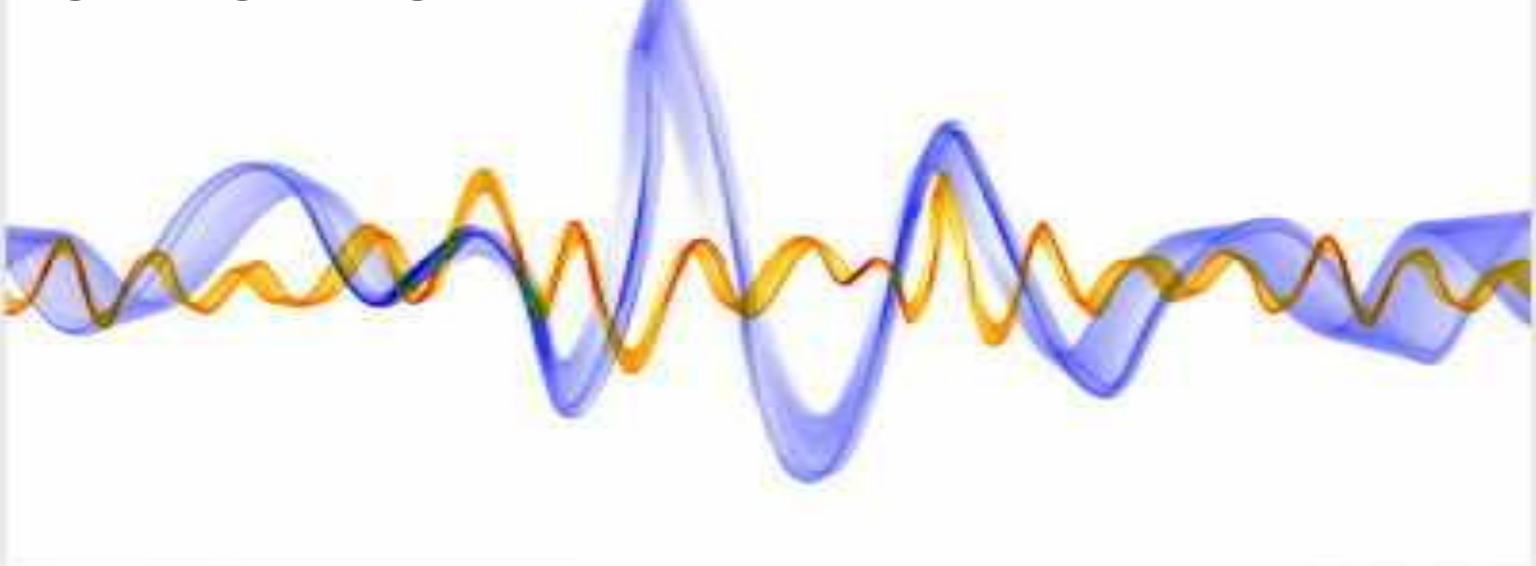
**Digital Logic Design**



# Lecture 3:
## Chapter 2: Boolean Algebra and Logic Gates

Mirvat Al-Qutt, Ph.D
Computer Systems Department , FCIS,
Ain Shams University

# Basic Definitions

- **The Postulates Boolean Algebra**

**Table 2.1**
*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) | $x(x + y) = x$ |

# Basic Definitions

- **Duality Principle ( DeMorgan's Theorem)**
- Verify DeMorgan's Theorem

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

| $x$ | $y$ | $x'$ | $y'$ | $x+y$ | $(x+y)'$ | $x'y'$ | $Xy$ | $x'+y'$ | $(xy)'$ |
|---|---|---|---|---|---|---|---|---|---|
| 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 |

# Basic Definitions

- **Duality Principle ( DeMorgan's Theorem)**
- Verify DeMorgan's Theorem

$x'y + xz'$

$= x'y + xz'$

$= ((x'y)' \cdot (xz')')'$

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

# Basic Definitions

- **Consensus Theorem**

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

**Proof:**

$xy + x'z + yz$

$= xy + x'z + 1.yz$

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

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

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

$= xy\,(1+z) + x'z\,(1+ y)$

$= xy + x'z$

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

**Proof:**

$(x+y)\bullet(x'+z)\bullet(y+z)$

$= (x+y)\bullet(x'+z)\bullet(0+y+z)$

$= (x+y)\bullet(x'+z)\bullet((xx')+y+z)$

$= (x+y)\bullet(x'+z)\bullet(x+y+z)\bullet(x'+y+z)$

$= (x+y)\bullet(0\bullet z)(x'+z)\bullet(0\bullet y)$

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

# Operator Precedence

- The operator precedence for evaluating Boolean Expression is
  - Parentheses
  - NOT
  - AND
  - OR
- Examples
  - $x\, y' + z$
  - $(x\, y + z)'$

