



EC/EE/CS/IN

Digital Electronics

Maha
Revision



Chandan Jha Sir (CJ Sir)

Today's Targets



- 1 Logic GATE & Minimization
- 2 Combinational circuit
- 3 Sequential circuit
- 4 ADC/DAC

Inverter & and Gate

$$A \rightarrow \bar{A}$$

LOGIC GATE

1. NOT GATE

When Even no. of NOT GATES in Loop



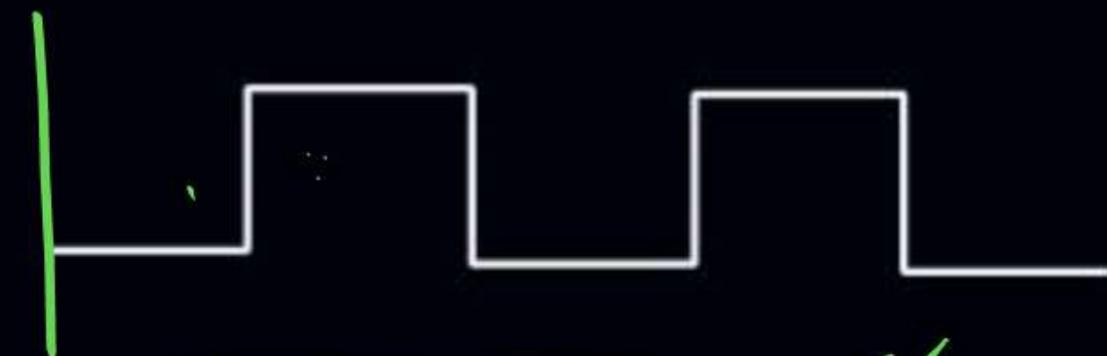
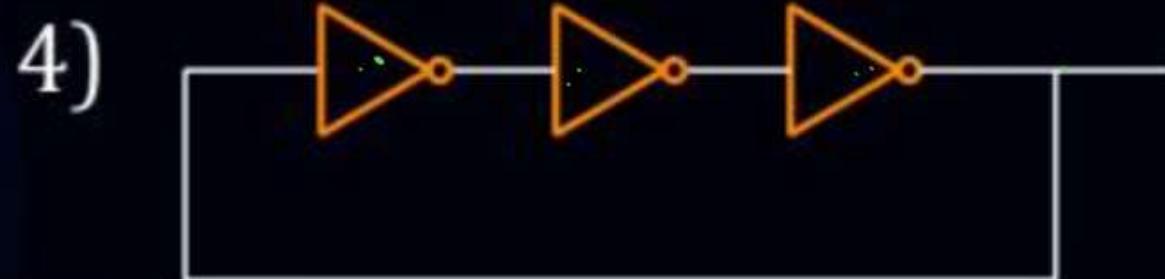
- 1> Basic memory element
- 2> Bistable multivibrator

Inverter & and Gate

LOGIC GATE

1. NOT GATE

Odd number of NOT GATE in loop :-

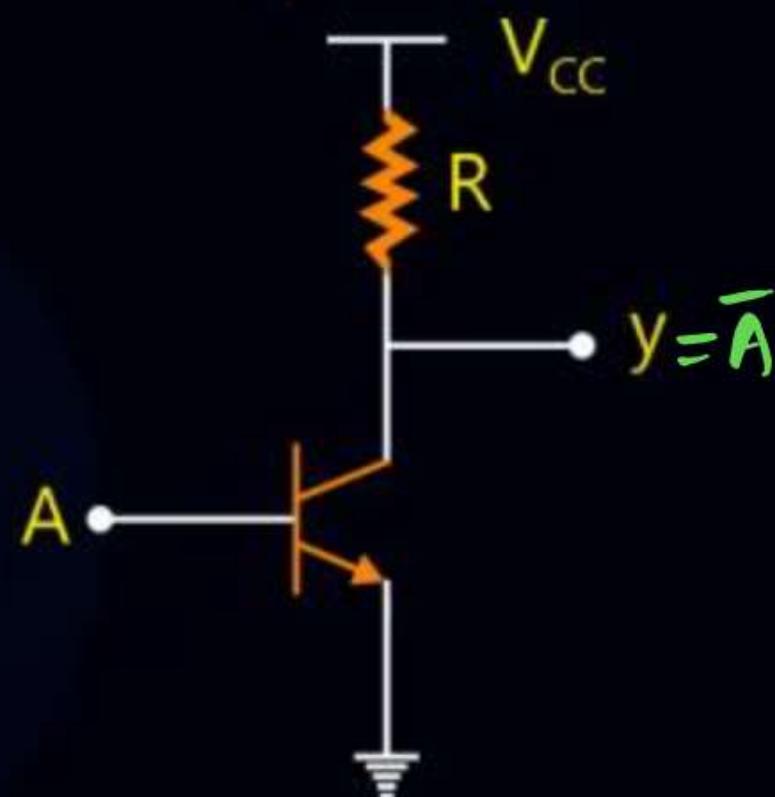


- 1> Astable multivibrator
- 2> Square wave generator
- 3> Free Running circuit
- 4> Ring oscillator

$$f = \frac{1}{2IN \times T_{PD}}$$

Inverter & and Gate

NPN BJT



A(Input)	y(Output)
0	1
1	0

Table 1: Truth table of AND gate

Figure 1: Transistor circuit for NOT gate

Inverter & and Gate

NOTE

1) Whenever logic are designed by TTL (Transistor transistor logic) then floating terminal always works as a high.

2) Whenever logic are designed by ECL (Emitter coupled logic) then floating terminal always works as a low.

3) Noise margin

4) Fan out

The number of logic driven by the logic are called fan out

5) Fan in

Number of input of a logic are called fan in.

6) ECL is the fastest logic among all the logic family

→ fastest Logic

Inverter & and Gate

AND GATE

6. Circuit Diagram

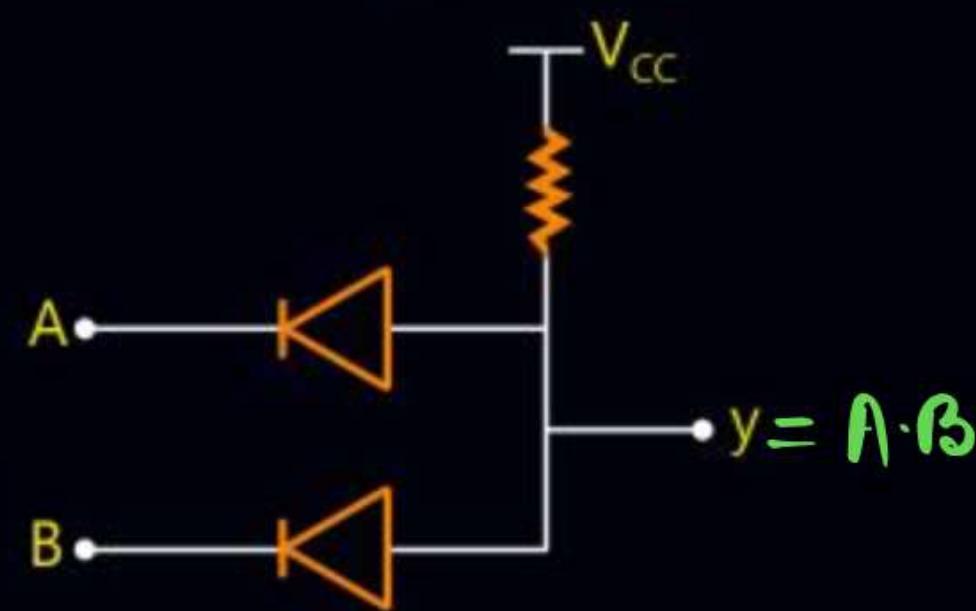
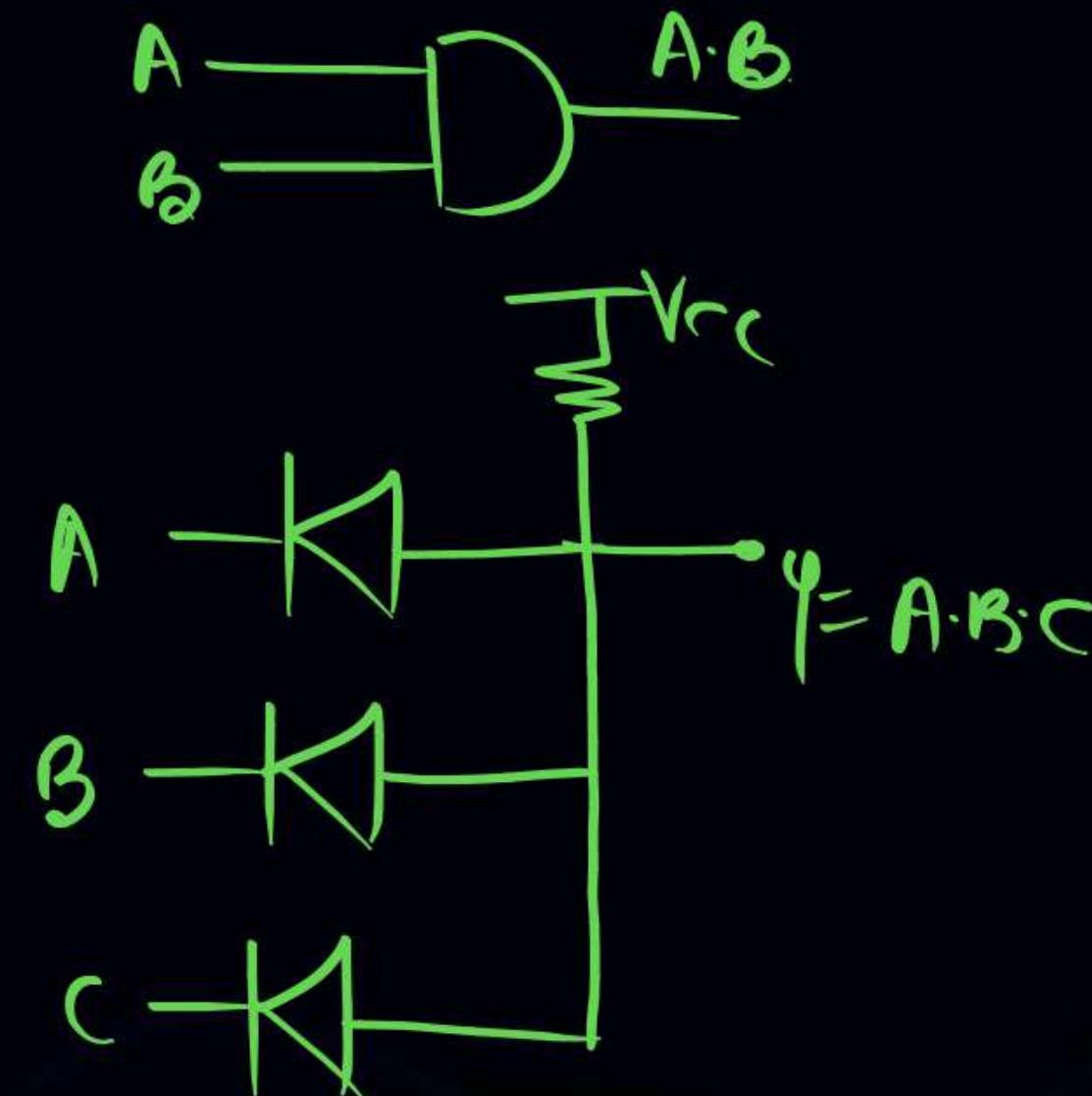


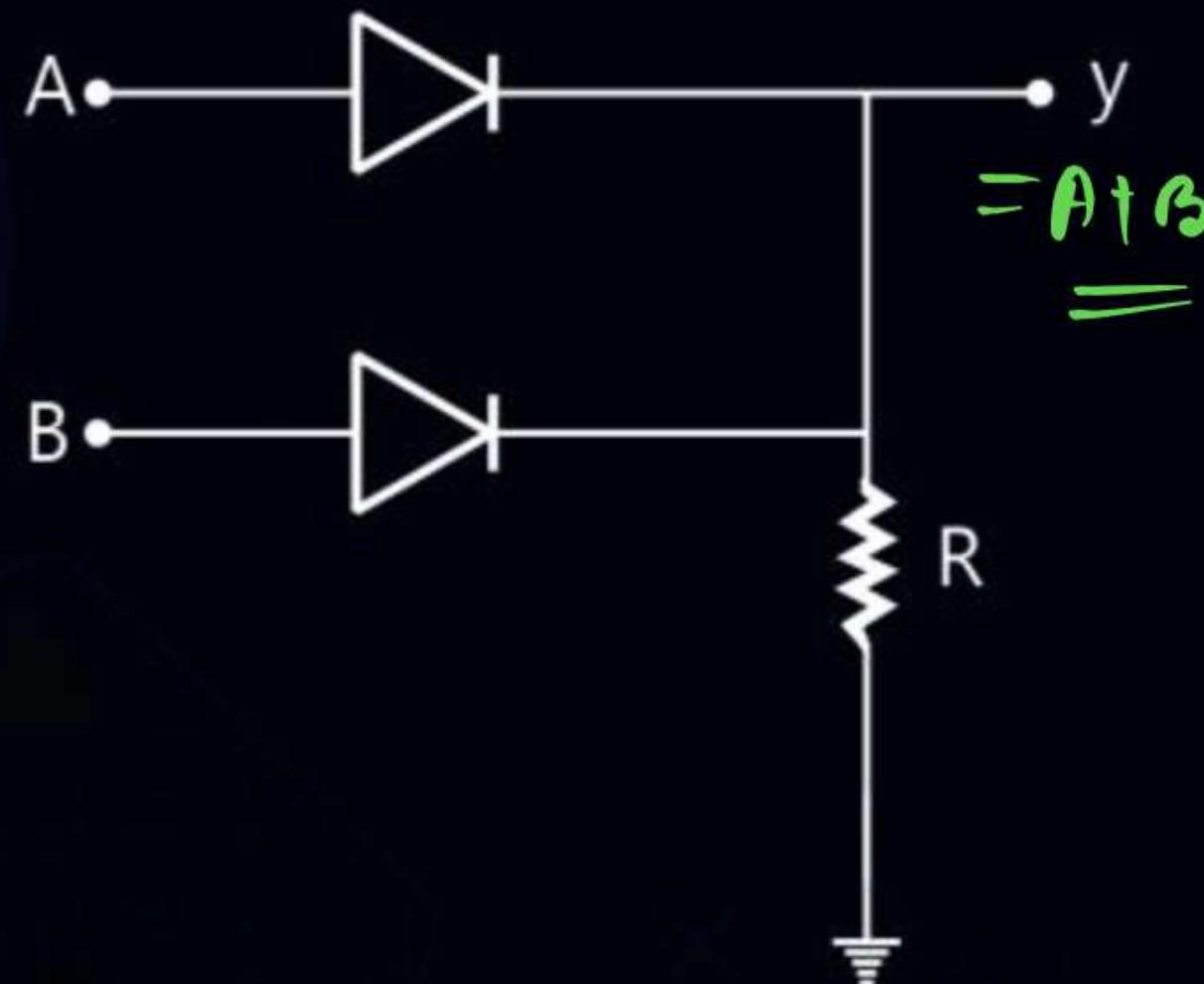
Figure 2: Transistor circuit for AND gate



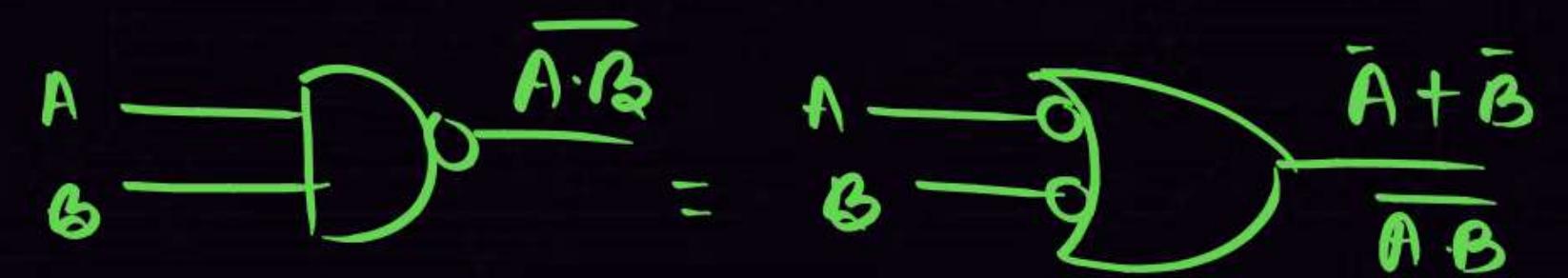
OR GATE, NAND, NOR GATE

OR GATE

7) Circuit Diagram



A	B	D_A	D_B	y
0	0			
0	1			
1	0			
1	1			

NAND

commutative Law ✓

Associative Law X

NOR



$$\text{Case (i)} \quad Y = \bar{A} \cdot \bar{B} \cdot C \cdot D \cdot \bar{E}$$

$n \rightarrow$ number of variables.

NAND

$K \rightarrow$ no. of complement

$$\underline{Q} \quad f = \bar{A} \cdot B \cdot C \quad n = 3$$

$$k = 1$$

$$\text{NAND} = (2 \times 3 - 2) + 1$$

$$= \underline{\underline{5}}$$

NOR

$$(3n-3) - k$$

$$n = 3, \quad k = 1$$

$$\text{NOR} = (3 \times 3 - 3) - 1$$

$$= \underline{\underline{5}}$$

case(2) $f = \bar{A} + B + \bar{C} + D + \bar{E} + \dots$

$n \rightarrow$ no. of Variables.

$k \rightarrow$ no. of complement Variables.

NAND

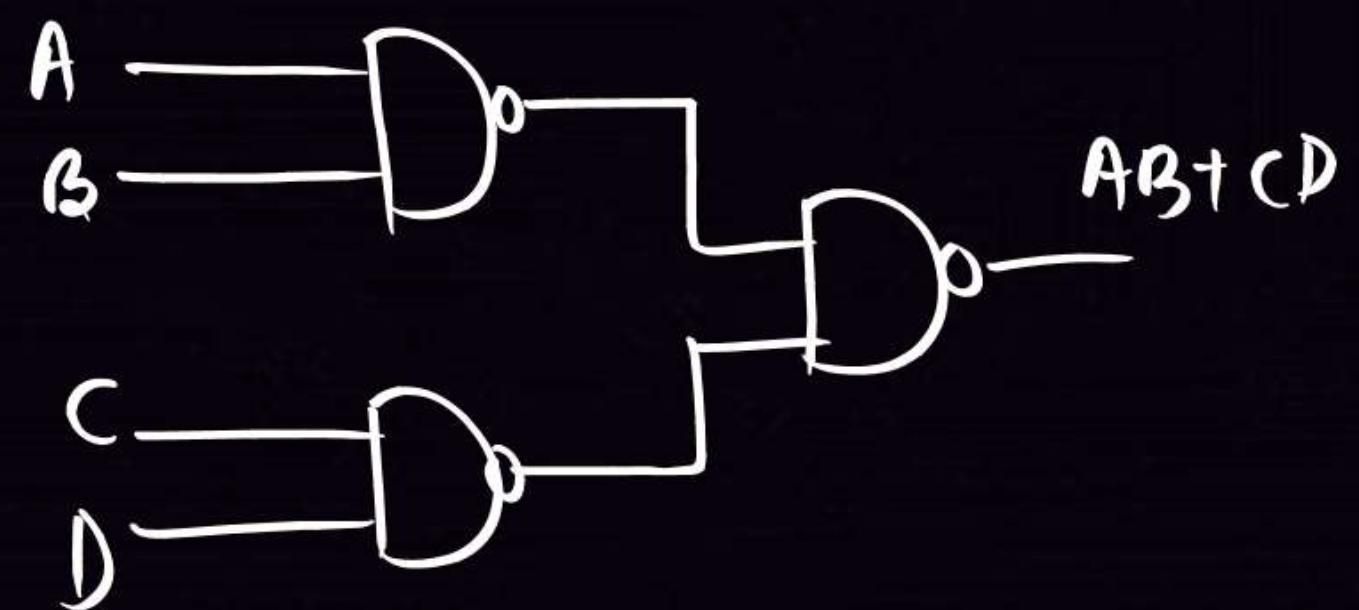
$$(3n-3)-k$$

NOR

$$(2n-2)+k$$

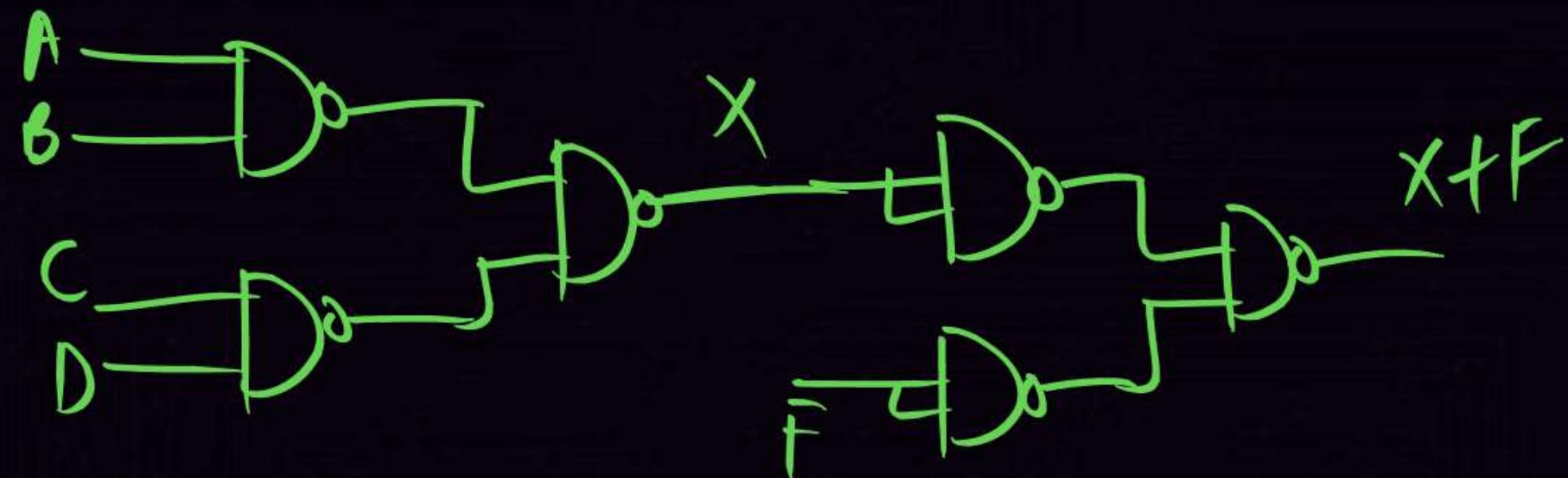
case(3) $f = AB + CD$

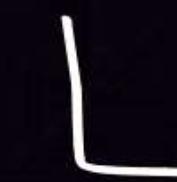
Number of NAND = 3



$$\overbrace{AB + CD + F}^3 \quad = \quad \overbrace{X + F}^{OR = 3}$$

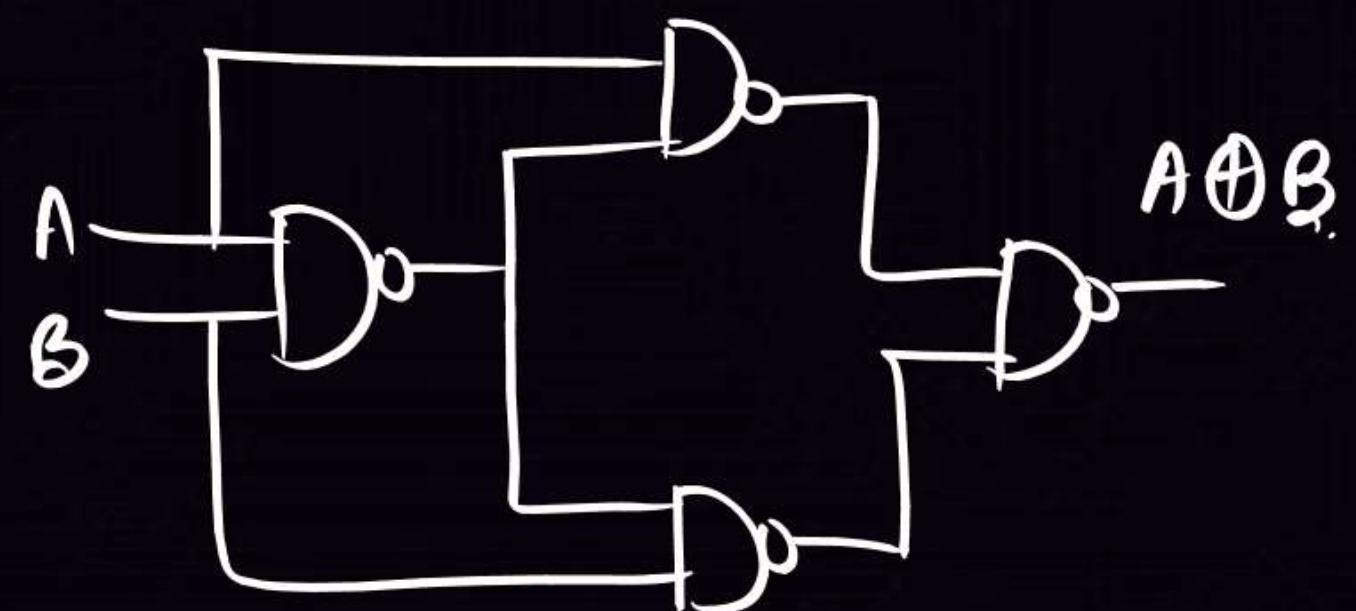
$$3+3=6$$

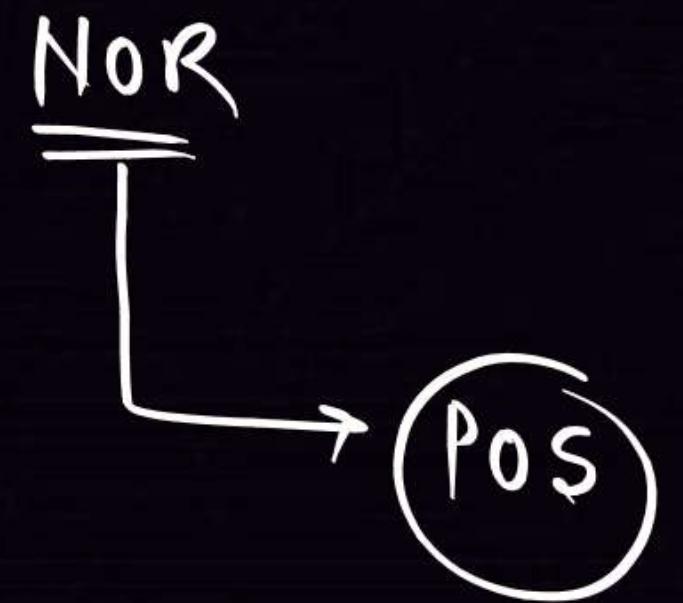


case(4)X-OR

NAND = ④

$$\begin{aligned} A \oplus B &= \bar{A}B + A\bar{B} \\ &= (A+B)(\bar{A}+\bar{B}) \end{aligned}$$

