

Digital Logic Design (Summary note) (Advanced)

① AND, OR, NOT Gates

Ex: I take an umbrella if it is cloudy or if it is raining.

[boolean variables.]

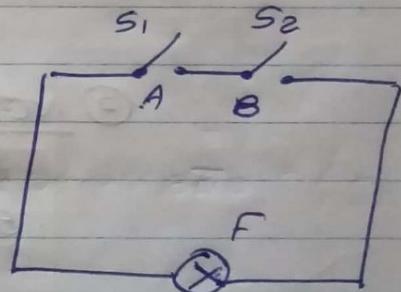
(OR gate applies)

cloudy	raining	takes an umbrella
0	0	0
0	1	1
1	0	1
1	1	1

② AND gate

A	B	$A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

(Truth table)



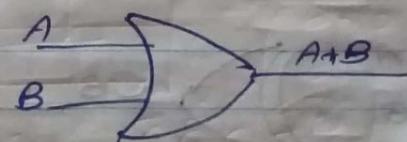
$$A \text{ and } B = 1$$

$$\underline{\underline{F = 1}}$$

③ OR gate

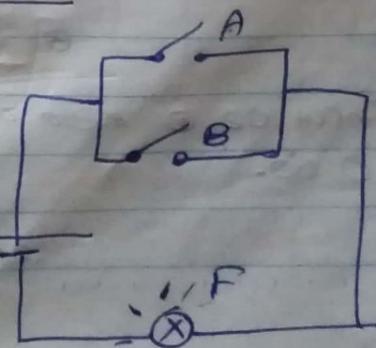
A	B	$A + B$
0	0	0
0	1	1
1	0	1
1	1	1

(Truth table)



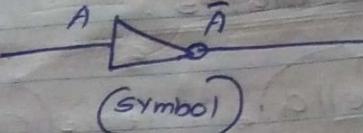
$$A \text{ OR } B = 1$$

$$\underline{\underline{F = 1}}$$



③ NOT gate

A	\bar{A}
0	1
1	0



Combinational gates

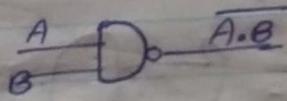
① NAND

② NOR

③ ~~XOR~~

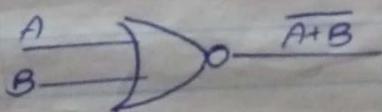
④ ~~X-NOR~~

① NAND



A	B	$A \cdot B$	$A \cdot B$
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

② NOR



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

③ X-OR



A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

④ Input 02 എന്നും തന്റെ Output
0 എംബ.

③ NOT gate

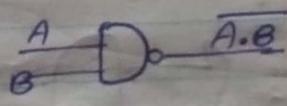
A	\bar{A}
0	1
1	0

(Symbol)

Combinational gates

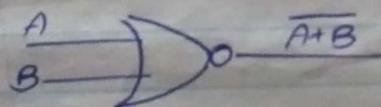
- ① NAND
- ② NOR
- ③ XOR
- ④ XNOR

① NAND



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

② NOR



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

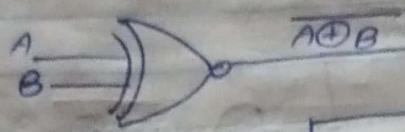
③ X-OR



A	B	$A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

④ Input 02 නිල සමඟ තුළ ඇත්තේ Output
0 යි.

④ X-NOR



⑤ Inputs 0 2 0 සමඟ මි
Output = 1 යේ.

A	B	$A \oplus B$
0	0	1
0	1	0
1	0	0
1	1	1

Universal Gate Implementing AND, OR NOT gates using NAND.

⑥ NAND සහ NOR න්‍යුත්තාක් පෙනී Circuits එහි logic gates
System of Universal Gate යේ.

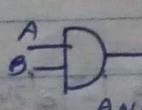
Benefits :-

- ① Economical, (Least cost)
- ② Easy to design.

⑦ NAND \rightarrow NOT design.



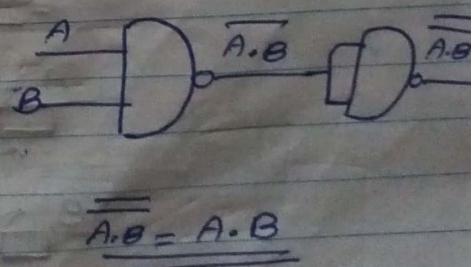
⑧ NAND \rightarrow AND design



A	B	$A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

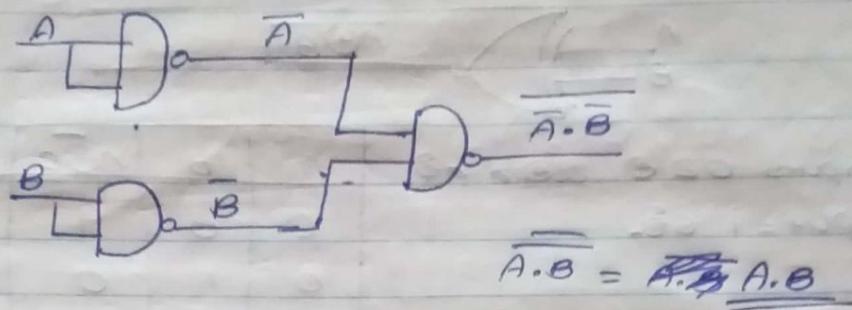
NAND \rightarrow AND

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



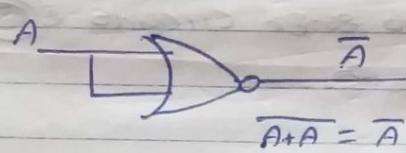
$$\overline{A \cdot B} = A \cdot B$$

① NAND \rightarrow OR



② Implementing AND, OR gates using NOR gate.

