# CHAPTER 4 BOOLEAN ALGEBRA

### 4.1 Definition

Boolean algebra is the mathematics of digital logic in which the values of the variables are the truth values true and false, usually denoted 1 and 0, respectively. Basic knowledge of Boolean algebra is indispensable to the study and analysis of logic circuits. *Variable, complement*, and literal are terms used in Boolean algebra. A variable is a symbol (usually an italic uppercase letter or word) used to represent an action, a condition, or data. Any single variable can have only a 1 or a 0 value. The complement is the inverse of a variable and is indicated by a bar over the variable (overbar).

There are four connecting symbols used in Boolean algebra.

- 1. Equal sign (=): This refers to the sign of equality as in mathematics.
- 2. Multiplication sign ( $\cdot$ ): It refers to the AND operation.
- 3. Plus sign (+): This refers to the OR operation.

4. Inversion sign (') or (-): This operation performs a complement of the input given to the logic gate.

### 4.2 Laws of Boolean Algebra

The basic laws of Boolean algebra—the commutative laws for addition and multiplication, the associative laws for addition and multiplication, and the distributive law—are the same as in ordinary algebra. Each of the laws is illustrated with two or three variables, but the number of variables is not limited to this.

#### 4.2.1 Commutative Laws

The commutative law of addition for two variables is written as

```
A + B = B + A
```



Application of commutative law of multiplication.

#### 4.2.2 Associative Laws

The associative law of addition is written as follows for three variables:



Application of associative law of addition.

The associative law of multiplication is written as follows for three variables:

A (BC) = (AB) C

This law states that it makes no difference in what order the variables are grouped when ANDing more than two variables. Figure illustrates this law as applied to 2-input AND gates.



#### 4.2.3 Distributive Law

The distributive law is written for three variables as follows:

A (B + C) = AB + AC



### 4.3 Rules of Boolean Algebra

The table below lists 12 basic rules that are useful in manipulating and simplifying Boolean expressions.

Basic reles of Boolea	in algeora
<b>1.</b> $A + 0 = A$	7. $A \cdot A = A$
<b>2.</b> $A + 1 = 1$	8. $A \cdot \overline{A} = 0$
3. $A \cdot 0 = 0$	9. $\overline{\overline{A}} = A$
<b>4.</b> $A \cdot 1 = A$	<b>10.</b> $A + AB = A$
5. $A + A = A$	<b>11.</b> $A + \overline{AB} = A + B$
<b>6.</b> $A + \overline{A} = 1$	12. $(A + B)(A + C) = A + BC$

A, B, or C can represent a single variable or a combination of variables.

#### **Rule 1:** A + 0 = A

A variable ORed with 0 is always equal to the variable. If the input variable A is 1, the output variable X is 1, which is equal to A. If A is 0, the output is 0, which is also equal to A.

$$A = 1 \qquad X = 1 \qquad A = 0 \qquad X = 0$$

$$X = A + 0 = A$$

**Rule 2:** A + 1 = 1 A variable ORed with 1 is always equal to 1. A 1 on an input to an OR gate produces a 1 on the output, regardless of the value of the variable on the other input.



### Rule 3: $A \cdot 0 = 0$

A variable ANDed with 0 is always equal to 0. Any time one input to an AND gate is 0, the output is 0, regardless of the value of the variable on the other input.

$$A = 1$$

$$0$$

$$X = 0$$

$$X = 0$$

$$X = 0$$

$$X = 0$$

#### Rule 4: $A \cdot 1 = A$

A variable ANDed with 1 is always equal to the variable. If A is 0, the output of the AND gate is 0. If A is 1, the output of the AND gate is 1 because both inputs are now 1s.



#### Rule 5: A + A = A

A variable ORed with itself is always equal to the variable. If A is 0, then 0 + 0 = 0; and if A is 1, then 1 + 1 = 1.



### Rule 6: $A + \overline{A} = 1$

A variable ORed with its complement is always equal to 1. If A is 0, then 0 + 0 = 0 + 1 = 1. If A is 1, then 1 + 1 = 1 + 0 = 1.

$$\begin{array}{c} A = 0 \\ \overline{A} = 1 \end{array} \qquad X = 1 \qquad \qquad A = 1 \\ \overline{A} = 0 \end{array} \qquad X = 1 \\ \overline{A} = 0 \end{array}$$

X = A + A = 1

#### **Rule 7:** $A \cdot A = A$

A variable ANDed with itself is always equal to the variable. If A = 0, then 0 # 0 = 0; and if A = 1, then 1 # 1 = 1.



### Rule 8: $A \cdot \overline{A} = 0$

A variable ANDed with its complement is always equal to 0. Either A or A will always be 0; and when a 0 is applied to the input of an AND gate, the output will be 0 also.