# Boolean Functions

- A Boolean function my include:
  - Binary variables
  - Binary operators OR and AND
  - Unary operator NOT
  - Parentheses
- Examples
  - $F_1 = x\, y\, z'$
  - $F_2 = x + y'z$
  - $F_3 = x'\, y'\, z + x'\, y\, z + x\, y'$
  - $F_4 = x\, y' + x'\, z$

- **The truth table of $2^n$ entries (n=number of variables)**
- **Two Boolean expressions may specify the same function $F_3 = F_4$**

| $x$ | $y$ | $z$ | $F_1$ | $F_2$ | $F_3$ | $F_4$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 | 1 | 0 | 0 |

# Boolean Functions

- Different representation of Boolean Function
  - Boolean Expression (Many)
  - Truth Table (Unique)
  - Logic Gates Diagram (Many)

- Examples
  - $F_1 = x\,y\,z'$
  - $F_2 = x + y'z$
  - $F_3 = x'\,y'\,z + x'\,y\,z + x\,y'$
  - $F_4 = x\,y' + x'\,z$

| $x$ | $y$ | $z$ | $F_1$ | $F_2$ | $F_3$ | $F_4$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 | 1 | 0 | 0 |

# Boolean Functions

- Implementation
  with logic gates

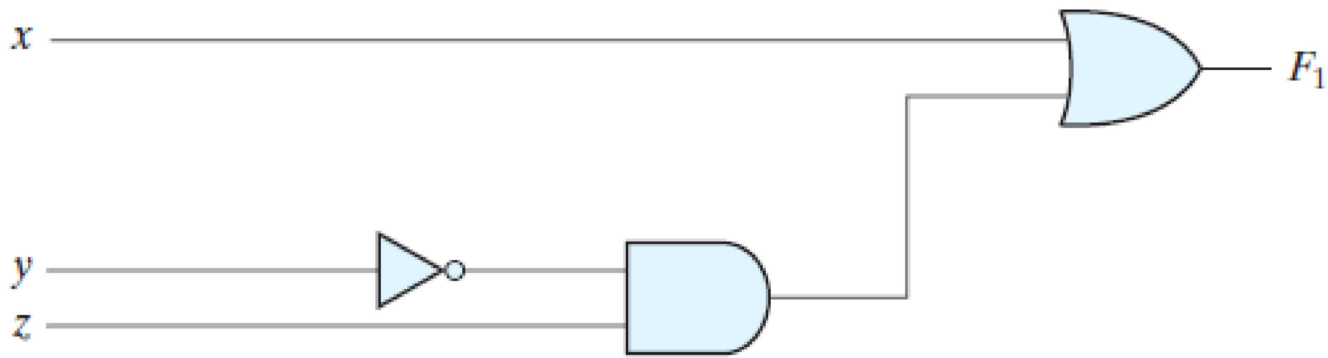



**FIGURE 2.1**
Gate implementation of $F_1 = x + y'z$

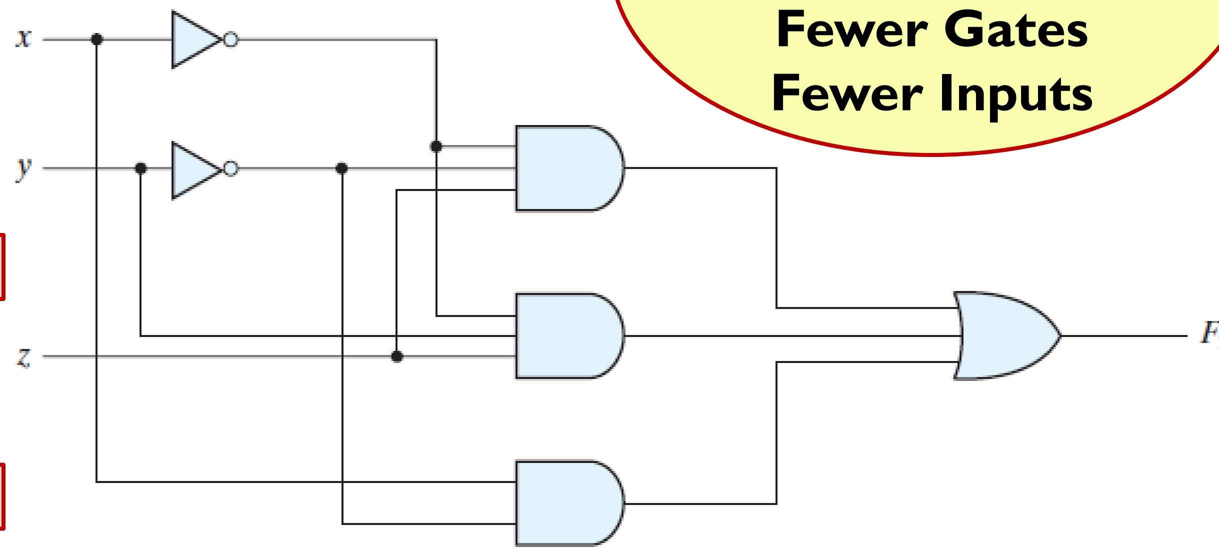
# Boolean Functions

- Economical
- Simpler ,
- Less Cost
- Fewer Gates
- Fewer Inputs

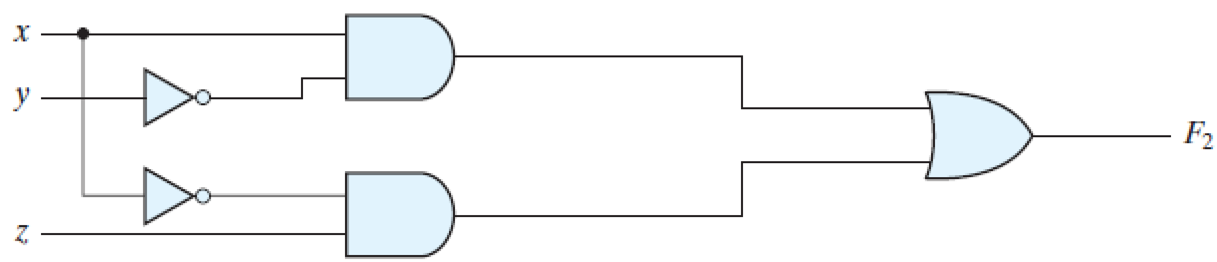
- Implementation with logic gates

$$F_3 = x'y'z + x'yz + xy'$$
$$= x'z(y'+y) + xy'$$
$$= x'z(1) + xy'$$
