CSET105 – DIGITAL DESIGN

BOOLEAN ALGEBRA

Bennett University

MODULE-2

Boolean algebra

- Basic Definitions, Axiomatic Definition of Boolean Algebra, Basic Theorems and Properties of Boolean Algebra, Boolean Functions, Canonical and Standard Forms

BOOLEAN ALGEBRA

BOOLEAN ALGEBRA

BOOLEAN ALGEBRA INTRODUCTION

- Boolean Algebra is the mathematics we use to analyze digital gates and circuits.
- We can use these “Laws of Boolean” to both reduce and simplify a complex Boolean expression to reduce the number of logic gates required.
- Boolean Algebra is therefore a system of mathematics based on logic that has its own set of rules or laws which are used to define and reduce Boolean expressions.

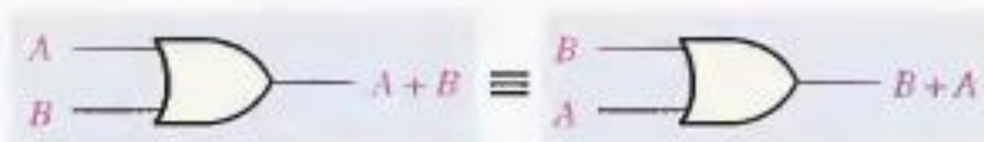
BOOLEAN ALGEBRA

LAWS OF BOOLEAN ALGEBRA

- **Commutative Laws:** The commutative law of addition for two variables is written as: $A+B=B+A$

FIGURE 4–1

Application of commutative law of addition.



- The commutative law of multiplication for two variables is, $AB = BA$

FIGURE 4–2

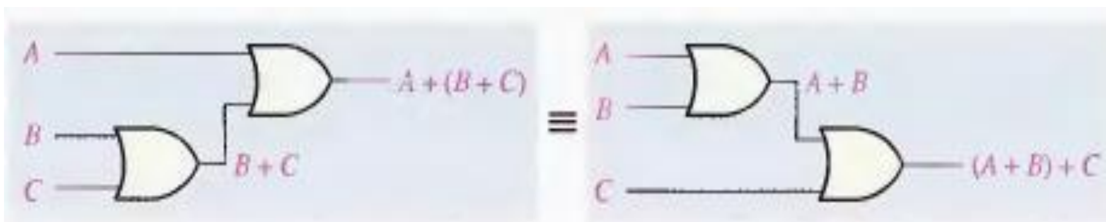
Application of commutative law of multiplication.

