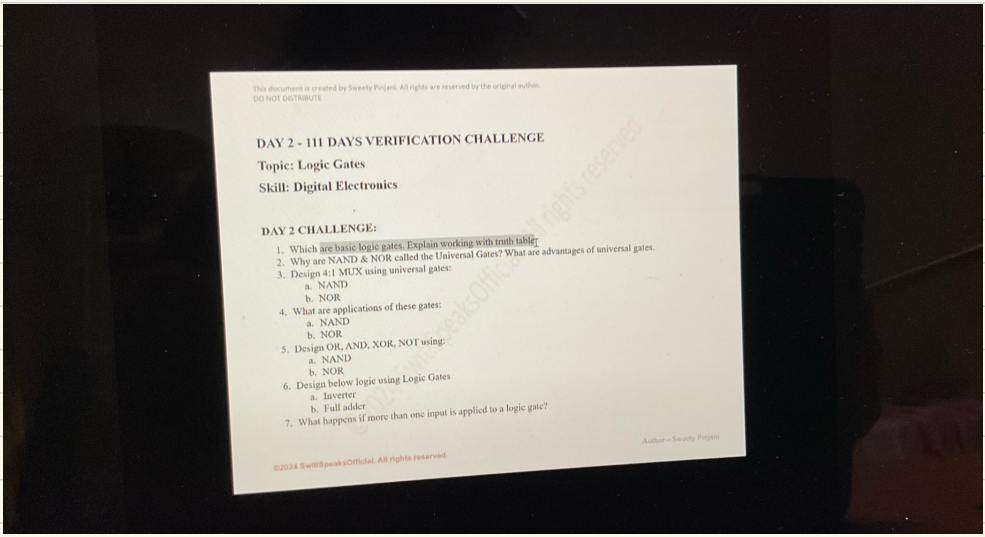
Day 2. 2nd Sept / 3rd Sept



DAY 2 - 111 DAYS VERIFICATION CHALLENGE

Topic: Logic Gates

Skill: Digital Electronics

DAY 2 CHALLENGE:

1. Which are basic logic gates. Explain working with truth table.
2. Why are NAND & NOR called the Universal Gates? What are advantages of universal gates.
3. Design 4:1 MUX using universal gates:
   a. NAND
   b. NOR
4. What are applications of these gates:
   a. NAND
   b. NOR
5. Design OR, AND, XOR, NOT using:
   a. NAND
   b. NOR
6. Design below logic using Logic Gates
   a. Inverter
   b. Full adder
7. What happens if more than one input is applied to a logic gate?

Author – Sweety Pinjani

Day 2

2nd Sept 2024

A1) Basic logic gates

(a) AND

A	B	out
0	0	0
0	1	0
1	0	0
1	1	1

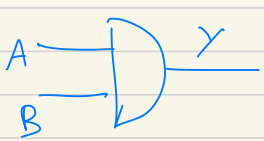
(b) OR

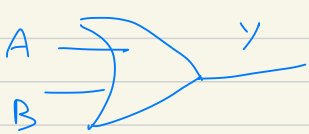
A	B	out
0	0	0
0	1	1
1	0	1
1	1	1

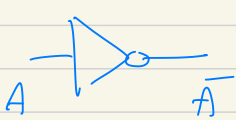
(c) NOT

A	out
0	1
1	0

These are called basic gates as they make the building blocks of digital electronics.







A2) NAND & NOR are called universal gates because using only NAND or NOR all the gates can be implemented. Any circuit can be implemented from those as well

The advantage of universal gates?

The main advantage is the number of transistors used. The no. of tx used is more in case of OR & AND gates. ∵ of the inverting nature of the gates.