### Rule 9: $\overline{A} = A$

The double complement of a variable is always equal to the variable. If you start with the variable A and complement (invert) it once, you get A. If you then take A and complement (invert) it, you get A, which is the original variable.

$$A = 0 - \boxed{A = 1} - \boxed{A = 0} - \overline{A} = 0$$

$$A = 1 - \boxed{A = 1} - \boxed{A$$

A = A

#### Rule 10: A + AB = A

This rule can be proved by applying the distributive law, rule 2, and rule 4 as follows:

$$A + AB = A \cdot 1 + AB = A (1 + B)$$
 Factoring (distributive law)

= A.1 Rule 2: (1 + B) = 1

 $= A \qquad \text{Rule 4: } A . 1 = A$ 

The proof is shown in below, which shows the truth table and the resulting logic circuit simplification.

Rule 10: A-	+AB = A			-
A	В	AB	A + AB	_
0	0	0	0	
0	1	0	0	
1	0	0	1	
1	1	1	1	
t	equ	ual	1	Astraight connection

### Rule 11: $A + \overline{A}B = A + B$

This rule can be proved as follows:

 $A + \overline{AB} = (A + AB) + \overline{AB}$  Rule 10: A = A + AB =(AA + AB) +  $\overline{AB}$  Rule 7: A = AA =AA + AB + A $\overline{A}$  +  $\overline{AB}$  Rule 8: adding A $\overline{A}$  = 0 = (A +  $\overline{A}$ )(A + B) Factoring = 1. (A + B) Rule 6: A + A = 1 = A + B Rule 4: drop the 1 The proof is shown in the table below, which shows the truth table and the resulting logic circuit simplification.

Rule 11: $A + \overline{AB} = A + B$								
A	В	ĀB	$A + \overline{AB}$	A + B	_			
0	0	0	0	0				
0	1	1	1	1				
1	0	0	1	1				
1	1	0	1	1	A			
			equ	ual 🔟				

#### **Rule 12:** (A + B) (A + C) = A + BC

This rule can be proved as follows:

(A + B)(A + C) = AA + AC + AB + BC Distributive law = A + AC + AB + BC Rule 7: AA = A = A(1 + C) + AB + BC Factoring (distributive law)  $= A \cdot 1 + AB + BC$  Rule 2: 1 + C = 1 = A(1 + B) + BC Factoring (distributive law)  $= A \cdot 1 + BC$  Rule 2: 1 + B = 1 = A + BC Rule 4: A. 1 = A

The proof is shown in the table below, which shows the truth table and the resulting logic circuit simplification.

	B	C	A + B	A + C	(A+B)(A+C)	BC	A + BC	
9	0	0	0	0	0	0	0	
8	0	1	0	1	0	0	0	A
	1	0	1	0	0	0	0	
	1	1	1	1	1	1	1	c
	0	0	1	1	1	0	1	850 - B
	0	10	1	1	1	0	1	
	1	0	1	1	1	0	1	A
- L	1	1	1	1	1	1	1	р С

### **H.W:**

- 1. Apply the associative law of addition to the expression A + (B + C + D).
- 2. Apply the distributive law to the expression A(B + C + D).

### 4.4 DeMorgan's Theorems

DeMorgan, a mathematician, proposed two theorems that are an important part of Boolean algebra. In practical terms, DeMorgan's theorems provide mathematical verification of the equivalency of the NAND and negative-OR gates and the equivalency of the NOR and negative-AND gates.

### 4.4.1 DeMorgan's first theorem

The complement of a product of variables is equal to the sum of the complements of the variables.

Stated another way,

The complement of two or more ANDed variables is equivalent to the OR of the complements of the individual variables.

The formula for expressing this theorem for two variables is

$$\overline{XY} = \overline{X} + \overline{Y} \qquad \dots \qquad (1)$$

#### 4.4.2 DeMorgan's second theorem

The complement of a sum of variables is equal to the product of the complements of the variables.

Stated another way,

The complement of two or more ORed variables is equivalent to the AND of the

complements of the individual variables.

The formula for expressing this theorem for two variables is

## $\overline{X+Y} = \overline{X}\overline{Y} \quad \dots \dots$

Figure below shows the gate equivalencies and truth tables for Equations 1 and 2.

(2)

		Inpu X	ts Y	$\begin{array}{c} \mathbf{Output} \\ \overline{XY}  \overline{X} + \overline{Y} \end{array}$	
	$\equiv \frac{X}{V} - \frac{1}{X} + \frac{1}{Y}$	0	0	1	1
	1 2	0	1	1	1
NAND	Negative-OR	1	0	1	1
		1	1	0	0
		Inpu X	ts Y	$\frac{\text{Out}_{I}}{\overline{X+Y}}$	$\overline{X} \overline{Y}$
x	- x			411	
$X \rightarrow X + Y$	$x \equiv \frac{\pi}{y} = 0$ $(x = 1)$	0	0	1	1
		0	1	0	0
NOR	Negative-AND	1	0	0	0
		1	1	0	0

Example: Apply DeMorgan's theorems to the expressions  $\overline{XYZ}$  and  $\overline{X + Y + Z}$ . Sol.

$$\overline{XYZ} = \overline{X} + \overline{Y} + \overline{Z}$$
$$\overline{X + Y + Z} = \overline{X} \overline{Y} \overline{Z}$$

Example: Apply DeMorgan's theorems to the expressions  $\overline{WXYZ}$  and  $\overline{W + X + Y + Z}$ . Sol.

