Number systems

convert 10111_2 to decimal

10111_2

$1 \times 2^0 = 1$

$1 \times 2^1 = 2$

$1 \times 2^2 = 4$ $\Rightarrow 23_{10}$

$0 \times 2^3 = 0$

$1 \times 2^4 = 16$

② convert 1011101001_2

1011101001

$1 \times 2^0 = 1$

$0 \times 2^1 = 0$

$0 \times 2^2 = 0$ $\Rightarrow 745_{10}$

$1 \times 2^3 = 8$

$0 \times 2^4 = 0$

$1 \times 2^5 = 32$

$1 \times 2^6 = 64$

$1 \times 2^7 = 128$

$0 \times 2^8 = 0$

$1 \times 2^9 = 512$

128
128
256

256
256
512

Convert 43_{10} to binary

(Successive division method)

$$
\begin{array}{l|l}
2 & 43 \\
2 & 21 - 1 \\
2 & 10 - 1 \\
2 & 5 - 0 \\
2 & 2 - 1 \\
2 & 1 - 0 \\
\end{array}
$$

$\Rightarrow 101011_2$

Convert 200_{10} to binary

$$
\begin{array}{l|l}
2 & 200 \\
2 & 100 - 0 \\
2 & 50 - 0 \\
2 & 25 - 0 \\
2 & 12 - 1 \\
2 & 6 - 0 \\
2 & 3 - 0 \\
2 & 1 - 1 \\
\end{array}
$$

$\Rightarrow 11001000_2$

Binary Addition

Add 1011_2 & 110_2

$$
\begin{array}{r}
1011 \\
110 \\
\hline
10001_2
\end{array}
$$

Binary subtraction

Sub $100_2 - 01_2$

$$
\begin{array}{r}
100 \\
1 \\
\hline
11_2
\end{array}
$$

Hexadecimal Number System:-

Convert 11011110101110_2 to hexadecimal

$11,0111,1010,1110_2$

$3 \quad 7 \quad A \quad E \implies 37AE$.

Convert $4A8C_{16}$ to binary

$0100\,1010\,1000\,1100_2$

Convert $2C9_{16}$ to Decimal

$2C9_{16}$

$9 \times 16^0 \longrightarrow 9$

$C \times 16^1 \longrightarrow 12 \times 16 = 192$

$2 \times 16^2 \longrightarrow 512$

Result $= 713_{10}$

Convert Decimal to hexadecimal

423_{10}

$$\begin{array}{r|l} 16 & 423 \\ \hline 16 & 26 - 7 \\ \hline & 1 - 10 \end{array} \Rightarrow 1A7_{16}$$

Introduction to Boolean Algebra &

logic gates:

1) Associateve Law:- $(a * b) * c = a \times (b \times c)$
for all $a, b, c \in S$ set.

2) Commutative law: $a \times b = b \times a$ for all $a, b \in S$

3) Identity element: $I \times a = a \times I$
$I =$ either 0 or 1

4) Inverse : $a . a^{-1} = 0$

5) Distributive law:
$a \times (bc) = (ab) c$
$a(b+c) = ab + ac$

Axiomatic definition of boolean Algebra

In 1854 George boole developed an algebraic system now called Boolean Algebra.

In 1938 C.E. Shannon introduced 2 Variable or Value boolean algebra called switching algebra, it represent the properties of bistable electrical switching circuits.

formal definition of boolean algebra one can use E.V. huntington in 1904.

Boolean Algebra is an algebraic structure defined by a set of elements together with 2 binary operators.

Difference b/w boolean Algebra & ordinary algebra

1) Distributive law suit for boolean Algebra not for ordinary Algebra

2) Boolean Algebra don't have Subtraction & division operations.

3) Compliment operator available in boolean not in ordinary Algebra.

4) Boolean Algebra Constitute with 0 and 1 only.

Basic theorem and properties of Boolean Algebra:-

$a+0 = a$

$a+\bar{a} = 1$

$a+a = a$

$a+1 = 1$

$\bar{\bar{a}} = a$ Involution Theorem

$x+y = y+x$
Commutative

$a+(b+c) = (a+b)+c$
Associate

$a \cdot (b+c) = ab+ac$
Distributive

$a \cdot 0 = 0$

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

$a \cdot a = a$

$a \cdot 1 = 1$

$a \cdot 0 = 0$

$xy = yx$

$a(bc) = (ab)c$

$a+bc = (a+b)(a+c)$

Demorgan's law

$$\overline{(a+b)} = \bar{a}\cdot\bar{b} \quad , \quad \overline{ab} = \bar{a}+\bar{b}$$

Absorption law

$$x + xy = x \quad , \quad x(x+y) = x$$

Consensus theorem

SOP $ab + \bar{a}c + bc = ab + \bar{a}c$.

POS $(a+b)\cdot(\bar{a}+c)(b+c) = (a+b)(\bar{a}+c)$

Simplify the following function to a minimum numbers of literals.

① $x(\bar{x}+y) = x\bar{x} + xy = 0 + xy = xy$

② $x + \bar{x}y = (x+\bar{x})\cdot(x+y) = 1\cdot(x+y)$
$= (x+y)$

③ $(x+y)(x+\bar{y}) = x\cdot x + x\bar{y} + yx + y\cdot\bar{y}$
$= x + x\bar{y} + xy + 0$
$= x + xy + x\bar{y}$
$= x + x\bar{y} = x.$

④ $xy + \bar{x}z + yz$

$\Rightarrow xy + \bar{x}z + yz(x+\bar{x})$

$\Rightarrow xy + \bar{x}z + xyz + \bar{x}yz$

$\rightarrow xy[1+z] + \bar{x}z[1+y]$

$\Rightarrow xy + \bar{x}z$

⑤ $(x+y)(\bar{x}+z)(y+z) = (x+y)(\bar{x}+z)$

by duality function ref sum ④

find the complement of the functions

①

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

$F_c = \bar{F} = \overline{\bar{x}y\bar{z} + \bar{x}\bar{y}z}$

Apply Demorgan's Law

$F_c = \overline{\bar{x}y\bar{z}} \cdot \overline{\bar{x}\bar{y}z}$

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

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

② $F = x(\bar{y}\bar{z} + yz)$

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

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

$$= \bar{x} + \left[\overline{\bar{y}\bar{z}} \cdot \overline{yz}\right]$$

$$= \bar{x} + \left[(y+z) \cdot (\bar{y} + \bar{z})\right]$$

$$= \bar{x} + \left[0 + y\bar{z} + z\bar{y} + 0\right]$$

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

Canonical & standard forms:-

Sum of Product (SOP): Minterm - AND gate

Product of sum (POS): MaxTerm - OR gate

Sum of Minterms (Sum of Product Term)

Ex: $F = A+\bar{B}C$

Find out the minterms:

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

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

$ABC + AB\bar{C} + A\bar{B}C + A\bar{B}\bar{C} + \bar{B}CA + \bar{B}C\bar{A}$

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

$111 + 110 + 101 + 100 + 101 + 001$

$7+6+5+4+5+1$

$F(A,B,C) = \sum_m (1,4,5,6,7)$

Truth Table : $F = A + \bar{B}C$

A	B	$\bar{B}$	C	F	
0	0	1	0	0	0
0	0	1	1	1	1
0	[illegible]	0	0	0	2
0	1	0	1	0	3
1	0	1	0	1	4
1	0	1	1	1	5
1	1	0	0	1	6
1	1	0	1	1	7

Product of Max Term (Product of Sum)

Express the boolean function $F = xy + \bar{x}z$ as a product of maxterms

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

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

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

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

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

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

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

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

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

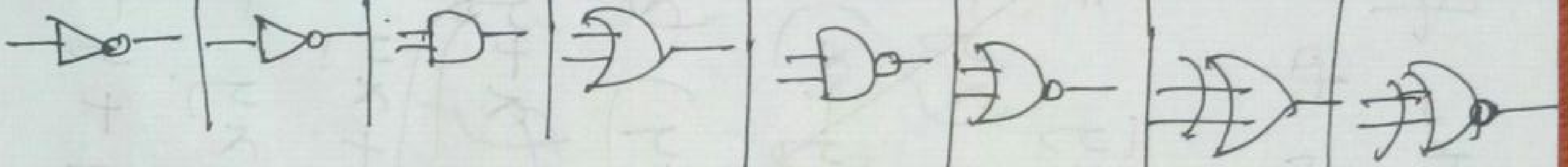
$F = 100, 101, 000, 010$

$F = M4, M5, M0, M_2$

$F(x, y, z) = \prod(0, 2, 4, 5)$

Digital logic gates

A	B	$\overline{A}$	$\overline{B}$	AB	$A+B$	$\overline{AB}$	$\overline{A+B}$	$A \oplus B$	$\overline{A \oplus B}$
0	0	1	1	0	0	1	1	0	1
0	1	1	0	0	1	1	0	1	0
1	0	0	1	0	1	1	0	1	0
1	1	0	0	1	1	0	0	0	1
		InV	InV	AND	OR	NAND	NOR	XOR	XNOR



Simplify the following Boolean expression to a minimum no of literals.

1) $xy + x\bar{y} \Rightarrow x(y+\bar{y}) \Rightarrow x$

$\Rightarrow (xy+x)(xy+\bar{y})$

$\Rightarrow (x+x)(y+x)(x+\bar{y})(y+\bar{y})$

$\Rightarrow x(x+y)(x+\bar{y})$

2) $(x+y)(x+\bar{y})$

$\Rightarrow xx + xy + x\bar{y} + y\bar{y}$

$\Rightarrow x + xy + x\bar{y} + 0$

$\Rightarrow x[1+y+\bar{y}] + 0$

$\Rightarrow x$

3) $xyz + \bar{x}y + xy\bar{z}$

$\Rightarrow xy(z+\bar{z}) + \bar{x}y$

$\Rightarrow xy + \bar{x}y \Rightarrow y(x+\bar{x}) \Rightarrow y$.

④ $(\overline{A+B})\ \overline{\bar{A}+\bar{B}}$

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

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

$\Rightarrow 0$

⑤ $xy\bar{z}+\bar{x}yz+xyz+\bar{x}y\bar{z}$

$\Rightarrow xy(\bar{z}+z)+\bar{x}y(z+\bar{z})$

$\Rightarrow xy+\bar{x}y \Rightarrow y(x+\bar{x}) \Rightarrow y$

⑥ $(x+y+\bar{z})(\bar{x}+\bar{y}+z)$

$\Rightarrow x\bar{x}+x\bar{y}+xz+y\bar{x}+y\bar{y}+yz+\bar{z}\bar{x}+\bar{z}\bar{y}+\bar{z}z$

$\Rightarrow 0+x\bar{y}+xz+\bar{x}y+0+yz+\bar{z}\bar{x}+\bar{z}\bar{y}+0$

$\Rightarrow x\bar{y}+xz+\bar{x}y+yz+\bar{z}\bar{x}+\bar{z}\bar{y}$

⑦ $xy+x\bar{y}+\bar{x}y$

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

$=x+y$

(8) $ABC + \bar{A}B + AB\bar{C} + AC$

$\Rightarrow AB(C+\bar{C}) + \bar{A}B + AC$

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

$\Rightarrow B + AC \Rightarrow (A+B)(B+C)$.

(9) $\bar{x}yz + xz + \bar{x}z$

$\Rightarrow \bar{x}z(y+1) + xz$

$\Rightarrow \bar{x}z + xz \Rightarrow z$

(10) $(\overline{x+y})(\bar{x}+\bar{y})$

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

$\bar{x}.\bar{y}\bar{x} + \bar{x}\bar{y}\bar{y}$

$\bar{x}\bar{y} + \bar{x}\bar{y} \Rightarrow \bar{x}\bar{y}$

(11) $xy + x(wz + w\bar{z})$

$\Rightarrow xy + xw$

$\Rightarrow x(y+w)$

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

$\Rightarrow B\bar{C}A\bar{B} + B\bar{C}C\bar{D} + \bar{A}DA\bar{B} + \bar{A}DC\bar{D}$

$\Rightarrow 0 + 0 + 0 + 0$

$\Rightarrow 0$

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

$\Rightarrow 0 + \bar{x}\bar{y} + \bar{x}\bar{z} + \bar{z} + \bar{z}\bar{y} + \bar{z}x$

$\Rightarrow 0 + \bar{x}\bar{y} + \bar{x}\bar{z} + \bar{z}\bar{y} + \bar{z}$

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

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

Reduce the following Boolean expressio to the indicated number of literals.

(1) $\bar{A}\bar{C} + ABC + A\bar{C} + A\bar{B}$ (2 literals)

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

$\Rightarrow \bar{C} + ABC + A\bar{B}\bar{C} + A\bar{B}C$

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

Use boolean algebra to simplify the function to a minimum no of literals

$F = x\bar{y}z + \bar{x}\bar{y}z + \bar{w}xy + w\bar{x}y + wxy$

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

$\Rightarrow \bar{y}z + xy + \bar{x}yw$

$\Rightarrow \bar{y}z + y[x + \bar{x}w]$

$\Rightarrow \bar{y}z + y[(x+\bar{x})(x+w)]$

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

Map Method (or) Karnaugh Map (or) K-Map

(i) simple, straight forwarded method.

2 inputs $\Rightarrow 2^n = 0, 1, 2, 3$

3 inputs $= 2^3 = 0, 1, 2, 3, 4, 5, 6, 7.$

4 inputs $= 2^4 = 0$ to 15

5 inputs $= 2^5 = 0$ to 32

Two Variable Map

	$\bar{b}$	b	
$\bar{a}$	0	1	0
a	2	3	1

	$\bar{b}$	b
$\bar{a}$	0	1
a	2	3

$$F = \bar{x}y + x\bar{y} + xy$$

$$= y[\bar{x} + x] + x\bar{y}$$

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

$$\Rightarrow (y+x)$$

	$\bar{y}$	y
$\bar{x}$	0	1 (1)
x	1 (2)	1 (3)

$$f = \bar{x}y + x\bar{y} + xy$$

$$= 01, 10, 11$$

$\rightarrow x \quad f = x + y$

$\downarrow y$

Simplify the boolean function

① $f(x, y, z) = \Sigma(2, 3, 4, 5)$

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

$\bar{x}y$

$x\bar{y}$

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

(2) $F(x,y,z) = \sum(3,4,6,7)$

$x \backslash yz$	$\bar{y}\bar{z}$	$\bar{y}z$	yz	$y\bar{z}$
$\bar{x}$	0	1	1 3	2
x	1 4	5	1 7	1 6

$\cancel{x}y$

$yz \implies yz + x\bar{z}$

$x\bar{z}$

$$F(x,y,z) = \sum(0,2,4,5,6) = \bar{z} + x\bar{y}$$

$x \backslash yz$	$\bar{y}\bar{z}$	$\bar{y}z$	yz	$y\bar{z}$
$\bar{x}$	1 0	1		1 2
x	1 4	1 5	7	1 6

$y\bar{z}$

$\bar{y}\bar{z}$ $\quad$ $x\bar{y}$ $\quad$ $\cancel{\bar{x}\bar{z}} + \cancel{y\bar{z}}$

$F(a,b,c) = \sum(1,2,3,5,7) = c + \bar{a}b$

$a \backslash bc$	$\bar{b}\bar{c}$	$\bar{b}c$	bc	$b\bar{c}$
$\bar{a}$	0	1 1	1 3	1 2
a	4	1 5	1 7	6

$\Rightarrow \bar{a}b$

c

Four Varible Map

$F(w,x,y,z) = \sum(0,1,2,4,5,6,8,9,12,13,14)$

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

$\Rightarrow \bar{w}\bar{z}$

$\Rightarrow x\bar{z}$

$\bar{y}$

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

Simplify the boolean function

$F = \bar{a}\bar{b}\bar{c} + \bar{b}c\bar{d} + \bar{a}bc\bar{d} + a\bar{b}\bar{c}$

ab \ cd	$\bar{c}\bar{d}$	$\bar{c}d$	cd	$c\bar{d}$
$\bar{a}\bar{b}$	1 (0)	1 (1)	1 (3)	(2)
$\bar{a}b$	(4)	(5)	(7)	1 (6)
ab	(12)	(13)	(15)	(14)
$a\bar{b}$	1 (8)	1 (9)	1 (11)	(10)

$\rightarrow \bar{a}bc\bar{d}$

$\bar{b}\bar{c}$ $\quad$ $\bar{b}d$

Simplify the Boolean function

$F(w,x,y,z) = \Sigma(1,3,7,11,15) + d(0,2,5)$

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

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

$\rightarrow \bar{w}z$

yz

$F(w,x,y,z) = \sum(1,3,10) + \sum_d(0,2,8,12)$

	$\bar y\bar z$	$\bar y z$	yz	$y\bar z$
$\bar w\bar x$	X 0	1 1	1 3	X 2
$\bar w x$	4	5	7	6
wx	X 12	13	15	14
$w\bar x$	X 8	9	11	1 10

$\bar w\bar x$

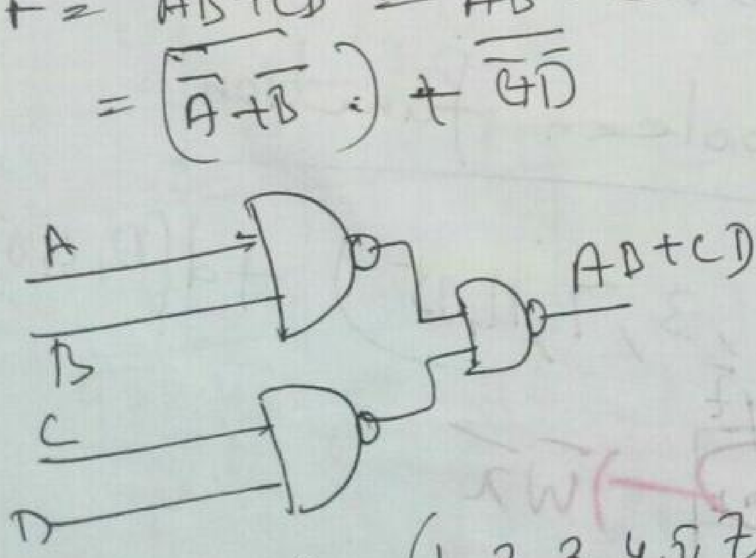
$\bar x\bar z$

NAND & NOR implementation

$F = AB + CD$

$F = \overline{\overline{AB+CD}} = \overline{\overline{AB}\cdot\overline{CD}}$ $\quad \overline{\bar A+\bar B \cdot \bar C+\bar D}$

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



$F(x,y,z) = (1,2,3,4,5,7)$

	$\bar y\bar z$	$\bar y z$	yz	$y\bar z$
$\bar x$	0	1 1	1 3	1 2
x	1 4	1 5	1 7	6

$\bar x y$

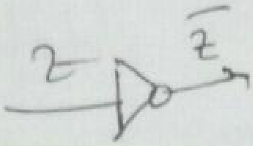
$x\bar y$

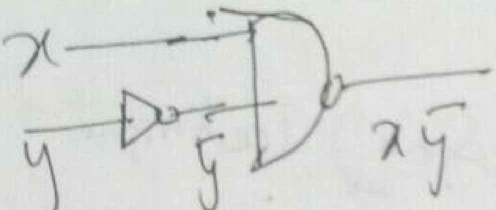
z

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

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

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





Implement the following Boolean function with NAND gates

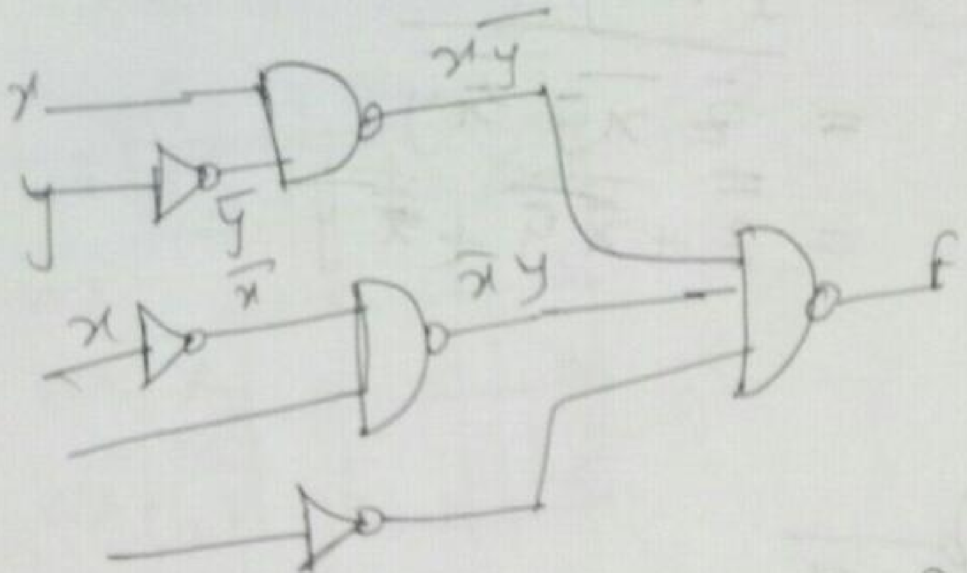
$F(x,y,z) = \sum m(1,2,3,4,5,7)$

$x \backslash yz$	$\bar{y}\bar{z}$	$\bar{y}z$	yz	$y\bar{z}$
$\bar{x}$	0	1 $_1$	1 $_3$	1 $_2$
x	1 $_4$	1 $_5$	1 $_7$	$_6$

$\bar{x}y$

$x\bar{y}$ $\quad z$

$$f = \overline{\overline{x\bar{y} + \bar{x}y + z}} = \overline{\overline{x\bar{y}} \cdot \overline{\bar{x}y}\;\bar{z}}$$



Quine Mc-cluskey (QM) Technique

For increased no of Variables QM method can be adopted.
QM method is more effective since a digital computer can be used to simplify the given boolean switching function.
Kmap is graphical method
QM method is a tabular method.

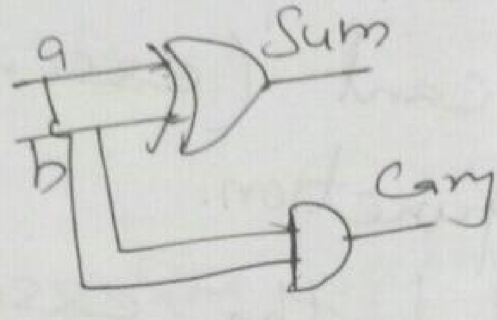
The basic QM cluskey procedure is as follows

1. Find the Prime implicants of the function
2. Construct the Prime implicant table and find the essential prime implicant (essential rows) of the function.
3. Include the essential prime implicants in the minimum sum.
4. ~~After~~ all essential prime implicants are deleted from the Prime implicant table, determine the dominated rows & dominated columns in the table, delete all dominated rows and dominating columns and find the (II^ry) essential prime implicants.
5. Repeat steps 3 & 4 as many times as they are applicable until a minimal cover of the function is found.

