จุดประสงค์การสอน สัปดาห์ที่ 1

1.1 ระบบตัวเลขและรหัส

1.1.1 แปลงเลขฐานต่าง ๆ

1.1.2 อธิบายรหัสในระบบดิจิตอล

1.1.3 คำนวณเลขฐานสอง

1.2 การหาฟังก์ชันเอาต์พุต

1.2.1 ใช้พีชคณิตบูลีน

1.2.2 ใช้ Kanaugh Map

1.2.3 ใช้ Variable-Entering Map

ระบบตัวเลข

ระบบตัวเลข มี 4 ชนิด ฐาน 10 , 2 , 8,16

ฐานสิบ 0,1,2,3,4,5,6,7,8,9

ฐานสอง 0,1

ฐานแปด 0,1,2,3,4,5,6,7

ฐานสิบหก 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

ระบบตัวเลข

การแปลงเลข ฐานสอง เป็น ฐานสิบ

ตัวอย่างที่ 1 **1257 = (1 x 10^3) + (2 x 10^2) + (5 x 10^1) + (7 x 10^0)**

วิธีทำ $\mathbf{N} = d_3R^3 + d_2R^2 + d_1R^1 + d_0R^0$

เมื่อ **R = 10, $\mathbf{d_3}$ = 1, $\mathbf{d_2}$ = 2, $\mathbf{d_1}$ = 5, $\mathbf{d_0}$ = 7**

ตัวอย่างที่ 2 จงแปลงเลขฐานสองต่อไปนี้ให้เป็นเลขฐานสิบ

$(1011.1011)_2 = (……)_{10}$

วิธีทำ

$N=(1x2^3) + (0x2^2) + (1x2^1) + (1x2^0) + (1x2^{-1}) + (0x2^{-2}) + (1x2^{-3}) + (1x2^{-4})$

$$= 8+0+2+1+\frac{1}{2}+0+\frac{1}{2^3}+\frac{1}{2^4}$$

$$N = d_{n-1}R^{n-1} + d_{n-2}R^{n-2} + \ldots + d_1R^1 + d_0R^0 + d_{-1}R^{-1} + d_{-2}R^{-2} + d_{-3}R^{-3} + d_{-m}R^{-m}$$

แปลงเลขฐานต่าง ๆ

เต็ม : การแปลงเลขฐานสิบเป็นฐานสอง

1. ฐานสิบหาร 2 เศษที่ได้คือผลลัพธ์เลขฐานสอง
2. ผลลัพธ์นำไปหารสองต่อไปเรื่อยๆ จนผลลัพธ์เป็นศูนย์ การหารแต่ละครั้งเศษที่ได้จะเป็นผลลัพธ์คือเลขฐานสอง
3. ผลลัพธ์เศษตัวแรกจะเป็น (Least Significant Bit : LSB) และ เศษตัวสุดท้ายจะเป็น (Most Significant Bit : MSB)

แปลงเลขฐานต่าง ๆ

ทศนิยม : การแปลง เลขฐานสิบ เป็น ฐานสอง

1. นำเอาเฉพาะหลังจุดทศนิยมเท่านั้นมาตั้งแล้วคูณด้วย 2
2. ได้ผลลัพธ์เท่าใดให้นำเอาผลลัพธ์หลังจุดทศนิยมขึ้นมาตั้ง แล้วคูณด้วย 2 ทำเช่นนี้ไปเรื่อยๆ
3. ผลลัพธ์ที่ได้เป็นเลขฐานสอง คือนำเอาหน้าจุดทศนิยมเป็นผลลัพธ์

อธิบายรหัสในระบบดิจิตอล

รหัส BCD8421,5421,2421,7536’ , Excess-3, Gray, ASCII

BCD8421 คือ ฐานสิบในรูปเลขฐานสอง

BCD คือBinary Code Decimal

Excess-3 คือ BCD8421 บวกค่าเพิ่ม 3

Gray คือ รหัสที่แปลงจากฐานสองแต่ละลำดับต่างกัน 1 บิท

ASCII คือ รหัสมาตราฐานสหรัฐอเมริกา

โดยเป็นเลขฐานสอง 7 บิท

Table 1.7 Binary coded decimal codes.

Decimal digit	8421 code	5421 code	2421 code	Excess 3 code	2 of 5 code
0	0000	0000	0000	0011	11000
1	0001	0001	0001	0100	10100
2	0010	0010	0010	0101	10010
3	0011	0011	0011	0110	10001
4	0100	0100	0100	0111	01100
5	0101	1000	1011	1000	01010
6	0110	1001	1100	1001	01001
7	0111	1010	1101	1010	00110
8	1000	1011	1110	1011	00101
9	1001	1100	1111	1100	00011
unused	1010	0101	0101	0000	any of
	1011	0110	0110	0001	the 22
	1100	0111	0111	0010	patterns
	1101	1101	1000	1101	with 0, 1,
	1110	1110	1001	1110	3, 4, or 5
	1111	1111	1010	1111	1's

การคำนวณเลขฐานสอง

บวก ลบ คูณ หาร

0 + 0 = 0

0 + 1 = 1

1 + 0 = 1

1 + 1 = 0 ทด 1

0 – 0 = 0

1 – 0 = 1

1 – 1 = 0

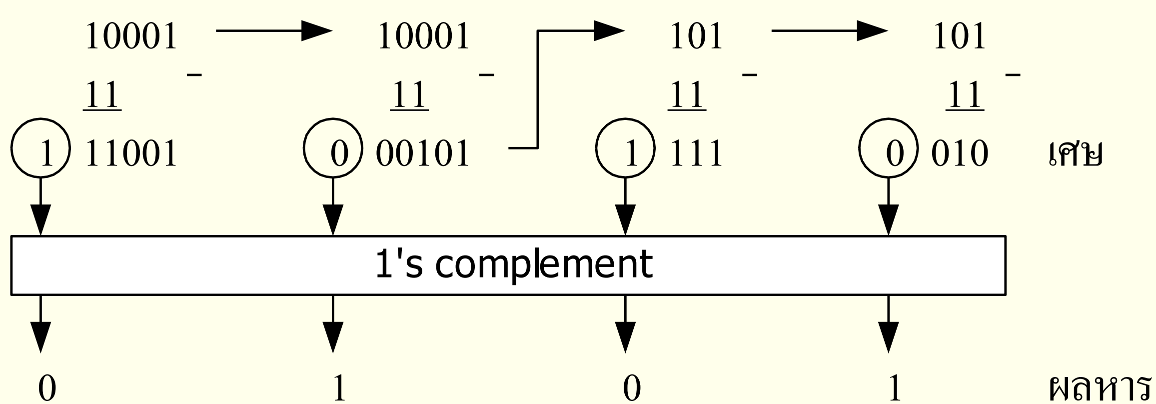
0 – 1 = 1 ต้องยืมจากหลักที่สูงกว่ามา 1

0 x 0 = 0

1 x 0 = 0

0 x 1 = 0

1 x 1 = 1



ลอจิกเกตพื้นฐาน (TTL Logic Circuits)

