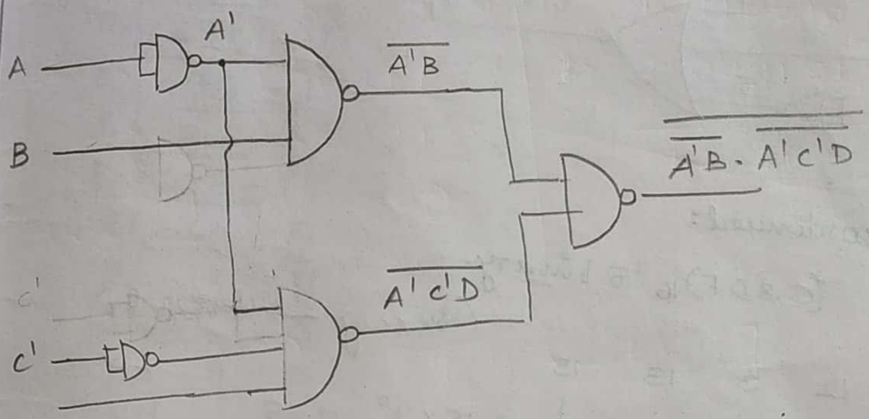
$= A'(B+C'D)$

implementation using universal logic gates:

ONLY USING $\boxed{\text{NAND}}$ AND $\boxed{\text{NOR}}$

$$Y = A'B + A'C'D$$

$$\overline{\overline{Y}} = \overline{\overline{A'B} \cdot \overline{A'C'D}} \Rightarrow \text{only using NAND}$$



$\Rightarrow$ only using NOR

$$Y = A'B + A'C'D$$

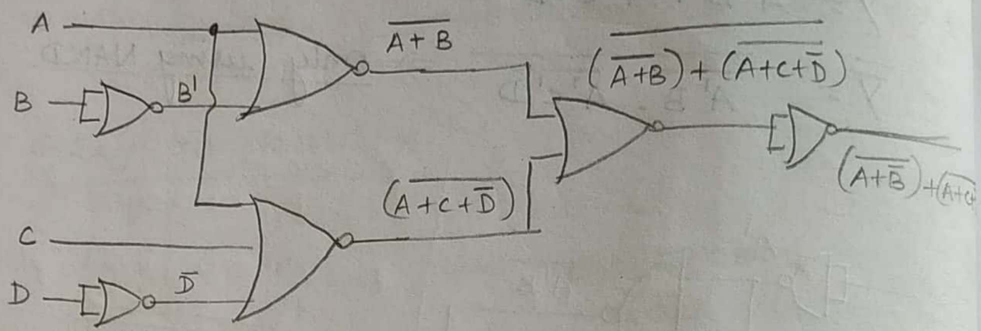
$$Y = \overline{\overline{A'B} \cdot \overline{A'C'D}}$$

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

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

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

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



1. (B) continued:

$(C3DF)_{16}$ to binary.

(hexa → decimal)

12 3 13 15

$15 \times 16^0 = 15$

$13 \times 16 = 208$

$3 \times 16^1 = 768$

$12 \times 16^2 = 49,152$

$(50143)_{10}$

(decimal → binary)

- Repeated division by 2

$2 \underline{| 50143}$

Ans ⇒ $(1100\ 0011\ 1101\ 1111)_2$

Scanned by CamScanner

decimal to binary

$$
\begin{array}{r|l}
2 & 50143 \\
2 & 25071 - 1 \\
2 & 12535 - 1 \\
2 & 6267 - 1 \\
2 & 3133 - 1 \\
2 & 1566 - 1 \\
2 & 783 - 0 \\
2 & 391 - 1 \\
2 & 195 - 1 \\
2 & 97 - 1 \\
2 & 48 - 1 \\
2 & 24 - 0 \\
2 & 12 - 0 \\
2 & 6 - 0 \\
2 & 3 - 0 \\
 & 1 - 1
\end{array}
$$

$\Rightarrow (1100001111011111)_2$

7. Express the function $(cd + b'c + bd')(b+d)$ sum of min terms & pdt of max terms (CANONICAL FORM)

$\Rightarrow (cd + b'c + bd')(b+d)$

Sum of min terms :-

$F = bcd + \cancel{bb'c}^{\,0} + bd' + cd + b'cd + \cancel{bdd'}^{\,0}$

$= bcd + b(c+c')d' + (b+b')cd + b'cd$

$= bcd + bcd' + bc'd' + \cancel{bcd} + b'cd + \cancel{b'cd}$

$\underset{7}{111} \quad \underset{6}{110} \quad \underset{4}{100} \quad \underset{3}{011}$

$\Sigma m(3, 4, 6, 7)$

Pdt of max terms:-

$= \underline{bcd} + bcd' + bc'd' + \underline{b'cd}$

$= cd(b+b') + bd'(c+c')$

$= cd + bd'$

$\Rightarrow bd' + cd$ By distributive law.

$= (bd'+c)(bd'+d)$

$= (b+c)(d'+c)(b+d)\cancel{(d+d')}$

$= (b+c+dd')(bb'+c+d')(b+cc'+d)$

$= (b+c+d)\cancel{(b+c+d')}$

$(b+c+d')(b'+c+d')$

$\cancel{(b+c+d)}(b+c'+d)$

$= (b+c+d)(b+c+d')(b'+c+d')(b+c'+d)$

000 001 101 010

0 1 5 2

$= \Pi m(0,1,2,5)$

8. Convert

A) $F(x,y,z) = \Sigma(1,3,5)$

$= \Pi(0,2,4,6,7)$

B) $F(A,B,C,D) = \Pi(3,5,8,11)$

$= \Sigma(0,1,2,4,6,7,9,10,12,13,14,15)$

$= \Sigma(0,1,2,4,6,7,9,A,C,D,E,F)$

9. conv

A. (u

pd

sur

=

=

Pdt

$\Rightarrow$

$\Rightarrow$

$\Rightarrow$

10. Re

A. F

$\bar{w}\bar{x}$

$\bar{w}x$

wx

$w\bar{x}$

Scanned by CamScanner

9. convert:

A. $(u+xw)(x+u'v)$ into sum of pdt and pdt of sum [standard form]

sum of pdt:

$(u+xw)(x+u'v)$

$= \cancel{ux + x'}\; ux + \cancel{uu'v} + xwx + xwu'v$

$= ux + xwu'v + xw \Leftarrow$ sum of pdt.

Pdt of sum:

$(u+xw)(x+u'v)$ Using distribution law,

$(u+xwu'v)(x+xwu'v)$

$\Rightarrow \cancel{(u+x)(u+w)(u+u')(u+v)(x+x)(x+w)(x+u')(x+v)}$

$\Rightarrow (u+x)(u+w)(u+v)(x+w)(x+u')(x+v)(x+x)$

$\Downarrow$

$\Rightarrow (u+x)(u+w)(x+u')(x+V)$ Pdt of sum.

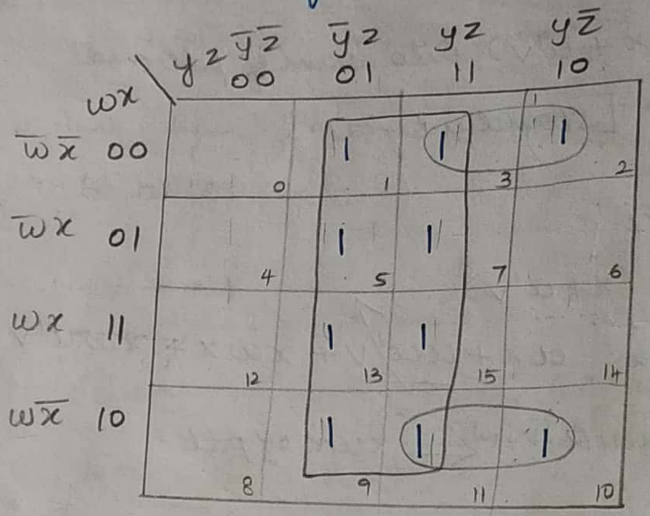
10. Reduced using K-map.

A. $F(w,x,y,z) = \Sigma(2,3,12,13,14,15)$

wx \ yz	00 ($\bar{y}\bar{z}$)	01 ($\bar{y}z$)	11 (yz)	10 ($y\bar{z}$)
$\bar{w}\bar{x}$ 00	0	1	1 (3)	1 (2)
$\bar{w}x$ 01	4	5	7	6
wx 11	1 (12)	1 (13)	1 (15)	1 (14)
$w\bar{x}$ 10	8	9	11	10

$F(w,x,y,z) = wx + \bar{w}\bar{x}y$.

B. $F = w'z + xz + x'y + wx'z$



$F(w,x,y,z) = z + x'y$

11. Using tabulation method find the reduced expression:-

A. $F = \Sigma(1,3,4,5,10,11,12,13,14,15)$

	A	B	C	D	
1	0	0	0	1	0
3	0	0	1	1	2
4	0	1	0	0	0
5	0	1	0	1	2
10	1	0	1	0	2
11	1	0	1	1	3
12	1	1	0	0	2
13	1	1	0	1	3
14	1	1	1	0	3
15	1	1	1	1	4

	A	B	C	D
1	0	0	0	1 ✓
4	0	1	0	0 ✓
3	0	0	1	1 ✓
5	0	1	0	1 ✓
10	1	0	1	0 ✓
12	1	1	0	0 ✓
11	1	0	1	1 ✓
13	1	1	0	1 ✓
14	1	1	1	0 ✓
15	1	1	1	1 ✓

Iteration-I

(1,3) 00–1
(1,5) 0–01
(4,5) 010– ✓
(4,12) –100 ✓
(3,11) –011
(5,13) –101 ✓
(10,11) 101– ✓
(10,14) 1–10 ✓
(12,13) 110– ✓
(12,14) 11–0 ✓
(11,15) 1–11 ✓
(13,15) 11–1
(14,15) 111– ✓

Iteration-II

ABCD
(4,12,5,13) –10–
(10,11,14,15) 1–1–
~~(10,14,11,15) 1–1–~~
~~(4,5,12,13) –10–~~
(12,13,14,15) 11––
~~(12,14,13,15) 11––~~

PRIME IMPLICANTS:

(1,3) – $\bar{A}\bar{B}D$
(1,5) – $\bar{A}\bar{C}D$
(3,11) – $\bar{B}CD$
(4,12,5,13) – $B\bar{C}$
(10,11,14,15) – AC
(12,13,14,15) – AB

Min terms	P.I	1?	3?	4 ✓	5 ✓	10 ✓	11 ✓	12 ✓	13	14 ✓	15 ✓
✓(1,3)	$\bar{A}\bar{B}\bar{D}$	X	X								
1,5	$\bar{A}\bar{C}D$	X			X						
3,11	$\bar{B}CD$		X				X				
✓4,12,5,13	$B\bar{C}$			X	X			X	X		
✓10,11,14,15	AC					X	X			X	X
12,13,14,15	AB							X	X	X	X

$$F = \bar{A}\bar{B}\bar{D} + B\bar{C} + AC.$$

$$F = \bar{A}\bar{B}D + B\bar{C} + AC.$$

Verification using K-map :- $F=\Sigma(1,3,4,5,10,11,12,13,14,15)$

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

$$\boxed{F = B\bar{C} + AC + \bar{A}\bar{B}D}$$

Hence verified.

B. $F = \Sigma(5,6,7,12,14,15)$

$d = \Sigma(3,9,11)$

	A	B	C	D
3	0	0	1	1
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
9	1	0	0	1
11	1	0	1	1
12	1	1	0	0
14	1	1	1	0
15	1	1	1	1

5, 10, 11, 12, 13, 14, 15

	A	B	C	D
3	0	0	1	1 ✓
5	0	1	0	1 ✓
6	0	1	1	0 ✓
9	1	0	0	1 ✓
12	1	1	0	0 ✓
7	0	1	1	1 ✓
11	1	0	1	1 ✓
14	1	1	1	0 ✓
15	1	1	1	1 ✓

Iteration-1

(3,7) 0–11 ✓
(3,11) –011 ✓
(5,7) 01–1 °
(6,7) 011– ✓
(6,14) –110 ✓
(9,11) 10–1 °
(12,14) 11–0 °
(7,15) –111 ✓
(11,15) 1–11 ✓
(14,15) 111– ✓

Iteration-2

	A B C D
(3,7,11,15)	– – 1 1
~~(3,11,7,15)~~	– – 1 1
(6,7,14,15)	– 1 1 –
(6,14,7,15)	– 1 1 –

PRIME IMPLICANTS.

(5,7) $\bar{A}BD$
(9,11) ~~$A\bar{B}$~~ $A\bar{B}D$
(12,14) $AB\bar{D}$
(3,7,11,15) → CD
(6,7,14,15) → BC

Min terms	P.I	~~3~~	5	6	7	~~9~~	~~11~~	12	14	15
✓ 5,7	$\bar{A}BD$		X		X					
9,11	$A\bar{B}D$									
✓ 12,14	$AB\bar{D}$							X	X	
3,7,11,15	CD				X					X
✓ 6,7,14,15	BC			X	X				X	X

$F = \bar{A}BD + AB\bar{D} + BC$

verification using K-map.

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

$\boxed{F = BC + \bar{A}BD + AB\bar{D}}$

Hence Verified.

31.07.18

Unit - II Combinational circuits.

Digital logic circuits

Combinational

o/p depends upon present i/p

→ Half Adder

→ Full Adder.

→ Ripple carry adder (parallel adder)

→ Mux

→ De Mux

→ Decoder

→ Encoder

→ Comparator

Sequential.

(Flip flops)

→ Register

→ Counter

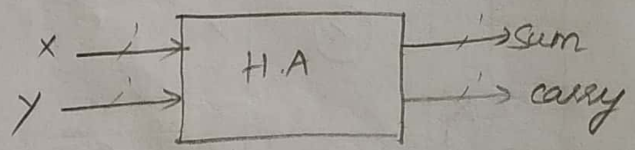
→ Shift register

→ ALU

→ Memory

→ Control Unit

Eg:- 1. **Half Adder.**

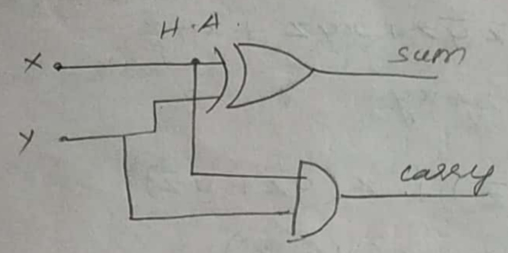


1 bit

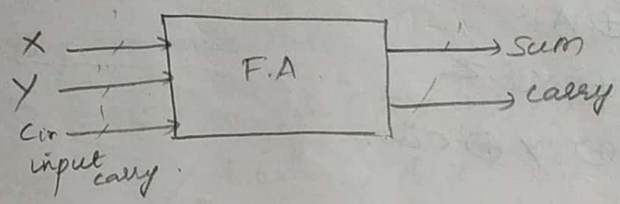
x	y	sum	carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$Sum = \overline{x}y + x\overline{y} = x \oplus y$

$carry = xy$



Full Adder:



	x	y	cin	sum	carry
0	0	0	0	0	0
1	0	0	1	1	0
2	0	1	0	1	0
3	0	1	1	0	1
4	1	0	0	1	0
5	1	0	1	0	1
6	1	1	0	0	1
7	1	1	1	1	1

sum

x \ yz	$\bar{y}\bar{z}$ 00	$\bar{y}z$ 01	yz 11	$y\bar{z}$ 10
$\bar{x}$ 0	(0)	1 (1)	(3)	1 (2)
x 1	1 (4)	(5)	1 (7)	(6)

$$F \Rightarrow sum \Rightarrow x\bar{y}\bar{z} + \bar{x}\bar{y}z + xyz + \bar{x}y\bar{z}$$

x \ yz	$\bar{y}\bar{z}$ 00	$\bar{y}z$ 01	yz 11	$y\bar{z}$ 10
$\bar{x}$ 0	(0)	(1)	1 (3)	(2)
x 1	(4)	1 (5)	1 (7)	1 (6)

$$carry \Rightarrow \cancel{yz + xz + xy}$$
$$yz + xz + xy$$

Sum: $x\bar{y}\bar{z} + \bar{x}\bar{y}z + xyz + \bar{x}y\bar{z}$

Carry: $yz + xz + xy$

sum:

$$\bar{x}(\bar{y}z + y\bar{z}) + x(yz + \bar{y}\bar{z})$$

$$\bar{x}(\underbrace{y \oplus z}_{A}) + x(\underbrace{y \odot z}_{\bar{A}})$$

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

$$= x \oplus A$$

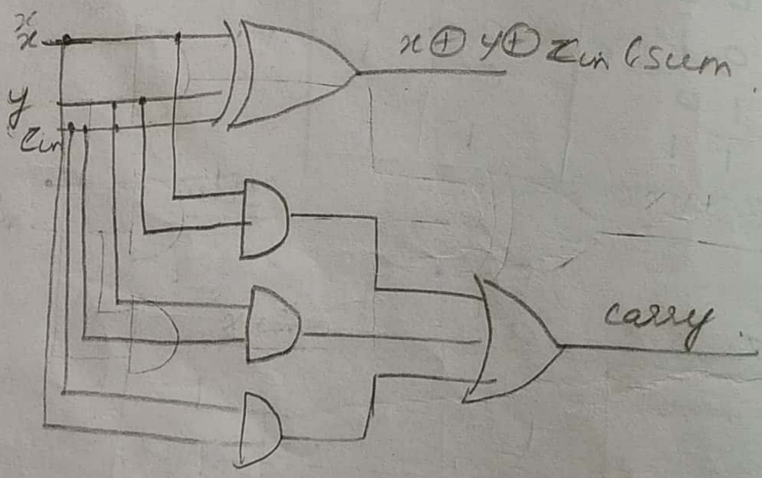
$$= x \oplus y \oplus z$$

$$= x \oplus y \oplus C_{in}.$$

Carry: $yz + xz + xy$.

$$\cancel{x(z+y) + yz}$$

$$= xy + yC_{in} + xC_{in}.$$



Half Adder..