Implementation of Combinational logic Ckts using HA/FA using Verilog HDL

HA -> Half Adder

a	b	sum	carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1



Gate level modelling

```
module halfadder(a,b,sum,carry);
input a,b; output sum,carry;
xor(sum,a,b);
and(carry,a,b); endmodule
```

Data flow modelling

```
assign sum=a^b;
assign carry= a&b;
```

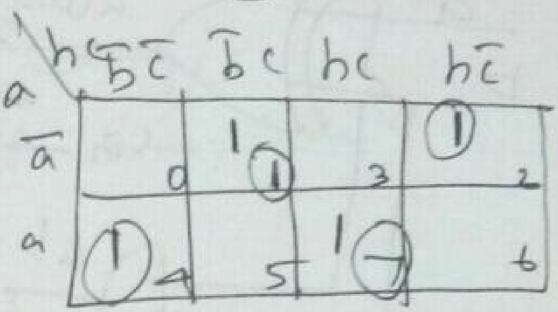
Behavioural

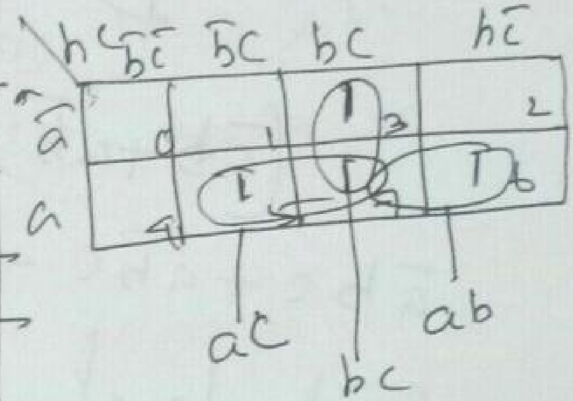
```
reg sum,carry;
always@(a,b)
begin
sum=a^b;
carry=a&b;
end
endmodule
```

FullAdder

	a	b	c	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

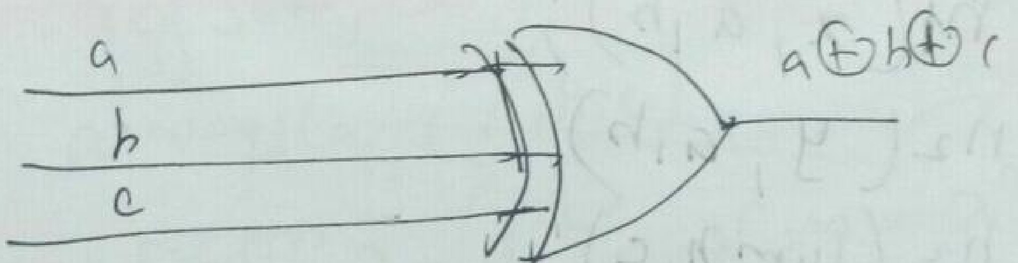


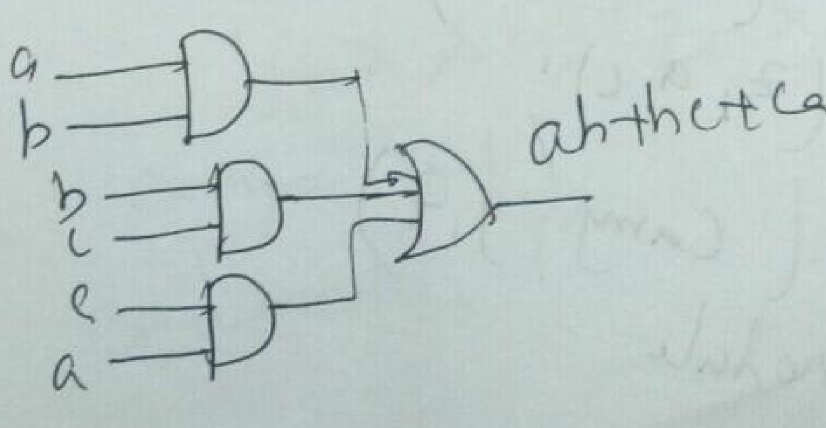


$ab+bc+ca = \text{Carry}$

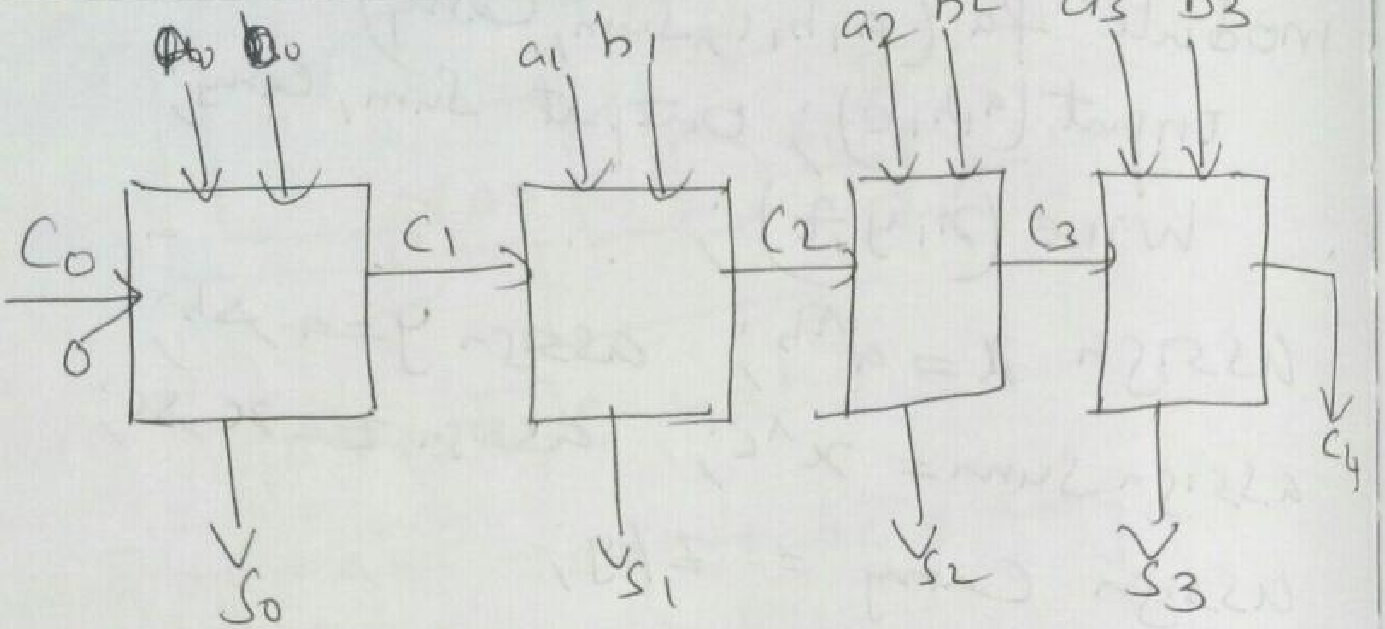
$\text{Sum} = \bar{a}\bar{b}c + \bar{a}b\bar{c} + a\bar{b}\bar{c} + abc$

$= a \oplus b \oplus c$





Binary Adder (4 bit) (RPA) ripple carry adder



A = 1 0 1 1   B = 0 0 1 1

a = 1 0 1 1   At C0 = 0

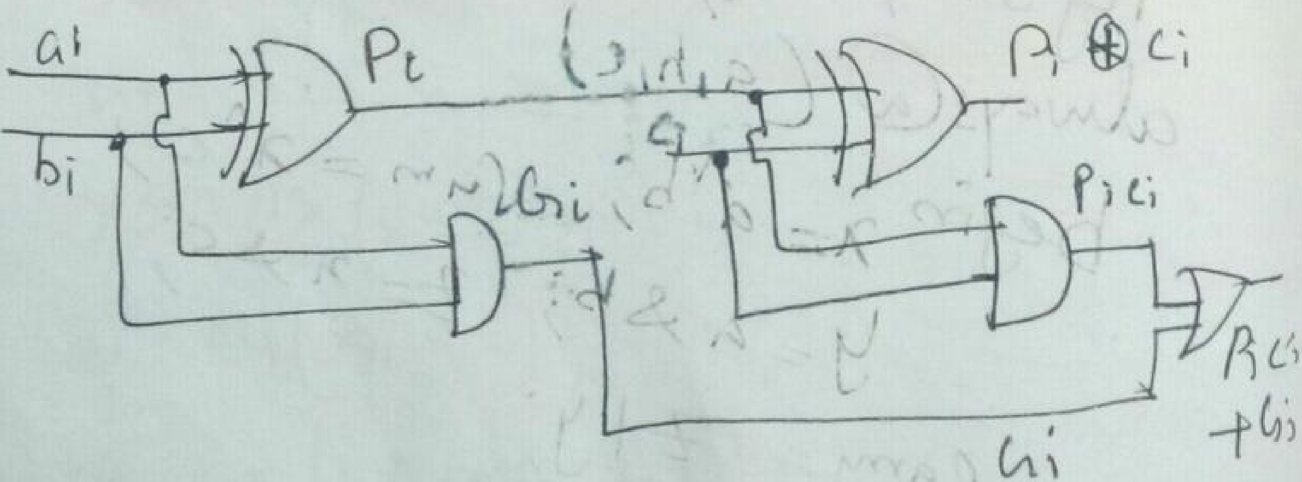
b = 0 0 1 1

S 1 1 1 0

C 0 0 1 1

a = augend

b = addend



C_0 = input carry **Carry lookahead generator**

$C_1 = G_0 + P_0 C_0$; $C_2 = G_1 + P_1 C_1$

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

$C_3 = G_2 + P_2 C_2 \Rightarrow G_2 + P_2 (G_1 + P_1 (G_0 + P_0 C_0))$

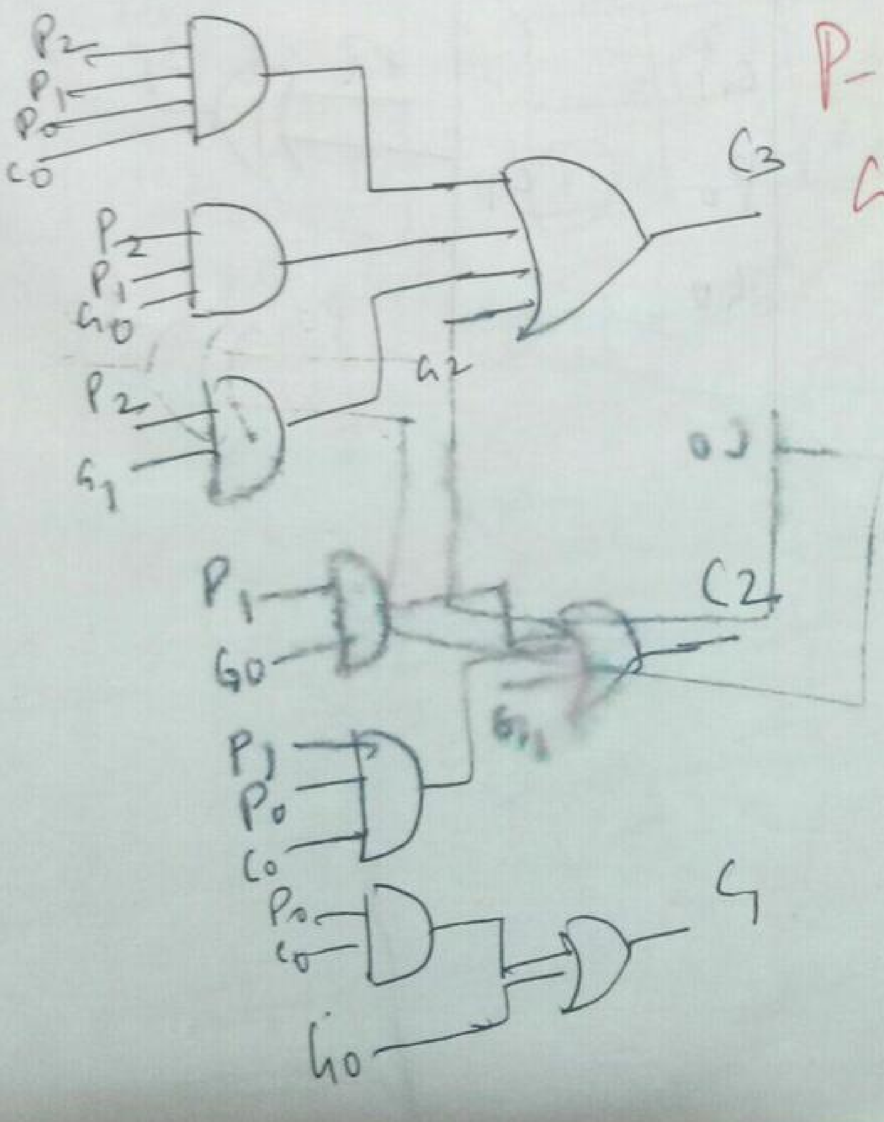
$= G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 C_0$

$P_i = A_i \oplus B_i$ $G_i = a_i b_i$ Sum $= P_i \oplus C_i$

Carry $= C_{i+1}$

Carry $=$

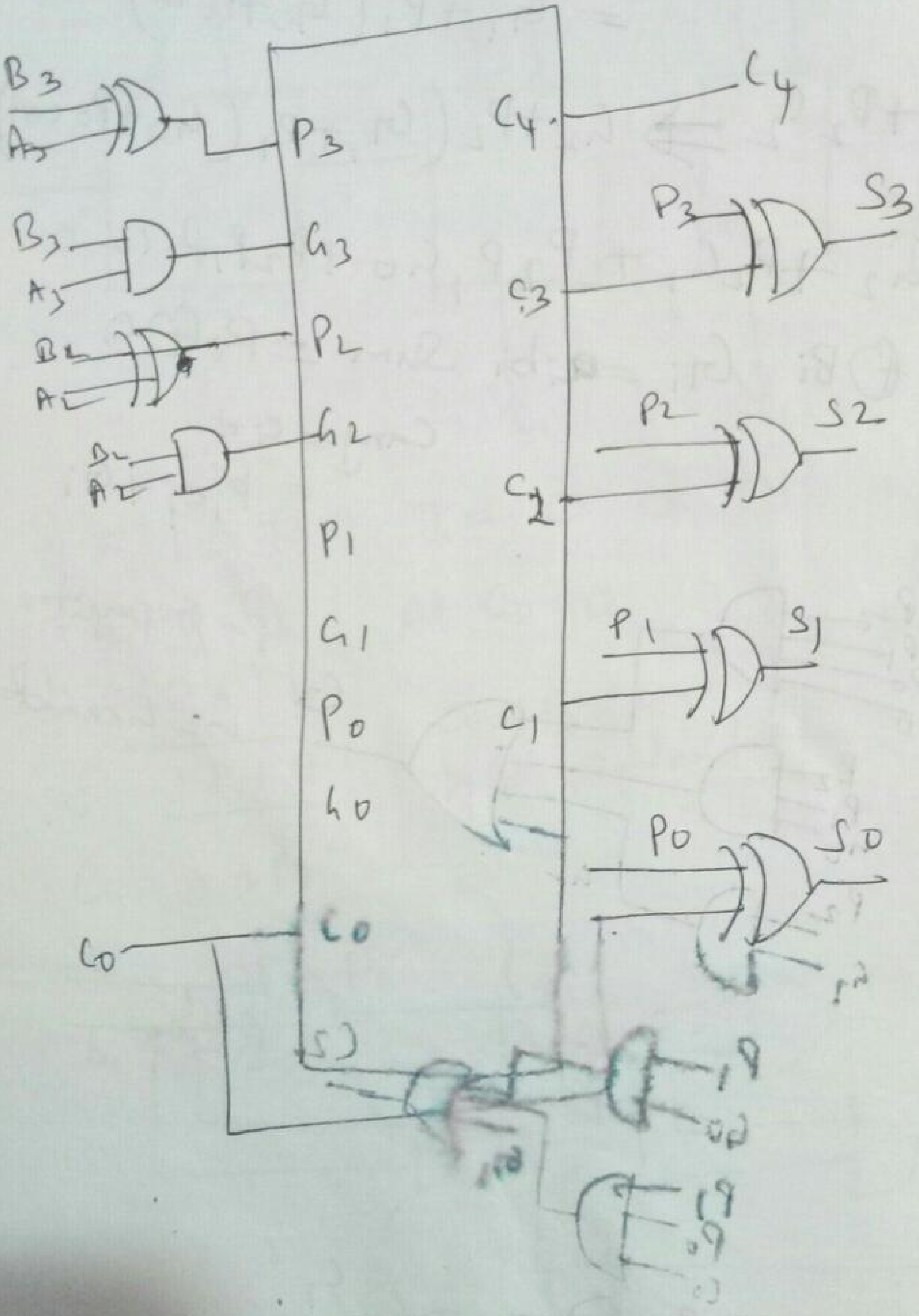
$= P_i C_i + G_i$



P- propagator

G= Generator

Fourbit adder with carry look ahead



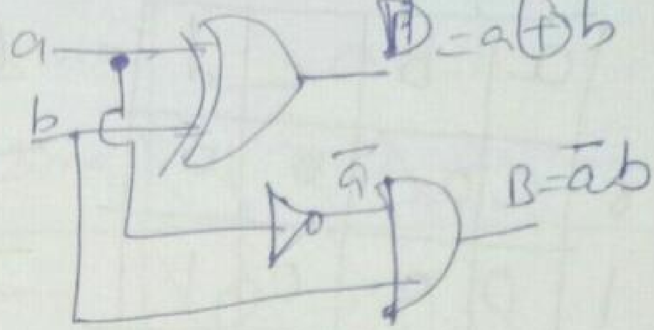
Half subtractor

a	b	B	D
0	0	0	0
0	1	1	1
1	0	0	1
1	1	0	0

Borrow = $B = \bar{a}b$

$\bar{a}b + a\bar{b} = a \oplus b$

difference $= a \oplus b$



Full subtractor

	LSB a	b	MSB c	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

a \ bc	$\bar{b}\bar{c}$	$\bar{b}c$	bc	$b\bar{c}$
$\bar{a}$	0	1 1	3	1 2
a	1 4	5	1 7	6

Diff $= \bar{a}\bar{b}c + \bar{a}b\bar{c} + a\bar{b}\bar{c} + abc$
$= a \oplus b \oplus c$

a \ bc	$\bar{b}\bar{c}$	$\bar{b}c$	bc	$b\bar{c}$
$\bar{a}$	0	1 1	1 3	1 2
a	4	5	1 7	6

$\bar{a}c$ $\quad$ $\bar{a}b$

Borrow $\quad bc$

Borrow $= \bar{a}c + \bar{a}b + bc$

Quine Mc cluskey method (Tabular method)

$$F(a,b,c,d) = \sum_m (0,1,3,7,8,9,11,15)$$

Step ①

	a	b	c	d
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

Step ②

Group	min terms	Binary code
0	m_0	0 0 0 0
1	m_1 m_8	0 0 0 1 1 0 0 0
2	m_3 m_9	0 0 1 1 1 0 0 1
3	m_7 m_{11}	0 1 1 1 1 0 1 1
4	m_{15}	1 1 1 1

step 3

Group	(Decimal no) minterms with matched pair	First Reduction
0	m0 & m1	0 0 0 —
	m0 & m8	— 0 0 0
1	m1 & m3	0 0 — 1
	m1 & m9	— 0 0 1
	m8 & m9	1 0 0 —
2	m3 & m7	0 — 1 1
	m3 & m11	— 0 1 1
	m9 & m11	1 0 — 1
3	m7 & m15	— 1 1 1
	m11 & m15	1 — 1 1

step 4		a b c d	
0	m0 & m1, m8 & m9	— 0 0 —	$\bar{b}\bar{c}$
	m0 - m8 - m1 - m9	— 0 0 —	
1	m1 - m3 - m9 - m11	— 0 — 1	$\bar{b}d$
	m1 - m9 - m3 - m11	— 0 — 1	
2	m3 - m7 - m11 - m15	— — 1 1	cd
3, 11, 7, 15	m3 - m11 - m7 - m15	— — 1 1	

Prime Implicant Step ⑤

Prime Implicant	Individual matched Pair	0	1	3	7	8	9	11
$\bar{B}\bar{c}$	0,1,8,9	X	X			X	X	
$\bar{b}d$	1,3,9,11		X	X			X	X
cd	3,7,11,15			X	X			X

$F = \cancel{\bar{a}\bar{b}\bar{c}\bar{d} + \bar{a}\bar{b}\bar{c}}\ ab\bar{c}\bar{d}$

$F = \bar{a}\bar{b}\bar{c}\bar{d} + a\bar{b}\bar{c}\bar{d} + \bar{a}bcd + abcd$

$= \bar{b}\bar{c}\bar{d} + bcd$

Ans. $\boxed{F = \bar{b}\bar{c} + cd}$

Minimization using Tabular method

$$f(a,b,c,d) = \sum_m (6,7,8,9) + d(10,11,12,13,14,15)$$

Step 1

Dec	a	b	c	d
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

Step 2

Group	minterm	Binary Representation
0	—	—
1	m_8	1000
2	m_6 m_9 m_{10} m_{12}	0110 1001 1010 1100
3	m_7 m_{11} m_{13} m_{14}	0111 1011 1101 1110
4	m_{15}	1111

Step III

Group	matced pairs	First Reduction
1	$m_8 - m_9$	1 0 0 –
	$m_8 - m_{10}$	1 0 – 0
	$m_8 - m_{12}$	1 – 0 0
2	$m_6 - m_7$	0 1 1 –
	$m_6 - m_{14}$	– 1 1 0
	$m_9 - m_{11}$	1 0 – 1
	$m_9 - m_{13}$	1 – 0 1
	$m_{10} - m_{11}$	1 0 1 –
	$m_{12} - m_{14}$	1 1 – 0
	$m_{12} - m_{13}$	1 1 0 –
3	$m_7 - m_{15}$	– 1 1 1
	$m_{14} - m_{15}$	1 1 1 –
	$m_{13} - m_{15}$	1 1 – 1
	$m_{11} - m_{15}$	1 – 1 1

Step IV Group	Matched pair	II n2 Reduction
1	$m_8 - m_9 - m_{10} - m_{11}$	10--
	$m_8 - m_9 - m_{12} - m_{13}$	1-0-
	$m_8 - m_{10} - m_9 - m_{11}$	10--
	$m_8 - m_{10} - m_{12} - m_{14}$	1--0
	$m_8 - m_{12} - m_9 - m_{13}$	1-0-
	~~$m_8 - m_{12}$~~	
2	$m_6 - m_7 - m_{14} - m_{15}$	-11-
	$m_6 - m_{14} - m_7 - m_{15}$	-11-
	$m_9 - m_{11} - m_{13} - m_{15}$	1--1
	$m_9 - m_{13} - m_{11} - m_{15}$	1--1
	$m_{10} - m_{11} - m_{14} - m_{15}$	1-1-
	$m_{12} - m_{14} - m_{13} - m_{15}$	11--
	$m_{12} - m_{13} - m_{14} - m_{15}$	11--

Step 5 Prime Implicant	Indicated terms	minterms 6	7	8	9
	8, 9, 10, 11, 12, 13, 14, 15			X	X
	6, 7, 9, 10,				

19/7/18 Reduce the following Boolean expression

(1) $AA + BB = A + B$

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

(3) $BC + BBC = BC$

(4) $WWX + WXX + WWX\bar{X} = \underline{WX}$

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

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

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

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

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

(8) $w\bar{x}(w + y) + wy(\bar{w} + \bar{x})$

$\Rightarrow w\bar{x} + w\bar{x}y + 0 + wy\bar{x}$

$\Rightarrow w\bar{x} + wy\bar{x} \Rightarrow w\bar{x}$

(9) $(AB + C)(AB + D)$

$\cancel{(A+C)}\cancel{(A+B)}\ AB + ABD + ABC + CD$

$\Rightarrow AB(1 + D) + ABC + CD$

$\Rightarrow AB(1 + C) + CD \Rightarrow \underline{AB + CD}$

(10) $A + \bar{B}C(\overline{A + \bar{B}C})$

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

$= \underline{A}$

(11) $C(\overline{ABC} + A\overline{B}C)$

$C[\overline{A}+\overline{B}+\overline{C}+A\overline{B}C]$

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

$C[\overline{A}+\overline{B}+A\overline{B}]$

$C[\overline{A}+\overline{B}(1+A)] = C[\overline{A}+\overline{B}]$

(12) $\overline{\overline{ABC+\overline{A}\,\overline{B}}+BC}$

$\Rightarrow \cancel{ABC} \cdot \overline{\overline{ABC+\overline{A}\,\overline{B}}}\cdot\overline{BC}$