- **TTL** = Transistor-Transistor Logic
- Logic Gates
  - AND
  - OR
  - NOT (inverter)
  - NAND
  - NOR
  - XOR
  - Equivalence Gate
  - Buffer

AND

- สัญลักษณ์ คือ จุด

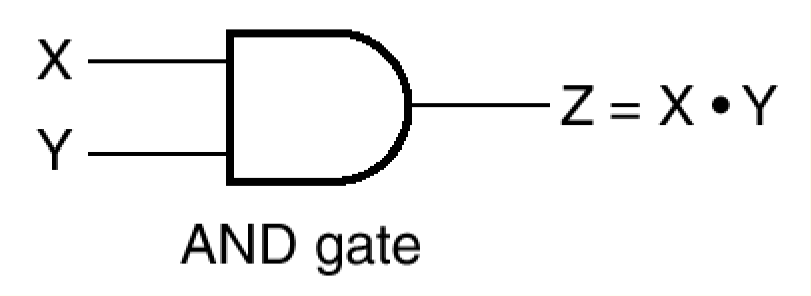
  Z = A · B
- หรือไม่ใส่จุด

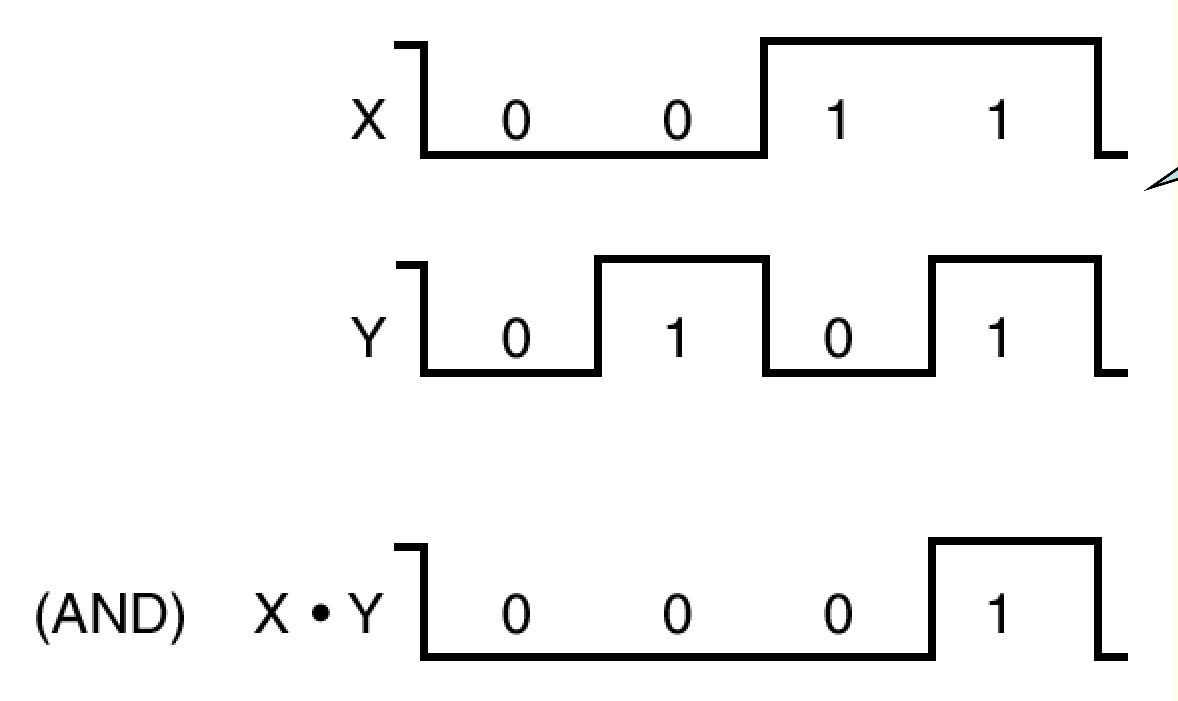
  Z = AB
- Z จะเป็น 1
- เมื่อ A เป็น 1 และ B เป็น 1
- *Truth table: a table of combinations of the binary variables showing the relationship between the values of variables and the result.*

AND

X	Y	Z = X · Y
0	0	0
0	1	0
1	0	0
1	1	1

AND Gate

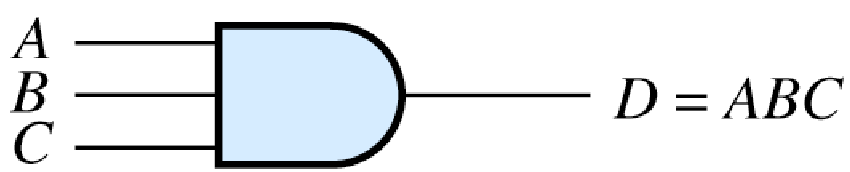




Timing Diagrams

A	B	C	$D = ABC$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

(a) Truth table



(b) Symbol for three-input AND gate

Figure 7.13 Three-input AND gate.