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

**Example:** Prove that A + AB = A

Sol.

A+AB = A (1+B) $= A \cdot 1 = A$ 

**Example:** Prove that  $A + \overline{AB} = A + B$ 

Sol.

$$A + \overline{AB} = (A + AB) + \overline{AB}$$
$$= A + B(A + \overline{A})$$
$$= A + B.1$$
$$= A + B$$

**Example:** Apply DeMorgan's theorems to each expression:

(a) 
$$\overline{(\overline{A} + \overline{B}) + \overline{C}}$$
  
(b)  $\overline{(\overline{A} + \overline{B}) + CD}$   
(c)  $\overline{(A + \overline{B})\overline{C}\overline{D} + E + \overline{F}}$ 

Sol.

(a) 
$$(\overline{A + B}) + \overline{C} = (\overline{A + B})\overline{C} = (A + B)C$$
  
(b)  $(\overline{A + B}) + C\overline{D} = (\overline{\overline{A} + B})\overline{CD} = (\overline{\overline{A}}\overline{B})(\overline{C} + \overline{D}) = A\overline{B}(\overline{C} + \overline{D})$   
(c)  $(\overline{A + B})\overline{C}\overline{D} + E + \overline{F} = ((A + B)\overline{C}\overline{D})(\overline{E + F}) = (\overline{A}\overline{B} + C + D)\overline{E}F$ 

**Example:** Prove that (A + B)(A + C) = A + BC

Sol.

$$(A+B)(A+C) = AA + AC + AB + BC$$
  
= A + AC + AB + BC  
= A + (1+C) + AB + BC  
= A.1 + AB + BC  
= A + AB + BC  
= A(1 + B) + BC  
= A.1 + BC  
= A + BC

### **H.W.:**

Apply DeMorgan's theorems to the following expressions:

- 1.  $\overline{ABC} + (\overline{\overline{D} + E})$  2.  $\overline{(A + B)C}$  3.  $\overline{A + B + C} + \overline{\overline{D}E}$
- 2. Apply DeMorgan's theorems to the expression  $\overline{\overline{ABC} + D + E}$ .

#### 4.5 Logic Simplification Using Boolean Algebra

A logic expression can be reduced to its simplest form or changed to a more convenient form to implement the expression most efficiently using Boolean algebra.

Example: Using Boolean algebra techniques, simplify this expression: AB + A(B + C) + B(B + C)

Sol.

AB + AB + AC + BB + BC distributive law to the second and third terms AB + AB + AC + B + BC (BB = B) rule 7

AB + AC + B + BC (AB + AB = AB) rule 5

AB + AC + B (B + BC = B) rule 10

B + AC (AB + B = B) rule 10

The figure below shows that the simplification process significantly reduced the number of logic gates required to implement the expression. Part (a) shows that 5 gates are required to implement the expression in its original form; however, only 2 gates are needed for the simplified expression, shown in part (b). It is important to realize that these two gate circuits are equivalent.



**Example:** Simplify the following Boolean expression:

$$\overline{AB + AC} + \overline{ABC}$$

Sol.

 $(\overline{AB})(\overline{AC}) + \overline{ABC} \quad \text{DeMorgan's theorem}$   $(\overline{A} + \overline{B})(\overline{A} + \overline{C}) + \overline{ABC} \quad \text{DeMorgan's theorem}$   $\overline{AA} + \overline{AC} + \overline{AB} + \overline{BC} + \overline{ABC} \quad \text{distributive law to the two terms}$   $\overline{AB} + \overline{ABC} = \overline{AB}(1 + C) = \overline{AB} \quad \text{rule 7} (\overline{AA} = \overline{A}) = A) \text{ to the first term}$   $\overline{A} + \overline{AC} + \overline{AB} + \overline{BC} \quad \text{rule 10 to the third and last terms.}$   $\overline{A} + \overline{AC} = \overline{A}(1 + \overline{C}) = \overline{A} \quad \text{rule 10 to the first and second terms}$   $\overline{A} + \overline{AB} + \overline{BC}$   $\overline{A} + \overline{AB} = \overline{A}(1 + \overline{B}) = \overline{A} \quad \text{rule 10 to the first and second terms.}$   $\overline{A} + \overline{BC}$ 

#### **H.W.:**

1. Simplify the following Boolean expressions:

a.  $A + AB + A\overline{B}C$  b.  $(\overline{A} + B)C + ABC$  c.  $A\overline{B}C(BD + CDE) + A\overline{C}$ 

2. Implement each expression in Question 1 as originally stated with the appropriate logic gates. Then implement the simplified expression, and compare the number of gates.