I) NOR \rightarrow NOT



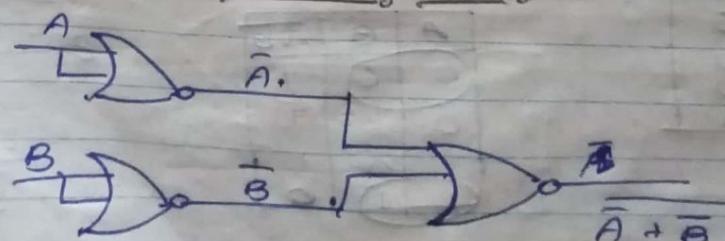
A	B	$A + B$
0	0	1
0	1	0
1	0	0
1	1	0

II) NOR \rightarrow OR



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

III) NOR \rightarrow AND



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

XOR NAND 3220020202 (IC)

A	B	F
0	0	0
0	1	1
1	0	1
1	1	0

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

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

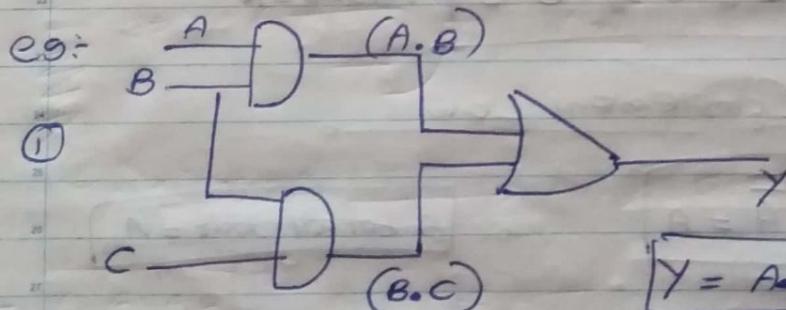
(Truth table)

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

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

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

Design of Circuits And truth tables.



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

Truth table :-

A	B	C	$A \cdot B$	$B \cdot C$	$A \cdot B + B \cdot C$
0	0	0	0	0	0
0	0	1	0	0	0
0	1	0	0	0	0
0	1	1	0	1	1
1	0	0	0	0	0
1	0	1	0	0	0
1	1	0	1	0	1
1	1	1	1	1	1

Laws of Boolean Algebra

① Idempotent Laws

$$\boxed{\begin{array}{l} A \cdot A = A \\ A + A = A \\ \bar{A} \cdot \bar{A} = \bar{A} \\ \bar{A} + \bar{A} = \bar{A} \end{array}}$$

② Identity Law

$$\boxed{\begin{array}{l} 1 \cdot A = A \\ 0 + A = A \\ 0 \cdot A = 0 \\ 1 + A = 1 \end{array}}$$

③ Complement / Inverse Law

$$\boxed{\begin{array}{l} A \cdot \bar{A} = 0 \\ A + \bar{A} = 1 \end{array}}$$

④ De-Morgan's Law

$$\boxed{\begin{array}{l} \overline{A+B} = \bar{A} \cdot \bar{B} \\ \overline{A \cdot B} = \bar{A} + \bar{B} \end{array}} \quad \begin{array}{l} \text{OR} \rightarrow \text{AND} \\ \text{AND} \rightarrow \text{OR} \end{array}$$

⑤ Double Complement Law

$$\boxed{\bar{\bar{A}} = A}$$

$$\boxed{\text{A bar of bar} = A}$$

⑥ Commutative Law

$$\boxed{\begin{array}{l} A \cdot B = B \cdot A \\ A + B = B + A \end{array}}$$

⑦ Associative Law

$$\begin{array}{l} (A \cdot B) \cdot C = A \cdot (B \cdot C) \\ (A+B)+C = A+(B+C) \end{array} \quad \begin{array}{l} = A \cdot B \cdot C \\ = A+B+C \end{array}$$

⑧ Redundancy Law

$$A + AB = A$$

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

$$\therefore A = \underline{\underline{A+AB}}$$

Boolean Expression Simplifying Using Boolean Rules.

$$\text{ex: } xy + x(y+z) + y(y+z)$$

①

(Distributive law)

$$\underbrace{xy + xy}_{xy} + xz + \underbrace{yy + yz}_{y + yz} \quad (\cancel{\text{Redundancy}})$$

(Idempotent law) $AA = A$
 $A+A = A$

$$xy + xz + y$$

(Redundancy law) $A+AB = A$

$$\underbrace{y + yx}_{y + y} + xz$$

(Commutative law) $xy = yx$

$$\boxed{y + xz}$$

(Redundancy law)

$$y + yz = y$$

ex- ② $(A+B)(A+C)$

$\leftarrow \oplus$

$$\underbrace{AA + AC}_{A + AC} + BA + BC \quad (\text{Distributive law})$$

$$\underbrace{A + AC}_{A + A} + BA + BC \quad (\text{Idempotent law}) \quad A + A = A$$

$$A + BA + BC \quad (\text{Redundancy law}) \quad A + AC = A$$

$$A + AB + BC$$

$$A + BC$$

↓

$$\boxed{\overline{A+BC}}$$

Karnaugh Map (SOP Concepts) ($\Sigma \rightarrow 1$) (Sum of Product)

ex- ① $AB = 2^2$

AB	00	01	10	11
00	0	0	1	1
01	0	1	0	1
10	1	0	0	1

$$0 = \bar{A} \quad \bar{A}B$$

$$1 = A$$

ex- ② $ABC (2^3)$

ABC	000	011	110	111
0	0	0	1	1
1	1	1	1	1

③ $ABCD = 2^4$

$ABCD$	0000	0101	1111	1010
00	0	0	1	1
01	0	1	0	1
11	1	0	0	1
10	1	1	1	0

$$ABC\bar{D} - ABC\bar{D}$$

$$es_1 = \textcircled{1} AB + A\bar{B}$$

(AB) ← two variables
 $2^2 = 4$

A	B	0	1
0	0		
1	1	1	1

$$\begin{matrix} AB \\ A\bar{B} \end{matrix}$$

$$\begin{matrix} A\bar{B} \\ AB \end{matrix} \rightarrow \underline{\underline{A}}$$

$$es_2 = \textcircled{2} \bar{A}\bar{B}C + \bar{A}\bar{B}C + A\bar{B}C + A\bar{B}C$$

