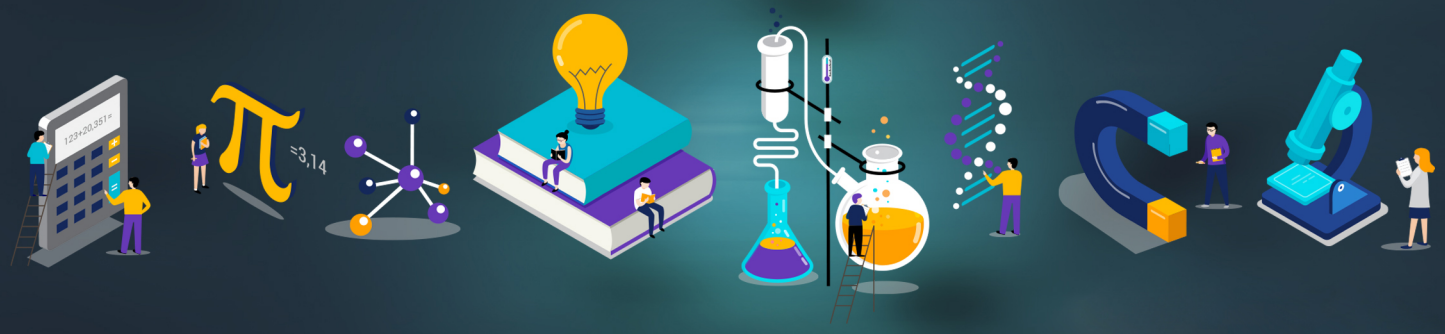


1. Digital Electronics

Digital electronics is essential to understanding the design and working of a wide range of applications. Digital systems contain devices that process the physical quantities represented in digital form.

1.1 Number system

Number System is a way to represent the numbers in the computer architecture. There are four different types of the number system,

1. Decimal number sysytem(Base 10)
2. Binary number system (base 2)
3. Octal number system(base 8)
4. Hexadecimal number system (base 16)

1.1.1 Decimal number system

In the decimal number system, is the radix-10 number system and therefore has 10 different digits or symbols. These are $0,1,2,3,4,5,6,7,8,9$. Each digit in the decimal system has a position and every digit is ten times more significant than the previous digit. Suppose, 25 is a decimal number, then 2 is ten times more than 5 . The place values of different digits in a mixed decimal number, starting from the decimal point, are $10^0, 10^1, 10^2$ and so on (for the integer part) and $10^{-1}, 10^{-2}, 10^{-3}$ and so on (for the fractional part). The value or magnitude of a given decimal number can be expressed as the sum of the various digits multiplied by their place values or weights.

■ **Example 1.1**

$$
\begin{aligned}
(92)_{10} &= 9\times 10^1 + 2\times 10^0 \\
(200)_{10} &= 2\times 10^2 + 0\times 10^1 + 0\times 10^0 \\
(30.2)_{10} &= 3\times 10^1 + 0\times 10^0 + 2\times 10^{-1} \\
(212.367)_{10} &= 2\times 10^2 + 1\times 10^1 + 2\times 10^0 + 3\times 10^{-1} + 6\times 10^{-2} + 7\times 10^{-3}
\end{aligned}
$$

■

1.1.2 Binary number system

The binary number system is a radix-2 number system with '0' and '1' as the two independent digits. Starting from the binary point, the place values of different digits in a mixed binary number are $2^0, 2^1, 2^2$ and so on (for the integer part) and $2^{-1}, 2^{-2}, 2^{-3}$ and so on (for the fractional part). The binary system is applied internally by almost all latest computers and computer-based devices because of it's direct implementation in electronic circuits using logic gates. Every digit is referred to as a bit.

■ **Example 1.2** 10101 is a five-bit binary number, 101 is a three-bit binary number,100001 is a six-bit binary number ■

Binary numbers			
Decimal Number	Binary Equivalent	Decimal Number	Binary Equivalent
1	1	11	1011
2	10	12	1100
3	11	13	1101
4	100	14	1110
5	101	15	1111
6	110	16	10000
7	111	17	10001
8	1000	18	10010
9	1001	19	10011
10	1010	20	10100

Table 1.1: Table of binary numbers

1.1.3 Decimal to binary conversion

Decimal to binary conversion is done by a method called double which is more popular for decimal to binary conversion in this method integers and fractions are handled seperately. Here the decimal numner is repeatdly divided by 2 and reminder after each division is taken in the reverse order.

Exercise 1.1 Convert the decimal number 25 into binary .

Divisor	Dividend	Remainder
2	25	–
2	12	1
2	6	0
2	3	0
2	1	1
–	0	1

$$(25)_{10} = (11001)_2$$ ■

Exercise 1.2 Convert the decimal number 125 into binary .

Divisor	Dividend	Remainder
2	125	–
2	62	1
2	31	0
2	15	1
2	7	1
2	3	1
2	1	1
–	0	1

$$(125)_{10} = (1111101)_2$$ ■

Exercise 1.3 Convert the decimal number 68 into binary .

Divisor	Dividend	Remainder
2	68	–
2	34	0
2	17	0
2	8	1
2	4	0
2	2	0
2	1	0
–	0	1

$$(68)_{10} = (1000100)_2$$

■

Conversion of decimal fraction to binary fraction

Instead of division, multiplication by 2 is carried out and the integer part of the result is saved and placed after the decimal point. The fractional part is again multiplied by 2 and the process repeated.

Exercise 1.4 Convert $(0.375)_{10}$ to binary fraction.

$$\begin{aligned}
\text{The fractional part } &= 0.375 \\
0.375 \times 2 &= 0.75 \quad \text{with a carry of 0} \\
0.75 \times 2 &= 0.5 \quad \text{with a carry of 1} \\
0.5 \times 2 &= 0 \quad \text{with a carry of 1} \\
\text{The binary equivalent of } (0.375)_{10} &= (.011)_2
\end{aligned}$$

■

Exercise 1.5 Convert $(0.68)_{10}$ to binary fraction.

$$\begin{aligned}
\text{The fractional part } &= 0.68 \\
0.68 \times 2 &= 1.36 \quad \text{with a carry of 1} \\
0.36 \times 2 &= 0.72 \quad \text{with a carry of 0} \\
0.72 \times 2 &= 1.44 \quad \text{with a carry of 1} \\
0.44 \times 2 &= 0.88 \quad \text{with a carry of 0} \\
\text{The binary equivalent of } (0.68)_{10} &= 0.1010\ldots.
\end{aligned}$$

■

Exercise 1.6 Convert the decimal number $7\frac{5}{8}$ into binary ■

Solution: $7\frac{5}{8}$ is written as 7.625
The binary number of the integer part 7 is 111. The decimal fraction 0.625 can be converted in to binary as before.

$$\begin{aligned}
0.625 \times 2 &= 1.250 \qquad \text{carry} = 1 \\
0.250 \times 2 &= 0.5 \qquad \text{carry} = 0 \\
0.5 \times 2 &= 1.0 \qquad \text{carry} = 1 \\
\therefore \left(7\tfrac{5}{8}\right) &= (111.101)_2
\end{aligned}$$

Exercise 1.7 Convert $\frac{1}{5} = 0.2$ in to binary. ■

Solution:

$$\begin{array}{lll} 0.2 \times 2 = 0.4 & \text{carry} & 0 \\ 0.4 \times 2 = 0.8 & \text{carry} & 0 \\ 0.8 \times 2 = 1.6 & \text{carry} & 1 \\ 0.6 \times 2 = 1.2 & \text{carry} & 1 \\ 0.2 \times 2 = 1.4 & \text{carry} & 0 \\ 0.4 \times 2 = 0.8 & \text{carry} & 0 \\ 0.8 \times 2 = 1.6 & \text{carry} & 1 \\ 0.6 \times 2 = 1.2 & \text{carry} & 1 \end{array}$$

The binary fractions is 0.00110011.....it never terminate.

1.1.4 Binary to Decimal Conversion

11011001 to decimal

$$\begin{aligned} 1 \times 2^0 &= 1 \times 1 = 1 \\ 0 \times 2^1 &= 0 \times 2 = 0 \\ 0 \times 2^2 &= 0 \times 4 = 0 \\ 1 \times 2^3 &= 1 \times 8 = 8 \\ 1 \times 2^4 &= 1 \times 16 = 16 \\ 0 \times 2^5 &= 0 \times 32 = 0 \\ 1 \times 2^6 &= 1 \times 64 = 64 \\ 1 \times 2^7 &= 1 \times 128 = 128 \\ 1 + 8 + 16 + 64 + 128 &= 217 \\ (11011001)_2 &= (217)_{10} \end{aligned}$$

Exercise 1.8 Convert 1101.1001 into decimal. According to the positional weight 1101.1001 can be written as ■

Solution:

$$\begin{aligned} 1101.1001 &= 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 2^0 \\ &\quad + 1 \times 2^{-1} + 0 \times 2^{-2} + 0 \times 2^{-3} + 1 \times 2^{-4} \\ &= 8 + 4 + 0 + 1 + \frac{1}{2} + 0 + 0 + \frac{1}{16} \\ &= 13\frac{9}{16} \quad \text{or} \quad 13.5625 \end{aligned}$$

Exercise 1.9 Convert the binary number 1011.11 into decimal. ■

Solution:

$$\begin{aligned}1011.11 &= 1\times 2^3 + 0\times 2^2 + 1\times 2^1 + 1\times 2^0 \\ &+ 1\times 2^{-1} + 1\times 2^{-2} \\ &= 11 + 0.5 + 0.25 = 11.75\end{aligned}$$

1.1.5 Binary addition

Adding two binary numbers will give us a binary number itself. It is the simplest method. Addition of two single digit binary number is given in the table below.

Binary Numbers	Addition	
0	0	0
0	1	1
1	0	1
1	1	0; Carry $\rightarrow$ 1

Exercise 1.10 Add the binary numbers 1110 and 1011 ■

Solution:

$$\begin{array}{ccccc} 1 & 1 & 1 & \text{carry} & \\ & 1 & 1 & 1 & 0 \\ + & 1 & 0 & 1 & 1 \\ \hline 1 & 1 & 0 & 0 & 1 \end{array}$$

Exercise 1.11 Add $8\frac{1}{4}$ and $6\frac{3}{4}$ in binary ■

Solution:

$$\begin{array}{ccccccc} & & & 1 & & 1 & \text{carry} \\ 1 & 0 & 0 & 0 & . & 0 & 1 \\ & 1 & 1 & 0 & . & 1 & 1 \\ \hline 1 & 1 & 1 & 1 & . & 0 & 0 \end{array}$$

$$\begin{aligned} 8\frac{1}{4} &= 8.25 = 1000.01_2 \\ 6\frac{3}{4} &= 6.75 = 110.11_2 \\ 8.25 + 6.75 &= 15 \\ \text{Answer is } &= (1111.00)_2 \end{aligned}$$

1.1.6 Binary substraction

Subtracting two binary numbers will give us a binary number itself. It is also a straightforward method. Subtraction of two single-digit binary number is given in the table below.

Binary Numbers	Subtraction	
0	0	0
0	1	1; Borrow 1
1	0	1
1	1	0

Exercise 1.12 Subtract 6 from 12 in binary system. ■

Solution:

Number Systems

$$(6)_{10} = (110)_2; (12)_{10} = (1100)_2$$

$$\begin{array}{rcccc} & 1 & 1 & 0 & 0 \\ - & & 1 & 1 & 0 \\ \hline & & 1 & 1 & 0 \end{array}$$

Exercise 1.13 Subtract 15 from 25 in the binary system ■

Solution:

$$(15)_{10} = (1111)_2; \quad (25)_{10} = (11001)_2$$

$$\begin{array}{rccccc} & 1 & 1 & 0 & 0 & 1 \\ - & & 1 & 1 & 1 & 1 \\ \hline & & 1 & 0 & 1 & 0 \end{array}$$

Exercise 1.14 Subtract $\left(9\frac{1}{2}\right)_{10}$ from $13\frac{3}{4}$ ■

Solution:

$$9\frac{1}{2}_{10} = 1001.1_2; 13\frac{3}{4}_{10} = 1101.11_2$$

$$\begin{array}{rccccccc} & 1 & 1 & 0 & 1 & . & 1 & 1 \\ - & 1 & 0 & 0 & 1 & . & 1 & \\ \hline & & 1 & 0 & 0 & . & 0 & 1 \end{array}$$

In the octal number system, we have the 7's and 8's complements. The 7's complement of a given octal number is obtained by subtracting each octal digit from 7 . For example, the 7 's complement of $(562)_8$ would be $(215)_8$. The 8 's complement is obtained by adding ' 1 ' to the 7 's complement. The 8 's complement of $(562)_8$ would be $(216)_8$ 1.7.4 Hexadecimal Number System The 15's and 16's complements are defined with respect to the hexadecimal number system. The 15's complement is obtained by subtracting each hex digit from 15. For example, the 15 's complement of $(3BF)_{16}$ would be $(C40)_{16}$. The 16's complement is obtained by adding ' 1 ' to the 15 's complement. The 16's complement of $(2AE)_{16}$ would be (D52) $)_{16}$.