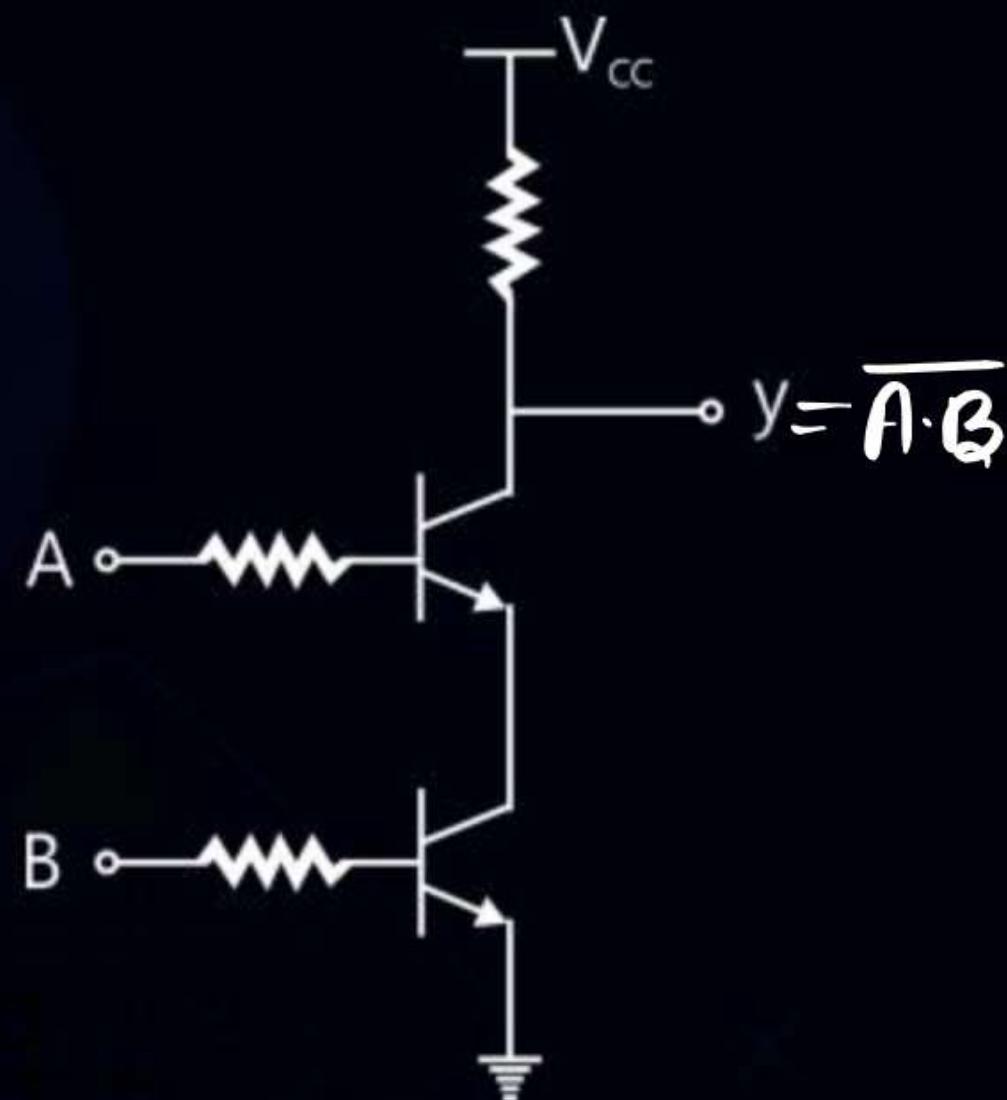




OR GATE, NAND, NOR GATE

NAND GATE

5) Circuit Diagram

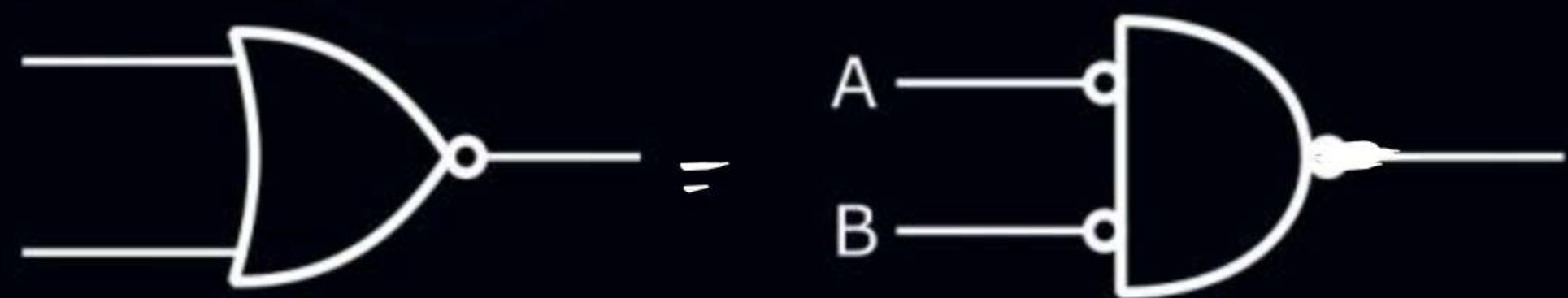


A	B	T _A	T _B	y
0	0			
0	1			
1	0			
1	1			

OR GATE, NAND, NOR GATE

NOR GATE

1) Symbol



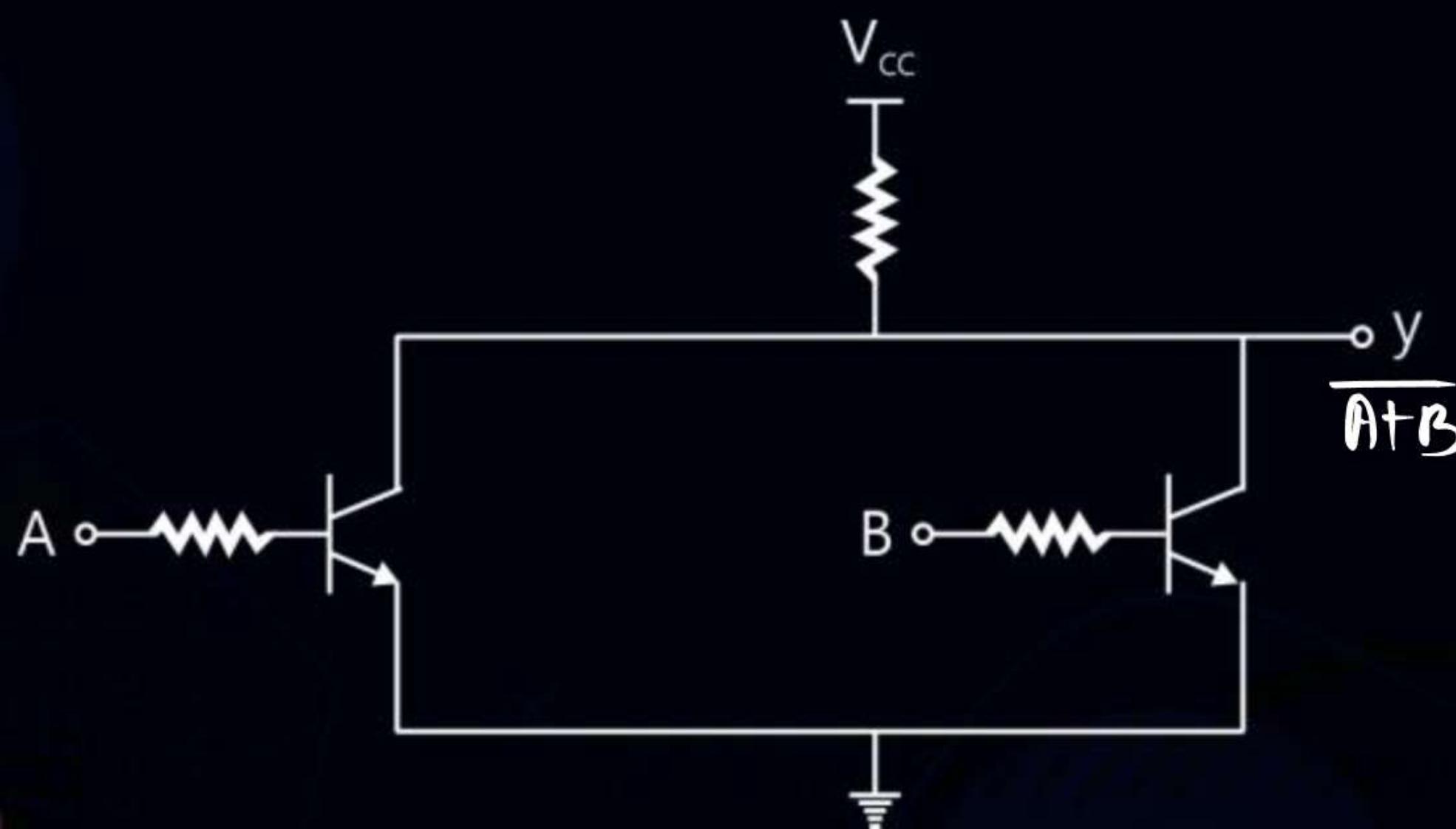
2) Truth Table

A	B	Y
0	0	
0	1	
1	0	
1	0	

OR GATE, NAND, NOR GATE

NOR GATE

5) Circuit Diagram



A	B	T_A	T_B	y
0	0			
0	1			
1	0			
1	1			

OR GATE, NAND, NOR GATE

NAND AS UNIVERSAL LOGIC

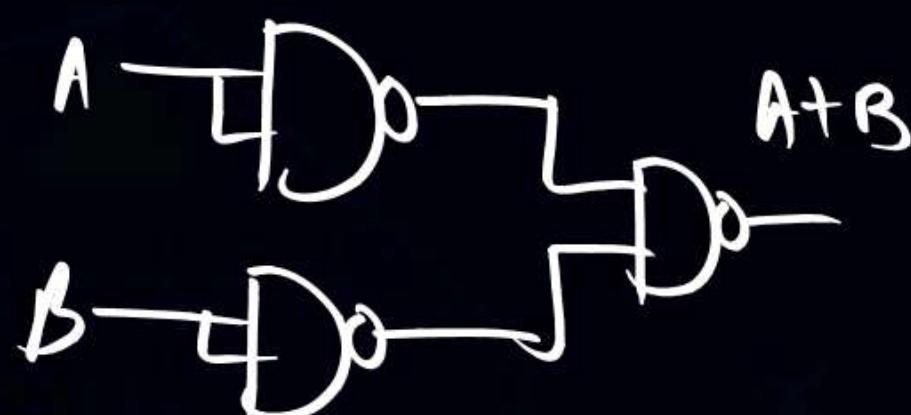
1) NOT GATE



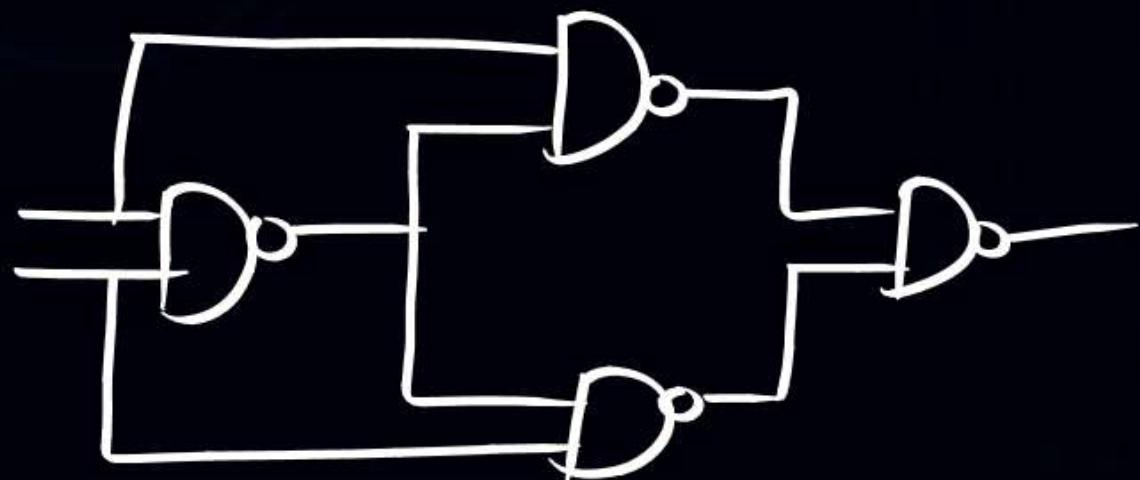
2) AND GATE



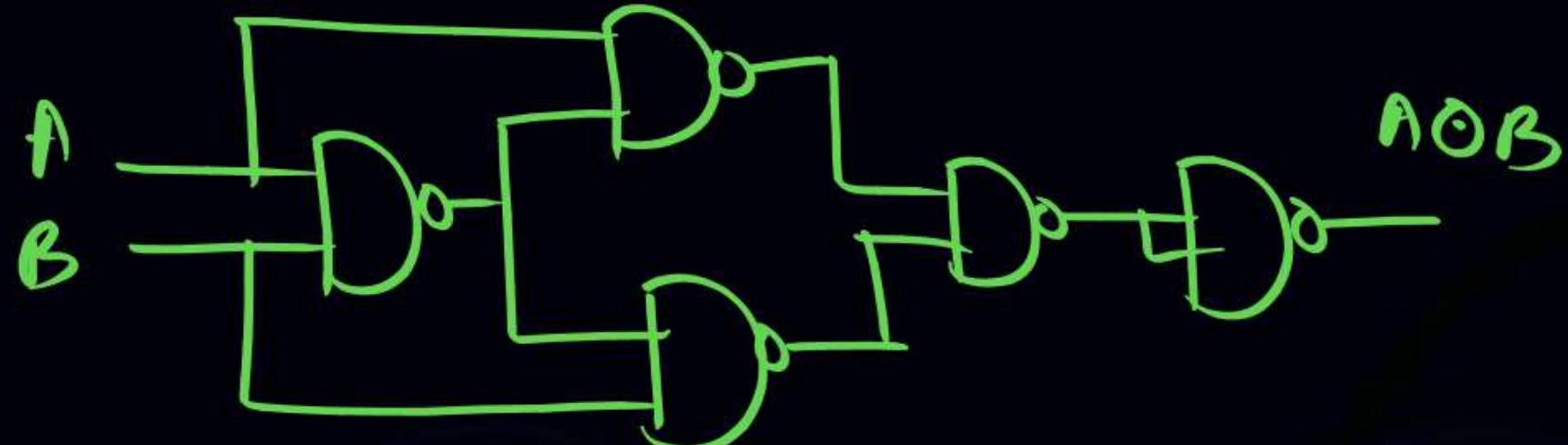
3) OR GATE



4) XOR GATE



5) XNOR GATE



OR GATE, NAND, NOR GATE

Alternate Symbol



OR GATE, NAND, NOR GATE

Discussion

XOR GATE, X-NOR GATE

XOR GATE

1) Symbol



2) Truth Table

A	B	y = A \oplus B
0	0	0
0	1	1
1	0	1
1	1	0

XOR GATE, X-NOR GATE

XOR GATE

$$3) A \oplus A = 0$$

$$A \oplus 0 = A$$

$$A \oplus \bar{A} = 1$$

$$A \oplus 1 = \bar{A}$$

- 4) Odd parity detector
Even parity generator

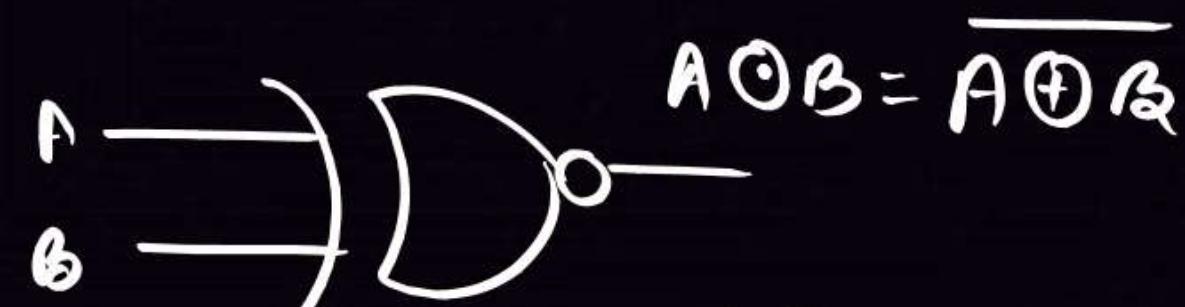
X-OR

commutative Law ✓

Associative Law ✓

$$A \oplus B \oplus C = \sum (1, 3, 4, 7)$$

$$A \oplus A \oplus A \dots \dots \oplus A = \begin{cases} A & \text{odd number of "A"} \\ 0 & \text{Even number of "A"} \end{cases}$$

X-NOR

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

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

A	B	y = A ⊙ B
0	0	1
0	1	0
1	0	0
1	1	1

X-NOR



commutative Law ✓

Associative Law ✓

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

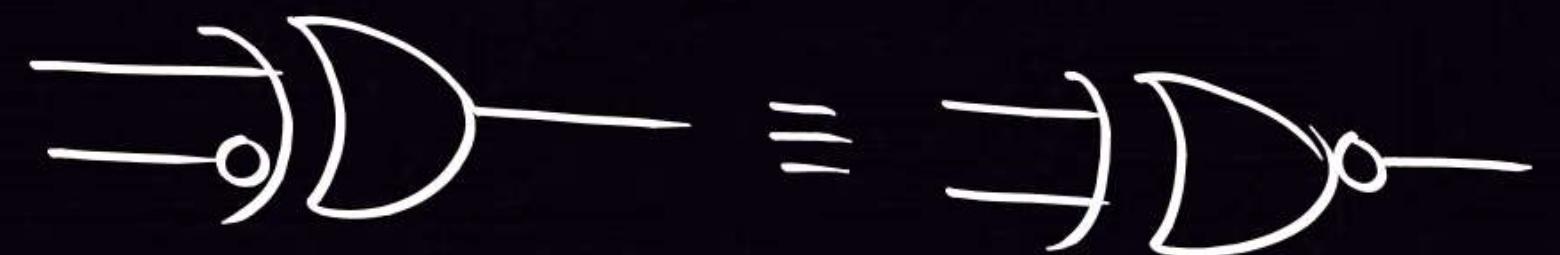
$$A \odot A = 1$$

$$A \odot 1 = A$$

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

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

$$(A \odot B) \odot C = A \oplus B \oplus C = \text{sum}(1, 2, 4, 8)$$



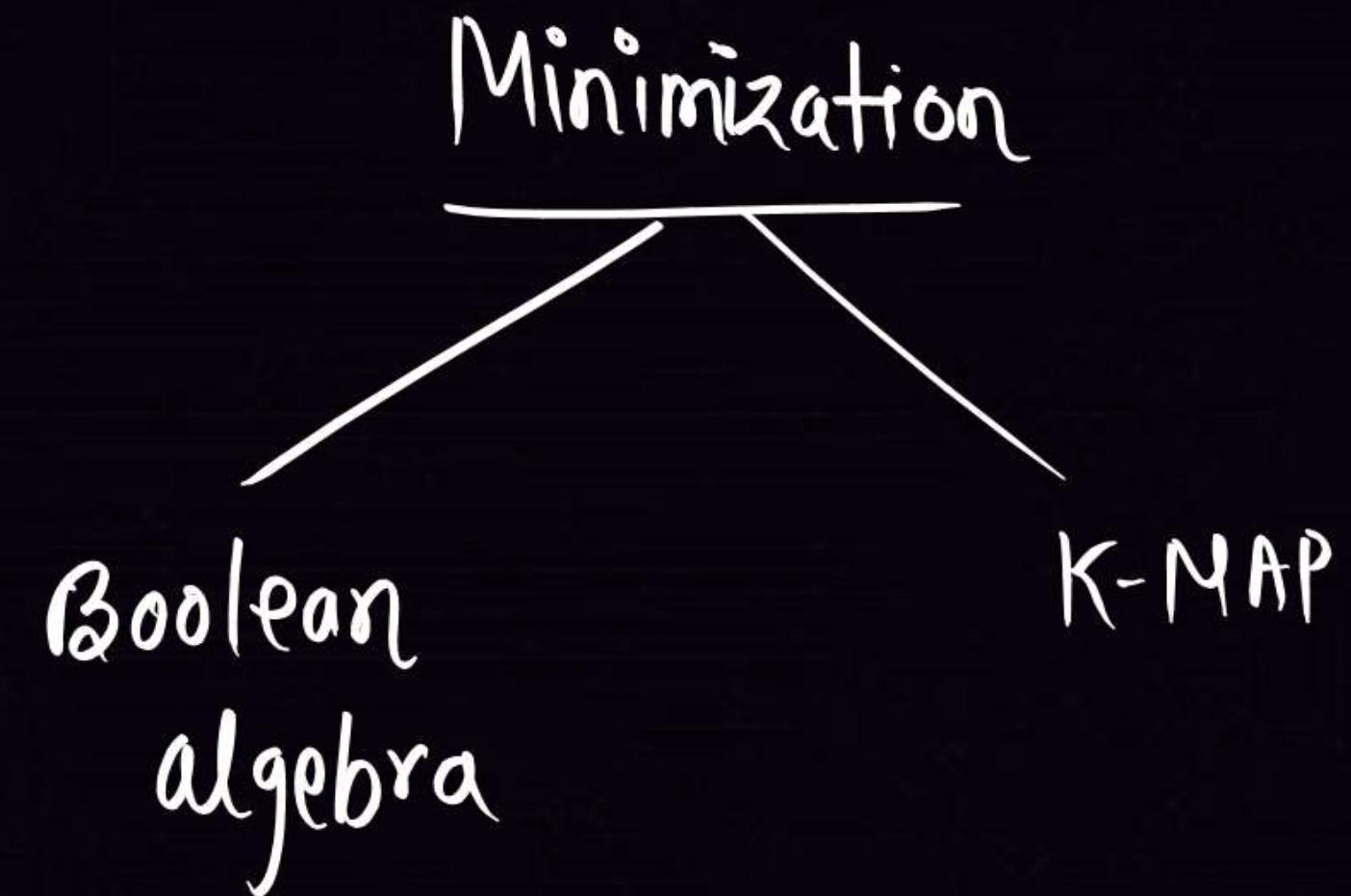
XOR GATE, X-NOR GATE }

7) X-OR follows the commutative as well as associative Law

$$A \oplus B = B \oplus A \Rightarrow \text{commutative Law}$$

$$A \oplus B \oplus C = (A \oplus B) \oplus C \quad \text{Associative Law}$$

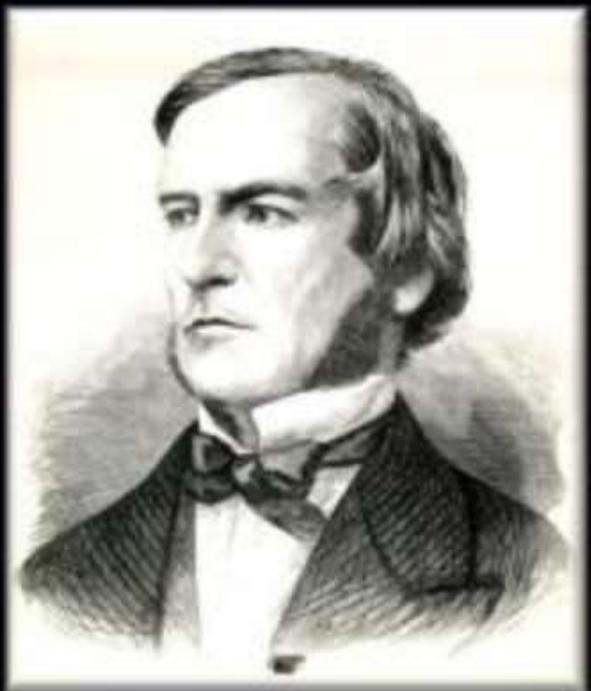
A B C	$A \oplus B \oplus C$	$(A \oplus B) \oplus C$
0 0 0	0	
0 0 1	1	
0 1 0	1	
0 1 1	0	
1 0 0	1	
1 0 1	0	
1 1 0	0	
1 1 1	1	



