



## 1.9 INTRODUCTION

A logic circuit is so named, because it can simulate human mental process like to take decisions. It is a combination of different electronic components, which obeys laws of Boolean algebra and digital electronics.

Thus, it produces required result at its output. Logic high or logic-1 means a state in digital circuit, when its input/output is positive or above ground level (*i.e. in I quadrant, above x-axis*). This state is indicated by a '1'.

Similarly, logic low or logic-0 means a state in digital circuit, when its input/output is negative (*i.e. at ground level or below ground level*). This logic state of the circuit is indicated by a '0'. But the inverse of binary state is also produced by some logic circuits. For example, if input is 1, then the output will be 0 and vice versa. This process is called the inversion of binary state is called *complement* of a bit.

## 1.10 BASIC GATES

There are three types of basic gates. They are AND gate, OR gate and NOT gate. These are also known as logic circuits because they can perform different logical functions.

### 1.10.1 AND LOGIC GATE

**Definition:** An AND gate is a logic circuit, whose output becomes high ONLY WHEN ALL ITS INPUTS ARE HIGH. Its logic equation is  $Y = A \cdot B$ .

This equation shows that inputs A & B are related to each other with AND mathematical operator. Thus, when A AND B both are high, then only Y is high, otherwise low. In this way, AND multiplication rules for all possible inputs combinations are –

When  $A = B = 0$ , then  $Y = A \cdot B = 0 \cdot 0 = 0$   
 When  $A = 0, B = 1$ , then  $Y = A \cdot B = 0 \cdot 1 = 0$   
 When  $A = 1, B = 0$ , then  $Y = A \cdot B = 1 \cdot 0 = 0$   
 When  $A = B = 1$ , then only  $Y = A \cdot B = 1 \cdot 1 = 1$



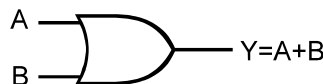
A	B	$Y = A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

### 1.10.2 OR LOGIC GATE

**Definition:** An OR gate is a logic circuit whose output becomes high WHEN ANY ONE OF ITS INPUTS IS HIGH. Its logic equation is  $Y = A + B$ . It is also called inclusive OR gate.

This equation shows that inputs A and B are related with OR mathematical operator. Thus, when A OR B is high, Y is high. The OR addition rules for all possible combinations of inputs are –

When  $A = B = 0$ ,  $Y = A + B = 0 + 0 = 0$   
 When  $A = 0, B = 1$ ,  $Y = A + B = 0 + 1 = 1$   
 When  $A = 1, B = 0$ ,  $Y = A + B = 1 + 0 = 1$   
 When  $A = B = 1$ ,  $Y = A + B = 1 + 1 = 1$

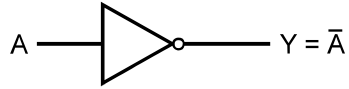


A	B	$Y = A + B$
0	0	0
0	1	1
1	0	1
1	1	1

### 1.10.3 NOT LOGIC GATE

**Definition:** A NOT gate has only one input and one output. It gives high output WHEN ITS INPUT IS LOW. Its logic equation is  $Y = \bar{A}$ . Thus, it shows that Y is complement of A. The rules of NOT operations are –

When  $A = 0$ , then  $\bar{A} = 1$   
 When  $A = 1$ , then  $\bar{A} = 0$



A	$Y = \bar{A}$
0	1
1	0

### 1.11 BOOLEAN ALGEBRA

When we deal with logic, we actually face with two-valued question. Its answer will be either in YES or NO! The English mathematician **George Boole** assigned two symbols to the solutions: **Yes = 1 and No = 0**. This is called method of reasoning to answer a two-valued question. Thus, the Boolean algebra is mainly used in digital circuits to manipulate, design and analyze the working of electronic circuits in terms of high or low, 1 or 0, yes or no, true or false etc.

#### 1.11.1 LAWS OF BOOLEAN ALGEBRA

According to Boolean algebra, A, B, C... are the variables which can take two possible values. They can be either a 1 or a 0. The laws are as follows –

##### Law of commutativity

$$A + B = B + A$$

$$A \cdot B = B \cdot A$$

##### Law of associativity

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

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

##### Distributive Law

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

#### 1.11.2 BASIC RULES OF BOOLEAN ALGEBRA

##### Rules of addition

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

##### Rules of multiplication

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

These rules should be remembered in order to solve different problems on logic circuits. They are useful to reduce the complicated logical equations into simple form. Above rules are also applicable to the variables like A, B, C. Thus, we can have –

$$A + A = A \quad \text{and} \quad A \cdot A = A$$

**EXERCISE ON THE TOPIC**

1. Draw the symbol of AND logic circuit and explain it with its truth table and its symbol.
2. How an OR gate logic circuit works? Explain its working with the help of truth table and symbol.
3. Why the output condition of NOT logic circuit is opposite to its input? Draw the symbol of NOT gate and explain its working with truth table.
4. If  $A + A = A$ , then  $AB + AB = ?$  By putting the values of the variables as 1 or 0 in the given equation and obtain the result.
5. If  $A.(B+C) = AB + AC$ , then calculate the result of  $AB.(A + B) = ?$
6. If  $A + A = A$  and  $A.A = A$ , then is it true that  $A + A = A.A$ ? By putting the values of the variables as 1 or 0 in the given equation check the result.

