# Chapter 6
# Boolean Algebra

6.1 Introduction

6.2 Operations on Boolean Algebra

6.3 Boolean Function

6.4 Minimal Sums of Products: Karnaugh Map (*K*-map)

# 6.1 Introduction

- Boolean algebra is a branch of mathematics concerned with a logical calculus in which structures and rules are applied to logical symbols in the same way that ordinary algebra is applied to symbols representing numerical quantities.

# 6.1 Introduction (cont)

- Boolean variables are the variables that can take only value 0 or 1.
- The simplest Boolean algebra consists of set {0, 1} together with the operations of disjunction ($\vee$), conjunction ($\wedge$), and negation (’).

# 6.2 Operations on Boolean Algebra

- Disjunction

| $\vee$ | 0 | 1 |
|---|---|---|
| 0 | | |
| 1 | | |

- Conjunction

| $\wedge$ | 0 | 1 |
|---|---|---|
| 0 | | |
| 1 | | |

# 6.2 Operations on Boolean Algebra (cont)

- Negation
  - $0' = 1$
  - $1' = 0$
- For Boolean variables *x* and *y*, the operations conjunction, disjunction, and negation can be represented in a table similar to the truth table, where the truth values F and T are replaced by 0 and 1, not $x$ by $x'$, $x$ and $y$ by $x \wedge y$ ($xy$), $x$ or $y$ by $x \vee y$ ($x + y$).

# 6.2 Operations on Boolean Algebra (cont)

| $x$ | $x'$ |
|---|---|
| 0 | 1 |
| 1 | 0 |

| $x$ | $y$ | $x \wedge y$ | $x \vee y$ |
|---|---|---|---|
| 0 | 0 | | |
| 0 | 1 | | |
| 1 | 0 | | |
| 1 | 1 | | |

# 6.2 Operations on Boolean Algebra (cont)

- Boolean expression is an expression composed of Boolean variables and the operators $\vee$, $\wedge$, and ’.
- Two Boolean expressions are equivalent if they have the same truth table.

# 6.2 Operations on Boolean Algebra (cont)

- Laws of Boolean algebra

| | |
|---|---|
| 1. $(x')' = x$ | Involution Property |
| 2. $(x \wedge y)' = x' \vee y'$<br>$(x \vee y)' = x' \wedge y'$ | De Morgan's Laws |
| 3. $x \wedge y = y \wedge x$<br>$x \vee y = y \vee x$ | Commutative Laws |

# 6.2 Operations on Boolean Algebra (cont)

| | |
|---|---|
| 4. $(x \wedge y) \wedge z = x \wedge (y \wedge z)$<br>$(x \vee y) \vee z = x \vee (y \vee z)$ | Associative Laws |
| 5. $x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$<br>$x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$ | Distributive Laws |
| 6. $x \wedge x = x$<br>$x \vee x = x$ | Idempotent Laws |

# 6.2 Operations on Boolean Algebra (cont)

| | |
|---|---|
| 7. $x \wedge (x \vee y) \equiv x$<br>$x \vee (x \wedge y) = x$ | Absorption Laws |
| 8. $x \wedge x' = 0$<br>$x \vee x' = 1$ | Inverse Laws |
| 9. $x \wedge 0 = 0$<br>$x \vee 1 = 1$ | Dominance Laws |
| 10. $x \wedge 1 = x$<br>$x \vee 0 = x$ | Identity Laws |

# 6.2 Operations on Boolean Algebra (cont)

- The following table summarises the correspondence between Boolean operations, the logical operators in propositional calculus and set operations:

| Logical Operation | Set Operation | Boolean Operation |
|---|---|---|
| not, ~ | — | ’ |
| or, $\vee$ | $\cup$ | $\vee$ |
| and, $\wedge$ | $\cap$ | $\wedge$ |

# E.g.1

Show that $(x' \wedge y)' \wedge (x \vee y) = x$.

# 6.3 Boolean Function

- A Boolean function of the $n$ Boolean variables, $x_1$, $x_2$, …, $x_n$, is a function $f : B^n \rightarrow B$ such that $f(x_1, x_2, \ldots, x_n)$ is a Boolean expression.
- Boolean function can be expressed in an equivalent standard form, called the disjunctive normal form (sum of products) (refer Chapter 2).

# E.g.2

Construct a truth table for the Boolean function $f : B^3 \to B$ where

$f(x, y, z) = (x \wedge y) \vee (x' \wedge z)$.

Then write the disjunctive normal form for $f(x, y, z)$.

| $x$ | $y$ | $z$ | $x \wedge y$ | $x'$ | $x' \wedge z$ | $(x \wedge y) \vee (x' \wedge z)$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | | | | |
| 0 | 0 | 1 | | | | |
| 0 | 1 | 0 | | | | |
| 0 | 1 | 1 | | | | |
| 1 | 0 | 0 | | | | |
| 1 | 0 | 1 | | | | |
| 1 | 1 | 0 | | | | |
| 1 | 1 | 1 | | | | |

# 6.4 Minimal Sums of Products: Karnaugh Map (*K*-map)

- To find a ‘simpler’ equivalent equation, i.e. use fewer symbols than the original expression.
- Applied to disjunctive normal form of the given expression.

# 6.4 Minimal Sums of Products: Karnaugh Map (*K*-map) (cont)

- E.g.

  $(x' \wedge y' \wedge z) \vee (x' \wedge y \wedge z) \vee (x \wedge y' \wedge z)$

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

  $=$

# 6.4 Minimal Sums of Products: Karnaugh Map (*K*-map) (cont)

- A Karnaugh map is a device invented as an aid to logical circuit design that uses a visual display to simplify a sum-of-product Boolean expression by indicating which pairs of minterms can be brought together and merged into a single simpler expression.