$(ABC+\overline{A}\,\overline{B})B\overline{C} = 0$

(13) $A[B+C(\overline{AB+AC})]$

$\Rightarrow A[B+C(\overline{AB}\cdot\overline{AC})]$

$\Rightarrow A[B+C[(\overline{A}+\overline{B})(\overline{A}+\overline{C})]]$

$\Rightarrow A[B+C[\overline{A}+\overline{A}\,\overline{C}+\overline{A}\,\overline{B}+\overline{B}\,\overline{C}]$

$\Rightarrow A[B+C\overline{A}+0+\overline{A}\,\overline{B}+0]$

$\Rightarrow A[B+C\overline{A}+\overline{A}\,\overline{B}] \Rightarrow AB$

(12) $\overline{\overline{ABC + \bar{A}\bar{B}} + BC} = \bar{A}\bar{B}$

$\Rightarrow \overline{\overline{ABC} \cdot \overline{\bar{A}\bar{B}} + BC}$

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

$\Rightarrow \overline{0 + A\bar{B} + A\bar{C} + \bar{A}B + 0 + B\bar{C} + BC}$

$\Rightarrow \overline{A\bar{B} + A\bar{C} + \bar{A}B + B\bar{C} + BC}$

$\Rightarrow \overline{A\bar{B} + A\bar{C} + \bar{A}B + B}$

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

$\Rightarrow \overline{A\bar{B}} \cdot \overline{A\bar{C}} \cdot \bar{B} \Rightarrow \bar{A}B \cdot \bar{A}C \cdot \bar{B} = 0$

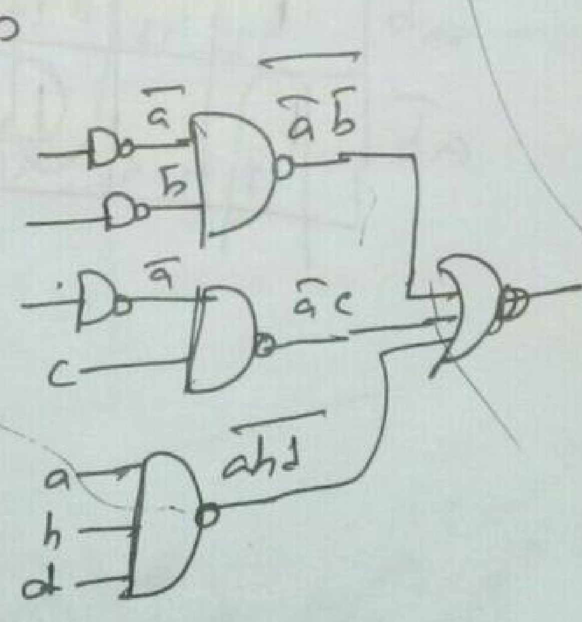
Reduce the expression $f = \sum_m (0,1,2,3,6,7,13,15)$ by mapping & implement in NAND logic

ab \ cd	$\bar{c}\bar{d}$	$\bar{c}d$	cd	$c\bar{d}$
$\bar{a}\bar{b}$	1 (0)	1 (1)	1 (3)	1 (2)
$\bar{a}b$	(4)	(5)	1 (7)	1 (6)
ab	(12)	1 (13)	1 (15)	(14)
$a\bar{b}$	(8)	(9)	(11)	(10)

$\bar{a}\bar{b}$

$\bar{a}c$

abd



$f = \bar{a}\bar{b} + \bar{a}c + abd$

$= \overline{\overline{\bar{a}\bar{b} + \bar{a}c + abd}}$ (NAND)

$= \overline{\overline{\bar{a}\bar{b}} \cdot \overline{\bar{a}c} \cdot \overline{abd}}$

Reduce $F = \sum_m (0,2,3,4,5,7)$ using Mapping

$a \backslash bc$	$\bar{b}\bar{c}$	$\bar{b}c$	bc	$b\bar{c}$
$\bar{a}$	1 $_0$	$_1$	1 $_3$	1 $_2$
a	1 $_4$	1 $_5$	1 $_7$	$_6$

$\bar{b}\bar{c} + ac + \bar{a}b$

$a \backslash bc$	$\bar{b}\bar{c}$	$\bar{b}c$	bc	$b\bar{c}$
$\bar{a}$	1 $_0$	$_1$	1 $_3$	1 $_2$
a	1 $_4$	1 $_5$	1 $_7$	$_6$

$a\bar{b} + bc + \bar{a}\bar{c}$

both is correct

Reduce by mapping & implement is SOP NAND logic

$N = \sum_m (0,2,3,6,7,8,10,11,12,15)$

$ab \backslash cd$	$\bar{c}\bar{d}$	$\bar{c}d$	cd	$c\bar{d}$
$\bar{a}\bar{b}$	1 $_0$	$_1$	1 $_3$	1 $_2$
$\bar{a}b$	$_4$	$_5$	1 $_7$	1 $_6$
ab	1 $_{12}$	$_{13}$	1 $_{15}$	$_{14}$
$a\bar{b}$	1 $_8$	$_9$	1 $_{11}$	1 $_{10}$

X

1111 - 1111

0111 $\quad$ 1000

$\quad\quad\quad$ ---

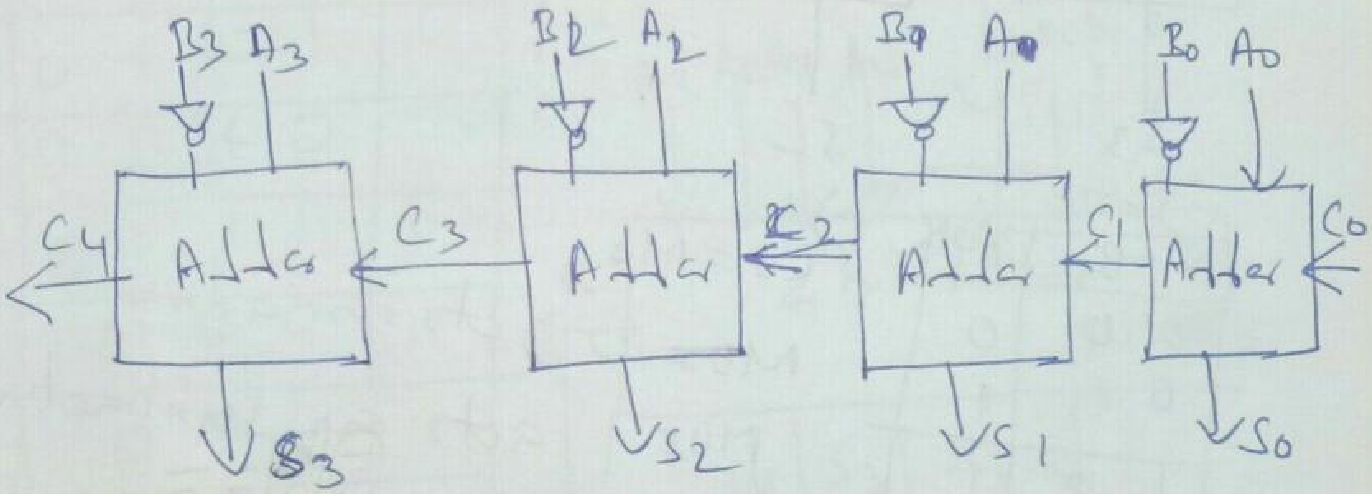
$\quad\quad\quad$ 111

Solve the expression by QM & verify with mapping $f = \sum m(0, 2, 3, 6, 7, 8, 9, 10, 13)$

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

---

4 bit binary Subtractor



$A = 15 = 1111$ 15   1111   15 1111

$B = 07 = 0111$ −7   1000   2 0010

10000   10111   1101

1   1000

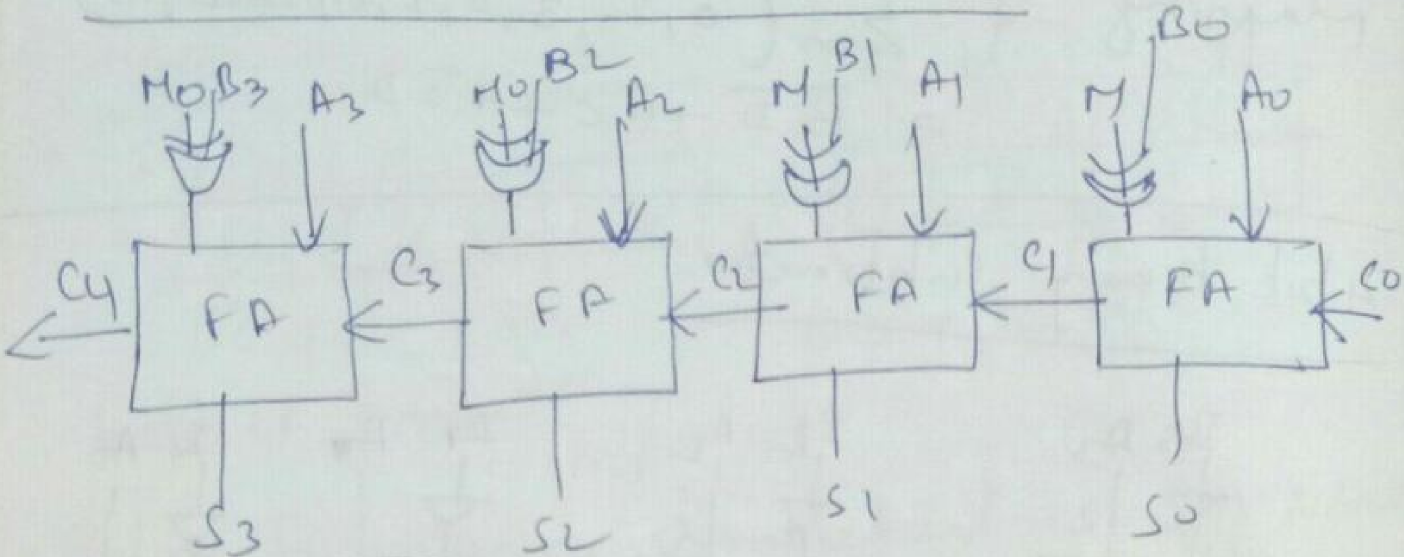
1111
1101
11100
1

For 4 bit Subtracter $C_0 = 1$

Carry look Ahead Adder. Carry Generate & Propagate.

4 bit ||el Adder & subtrator

XOR



A	B0	XOR
0	0	0
0	1	1
1	0	1
1	1	0

C0 = M0

M0 = 0 acts as adder

M0 = 1 acts as subtracter

1000
1001
(+) 10001

0000 1000
1 0

Serial Adder

0000
0001
0010
0011
0100
0101
0110
0111
1000
1001

BCD Adder

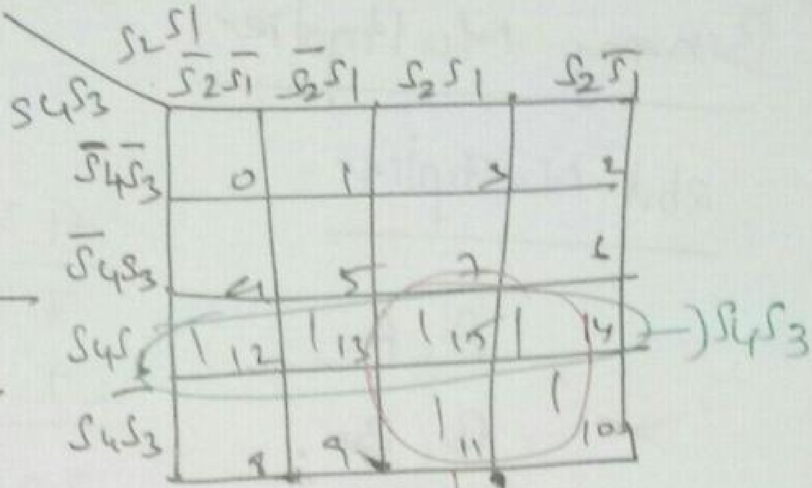
(1) 1010
(2) 1011
(3) 1100
(4) 1101
(5) 1110
(6) 1111
(10) 10000

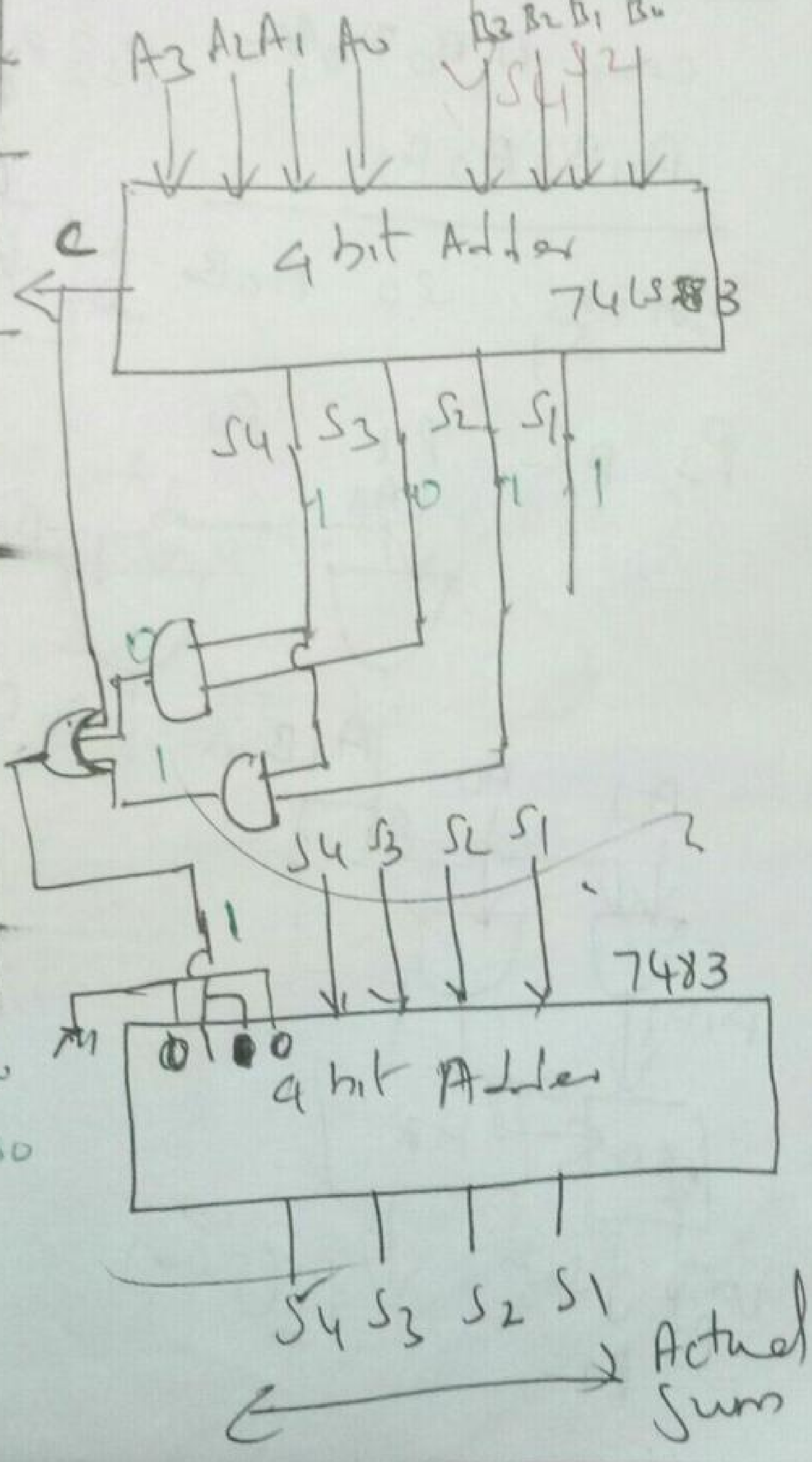
0001 0000

correction required

Each decimal value is coded by binary

	S_4	S_3	S_2	S_1	Y
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	0
4	0	1	0	0	0
5	0	1	0	1	0
6	0	1	1	0	0
7	0	1	1	1	0
8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	1	0	1
11	1	0	1	1	1
12	1	1	0	0	1
13	1	1	0	1	1
14	1	1	1	0	1
15	1	1	1	1	





0110 A3 A2 A1 A0
0101 B3 B2 B1 B0
1011
0110
10001

Binary Multiplier

43
32

2bit Multiplier

43×32

86
129

1376

$A_1 A_0$
$B_1 B_0$

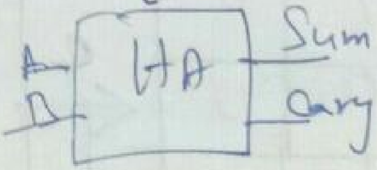
Carry A_1B_0 A_0B_0
A_1B_1 A_0B_1

~~A_1B~~ S_1 S_0 A_0B_0

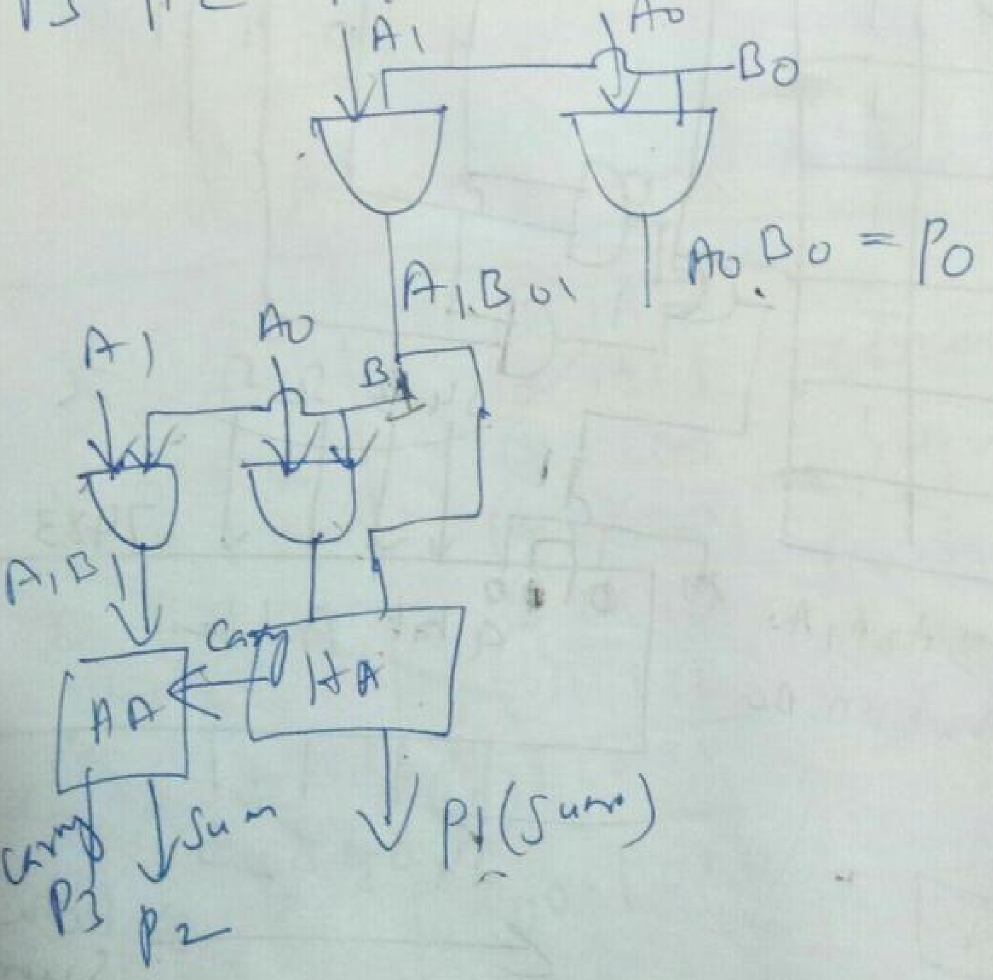
$A_1A_0 \times B_1B_0$

A_1B_0 A_0B_0
$P_3P_2P_1P_0$ A_0B_1
A_1B_1

Carry S_1 S_0 A_0B_0



P_3 P_2 P_1 P_0



Magnitude Comparator

Binary division

```
       10011.1
100 | 1001110
      100
      000111
         100
         0110
          100
          0100
           100
             0
```

Introduction to MUX (Multi plexer)

It is a combinational ckt that selects binary information from one of many input lines and directs it to o/p line.

It is simply called data selector.

Advantages :- 1) Reduce no of wires

2) Reduces circuit complexity & cost

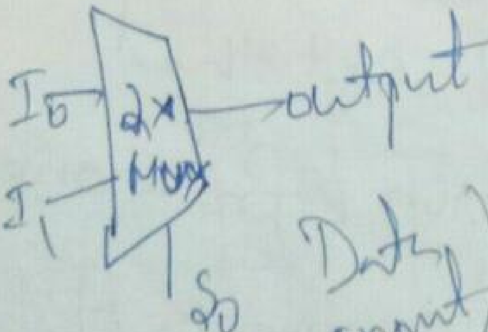
3) Implementation of various circuit using MUX.

n = no of inputs = 2^m → no of selection lines.

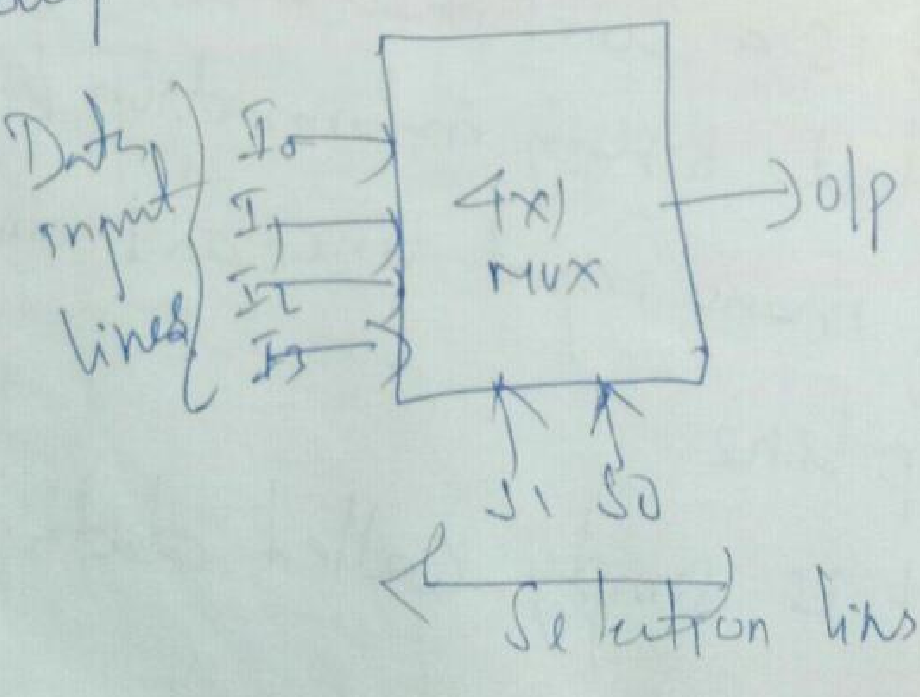
$$\boxed{m = \log_2 n}$$

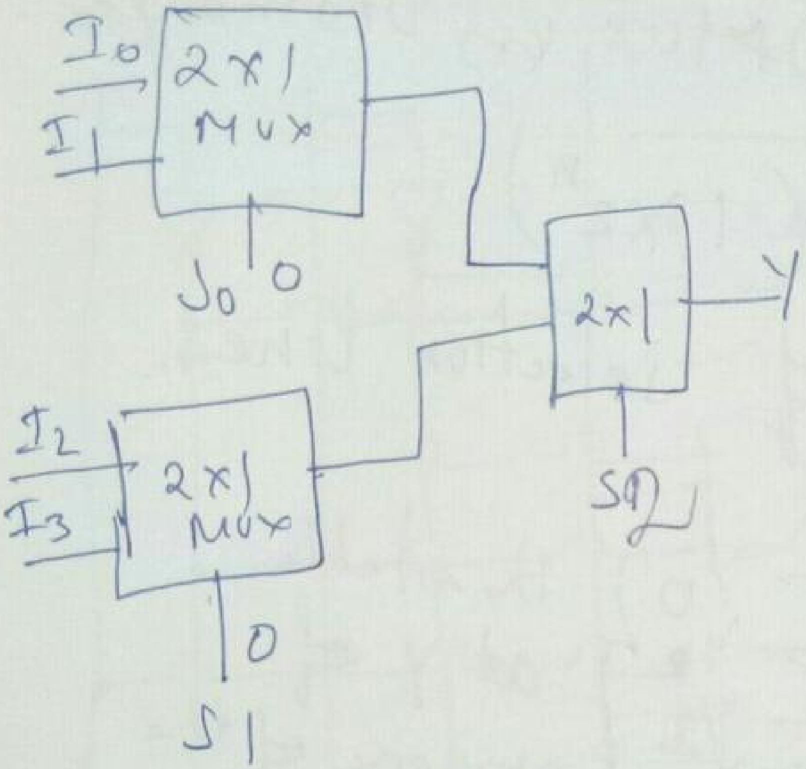
if $n = 4$; $m = \log_2 2^2 = 2$.