BOOLEAN ALGEBRA

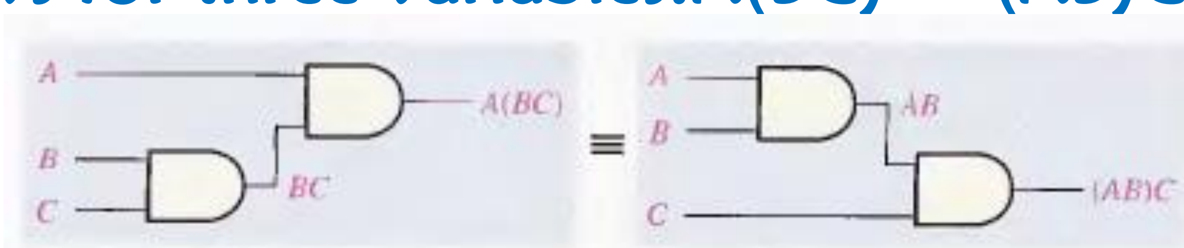
LAWS OF BOOLEAN ALGEBRA

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

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



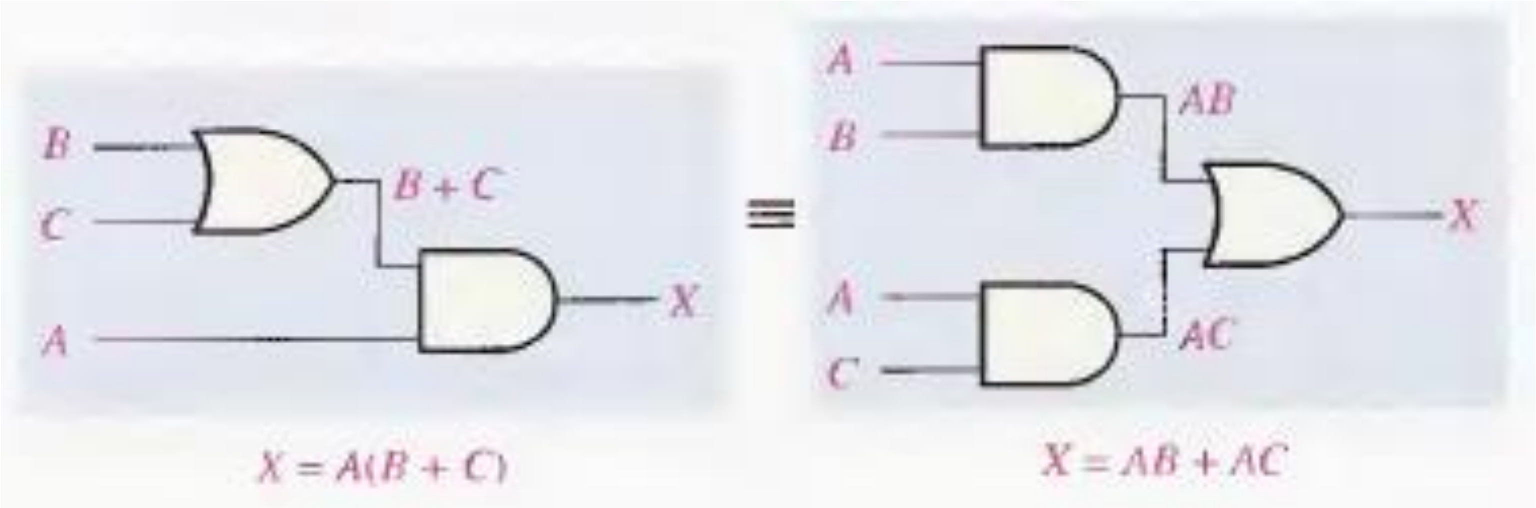
- The associative law of multiplication is written as follows for three variables: $A(BC) = (AB)C$



LAWS OF BOOLEAN ALGEBRA

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

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



BOOLEAN ALGEBRA

RULES OF BOOLEAN ALGEBRA

- Basic rules that are useful in manipulating and simplifying Boolean expressions.

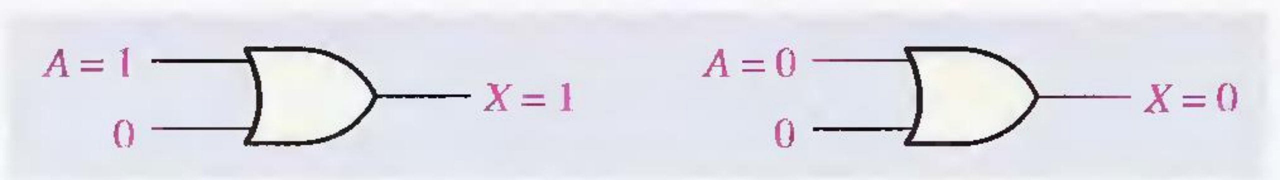
1. $A + 0 = A$	7. $A \cdot A = A$
2. $A + 1 = 1$	8. $A \cdot \overline{A} = 0$
3. $A \cdot 0 = 0$	9. $\overline{\overline{A}} = A$
4. $A \cdot 1 = A$	10. $A + AB = A$
5. $A + A = A$	11. $A + \overline{A}B = A + B$
6. $A + \overline{A} = 1$	12. $(A + B)(A + C) = A + BC$