1. Full Adder using 2 half adder & one OR gate.

carry: $xy + y\,c_{in} + x\,c_{in}$.

$$= xy + (x+\bar{x})\,y\,c_{in} + x(y+\bar{y})\,c_{in}$$

$$= xy + x\,y\,c_{in} + \bar{x}\,y\,c_{in} + \cancel{x\,y\,c_{in}} + x\bar{y}\,c_{in}.$$

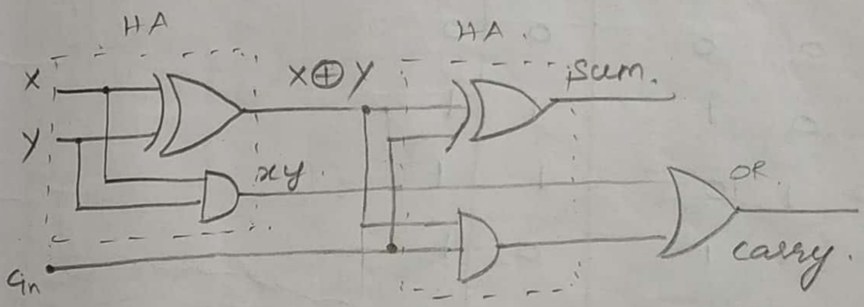
$$= \underbrace{xy + xy\,c_{in}} + \bar{x}\,y\,c_{in} + x\,\bar{y}\,c_{in}.$$

Using absorption law; $(A + AB = A)$

$$= xy + c_{in}(\bar{x}y + x\bar{y})$$

<u>carry</u> $= xy + c_{in}(x \oplus y)$

<u>Sum</u> $= x \oplus y \oplus c_{in}$



2. Half Subtractor.

x	y	diff (D)	borrow (B)
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

Difference:

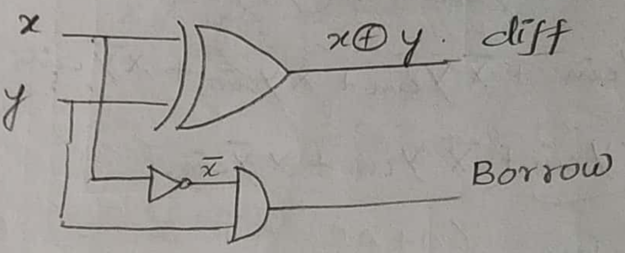
$$D = \bar{x}y + x\bar{y}$$
$$= x \oplus y$$

Borrow

$$B = \bar{x}\,y$$

Difference : $x \oplus y$

Borrow : $\bar{x}y$.



FULL SUBTRACTOR:

	x	y	c_{in}	D	B
0	0	0	0	0	0
1	0	0	1	1	1
2	0	1	0	1	1
3	0	1	1	0	1
4	1	0	0	1	0
5	1	0	1	0	0
6	1	1	0	0	0
7	1	1	1	1	1

Difference

x \ $y c_{in}$	$\bar{y}\bar{c}$ 00	$\bar{y}c$ 01	yc 11	$y\bar{c}$ 10
$\bar{x}$ 0	(0)	1 (1)	(3)	1 (2)
x 1	1 (4)	(5)	1 (7)	(6)

Difference $D = x\bar{y}\bar{c}_{in} + \bar{x}\bar{y}c_{in} + xyc_{in} + \bar{x}y\bar{c}_{in}$

$$\boxed{D = x \oplus y \oplus c_{in}}$$

Borrow

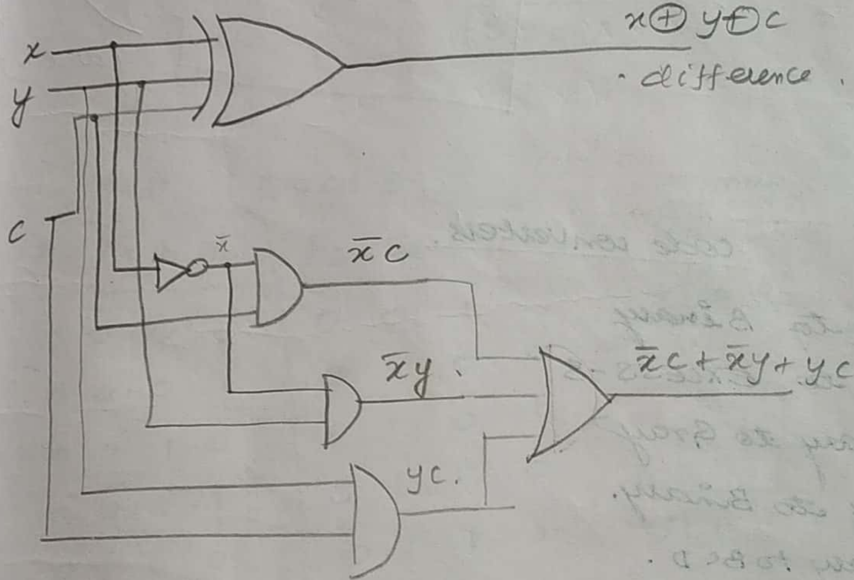
x \ $y c_{in}$	$\bar{y}\bar{c}$ 00	$\bar{y}c$ 01	yc 11	$y\bar{c}$ 10
$\bar{x}$ 0	0	1 (1)	1 (3)	1 (2)
x 1	4	5	1 (7)	6

$$B = \bar{x}c_{in} + \bar{x}y + yc$$

$$\boxed{B = \bar{x}c + \bar{x}y + yc}$$

$$= \cancel{\bar{x}y + c\ell}$$

2 half subt + R.



$$B = \bar{x}c + \bar{x}y + yc.$$

$$= \bar{x}y + \bar{x}(y+\bar{y})c + (x+\bar{x})yc$$

$$= \underbrace{\bar{x}y + \bar{x}yc}_{} + \bar{x}\bar{y}c + xyc + \cancel{\bar{x}yc}$$

$$\bar{x}y(1+c) = \cancel{\bar{x}y + c(\bar{x}\bar{y} + xy) + \bar{x}yc}$$

$$= \cancel{\bar{x}y + c \odot xy + \bar{x}yc}$$

$$= \bar{x}y + \bar{x}\bar{y}c + xyc$$

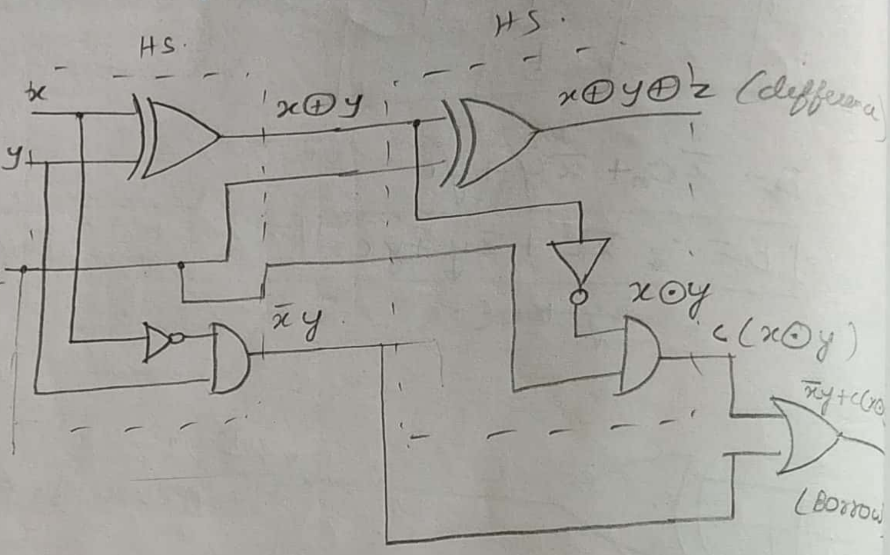
$$= \bar{x}y + c(\bar{x}\bar{y} + xy)$$

$$\cancel{= \bar{x}y + c \odot \bar{x}y}\ c(x \odot y)$$

$$B = \bar{x}y + c(x \odot y)$$

$B = \bar{x}y + c(x \odot y)$

$D = x \oplus y \oplus c.$



07.08.18

code converters.

1. BCD to Binary.
2. BCD to EXCESS-3
3. Binary to Gray
4. Gray to Binary.
5. Binary to BCD.
6. Excess 3 to Binary.

I. 0-15

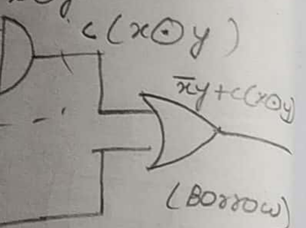
	BCD A_3	A_2	A_1	A_0	BINARY B_3	B_2	B_1	B_0
0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	1
2	0	0	1	0	0	0	1	0
3	0	0	1	1	0	0	1	1
4	0	1	0	0	0	1	0	0
5	0	1	0	1	0	1	0	1
6	0	1	1	0	0	1	1	0
7	0	1	1	1	0	1	1	1
8	1	0	0	0	1	0	0	0
9	1	0	0	1	1	0	0	1
	1	0	1	0				
	1	0	1	1				
	1	1	0	0				
	1	1	0	1				
	1	1	1	0				

Dont care nos.

$y \oplus z'$ (difference)

$x \odot y$

$c(x \odot y)$



BCD				
A_3	A_2	A_1	A_0	
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0				

~~BINARY~~

Draw k maps for B_3 B_2 B_1 B_0

For B_3

$A_3A_2 \backslash A_1A_0$	00	01	11	10
00	0_0	0_1	0_3	0_2
01	0_4	0_5	0_7	0_6
A_3A_2 11	x	x	x	x
$A_3\bar{A}_2$ 10	1_8	1_9	x	x

$\boxed{B_3 = A_3}$.

$B_2 = A_2$

$B_1 = A_1$

$B_0 = A_0$

	BCD				Excess 3.			
	A_3	A_2	A_1	A_0	E_3	E_2	E_1	E_0
0	0	0	0	0	0	0	1	1
1	0	0	0	1	0	1	0	0
2	0	0	1	0	0	1	0	1
3	0	0	1	1	0	1	1	0
4	0	1	0	0	0	1	1	1
5	0	1	0	1	1	0	0	0
6	0	1	1	0	1	0	0	1
7	0	1	1	1	1	0	1	0
8	1	0	0	0	1	0	1	1
9	1	0	0	1	1	1	0	0.

K map

E_3

$A_3A_2 \backslash A_1A_0$	00	01	11	10
00	0	0	0	0
01	0	1	1	1
11	x	x	x	x
10	1	1	x	x

$E_3 = A_3 + A_2A_0 + A_2A_1$

$= A_3 + A_2(A_0 + A_1)$

k map for E_2.

$A_3A_2 \backslash A_1A_0$	$\bar{A}_1\bar{A}_0$ 00	$\bar{A}_1A_0$ 01	A_1A_0 11	$A_1\bar{A}_0$ 10
$\bar{A}_3\bar{A}_2$ 00	0_0	1	1	1_2
$\bar{A}_3A_2$ 01	1_4	0_5	0_7	0_6
A_3A_2 11	x_{12}	x_{13}	x_{15}	x_{14}
$A_3\bar{A}_2$ 10	0_8	1_9	x_{11}	x_{10}

$E_2 = \cancel{A_3}A_0 + \bar{A}_2A_1 + \cancel{\bar{A}_3\bar{A}_2}$

$A_0\bar{A}_2 + A_2\bar{A}_1\bar{A}_0$

K map for E_0

$A_3A_2 \backslash A_1A_0$	00	01	11	10
00	1	0	0	1
01	1	0	0	1
11	X	X	X	X
10	1	0	X	X

$$E_0 = \bar{A}_1\bar{A}_0 + A_1\bar{A}_0$$

$$E_0 = \bar{A}_0(\bar{A}_1 + A_1)$$

K map for E_1

$A_3A_2 \backslash A_1A_0$	00	01	11	10
00	1	0	1	0
01	1	0	1	0
11	X	X	X	X
10	1	0	X	X

$$E_1 = \bar{A}_1\bar{A}_0 + A_1A_0 = A_1 \odot A_0$$

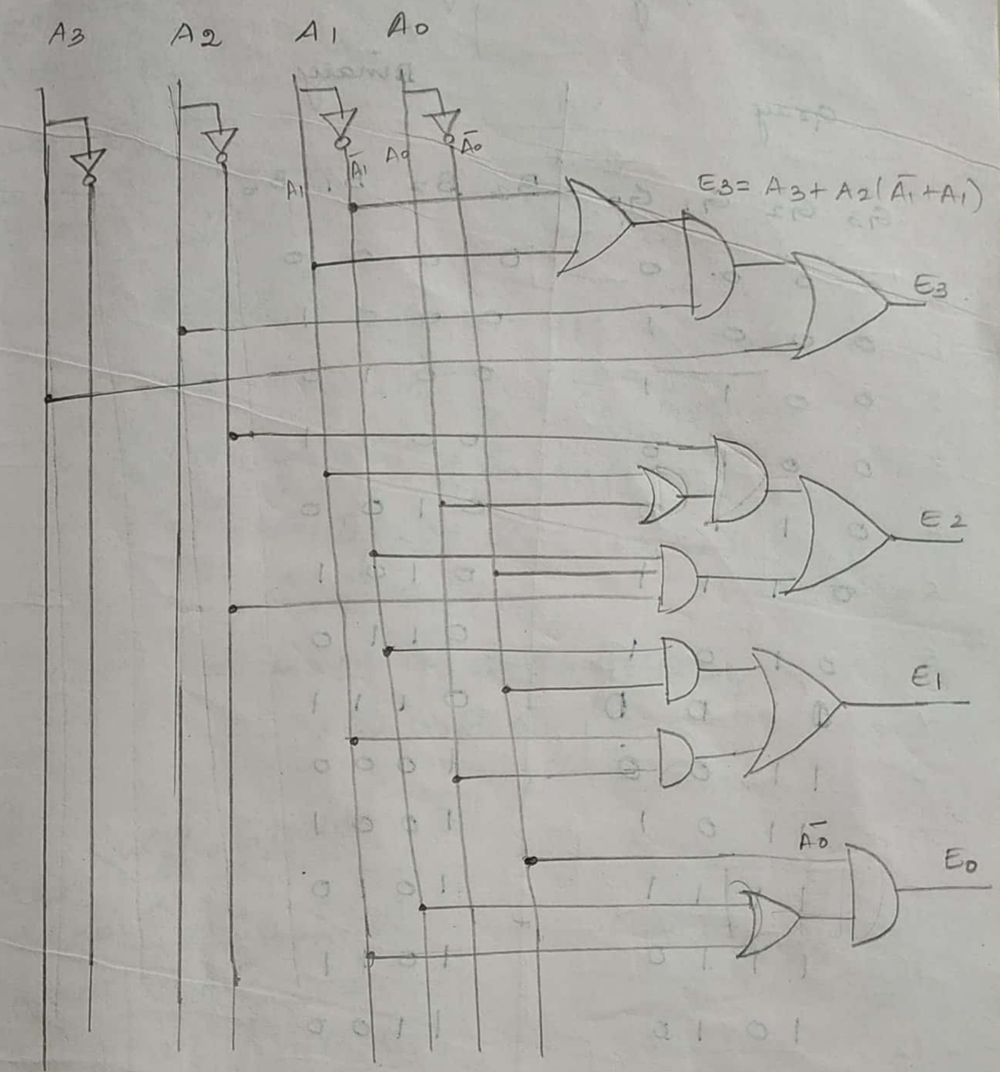
$$E_3 = A_3 + A_2(\bar{A}_1 + A_1)$$

$$E_2 = \bar{A}_2(A_1 + A_0) + \bar{A}_1\bar{A}_0A_2$$

$$E_1 = \bar{A}_1\bar{A}_0 + A_1A_0$$

$$E_0 = \bar{A}_0(\bar{A}_1 + A_1)$$

LOGIC DIAGRAM:



$E_3 = A_3 + A_2(\bar{A_1} + A_1)$

$E_2 = \bar{A_2}(A_1 + A_0) + \bar{A_1}\,\bar{A_0}\,A_2$

$E_1 = \bar{A_1}\bar{A_0} + A_1 A_0$

$E_0 = \bar{A_0}(\bar{A_1} + A_1)$

Using this logic diagram is drawn above

Gray to Binary

	Gray				Binary			
	G_3	G_2	G_1	G_0	B_3	B_2	B_1	B_0
0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	1
2	0	0	1	1	0	0	1	0
3	0	0	1	0	0	0	1	1
4	0	1	1	0	0	1	0	0
5	0	1	1	1	0	1	0	1
6	0	1	0	1	0	1	1	0
7	0	1	0	0	0	1	1	1
8	1	1	0	0	1	0	0	0
9	1	1	0	1	1	0	0	1
10	1	1	1	1	1	0	1	0
11	1	1	1	0	1	0	1	1
12	1	0	1	0	1	1	0	0
13	1	0	1	1	1	1	0	1
14	1	0	0	1	1	1	1	0
15	1	0	0	0	1	1	1	1

B_2

$G_3G_2 \backslash G_1G_0$	00	01	11	10
00	0	0	0	0
01	1	1	1	1
11	0	0	0	0
10	1	1	1	1