**1.12 PROOFS OF BOOLEAN ALGEBRA**

There are some important proofs used in Boolean algebra. The student must prepare all these proofs to solve complicated logic equations effectively. In the following proofs we have to prove that for given statement LHS = RHS. Also to prove these equations, we shall consider that the variables take two possible values either 1 or 0.

1.  $A + 1 = 1$
2.  $A + A = A$
3.  $A.1 = A$
4.  $\bar{A} + A = 1$
5.  $\bar{A}.A = 0$
6.  $A.A = A$
7.  $\bar{\bar{A}} = A$
8.  $A + AB = A$

**EXERCISE ON THE TOPIC**

1. If  $A + AB = A$ , then calculate the output of  $A + AB + ACB = ?$
2. If  $\bar{A} = 0$ , then  $A = ?$
3. According to the basic proofs, if  $A + 1 = A$ , then  $\bar{A}.B + 1 = ?$
4. Suppose  $\bar{A} = 1$  and  $\bar{A}.B = 0$ , then  $B = ?$
5. Calculate the output result of the equation  $A + AB + CA + BC = ?$
6. If  $\bar{A} + A = 1$ , then calculate  $\bar{A}\bar{B} + AB = ?$
7.  $\bar{A}.A = 0$ , then calculate the output result of  $\bar{A}\bar{B}.AB = ?$
8.  $AC + AB + ABC + ACDB + ABCDEA = ?$
9.  $C\bar{A} + \bar{A}BC + BC + A = ?$
10.  $\bar{A}.1 + A + 1 = ?$
11.  $\bar{A}.A + 1 = ?$
12.  $\bar{A}.(B + C + AB) = ?$
13.  $A.(\bar{A} + A) = ?$

### 1.13 DE MORGAN'S THEOREMS

It is an important tool in solving the complicated logical equations while analyzing the logical circuits. There are two theorems of De Morgan. They are known as theorem #1 and theorem #2. The sequence of these theorems may change, but the basic idea of both the theorems remains the same.

#### 1.13.1 FIRST THEOREM

**Definition:** It states that the complement of product is equal to the sum of the complements.

**Special case:**  $\overline{A \cdot B} = \bar{A} + \bar{B}$ , **General case:**  $\overline{A \cdot B \cdot C \dots} = \bar{A} + \bar{B} + \bar{C} \dots$

**Proof:** We shall prove above theorem by showing that LHS & RHS of given equation are equal.

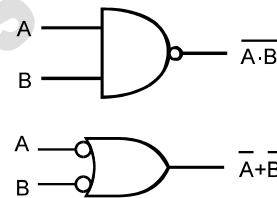
Suppose,  $LHS = \overline{AB}$

Let  $A = B = 0$ , then  $\overline{AB} = \overline{0 \cdot 0} = \overline{0} = 1$   
 Let  $A = 0, B = 1$ , then  $\overline{AB} = \overline{0 \cdot 1} = \overline{0} = 1$   
 Let  $A = 1, B = 0$ , then  $\overline{AB} = \overline{1 \cdot 0} = \overline{0} = 1$   
 Let  $A = 1, B = 1$ , then  $\overline{AB} = \overline{1 \cdot 1} = \overline{1} = 0$

A	B	LHS	RHS
0	0	1	1
0	1	0	1
1	0	0	1
1	1	0	0

Suppose,  $RHS = \bar{A} + \bar{B}$

Let  $A = B = 0$ , then  $\bar{A} + \bar{B} = \bar{0} + \bar{0} = 1 + 1 = 1$   
 Let  $A = 0, B = 1$ , then  $\bar{A} + \bar{B} = \bar{0} + \bar{1} = 1 + 0 = 1$   
 Let  $A = 1, B = 0$ , then  $\bar{A} + \bar{B} = \bar{1} + \bar{0} = 0 + 1 = 1$   
 Let  $A = 1, B = 1$ , then  $\bar{A} + \bar{B} = \bar{1} + \bar{1} = 0 + 0 = 0$



#### 1.13.2 SECOND THEOREM

**Definition:** It states that the complement of sum is equal to the product of the complements.

**Special case:**  $\overline{A + B} = \bar{A} \cdot \bar{B}$ , **General case:**  $\overline{A + B + C \dots} = \bar{A} \cdot \bar{B} \cdot \bar{C} \dots$