$$= x'z + xy'$$

**Simplification**




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




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

**FIGURE 2.2**
Implementation of Boolean function $F_2$ with gates

# Boolean Functions

- **Simplify the following functions**

$$
\begin{aligned}
F &= x(x' + y) \\
&= xx' + xy \\
&= 0 + xy \\
&= xy
\end{aligned}
$$

$$
\begin{aligned}
F &= x + x'y \\
&= (x + x')(x + y) \\
&= 1(x + y) \\
&= (x + y)
\end{aligned}
$$

$$
\begin{aligned}
F &= (x + y)(x + y') \\
&= x + xy + xy' + yy' \\
&= x(1 + y + y') \\
&= x
\end{aligned}
$$

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

**Consensus Theorem**

# Complement of a Function

- The complement of a function *F* is *F'* and is obtained from an interchange of 0's for 1's and 1's for 0's in the value of *F*.

- The complement of a function may be derived algebraically through DeMorgan's theorems,
  - 3 variables DeMorgan's theorem
  - $(A+B+C)' = (A+X)'$ let $B+C = X$

| | |
|---|---|
| $(A+B+C)' = (A+X)'$ | let $B+C = X$ |
| $= A'X'$ | by theorem 5(a) (DeMorgan's) |
| $= A'(B+C)'$ | substitute $B+C = X$ |
| $= A'(B'C')$ | by DeMorgan's theorem |
| $= A'B'C'$ | by associative theorem |

# Complement of a Function

- The complement of a function *F* is *F'* and is obtained from an interchange of 0's for 1's and 1's for 0's in the value of *F*.