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

BOOLEAN ALGEBRA

RULES OF BOOLEAN ALGEBRA

RULE - 1



$$X = A + 0 = A$$

RULE - 2

$$X = A + 1 = 1$$

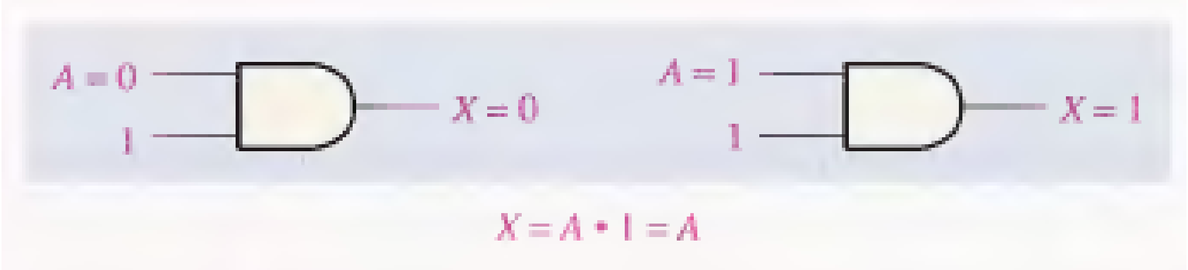
RULE - 3

$$X = A \cdot 0 = 0$$

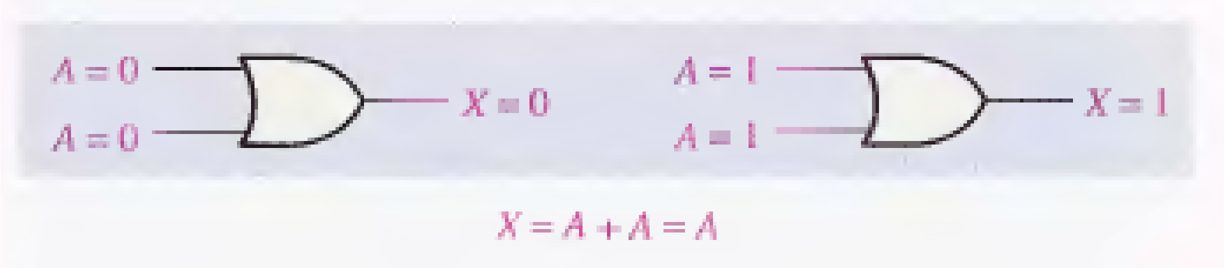
BOOLEAN ALGEBRA

RULES OF BOOLEAN ALGEBRA

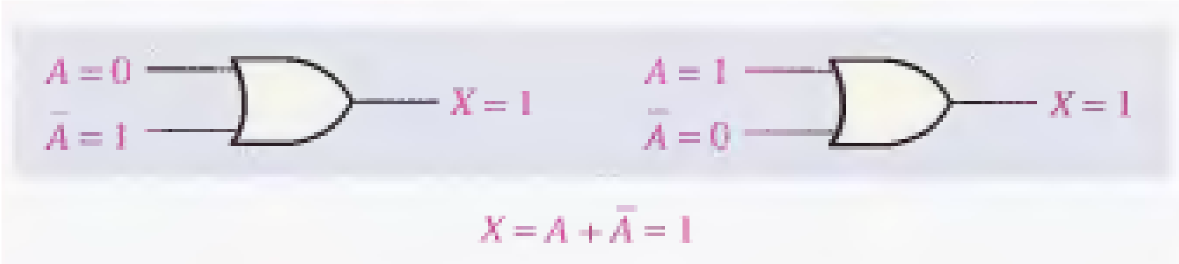
RULE - 4



RULE - 5



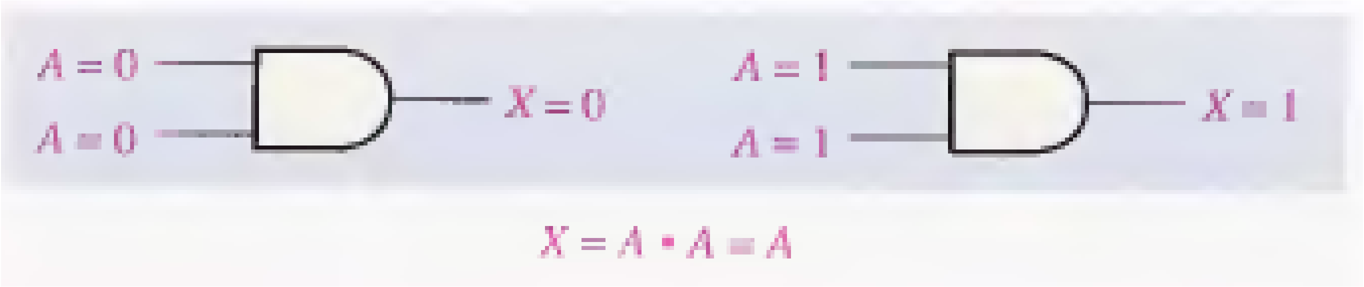
RULE - 6



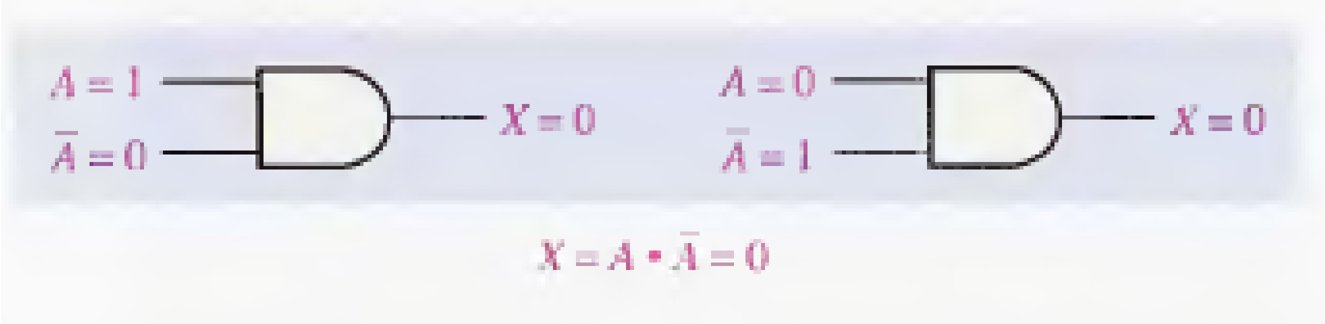
BOOLEAN ALGEBRA

RULES OF BOOLEAN ALGEBRA

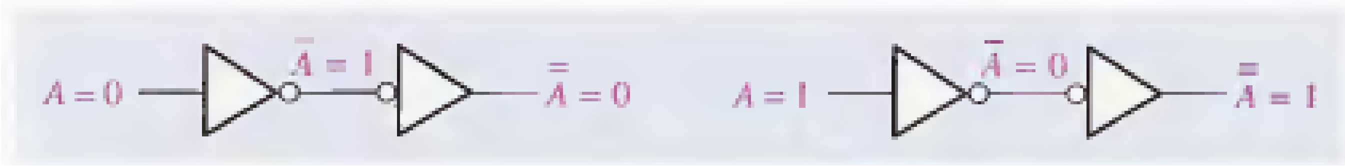
RULE - 7



RULE - 8



RULE - 9



BOOLEAN ALGEBRA

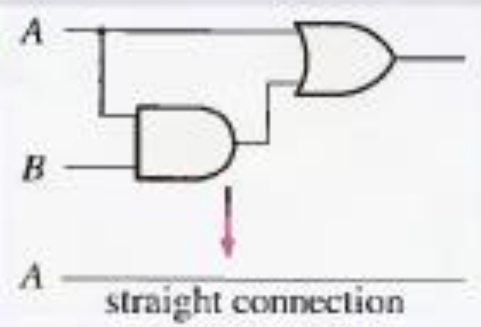
RULES OF BOOLEAN ALGEBRA

Rule 10. $A + AB = A$ This rule can be proved by applying the distributive law, rule 2, and rule 4 as follows:

$$A + AB = A(1 + B) \quad \text{Factoring (distributive law)}$$
$$= A \cdot 1 \quad \text{Rule 2: } (1 + B) = 1$$
$$= A \quad \text{Rule 4: } A \cdot 1 = A$$

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

equal



BOOLEAN ALGEBRA

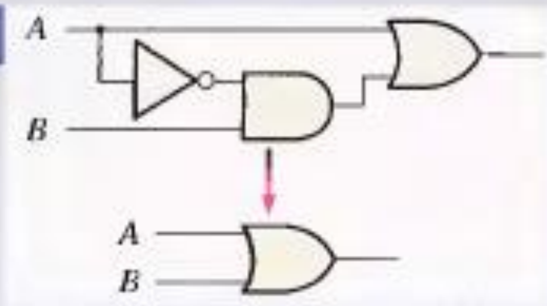
RULES OF BOOLEAN ALGEBRA

