Lecture (04)
Boolean Algebra and Logic Gates

By:
Dr. Ahmed ElShafee

Boolean algebra **properties**

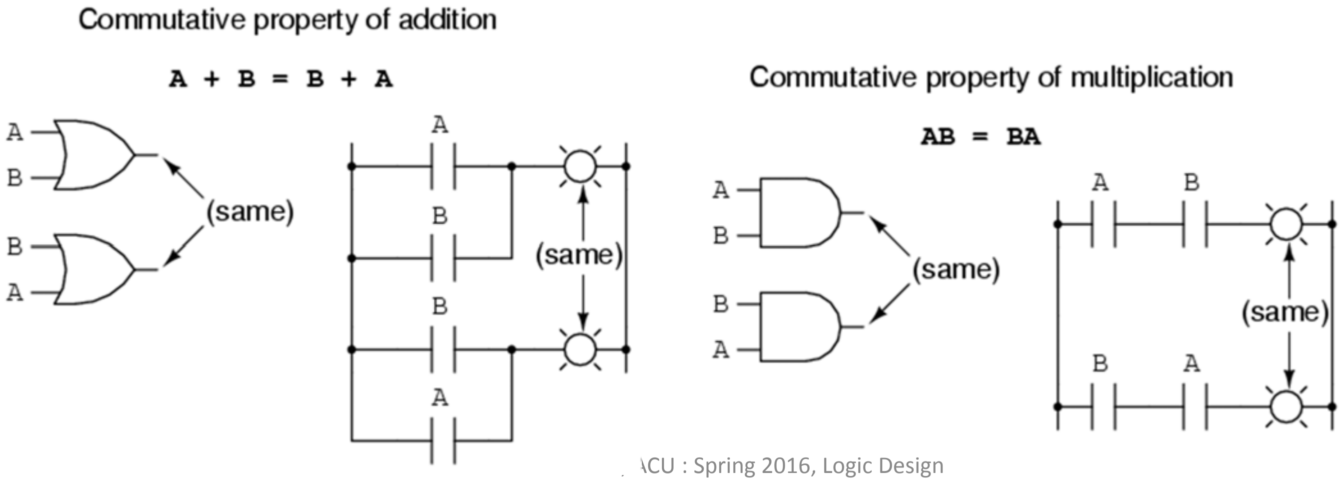
basic assumptions and properties:

- *Closure law*
  - A set *S* is closed with respect to a binary operator, for every pair of elements of *S*,
  - the binary operator specifies a rule for obtaining a unique element of *S*.

$a, b \in N$, there is a unique $c \in N$ such that $a + b = c$.

- *Commutative law.*
  - A binary operator * on a set *S* is said to be commutative whenever

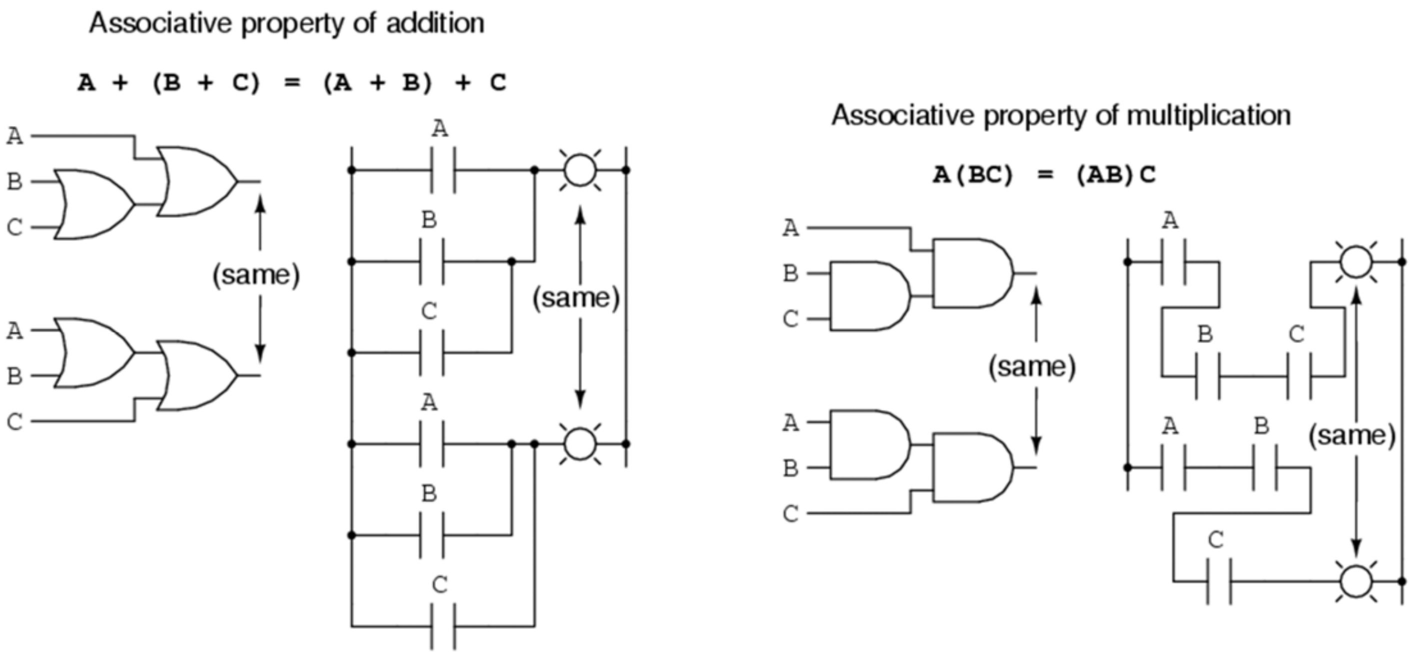
$$x * y = y * x \text{ for all } x, y \in S$$