- <u>Generalization:</u> a function is obtained by interchanging AND and OR operators and complementing each literal.
  - F = $A+B+C+D+ \ldots$ <u>*Then*</u> $F' = (A+B+C+D+ \ldots)' = A'B'C'D' \ldots$
  - F = $ABCD \ldots$ <u>*Then*</u> $F' = (ABCD \ldots)' = A' + B' + C' + D' \ldots$

# Complement of a Function

- Find the Complement of the following functions

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

- $F_2 = x(y' z' + y z)$
- $F_2' = [x(y'z'+yz)]'$
  $= x' + (y'z'+yz)'$
  $= x' + (y'z')' (yz)'$
  $= x' + (y+z) (y'+z')$
  $= x' + yz'+y'z$

# Canonical and Standard Forms

## Drive truth table of the following function

F = x’ y’

| $x$ | $y$ | $x'$ | $y'$ | $F$ |
|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 |

# Canonical and Standard Forms

**Drive truth table of the following function**

F = x' y' + x'y

| $x$ | $y$ | $x'$ | $y'$ | $x'y'$ | $x'y$ | $F$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 |

# Canonical and Standard Forms

## Drive truth table of the following function

F = x' y' + x'y + xy'

| $x$ | $y$ | $x'$ | $y'$ | $x'y'$ | $x'y$ | $xy'$ | $F$ |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |

# Canonical and Standard Forms

## Drive truth table of the following function

F = x'y' + x'y + xy' + xy

| $x$ | $y$ | $x'$ | $y'$ | $F$ |
|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 |

# Canonical and Standard Forms

## Drive truth table of the following function

F = x' y' z'

| $x$ | $y$ | $z$ | $x'$ | $y'$ | $z'$ | $F$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 |

# Canonical and Standard Forms

## Drive truth table of the following function

F = x' y' z' + x'y'z

| $x$ | $y$ | $z$ | $x'$ | $y'$ | $z'$ | $F$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 0 | 1 | 1 | 1 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 |

# Canonical and Standard Forms

## Minterms

- A minterm (standard product): an AND term consists of all literals in their normal form or in their complement form.

- For example,
- two binary variables *x* and *y*,
  - ***xy, xy', x'y, x'y'***
- It is also called a standard product.
- *n* variables can be combined to form $2^n$ minterms.

| x | y | Term | Symbol |
|---|---|---|---|
| 0 | 0 | x'y' | $m_0$ |
| 0 | 1 | x'y | $m_1$ |
| 1 | 0 | xy' | $m_2$ |
| 1 | 1 | xy | $m_3$ |

Minterms and Maxterms for Three Binary Variables

| | | | Minterms | |
|---|---|---|---|---|
| *x* | *y* | *z* | Term | Designation |
| 0 | 0 | 0 | $x'y'z'$ | $m_0$ |
| 0 | 0 | 1 | $x'y'z$ | $m_1$ |
| 0 | 1 | 0 | $x'yz'$ | $m_2$ |
| 0 | 1 | 1 | $x'yz$ | $m_3$ |
| 1 | 0 | 0 | $xy'z'$ | $m_4$ |
| 1 | 0 | 1 | $xy'z$ | $m_5$ |
| 1 | 1 | 0 | $xyz'$ | $m_6$ |
| 1 | 1 | 1 | $xyz$ | $m_7$ |

# Canonical and Standard Forms

## Minterms

- Sum of minterms for each combination of variables that produces a (1) in the function

$$\sum$$

*Minterms and Maxterms for Three Binary Variables*

| | | | Minterms | |
|---|---|---|---|---|
| **x** | **y** | **z** | **Term** | **Designation** |
| 0 | 0 | 0 | $x'y'z'$ | $m_0$ |
| 0 | 0 | 1 | $x'y'z$ | $m_1$ |
| 0 | 1 | 0 | $x'yz'$ | $m_2$ |
| 0 | 1 | 1 | $x'yz$ | $m_3$ |
| 1 | 0 | 0 | $xy'z'$ | $m_4$ |
| 1 | 0 | 1 | $xy'z$ | $m_5$ |
| 1 | 1 | 0 | $xyz'$ | $m_6$ |
| 1 | 1 | 1 | $xyz$ | $m_7$ |

