

DIGITAL Revision Class

Syllabus

- 1) Number System
- 2) Boolean Algebra
- 3) Logic Gates
- 4) K-maps
- 5) Combinational Logic Circuits
- 6) Sequential Logic Circuit

Gnaneshwar

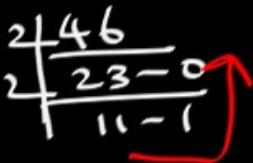
gnani412@gmail.com

15 years of Experience

Chapter 1 :-

Number System

Topic :- Number System Conversions:-

I) Integers →  → Division

II) Fractional
↓
multiplication



Number System Conversions for Fractional

Gnaneshwar



Decimal to Binary

$$(0.25)_{10} = (?)_2$$

fraction \times Required Radix = Result Integer

0.25	\times	2	=	0.5	0
0.5	\times	2	=	.10	1
0.0	\times	2	=	0	

Ans:⇒

$$(0.25)_{10} = (0.01)_2$$

$$\text{I) } \frac{10}{2} = 5$$

$$\text{II) } \frac{10}{3} = 3.\overline{3333\ldots} \approx 3.33$$

Q.) $(0.95)_{10} = (?)_2$

Decimal to Binary

Fraction	\times	Required Radix	= Result	Integer
0.95	\times	2	$= 1.9$	1
0.9	\times	2	$= 1.8$	1
0.8	\times	2	$= \underline{1.6}$	1
0.6	\times	2	$= 1.2$	1
0.2	\times	2	$= 0.4$	0
0.4	\times	2	$= 0.8$	0
0.8	\times	2	$= \underline{1.6}$	1

Fraction	\times	Required Radix	= Result	Integers
0.95	\times	2	= 1.9	1
0.9	\times	2	= 1.8	1
0.8	\times	2	= 1.6	1
0.6	\times	2	= 1.2	1
0.2	\times	2	= 0.4	0
0.4	\times	2	= 0.8	0
0.8	\times	2	= 1.6	1

Ans:- $(0.95)_{10} \approx (0.1111001)_2$



Binary to Decimal

Gnaneshwar



$$Q. (1011111.11)_2 = (\quad)_{10}$$

$$\begin{aligned} &= (\overset{2^6}{1} \overset{2^5}{0} \overset{2^4}{1} \overset{2^3}{1} \overset{2^2}{1} \overset{2^1}{1} \overset{2^0}{1} \cdot \overset{2^{-1}}{1} \overset{2^{-2}}{1})_2 \\ &= (1 \times 2^6) + (0 \times 2^5) + (1 \times 2^4) + (1 \times 2^3) + (1 \times 2^2) + (1 \times 2^1) + (1 \times 2^0) + (1 \times 2^{-1}) + (1 \times 2^{-2}) \\ &= (95.75)_{10} \end{aligned}$$



Binary to Octal Conversion

$$(\gamma = 2^2 = 4) \text{ to } (\gamma = 8 = 2^3)$$

one digit in octal = 3 bits in binary

E_x)

$$[10\ 110\ 011 + 101\ 1]_2 = [?]_8$$

Soln

Given = [0 1 0]
place extra bit ↓

place extra bit

110 011

← , →

• 101 100 → extra zeros

$$= \begin{bmatrix} 2 & 6 & 3 & . & 5 & 4 \end{bmatrix}_8$$

Binary to Hexadecimal Conversion

 $(\tau = 2 = 2^1)$ to $(\tau = 16 = 2^4)$

4 bits in Binary = 1 digit in Hexa decimal

Ex)

$$[11 \ 1100 \ 0101 \cdot 1011 \ 1]_2 = [?]_{16}$$

Given

$$\begin{array}{ccccccccc}
 & & & & \leftarrow & \cdot & \rightarrow & & \\
 & & & & & & & & \\
 & 0 & 0 & 1 & 1 & & 1 & 0 & 1 \\
 & \downarrow & & \downarrow & & & \downarrow & & \downarrow \\
 [& 3 & & C & & 5 & & B & 8]_{16} \\
 & & & & & & & & \\
 & & & & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\
 & & & & \circled{01} & \circled{10} & \circled{01} & \circled{10} & \circled{00} \\
 & & & & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\
 & & & & 1 & 1 & 0 & 0 & 0
 \end{array}$$



Octal to Hexadecimal Conversion

Procedure : →

Step 1) Convert Octal to Binary

Step 2) Convert Binary to Hexadecimal;

Previous Questions

Gnaneshwar





Previous Questions



Q. If $(11X1Y)_8 = (12C9)_{16}$ then the values of X and Y are

- ~~x(a) 9 and 1~~
- ~~x(b) 5 and 9~~
- ~~✓(c) 3 and 1~~
- ~~x(d) 1 and 8~~

Soln All digits < Radix \Rightarrow
 $X < 8$ and $Y < 8$

Multiple Select Question (MSQ)

Q. Which of the following is/are correct [B 9 F. A E]₁₆?

- (A) [1011 1010 1111. 1010 1110]₂
- (B.) [1011 1001 1111. 1010 1110]₂
- (C) [5637. 532]₈
- (D.) [5637. 534]₈

Given =

$$[B \ 9 \ F. A \ E]_{16} \rightarrow \text{one digit in Hex = 4 bits in Binary}$$

↓ ↓ ↓ ↓ ↓
 1011 1001 1111 0101 1110

Binary =

$$1011 \ 1001 \ 1111 \ 0101 \ 1110 \rightarrow \text{option (B)}$$

Binary \Rightarrow 1011 1001 1111 = 3 digits in binary

Binary \rightarrow

$$1011 \ 1001 \ 1111 \rightarrow 1 \text{ digit in octal}$$

↓ ↓ ↓
 1 0 1 1 1 0 0 1 1 1 1

Octal \rightarrow

$$5 \ 6 \ 3 \ 7 \ . \ 5 \ 3 \ 4 \rightarrow \text{option (D)}$$

Ans: Both B & D are right.



Previous Questions



Q. Consider the equation $(43)_X = (Y3)_8$. Where X and Y are unknown. The number of possible solution is _____

(CSE-GATE)

Soln

$$(\begin{smallmatrix} 4 & 3 \\ x & y \end{smallmatrix})_X = (\begin{smallmatrix} 8 & 3 \\ Y & z \end{smallmatrix})_8 \Rightarrow \boxed{x > 4 \text{ and } y < 8}$$

$$4x + 3 = 8y + z$$

$$4x = \cancel{8}y$$

$$\boxed{x = 2y}$$

$$y < 8 ; \quad x > 4$$

y	$x = 2y$	
1 ^v	2 ^x	X
2 ^v	4 ^x	X
3 ^v	6 ^v	✓
4 ^v	8 ^v	✓
5 ^v	10 ^v	✓
6 ^v	12 ^v	✓
7 ^v	14 ^v	✓
8 ^v		
9 ^{all as well invalid}		

number of valid
solutions are five

Ans: $\rightarrow 5$

Previous Questions



Q. Let the representation of a number in base 3 be 210.
What is the hexadecimal representation of the number?

(GATE-21-Set1)

- (a) 528
- (b) 21
- (c.) 15
- (d) D2

Soln Given $(_{210})_3 = (?)_{16}$

Step1) Convert Radix 3 to decimal

$$\left[\begin{smallmatrix} 3^2 & 3^1 & 3^0 \\ 2 & 1 & 0 \end{smallmatrix} \right]_3 = 2 \times 3^2 + 1 \times 3^1 + 0 \times 3^0 = (21)_{10}$$

Step2) Convert decimal to Hexa decimal

$$16 \Big| \begin{array}{r} 21 \\ -5 \end{array} \Rightarrow (21)_{10} = (15)_{16}$$

$$(_{210})_3 = (21)_{10} = ((5))_{16}$$

option (c) ✓



Previous Questions

Q. If x and y are two decimal digits and $(0.1101)_2 = (0.8xy5)_{10}$,
the decimal value of (x + y) is _____

(GATE-21-Set2)

Given $(0.1101)_2 = (0.8XY5)_{10}$

Soln Now Convert $(0.1101)_2$ to Decimal

$$\left[0. \overbrace{1 \ 1 \ 0 \ 1}^{\overset{\rightarrow}{2^{-1} \ 2^{-2} \ 2^{-3} \ 2^{-4}}} \right]_2 = \\ 1 \times 2^{-1} + 1 \times 2^{-2} + 0 \times 2^{-3} + 1 \times 2^{-4} = (0.8125)_{10}$$

$$[0.8 \overbrace{1 \ 2 \ 5}^{x \ y}]_{10} = (0.8XY5)_{10}$$

$$x=1, y=2 \Rightarrow x+y = 1+2=3$$

Ans! → 3

$$\theta) \quad \left[\begin{smallmatrix} 3 & 1 & 2 \\ \hline 2 & 0 \end{smallmatrix} \right]_{\gamma} = \left[\begin{smallmatrix} 1 & 3 & 1 \end{smallmatrix} \right]_{\gamma}$$

$$\left[\begin{smallmatrix} \gamma^2 & \gamma^1 & \gamma^0 \\ 3 & 1 & 2 \end{smallmatrix} \right]_{\gamma} = \left[\begin{smallmatrix} \gamma^1 & \gamma^0 \\ 2 & 0 \end{smallmatrix} \right]_{\gamma} \times \left[\begin{smallmatrix} \gamma^1 \gamma^0 & \gamma^1 \\ 1 & 1 \end{smallmatrix} \right]_{\gamma}$$

$$3\gamma^2 + \gamma + 2 = [2\gamma] \times [\gamma + 3 + 1/\gamma]$$

$$3\gamma^2 + \gamma + 2 = 2\gamma^2 + 6\gamma + \cancel{2}$$

$$\gamma^2 = 5$$

$$\gamma = 5$$



Previous Questions



Q. Which of the following represents ' $(E3)_{16}$ '

(GATE)

- (a) $(CE)_{16} + (A2)_{16}$
- (b) $(1BC)_{16} - (DE)_{16}$
- (c) $(2BC)_{16} - (1DE)_{16}$
- (d) $(200)_{16} - (11D)_{16}$

(a) $(CE)_{16} + (A2)_{16}$

$$\left\{ \begin{array}{l} A=10, \quad C=12, \quad E=14 \\ B=11, \quad D=13, \quad F=15 \end{array} \right.$$

C E

$$+ \begin{matrix} 0 & 1 & 0 \\ & \downarrow & \\ 1 & 7 & 0 \end{matrix} \begin{matrix} 2 \\ 1 \\ 0 \end{matrix}$$

$[170]_{16}$

$$E+2 = 14+2 = 16 = 16 + 0$$

$$C+A+1 = 12+10+1 = 23 = 16 + 7$$

$$(b) (1BC)_{}_{16} - (DE)_{16}$$

$$\left\{ \begin{array}{l} A=10, \quad C=12, \quad E=14 \\ B=11, \quad D=13, \quad F=15 \end{array} \right.$$

$\begin{matrix} & +15 & +16 \\ C & B & C \\ - & 0 & D \\ \hline [D & E]_{16} \end{matrix}$

$$16 + C - E = 16 + 12 - 14 = 14 = E$$

$$15 + B - D = 15 + 11 - 13 = 13 = D$$

(c) $(2BC)_{16} - (1DE)_{16}$

$$\left\{ \begin{array}{l} A=10, \quad C=12, \quad E=14 \\ B=11, \quad D=13, \quad F=15 \end{array} \right.$$

$$(d) (200)_{16} - (11D)_{16}$$

$$\left\{ \begin{array}{l} A=10, \quad C=12, \quad E=14 \\ B=11, \quad D=13, \quad F=15 \end{array} \right.$$

$$\begin{array}{r}
 & +15 & +16 \\
 \textcircled{1} & \textcircled{2} & \textcircled{0} & \textcircled{0} \\
 - & 1 & 1 & \textcircled{D} \\
 \hline
 & \textcolor{blue}{[E \quad 3]}_{16}
 \end{array}$$

$$16+0-D = 16-13 = 3$$

$$15+0-1 = 14 = E$$



Previous Questions



Q. Which of the following represents ' $(E3)_{16}$ '

(GATE)

- ~~a) $(CE)_{16} + (A2)_{16} = (170)_{16} \neq (E3)_{16}$~~
- ~~b) $(1BC)_{16} - (DE)_{16} = (D\bar{E})_{16} \neq (\bar{E}3)_{16}$~~
- ~~c) $(2BC)_{16} - (1DE)_{16} = (D\bar{E})_{16} \neq (\bar{E}3)_{16}$~~
- d) $(200)_{16} - (11D)_{16} = (\bar{E}3)_{16}$

Complements

Gnaneshwar



Binary (r = 2) Complements



1's Complements

Find 1's complement of $N = (10101)_2$

Ans 1's complement of $N = 01010$

2's Complements

10) Find 2's complement of $N = (10101)_2$

Soln Given number $\Rightarrow N = \underbrace{1\ 0\ 1\ 0\ 1}_{\text{logical Complement}} \quad |$
 \downarrow as it is

2's complement of $N = 0\ 1\ 0\ 1\ 1$

2's Complements

Q) Find 2's complement of $N = (10110)_2$

Soln

Given

$$N = (\underbrace{101}_{\text{logical complement}} \underbrace{10}_\text{write as it is})_2$$

2's Complement of N = 01010

2's Complements

3θ) Find 2's complement of $N = (1101000)_2$

Sol:

Given

$$N = (\underbrace{110}_{\text{logical complement}} \underbrace{1000}_\text{as it is})_2$$

$$2^1s \text{ complement of } N = 0011000$$

Signed Numbers

There are 3 methods to represent Signed Numbers

1) Sign Magnitude [SM] method

2) Signed 1's complement [S 1's] method

(3) Signed 2's complement [S 2's] method

In above three methods

If $MSB = 0 \Rightarrow$ It is positive number

If $MSB = 1 \Rightarrow$ It is negative number.

Sign Extension

for positive numbers Placing (or removing)
many zeros at MSB \Rightarrow no effect on result.

for negative numbers placing (or removing)
many 1's at MSB \Rightarrow no effect on result

Previous Questions

Q. In 16-bit 2's complement representation, the decimal number -28 is

(GATE-CSE 2019)

- (a) 1000 0000 1110 0100
- ~~(b) 0000 0000 1110 0100~~
- (c) 1111 1111 1110 0100
- (d) 1111 1111 0001 1100

$$(28)_{10} = (?)_2$$

2	28
2	14 - 0
2	7 - 0
2	3 - 1
2	- 1

$$(28)_{10} = 11100 \rightarrow \text{unsigned}$$

$$+ (28)_{10} = \underbrace{011100}_{\text{apply 2's complement}}$$

$$-(28)_{10} = 100100 \Rightarrow n=6 \text{ bits}$$

$$(-28)_{10} = 1111111100100 \Rightarrow n=16 \text{ bits}$$

option (C) ✓

Previous Questions

Q. In 16-bit 2's complement representation, the decimal number -28 is

(GATE-CSE 2019)

- (a) 1000 0000 1110 0100
- ~~(b) 0000 0000 1110 0100~~
- ~~(c) 1111 1111 1110 0100~~
- ~~(d) 1111 1111 0001 1100~~

$$\begin{aligned}
 &= -2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0 \\
 &= -32 + 0 + 0 + 4 + 0 + 0 = (-28)_{10} \\
 &= -2^7 + 2^6 + 2^5 + 2^4 + 2^3 + 2^2 + 2^1 + 2^0 \neq (-28)_{10}
 \end{aligned}$$



Previous Questions

Gnaneshwar



Q. 9's complement of the number $[36800]_9$ is

- (a) 63199
- (b) 63100
- (c) 52100
- (d) None

$$N = [3\ 6\ 8\ 0\ 0]_9$$

for 9's complement write zero LSB bits as it is
 subtract if non zero digit from " ∞ " &
 remaining digits from $(\infty - 1)$

Given $N =$

$$\begin{array}{r}
 8 \quad 8 \quad 9 \\
 3 \quad 6 \quad 8 \\
 \hline
 0 \quad 0
 \end{array}$$

↓ write as it is

9's complement of $N = [5 \quad 2 \quad 1 \quad 0 \quad 0]_9$

Previous Questions

Q. The range of signed decimal numbers that can be represented by 6-bit 1's complement number is

(GATE)

- (A) -31 to + 31
- (B) -63 to + 63
- (C) -64 to + 63
- (D) -32 to + 31

The Range for n-bit number is

$$S-M \Rightarrow -\left(\frac{2^{n-1}-1}{2}\right) \text{ to } +\left(\frac{2^{n-1}-1}{2}\right) \quad \left. \begin{array}{l} \\ \end{array} \right\} \text{Both are same}$$

$$S-1's \Rightarrow -\left(\frac{2^{n-1}-1}{2}\right) \text{ to } +\left(\frac{2^{n-1}-1}{2}\right)$$

$$S-2^1s \Rightarrow -\left(\frac{2^{n-1}}{2}\right) \text{ to } +\left(\frac{2^{n-1}}{2}\right)$$

Q. The range of signed decimal numbers that can be represented by 6-bit 1's complement number is

- (A) -31 to +31
- (B) -63 to +63
- (C) -64 to +63
- (D) -32 to +31

formula:-

$$S-1 \Rightarrow -(2^{n-1}-1) \text{ to } +(2^{n-1}-1), \text{ Here } n=6$$

$$-(2^{6-1}-1) \text{ to } +(2^{6-1}-1)$$

$$-31 \text{ to } +31$$

∴

option (A) ✓



Multiple Select Question (MSQ)

- Q. Which of the following is/are the correct signed 2's complement representation of -14?
- (A) 10011
 - (B.) 10010
 - (C) 10001100
 - (D.) 11110010

(A) $\begin{array}{r} -2^4 \\ \hline 1 \end{array} \quad \begin{array}{r} 2^3 \\ 0 \end{array} \quad \begin{array}{r} 2^2 \\ 0 \end{array} \quad \begin{array}{r} 2^1 \\ 1 \end{array} \quad \begin{array}{r} 2^0 \\ 1 \end{array} = -16 + 0 + 0 + 2 + 1 = (-13)_{10} \neq (-4)_{10}$

(B) $\begin{array}{r} -2^4 \\ \hline 1 \end{array} \quad \begin{array}{r} 2^3 \\ 0 \end{array} \quad \begin{array}{r} 2^2 \\ 0 \end{array} \quad \begin{array}{r} 2^1 \\ 1 \end{array} \quad \begin{array}{r} 2^0 \\ 0 \end{array} = -16 + 0 + 0 + 2 = (-14)_{10}$

(C) $\begin{array}{r} -2^7 \\ \hline 1 \end{array} \quad \begin{array}{r} 2^6 \\ 0 \end{array} \quad \begin{array}{r} 2^5 \\ 0 \end{array} \quad \begin{array}{r} 2^4 \\ 0 \end{array} \quad \begin{array}{r} 2^3 \\ 1 \end{array} \quad \begin{array}{r} 2^2 \\ 1 \end{array} \quad \begin{array}{r} 2^1 \\ 0 \end{array} \quad \begin{array}{r} 2^0 \\ 0 \end{array} \neq (-14)_{10}$

d) $\begin{array}{r} -1 \ 1 \ 1 \ 1 \ 0 \ 0 \ 1 \ 0 \end{array} = -(14)_{10} \checkmark$

Ans: options (B) & (D) are correct

Binary Codes

Gnaneshwar



- 1) Weighted Codes
- 2) Non - Weighted Codes



Weighted Codes

BCD (Binary Coded Decimal) Code

- 1.BCD is 4 bit weighted code.
- 2.BCD is Sequential code.

Decimal	8 4 2 1 (or BCD)				Decimal	3 3 2 1				Decimal	7 4 -2 -1			
	0	0	0	0		0	0	0	0		0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	0	0	0	1	1	0	0	0	1	1	0	1	1	
2	0	0	1	0	2	0	0	1	0	2	0	1	1	
3	0	0	1	1	3	0	0	1	1	3	0	1	0	
4	0	1	0	0	4	0	1	0	1	4	0	1	0	
5	0	1	0	1	5	1	0	1	0	5	1	0	1	
6	0	1	1	0	6	1	1	0	0	6	1	0	0	
7	0	1	1	1	7	1	1	0	1	7	1	0	0	
8	1	0	0	0	8	1	1	1	0	8	1	1	1	
9	1	0	0	1	9	1	1	1	1	9	1	1	1	



Self Complement Code

A code and its 9's complement are logical complement to each other , then that code is called as self complement code

Decimal	8 4 2 1 (or BCD)				Decimal	3 3 2 1				Decimal	7 4 -2 -1			
	0	0	0	0		0	0	0	0		0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	0	0	0	1	1	0	0	0	1	1	0	1	1	
2	0	0	1	0	2	0	0	1	0	2	0	1	1	
3	0	0	1	1	3	0	0	1	1	3	0	1	0	
4	0	1	0	0	4	0	1	0	1	4	0	1	0	
5	0	1	0	1	5	1	0	1	0	5	1	0	1	
6	0	1	1	0	6	1	1	0	0	6	1	0	0	
7	0	1	1	1	7	1	1	0	1	7	1	0	0	
8	1	0	0	0	8	1	1	1	0	8	1	1	1	
9	1	0	0	1	9	1	1	1	1	9	1	1	1	

8421 is not self complementing, sequential code

3321 is self complementing, but not sequential code

74–2–1 is neither a self complementary, nor a sequential code



Self Complement Code

A code and its 9's complement are logical complement to each other , then that code is called as self complement code

If sum of weights equal to 9 then that code is called as self complement code

Non - Weighted Codes

Gnaneshwar



Excess – 3 Code

Gnaneshwar





Gray Code

I) Binary to Gray code Conversion

II) Gray Code to Binary Conversion

I) Binary to Gray code Conversion

Q) Convert given Binary (1011)₂ to Gray code

II) Gray Code to Binary Conversion

Q) Convert given Gray code (1110) to Binary

Gray Code

Gnaneshwar



1. Gray Code is Unit distance code
2. Gray Code is called as Reflective code

Chapter 2

BOOLEAN ALGEBRA

- 
- *Introduction*
 - *Laws , Postulates*
 - *Duality , Demorgan's Theorems*
 - *Self Dual function*
 - *Canonical forms*
 - *Previous Questions*

**WORK HARD
IN SILENCE
LET SUCCESS
MAKE THE
NOISE**

Chapter 2

BOOLEAN ALGEBRA

Purpose:

By using Boolean algebra complicated Boolean functions can be converted to simple form

Advantages:

1. Reduce hardware complexity
2. Number of gates required is less
3. less cost
4. power consumption is less

Postulates in Boolean Algebra:

1. $A + 0 = A$
2. $A + 1 = 1$
3. $A \cdot 0 = 0$
4. $A \cdot 1 = A$
5. $A + A = A$
6. $A \cdot A = A$
7. $A + \bar{A} = 1$
8. $A \cdot \bar{A} = 0$
9. $(\bar{\bar{A}}) = A$
10. $A + AB = A$
11. $(A + B)(A + C) = A + BC$
12. $A + \bar{A}B = A + B$
13. $AB + \bar{A}C = (A + C)(\bar{A} + B)$

(11) ~~$(A+B)(A+C) = A + B \cdot C$~~
$$\begin{matrix} (A+B)(A+C) \\ (1+2)(1+3) \end{matrix} = \begin{matrix} A + B \cdot C \\ 1 + 2 \cdot 3 \end{matrix}$$

(12) ~~$A + \bar{A}B = A + B$~~
$$\begin{matrix} A + \bar{A}B \\ 1 + \bar{1} \cdot 2 \end{matrix} = \begin{matrix} A + B \\ 1 + 2 \end{matrix}$$

Practice Questions

. Simplify the Boolean function $f = (A + \bar{B})(\bar{A} + \bar{B} + D)(\bar{B} + C + \bar{D})$

Soln

Given $f = [\bar{B} + A] [\bar{B} + (\bar{A} + D)] (\bar{B} + C + \bar{D})$
use 11th $(1 + 2) \cdot (1 + 3) = 1 + 2 \times 3$

$f = [\bar{B} + (A + D)] [\bar{B} + (C + \bar{D})]$
use 11th $(1 + 2) \cdot (1 + 3) = 1 + 2 \times 3$

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

Duality Theorem

$$1) \quad 1 \iff 0$$

$$2) \quad \text{AND} \iff \text{OR}$$
$$(\cdot) \iff (+)$$

Ex1) $f = [B] + [\bar{A} \cdot C]$

Duality $= f_D = [B] \cdot [\bar{A} + C]$

Ex2) $\bar{A} + 1 = 1$

Apply duality

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

NOTE: In duality variables are not going to effect.