1.2 Octal number system

The octal number system has a radix of 8 and therefore has eight distinct digits. The independent digits are 0,1 , $2, 3, 4, 5, 6$, and 7 . The next ten numbers that follow 7 , for example, would be $10, 11, 12, 13, 14, 15, 16, 17, 20$

and 21 . In fact, if we omit all the numbers containing the digits 8 or 9 or both from the decimal number system, we end up with octal number system. The place values for different digits in the octal number system are $8^0, 8^1$, 8^2 and so on (for the integer part) and $8^{-1}, 8^{-2}, 8^{-3}$ and so on (for the fractional part).

Octal Number	Binary Equivalent
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

Decimal to Octal number

To convert decimal to octal number, octal dabble method is used. In this method, the decimal number is divided by 8 each time, it yields or gives a remainder. The first remainder we get is the Least Significant digit(LSD) and the last remainder is the Most Significant Digit (MSD). Let us understand the conversion with the help example.

Exercise 1.15 Suppose 560 is a decimal number. Convert it into an octal number.

Divisor	Dividend	Remainder
8	560	–
8	70	0
8	8	6
8	1	0
–	1	1

$$(560)_{10} = (1060)_8$$ ■

Exercise 1.16 Convert 0.52 into an octal number. ■

Solution: The fraction part of the decimal number has to be multiplied by 8 .
$0.52 \times 8 = 0.16$ with carry 4
$0.16 \times 8 = 0.28$ with carry 1
$0.28 \times 8 = 0.24$ with carry 2
$0.24 \times 8 = 0.92$ with carry 1
So, for the fractional octal number, we read the generated carry from up to down. Therefore, 4121 is the octal number.

Octal to Decimal

To convert an octal number to decimal number we need to multiply each digit of the given octal with the reducing power of 8 .

Exercise 1.17 Suppose215_8 is an octal number, then it's decimal form will be? ■

Solution:

$$\begin{aligned} 215_8 &= 2 \times 8^2 + 1 \times 8^1 + 5 \times 8^0 \\ &= 2 \times 64 + 1 \times 8 + 5 \times 1 = 128 + 8 + 5 \\ &= 141_{10} \end{aligned}$$

1.3 Hexadecimal number system

The hexadecimal number system is a radix- 16 number system and its 16 basic digits are $0, 1, 2, 3, 4, 5, 6, 7, 8, 9$, A, B, C, D, E, and F. The place values or weights of different digits in a mixed hexadecimal number are 16^0, $16^1, 16^2$ and so on (for the integer part) and $16^{-1}, 16^{-2}$, 16^{-3} and so on (for the fractional part). The decimal equivalent of A, B, C, D, E and F are $10, 11, 12, 13, 14$ and 15 , respectively, for obvious reasons. Hexadecimal number system provides a condensed way of representing large binary numbers stored and processed inside the computer.

hexadecimal	Binary representation of Hexadecimal number
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
A	1010
B	1100
C	1101
D	1101
E	1110
F	1111

Decimal to hexadecimal

Exercise 1.18 Convert the decimal number 325 into hexadecimal

Divisor	Dividend	Remainder
16	325	–
16	20	5
16	1	4
–	1	1

$(325)_{10} = (145)_{16}$ ■

Exercise 1.19 Convert the decimal number 58 into hexadecimal

Divisor	Dividend	Remainder
16	58	–
16	3	A
–	3	3

$(58)_{10} = (3A)_{16}$ ■

Hexadecimal to decimal

Exercise 1.20

1.Convert $(256)_{16}$ in to decimal?

2.Convert $(4F)_{16}$ into decimal? ■

Solution:

$$
\begin{aligned}
1.\ (256)_{16} &= 2\times 16^2 + 5\times 16^1 + 6\times 16^0 \\
&= 512 + 80 + 6 \\
&= (598)_{10} \\
2.\ (4F)_{16} &= 4\times 16 + 15\times 16^0 \\
&= (79)_{10}
\end{aligned}
$$

Hexadecimal to binary

Since most of the microcomputer memories are organized into sets of bytes, hexadecimal numbers are very useful to represent the instruction code. A byte means 8 bits. The highest digit in hexadecimal is F which is 1111 in binary. A byte is grouped into two having 4 bits each. So hexadecimal is another short hand notation for binary.

$$7\text{E} = \underbrace{0111}_{7}\underbrace{1110}_{E}$$

Note

- A group of four bit is called a nibble.It is the half of an 8 bit byte.Since the highest hexadecimal has four bits, it stands for a nibble.

Conversion Table

Decimal	Octal	Hexadecimal	Binary
0	0	0	0
1	1	1	1
2	2	2	10
3	3	3	11
4	4	4	100
5	5	5	101
6	6	6	110
7	7	7	111
8	10	8	1000
9	11	9	1001
10	12	A	1010
11	13	B	1011
12	14	C	1100
13	15	D	1101
14	16	E	1110
15	17	F	1111

Table 1.2: Conversion table for Number system.

1.4 Boolean Algebra

Boolean algebra is mathematics of logic. It is one of the most basic tools available to the logic designer and thus can be effectively used for simplification of complex logic expressions. Boolean algebra too is composed of a set of symbols and a set of rules to manipulate these symbols.

- In Boolean algebra, the alphabetical symbols can take on either of the two values, that is, 1 and 0 (True or False).

- Boolean algebra captures the essential properties of both logic operations such as AND, OR and NOT and set operations such as intersection, union and complement.

Value Bit	Alternative Names
0	F, False, No, OFF, LOW
1	T, True, Yes, ON, High

1.4.1 Boolean Operations

Boolean algebra operates with three fuctional operator-The building block of digital logic design complement, OR, and AND. These building block are comparable to taking the negative, adding and multiplying in ordinary algebra.

Operator Name	Alternate Name	Example	Altemate Representations
NOT	Complement inversion	$\bar{X}$	X'
OR	Union logical addition	$X+Y$	$X\bigcup Y, X\vee Y, X\vee Y$
AND	Intersection logical multiplication	$X\cdot Y$	$X\cdot Y, X\bigcap Y, X\&y$

Input		Output		
X	Y	NOT $(\bar{X})$	O R $(X+Y)$	AND $(X\cdot Y)$
0	0	1	0	0
0	1	1	1	0
1	0	0	1	0
1	1	0	1	1

Table 1.3: Truth Table

1.4.2 The important postulates of Boolean algebra

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

2. $1\cdot 0=0\cdot 1=0 \quad , \quad 0+1=1+0=1$

3. $0\cdot 0=0 \quad , \quad 1+1=1$

4. $\overline{1}=0 \quad , \quad \overline{0}=1$

1.4.3 Boolean Identities

Boolean expression can be simplified but we need identities or laws, that apply to boolean algebra instead of regular algebra.

Identity Name	AND Form	OR Form
Identity Law	$1 \cdot X = X$	$0 + X = X$
Null (or Dominance) Law	$0 \cdot X = 0$	$1 + X = 1$
Idempotent Law	$X \cdot X = X$	$X + X = X$
Inverse Law	$X\bar{X} = 0$	$X + \bar{X} = 1$
Commutative Law	$X \cdot Y = Y \cdot X$	$X + Y = Y + X$
Associative Law	$(X \cdot Y)Z = X(Y \cdot Z)$	$(X + Y) + Z = X + (Y + Z)$
Distributive Law	$(X + Y) \cdot Z = (X + Y) \cdot (X + Z)$	$X \cdot (Y + Z) = X \cdot Y + X \cdot Z$
Absorption Law	$X \cdot (X + Y) = X$	$X + X \cdot Y = X$
DeMorgan's Law	$(\overline{X \cdot Y}) = \bar{X} + \bar{Y}$	$(\overline{X + Y}) = \bar{X} \cdot \bar{Y}$
Double Complement Law	$\bar{\bar{X}} = X$	

Dual of a Boolean Expression

The dual of a Boolean expression is obtained by replacing all '.' by '+' operations, all '+' operations by '.' operations, all 0's by 1's, all 1's by 0's and leaving all literals unchanged.

$$\text{Given Boolean expression : } \bar{A} \cdot B + A \cdot \bar{B}$$
$$\text{Corresponding dual expression : } (\bar{A} + B) \cdot (A + \bar{B})$$

Exercise 1.21 Use boolean algebra techniques, simplify this expresssion? ■

Solution:

$$\begin{aligned} AB + AB + AC + B &= AB + AC + B \\ B(B + C) &= B \\ AB + AB &= AB \\ &= AB + B + AC \\ &= B + AC \end{aligned}$$

Exercise 1.22 Simplify the boolean expressions

1. $A\bar{B} + A(B + C) + B(B + C)$
2. $\left[A\bar{B}(C + BD) + \bar{A}\bar{B}\right] C$

■

Solution:

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

$$\begin{aligned} & A\bar{B} + AB + AC + B && B(B + C) = B \\ & A\bar{B} + AC + AB + B && AB + B = B(A + 1) = B \\ & A\bar{B} + AC + B \end{aligned}$$

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

2. $$\left[A\bar{B}(C+BD)+\bar{A}\bar{B}\right]C=[A\bar{B}C+A\bar{B}BD]C+\bar{A}\bar{B}C \qquad B\bar{B}=0$$
$$=A\bar{B}CC+\bar{A}\bar{B}C \qquad (A+\bar{A})=1$$
$$=A\bar{B}C+\bar{A}\bar{B}C$$
$$=\bar{B}C(A+\bar{A})$$
$$=\bar{B}C$$

Exercise 1.23 Expression $A+\bar{A}B+\bar{A}\bar{B}C+\bar{A}\bar{B}\bar{C}D+\bar{A}\bar{B}\bar{C}\bar{D}E$ would be similar to ■

Solution:

$$=A+\bar{A}B+\bar{A}\bar{B}C+\bar{A}\bar{B}\bar{C}(D+\bar{D}E)$$
$$=A+\bar{A}B+\bar{A}\bar{B}C+\bar{A}\bar{B}\bar{C}(D+E)$$
$$=A+\bar{A}B+\bar{A}\bar{B}\left(C+\bar{C}(D+E)\right) \qquad C+\bar{C}(D+E)=C+D+E$$
$$=A+\bar{A}B+\bar{A}\bar{B}(C+D+E) \qquad B+\bar{B}(C+D+E)=B+C+D+C$$
$$=A+\bar{A}(B+\bar{B})(C+D+E)$$
$$=A+\bar{A}(B+C+D+C)$$
$$=A+B+B+D+C \qquad [\therefore A+\bar{A}B=A+B]$$

1.5 Demorgan's Law

This law provides an easy way of finding the complement of a boolean function. Demorgan's theorem provides mathematical verification of the equivalency of the NAND and negative OR gates and the equivalency of the NOR and negative AND gates.

First Law

The compliment of a product of variables is equal to the sum of the complement of the variables.
Or,
The compliment of two or more ANDed variables is equivalent to the OR of the compliment of the individual variables.

Formula

$$\overline{X\cdot Y}=\overline{X}+\overline{Y}$$

Truth Table

X	Y	$\overline{X}$	$\overline{Y}$	XY	$\overline{XY}$	$\overline{X}+\overline{Y}$
0	0	1	1	0	1	1
1	0	0	1	0	1	1
0	1	1	0	0	1	1
1	1	0	0	1	0	0

Second Law

The complement of a sum of variable is equals to the prodict of the complement of the variables.
Or,
The complement of two or ORed variables is equivalent to the AND of the complement of the individual variables.

$$\overline{X+Y} = \bar{X}\bar{Y} \tag{1.1}$$

Truth Table

X	Y	$\overline{X}$	$\overline{Y}$	X+Y	$\overline{X+Y}$	$\bar{X}\bar{Y}$
0	0	1	1	0	1	1
1	0	0	1	1	0	0
0	1	1	0	1	0	0
1	1	0	0	1	0	0

Note

- $\overline{XYZ} = \bar{X} + \bar{Y} + \bar{Z}$
- $\overline{X+Y+Z} = \bar{X}\bar{Y}\bar{Z}$
- $\overline{WXYZ} = \bar{W} + \bar{X} + \bar{Y} + \bar{Z}$
- $\overline{W+X+Y+Z} = \bar{W}\bar{X}\bar{Y}\bar{Z}$

Exercise 1.24 Solve $\overline{\overline{A+B\overline{C}}+D(\overline{E+\overline{F}})}$ ■

Solution:

$$\overline{\overline{A+B\overline{C}}+D(\overline{E+\overline{F}})}$$
$$\overline{(A+B\overline{C})}\ \overline{(D(\overline{E+\overline{F}}))}$$
$$(A+B\overline{C})(D+\overline{(\overline{E+\overline{F}})})$$
$$(A+B\overline{C})(\overline{D}+E+\overline{F})$$

Exercise 1.25 Apply Demorgan's theorems to each of the following expressions.

a). (A+B+C)D **b).** ABC+DEF **c).** $A\overline{B}+C\overline{D}+EF$ ■

Solution:

a). $$(A+B+C)D = \left(\overline{A+B+C}\right)+\overline{D}$$
$$= \overline{A}\,\overline{B}\,\overline{C}+\overline{D}$$

b). $$= \left(\overline{ABC}\right)\left(\overline{DEF}\right)$$
$$= \left(\overline{A}+\overline{B}+\overline{C}\right)\left(\overline{D}+\overline{E}+\overline{F}\right)$$

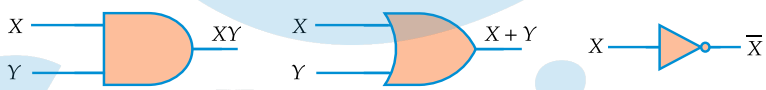
c). $$A\overline{B}+C\overline{D}+EF = \left(\overline{A\overline{B}}\right)\left(\overline{C\overline{D}}\right)\left(\overline{EF}\right)$$
$$= \left(\overline{A}+B\right)\left(\overline{C}+D\right)\left(\overline{E}+\overline{F}\right)$$

1.6 Logic Gates

- We see that Boolean functions are implemented in digital computer circuits called gates
- A gate is an electronic device that produces a result based on two or more input values.
- In reality, gates consist of one to six transistors, but digital designers think of them as a single unit.
- Integrated circuits contain collections of gates suited to a particular purpose

1.6.1 Symbols for Logic Gates AND,OR and NOT

- The three simplest gates are the AND, OR, and NOT gates



X AND Y

X	Y	XY
0	0	0
0	1	0
1	0	0
1	1	1

X OR Y

X	Y	X + Y
0	0	0
0	1	1
1	0	1
1	1	1

NOT X

X	$\overline{X}$
0	1
1	0

Figure 1.1

1.6.2 XOR Gate

- Another very useful gate is the exclusive OR (XOR) gate.
- The output of the XOR operation is true only when the values of the inputs differ.