# Canonical and Standard Forms

## Drive truth table of the following function

F = x' y'

| $x$ | $y$ | $x'$ | $y'$ | $F$ |
|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 |

F = $\sum m_0$

# Canonical and Standard Forms

## Drive truth table of the following function

F = x' y' + x'y

| $x$ | $y$ | $x'$ | $y'$ | $F$ |
|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 |

F = $\sum(m_0, m_1)$

F = x' y' + x'y
=(00,01)
=$\sum(m_0, m_1)$

# Canonical and Standard Forms

## Drive truth table of the following function

F = x'y' + x'y + xy' + xy

| $x$ | $y$ | $x'$ | $y'$ | $F$ |
|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 |

F = $\sum(m_0, m_1, m_2, m_3)$

**Much More Compact Form**
**F = 1**

# Canonical and Standard Forms

## Drive truth table of the following function

F = x' y' z' = $\sum(m_0)$

| $x$ | $y$ | $z$ | $x'$ | $y'$ | $z'$ | $F$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 |

# Canonical and Standard Forms

## Drive truth table of the following function

F = x' y' z' + x'y'z $= \sum(m_0, m_1)$

000, 001

| $x$ | $y$ | $z$ | $x'$ | $y'$ | $z'$ | $F$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 0 | 1 | 1 | 1 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 |

# Canonical and Standard Forms

**Write the following function in terms sum of its minterms**

| $x$ | $y$ | $z$ | $F1$ | $F1$ | $F3$ |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 0 | 1 |

F1 $= \sum(m_0, m_1, m_7)$

F1 = x'y'z' + x'y'z+ xyz

F2 $= \sum(m_1, m_4)$

F2 = x'y'z+ xy'z'

F3 $= \sum(m_{0,} m_1, m_5, m_6, m_7)$

# Canonical and Standard Forms

## Minterms and Maxterms

- A **minterm (standard product)**: an AND term consists of all literals in their normal form or in their complement form.

$\sum$

**Sum means ORing**

- A **maxterm (standard sums)**: an OR term
  - It is also called a standard sum.
  - $2^n$ maxterms.

$\prod$

**Product means ANDing**

# Canonical and Standard Forms

**Drive truth table of the following function**

F = x + y

| $x$ | $y$ | $x'$ | $y'$ | $F$ |
|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 |

# Canonical and Standard Forms

## Drive truth table of the following function

F = ( x + y ) ( x + y' )

| $x$ | $y$ | $x'$ | $y'$ | $x+y$ | $x+y'$ | $F$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 | 1 | 1 |

# Minterms and Maxterms

- Each maxterm is the complement of its corresponding minterm, and vice versa.

**Table 2.3**
*Minterms and Maxterms for Three Binary Variables*

| | | | Minterms | | Maxterms | |
|---|---|---|---|---|---|---|
| **x** | **y** | **z** | **Term** | **Designation** | **Term** | **Designation** |
| 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 | $xyz$ | $m_7$ | $x' + y' + z'$ | $M_7$ |

# Minterms and Maxterms

Challenge

- Express the following functions in terms of

1. sum of standard product terms (minterms)
2. product of standard sum terms (Maxterms)

**Table 2.4**
*Functions of Three Variables*

| $x$ | $y$ | $z$ | Function $f_1$ | Function $f_2$ |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

F1= $\sum(m_1, m_4, m_7)$

= x'y'z + xy'z'+ xyz

= $\prod(M_0 M_2 M_3 M_5 M_6)$

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

(x'+y+z') (x'+y'+z)

# Minterms and Maxterms

**Challenge**

- Convert from any form to the other

**Table 2.4**
*Functions of Three Variables*

| $x$ | $y$ | $z$ | Function $f_1$ | Function $f_2$ |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

F1= $\sum(m_1, m_4, m_7)$

F1=x'y'z + xy'z'+ xyz

F1=(x+y+z) (x+y'+z) (x+y'+z')(x'+y+z') (x'+y'+z)