Self Dual function

If $f_D = f$ then “f” is called as Self Dual

Previous Questions

Q. By using 2-variables, number of possible Boolean functions

- (a) 2
- (b) 4
- (c) 8
- (d) 16

for n -variables, no. of possible Boolean functions = 2^{2^n}
Here $n=2 \Rightarrow$ so Ans = $\frac{2^2}{2} = 2^4 = 16$

option (D) ✓

Previous Questions

Q. By using 2-variables, number of possible Self Dual Boolean functions

- (a) 2
- (b) 4
- (c) 8
- (d) None

Soln

By using 2-variables, no of possible Self dual functions = $\frac{n-1}{2}$

Here $n=2 \Rightarrow \frac{2-1}{2} = \frac{1}{2} = 2^2 = 4$

option (B) ✓

Conclusions Table

Variable	No. of Boolean functions	No. of Self Dual
2	16	4
3	256	16
.	.	.
n	2^{2^n}	$2^{2^{n-1}}$

Possible Boolean functions

n	2^{2^n}
S.D.F	$2^{2^{n-1}}$
N.S.D.F	$2^{2^n} - 2^{2^{n-1}}$

De Morgan's Theorems

$$① \quad \overline{A + B} = \overline{A} \cdot \overline{B}$$

$$② \quad \overline{A \cdot B} = \overline{A} + \overline{B}$$

Practice Questions

Simplify the following Boolean function

$$f = \underbrace{B + AD + BC}_{\text{Simplification}} + \overline{B + A[C + D]}$$

Soln

$$f = B \underbrace{[1 + C]}_1 + AD + \overline{(B + AC + AD)} \quad \left\{ \because \overline{x+y} = \overline{x} \cdot \overline{y} \right.$$

$$f = (B + AD) + \underbrace{\overline{(B+AD)}}_{1} + \overline{AC} \quad \left\{ \because \overline{x+y} = \overline{x} \cdot \overline{y} \right.$$

$$f = (B + AD) + \overline{(B+AD)} \cdot \overline{(AC)} \quad \left\{ \begin{array}{l} \text{use } (12)^{\text{th}} \text{ formula} \\ \overline{x+\overline{x} \cdot y} = x+y \\ 1 + \overline{1} \cdot 2 = 1+2 \end{array} \right.$$

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

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

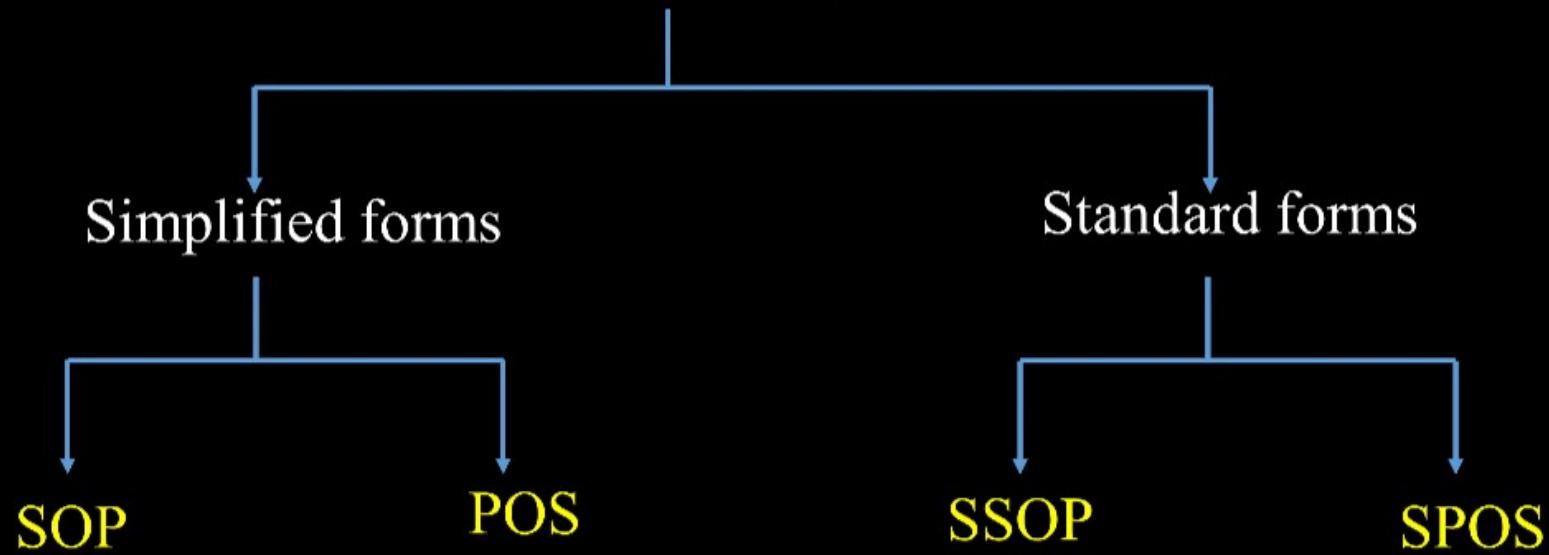
$$f = [\overline{A} + (\overline{A}) \cdot D] + B + \overline{C}$$

(12)th
 $x + \overline{x} \cdot y = x + y$

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

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

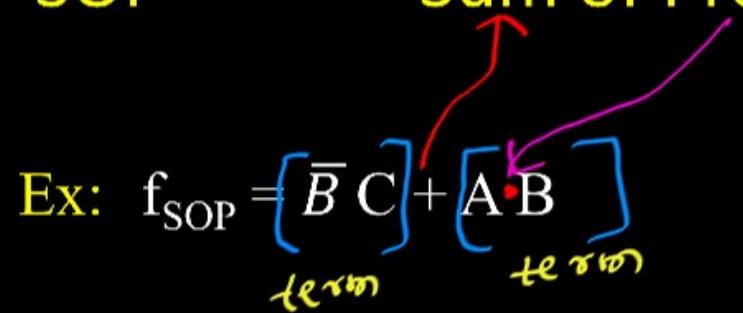
Boolean function forms



SOP → Sum of Product

Ex: $f_{SOP} = [\bar{B} \cdot C] + [A \cdot \bar{B}]$

term *term*



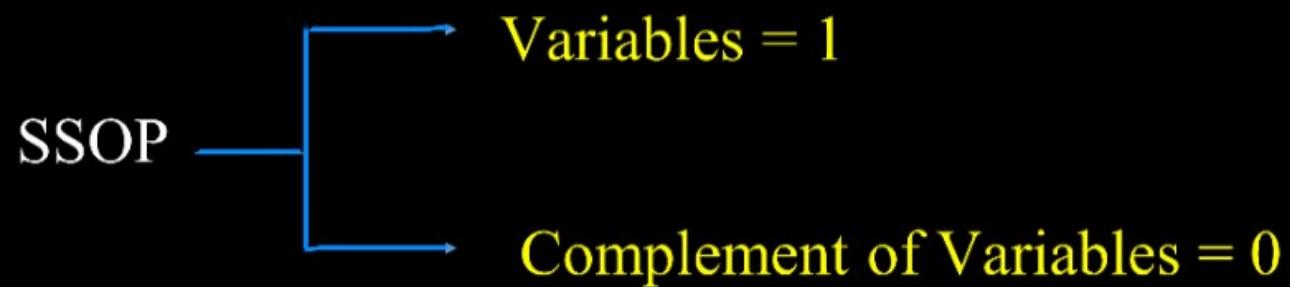
SSOP → Standard Sum of Product

Convert given SOP to SSOP

$$f_{SOP} = \bar{B} C + A B$$

Each term in SSOP → min term "m"

SSOP Summary



POS → Product of Sum

$$\text{Ex : } f_{\text{POS}} = (x + y)(\bar{y} + z)$$

term *term*

SOP \longrightarrow Sum of Product

$$\text{Ex: } f_{\text{SOP}} = [\bar{B} C] + [A B]$$

POS \longrightarrow Product of Sum

$$\text{Ex : } f_{\text{POS}} = (x + y) \cdot (\bar{y} + z)$$

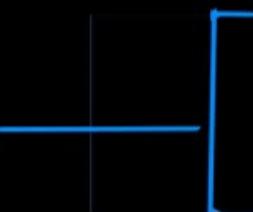
SPOS → Standard Product of Sum

Convert given POS to SPOS

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

each term in SPOS is called as
Max term → "M"

SPOS Summary

SPOS  [Variables = 0
 Complement of Variables = 1]

Comparison

SSOP Summary

SSOP 
Variables = 1
Complement of Variables = 0

SPOS Summary

SPOS 
Variables = 0
Complement of Variables = 1

Previous Questions

Q. The simplified SOP (Sum of Product) form of the Boolean expression
 $(P + \bar{Q} + \bar{R}) \cdot (P + \bar{Q} + R) \cdot (P + Q + \bar{R})$ is _____
(GATE)

- (a) $(\bar{P} \cdot Q + \bar{R})$
- (b.) $(P + \bar{Q} \cdot \bar{R})$
- (c) $(\bar{P} \cdot Q + R)$
- (d) $(P \cdot Q + R)$

Given $f = \left[\underbrace{(P + \bar{Q} + \bar{R})}_{(A + B)}, \underbrace{(P + \bar{Q} + R)}_{(A + C)} \right] (P + Q + \bar{R})$

use (ii)th $(A + B) \cdot (A + C) = A + BC$

$$f = \left[P + \overline{Q} \right] \left[(P) + \underbrace{\cancel{Q} + \bar{R}}_{(A + C)} \right]$$

(ii)th $(A + B) \cdot (A + C) = A + BC$

$$f = P + \overline{Q} \bar{R}$$

Q. The simplified form of the following

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

Soln

$$\text{Let } \alpha = A\bar{B}\bar{C} + C\bar{D}$$

$$\text{Now } \bar{\alpha} = \overline{[A\bar{B}\bar{C} + C\bar{D}]} = \overline{\alpha} \cdot \bar{y}$$

$$\bar{\alpha} = \overline{[A \cdot \bar{B} \cdot \bar{C}]} \cdot \overline{[C \cdot \bar{D}]}$$

$$\bar{\alpha} = [\bar{A} + \bar{\bar{B}} + \bar{\bar{C}}] \cdot [\bar{\bar{C}} + \bar{\bar{D}}]$$

$$\bar{\alpha} = [\bar{A} + B + C] \cdot (\bar{C} + \bar{D})$$

$$f = \alpha + \bar{\alpha} = 1$$

Ans: $\rightarrow 1$

Previous Questions

Q. The Boolean expression $xy + (\bar{x} + \bar{y})z$ is equivalent to

- (a) $xyz + \bar{x}\bar{y}z$
- (b) $\bar{x}\bar{y}\bar{z} + xyz$
- (c.) $(x+z)(y+z)$
- (d) None

Soln

is

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

$$\begin{aligned} f &= \cancel{xy} + \cancel{\bar{x}\bar{y}} \cdot z \\ \text{12th} &\quad A + \bar{A} \cdot B = A + B \end{aligned}$$

$$\begin{aligned} f &= \cancel{xy} + z \\ \text{11th} &\quad 1 = (1+2)(1+3) \end{aligned}$$

$$f = (z+x)(z+y)$$

$$f = (x+z)(y+z)$$

option: (c) ✓



Chapter 3

LOGIC GATES

-  *Introduction*
-  *Ex-OR Gate Properties*
-  *Universal Logic GATEs*
-  *Logic function Design using NAND*
-  *Logic function Design using NOR*
-  *Previous Questions*



Study while

Others are sleeping

Work while

Others are loafing

Prepare while

Others are playing &

Dream while

Others are wishing

Chapter 3

Logic Gates

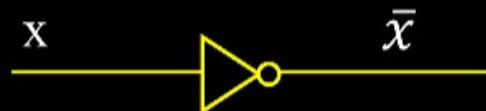
Gnaneshwar



Logic Gates are Basic Building Block of Digital System

NOT Gate (Inverter)

Symbol



Truth Table

Input (x)	Output ($Y = \bar{x}$)
0	1
1	0

AND Gate

Symbol

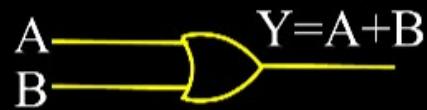


Truth Table

Inputs		Output
A	B	$Y = A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

OR Gate

Symbol



Truth Table

Inputs		Output
A	B	$Y = A + B$
0	0	0
0	1	1
1	0	1
1	1	1



Note

NOT , AND , OR Gates are called as Basic Gates

AOI means AND , OR , Inverter

NAND Gate

Symbol

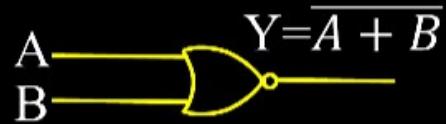


Truth Table

Inputs		Output
A	B	$Y = \overline{AB}$
0	0	1
0	1	1
1	0	1
1	1	0

NOR Gate

Symbol



Truth Table

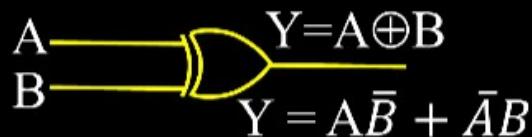
Inputs		Output
A	B	$Y = \overline{A + B}$
0	0	1
0	1	0
1	0	0
1	1	0

Exclusive-OR or Ex-OR or XOR Gate

Gnaneshwar



Symbol



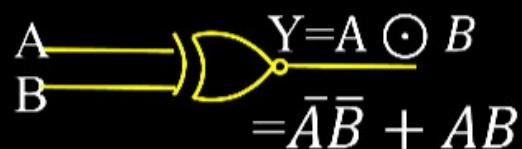
Truth Table

Inputs		Output
A	B	Y = A \oplus B
0	0	0
0	1	1
1	0	1
1	1	0

NOTE:- This XOR Gate is called as Inequality detector
or Anti coincidence detector.

Ex-NOR or XNOR Gate

Symbol



Truth Table

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

NOTE

2 input XNOR gate is called as
Equality Detector (or) coincidence detector

Properties of Ex-OR Gate:

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

1. $A \oplus 0 = A\bar{0} + \bar{A}0 = A(1) + 0 = A$
2. $A \oplus 1 = A\bar{1} + \bar{A}1 = A(0) + \bar{A}.1 = \bar{A}$
3. $A \oplus A = A\bar{A} + \bar{A}A = 0$
4. $A \oplus \bar{A} = A\bar{\bar{A}} + \bar{A}\bar{A} = AA + \bar{A}\bar{A} = A + \bar{A} = 1$
5. $A(B \oplus C) = AB \oplus AC$
6. $(A + B) \oplus C \neq [A \oplus C] + [B \oplus C]$

Relation Between XOR , XNOR

for odd number of variable $\Rightarrow \text{XOR} = \text{XNOR} \Rightarrow$ equal relation

$$\text{Ex} \quad A \oplus B \oplus C = A \odot B \odot C$$

for Even number of variables $\Rightarrow \overline{\text{XOR}} = \text{XNOR} \Rightarrow$ complement Relation

$$\text{Ex} \quad \overline{A \oplus B} = A \odot B$$



Universal Logic Gates

NAND , NOR Logic Gates are called as Universal Logic Gates

Reason :

Any logic Gate and Any logic function can be designed

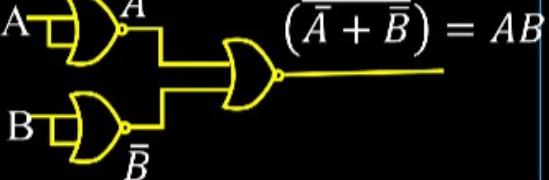
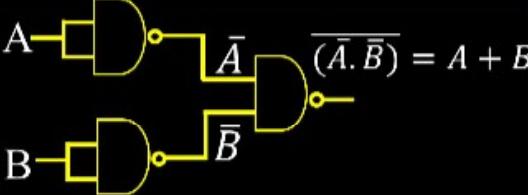
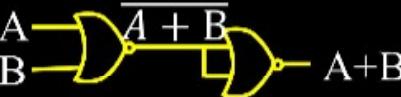
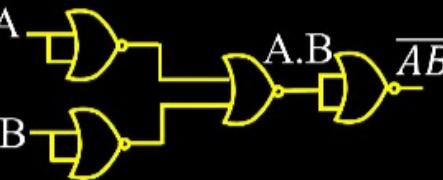
Design (or) implement all logic gates using NAND, NOR Gates

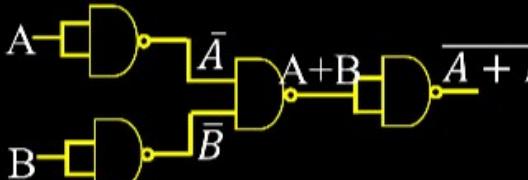
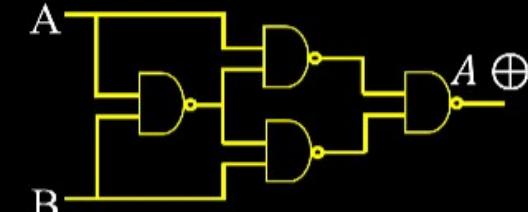
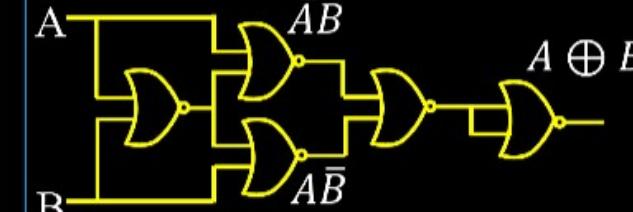
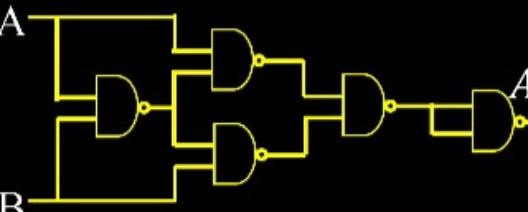
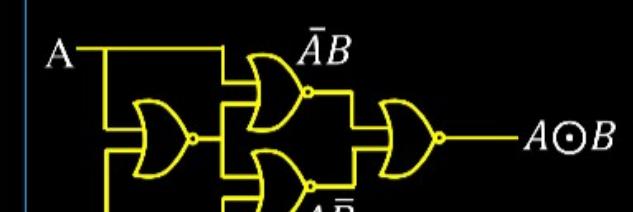
Gnaneshwar



Basic gates	No. of NAND Gate	No. of NOR Gate
NOT	1	1
AND	2	3
OR	3	2
NAND	1	4
NOR	4	1
XOR	4	5
XNOR	5	4

Design (or) implement all logic gates using NAND, NOR Gates

Gate	Basic symbol	Using NAND gate	No. of gates required	Using NOR gate	No. of Gates required
NOT			1	 $(\overline{A + A}) = \bar{A}$	1
AND		 $A \cdot B = \overline{\overline{AB}} = \overline{AB}$	2	 $\overline{A} \rightarrow \bar{A}$ $\overline{B} \rightarrow \bar{B}$ $(\bar{A} + \bar{B}) = AB$	3
OR		 $A + B = \overline{(\bar{A} \cdot \bar{B})} = \overline{\bar{A} \cdot \bar{B}}$	3	 $\overline{A + B} \rightarrow \overline{A} + \overline{B}$ $\overline{A} + \overline{B} \rightarrow A + B$	2
NAND			1	 $\overline{A} \rightarrow \bar{A}$ $\overline{B} \rightarrow \bar{B}$ $\overline{A} \cdot \overline{B} \rightarrow A \cdot B$ $A \cdot B \rightarrow \overline{AB}$	4

Gate	Basic symbol	Using NAND gate	No.of gates Required	Using NOR gate	No.of Gates Required
NOR			4		1
XOR			4		5
XNOR			5		4

Design (or) implement all logic gates using NAND, NOR Gates

Gnaneshwar



Basic gates	No. of NAND Gate	No. of NOR Gate
NOT	1	1
AND	2	3
OR	3	2
NAND	1	4
NOR	4	1
XOR	4	5
XNOR	5	4

Gnaneshwar



Q. functions f_1 , f_2 and f_3 , which are expressed in sum of min terms
 Consider three 4 variable as

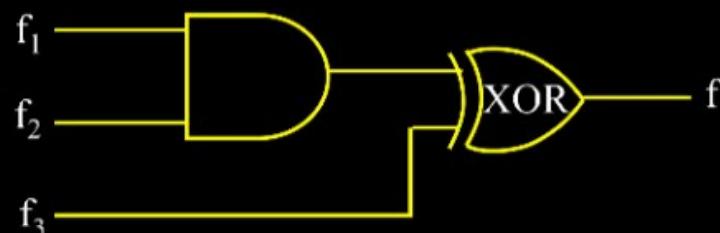
$$f_1 = \Sigma (0, 2, 5, 8, 14)$$

$$f_2 = \Sigma (2, 3, 6, 8, 14, 15)$$

$$f_3 = \Sigma (2, 7, 11, 14)$$

For the following circuit with one AND gate and one XOR gate, the output function f can be expressed as:

(GATE - 2019)



- (a) $\Sigma (2, 14)$
- (b) $\Sigma (2, 7, 8, 11, 14)$
- (c) $\Sigma (0, 2, 3, 5, 6, 7, 8, 11, 14, 15)$
- (d.) $\Sigma (7, 8, 11)$

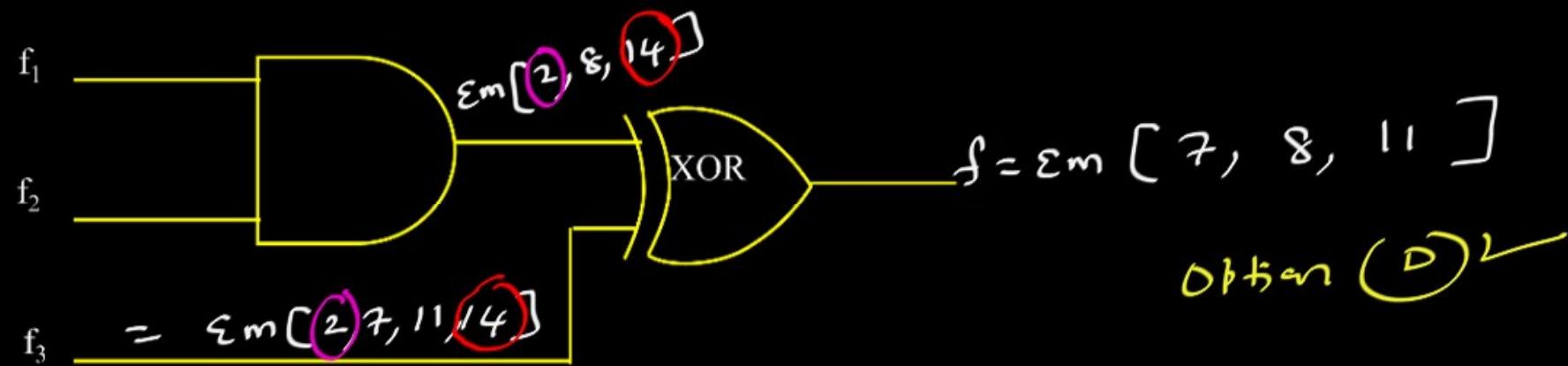
$$f_1 = \Sigma (0, 2, 5, 8, 14)$$

$$f_2 = \Sigma (2, 3, 6, 8, 14, 15)$$

$$f_3 = \Sigma (2, 7, 11, 14)$$

Inputs		AND Gate	XOR
A	B		
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

Gnaneshwar



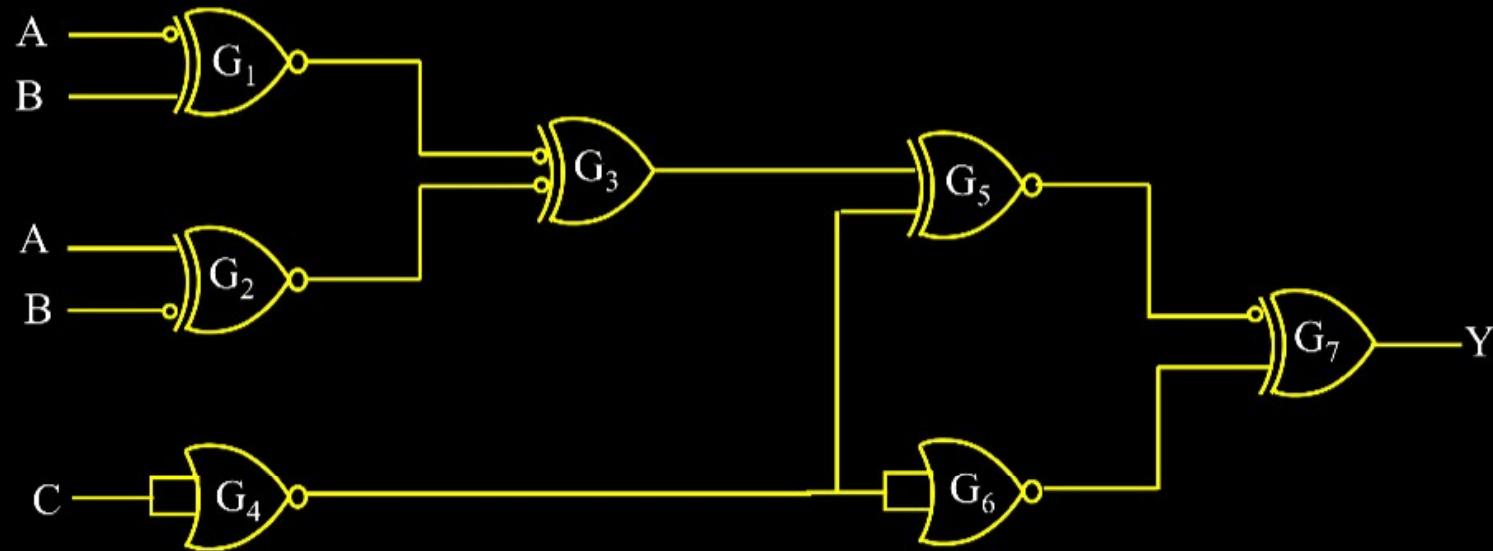
Previous Questions

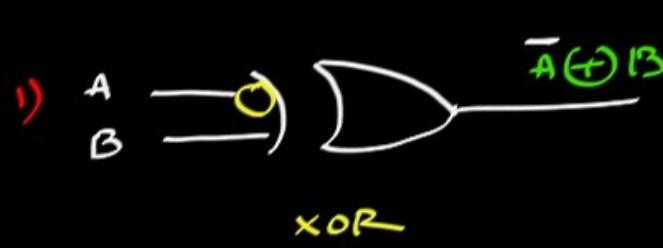
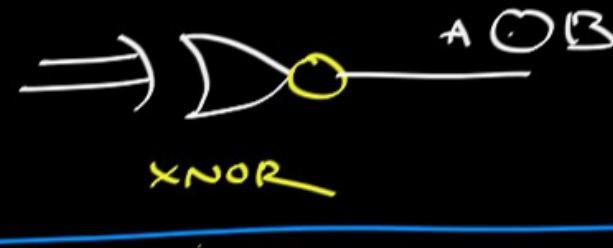
Gnaneshwar



ACE

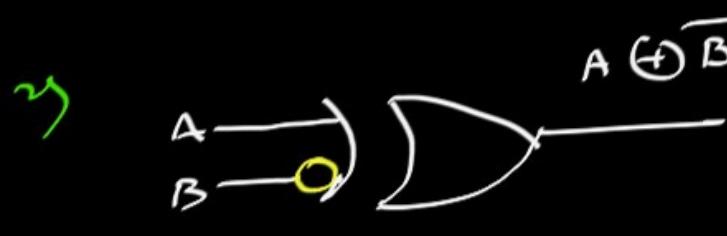
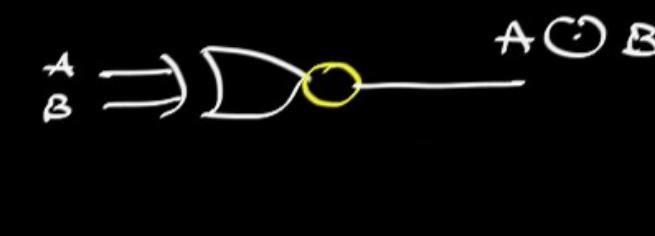
Q. The output (Y) of the following circuit is