Laws of Boolean Algebra

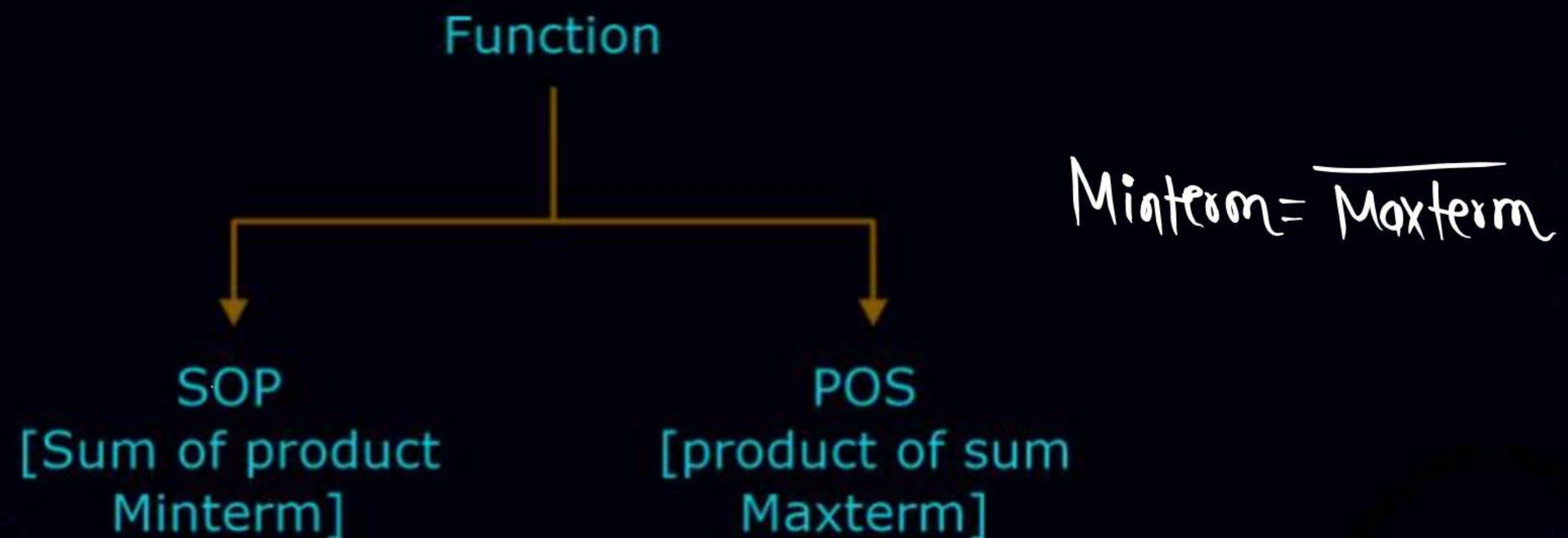
1854- George Boole

“An Investigation of Law of Thoughts”



Laws of Boolean Algebra

BOOLEAN ALGEBRA



① Distribution Law

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

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

② Consensus theorem

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

③ Transpose Theorem

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

④ D-Morgan's Law

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

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

Laws of Boolean Algebra

Theorem

1) Distribution theorem

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

2) Consensus theorem

$$AB + \overline{A}C + BC$$

3) Transpose theorem

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

4) D-Morgan's Law

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

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

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

$$AB + \bar{A}C + BC \perp$$

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

$$\underline{\underline{AB}} + \underline{\underline{\bar{A}C}} + \underline{\underline{\bar{A}B}} + \underline{\underline{AC}}$$

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

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

Laws of Boolean Algebra

Theorem

5) Annulment Law

$$A \cdot 0 = 0, \quad A + 1 = 1$$

6) Identity Law

$$A + 0 = A, \quad A \cdot 1 = A$$

7) Idempotent Law

$$A + A = A, \quad A \cdot A = A$$

8) Absorptive Law

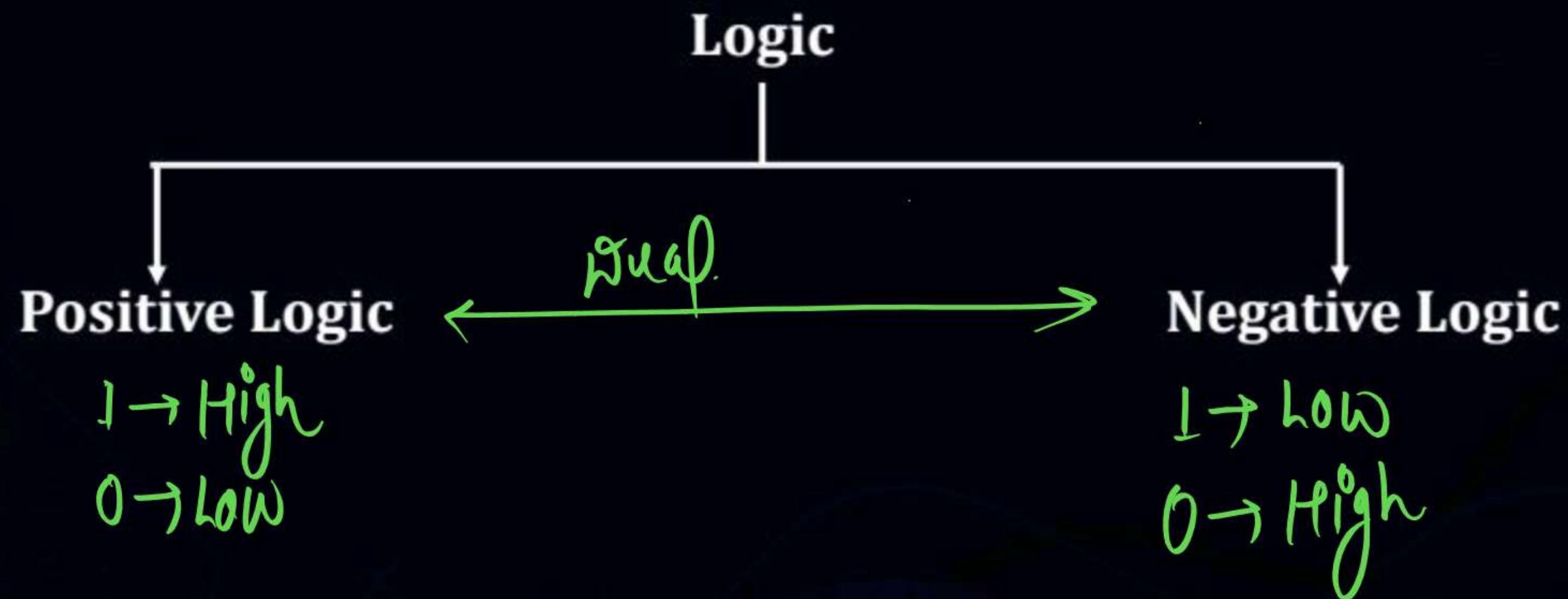
$$A + AB = A$$

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

Laws of Boolean Algebra

Dual

$$\left. \begin{array}{l} 1 \rightarrow -3.68V \\ 0 \rightarrow -9.2V \end{array} \right\} +ve \text{ Logic}$$



Laws of Boolean Algebra

AND

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

OR

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

"P"

Laws of Boolean Algebra

Dual

i) Positive logic $\xrightarrow{\text{Dual}}$ negative logic ✓

ii) NPN trans \rightarrow PNP trans ✓

iii) N-MOS \rightarrow P-MOS ✓

iv) 0 \leftrightarrow 1 ✓

v) $\cdot \leftrightarrow +$

AND \longleftrightarrow OR

$\cdot \longleftrightarrow +$

Laws of Boolean Algebra

Dual

vi) AND \leftrightarrow OR

vii) NAND \leftrightarrow NOR

Viii) EX - OR \leftrightarrow EX - NOR

ix) BUFFER \leftrightarrow BUFFER

X) INVERTER \leftrightarrow INVERTER



Dual

$$\bar{A} + \bar{B} + C \xrightleftharpoons{R} \bar{A} \cdot \bar{B} \cdot C$$

Complement

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

Laws of Boolean Algebra

Self Dual Functions ✓

$$\begin{array}{ccc} A & \xrightarrow{\text{D}} & A \\ \overline{A} & \xrightarrow{\text{D}} & \overline{A} \end{array} \quad \left. \right\}$$

$$f^D = f$$

→ Self Dual

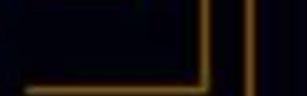
$$f = AB + BC + AC$$

$$f'$$

Self Rel.

Laws of Boolean Algebra

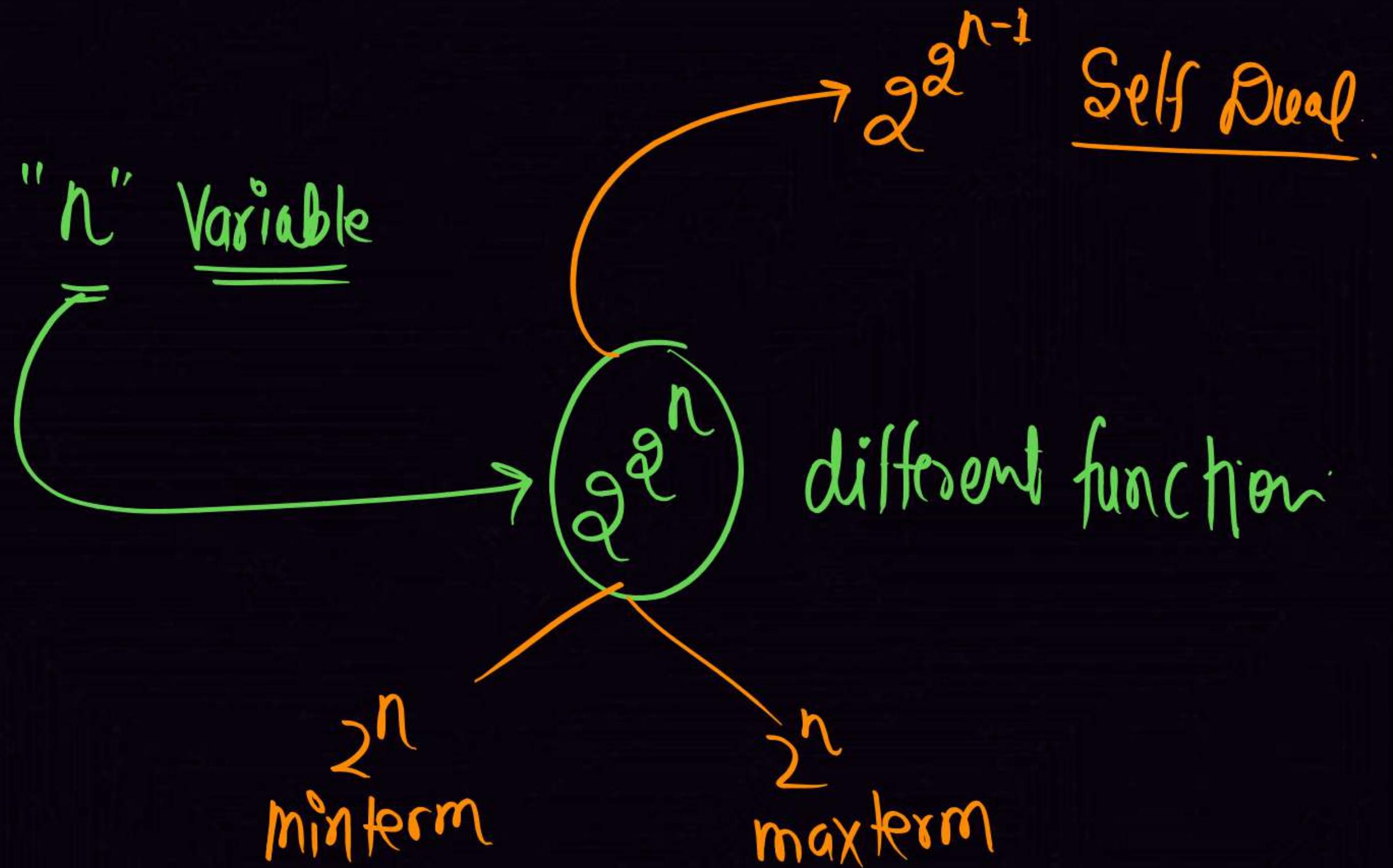
Mutual Exclusive Functions

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

(0, 7)
(1, 6)
(2, 5)
(3, 4)

Self Compliment

n-variable - [Maxterm = Minterm = 2^n]

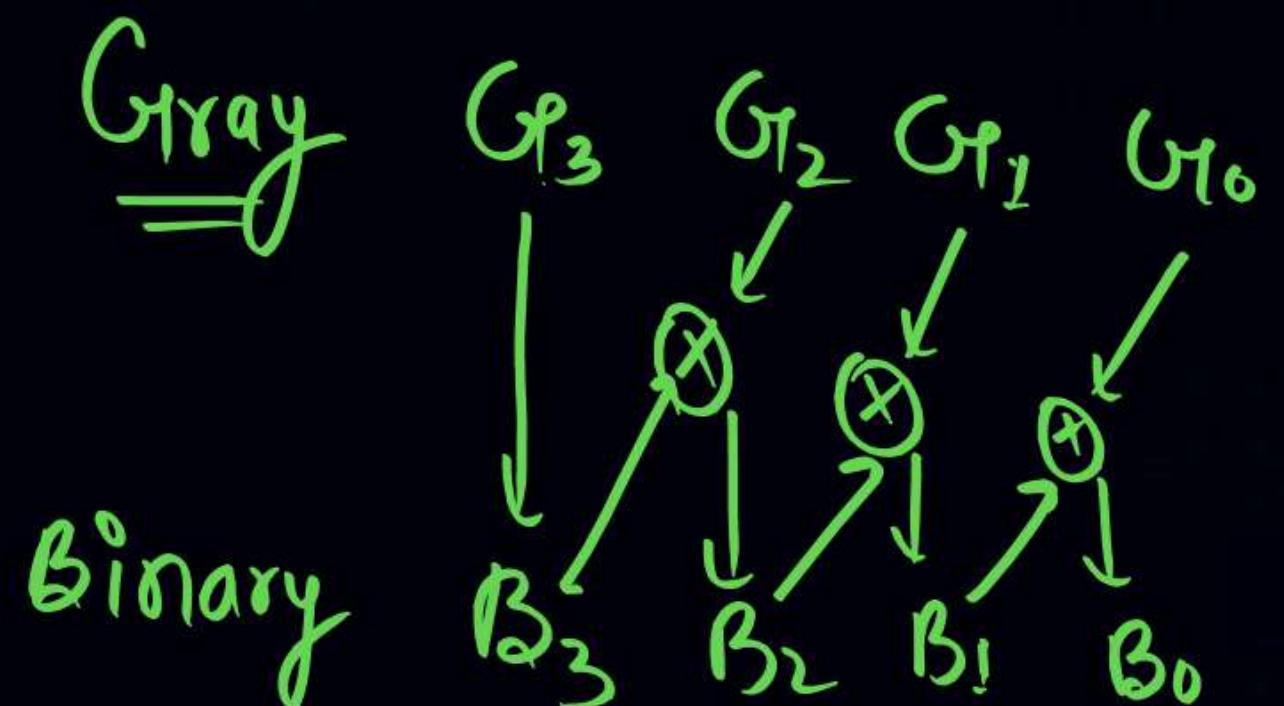
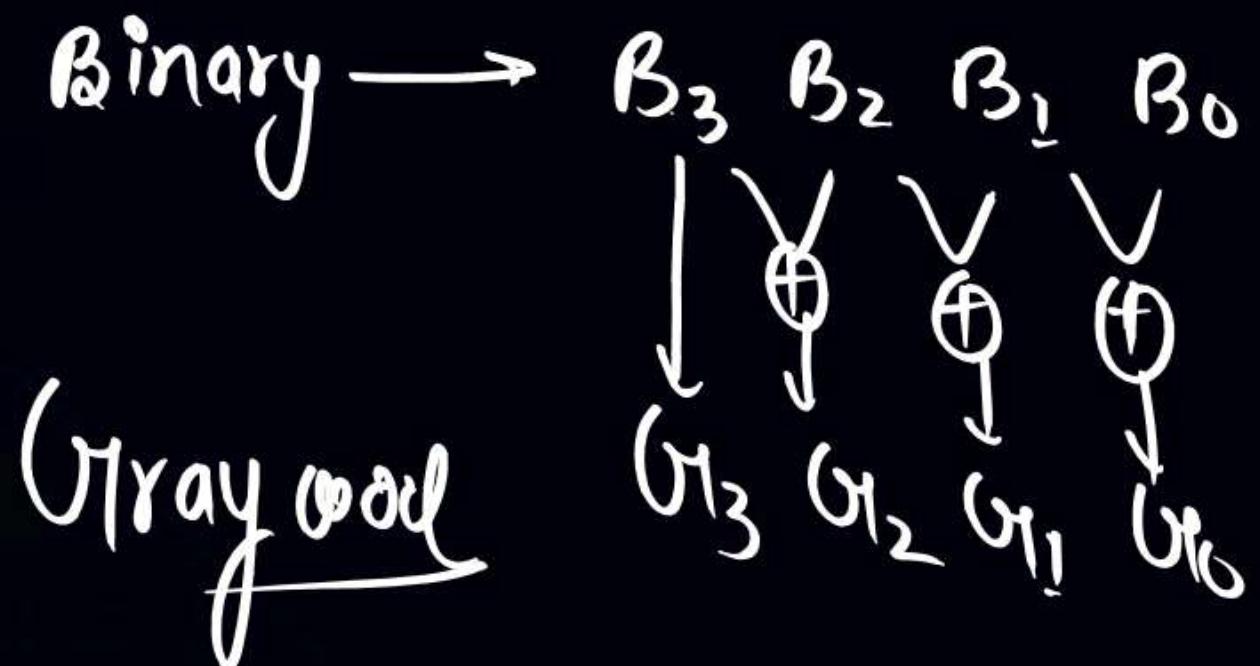


K Map - Basics

Minimization by K-Map

→ Based on gray code. ✓

→ Gray code ✓



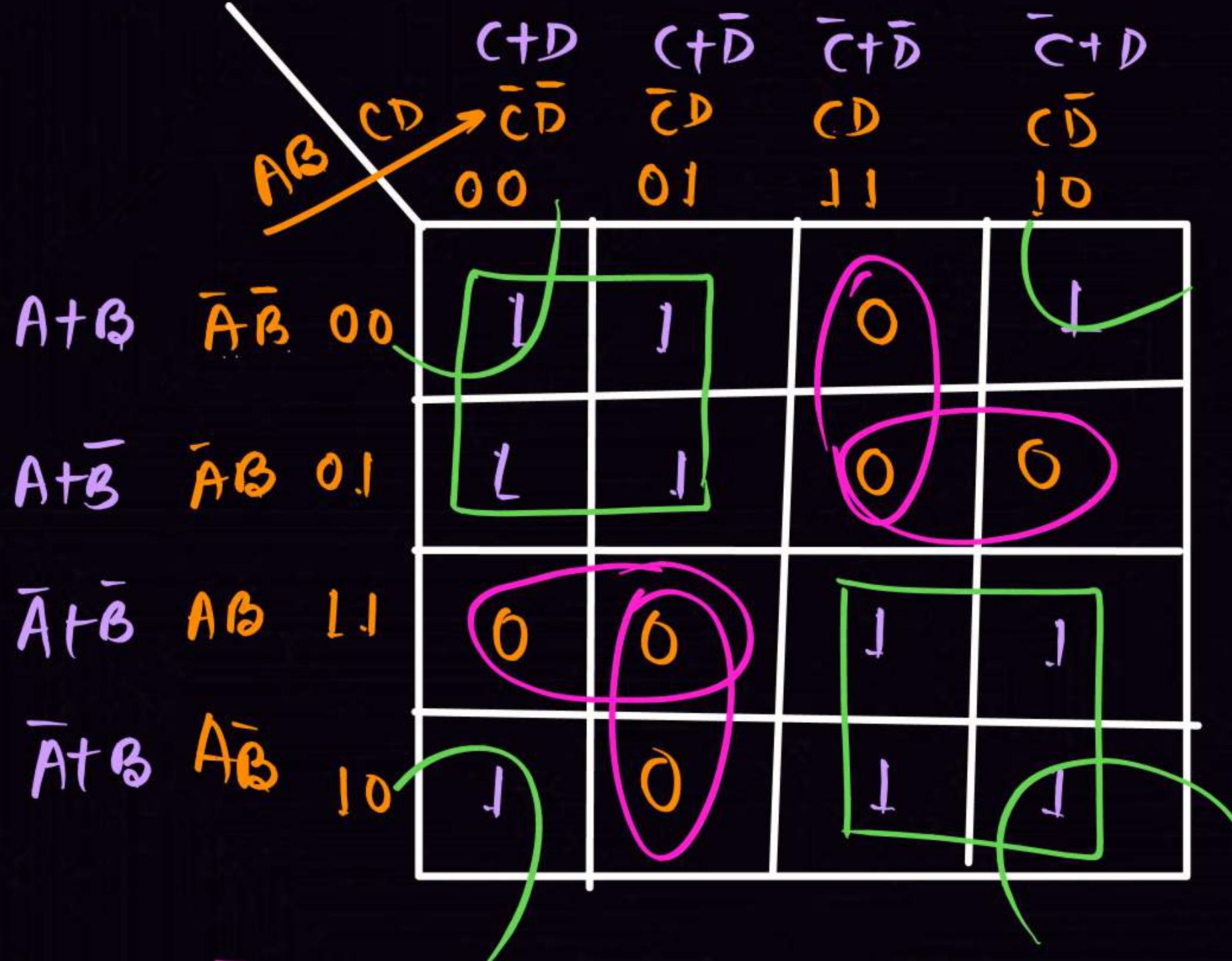
K-MAP

C^T = Baba Rule

Variable will mini.
Kam se kam group banana hai and Bade se bada group
terms will minimize banana hai