A	BC	00	01	12 ³	18	10
0	0	1	1			
1	1	1	1			

$$\begin{matrix} \bar{A}\bar{B}C \\ \bar{A}\bar{B}C \\ A\bar{B}C \\ A\bar{B}C \end{matrix} = \underline{\underline{B}}$$

es₃

$$\textcircled{3} \bar{A}\bar{B}CD + \bar{A}\bar{B}C\bar{D} + A\bar{B}\bar{C}D + A\bar{B}CD + A\bar{B}\bar{C}\bar{D} + ABCD$$

$$2^4 = 16$$

AB	CD	00	01	11	10
00	0			1	
01	1			1	
11	1	1	1		
10	1	1	1		

$$\begin{matrix} \bar{A}\bar{B}CD \\ \bar{A}\bar{B}C\bar{D} \\ A\bar{B}\bar{C}D \\ A\bar{B}C\bar{D} \end{matrix}$$

$$\begin{matrix} \bar{A}\bar{B}CD \\ \bar{A}\bar{B}C\bar{D} \end{matrix} \quad \text{group 1} = \bar{A}\bar{B}C$$

$$\begin{matrix} A\bar{B}\bar{C}D \\ A\bar{B}C\bar{D} \\ A\bar{B}\bar{C}\bar{D} \\ A\bar{B}C\bar{D} \end{matrix}$$

$$A\bar{D}$$

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

$$\cancel{AB}$$

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

$\pi \rightarrow 0$

Karnaugh Map Using POS (Product of Sums)

(grouping group 0000, 0010, 0110, 0100)

$$\text{eg: } \text{OF}(ABC) = \sum (0, 3, 6, 7)$$

2^3 (3 elements)

		BC	00	01	11	10	
		A	0	0	1	0	2
		B	0	4	5	7	6
0	1	C	0	1	0	1	

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

$$\begin{array}{c} \boxed{ABC} \\ \boxed{AB\bar{C}} \end{array} \quad AB \longrightarrow (\bar{A} + \bar{B}) \quad (\text{POS})$$

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

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

$$\text{eg: } \text{Variable } 04, F(ABCD) = \sum (3, 5, 7, 8, 10, 11, 12, 13)$$

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

$$\begin{array}{c} \boxed{A\bar{B}CD} \\ \boxed{\bar{A}BCD} \end{array} \rightarrow \bar{A}CD \quad (\bar{A} + \bar{C} + D)$$

$$\begin{array}{c} \boxed{\bar{A}B\bar{C}D} \\ \boxed{A\bar{B}\bar{C}D} \end{array} \rightarrow B\bar{C}D \quad (\bar{B} + \bar{C} + D)$$

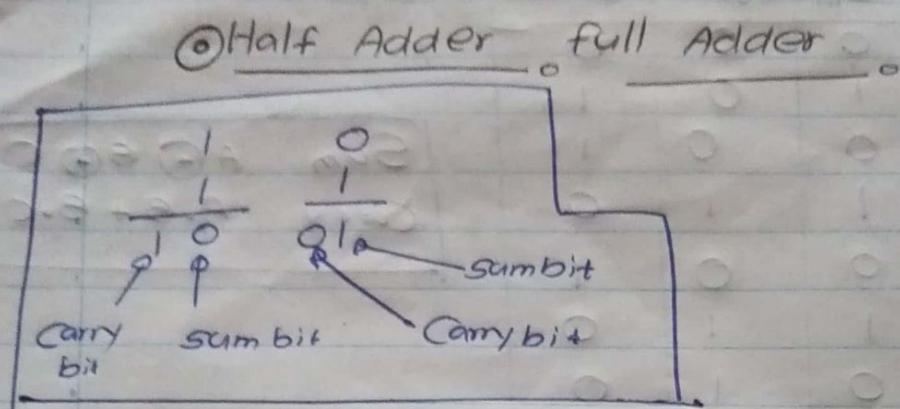
$$\begin{array}{c} \boxed{ABC\bar{D}} \\ \boxed{A\bar{B}\bar{C}\bar{D}} \end{array} \rightarrow A\bar{C}\bar{D} \rightarrow (\bar{A} + C + D)$$

$$\begin{array}{c} \boxed{A\bar{B}CD} \\ \boxed{A\bar{B}C\bar{D}} \end{array} \rightarrow A\bar{B}C \quad (\bar{A} + B + \bar{C})$$

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

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

DLD Part 02

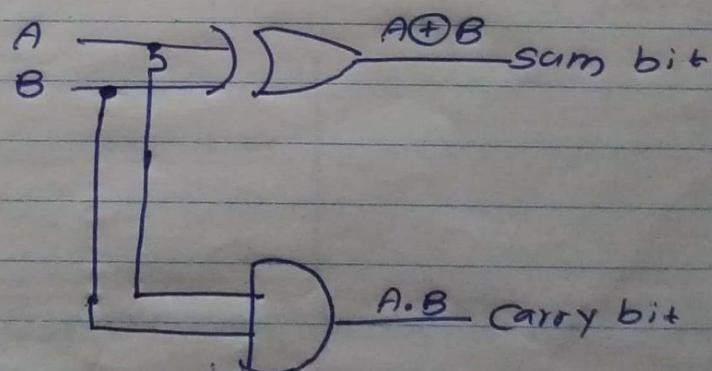


(XOR)

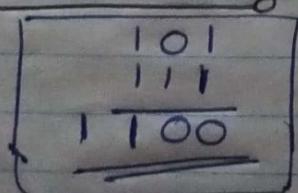
A	B	Carry bit	Sum bit
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

(AND)

② In Half adder,



③ Full adder



A	B	Cin	Code	Sum
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Sum $A \oplus B \oplus C$
 Carry $A \cdot B + B \cdot C + A \cdot C$

Design Of a full Adder

Full adder takes three inputs (A , B and Cin) and produces two outputs. (Sum , $Cout$)

① Truth table :

A	B	Cin	Sum	$Cout$
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

② For Sum (S) :

S	A	B	Cin
00	00	01	11 10
0	00	01	10
1	1	0	01

Simplified expression : $S = \bar{A}\bar{B}Cin + \bar{A}BCin + A\bar{B}Cin + ABCin$

For Cout (C) :

	A	B	Cin
00	00	01	11 10
0	0	0	11 11
1	0	1	11 11

Simplified expression : $C = \bar{A}B + A\bar{B}Cin + \bar{A}Cin$

No.....
For Sum $\Sigma m(1,3,7)$ \rightarrow $S = A\bar{B}\bar{C}in + \bar{A}BCin + A\bar{B}Cin + ABCin$

* $S = A\bar{B}\bar{C}in + \bar{A}BCin + A\bar{B}Cin + ABCin$ (Apply De Morgan theorem)

* $S = (\bar{A}\bar{B}Cin) * (\bar{A}BCin) * (A\bar{B}Cin) * (ABCin)$ (NOR equivalent)

* $S = (A+B+Cin) * (A+\bar{B}+Cin) * (\bar{A}+B+Cin) * (A+B+\bar{C}in)$ (NOR equivalent)