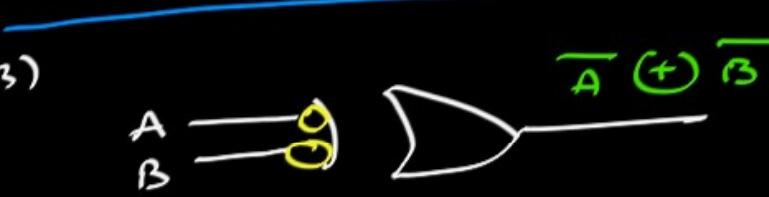
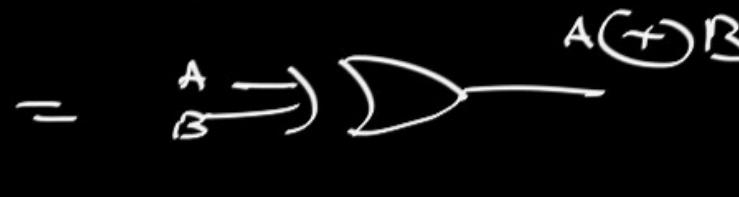
1)  = 

$A \oplus B$

xOR

2)  = 

$A \oplus \overline{B}$

3)  = 

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

$A \oplus B$

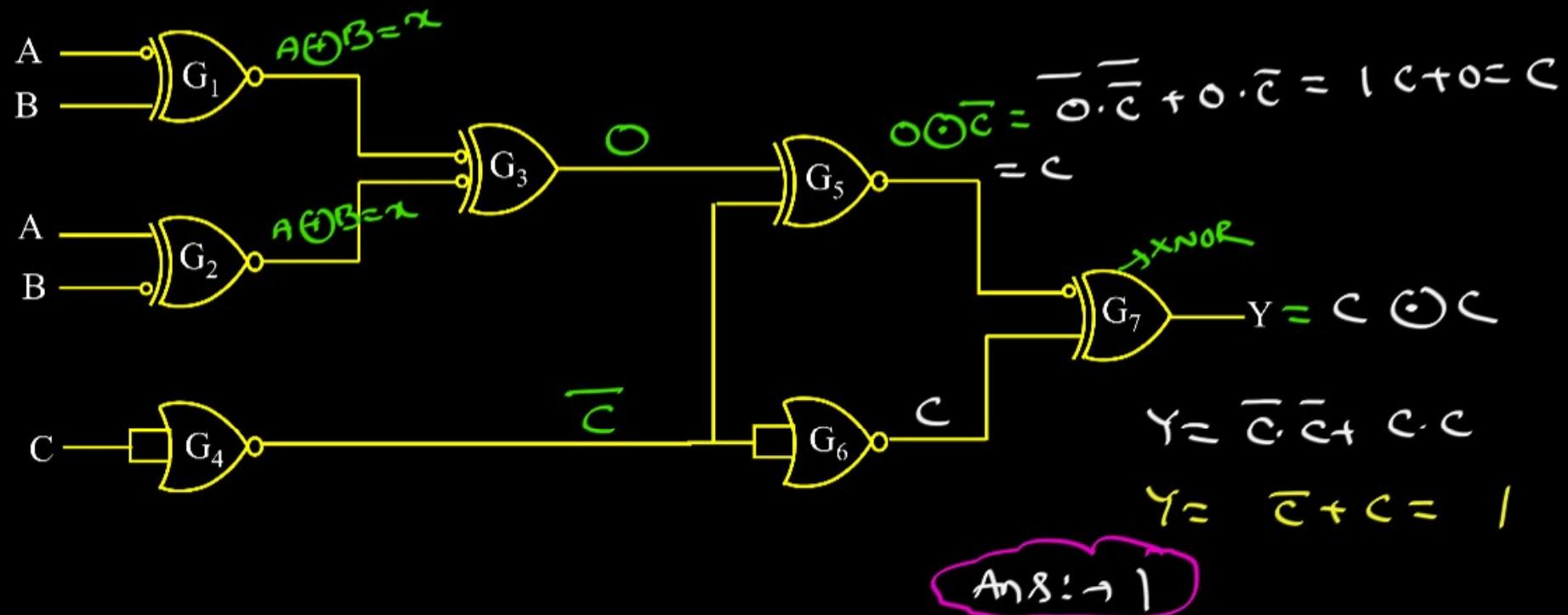
Previous Questions

Gnaneshwar



ACE

Q. The output (Y) of the following circuit is

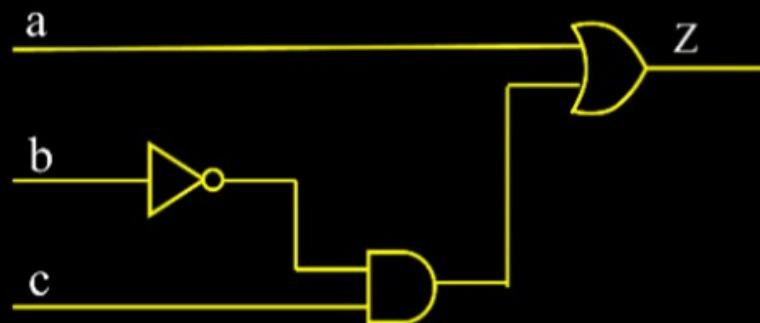


Previous Questions

Gnaneshwar



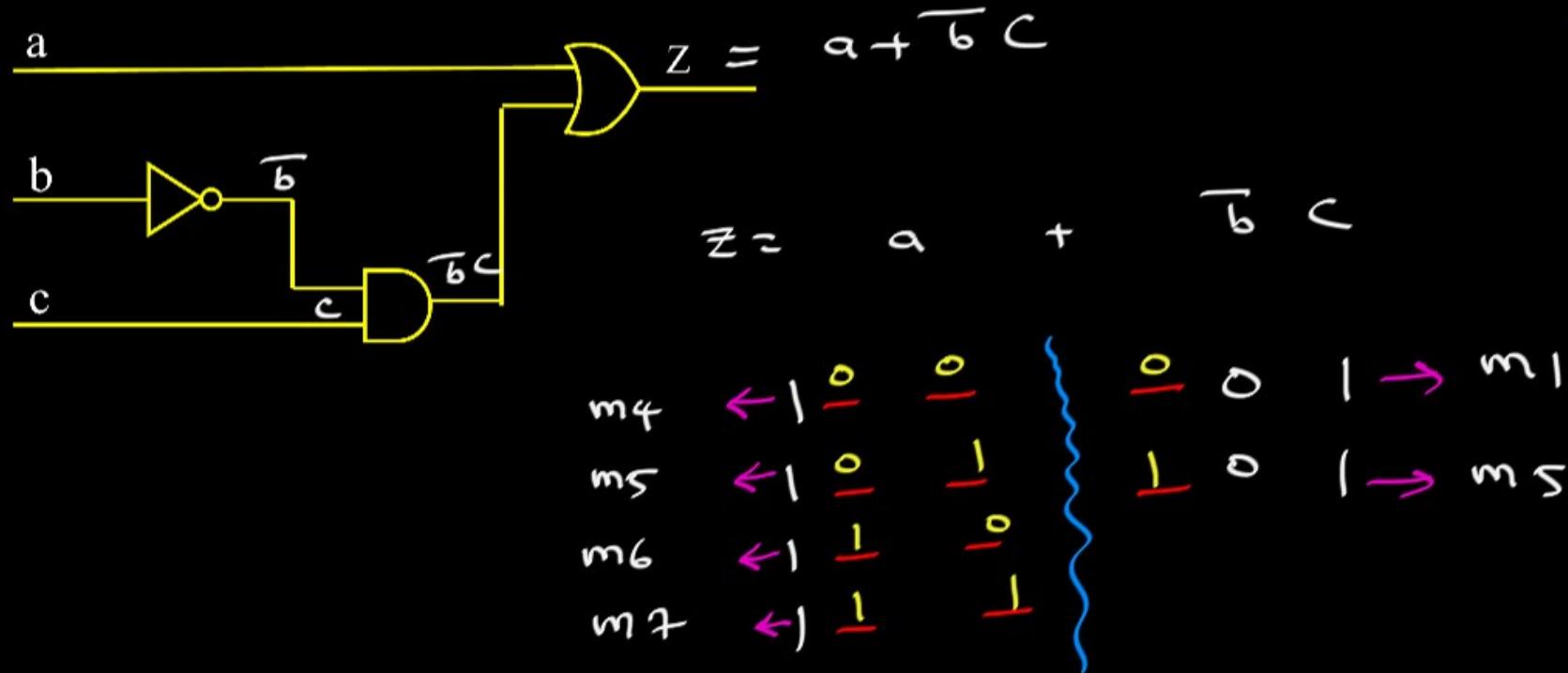
Q. Consider the Boolean function $z(a, b, c)$.



Which of the following min term lists represents the circuit given above?

(GATE – 2020)

- (a) $z = \Sigma(2, 4, 5, 6, 7)$
- (b.) $z = \Sigma(1, 4, 5, 6, 7)$
- (c) $z = \Sigma(0, 1, 3, 7)$
- (d) $z = \Sigma(2, 3, 5)$



$$f = \sum m[1, 4, 5, 6, 7] \rightarrow \text{option (B)}$$



Chapter 4

K-MAPS

- Introduction
- K-map for SSOP
- K-map for SPOS
- K-map with Don't cares
- Prime Implicants
- Previous Questions

**MORE YOU SWEAT
IN TIMES OF PEACE**

**LESS YOU BLEED
IN TIMES OF WAR**

**MORE YOU SWEAT
IN TIMES OF PEACE**

**LESS YOU BLEED
IN TIMES OF WAR**

K-Maps (Karnaugh map)

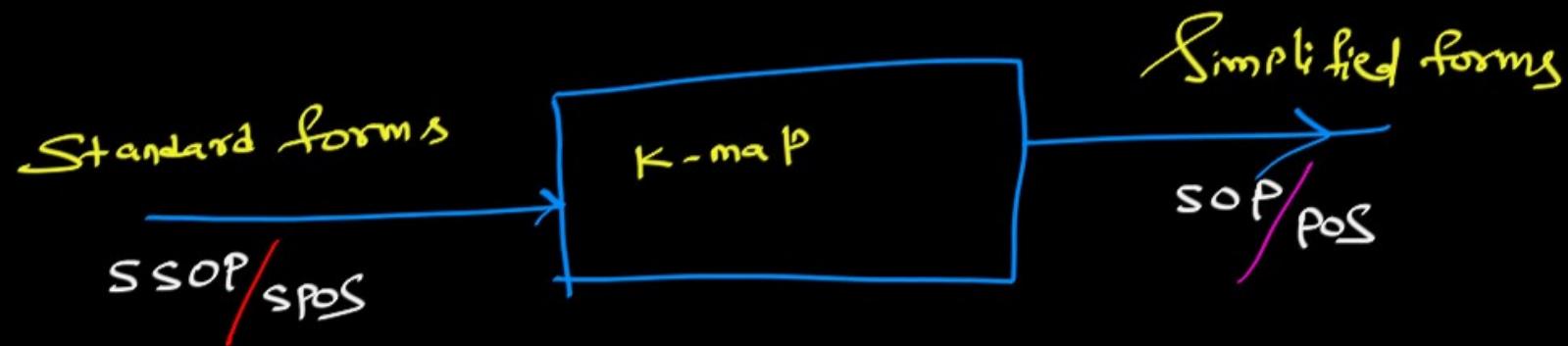
Purpose

Complicated Boolean functions can be simplified

In Boolean Algebra , if number of terms increases,
Then complexity also increases.

K-Maps

For K-map inputs are Standard forms ,
where as outputs are simplified forms



K-Maps

I) K- map Simplification for SSOP

II) K- map Simplification for SPOS

I) K- map Simplification for SSOP

Grouping Priority

To get min literals, Group Max Combinations

Priority $\Rightarrow \dots$ 16 terms or 8 terms or 4 terms or 2 terms or 1 term

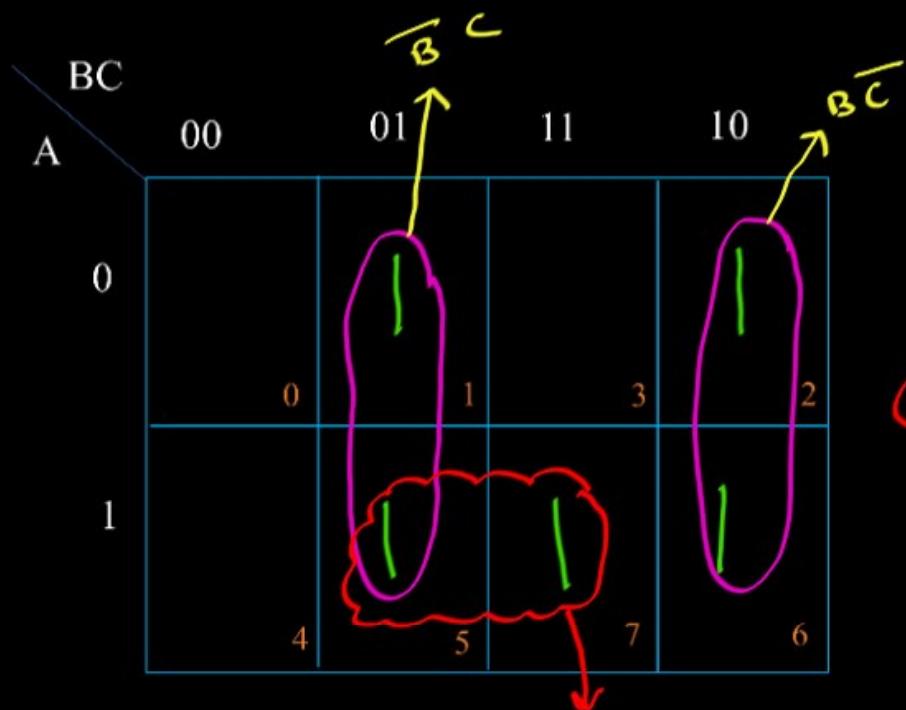
K-Map

Group of 2 terms is called as Pair

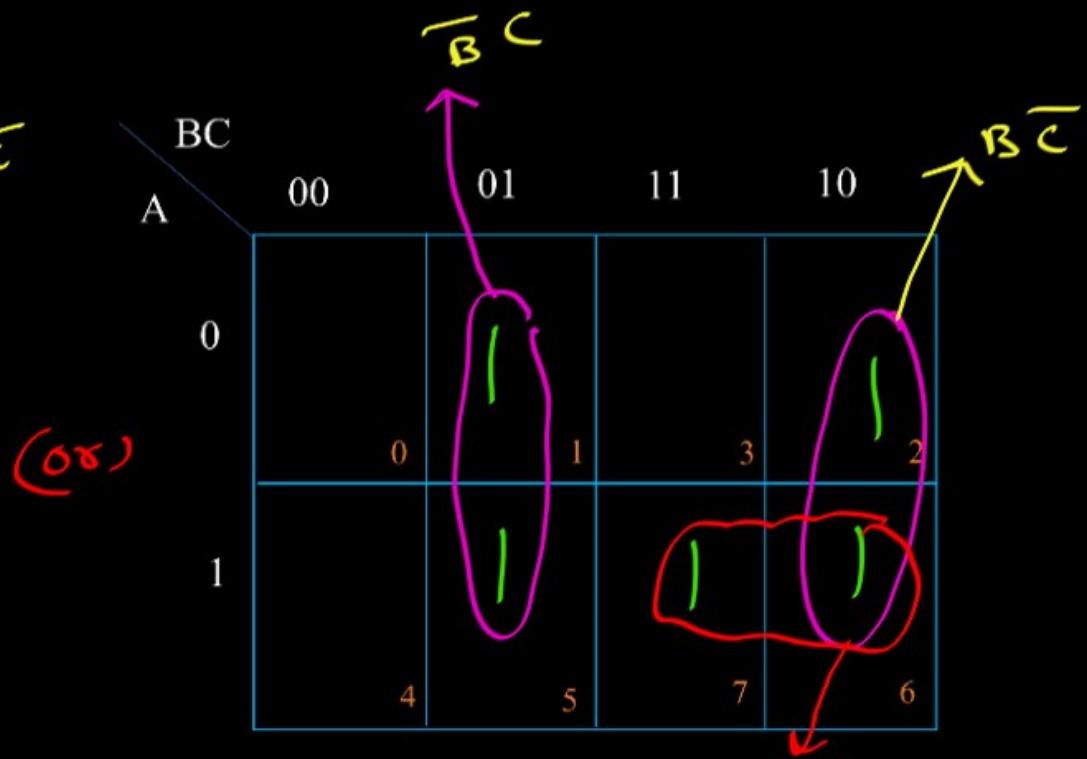
Group of 4 terms is called as Quad

Group of 8 terms is called as Octet

Q) Simplify $f(A, B, C) = \sum m(1, 2, 5, 6, 7)$



$$f = \overline{B}C + BC + AC$$



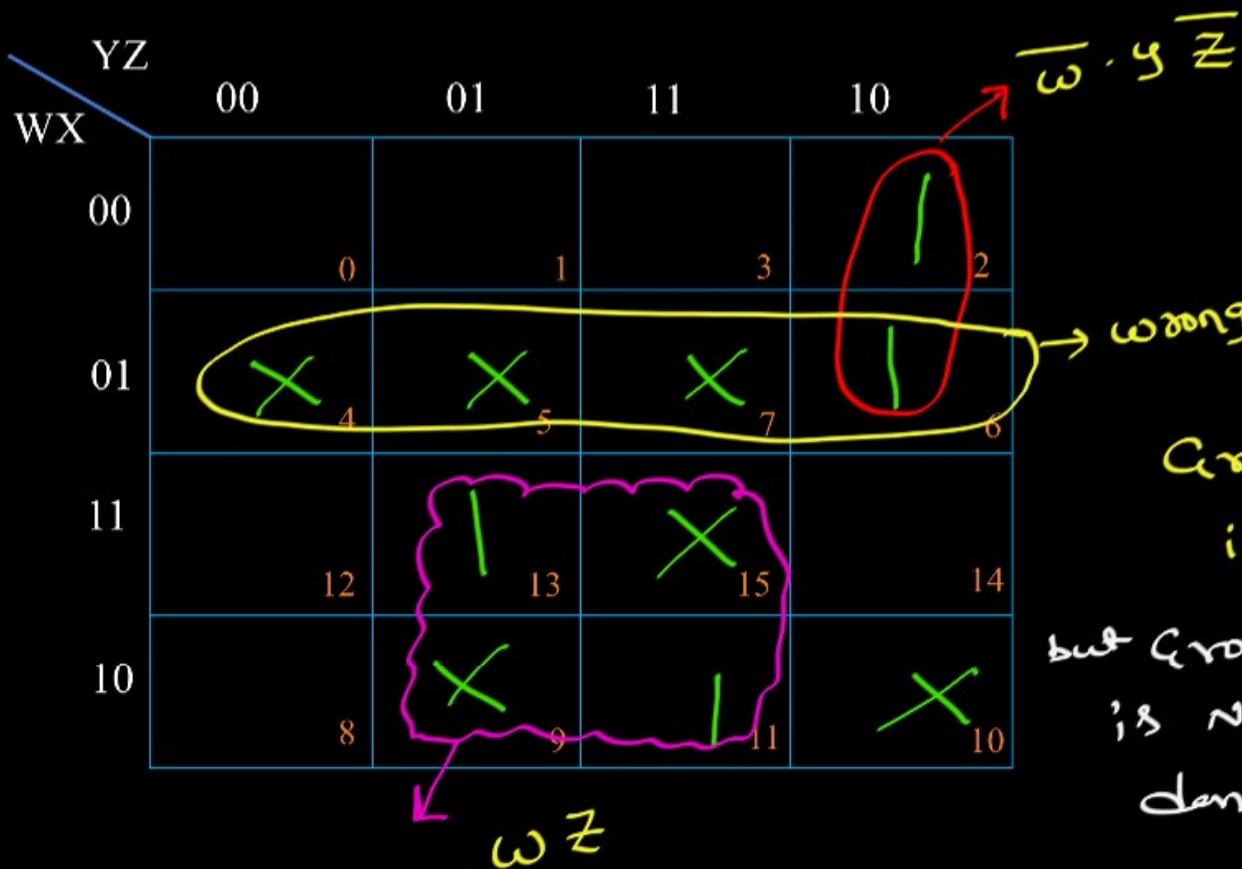
Here Both are correct;

$$f = \overline{B}C + BC + AB$$

K-map Simplification with Don't Care

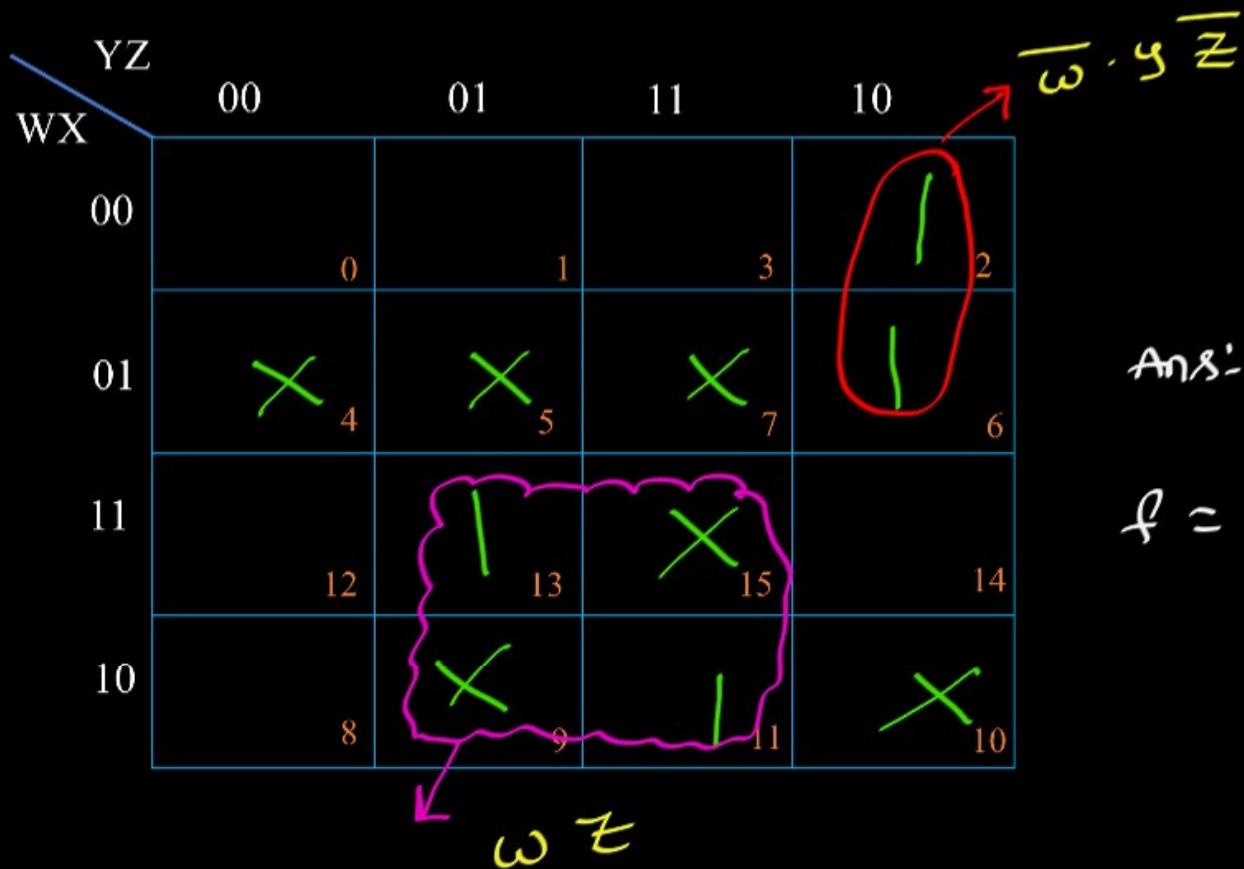
$$X \rightarrow \sum_{i=1}^{\infty} (0^i)$$

Q) Simplify $F(w,x,y,z) = \sum m(2,6,11,13) + \sum d(4,5,7,9,10,15)$



wrong \rightarrow Reason
 Grouping of all minterms
 is compulsory
 but grouping of all don't care
 is not compulsory, some of
 don't care may be empty.