$\boxed{B_3 = G_3}$

$B_2 = \bar{G}_3 G_2 + G_3 \bar{G}_2$

$\boxed{B_2 = G_3 \oplus G_2.}$

$\underline{B_1}$

$G_3G_2 \backslash G_1G_0$	$\bar{G_1}\bar{G_0}$ 00	$\bar{G_1}G_0$ 01	G_1G_0 11	$G_1\bar{G_0}$ 10
$\bar{G_3}\bar{G_2}$ 00	0	0	1	1
$\bar{G_3}G_2$ 01	1	1	0	0
G_3G_2 11	0	0	1	1
$G_3\bar{G_2}$ 10	1	1	0	0

$$B_1 = G_3\bar{G_2}\bar{G_1} + G_3G_2G_1 + \bar{G_3}\bar{G_2}G_1 + \bar{G_3}G_2\bar{G_1}$$

$$= G_3(\bar{G_2}\bar{G_1} + G_2G_1) + \bar{G_3}(\bar{G_2}G_1 + G_2\bar{G_1})$$

$$= G_3(G_2 \odot G_1) + \bar{G_3}(G_1 \oplus G_2)$$

$$= \bar{G_3}(G_1 \oplus G_2) + G_3(\overline{G_1 \oplus G_2})$$

$$= \bar{G_3} \oplus G_1 \oplus G_2$$

$$= \cancel{G_3 \oplus G} \; \cancel{G_1 \oplus G_2 \oplus G_3}$$

$$= G_3 \oplus G_2 \oplus G_1$$

$\underline{B_0}$

$G_3G_2 \backslash G_1G_0$	$\bar{G_1}\bar{G_0}$ 00	$\bar{G_1}G_0$ 01	G_1G_0 11	$G_1\bar{G_0}$ 10
$\bar{G_3}\bar{G_2}$ 00	0_0	1_1	0_3	1_2
$\bar{G_3}G_2$ 01	1_4	0_5	1_7	0_6
G_3G_2 11	0_{12}	1_{13}	0_{15}	1_{14}
$G_3\bar{G_2}$ 10	1_8	0_9	1_{10}	0_{11}

$$B_0 = \bar{G_3}\bar{G_2}\bar{G_1}G_0 + \bar{G_3}\bar{G_2}G_1\bar{G_0} +$$
$$\bar{G_3}G_2\bar{G_1}\bar{G_0} + \bar{G_3}G_2G_1G_0 +$$
$$G_3G_2\bar{G_1}G_0 + G_3G_2G_1\bar{G_0} +$$
$$G_3\bar{G_2}\bar{G_1}\bar{G_0} + G_3\bar{G_2}G_1G_0$$

$$= \bar{G_3}\bar{G_2}(\bar{G_1}G_0 + G_1\bar{G_0}) + \bar{G_3}G_2(\bar{G_1}\bar{G_0} + G_1G_0) +$$
$$G_3G_2(\bar{G_1}G_0 + G_1\bar{G_0}) + G_3\bar{G_2}(\bar{G_1}\bar{G_0} + G_1G_0)$$

$$= \bar{G_3}\bar{G_2}(G_1 \oplus G_0) + \bar{G_3}G_2(G_1 \odot G_0) + G_3G_2(G_1 \oplus G_0)$$
$$+ G_3\bar{G_2}(G_1 \odot G_0)$$

$$= (\bar{G_3}\bar{G_2} + G_3G_2)(G_1 \oplus G_0) + (\bar{G_3}G_2 + \bar{G_2}G_3)(G_1 \odot G_0)$$

$$= (G_3 \odot G_2)(G_1 \oplus G_0) + \cancel{G_3G_2}(G_3 \oplus G_2)(G_1 \odot G_0)$$

$= (G_3 \odot G_2)(G_1 \oplus G_0) + \overline{\overline{(G_3 \oplus G_2)}}\ \overline{\overline{(G_1 \odot G_0)}}$

$= (G_3 \odot G_2)(G_1 \oplus G_2) + (\overline{G_3 \odot G_2})(\overline{G_1 \oplus G_0})$

$\cancel{AB + \bar{A}\bar{B}}$

$= \cancel{G_3 \odot G_2 \odot G_1}$

$= (G_3 \odot G_2)(G_1 \oplus G_0) + (\overline{G_3 \odot G_2})(\overline{G_0 \oplus G_1})$

$\quad A \qquad B \qquad \bar{A} \qquad \bar{B}$

$\cancel{(G_3 \odot G_2) \odot G_1 \oplus G_0}$

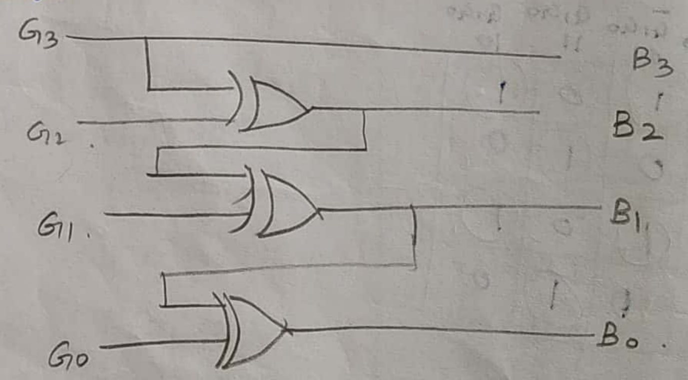
$= (\overline{G_3 \oplus G_2})(G_1 \oplus G_0) + (\overline{G_0 \oplus G_1})(G_2 \oplus G_3)$

$\quad \bar{A} \qquad B \qquad \bar{B} \qquad A$

$= G_3 \oplus G_2 \oplus G_1 \oplus G_0$

$B_0 = G_0 \oplus B_2 \oplus G_1$

Design:-



Excess 3 to Binary

Excess -3 E_3 E_2 E_1 E_0	Binary B_3	B_2	B_1	B_0	
	0	0	0	0	0
	0	0	0	1	1
	0	0	1	0	2
	0	0	1	1	3
	0	1	0	0	4
	0	1	0	1	5
	0	1	1	0	6
	0	1	1	1	7
	1	0	0	0	8
	1	0	0	1	9
	1	0	1	0	10
	1	0	1	1	11
	1	1	0	0	12
X	1	1	0	1	13
Dont cares X	1	1	1	0	14
X	1	1	1	1	15

10.8.18

Binary parallel Adders.

Half adders – 2 bits } each input
Full adder – 3 bits } 1 bits.

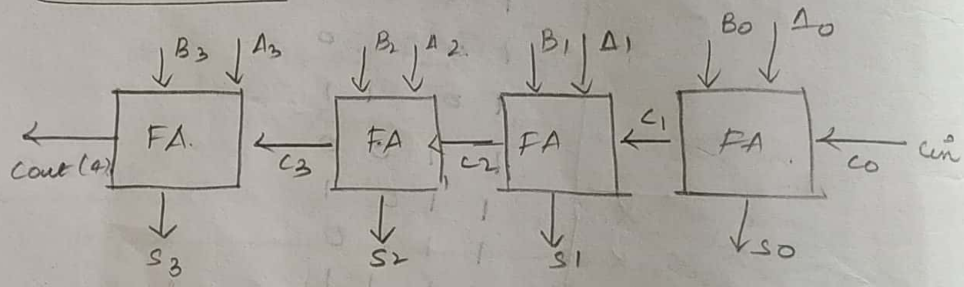
1. Ripple carry Adder (carry propagate Adder)
2. carry look ahead adder.

1. Ripple carry Adder:

(made of full adders)

If n bits are required, the n-number of full adders are required.

4-bit RCA.



$A = A_3 A_2 A_1 A_0 = 1101$

$B = B_3 B_2 B_1 B_0 = 1001$

Initially i/p carry is zero.

$C_3 \quad C_2 \quad C_1 \quad C_0 = 0$

$A = A_3 \; A_2 \; A_1 \; A_0$

$B = B_3 \; B_2 \; B_1 \; B_0$

$S_3 \; S_2 \; S_1 \; S_0$

C_4

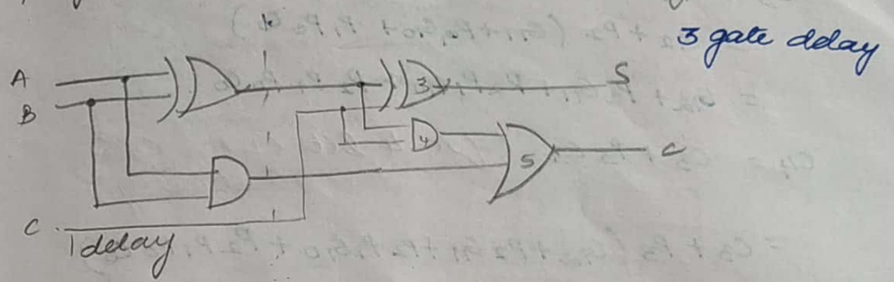
Cout

3 inputs are added

∴ full adders are used

Propagation delay: (Generally in nano second)
Time Delay between the input arrivals to the gate & the generation of carry.

Propagation delay depends upon the gate.



In full adder → 2 gate delay. (in general)

Disadvantage:

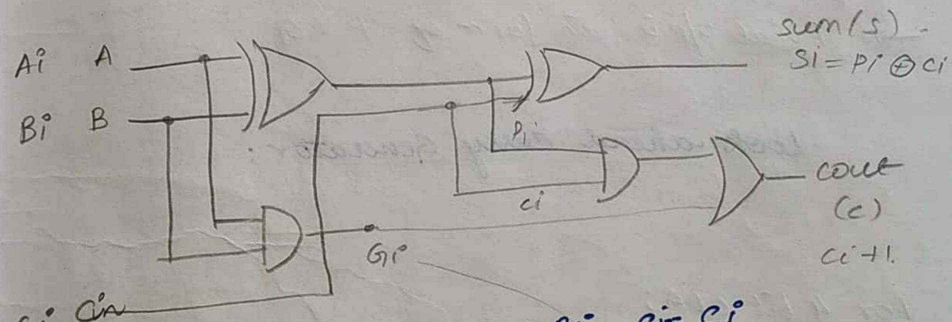
Longest propagation delay.

2. Carry Look ahead adder: (high speed parallel adder)

All the carry are generated simultaneously.

We modify based on full adder

[consider it as single bit]



Assume $A \to A_i$, $B \to B_i$, $c_{in} = C_i$

$P_i \to$ Propagate function $= A_i \oplus B_i$ (for i/p)

$G_i \to$ Generate function $= A_i B_i$ (And combination of inputs)

$$C_{i+1} = G_i + P_i C_i$$

apply $i=0$

$\rightarrow C_1 = G_0 + P_0 C_0$, where $C_0 \to$ input carry.

when $i=2$

$\rightarrow C_2 = G_1 + P_1 C_1$

sub C_1 in C_2

$\rightarrow C_2 = G_1 + P_1(G_0 + P_0 C_0)$

$= G_1 + P_1 G_0 + P_1 P_0 C_0$

$C_3 = G_2 + P_2 C_2$

$= G_2 + P_2 (G_1 + P_1 G_0 + P_1 P_0 C_0)$

$= G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 C_0$.

$C_4 = C_3 + P_3 C_3$ (for 4 bit CLA, C_4 is the final carry)

$= C_3 + P_3 (G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 C_0)$

$= C_3 + P_3 G_2 + P_3 P_2 G_1 + P_3 P_2 P_1 G_0 + P_3 P_2 P_1 P_0 C_0$

$C_n = G_{n-1} + P_{n-1} G_{n-2} + P_{n-1} P_{n-2} G_{n-3} + \ldots \ldots$

$\ldots + (P_{n-1} P_{n-2} \ldots P_1 P_0 C_0)$

For 8 bit CLA:

$C_{out} = C_7 = G_7 + P_7 G_6 + P_7 P_6 G_5 + P_7 P_6 P_5 G_4 + P_7 P_6 P_5 P_4 G_3 + P_7 P_6 P_5 P_4 P_3 G_2 + P_7 P_6 P_5 P_4 P_3 P_2 G_1 + P_7 P_6 P_5 P_4 P_3 P_2 P_1 G_0 + P_7 P_6 P_5 P_4 P_3 P_2 P_1 P_0 \underline{\underline{C_0}}$ input carry

→ Final o/p is in form of P & G

Look ahead carry Generator:

⟹

For 4 bit CLA;

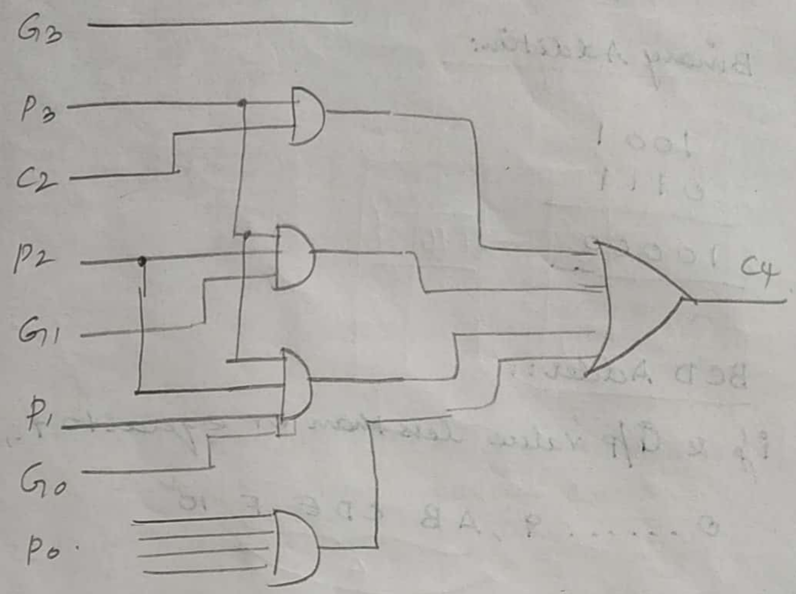
we get 4 gate delay (fixed) from i/p to output sum.

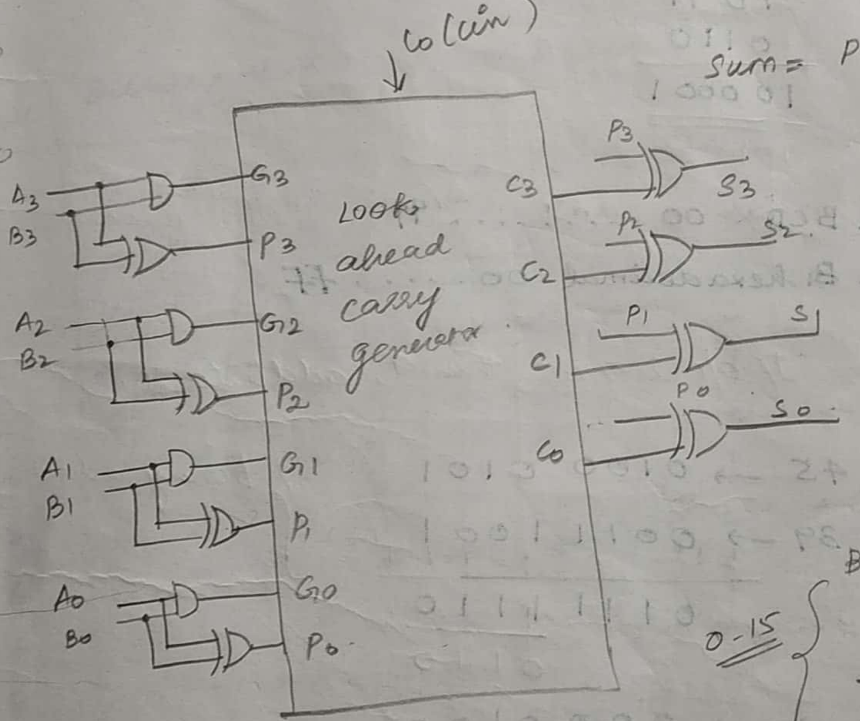
(Very high speed parallel adder)

to generate G, P → 1 gate

to generate sum = 1 gate.

Look ahead carry generator





Sum = $P_i \oplus C_i$

Binary add

$$0\text{-}15 \left\{ \begin{array}{r} 1001 \\ 0111 \\ \hline 10000 \\ \hline 16 \end{array} \right.$$

BCD addition

0 to 9

From 10 it repeats.

BCD Adder. (4 bit)

Binary Addition:

```
 1001
 0111
-----
10000  → [16]
```

BCD Adder:

i/p & O/p value less than or equal to 9.

0 ...... 9, A B C D E F 10

$\underbrace{\text{A B C D E F}}_{6}$

add 6

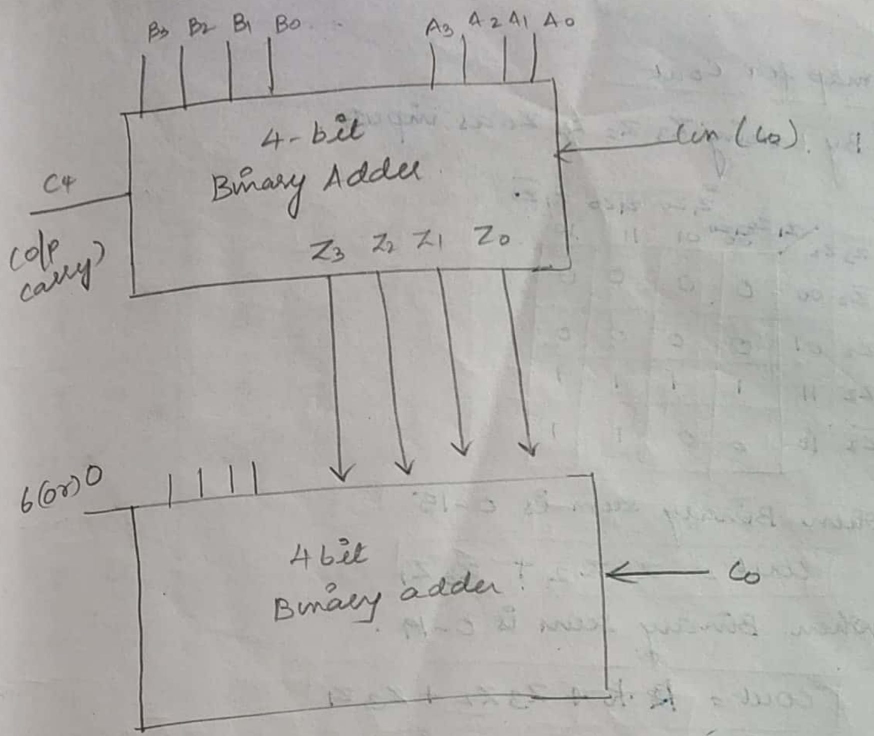
```
 1011
 0110
------
100001
```

In BCD 00 .... 99 } highest value

In Bi hexadecimal 00 ..... FF

If o/p is greater than 9, add 6 to it.

```
45 → 0100 0101
39 → 0011 1001 (+)   } binary addition
     ---------
     0111 1110
          0110
     ---------
84 → 10000100
```



DECIMAL	Binary sum K	Z_3	Z_2	Z_1	Z_0	BCD sum Cout	S_3	S_2	S_1	S_0
0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	1	0	0	0	0	1
2	0	0	0	1	0	0	0	0	1	0
3	0	0	0	1	1	0	0	0	1	1
4	0	0	1	0	0	0	0	1	0	0
5	0	0	1	0	1	0	0	1	0	1
6	0	0	1	1	0	0	0	1	1	0
7	0	0	1	1	1	0	0	1	1	1
8	0	1	0	0	0	0	1	0	0	0
9	0	1	0	0	1	0	1	0	0	1
10	0	1	0	1	0	1	0	0	0	0
11	0	1	0	1	1	1	0	0	0	1
12	0	1	1	0	0	1				
13	0	1	1	0	1	1				
14	0	1	1	1	0	1				
15	0	1	1	1	1	1				
⋮										
19										

K map for Cout

By taking $Z_3\ Z_2\ Z_1\ Z_0$ as inputs.

