CE232 DIGITAL SYSTEM

# Topic 2. Boolean Algebra and Canonical Form

Prepared by Nabila Husna Shabrina

Contact : nabila.husna@umn.ac.id

# Subtopic

2.1 Boolean Algebra

2.2 Sum of Product

2.3 Product of Sum

2.4 Basic Canonical Form

# 2.1 Boolean Algebra

# 2.1 Boolean Algebra

- Set of rules used to simplify the given logic expression without changing its functionality – should be check using truth table
- Used when number of variables are less (1,2,3 variable. More than that, use K-MAP Method)
- “Laws of Boolean” use to both reduce and simplify a complex Boolean expression in an attempt to reduce the number of logic gates required

# 2.1 Boolean Algebra

- The variables used in Boolean Algebra only have one of two possible values, a logic “0” and a logic “1”
- However, expression can have an infinite number of variables all labelled individually

For example, variables A, B, C etc, giving us a logical expression of

$A + B = C$, but each variable can ONLY be a 0 or a 1.

# 2.1 Boolean Algebra

## Truth Table

- Truth table **shows relationship,** in tabular form, between the input values and the result of a specific Boolean operator or function on the input variables
- Some operator → AND, OR, NOT

# 2.1 Boolean Algebra

## AND OPERATOR

- Also known as a Boolean product
- The Boolean expression xy is equivalent to the expression x * y and is read “x and y.”

| Inputs | | Outputs |
|---|---|---|
| $x$ | $y$ | $xy$ |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

# 2.1 Boolean Algebra

## OR OPERATOR

- Often referred to as a Boolean sum
- The expression x+y is read“x or y”

| Inputs | | Outputs |
|---|---|---|
| $x$ | $y$ | $x+y$ |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

# 2.1 Boolean Algebra

## NOT OPERATOR

- Both $\bar{x}$ and $x$ are read as “NOT x
- The rule of precedence for Boolean operators **give NOT top priority**, followed by AND, and then OR

| Inputs | Outputs |
|---|---|
| $x$ | $\bar{x}$ |
| 0 | 1 |
| 1 | 0 |

# 2.1 Boolean Algebra

Example

Truth table for
$F(x, y, z) = x + y'z$

| Inputs | | | | | Outputs |
|---|---|---|---|---|---|
| $x$ | $y$ | $z$ | $\bar{y}$ | $\bar{y}z$ | $x + \bar{y}z = F$ |
| 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 | 0 | 1 |

# 2.1 Boolean Algebra

Boolean Laws/Identities

- To simplified Boolean expression

| Identity Name | AND Form | OR Form |
|---|---|---|
| Identity Law | $1x = x$ | $0 + x = x$ |
| Null (or Dominance) Law | $0x = 0$ | $1 + x = 1$ |
| Idempotent Law | $xx = x$ | $x + x = x$ |
| Inverse Law | $x\bar{x} = 0$ | $x + \bar{x} = 1$ |
| Commutative Law | $xy = yx$ | $x + y = y + x$ |
| Associative Law | $(xy)z = x(yz)$ | $(x+y) + z = x + (y+z)$ |
| Distributive Law | $x + yz = (x+y)(x+z)$ | $x(y+z) = xy + xz$ |
| Absorption Law | $x(x+y) = x$ | $x + xy = x$ |
| DeMorgan's Law | $(\overline{xy}) = \bar{x} + \bar{y}$ | $(\overline{x+y}) = \bar{x}\bar{y}$ |
| Double Complement Law | $\bar{\bar{x}} = x$ | |

# 2.1 Boolean Algebra

- **DeMorgan's law** provides an easy way of **finding the complement of a Boolean function.**

$$\overline{(xy)} = \overline{x} + \overline{y} \quad \text{and} \quad \overline{(x+y)} = \overline{x}\,\overline{y}$$

| $x$ | $y$ | $(xy)$ | $\overline{(xy)}$ | $\overline{x}$ | $\overline{y}$ | $\overline{x}+\overline{y}$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 |

# 2.1 Boolean Algebra

## Boolean Simplification

- There is **no defined set of rules** for using these identities to minimize a Boolean expression: it is simply something tat comes with **experience**
- To prove the equality of two Boolean expressions, you can **also create the truth tables for each and compare**. If the truth tables are **identical**, the expressions **are equal.**

# 2.1 Boolean Algebra

Example.

| Proof | Identity Name |
|---|---|
| $(x+y)(\bar{x}+y) = x\bar{x}+xy+y\bar{x}+yy$ | Distributive Law |
| $= 0+xy+y\bar{x}+yy$ | Inverse Law |
| $= 0+xy+y\bar{x}+y$ | Idempotent Law |
| $= xy+y\bar{x}+y$ | Identity Law |
| $= y(x+\bar{x})+y$ | Distributive Law (and Commutative Law) |
| $= y(1)+y$ | Inverse Law |
| $= y+y$ | Identity Law |
| $= y$ | Idempotent Law |