Q) Simplify $F(w,x,y,z) = \sum m(2,6,11,13) + \sum d(4,5,7,9,10,15)$



Ans: \rightarrow

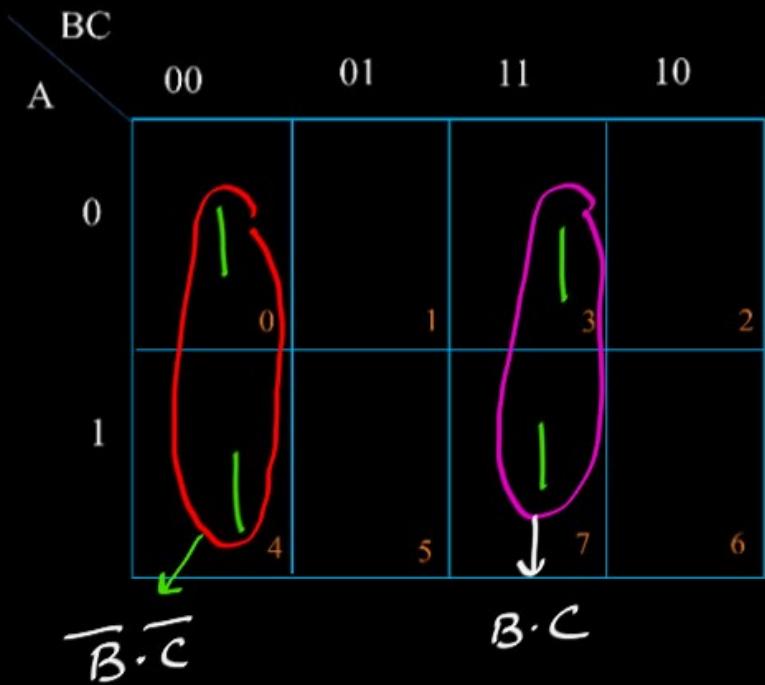
$$f = \omega z + \overline{\omega} y \cdot \overline{z}$$

II) K- map Simplification for SPOS

Q. Simplify $F(A,B,C) = \prod M(0,3,4,7)$

SOP

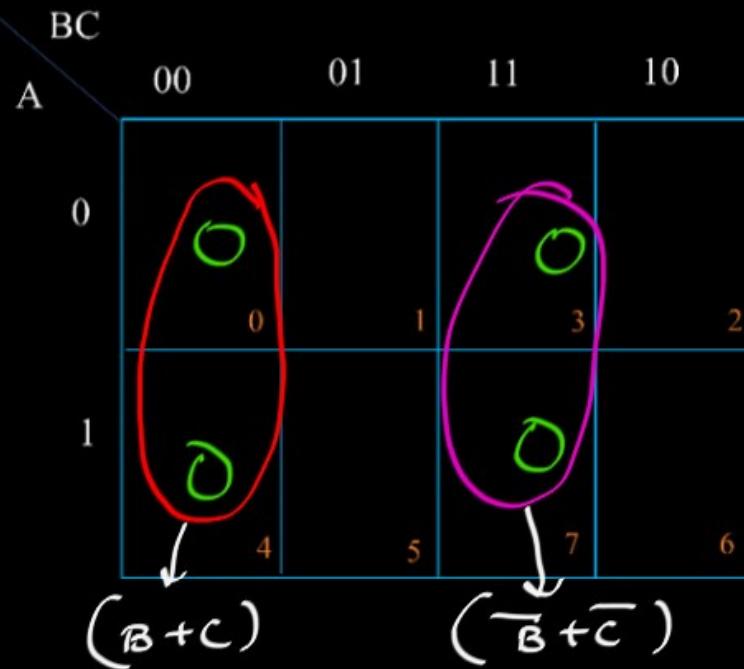
Q. $F(A, B, C) = \Sigma m(0, 3, 4, 7)$



Ans: $f_{SOP} = \overline{B} \cdot \overline{C} + B \cdot C$

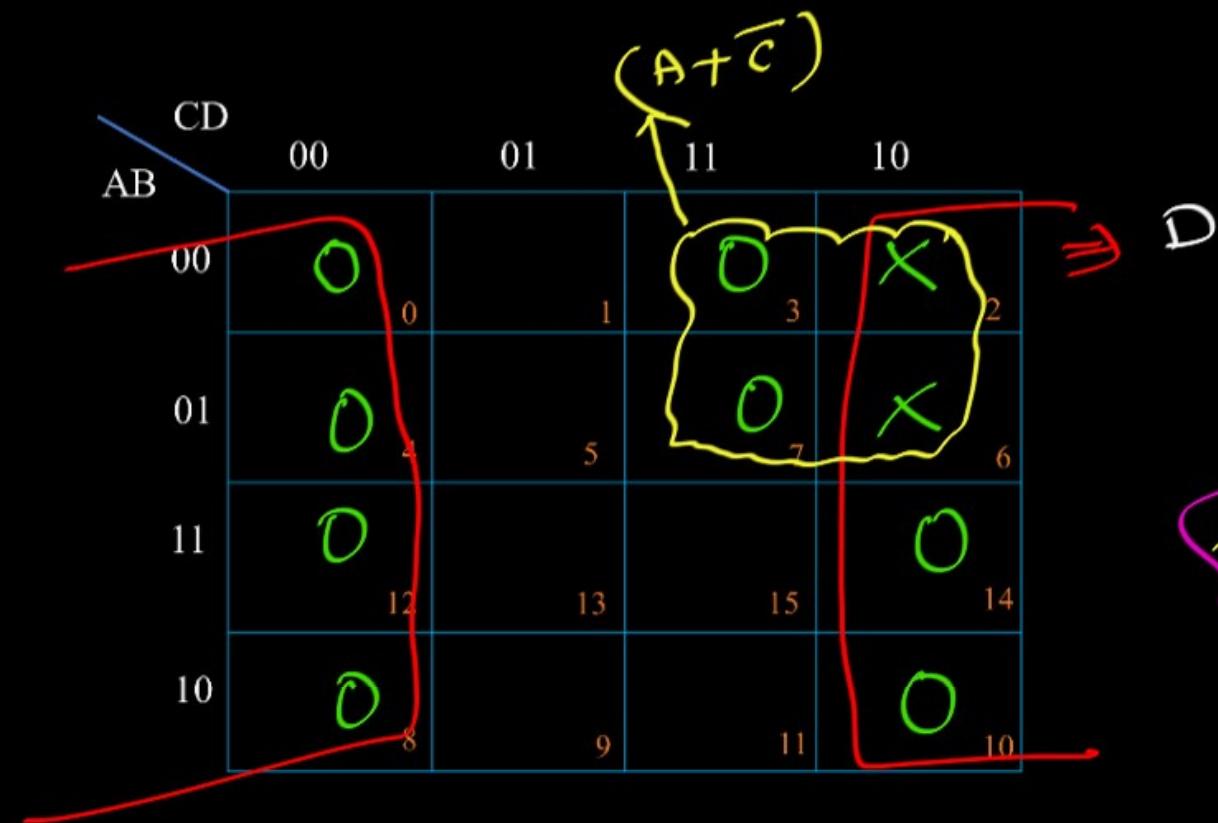
SPOS

Q. $F(A, B, C) = \prod M(0, 3, 4, 7)$



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

$$Q. \quad F(A,B,C,D) = \prod M (0,3,4,7,8,10,12,14) + d (2,6)$$



Ans: $f = D (A+C)$

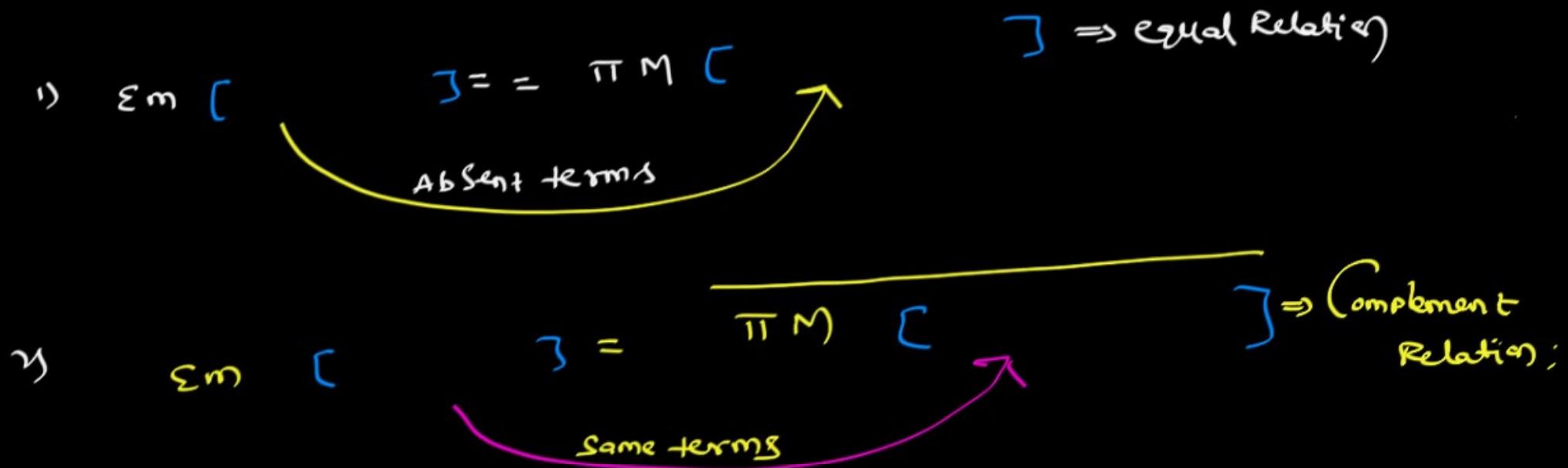
Consensus theorem

$$\alpha y + \bar{\alpha} z + yz = \alpha y + \bar{\alpha} z$$

$$\alpha y + \bar{\alpha} z + yz = \alpha y + \bar{\alpha} z$$

1. 2 \rightarrow 1. 3 \rightarrow $\underbrace{2 \times 3}_{\text{Neglect}}$

Relation Between Σm and $\prod M$



Relation Between ΣM and ΠM

$$\Sigma M [] \quad \exists = \Pi M [] \Rightarrow \text{equal Relation}$$

Absent terms

$$\Sigma M [0, 3, 4, 7] = \Pi M [1, 2, 5, 6] \Rightarrow \text{equal Relation}$$

$$\Sigma M [0, 3, 4, 7] = \overline{\Pi M [0, 3, 4, 7]} \Rightarrow \text{complement relation}$$

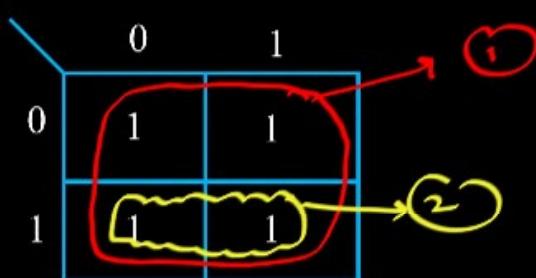
Prime Implicant



Implicant : It is a set of all adjacent min terms , nothing but group of Possible combinations of octet , Quads , pairs etc.
(No need to follow K-map rules)

Prime Implicant (PI) : It is an Implicant which is not a subset of another Implicant
(No need to follow K-map redundant property)

Ex :



Implicants : $\Rightarrow \textcircled{1}, \textcircled{2}$

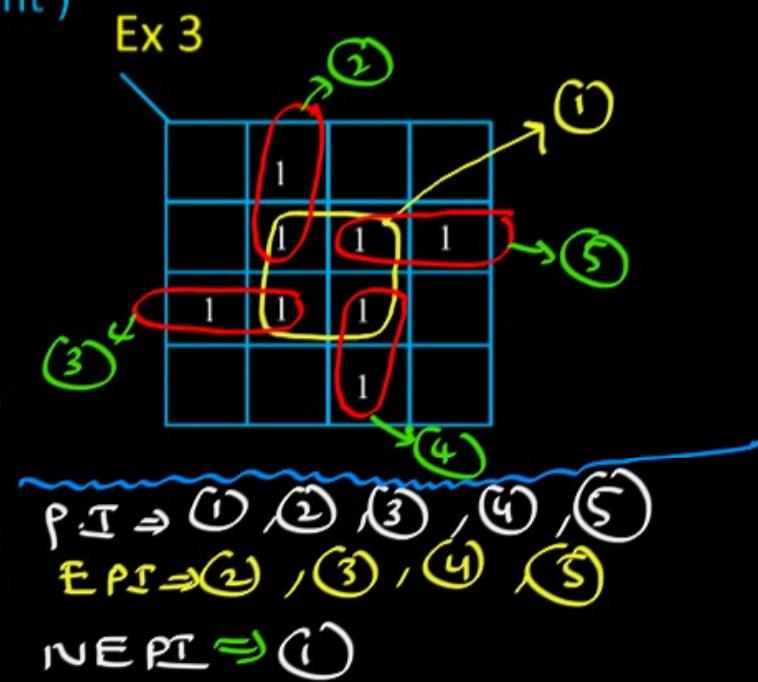
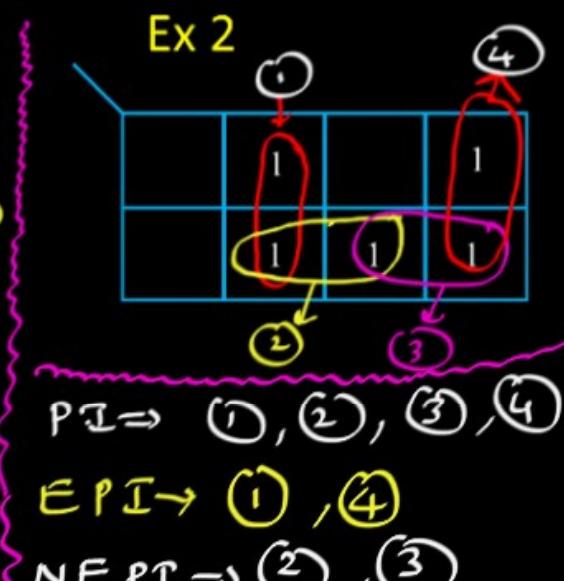
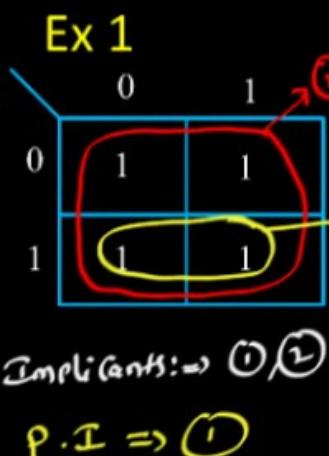
Prime Implicant : $\Rightarrow \textcircled{1}$

$\textcircled{2}$ is not a P.I reason, it is subset of $\textcircled{1}$ 't one

Prime Implicant (PI) : It is an Implicant which is not a subset of another Implicant
 (No need to follow K-map redundant property)

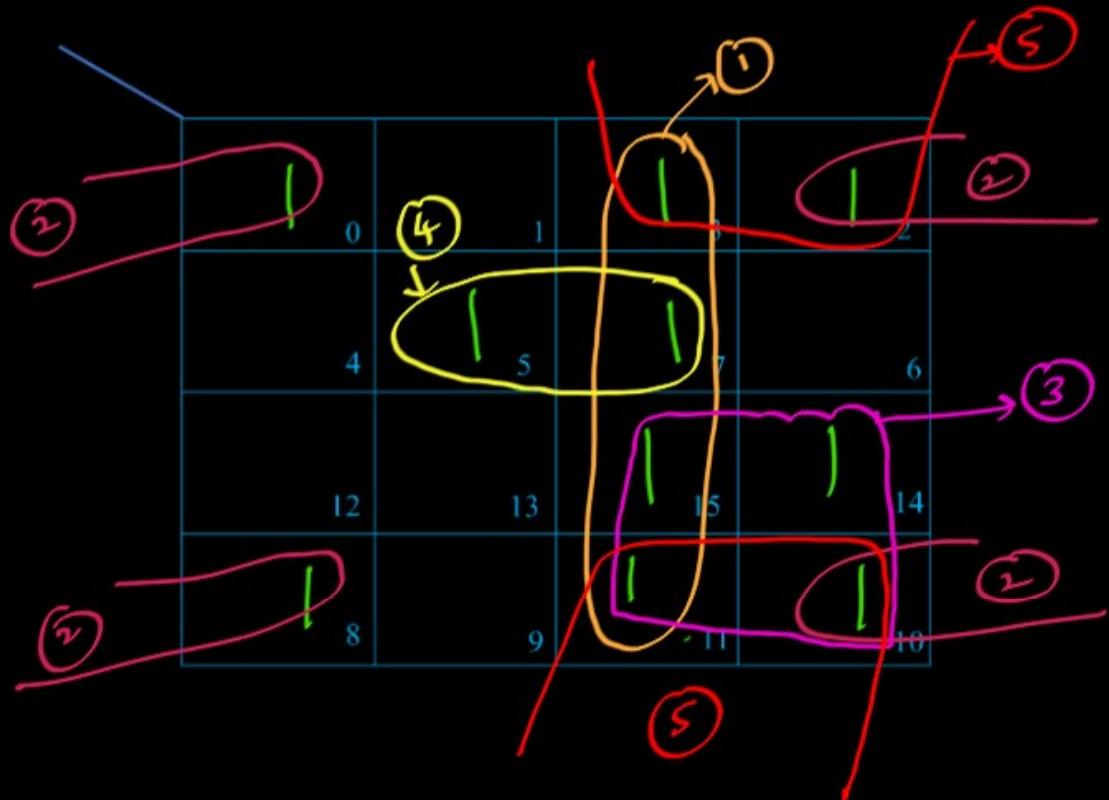
Essential Prime Implicant (EPI)

It is prime implicant which contains at least one Independent min term
 (can not be covered by any other Prime implicant)



$$F(A,B,C,D) = \Sigma m(0, 2, 3, 5, 7, 8, 10, 11, 14, 15)$$

Number of P.I , E.P.I respectively are ?



P.I \Rightarrow no. of P.I are five

E.P.I \Rightarrow (2), (3), (4)

NEPI \Rightarrow (1), (5)

Prime Implicant (PI) : It is an Implicant which is not a subset of another Implicant
(No need to follow K-map redundant property)

Essential Prime Implicant (EPI)

It is prime implicant which contains at least one Independent min term
(can not be covered by any other Prime implicant)

Previous Questions

Previous Questions

Q. $F(A,B,C) = \Sigma m(0, 1, 2, 5, 6, 7)$

Number of PI, EPI respectively are

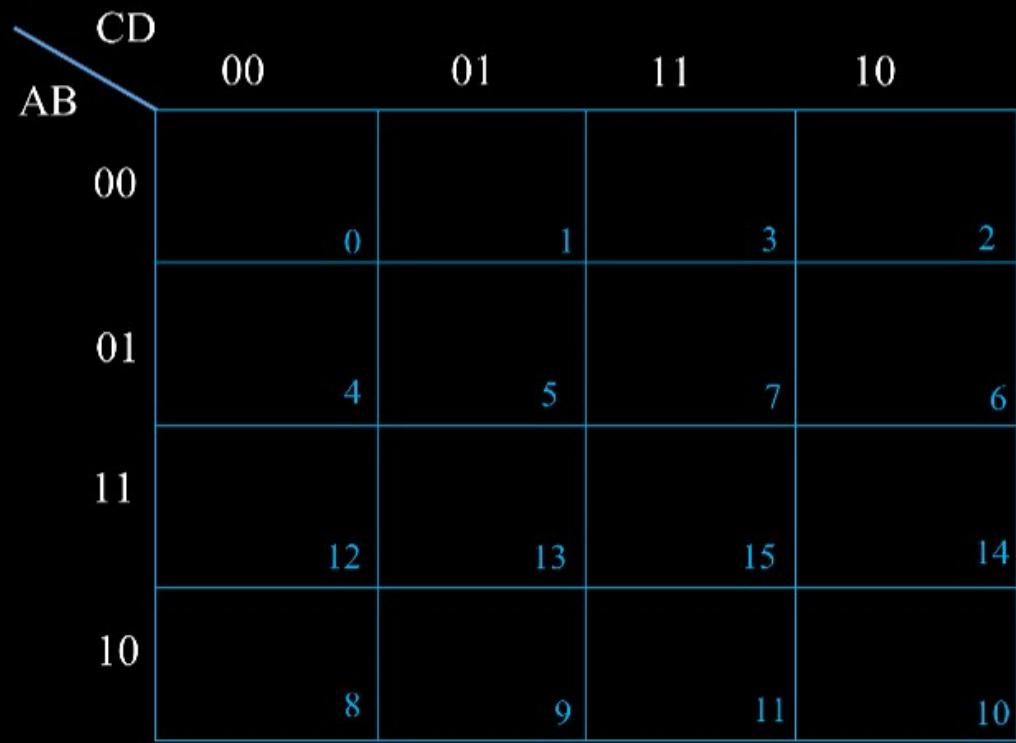
		BC	00	01	11	10
		A	0	1	3	2
		0	0	1	3	2
		1	4	5	7	6

Previous Questions

Q. Consider the function $f(A,B,C,D) = \Sigma m [0,1,4,6,7,8,10,14,15]$

The number of Prime Implicants are _____

$$f(A, B, C, D) = \Sigma m [0, 1, 4, 6, 7, 8, 10, 14, 15]$$



Previous Questions

Q. What is the minimum number of 2 input NOR gates required to implement a 4 variable function expressed in sum of min terms form as

$$f = \Sigma (0, 2, 5, 7, 8, 10, 13, 15) ?$$

Assume that all the inputs and their complements are available. Answer: _____.

(CSE-GATE-2019 – 2M)

Given $f = \Sigma (0, 2, 5, 7, 8, 10, 13, 15)$

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

Chapter 4 ended
Now Chapter 5

Chapters

5) Combinational Logic Circuits



Introduction



Adders, Subtractors



Comparator



Encoders & Decoders



Multiplexers



Previous Questions

**Unless you don't respect your Parents
The world won't respect you**
**Unless you won't love your parents
The world won't love you**

So,

RESPECT and LOVE your parents
rather than doing it to someone
Who don't even deserve it

Combinational Logic Circuits

Combinational Logic Circuits

Definition

A circuit in which present output depends only on present input but not on previous input is called Combinational Logic Circuit.

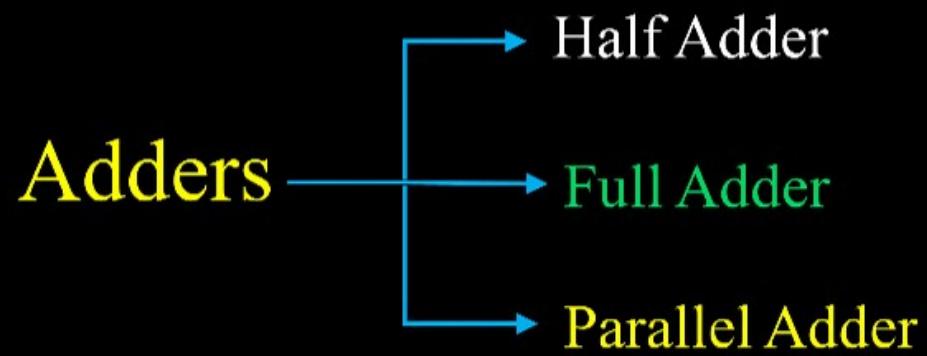
for combinational logic circuit memory element is not required to store previous data.

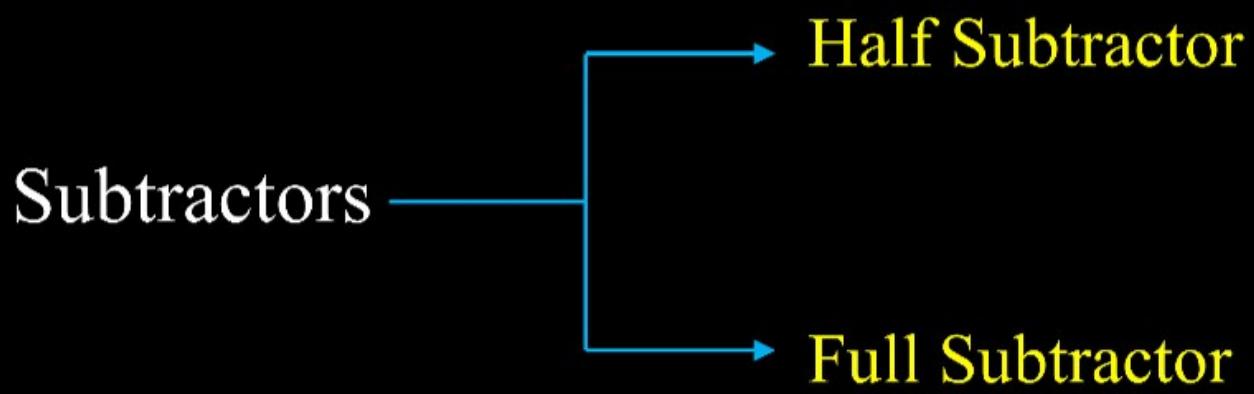
Combinational Logic Circuits

Examples

1. adders
2. Subtractors
3. Parallel Adder
4. Encoders
5. Decoders
6. Multiplexers
7. De multiplexers

Adders and Subtractors





Half Adder

Step 1) Definition

Half adder is a Combinational Logic Circuit which consists of 2 inputs and 2 outputs (Sum , Carry)

Step 2) Block Diagram



Half Adder

Step 3) Truth table

Inputs		Outputs	
		Sum	Carry
A	B		
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

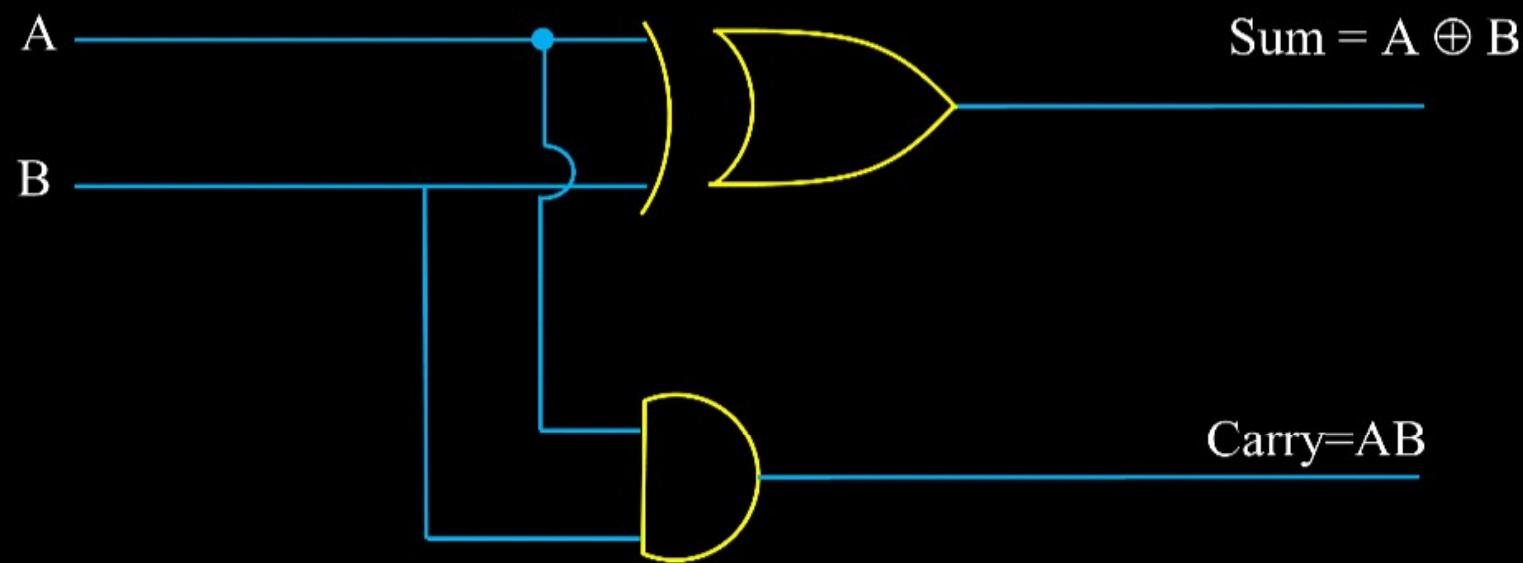
Step 4) Output Expressions

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

$$\text{Carry} = A \cdot B$$

Half Adder

Step 5) Logic diagram

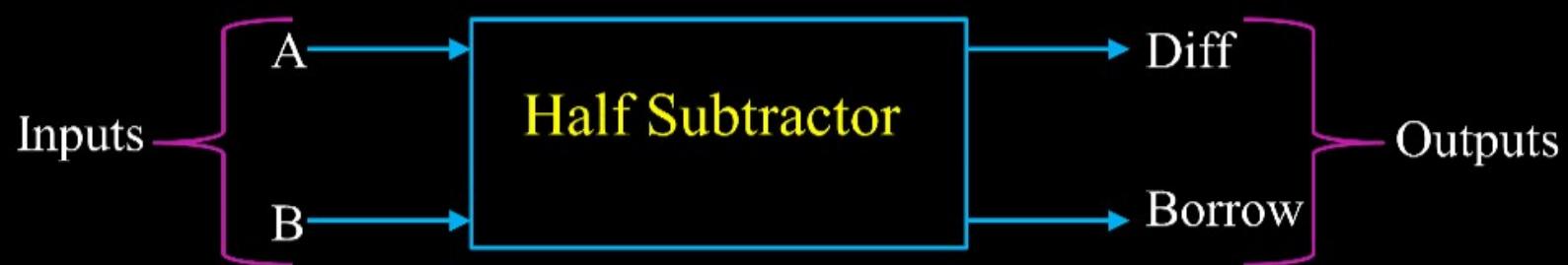


Half Subtractor

Step 1) Definition

Half Subtractor is a Combinational Logic Circuit which consists of 2 inputs and 2 outputs (Difference , Borrow)