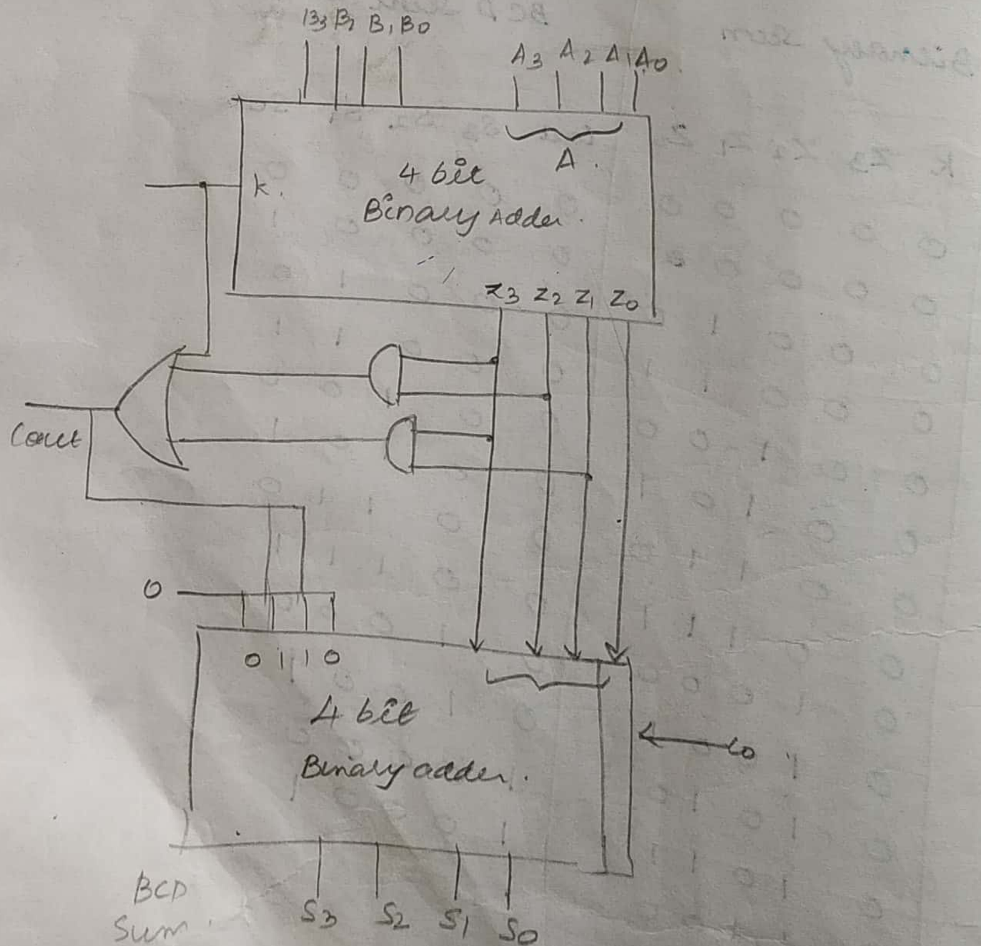
$Z_3Z_2 \backslash Z_1Z_0$	$\bar{Z}_1\bar{Z}_0$ 00	$\bar{Z}_1Z_0$ 01	Z_1Z_0 11	$Z_1\bar{Z}_0$ 10
$\bar{Z}_3\bar{Z}_2$ 00	0	0	0	0
$\bar{Z}_3Z_2$ 01	0	0	0	0
Z_3Z_2 11	1	1	1	1
$Z_3\bar{Z}_2$ 10	0	0	1	1

When Binary sum is 0-15

$$\boxed{C_{out} = Z_3Z_2 + Z_3Z_1}$$

When Binary sum is 0-19.

$$\boxed{C_{out} = K + Z_3Z_2 + Z_3Z_1}$$



.|08|18

Magnitude comparator

Inputs				Outputs			
A		B		A > B	A = B	A < B	
A_1	A_0	B_1	B_0				
0	0	0	0	0	1	0	0
0	0	0	1	0	0	1	1
0	0	1	0	0	0	1	2
0	0	1	1	0	0	1	3
0	1	0	0	1	0	0	4
0	1	0	1	0	1	0	5
0	1	1	0	0	0	1	6
0	1	1	1	0	0	1	7
1	0	0	0	1	0	0	8
1	0	0	1	1	0	0	9
1	0	1	0	0	1	0	10
1	0	1	1	0	0	1	11
1	1	0	0	1	0	0	12
1	1	0	1	1	0	0	13
1	1	1	0	1	0	0	14
1	1	1	1	0	1	0	15

Kmap

A > B

(i)

A_1A_0 \ B_1B_0	$\bar B_1 \bar B_0$ 00	$\bar B_1 B_0$ 01	$B_1 B_0$ 11	$B_1 \bar B_0$ 10
$\bar A_1 \bar A_0$ 00	0	0	0	0
$\bar A_1 A_0$ 01	1	0	0	0
$A_1 A_0$ 11	1	1	0	1
$A_1 \bar A_0$ 10	1	1	0	0

$\boxed{A > B}$

$$= A_0 \bar B_1 \bar B_0 + A_1 A_0 \bar B_0 + A_1 \bar B_1$$

(ii) A = B

A = B

$A_1A_0 \backslash B_1B_0$	00	01	11	10
00	1 (0)	0 (1)	0 (3)	0 (2)
01	0 (4)	1 (5)	0 (7)	0 (6)
11	0 (12)	0 (13)	1 (15)	0 (14)
10	0 (8)	0 (9)	0 (11)	1 (10)

$$\boxed{A=B} = \bar{A}_1\bar{A}_0\bar{B}_1\bar{B}_0 + \bar{A}_1A_0\bar{B}_1B_0 + A_1A_0B_1B_0 + A_1\bar{A}_0B_1\bar{B}_0$$

(iii) A < B

$A_1A_0 \backslash B_1B_0$	00 ($\bar{B}_1\bar{B}_0$)	01 ($\bar{B}_1B_0$)	11 (B_1B_0)	10 ($B_1\bar{B}_0$)
00 ($\bar{A}_1\bar{A}_0$)	0 (0)	1 (1)	1 (3)	1 (2)
01 ($\bar{A}_1A_0$)	0 (4)	0 (5)	1 (7)	1 (6)
11 (A_1A_0)	0 (12)	0 (13)	0 (15)	0 (14)
10 ($A_1\bar{A}_0$)	0 (8)	0 (9)	1 (11)	0 (10)

$$\boxed{A<B} = \bar{A}_1B_1 + \bar{A}_1\bar{A}_0B_0 + \bar{A}_0B_1B_0$$

case(i) : $\boxed{A>B}$

$$A>B = A_1\bar{B}_1 + A_0\bar{B}_1\bar{B}_0 + A_1A_0\bar{B}_0$$

$$= A_1\bar{B}_1 + (A_1+\bar{A}_1)A_0\bar{B}_1\bar{B}_0 + A_1A_0(B_1+\bar{B}_1)\bar{B}_0$$

$$= A_1\bar{B}_1 + A_1A_0\bar{B}_1\bar{B}_0 + \bar{A}_1A_0\bar{B}_1\bar{B}_0 + A_1A_0B_1\bar{B}_0 + \cancel{A_1A_0\bar{B}_1\bar{B}_0}$$

$$= A_1\bar{B}_1 + A_1A_0\bar{B}_1\bar{B}_0 + A_0\bar{B}_0(\bar{A}_1\bar{B}_1 + A_1B_1)$$

$$= \underbrace{A_1\bar{B}_1 + A_1A_0\bar{B}_1\bar{B}_0}_{\text{use adsorption law}} + (A_1 \odot B_1)A_0\bar{B}_0$$

$$= A_1\bar{B}_1(1 + A_0\bar{B}_0) + (A_1 \odot B_1)A_0\bar{B}_0$$

$$\boxed{A>B = A_1\bar{B}_1 + (A_1 \odot B_1)A_0\bar{B}_0}$$

From (i), (ii) and (iii) cases:-

$$(A=B) = x_1x_0$$

1101
1101

$$(A>B) = A_1\bar{B}_1 + x_1A_0\bar{B}_0$$

$$(A<B) = \bar{A}_1B_1 + x_1\bar{A}_0B_0$$

case (iii): $\boxed{A<B}$

$$A<B = \bar{A}_1 B_1 + \bar{A}_1 \bar{A}_0 B_0 (B_1+\bar{B}_1) + \bar{A}_0 B_1 B_0 (A_1+\bar{A}_1)$$

$$= \bar{A}_1 B_1 + \bar{A}_1 \bar{A}_0 B_0 \bar{B}_1 + \bar{A}_1 \bar{A}_0 \bar{B}_1 B_0 + A_1 \bar{A}_0 B_1 B_0 + \bar{A}_1 \bar{A}_0 B_1 B_0$$

$$= \bar{A}_1 B_1 (1+\bar{A}_0 B_0) + \bar{A}_0 B_0 (\bar{A}_1 \bar{B}_1 + A_1 B_1)$$

$$= \bar{A}_1 B_1 + \bar{A}_0 B_0 \underbrace{(A_1 \odot B_1)}_{X_1} = \bar{A}_1 B_1 + \bar{A}_0 B_0 X_1$$

From ①

case(ii) $\boxed{A=B}$:-

$$= \bar{A}_1 \bar{A}_0 \bar{B}_1 \bar{B}_0 + \bar{A}_1 A_0 \bar{B}_1 B_0 + A_1 A_0 B_1 B_0 + A_1 \bar{A}_0 B_1 \bar{B}_0$$

$$= \bar{A}_1 \bar{B}_1 (\bar{A}_0 \bar{B}_0 + A_0 B_0) + A_1 B_1 (A_0 B_0 + \bar{A}_0 \bar{B}_0)$$

$$= \bar{A}_1 \bar{B}_1 (A_0 \odot B_0) + A_1 B_1 (A_0 \odot B_0)$$

$$= (A_0 \odot B_0)(\bar{A}_1 \bar{B}_1 + A_1 B_1)$$

$$= (A_0 \odot B_0)(A_1 \odot B_1)$$

$$= \cancel{X_0 X_0 +} X_1 X_0$$

Let $X_1 = A_1 \odot B_1$
$X_0 = A_0 \odot B_0$ } → ①

similarly $\boxed{X_i = A_i \odot B_i}$

While comparing: LHS → RHS
left to Right.

eg:- $X > Y$

X = 5 3 4 3 2 1 7 8

Y = 5 3 4 2 1 7 8 7

1101
1101

4-bit

$A = A_3\, A_2\, A_1\, A_0$

$B = B_3\, B_2\, B_1\, B_0$

$$(A>B) = A_3\overline{B_3} + X_3 A_2 \overline{B_2} + X_3 X_2 A_1 \overline{B_1} + X_3 X_2 X_1 A_0 \overline{B_0}$$

$$(A<B) = \overline{A_3} B_3 + X_3 \overline{A_2} B_2 + X_3 X_2 \overline{A_1} B_1 + X_3 X_2 X_1 \overline{A_0} B_0$$

$$(A=B) = X_3 X_2 X_1 X_0$$

21.08.18.

1Q. Design a combinational circuit with 3 inputs and 1 output. The output is 1, ~~when~~ when the binary value of input is less than 3. The output is 0 otherwise.

$2^3 \rightarrow 8$ combination

Full adder

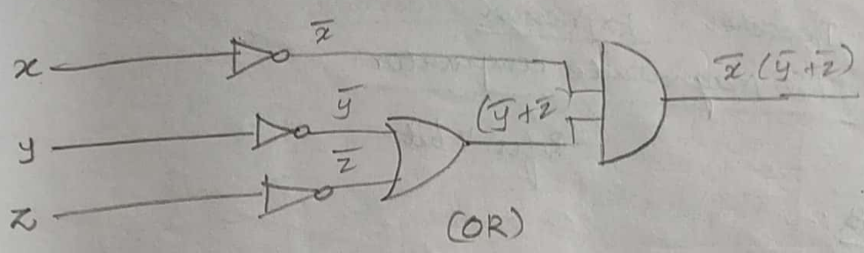
⇒

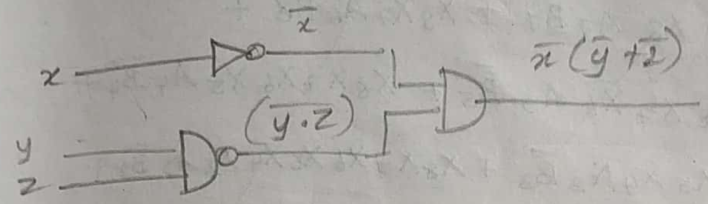
	x	y	z	o/p
0	0	0	0	1
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	0
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

x \ yz	$\bar{y}\bar{z}$ 00	$\bar{y}z$ 01	yz 11	$y\bar{z}$ 10
$\bar{x}$ 0	1 (0)	1 (1)	0 (3)	1 (2)
x 1	0 (4)	0 (5)	0 (7)	0 (6)

$$= \bar{x}\bar{y} + \bar{x}\bar{z}$$

$$= \bar{x}(\bar{y} + \bar{z})$$



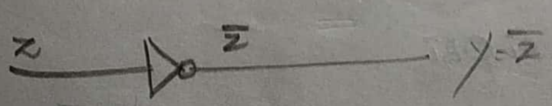


2. Q) Design a combinational circuit with 3 i/p and 1 output. The output is 1 when the binary value of i/p is even no & otherwise 0.

	x	y	z	o/p
0	0	0	0	1
1	0	0	1	0
2	0	1	0	1
3	0	1	1	0
4	1	0	0	1
5	1	0	1	0
6	1	1	0	1
7	1	1	1	0

x \ yz	$\bar{y}\bar{z}$ 00	$\bar{y}z$ 01	yz 11	$y\bar{z}$ 10
$\bar{x}$ 0	1 (0)	0 (1)	0 (3)	1 (2)
x 1	1 (4)	0 (5)	0 (7)	1 (6)

$= \bar{z}$



28.08.18

~~Decoder.~~ Expression

8bit Magnitude comparator:

Refer 4 bit.

⇒ $A>B$:-

$$= A_8\bar{B}_8 + X_8 A_7\bar{B}_7 + X_8X_7A_6\bar{B}_6 +$$
$$X_8X_7X_6A_5\bar{B}_5 + X_8X_7X_6X_5A_4\bar{B}_4 +$$
$$X_8X_7X_6X_5X_4A_3\bar{B}_3 + X_8X_7X_6X_5X_4X_3A_2\bar{B}_2 +$$
$$X_8X_7X_6X_5X_4X_3X_2A_1\bar{B}_1 + X_8X_7X_6X_5X_4X_3X_2X_1A_0$$

⇒ ~~A<~~ $A>B$

$$= A_7\bar{B}_7 + X_7A_6\bar{B}_6 + X_7X_6A_5\bar{B}_5 +$$
$$X_7X_6X_5A_4\bar{B}_4 + X_7X_6X_5X_4A_3\bar{B}_3 +$$
$$X_7X_6X_5X_4X_3A_2\bar{B}_2 + X_7X_6X_5X_4X_3X_2A_1\bar{B}_1 +$$
$$X_7X_6X_5X_4X_3X_2X_1A_0\bar{B}_0$$

⇒ $A<B$

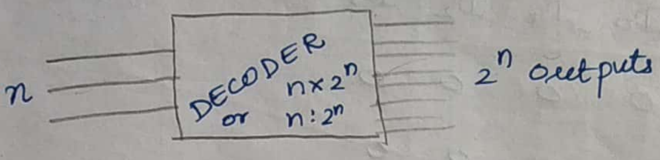
$$= \bar{A}_7B_7 + X_7\bar{A}_6B_6 + X_7X_6\bar{A}_5B_5 +$$
$$X_7X_6X_5\bar{A}_4B_4 + X_7X_6X_5X_4\bar{A}_3B_3 +$$
$$X_7X_6X_5X_4X_3\bar{A}_2B_2 + X_7X_6X_5X_4X_3X_2\bar{A}_1B_1 +$$
$$X_7X_6X_5X_4X_3X_2X_1\bar{A}_0B_0$$

⇒ $A=B$

$$X_7X_6X_5X_4X_3X_2X_1X_0$$

$$X_i = A_i \odot B_i$$

DECODER



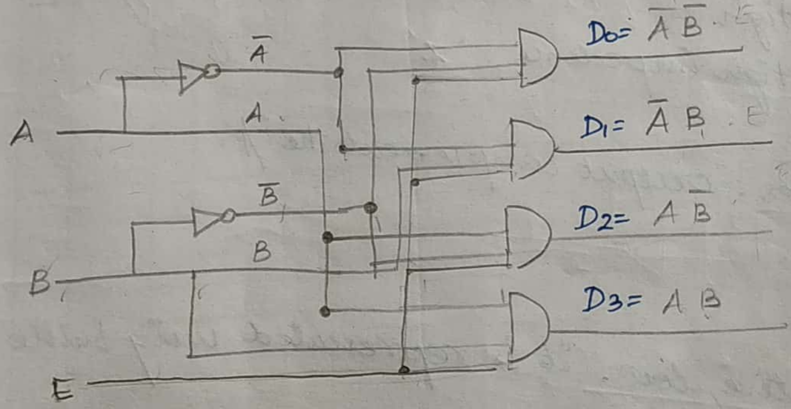
n	2^n
2	$2^2 = 4$
3	$2^3 = 8$
4	$2^4 = 16$

example:

2 : 4
3 : 8
4 : 16
input output

Accepts n inputs and gives 2^n outputs.
Mostly we USE AND gates.
if 4 outputs are required, we use 4 and gates.

2 to 4 Decoder:



Truth table.

i/p	D_0	D_1	D_2	D_3
0 0	1	0	0	0
0 1	0	1	0	0
1 0	0	0	1	0
1 1	0	0	0	1

[Decoder generates 1 in one of the bits]

Use :-
Interconnecting ~~proc~~ components bet i/p and processor
(At the time any 1 of the o/p generates!)

