Frantz Fanon University

Break chains, visible or invisible - build up from ruins, personal or social.

Digital Logic Design

FRANTZ FANON UNIVERSITY

COLLAGE OF COMPUTING AND IT

DEPARTMENT OF ICT

Suleiman Gargaare

MSc in Computer Science, MSc in Project Management & BSc in Information Technology.

Lecturer

suleiman.gargaare@gmail.com

Course Contents

- Chapter 1 - Number Systems
- Chapter 2 - Binary Arithmetic
- Chapter 3 - Logic Gates
- **Chapter 4 - Boolean Algebra**
- Chapter 5 - K-Map

Digital Logic Design

Chapter 4 – Boolean Algebra

Introduction

- In 1854, **George Boole** published a work titled *An Investigation of the Laws of Thought*, on which are founded the Mathematical Theories of Logic and Probabilities. It was in this publication that a “logical algebra”, known today as **Boolean algebra**, was formulated. Boolean algebra is a convenient and systematic way of expressing and analyzing the operation of logic circuits.
- This chapter covers the **Laws**, **Rules** and **Theories** of Boolean Algebra.

Key Terms

- Variable
- Complement
- Sum term
- Product term
- Karnaugh Map? Assignment?

Cont..

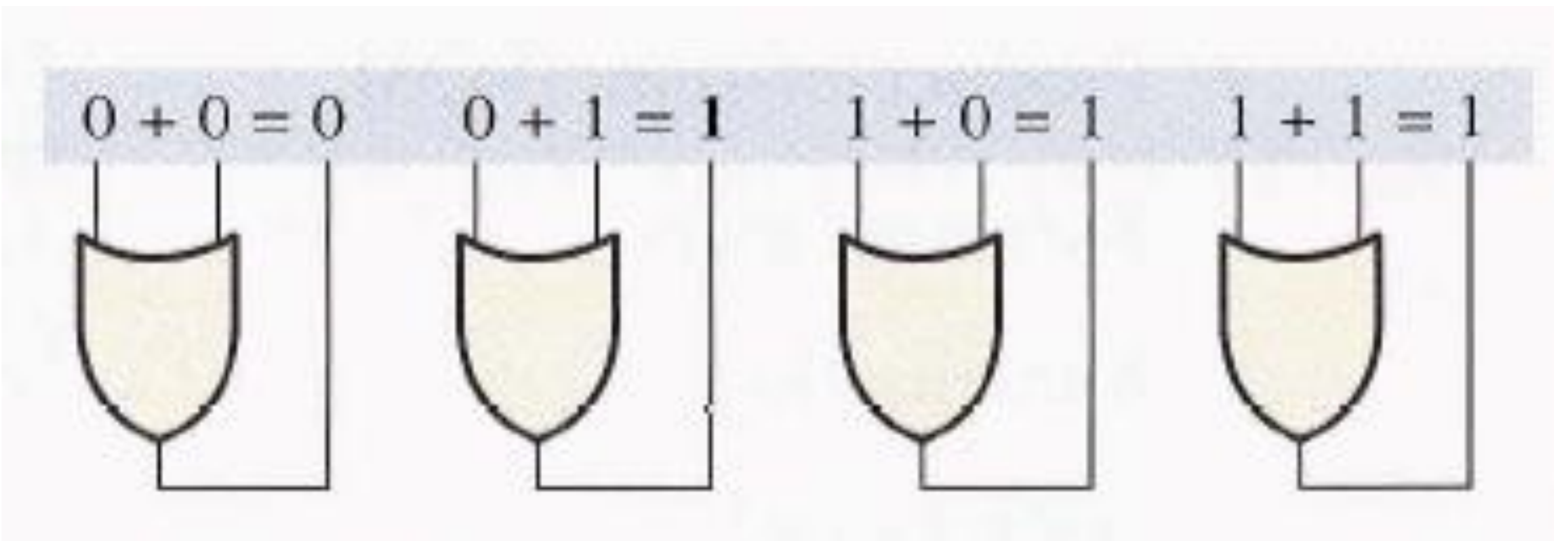
- Boolean algebra *is the mathematics of digital systems*. A basic knowledge of Boolean algebra is indispensable to the study and analysis of logic circuits. In the last chapter, Boolean operations and expressions in terms of their relationship to **NOT**, **AND**, **OR**, **NAND**, and **NOR** gates were introduced.
- The **Variable**, **Complement** and **Literal** are terms used Boolean Algebra.

Cont..

- A variable is a symbol *(usually an italic uppercase letter)* used to represent a logical quantity. Any single variable can have a **1** or a **0** value.
- The complement is the inverse of a variable and is indicated by a bar over the variable *(overbar).* For example, the complement of the variable **A** is **A'**. If **A = 1**, then **A' = 0**.
- The complement of the variable **A** is read as "**not A**" or "**A bar**".
- A literal is a variable or the complement if a variable.

Boolean Addition

- Recall from Chapter 3 that Boolean addition is equivalent to the **OR** operation and the basic rules are illustrated with their relation to the **OR** gate as follows:



Cont..

- In Boolean algebra, a sum term is a sum of literals. In logic circuits, a sum term is produces by an **OR** operation with no **AND** operation involved. Some examples of sum terms are **A + B**, **A + B’**, **A + B + C’**.
- A sum term is equal to **1** when one or more of the literals in the term are **1**. A sum term is equal to **0** only if each of the literals is **0**.

