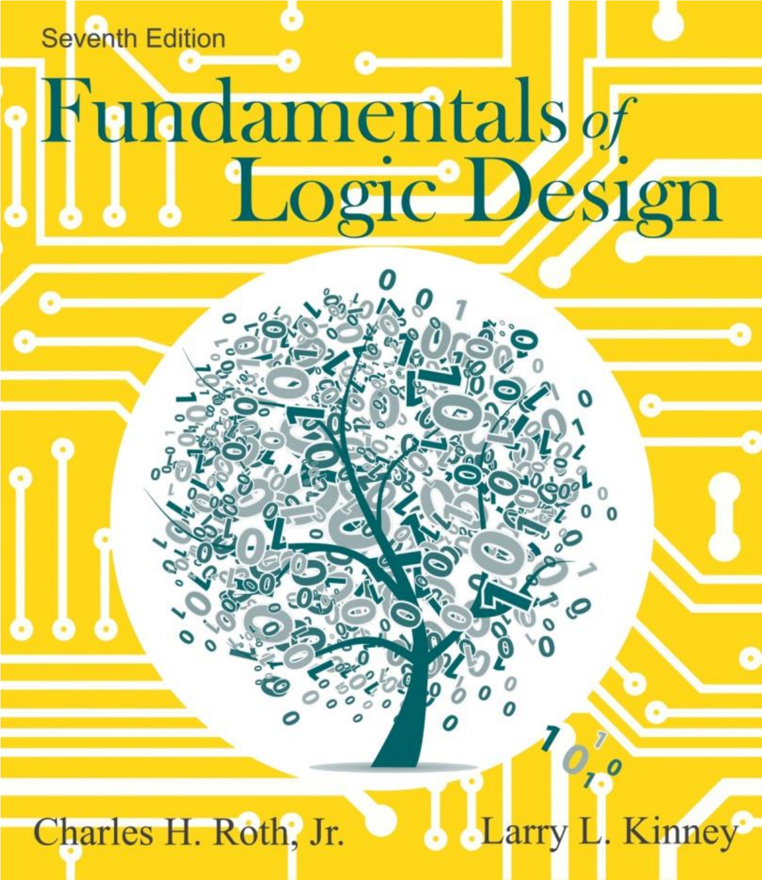


UNIT 2

Boolean Algebra

This chapter includes:

Learning Objectives

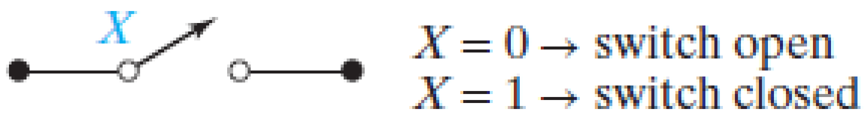
- Understand the basic operations and laws of Boolean algebra.
- Relate these operations and laws to circuits composed of AND gates, OR gates, INVERTERS and switches.
- Prove any of these laws in switching algebra using a truth table.
- Apply these laws to the manipulation of algebraic expressions including: obtaining a sum of products or product of sums, simplifying an expression and/or finding the complement of an expression

Introduction

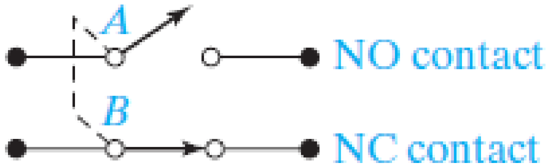
- All switching devices we will use are two-state devices, so we will emphasize the case in which all variables assume only one of two values.
- Boolean variable X or Y will be used to represent input or output of switching circuit.
- Symbols “0” and “1” represent the two values any variable can take on. These represent states in a logic circuit, and do not have numeric value.
- Logic gate: 0 usually represents range of low voltages and 1 represents range of high voltages
- Switch circuit: 0 represents open switch and 1 represents closed
- 0 and 1 can be used to represent the two states in any binary valued system.

Basic Operations

- The basic operations of Boolean (switching) algebra are called AND, OR, and complement (or inverse).
- To apply switching algebra to a switch circuit, each switch contact is labeled with a variable. See diagram below:



- NC (normally closed) and NO (normally open) contacts are always in opposite states.

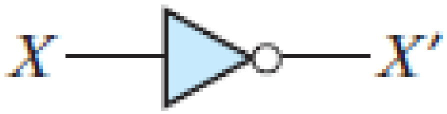


- If variable X is assigned to NO contact, then X′ will be assigned for NC.