# 2.1 Boolean Algebra

Complements

F(x, y, z) = x' + yz' and its complement, F'(x, y, z) = x(y' + z)

| $x$ | $y$ | $z$ | $y\bar{z}$ | $\bar{x}+y\bar{z}$ | $\bar{y}+z$ | $x(\bar{y}+z)$ |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 | 1 | 0 |
| 0 | 0 | 1 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 | 0 | 1 | 1 |

# 2.2 Sum of Product

# 2.2 Sum of Product

- SoP is a group of product terms, summed together

Example :

PRODUCT TERMS

AB + AB’C + BC

SUM

Also called disjunctive normal form (DNF)

# 2.3 Product of Sum

# 2.3 Product of Sum

- PoS is group of sum terms multiplied together

SUM TERMS

Example :

$$(A+B)\ .\ (B+C)$$

Multiplication

Also called as Conjunctive Normal Form (CNF)

# 2.4 Basic Canonical Form

# 2.4 Basic Canonical Form

- A Boolean function can be uniquely described by its truth table, or in one of the **canonical forms.**
- Two dual canonical forms of a Boolean function are available:
  - Standard SoP (SSOP) – or Sum of Minterms
  - Standard PoS (SSOP) – or Sum of Maxterms

# 2.4 Basic Canonical Form

SSOP

- Each Product term contains all the variables of the function
- Example :

$f(A,B,C) = A'BC + ABC'$ → SOP and SSOP Form

$f(A,B,C) = AB + BC'A'$ → SOP but not SSOP Form

# 2.4 Basic Canonical Form

SPOS

- Each sum terms contains all the variables of the function
- Example :

$f(A,B,C) = (A + B' + C) . (A' + B' + C')$ → POS and SPOS form

$f(A,B,C) = (A + B) . (A' + B + C')$→ POS but not SPOS Form

# 2.4 Basic Canonical Form

Now your turn !

- $f(A, B) = A \,.\, (A + B')$
- $f(A, B, C, D) = A'B'CD + ABC$
- $f(A, B, C) = ABC' + AB'C + A'B'C'$
- $f(A, B, C) = (A + B' + C') \,.\, (A' + B + C')$

# 2.4 Basic Canonical Form

MINTERMS AND MAXTERMS

- **MINTERMS** is each individual term in SSOP
- **MAXTERMS** is each individual term in SPOS

Example of 2 variables minterms and maxterms

| Variable A and B | Minterms SSOP | Maxterms SPOS |
|---|---|---|
| 0 0 | A’ B’ → $m_0$ | A + B → $M_0$ |
| 0 1 | A’ B → $m_1$ | A + B’ → $M_1$ |
| 1 0 | A B’ → $m_2$ | A’ + B → $M_2$ |
| 1 1 | A B → $m_3$ | A’ + B’ → $M_3$ |

# 2.4 Basic Canonical Form

Index keypoints

- For Minterms:

  “1” → “Not Complemented”

  “0” → “Complemented”

- For Maxterms:

  “0” → “Not Complemented”

  “1” → “Complemented”.

# 2.4 Basic Canonical Form

Example of 3 variable minterms and maxterms

| | | | 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 | $xyz$ | $m_7$ | $x' + y' + z'$ | $M_7$ |

# 2.4 Basic Canonical Form

Example of 4 variables minterms and maxterms

| Index<br>i | Binary<br>Pattern | Minterm<br>$m_i$ | Maxterm<br>$M_i$ |
|---|---|---|---|
| 0 | 0000 | $\bar{a}\bar{b}\bar{c}\bar{d}$ | $a+b+c+d$ |
| 1 | 0001 | $\bar{a}\bar{b}\bar{c}d$ | ? |
| 3 | 0011 | ? | $a+b+\bar{c}+\bar{d}$ |
| 5 | 0101 | $\bar{a}b\bar{c}d$ | $a+\bar{b}+c+\bar{d}$ |
| 7 | 0111 | ? | $a+\bar{b}+\bar{c}+\bar{d}$ |
| 10 | 1010 | $a\bar{b}c\bar{d}$ | $\bar{a}+b+\bar{c}+d$ |
| 13 | 1101 | $ab\bar{c}d$ | ? |
| 15 | 1111 | $abcd$ | $\bar{a}+\bar{b}+\bar{c}+\bar{d}$ |

# 2.4 Basic Canonical Form

Write SSOP using minterms

$$\begin{array}{c} \quad\quad\quad\quad 1\,1 \quad\quad\quad 0\,1 \\ f(A,B) = A\,B + A'B \end{array}$$

$$f(A,B) = \sum m\,(1,3)$$

# 2.4 Basic Canonical Form

Write SSOP using minterms

In general, you can write the equation as

$$y = f(x_{n-1}, \ldots, x_0) = \sum_{\text{for all } j \text{ such that } y_j = 1} m_j$$