Step 2) Block Diagram



Half Subtractor

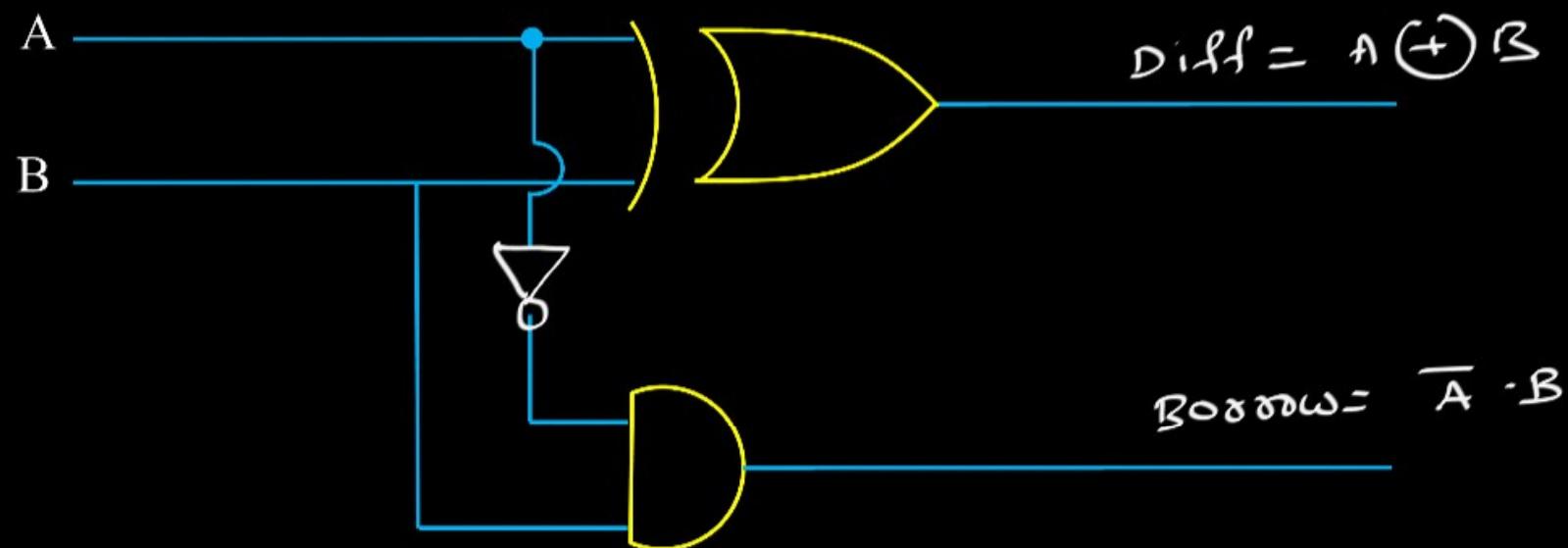
Step 3) Truth table

Step 4) Output Expressions

Inputs		Outputs	
A	B	Diff	Borrow
0	0		
0	1		
1	0		
1	1		

Half Subtractor

Step 5) Logic diagram



Full Adder

Step 1) Definition

Full adder is a Combinational Logic Circuit which consists of 3 inputs and 2 outputs (Sum , Carry)

Step 2) Block Diagram



Step 3) Truth table

Full Adder

Decimal

Inputs

Outputs

Step 4) Output Expressions

A

B

C

Sum

Carry

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

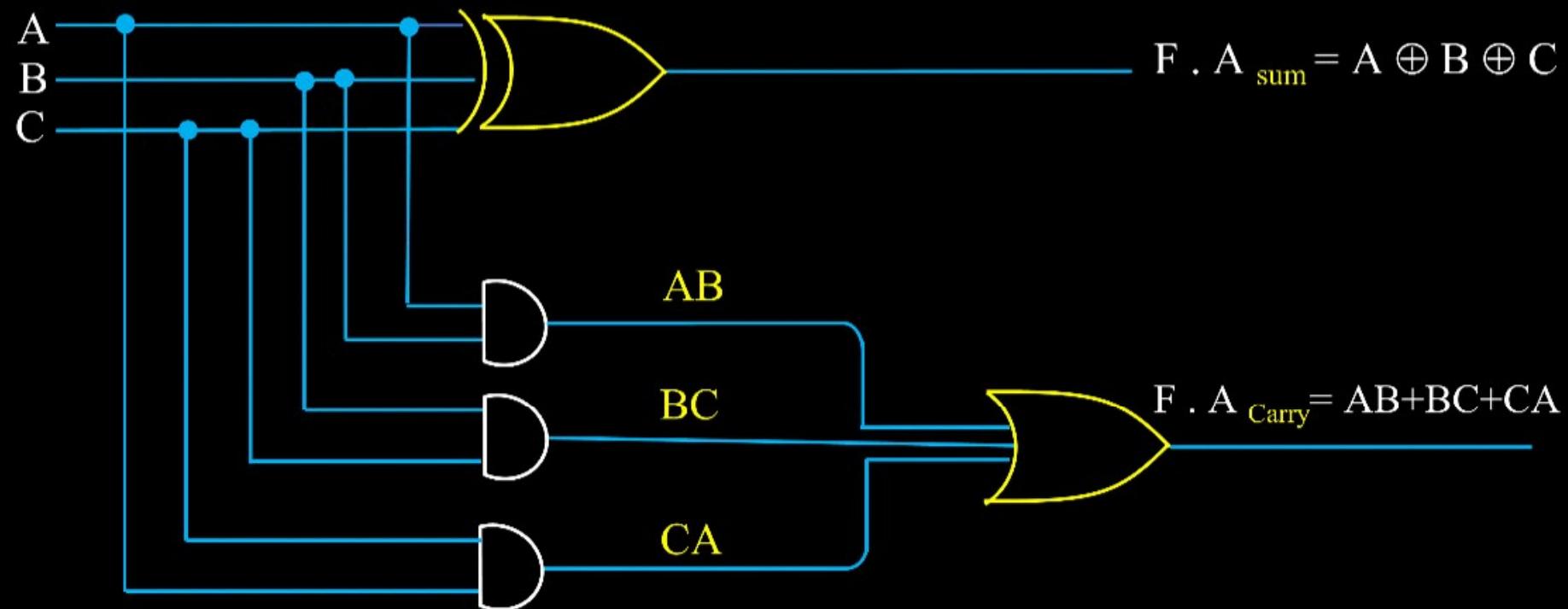
Step 5) Output Expression Simplification

		BC	A		
		00	01	11	10
0		0	1	3	2
1	0				
	1	4	5	7	6

		BC	A		
		00	01	11	10
0		0	1	3	2
1	0				
	1	4	5	7	6

Full Adder

Step 5) Logic diagram

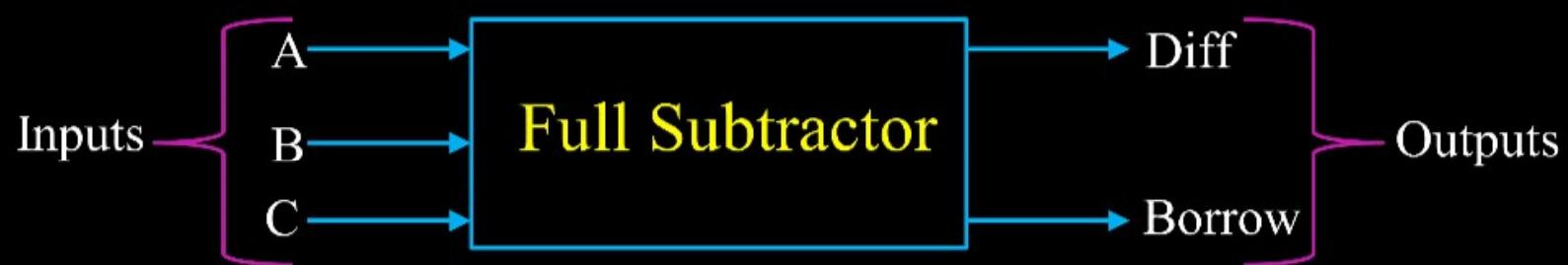


Full Subtractor

Step 1) Definition

Full Subtractor is a Combinational Logic Circuit which consists of 3 inputs and 2 outputs (Difference , Borrow)

Step 2) Block Diagram



Step 3) Truth table

Full Subtractor

Decimal

Inputs

Outputs

Step 4) Output Expressions

A

B

C

Diff

Borrow

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

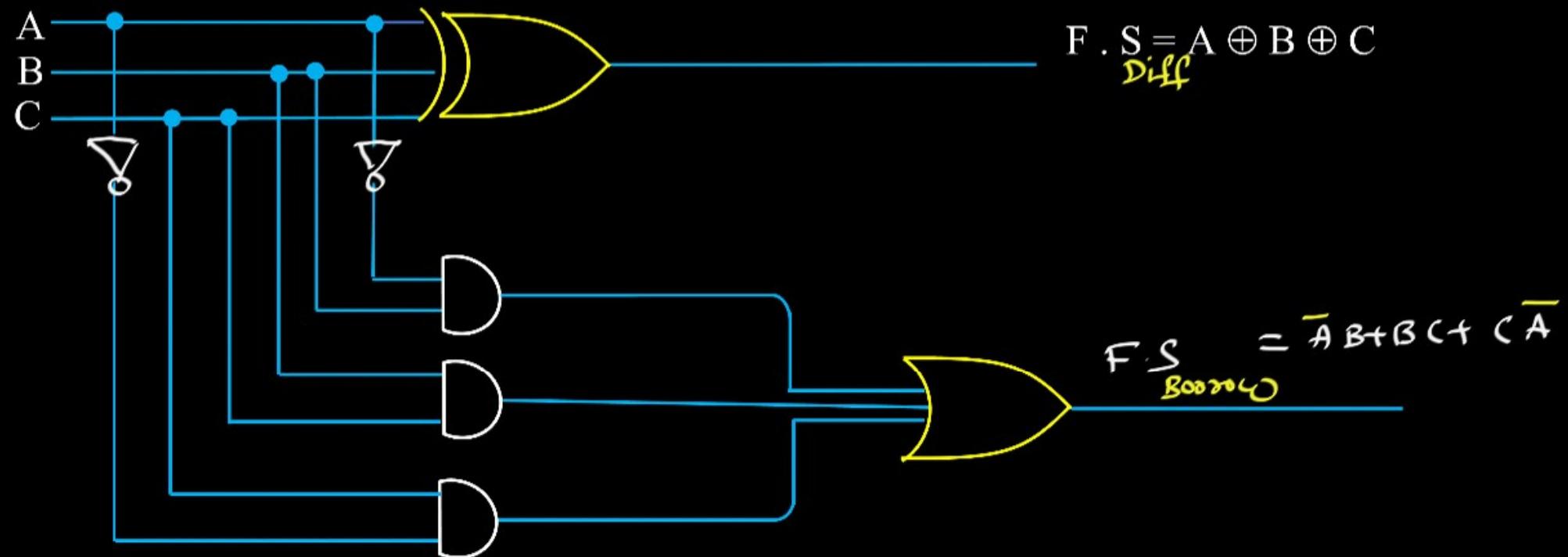
Step 5) Output Expression Simplification

		BC	A		
		00	01	11	10
0		0	1	3	2
1	0				
	1	4	5	7	6

		BC	A		
		00	01	11	10
0		0	1	3	2
1	0				
	1	4	5	7	6

Full Subtractor

Step 5) Logic diagram



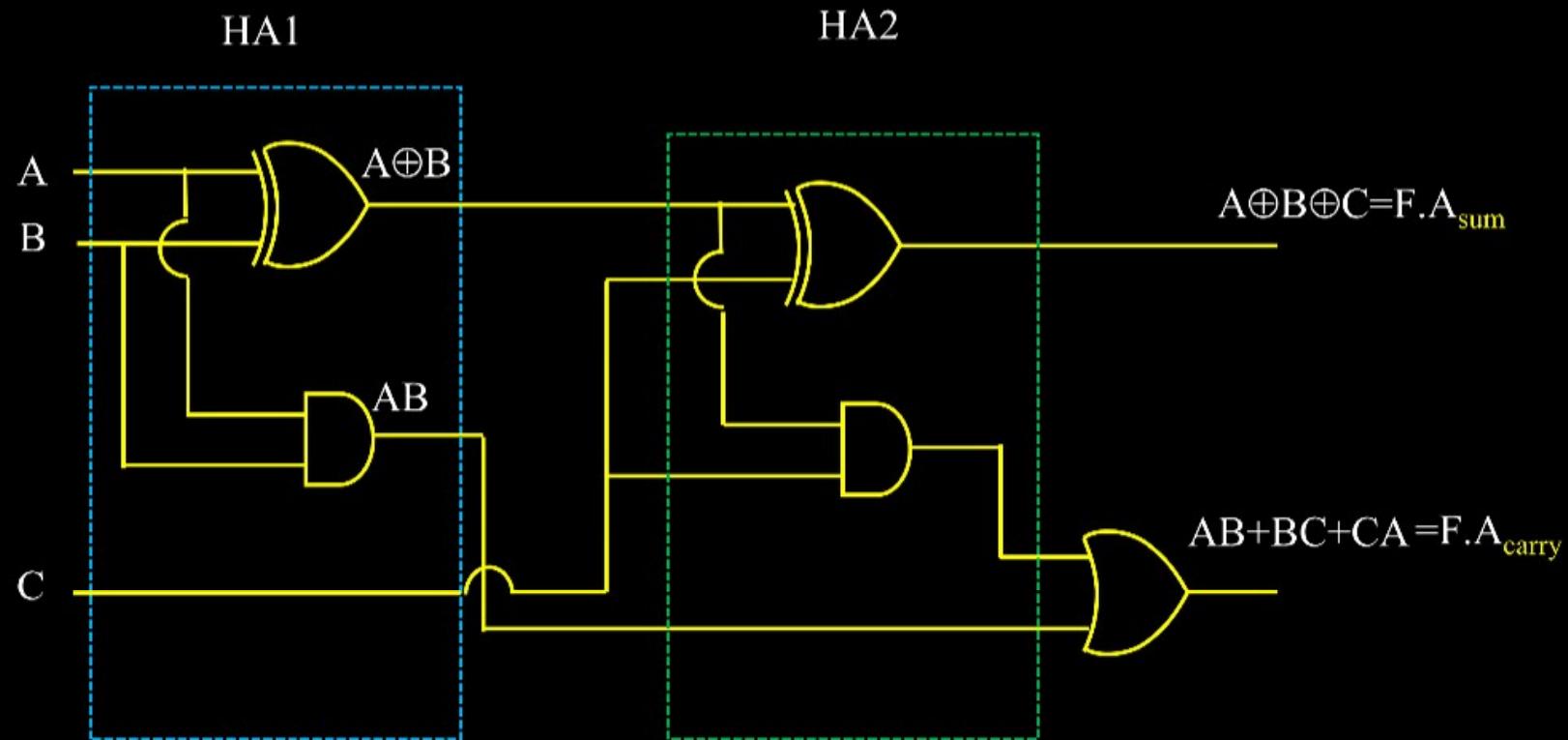
Short Notes

Full Adder design using Half adders

Q) Number of Half adders, Basic gates required to implement Full adder respectively are _____

- (a) 2,2
- (b) 2,1
- (c) 1,2
- (d) 2,3

Design Full Adder using Half adders



Q) Number of Half adders, Basic gates required to implement Full adder respectively are _____

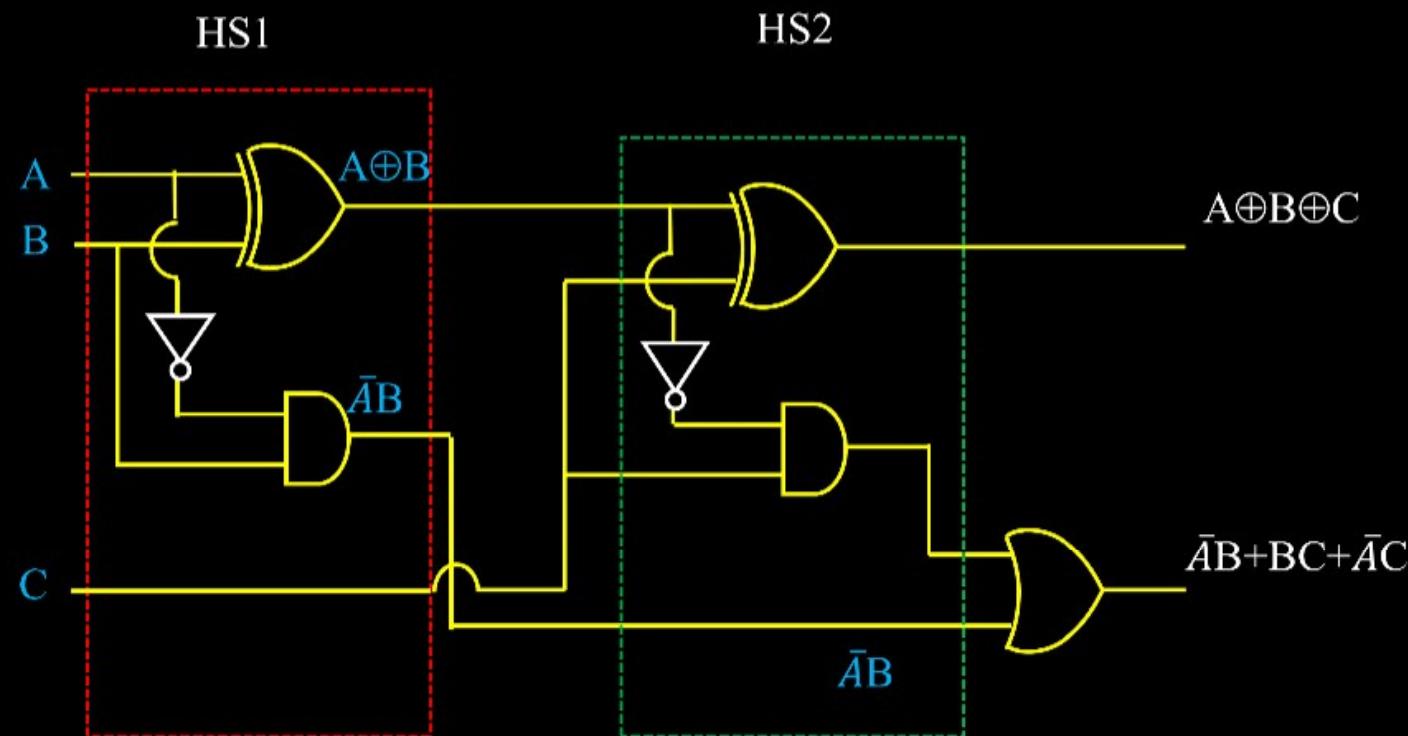
- (a) 2,2
- (b) 2,1
- (c) 1,2
- (d) 2,3

Full Subtractor design using Half subtractors

Q) Number of Half subtractor, Basic gates required to implement Full subtractor

- (a) 2, 1
- (b) 2, 2
- (c) 2,3
- (d) 2,5

Design Full Subtractor using Half Subtractors



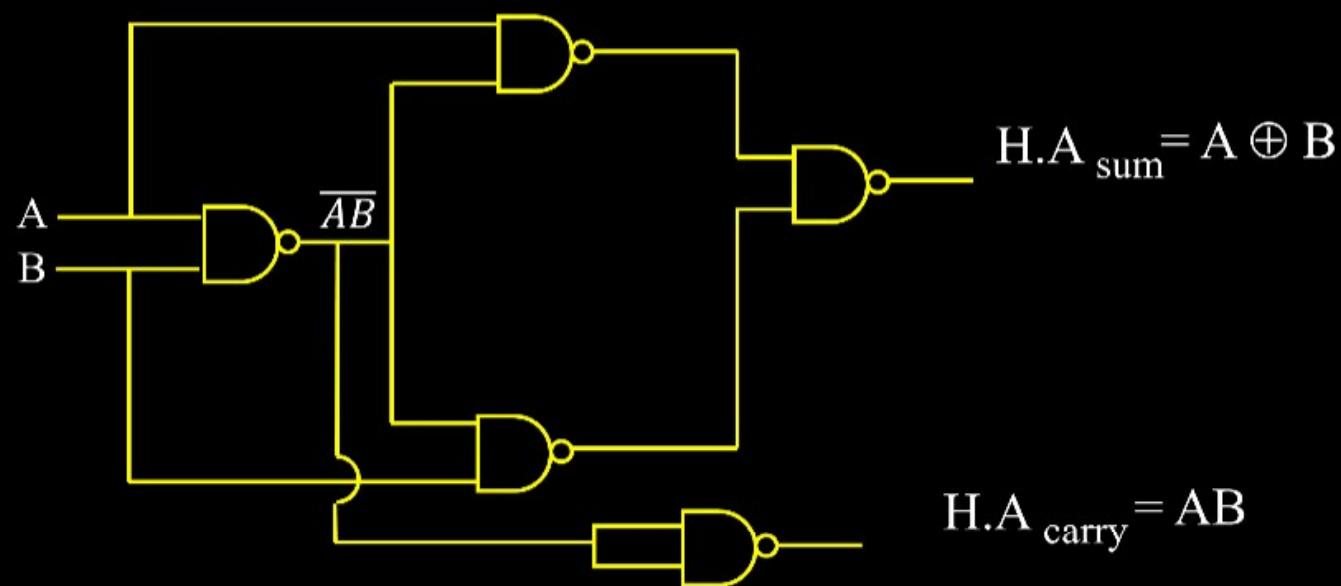
Q) Number of Half subtractor, Basic gates required to implement Full subtractor

- (a) 2, 1
- (b) 2, 2
- (c) 2,3
- (d) 2,5

Adders and Subtractors Design Using universal Logic Gates

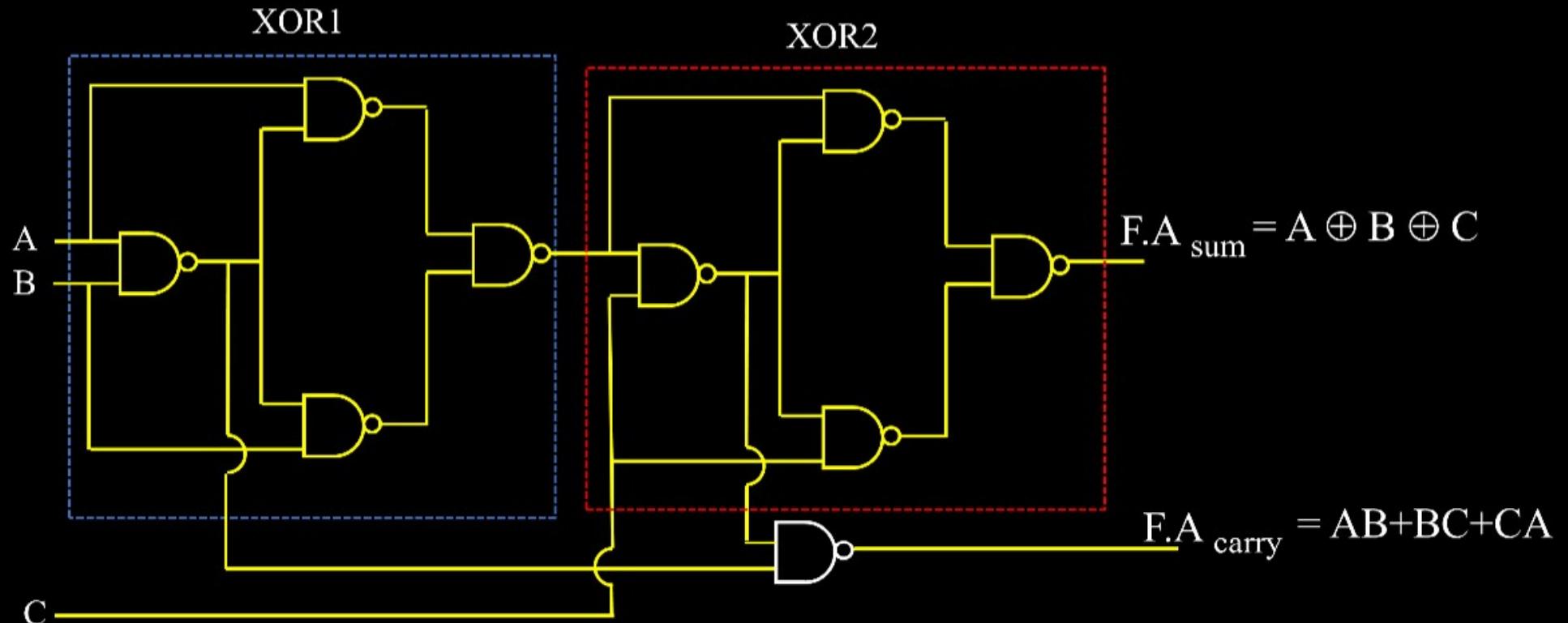
Design of Half Adder using NAND Gates

Design of Half Adder using NAND Gates



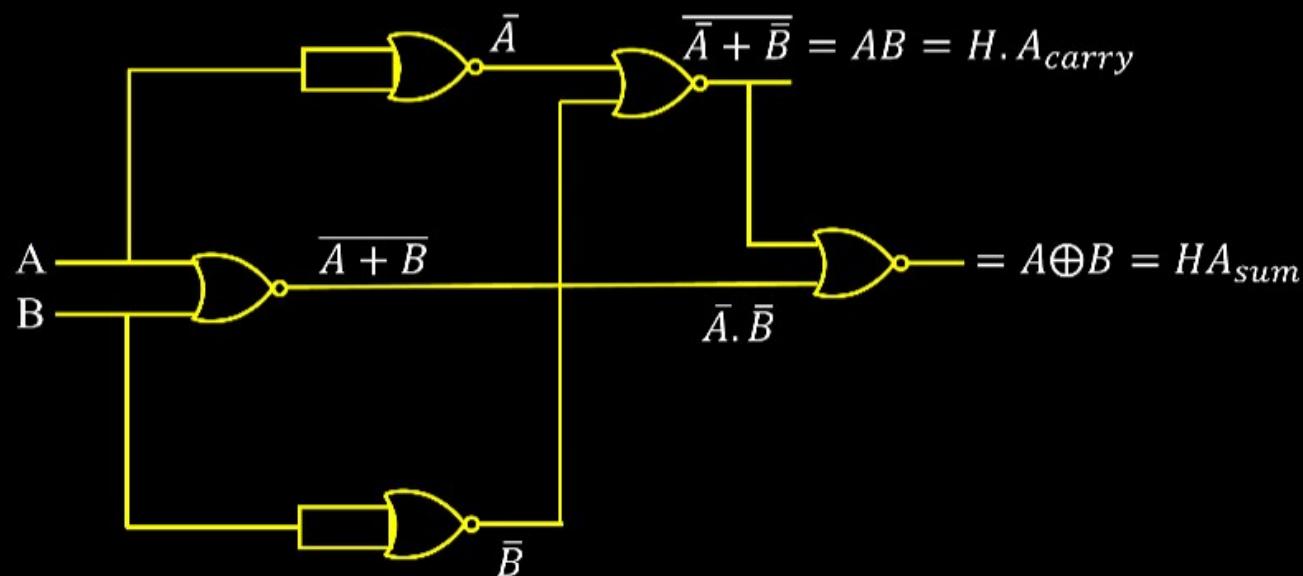
To design H.A , total 5 NAND gates are required