Basic Operations

Complementation/ Inversion:

- Prime (’) denotes complementation.
- 0’=1 and 1’=0
- For a switching variable, X:
    - X’=1 if X=0 and X’=0 if X=1’
- Complementation is also called inversion. An inverter is represented as shown below, where circle at the output denotes inversion:



Basic Operations

Series Switching Circuits/ AND Operation:

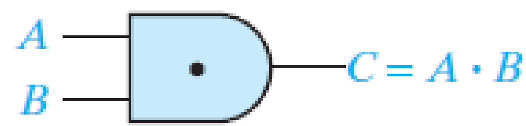
Series:
A) Truth table B) Logic gate diagram
C) Switch circuit diagram

The operation defined by the table is called AND. It is written algebraically as *C=A·B. We will usually write AB instead of A·B. The AND operation is also referred to as logical (or Boolean) multiplication.*

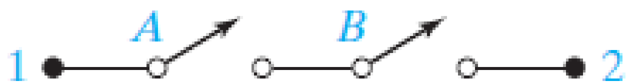
A)

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

B)



C)



$C = 0 \rightarrow$ open circuit between terminals 1 and 2
$C = 1 \rightarrow$ closed circuit between terminals 1 and 2

Basic Operations

Parallel Switching Circuits/ OR Operation:

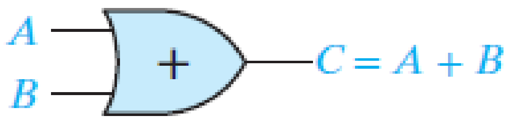
Series:
A) Truth table B) Logic gate diagram
C) Switch circuit diagram

The operation defined by the table is called OR. It is written algebraically as *C=A+B. The OR operation is also referred to as logical (or Boolean) addition.*

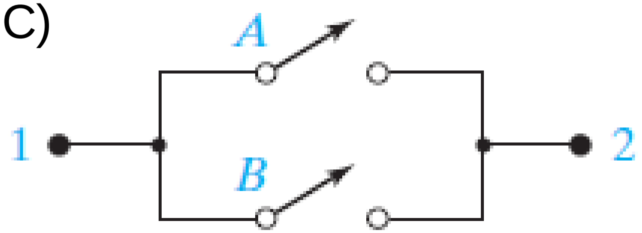
A)

A	B	$C = A + B$
0	0	0
0	1	1
1	0	1
1	1	1

B)



C)



If switches A and B are connected in parallel, there is a closed circuit if either A or B, or both, are closed and an open circuit only if A and B are both open.

Boolean Operations and Truth Tables

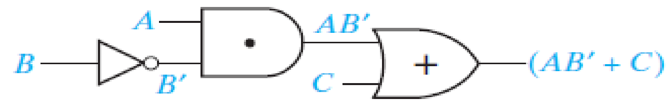
Examples of Boolean Expressions and Corresponding Diagrams:

Expressions

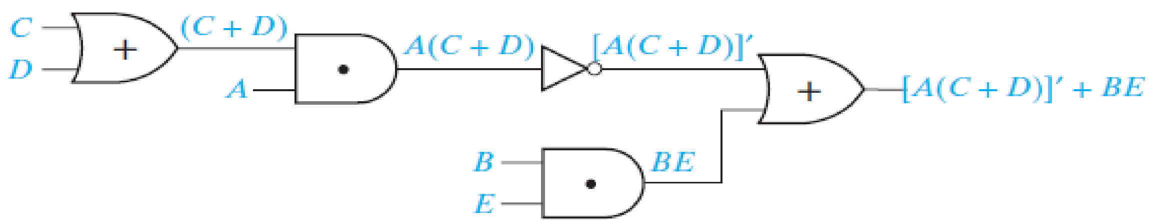
$$AB' + C \tag{2-1}$$
$$[A(C + D)]' + BE \tag{2-2}$$

Order of operations- Parentheses, Inversion, AND, OR

Logic Diagrams



(a)



(b)

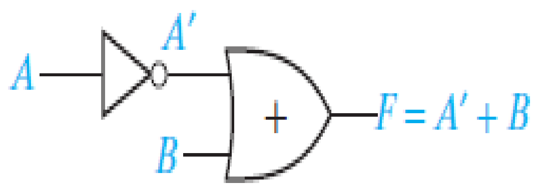
Boolean Operations and Truth Tables

Truth Tables:

A truth table specifies the values of a Boolean expression for every possible combination of values of the variables in the expression.