2×1 Multiplexer



4×1 MUX





S_1	S_0	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

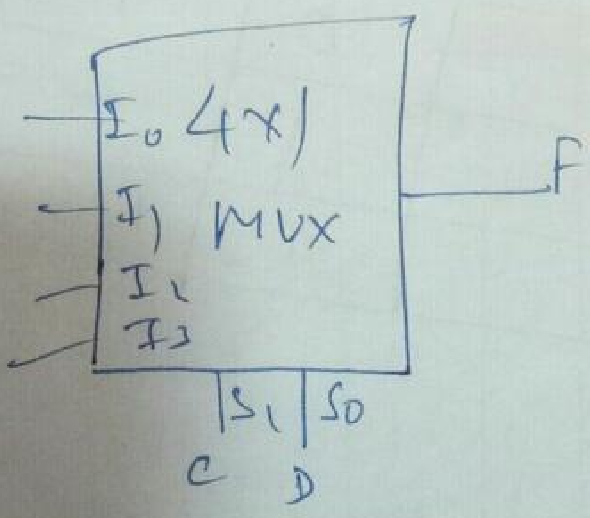
Implementation using Multiplexer

$F(a,b,c,d) = \Sigma(1,3,4,11,12,13,14,15)$

Using 4x1 Mux with a,b as Input Variable and C,D as Selection Variable.

	I_0	I_1	I_2	I_3
$\bar{a}\bar{b}$	0	1	2	3
$\bar{a}b$	4	5	6	7
$a\bar{b}$	8	9	10	11
ab	12	13	14	15

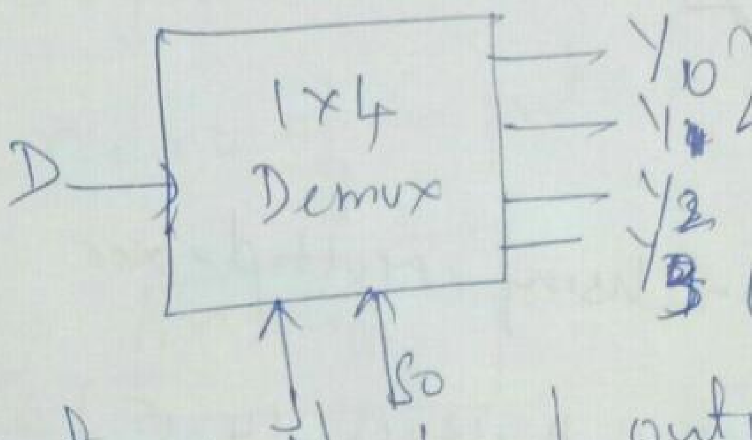
b ab ab a-15



Demultiplexer : ~~D~~MUX or Distributer

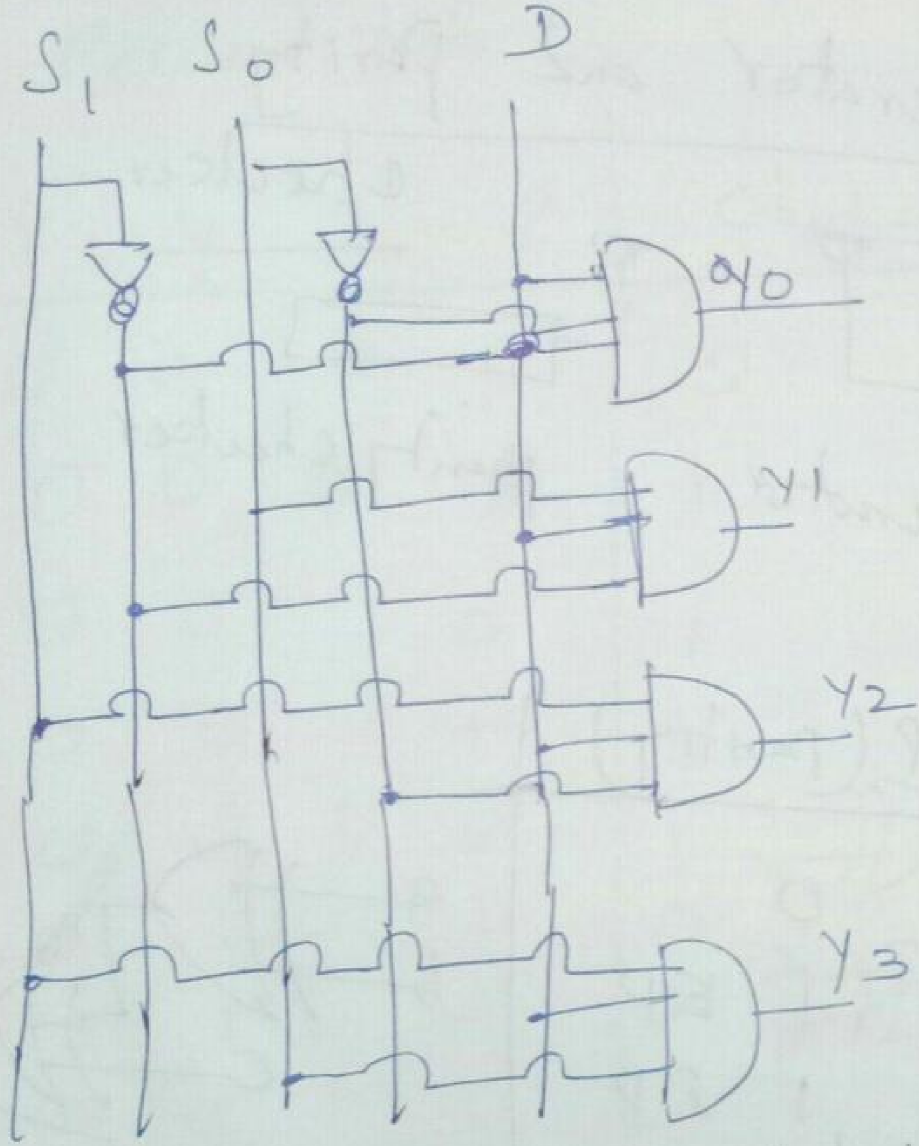
One to Many (1×2^n)

$n \Rightarrow$ no of selection lines.



Available at Y of any one of the outputs.

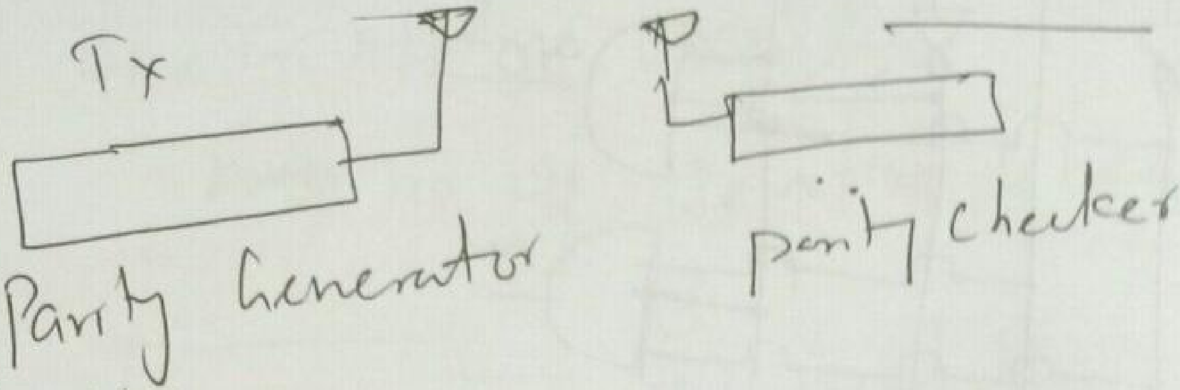
Input D	Selection lines S_1	S_0	outputs Y_0	Y_1	Y_2	Y_3
	0	0	D	0	0	0
	0	1	0	D	0	0
	1	0	0	0	D	0
	1	1	0	0	0	D



Decoder → 3 to 8 decoder. (Full Adder SFS)

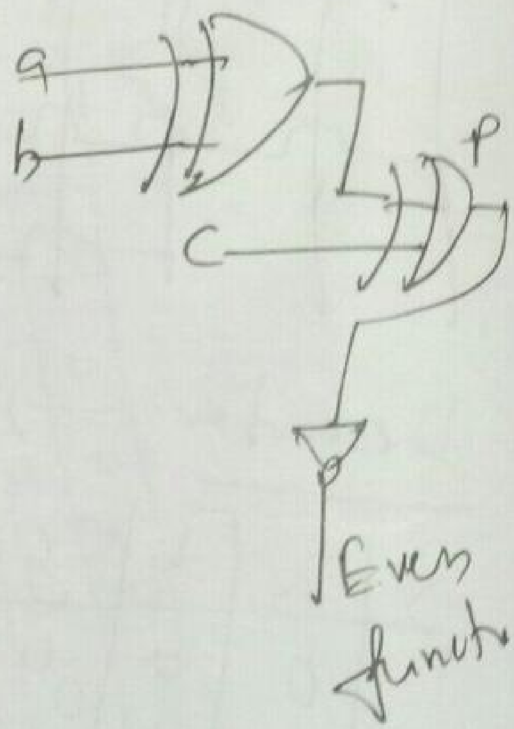
D	a	b	c	D1	D2	D3	D4	D5	D6	D7
1	0	0	0	1	0	0	0	0	0	0
1	0	0	1	0	1	0	0	0	0	0
1	0	1	0	0	0	1	0	0	0	0
1	1	0	0	0	0	0	1	0	0	0
1	1	0	1	0	0	0	0	1	0	0
1	1	1	0	0	0	0	0	0	1	0
1	1	1	1	0	0	0	0	0	0	1

even Parity generator and Parity checker



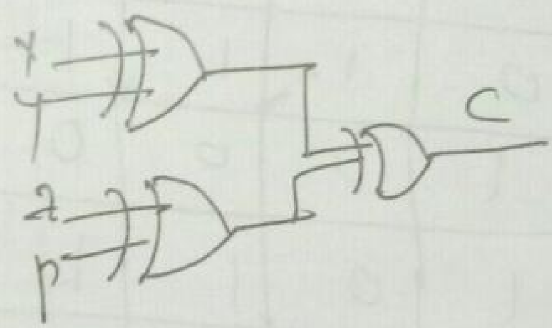
message

a	b	c	P (parity)	
0	0	0	0	
0	0	1	1	EP
0	1	0	1	EP
0	1	1	0	
1	0	1	0	
1	1	0	0	
1	1	1	1	



Received bits

X	Y	Z	P	Checker
0	0	0	0	0
0	0	0	[illegible]	1
0	0	1	0	1
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	1
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	1
1	1	1	0	1
1	1	1	1	0



When C = 0
no error
When C = 1
error.

odd parity generator

message bit			Parity (odd)
a	b	c	p
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

odd parity checker

a	b	c	p	Checker
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	0
1	0	0	0	0
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

Binary to Gray

1 0 0 0 → Binary

1 1 0 0 → Gray

Gray to Binary

1 0 0 0 → Gray

1 1 1 1 → Binary

1 0 1 0

1 1 1 1

1 0

1 1 1 1

1 0 1 0

1 0 0 0

code converter

Binary to Graycode (4 bit)

Decimal No	a	b	c	d	G_1	G_2	G_3	G_4	
0	0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	1	1
2	0	0	1	0	0	0	1	1	3
3	0	0	1	1	0	0	1	0	2
4	0	1	0	0	0	1	1	0	6
5	0	1	0	1	0	1	[illegible]	1	7
6	0	1	1	0	0	1	0	1	5
7	0	1	1	1	0	1	0	0	4
8	1	0	0	0	[illegible]	1	0	0	12
9	1	0	0	1	[illegible]	1	0	1	13
10	1	0	1	0	[illegible]	1	1	1	15
11	1	0	1	1	[illegible]	1	1	0	14
12	1	1	0	0	[illegible]	0	1	0	10
13	1	1	0	[illegible]	1	0	1	1	11
14	1	1	[illegible]	0	1	0	0	1	9
15	1	1	1	1	1	0	0	0	8

The

(4)

=

$= \bar{d}$

$+d)$

$+C$

$\bar{b})+$

$+C)$

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

$(\bar{b}+c$

Binar

$G_1 = \sum m(8, 9, 10, 11, 12, 13, 14, 15)$

ab \ cd	$\bar{c}\bar{d}$	$\bar{c}d$	cd	$c\bar{d}$
$\bar{a}\bar{b}$	0	1	3	2
$\bar{a}b$	4	5	7	6
ab	1 (12)	1 (13)	1 (15)	1 (14)
$a\bar{b}$	1 (8)	1 (9)	1 (11)	1 (10)

$\Rightarrow a$

$G_2 = \sum m(4, 5, 6, 7, 8, 9, 10, 11)$

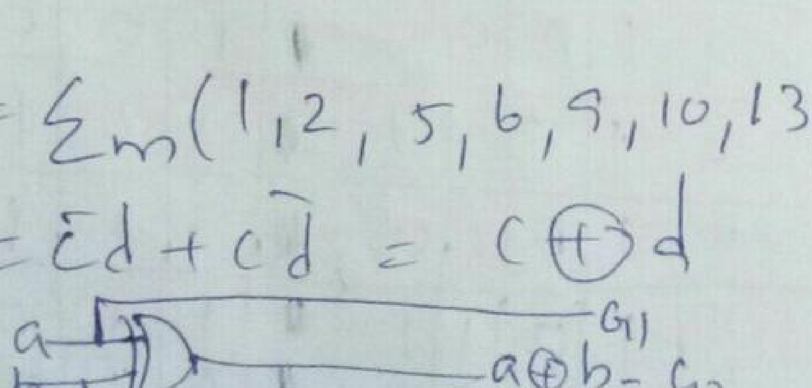
$= \bar{a}b + a\bar{b} = A \oplus B$

$G_3 = \sum m(2, 3, 4, 5, 12, 13, 10, 11)$

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

$G_4 = \sum m(1, 2, 5, 6, 9, 10, 13, 14)$

$G_4 = \bar{c}d + c\bar{d} = c \oplus d$



Gray to binary

1 0 0 1   1 0 0 0

1 1 1 0   1 1 1 1

K-Map $\Rightarrow$ $A_m = A, A \oplus B,$

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

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

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

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

$= A \oplus B \oplus C$

$B = \bar{A}\bar{B}\bar{C}D + \bar{A}\bar{B}CD + \bar{A}B\bar{C}\bar{D} + \bar{A}BCD +$
$AB\bar{C}D + ABC\bar{D} + A\bar{B}\bar{C}\bar{D} + A\bar{B}CD$

$= A \oplus b \oplus c \oplus d$

ab \ cd				
	0	$\boxed{1}$ 1	3	$\boxed{1}$ 2
	$\boxed{1}$ 4	5	$\boxed{1}$ 7	6
	12	$\boxed{1}$ 13	15	$\boxed{1}$ 14
	$\boxed{1}$ 8	9	$\boxed{1}$ 11	10

, the

, (4) +

c

$\bar{c}$, $\bar{\bar{d}}$

$+d)$

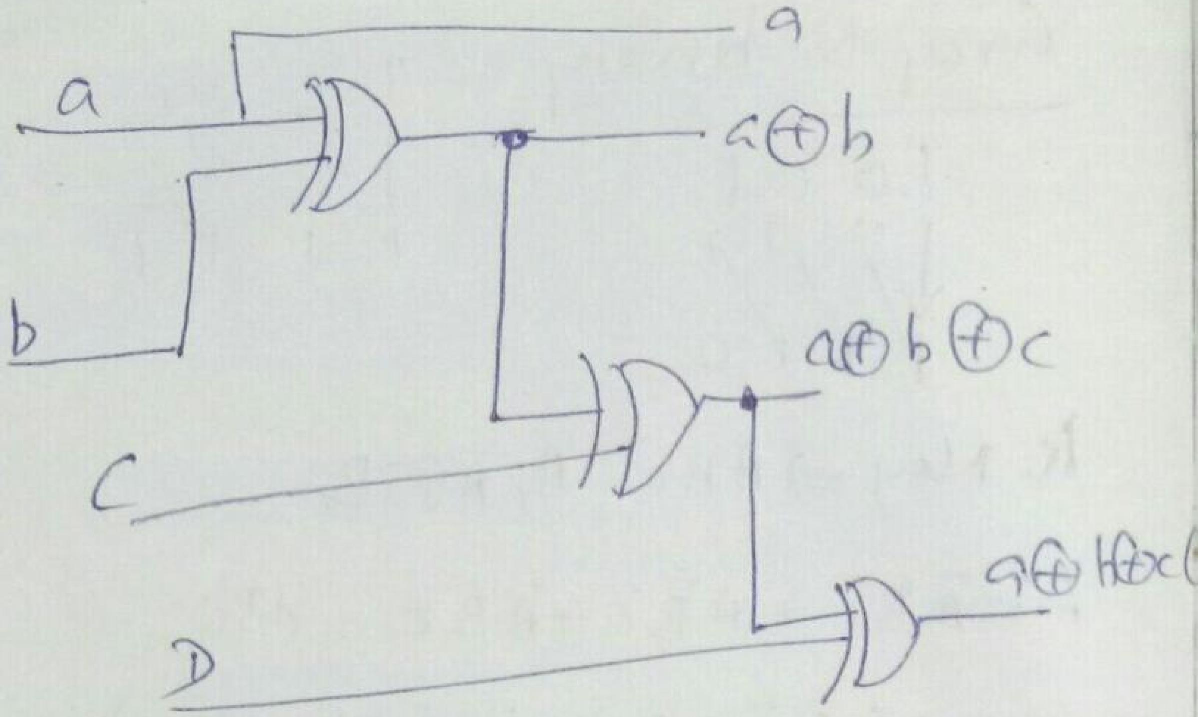
$c + c$

$+\bar{b}) +$

$\bar{b} + c)$

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

$(\bar{b} + c)$



BCD (8421) → Excess 3 code

$D = \bar{z}$ $B = \bar{x}y + \bar{x}z + xyz$

$C = y \oplus \bar{z}$ $A = wx + wyz$

$(c+d)(\bar{c}+d) \cdot (a+b)(\bar{a}+b)$

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

$z = \bar{D}$ $w = a + bc + bd$

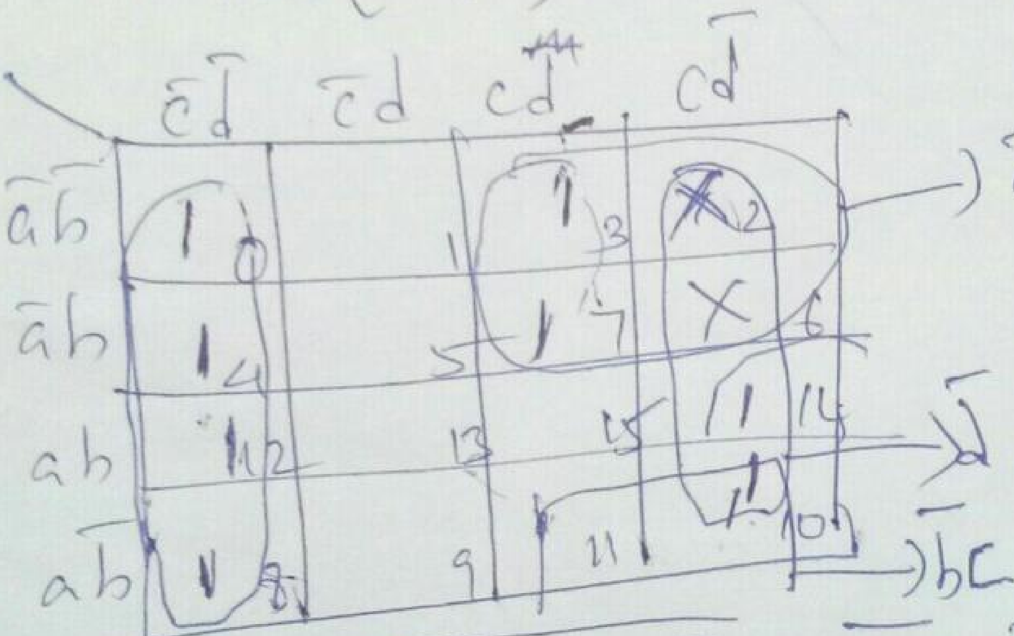
$y = CD + \bar{C}\bar{D}$ $x = \bar{b}c + \bar{b}d + b\bar{c}\bar{d}$

abcD i/p   upto 9, 10, 11, 12, 13, 1... X X X X

wxyz o/p

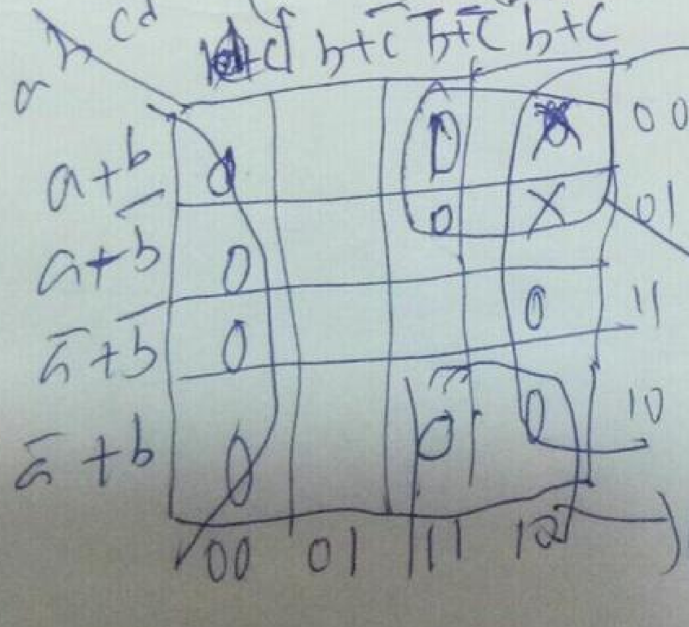
Realize / implement the logic function using NAND gates specified by the function

$$f = \Pi_M(0,3,4,7,8,10,12,14) + \Pi d(2,6)$$



$$f = \overline{\bar a c + \bar b c + \bar d} = \overline{\bar a c} \cdot \overline{\bar b c} \cdot \bar{\bar d}$$

$$= (a+\bar c)(b+\bar c)\, d$$



$(\bar c+d)(c+d)$

$bc + \bar b c + c$

$(a+b)(a+\bar b) + (\bar b+\bar c)(\bar b+c)$

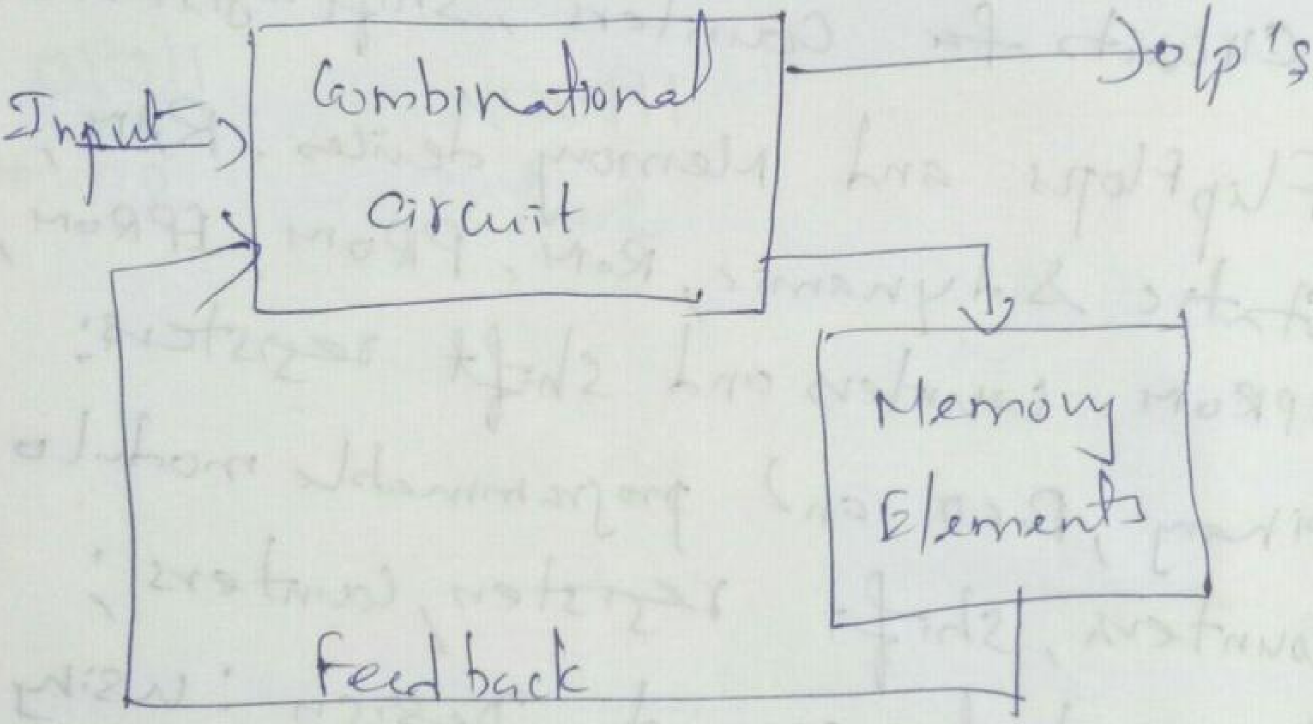
$(a+b)(\bar a+b)(\bar b+\bar c)(\bar b+c)$