- *Associative law.*
  - A binary operator * on a set *S* is said to be associative whenever

$$(x * y) * z = x * (y * z) \text{ for all } x, y, z, \in S$$

•



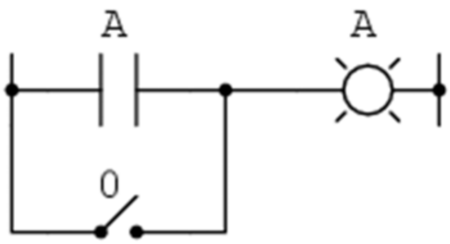
- *Identity element.*
  - A set *S* is said to have an identity element with respect to a binary operation * on *S*
  - if there exists an element $e \in S$ with the property that

$$e * x = x * e = x \text{ for every } x \in S$$

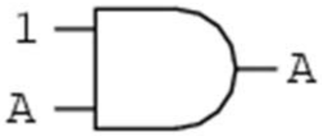
- *Example:* The element 0 is an identity element with respect to the binary operator + on the set of integers *I* = {…, -3, -2, -1, 0, 1, 2, 3,…}, since
- $x + 0 = 0 + x = x$ for any $x \in I$
- *Example:* The element 1 is an identity element with respect to the binary operator . on the set of integers *I* = {…, -3, -2, -1, 0, 1, 2, 3,…}, since
- $x \cdot 1 = 1 \cdot x = x$ for any $x \in I$

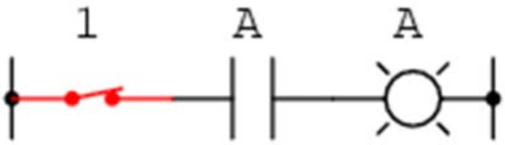
•

$$A + 0 = A$$



$$1A = A$$





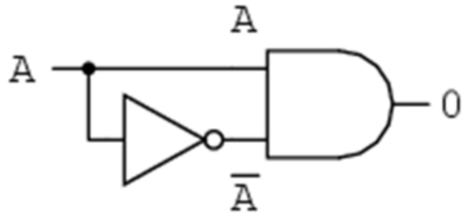
- *Inverse.*
  - A set *S* having the identity element *e* with respect to a binary operator * is said to have an inverse whenever, for every $x \in S$, there exists an element $y \in S$ such that

$$x * y = e$$

- *Example:* In the set of integers, *I*, and the operator +, with *e* = 0, the inverse of an element *a* is (-*a*), since *a* + (-*a*) = 0.