FIGURE 2-2
Two-Input Circuit and Truth Table

© Cengage Learning 2014



(a)

A	B	A'	$F = A' + B$
0	0	1	1
0	1	1	1
1	0	0	0
1	1	0	1

(b)

Boolean Operations and Truth Tables

Equal Boolean Expressions:

Two boolean expressions are said to be **equal** if they have the same value for every possible combination of the variables.

TABLE 2-1

© Cengage Learning 2014

A B C	B'	AB'	$AB' + C$	$A + C$	$B' + C$	$(A + C)(B' + C)$
0 0 0	1	0	0	0	1	0
0 0 1	1	0	1	1	1	1
0 1 0	0	0	0	0	0	0
0 1 1	0	0	1	1	1	1
1 0 0	1	1	1	1	1	1
1 0 1	1	1	1	1	1	1
1 1 0	0	0	0	1	0	0
1 1 1	0	0	1	1	1	1

$$AB' + C = (A + C)(B' + C) \tag{2-3}$$

An n-variable expression will have 2^n rows in its truth table.

Basic Theorems

Single Variable Basic Theorems:

Operations with 0 and 1:

$X + 0 = X$	(2-4)	$X \cdot 1 = X$	(2-4D)
$X + 1 = 1$	(2-5)	$X \cdot 0 = 0$	(2-5D)

Idempotent laws:

$X + X = X$	(2-6)	$X \cdot X = X$	(2-6D)

Involution law:

$(X')' = X$	(2-7)

Laws of complementarity:

$X + X' = 1$	(2-8)	$X \cdot X' = 0$	(2-8D)

See book for switch circuit diagrams that illustrate these basic theorems (Roth-Kinney 7th edition, pg 42-43).

Commutative, Associative, Distributive and DeMorgan's Laws

Commutative and Associative Laws:

Commutative: Order in which variables are written does not affect result of applying AND and OR operations.

$$XY = YX \text{ and } X+Y = Y+X$$

Associative: Result of AND and OR operations is independent of which variables we associate together first.

$$(XY)Z = X(YZ) = XYZ$$

$$(X+Y)+Z = X+(Y+Z) = X+Y+Z$$

Commutative, Associative, Distributive and DeMorgan's Laws

Distributive Law:

The distributive law of boolean algebra is as follows:

$$X(Y+Z)= XY+XZ$$

Furthermore, a second distributive law is valid for Boolean algebra but not ordinary algebra:

$$X+YZ= (X+Y)(X+Z)$$

Proof of this second distributive law can be found on page 45.

Commutative, Associative, Distributive and DeMorgan's Laws

DeMorgan's Laws:

DeMorgan's Law is stated as follows:

$$(X+Y)'=X'Y'$$

$$(XY)'=X'+Y'$$

Truth table proof of DeMorgan's Laws is shown below:

X	Y	X'	Y'	$X+Y$	$(X+Y)'$	$X'Y'$	XY	$(XY)'$	$X'+Y'$
0	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	0	0	1	1
1	0	0	1	1	0	0	0	1	1
1	1	0	0	1	0	0	1	0	0

Commutative, Associative, Distributive and DeMorgan's Laws

Laws of Boolean Algebra(Table 2-3):

Operations with 0 and 1:

1. $X + 0 = X$ 1D. $X \cdot 1 = X$
2. $X + 1 = 1$ 2D. $X \cdot 0 = 0$

Idempotent laws:

3. $X + X = X$ 3D. $X \cdot X = X$

Involution law:

4. $(X')' = X$

Laws of complementarity:

5. $X + X' = 1$ 5D. $X \cdot X' = 0$

Commutative, Associative, Distributive and DeMorgan's Laws

Laws of Boolean Algebra- (continued):

Commutative laws:

6. $X + Y = Y + X$ 6D. $XY = YX$

Associative laws:

7. $(X + Y) + Z = X + (Y + Z) = X + Y + Z$ 7D. $(XY)Z = X(YZ) = XYZ$

Distributive laws:

8. $X(Y + Z) = XY + XZ$ 8D. $X + YZ = (X + Y)(X + Z)$

DeMorgan's laws:

9. $(X + Y)' = X'Y'$ 9D. $(XY)' = X' + Y'$

Simplification Theorems

Simplification theorems:

Theorems used to replace an expression with a simpler expression are called **simplification theorems**.