X XOR Y

X	Y	$X \oplus Y$
0	0	0
0	1	1
1	0	1
1	1	0

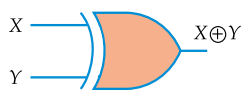


Figure 1.2

Equivalent Circiut of XOR Gate

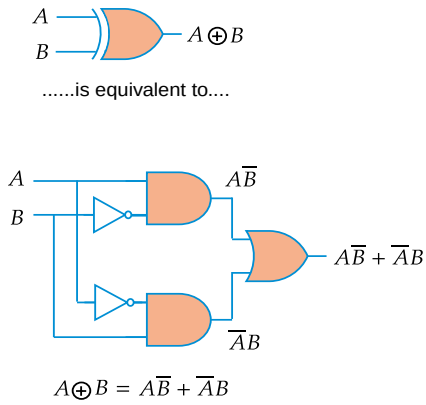


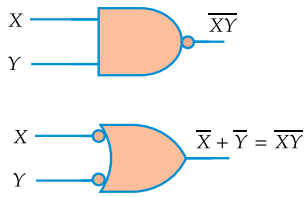
Figure 1.3

1.6.3 Universal gates NAND and NOR

- Two other common gates are NAND and NOR, which produce complementary output to AND and OR.

X NAND Y

X	Y	X NAND Y
0	0	1
0	1	1
1	0	1
1	1	0



X NOR Y

X	Y	X NOR Y
0	0	1
0	1	0
1	0	0
1	1	0

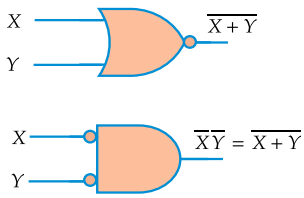


Figure 1.4

- NAND and NOR are known as universal gates because they are inexpensive to manufacture and any Boolean function can be constructed using only NAND or only NOR gates.

1.6.4 Basic gates using NAND gates

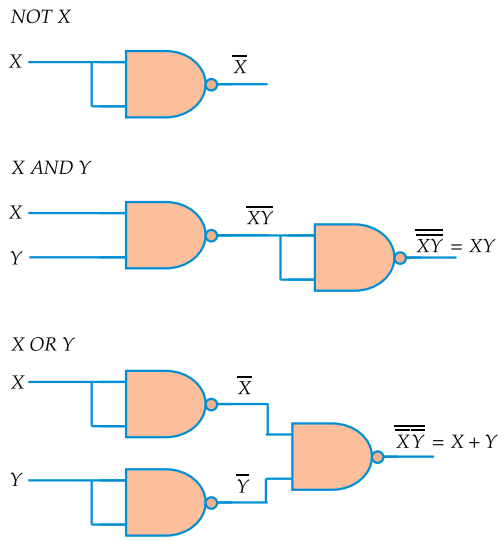


Figure 1.5

1.6.5 Basic gates using NOR gates

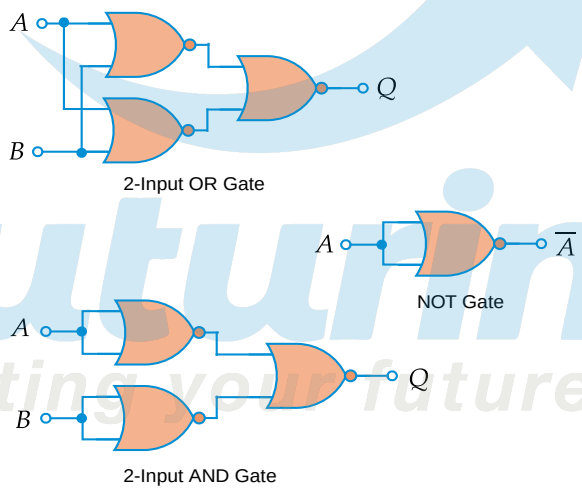


Figure 1.6

1.6.6 Multiple Input Gates

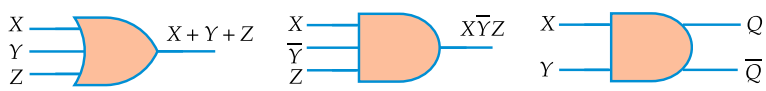
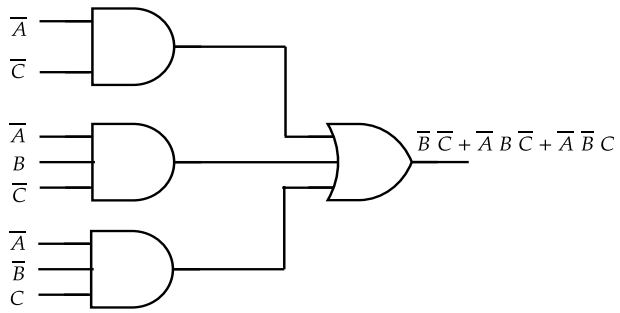


Figure 1.7

Exercise 1.26 Simplify the expression $A\bar{B}\bar{C}+\bar{A}\bar{B}\bar{C}+\bar{A}B\bar{C}+\bar{A}\bar{B}C$ and draw block diagram of the circuit using OR and AND gate ■

Solution:

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



Exercise 1.27 Simplify the expression $(A+B)(B+C)(A+C)$ and draw the block diagram using OR and AND gates, ■

Solution:

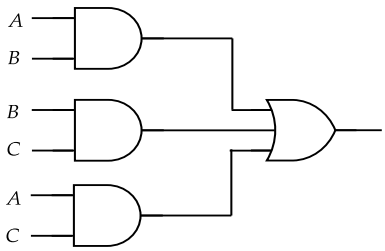
The first two terms are

$$\begin{aligned}(A+B)(B+C) &= AB+AC+BB+BC\\ &= AB+AC+B+BC\\ &= AB+AC+B(1+C) = AB+AC+B\\ &= B(A+1)+AC = B+AC\end{aligned}$$

Multiply logically with the third term

$$\begin{aligned}(B+AC)(A+C) &= BA+BC+ACA+ACC\\ &= BA+BC+AC+AC\\ &= BA+BC+AC\end{aligned}$$

The circuit diagram is given below



Exercise 1.28 Complement the following expressions

a. $A+BC+AB$

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

c. $AB(\bar{C}D+\bar{B}C)$

■

Solution:

(a) complement of $A+BC+AB$ equals $\overline{A+BC+AB}$

$$\begin{aligned}
\overline{A+BC+AB} &= \bar{A}(\bar{B}+\bar{C})\overline{(A+\bar{B})} \\
&= \overline{(A\bar{B}+\bar{A}\bar{C})}\overline{(A+\bar{B})} \\
&= \bar{A}\bar{B}\bar{A}+\bar{A}\bar{B}\bar{B}+\bar{A}\bar{C}\bar{A}+\bar{A}\bar{C}\bar{B} \\
&= \bar{A}\bar{B}+\bar{A}\bar{B}+\bar{A}\bar{C}+\bar{A}\bar{C}\bar{B} \\
&= \bar{A}\bar{B}+\bar{A}\bar{C}+\bar{A}\bar{C}\bar{B} \\
&(\text{ since } \bar{A}\bar{B}+\bar{A}\bar{B}=\bar{A}\bar{B}) \\
&= \bar{A}\bar{B}+\bar{A}\bar{C}(1+\bar{B}) = \bar{A}\bar{B}+\bar{A}\bar{C}
\end{aligned}$$

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

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

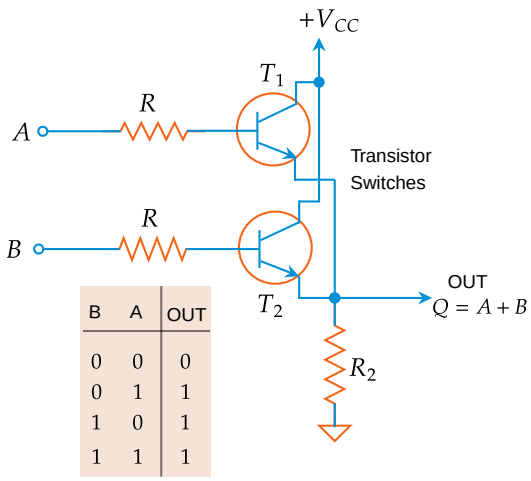
Boolean Algibra and Logic Gates

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

1.7 Logic Gates Using Transisitors

1.7.1 2-input Transistor OR Gate

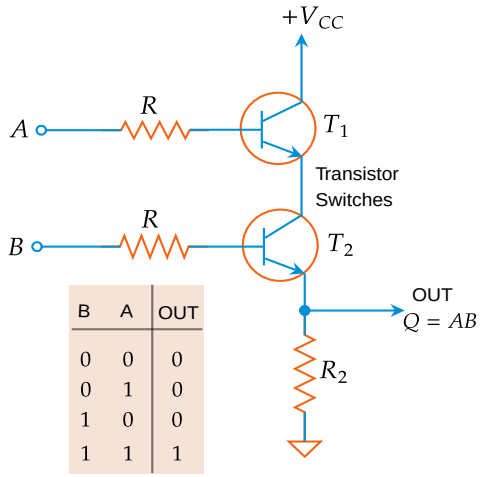
A simple 2-input inclusive OR gate can be constructed using RTL Resistor-transistor switches connected together as shown below with the inputs connected directly to the transistor bases. Either transistor must be saturated "ON" for an output at Q.



B	A	OUT
0	0	0
0	1	1
1	0	1
1	1	1

1.7.2 2-input Transistor AND Gate

A simple 2-input logic AND gate can be constructed using RTL Resistor-transistor switches connected together as shown below with the inputs connected directly to the transistor bases. Both transistors must be saturated "ON" for an output at Q.

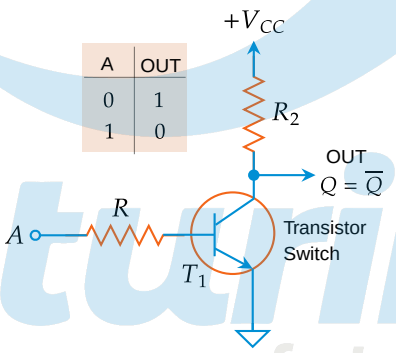


B	A	OUT
0	0	0
0	1	0
1	0	0
1	1	1

Figure 1.8

1.7.3 Transisitor NOT gate

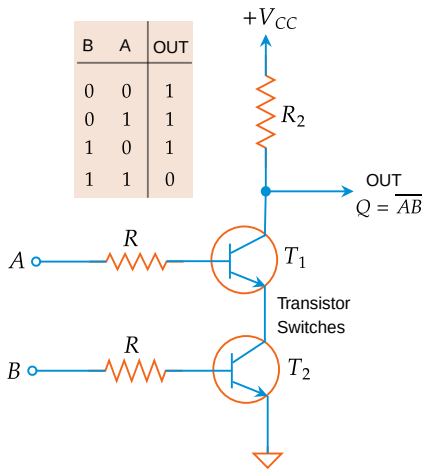
A simple 2-input logic NOT gate can be constructed using a RTL Resistor-transistor switches as shown below with the input connected directly to the transistor base. The transistor must be saturated "ON" for an inverted output "OFF" at Q.



A	OUT
0	1
1	0

Figure 1.9

1.7.4 Transistor NAND gate



B	A	OUT
0	0	1
0	1	1
1	0	1
1	1	0

1.7.5 Transisitor NOR gate

A simple 2-input logic NOR gate can be constructed using RTL Resistor-transistor switches connected together as shown below with the inputs connected directly to the transistor bases. Both transistors must be cut-off "OFF"

for an output at Q.