E → enable

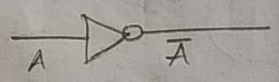
E	A	B	D_0	D_1	D_2	D_3
0	0	0	0	0	0	0
1	0	0	1	0	0	0
1	0	1	0	1	0	0
1	1	0	0	0	1	0
1	1	1	0	0	0	1

E is the extra bit given to i/p

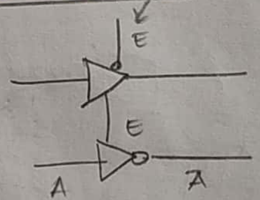
The above concept represents
TRISTATE device.

1. Low → If enable is properly given '0'
2. High → If the enable is not properly given '1'
3. High impedence. 'Z'

Inverter: Output complements the i/p.



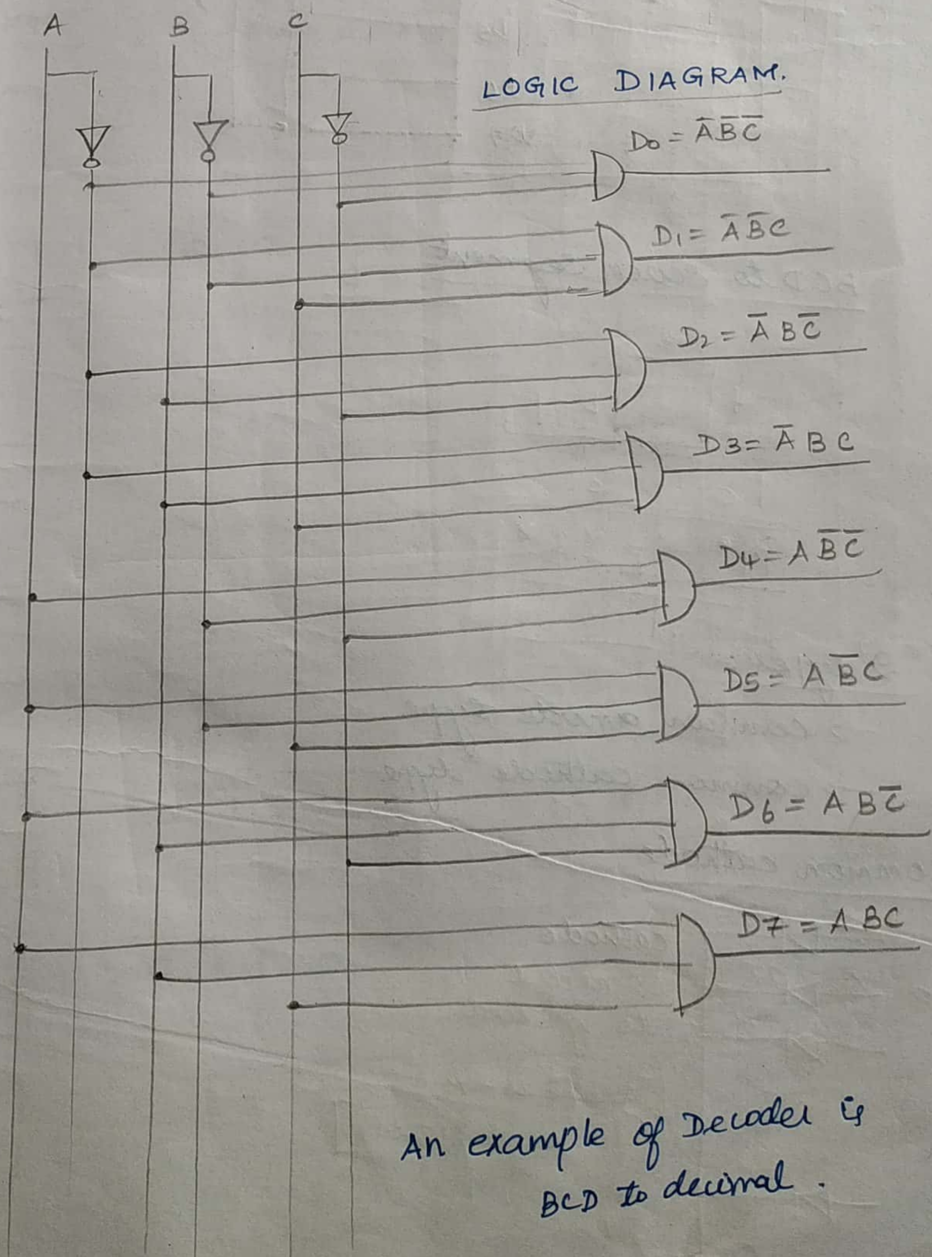
If active low, it is represented using bubble.



3 to 8 is widely used. $\boxed{74LS138}$

3 to 8 decoder without Enable

i/p			o/p							
A	B	C	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

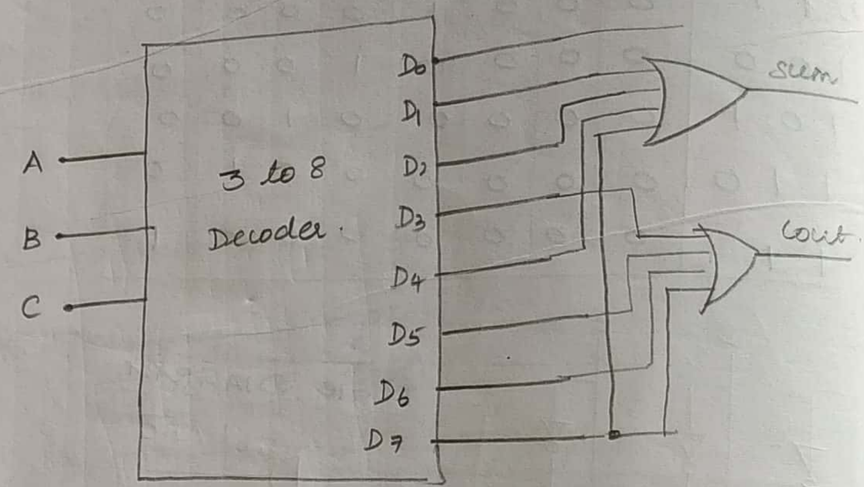


An example of Decoder is BCD to decimal.

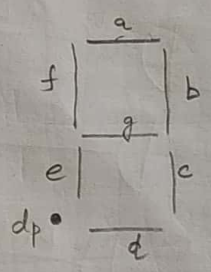
Full adder using Decoder.

$Sum = \Sigma(\overset{D_1}{1}, \overset{D_2}{2}, \overset{D_4}{4}, \overset{D_7}{7})$

$C_{out} = \Sigma(\overset{D_3}{3}, \overset{D_5}{5}, \overset{D_6}{6}, \overset{D_7}{7})$



BCD to seven segment.

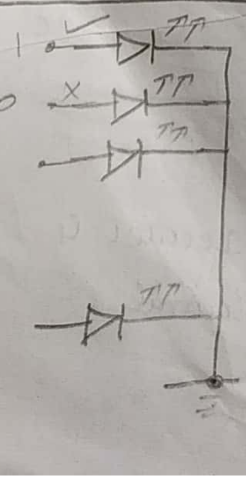


2 types :-

→ common anode type

→ common cathode type.

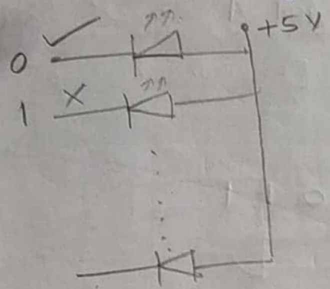
* common cathode



cathode connected to ground

[(Anode is +ve forward biased)]

* common anode type:



HW

Q) How will you design 7-segment display

31.08.18

~~Digit~~ ~~abcde~~

DIGIT	$A_3\ A_2\ A_1\ A_0$	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

Digit	$A_3 A_2 A_1 A_0$	a b c d e f g
8	1 0 0 0	1 1 1 1 1 1 1
9	1 0 0 1	1 1 1 0 0 1 1

Kmap for a:

$A_3A_2 \backslash A_1A_0$	$\bar{A}_1\bar{A}_0$ 00	$\bar{A}_1A_0$ 01	A_1A_0 11	$A_1\bar{A}_0$ 10
$\bar{A}_3\bar{A}_2$ 00	1	0	1	1
$\bar{A}_3A_2$ 01	0	1	1	1
A_3A_2 11	X	X	X	X
$A_3\bar{A}_2$ 10	1	1	X	X

$$a = A_3 + A_1 + A_2A_0 + A_3\bar{A}_0$$

For b:

$A_3A_2 \backslash A_1A_0$	$\bar{A}_1\bar{A}_0$ 00	$\bar{A}_1A_0$ 01	A_1A_0 11	$A_1\bar{A}_0$ 10
$\bar{A}_3\bar{A}_2$ 00	1	1	1	1
$\bar{A}_3A_2$ 01	1	0	1	0
A_3A_2 11	X	X	X	X
$A_3\bar{A}_2$ 10	1	1	X	X

$$b = A_3 + \bar{A}_2 + \bar{A}_1\bar{A}_0 + A_1A_0$$

For c:

For d:

For e:

e=

$+ A_2 A_0 +$

$A_1 A_0$

For c:

$A_3A_2 \backslash A_1A_0$	00 ($\bar{A}_1\bar{A}_0$)	01 ($\bar{A}_1 A_0$)	11 ($A_1 A_0$)	10 ($A_1\bar{A}_0$)
00 ($\bar{A}_3\bar{A}_2$)	1_0	1_1	1_3	0_2
01 ($\bar{A}_3 A_2$)	1_4	1_5	1_7	1_6
11 ($A_3 A_2$)	X_{12}	X_{13}	X_{15}	X_{14}
10 ($A_3\bar{A}_2$)	1_8	1_9	X_{11}	X_{10}

$A_2 + \bar{A}_1 + A_1 A_0 = c$

For d:

$A_3A_2 \backslash A_1A_0$	00 ($\bar{A}_1\bar{A}_0$)	01 ($\bar{A}_1 A_0$)	11 ($A_1 A_0$)	10 ($A_1\bar{A}_0$)
00 ($\bar{A}_3\bar{A}_2$)	1_0	0_1	1_3	1_2
01 ($\bar{A}_3 A_2$)	0_4	1_5	0_7	1_6
11 ($A_3 A_2$)	X	X	X	X
10 ($A_3\bar{A}_2$)	1_8	0_9	X	X

$\bar{A}_2\bar{A}_1$

$d = A_1\bar{A}_0 + \bar{A}_2\bar{A}_0 + \cancel{\bar{A}_2\bar{A}_0} +$

$A_2\bar{A}_1 A_0$

For e:

$A_3A_2 \backslash A_1A_0$	00 ($\bar{A}_1\bar{A}_0$)	01 ($\bar{A}_1 A_0$)	11 ($A_1 A_0$)	10 ($A_1\bar{A}_0$)
00 ($\bar{A}_3\bar{A}_2$)	1_0	0_1	0_3	1_2
01 ($\bar{A}_3 A_2$)	0_4	0_5	0_7	1_6
11 ($A_3 A_2$)	X	X	X	X
10 ($A_3\bar{A}_2$)	1_8	0_9	X	X

$e = \bar{A}_2\bar{A}_0 + A_1\bar{A}_0$

For f

$A_3A_2 \backslash A_1A_0$	00	01	11	10
00	1_0	0_1	0_3	0_2
01 ($\bar{A}_3 A_2$)	1_4	1_5	0_7	1_6
11 ($A_3 A_2$)	X	X	X	X
10	1_8	1_9	X	X

$\bar{A}_1\bar{A}_0 + A_3 + \cancel{\bar{A}_1\bar{A}_0 A_2} + A_2\bar{A}_1$

$A_2\bar{A}_0 \Rightarrow f$

For g

$A_3A_2 \backslash A_1A_0$	00	01	11 ($A_1 A_0$)	10 ($A_1\bar{A}_0$)
00 ($\bar{A}_3\bar{A}_2$)	0_0	0_1	1_3	1_2
01 ($\bar{A}_3 A_2$)	1_4	1_5	0_7	1_6
11 ($A_3 A_2$)	X	X	X	X
10 ($A_3\bar{A}_2$)	1_8	1_9	X	X

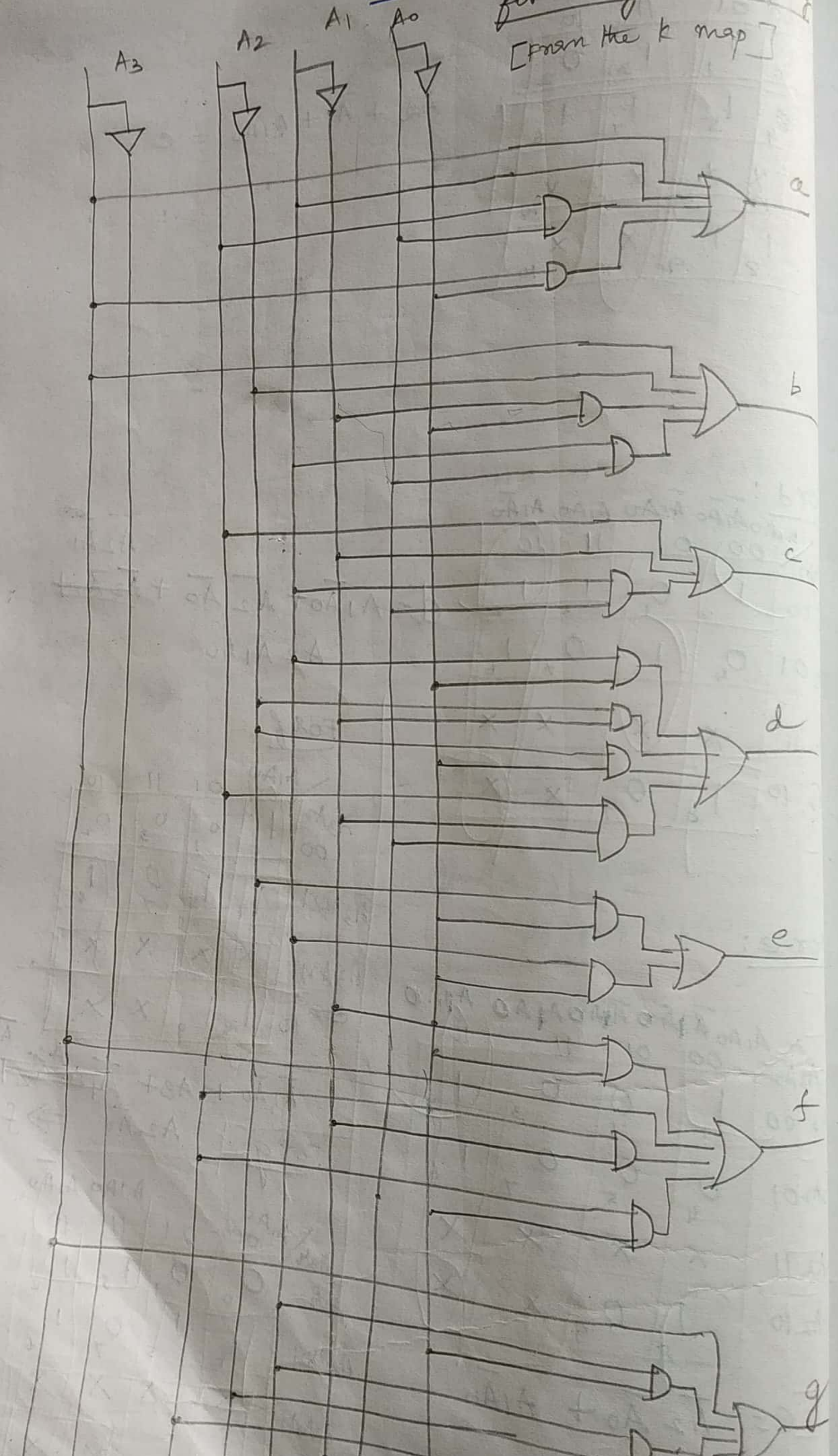
$g = A_3 + A_1\bar{A}_0 + A_1\bar{A}_2 +$
$A_2\bar{A}_1$

Scanned by CamScanner

LOGIC DIAGRAM

for 7 segment display

[from the k map]



2^n inputs

4

Inputs

D0	D1
1	0
0	1
0	0
0	0

D0

D1

D2

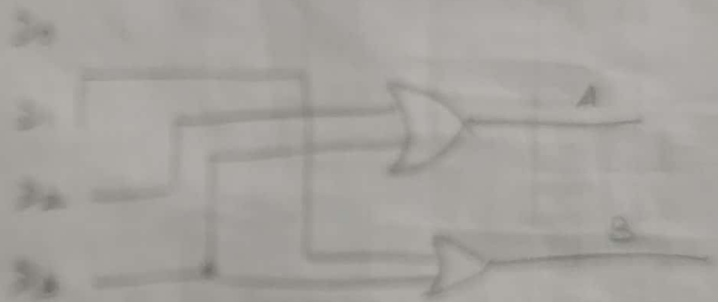
D3

<u>Encoder</u>

2^n inputs and n outputs

4:2 Encoder

Inputs				Outputs	
D_0	D_1	D_2	D_3	A	B
1	0	0	0	0	0
0	1	0	0	0	1
0	0	1	0	1	0
0	0	0	1	1	1



<u>Logic diagram:</u>

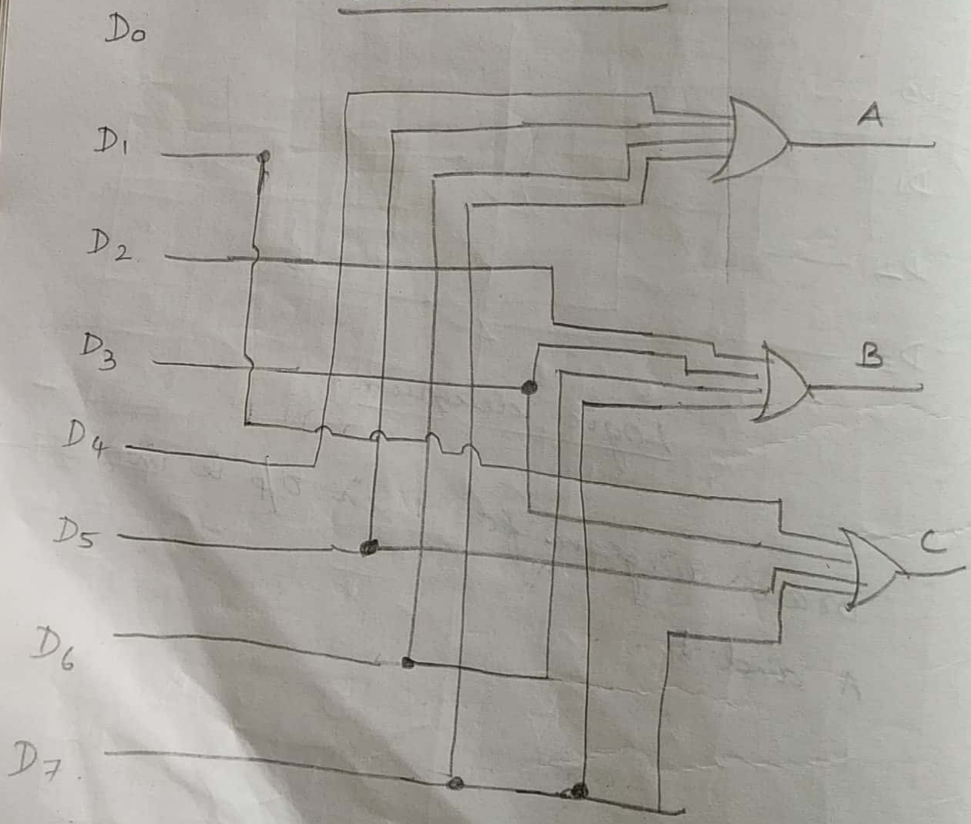
Draw diagram for 1 in o/p ie in A and B.

