Boolean Logic and Boolean Algebra

CSE 4205: Digital Logic Design

Aashnan Rahman
Junior Lecturer
Department of Computer Science and Engineering (CSE)
Islamic University of Technology

01

Binary Logic

Zero or One?

Algebra

A mathematical system, defined with a set of elements, operators and a number of axioms or postulates.

- **A set of elements :** Objects having common properties
- **Operators :** A set of rules defined on the elements.
- **Postulates :** Basic assumptions from where rules, theorems, etc. are deduced.

Binary Logic

Binary Algebra consists of **binary variables** and **set of logical operation**

Binary variables take on only one of two possible values –

- High or 1
- Low or 0

Logical Operations?

AND, OR, NOT, NAND, NOR, X–OR, X–NOR

02

Postulates

Fundamental principle or statement assumed to be true

Postulates

A postulate (also called an **axiom**) is a **basic, foundational assumption** that is **accepted** as **true without proof** and serves as a **starting point** for **further reasoning** or **theory-building.**

Postulates

1. Closure

A set is said to satisfy closure property if it is closed under an operation or collection of operations.

In other words, A set is closed with respect to binary operation if performance of that operation on the members of the set always produce a result from the same set.

Postulates

2. Associative Law

$(x \# y) \# z = x \# (y \# z)$ for all, $x,y,z \in S$

3. Commutative Law

$x \# y = y \# x$ for all, $x,y \in S$

4. Distributive Law

$x \# (y \$ z) = (x \# y) \$ (x \# z)$ for all, $x,y,z \in S$

Postulates

5. Identity

$x \# e = e \# x = x$ for all, $x,e \in S$

6. Inverse

$x \# y = e$ for all, $x,y,e \in S$

History

- **In 1854, George Boole** introduced **systematic treatment of logic** and developed **Boolean algebra.**

- **In 1938, C. E. Shannon** materialized **a two-level Boolean algebra.** Also called **switching circuit/algebra.**

- For formal definition of Boolean algebra, we follow the **postulates** formulated by **E. V. Huntington (1904).**

  Boolean algebra is a field (implementation of algebraic structure) with a set of elements B, together with two binary operators (+) and (.) and follows the **Huntington postulates**.

Huntington Postulates

1. a. **Closure** with respect to binary operator +

   b. **Closure** with respect to binary operator .

2. a. An **identity element** with respect to +, is designated by **0** :

   $$x + 0 = 0 + x = x$$

   b. An **identity element** with respect to ., is designated by **1** :

   $$x \cdot 1 = 1 \cdot x = x$$

3. a. **Commutative** with respect to + : $x + y = y + x$

   b. **Commutative** with respect to . : $x \cdot y = y \cdot x$

4. a. . is **Distributed over** + : $x \cdot (y + z) = (x \cdot y) + (x \cdot z)$

   b. + is **Distributed over** . : $x + (y \cdot z) = (x + y) \cdot (x + z)$

Huntington Postulates

5. For every element x ∈ B there exists an inverse element x’ ∈ B (**complement of B**) such that :

$$x + x' = 1$$
$$x \cdot x' = 0$$

6. There exists at least **two elements** x,y ∈ B such that: $x \neq y$

Boolean Algebra vs Ordinary Algebra

- **Huntington postulates do not mention associative law, but it holds here.**
- **Distributive law of (+) over (-)** is **only valid** for **boolean algebra but not ordinary algebra.**

$$x + (y \cdot z) = (x + y) \cdot (x + z)$$

- **No existence of additive or multiplicative inverse in boolean algebra.**
- **No existence of division or multiplication in boolean algebra either.**
- **Complement is only available in Boolean algebra.**
- **For boolean algebra,** x ∈ B where B = [0,1]
- **For ordinary algebra,** x ∈ **R**