Rule 11. $A + \bar{A}B = A + B$ This rule can be proved as follows:

$A + \bar{A}B = (A + AB) + \bar{A}B$	Rule 10: $A = A + AB$
$= (AA + AB) + \bar{A}B$	Rule 7: $A = AA$
$= AA + AB + A\bar{A} + \bar{A}B$	Rule 8: adding $A\bar{A} = 0$
$= (A + \bar{A})(A + B)$	Factoring
$= 1 \cdot (A + B)$	Rule 6: $A + \bar{A} = 1$
$= A + B$	Rule 4: drop the 1

A	B	$\bar{A}B$	$A + \bar{A}B$	$A + B$
0	0	0	0	0
0	1	1	1	1
1	0	0	1	1
1	1	0	1	1

↑ equal ↑ ($A + \bar{A}B$ and $A + B$ columns)



BOOLEAN ALGEBRA

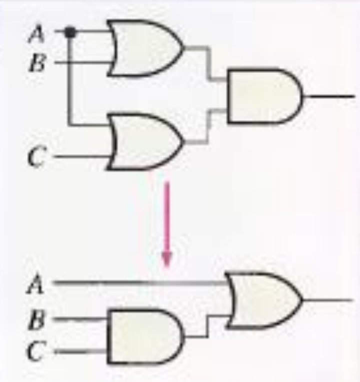
RULES OF BOOLEAN ALGEBRA

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: $A \cdot 1 = A$

A	B	C	A + B	A + C	(A + B)(A + C)	BC	A + BC
0	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0
0	1	0	1	0	0	0	0
0	1	1	1	1	1	1	1
1	0	0	1	1	1	0	1
1	0	1	1	1	1	0	1
1	1	0	1	1	1	0	1
1	1	1	1	1	1	1	1

equal

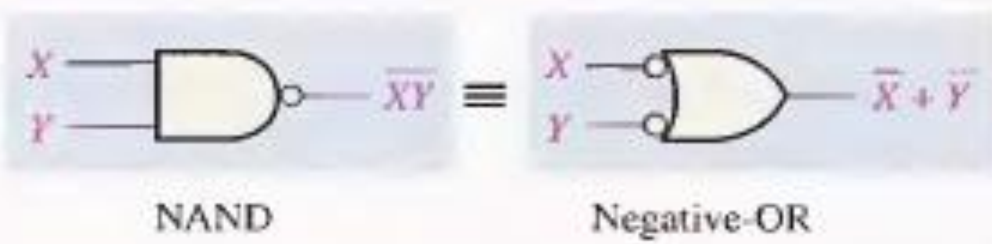


BOOLEAN ALGEBRA

DEMORGAN'S LAW – FIRST LAW

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

- The formula for expressing this theorem for two variables is $\overline{XY} = \overline{X} + \overline{Y}$