OR

- สัญลักษณ์ คือ บวก

  ไม่ได้มีเป็นหาผลรวม

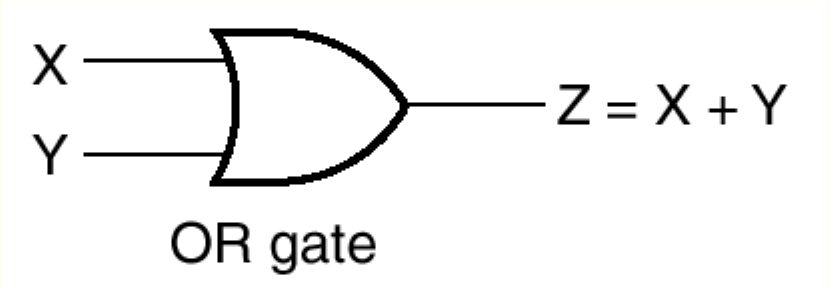
  Z = A + B
- Z เป็น 1 เมื่อ

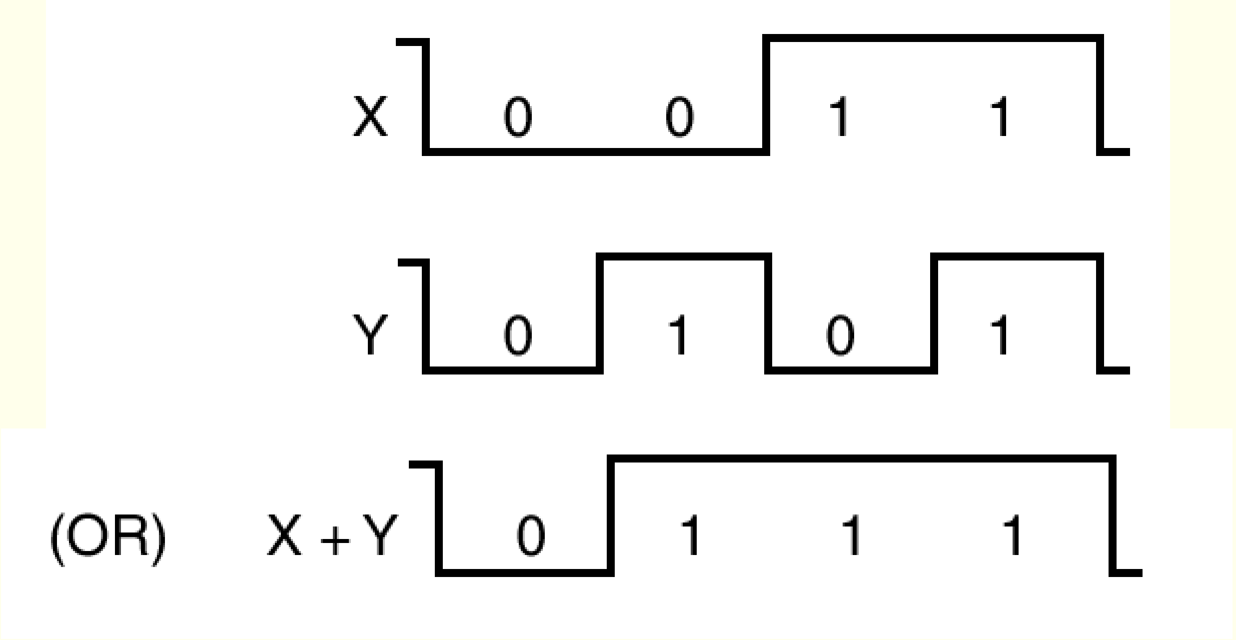
ตัวใดตัวหนึ่งเป็น 1

X	Y	Z = X + Y
0	0	0
0	1	1
1	0	1
1	1	1

OR

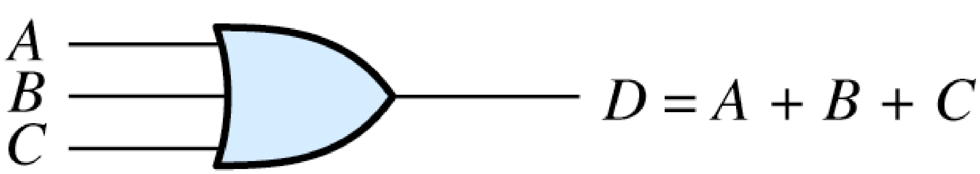
OR Gate





A	B	C	$D = A + B + C$
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

(a) Truth table



(b) Circuit symbol

Figure 7.16 Three-input OR gate.