Unit II

CO2 → Apply the concept of sequential circuits for counters, shift register etc

Flipflops and Memory devices: RAM Static & dynamic, ROM, PROM, EPROM, EEPROM, counters and shift registers: Binary, BCD and programmable modulo counters, shift register, counters; Sequential circuit Design: using Mealy and Moore model.

Block Diagram of Sequential circuits



In sequential o/p's depends present Input as well as previous output (or previous state of the condition.

O/p depends Input as well as clock pulse.

Sequential
- Synchronous ckt
- Asynchronous ckt

A Synchronous sequential circuits is a system whose behaviour can be defined from the knowledge of its signal at discrete instant of time.

The behaviour of asynchronous ckt depend upon the input signals at any instant of time and the order in which input change.

The storage elements (memory) used in clocked sequential ckts are called flip flops.

<u>**Flip Flop**</u>: It is a binary storage device capable of storing one bit of information. Its stable state the the o/p of a flip flop is either 0 or 1.

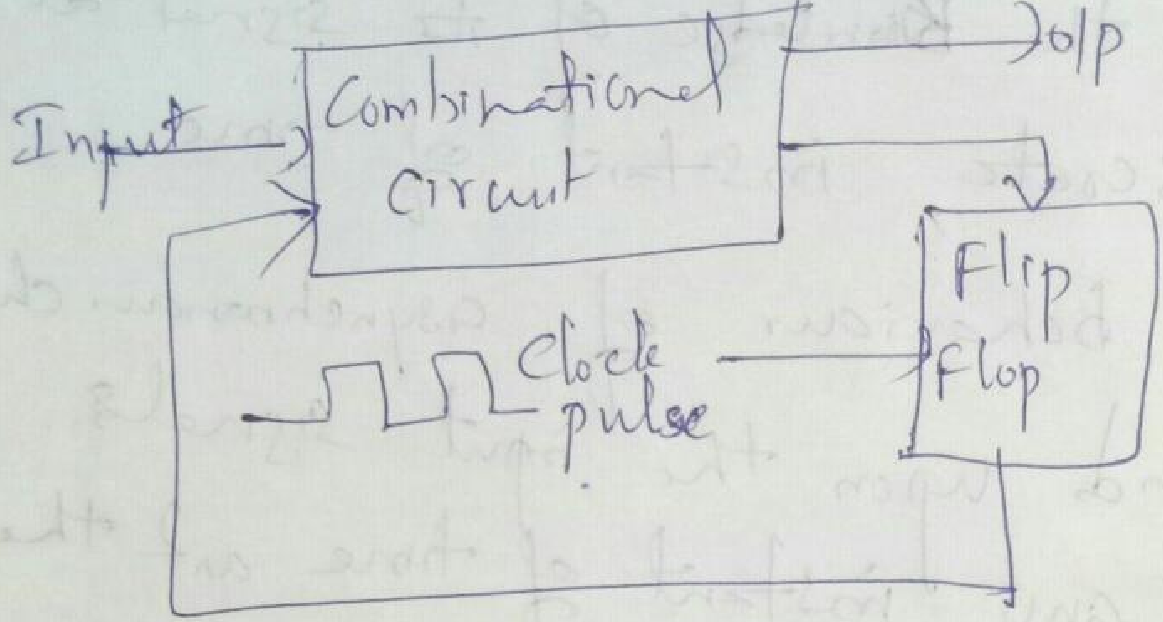
A synchronous sequential circuits is a system whose behaviour can be defined from the knowledge of its signal at discrete instant of time.

The behaviour of asynchronous ckt depend upon the input signals at any instant of time and the order in which input change.

The storage elements (memory) used in clocked sequential ckts are called flip flops.

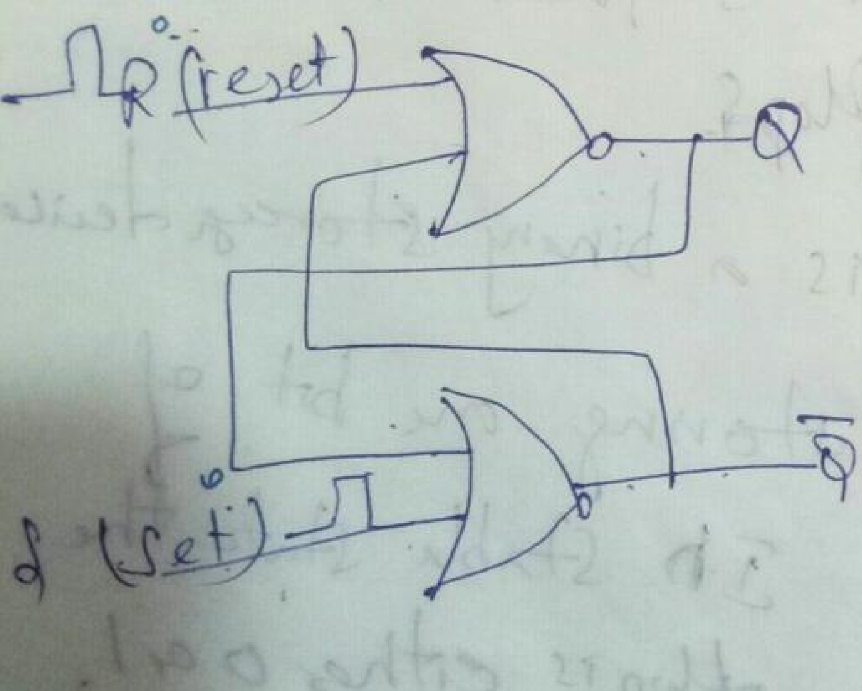
<u>**FlipFlop**</u>: It is a binary storage device capable of storing one bit of information. In stable state the the o/p of a flip flop is either 0 or 1.

Synchronous clocked Sequential circuit



Storage Elements: (Latches)

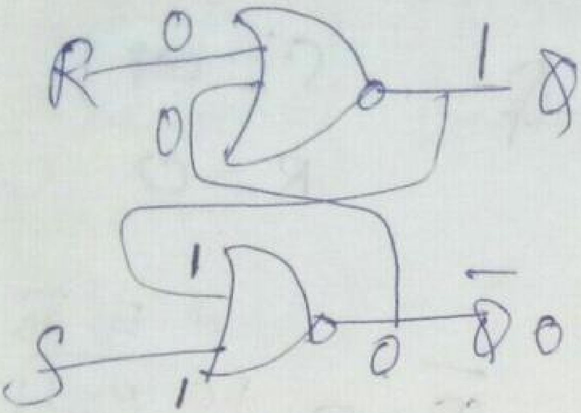
SR Latch with NOR gate



Function Table

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

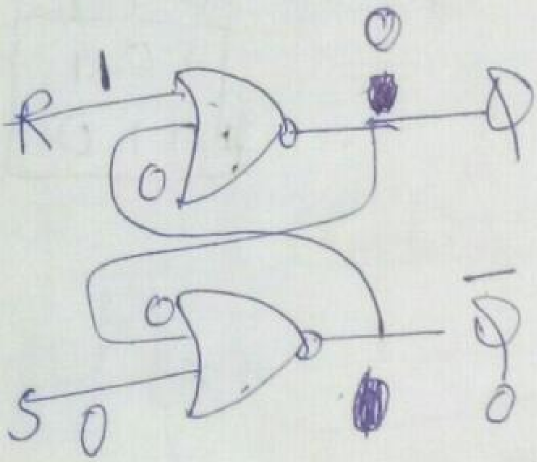
Forbidden



Function table

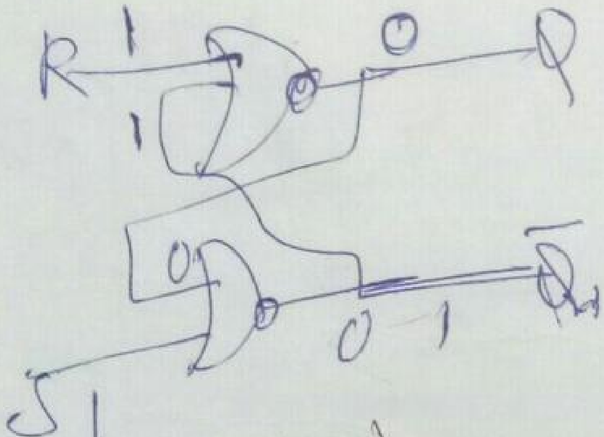
S	R	Q	$\bar{Q}$
[illegible]	0	1	0
0	[illegible]	1	0
1	1	0	0

when
$\bar{Q}=0$
$S=1$
$R=0$.



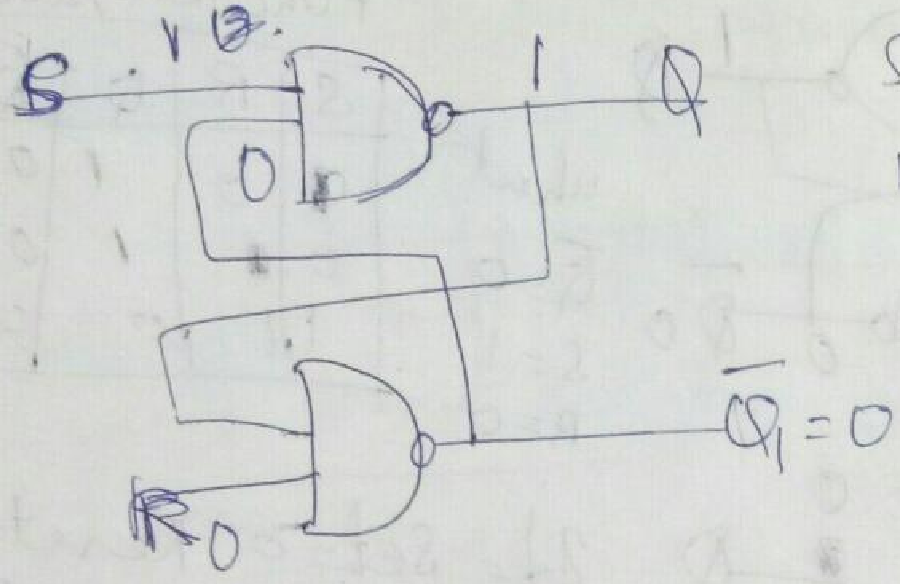
If Set=0 Reset=1
$\bar{Q}=0$ (previous state)
Current state $\bar{Q}_1=1$

If set=1 & Reset=1
prev. state $\bar{Q}_1=1$



Current state $Q_1=Q_0=0$
It is forbidden state
undefined state,
metastable state.

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



$S = 1$

$R = 0$

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

① Which of the following represents the sum of Products of expression for the function.

$f(w,x,y,z) = \sum(0,1,3,5,14) + \sum_d(8,15)$

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

$\Rightarrow \bar{w}\bar{x}\bar{z}$ ✓

$\Rightarrow wxy$ ✓

$\Rightarrow \bar{w}\bar{y}z$ ✓

$\Rightarrow x\bar{y}\bar{z}$

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

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

$\Rightarrow \cancel{wx\bar{y}}$

$\Rightarrow \bar{x}\bar{y}$

$\bar{x}\bar{z}$

$\Rightarrow \bar{w}\bar{x} + \bar{x}\bar{y} + \bar{x}.\bar{z}$

$f(w,x,y,z) = \sum(0,1,2,8,11) + \sum_d(3,9,15)$

(3) which of the following represents the complement of the function $x+yz$

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

(4) which of the following 2 input logic functions can act as a universal gate?

$a\oplus b$, ab, $\overline{(a+b)}$, none

(5) What is the minimum no of 2 input NOR gates that are required for implementing a 2 input XNOR functions

$\overline{a+b} = NOR$ $\quad \overline{a\oplus b} = XNOR$ $\quad \bar{a}\bar{b}$

$\overline{a\oplus b} = \overline{\bar{a}b+a\bar{b}} = \overline{\bar{a}b}\cdot\overline{a\bar{b}} = \bar{x}\cdot\bar{y}$

$\overline{\overline{a+b}} = \overline{\overline{ab}}$ $\quad$ (4) ans

(6) which of the following is not an essential prime implicant of the function. $f(a,b,c,d) = \Sigma(1,4,5,10,12,13,14,15)$

$\bar{a}\bar{b}\bar{c}d$, $\bar{a}b\bar{c}\bar{d}$, $\bar{a}b\bar{c}d$, $a\bar{b}c\bar{d}$, $ab\bar{c}\bar{d}$

$ab\bar{c}d$ $abcd$

	$\bar{c}\bar{d}$	$\bar{c}d$	cd	$c\bar{d}$	
$\bar{a}\bar{b}$	0	1 1	3	2	$\rightarrow \bar{a}\bar{c}d$
$\bar{a}b$	1 4	1 5	7	6	$\rightarrow \bar{a}b\bar{c}$
ab	1 12	1 13	1 15	1 14	$\rightarrow ab$
$a\bar{b}$	8	9	11	1 10	$\rightarrow ac\bar{d}$

(a) $\bar{a}\bar{c}d$ (b) $ac\bar{d}$ (c) $b\bar{c}d$ d) ab

(7) This boolean function $f(a,b,c,d) = \Sigma_m(3,7,11,13,14,15)$ simplifies to

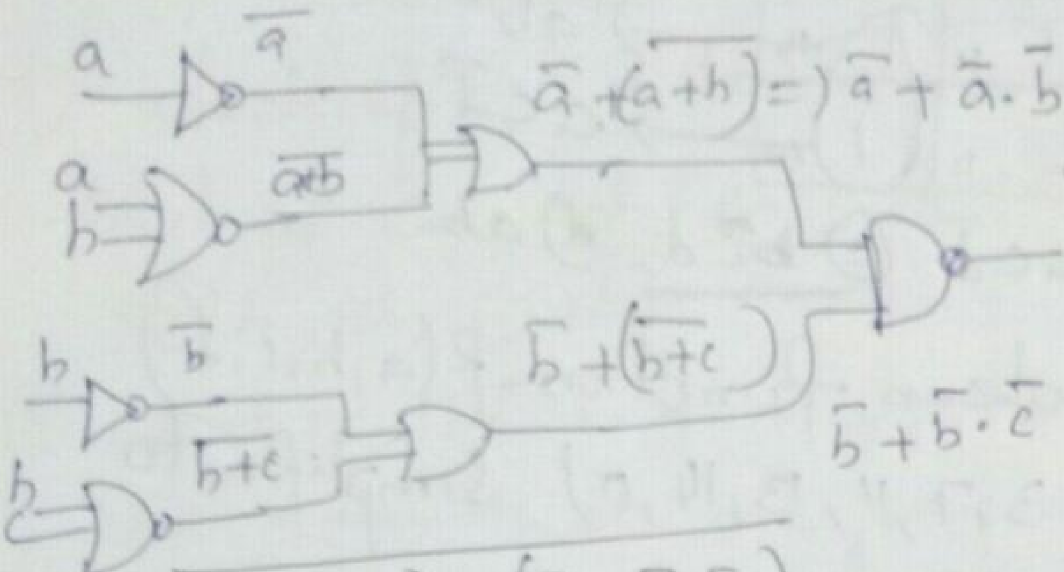
	$\bar{c}\bar{d}$	$\bar{c}d$	cd	$c\bar{d}$	
$\bar{a}\bar{b}$	0	1	1 3	2	$\rightarrow cd$
$\bar{a}b$	4	5	1 7	6	
ab	12	1 13	1 15	1 14	$\rightarrow abc$
$a\bar{b}$	8	9	1 11	10	$\rightarrow abd$

(8) With XOR gate which of the following logic gate may be taken to form a universal logic gate

$a \oplus b = \overline{\overline{\bar{a}b + a\bar{b}}} = \overline{\bar{a}b} \cdot \overline{a\bar{b}} = \overline{a+\bar{b}}$

$= \bar{a} \; \overline{(a+\bar{b})} = \overline{ab} = a$

(9) which of the following gate is equivalent to the given circuit.



$\overline{(\bar a + \bar a \cdot \bar b) \cdot (\bar b + \bar b \cdot \bar c)}$

$\overline{\bar a + \bar a \cdot \bar b} + \overline{\bar b + \bar b\, \bar c}$

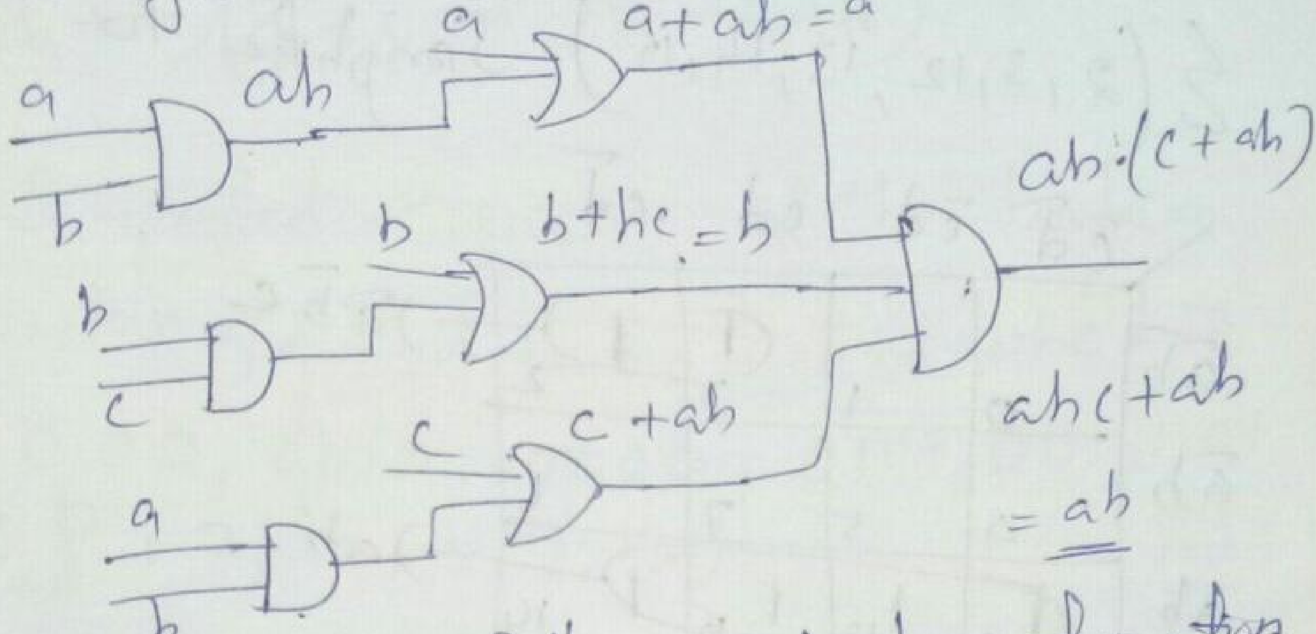
$\bar{\bar a} \cdot \overline{\bar a \cdot \bar b} + \bar{\bar b} \cdot \overline{\bar b\, \bar c}$

$a \cdot (a+b) + b \cdot (b+c)$

$(a + ab) + (b + bc)$

$\underline{a+b} \rightarrow$ OR

(10) which of the following gate is equivalent to the given circuit.



abc+ab

= $\underline{\underline{ab}}$

(11) which of the following boolean function is equivalent to the expression

$f(a,b,c,d) = \pi(0,1,2,3,4,7)$

	$\bar b\bar c$	$\bar b c$	bc	$b\bar c$
$\bar a\bar b$	1 (0)	1 (1)	1 (3)	1 (2)
$\bar a b$	1 (4)	(5)	1 (7)	(6)
ab	(12)	(13)	(15)	(14)
$a\bar b$	(8)	(9)	(11)	(10)