**Proof:** We shall prove above theorem by showing that LHS & RHS of given equation are equal.

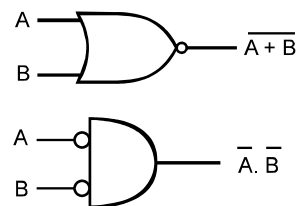
Suppose,  $LHS = \overline{A + B}$

Let  $A = B = 0$ , then  $\overline{A + B} = \overline{0 + 0} = \overline{0} = 1$   
 Let  $A = 0, B = 1$ , then  $\overline{A + B} = \overline{0 + 1} = \overline{1} = 0$   
 Let  $A = 1, B = 0$ , then  $\overline{A + B} = \overline{1 + 0} = \overline{1} = 0$   
 Let  $A = 1, B = 1$ , then  $\overline{A + B} = \overline{1 + 1} = \overline{1} = 0$

A	B	LHS	RHS
0	0	1	1
0	1	0	0
1	0	0	0
1	1	0	0

Suppose,  $RHS = \bar{A} \cdot \bar{B}$

Let  $A = B = 0$ , then  $\bar{A} \cdot \bar{B} = \bar{0} \cdot \bar{0} = 1 \cdot 1 = 1$   
 Let  $A = 0, B = 1$ , then  $\bar{A} \cdot \bar{B} = \bar{0} \cdot \bar{1} = 1 \cdot 0 = 0$   
 Let  $A = 1, B = 0$ , then  $\bar{A} \cdot \bar{B} = \bar{1} \cdot \bar{0} = 0 \cdot 1 = 0$   
 Let  $A = 1, B = 1$ , then  $\bar{A} \cdot \bar{B} = \bar{1} \cdot \bar{1} = 0 \cdot 0 = 0$



## EXERCISE ON THE TOPIC

1. Calculate the output result of the logic equation:  $Y = A + \overline{A.B}$
2. If  $Y = (\overline{A + B}).(\overline{A + B}).(A + \overline{B})$ , then calculate the value of  $Y$ .
3.  $Y = (\overline{AB}) + (\overline{BC}) + (\overline{AC})$ , then calculate the value of  $Y$ .
4. What is the output result of the equation:  $Y = (\overline{A + B}).(A + B)$ .
5. Calculate the output result of the logic equation:  $Y = (\overline{A + B + C}).(A + \overline{B + C}).(A + B + \overline{C})$ .

### 1.14 DERIVED GATES

When basic gates are connected in different combination, a new type of gate is produced known as derived gate. Basically there are two types of derived gates known as NAND gate and NOR gate.

#### 1.14.1 NAND GATE

**Definition:** A NAND gate is a logic circuit, whose output becomes low only when all its inputs are high. In other words, its output is high when at least one of its inputs is low. Its output equation, truth table and the proof of logical conditions are given below. Since it is produced by combining NOT gate and AND gate, the circuit is called NOT-AND gate, i.e. NAND gate.

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

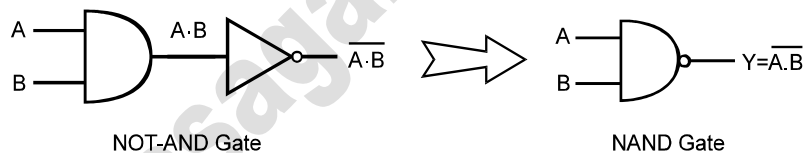
Let  $A = B = 0$ , then  $\overline{A.B} = \overline{0.0} = \overline{0} = 1$

Let  $A = 0$ ,  $B = 1$ , then  $\overline{A.B} = \overline{0.1} = \overline{0} = 1$

Let  $A = 1$ ,  $B = 0$ , then  $\overline{A.B} = \overline{1.0} = \overline{0} = 1$

Let  $A = 1$ ,  $B = 1$ , then  $\overline{A.B} = \overline{1.1} = \overline{1} = 0$

A	B	$Y = \overline{A.B}$
0	0	1
0	1	1
1	0	1
1	1	0



#### 1.14.2 NOR GATE

**Definition:** A NOR gate is a logic circuit whose output becomes low when any one of its inputs is low. In other words, i.e. its output becomes high when both of its inputs are low. Its output equation, truth table and the proof of logical conditions are given below. Since it is produced by combining NOT gate and OR gate, the circuit is called NOT-OR gate, i.e. NOR gate.

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

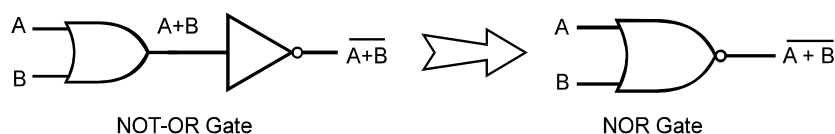
Let  $A = B = 0$ , then  $\overline{A + B} = \overline{0 + 0} = \overline{0} = 1$

Let  $A = 0$ ,  $B = 1$ , then  $\overline{A + B} = \overline{0 + 1} = \overline{1} = 0$

Let  $A = 1$ ,  $B = 0$ , then  $\overline{A + B} = \overline{1 + 0} = \overline{1} = 0$

Let  $A = 1$ ,  $B = 1$ , then  $\overline{A + B} = \overline{1 + 1} = \overline{1} = 0$

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

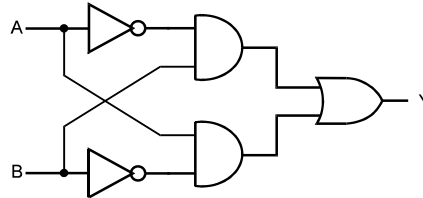


## EXERCISE ON THE TOPIC

- Calculate the value of output  $Y$ , of the following logic circuits if inputs are  $A$  and  $B$ .