NOT

- สัญลักษณ์ คือ ขีดบน

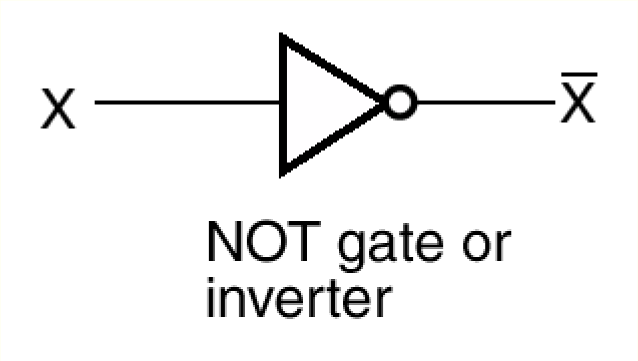
  $Z = \bar{A}$

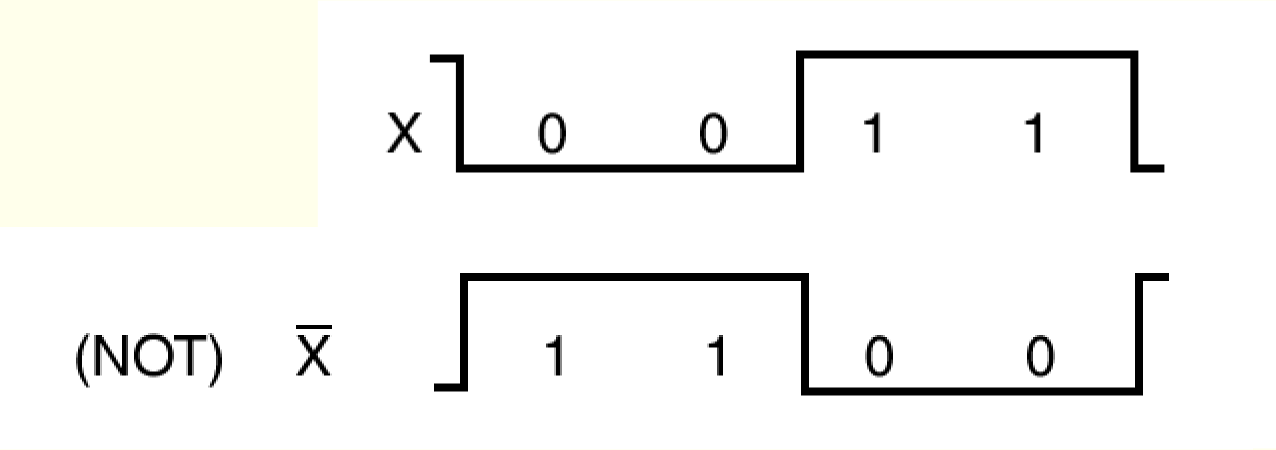
- ค่าตรงข้าม

NOT

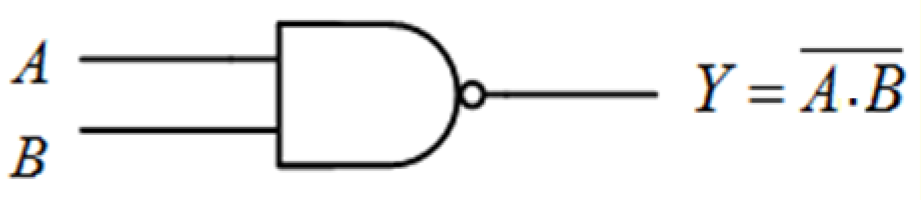
X	$Z = \bar{X}$
0	1
1	0

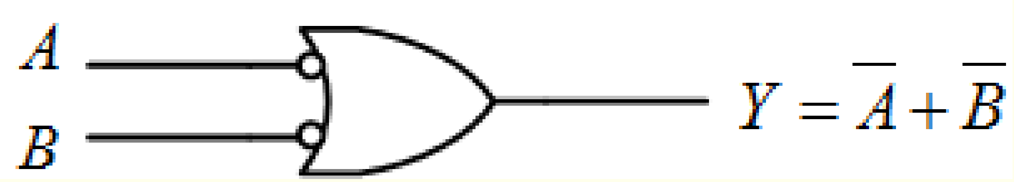
Inverter





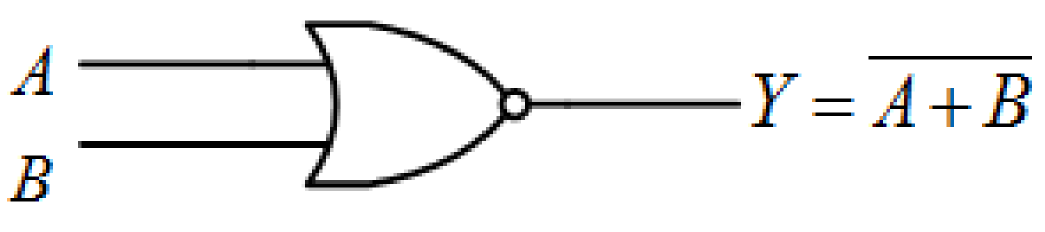
NAND

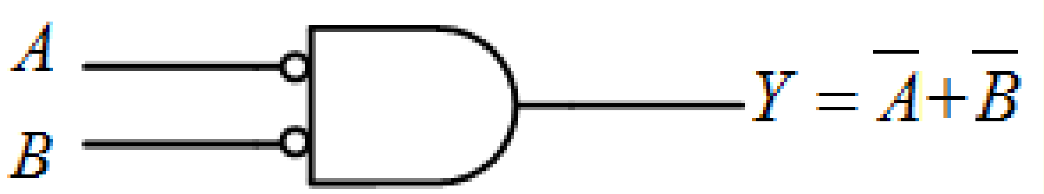




Input		Output
A	B	Y
0	0	1
0	1	1
1	0	1
1	1	0

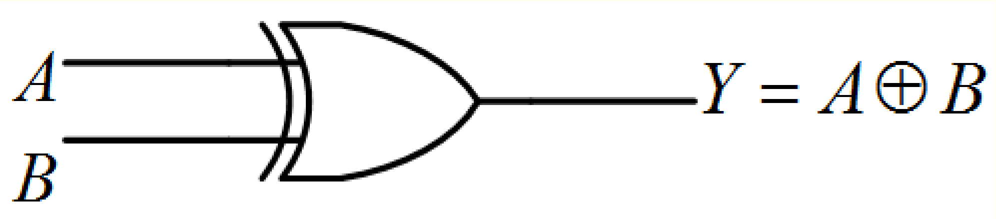
NOR

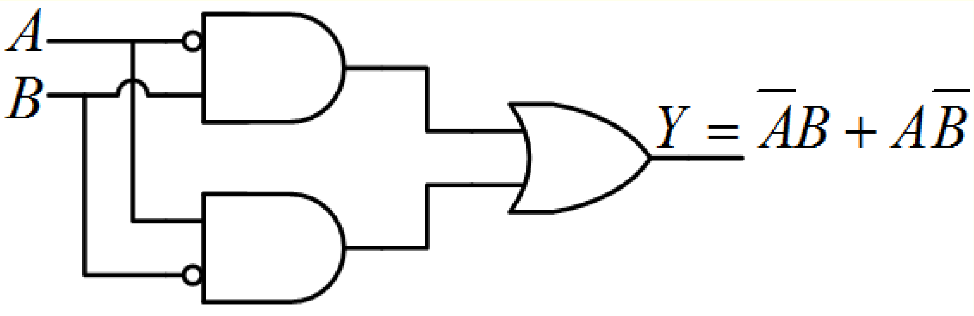




Input		Output
A	B	Y
0	0	1
0	1	0
1	0	0
1	1	0

XOR





Input		Output
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	0

Representation: Schematic

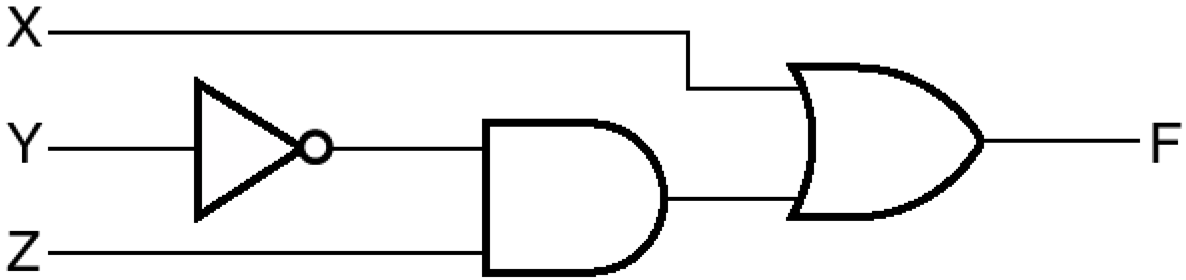


Fig. 2-3 Logic Circuit Diagram for $F = X + \overline{Y}Z$

การลดรูปฟังก์ชัน

1. ใช้ทฤษฎีพีชคณิตบูลีน
2. เขียนตาราง Karnaugh map
3. วิธี Variable entering map
4. หลักการ Quine McCluskey

การลดรูปฟังก์ชัน

1. ใช้ทฤษฎีพีชคณิตบูลีน

นักคณิตศาสตร์ชาวอังกฤษชื่อ George Boole

ได้เป็นผู้กำหนดพีชคณิตบูลีนขึ้น ในพีชคณิตบูลีนเรา

ใช้อักษร A B C … แทนตัวแปรค่า 2 สภาวะ คือ 0 หรือ 1

และใช้เครื่องหมายทางเลขคณิตแทนความสัมพันธ์ระหว่าง

ตัวแปรค่านั้นๆ ได้แก่

เครื่องหมาย . แทน แอนด์ (AND)

เครื่องหมาย + แทนความหมาย ออร์ (OR)

เครื่องหมาย - แทนความหมาย น็อต (NOT)