Scanned by CamScanner

8:3 Encoder:

D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	A	B	C
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

LOGIC DIAGRAM.

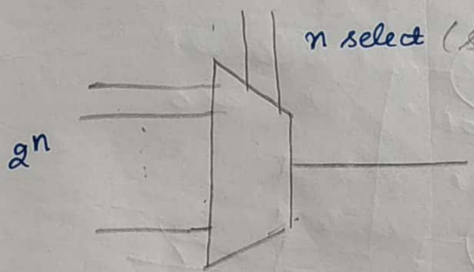


Multiplexer (or) Mux

2^n inputs 1 output

& has n number of select lines

Also called as Data Selector

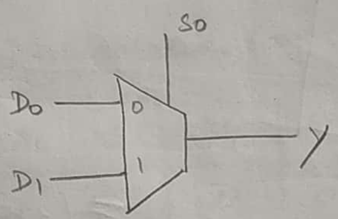


Examples

Mux	Select lines
2 : 1	1
4 : 1	2
8 : 1	3
16 : 1	4

examples:

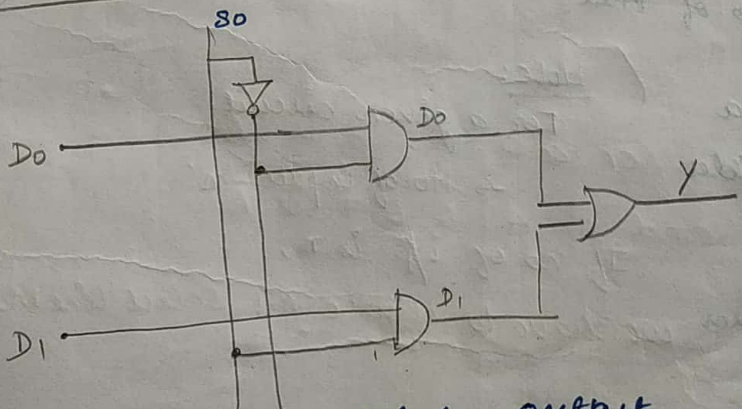
1 ⟹ 2:1 Mux



$S_0 = 0$ $Y = D_0$

$S_0 = 1$ $Y = D_1$

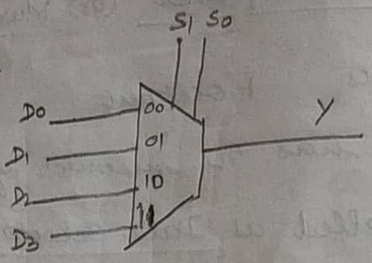
LOGIC DIAGRAM:

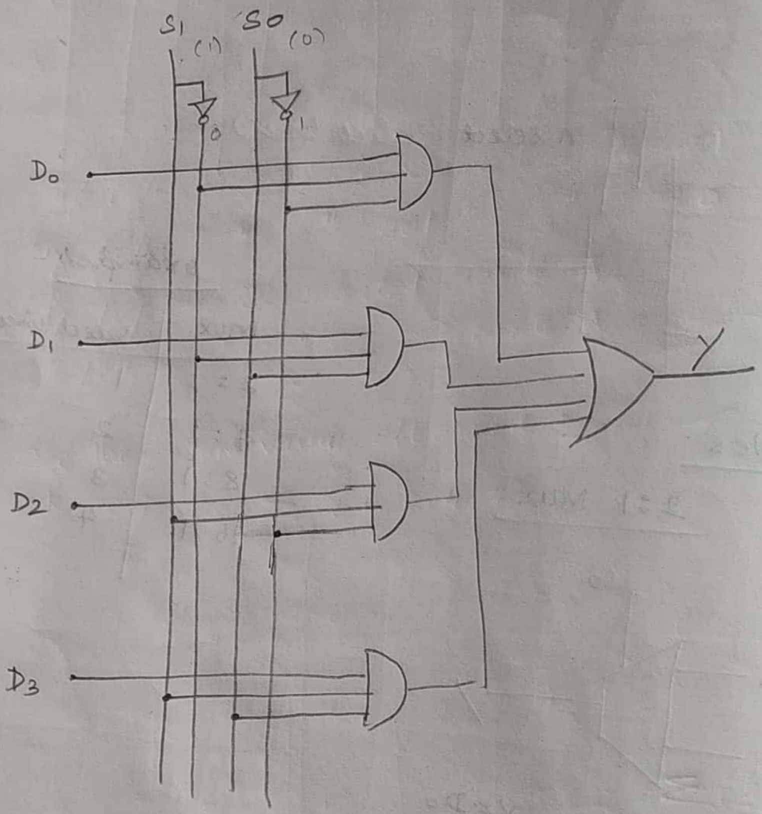


Select	Output
S_0	Y
0	D_0
1	D_1

TRUTH TABLE

2 ⇒ 4:1 MUX





Advantage:

We can design any digital ckt with the help of Mux

Q. Design a full adder using Multiplexer

(Note:
For a given circuit
→ No of inputs 'n'
If no of i/p is n,
we have to choose n-1 select line Mux)

⊖

From the truth table of full adder

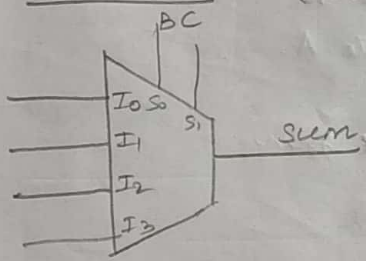
$sum = \Sigma(1, 2, 4, 7)$

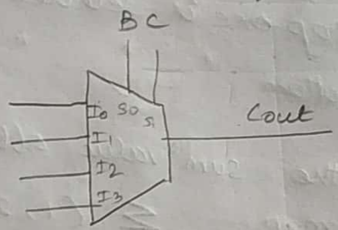
$cout = \Sigma(3, 5, 6, 7)$

i/p → 3 | output
(A B C) | sum, carry.

2 select lines → so choose 4:1 MUX
(they have 2 select lines)

4:1 MUX (we use separate mux for sum & carry)





For Implementation table draw the truth table of full adder first.

		A	B	C	sum	carry
			select lines			
$\bar{A}$	0	0	0	0	0	0
	1	0	0	1	1	0
	2	0	1	0	1	0
	3	0	1	1	0	1
A	4	1	0	0	1	0
	5	1	0	1	0	1
	6	1	1	0	0	1
	7	1	1	1	1	1

Implementation table for sum.

	I_0	I_1	I_2	I_3
$\bar{A}$	0	1	2	3
A	4	5	6	7
	A	$\bar{A}$	$\bar{A}$	A

② If both the nos are circled write as 1
If both " " " not " " as 0
If top one is selected write $\bar{A}$
" bottom " " " write A

① See the sum value obtained from k map $\Sigma(1, 2, 4, 7)$

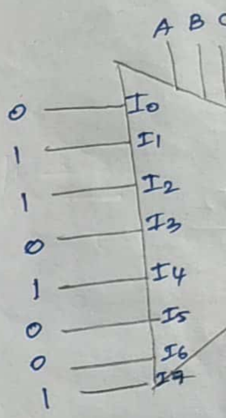
Implementation table for carry

	I_0	I_1	I_2	I_3
$\bar{A}$	0	1	2	3
$A.$	~~3~~ 4	5	6	7
	0	A	A	1

$\Sigma(3, 5, 6, 7)$
(from full addr

always for a 0, 1 2 3 is A
and 4 5 6 7 is $\bar{A}$

Using 8



Q. $F_1 = \Sigma(0, 1$
$F_2 = \Sigma(1$

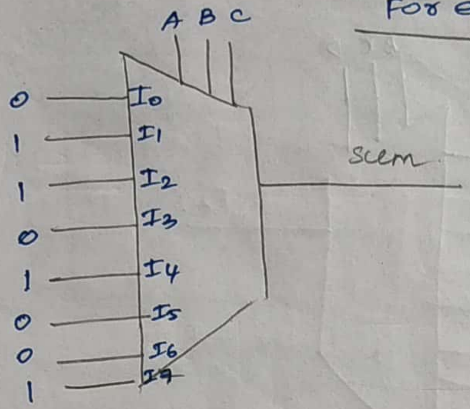
⇒ i/p –

8:1 MU

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

Using 8:1 MUX

For example



Q. $F_1 = \Sigma(0, 1, 3, 5, 7, 9, 11, 12)$

$F_2 = \Sigma(1, 5, 7, 11, 12, 13, 15)$

$\Rightarrow$ i/p $\rightarrow$ 4 $\Rightarrow n$

no of select lines $n-1 = 3$

(choose 8:1 MUX)

8:1 MUX

		A	B	C	D
$\bar{A}$	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
A	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

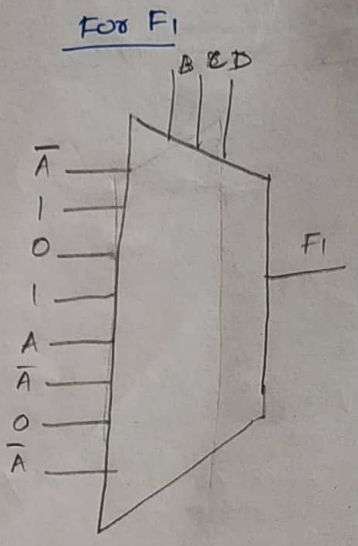
For F_1

	I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7
$\bar{A}$	0	1	2	3	4	5	6	7
A	8	9	10	11	12	13	14	15
	$\bar{A}$	1	0	1	A	$\bar{A}$	0	$\bar{A}$

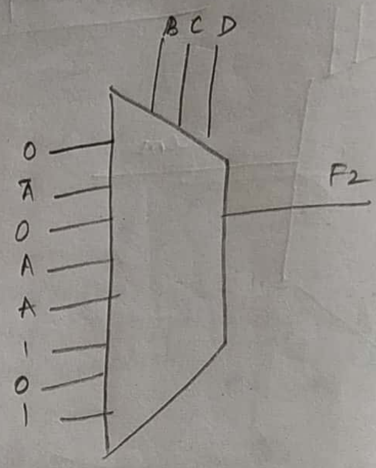
For F_2

	I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7
$\bar{A}$	0	1	2	3	4	5	6	7
A	8	9	10	11	12	13	14	15
	0	$\bar{A}$	0	A	A	1	0	1

For F1



For F2



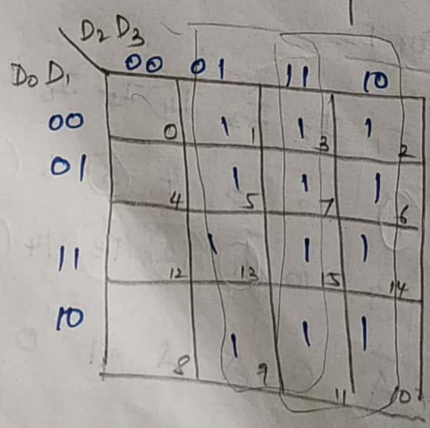
4/9/18

Priority Encoder.

4:2

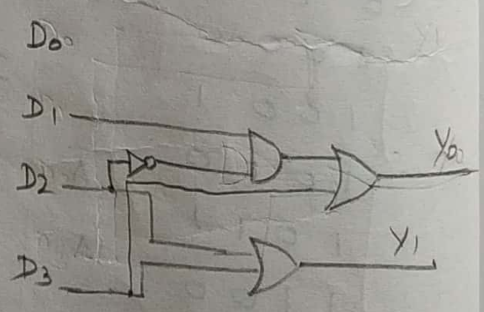
	D_0	D_1	D_2	D_3	Y_1	Y_0
8	1	0	0	0	0	0
4, 12	x	1	0	0	0	1
2, 6, 10, 14	x	x	1	0	1	0
1, 3, 5, 7, 9, 11, 13, 15.	x	x	x	1	1	1

$Y_1 \Rightarrow$



$Y_1 = D_3 + D_2$

LOGIC Diagram



For Y_0

$\Rightarrow$ 8:3

$Y_2 = \Sigma$

$Y_1 = \Sigma(2,$

$= D_2 \bar{D}$

$Y_0 = \Sigma($

$= D_7$

Draw

For Y_0

D_0D_1 \ D_2D_3	00	01	11	10
00		1	1	
01	1	1	1	
11	1	1	1	
10		1	1	

in priority encoder we get simultaneous encoder

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

⇒ **8 : 3 PE**

	D_0	D_1	D_2	D_3	D_4	D_5	D_6	D_7	Y_2	Y_1	Y_0
	128	64	32	16	8	4	2	01			
0	1	0	0	0	0	0	0	0	0	0	0
1	x	1	0	0	0	0	0	0	0	0	1
2	x	x	1	0	0	0	0	0	0	1	0
3	x	x	x	1	0	0	0	0	0	1	1
4	x	x	x	x	1	0	0	0	1	0	0
5	x	x	x	x	x	1	0	0	1	0	1
6	x	x	x	x	x	x	1	0	1	1	0
7	x	x	x	x	x	x	x	1	1	1	1

expression write from MSB

$$Y_2 = \Sigma(4,5,6,7)$$

$$= D_4 + D_5\overline{D_6}\,\overline{D_7} + D_6\overline{D_7} + D_7$$

$$Y_1 = \Sigma(2,3,6,7)$$

$$= D_2\overline{D_3}\,\overline{D_4}\,\overline{D_5}\,\overline{D_6}\,\overline{D_7} + D_3\overline{D_4}\,\overline{D_5}\,\overline{D_6}\,\overline{D_7} + D_6\overline{D_7} + D_7$$

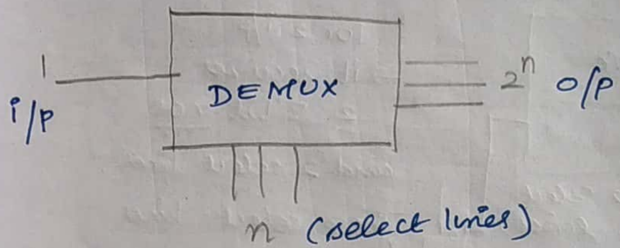
$$Y_0 = \Sigma(1,3,5,7)$$

$$= D_7 + D_5\overline{D_6}\,\overline{D_7} + D_3\overline{D_4}\,\overline{D_5}\,\overline{D_6}\,\overline{D_7} + D_1\overline{D_2}\,\overline{D_3}\,\overline{D_4}\,\overline{D_5}\,\overline{D_6}\,\overline{D_7}$$

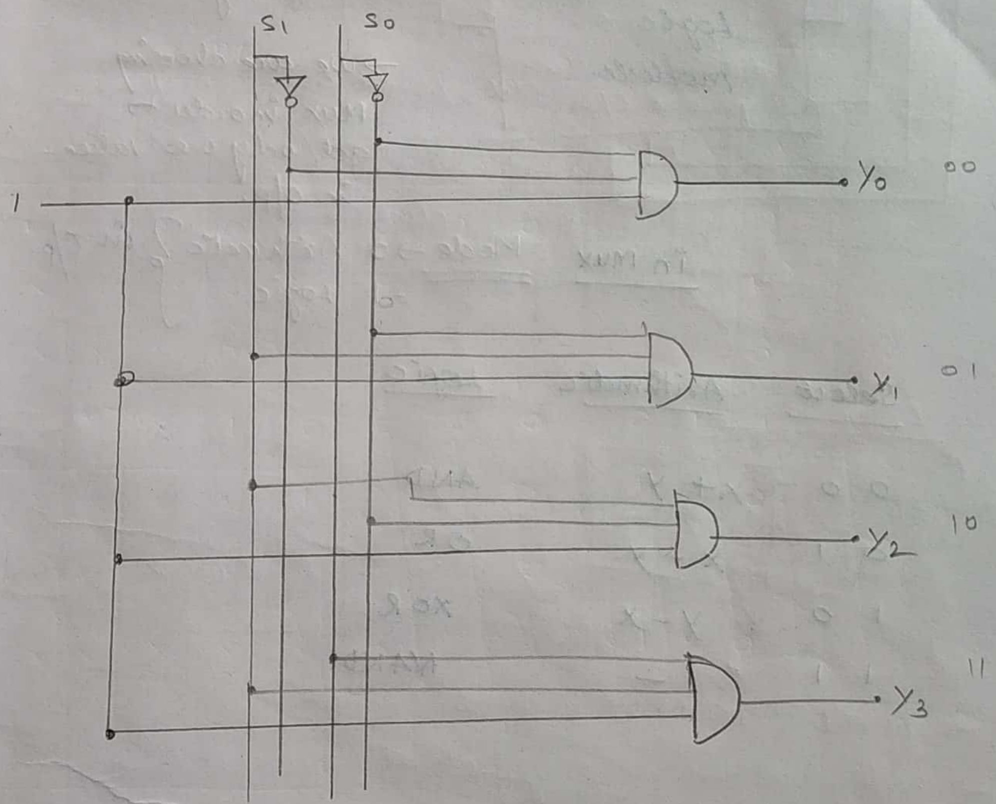
Draw logic diagram :-

Scanned by CamScanner

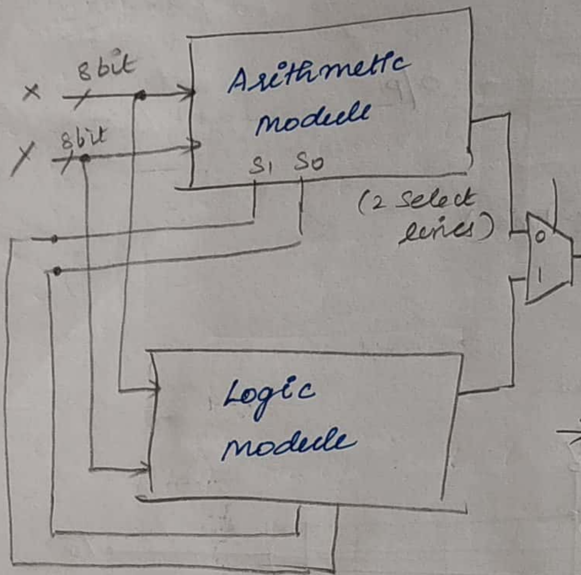
Demultiplexer



1:4 Demux



8-bit ALU:-



For 2 i/p
→ it receives 2 8'bit nos
and 2 select lines
These select lines are fed to the select lines of Logic module.