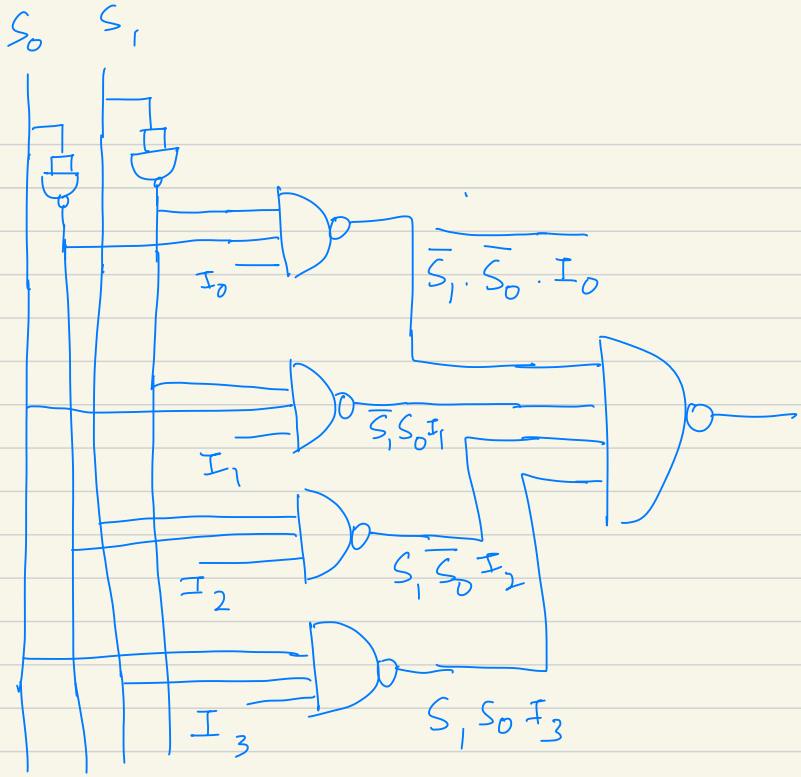
A3) 4:1 Mux using NAND

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

$$Y = \overline{S_1} \cdot \overline{S_0} I_0 + \overline{S_1} \cdot S_0 I_1 + S_1 \cdot \overline{S_0} I_2 + S_1 \cdot S_0 \cdot I_3$$

(A = $\overline{S_1} \cdot \overline{S_0} I_0$, B = $\overline{S_1} \cdot S_0 I_1$, C = $S_1 \cdot \overline{S_0} I_2$, D = $S_1 \cdot S_0 \cdot I_3$)

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



4 : 1 Mux using NAND gates only

4 : 1 Mux using NOR gates only

P. F. B

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

A B C

$$Y = \bar{S}_1 \bar{S}_0 I_0 + \bar{S}_1 S_0 I_1 + S_1 \bar{S}_0 I_2 + S_1 S_0 I_3$$