α^n → group → "n" Variables minimized

$16 = 2^4$ → 4 Variables minimized



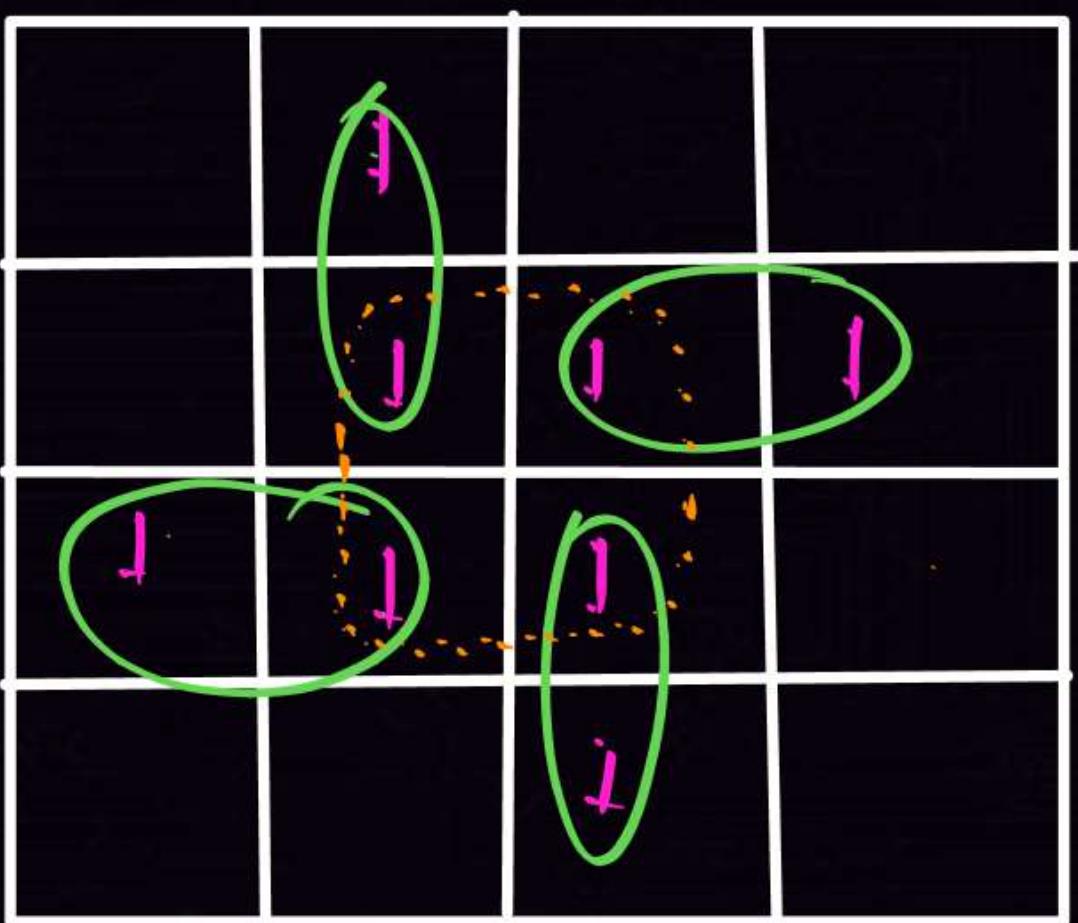
$$\bar{f} = \bar{A}CD + \bar{A}B^{\perp}C + AB^{\perp}\bar{C} + \bar{A}\bar{C}D$$

SOP

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

POS

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



$$I = 8$$

$$PI = 5$$

$$EPI = 4$$

$$RPI = 1$$

Combinational Circuit :-

comparator
→ "n" bit

$$\text{Total combination} = 2^{2n}$$

$$\text{Equal combination} = 2^n$$

$$\text{Unequal} = 2^{2n} - 2^n$$

$$\text{Greater-Less} = \frac{2^{2n} - 2^n}{2}$$

2 bit comparator



(A > B)

$$A_1 \bar{B}_1 + (A_1 \odot B_1) A_0 \bar{B}_0 \quad \checkmark$$

$$A_1 \bar{B}_1 + (A_1 + \bar{B}_1) A_0 \bar{B}_0 \quad \checkmark$$

(A < B)

$$\bar{A}_1 B_1 + (A_1 \odot B_1) \bar{A}_0 B_0 \quad \checkmark$$

$$\bar{A}_1 B_1 + (\bar{A}_1 + B_1) \bar{A}_0 B_0 \quad \checkmark$$

(A = B)

$$(A_1 \odot B_1) (A_0 \odot B_0) \quad \checkmark$$

HA

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

$$\text{Carry} = AB$$

$$\text{Minimum no. of NAND} = 5$$

$$\text{Minimum no. of NOR} = 5$$

FA \Rightarrow 2HA + 1OR UNIT.

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

$$\text{Carry} = \sum m(3, 5, 6, 7)$$

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

$$= AB + BC + AC$$

$$\text{NAND} = 9$$

$$\text{NOR} = 9$$

H.S

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

$$\text{Borrow} = \bar{A}B$$

$$\text{NAND} = 5$$

$$\text{NOR} = 5$$

F.S.

$$\text{Hiff} = A \oplus B \oplus C$$

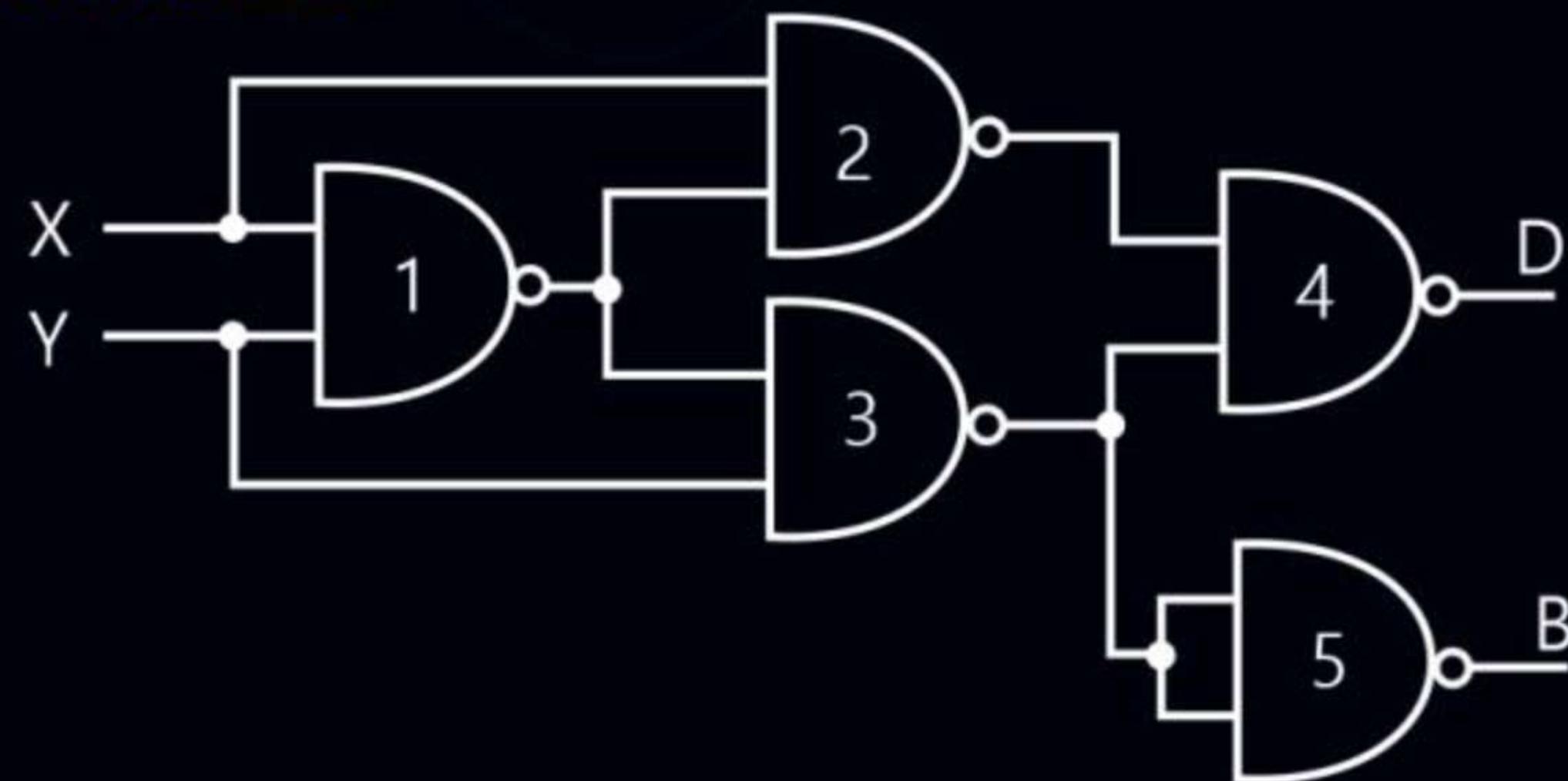
$$\text{Borrow} = \bar{A}B + \bar{A}C + BC$$

$$\begin{array}{l} \text{NAND} \\ \text{NOR} \end{array} \left. \right\} q$$

$$J \cdot F.S = 2H.S + 10R$$

HS, FS, Serial Adder

HALF SUBTRACTOR



Half-Subtractor using NAND gates

Figure 2: Implementation of Half Subtractor using NAND Gates

HS, FS, Serial Adder

HALF SUBTRACTOR

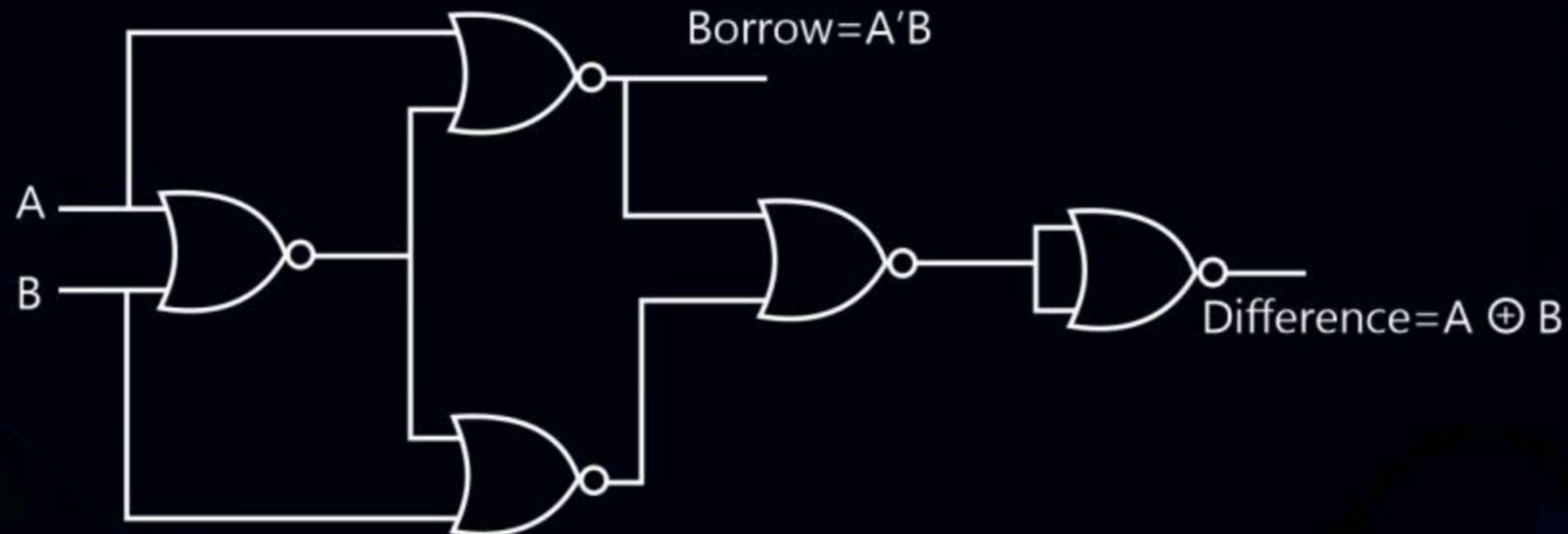


Figure 3: Implementation of Half Subtractor using NOR Gates

HS, FS, Serial Adder

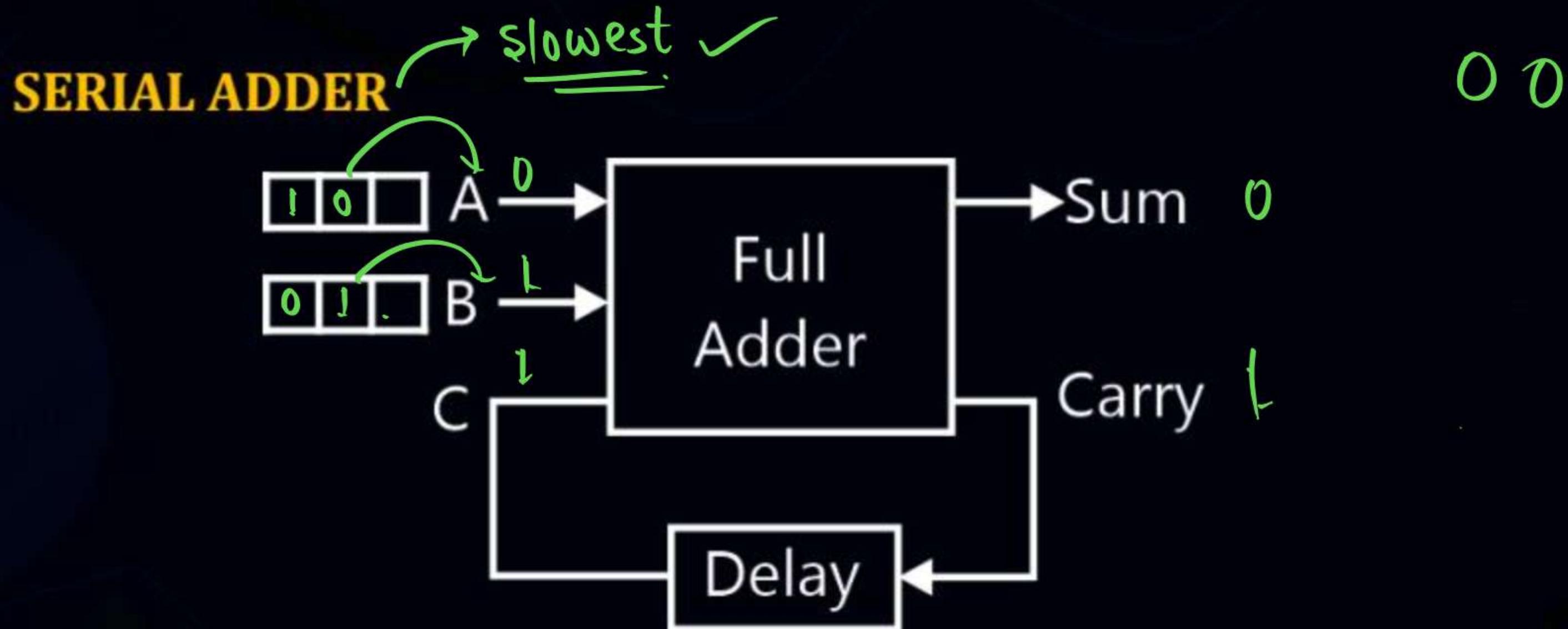


Figure 8: Serial Adder

HS, FS, Serial Adder

PARALLEL ADDER [Ripple carry adder]

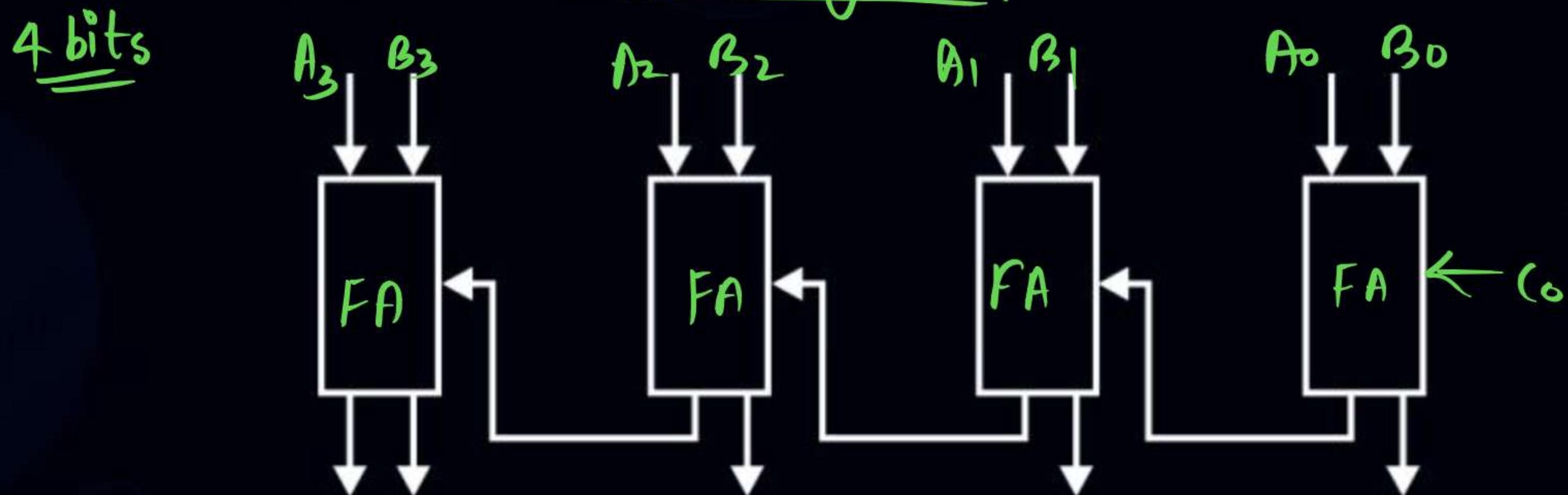


Figure 9: Parallel Adder

"n" bit.

① $(n-1)FA + 1HA$

② nFA

③ $(2n-1)HA + (n-1)OR$

Delay

T_{sum}, T_{carry}

$$T = (n-1)T_{carry} + \max\{T_{sum}, T_{carry}\}$$

Parallel Adder, LACA

PARALLEL ADDER

$$T = \underbrace{(n-1) \{ T_{AND} + T_{OR} \}} + \text{Max} \{ T_{sum}, T_{carry} \}$$