การเขียนสมการ

Boolean Algebra

- เขียนด้วยตัวแปร และ คำสั่ง
- คำสั่งประกอบด้วย AND, OR, and NOT
- บูลีนฟังก์ชั่น : อธิบายโดยสมการบูลีน
- สมการบูลีน : เกิดจากความสัมพันธ์ระหว่างตัวแปรหลายเทอม

บูลีนฟังก์ชั่น

เทอม

$$F(X,Y,Z) = X + \bar{Y}Z$$

การเขียน :ตารางความจริง

- 2^n แถว

n คือ ตัวแปร

TABLE 2-2
Truth Table
for the Function $F = X + \overline{Y}Z$

X	Y	Z	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

การลดรูปฟังก์ชัน

1. ใช้ทฤษฎีพีชคณิตบูลีน

ทฤษฎีบทที่ 1 กฎการสลับที่ (Commutative law)

$A + B = B + A$

$A . B = B . A$

ทฤษฎีบทที่ 2 กฎการเปลี่ยนกลุ่ม (Associative law)

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

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

ทฤษฎีบทที่ 3 กฎการกระจาย (Distributive law)

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

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

การลดรูปฟังก์ชัน

ทฤษฎีบทที่ 4 กฎการเท่ากัน

$A + A = A$

$A \cdot A = A$

ทฤษฎีบทที่ 5 กฎการกลับค่า (Negation law)

$(\overline{A}) = \overline{A}$

$\overline{\overline{A}} = A$

ทฤษฎีบทที่ 6

$0 + A = A$	$1 \cdot A = A$
$1 + A = 1$	$0 \cdot A = 0$

การลดรูปฟังก์ชัน

ทฤษฎีบทที่ 7

$\overline{A} + A = 1$

$\overline{A} . A = 0$

ทฤษฎีบทที่ 8 กฎการลดทอน (Redundance law)

$A + (A . B) = A$

$A . (A + B) = A$

ทฤษฎีบทที่ 9

$A + (\overline{A} . B) = A + B$

$A . (\overline{A} + B) = A . B$

การลดรูปฟังก์ชัน

ทฤษฎีบทที่ 10 ทฤษฎีของเดอร์มอร์แกน

(De morgan's theorem)

$$\overline{A + B} = \overline{A} \cdot \overline{B}$$

$$\overline{A \cdot B} = \overline{A} + \overline{B}$$

ฟังก์ชั่น

- จากตารางความจริงเดียวกันอาจจะได้ ฟังก์ชั่นต่างกันได้

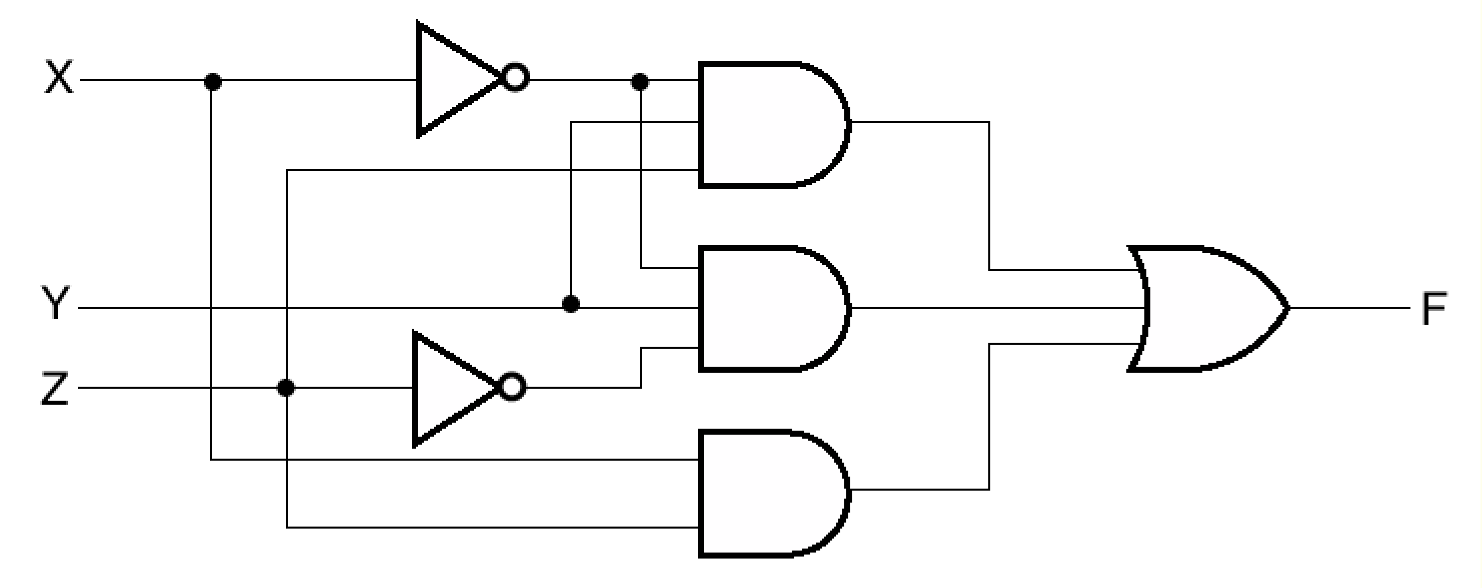
$$F = X + \bar{Y}Z$$

$$F = (X + \bar{Y})(X + Z)$$

- การใช้งานง่าย
  - *ใช้ลดเกต* หรือ *เลือกใช้ชนิดเกต*

ตัวอย่างการเขียน วงจรจากฟังก์ชั่น

$$F = \overline{X}YZ + \overline{X}Y\overline{Z} + XZ$$



(a) $F = \overline{X}YZ + \overline{X}Y\overline{Z} + XZ$

วิธีการเขียนฟังก์ชั่นจากตารางความจริง

- สามารถ เขียนฟังก์ชั่น F จากตารางโดยดูจาก บรรทัดที่มีค่าเป็น 1 “Standard Form” of the function

Truth Table
for the Function $F = X + \overline{Y}Z$

X	Y	Z	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

$$F = \overline{X}\overline{Y}Z + X\overline{Y}\overline{Z} + X\overline{Y}Z + XY\overline{Z} + XYZ$$

Standard Forms

- Definitions:
  - Product terms – AND → $\bar{A}BZ$ หรือ $\bar{A}.B.Z$
  - Sum terms – OR → $X + \bar{A}$

เป็นสัญลักษณ์ทางด้านลอกจิก ไม้มีผลการคำนวณคณิตศาสตร์

แสดงการเขียนมินเทอม : Minterm

- ทำการเขียนโดย Product term ต่อกันทุกตัวแปร
- จะได้ 1 เอาพุต ตรงกับเลขฐานสองในตารางความจริง