) $\bar a$

) bc

) $\bar a cd$

) $\bar a\bar c\bar d$ $\bar b\bar c$

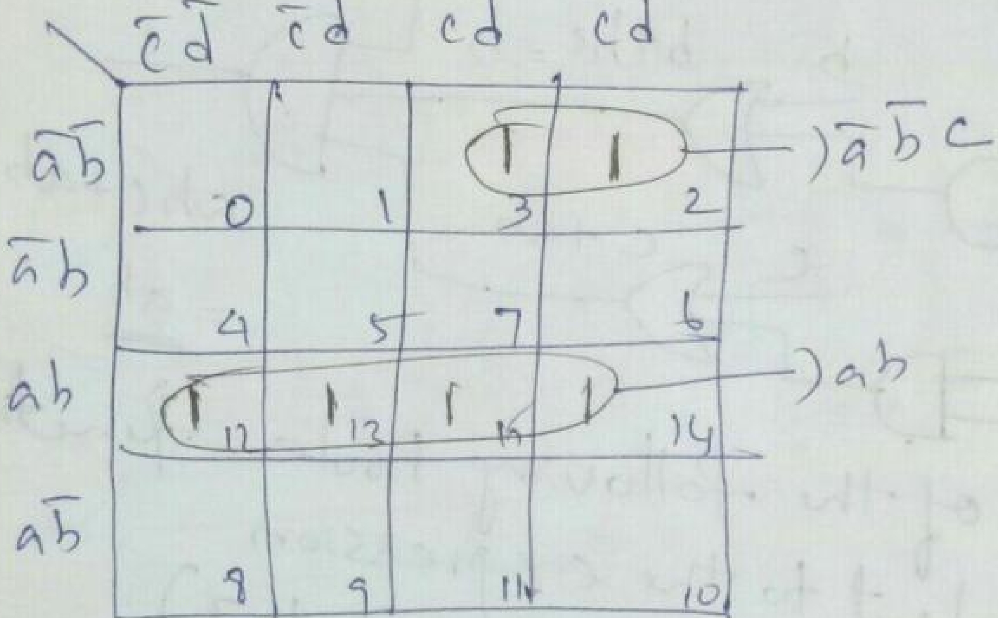
$= \overline{\bar a\bar b + \bar a cd + \bar a\bar c\bar d}$

$\Rightarrow \overline{\bar a\bar b}\cdot\overline{\bar a cd}\cdot\overline{\bar a\bar c\bar d}$

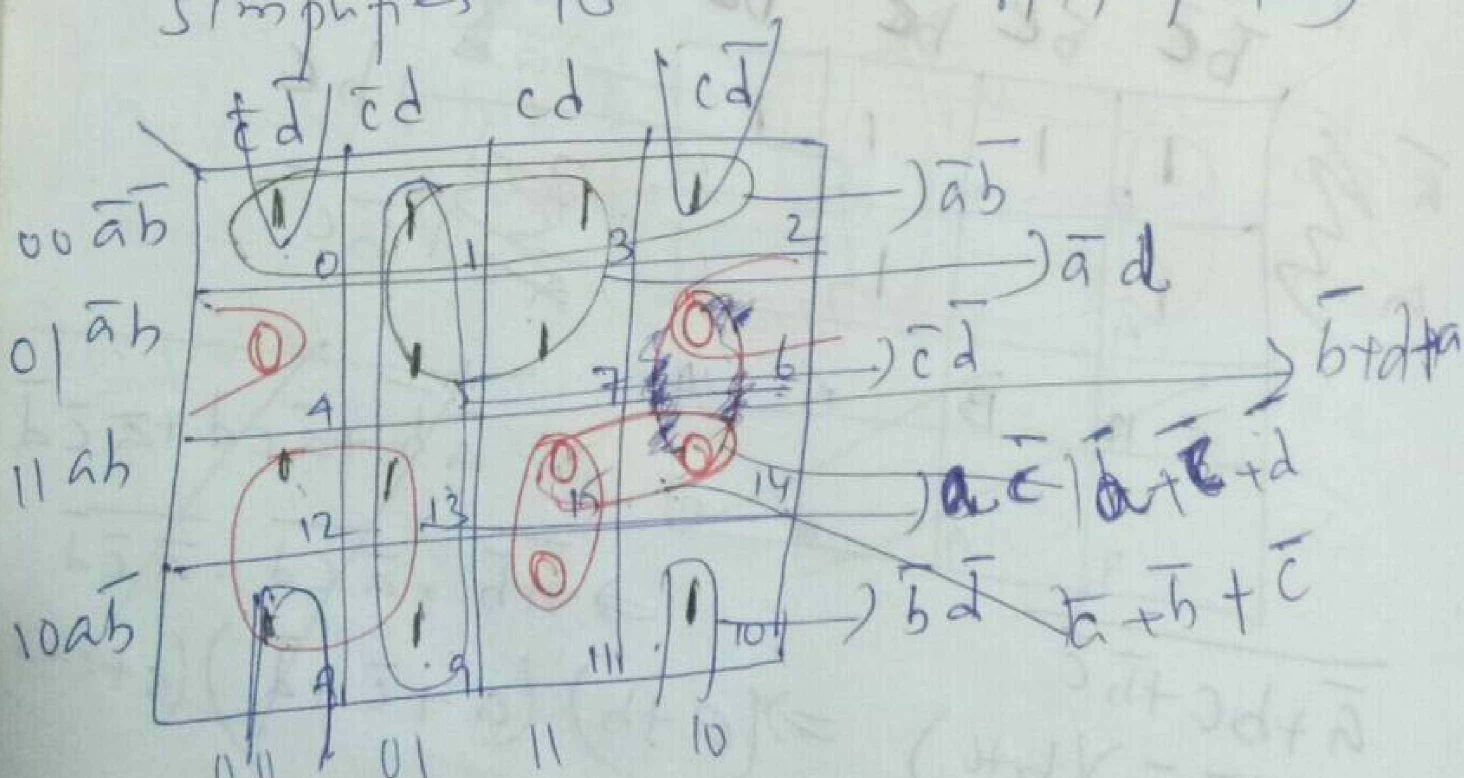
$\overline{\bar a + bc + \bar b\bar c}$

$= a(\bar b+\bar c)(b+c)$ $\Rightarrow (a+b)(a+\bar c+\bar d)(a+c+d)$

(12) The boolean function $f(a,b,c,d) = \sum(2,3,12,13,14,15)$ simplifies to



(13) The boolean function $f(a,b,c,d) = \sum(0,1,2,3,5,7,8,9,10,12,13)$ simplifies to



$\cancel{\bar a \bar b} \cancel{+ \bar a d} + \bar c \bar d + a\bar c + \bar b \bar d$

$(\bar a + \bar b + \bar c)(\bar a + \bar c + \bar d)(a + \bar b + d)$

(14) Simplified expression of the boolean function $f = \bar a \bar b \bar d + \bar a \bar b \bar c \bar d + \bar a b d + a b \bar c d$ is

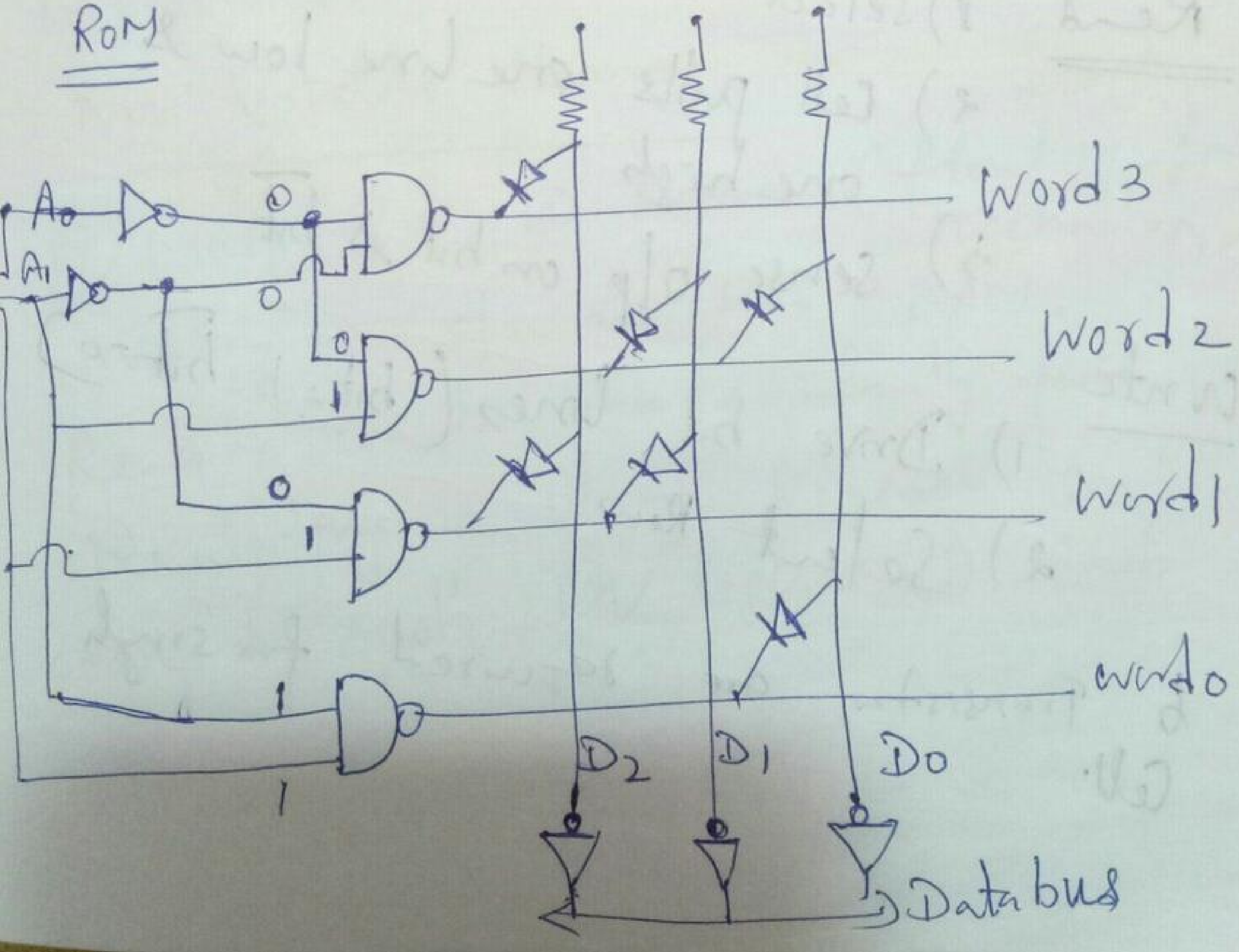
0000, 0010, 0000, 0101, 0111, 1101

$\Sigma_m(0, 1, 2, 0, 5, 7, 13)$

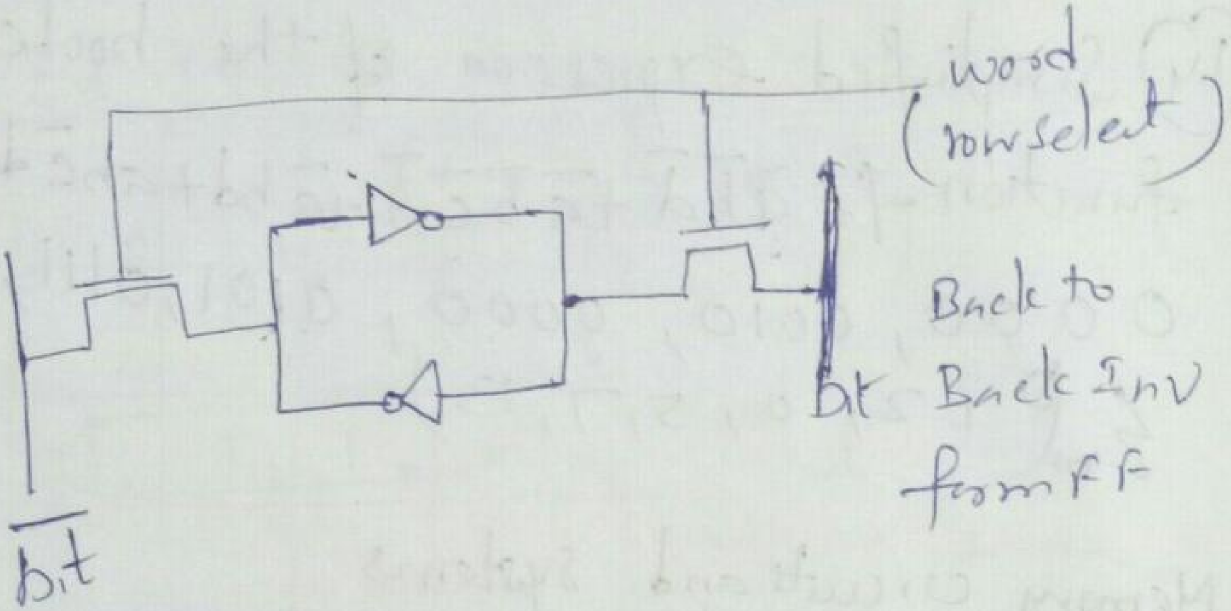
1010
1011
1100
1101

Memory circuits and systems

ROM



Basic static RAM Cell (Random Access Memory)



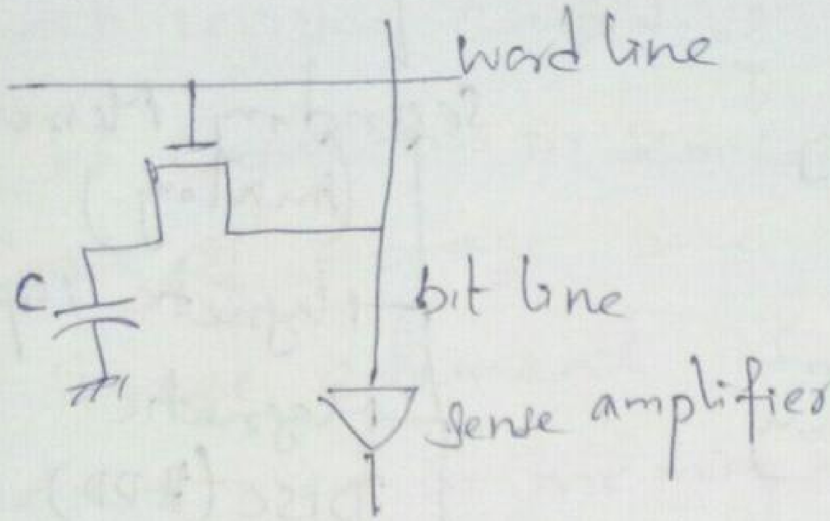
Back to Back Inv from FF

Read
1) Select Row
2) Cell pulls one line low & one high
3) Sense o/p on bit & $\overline{bit}$

Write
1) Drive bit lines (bit = 1 $\overline{bit}$ = 0)
2) Select Row

6 Transistor are required for single Cell.

DRAM :- Dynamic RAM



SRAM cells exhibit high speed / poor density

DRAM :- Simple Tr / capacitor pairs in high density form. Refresh at regular intervals

Non Volatile Memory

① Mask ROM ② EPROM (floating gate Technology)

writing is slower than RAM, Reading can be made fairly fast. Rewriting is slow. Erasable can be done by apply UV light ③ Flash

Memory

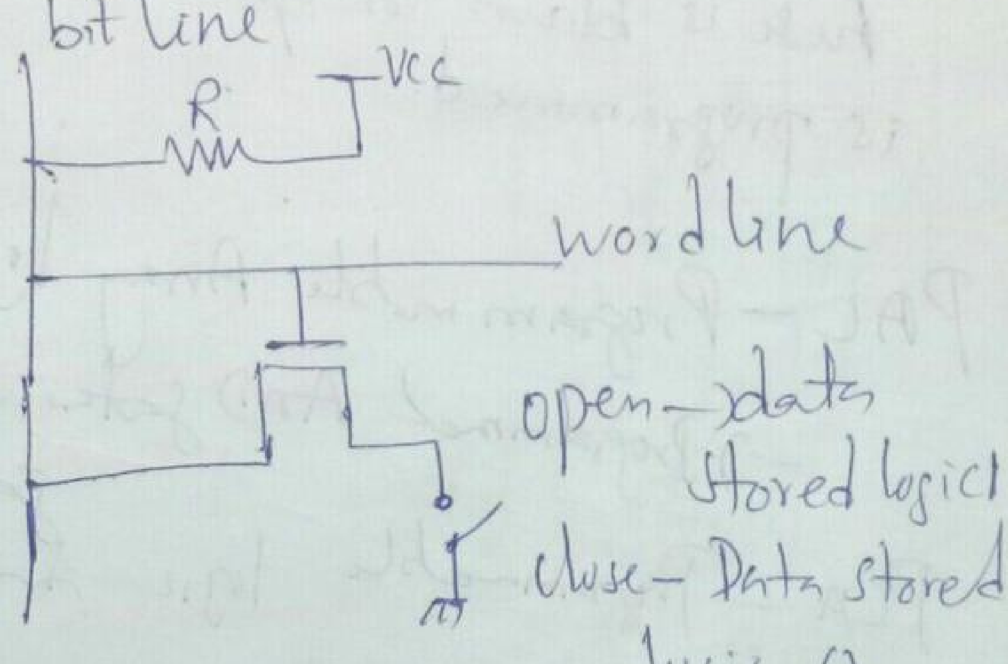
- Primary Memory
  - Cache Memory
  - Main Memory
    - RAM
      - SRAM
      - DRAM

      Volatile memory
    - ROM → Non Volatile
      - PROM
      - EPROM
      - EEPROM
- Secondary Memory (Auxiliary)
  - Magnetic Tape
  - Magnetic Disc (HDD)
  - Optical Disk
    - CD (700MB)
    - DVD (4.7GB)
    - HVD
  - Flash Memory ← Non Volatile
    - Memory card
    - Pen drive

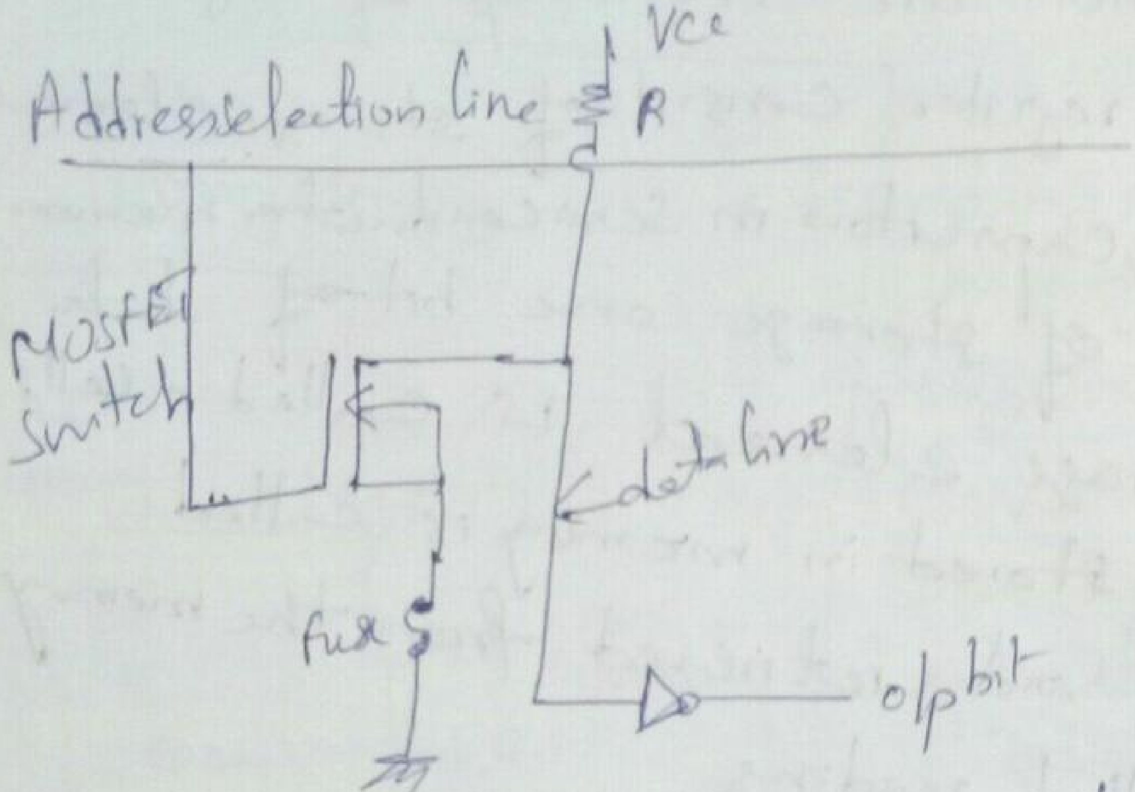
Non Volatile memory even if power is turned off can hold the data. But Volatile memory can hold data during power only.

Memories are made up of registers.
Each register consists of storage elements (FF, capacitors in semiconductor memories) each of storage one bit of data.
A storage element is called a cell.
Data stored in memory is called writing and retrieved from the memory is called reading

ROM cell



PROM: Programmable ROM



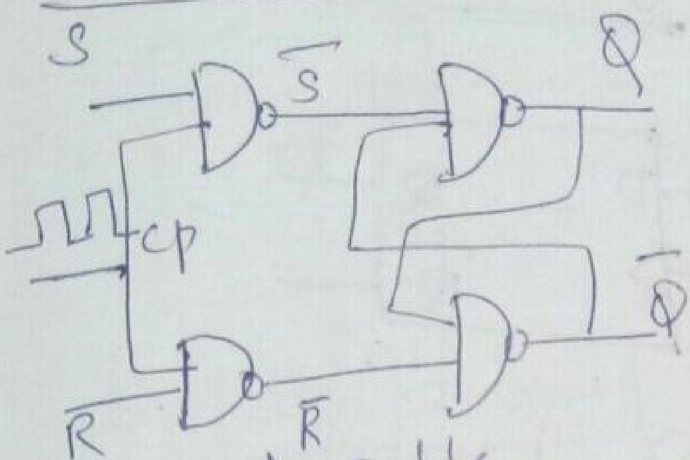
fuse is blown or open means → logic 0 is programmed.