For Cout \therefore

$$C = AB + A\bar{B}Cin + \bar{A}Cin \quad (\text{Apply De Morgan theorem})$$

$$C = (\bar{A}B) * (\bar{A}\bar{B}Cin) * (\bar{A}Cin) \quad (\text{NOR equivalent})$$

$$C = (A+\bar{B}) * (\bar{A}+B+\bar{C}in) * (A+\bar{B}+\bar{C}in) \quad (\text{NOR equivalent})$$

② Design Circuit

$$\text{Sum} = A \oplus B$$

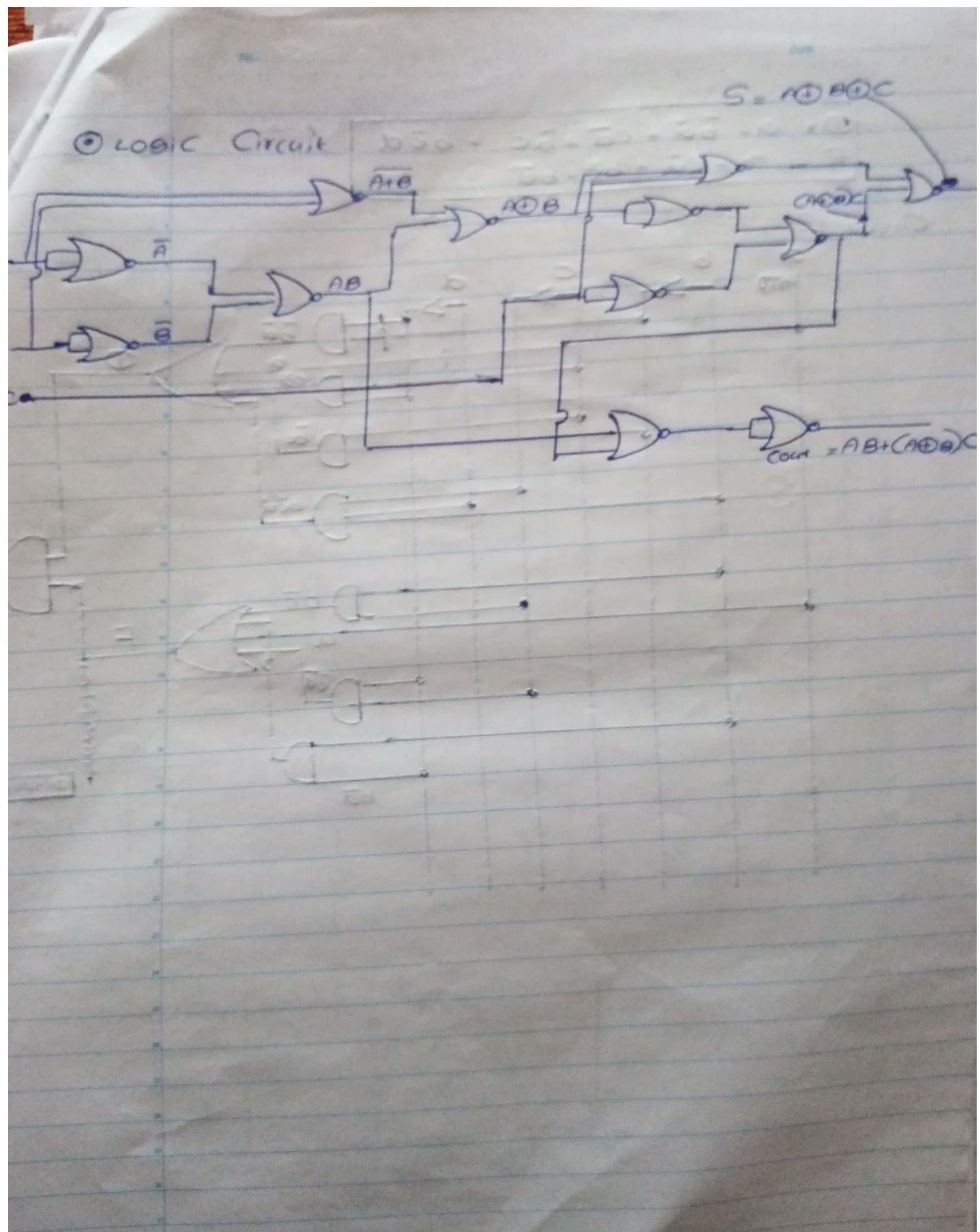
$$S = A \oplus B \oplus C$$

$$\text{Cout} = AB = \frac{\bar{A}\bar{D}}{\bar{A}+\bar{D}}$$

$$C = AB + (A \oplus B)C$$

$$\begin{aligned} \text{Sum} &= \bar{A}B + A\bar{B} \longrightarrow \bar{A}B + A\cdot\bar{A} + A\bar{B} + B\bar{B} \\ &= \bar{A}(B+A) + \bar{B}(A+\bar{B}) \\ &= (\bar{A}+\bar{B})(A+B) \end{aligned}$$