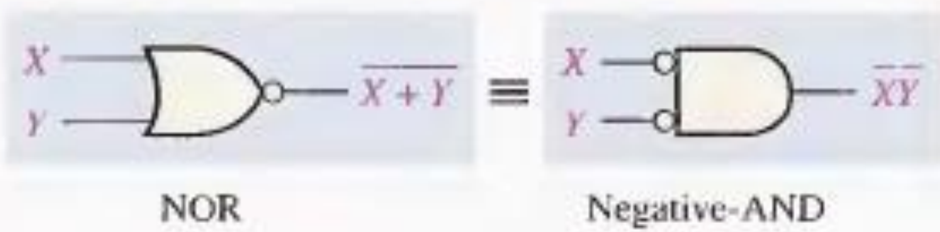
Inputs		Output	
X	Y	$\overline{XY}$	$\overline{X} + \overline{Y}$
0	0	1	1
0	1	1	1
1	0	1	1
1	1	0	0

BOOLEAN ALGEBRA

DEMORGAN'S LAW – SECOND LAW

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

- The formula for expressing this theorem for two variables is $\overline{X+Y} = \overline{X}\,\overline{Y}$



Inputs		Output	
X	Y	$\overline{X+Y}$	$\overline{X}\,\overline{Y}$
0	0	1	1
0	1	0	0
1	0	0	0
1	1	0	0

BOOLEAN ALGEBRA

DEMORGAN'S LAW – EXAMPLES

Apply DeMorgan's theorems to each of the following expressions:

(a) $\overline{(A + B + C)D}$ (b) $\overline{ABC + DEF}$ (c) $\overline{A\overline{B} + \overline{C}D + EF}$

Solution (a) Let $A + B + C = X$ and $D = Y$. The expression $\overline{(A + B + C)D}$ is of the form $\overline{XY} = \overline{X} + \overline{Y}$ and can be rewritten as

$$\overline{(A + B + C)D} = \overline{A + B + C} + \overline{D}$$

Next, apply DeMorgan's theorem to the term $\overline{A + B + C}$.

$$\overline{A + B + C} + \overline{D} = \overline{A}\,\overline{B}\,\overline{C} + \overline{D}$$

(b) Let $ABC = X$ and $DEF = Y$. The expression $\overline{ABC + DEF}$ is of the form $\overline{X + Y} = \overline{X}\,\overline{Y}$ and can be rewritten as

$$\overline{ABC + DEF} = (\overline{ABC})(\overline{DEF})$$

Next, apply DeMorgan's theorem to each of the terms $\overline{ABC}$ and $\overline{DEF}$.

$$(\overline{ABC})(\overline{DEF}) = (\overline{A} + \overline{B} + \overline{C})(\overline{D} + \overline{E} + \overline{F})$$

1. Apply DeMorgan's theorems to the following expressions:

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

BOOLEAN ALGEBRA

MINIMIZATION OF BOOLEAN EXPRESSION

$(x' + y)(x + y)$

$= x'.x + x'y + yx + y.y$

$= 0 + x'y + xy + y$ $[x.x' = 0]; [y.y = y]$

$= y(x' + x + 1)$

$= y(1)$ $[1 + x = 1]$

$= y.$

BOOLEAN ALGEBRA

MINIMIZATION OF BOOLEAN EXPRESSION

xy+ xyz+ xyz'+ x'yz

= xy (1+ z+ z')+ x'yz

= xy (1) + x'yz [1+ x = 1]

= xy+ x'yz

= y (x+ x'z) [x+ x'y = x+ y]

= y (x+ z).

BOOLEAN ALGEBRA

MINIMIZATION OF BOOLEAN EXPRESSION

$x'yz + xy'z' + x'y'z' + xy'z + xyz$

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

$= yz(1) + y'z'(x + x') + xy'z$ **[$x + x' = 1$]**

$= yz + y'z'(1) + xy'z$ **[$x + x' = 1$]**

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

$= yz + y'(z' + xz)$

$= yz + y'(z' + x)$ **[$x' + xy = x' + y$]**

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

BOOLEAN ALGEBRA

MINIMIZATION OF BOOLEAN EXPRESSION

$xy + xy'(x'z')'$

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

$= xy + xy'(x + z)$ $[x'' = x]$

$= xy + xy'x + xy'z$

$= xy + xy' + xy'z$ $[x.x = x]$

$= xy + xy'[1 + z]$

$= xy + xy'[1]$ $[1 + x = 1]$

$= xy + xy'$

$= x(y + y')$

$= x[1]$ $[x + x' = 1]$

$= x$

BOOLEAN ALGEBRA

MINIMIZATION OF BOOLEAN EXPRESSION

$[(xy)' + x' + xy]'$

$= [x' + y' + x' + xy]'$	
$= [x' + y' + xy]'$	$[x + x = x]$
$= [x' + y' + x]'$	$[x' + xy = x' + y]$
$= [y' + 1]'$	$[x + x' = 1]$
$= [1]'$	$[1 + x = 1]$
$= 0.$	