→ We are choosing Mux inorder to get only one value in o/p.

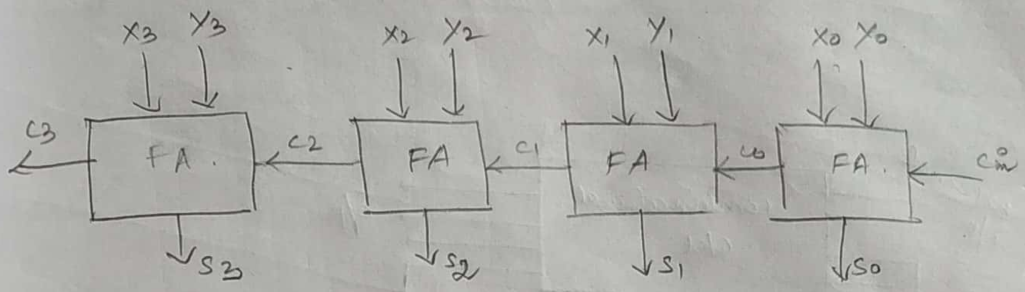
in Mux Mode → 0 Arithmetic
= 1 Logic } in o/p

Select	Arithmetic	Logic
0 0	x + y	AND
0 1	x − y	OR
1 0	y − x	XOR
1 1	—	NAND

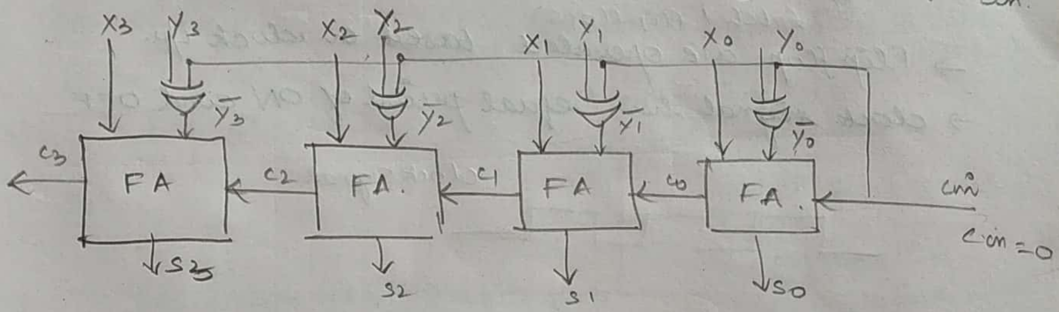
For performing Arithmetic operations :- x + y and x − y.

Using 4 bit → construct 4 F.A (RCA)

4 bit Adder & Subtractor



For X+Y & X-Y



	cin	y	o/p
x+y	0	0	0
	0	1	1
x−y	1	0	1
	1	1	0

When cin = 0

$x + y + 0$

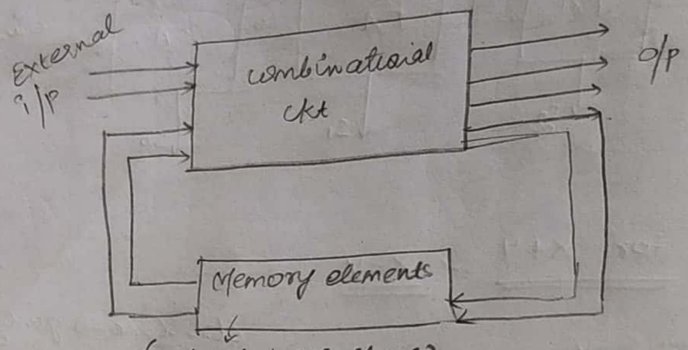
When cin = 1 → 2's complement

$x + \bar{y} + 1$

$= x - y$

7.9.18

UNIT - 3 SYNCHRONOUS SEQUENTIAL CIRCUITS.



(latch / flipflops)

→ Flip flops are operated based on clock i/p.

→ clock signal has equal period of ON and OFF

clock signal.

LATCHES:-

→ Does not have clock signal

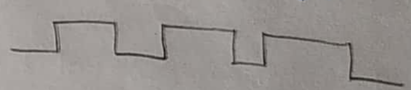
→ It has combination of gates

→ They are mostly used as feed back elements in asynchronous ckt

→ LEVEL TRIGGERED signals (Latches)

→ o/p responds during the level

→ Flip flops are edge triggered clock signal.

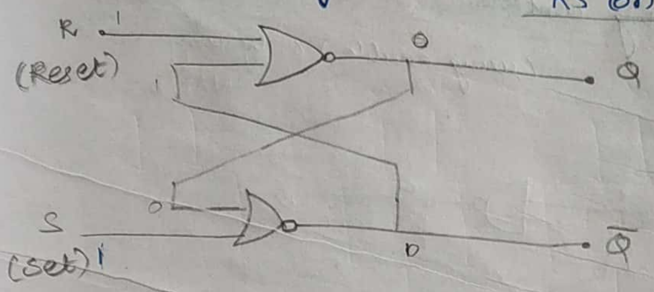


* SR latch (Basic latch)

NOR NAND

Using NOR

RS (or) Set Reset Latch



* When i/p is 0,0 the it will retain the previous o/p.

* If we give 1,0, it will change the previous.

NOR

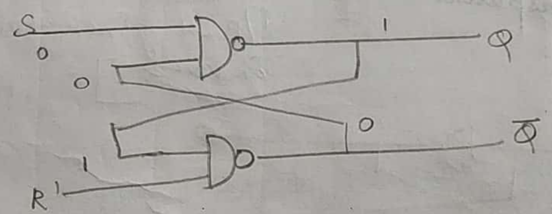
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

Truth table from the above diagram

R	S	Q	$\overline{Q}$	
0	1	1	0	
0	0	1	0	(the value is retained)
1	0	0	1	
0	0	0	1	
1	1	0	0	(invalid)

(in flip flop, it is indetermined state)

SR latch using NAND gate



initial case assume 0,1; depending upon o/p change the previous value.

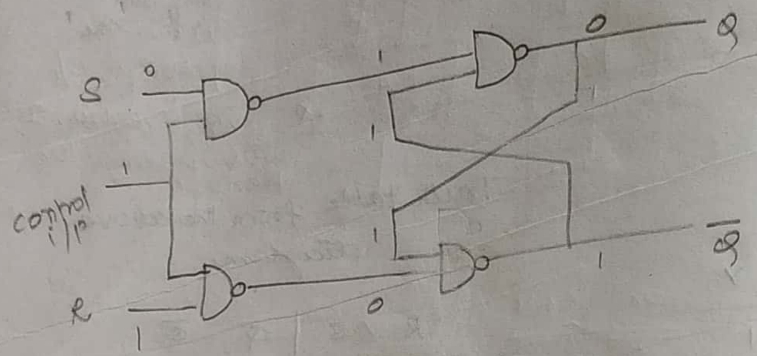
NAND

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

S	R	Q	$\overline{Q}$	
0	1	1	0	
1	1	1	0	(retain the previous value)
		0	1	
1	0			
1	1	0	1	
0	0	1	1	invalid

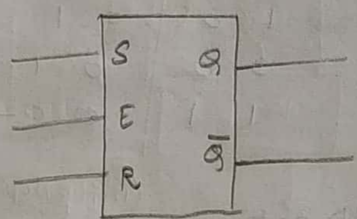
SR latch using controls.

(using enable) only using NAND.



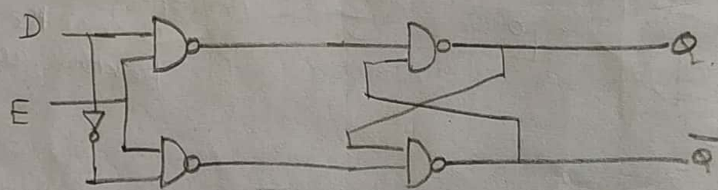
symbol of Latch

For '0' we put bubble



D LATCH :- (Transparent latch)

[In SR latch, one i/p is given to inverter.]



(Mostly 'D' flipflops is used for designing)

* Respond to changes given to level (in lab)
* concept of latch & D-flipflop is it follows the i/p (o/p follows i/p)

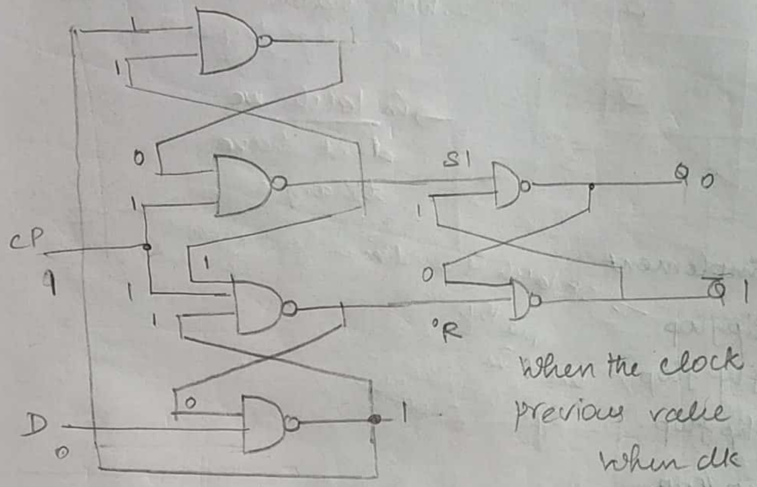
if D is 1, Q is 1
if " " 0, " " 0

if E is 0, $\overline{Q}$ is 0
if " " 1, $\overline{Q}$ " 1

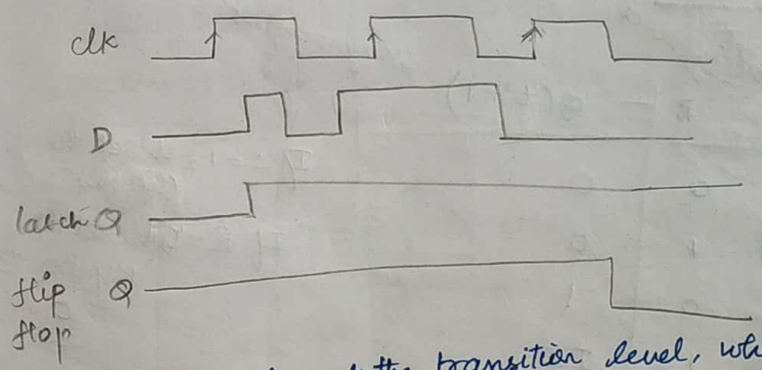
Positive edge triggered D-flip flop.

→ one of i/p is 0, the o/p is 1 (first part)

both i/p => 1 o/p is 0



When the clock pulse is 0, previous value is retained
when clk pulse is 1 there is a change, output follows the i/p



→ Only at the transition level, when the o/p changes, it is called as flipflop.

TRUTH TABLE. for D flip flop (characteristic Table)

D	Q
0	0
1	1

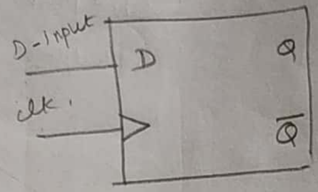
before clock

assume clk transition is from low to high

Q(t)	D	Q(t+1)
0	0	0
0	1	1
1	0	0
1	1	1

D value retains as it is.

SYMBOL



→ Without bubble +ve edge trigger
→ With bubble -ve edge trigger.
→ in latch we dont have triangle

We can implement

S-R flipflop → SR latch is used in asynchronous circuit.
J-k flip flop
(Toggle) T flip flop.
D flip flop

} mostly used in synchronous circuits.

SR FLIP FLOP

Using NOR gate.

Q	S (set)	R (Reset)	Q(t+1)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	I.D.
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	I.D

If S=1, it set the flip flop
If R=1, it reset the flipflop 0

General characteristic table of SR Latch.

S	R	Q(t)
0	0	Q(t)
0	1	0 → (Reset)
1	0	1 → set
1	1	Indetermined

JK FLIP flop

J	K	Q(t+1)
0	0	Q(t)
0	1	0
1	0	1
1	1	$\overline{Q(t)}$

Q	J	K	Q(t+1)	
0	0	0	0	0
0	0	1	0	1
0	1	0	1	2
0	1	1	1	3
1	0	0	1	4
1	0	1	0	5
1	1	0	1	6
1	1	1	0	7

T flipflop is implemented from JK flipflop to avoid racing condition (complemented form)

Apply k map for Q(t+1)

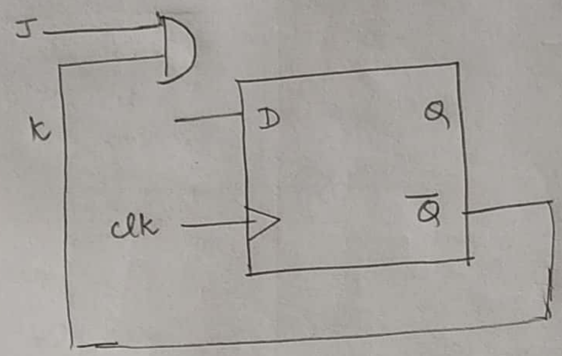
For J-K

Q \ JK	$\bar{J}\bar{K}$ 00	$\bar{J}K$ 01	JK 11	$J\bar{K}$ 10
$\bar{Q}$ 0	0	0	1	1
Q 1	1	0	0	1

$\Rightarrow \bar{Q}J + Q\bar{K}$ characteristic eqn of JK flip flop

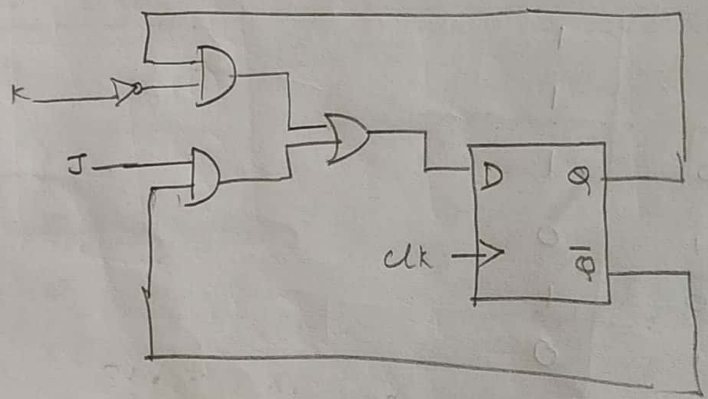
$\Rightarrow Q(t+1) = \overline{Q(t)}J + Q(t)\cdot \bar{K}$

$= \bar{Q}J + Q\bar{K}$



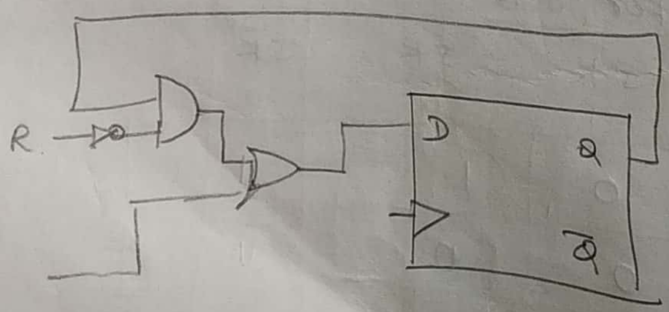
JK flip flop D-flip flop.

char eqn of D flip flop is D for $Q(t+1)$
(since it follows the i/p)



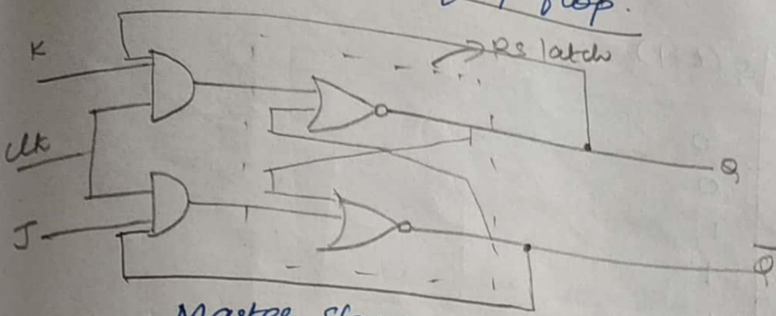
Q \ SR	$\bar{S}\bar{R}$	$\bar{S}R$	SR	$S\bar{R}$
$\bar{Q}$	0	1	x	1
Q	1		x	1

$Q(t+1) = S + Q\bar{R}$



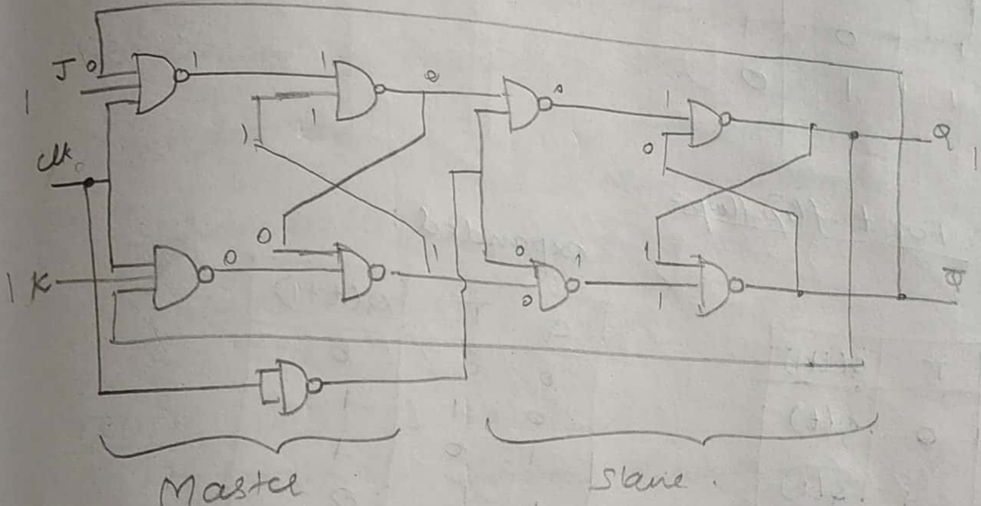
11.09.18

JK flip flop.