- What is the output  $Y$ , of the following logic circuits if inputs are  $A$  and  $B$ .



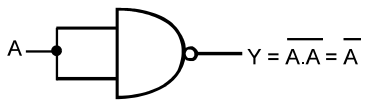
- What will happen if the output of one NOT gate is connected to the input of the second NOT gate? Explain with diagram and obtain the output logic equation.

### 1.15 UNIVERSAL BUILDING BLOCKS

The above logic circuits NAND gate and NOR gate are called universal building blocks, because using these logic circuits, we can construct any type of logic circuit i.e. AND, OR, NOT, NAND or NOR gate. For this, the rules mentioned in *Topic #2.4* and the De Morgan's both theorems are important.

#### 1.15.1 NOT GATE USING NAND GATE

When both inputs of a NAND gate are connected to each other and *treated as single input* it forms a NOT gate. Here both input terminals of NAND gate are connected together. So input signal at terminal  $A$  will be equally distributed to both inputs of NAND gate. Hence, the output will be the same as NOT gate with a single input. Thus, any NAND gate with any number inputs can be converted into equivalent NOT gate using this method.



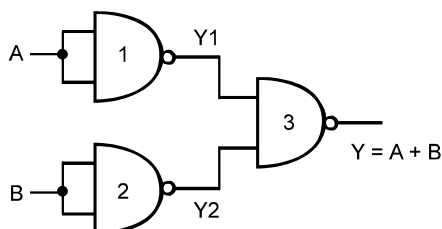
#### Mathematical analysis of the logic circuit:

$$\text{When } A = 0, Y = \overline{0 \cdot 0} = \overline{0} = 1 = \bar{A} \quad i.e. \quad Y = \bar{A}$$

$$\text{When } A = 1, Y = \overline{1 \cdot 1} = \overline{1} = 0 = \bar{A} \quad i.e. \quad Y = \bar{A}$$

#### 1.15.2 OR GATE USING NAND GATE

When three NAND gates are connected in the circuit as shown below, an OR gate is formed. In the following circuit, NAND gate 1 and 2 are used as NOT gates and their outputs are connected to the input terminals of NAND gate 3. The mathematical analysis of the circuit is given below –



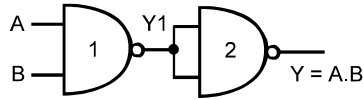
$$Y_1 = \bar{A}, \quad Y_2 = \bar{B} \quad \therefore Y = \overline{Y_1 \cdot Y_2}$$

$$\therefore Y = \overline{\bar{A} \cdot \bar{B}} = \bar{\bar{A}} + \bar{\bar{B}} = A + B$$

Thus, by applying De Morgan's theorem, we obtain the output logical equation of the circuit. So using three NAND gates we can obtain one OR gate. However, this circuit is only experimental idea to understand basic working of NAND gate.

### 1.15.3 AND GATE USING NAND GATE

When two NAND gates are connected one after another, as shown in the following circuit, an AND gate is formed. Here, NAND gate 1 has two input terminals and NAND gate 2 is used as NOT gate. The mathematical analysis of the circuit is given below.



$$Y_1 = \overline{AB} \quad \text{and} \quad Y = \overline{Y_1}$$

$$\therefore Y = \overline{\overline{AB}} = A.B$$

Thus, by applying simple Boolean algebra, we obtain the output logical equation of the circuit. So using two NAND gates we can obtain one AND gate. However, this circuit is only experimental idea to understand basic working of NAND gate.

### 1.16 IDENTITIES OF BOOLEAN ALGEBRA

Understanding the logic circuits give us an effective tool in solving complicated logical equations. Following are the identities, in which we have to prove that LHS = RHS.

1.  $(\bar{A} + \bar{B}).(\bar{A} + B).(A + \bar{B}) = AB$
2.  $(\bar{A} + B + C).(A + \bar{B} + C).(A + B + \bar{C}) = AB + BC + AC + \bar{A}.\bar{B}.\bar{C}$
3.  $A + \bar{A}.B = A + B$
4.  $(A + B).(A + C) = A + BC$
5.  $\overline{AB + BC + CA} = \bar{A}.\bar{B} + \bar{B}.\bar{C} + \bar{C}.\bar{A}$
6.  $AB + AC + B\bar{C} = AC + B\bar{C}$
7.  $\overline{\bar{A} + B} + \bar{A} + \bar{B} = A$
8.  $AB + CAB + DBCA + DBACE = A$
9.  $A.(\bar{A} + B) = AB$
10.  $\bar{B}.\bar{C} + \bar{D}.\bar{A}.\bar{C} + \bar{D}.A.B.\bar{C} + D.B.\bar{C} = \bar{C}$

### 1.17 EXCLUSIVE OR GATE (EX-OR GATE)

**Definition:** An Exclusive OR gate is defined as a logic circuit whose output becomes high only when its both inputs are UNEQUAL.