•

$$A\bar{A} = 0$$



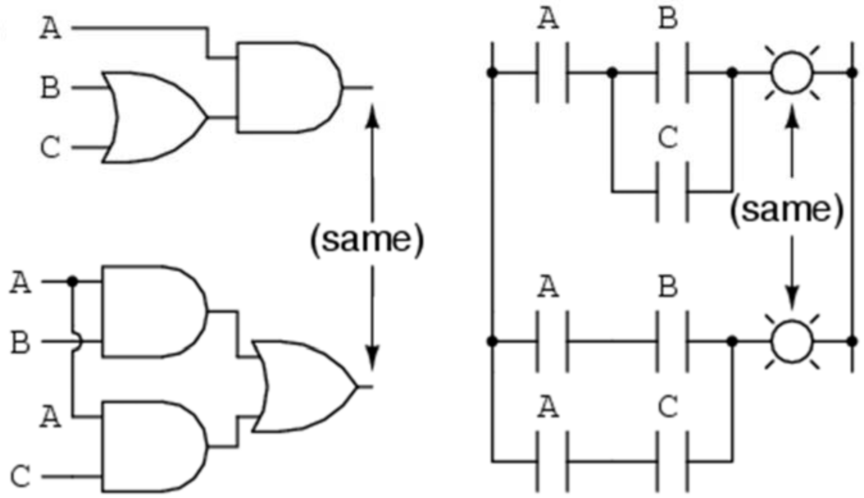
- *Distributive law*
  - If * and . are two binary operators on a set *S*, * is said to be distributive over . Whenever

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

Distributive property

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

•



Basic Boolean algebraic properties

Additive	Multiplicative
$A + B = B + A$	$AB = BA$
$A + (B + C) = (A + B) + C$	$A(BC) = (AB)C$

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

Summery

- The binary operator + defines addition (OR).
- The additive identity is 0.
- The additive inverse defines subtraction.
- The binary operator . defines multiplication (AND).
- The multiplicative identity is 1.
- For a 0, the multiplicative inverse of a = 1/a defines division (i.e., a . 1/a = 1).
- The only distributive law applicable is that of . over +:

$$a \,.\, (b + c) = (a \,.\, b) + (a \,..\, c)$$

Ordinary and Boolean Algebra

- George Boole developed an algebraic system now called *Boolean algebra*.
- Claude E. Shannon introduced a two-valued Boolean algebra called *switching algebra* that represented the properties of bistable electrical switching circuits.

1. (a) The structure is closed with respect to the operator +.

   (b) The structure is closed with respect to the operator . .
2. (a) The element 0 is an identity element with respect to +; that is, $x + 0 = 0 + x = x$.

   (b) The element 1 is an identity element with respect to . ; that is, $x . 1 = 1 . x = x$.
3. (a) The structure is commutative with respect to +; that is, $x + y = y + x$.

   (b) The structure is commutative with respect to . ; that is, $x . y = y . x$.

4. (a) The operator . is distributive over +; that is, $x \cdot (y + z) = (x \cdot y) + (x \cdot z)$.

   (b) The operator + is distributive over . ; that is, $x + (y \cdot z) = (x + y) \cdot (x + z)$.

5. For every element $x \in B$, there exists an element $x \in B$ (called the *complement* of x) such that

   (a) $x + x = 1$ and

   (b) $x \cdot x = 0$,

6. There exist at least two elements $x, y \in B$ such that $x \neq y$.

Comparing Boolean algebra with arithmetic and ordinary algebra:

- The distributive law of + over . (i.e., $x + (y \cdot z) = (x + y) \cdot (x + z)$) is valid for Boolean algebra, but not for ordinary algebra.
- Boolean algebra does not have additive or multiplicative inverses; therefore, there are no subtraction or division operations.
- defines an operator called the *complement* that is not available in ordinary algebra.

- Ordinary algebra deals with the real numbers, which constitute an infinite set of elements. Boolean algebra deals with the as yet undefined set of elements, *B*, but in the two-valued Boolean algebra defined next (and of interest in our subsequent use of that algebra), *B* is defined as a set with only two elements, 0 and 1.

Two-Valued Boolean Algebra

- defined on a set of two elements, *B* = {0, 1}, with rules for the two binary operators + and .

x	y	$x \cdot y$
0	0	0
0	1	0
1	0	0
1	1	1

x	y	$x + y$
0	0	0
0	1	1
1	0	1
1	1	1

x	x'
0	1
1	0

Truth table

- rules are exactly the same as the AND, OR, and NOT operations

- the structure is *closed* with respect to the two operators, each operation is either 1 or 0 and 1, 0 ∈ *B*.
- From the tables, we see that

  (a) 0 + 0 =0 0+ 1 = 1 + 0 = 1;

  (b) 1 . 1 = 1 1. 0 = 0 . 1 = 0.