# 2.4 Basic Canonical Form

Write SPOS using maxterms

$$f(A,B) = \overset{0\quad 0}{(A + B)} + \overset{1\quad 0}{(A' + B)}$$

$$f(A,B) = \prod M(0,2)$$

# 2.4 Basic Canonical Form

Write SPOS using maxterms

In general, you can write the equation as

$$\boxed{y = f(x_{n-1}, \ldots, x_0) = \prod_{\text{for all } j \text{ such that } y_j=0} M_j}$$

# 2.4 Basic Canonical Form

Convert SOP to SSOP

Steps

- Identify the missing variables in product terms
- Multiply with the missing variables + its complements
- Neglect the repeated terms

# 2.4 Basic Canonical Form

Example

$$f(A,B,C) = AB + AB'C + BC$$

Missing C Missing A

$$= AB\,(C + C') + ABC' + BC(A + A')$$
$$= ABC + ABC' + ABC' + ABC + BCA'$$
$$= ABC + ABC' + A'BC$$
$$= \sum m(7,6,3)$$

# 2.4 Basic Canonical Form

Convert POS to SPOS

Steps

- Identify the missing variables in product terms
- Add with the missing variables and its complements separately
- Neglect the repeated terms

# 2.4 Basic Canonical Form

Example

$$f(A,B,C) = A\,.\,(A+C)$$

Missing BC Missing B

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

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

$$= \prod M(0,1,2,3)$$

# 2.4 Basic Canonical Form

Convert SSOP to SPOS

$$f(A,B,C) = \sum m\,(0,1,3,4,7)$$

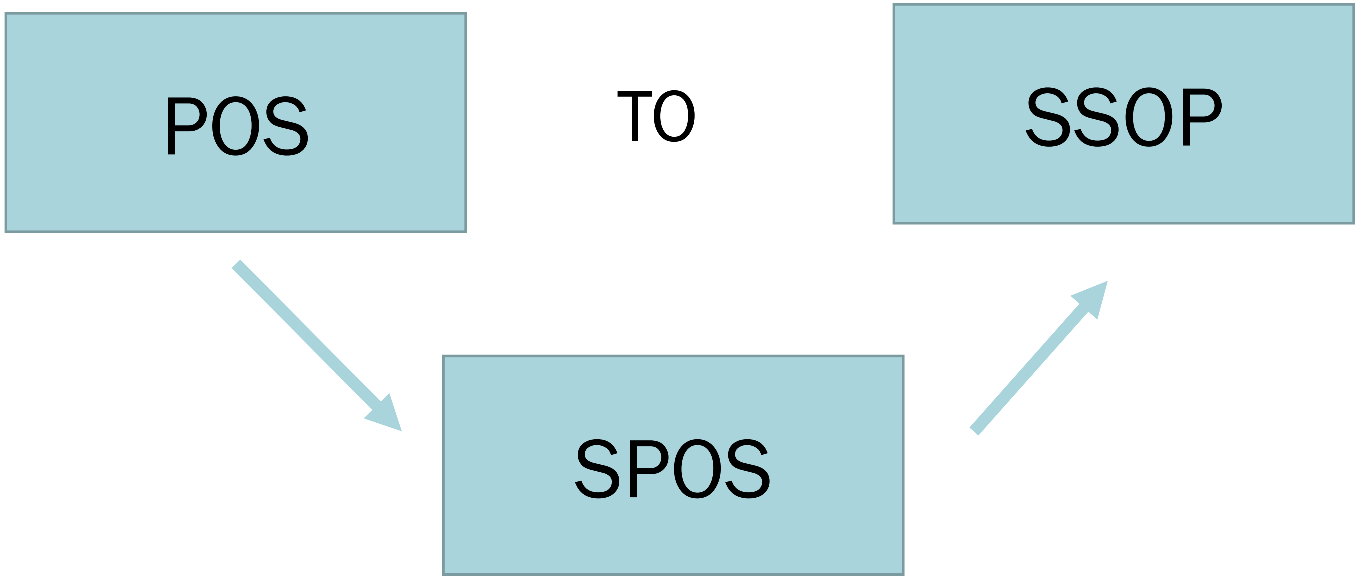
$$\overline{f(A,B,C)} = \prod M(2,5,6)$$

010 101 110

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

# 2.4 Basic Canonical Form

Convert SSOP to SPOS



# 2.4 Basic Canonical Form

Example of POS to SSOP : $f(A,B) = A(A+B)$

In SPOS form $= (A+B).(A+B').(A+B) = (A+B).(A+B')$

$$\underset{00}{(A+B)}.\underset{01}{(A+B')}$$

$$= \prod M(0,1)$$

$$= \sum m(\underset{10}{2},\underset{11}{3})$$

In SSOP form $= (AB') + (AB)$

# References

M. Morris Mano, Digital Design, 5th ed, Prentice Hall, 2012, **Chapter 2**

# Next Topic : Logic Gates