It is a special type of logic circuit used in digital electronics. Its logic equations are –

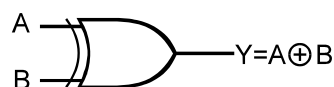
$$Y = \bar{A}.B + A.\bar{B}$$

$$Y = A \oplus B$$

The Ex–OR addition rules for all possible combinations of inputs are given below –

- |   |                               |
|---|-------------------------------|
| a) When $A = B = 0$ , $Y = \bar{A}.B + A.\bar{B} = 0$       | $A \oplus B = 0 \oplus 0 = 0$ |
| b) When $A = 0$ , $B = 1$ , $Y = \bar{A}.B + A.\bar{B} = 1$ | $A \oplus B = 0 \oplus 1 = 1$ |
| c) When $A = 1$ , $B = 0$ , $Y = \bar{A}.B + A.\bar{B} = 1$ | $A \oplus B = 1 \oplus 0 = 1$ |
| d) When $A = B = 1$ , $Y = \bar{A}.B + A.\bar{B} = 0$       | $A \oplus B = 1 \oplus 1 = 0$ |

A	B	$Y = A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0





### 1.17.1 EX-OR GATE USING NAND GATES

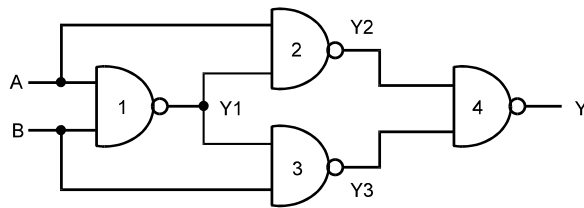
When four NAND gates are connected in particular combination as shown in the following circuit, one Ex-OR gate is formed. In this circuit, gate 1 has two inputs A & B. Its output is connected in parallel to one input terminal of gate 2 and gate 3. Their other inputs are connected to terminal A & B. Finally, the outputs of gate 2 and gate 3 are connected to two input terminals of gate 4. The output of gate 4 is Y.

Now  $Y_1$  is the output of gate 1, which is given by –

$$Y_1 = \overline{A \cdot B}$$

Now  $Y_2$  and  $Y_3$  are the outputs of gate 2 and gate 3. They are given as –

$$Y_2 = \overline{A \cdot Y_1} = \overline{A \cdot (\overline{A \cdot B})} \quad \text{and} \quad Y_3 = \overline{B \cdot Y_1} = \overline{B \cdot (\overline{A \cdot B})}$$



A	B	Y = A ⊕ B
0	0	0
0	1	1
1	0	1
1	1	0

Finally, the Y is the output of gate 4, which is given by –

$$Y = \overline{Y_2 \cdot Y_3} = \overline{(\overline{A \cdot (\overline{A \cdot B})}) \cdot (\overline{B \cdot (\overline{A \cdot B})})}$$

Now applying De Morgan's theorem to the equation,  $Y = \overline{Y_2 \cdot Y_3}$ , we get –

$$\begin{aligned} \therefore Y &= \overline{Y_2} + \overline{Y_3} = \overline{(\overline{A \cdot (\overline{A \cdot B})})} + \overline{(\overline{B \cdot (\overline{A \cdot B})})} \\ \therefore Y &= A \cdot (\overline{A \cdot B}) + B \cdot (\overline{A \cdot B}) \end{aligned}$$

Now finally applying De Morgan's theorem to the brackets, we get –

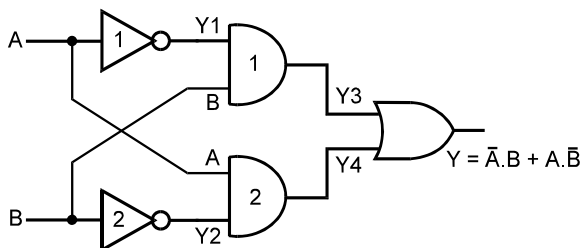
$$\begin{aligned} \therefore Y &= A \cdot (\overline{A} + \overline{B}) + B \cdot (\overline{A} + \overline{B}) \\ \therefore Y &= A \cdot \overline{A} + A \cdot \overline{B} + B \cdot \overline{B} + \overline{A} \cdot B \end{aligned}$$

But according to Boolean algebra,  $A \cdot \overline{A} = 0$  and  $B \cdot \overline{B} = 0$ . So finally, we get –

$$Y = A \cdot \overline{B} + \overline{A} \cdot B = Y = A \oplus B$$

### 1.17.2 EX-OR GATE USING BASIC GATES

When all the basic gates are connected in particular combination as shown below, we get Ex-OR logic circuit. The mathematical analysis of the circuit is given below –



$$Y_1 = \overline{A} \text{ and } Y_2 = \overline{B}$$

$$\text{Similarly, } Y_3 = \overline{A} \cdot B \text{ and } Y_4 = A \cdot \overline{B}$$