  This establishes the two *identity elements,* 0 for + and 1 for . ,
- The *commutative* laws are obvious from the symmetry of the binary operator table (A+B = B+A) & (A.B = B.A)

- The *distributive* law *x* . (*y* + *z*) = (*x* . *y*) + (*x* . *z*) can be shown to hold from the operator tables
- The *distributive* law of + over . can be shown to hold by means of a truth table

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

$y+z$	$x \cdot (y+z)$
0	0
1	0
1	0
1	0
0	0
1	1
1	1
1	1

$x \cdot y$	$x \cdot z$	$(x \cdot y) + (x \cdot z)$
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	1	1
1	0	1
1	1	1

Truth table

- the complement table

  (a) $x + x = 1$, since $0 + 0 = 0 + 1 = 1$ and $1 + 1 = 1 + 0 = 1$.

  (b) $x \cdot x = 0$, since $0 \cdot 0 = 0 \cdot 1 = 0$ and $1 \cdot 1 = 1 \cdot 0 = 0$.

Postulates and Basic theorem of Boolean algebra

Postulates and Theorems of Boolean Algebra

Postulates and Theorems of Boolean Algebra

Postulate 2	(a)	$x + 0 = x$	(b)	$x \cdot 1 = x$
Postulate 5	(a)	$x + x' = 1$	(b)	$x \cdot x' = 0$
Theorem 1	(a)	$x + x = x$	(b)	$x \cdot x = x$
Theorem 2	(a)	$x + 1 = 1$	(b)	$x \cdot 0 = 0$
Theorem 3, involution		$(x')' = x$		
Postulate 3, commutative	(a)	$x + y = y + x$	(b)	$xy = yx$
Theorem 4, associative	(a)	$x + (y + z) = (x + y) + z$	(b)	$x(yz) = (xy)z$
Postulate 4, distributive	(a)	$x(y + z) = xy + xz$	(b)	$x + yz = (x + y)(x + z)$
Theorem 5, DeMorgan	(a)	$(x + y)' = x'y'$	(b)	$(xy)' = x' + y'$
Theorem 6, absorption	(a)	$x + xy = x$	(b)	$x(x + y) = x$

Duality *principle*

- postulates were listed in pairs and designated by part (a) and part (b).
- One part may be obtained from the other if the binary operators and the identity elements are interchanged
- we simply interchange OR and AND operators and replace 1's by 0's and 0's by 1's.

Postulate 2	(a)	$x + 0 = x$	(b)	$x \cdot 1 = x$
Postulate 5	(a)	$x + x' = 1$	(b)	$x \cdot x' = 0$
Theorem 2	(a)	$x + 1 = 1$	(b)	$x \cdot 0 = 0$

Theorem 5.a

the truth table for the first DeMorgan's theorem, $(x + y)' = x'y'$, is as follows:

x	y	$x + y$	$(x + y)'$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

x'	y'	$x'y'$
1	1	1
1	0	0
0	1	0
0	0	0

Truth table

THEOREM 6(b): $x(x + y) = x$

x	y
0	0
0	1
1	0
1	1

xy	$x + xy$
0	0
0	0
0	1
1	1

Truth table

- **Operator Precedence**
  - (1) parentheses,
  - (2) NOT,
  - (3) AND, and
  - (4) OR.
- expressions inside parentheses must be evaluated before all other operations. The next operation that holds precedence is the complement, and then follows the AND and, finally, the OR.
- example: demorgan's $(x + y)' = x'y'$