Postulates

Laws	Examples	
Identity Laws	$A+0=A$	$A\cdot 1=A$
	$A+1=1$	$A\cdot 0=0$
Idempotent Law	$A+A=A$	$A.A=A$
Complement Law	$A+\overline{A}=1$	$A+\overline{A}=1$
Commutative Law	$A+B=B+A$	$A\cdot B=B\cdot A$
Associative Law	$A+(B+C)=(A+B)+C$	$A\cdot(B\cdot C)=(A\cdot B)\cdot C$
Distributive Law	$A\cdot(B+C)=(A\cdot B)+(A\cdot C)$	$A+B\cdot C=(A+B)\cdot(A+C)$

Postulates

Laws	Examples	
Double Negation	$\bar{\bar{A}} = A$	
Absorption Law	$A + AB = A$	$A \cdot (A + B) = A$
De Morgan's Law	$\overline{A + B} = \overline{A} \cdot \overline{B}$	$\overline{A \cdot B} = \overline{A} + \overline{B}$

Duality

Every algebraic expression deducible from the postulates of the Boolean algebra remains **valid** if the operators and identity elements are interchanged.

$$x + x' = 1 \rightarrow x \cdot x' = 0$$

Self-Dual

If the dual expression = the original expression.

Find the dual of the following expressions, and determine if its there is a self dual.

1. $F = (A + C) \cdot B + 0$ **Ans :** $A \cdot C + B$
2. $G = X \cdot Y + (W + Z)$ **Ans :** $(X+Y) \cdot (W \cdot Z)$
3. $H = A \cdot B + A \cdot C + B \cdot C$ **Ans :** $(A + B)(A + C)(B + C)$

Proof of Basic Theorems

THEOREM 1(a): $x + x = x$.

Statement	Justification
$x + x = (x + x) \cdot 1$	postulate 2(b)
$= (x + x)(x + x')$	5(a)
$= x + xx'$	4(b)
$= x + 0$	5(b)
$= x$	2(a)

Proof of Basic Theorems

THEOREM 1(b): $x \cdot x = x$.

Statement	Justification
$x \cdot x = xx + 0$	postulate 2(a)
$= xx + xx'$	5(b)
$= x(x + x')$	4(a)
$= x \cdot 1$	5(a)
$= x$	2(b)

Proof of Basic Theorems

THEOREM 2(a): $x + 1 = 1$.

Statement	Justification
$x + 1 = 1 \cdot (x + 1)$	postulate 2(b)
$= (x + x')(x + 1)$	5(a)
$= x + x' \cdot 1$	4(b)
$= x + x'$	2(b)
$= 1$	5(a)

THEOREM 2(b): $x \cdot 0 = 0$ by duality.

Proof of Basic Theorems

THEOREM 6(a): $x + xy = x$.

Statement	Justification
$x + xy = x \cdot 1 + xy$	postulate 2(b)
$= x(1 + y)$	4(a)
$= x(y + 1)$	3(a)
$= x \cdot 1$	2(a)
$= x$	2(b)

THEOREM 6(b): $x(x + y) = x$ by duality.

De-Morgan's Law

The complement of a sum is the product of the complements.

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

The complement of a product is the sum of the complements.

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

03

Truth Table

Time for the truth (table)

Truth Tables

A truth table shows all possible input combinations of Boolean variables and their corresponding output for a given Boolean expression.

For **n inputs**, there will be **2^n possible input** and **output combinations**.