PAL – Programmable Array logic
→ Programmed AND gates & fixed OR gate array

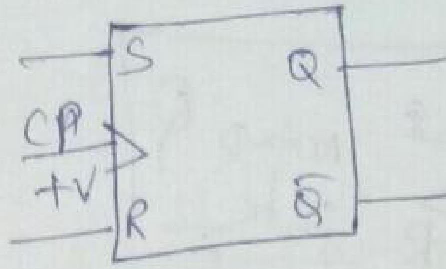
PLA – Programmable logic Array.
Both AND and OR are programmed

PROM → Fixed AND array constructed as decoder and programmable OR

SR clocked f/f



logic symbol



Truth table

CP	S	R	Q_n	Q_{n+1}	state
↑	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	X	Intermediate
↑	1	1	1	X	

SR \ Q_n	0	1
00		1
01		
11	X	X
10	1	1

$S\bar{Q}_n +$

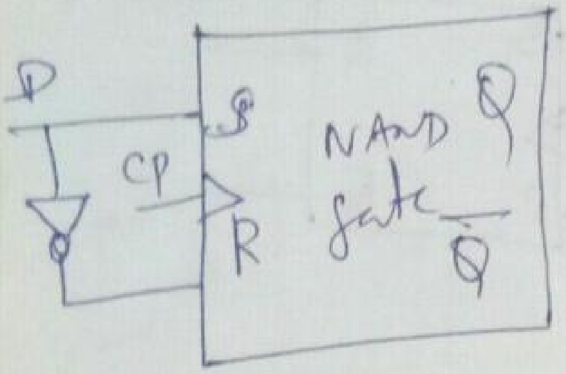
characterstic equation $\Rightarrow S + \bar{R}Q_n$

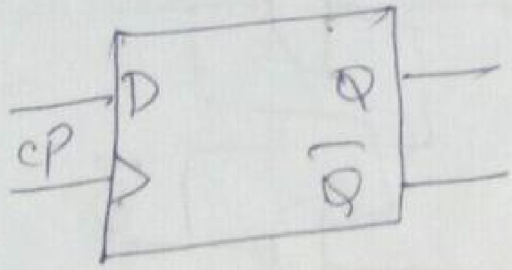
Q_n \ SR	00	01	11	10
0		1		X
1	1	1	X	

CP	S	R	Q_{n+1} NS
En	X	X	NC
ahl	0	0	NC
1	0	1	Q=0 Rst
1	1	0	Q=1 set
1	1	1	Intermediat

D flip flop

logic symbol





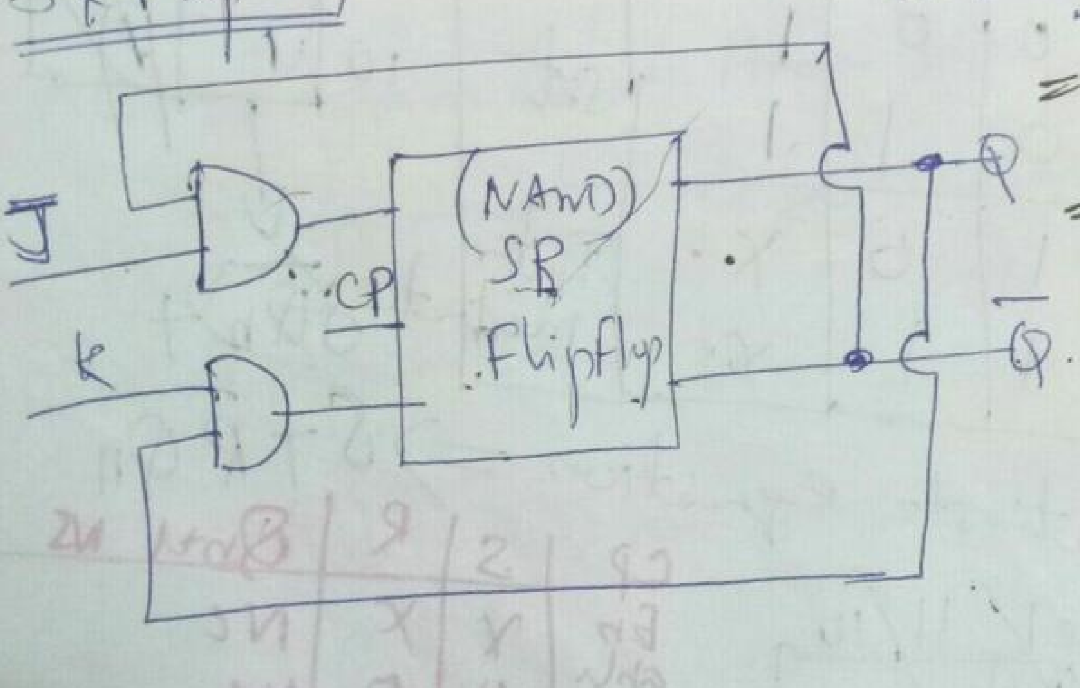
Truth table

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

$Q_{n+1} = D$

$= \bar{Q}_n$

JK flip flop



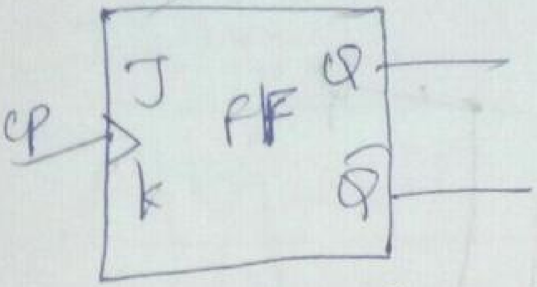
$Q_{n+1} = \bar{J}\bar{Q} + KQ$

$= 1 \times \bar{1} + 1 \times 1$

$= 0 + 1$

$= 1$

Logic symbol

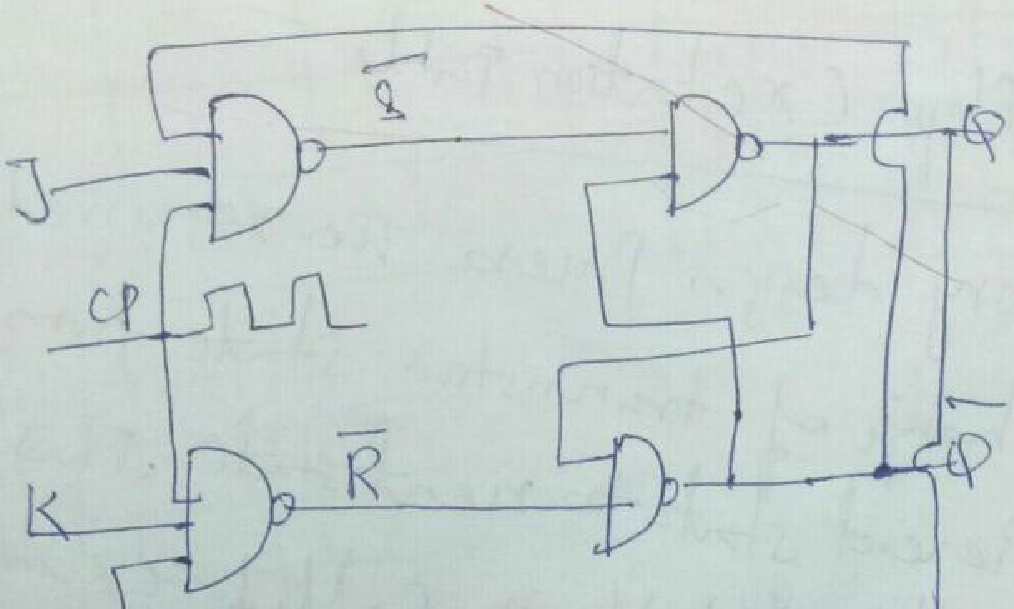


Q_n	J	K	Q_{n+1}
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

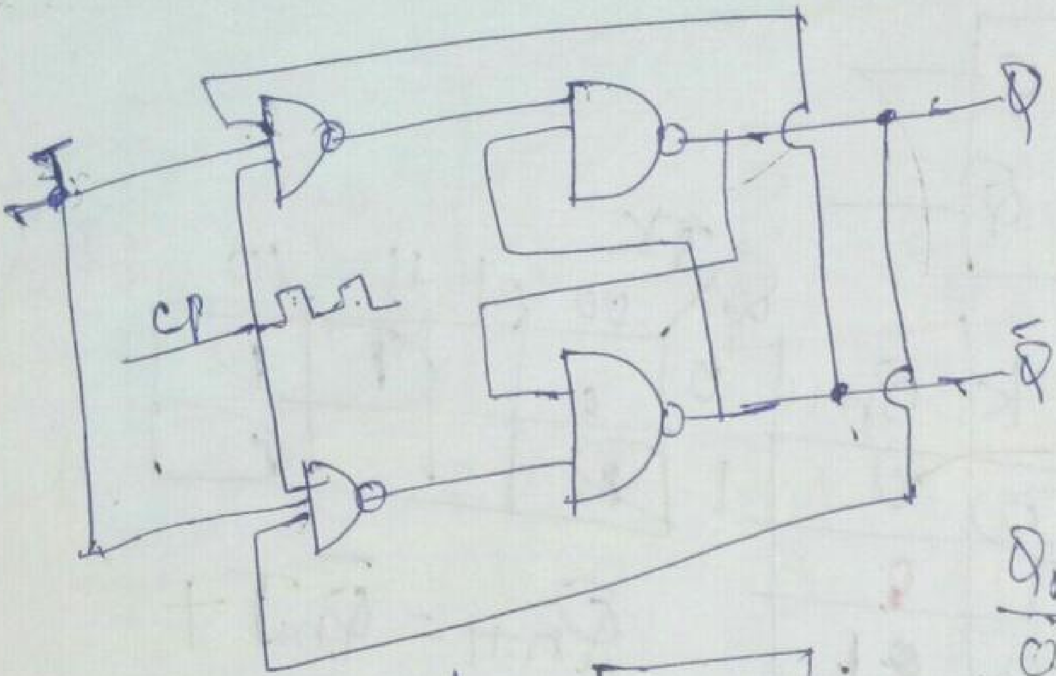
Q_n \ JK	00	01	11	10
0	0		1	1
1	1			1

$$Q_{n+1} = \bar{Q}_n J + Q_n \bar{K}$$

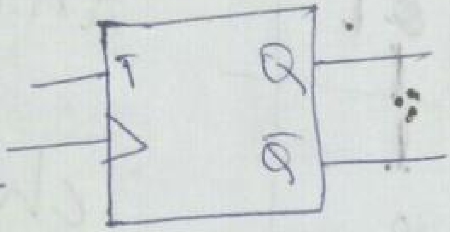
characteristic equation.



T FlipFlop



Logic symbol



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

characteristic equation: $Q_{n+1} = T\bar{Q}_n + \bar{T}Q_n$

FlipFlop Excitation Table

During design Process we required change of transistion state from Present state to next state. This state of table is called excitation table.

SR Flipflop

Truth Table

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

Excitation Table

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

D flipflop Exitation Table

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

JK f.f Excitation 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 FF excitation Table

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

Realization of one flipflop using other flipflop

SR flipflop to D flipflop

SR D

Input D	PS Q_n	NS Q_{n+1}	Flipflop Inputs S	R
0	0	0	0	X
1	0	1	1	0
0	1	0	0	1
1	1	1	X	0

K Map simplification

For S

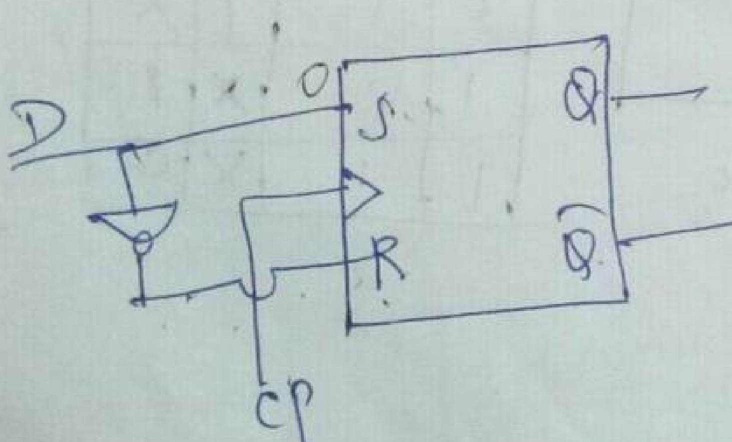
D \ Q_n	0	1
0		
1	1	X

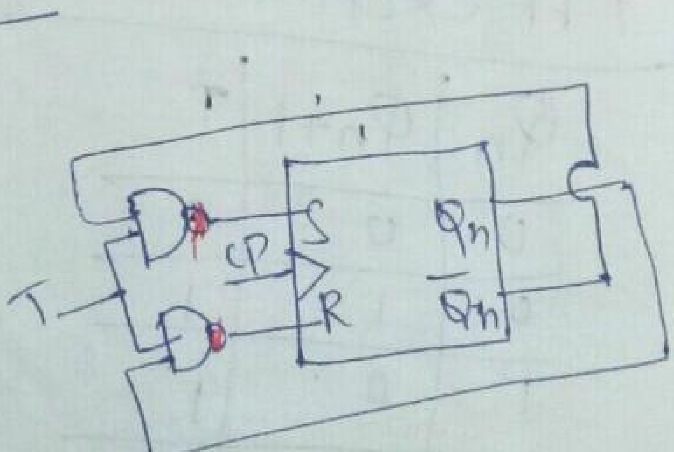
$: D$

For R

D \ Q_n	0	1
0	X	1
1		

$\bar{D}$





SR to JK flipflop

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

For S

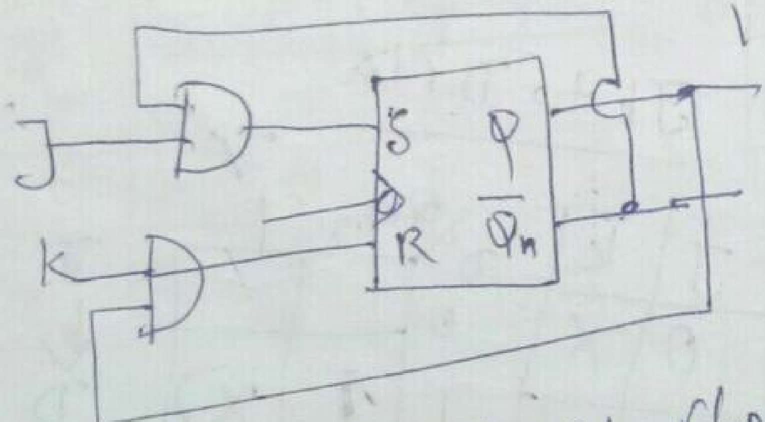
J \ KQ_n	00	01	11	10
0	$_0$	X_1	$_3$	$_2$
1	1_4	X_5	$_7$	1_6

$S = J \cdot \overline{Q}_n$

For K

J \ KQ_n	00	01	11	10
0	X_0	$_1$	1_3	X_2
1	$_4$	$_5$	1_7	$_6$

$K = KQ_n$



SR flipflop to T flipflop

Input T	PS Q_n	NS Q_{n+1}	S	R
0	0	0	0	X
0	1	1	X	0
1	0	1	1	0
1	1	0	0	1

S//

T \ Q_n	0	1
0		X
1	1	

$S = T\overline{Q}_n$

T \ Q_n	0	1
0	X	
1		1

$R = TQ_n$

JK flipflop to T flipflop

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

SJ

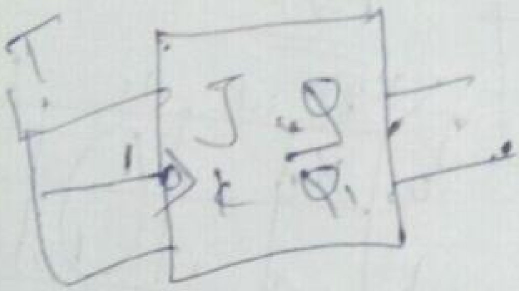
T \ Q_n	0	1
0		X
1	1	X

$\rightarrow J = T$

K

T \ Q_n	0	1
0	X	
1	X	1

$K = T$



~~D ff to T ff~~ JK to D f/f

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

J

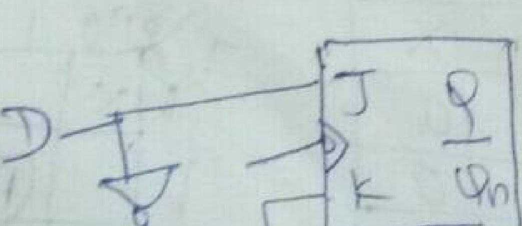
D \ Q_n	0	1
0		
1	1	X

$J \downarrow D$

K

D \ Q_n	0	1
0	X	1
1	X	

$K \downarrow \bar{D}$

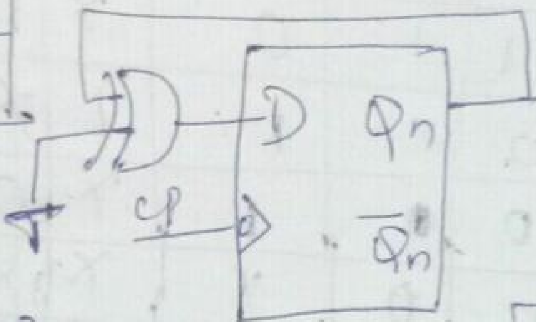


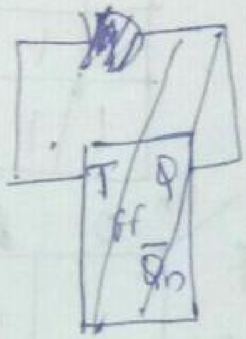
D flipflop T flipflop

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

T \ Q_n	0	1
0		1
1	1	

$$D = T\overline{Q_n} + \overline{T}\,Q_n$$



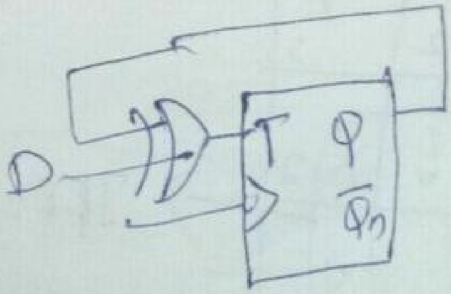


T flipflop to D flop flop

Input D	PS Q_n	NS Q_{n+1}	ff Input T
0	0	0	0
0	1	[illegible]	1
1	0	[illegible]	1
1	1	[illegible]	0

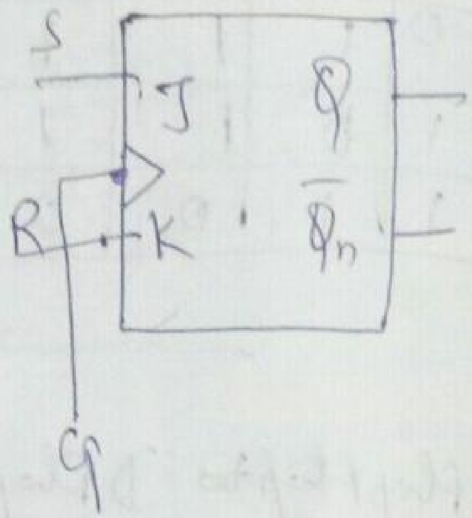
D \ Q_n	0	1
0		1
1	1	

$$T = D\overline{Q_n} + \overline{D}\,Q_n$$



JK flipflop to SR flipflop

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



SR \ Q_n	0	1
00	0	X 1
01	2	X 3
11	X 6	X 7
10	1 4	X 5

→ JFF = S

JFF

SR \ Q_n	0	1
00	X 0	1
01	X 2	1 3
11	X 6	X 7
10	X 4	5

→ R

K flipflop

D flipflop to SR flipflop

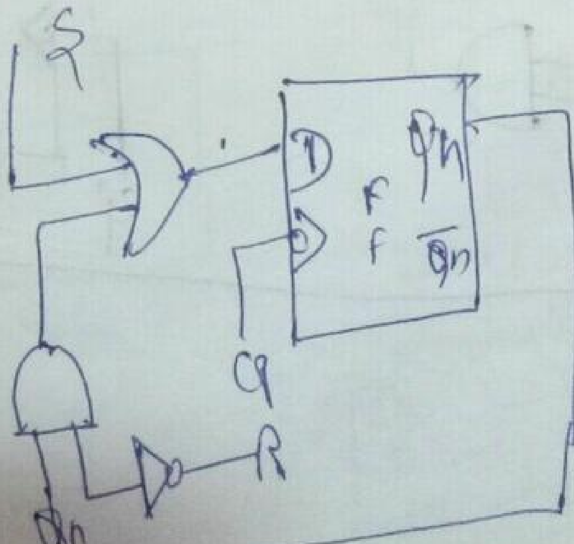
	S	R	Q_n	Q_{n+1}	D
0	0	0	0	0	0
1	0	0	1	1	1
2	0	1	0	0	0
3	0	1	1	0	0
4	1	0	0	1	1
5	1	0	1	1	1
6	1	1	0	X	X
7	1	1	1	X	X

SR \ Q_n	0	1
00		1
01		
11	X	X
10	1	1

→ $\bar{R} Q_n$

→ S

$$D = S + \bar{R} Q_n$$



T flipflop to SR flipflop

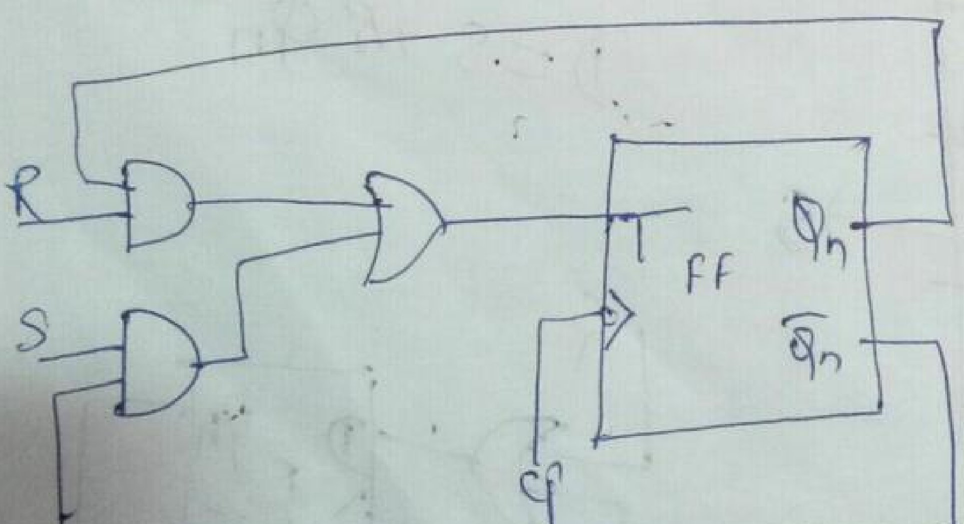
	S	R	Q_n	Q_{n+1}	T
	0	0	0	0	0
	0	0	1	1	0
	0	1	0	0	0
3	0	1	1	0	1
4	1	0	0	1	1
	1	0	1	1	0
6	1	1	0	X	X
7	1	1	1	X	X