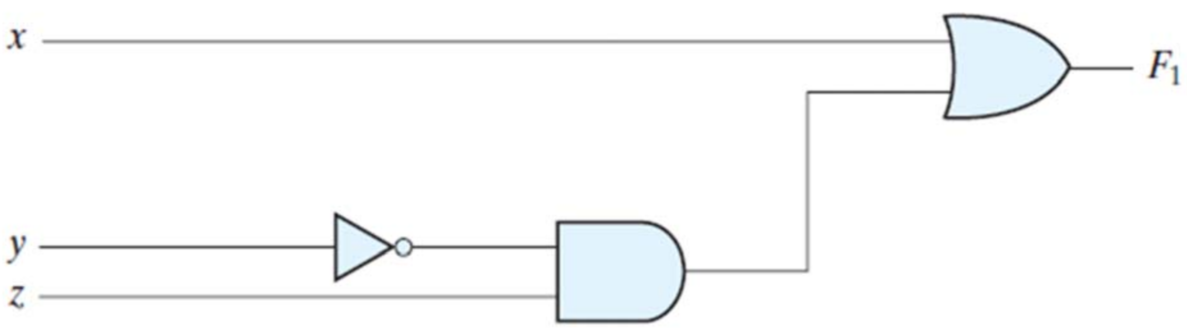
Boolean functions

- Boolean function described by an algebraic expression consists of binary variables

$$F_1 = x + y'z$$

- A Boolean function can be represented in a truth table. The number of rows in the truth table is 2^n, where n is the number of variables in the function.
- The interconnection of gates will dictate the logic expression.

$F_1 = x + y'z$



Logic diagram

$'F_1 = x + y'z$

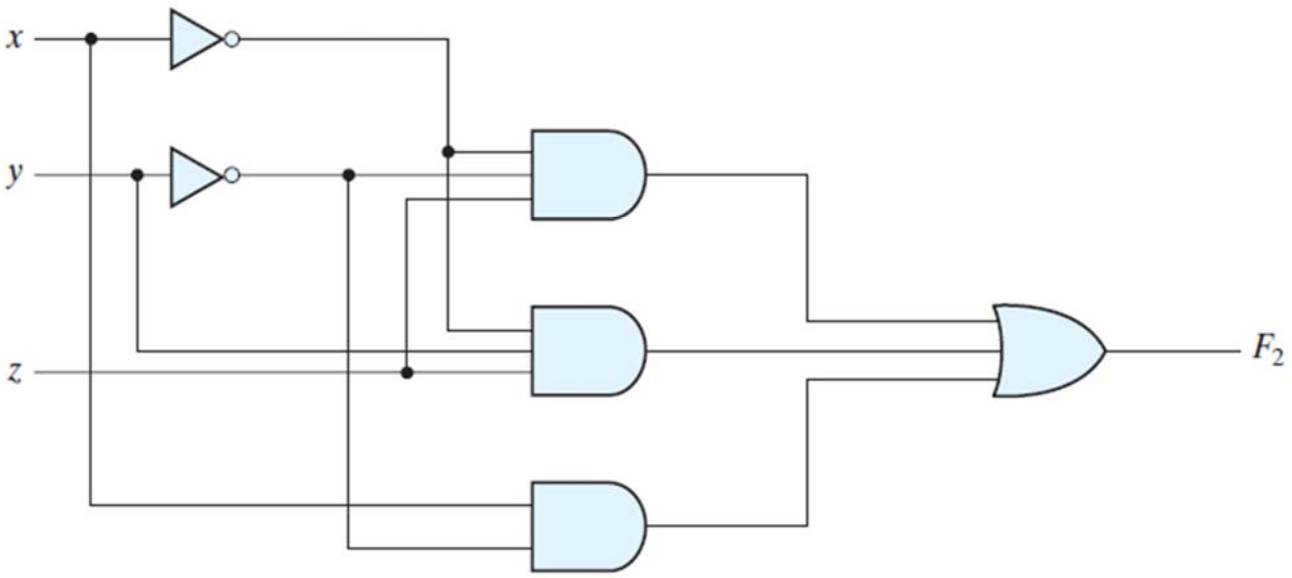
x	y	z
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

Y'	y' z	x	F1
1	0	0	0
1	1	0	1
0	0	0	0
0	0	0	0
1	0	1	1
1	1	1	1
0	0	1	1
0	0	1	1

- it is sometimes possible to obtain a simpler expression for the same function and thus reduce the number of gates in the circuit and the number of inputs to the gate.