D

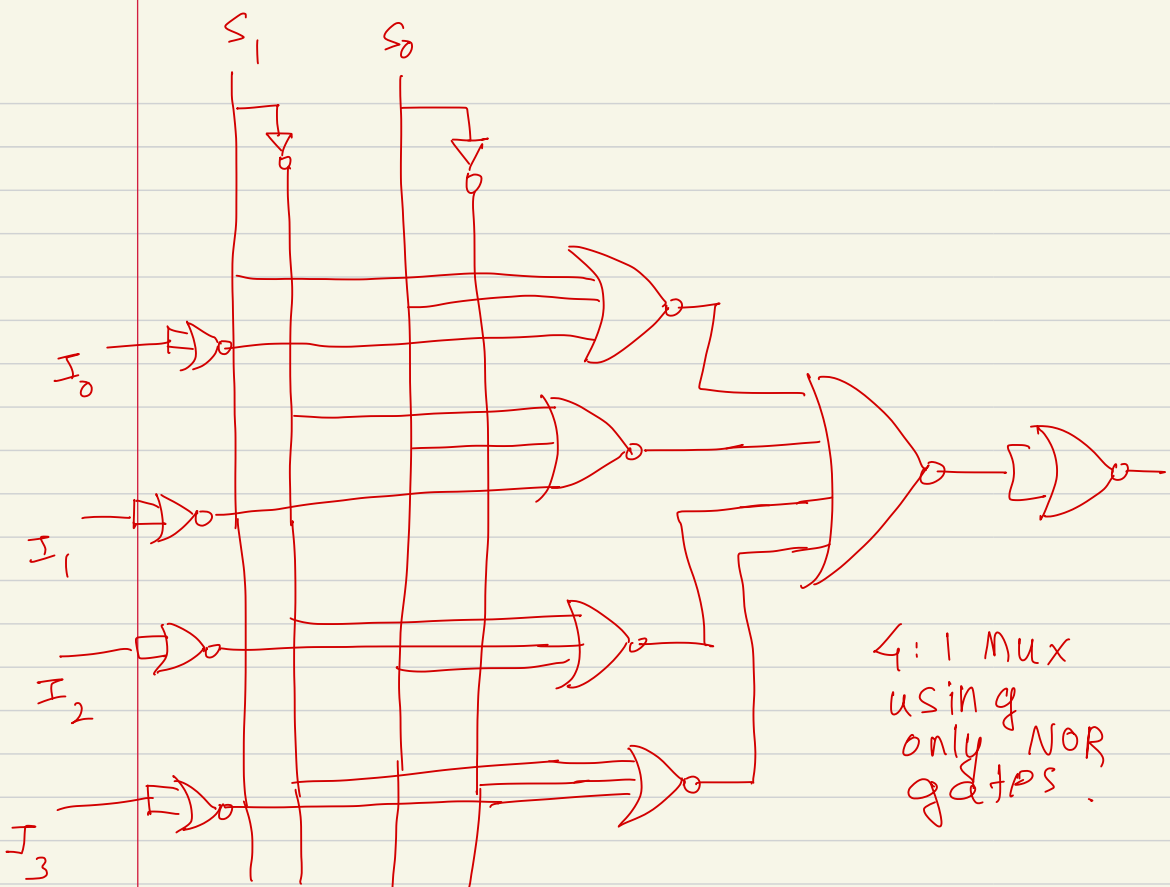
$$\overline{\bar{S}_1 \bar{S}_0 I_0} = \overline{\bar{S}_1 + \bar{S}_0 + \bar{I}_0} = S_1 + S_0 + \bar{I}_0$$

$$= \overline{\overline{S_1 \bar{S}_0 I_0} + \overline{\bar{S}_1 S_0 I_1} + \overline{S_1 \bar{S}_0 I_2} + \overline{S_1 S_0 I_3}}$$

$$\left(\text{or } \overline{A \cdot B \cdot C \cdot D} = \bar{A} + \bar{B} + \bar{C} + \bar{D} \right)$$

$$= \overline{\overline{S_1 + \bar{S}_0 + \bar{I}_0} + \left(\overline{\bar{S}_1 + \bar{S}_0 + \bar{I}_1}\right) + \left(\overline{\bar{S}_1 + S_0 + \bar{I}_2}\right) + \left(\overline{\bar{S}_1 + \bar{S}_0 + \bar{I}_3}\right)}$$

$$= \left(\overline{S_1 + S_0 + \bar{I}_0}\right) + \left(\overline{S_1 + \bar{S}_0 + \bar{I}_1}\right) + \left(\overline{\bar{S}_1 + S_0 + \bar{I}_2}\right) + \left(\overline{\bar{S}_1 + \bar{S}_0 + \bar{I}_3}\right)$$



4:1 Mux using only NOR gates.

A4) (a) Applications of NAND

i) Detects if a signal has gone low. This can find applicatⁿ in house alarms (Home Security)

ii) Encoder & decoders: The hidden element in barcode is NAND gates.

iii) Flash memories

Truth table

A	B	Output
0	0	1
0	1	1
1	0	1
1	1	0

Applications of NOR gate

(i) Home automation & security.

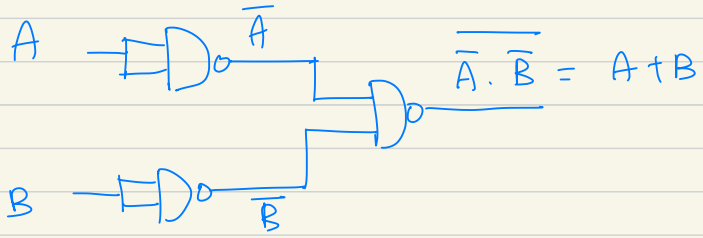
(ii) Data storage.

(iii) Traffic light systems. * How? & is NAND used or NOR.