X	Y	Z	Product Term	Symbol	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7
0	0	0	$\overline{X}\,\overline{Y}\,\overline{Z}$	m_0	1	0	0	0	0	0	0	0
0	0	1	$\overline{X}\,\overline{Y}Z$	m_1	0	1	0	0	0	0	0	0
0	1	0	$\overline{X}Y\overline{Z}$	m_2	0	0	1	0	0	0	0	0
0	1	1	$\overline{X}YZ$	m_3	0	0	0	1	0	0	0	0
1	0	0	$X\overline{Y}\,\overline{Z}$	m_4	0	0	0	0	1	0	0	0
1	0	1	$X\overline{Y}Z$	m_5	0	0	0	0	0	1	0	0
1	1	0	$XY\overline{Z}$	m_6	0	0	0	0	0	0	1	0
1	1	1	XYZ	m_7	0	0	0	0	0	0	0	1

Number of Minterms

- จะมี *n* *ตัวแปร จะมี* *2^n* *เทอม*
- สัญญาลักณ์แทน minterm 0 คือ m_0

  และ minterm 2^n-1 คือ m_{2^n-1}

Definition: Maxterms

- ทำการ Sum ตัวแปร ที่เอาท์พุตเป็น 0

X	Y	Z	Sum Term	Symbol	M_0	M_1	M_2	M_3	M_4	M_5	M_6	M_7
0	0	0	$X+Y+Z$	M_0	0	1	1	1	1	1	1	1
0	0	1	$X+Y+\overline{Z}$	M_1	1	0	1	1	1	1	1	1
0	1	0	$X+\overline{Y}+Z$	M_2	1	1	0	1	1	1	1	1
0	1	1	$X+\overline{Y}+\overline{Z}$	M_3	1	1	1	0	1	1	1	1
1	0	0	$\overline{X}+Y+Z$	M_4	1	1	1	1	0	1	1	1
1	0	1	$\overline{X}+Y+\overline{Z}$	M_5	1	1	1	1	1	0	1	1
1	1	0	$\overline{X}+\overline{Y}+Z$	M_6	1	1	1	1	1	1	0	1
1	1	1	$\overline{X}+\overline{Y}+\overline{Z}$	M_7	1	1	1	1	1	1	1	0

Minterm related to Maxterm

- Minterm and maxterm with same subscripts are complements

$$\overline{m_j} = Mj$$

- Example

$$\overline{m_3} = \overline{\overline{X}YZ} = X + \overline{Y} + \overline{Z} = M_3$$

Sum of Minterms

- OR ทุกเทอม ที่เอาท์พุตเป็น 1 จากตาราง
- $F = m_0 + m_2 + m_5 + m_7$
- A function that includes all the minterms is equal to logic 1

Ex: G(X,Y,Z)=Σm(0,2,5,7)=1

$$F = \overline{X}\,\overline{Y}\,\overline{Z} + \overline{X}Y\overline{Z} + X\overline{Y}Z + XYZ$$

X	Y	Z	F	$\overline{F}$
0	0	0	1	0
0	0	1	0	1
0	1	0	1	0
0	1	1	0	1
1	0	0	0	1
1	0	1	1	0
1	1	0	0	1
1	1	1	1	0

Product of Maxterms

- Can express F as AND of all Maxterms of rows that should evaluate to 0

$$F = M_1 \bullet M_3 \bullet M_4 \bullet M_6$$

or

$$F = (X + Y + \overline{Z})(X + \overline{Y} + \overline{Z})$$
$$(\overline{X} + Y + Z)(\overline{X} + \overline{Y} + Z)$$

X	Y	Z	F	$\overline{F}$
0	0	0	1	0
0	0	1	0	1
0	1	0	1	0
0	1	1	0	1
1	0	0	0	1
1	0	1	1	0
1	1	0	0	1
1	1	1	1	0