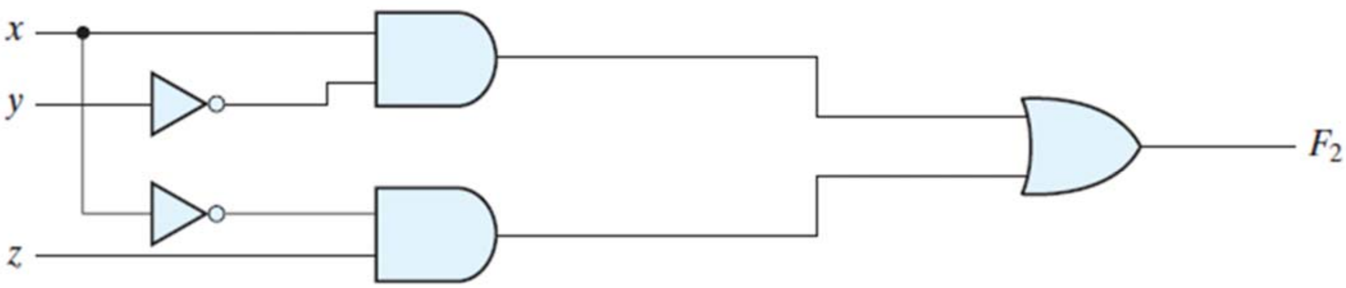
$$F_2 = x'y'z + x'yz + xy'$$

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



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

Logic diagram



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

Logic diagram

• $F_2 = xy' + x'z$

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

X	Y’	Xy’	X’	z	X’z	f2
0	1	0	1	0	0	0
0	1	0	1	1	1	1
0	0	0	1	0	0	0
0	0	0	1	1	1	1
1	1	1	0	0	0	1
1	1	1	0	1	0	1
1	0	0	0	0	0	0
1	0	0	0	1	0	0

Truth table

Algebraic Manipulation

- When a Boolean expression is implemented with logic gates, each term requires a gate and each variable within the term designates an input to the gate.
- We define a *literal* to be a single variable within a term, in complemented or un-complemented form.
- By reducing the number of terms, the number of literals, or both in a Boolean expression, it is often possible to obtain a simpler circuit

Example 01

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

1. $x(x' + y)$

2. $x + x'y$

3. $(x + y)(x + y')$

4. $xy + x'z + yz$

5. $(x + y)(x' + z)(y + z)$

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

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

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

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

4.
$$\begin{aligned} xy + x'z + yz &= xy + x'z + yz(x + x') \\ &= xy + x'z + xyz + x'yz \\ &= xy(1 + z) + x'z(1 + y) \\ &= xy + x'z. \end{aligned}$$

5. $(x + y)(x' + z)(y + z) = (x + y)(x' + z)$, by duality from function 4.

Complement of a Function

- The complement of a function may be derived algebraically through DeMorgan's theorems

$$(x + y)' = x'y' \qquad (xy)' = x' + y'$$

- DeMorgan's theorems can be extended to three or more variables.

$$\begin{aligned}(A + B + C)' &= (A + x)' && \text{let } B + C = x \\ &= A'x' && \text{by theorem 5(a) (DeMorgan)} \\ &= A'(B + C)' && \text{substitute } B + C = x \\ &= A'(B'C') && \text{by theorem 5(a) (DeMorgan)} \\ &= A'B'C' && \text{by theorem 4(b) (associative)}\end{aligned}$$

- $$(A + B + C + D + \cdots + F)' = A'B'C'D' \ldots F'$$
$$(ABCD \ldots F)' = A' + B' + C' + D' + \cdots + F'$$

Example 02

Find the complement of the functions $F_1 = x'yz' + x'y'z$ and $F_2 = x(y'z' + yz)$.

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

$$F_1' = (x'yz' + x'y'z)' = (x'yz')'(x'y'z)' = (x + y' + z)(x + y + z')$$

$$\begin{aligned} F_2' = [x(y'z' + yz)]' &= x' + (y'z' + yz)' = x' + (y'z')'(yz)' \\ &= x' + (y + z)(y' + z') \\ &= x' + yz' + y'z \end{aligned}$$

Example 03

Find the complement of the functions F_1 and F_2 of Example 2.: by taking their duals and complementing each literal.

$$F_1 = x'yz' + x'y'z \text{ and } F_2 = x(y'z' + yz).$$

1. $F_1 = x'yz' + x'y'z.$
   The dual of F_1 is $(x' + y + z')(x' + y' + z)$.
   Complement each literal: $(x + y' + z)(x + y + z') = F_1'$.
2. $F_2 = x(y'z' + yz)$.
   The dual of F_2 is $x + (y' + z')(y + z)$.
   Complement each literal: $x' + (y + z)(y' + z') = F_2'$.

Thanks,..

See you next week (ISA),…