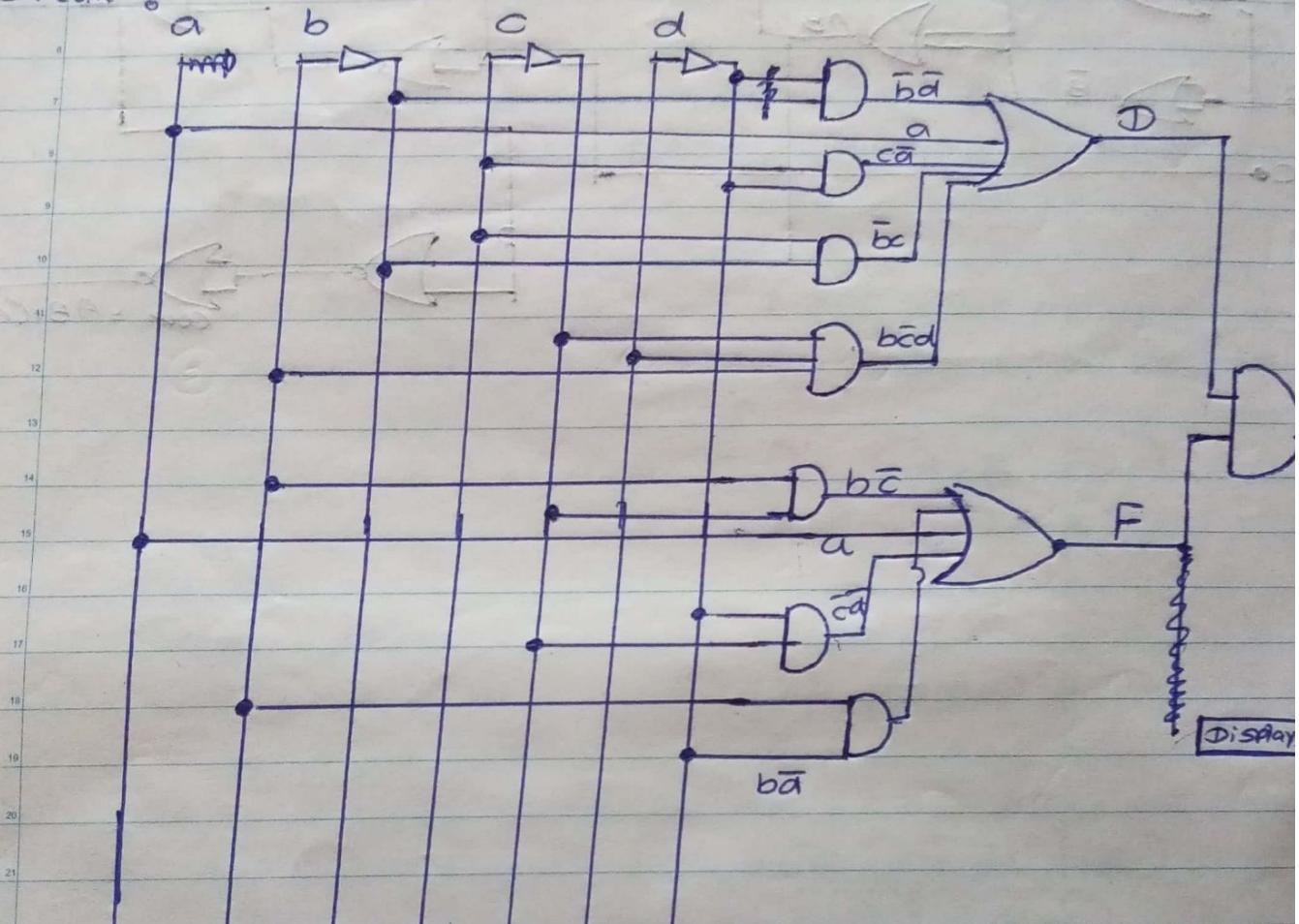
$$\text{Sum} = \underline{\underline{(A+\bar{B}) + (A+B)}}$$



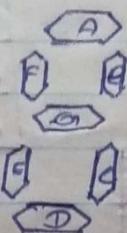
$$\textcircled{1} = a + \bar{b}\bar{d} + \bar{c}\bar{d} + \bar{b}c + b\bar{c}d$$

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

Circuit

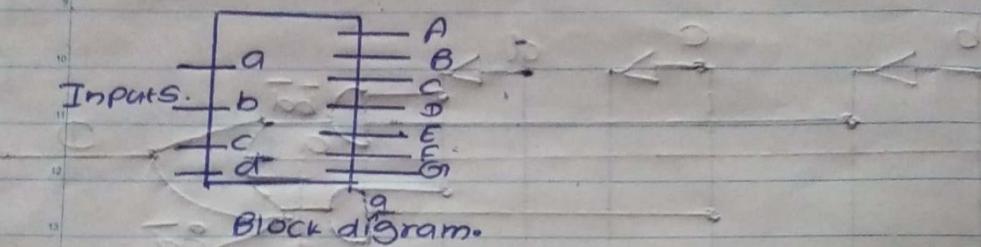


Design of a Driver for 7 segment Display.



segments.

① step :



② Step : Truth table

Inputs	Display				Seven Segment Code						
	a	b	c	d	A	B	C	D	E	F	G
0 0 0 0	0	1	1	1	1	1	1	1	1	1	0
0 0 0 1	1	0	1	1	0	1	1	0	0	0	0
0 0 1 0	2	1	1	0	1	1	0	1	1	0	1
0 0 1 1	3	1	1	1	1	1	1	0	0	0	1
0 1 0 0	4	0	1	1	0	1	1	0	0	1	1
0 1 0 1	5	1	0	1	1	0	1	1	0	1	1
0 1 1 0	6	1	0	1	1	1	0	1	1	1	1
0 1 1 1	7	1	1	1	1	1	1	0	0	0	0
1 0 0 0	8	1	1	1	1	1	1	1	1	1	1
1 0 0 1	9	1	1	1	1	1	1	1	0	1	1