การลดรูปฟังก์ชัน

เขียนตาราง Karnaugh map

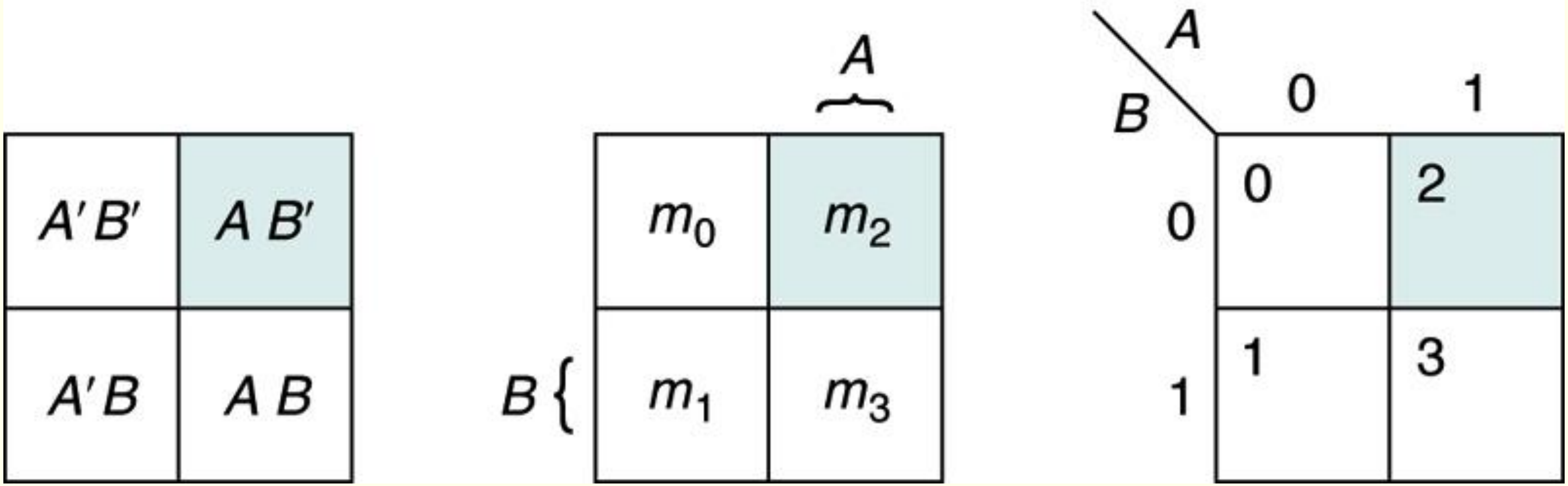
Y

A \ BC	00	01	11	10
0	1	0	1	0
1	1	0	1	1

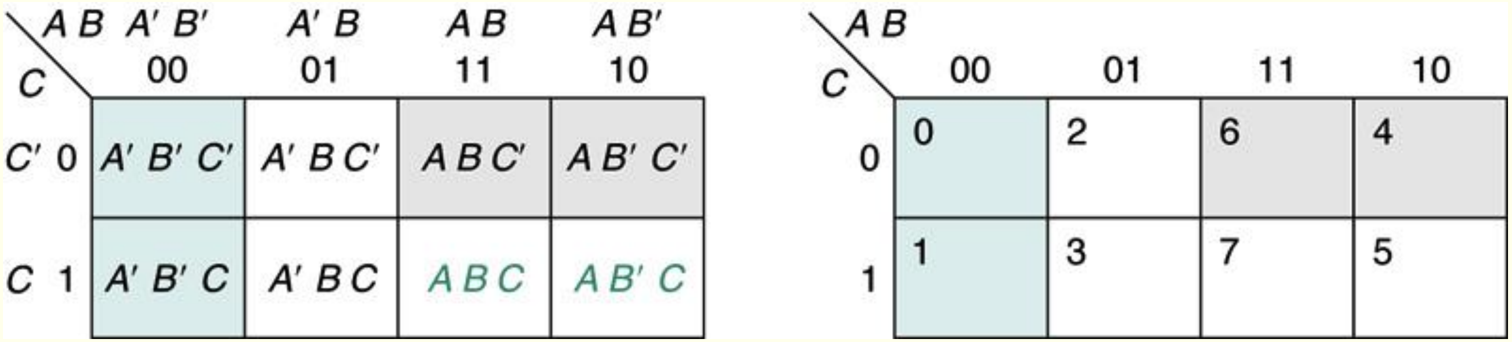
Karnaugh Mapping

1. เขียนตาราง Karnaugh map
2. นำค่า 1’s และ 0’s จากตารางความจริงมาใส่ K-map
3. วงรอบจัดกลุ่ม 1’s. วงรอบได้ตั้งแต่ 2, 4, 8,2^N.
4. เขียนสมการจากการจัดกลุ่มข้อที่ 3

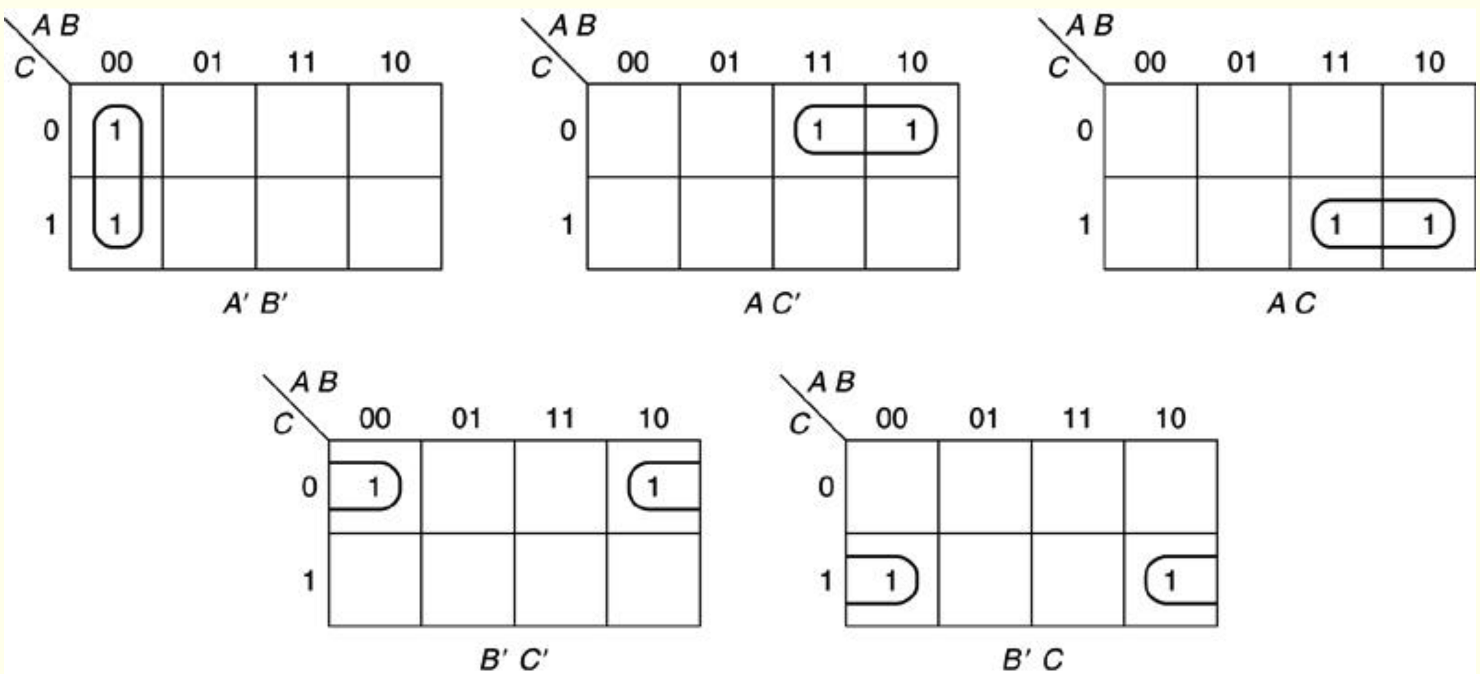
Map 2.1 Two-variable Karnaugh maps.



Map 2.3 Three-variable maps.



Map 2.5 Product terms corresponding to groups of two.



Map 2.8 The four-variable map.

$CD \backslash AB$	00	01	11	10
00	0	4	12	8
01	1	5	13	9
11	3	7	15	11
10	2	6	14	10

$CD \backslash AB$	00	01	11	10
00	$A'B'C'D'$	$A'BC'D'$	$ABC'D'$	$AB'C'D'$
01	$A'B'C'D$	$A'BC'D$	$ABC'D$	$AB'C'D$
11	$A'B'CD$	$A'BCD$	$ABCD$	$AB'CD$
10	$A'B'CD'$	$A'BCD'$	$ABCD'$	$AB'CD'$