BOOLEAN ALGEBRA

MINIMIZATION OF BOOLEAN EXPRESSION - EXERCISE

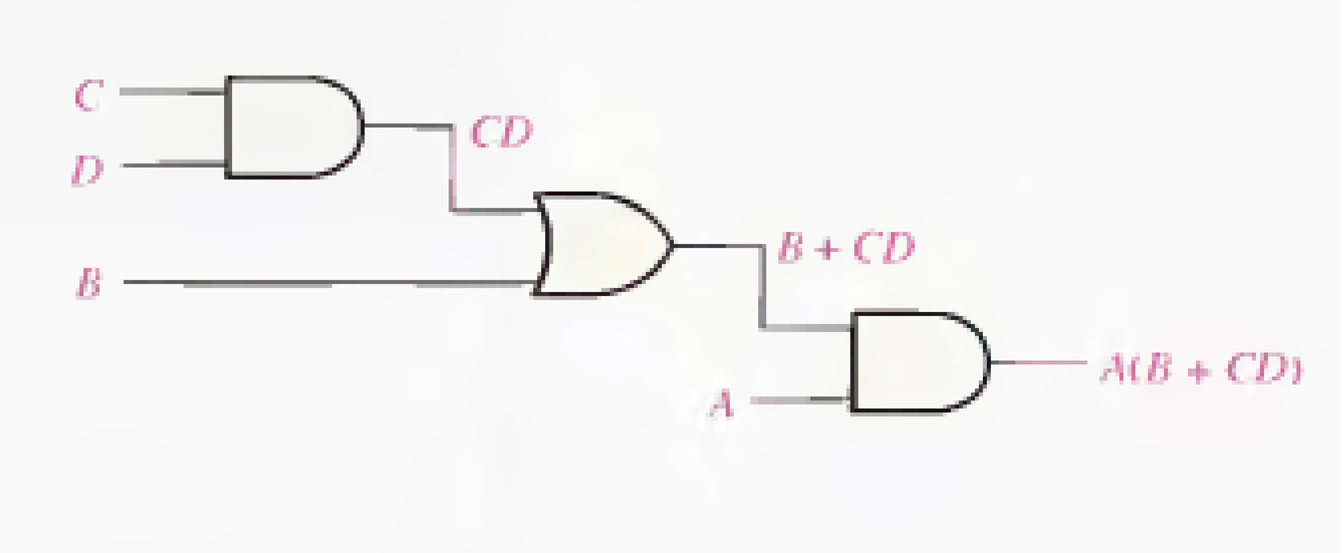
1. $xyz + xy'z + xyz' = x(y + z)$
2. $w'xyz' + xyz' + xy'z' + xy'z = xz$
3. $w'xy'z + w'xyz + wxz = xz$
4. $AB + (AC)' + AB'C\,(AB + C) = 1$
5. $x'y'z' + x'y'z + x'yz' + x'yz + xy'z' = x' + y'z'$
6. $(x + y)\,(x'z' + z)\,(y' + xz)' = x'y$

CANONICAL AND STANDARD FORMS

CANONICAL AND STANDARD FORMS

BOOLEAN EXPRESSION FOR A LOGIC CIRCUIT

- To derive the Boolean expression for a given logic circuit, begin at the left-most inputs and work toward the final output, writing the expression for each gate.



CANONICAL AND STANDARD FORMS

CONSTRUCTING A TRUTH TABLE FOR A LOGIC CIRCUIT

- Truth table shows the output for all possible values of the input variables can be developed.
- The procedure requires that you evaluate the Boolean expression for all possible combinations of values for the input variables.
- The first step is to list the sixteen input variable combinations of 1's and 0's in binary sequence.
- Next apply the inputs to the circuit and find the output, enter it in the output column of truth table.