5) Design of : OR, AND, XOR, NOT using (i) NAND, (ii) NOR.

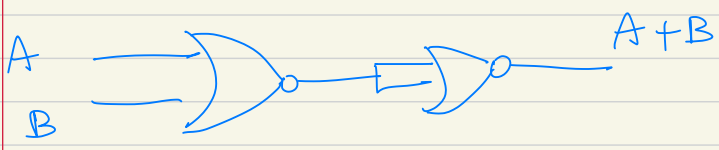
$A + B \qquad \overline{\overline{A \cdot B}} = \overline{A \cdot B}$

(i) $\overline{\overline{A} \cdot \overline{B}} = A + B$ {OR gate} using NAND

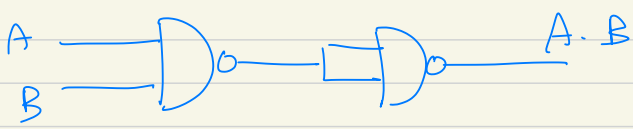


OR gate using NOR

$\overline{\overline{A+B}} = \overline{\overline{A}.\overline{B}} = A+B$

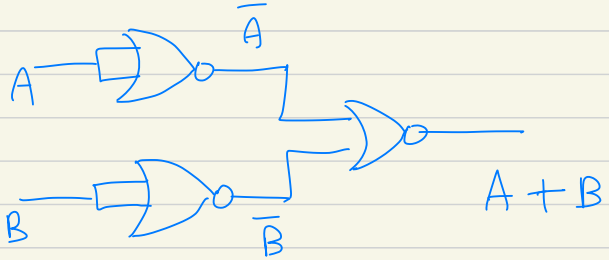


ii) AND gate (using NAND)



AND gate (using NOR)

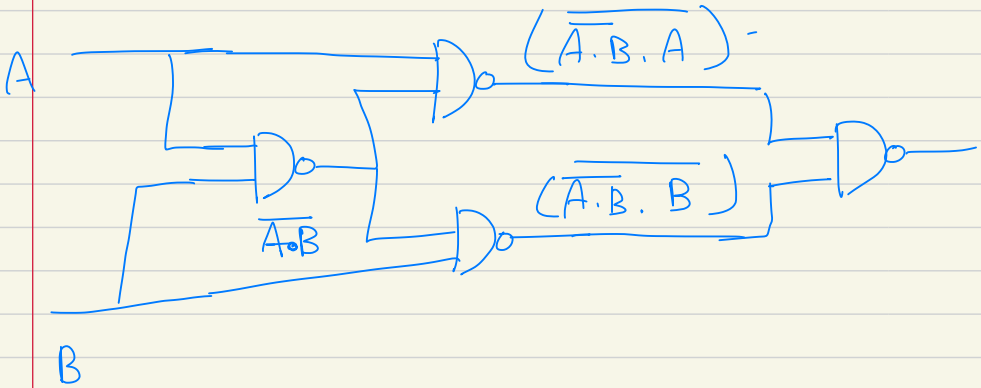
$\overline{\overline{A}+\overline{B}} = \overline{\overline{A}}.\overline{\overline{B}} = A.B$