Design of Full adder using NAND Gates



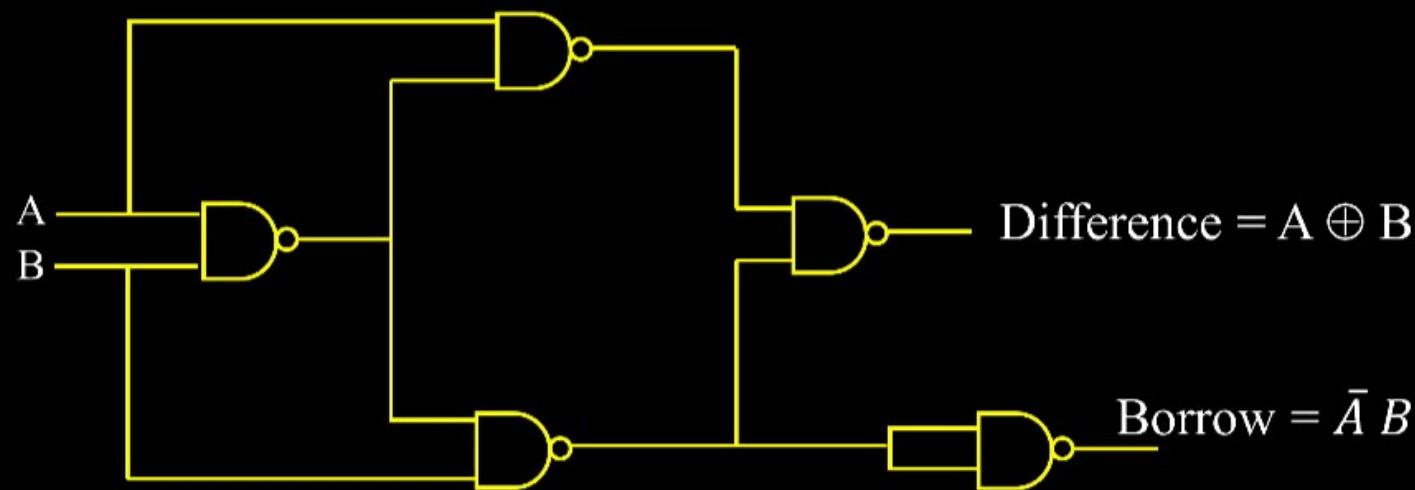
To design F.A , total 9 NAND gates are required

Design of Half Adder using NOR Gates



To design H.A , total 5 NOR gates are required

Design of Half subtractor using NAND gates



To design H.S , total 5 NAND gates are required

Summary

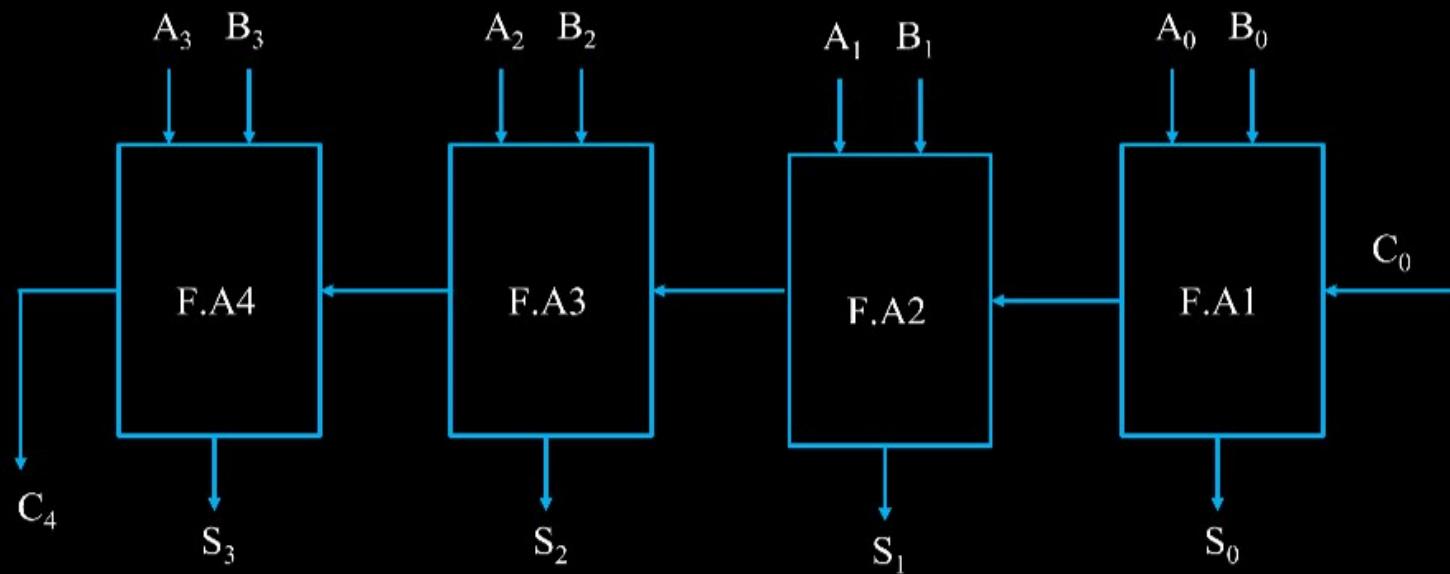
	Min no. of NAND gates	Min no. of NOR gates
Half Adders	5	5
Half Subtractors	5	5
Full Adder	9	9
Full Subtractor	9	9

Half = 5
Full = 9

Parallel Adder or Binary Adder or Ripple Carry adder

Circuit Diagram

4 – bit Parallel Adder as shown below



Previous Questions

Q) To Design 17 bit Parallel Adder, minimum number of Half Adder , Basic Gates required respectively

ISRO

To design "n-bit" parallel adder we required
= "n" F.A (or) $1 \text{ H.A} + (n-1) \text{ F.A}$ (or) $(2^{n-1}) \text{ H.A} + (n-1) \text{ OR gates}$

Here $n = 17$ bit

$$(2^{n-1}) \text{ H.A} = 2^{17-1} = 33 \text{ H.A}$$

$$\text{and } 17-1 = 16 \text{ OR gates}$$

$(n-1) \text{ OR} = 17-1 = 16 \text{ OR gates}$
option (d) is right.

Previous Questions

Q) In a 4 bit parallel binary adder, a full adder takes 32-nano seconds to produce sum and 14 ns to produce carry then total time required for addition is ?

- (a) 226 ns
 - (b) 88 ns
 - (c) 14 ns
 - (d.) 74 ns

Soln

$$\text{Total time} = (n-1) t_{\text{carry}} + \text{Max}[t_{\text{sum}}, t_{\text{carry}}]$$

Here given $n = 4$ bit, $t_{\text{sum}} = 32 \text{ ns}$, $t_{\text{carry}} = 14 \text{ ns}$

$$\text{Total time} = (4-1) \times 14 \text{ ns} + \text{Max}[32, 14]$$

$$= 3 \times 14 + 32 = 74 \text{ nsec}$$

Overflow

Introduction

When two numbers of “ n ” bits are added and the sum Occupies “ n+1 ” bits then we can say overflow occurred.

EX) Add 2 ,3

Overflow in Signed Numbers

In signed 2's complement Arithmetic If the result exceeds the range $-(2^{n-1})$ to $+(2^{n-1}-1)$ then we can say overflow occurs.

Overflow in Signed Numbers



Conclusions

- For n-bit number condition to detect over flow $\Rightarrow C_{n-1} \oplus C_n = 1$
- For n-bit number condition to occur over flow
 $\Rightarrow \overline{A}_{n-1} \cdot \overline{B}_{n-1} \cdot S_{n-1} + A_{n-1} \cdot B_{n-1} \cdot \overline{S}_{n-1} = 1$

Previous Question

Q. When two 8-bit numbers $A_7 \dots A_0$ and $B_7 \dots B_0$ in 2's complement representation (with A_0 and B_0 as the least significant bits) are added using a ripple-carry adder, the sum bits obtained are $S_7 \dots S_0$ and the carry bits are $C_7 \dots C_0$.

An overflow is said to have occurred if

(GATE-2017)

- (a) The carry bit C_7 is 1
- (b) All the carry bits ($C_7 \dots C_0$) are 1
- (c) $(A_7 \cdot B_7 \cdot \bar{S}_7 + \bar{A}_7 \cdot \bar{B}_7 \cdot S_7) \text{ is } 1$
- (d) $(A_0 \cdot B_0 \cdot \bar{S}_0 + \bar{A}_0 \cdot \bar{B}_0 \cdot S_0) \text{ is } 1$

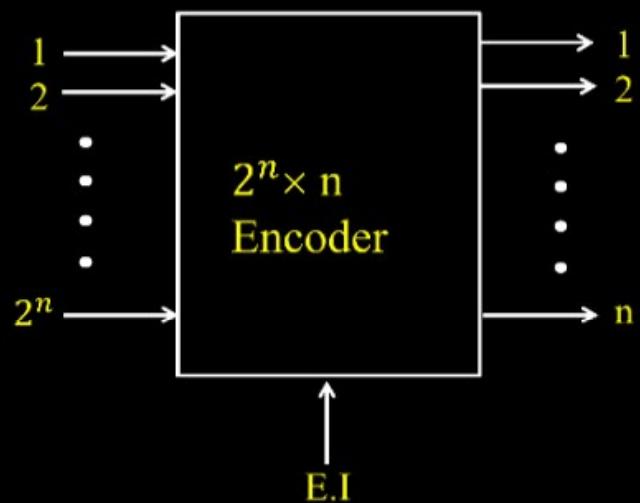
Encoders and Decoders

Encoders

Step 1) Definition

It is a Combinational Logic Circuit consists of “ 2^n ” inputs and “ n ” outputs

Step 2) Block Diagram

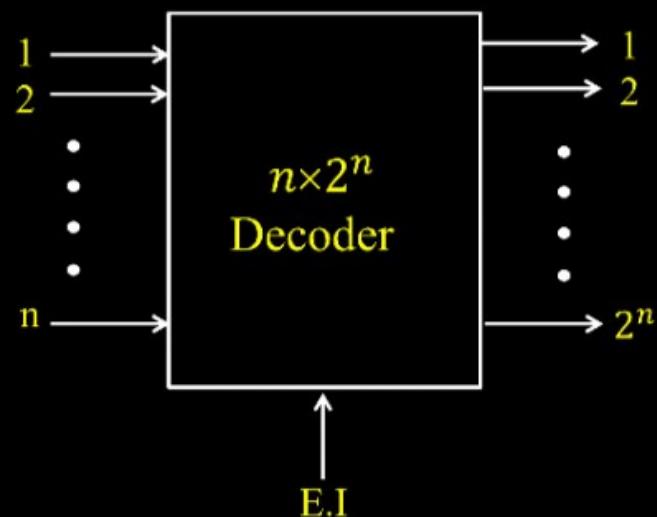


Decoders

Step 1) Definition

It is a Combinational Logic Circuit consists of “ n “ inputs and “ 2^n “ outputs

Step 2) Block Diagram



Types of Decoders

Based on output nature , there are two types of decoders

- 1) Active High output Decoder
- 2) Active Low output Decoder

Enable Input (E.I)

It is used to control the operation of Encoder and Decoder

Types of Enable Inputs

- 1) Active High Enable Input
- 2) Active Low Enable Input

Previous Question

Previous Question

Q) To design 4×16 decoder, minimum number of 2×4 decoder required is _____

To design 4×16 decoder, minimum number of 2×4 decoder required is _____

Shortcut Technique

$\frac{\text{Required output}}{\text{Given output}}$

$$\text{Step 1) } \frac{16}{4} = 4$$

$$\text{Step 2) } \frac{4}{4} = 1$$

5

Ans: $\Rightarrow 5$

To design 10×1024 decoder, minimum number of 3×8 decoders required is ____.

(a) 144

(b) 146

(c) 147

(d) 148

Shortcut Technique

Required output
Given output

$$\left\{ \begin{array}{l} \text{Step 1)} \quad \frac{1024}{8} = 128 \\ \text{Step 2)} \quad \frac{128}{8} = 16 \\ \text{Step 3)} \quad \frac{16}{8} = 2 \\ \text{Step 4)} \quad \frac{2}{8} = 0.25 \approx \frac{1}{4} \end{array} \right\} 146$$

Sweetcut Technique

To Design larger decoder with smaller decoder , always Answer is odd number

Decoder Shortcut Technique

$$\frac{\text{Required output}}{\text{Given output}}$$

Encoder Shortcut Technique

$$\frac{\text{Required input}}{\text{Given input}}$$

To construct 64×6 Encoder minimum nor of 4×2 Encoders required is ____.

Encoder Shortcut Technique

Required input

Given input

$$\left. \begin{array}{l} \text{Step1) } \frac{64}{4} = 16 \\ \text{Step2) } \frac{16}{4} = 4 \\ \text{Step3) } \frac{4}{4} = 1 \end{array} \right\} \xrightarrow{\text{21} \rightsquigarrow \text{odd}}$$

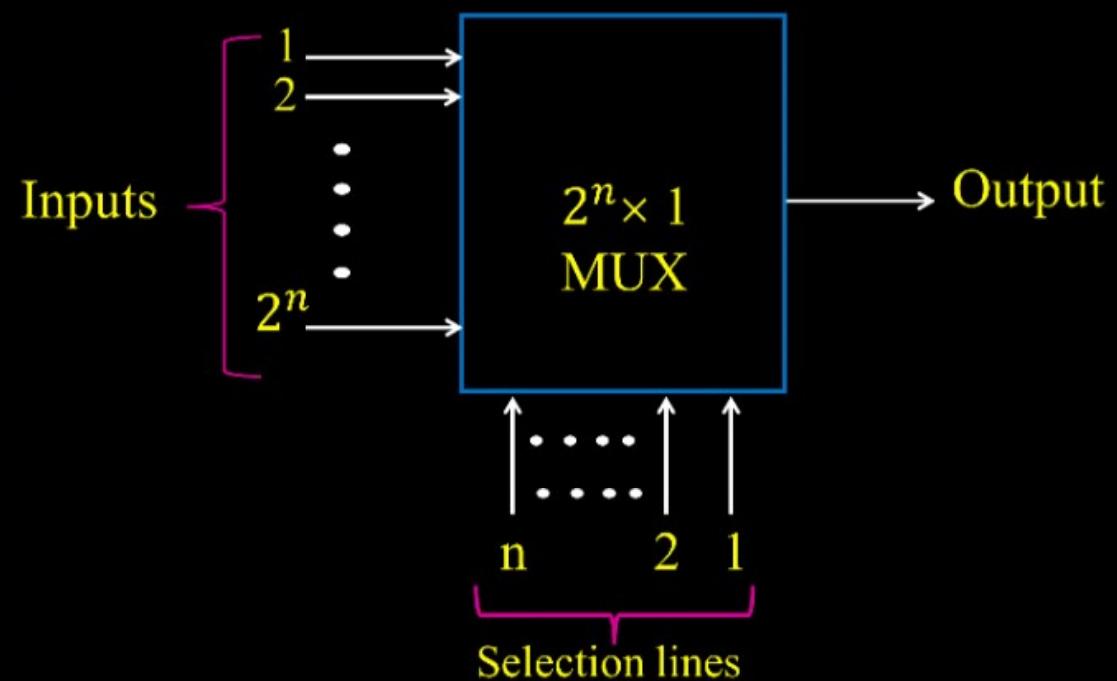
Multiplexer and De-Multiplexer

Multiplexer

Step 1) Definition

It is a Combinational Logic Circuit consists of “ 2^n ” inputs and “ Single ” output With “ n ” selection line inputs .

Step 2) Block Diagram

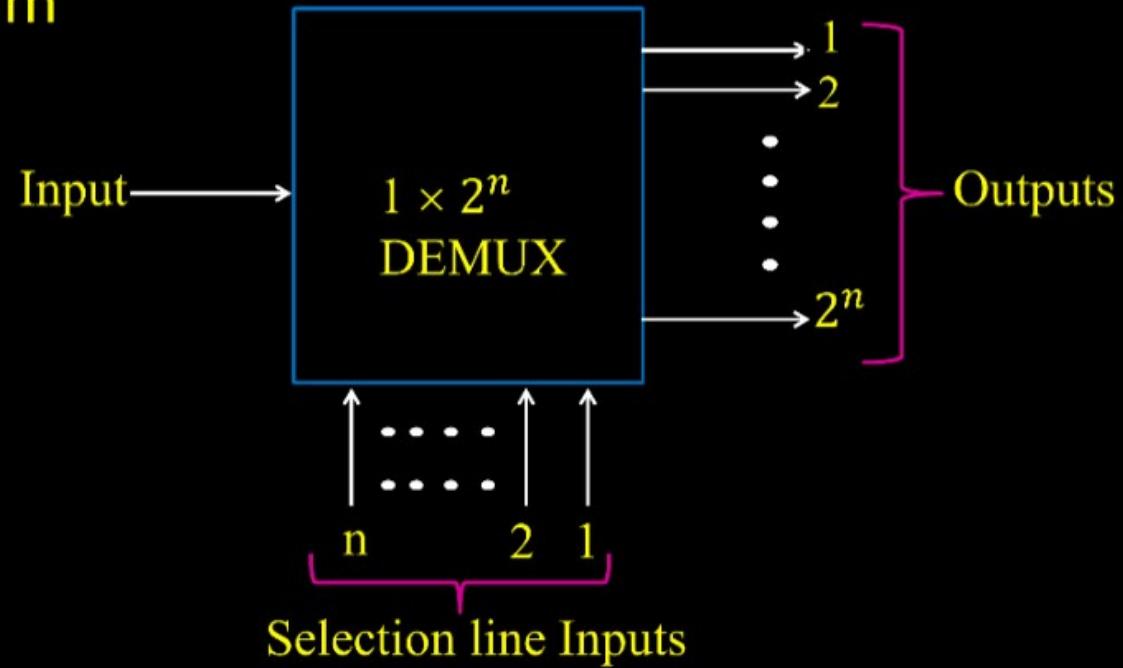


De-Multiplexer

Step 1) Definition

It is a Combinational Logic Circuit consists of “ Single “ input and “ 2^n “ outputs.

Step 2) Block Diagram



Multiplexer

1. Mux is called as Data Selector
2. Mux is used for Parallel to Serial data conversion

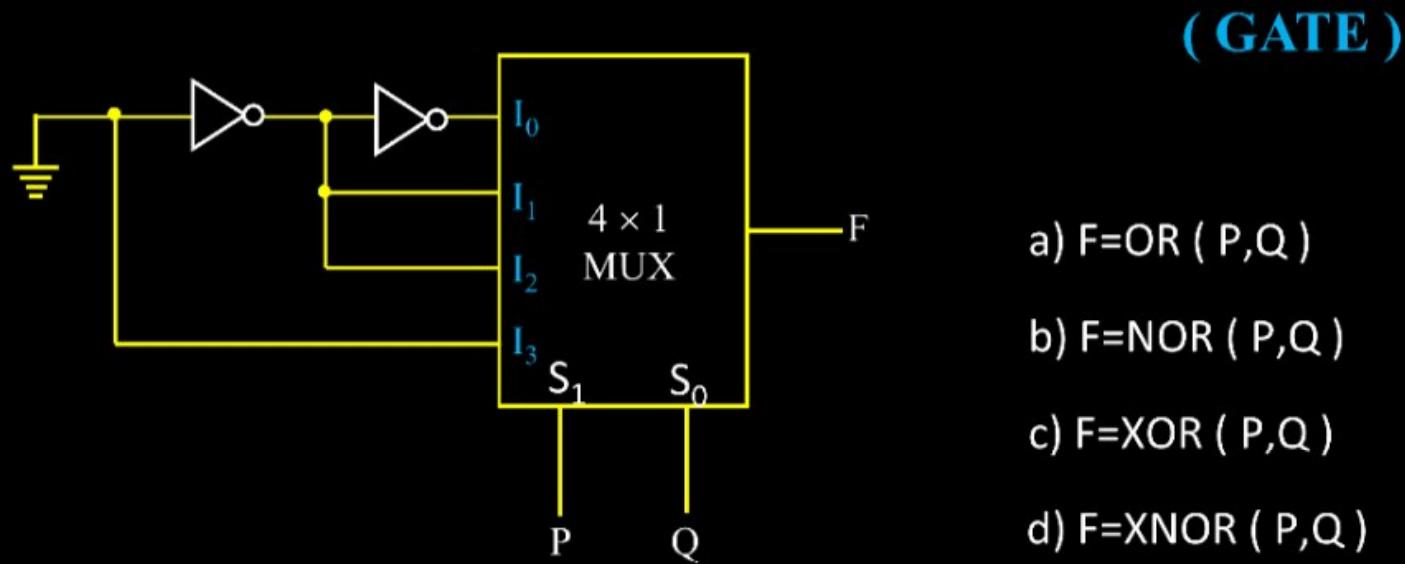
De-Multiplexer

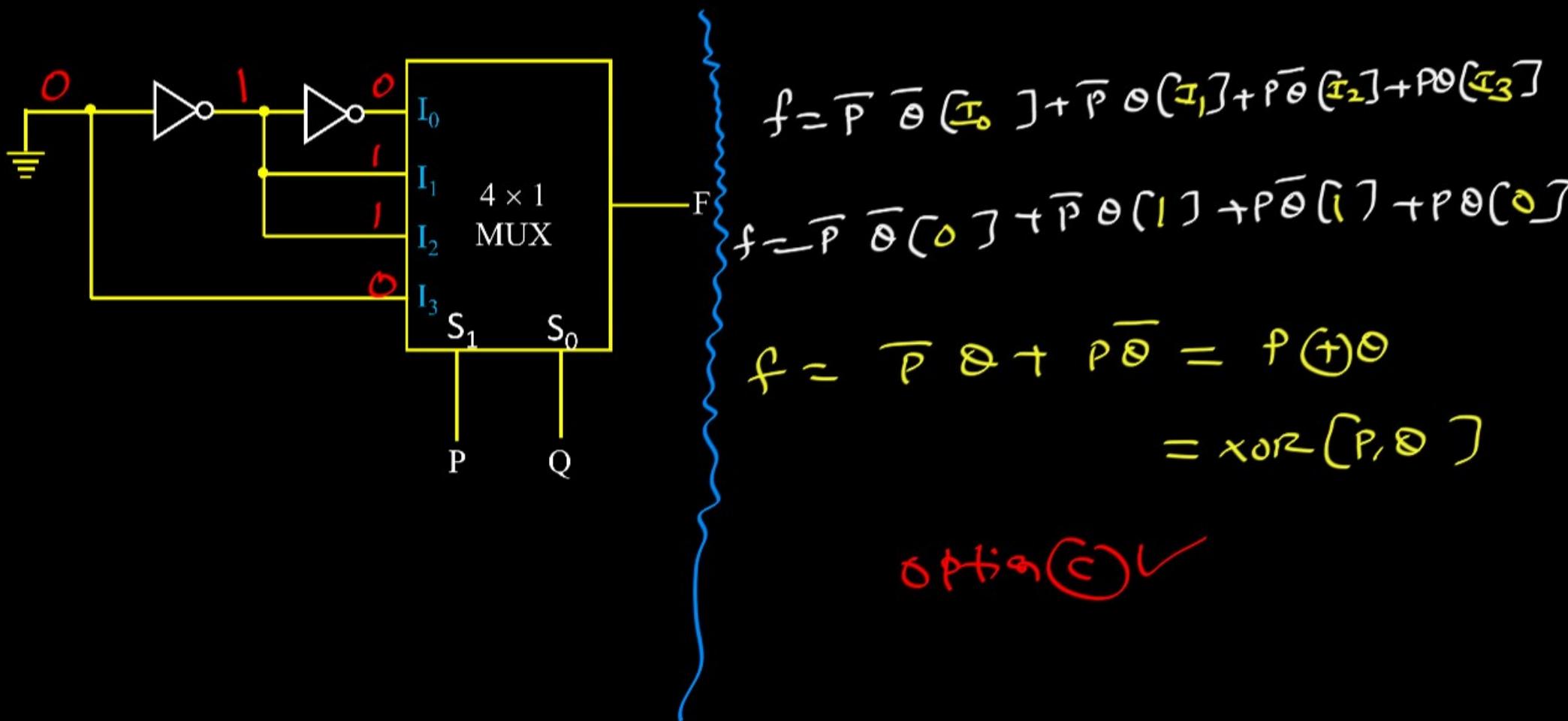
1. De-Mux is called as Data distributor
2. De-Mux is used for Serial to Parallel data conversion