CANONICAL AND STANDARD FORMS

CONSTRUCTING A TRUTH TABLE FOR A LOGIC CIRCUIT

INPUTS A	B	C	D	OUTPUT A(B + CD)
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

CANONICAL AND STANDARD FORMS

- **Standard form:** In a standard form we don't have to compulsorily write all the literals in all the terms of an expression.

  e.g. $f = xyz + y + x$
- Literal is a variable or its complement
- **Canonical form:** In a canonical form we have to compulsorily write all the literals in all the terms of an expression.

  e.g. $f = xyz+x'yz'+xy'z'$

CANONICAL AND STANDARD FORMS

- All Boolean expressions, regardless of their form, can be represented into either of two forms:
  - Sum-of-Products Form (SOP)
  - Product-of Sums Form (POS)

- Sum of Products(SOP): The logical sum of two or more logical product terms is referred to as a sum of products expression.

- It is basically an OR operation on AND operated variables. For example,

  Y = A + BC + AB’C or Y = AB’C’ + A’BC + AB’C

CANONICAL AND STANDARD FORMS

- **Product of Sums(POS):** Similarly, the logical product of two or more logical sum terms is called a product of sums expression.

- It is an AND operation on OR operated variables. For example, Y=(A) (B+C) (A+B'+C)

  or

  Y= (A+B'+C')(A'+ B+C) (A+B'+C)

- **Minterm:** A binary variable may appear either in its normal form (x) or in its complement form (x').

CANONICAL AND STANDARD FORMS

- When two binary variables x and y combined with an AND operation, there are four possible combinations:

  x'y', x'y, xy' and xy

  Each of these four AND terms is called as 'minterm'. In 'minterm' all the variables appears exactly once.
- **Maxterm:** In a similar fashion, when two binary variables x and y combined with an OR operation, there are four possible combinations:

  x+ y, x+ y', x'+ y and x'+ y'

  Each of these four OR terms is called as 'maxterm'.

CANONICAL AND STANDARD FORMS

The minterms and maxterms of a 3- variable function can be represented as in table below.

Variables			Minterms	Maxterms
X	Y	Z	m_i	M_i
0	0	0	$x'y'z' = m_0$	$x + y + z = M_0$
0	0	1	$x'y'z = m_1$	$x + y + z' = M_1$
0	1	0	$x'yz' = m_2$	$x + y' + z = M_2$
0	1	1	$x'yz = m_3$	$x + y' + z' = M_3$
1	0	0	$xy'z' = m_4$	$x' + y + z = M_4$
1	0	1	$xy'z = m_5$	$x' + y + z' = M_5$
1	1	0	$xyz' = m_6$	$x' + y' + z = M_6$
1	1	1	$xyz = m_7$	$x' + y' + z' = M_7$

CANONICAL AND STANDARD FORMS

DERIVING SOP AND POS FROM TRUTH TABLE

Inputs			*Output*	*Product terms*	*Sum terms*
A	*B*	*C*	*Y*		
0	0	0	0		$A + B + C$
0	0	1	0		$A + B + C'$
0	1	0	1	$A'BC'$	
0	1	1	0		$A + B' + C'$
1	0	0	1	$AB'C'$	
1	0	1	1	$AB'C$	
1	1	0	1	ABC'	
1	1	1	0		$A' + B' + C'$

$$Y = A'BC' + AB'C' + AB'C + ABC' \quad \text{(SOP)}$$

$$Y = (A + B + C)(A + B + C')(A + B' + C')(A' + B' + C') \quad \text{(POS)}$$

CANONICAL AND STANDARD FORMS

- Canonical Sum of Product Expression: If each term in SOP form contains all the literals then the SOP is known as Standard (or) Canonical SOP form.

- Example:

$$F(A, B, C) = AB'C + ABC + ABC'$$

- The same can be expressed in a compact form by listing the corresponding decimal-equivalent codes of the minterms containing a function value of 1.

CANONICAL AND STANDARD FORMS

- For example, if the canonical sum of product form of a three-variable logic function F has the minterms A'BC, AB'C, and ABC', this can be expressed as the sum of the decimal codes corresponding to these minterms as below.

$$F(A,B,C) = \Sigma(3,5,6)$$
$$= m3 + m5 + m6$$
$$= A'BC + AB'C + ABC'$$

where Σ(3,5,6) represents the summation of minterms corresponding to decimal codes 3, 5, and 6.

CANONICAL AND STANDARD FORMS

- Example-1: .Obtain the canonical sum of product form of the following function,   F (A, B, C) = A + BC

Solution:

- The given function contains three variables A, B, C.
- The variables B and C are missing from the first term of the expression and the variable A is missing from the second term of the expression.
- Therefore, first term is to be multiplied by (B+B') and (C+C').The second term is to be multiplied by (A+A').