# 6.4 Minimal Sums of Products: Karnaugh Map (*K*-map) (cont)

- Given a sum-of-product Boolean expression.
  - If two variables $x$ and $y$ are involved, a Karnaugh map consists of a table with two rows and two columns. Each cell corresponds to a minterm.

| | $y'$ | $y$ |
|---|---|---|
| $x'$ | $x' \wedge y'$ | $x' \wedge y$ |
| $x$ | $x \wedge y'$ | $x \wedge y$ |

| | |
|---|---|
| 00 | 01 |
| 10 | 11 |

# 6.4 Minimal Sums of Products: Karnaugh Map (*K*-map) (cont)

- A rectangle of size 2 × 4 is used for three variables *x*, *y*, and *z* in a Boolean expression.

| | $y'$ | $y'$ | $y$ | $y$ |
|---|---|---|---|---|
| $x'$ | $x' \wedge y' \wedge z'$ | $x' \wedge y' \wedge z$ | $x' \wedge y \wedge z$ | $x' \wedge y \wedge z'$ |
| $x$ | $x \wedge y' \wedge z'$ | $x \wedge y' \wedge z$ | $x \wedge y \wedge z$ | $x \wedge y \wedge z'$ |
| | $z'$ | $z$ | $z$ | $z'$ |

| | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | 000 | 001 | 011 | 010 |
| 1 | 100 | 101 | 111 | 110 |

# 6.4 Minimal Sums of Products: Karnaugh Map (*K*-map) (cont)

- *K*-map with four variables, *w*, *x*, *y*, and *z*.

| | $z'$ | $z'$ | $z$ | $z$ | |
|---|---|---|---|---|---|
| $x'$ | | | | | $y'$ |
| $x'$ | | | | | $y$ |
| $x$ | | | | | $y$ |
| $x$ | | | | | $y'$ |
| | $w'$ | $w$ | $w$ | $w'$ | |

| | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 00 | 0000 | 0001 | 0011 | 0010 |
| 01 | 0100 | 0101 | 0111 | 0110 |
| 11 | 1100 | 1101 | 1111 | 1110 |
| 10 | 1000 | 1001 | 1011 | 1010 |

## 6.4 Minimal Sums of Products: Karnaugh Map (*K*-map) (cont)

- For a given Boolean expression in disjunctive normal form we write a one in each of the box representing minterms which appear.
- In a *K*-map, two cells are adjacent if their minterms differ in only one variable.
- The required simplification is to group the 1's in adjacent cells and the variable that appears in pair will be eliminated.

# 6.4 Minimal Sums of Products: Karnaugh Map (*K*-map) (cont)

- Steps:

1) Make rectangles around groups of one, two, four or eight 1s.
   - All of the 1s in the map should be included in at least one rectangle.
   - Do not include any of the 0s.

# 6.4 Minimal Sums of Products: Karnaugh Map (*K*-map) (cont)

2) Each group corresponds to one product term. For the simplest result:
   - Make as few rectangles as possible, to minimize the number of products in the final expression.
   - Make each rectangle as large as possible, to minimize the number of literals in each term.
   - It’s all right for rectangles to overlap, if that makes them larger.

# 6.4 Minimal Sums of Products: Karnaugh Map (*K*-map) (cont)

- May have more than one valid solution.

# E.g.3

Simplify the following Boolean expressions using Karnaugh map.

i. $(x \wedge y) \vee (x' \wedge y) \vee (x' \wedge y')$

# E.g.3 (cont)

ii. $(x \wedge y \wedge z) \vee (x' \wedge y \wedge z') \vee (x \wedge y' \wedge z)$

| | $y'$ | $y'$ | $y$ | $y$ |
|---|---|---|---|---|
| $x'$ | | | | |
| $x$ | | | | |
| | $z'$ | $z$ | $z$ | $z'$ |

# E.g.3 (cont)

iii. $$(x \wedge y \wedge z) \vee (x' \wedge y' \wedge z') \vee (x' \wedge y \wedge z) \vee (x \wedge y \wedge z') \vee (x' \wedge y \wedge z')$$

| | $y'$ | $y'$ | $y$ | $y$ |
|---|---|---|---|---|
| $x'$ | | | | |
| $x$ | | | | |
| | $z'$ | $z$ | $z$ | $z'$ |

# E.g.4

Simplify the Boolean expression

$f\ (x, y, z) = [(x \vee y)' \wedge z] \vee (y \vee z)'.$

| $x$ | $y$ | $z$ | $x \vee y$ | $(x \vee y)'$ | $(x \vee y)' \wedge z$ | $y \vee z$ | $(y \vee z)'$ | $f(x, y, z)$ |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | | | | | | |
| 0 | 0 | 1 | | | | | | |
| 0 | 1 | 0 | | | | | | |
| 0 | 1 | 1 | | | | | | |
| 1 | 0 | 0 | | | | | | |
| 1 | 0 | 1 | | | | | | |
| 1 | 1 | 0 | | | | | | |
| 1 | 1 | 1 | | | | | | |

$$f(x, y, z) = (x' \wedge y' \wedge z') \vee (x' \wedge y' \wedge z) \vee (x \wedge y' \wedge z')$$

| | $y'$ | $y'$ | $y$ | $y$ |
|---|---|---|---|---|
| $x'$ | | | | |
| $x$ | | | | |
| | $z'$ | $z$ | $z$ | $z'$ |

# E.g.5

Refer to the given Karnaugh maps, simplify the Boolean expressions.

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

# E.g.5 (cont)

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