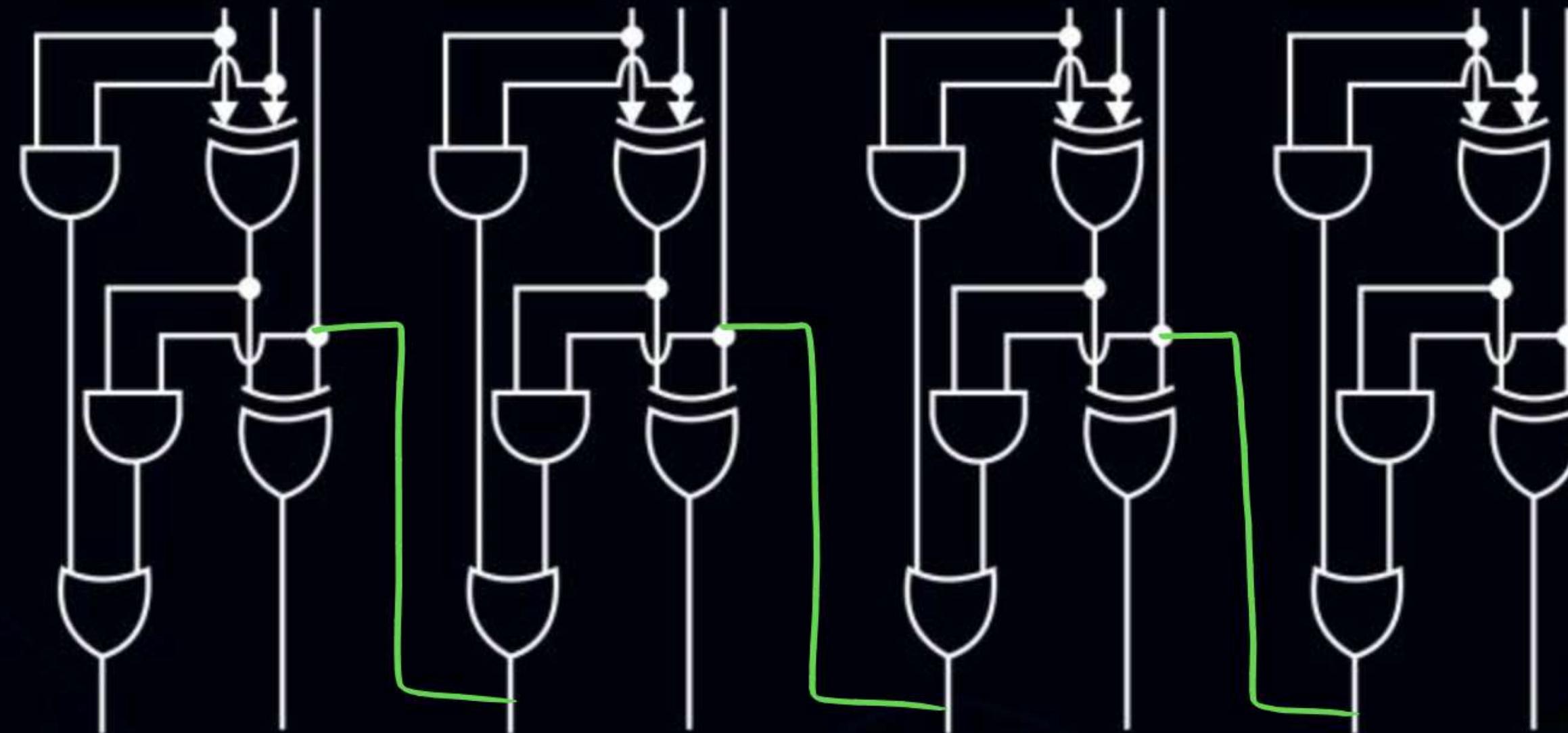


Figure 2: Designing of Parallel Adder using logic gates

LACA, Multiplier }

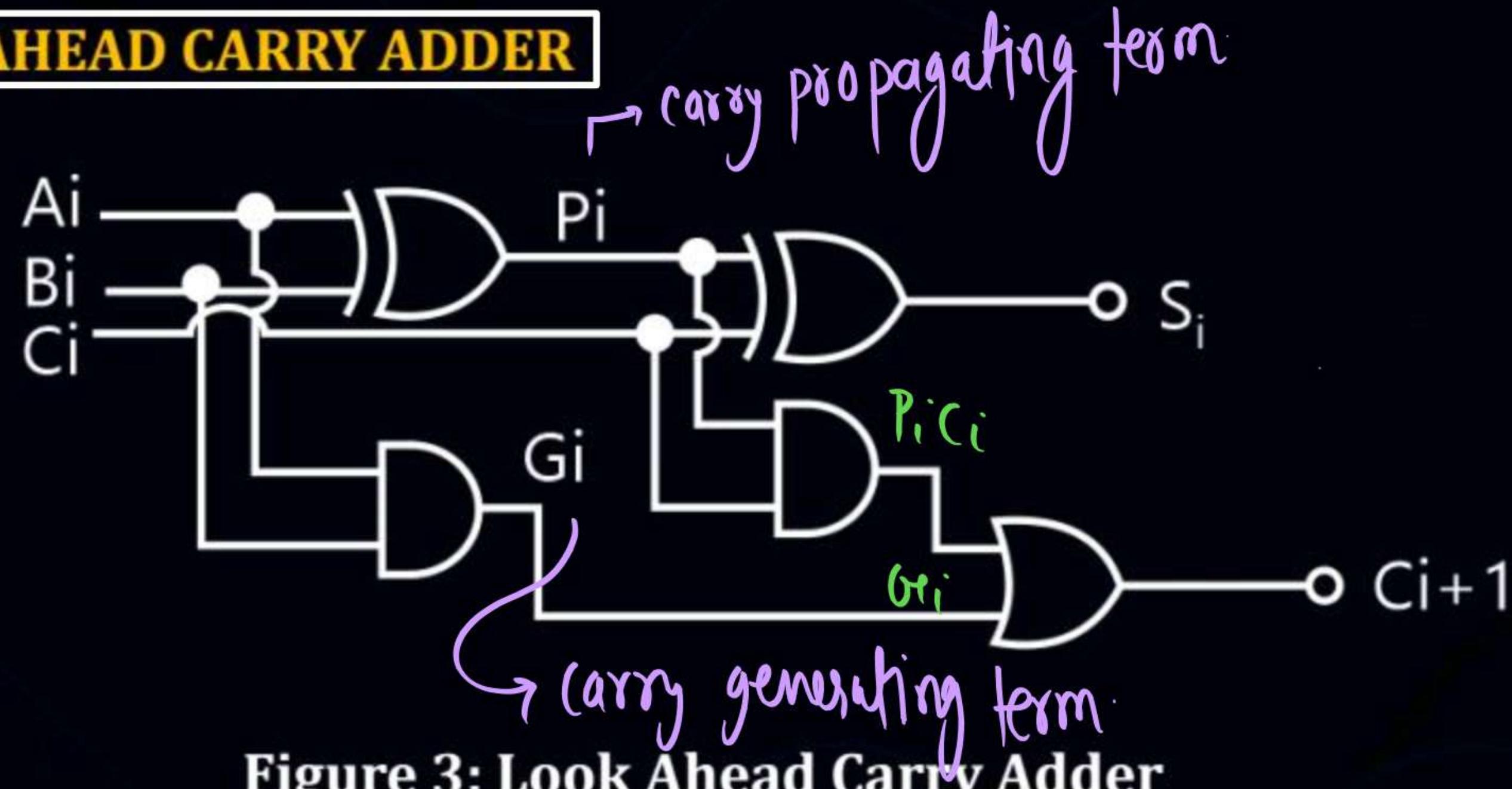
LOOK AHEAD CARRY ADDER

Figure 3: Look Ahead Carry Adder

$$P_i = A_i \oplus B_i$$

$$G_i = A_i B_i$$

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

$$C_1 = G_0 + P_0 C_0$$

$$S_i = P_i \oplus C_i$$

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

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

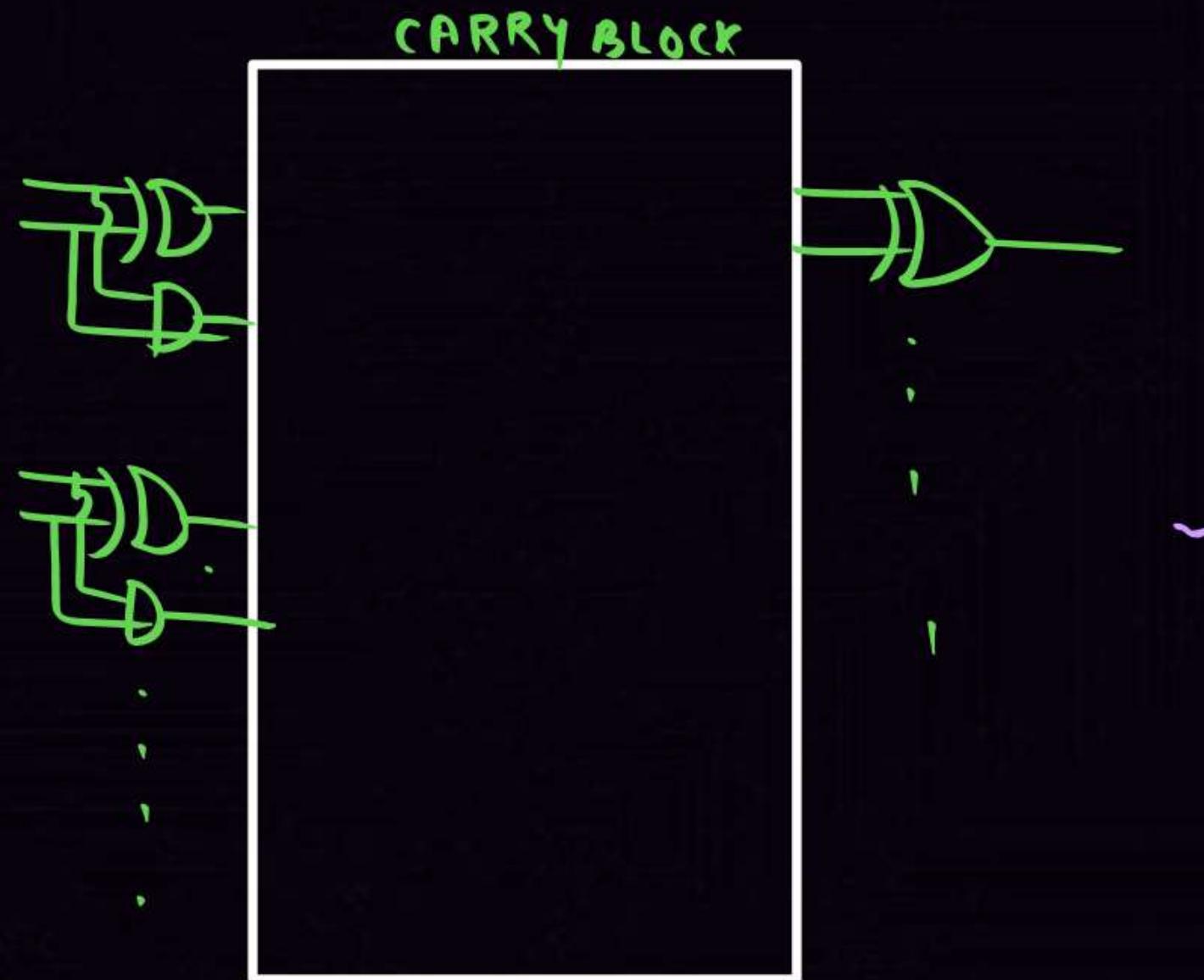
CARRY BLOCK

✓ NO. OF AND GATE

$$= \frac{n(n+1)}{2}$$

✓ OR GATE = n

$$\text{Delay} = 2 T_{pd}$$



Complete

$$\text{Total} = 4 T_{pd}$$

LACA, Multiplier }

LOOK AHEAD CARRY ADDER

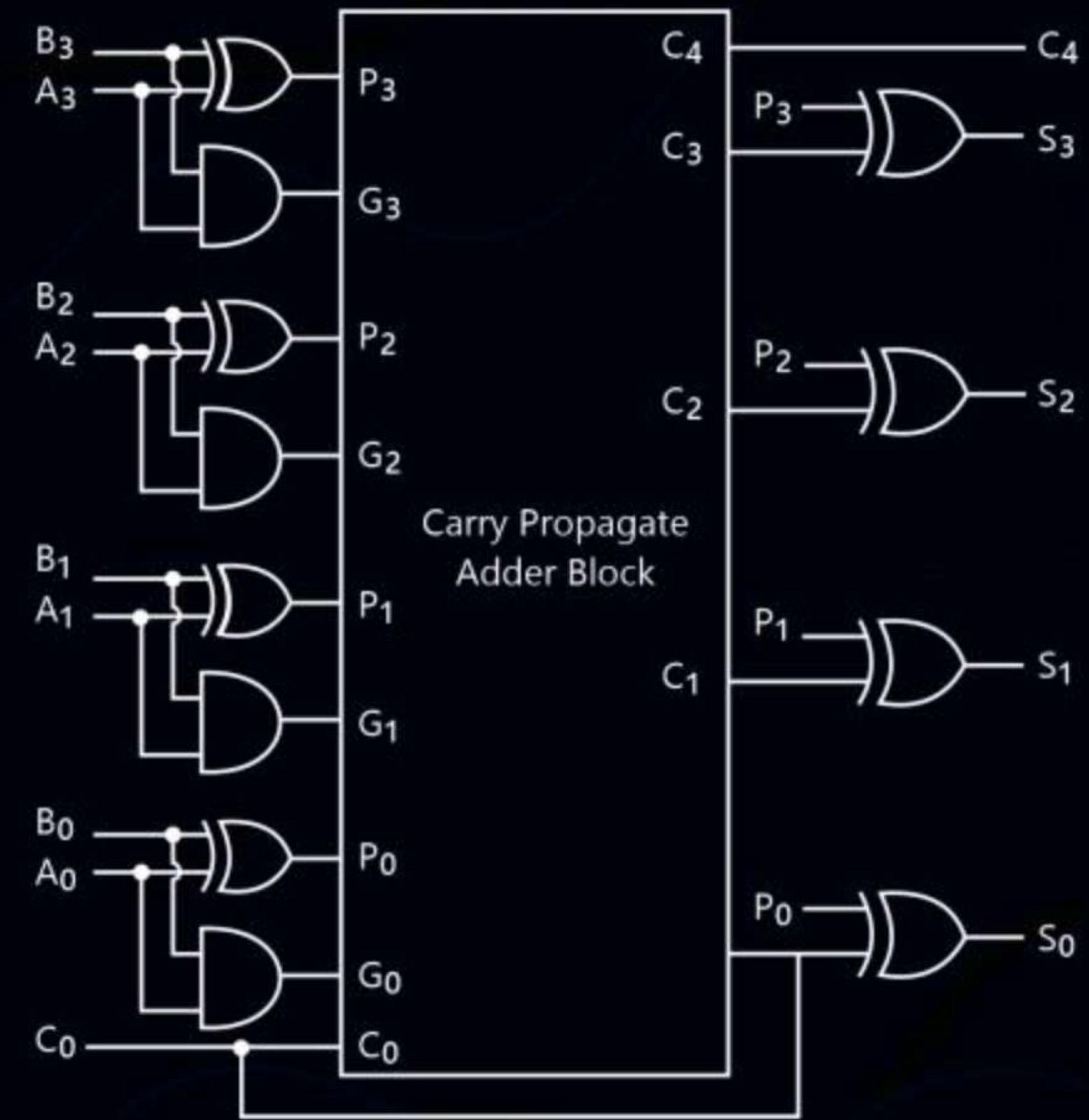


Figure 6: Look Ahead Carry Adder

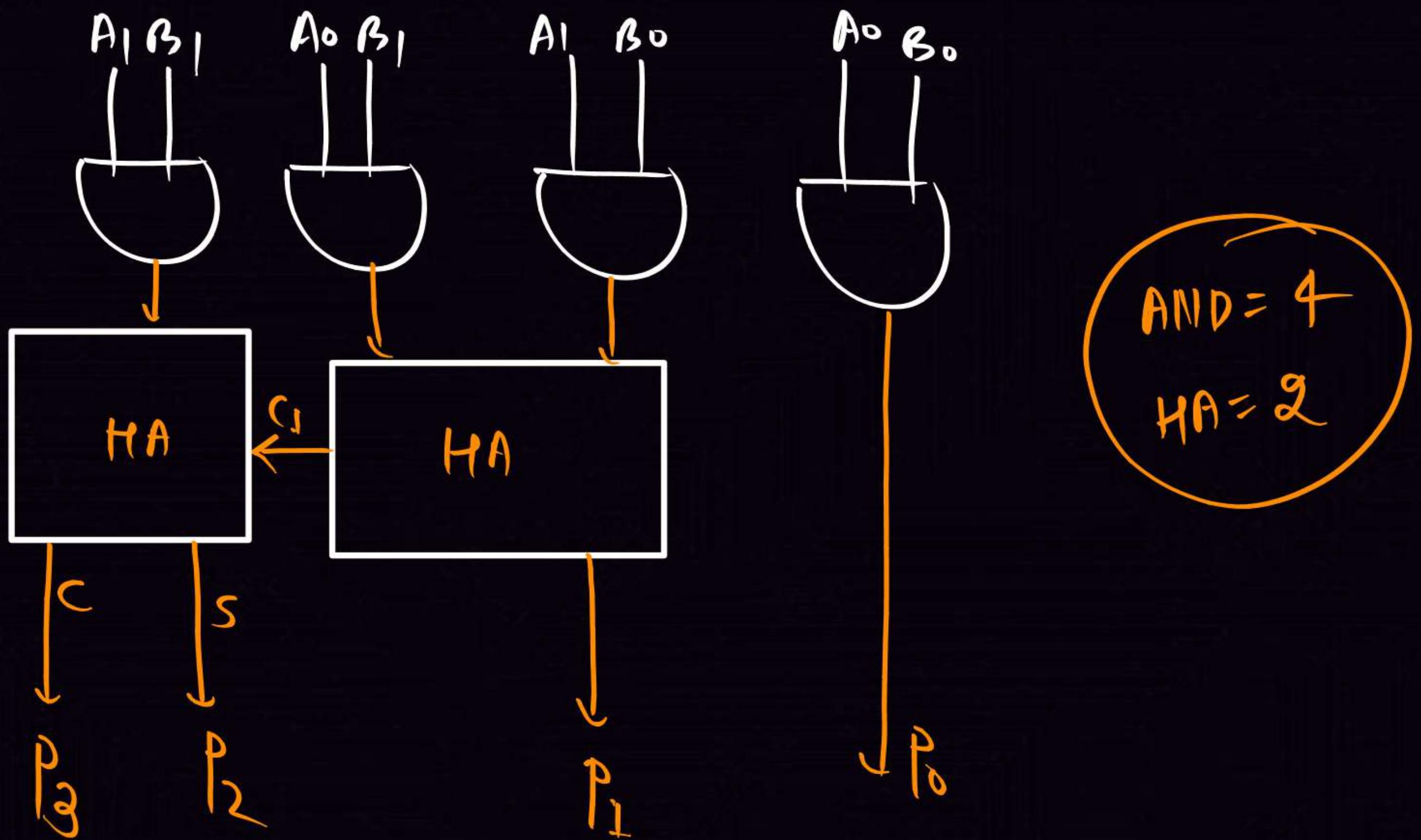
LACA, Multiplier }

MULTIPLIER2 bit Multiplier

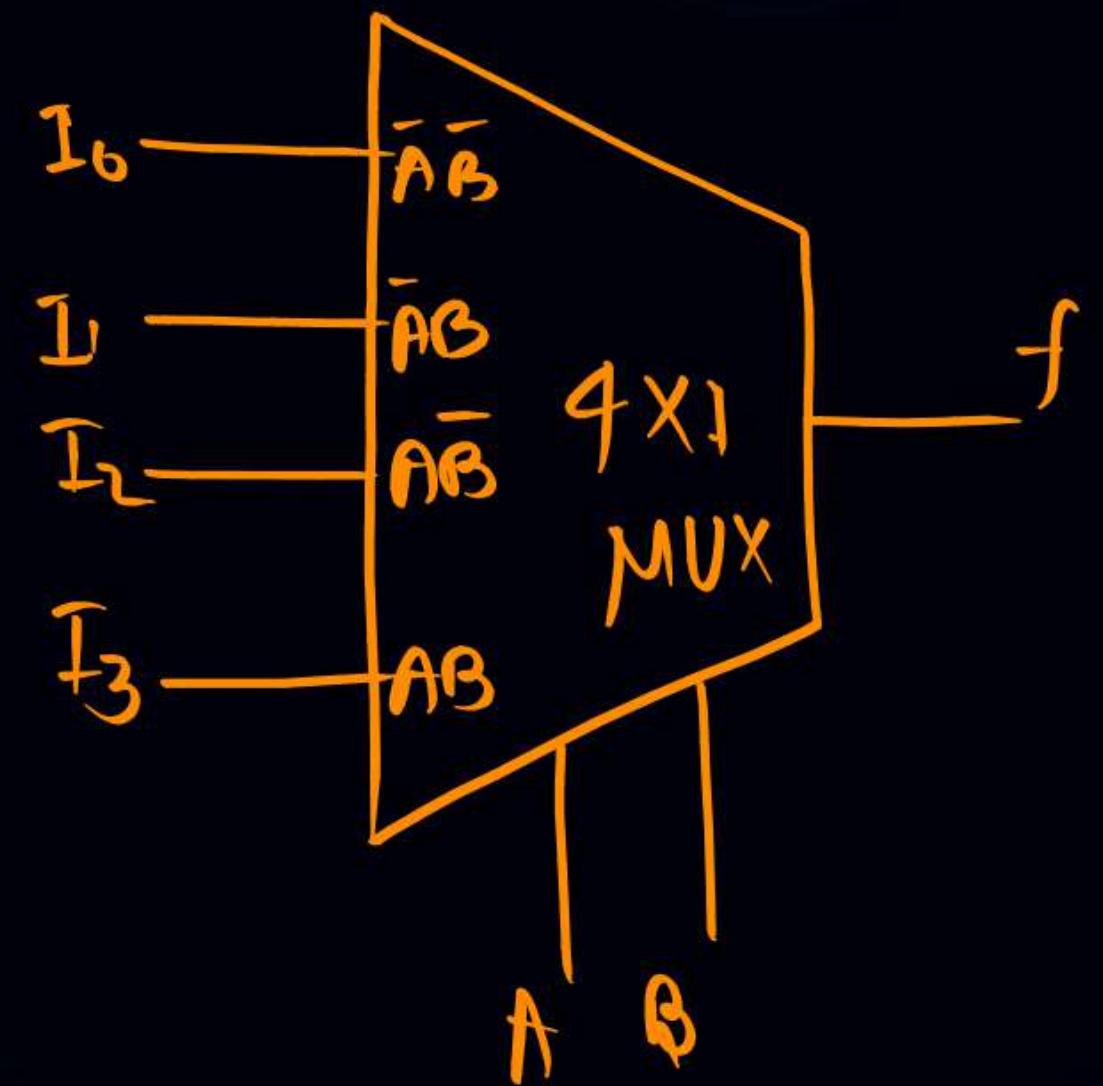
$$A = A_1 A_0$$

$$B = B_1 B_0$$

$$\begin{array}{r} A_1 \quad A_0 \\ B_1 \quad B_0 \\ \hline C_1 & A_1 B_0 & A_0 B_0 \\ A_1 B_1 & A_0 B_1 & X \\ \hline C_2 & S_2 & S_1 & S_0 \\ & \downarrow P_4 & \downarrow P_2 & \downarrow P_1 \\ & P_4 & P_2 & P_1 \\ & & & P_0 \end{array}$$



Multiplexer



TYPE - 1 :-

$$\begin{array}{c} \text{4x1 MUX} \\ = \end{array} \xrightarrow{\frac{16}{4} + \frac{4}{4}} \begin{array}{c} \text{16x1 MUX} \\ = \end{array}$$
$$4+1=5$$

$$\begin{array}{c} \text{8x1 MUX} \\ = \end{array} \xrightarrow{\frac{256}{8} + \frac{32}{8} + \frac{4}{8}} \begin{array}{c} \text{256x1 MUX} \\ = \end{array}$$
$$32+4+1=37$$

TYPE - 2. MUX as a universal Logic.

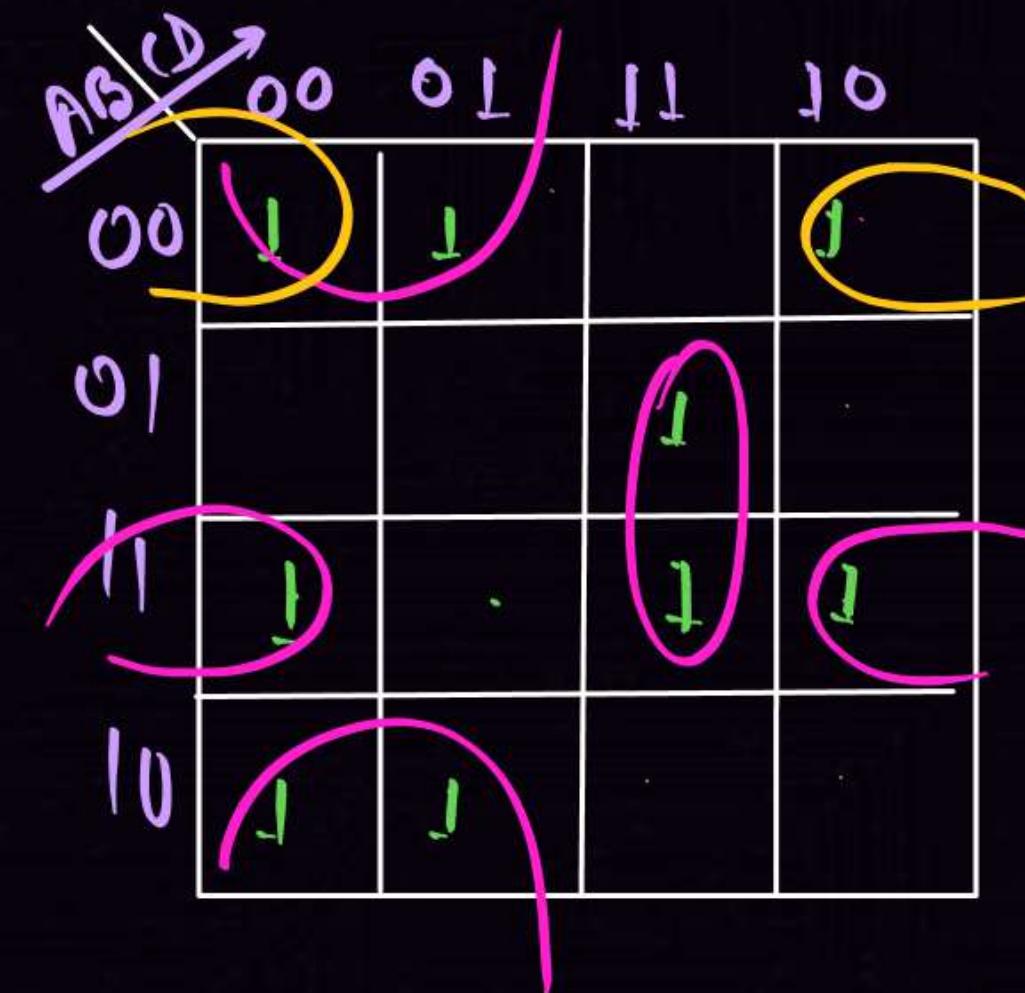
2X1 MUX

$$\begin{matrix} \text{NOT} \\ \text{AND} \\ \text{OR} \end{matrix} \left\{ \begin{matrix} 1 \rightarrow 2 \times 1 \text{ MUX} \end{matrix} \right.$$

$$\begin{matrix} \text{NAND} \\ \text{NOR} \\ \text{XOR} \\ \text{XINOR} \end{matrix} \left\{ \begin{matrix} 2 \rightarrow 2 \times 1 \text{ MUX} \end{matrix} \right.$$

TYPE - 3. Minimization

$$f(A, B, C, D) = \overline{m}(0, 1, 2, 7, 8, 9, 12, 14, 15)$$

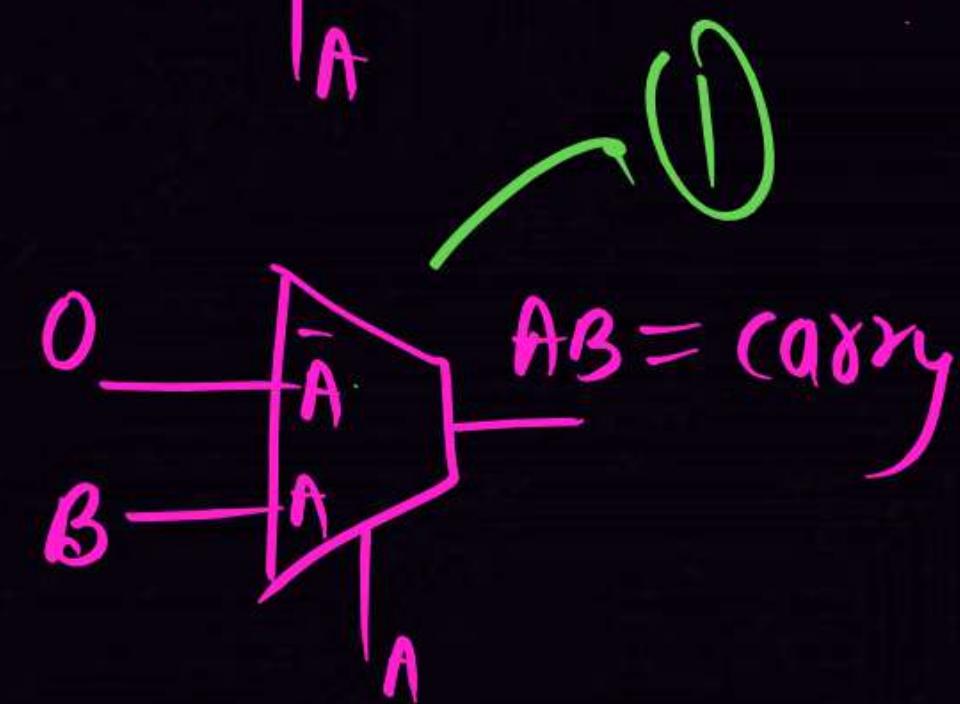
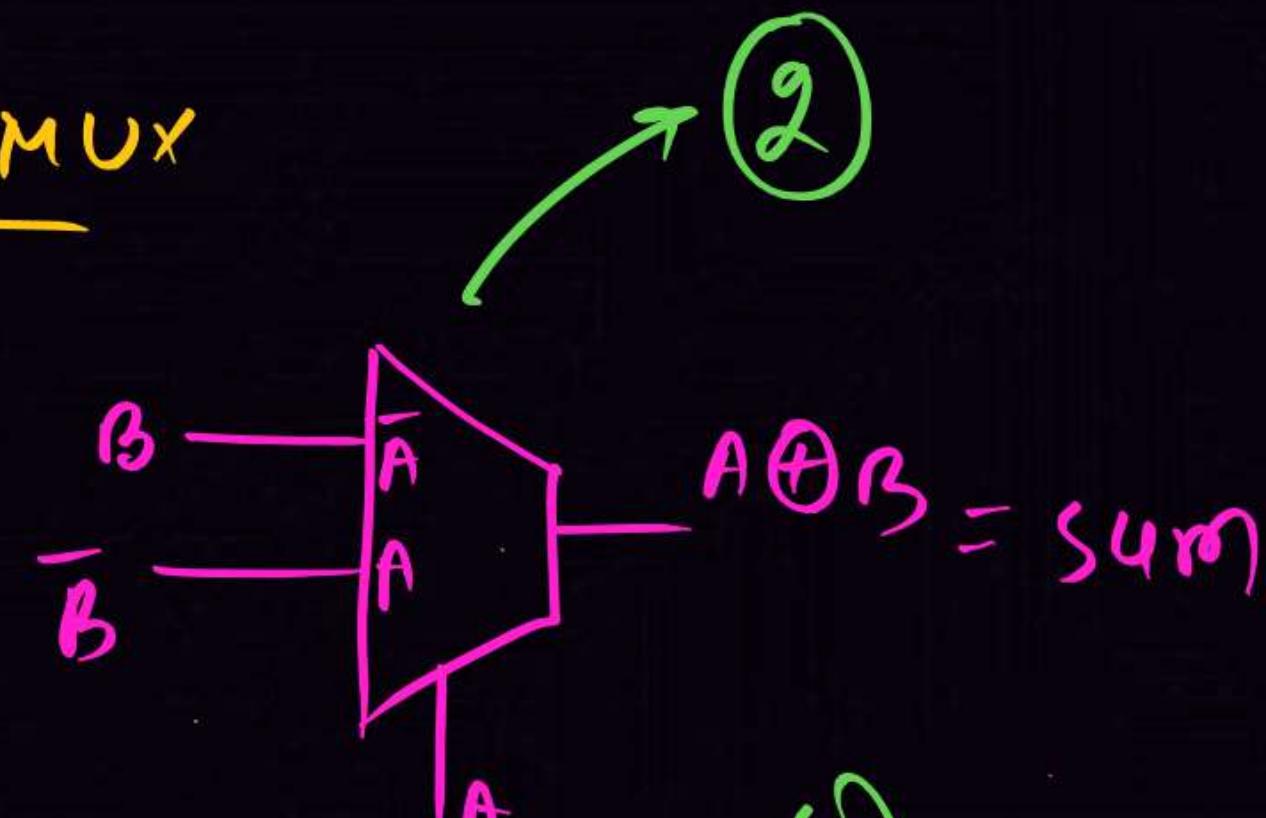