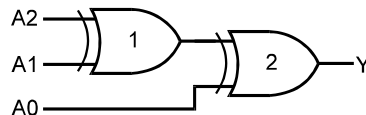
Now the final output Y will be the output of OR gate, so we get –

$$Y = Y_3 + Y_4 = A \cdot \overline{B} + \overline{A} \cdot B \quad \text{i.e. } Y = A \oplus B$$

### 1.17.3 PARITY AND PARITY CHECKER

Always remember that parity means a single binary 1. Thus, if a binary number is given as:  $(1101)_2$  then it contains three parities. Such binary number is called as odd parity binary “word”. But if a binary number is given as:  $(1001101)_2$  then it contains four parities. Then such number is called even parity binary “word”.

The Ex-OR gate is ideal for testing the number of parities present in very large binary word. This logic circuit (gate) can recognize odd number of parities. Thus, odd parity input to the following circuit produces the output as  $Y = 1$  but when even parity input gives  $Y = 0$ . This is shown in the following circuit –



Here, if  $Y = A_3A_2A_1 = 110$ , then the output of the circuit will be –

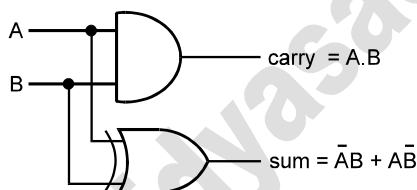
$$Y = 0$$

Here, if  $Y = A_3A_2A_1 = 010$ , then the output of the circuit will be –

$$Y = 1$$

### 1.18 HALF ADDER

It is combination of basic logic gates. Using this circuit, we can add two binary digits at a time. In half adder, there are two inputs  $A$  and  $B$  with two outputs SUM and CARRY. The SUM is the output of Ex-OR gate and CARRY is the output of AND gate. Remember that SUM is at LSB position and CARRY is at MSB position.



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

#### Mathematical analysis of the logic circuit:

Let  $A = B = 0$ , then SUM = 0 and CARRY = 0, so the result of addition will be 00.

Let  $A = 0, B = 1$ , then SUM = 1 and CARRY = 0, so the result of addition will be 01.

Let  $A = 1, B = 0$ , then SUM = 1 and CARRY = 0, so the result of addition will be 01.

Let  $A = B = 1$ , then SUM = 0 and CARRY = 1, so the result of addition will be 10.

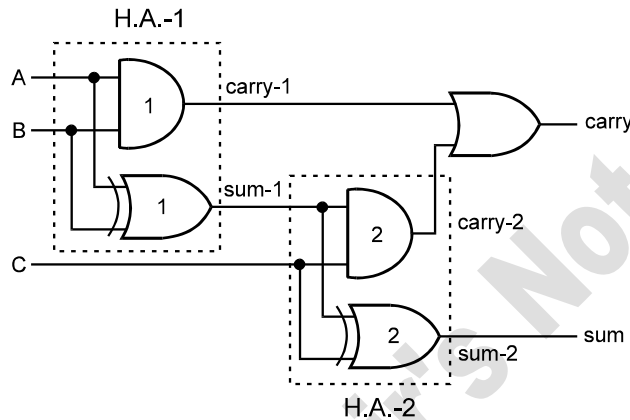
So this analysis shows that any two binary digits can be added using the circuit. Remember that the circuit uses rules of binary addition, as follows –

$$\begin{aligned} 0 + 0 &= 0 \\ 0 + 1 &= 1 \\ 1 + 0 &= 1 \\ 1 + 1 &= 10 \end{aligned}$$

### 1.18.1 FULL ADDER

It is also a combination of basic logic gates. It uses two half adder circuits connected one after another, as shown below. Using this circuit, we can add three binary digits at a time. In full adder, there are three inputs  $A$ ,  $B$  and  $C$  with two outputs  $SUM$  and  $CARRY$ . The  $SUM$  and  $CARRY$  are the outputs. Remember that  $SUM$  is at LSB position and  $CARRY$  is at MSB position.

Here the outputs of first half adder are  $CARRY-1$  and  $SUM-1$ . The output  $SUM-1$  is connected to one input of next half adder. Its  $SUM$  output is taken as independent output. The  $CARRY-1$  and  $CARRY-2$  outputs of both half adders are **ORed** together to form a single  $CARRY$  output.