Cont..

- Examples:

Determine the values of A, B, C, and D that make the sum term $A + \overline{B} + C + \overline{D}$ equal to 0.

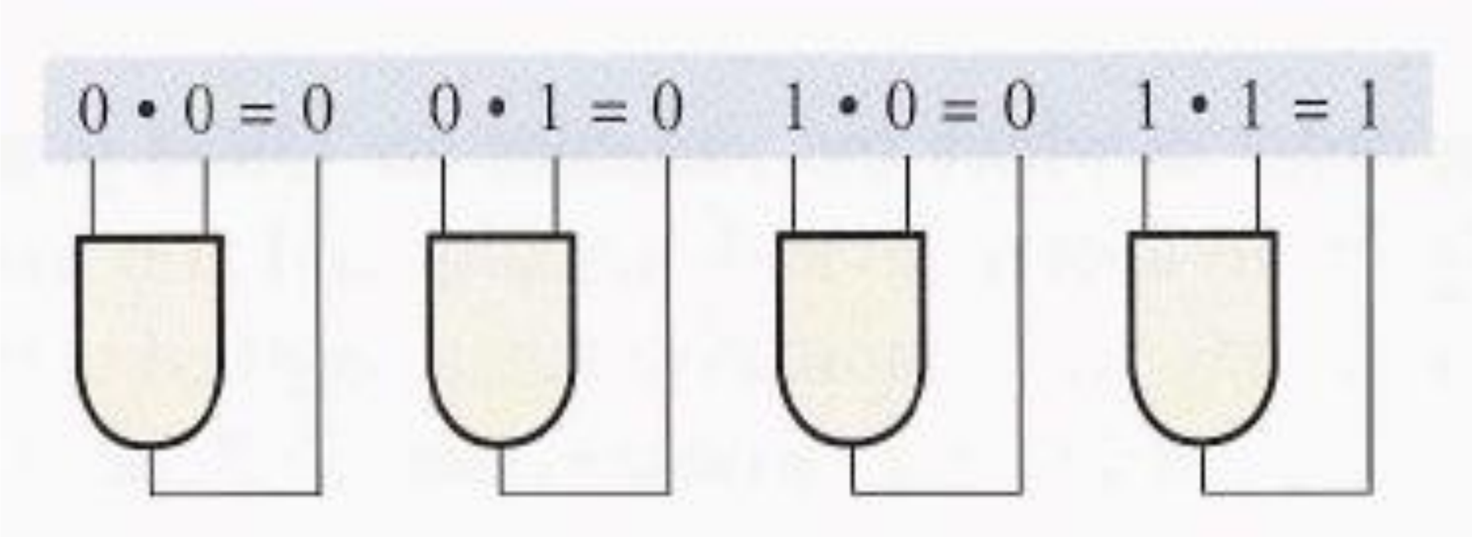
Solution For the sum term to be 0, each of the literals in the term must be 0. Therefore, $A = \mathbf{0}$, $B = \mathbf{1}$ so that $\overline{B} = 0$, $C = \mathbf{0}$, and $D = \mathbf{1}$ so that $\overline{D} = 0$.

$$A + \overline{B} + C + \overline{D} = 0 + \overline{1} + 0 + \overline{1} = 0 + 0 + 0 + 0 = 0$$

*Related Problem** Determine the values of A and B that make the sum term $\overline{A} + B$ equal to 0.

Boolean Multiplication

- Also recall from Chapter 3 that Boolean Multiplication is equivalent to the AND operation and the basic rules are illustrated with their relation to the AND gate as follows:



Cont..

- In Boolean algebra, a product term is the product of literals. In logic circuits, a product term is produced by an AND operation with no OR operations involved. Some examples of product terms are AB, AB’, ABC, and AB’CD’.
- A product term is equal to 1 only if each of the literals in the term is 1. A product term is equal to 0 when one or more of the literals are 0.

Cont..

- Examples

Determine the values of A, B, C, and D that make the product term $A\overline{B}C\overline{D}$ equal to 1.

Solution For the product term to be 1, each of the literals in the term must be 1. Therefore, $A = 1$, $B = 0$ so that $\overline{B} = 1$, $C = 1$, and $D = 0$ so that $\overline{D} = 1$.

$$A\overline{B}C\overline{D} = 1 \cdot \overline{0} \cdot 1 \cdot \overline{0} = 1 \cdot 1 \cdot 1 \cdot 1 = 1$$

Related Problem Determine the values of A and B that make the product term $\overline{A}\,\overline{B}$ equal to 1.

Rules of Boolean Algebra

- As the below table lists, there are 12 basic rules that are useful in manipulating and simplifying Boolean expressions. Rules 1 through 9 will be viewed in terms of their application to logic gated. Rules 10 through 12 will be derived in terms of the simpler rules and the laws of the Boolean algebra.

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

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

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.

Cont..

- Equation 1.1

Commutative Laws The *commutative law of addition* for two variables is written as

$$A + B = B + A$$

- Equation 1.2

The *commutative law of multiplication* for two variables is

$$AB = BA$$

Cont..

- Equation 2.1

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

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

- Equation 2.2

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

$$A(BC) = (AB)C$$

- Equation 3.1

Distributive Law The distributive law is written for three variables as follows:

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

Thank You