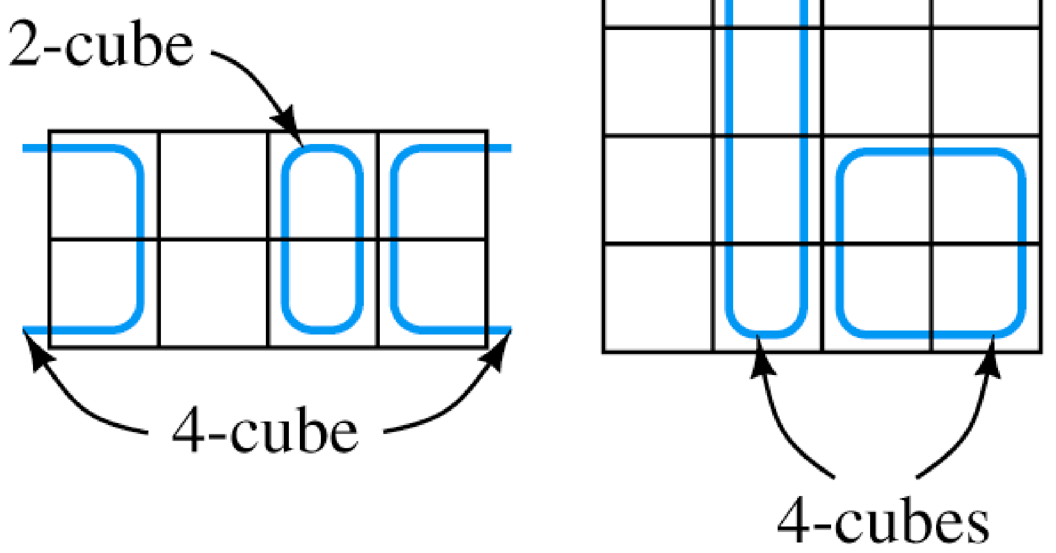


Figure 7.29 Karnaugh maps illustrating cubes.

ตัวอย่างการวงกลมรอมรอบ 1’s

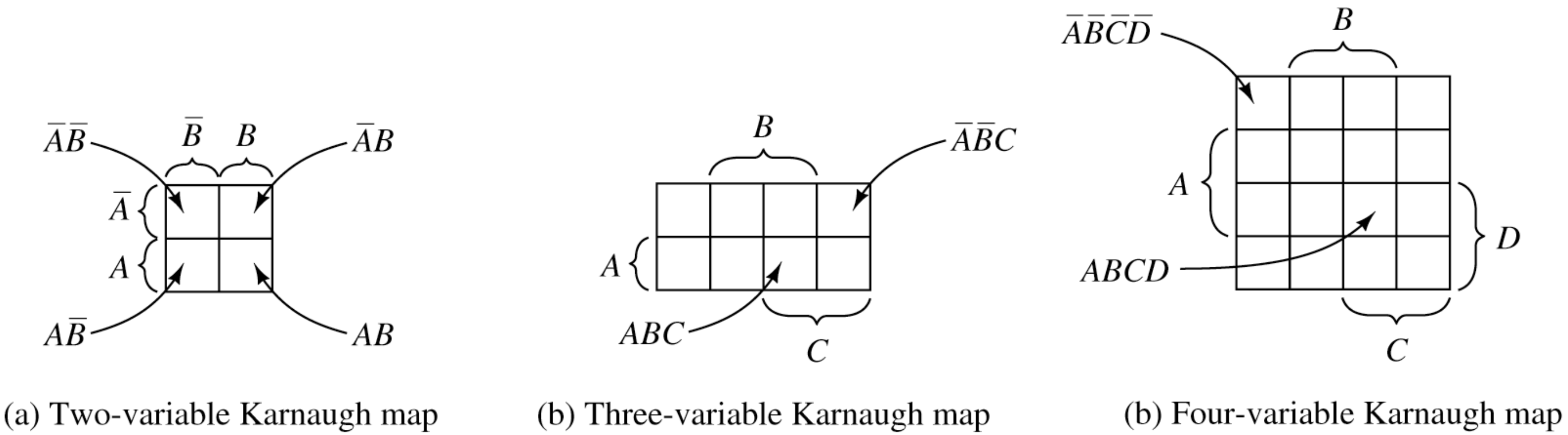


Figure 7.28 Karnaugh maps showing the minterms corresponding to some of the squares.

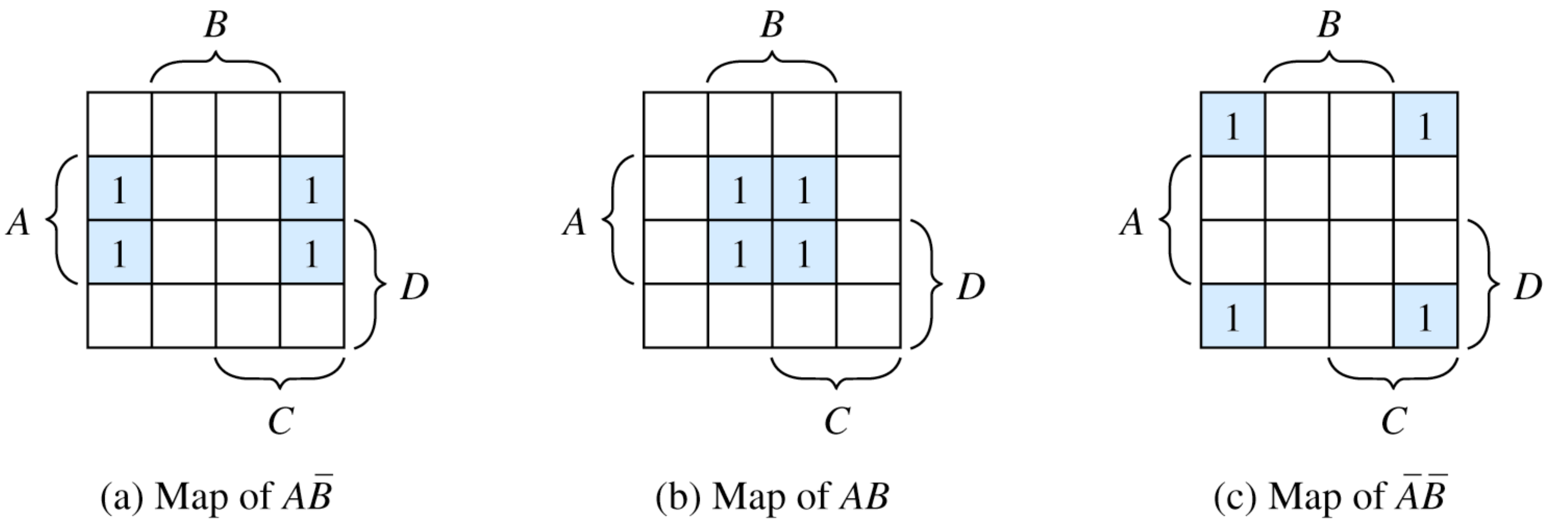


Figure 7.30 Products of two variables map into 4-cubes on a 4-variable Karnaugh map.