The logic equations of the circuit are given below –

$$CARRY1 = A \cdot B$$

$$SUM1 = A \oplus B$$

$$CARRY2 = (A \oplus B) \cdot C$$

$$SUM2 = A \oplus B \oplus C$$

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

$$SUM = A \oplus B \oplus C$$

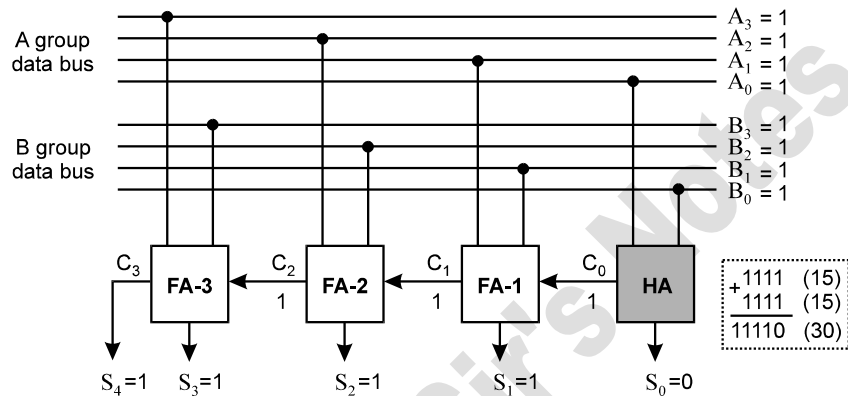
A	B	C	CARRY1	SUM1	CARRY2	SUM2	CARRY	SUM
0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0	1
0	1	0	0	1	0	1	0	1
0	1	1	0	1	1	0	1	0
1	0	0	0	1	0	1	0	1
1	0	1	0	1	1	0	1	0
1	1	0	1	0	0	0	1	0
1	1	1	1	0	0	1	1	1

In this way, the outputs  $CARRY$  and  $SUM$  will become 11, only when all the inputs are 1. And when only two inputs are 1, then the outputs  $CARRY$  and  $SUM$  become 10.

### 1.19 THE 4-BIT BINARY ADDER

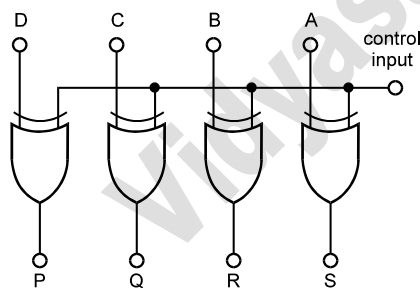
This circuit can add **two** 4-bit binary numbers. The process of addition in the circuit takes place as  $A_3A_2A_1A_0 + B_3B_2B_1B_0 = S_4S_3S_2S_1S_0$ . As shown in following figure, suppose  $A_3A_2A_1A_0 = 1111$  and  $B_3B_2B_1B_0 = 1111$ , then we get the result of addition as  $S_4S_3S_2S_1S_0 = 11110$  as follows –

$A_0 = 1$  and  $B_0 = 1, \therefore C_0 = 1$  and  $S_0 = 0$ .  
 $A_1 = 1$  and  $B_1 = 1, \therefore C_1 = 1$  and  $S_1 = 1$ .  
 $A_2 = 1$  and  $B_2 = 1, \therefore C_2 = 1$  and  $S_2 = 1$ .  
 $A_3 = 1$  and  $B_3 = 1, \therefore C_3 = S_4 = 1$  and  $S_3 = 1$ .  
 Finally, we get the output as  $S_4S_3S_2S_1S_0 = 11110$ .



### 1.20 CONTROLLED INVERTER USING EX-OR GATE

The Ex-OR gate can be used as controlled inverter circuit. So using four Ex-OR gates, four bit inverter circuit is shown in the following circuit diagram. When any one input terminal of Ex-OR gate is connected to logic-1, the gate works as NOT gate. Hence, Ex-OR gate can be used as controlled inverter. The circuit produces following results –



More examples of working of the circuit

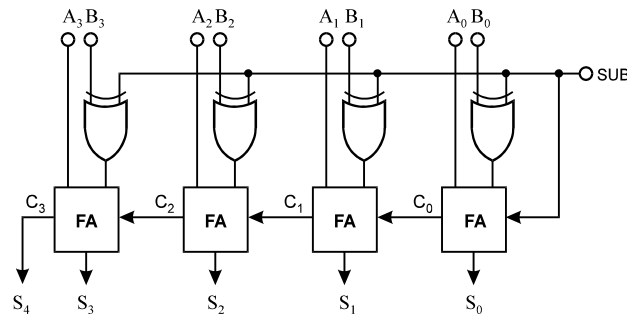
Control input	Inputs DCBA	Outputs PQRS
0	1101	1101
1	1101	0010
0	1000	1000
1	1000	0111

When  $DCBA = 1011$  and  $C.I. = 0$ , then  $PQRS = 1011$ .  
 When  $DCBA = 1011$  and  $C.I. = 1$ , then  $PQRS = 0100$ , which is the 1's complement of 1011.  
 When  $DCBA = 1111$  and  $C.I. = 0$ , then  $PQRS = 1111$ .  
 When  $DCBA = 1111$  and  $C.I. = 1$ , then  $PQRS = 0000$ , which is the 1's complement of 1111.

In this way the circuit produces the 1's complement of any number connected at its input when the value of  $C.I. = 1$ . This circuit is very useful when we want to convert large binary number into its equivalent 1's complement form.