HA

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

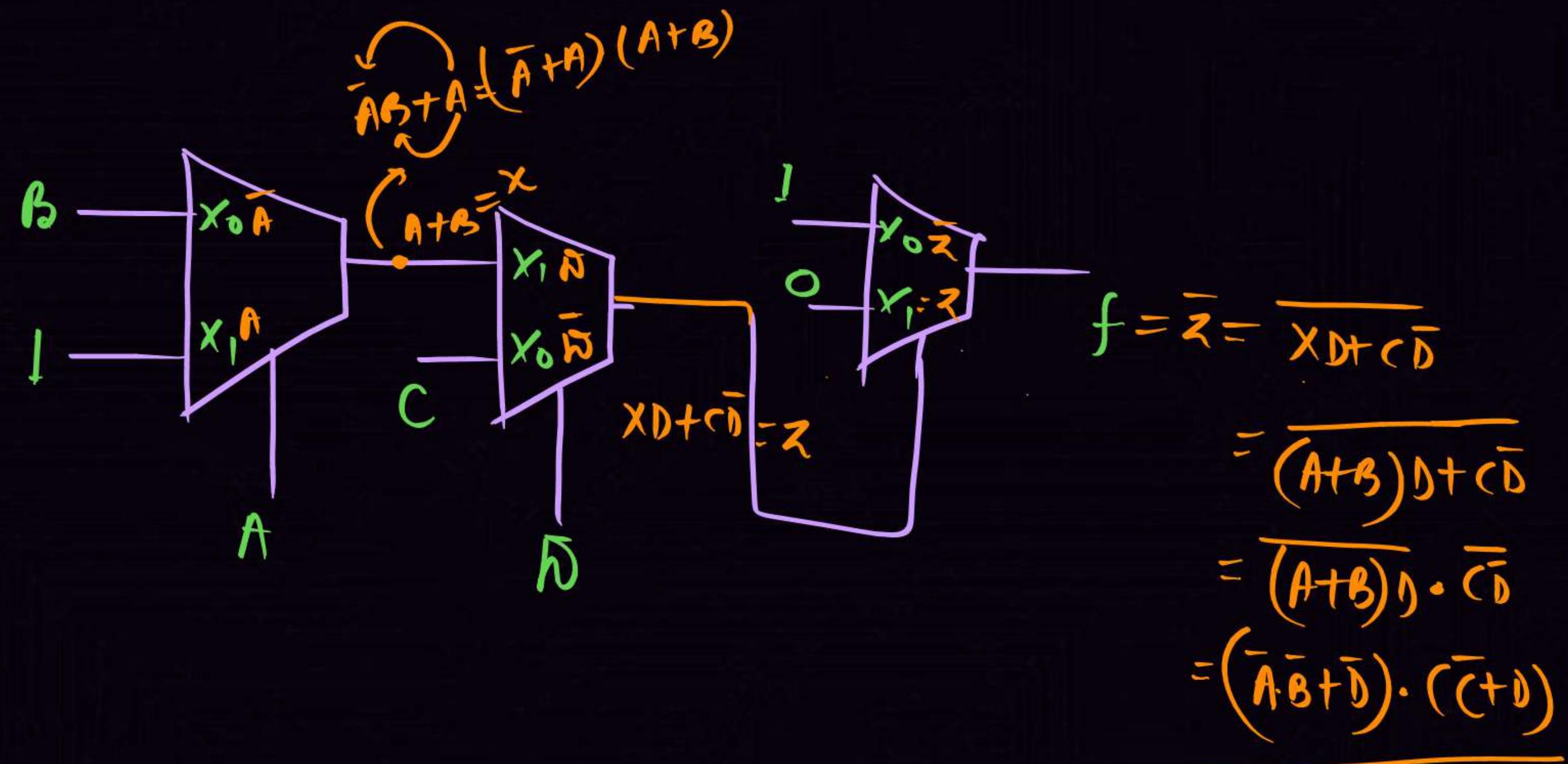
$$\text{carry} = AB$$

2x1 MUX



~~$2+1 = 3$~~

TYPE 4 Cascading of MUX.



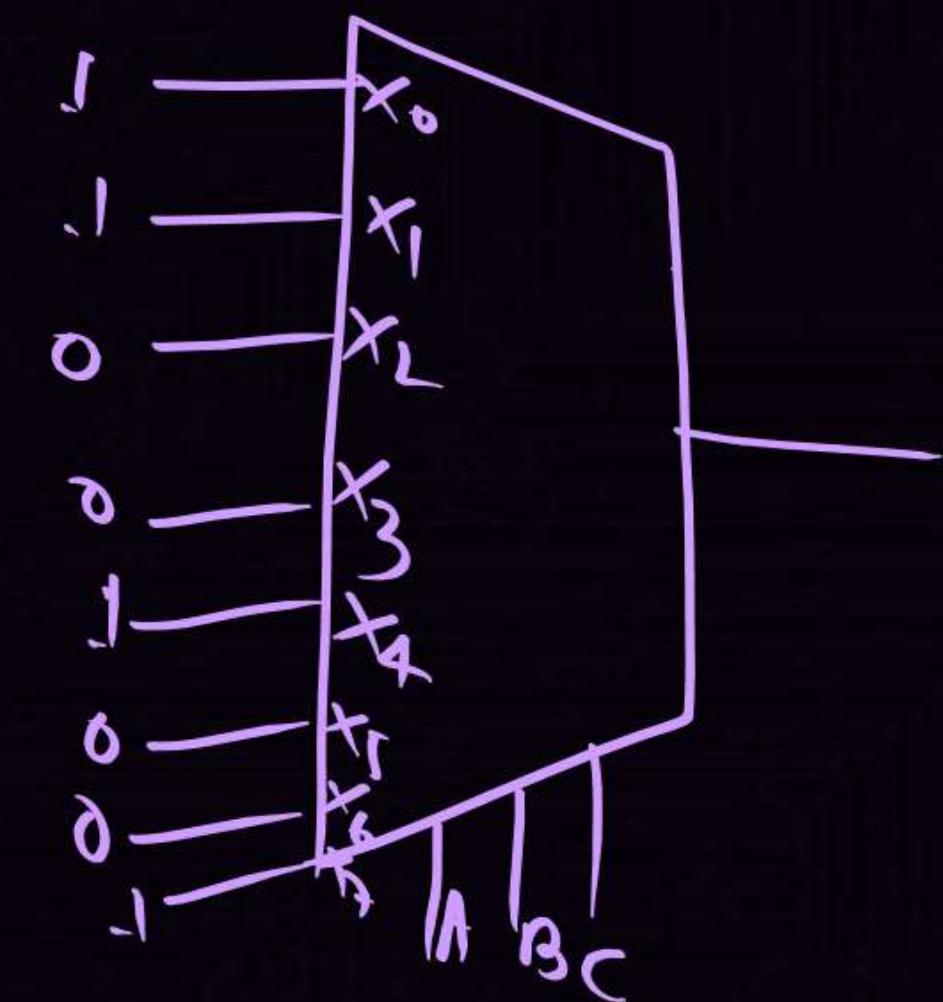
TYPE-5Implementation of function :-

$$f(A, B, C) = \sum m(0, 1, 4, 6, 7)$$

$$8x_1 \rightarrow AB$$

$$4x_L \rightarrow BC$$

$$2x_1 \rightarrow B$$



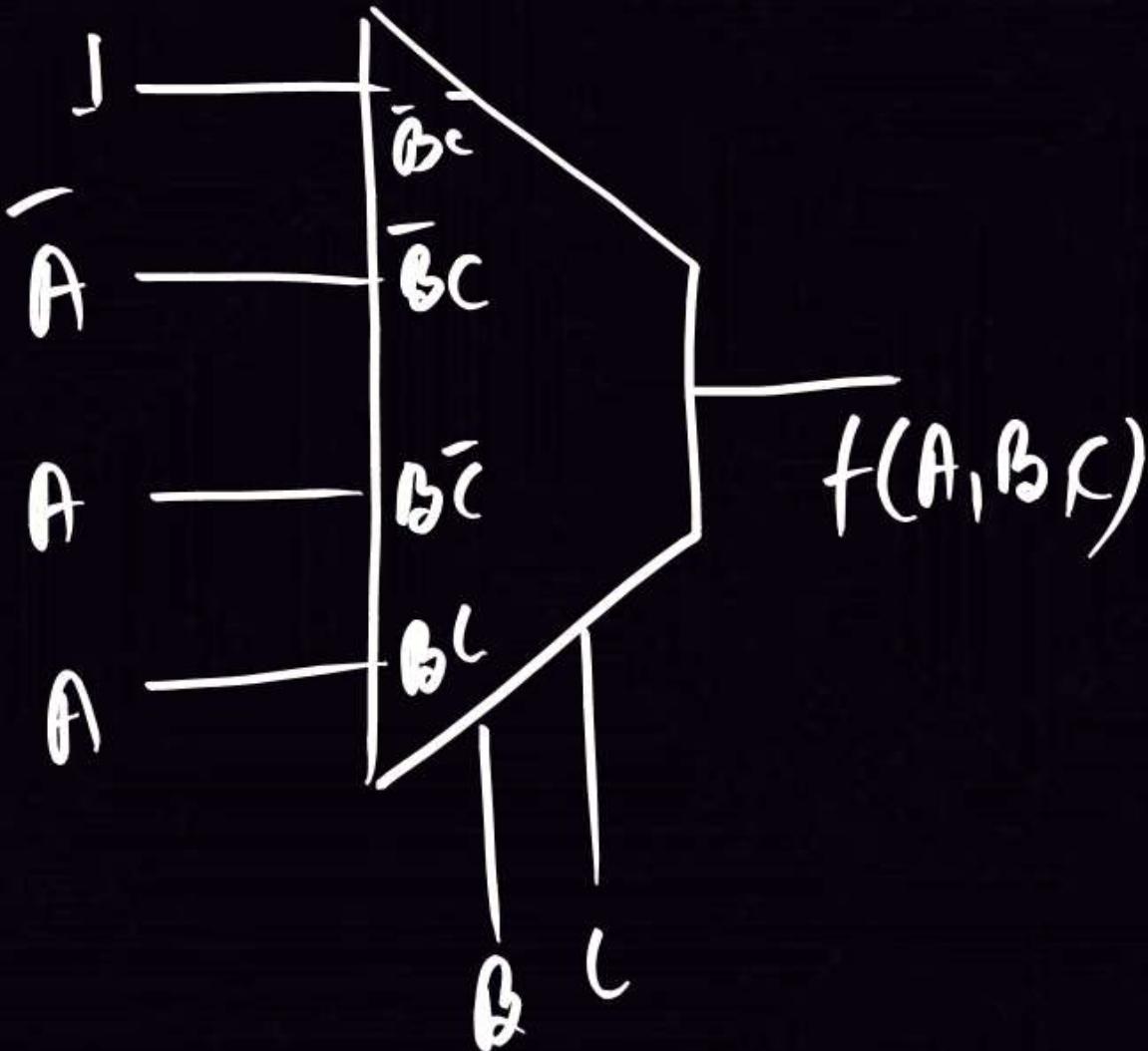
TYPE-5 Implementation of function :-

$$f(A, B, C) = \sum m(0, 1, 4, 5, 7)$$

4x1 MUX

(BC)

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



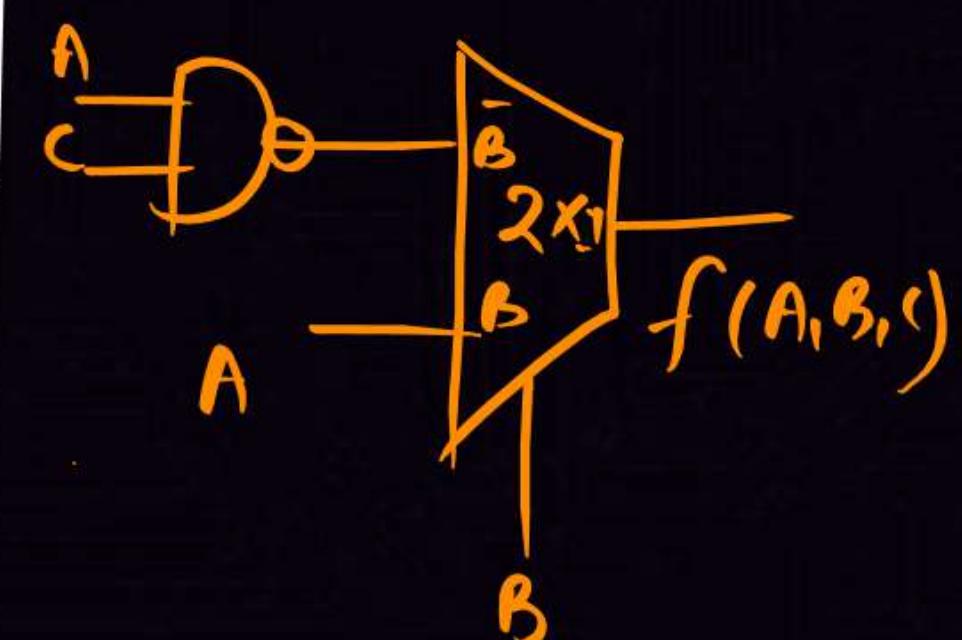
TYPE-5 Implementation of function :-

$$f(A, B, C) = \sum m(0, 1, 4, 6, 7)$$

~~2x1 MUX~~

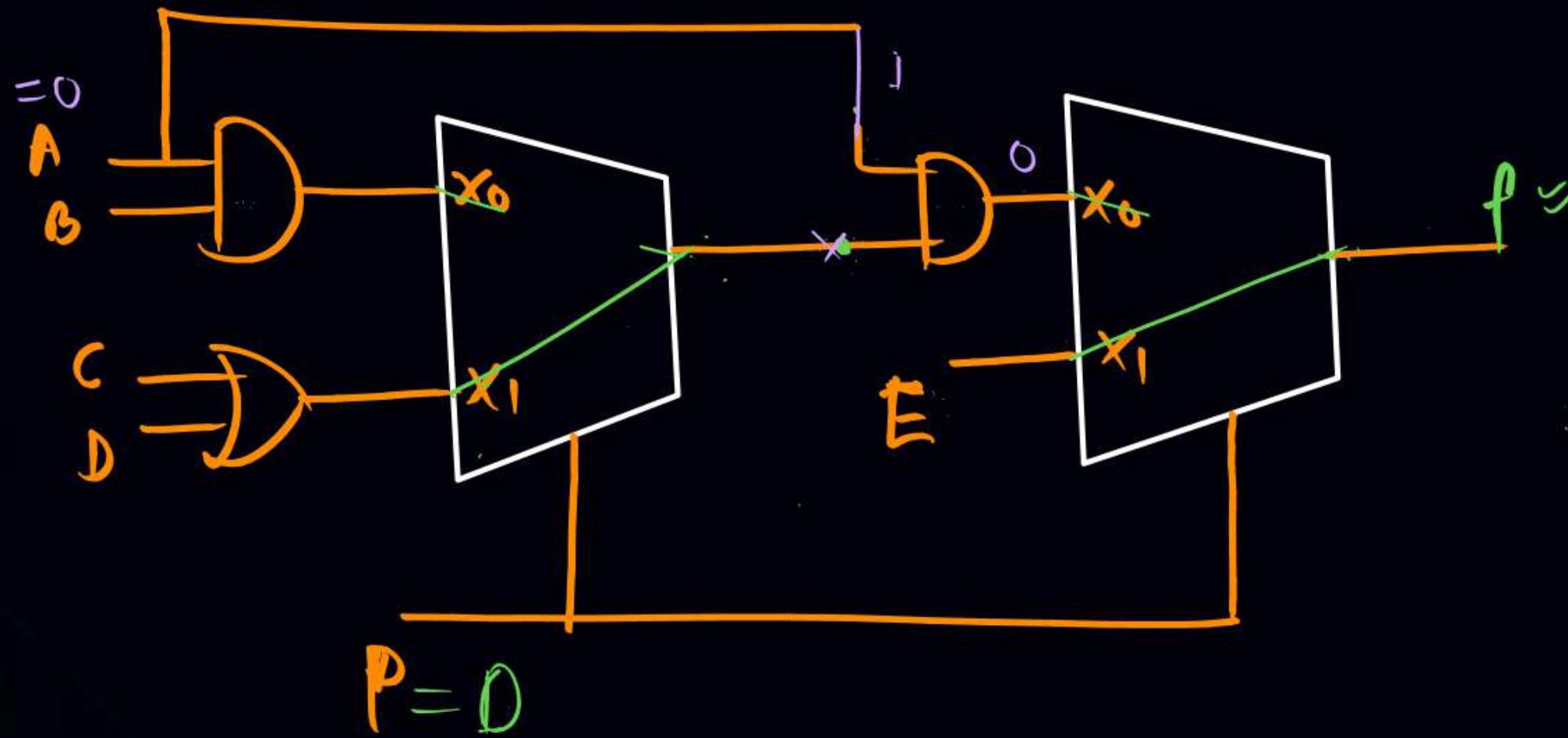
(BC)

	\bar{B}	B	
$\bar{A}\bar{C}$	$\bar{A}\bar{B}\bar{C}$	$\bar{A}B\bar{C}$	
$\bar{A}C$	$\bar{A}\bar{B}C$	$\bar{A}B\bar{C}$	
A \bar{C}	$A\bar{B}\bar{C}$	$AB\bar{C}$	
AC	$A\bar{B}C$	$AB\bar{C}$	
$\bar{A} + \bar{C}$			A
$\bar{A} \cdot C$			



Multiplexer

TYPE-6 "Delay"



$$\text{AND/OR} = 1.5 \mu\text{s}$$

$\text{MUX} \rightarrow 2 \mu\text{s}$

✓ Minimum Delay

✓ Maximum Delay

$$= 2 \mu\text{s}$$

$$= 7 \mu\text{s}$$

Case (1) P=0

$$(a) A=0$$

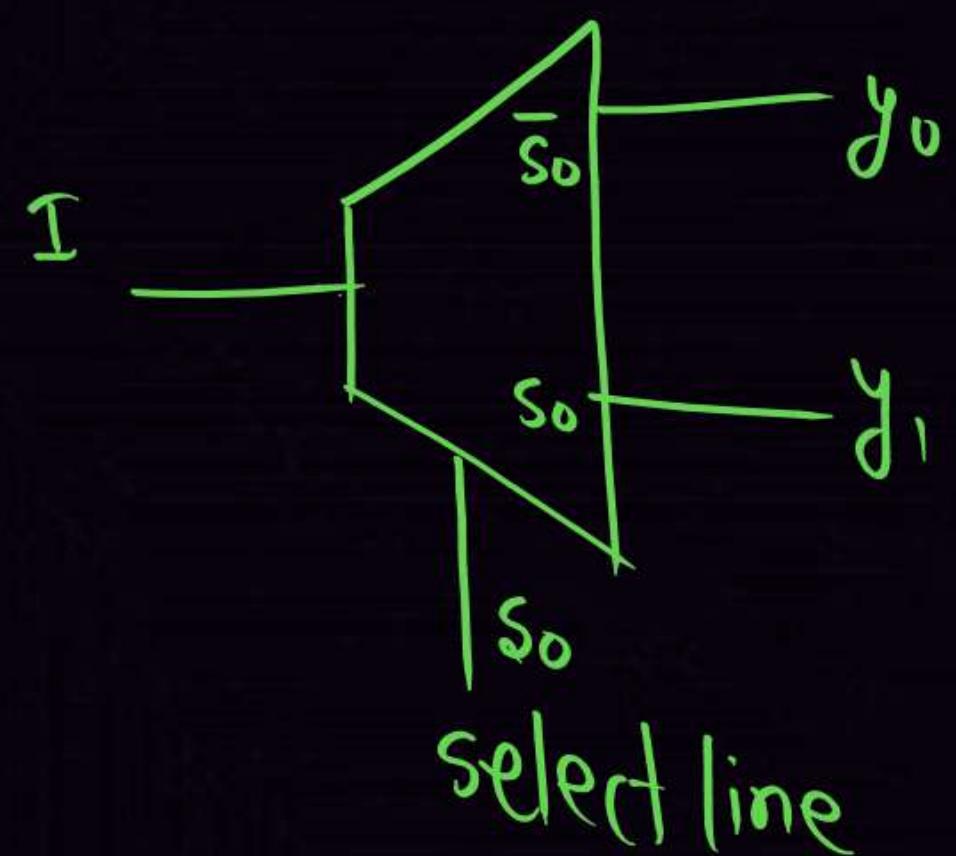
$$1.5 + 2 = 3.5 \mu\text{s}$$

$$(b) = A=1$$

$$3 + 4 = 7 \mu\text{s}$$

Case (2) P=1

$$9 \mu\text{s}$$

DE-MUX

Encoder

$\xrightarrow{4 \times 2}$

8×3

16×4
 \equiv

Weighted code

Self complemented

{
BCD
Excess-3
4221

Decoder



2x4 Decoder

3x8 Decoder

4x16 Decoder



BCD to Decimal Decoder

0-9

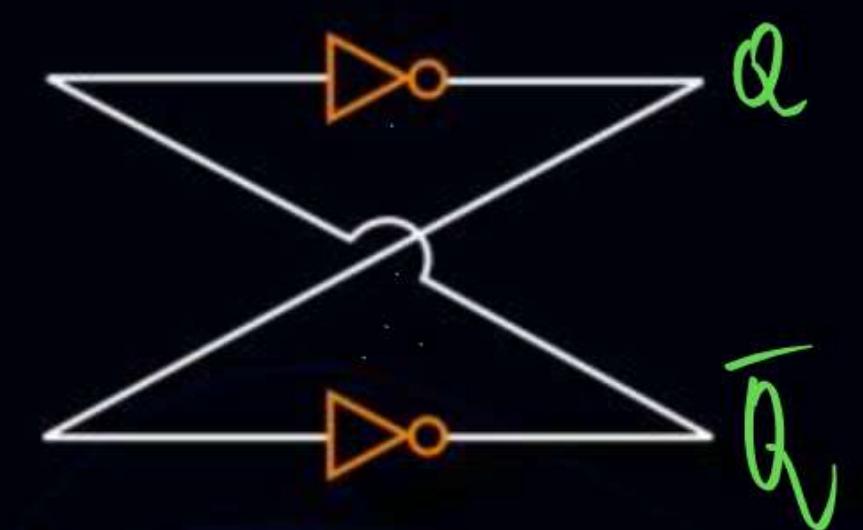
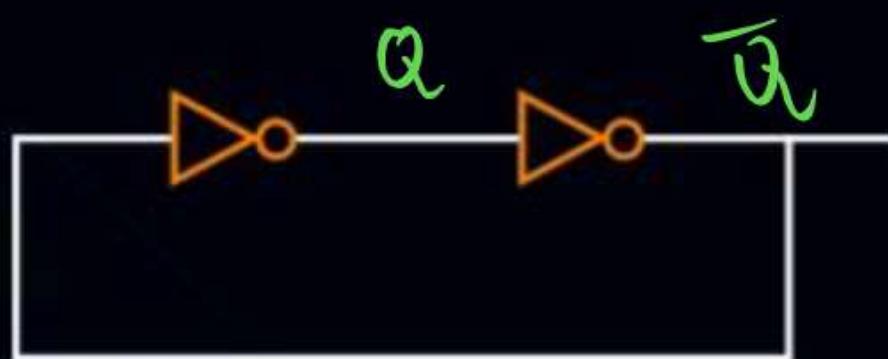
$10 \rightarrow 15 \rightarrow X =$