SR \ Q_n	0	1
00	0	1
01	2	1 3
11	X 6	X 7
10	1 4	5

→ $Q_n R$

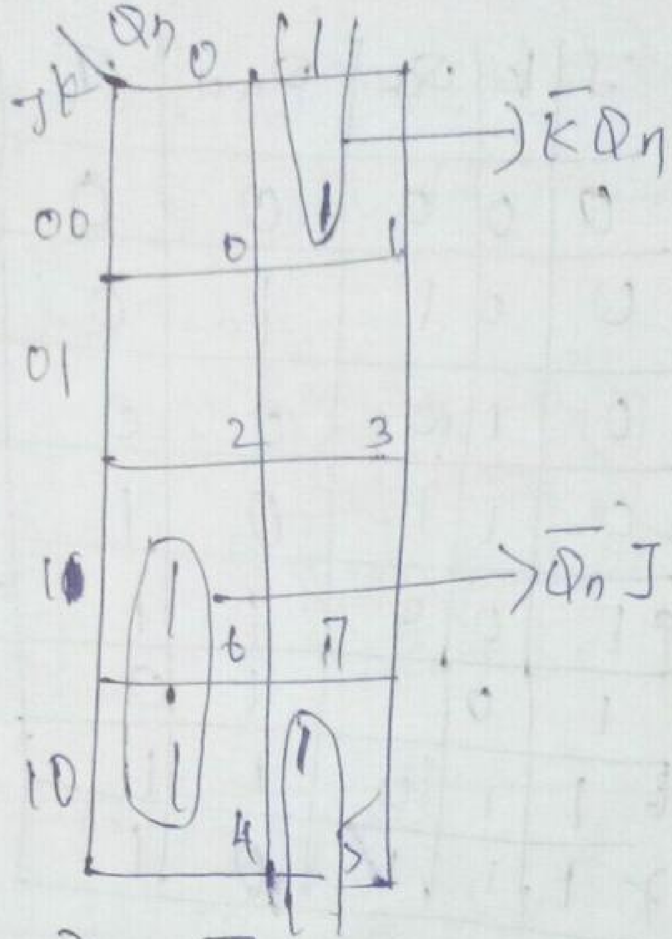
→ $\bar{Q}_n S$

$T = \bar{Q}_n S + Q_n R$

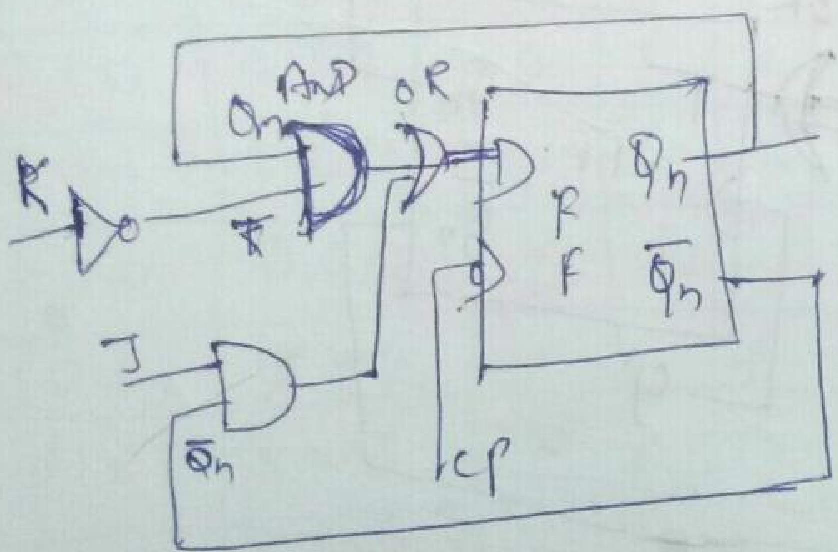


D flipflop to JK flipflop

	J	K	Q_n	Q_{n+1}	D
	0	0	0	0	0
1	0	0	1	1	1
	0	1	0	0	0
	0	1	1	0	0
4	1	0	0	1	1
5	1	0	1	1	1
6	1	1	0	1	1
	1	1	1	0	0



$$D = \bar{K}Q_n + \bar{Q}_n J$$



SR — JK, D, T } ③

JK — SR, D, T } ③

D — SR, JK, T

T — JK, D, SR

T flipflop to JK flipflop

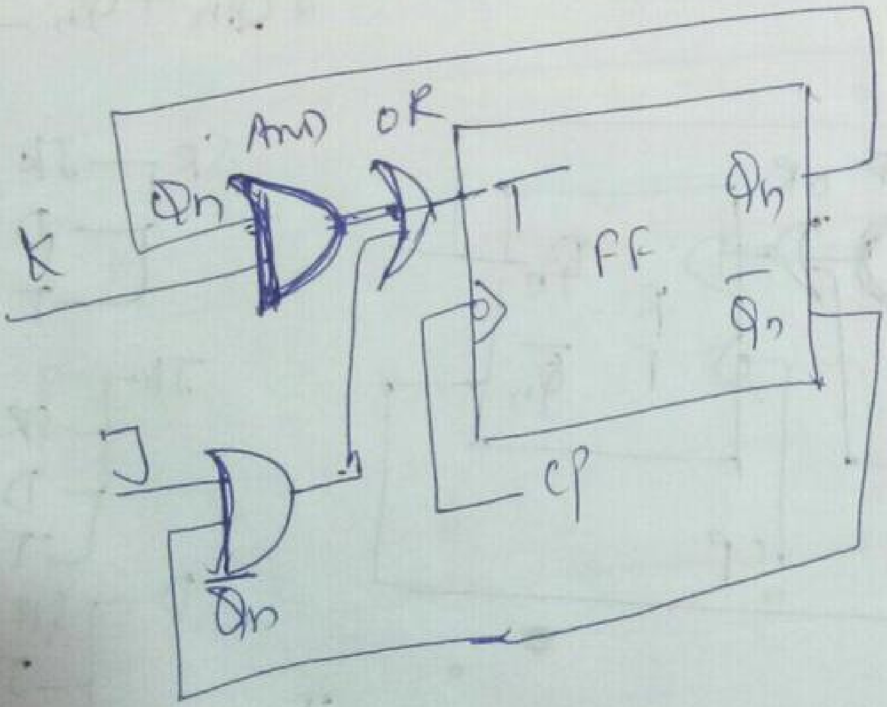
	J	K	Q_n	Q_{n+1}	T
	0	0	0	0	0
	0	0	1	1	0
	0	1	0	0	0
3	0	1	1	0	1
4	1	0	0	1	1
	1	0	1	1	0
6	1	1	0	1	1
7	1	1	1	0	1

JK \ Q_n	0	1
00	0	1
01	2	1 3
11	1 6	1 7
10	1 4	5

$\rightarrow K Q_n$

$\rightarrow \overline{Q_n} J$

$$T = K Q_n + \overline{Q_n} J$$



Determine a Mod 5 Synchronous Counter using JK flipflop and implement it.

Step 1: flipflop required 2^n = no of states

$2^3 = 8$ but in the sum 5. 3 flipflop required

Step 2: JK Excitation Table

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

a \ bc	00	01	11	10
0	1_0	1_1	1_3	1_2
1	1_4	5	7	6

$Y = \bar{a} + \bar{b}\bar{c}$

PS	NS	Flipflop Inputs					
Q_C Q_B Q_A	Q_{C+1} Q_{B+1} Q_{A+1}	J_C	K_C	J_B	K_B	J_A	K_A
0 0 0	0 0 1	0	X	0	X	1	X
0 0 1	0 1 0	0	X	1	X	X	1
0 1 0	0 1 1	0	X	X	0	1	X
0 1 1	1 0 0	1	X	X	1	X	1
1 0 0	0 0 0	X	1	0	X	0	X
1 0 1	~~X X X~~	X	X	X	X	X	X
1 1 0	X X X	X	X	X	X	X	X
1 1 1	X X X	X	X	X	X	X	X

By using Kmap

$Q_C \backslash Q_B Q_A$	00	01	11	10
0	0	1	1 3	2
1	X 4	X 5	X 7	X 6

For $J_C = Q_B Q_A$

$Q_C \backslash Q_B Q_A$	00	01	11	10
0	X 0	X 1	X 3	X 2
1	1 4	X 5	X 7	X 6

For $K_C = 1$

$Q_C \backslash Q_B Q_A$	00	01	11	10
0	0	1 1	X 3	X 2
0	4	X 5	X 7	X 6

For $J_B = Q_A$

$Q_C \backslash Q_B Q_A$	00	01	11	10
0	0	X 1	1 3	2
1	X 4	X 5	X 7	X 6

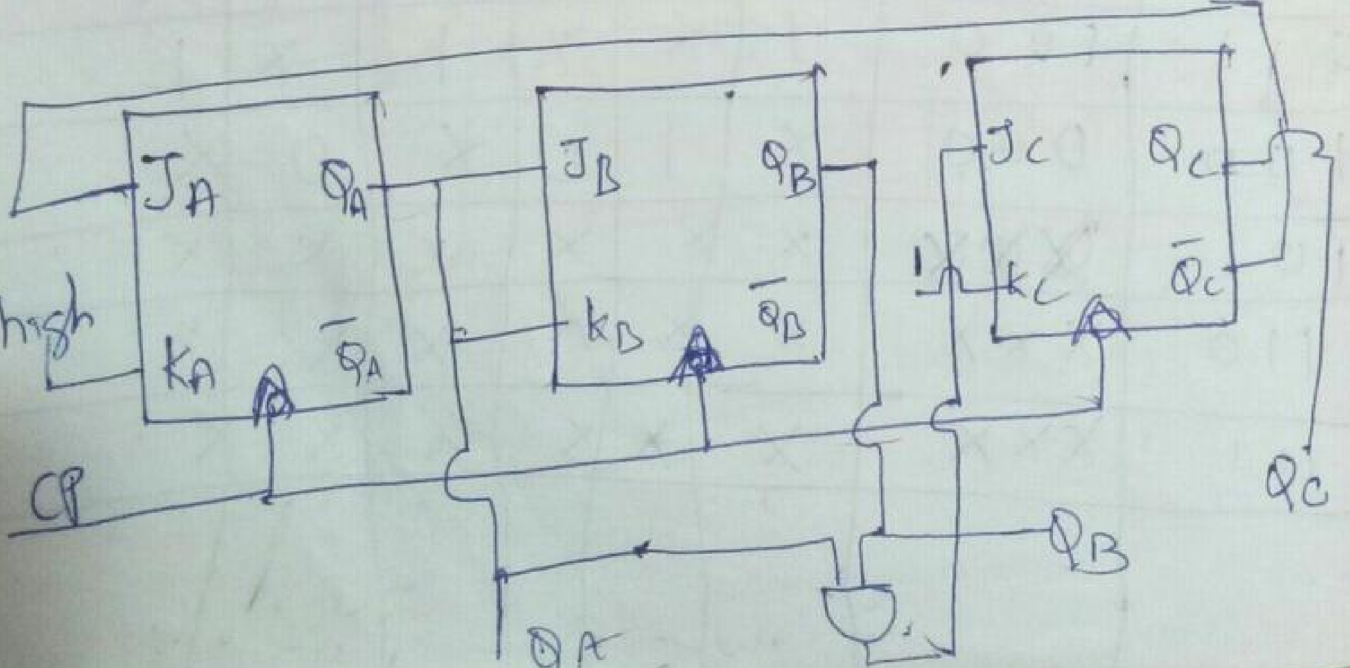
For $K_B = Q_A$

$Q_C \backslash Q_B Q_A$	00	01	11	10
0	1 0	X 1	X 3	1 2
1	4	X 5	X 7	X 6

For $J_A = \overline{Q}_C$

$Q_C \backslash Q_B Q_A$	00	01	11	10
0	X 0	1 1	1 3	X 2
1	X 4	X 5	X 7	X 6

For $K_A = 1$



Design and explain the working of a synchronous mod 3 counter

$N=3 \quad 2^2 = 4 \ (00, 01, 10, 11)$ PS $(00, 01, 10, 00)$ NS

	PS		NS		flip flop Input			
	Q_B	Q_A	Q_{B+1}	Q_{A+1}	J_B	K_B	J_A	K_A
0	0	0	0	1	0	X	1	X
1	0	1	1	0	1	X	X	1
2	1	0	0	0	X	1	0	X
3	1	1	X	X	X	X	X	X

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

$J_B = Q_A$

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

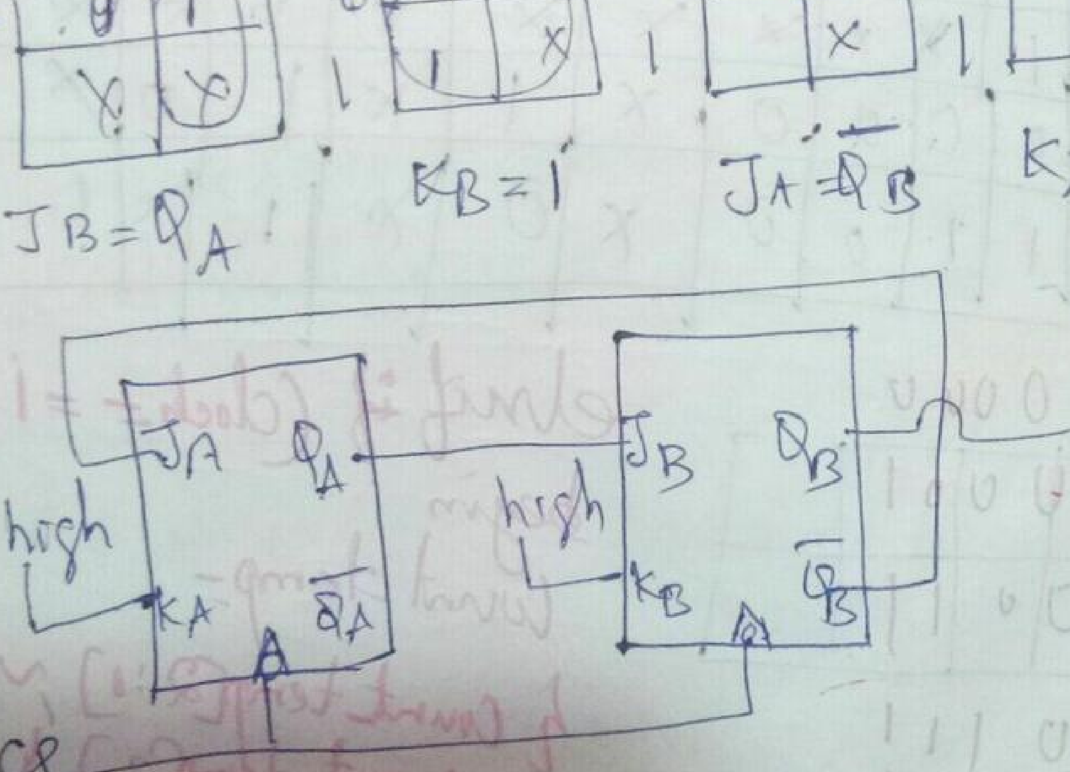
$K_B = 1$

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

$J_A = \overline{Q_B}$

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

$K_A = 1$



Design a Synchronous counter using JK FF to count the following sequence
7, 4, 3, 1, 6, 0, 7

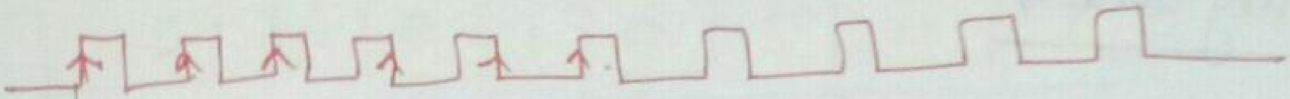
PS			NS			Flip flop Inputs					
C	B	A	C^+	B^+	A^+	J_C	K_C	J_B	K_B	J_A	K_A
0	0	0	1	1	1	1	X	1	X	1	X
0	0	1	1	1	0	1	X	1	X	X	1
0	1	0	X	X	X	X	X	X	X	X	X
0	1	1	0	0	1	0	X	X	1	X	0
1	0	0	0	1	1	X	1	1	X	1	X
1	0	1	X	X	X	X	X	X	X	X	X
1	1	0	0	0	0	X	1	X	1	0	X
1	1	1	1	0	0	X	0	X	1	X	1

0 0 0 0

	0 0	0 1	
0	0	1 1	

0 1 1 1
1 1 1
1 1 0 0
1 0 0 0
0 0 0 0

```
elsif if (clock = = 1'b1)
begin
count_temp =
{count_temp[2:0], ~count_temp[3]}; end
end assign count_out = count_temp
endmodule
```



```
module four_bit_ring_counter(
input clock, reset,
output [3:0] q); reg[3:0]a;
always@(posedge clock)
  if(reset)
        a=4'b0001;
  else
        begin
           a <= a<<1   blocking
           a[0] <= a[3]   Assignment
        end
assign q=a
end module
```

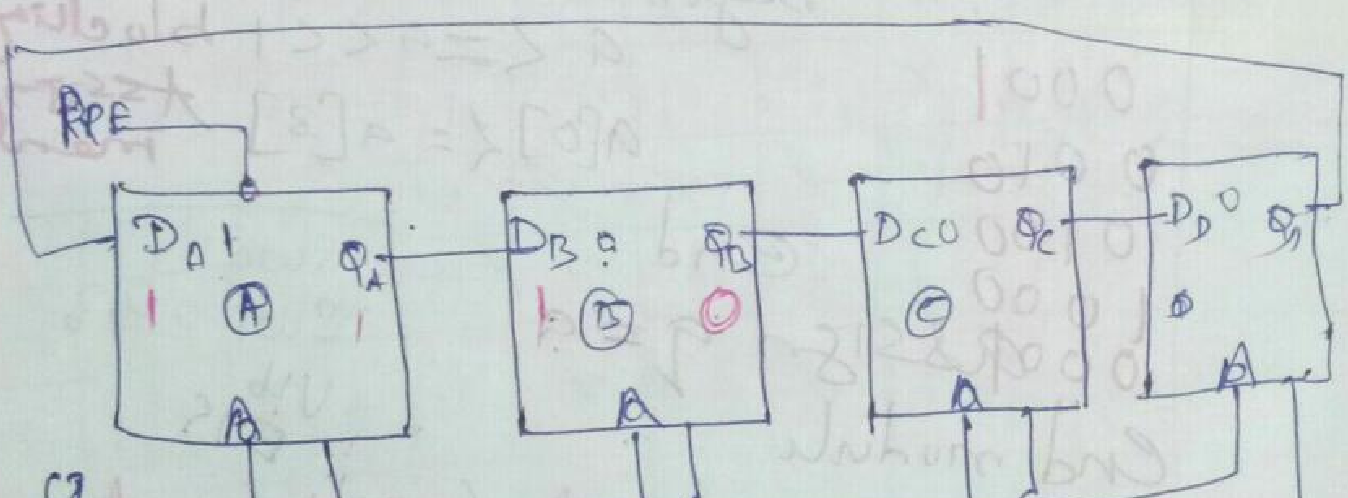
0001
0010
0100
1000
0001

Sw0
Sw1   U, 6.
U16
E15

```
module johnson_counter(clock, reset, count_out);
input clock, reset; output[3:0] count_out;
reg[3:0] count_temp;
always@(posedge clock or reset)
  begin if(reset == 1'b1) begin
     count_temp = 4'b0000; end
```

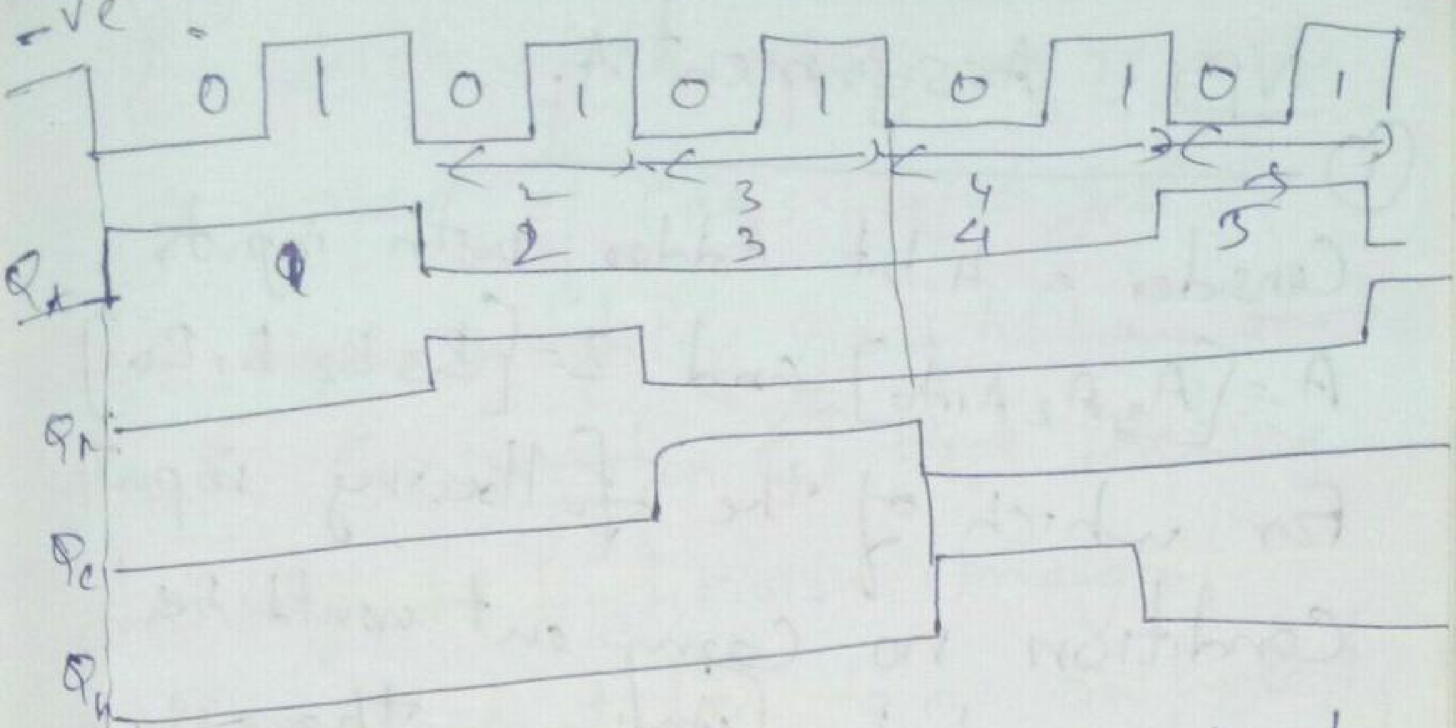
Ring counters

cp	Q_A	Q_B	Q_C	Q_D
0	1	0	0	0
1	0	1	0	0
2	0	0	1	0
3	0	0	0	1
4	1	0	0	0

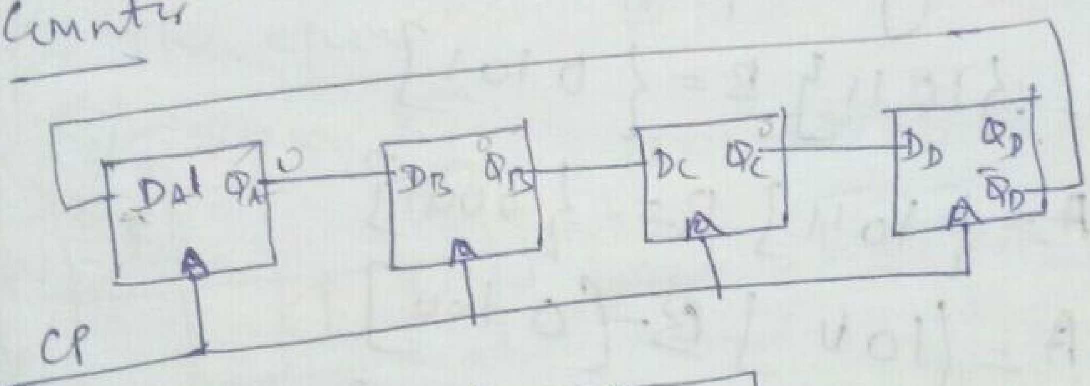


even we can give PRE = CLR

PRE is used for take prev value of FF $Q_D \Rightarrow$ to D.



Johnson (or) Twisting Ring (or) Switch Tail Counter



CP	Q_A	Q_D	Q_C	Q_D
↓	1	0	0	0
↓	1	1	0	0
↓	1	1	1	0
↓	1	1	1	1
↓	0	1	1	1
↓	0	0	1	1
	0	0	0	1

NPTEL Assignment 4.

(1)

Consider a 4 bit Adder with inputs $A = [A_3 A_2 A_1 A_0]$ and $B = [B_3 B_2 B_1 B_0]$ For which of the following input condition no carry out would be generated. [Assume there is no carry input]

a) A = {1011} B = {0101}

b) A = {1011} B = {0001}

c) A = [1011] B = [0100]

d) A = [1001] B = [0101]

a)
$$\begin{array}{r} 1011 \\ 0101 \\ \hline 10000 \end{array}$$

b)
$$\begin{array}{r} 1011 \\ 0001 \\ \hline 1100 \end{array}$$

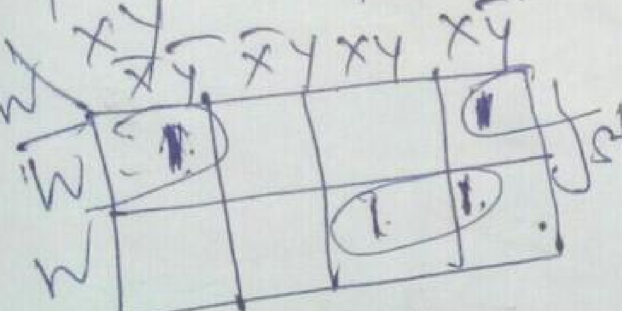
c) nc
$$\begin{array}{r} 1011 \\ 0100 \\ \hline 1111 \end{array}$$

d)
$$\begin{array}{r} 1001 \\ 0101 \\ \hline 1110 \end{array}$$

2) Which of the following is not a property of CMOS logic gates?

a) High switching speed b) Low static Power consumption c) High packing density d) High noise margin

3) The Product term to be included to remove possible static hazard for the function $wx+\bar{w}\bar{y}$ is



$wxy + wx\bar{y} +$

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

4) An odd parity circuit with 2^n inputs can be built with ____ XOR gates.

a) 2^n b) 2^{n+1} c) 2^n-1 d) $\log_2(n)$