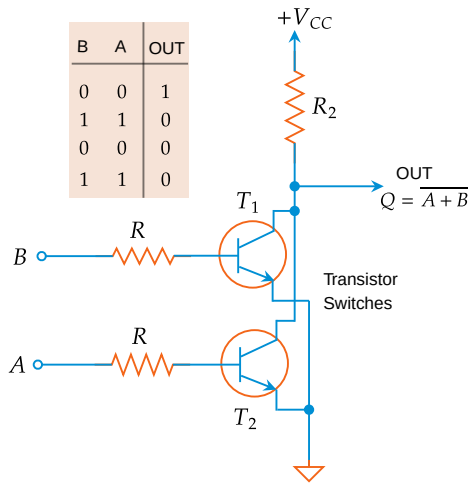


Figure 1.10

1.8 Simplification of Boolean expressions

1.8.1 Sum-of-Products and Product-of-Sums Boolean Expressions

The sum-of-products expression, which is also known as minterm, contains the sum of different terms with each term being either a single literal or a product of more than one literal. It can be obtained from the truth table directly by considering those input combinations which produce logic ' 1 ' at the output. Each such input combination produces a term. The different terms are given by product of corresponding literals. The sum of all terms gives the expression. Product-of-sums expression, which is also known as maxterm, contains the product of different terms with each term being either a single literal or a sum of more than one literal. It can be obtained from the truth table by considering those input combinations that produce logic ' 0 ' at the output. Each such input combination gives a term and product of all such terms gives the expression. Different terms are obtained by taking sum of corresponding literals. Here, ' 0 ' and ' 1 ' do mean the uncomplemented and complemented variables, respectively, unlike sum-of-products expressions where ' 0 ' and ' 1 ' do mean complemented and uncomplemented variables, respectively. Transforming given product-of-sums expression into an equivalent sum-of-products expression is a straight forward process. Multiplying out the given expression and carrying out the obvious simplification provides the equivalent sum-of-products expression. For example,

$$\begin{aligned}(A+B)\cdot(\bar{A}+\bar{B}) &= A\cdot\bar{A}+A\cdot\bar{B}+B\cdot\bar{A}+B\cdot\bar{B}\\ &= 0+A\cdot\bar{B}+B\cdot\bar{A}+0 = A\cdot\bar{B}+\bar{A}\cdot B\end{aligned}$$

A given sum-of-products expression can be transformed into an equivalent product-of-sums expression by (a) taking dual of given expression (b) multiplying out different terms to get the sum-of-products form (c) removing redundancy and (d) taking a dual to get the equivalent product-of-sums expression.
Let us consider the example, $A\cdot B+\bar{A}\cdot\bar{B}$:

$$\begin{aligned}\text{Dual of given expression} &= (A+B)\cdot(\bar{A}+\bar{B})\\ (A+B)\cdot(\bar{A}+\bar{B}) &= A\cdot\bar{A}+A\cdot\bar{B}+B\cdot\bar{A}+B\cdot\bar{B}\\ &= 0+A\cdot\bar{B}+B\cdot\bar{A}+0\\ &= A\cdot\bar{B}+\bar{A}\cdot B\\ \text{Dual of } (A\cdot\bar{B}+\bar{A}\cdot B) &= (A+\bar{B})\cdot(\bar{A}+B)\\ \text{Therefore, } A\cdot B+\bar{A}\cdot\bar{B} &= (A+\bar{B})\cdot(\bar{A}+B)\end{aligned}$$

1.8.2 Karnaugh Map Method

Karnaugh map (K-map) is a graphical representation of the logic system. It can be drawn directly from either minterm (sum-of-products) or maxterm (product-of- sums) Boolean expressions. Drawing a Karnaugh map from

truth table involves an additional step of writing the minterm or maxterm expression depending upon whether it is desired to have minimized sum-of-products or a minimized product-of-sums expression.

1.8.3 Construction of Karnaugh Map

An n-variable Karnaugh map has 2^n squares and each possible input is allotted a square. In case of a minterm Karnaugh map, ' 1 ' is placed in all those squares for which the output is ' 1 ' and ' 0 ' is placed in all those squares for which the output is ' 0 '. For simplicity, 0's are omitted. An X is placed in squares corresponding to 'don't care' conditions. In the case of a maxterm Karnaugh map, a ' 1 ' is placed in all those squares for which the output is ' 0 ' and a ' 0 ' is placed for input entries corresponding to a ' 1 ' output. Again 0's are omitted for simplicity and an X is placed in squares corresponding to 'don't care' conditions.
The process of construction of 2,3 and 4 variable Karnaugh maps is illustrated in the following examples.
The choice of terms identifying different rows and columns of a Karnaugh map is not unique for a given number of variables. The only condition to be satisfied is that the designation of adjacent rows and adjacent columns should be the same except for one of the literals being complemented. Also, the extreme rows and extreme columns are considered adjacent. Some of the possible designation styles for two-, three- and four-variable minterm Karnaugh maps are given in Figs. 1.11, 1.12 and 1.13, respectively.

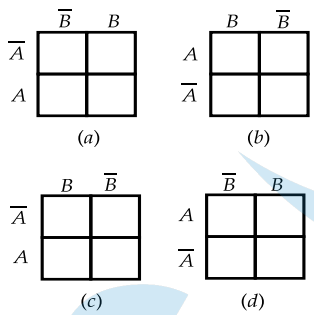


Figure 1.11: Two-variable Karnaugh map.

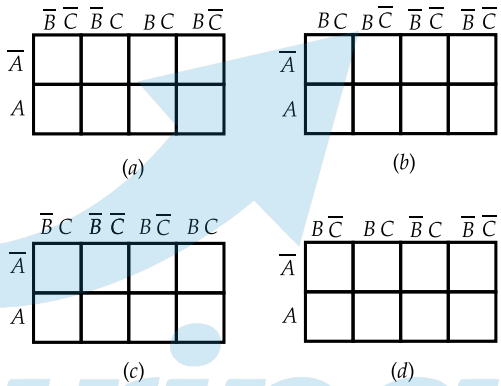


Figure 1.12: Three-Variable Karnaugh map

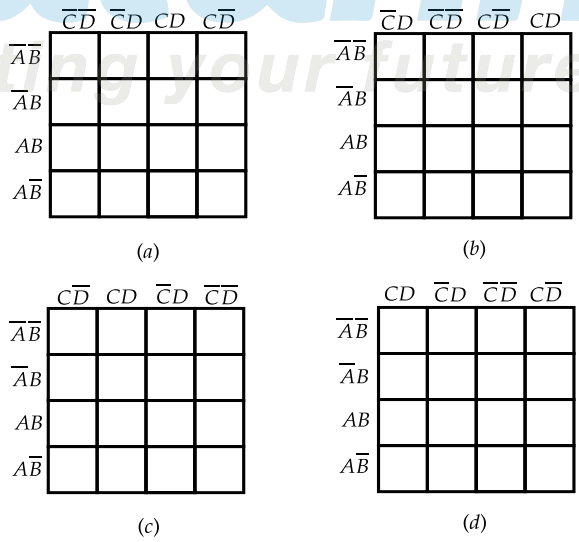


Figure 1.13: Four-Variable Karnaugh map

However not necessary to account for all optional entries. Only those optional combinations that can be used to advantage should be used.
Having made groups with all 1's having been accounted for, the minimum 'sum-of-products' or the 'product-of-sums' expressions can be written directly from Karnaugh map. Figures 1.15,1.16 and 1.17 illustrate the construction of minterm and maxterm Karnaugh maps for two-, three- and four-variable Boolean expressions, respectively.

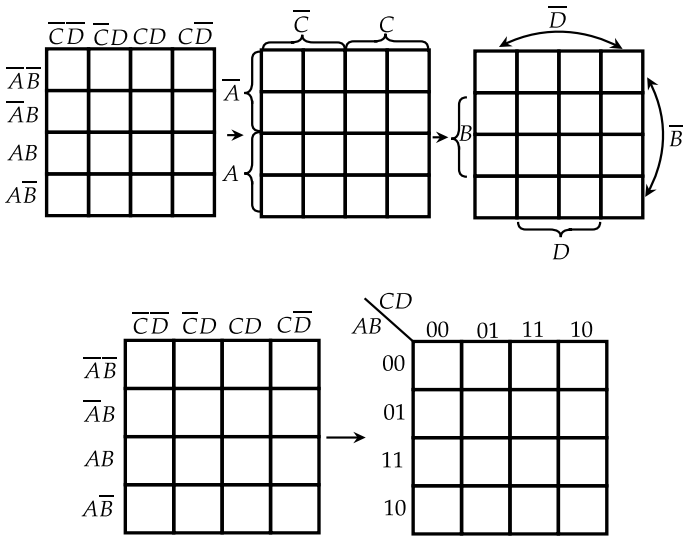


Figure 1.14: Different styles of row and columns identification.

Truth Table

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

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

Sum – of – products K – map

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

Product – of – sums K – map

Figure 1.15: Two-variable-Karnaugh maps.

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

	$\overline{B}\,\overline{C}$	$\overline{B}C$	BC	$B\overline{C}$
$\overline{A}$	1			1
A	1			1

Sum – of – products K – map

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

Products – of – sums K – map

Figure 1.16: Three-variable-Karnaugh maps.

Truth Table

A	B	C	D	Y
0	0	0	0	1
0	0	0	1	1
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	0
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	0
1	1	1	1	0

	$\overline{C}\,\overline{D}$	$\overline{C}D$	CD	$C\overline{D}$
$\overline{A}\,\overline{B}$	1	1		
$\overline{A}B$	1	1		
AB	1	1		
$A\overline{B}$	1	1		

Sum – of – products K – map

	$\overline{C}+\overline{D}$	$\overline{C}+D$	$C+D$	$C+\overline{D}$
$\overline{A}+\overline{B}$	1	1		
$\overline{A}+B$	1	1		
$A+B$	1	1		
$A+\overline{B}$	1	1		

Products – of – sums K – map

Figure 1.17: Four Variable-Karnaugh maps.

1.9 Multiplexer(MUX)

Multiplex means many into one. A multiplexer is a circuit with many inputs but only one output. By applying control signals, we can steer any input to the output. Thus it is also called a data selector and control inputs are termed select inputs. A MUX is represented by many input and single output line.The internal structure of MUX is a rotatory switch.
It is used for parellel to series conversion. It's also called as universal combinational circuit.

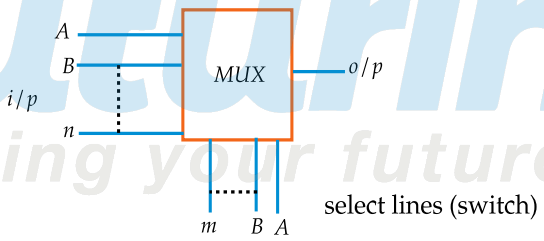


Figure 1.18

If the number of input lines in a MUX is N, then,

$$N = 2^n$$

Where, n = number of select lines.

1.9.1 2×1 MUX

Here, the number of input lines,N

$$N = 2^n = 2$$

Then,number of select lines $n = 1$

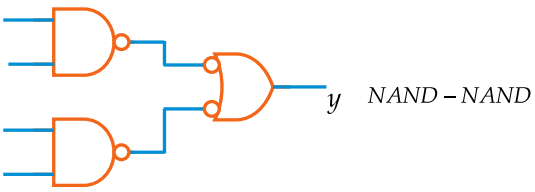
■ **Example 1.3** For a two level AND-OR circuit can be equally represented by

a. AND-AND

b. OR-AND

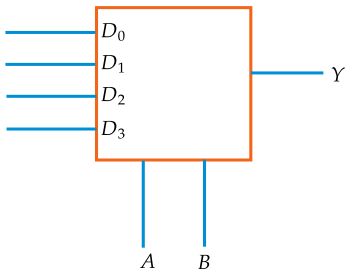
c. NOR-NOR

d. NAND-NAND



■

1.9.2 4× 1 MUX



Output y=$\bar{A}\bar{B}D_0 + \bar{A}BD_1 + A\bar{B}D_2 + ABD_3$

1.9.3 8×1 MUX

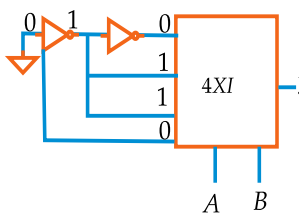
$$2^n = 8$$

number of select lines, $n = 3$

$$y = \bar{A}\bar{B}\bar{C}D_0 + \bar{A}\bar{B}CD_1 + \bar{A}B\bar{C}D_2 + \bar{A}BCD_3 + A\bar{B}\bar{C}D_4 + A\bar{B}CD_5 + AB\bar{C}D_5 + ABCD_7$$

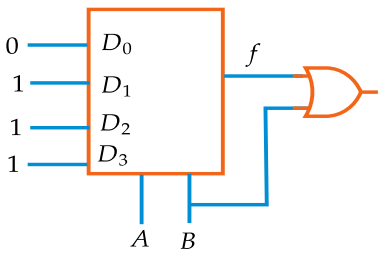
1.10 Minimization of logical expression of using MUX:

1. Minimize



$$y = \overline{A}\,\overline{B}D_0 + \overline{A}BD_1 + A\overline{B}D_2 + ABD_3$$
$$y = \overline{A}B + A\overline{B}$$
$$y = A \oplus B$$

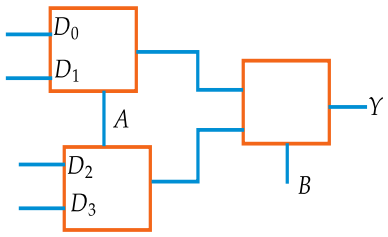
2. Minimize



$$\begin{aligned} y &= f + B \\ &= \overline{A}B + A\overline{B} + AB + B \\ &= B(\overline{A} + A + 1) + A\overline{B} \\ &= B + \overline{B}A = A + B \end{aligned}$$

1.10.1 Implementation of higher order MUX by using 2×1MUX:

1. 4×1 MUX by 2×1 MUX



Alternate method to find the number of MUX

2	4
2	2
	1

$2+1=3$

three, 2×1 MUX is required

16×1 MUX by 2×1 MUX

2	16
2	8
2	4
2	2
	1

$=8+4+2+1 \Rightarrow 15$, (Fifteen, 2×1 MUX Required.)

similarly,

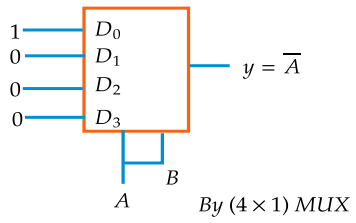
2. 256×1 MUX by 16×1 MUX

16	256
16	16
	1

$16+1=17$(Seventeen, 16×1MUX is required).