Uniting theorems:

1. $XY + XY' = X$ 1D. $(X + Y)(X + Y') = X$

Absorption theorems:

2. $X + XY = X$ 2D. $X(X + Y) = X$

Elimination theorems:

3. $X + X'Y = X + Y$ 3D. $X(X' + Y) = XY$

Duality:

4. $(X + Y + Z + \cdots)^D = XYZ\ldots$ 4D. $(XYZ\ldots)^D = X + Y + Z + \cdots$

Theorems for multiplying out and factoring:

5. $(X + Y)(X' + Z) = XZ + X'Y$ 5D. $XY + X'Z = (X + Z)(X' + Y)$

Consensus theorems:

6. $XY + YZ + X'Z = XY + X'Z$ 6D. $(X + Y)(Y + Z)(X' + Z) = (X + Y)(X' + Z)$

Simplification Theorems

Proof of Simplification Theorems:

❖Using switching algebra, the theorems on the previous slide can be proven using truth tables.

❖In general Boolean algebra, these theorems must be proven algebraically starting with basic theorems.

Proof of (2-15): $XY + XY' = X(Y + Y') = X(1) = X$

Proof of (2-16): $X + XY = X \cdot 1 + XY = X(1 + Y) = X \cdot 1 = X$

Proof of (2-17): $X + X'Y = (X + X')(X + Y) = 1(X + Y) = X + Y$

Proof of (2-18): $XY + X'Z + YZ = XY + X'Z + (1)YZ =$

$XY + X'Z + (X + X')YZ = XY + XYZ + X'Z + X'YZ =$

$XY + X'Z$ (using absorption twice)

❖Using duality property, (2-15D)-(2-18D) can be proven

Multiplying Out and Factoring

Sum of Products:

An expression is said to be in *sum-of-products* (SOP) form when all products are the products of single variables. This form is the end result when an expression is fully multiplied out.

For example:

$$AB' + CD'E + AC'E'$$

$$ABC' + DEFG + H$$

Multiplying Out and Factoring

Product of Sums:

An expression is in *product-of-sums* (POS) form when all sums are the sums of single variables. It is usually easy to recognize a product-of-sums expression since it consists of a product of sum terms.

For example:

$$(A + B')(C + D' + E)(A + C' + E')$$

$$(A + B)(C + D + E)F$$

Multiplying Out and Factoring

Examples:

Example 1

Factor $A + B'CD$. This is of the form $X + YZ$ where $X = A$, $Y = B'$, and $Z = CD$, so

$$A + B'CD = (X + Y)(X + Z) = (A + B')(A + CD)$$

$A + CD$ can be factored again using the second distributive law, so

$$A + B'CD = (A + B')(A + C)(A + D)$$

Example 2

Factor $AB' + C'D$.

$AB' + C'D = (AB' + C')(AB' + D)$ ← note how $X + YZ = (X + Y)(X + Z)$ was applied here

$= (A + C')(B' + C')(A + D)(B' + D)$ ← the second distributive law was applied again to each term

Complementing Boolean Expressions

Using DeMorgan's Laws to find Inverse Expressions:

- The complement or inverse of any Boolean expression can be found using DeMorgan's Laws.
- DeMorgan's Laws for n-variable expressions:

$$(X_1 + X_2 + X_3 + \cdots + X_n)' = X_1{}'X_2{}'X_3{}' \ldots X_n{}' \qquad (2\text{-}25)$$

$$(X_1X_2X_3 \ldots X_n)' = X_1{}' + X_2{}' + X_3{}' + \cdots + X_n{}' \qquad (2\text{-}26)$$

For example, for $n = 3$,

$$(X_1 + X_2 + X_3)' = (X_1 + X_2)'X'_3 = X'_1X'_2X'_3$$

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

Complementing Boolean Expressions

Examples:

Example 1

To find the complement of $(A' + B)C'$, first apply (2-13) and then (2-12).

$$[(A' + B)C']' = (A' + B)' + (C')' = AB' + C$$

Example 2

$$\begin{aligned}
[(AB' + C)D' + E]' &= [(AB' + C)D']'E' && \text{(by (2-12))} \\
&= [(AB' + C)' + D]E' && \text{(by (2-13))} \\
&= [(AB')'C' + D]E' && \text{(by (2-12))} \\
&= [(A' + B)C' + D]E' && \text{(by (2-13))} \quad (2\text{-}27)
\end{aligned}$$

Note that in the final expressions, the complement operation is applied only to single variables.