Sequential Circuit }

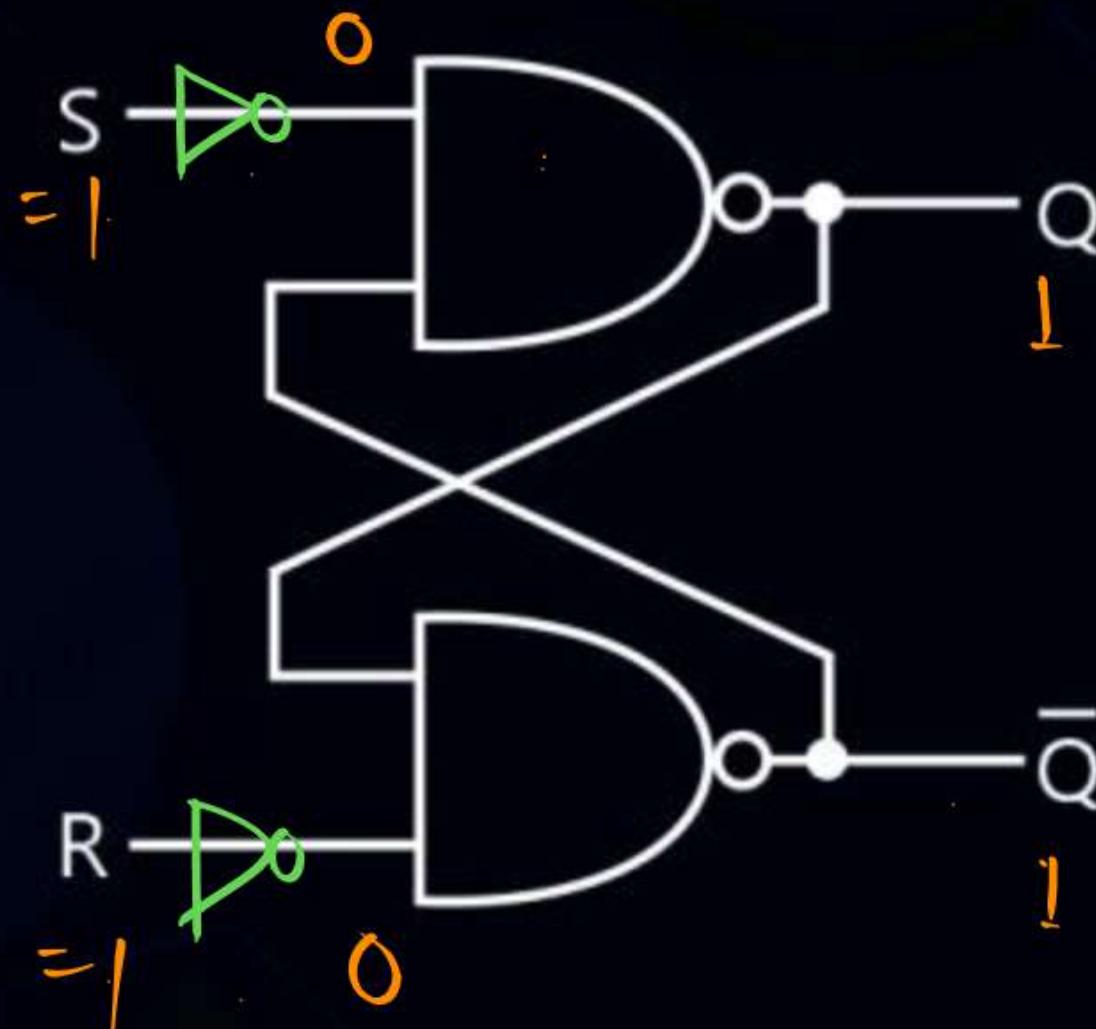
1. A circuit with feedback and memory are called sequential circuit.
2. Output of the sequential circuit depends on previous output as well as present state of input.

Latches

- Basic memory element ✓
- Latches are level triggered ✓
- Latches has two output which is complement of each other

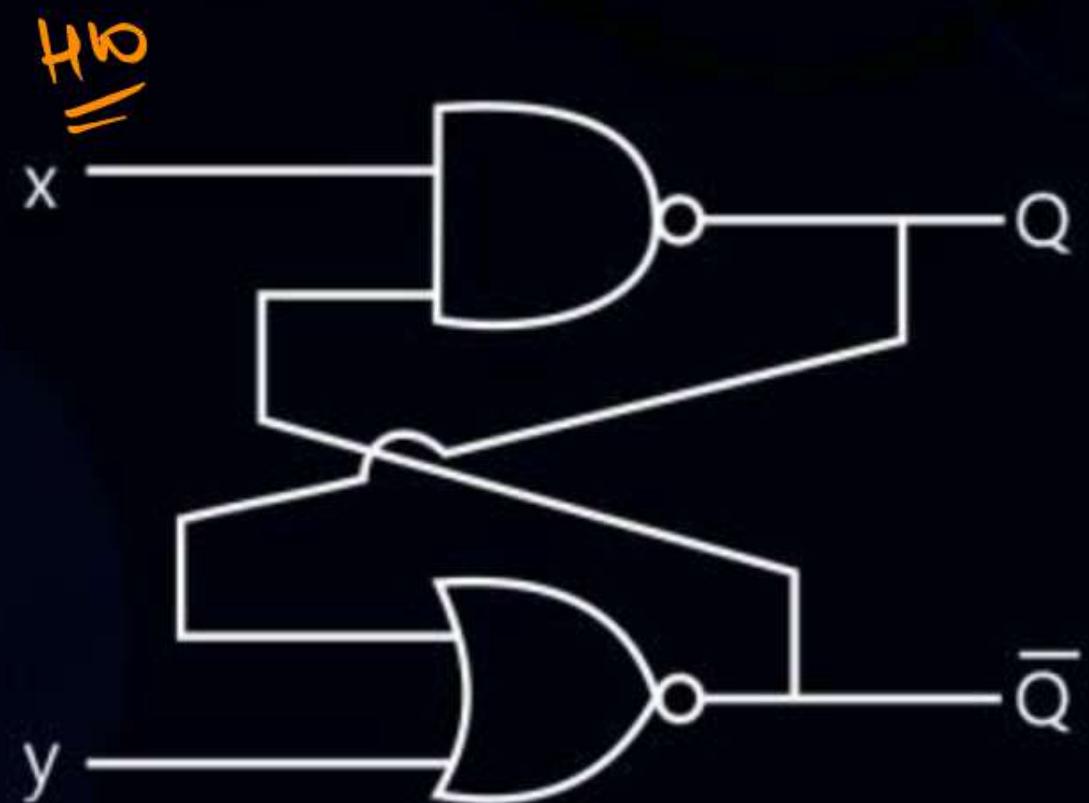


Latches



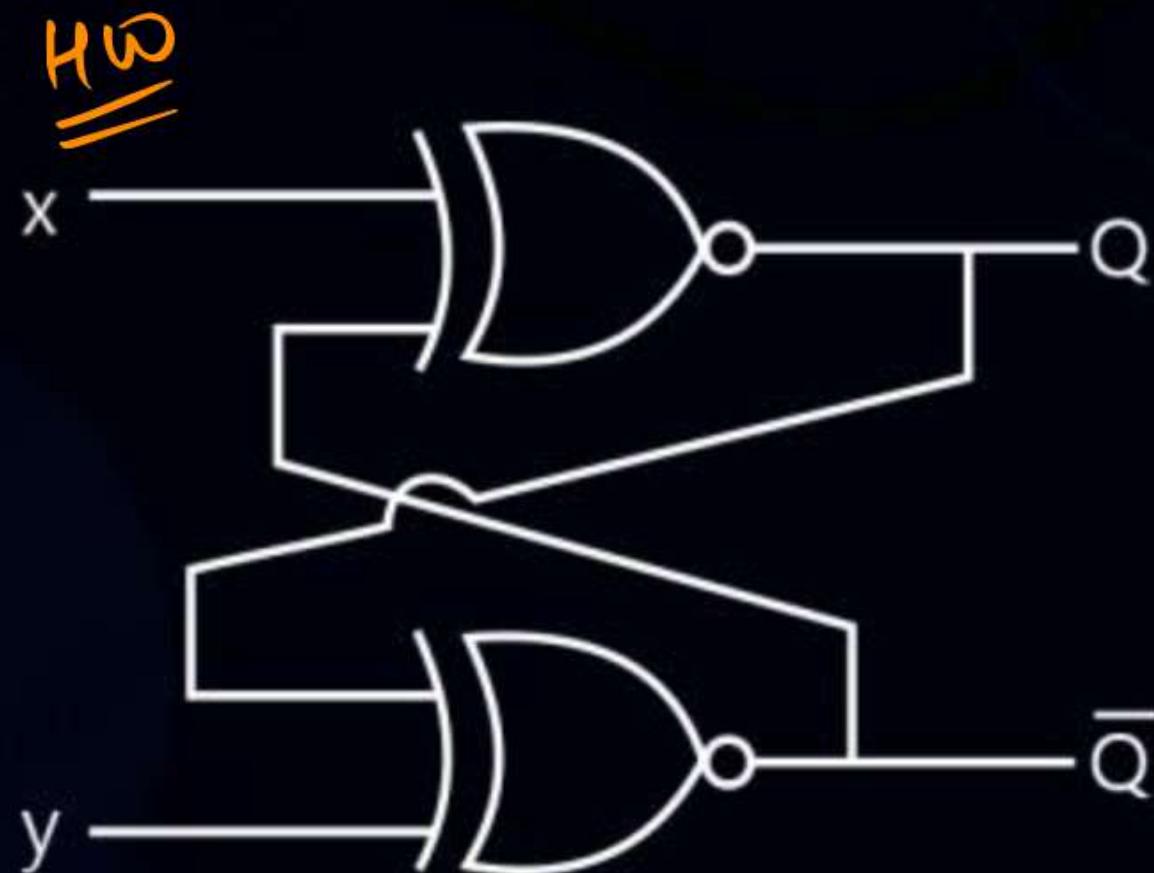
S	R	Q	\bar{Q}	
0	0	Q	\bar{Q}	→ HOLD
0	1	0	1	
1	0	1	0	
1	1	1	1	→ Invalid

Latches



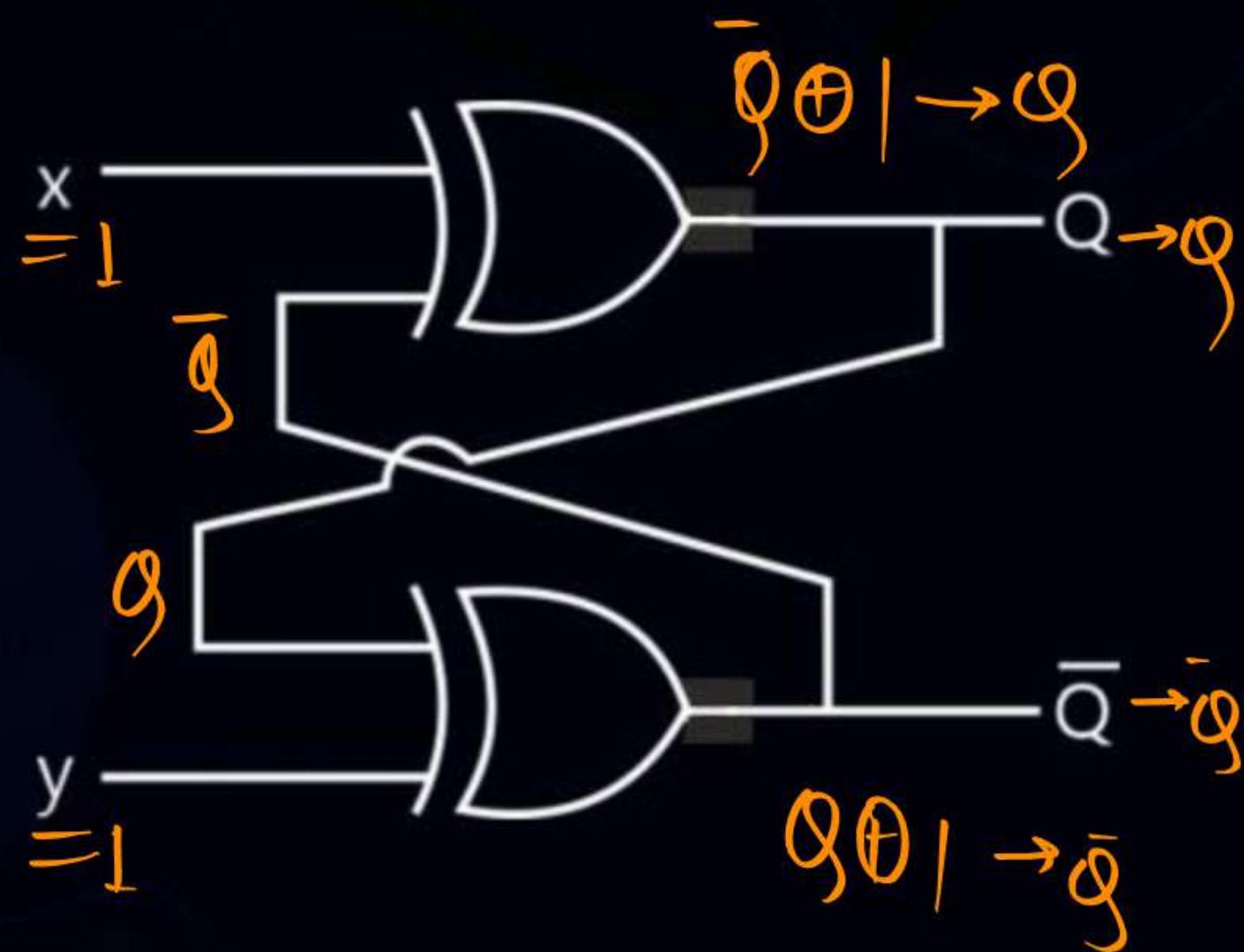
X	Y	Q	
0	0		
0	1		
1	0		
1	1		

Latches



X	Y	Q
0	0	
0	1	
1	0	
1	1	

Latches



$$Q \oplus Q = 0$$

$$Q \oplus 0 = Q$$

$$\bar{Q} \oplus 0 = \bar{Q}$$

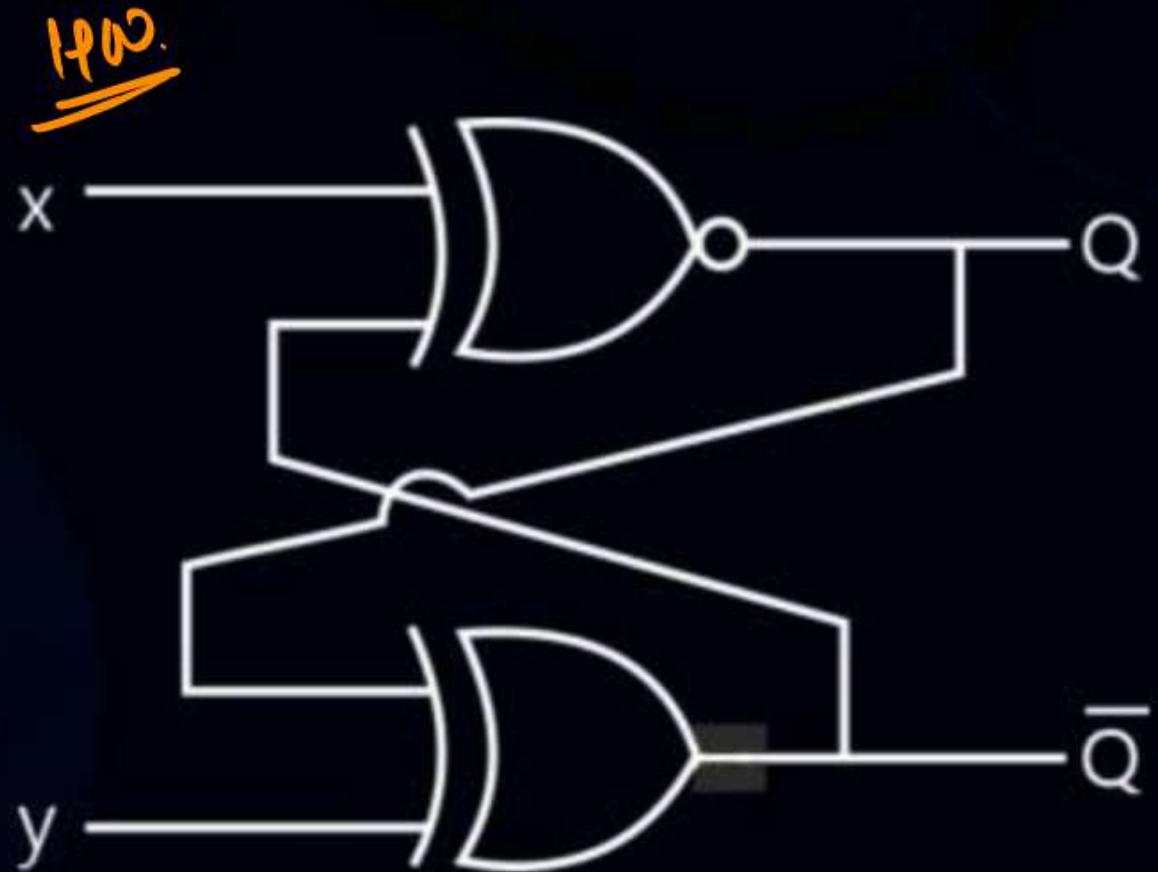
$$Q \oplus \bar{Q} = 1$$

$$Q \oplus 1 = \bar{Q}$$

$$\bar{Q} \oplus 1 = Q$$

X	Y	Q	\bar{Q}	
0	0	\bar{Q}	Q	→ Toggle
0	1	\bar{Q}	\bar{Q}	→ Invalid
1	0	Q	Q	→ Invalid
1	1	Q	\bar{Q}	→ Hold

Latches



X	Y	Q	
0	0		
0	1		
1	0		
1	1		

Latches

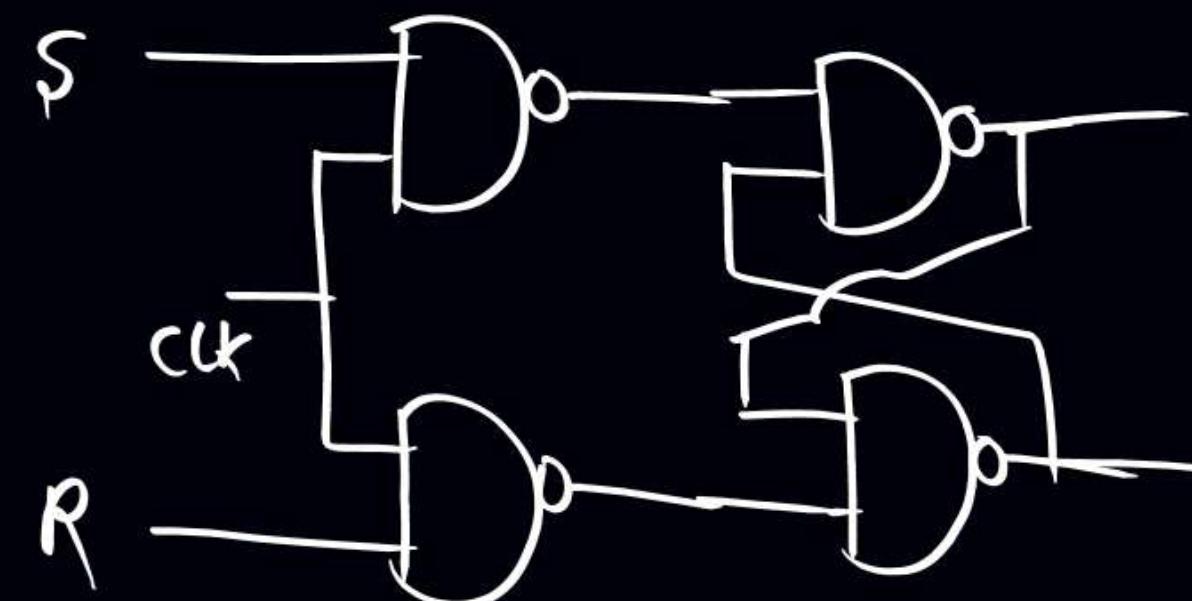
Discussion

Circuit → Truth table

(
≡
Delay)

Flip-Flop.

① SR FF.