Previous Questions

Previous Questions

Q. The logic function implemented by the circuit below is
(ground implies a logic “ 0 ”)

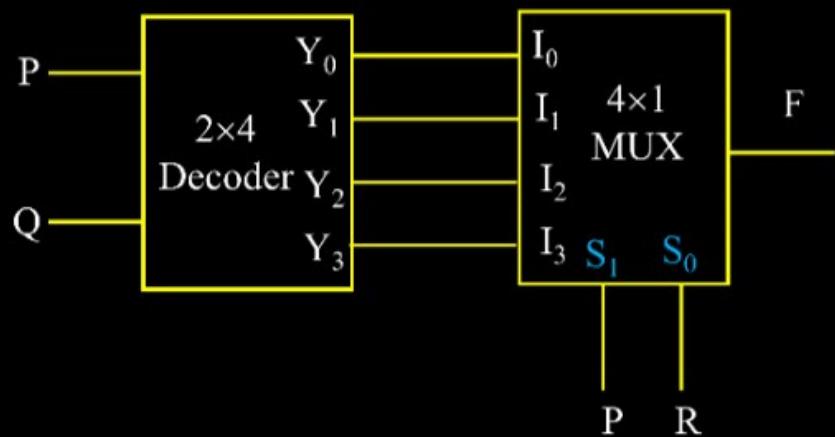




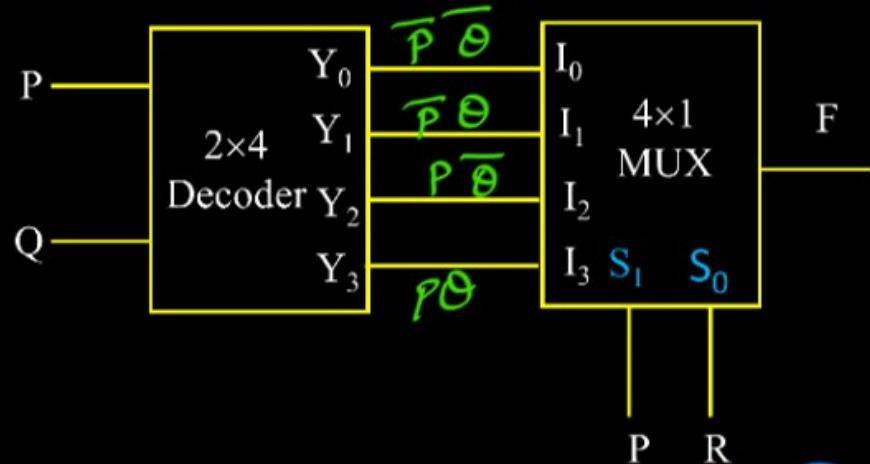
Multiple Select Question (MSQ)

Multiple Select Question (MSQ)

Q. The min term of $F(P,Q,R)$ are



- (A.) $\Sigma m (0, 3, 4, 7)$
- (B.) $\Pi M (0, 3, 4, 7)$
- (C.) $\Pi M (1, 2, 5, 6)$
- (D.) $\Sigma m (0, 3, 5, 7)$



$$f = \overline{P} \bar{R} [I_0] + \overline{P} R [I_1] + P \bar{R} [I_2] + PR [I_3]$$

$$f = \overline{P} \bar{R} [\overline{P} \bar{Q}] + \overline{P} R [\overline{P} Q] + P \bar{R} [P \bar{Q}] + PR [PQ]$$

$$f = \underbrace{\overline{P} \bar{Q} \bar{R}}_{\text{0 0 0}} + \underbrace{\overline{P} Q R}_{\text{0 1 1}} + \underbrace{P \bar{Q} \bar{R}}_{\text{1 0 0}} + \underbrace{P Q R}_{\text{1 1 1}}$$

$$f = \Sigma_m [0, 3, 4, 7] \rightarrow \text{option (A)}$$

$$f = \Pi M [1, 2, 5, 6] \rightarrow \text{option (C)}$$

Q. To Design 128×1 MUX, min number of 4×1 MUX required is?

MUX Shortcut Technique

Required input
Given input

Step1) $\frac{128}{4} = 32$

Step2) $\frac{32}{4} = 8$

Step3) $\frac{8}{4} = 2$

Step4) $\frac{2}{4} \approx 1$
43 \rightarrow odd

Q. To Design 1×64 De MUX, min number of 1×4 De MUX required is?

De-Mux Shortcut Technique

$\frac{\text{Required output}}{\text{Given output}}$

$$\left. \begin{array}{l} \text{Step 1: } \frac{64}{4} = 16 \\ \text{Step 2: } \frac{16}{4} = 4 \\ \text{Step 3: } \frac{4}{4} = 1 \\ \hline \text{21 } \rightsquigarrow \text{ odd} \end{array} \right\}$$

Chap^{to} 6

SEQUENTIAL LOGIC CIRCUITS



Introduction



Flip Flops



Race around Condition



Registers



Counters



Previous Questions

TIME IS FREE,
BUT IT'S PRICELESS

YOU CAN'T OWN IT,
BUT YOU CAN USE IT

YOU CAN'T KEEP IT,
BUT YOU CAN SPEND IT

ONCE YOU HAVE LOST IT,
YOU CAN NEVER GET IT BACK

Sequential Logic Circuits

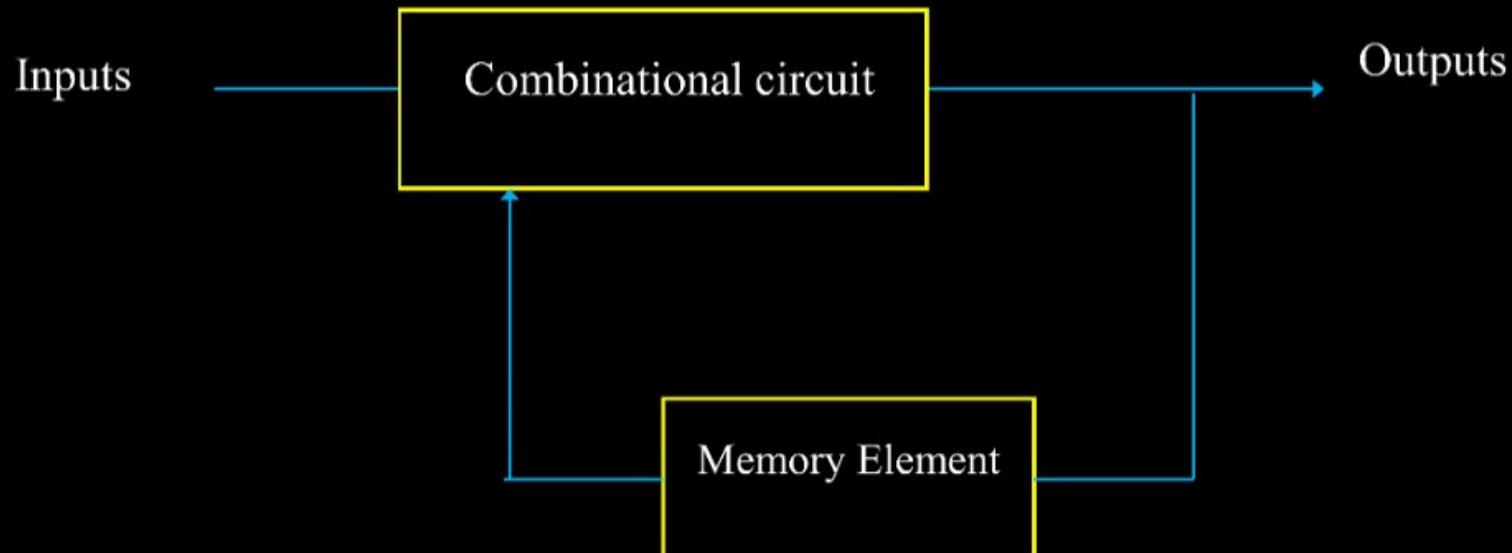
Definition

A circuit in which present output depends not only on present input but also on previous data is called Sequential Logic Circuit.

for Sequential Logic Circuit memory element is required to store previous data.

Sequential Logic Circuits

Block Diagram



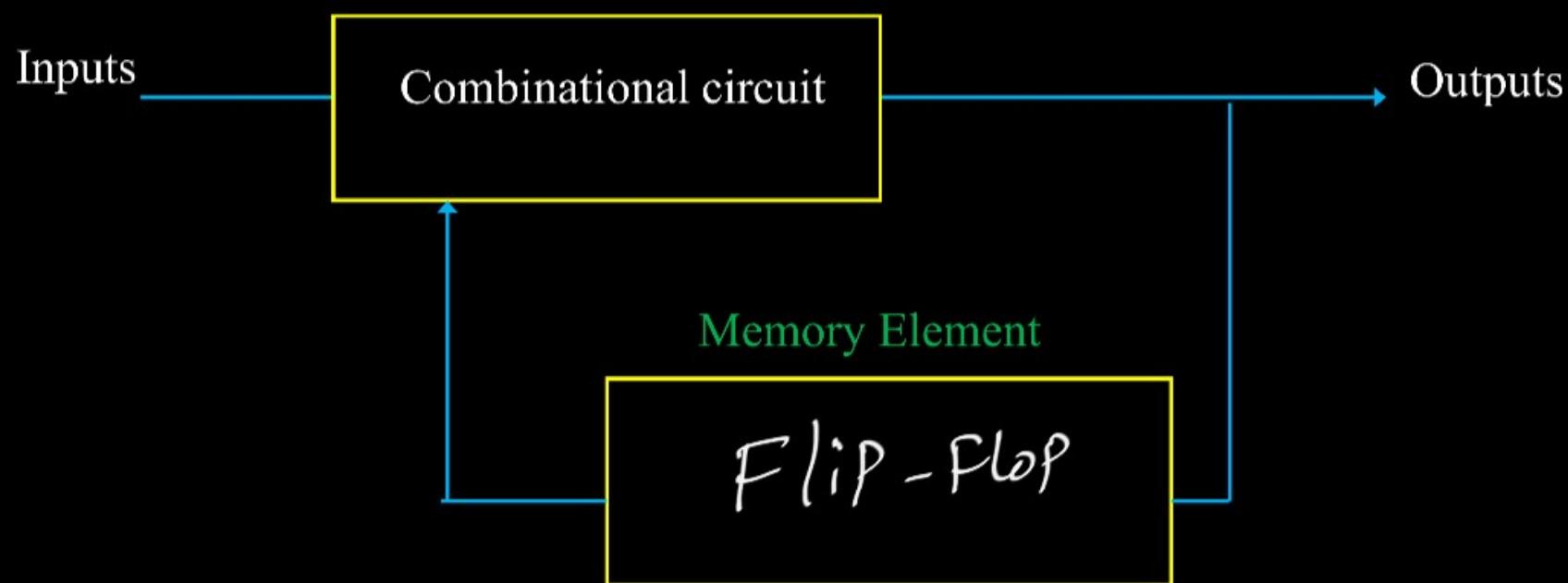
S L C = C L C + memory

Comparison Between Combinational and sequential

C.L.C	S.L.C
Faster (Advantage)	Slower (disadvantage)
Storing data is not possible Disadvantage	we can store the data. Advantage
Memory element is absent	Memory element is present.

Sequential Logic Circuits

Block Diagram



Sequential Logic Circuits

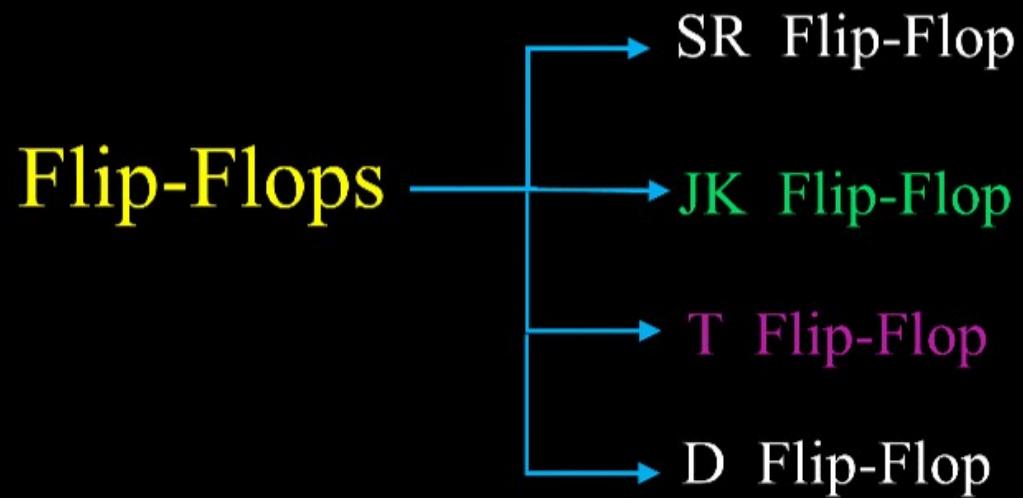
Examples

1. Flip-Flops
2. Registers
3. Counters

Flip - Flop

- It is a memory element used in sequential logic.
- It stores one bit data at a time.
- To store “ n “ bit data, we require “ n “ flip flops.

Types Of Flip - Flops



Types Of Flip - Flops

1) SR Flip-Flop

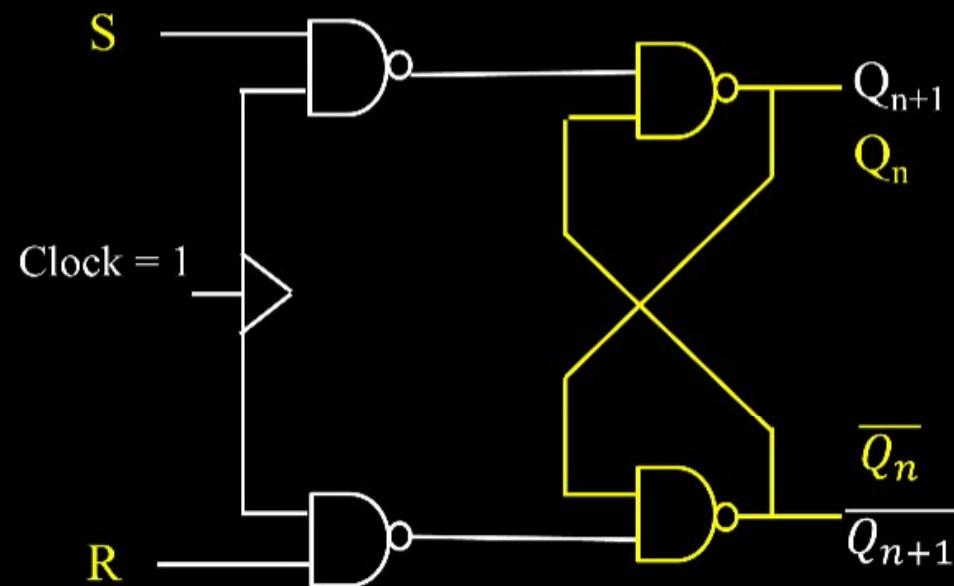
2) JK Flip-Flop

3) T Flip-Flop

4) D Flip-Flop

SR Flip-Flop

Step1) Circuit Diagram



Step2) Characteristic Table or Truth Table

SR Flip-Flop

Decimal	S	R	Q_n	Q_{n+1}	State
0	0	0	0	0	No Change
1	0	0	1	1	
2	0	1	0	0	Reset
3	0	1	1	0	
4	1	0	0	1	Set
5	1	0	1	1	
6	1	1	0	X	Invalid (or) Prohibited
7	1	1	1	X	

Step3) Output Expression

SR Flip-Flop

		00	01	11	10
		0	1	3	2
S	0				
	1	4	5	7	6
R	Q _n				

Step4) Excitation Table **or** Transition Table

SR Flip-Flop

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

Step2) Characteristic Table or Truth Table

SR Flip-Flop

Decimal	S	R	Q_n	Q_{n+1}	State
0	0	0	0	0	No Change
1	0	0	1	1	
2	0	1	0	0	Reset
3	0	1	1	0	
4	1	0	0	1	Set
5	1	0	1	1	
6	1	1	0	X	Invalid (or) Prohibited
7	1	1	1	X	

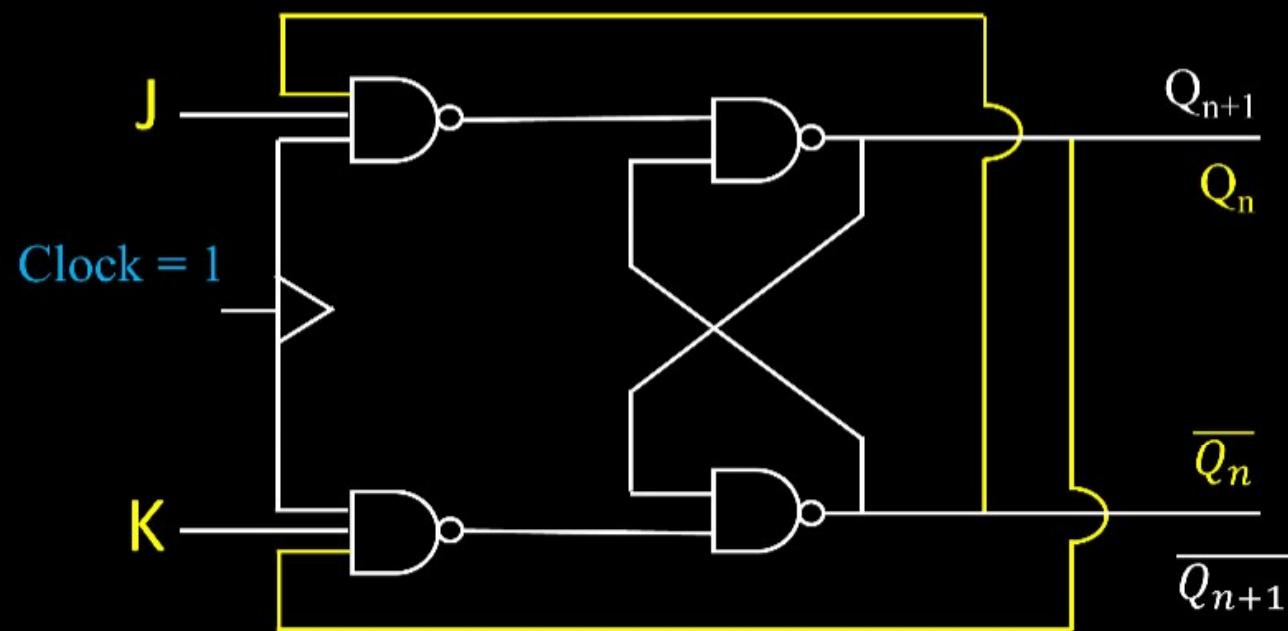
Step5) Simplified Truth table

SR Flip-Flop

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

JK Flip-Flop

Step1) Circuit Diagram



Step2) Characteristic Table or Truth Table

JK Flip-Flop

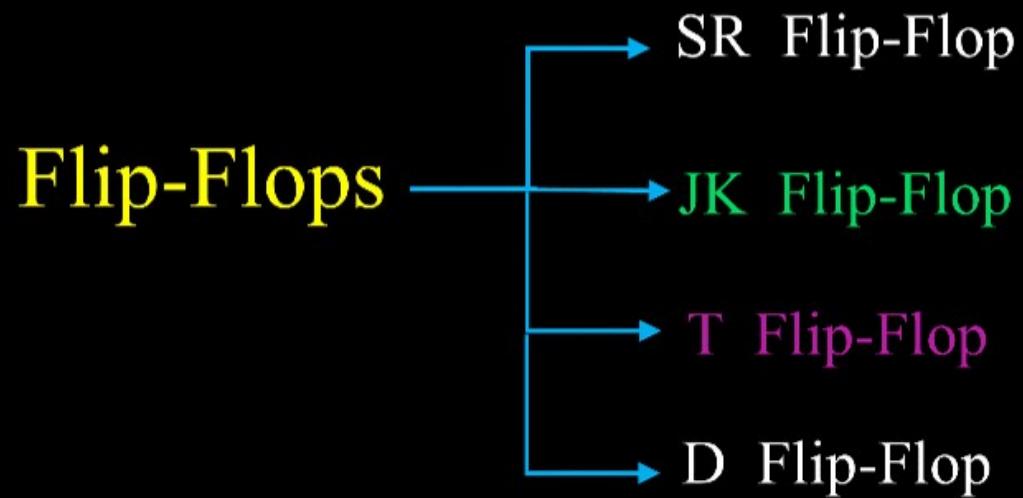
Decimal	J	K	Q_n	Q_{n+1}	State
0	0	0	0	0	No Change
1	0	0	1	1	
2	0	1	0	0	Reset
3	0	1	1	0	
4	1	0	0	1	Set
5	1	0	1	1	
6	1	1	0	1	Complement State
7	1	1	1	0	

Step5) Simplified Truth table

JK Flip-Flop

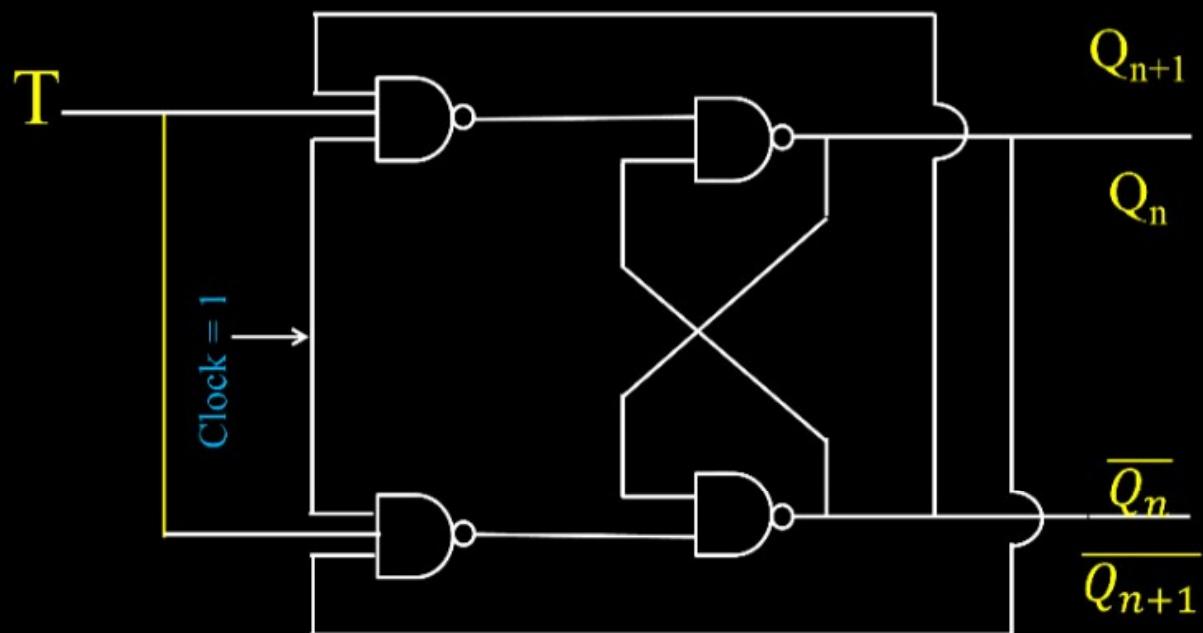
J	K	Q_{n+1}
0	0	Θ_n
0	1	0
1	0	1
1	1	$\overline{\Theta_n}$

Types Of Flip - Flops



T Flip-Flop

Step1) Circuit Diagram

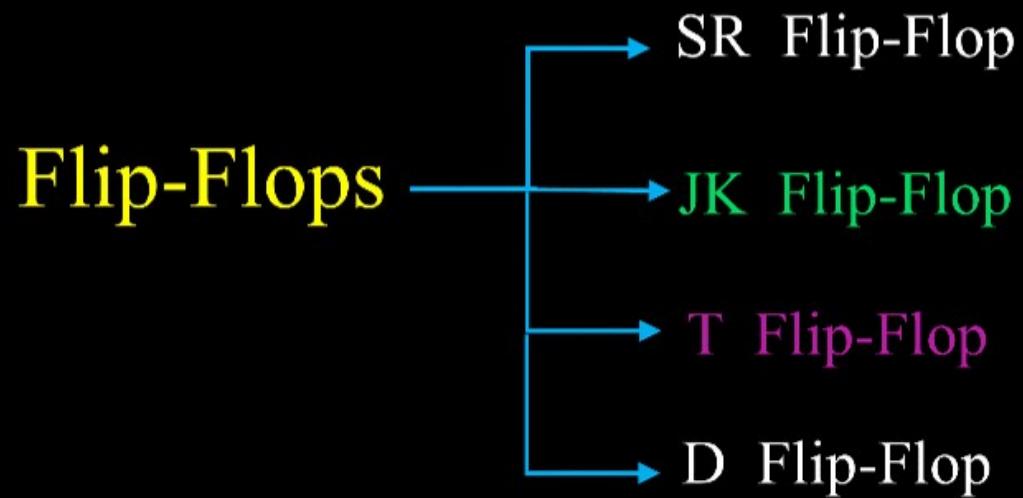


Step5) Simplified Truth table

T Flip-Flop

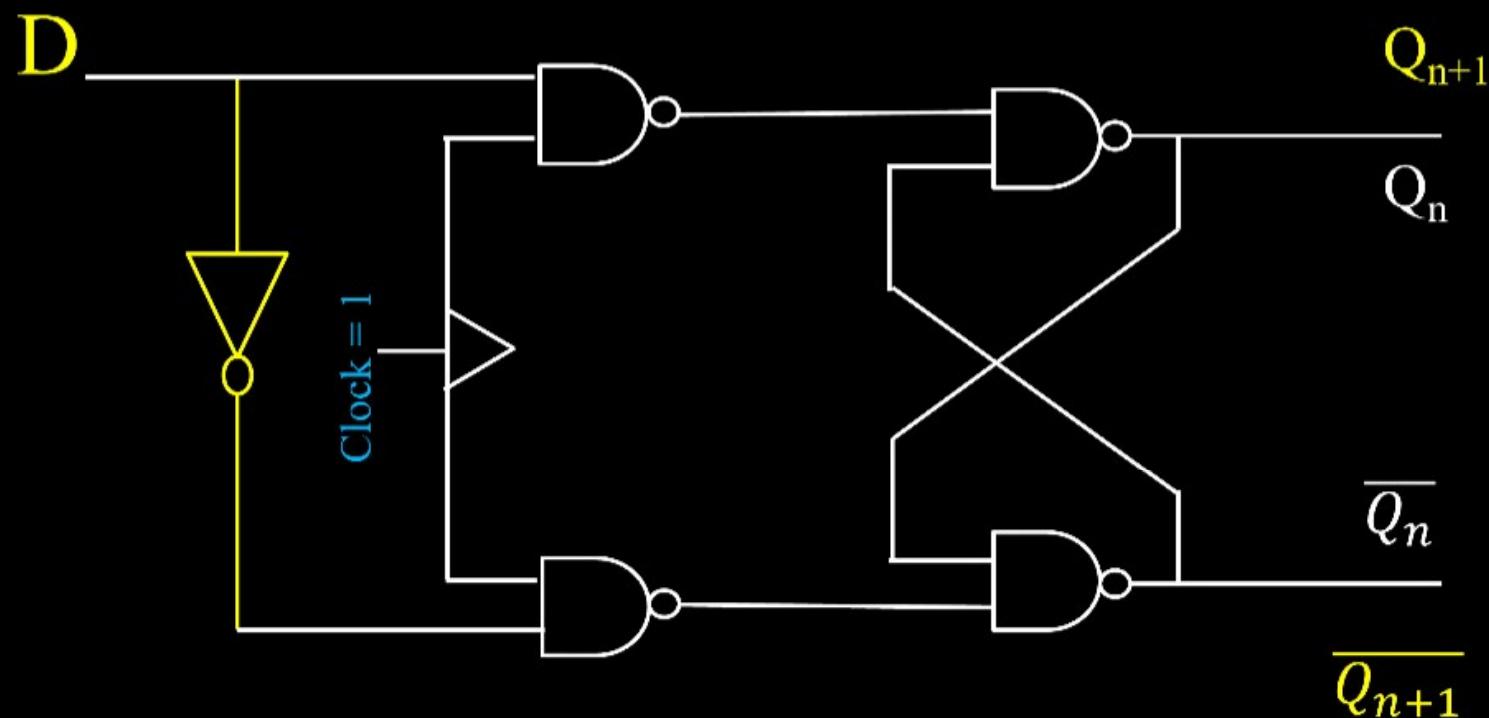
T	Q_{n+1}
0	θ_n
1	$\overline{\theta_n}$

Types Of Flip - Flops



D Flip-Flop

Step1) Circuit Diagram



Step2) Characteristic Table or Truth Table

D Flip-Flop

D	Q_n	Q_{n+1}	State
0	0	0	Reset
0	1	0	
1	0	1	Set
1	1	1	

Step5) Simplified Truth table

D Flip-Flop

D	Q_{n+1}
0	0
1	1

Summary

- ① $S R \text{ FF} \Rightarrow \Theta_{n+1} = S + \bar{R} \Theta_n$
- 2) $T K \text{ FF} \Rightarrow \Theta_{n+1} = T \bar{\Theta}_n + \bar{K} \Theta_n$
- ③ $T F \bar{F} \Rightarrow \Theta_{n+1} = T \Theta_n$
- 4) $D F F \Rightarrow \Theta_{n+1} = -D$

Clock

Clock is used to change the output of any circuit at a specified point o a triggering

Types of Clocks

Positive Edge triggered

Negative Edge triggered

Positive Level triggered

Negative Level triggered

Propagation Delay

It is the time required for an input to reach (propagate) the output

rACE around problem

Race around Condition

During one clock period output should change only one time but in JK flip flop when $J = 1$, $K = 1$ and if the clock pulse width is more, then the o/p Continuously keep on changes [toggles (or) racing] 0 to 1 and 1 to 0. This is called as Race around condition.