③ Index No last digit is = 3

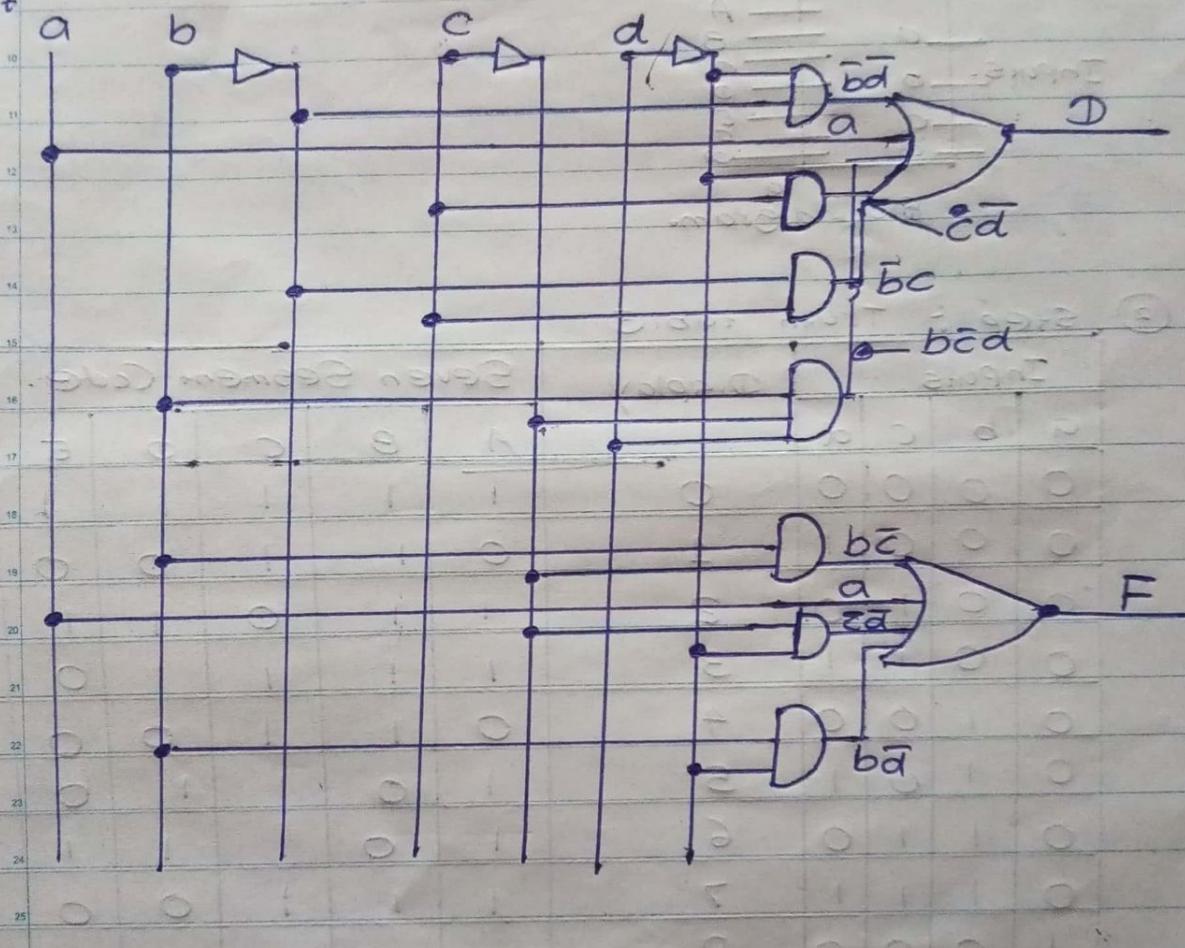
ab	cd	$\bar{c}\bar{a}$	$\bar{c}d$	$c\bar{d}$	cd	$\bar{c}\bar{a}$
$\bar{a}b$	1	0	1	1		
$\bar{a}b$	0	1	0	1		
$a\bar{b}$	0	0	0	0		
ab	0	0	0	0		
$a\bar{b}$	1	1	0	0		

$$\Phi = a + \bar{b}\bar{a} + c\bar{a} + \bar{b}c + b\bar{c}d$$

ab	cd	$\bar{c}\bar{a}$	$\bar{c}d$	$c\bar{d}$	cd	$\bar{c}\bar{a}$
$\bar{a}b$	1	0	0	0	0	0
$\bar{a}b$	0	1	0	1	0	1
$a\bar{b}$	0	0	0	0	0	0
ab	0	0	0	0	0	0
$a\bar{b}$	1	1	0	0	0	0

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

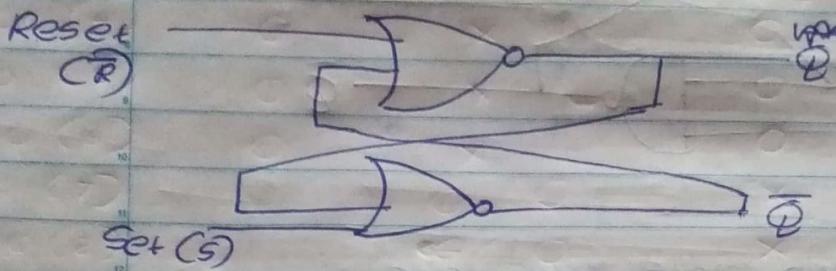
Circuit



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

SR Flip-Flop Latches

- I) SR Latch
II) Using NOR Gates.

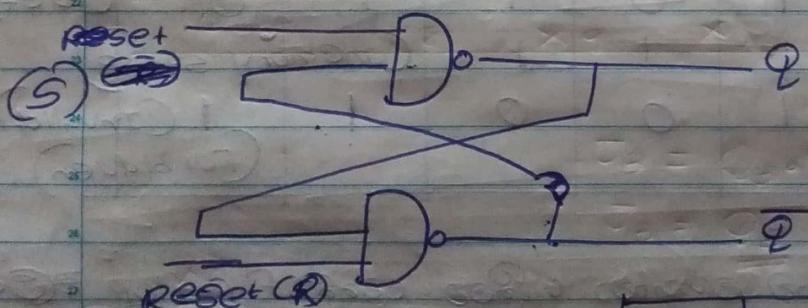


NOR Gate (truth table)

A	B	F
0	0	1
0	1	0
1	0	0
1	1	0

S	R	Q	\bar{Q}
0	1	0	1
0	0	(Memory state) Not change	
1	0	1	0
0	0	(Memory state) Not change	
1	1	0	0 (Invalid)

- II) Using NAND Gates.

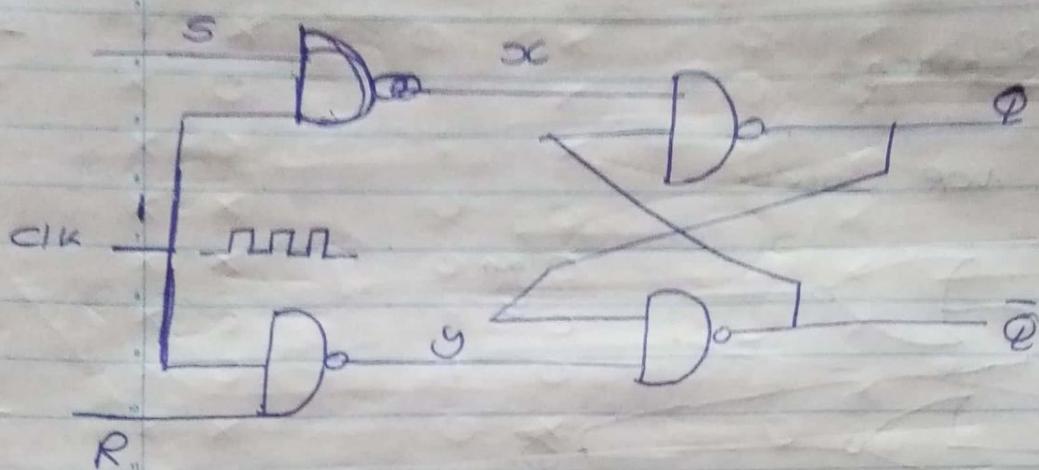


NAND Gate (truth table)

A	B	F
0	0	1
0	1	1
1	0	1
1	1	0

S	R	Q	\bar{Q}
0	1	1	0
1	1	(Memory state)	
1	0	0	1
1	1	(Memory state)	
0	0	(Invalid)	

SR Flip-Flop



NAND Gate truth table.