CANONICAL AND STANDARD FORMS

$$\begin{aligned} F(A, B, C) &= A + BC \\ &= A(B + B')(C + C') + BC(A + A') \\ &= (AB + AB')(C + C') + ABC + A'BC \\ &= ABC + AB'C + ABC' + AB'C' + ABC + A'BC \\ &= ABC + AB'C + ABC' + AB'C' + A'BC \text{ (as } ABC + ABC = ABC) \end{aligned}$$

Hence the canonical sum of the product expression of the given function is

$$F(A, B) = ABC + AB'C + ABC' + AB'C' + A'BC.$$

Example-2: $\mathbf{Y(A, B, C) = AC + AB + BC}$

$$\begin{aligned} &= AC(B + B') + AB(C + C') + BC(A + A') \\ &= \underline{ABC} + AB'C + \underline{ABC} + ABC' + \underline{ABC} + A'BC \\ &= ABC + AB'C + ABC' + A'BC \\ &= \sum m(3, 5, 6, 7). \end{aligned}$$

CANONICAL AND STANDARD FORMS

- **Canonical Product of Sum Expression:**
  When a Boolean function is expressed as the logical product of all the maxterms from the rows of a truth table, for which the value of the function is 0, it is referred to as the canonical product of sum expression.

- The same can be expressed in a compact form by listing the corresponding decimal equivalent codes of the maxterms containing a function value of 0.

CANONICAL AND STANDARD FORMS

- For example, if the canonical product of sums form of a three-variable logic function F has the maxterms A + B + C, A + B′+ C, and A′+ B + C′, this can be expressed as the product of the decimal codes corresponding to these maxterms as below,

$$F(A,B,C) = \Pi(0,2,5)$$
$$= M_0 M_2 M_5$$
$$= (A + B + C)(A + B' + C)(A' + B + C')$$

Where ∏(0,2,5) represents the product of maxterms corresponding to decimal codes 0, 2, and 5.

CANONICAL AND STANDARD FORMS

- **Example-1:** Obtain the canonical product of the sum form of the following function, F(A,B,C)=(A+B') (B+C) (A+C')

Solution:

- In the above three-variable expression, C is missing from the first term, A is missing from the second term, and B is missing from the third term.

- Therefore, CC' is to be added with first term, AA' is to be added with the second, and BB' is to be added with the third term.

CANONICAL AND STANDARD FORMS

$F(A, B, C) = (A + B')(B + C)(A + C')$

$= (A + B' + 0)(B + C + 0)(A + C' + 0)$

$= (A + B' + CC')(B + C + AA')(A + C' + BB')$

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

[using the distributive property, as $X + YZ = (X + Y)(X + Z)$]

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

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

Hence the canonical product of the sum expression for the given function is

$F(A, B, C) = (A + B' + C)(A + B' + C')(A + B + C)(A' + B + C)(A + B + C')$

CANONICAL AND STANDARD FORMS

Example-2:

$$Y = (A+B)(B+C)(A+C)$$
$$= (A+B+C.C')(B+C+A.A')(A+C+B.B')$$
$$= \underline{(A+B+C)}(A+B+C')\underline{(A+B+C)}(A'+B+C)\underline{(A+B+C)}(A+B'+C)$$
$$= (A+B+C)(A+B+C')(A'+B+C)(A+B'+C)$$
$$= M_0 . M_1 . M_4 . M_2$$
$$= \prod M(0, 1, 2, 4)$$

CANONICAL AND STANDARD FORMS

Exercises

Obtain the canonical Sum of products form of the following function

(i). Y (A, B, C, D) = AB+ ACD
(ii). Y (A, B, C) = A+ ABC

Obtain the canonical product of sum form of the following function

(i). Y= A. (B+ C+ A)
(ii). Y= (A+B’) (B+C) (A+C’)

CANONICAL AND STANDARD FORMS

CONVERTING CANONICAL SOP TO CANONICAL POS

- To convert from canonical SOP to canonical POS, the following steps are taken:

  Step 1. Evaluate each product term in the SOP expression. That is, determine the binary numbers that represent the product terms.

  Step 2. Determine all of the binary numbers not included in the evaluation in Step 1.

  Step 3. Write the equivalent sum term for each binary number from Step 2 and express in POS form.

CANONICAL AND STANDARD FORMS

CONVERTING CANONICAL SOP TO CANONICAL POS

Convert the following SOP expression to an equivalent POS expression:

$$\bar{A}\bar{B}\bar{C} + \bar{A}B\bar{C} + \bar{A}BC + A\bar{B}C + ABC$$

Solution The evaluation is as follows:

$$000 + 010 + 011 + 101 + 111$$

Since there are three variables in the domain of this expression, there are a total of eight (2^3) possible combinations. The SOP expression contains five of these combinations, so the POS must contain the other three which are 001, 100, and 110.

Remember, these are the binary values that make the sum term 0. The equivalent POS expression is

$$(A + B + \bar{C})(\bar{A} + B + C)(\bar{A} + \bar{B} + C)$$

Related Problem Verify that the SOP and POS expressions in this example are equivalent by substituting binary values into each.

THANK YOU