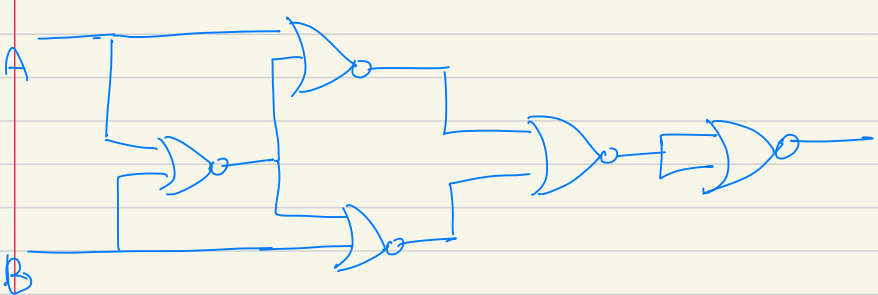


$\overline{\overline{C} \cdot \overline{D}}$

iii) XOR $A \oplus B = A \cdot \overline{B} + \overline{A} \cdot B$

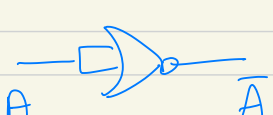
XOR using NAND





1 XOR using NOR gates only.

iv) NOT

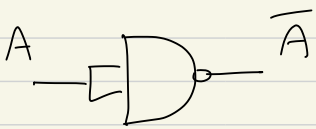


6) (a) Design of an Inverter:-

using NAND gate

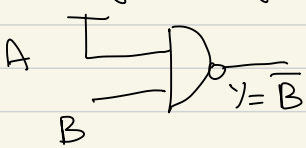
A	B	Out
0	0	1
0	1	1
1	0	1
1	1	0

A	Out
0	1
1	0

(1) 

If both the inputs are same; then o/p is complement of i/p.

(2) another soln is making A constant & getting inverted o/p



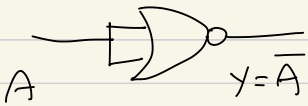
{ you get inverter function ality when A = 1 }.

using NOR gate.

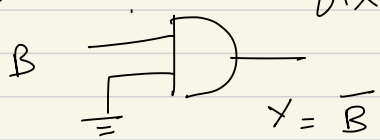
A	B	Out
0	0	1
0	1	0
1	0	0
1	1	0

A	out
0	1
1	0

(1) if both inputs same then we get inverted o/p

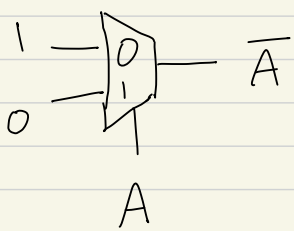


(2)



fixing A to gnd & giving B i/p gives the inverted O/P.

Using MUX (2:1)



(D) Full adder

Truth table.

A	B	C	Sum	Carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Sum

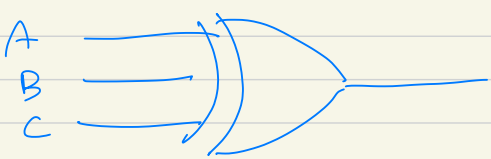
$$S = \bar{A}.\bar{B}.C + \bar{A}.B.\bar{C} + A.\bar{B}.\bar{C} + A.B.C$$

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

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

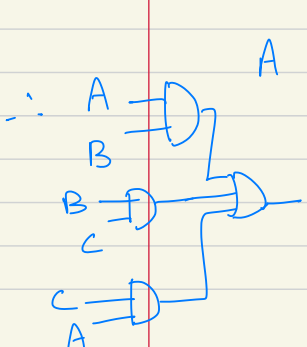
$$= \bar{A}.X + A\bar{X}$$

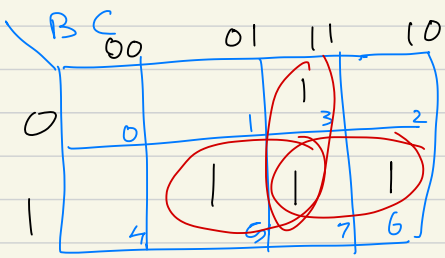
$$= A \oplus X \Rightarrow \boxed{S = A \oplus B \oplus C}$$



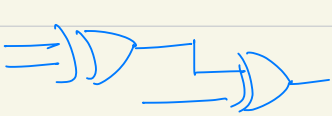
for carry

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





$$C = B.C + A.C + A.B$$



or

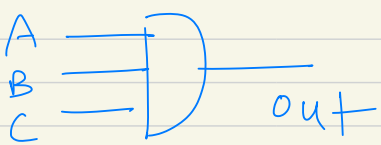
$$C = AB + C_{in}(A \oplus B)$$

7) What happens if more than 1 input is applied to the logic gate?

If more than 1 input is applied to the logic gate then the number of input combinations increases & o/p would change w.r.t the input.

AND gate 3 inputs

A	B	C	out
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1



If we compare the 2 no. of i/ps comb^{n} for 3 inputs

AND gate 2 inputs

A	B	out
0	0	0
0	1	0
1	0	0
1	1	1