F1= $\prod(M_0 M_2 M_3 M_5 M_6)$

# Minterms and Maxterms

- Express the following functions in terms of

1. sum of its minterms
2. product of its Maxterms

**Table 2.4**
*Functions of Three Variables*

| $x$ | $y$ | $z$ | Function $f_1$ | Function $f_2$ |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

F2= $\sum(m_3, m_5, m_6, m_7)$

$= \prod(M_0 M_1 M_2 M_4)$

# Minterms and Maxterms

- Any Boolean function can be expressed as:
  - A sum of minterms expressions ("sum" meaning the ORing of terms )

$$F=XYZ+XY'Z+X'Y'Z'+\ldots\ldots$$

  - A product of maxterms expressions ("product" meaning the ANDing of terms).

$$F=(X+Y+Z).(X+Y'+Z).(X'+Y'+Z')\ \ldots\ldots$$

  - Both Boolean functions are said to be in **Canonical** form.

# Sum of Minterms

- Sum of minterms: there are $2^n$ minterms and $2^{2n}$ combinations of functions with $n$ Boolean variables.
- Example : express $F = A+B'C$ as a sum of **minterms**.
  - $F = A+B'C$

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

    $= AB + AB' + B'C$

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

    $= ABC+ABC'+AB'C+AB'C'+A'B'C$

    $= A'B'C + AB'C' + AB'C + ABC' + ABC$

    $= m_1 + m_4 + m_5 + m_6 + m_7$

    $= \Sigma(1, 4, 5, 6, 7)$

1

2

or, built the truth table first

**Table 2.5**
*Truth Table for $F = A + B'C$*

| A | B | C | F |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

# Sum of Minterms

- Sum of minterms: there are $2^n$ minterms and $2^{2n}$ combinations of functions with *n* Boolean variables.
- Example: express $F = A+B'C$ as a product of **maxterms**
  - $F = A+B'C$
    $= (A+B')(A+C)$
    $= (A+B'+CC')(A+C+BB'')$
    $= (A+B'+C)(A+B'+C')(A+B+C)(A+B'+\cancel{C})$
    $=\prod(M_0M_2M_3)$

1

2

**or, built the truth table first**

**Table 2.5**
*Truth Table for F = A + B'C*

| A | B | C | F |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

# Product of Maxterms

- Product of maxterms: using distributive law to expand.

- Example : express $F = xy + x'z$ as a product of **maxterms**.
  - $F = xy + x'z$

1

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

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

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

$= (x'+y+zz')(x+z+yy')(y+z+xx')$

$=(x'+y+z)(x'+y+z')(x+z+y)\ (x+z+y')(y+\cancel{z}+x)(y+\cancel{z}+x')$

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

$= M_0M_2M_4M_5$

$= \Pi(0, 2, 4, 5)$

2

**or, built the truth table first**

**Table 2.6**

*Truth Table for $F = xy + x'z$*

| $x$ | $y$ | $z$ | $F$ |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

# Complement of a Function Expressed in Canonical Forms

- The complement of a function expressed as the sum of minterms equals the sum of minterms missing from the original function.
  - $F(A, B, C) = \Sigma(1, 4, 5, 6, 7)$
  - $F(A, B, C) = = \Pi(0, 2, 3)$

Thus,

- $F'(A, B, C) = \Sigma(0, 2, 3)$
- $F'(A, B, C) = \Pi(1, 4, 5, 6, 7)$

- By DeMorgan's theorem $m_j' = M_j$

| x | y | z | F1 | F1' |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 | 0 |

# Conversion between Canonical Forms

- To convert from one canonical form to another: **interchange** the symbols $\Sigma$ and $\Pi$ and list those numbers **missing** from the original form
  - $\Sigma$ of 1's
  - $\Pi$ of 0's

# Conversion between Canonical Forms

- Example
  - $F = xy + x'z$
  - $F(x, y, z) = \Sigma(1, 3, 6, 7)$
  - $F(x, y, z) = \Pi\ (0, 2, 4, 6)$

**Table 2.6**
*Truth Table for $F = xy + x'z$*

| $x$ | $y$ | $z$ | $F$ |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

# Canonical Forms vs. Standard Forms

## Canonical Forms

- Each minterm or maxterm must contain **all the variables** either complemented or uncomplemented,
- Sum of minterms (Product terms)
- OR Product of Maxterms (sum terms)