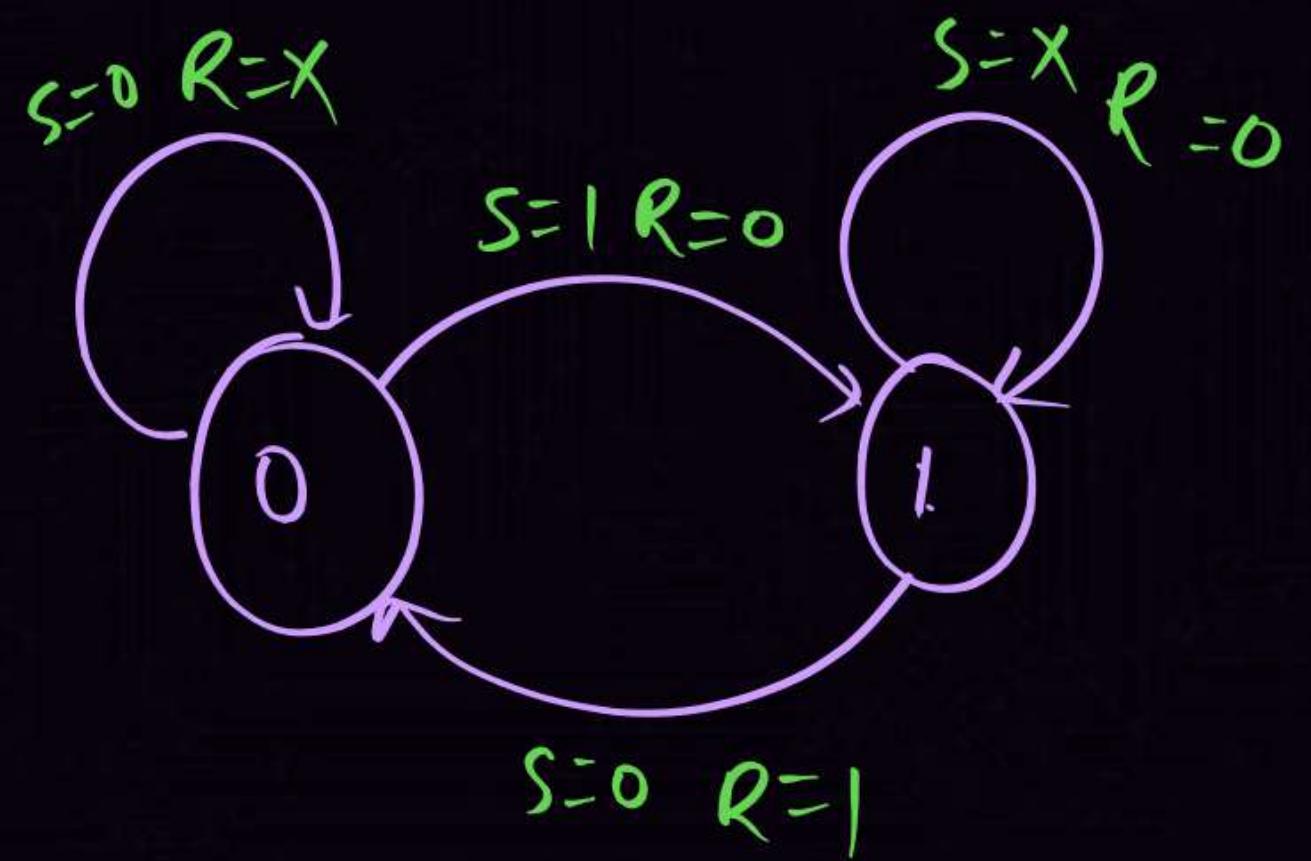


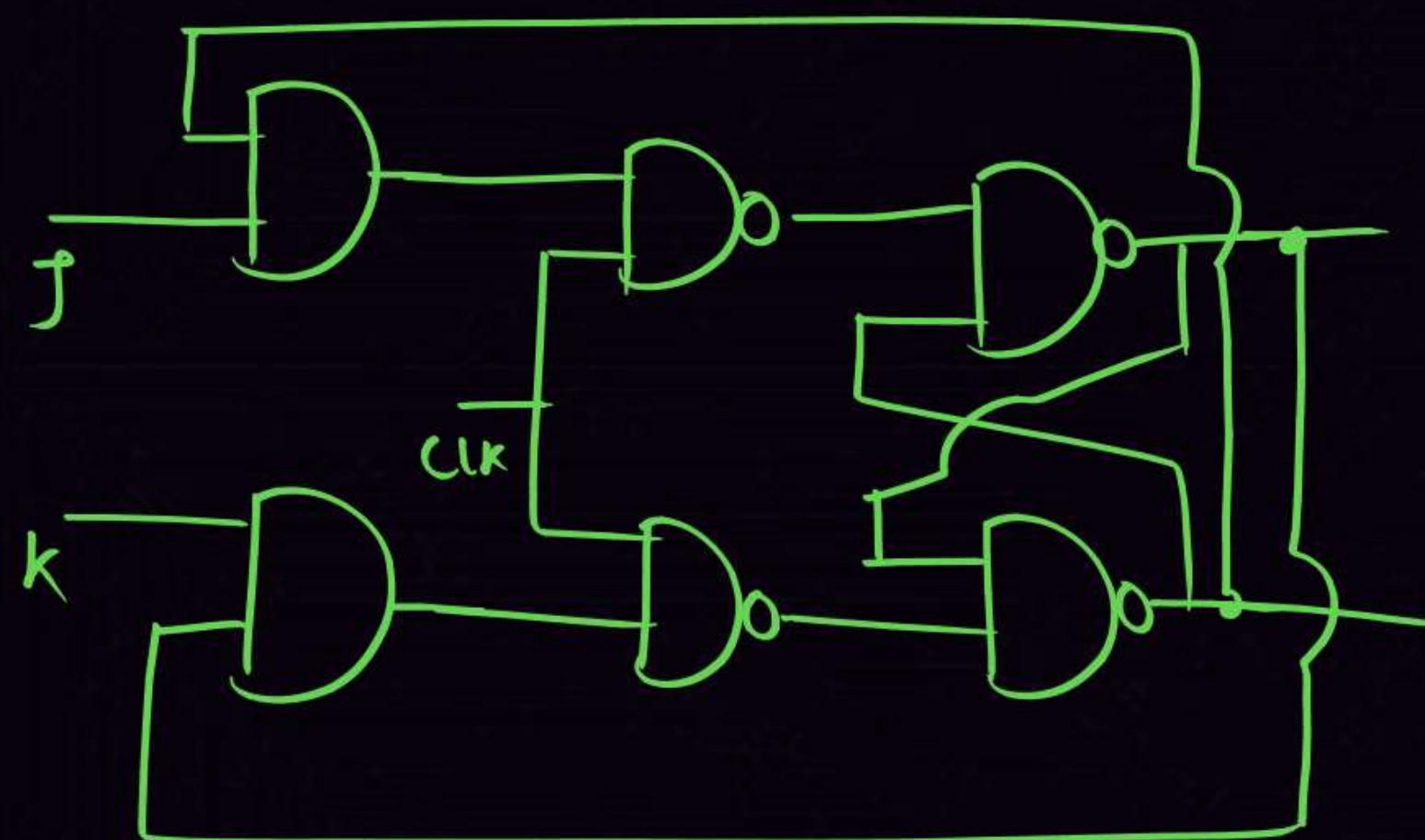
S	R	Q _{n+1}
0	0	Q _n
0	1	0
1	0	1
1	1	X

$$Q_{n+1} = S + \bar{R} Q_n$$

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

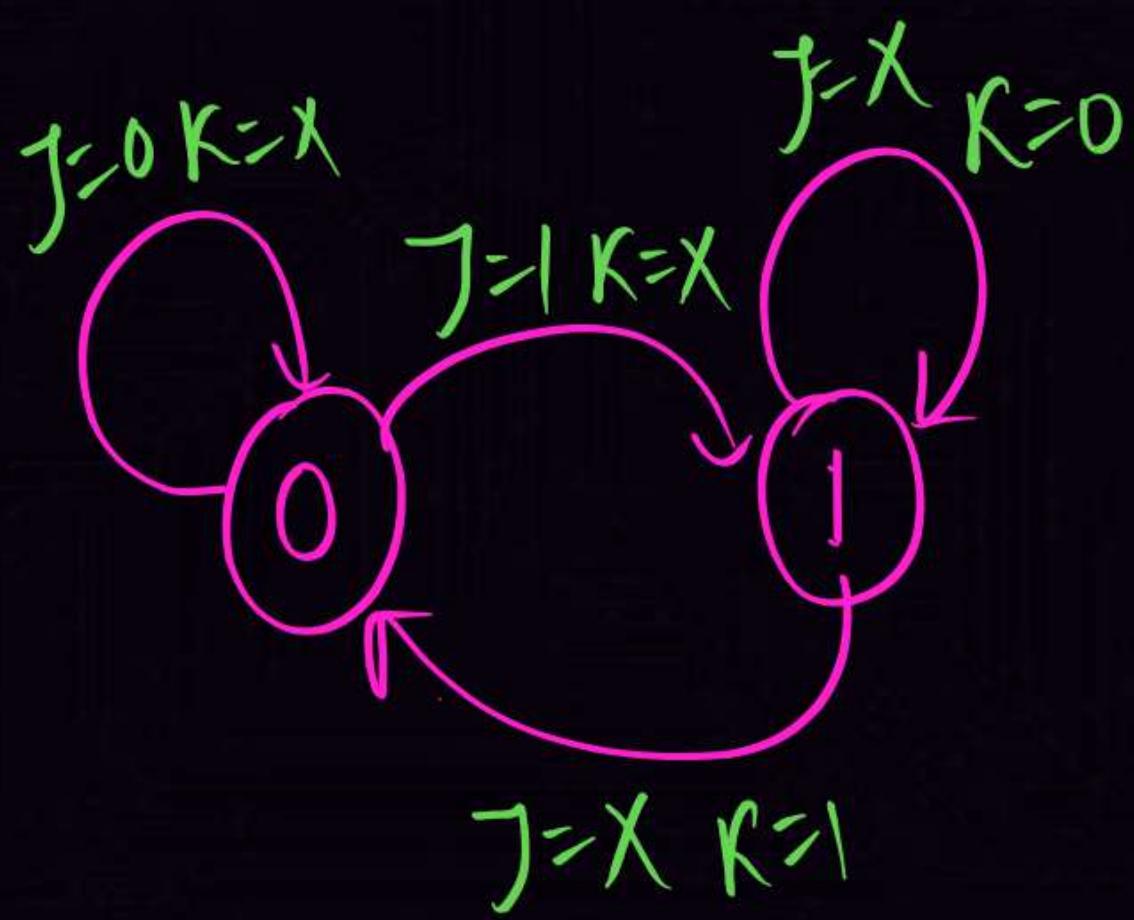


JK Flip

J	K	Q_{n+1}
0	0	Q_n
0	1	0
1	0	1
1	1	\bar{Q}_n

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

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



D-Flip-Flop

- (i) It is known as Delay FF or Transparent FF.
- (ii) In the D FF whatever the input is applied it will directly come to the output along with the clock.

1. Symbol

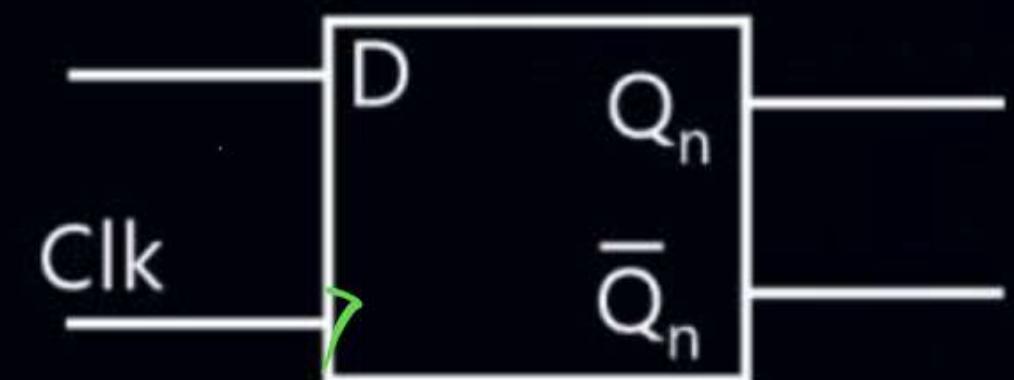


Figure 1: D Flip Flop

2. Truth Table

D	Q_{n+1}
0	0
1	1

Table 1: Truth Table of D Flip Flop

D-Flip-Flop

3. Characteristic table



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

Table 2: Characteristic Table of D Flip Flop

4. Characteristic Equation

$$\boxed{Q_{n+1} = D}$$

D-Flip-Flop

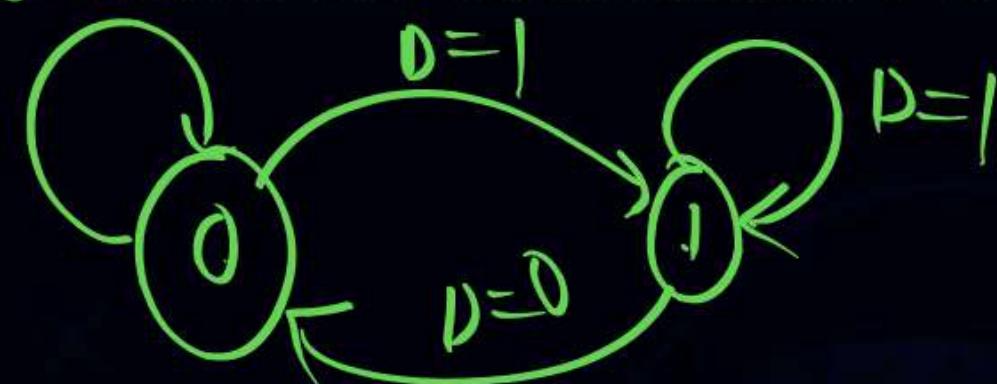
5. Excitation table:

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

$$Q_{n+1} = D$$

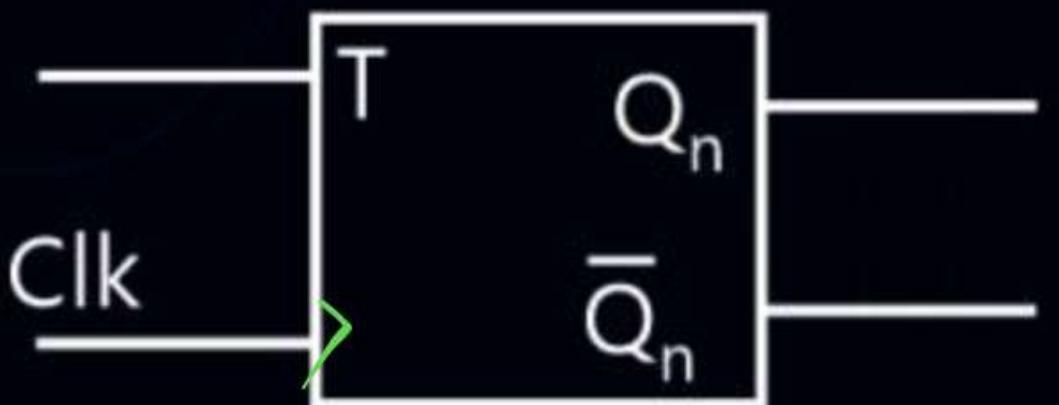
$V=0$ Table 3: Excitation Table of D Flip Flop

6. State diagram:



T-Flip-Flop(Toggle Flip-Flop)

1. Symbol



2. Truth Table

T	Q_{n+1}
0	Q_n
1	\overline{Q}_n

Table 4: Truth Table of T Flip Flop

T-Flip-Flop(Toggle Flip-Flop)

3. Characteristic table

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

Table 5: Characteristic Table of T Flip Flop

4. Characteristic Equation

$$Q_{n+1} = T \oplus Q_n$$

T-Flip-Flop(Toggle Flip-Flop)

5. Excitation table:

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

Table 6: Excitation Table of T Flip Flop

6. State diagram:

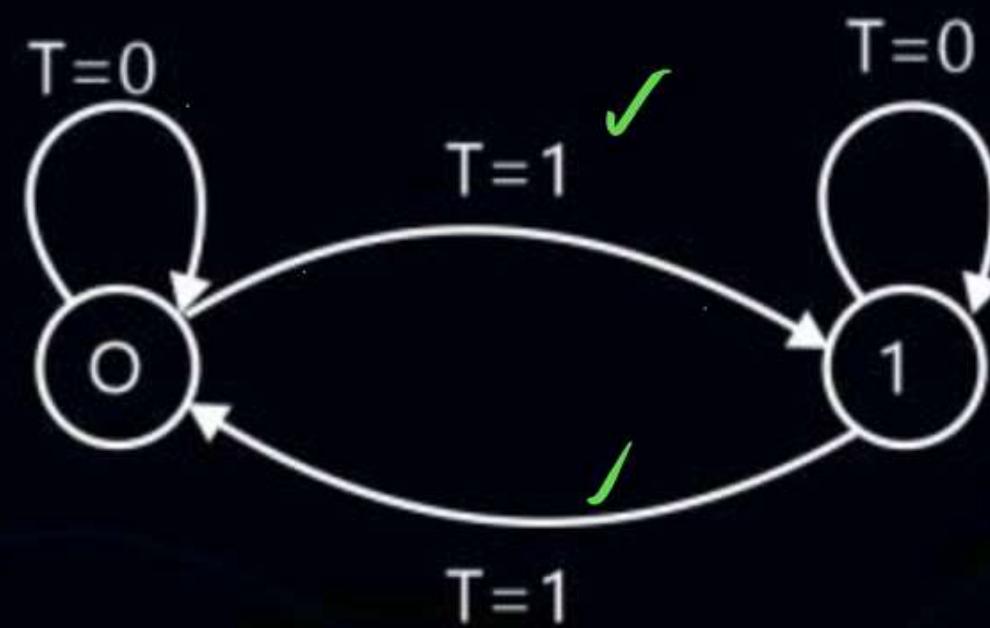


Figure 3: State diagram of T Flip Flop

Designing of one Flip-Flop by Using other Flip-Flop }



Step (1) : Write the characteristic table of desired Flip Flop.

Step (2) : Write the excitation table of available Flip Flop. ✓

Step (3) : Write the logical expression.

Step (4) : Minimize the logical expression.

Step (5) : Hardware implementation.

Q Design NK FF by using SR FF.

N	K	Q_{n+1}
0	0	X
0	1	\bar{Q}_n
1	0	1
1	1	0

Step 1

L

Step 2

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

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

S

$X \otimes n$

	00	01	11	10
0	\times	\times		\downarrow
1	\downarrow	\times		

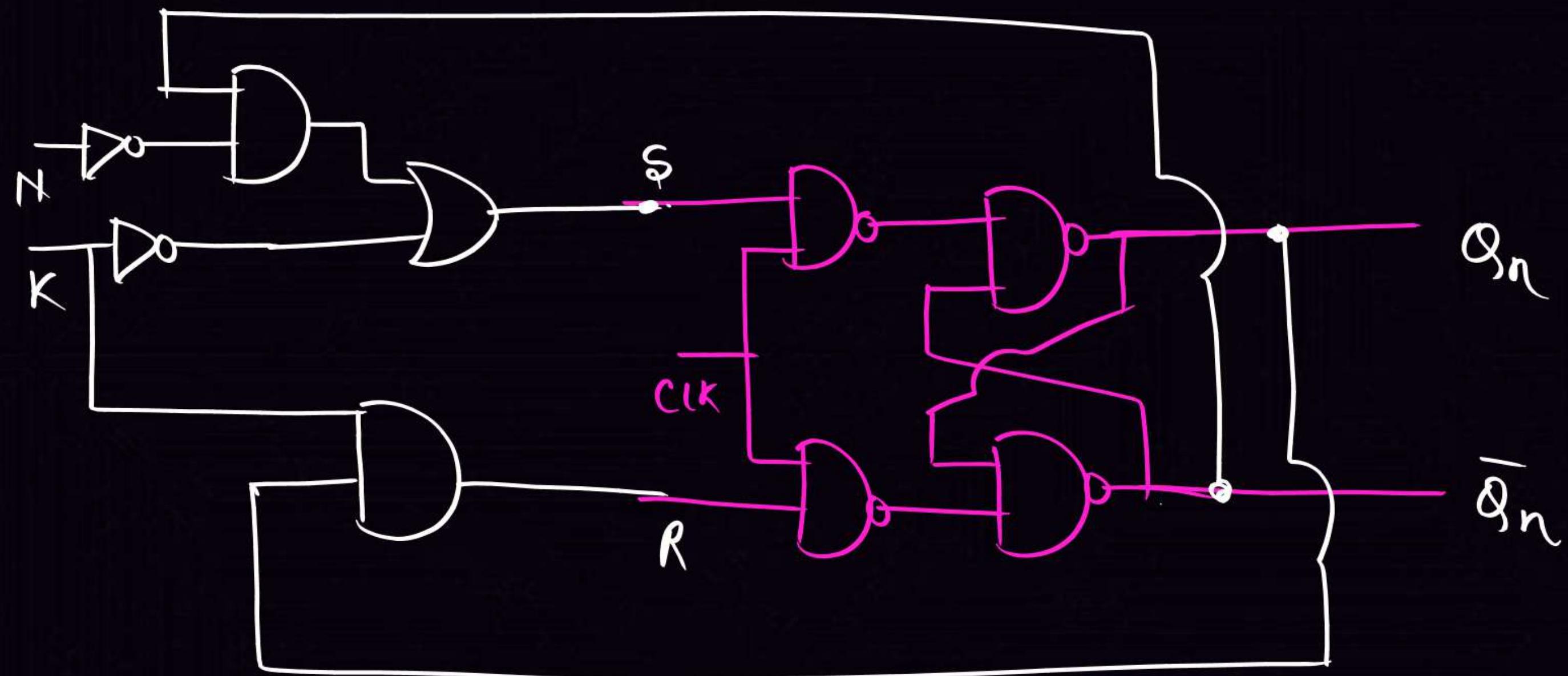
$$S = \bar{K} + \bar{N} \otimes n$$

R

$X \otimes n$

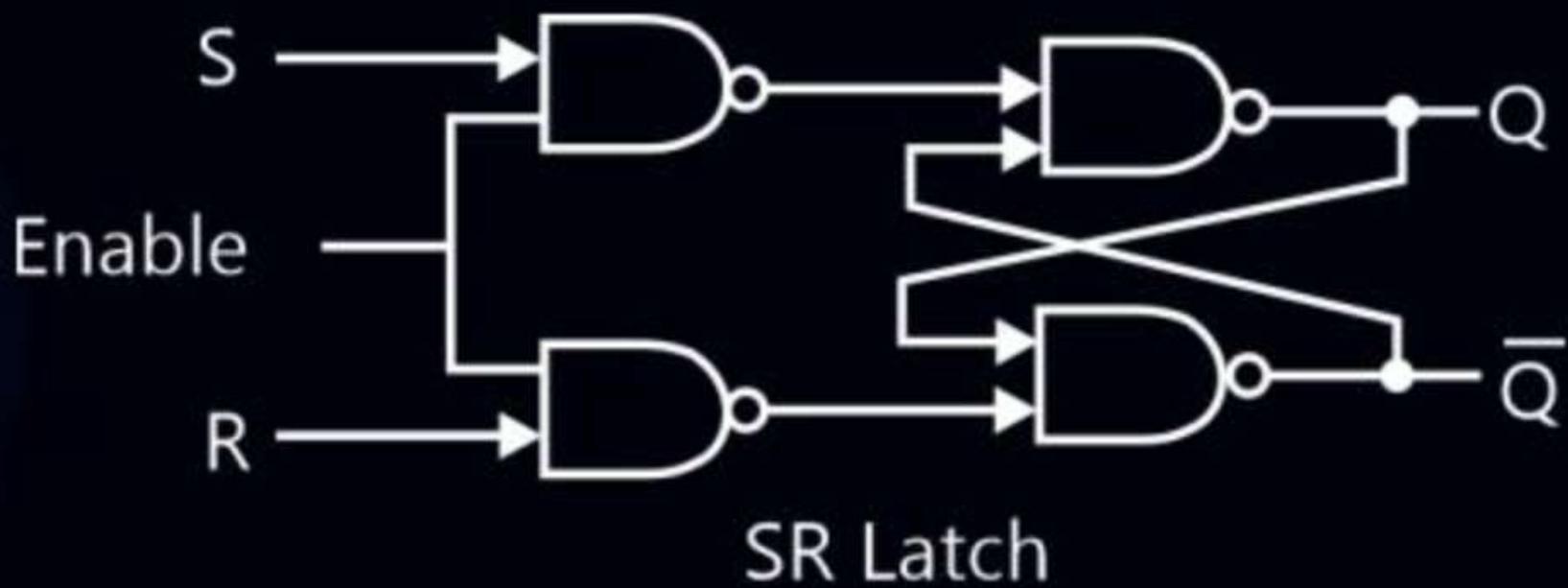
	00	01	11	10
0	\times	\times	\downarrow	
1			\downarrow	\times

$$R = K \otimes n$$



(1) SR Flip-Flop [Set Reset FF]

(i) Circuit Diagram :



(ii) Truth Table :

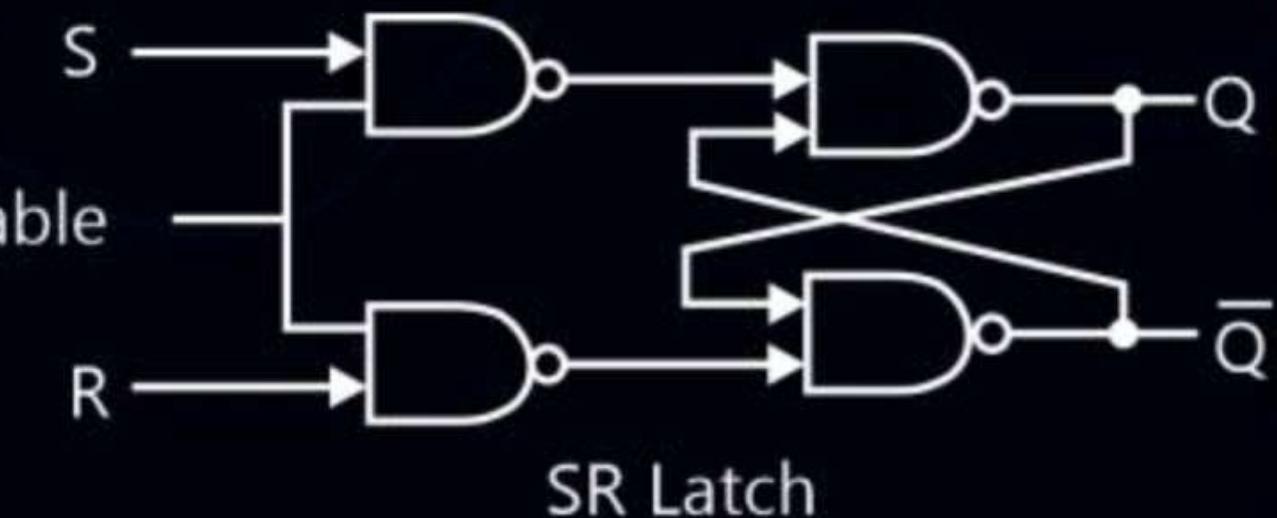


Figure 7: SR Flip-Flop

X	Y	Q	Q
0	0		
0	1		
1	0		
1	1		

Table 5: Truth Table of SR Flip-Flop

THANK
You! ☺