Implementation of Logic gates by using 2×1 MUX and $u\times 1$ MUX:

1. NOT gate:



Logic gate	2×1 MUX	4×1 MUX
NOT	1	1
AND	1	1
OR	1	1
NAND	2	1
NOR	2	1
EX-OR	2	1
EX-NOR	2	1

2. For the implementation of AND gate and XOR gate the number of 2X1 MUX required.

a. 1,1 **b.** 1,2 **c.** 2,1 **d.** 2,2

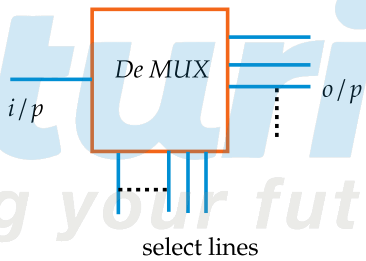
Solution:

I MUX for AND, 2MUX for XOR

So the correct answer is **Option (b)**

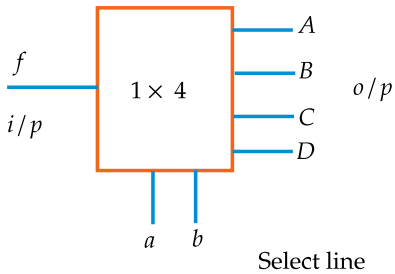
Demultiplexer:

A de MUX performs the reverse operation of MUX.



- Number of outputs $= 2^n$ where $n \rightarrow$ select lines
- De MUX is also called as data detector.
- By setting the input to true (1), the demux behaves as decoder.

1×4 **DEMUX**



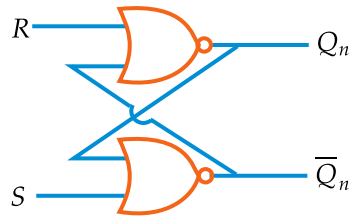
1.11 Flip-Flop

Sequential Circuit

Digital output are required to be generated in accordance with sequence in which input signals are received, which is not possible with the combinational circuit generated should depend on present and past history of

input. Such circuit is called as sequential circuit.

SR-Latch:

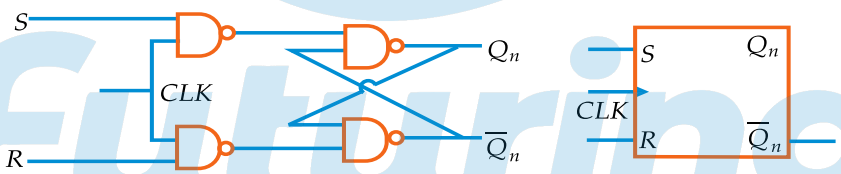


Truth Table:

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

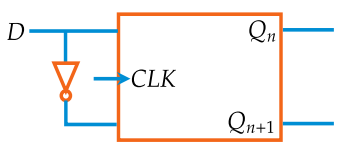
X : Input either 0 or 1

S-R Flip Flop with NAND Gate:



D-Flip Flop (Delay):

When S = D, $\bar{R} = D$, Now SR becomes D type Flip Flop.

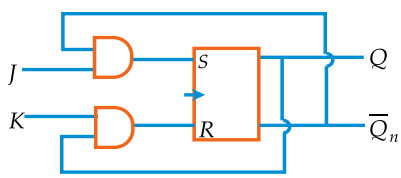


Truth Table

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

J-K Flip Flop:

$S = J\left(\bar{Q}_n\right); R = K\left(Q_n\right)$



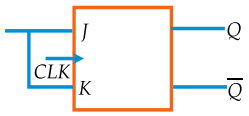
J	K	Q_{n+1}
0	0	Q_n
0	1	0
1	0	1
1	1	$\overline{Q}_n$

→ Toggle State

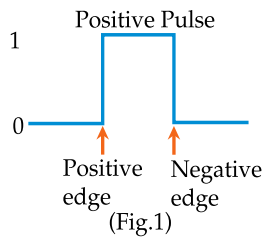
Note Problem in JK flip flop is race around condition.

T-type (Toggle) Flip Flop:

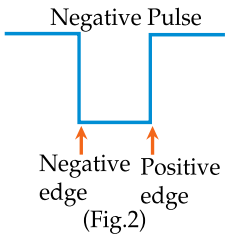
J = K = T then T = Flip Flop.



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



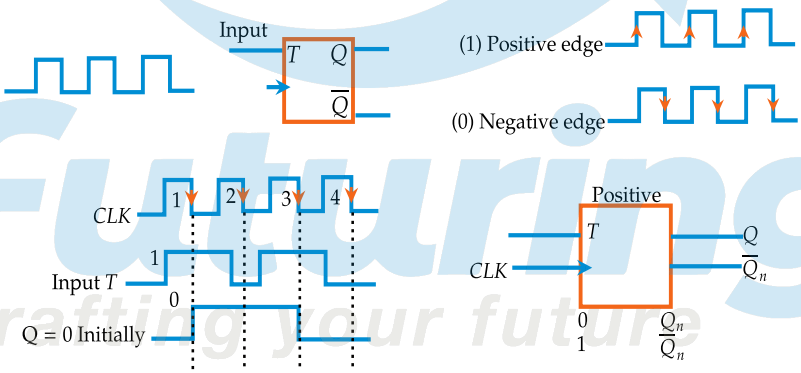
(Fig.1)



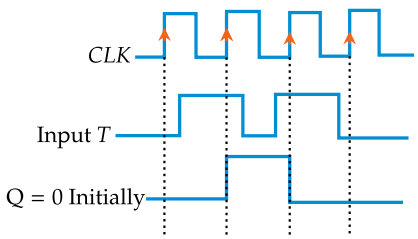
(Fig.2)

A clock pulse may be either positive or negative. A positive clock source remains at 0 during the interval between pulses and goes 0 to 1 during the occurrence of a pulse. The pulse goes through two signal transitions; from 0 to 1 and return from 1 to 0, positive transition is defined as the positive edge and the negative transition as the negative edge.

Time Diagram Representation:



If we take positive edge:



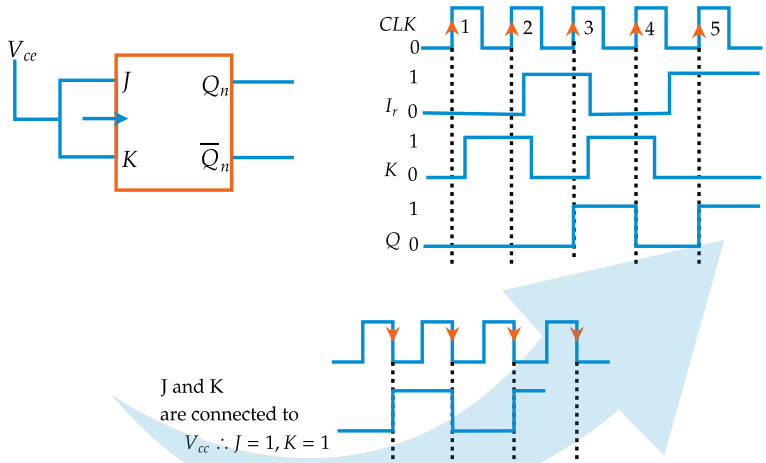
1.12 Counters

1. Asynchronous or ripple or serial
2. Synchronous or parallel or fast
3. (i) Ring counter (ii) Twisted tail or thomson or mobious (Type of shift registers)

1. **Asychronous Counter:** No, common clock-clock is output of previous flip-flop.
2. In synchronous counter common clock is used.

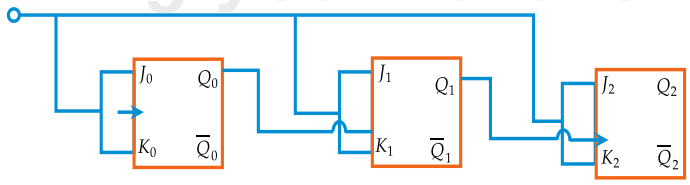
1. What is meaning of MOD-12 counter $\Rightarrow$ number of states are 12 .
   1. $0-11 \rightarrow 12$ states 2. $2-13 \rightarrow 12$ states
   3. $3-14 \rightarrow 12$ states 4. $1-12 \rightarrow 12$ states
   MOD 12 means divide by 12 .

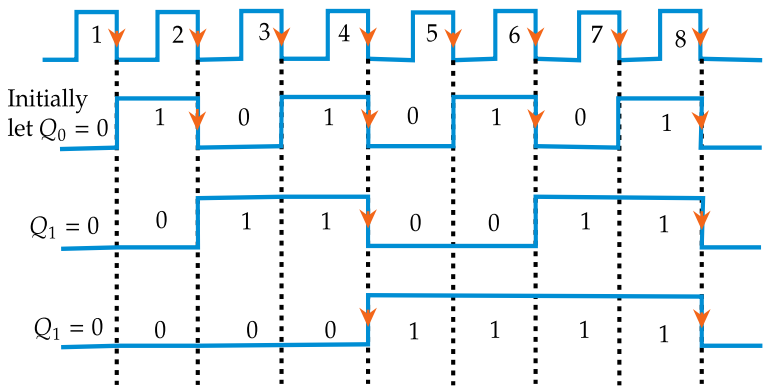
Asynchronous Counter:



Note
1. Total number of flip-flop required for Mod-N counter $N = 2^n$.
2. 3 bit means MOD-8 counter $\Rightarrow$ MOD $8 = 2^3$ means 3 Flip-Flop required.
3. 4 bit $\rightarrow$ 16MOD $\rightarrow 2^4 \rightarrow$ 4 Flip Flop required

MOD-8 Asychronous Counter:

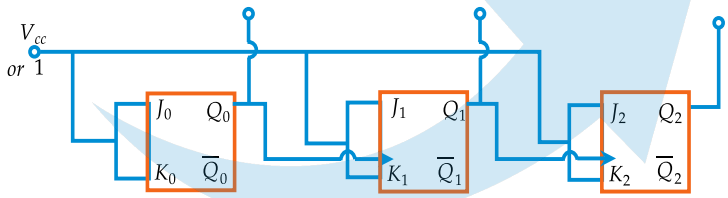




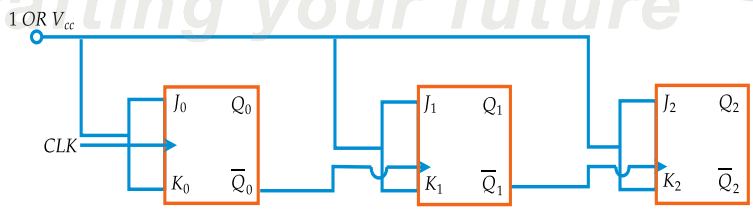
Truth Tuble

CLK	Q_2	Q_1	Q_0
1	0	0	0
2	0	0	1
3	0	1	0
4	0	1	1
5	1	0	0
6	1	0	1
7	1	1	0
8	1	1	1

Edge → Positive → (i) up counter (ii) Down counter
Edge → Negative → (i) up counter (ii) Down counter

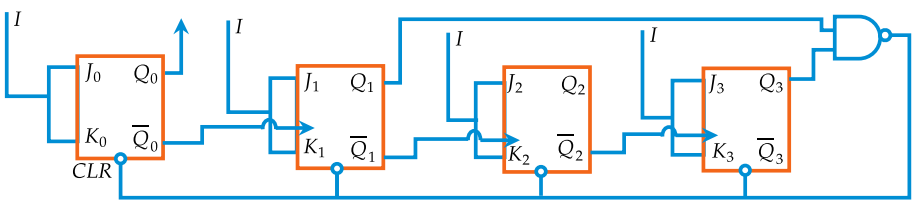


Q_2, Q_1, Q_0 are standard output
Case (i): If the output is of first flip-flop is given as circuit to next flip-flop it will act as up counter.
Case (ii): If $\bar{Q}$ of 1 st flip flop is given as circuit to next flip-flop it will act as down counter.



Synchronous Counter

(i) Common clock is there (ii) There are fast
Widely used if MOD is in form of 2N then design is simple. If MOD is not in form of 2N then design by use of K-map.
Example: MOD-10 UP counter.