J	K	Q(t+1)
0	0	Q(t)
0	1	0
1	0	1
1	1	$\overline{Q(t)}$

Master slave JK flip flop.



→ Master receives external i/p
→ Mostly 1 1 condition is used to avoid racing condition

⟶ During -ve pulse cycle there is no change in the MASTER.

→ no change in slave (positive)
change in Master.

→ NO RACING CONDITION (negative)
No change in master
change in slave.

2 flip flop connected in cascade, it is master flip flop.

Q	J	K	Q(t+1)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

For t flip flop:-

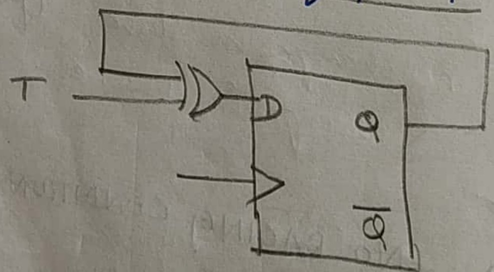
T	Q(t+1)
0	$Q(t)$
1	$\overline{Q(t)}$

expanded:

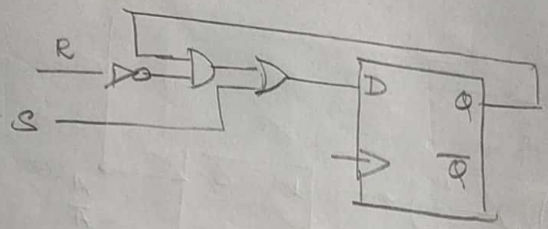
Q	T	Q(t+1)
0	0	0
0	1	1
1	0	1
1	1	0

$$Q(t+1) = \overline{Q}T + Q\overline{T}$$
$$= Q \oplus T$$

T flip flop from D flip flop.



SR from D



Synchronous seq. ckt.

Analysis

⇒ circuit or Equations
↓
Equations
↓
State table
↓
State diagram

Design.
↓ (From problem specification)
State diagram
↓
State table
↓
Transition Table.
↓
Equations
↓
logical Diagram.

1. TRUTH TABLE:

For known i/p, ~~for gat~~ depending upon the type of gate o/p is generated

2. STATE TABLE:

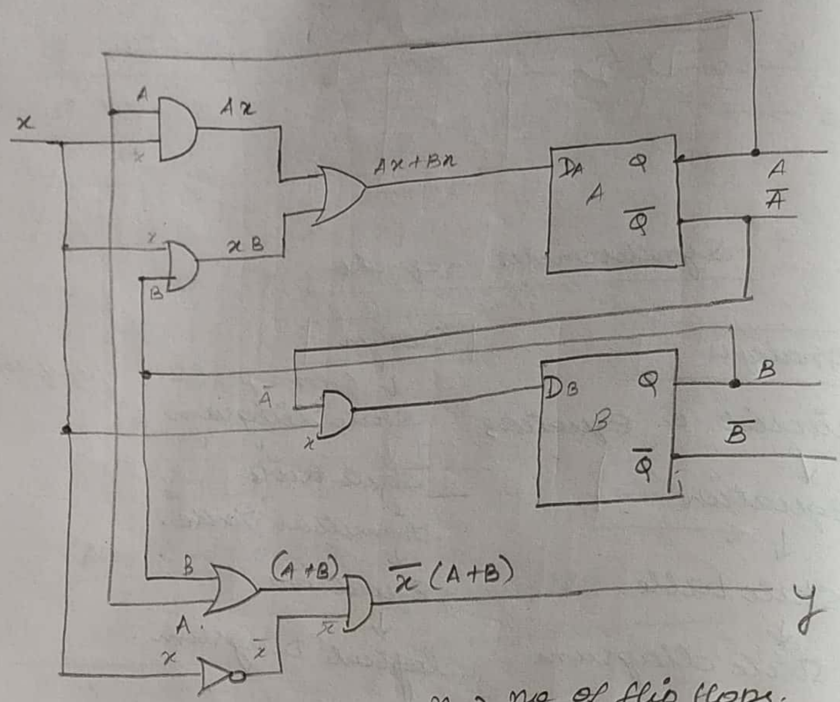
o/p for various i/p + Present state, ~~past~~ next State is obtained.

3. Transition table:

4. Excitation table: o/p is known, $Q(t)$ and $Q(t+1)$ is also known,
present state / next state
flip flop inputs are found

5. Characteristic table:

Next state value, we have to find. i/p & present states are known.



$n \rightarrow$ no of flip flops.

$2^n \rightarrow$ no of states.

Equations.

State equations:

$$A(t+1) = D_A = xA + xB = x(A+B)$$

$$B(t+1) = D_B = x\bar{A}$$

Output Equation:

$$y = \bar{x}(A+B)$$

State table: PS → present state (depends on 2 flip flops)

00 → a
01 → b
10 → c
11 → d

P S A	P S B	x	N.S A(t+1)	N.S B(t+1)	output $y = \bar{x}(A+B)$
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	1
0	1	1	1	1	0
1	0	0	0	0	1
1	0	1	1	0	0
1	1	0	0	0	1
1	1	1	1	0	0

Final State tabl

P S AB	N S x=0
a	a
b	a
c	a
d	a

State Diagr

change

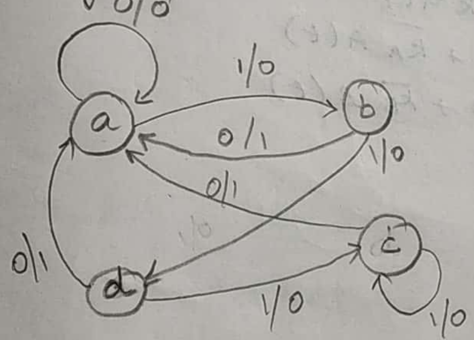
Final State table ⇒ Mealy

P S	N S		Output y	
AB	$x=0$	$x=1$	$x=0$	$x=1$
a	a	b	0	0
b	a	d	1	0
c	a	c	1	0
d	a	c	1	0

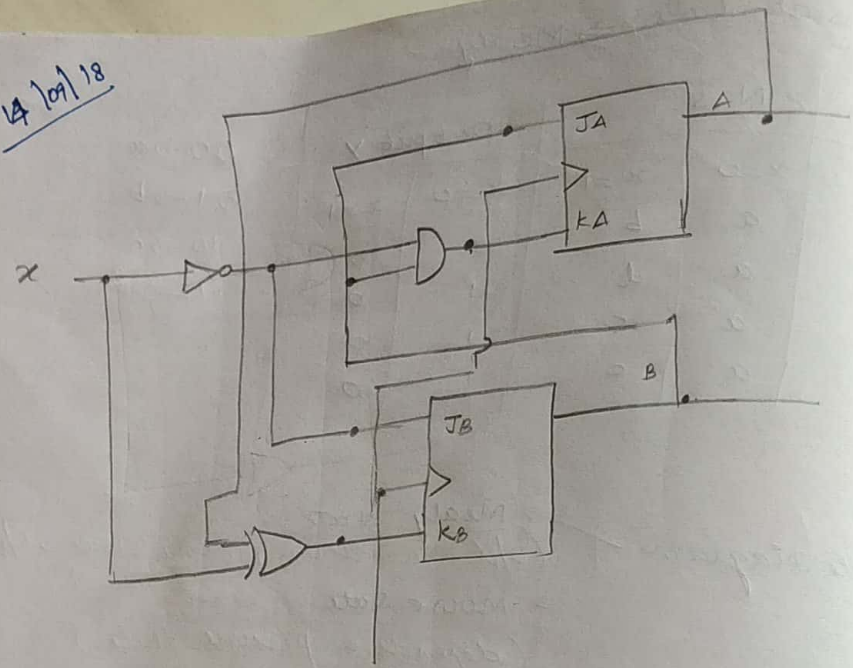
00 → a
01 → b
10 → c
11 → d

State Diagram → Mealy state diagram (o/p depends on present state, i/p)
→ Moore state diagram (depends on present state)

change in state → arrow (represented)



14/09/18



$$Q(t+1) = J\bar{Q}(t) + \bar{K}Q(t)$$

$$A(t+1) = J_A\bar{A}(t) + \bar{K}_A A(t)$$

$$B(t+1) = J_B\bar{B}(t) + \bar{K}_B B(t)$$

$J_A = B$

$J_B = \bar{x}$

$K_A = \bar{x}B$

$\bar{K}_A = \overline{\bar{x}B}$

$= x + \bar{B}$

$K_B = \cancel{\bar{x} \oplus \bar{A}} \quad x \oplus A$

$\bar{K}_B = \bar{x} \oplus \bar{A}$

$= xA + \bar{x}\bar{A}$

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

$B(t+1) = \bar{x}\bar{B}(t) + (xA + \bar{x}\bar{A})B(t)$

$$A(t+1) = B\bar{A}(t) + (x+\bar{B})A(t)$$
$$= B\bar{A}(t) + xA(t) + \bar{B}A(t)$$
$$= B\bar{A}(t) + \bar{B}A(t) + xA(t)$$
$$= \cancel{B \odot \bar{A}(t) + xA(t)}$$
$$= B \oplus A(t) + xA(t)$$

$$B(t+1) = \bar{x}\bar{B}(t) + xAB(t) + \bar{x}\bar{A}B(t)$$
$$= \bar{x}\bar{B}(t) + B(t)(x \odot A)$$

State Table:

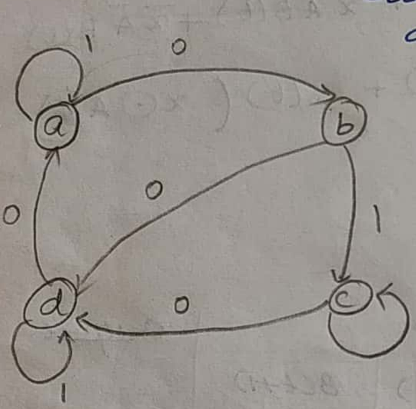
P	S	x	NS		~~Output~~.
A	B	x	A(t+1)	B(t+1)	
0	0	0	0	1	
0	0	1	0	0	
0	1	0	1	1	
0	1	1	1	0	
1	0	0	1	1	
1	0	1	1	0	
1	1	0	0	0	
1	1	1	1	1	

New state table:

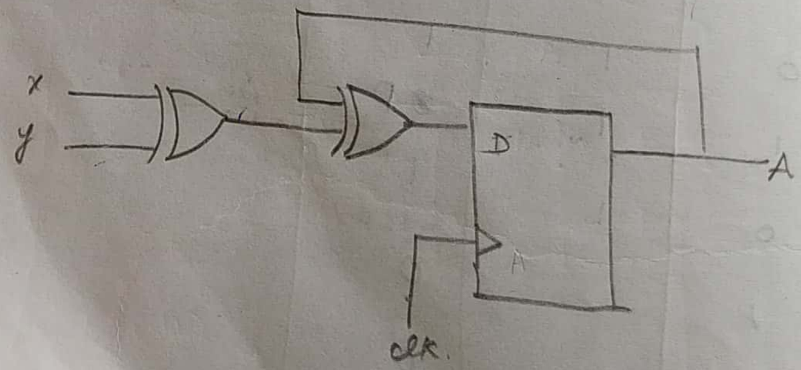
~~ou~~

PS	NS	
	$x=0$	$x=1$
a	b	a
b	d	c
c	d	c
d	a	d

State diagram



2.



$A(t+1) = D$

D flip flop.

$D = (x \oplus y) \oplus A$

$= A \oplus (x \oplus y)$

$= A \oplus x \oplus y.$

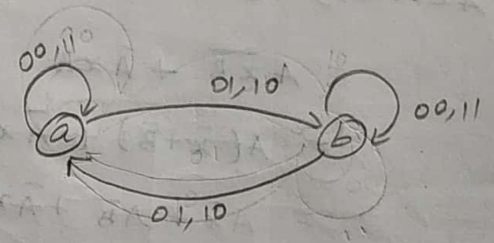
State Table:

	A	x	y	$Q(t+1) = D$
A	0	0	0	0
	0	0	1	1
	0	1	0	1
	0	1	1	0
B	1	0	0	1
	1	0	1	0
	1	1	0	0
	1	1	1	1

~~Final State~~:

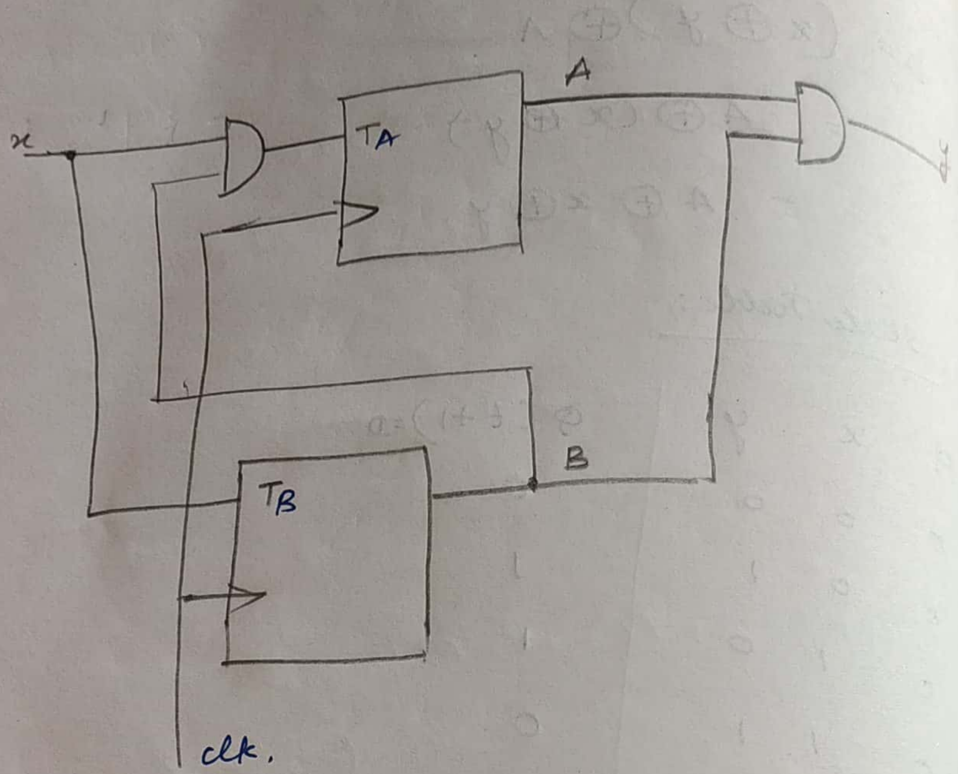
~~PS~~

State diagram



PS	NS			
A	xy=00	01	10	11
0	a	b	b	a
1	b	a	a	b

3.



$$Q(t+1) = Q \oplus T$$

$$A(t+1) = A \oplus T_A$$

$$B(t+1) = B \oplus T_B.$$

$$T_A = xB$$

$$T_B = x$$

$$A(t+1) = A \oplus xB$$

$$= A\overline{xB} + \bar{A}xB$$

$$= A(\bar{x} + \bar{B}) + \bar{A}xB$$

$$= A\bar{x} + A\bar{B} + \bar{A}xB.$$

$$y = A \cdot B$$

$$B(t+1) = B \oplus x$$

$$= B\bar{x} + \bar{B}x.$$

Scanned by CamScanner

Moore's state Table: II

P	S	N.S				o/p y
		x=0		x=1		
A	B	A	B	A	B	
0	0	00		b		0
0	1	01 b		c		0
1	0	10		d		0
1	1	11 d		a		1

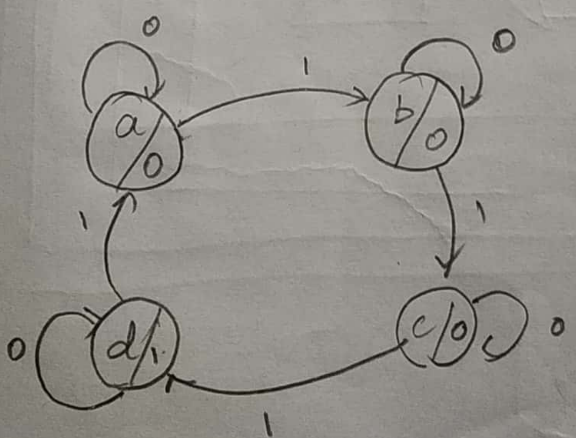
output is represented in moore's state table.

Table I.

	P.S			N.S		
	A	B	x	A(t+1)	B(t+1)	y = A . B
a	0	0	0	0	0	0
a	0	0	1	0	1	0
b	0	1	0	0	1	0
b	0	1	1	1	0	0
c	1	0	0	1	0	0
c	1	0	1	1	1	0
d	1	1	0	1	1	1
d	1	1	1	0	0	1

Moore's state diagram.

it depends only on B.



4. A sequential circuit has 2 JK flipflops `A' and `B', 2 inputs `x' and `y' and 1 output z. The flipflop i/p equations and ckt o/p eqns are

$$JA = \boxed{Bx + B'y'}$$

$$KA = B'xy'$$

$$JB = A'x$$

$$KB = A + xy'$$

$$z = Ax'y' + Bx'y' \Rightarrow (A+B)x'y'$$

(i) Draw the logic diagram of the ckt

(ii) Tabulate the state table

(iii) Derive the state eqns for A and B.

(iv) Draw the state diagram.

refer JK flip flop.