Proof using Truth Tables

Absorption Rule : $A + AB = A$

A	B	AB	A+AB
0	0	0	0
0	1	0	0
1	0	0	1
1	1	1	1

Proof using Truth Tables

De-Morgan's Law : $\overline{A+B} = \overline{A} \cdot \overline{B}$

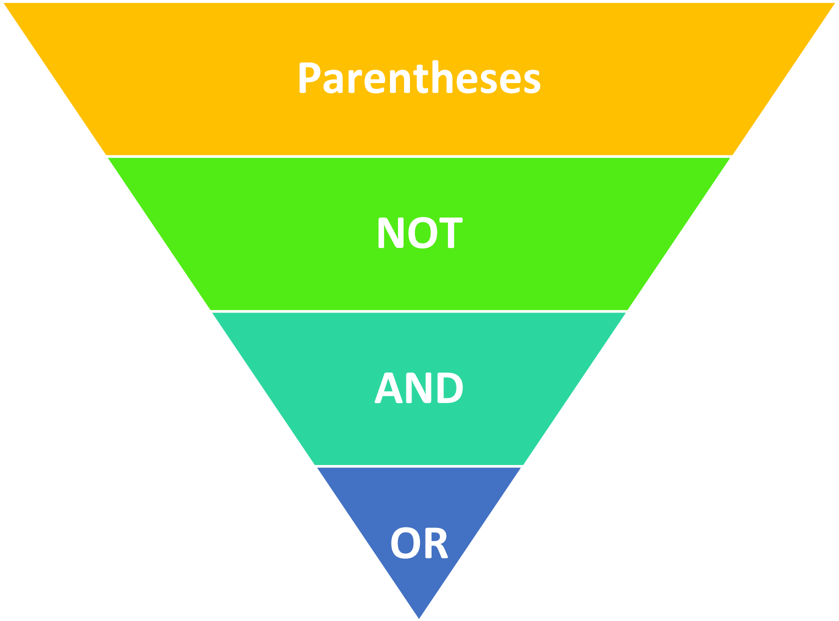
A	B	A+B	(A+B)'	A'	B'	A'B'
0	0	0	1	1	1	1
0	1	1	0	1	0	0
1	0	1	0	0	1	0
1	1	1	0	0	0	0

04

Boolean Algebra

The Logic Behind Every Digital Decision

Operator Precedence



Boolean Functions

A Boolean function is a mathematical expression that takes binary inputs (0s and 1s) and produces a binary output (0 or 1) using Boolean operations.