Truth Table

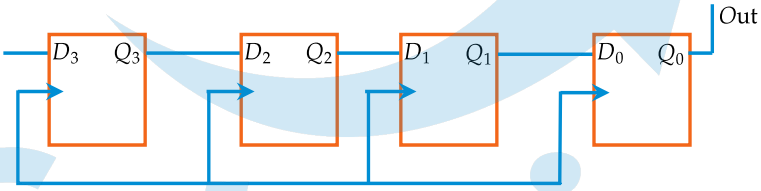
Q_3	Q_2	Q_1	Q_0
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0

Remove

Shift Register

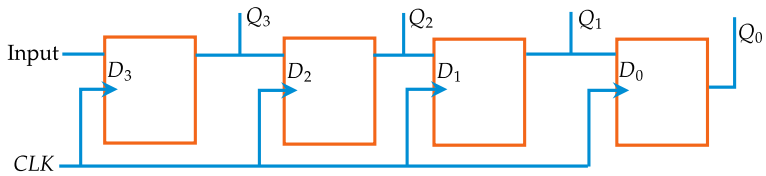
Register's are group of flip-flop.
To store n-bits n-bits n-flip-flop are required in register.
Depending upon input and output registers can be classified as
(1) SISO [Serial input serial output] (3) PISO [Parallel in serial output]
(2) SIPO [Serial input parallel out] (4) PIPO [Parallel input parallel output]

4-Bit SISO



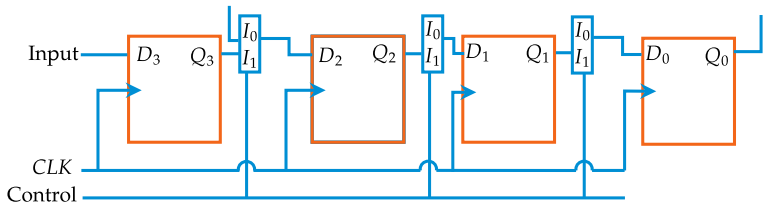
To provide n-bit data in $(n-1)$-clk pulse required,
To store n-bit data n-click pulse required.
S

IPO(4-Bit)



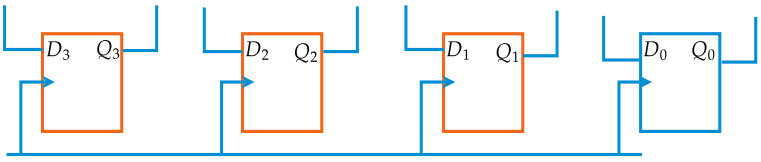
To provide n-bit data in n-clk pulse required, to provide parallel out no circuit pulse required.

PISO

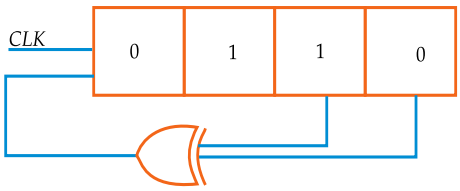


Control 0 → Parallel input, Control 1 → Serial output

PIPO



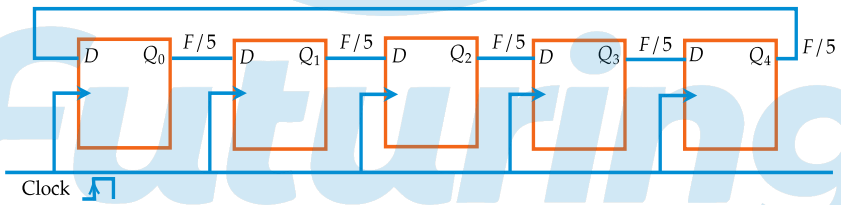
Initial contents of 4-bit SIPO, ring shift register, shown in figure is 0110 . After 3 clock pulses are applied, what are contents of shift register.



Solution: After 1 st clock $\rightarrow$ 1011, 2 nd clock $\rightarrow$ 0101, 3rd clock $\rightarrow$ 1010. So content are 1010 .

Ring Counter:

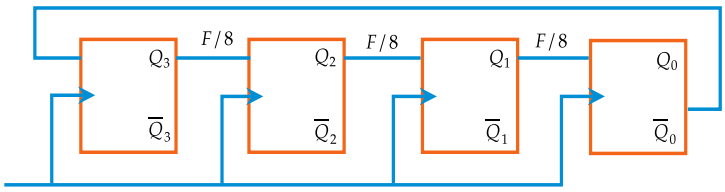
Design MOD-5 ring counter. After each 10 steps is reads again 0000.



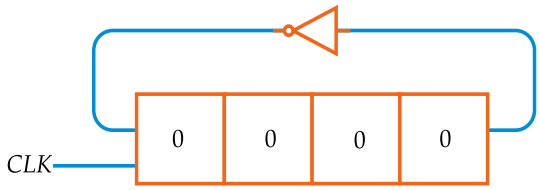
Ring counter is shift register with feedback applied last flip-flop output Q to input of first flip flop.
Ring counter is one bit is logic one and it will rotate with clock.
In n-bit ring counter number of use state is n.
Number of unused states in n-bit ring counter is $2^n - n$.

CLK	Q_4	Q_3	Q_2	Q_1	Q_0	
0	0	0	0	0	0	
1	1	0	0	0	0	
2	0	1	0	0	0	
3	0	0	1	0	0	5 State
4	0	0	0	1	0	
5	0	0	0	0	1	
6	1	0	1	0	0	
7	0	1	0	0	0	
8	0	0	1	0	0	5 State
9	0	0	0	1	0	
10	0	0	0	0	1	

Johnson Counter or (Twisted Ring Counter) or Switch Tail Counter or Creeping Counter or Mobies Counter or Walking Counter.



Equivalent Circuit:



Truth Table :

CLK	Q_3	Q_2	Q_1	Q_0
0	0	0	0	0
1	1	0	0	0
2	1	1	0	0
3	1	1	1	0
4	1	1	1	1
5	0	1	1	1
6	0	0	1	1
7	0	0	0	1
8	0	0	0	0

In Johnson counter with n-flip-flop maximum possible states are $2n$ states or maximum uses states. Unused states are $2^n - 2n$.
50% duty cycle.
When a Johnson counter is working in uses state the operation frequency $f/2n$.

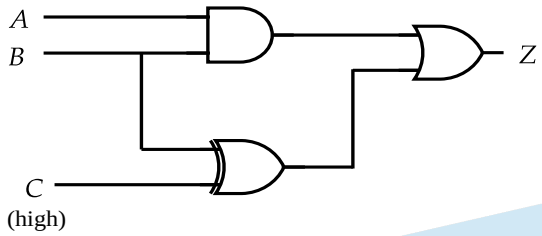
Practice set-1

1. A signal of frequency $10kHz$ is being digitalized by an A/D converter. A possible sampling time which can be used is

[NET/JRF(JUNE-2011)]

A. $100\mu s$ **B.** $40\mu s$ **C.** $60\mu s$ **D.** $200\mu s$

2. Consider the digital circuit shown below in which the input C is always high (1).



The truth table for the circuit can be written as

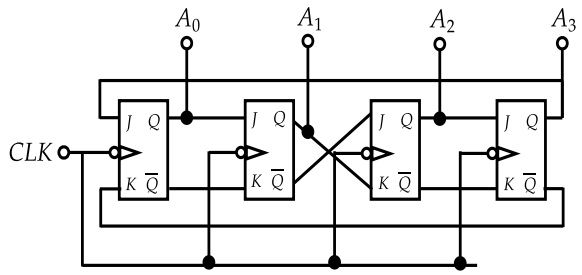
A	B	Z
0	0	1
0	1	0
1	0	1
1	1	1

The entries in the Z column (vertically) are

[NET/JRF(JUNE-2011)]

A. 1010 **B.** 0100 **C.** 1111 **D.** 1011

3. A counter consists of four flip-flops connected as shown in the figure:



If the counter is initialized as $A_0A_1A_2A_3 = 0110$, the state after the next clock pulse is

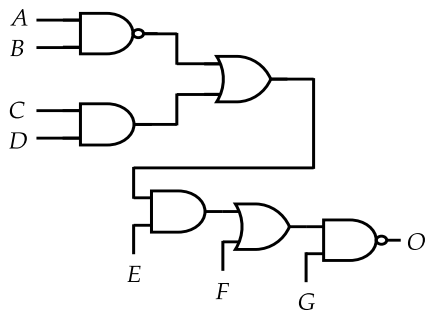
[NET/JRF(DEC-2011)]

A. 1000 **B.** 0001 **C.** 0011 **D.** 1100

4. The output, O, of the given circuit in cases I and II, where
Case I: $A, B = 1; C, D = 0; E, F = 1$ and $G = 0$
Case II: $A, B = 0; C, D = 0 : E, F = 0$ and $G = 1$ are respectively.

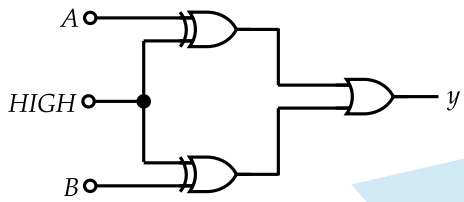
[NET/JRF(JUNE-2012)]

A. 1,0 **B.** 0,1 **C.** 0,0 **D.** 1,1



5. The logic circuit shown in the figure below Implements the Boolean expression

[NET/JRF(DEC-2012)]



A. $y=\overline{A\cdot B}$ **B.** $y=\bar{A}\cdot\bar{B}$ **C.** $y=A\cdot B$ **D.** $y=A+B$

6. If the analog input to an 8 -bit successive approximation ADC is increased from 1.0 V to 2.0 V, then the conversion time will

[NET/JRF(JUNE-2013)]

A. Remain unchanged **B.** Double

C. Decrease to half its original value **D.** Increase four times

7. If one of the inputs of a J-K flip flop is high and the other is low, then the outputs Q and $\bar{Q}$

[NET/JRF(DEC-2013)]

A. Oscillate between low and high in race around condition

B. Toggle and the circuit acts like a T flip flop

C. Are opposite to the inputs

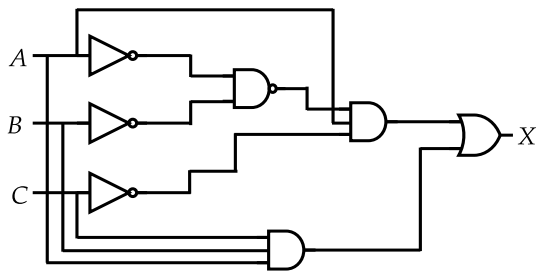
D. Follow the inputs and the circuit acts like an $R-S$ flip flop

8. A 4 -variable switching function is given by $f=\sum(5,7,8,10,13,15)+d(0,1,2)$, where d is the do-not-care-condition. The minimized form of f in sum of products (SOP) form is

[NET/JRF(DEC-2013)]

A. $\bar{A}\bar{C}+\overline{BD}$ **B.** $A\bar{B}+C\bar{D}$ **C.** $AD+BC$ **D.** $\overline{BD}+BD$

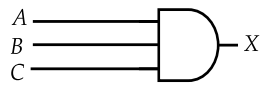
9. For the logic circuit shown in the below



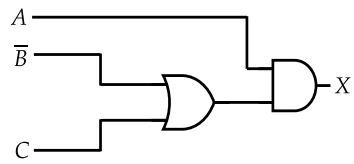
A simplified equivalent circuit is

[NET/JRF(JUNE-2014)]

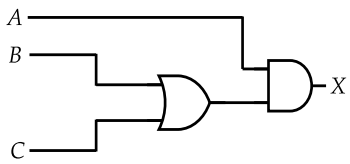
A.



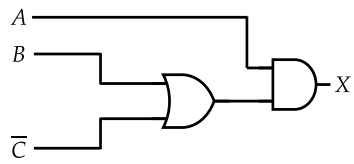
B.



C.

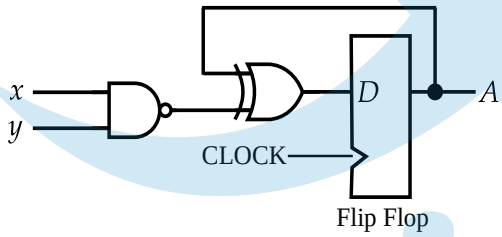


D.

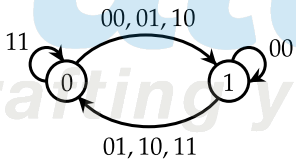


10. The state diagram corresponding to the following circuit is

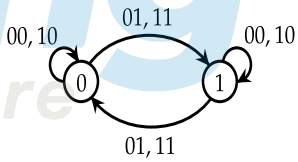
[NET/JRF(DEC-2015)]



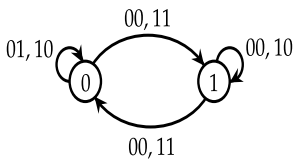
A.



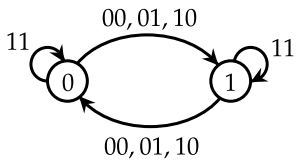
B.



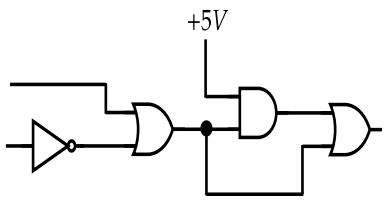
C.



D.