### 1.21 ADDER/SUBTRACTOR LOGIC CIRCUIT

This circuit can perform both addition & subtraction just by changing only one input known as “SUB” input. It is actually a combination of 4-bit adder circuit and controlled inverter circuit using Ex-OR gates. The four bit binary adder/subtractor circuit is given below –



**Addition process:** For addition keep  $SUB = 0$  and then connect two different binary numbers which we want to add. In this case, the circuit works as general-purpose binary adder circuit. All Ex-OR gates send the data at their outputs without any change, since the  $SUB$  input is in logic 0 state. Thus, at output we get addition result as follows  $A_3A_2A_1A_0 + B_3B_2B_1B_0 = S_4S_3S_2S_1S_0$ .

**Subtraction process:** For subtraction keep  $SUB = 1$  and then connect two different binary numbers at two groups of inputs. The binary number which we want to subtract must be connected as  $B_3B_2B_1B_0$  number. Now the circuit works as subtractor. The circuit performs subtraction process using 2's complement method. The Ex-OR gates generate 1's complement of 'B' group number. Then this 1's complement is added to a 1 through  $SUB = 1$  at the input of rightmost full adder. Thus, at output we get subtraction result as follows –

$$\begin{array}{r}
 A_3 A_2 A_1 A_0 \\
 + \quad D_3 D_2 D_1 D_0 \leftarrow \text{2's complement} \\
 \hline
 C_4 S_3 S_2 S_1 S_0 \\
 X \leftarrow \text{carry is ignored}
 \end{array}$$

of  $B_3 B_2 B_1 B_0$

## Self Examination

### Objective questions

1. The output of \_\_\_\_\_ logic circuit becomes equal to logic-1 when both of its inputs are at logic-0.
2. When any one input of OR gate is at logic-1, its output is at \_\_\_\_\_.
3. When a NOT gate is connected at the input of an AND gate, the new gate produced is called as \_\_\_\_\_.
4. In half adder circuit, when both of its inputs are logic-1, then its output is \_\_\_\_\_ and \_\_\_\_\_.
5. When one input of 2-input Ex-OR gate is connected to logic-1, then the circuit behaves as \_\_\_\_\_ logic circuit.
6. According to De Morgan's theorem,  $\overline{A + B + C} = \underline{\hspace{2cm}}$ .
7. To construct an OR gate using NAND gates, as universal building block, the number of NAND gates required will be \_\_\_\_\_.
8. When both inputs of an Ex-OR gate are at equal logic level, i.e. either at logic-1 or logic-0, then its output will be at \_\_\_\_\_.
9. If two NOT gates are connected one after another, such that the output of first is connected to the input of second, and if initial input is at logic-0 level, then the final output will be at \_\_\_\_\_.
10. The addition process of logic values can be performed using \_\_\_\_\_ gate.
11. Bubbled OR gate is equivalent to \_\_\_\_\_ gate.
12.  $C + 1 = \underline{\hspace{2cm}}$ .

### Short answer questions (3 Marks)

1. Draw the symbol of NOT gate and explain its working with truth table.
2. Explain why NAND gate is known as universal building block?
3. How Ex-OR gate can be used as NOT gate? Explain.
4. How to convert a NOR gate into a NOT gate? Explain with diagram.
5. Prove that  $C + AC + ABC + ABCD = C$
6. Explain the working of half adder using proper logic diagram.
7. What is parity? Explain the working of parity checker.
8. What is the basic difference between inclusive OR gate and exclusive OR gate? Explain with diagrams.

### Long answer questions (4 Marks)

1. Explain the working of NAND gate as universal building block by drawing different circuits as NOT, OR and AND gates.
2. How 4-bit binary adder circuit works? Explain with diagram and one example.
3. What is controlled inverter? How Exclusive OR gate can be used as controlled inverter? Explain with diagram.
4. Draw the circuit of 4-bit adder/subtractor circuit using full adders and the circuit of controlled inverter.
5. Define and prove De Morgan's both theorems with diagrams and truth tables.
6. Give any four logic equations to prove the Boolean algebra.
7. How full adder circuit works? Explain with diagram.
8. Explain the working of Ex-OR gate using basic gates. Draw the necessary diagram and explain the working of circuit with derivation of output equation.
9. How to use NAND gates to produce one exclusive OR gate? Explain with diagram and derivation of output equation.
10. Prove that NOR can gate be used as universal building block with diagram and derivations of each circuit.
11. Draw the logic circuit of 4-input Ex-OR gate and explain its working such that its output equation will be  $Y = A \oplus B \oplus C \oplus D$ .
12. Define and explain the three basic gates. Write their truth tables also.
13. Solve the following identities and prove that LHS = RHS.

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

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

$$AB + CAB + DBCA + DBACE =$$

$$A \cdot (\bar{A} + B) = AB$$

### Conceptual study questions



1. Is there an alternative to check the parities using some other logic circuit? Explain with reasons.
2. How you can produce 10 input Exclusive OR gate? Explain with diagram.
3. Can we replace controlled inverter circuit used in 4-bit binary adder with simple NOT logic circuit? Explain.
4. Try to solve the famous Boatman's puzzle by drawing its equivalent logic circuit and final output equation.

Vidyasagar Sir's Notes