$$L(D, X, A) = D\overline{X} + A$$

Boolean Functions

Truth Table for the Function $L = D\overline{X} + A$

D	X	A	L
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

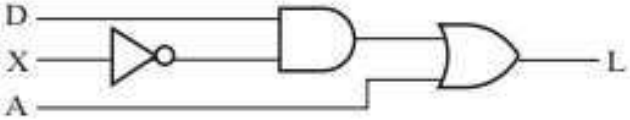


FIGURE 2-5
Logic Circuit Diagram for $L = D\overline{X} + A$

05

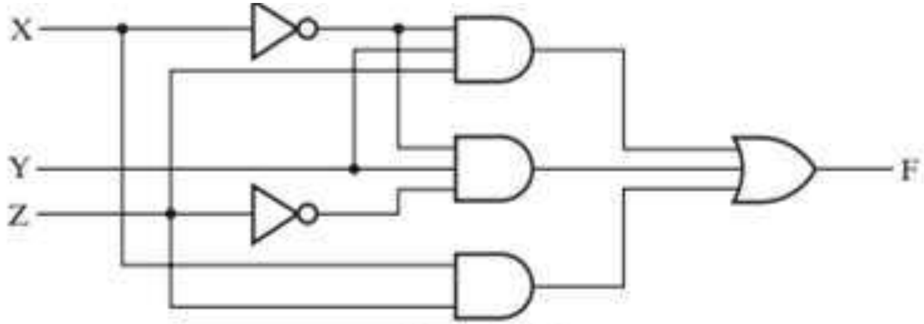
Circuit Diagram and Simplification

Let’s keep things simple

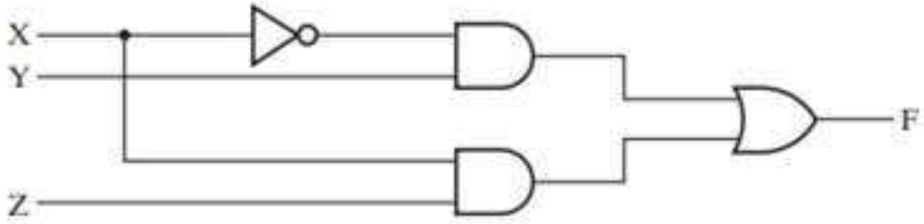
Boolean Functions

$$F = \overline{X}YZ + \overline{X}Y\overline{Z} + XZ$$

$$F = \overline{X}Y + XZ$$



(a) $F = \overline{X}YZ + \overline{X}Y\overline{Z} + XZ$



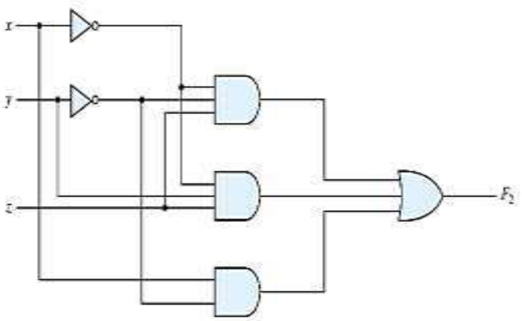
(b) $F = \overline{X}Y + XZ$

Boolean Functions

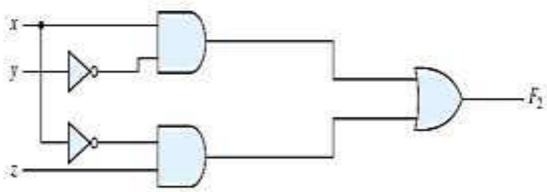
$$F_2 = x'y'z + x'yz + xy'$$

$$F_2 = x'y'z + x'yz + xy' = x'z(y' + y) + xy' = x'z + xy'$$

x	y	z	F_2
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0



(a) $F_2 = x'y'z + x'yz + xy'$



(b) $F_2 = xy' + x'z$

Boolean Functions

For practice

1. X + XY = X
2. XY + XY = X
3. X + XY = X+Y
4. X(X+Y)=X
5. (X+Y)(X+Y)= X
6. X (X+Y)=XY
7. $XY + \overline{X}Z + YZ = XY + \overline{X}Z$
8. $(X + Y)(\overline{X} + Z)(Y + Z) = (X + Y)(\overline{X} + Z)$

$$x(x' + y) = xx' + xy = 0 + xy = xy.$$
$$x + x'y = (x + x')(x + y) = 1(x + y) = x + y.$$
$$(x + y)(x + y') = x + xy + xy' + yy' = x(1 + y + y') = x.$$

Worksheet 1: https://web.mit.edu/6.111/www/s2007/PSETS/pset1.pdf
Worksheet 2: https://uomustansiriyah.edu.iq/media/lectures/6/6_2024_04_04!02_36_02_PM.pdf

Algebraic Manipulation

- In any Boolean expression,

  – A **Term** is represented by a **gate with inputs**.

  – Each **literal/variable** within a term designate an **input** in **primed** or **unprimed form.**

For example

$$F_1 = \overline{x}\overline{y}z + \overline{x}yz + x\overline{y}$$ 3 terms, 8 literals

$$F_2 = \overline{x}z + x\overline{y}$$ 2 terms, 4 literals

Complement

Interchanging the output of function F produces F’.

Methods

1. **Apply De-Morgan’s Law**
2. **Complement each literal and take its dual**

$$F_1 = \overline{X}Y\overline{Z} + \overline{X}\,\overline{Y}Z \qquad \overline{F_1} = (X + \overline{Y} + Z)(X + Y + \overline{Z})$$

$$F_2 = X(\overline{Y}\,\overline{Z} + YZ) \qquad \overline{F_2} = \overline{X} + (Y + Z)(\overline{Y} + \overline{Z})$$

Thank You !!

Feel free to ask any questions