11. In the schematic figure given below, assume that the propagation delay of each logic gate is t_{gate} .



The propagation delay of the circuit will be maximum when the logic inputs A and B make the transition

[NET/JRF(JUNE-2016)]

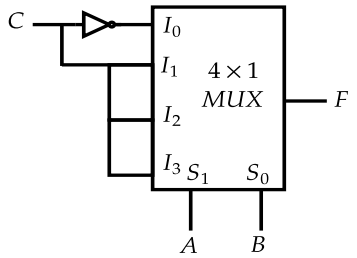
A. $(0,1) \to (1,1)$
B. $(1,1) \to (0,1)$
C. $(0,0) \to (1,1)$
D. $(0,0) \to (0,1)$

12. Which of the following circuits implements the Boolean function

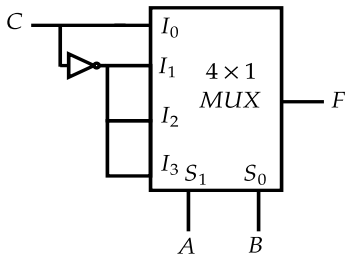
$$F(A,B,C) = \sum(1,2,4,6)?$$

[NET/JRF(DEC-2016)]

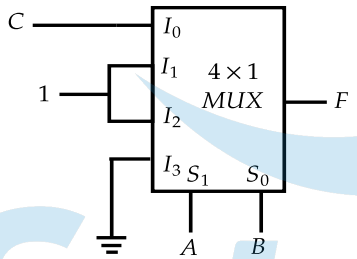
A.



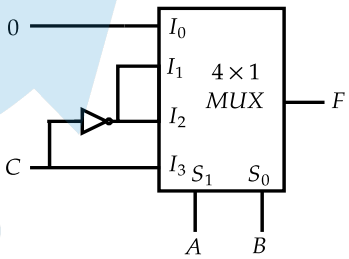
B.



C.



D.



13. A 2×4 decoder with an enable input can function as a

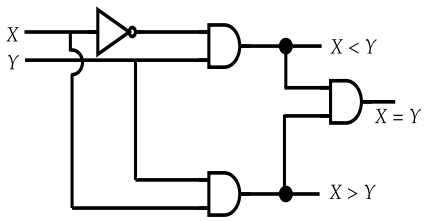
[NET/JRF(JUNE-2017)]

A. 4×1 multiplexer
B. 1×4 demultiplexer
C. 4×2 encoder
D. 4×2 priority encoder

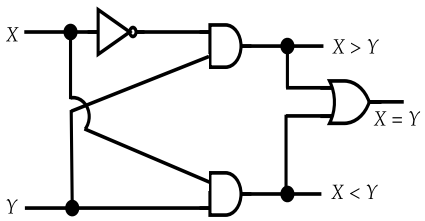
14. In the figures below, X and Y are one bit inputs. The circuit which corresponds to a one bit comparator is

[NET/JRF(JUNE-2017)]

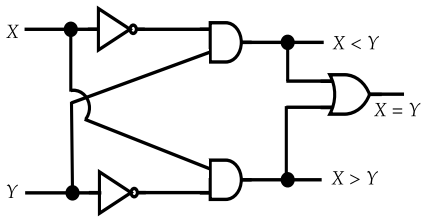
A.



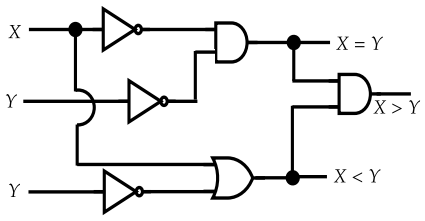
B.



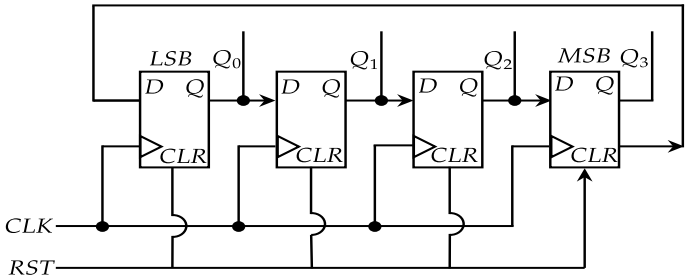
C.



D.



15. The circuit below comprises of D-flip flops. The output is taken from Q_3, Q_2, Q_1 and Q_0 as shown in the figure.



the binary number given by the string $Q_3, Q_2, Q_1 Q_0$ changes for every clock pulse that is applied to the CLK input. If the output is initialized at 0000 , the the corresponding sequence of decimal numbers that repeats itself, is

[NET/JRF(DEC-2017)]

A. $3, 2, 1, 0$

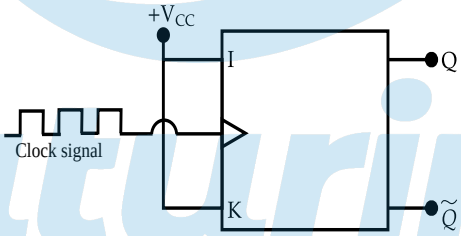
B. $1, 3, 7, 14, 12, 8$

C. $1, 3, 7, 15, 12, 14, 0$

D. $1, 3, 7, 15, 14, 12, 8, 0$

16. In the following JK flip-flop circuit, J and K inputs are tied together to $+V_{CC}$. If the input is a clock signal of frequency f, the frequency of the output Q is

[NET/JRF(JUNE-2018)]



A. f

B. $2f$

C. $4f$,

D. $\frac{f}{2}$

17. Which of the following gates can be used as a parity checker?

[NET/JRF(JUNE-2018)]

A. An OR gate

B. A NOR gate

C. An exclusive OR (XOR) gate

D. An AND gate

18. The full scale of a 3 -bit digital-to-analog (DAC) converter is $7V$. Which of the following tables represents the output voltage of this 3 -bit DAC for the given set of input bits?

[NET/JRF(JUNE-2018)]

A.

Input bits	Output voltage
000	0
001	1
010	2
011	3

B.

Input bits	Output voltage
000	0
001	1.25
010	2.5
011	3.75

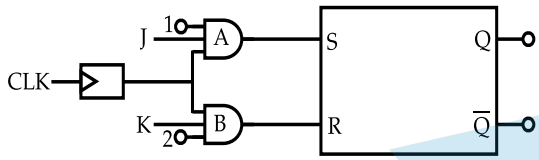
C.

Input bits	Output voltage
000	1.25
001	2.5
010	3.75
011	5

D.

Input bits	Output voltage
000	1
001	2
010	3
011	4

19. Consider the following circuit, consisting of an RS flip-flop and two AND gates.



Which of the following connections will allow the entire circuit to act as a JK flip-flop?

[NET/JRF(DEC-2018)]

A. Connect Q to pin 1 and $\bar{Q}$ to pin 2

B. Connect Q to pin 2 and $\bar{Q}$ to pin 1

C. Connect Q to K input and $\bar{Q}$ to J input

D. Connect Q to J input and $\bar{Q}$ to K input

20. The truth table below gives the value $Y(A,B,C)$ where A,B and C are binary variables. The output Y can be represented by

[NET/JRF(DEC-2018)]

A. $Y = \overline{AB}C + \bar{A}B\bar{C} + A\bar{B}C + AB\bar{C}$

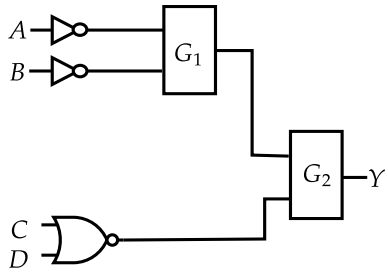
B. $Y = \bar{A}\bar{B}\bar{C} + \bar{A}BC + A\bar{B}\bar{C} + ABC$

C. $Y = \overline{ABC} + \bar{A}BC + A\overline{BC} + ABC$

D. $Y = \bar{A}\bar{B}\bar{C} + \bar{A}B\bar{C} + A\overline{BC} + AB\bar{C}$

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

21. Let Y denote the output in the following logical Circuit.



If $Y = AB + \overline{CD}$, the gates G_1 and G_2 must, respectively, be

[NET/JRF(JUNE-2019)]

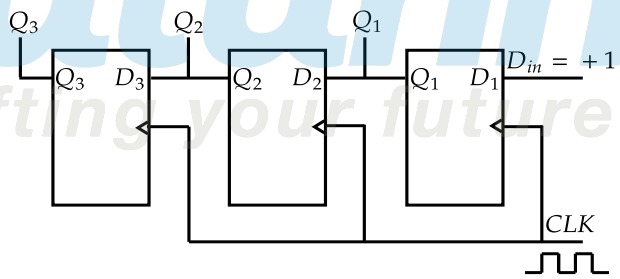
A. OR and NAND

B. NOR and OR

C. AND and NAND

D. NAND and OR

22. In the 3 -bit register shown below, Q_1 and Q_3 are the least and the most significant bits of the output, respectively.



If Q_1, Q_2 and Q_3 are set to zero initially, then the output after the arrival of the second falling clock (CLK) edge is

[NET/JRF(JUNE-2020)]

A. 001

B. 100

C. 011

D. 110

23. The Boolean equation $Y = \bar{A}BC + \bar{A}B\bar{C} + A\bar{B}\bar{C} + A\bar{B}C$ is to be implemented using only twoinput NAND gates. The minimum number of gates required is

[NET/JRF(JUNE-2020)]

A. 3

B. 4

C. 5

D. 6

Answer key			
Q.No.	Answer	Q.No.	Answer
1	**B**	2	**D**
3	**B**	4	**D**
5	**A**	6	**A**
7	**D**	8	**D**
9	**D**	10	**D**
11	**D**	12	**B**
13	**B**	14	**C**
15	**D**	16	**D**
17	**C**	18	**A**
19	**B**	20	**B**
21	**B**	22	**C**
23	**B**		

Practice set-2

1. The voltage resolution of a 12 -bit digital to analog converter (DAC), whose output varies from $-10V$ to $+10V$ is, approximately

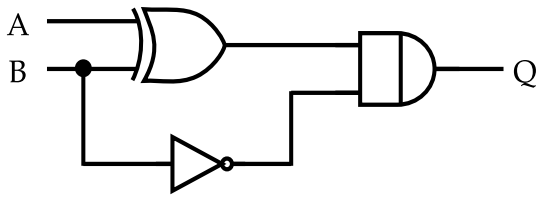
[GATE 2010]

A. 1 mV **B.** 5 mV **C.** 20 mV **D.** 100 mV

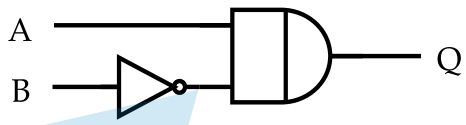
2. For any set of inputs, A and B, the following circuits give the same output, Q, except one. Which one is it?

[GATE 2010]

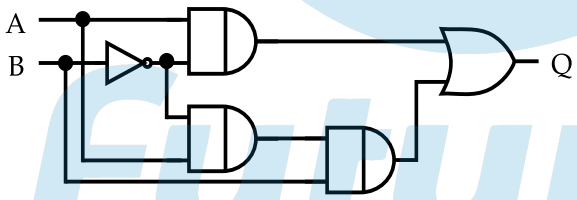
A.



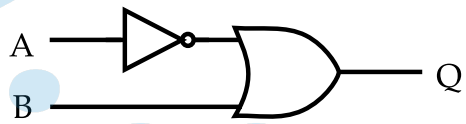
B.



C.



D.



3. The following Boolean expression

$$Y = A\cdot\bar{B}\cdot\bar{C}\cdot\bar{D}+\bar{A}\cdot B\cdot\bar{C}\cdot D+\bar{A}\cdot\bar{B}\cdot\bar{C}\cdot D+\bar{A}\cdot\bar{B}\cdot C\cdot D+\bar{A}\cdot B\cdot C\cdot D+A\cdot\bar{B}\cdot\bar{C}\cdot D \quad \text{can}$$

be simplified to

[GATE 2011]

A. $\bar{A}\bullet\bar{B}\bullet C+A\bullet\bar{D}$ **B.** $\bar{A}\bullet B\bullet\bar{C}+A\bullet\bar{D}$

C. $A\bullet\bar{B}\bullet\bar{C}+\bar{A}\bullet D$ **D.** $A\bullet\bar{B}\bullet C+\bar{A}\bullet D$

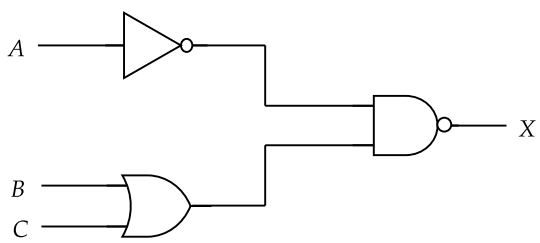
4. Which one of the following DOES NOT represent an exclusive OR operation for inputs A and B?

[GATE 2015]

A. $(A+B)\overline{AB}$ **B.** $A\bar{B}+B\bar{A}$ **C.** $(A+B)(\bar{A}+\bar{B})$ **D.** $(A+B)AB$

5. For the digital circuit given below, the output X is

[GATE 2016]



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

6. The minimum number of NAND gates required to construct an OR gate is:

[GATE 2017]

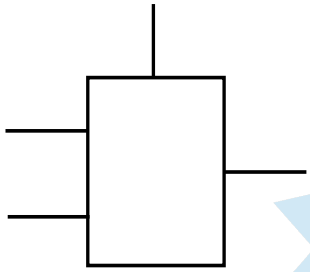
A. 2 **B.** 4 **C.** 5 **D.** 3

7. The logic expression $\bar{A}BC+\bar{A}\bar{B}C+AB\bar{C}+A\bar{B}\bar{C}$ can be simplified to

[GATE 2018]

A. A XOR C **B.** A AND C **C.** 0 **D.** 1

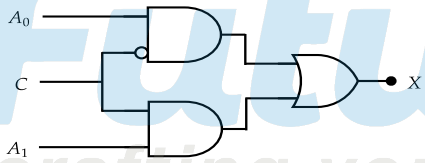
8. In a 2-to-1 multiplexer as shown below, the output $X=A_0$ if $C=0$ and $X=A_1$ if $C=1$.