Race around Condition

Note: Racing is uncontrolled output whereas toggling is controlled output

Condition to avoid Race around

1. $t_{pw} < t_{pd} < T_{clock}$
2. Using edge triggered

Registers

Group of Flip - Flops are called as Registers

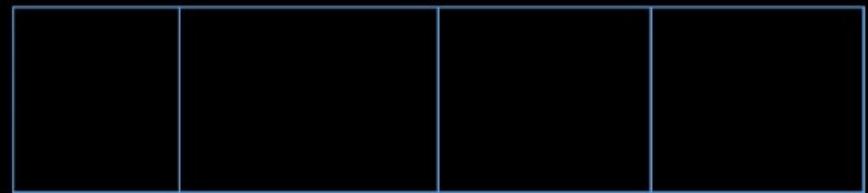
n-bit Register Consist of n- Flip - Flops and used to store n- bit data

Shift Register

A Register in which shifting of data is possible in one or both direction is called as Shift Register .

Types of Shift Registers

5) Parallel Shift in



6) Parallel Shift out



Previous Questions

Counters

A Counter is a Register that goes through a Predetermined Sequence of states

Modulus of a Counter

Modulus (MOD) of a Counter Indicates Number of different output states.

In MOD – N Counter, Number of different output states = N

Counters

Application

Frequency Division



MOD – N Counter is also called as Divided-by – N Counter

Relation between “ N ” & “ n ”

$$N \leq 2^n \text{ (or)} n \geq \log_2^N$$

Here N = Number of states

n = Number of flip flops

Previous Questions

Previous Questions

Q. The minimum number of D flip-flops needed to design a mod-258 counter is

(CSE-GATE)

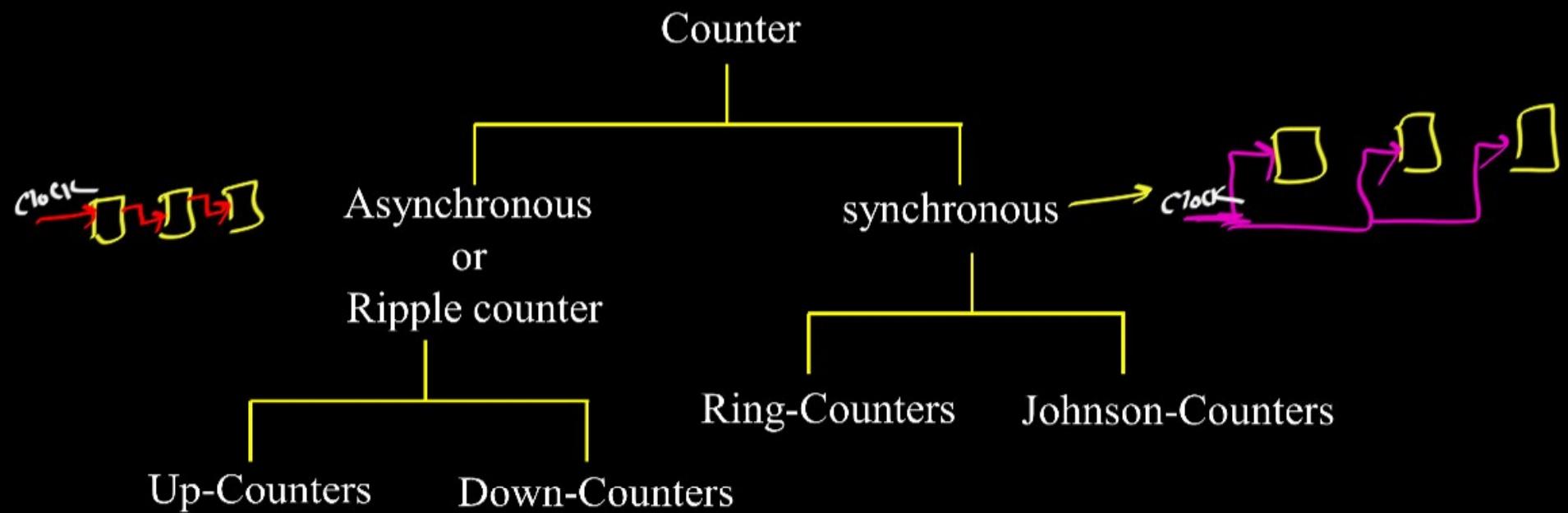
(a) 9

(c) 512

(b) 8

(d) 258

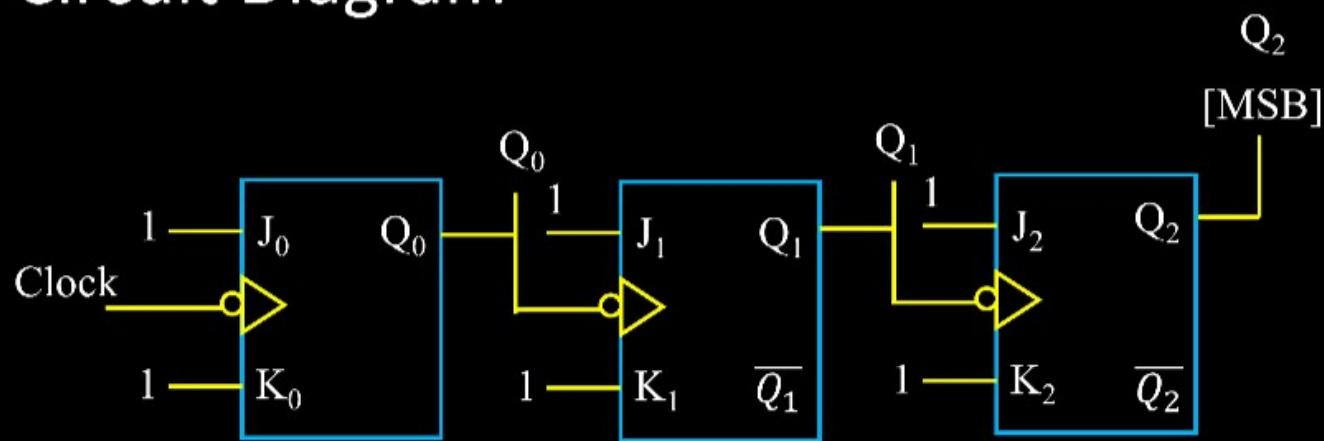
Types of a Counters



Asynchronous Up Counter

3 – bit Asynchronous Up Counter is as shown below

Circuit Diagram



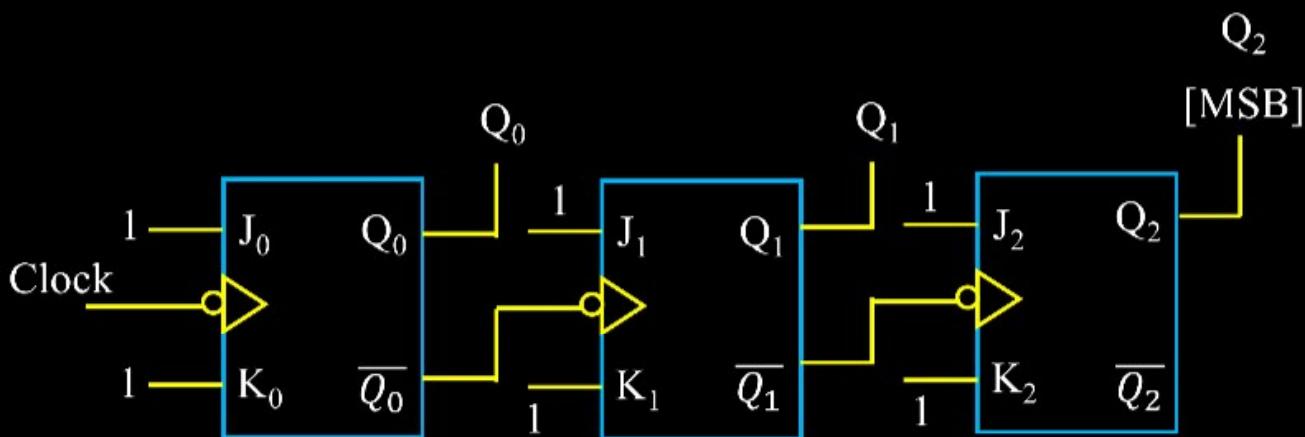
Up Counter Truth table

Clock	Q ₂	Q ₁	Q ₀
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
8	0	0	0

Asynchronous Down Counter

3 – bit Asynchronous Down Counter is as shown below

Circuit Diagram



Down Counter Truth table

Clock	Q ₂	Q ₁	Q ₀
0	0	0	0
1	1	1	1
2	1	1	0
3	1	0	1
4	1	0	0
5	0	1	1
6	0	1	0
7	0	0	1
8	0	0	0

Conclusion

3-bit Asynchronous Counter repeats same state after $8 = 2^3$ clock pulses.

n-bit Asynchronous Counter repeats same state after 2^n clock pulses.

MOD of n- bit Asynchronous Counter (N) == 2^n

Frequency in n-bit Asynchronous Counter

$$f = \frac{1}{n \times t_{pd}}$$

f = Clock frequency

t_{pd} = Propagation delay of each Flip-Flop

n = Number of flip flops

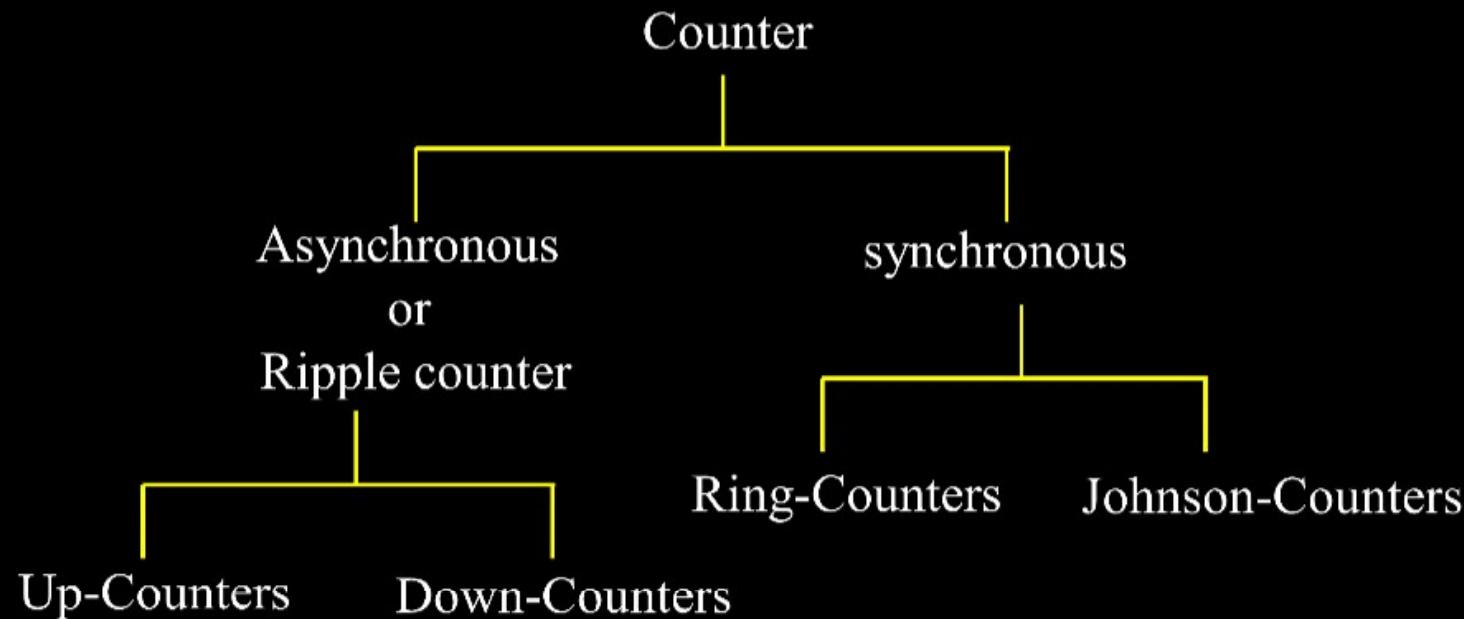
Previous Questions

A MOD 16 Ripple counter uses JK flip flop.

If the propagation delay of each FF is 50nsec,
then the maximum clock frequency is equal to

- (a) 20 MHz
- (b) 10 MHz
- (c) 5 MHz
- (d) 4 MHz

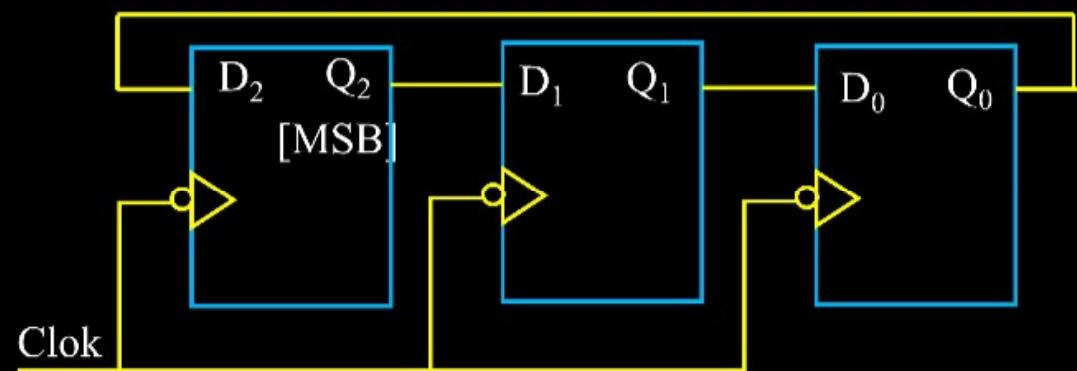
Types of a Counters



Ring Counter

3 – bit Ring Counter is as shown below

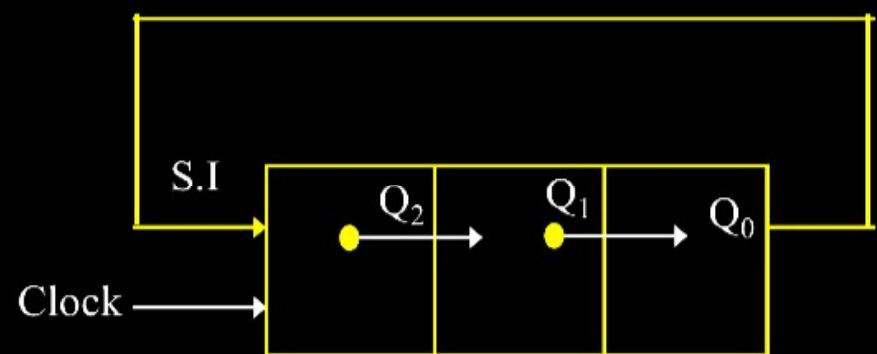
Circuit Diagram



Ring Counter

3 – bit Ring Counter is as shown below

Circuit Diagram



Conclusion

3-bit Ring Counter repeats same state after 3 Clock Pulses.

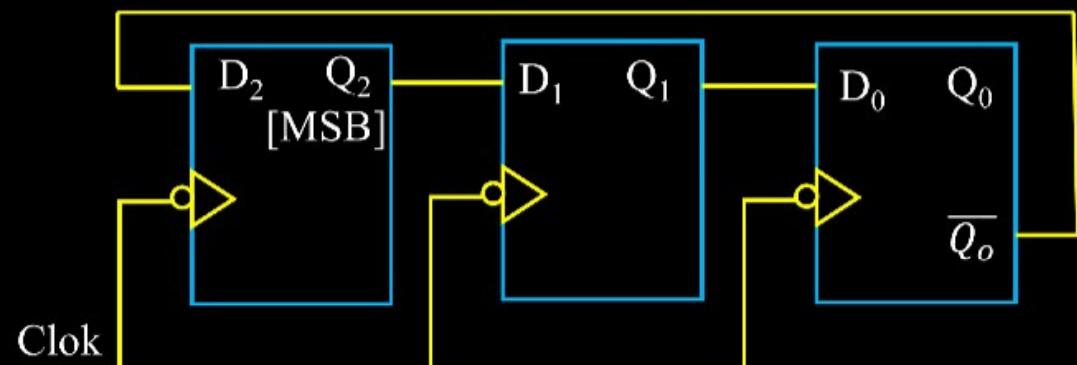
n-bit Ring Counter repeats same state after “ n ” Clock Pulses.

MOD of n- bit Ring Counter (N) == n

Johnson Counter

3 – bit Johnson Counter is as shown below

Circuit Diagram



Conclusion

3-bit Johnson Counter repeats same state after $6 = 2 \times 3$ Clock Pulses.

n-bit Johnson Counter repeats same state after “ $2 \times n$ ” Clock Pulses.

MOD of n- bit Johnson Counter (N) == $2 \times n$

Conclusions

MOD of n- bit Asynchronous Up & Down Counter (N) == 2^n

MOD of n- bit Ring Counter (N) == n

MOD of n- bit Johnson Counter (N) == $2 \times n$

Summary

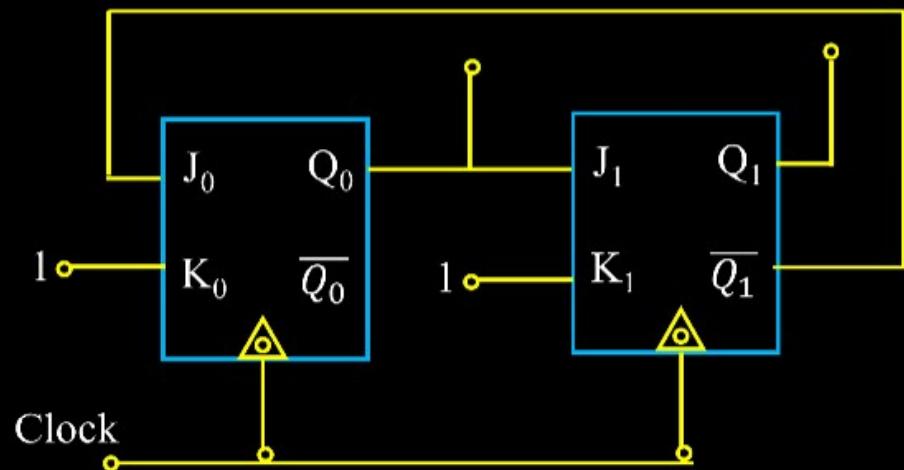
Counter Name	Repeats same State after	MOD	Output frequency
1) n-bit Asynchronous [for both up and down]	2^n clock pulses	$N = 2^n$	$f_0 = \frac{f_i}{N} = \frac{f_i}{2^n}$
2) n-bit Ring counter	n clock pulses	$N = 2^n$	$f_0 = \frac{f_i}{N} = \frac{f_i}{n}$
3) n-bit Johnson counter	2n clock Pulses	$N = 2 \times n$	$f_0 = \frac{f_i}{N} = \frac{f_i}{2 \times n}$

Counter Name	Repeats same State after	MOD	Output frequency
1) n-bit Asynchronous [for both up and down]	2^n clock pulses	$N = 2^n$	$f_0 = \frac{f_i}{N}$
2) n-bit Ring counter	n clock pulses	$N = n$	$f_0 = \frac{f_i}{N}$
3) n-bit Johnson counter	2n clock Pulses	$N = 2n$	$f_0 = \frac{f_i}{N}$

Previous Questions

Previous Questions

Q. Figure shows a MOD-K counter, the value of K is equal to
(Initially output = 00) (GATE)



- (a) 2 (b) 3 (c) 4 (d) 5

Procedure

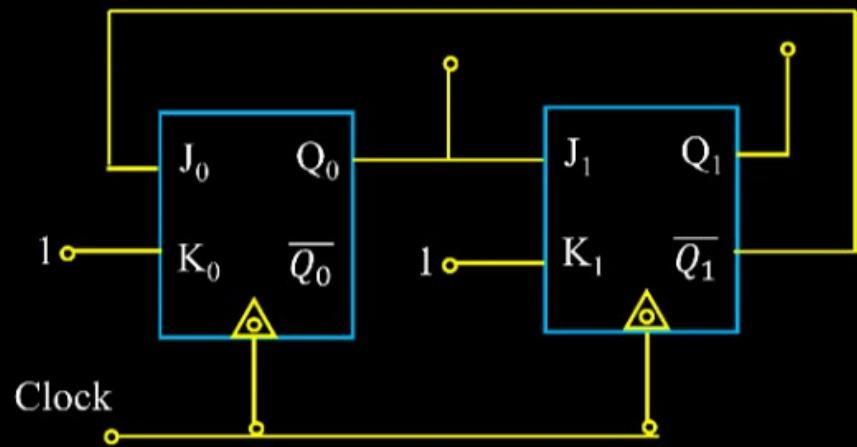
Step 1) write the given Flip Fop truth table

Step 2) write the relations from the given circuit

Step 3) write Input & output Relation table

Step 1) Given Flip Fop truth table

J	K	Q_{n+1}
0	0	Q_n
0	1	0
1	0	1
1	1	$\overline{Q_n}$



Step 2) Relations from given Circuit

Step 3) Input & output Relation table

$$\text{Given } J_0 = \overline{Q_1}$$

$$K_0 = 1$$

Clock	J_0	K_0	J_1	K_1	Q_0	Q_1
0	-	-	-	-	0	0
1	/	(0)	1	0
2	()	()	0	1
3	0	/	0)	0	0

$$J_1 = Q_0$$

$$K_1 = 1$$

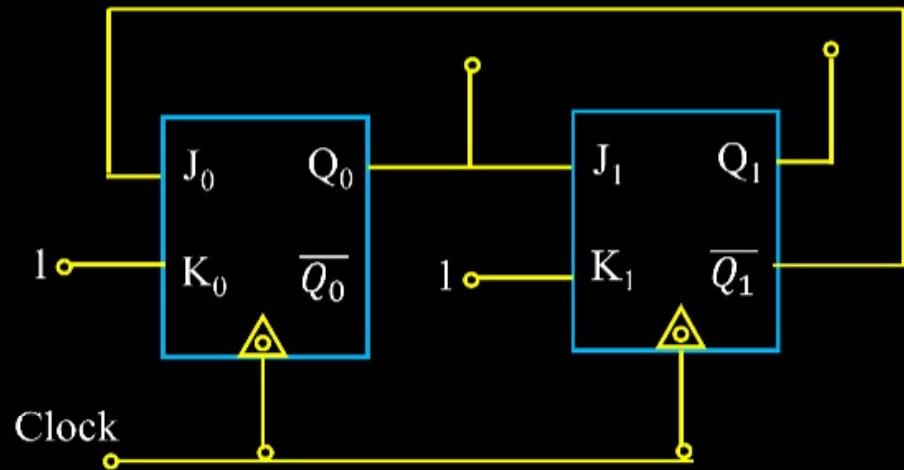
⇒ Initial State

Initial state is ⇒ hence
repeating It is
MOD-3 counter

Given Flip Fop truth table

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

Q. Figure shows a MOD-K counter, the value of K is equal to
(Initially output = 00) (GATE)



- (a) 2 (b) 3 (c) 4 (d) 5

Multiple Select Question (MSQ)

Multiple Select Question (MSQ)

Q. Which of the following is/are property of sequential logic circuits?

- (A.) A Flip-Flop is used to store 1-bit of Information
- (B.) Race around condition occurs in a JK Flip-Flop when $J = K = 1$
- (C) Master-slave configuration is used in Flip-Flop to store 2 bits of Information
- (D.) A transparent latch consists of a D-type of Flip-Flop

Gratitude

Ganeshwar

mail ID := ganeshwar412@gmail.com

mobile := 7900099555

Send what's up => your name, Branch,
Date writing like 23/08/24

complete or full notes

today's class => short notes:-

final word

No shortcut for success
↓
Hard work

Prepare Every day \Rightarrow 14 hours

Don't stop
when you're tired.

STOP

when you are **DONE**

Gnaneshwar