A	B	F
0	0	1
0	1	1
1	0	1
1	1	0

(Truth table for flip-flop)

S	R	C	x	y	Q	Q-bar
0	1	0	1	1		
0	1	1	1	0	0	1
1	0	1	0	1	1	0
0	0	1	1	1		
1	1	1	0	0	1	1

(Memory state)

(Reset)

(Set)

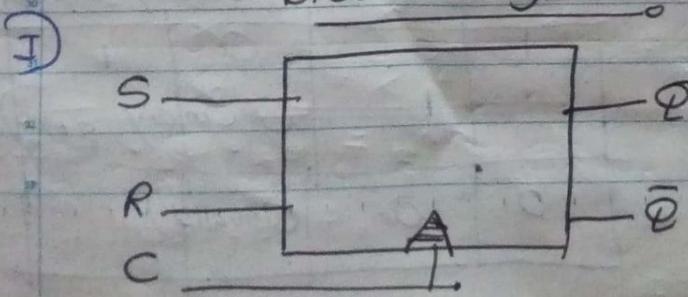
(Memory state)

(Invalid)

(Forbidden)

① Forbidden states entered in the SR flip-flop are more than 2. (J-K flip flop, T flip flop etc. have 2 states.)

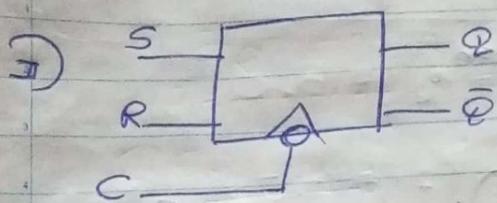
Block Diagram



② Q and Q-bar are complementary

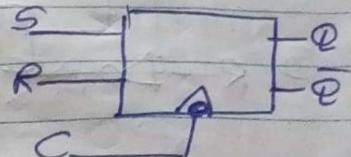
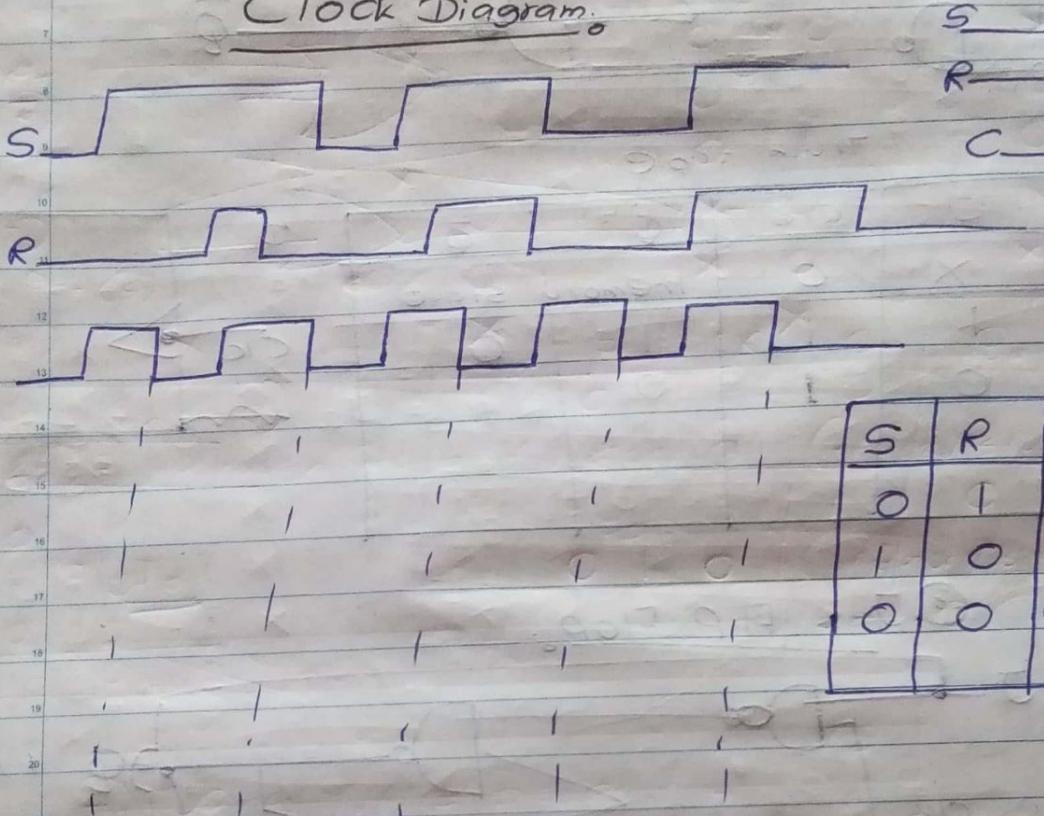
Clock signal 0/1 is

seen at the C terminal.



○ (Clock) Clock Signal 012.

Clock Diagram.



S	R	Q	\bar{Q}
0	1	0	1
1	0	1	0
0	0	(Memory state)	

② D-Flip flop

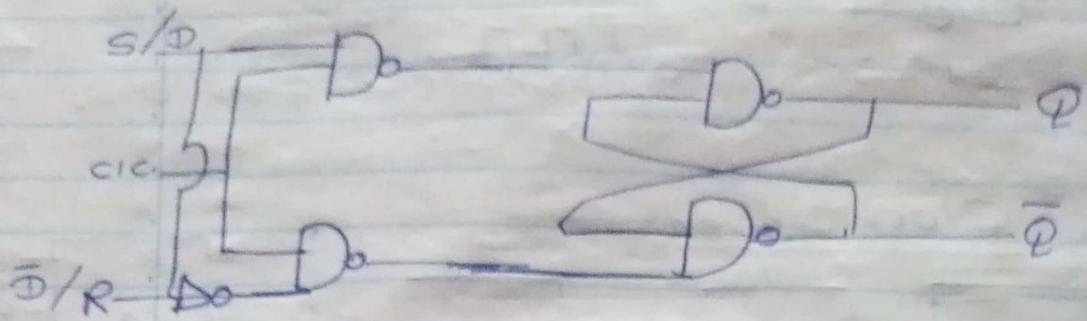
SR Latch Table

S	R	Q	\bar{Q}
0	1	1	0
1	0	0	1
0	0	Invalid	
1	1	Memory State.	