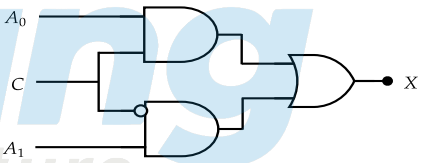
Which one of the following is the correct implementation of this multiplexer?

[GATE 2018]

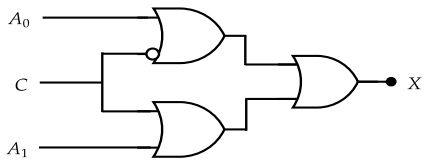
A.



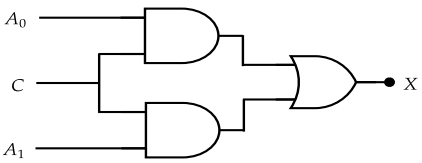
B.



C.



D.



9. Consider the following Boolean expression:

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

It can be represented by a single three-input logic gate. Identify the gate

[GATE 2019]

A. AND **B.** OR **C.** XOR **D.** NAND

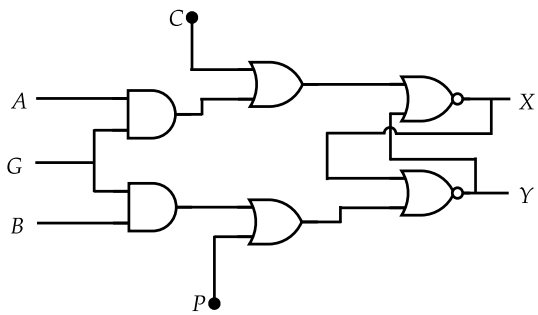
10. A 3 - bit analog-to-digital converter is designed to digitize analog signals ranging from $0V$ to $10V$. For this converter, the binary output corresponding to an input of $6V$ is

[GATE 2019]

A. 011 **B.** 101 **C.** 100 **D.** 010

11. For the following circuit, the correct logic values for the entries X_2 and Y_2 in the truth table areFor the following circuit, the correct logic values for the entries X_2 and Y_2 in the truth table are

[GATE 2019]



G	A	B	P	C	X	Y
1	0	1	0	0	0	1
0	0	0	1	0	X_2	Y_2
1	0	0	0	1	0	1

A. 1 and 0 **B.** 0 and 0 **C.** 0 and 1 **D.** 1 and 1

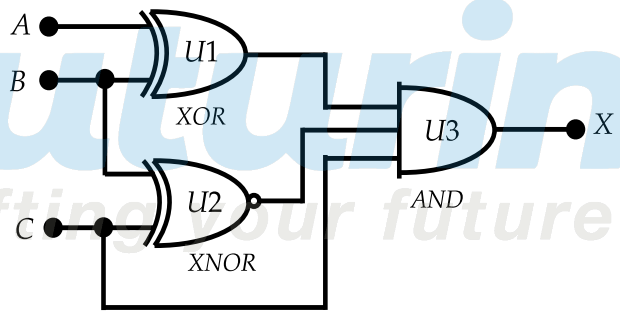
12. The net charge of an n -type semiconductor is

[JEST 2012]

A. Positive **B.** Zero **C.** Negative **D.** Dependent

13. For the logic circuit shown in figure, the required input condition (A,B,C) to make the output $(X) = 1$ is,

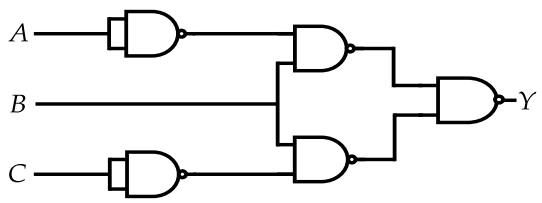
[JEST 2015]



A. $1,0,1$ **B.** $0,0,1$ **C.** $1,1,1$ **D.** $0,1,1$

14. What is Y for the circuit shown below?

[JEST 2017]



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

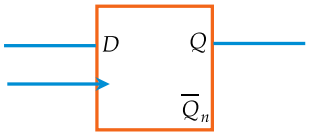
C. $Y = \overline{(\bar{A}+B)(\bar{B}+C)}$ **D.** $Y = \overline{(A+B)(\bar{B}+C)}$

Answer key			
Q.No.	Answer	Q.No.	Answer
1	**B**	2	**D**
3	**C**	4	**D**
5	**B**	6	**D**
7	**A**	8	**A**
9	**D**	10	**C**
11	**A**	12	**B**
13	**D**	14	**A**

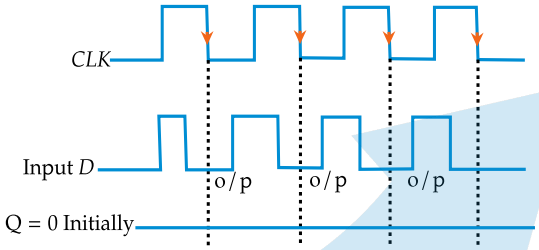
Practice set-3

1. For negative edge draw the time diagram of D Flip-Flop?

Solution:

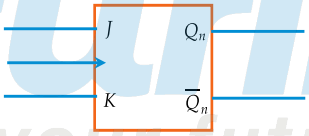


For negative edge draw time diagram of D type flip flop.

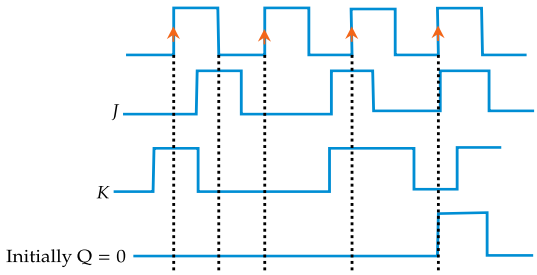


If D is high then output is high. If D is low output is low.

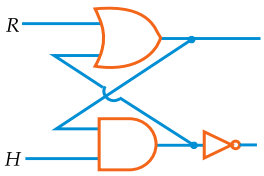
2. For the positive clock pulse find the timing diagram of JK flip flop.



Solution: Time diagram of above flip flop.



3. Consider a latch circuit shown in figure below, which of the following let of input is invalid for circuit.



a. $R = 0, H = 0$

b. $R = 0, H = 1$

c. $R = 1, H = 1$

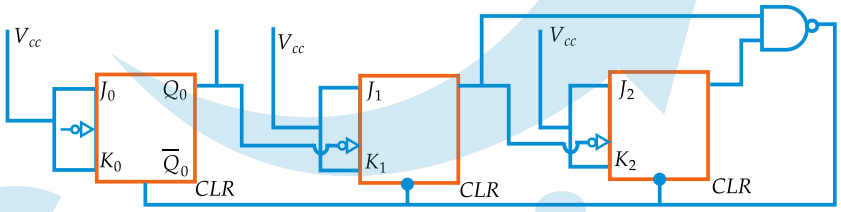
d. $R = 1, H = 0$

Solution:

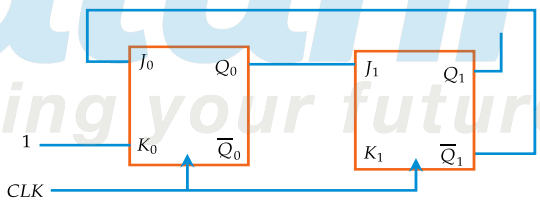
R	H	Q	Q or Q^{n+1}
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	X
1	0	1	X
1	1	0	1
1	1	1	1

4. Design MOD-6 UP Counter:

Solution:



5. Find MOD of the counter:



a. 1 **b.** 2 **c.** 3 **d.** 4

Solution:

$$\text{Module of counter } \rightarrow 3, J_0 = \bar{Q}_1, \bar{J}_1 = Q_0, K_0 = 1, K_1 = 1$$

$$\text{Let initially counter is at } \begin{matrix} Q_1 & Q_0 \\ 0 & 0 \end{matrix}$$

$$\text{i.e. } J_0 = 1 \qquad \mathrm{J}_1 = 0$$
$$\mathrm{K}_0 = 1 \qquad \mathrm{K}_1 = 1$$

After one clock pulse

$$Q_0 = 1, Q_1 = 0, J_0 = 1, J_1 = 1, K_0 = 1, K_1 = 1$$

After two clock pulse

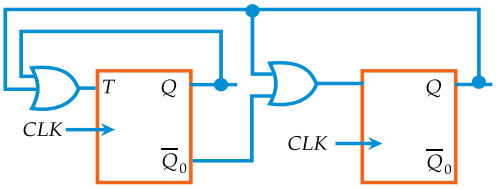
$$Q_0 = 0, Q_1 = 1, J_0 = 0, J_1 = 0, K_0 = 1, K_1 = 1, Q_0 = 0, Q_1 = 1$$

So reading

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

MOD-3 Counter

6. The circuit shown in figure below is



a. a MOD-2 counter

b. A MOD- 3 counter

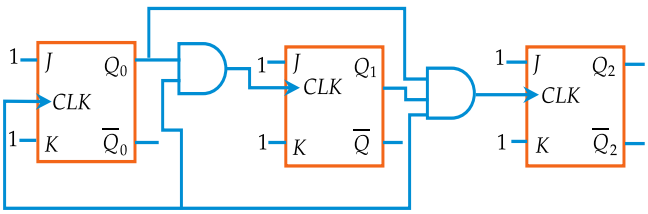
c. Generate sequence $00, 10, 01, 01\ldots$

d. Generate sequence $00, 10, 00, 00\ldots$.

Solution: The truth table is shown below:

Present State		Flip Flop Input		Next State	
Q_A	Q_B	T_A	T_B	Q_A^+	Q_B^+
0	0	0	1	0	1
0	1	1	1	1	0
1	0	1	0	0	0
1	1	1	1	0	0

7. Consider a sequential circuit shown in figure. Initially all the flip-flop are reset output $Q_0Q_1Q_2$ after 5th clock pulse is



a. 100

b. 101

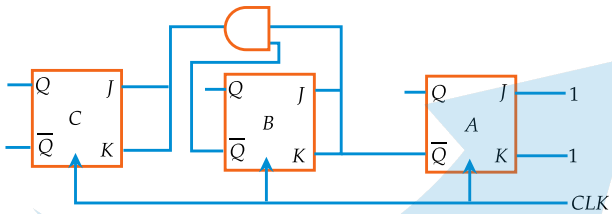
c. 110

d. 111

Solution: This is a 3 bit counter, so the output sequence is

CLK	Q_2	Q_1	Q_0
Initially	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. The counter shown in figure below is a



a. MOD-8 up counter

b. MOD-8 down counter

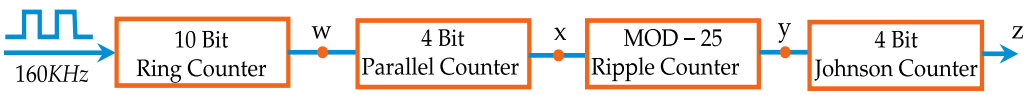
c. MOD-6 up counter

d. MOD-6 down counter

Solution:

FFC	FFB	FFA	
$JK\bar{C}$	$JK\bar{B}$	$JK\bar{A}$	$C^+B^+A^+$
111	111	111	111
000	000	110	110
000	110	111	101
000	001	110	100
111	111	111	011
001	000	110	010
001	110	111	001
000	001	110	000

9. The frequency of the pulse at Z in the N/W. Show in figure below is



a. 10 Hz

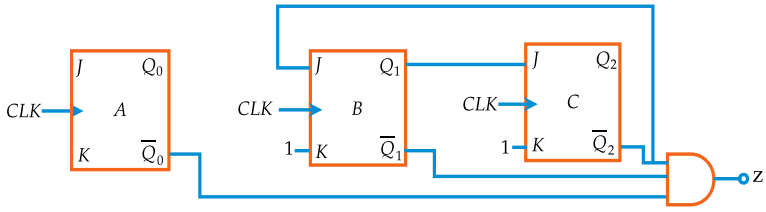
b. 160 Hz

c. 40 Hz

d. 5 Hz

Solution: 10-bit ring counter is a MOD-10, so it divides the 160KHz input by 10 . Therefore, $\mathbf{w} =$ 16KHz. The four bit parallel counter is a MOD-16. Thus, the frequency at $x = 1$KHz, the MOD $- 25$ ripple counter produces a frequency at $y = 40$ Hz.$(1\text{Khz}/25 = 40$ Hz$)$. The four bit Johnson counter is a MOD-8. This the frequency at z $= 5$ Hz.

10. Consider a sequential circuit using three J-K flip-flop and one AND gate shown in figure output of the circuit becomes ' 1 ' after every N-clock cycle. The value of N is



a. 4 **b.** 7 **c.** 8 **d.** 6

Solution: Let initially output is 1 , then

CLK	Q	Q_1	Q_0	Z
Initially	0	0	0	1
1	1	1	0	0
2	0	0	1	0
3	1	0	0	0
4	0	1	0	0
5	1	0	1	0
6	0	0	0	