## Standard forms

- the terms that form the function may obtain **one, two, or any number** of literals, .
- There are two types of standard forms:
  - Sum of products:

    $F_1 = y' + xy + x'yz'$
  - Product of sums:

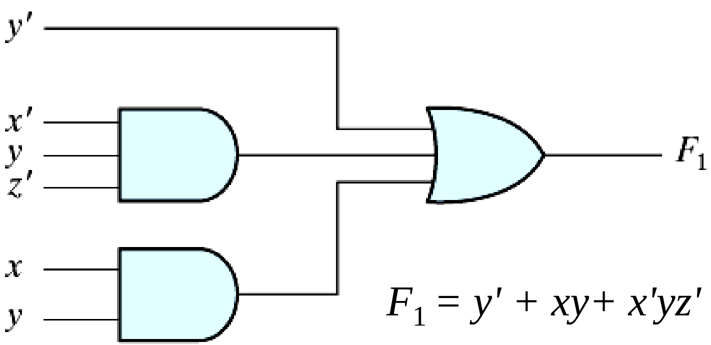
    $F_2 = x(y'+z)(x'+y+z')$

# Standard Forms

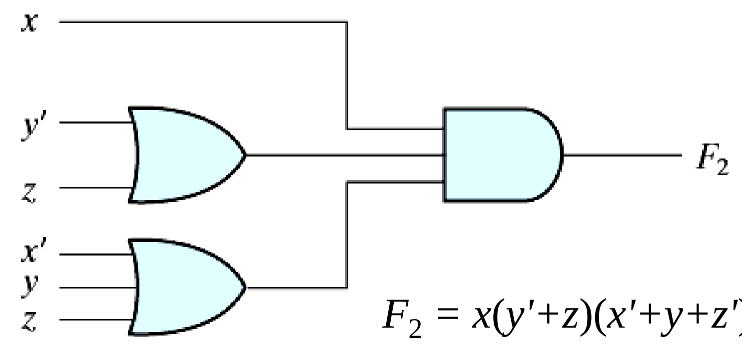
- A Boolean function may be expressed in a **nonstandard** form
  - $F_3 = AB + C(B + A)$
- But it can be changed to a standard form by using The distributive law
  - $F_3 = AB + C(B + A) = AB + BC + AC$
- And it can be changed to a canonical form by using The distributive law after adding missing literal
  - $F_3 = AB + BC + AC = AB(C+C')+BC(A+A')+AC(B+B')$
  - $=ABC+ABC'+\cancel{ABC}+A'BC+\cancel{ABC}+AB'C$
  - $=ABC+ABC'+A'BC+AB'C$

# Implementation

- Two-level implementation

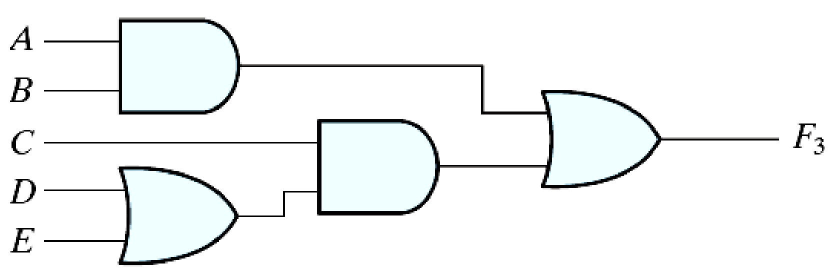



(a) Sum of Products

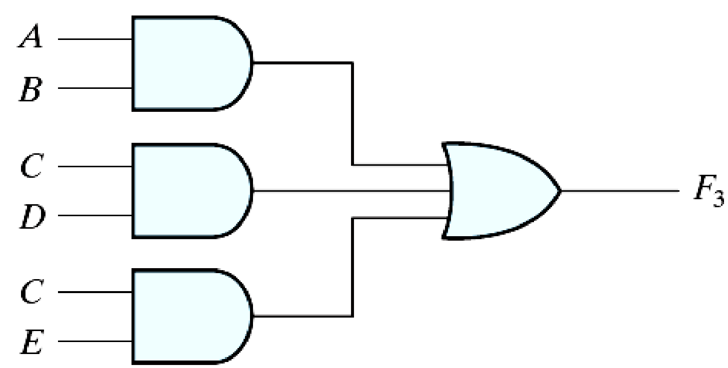



(b) Product of Sums

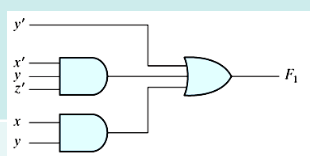
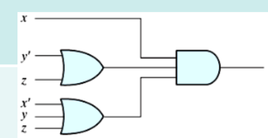
- Multi-level implementation