SR Flip Flop T-table.

S	R	C	Q	\bar{Q}
0	1	1	0	1
1	0	1	1	0
0	0	1	(Memory state)	
1	1	1	(Invalid)	

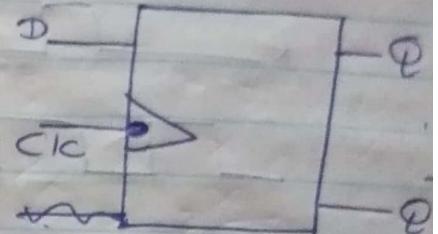
D-Flip Flop



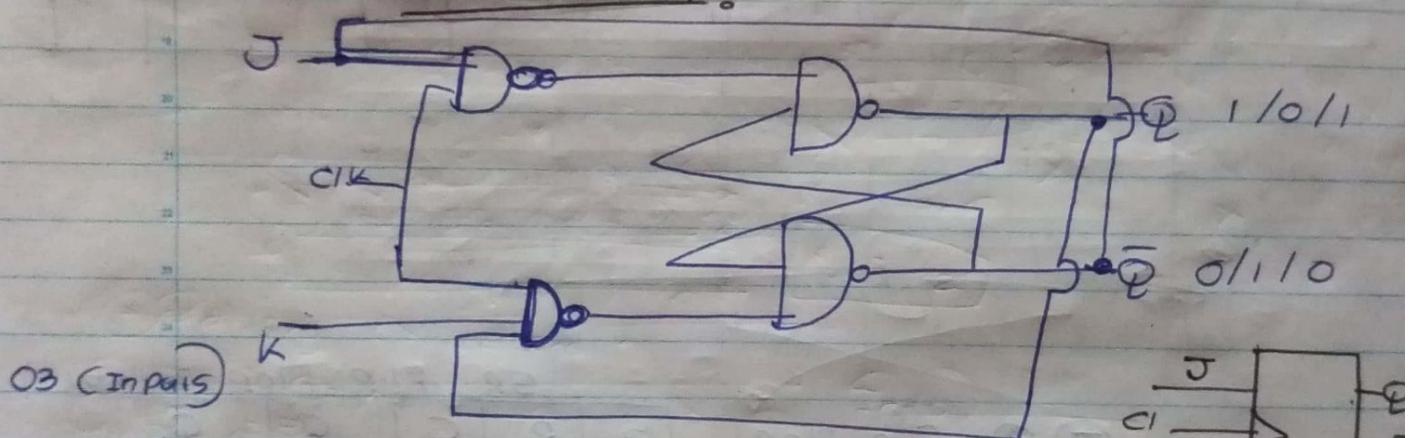
Truth table.

($D = 0/1$)

D	CIC	Q	Q-bar
X	0	memory state.	
1	1	1	0
0	1	0	1

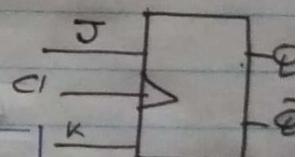


③ J-K Flip Flop



03 (Inputs)

C	J	K	Q	Q-bar
0	X	X	(memory state)	
1	0	1	0	1
1	1	0	1	0



Truth table

* ($J=1, K=1$) එම toggle යි.

ඉතුරු එම toggle එකට ප්‍රාග්ධනය කිරීමේදී Invalid නොවා

Ex: Design a Synchronous Counter Using JK
Flip-Flop for getting the following sequence.
 (0-1-3-5-7) → 0?

(Step 1) :

J-K transition table

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

Truth table.

Present State ($C=0$)	Next State ($C=1$)	B	C	D	B	C	D	J_2	K_2	J_1	K_1	J_0	K_0
0 0 0	0 0 1	0	0	0	0	0	1	0	X	0	X	1	0
0 0 1	0 1 1	0	1	1	0	1	1	0	0X	1	0	X	0
0 1 1	1 0 1	1	0	1	1	0	1	1	0	X	1	X	0
1 0 1	1 1 1	1	1	1	-	-	-	X	0	1	0	X	0
1 1 1	0 0 0	1	1	1	-	-	-	X	1	X	1	X	1

(Step 2) :

Present State table.

B	C	D	J_2	K_2	J_1	K_1	J_0	K_0
0	0	0	0	X	0	X	1	0
0	0	1	0	X	1	0	X	0
0	1	1	1	0	X	1	X	0
1	0	1	X	0	1	0	X	0
1	1	1	X	1	X	1	X	1

i) Karnaugh P-map to J_2

CD \ B		00	01	11	10
0	0	0	1	X	2
1	X	X	5	X	6

$\bar{B}CD$

$$J_2 = C$$

$$= \bar{B}CD$$

$$J_2 =$$

II) Karnaugh map to K_2 .

	CD 00	01	11	10
B 0	x_0	x_1	x_3	x_2
B 1	x_4	x_5	x_7	x_6

$$K_2 = B \cdot C$$

(B C D)

$$K_2 = B \cdot J_2$$

III) Karnaugh map to J_1 .

	CD 00	01	11	10
B 0	0, 0	1, 1	x_3	x_2
B 1	x_4	x_5	x_6	x_6

$$J_1 = D.$$

(B C D)

IV) Karnaugh map to K_1 .

	CD 00	01	11	10
B 0	x_0	0, 1	1, 3	x_2
B 1	x_4	x_5	1, 7	x_6

$$K_1 = C$$

(B C D)

$$K_1 = J_2$$

$$K_1 = C \cdot D$$

V) Karnaugh map to J_0 .

	CD 00	01	11	10
B 0	1, 0	x_1	x_3	x_2
B 1	x_4	x_5	x_7	x_6

$$J_0 = 1 \text{ as } A = 1$$

$$\boxed{J_0 = A} / J_0 = 1$$

VI) Karnaugh map to K_0 .

	CD 00	01	11	10
B 0	0, 0	0, 1	0, 3	x_2
B 1	x_4	0, 5	1, 7	x_6

$$K_0 = B \cdot C$$

$$\boxed{K_0 = K_2}$$

Step(3) :-

* Simplifying Expressions.

$$① J_2 = C = K_1$$

OB Bits.

(Info)

$$D = Q_0$$

$$C = Q_1$$

$$B = Q_2$$

$$② J_1 = D$$

$$③ J_0 = A = 1$$

$$④ K_2 = B \cdot J_2 = K_0$$

Step (4) \div Circuit Diagram