(a) $AB + C(D + E)$




(b) $AB + CD + CE$

| SOP | POS |
|---|---|
| Sum of minterms<br>$$F = \sum (m_0, m_2, \ldots . m_i)$$<br>Sum of terms that function gives 1 | Product of Maxterms<br>$$F = \prod (M_0 M_1 \ldots \ldots . M_i)$$<br>Product of terms that function gives 0 |
| Minterms ( Locate 1's)<br>$m_0$ = x'y'z' = 000<br>$m_1$ = x'y'z = 001<br>….<br>$m_7$ = xyz = 111 | Maxterms ( Locate 0's)<br>$M_0$ = x+y+z = 000<br>$M_1$ = x+y+z' = 001<br>….<br>$M_7$ =x'+y+'z' = 111 |
| Convert Boolean function to SOP<br>By multiplying each term by the missing variable Ored with its complement<br>F = xy = xy(z+z') = xyz +xyz'<br><br><br>(a) Sum of Products | Convert Boolean function to POS<br>By expanding using distributive law and then for each term add the missing variable ANDed with its complement<br>F= x+y = x+y+zz' = (x+y+z)(x+y+z')<br><br><br>(b) Product of Sums |
| Logic Diagram:<br>• 2 level implantation<br>• Level of AND gates followed by one OR gate | Logic Diagram:<br>• 2 level implantation<br>• Level of OR gates followed by one AND gate |

# Other Logic Operations

- $2^n$ rows in the truth table of n binary variables.
- $2^{2^n}$ functions for n binary variables.
- 16 functions of two binary variables.

**Table 2.7**
*Truth Tables for the 16 Functions of Two Binary Variables*

| $x$ | $y$ | $F_0$ | $F_1$ | $F_2$ | $F_3$ | $F_4$ | $F_5$ | $F_6$ | $F_7$ | $F_8$ | $F_9$ | $F_{10}$ | $F_{11}$ | $F_{12}$ | $F_{13}$ | $F_{14}$ | $F_{15}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |

- All the new symbols except for the exclusive-OR symbol are not in common use by digital designers.

# Boolean Expressions

**Table 2.8**
*Boolean Expressions for the 16 Functions of Two Variables*

| Boolean Functions | Operator Symbol | Name | Comments |
|---|---|---|---|
| $F_0 = 0$ | | Null | Binary constant 0 |
| $F_1 = xy$ | $x \cdot y$ | AND | $x$ and $y$ |
| $F_2 = xy'$ | $x/y$ | Inhibition | $x$, but not $y$ |
| $F_3 = x$ | | Transfer | $x$ |
| $F_4 = x'y$ | $y/x$ | Inhibition | $y$, but not $x$ |
| $F_5 = y$ | | Transfer | $y$ |
| $F_6 = xy' + x'y$ | $x \oplus y$ | Exclusive-OR | $x$ or $y$, but not both |
| $F_7 = x + y$ | $x + y$ | OR | $x$ or $y$ |
| $F_8 = (x + y)'$ | $x \downarrow y$ | NOR | Not-OR |
| $F_9 = xy + x'y'$ | $(x \oplus y)'$ | Equivalence | $x$ equals $y$ |
| $F_{10} = y'$ | $y'$ | Complement | Not $y$ |
| $F_{11} = x + y'$ | $x \subset y$ | Implication | If $y$, then $x$ |
| $F_{12} = x'$ | $x'$ | Complement | Not $x$ |
| $F_{13} = x' + y$ | $x \supset y$ | Implication | If $x$, then $y$ |
| $F_{14} = (xy)'$ | $x \uparrow y$ | NAND | Not-AND |
| $F_{15} = 1$ | | Identity | Binary constant 1 |


Thank You!