3

布林代數的化簡

概　　述

- 在設計數位系統時，將所得之布林代數，依所設計數位電路之實際狀況，進行**適當的化簡** (Minimization) 工作，以**降低布林代數之複雜度**，便可**減少**實現數位電路 (Digital Circuit) 所需之**邏輯閘數量與邏輯閘間連接線** (Interconnections) 之複雜度，如此可進一步降低製作**硬體電路所需之成本**。

- 布林代數之化簡 (Minimization) 是將布林函數 (Boolean Function) 表示式採用某些**簡化準則化簡**後，變成另外一種與**原函式等效**，且**較簡單**之布林函數式來表示。

- 若布林函數表示式以積項之和 (或和項之積) 來表示時，經化簡後所得之布林函數表示式，需同時符合 (1) **積項 (或和項) 之項數 (Number of the Terms) 最少**，(2) **積項 (或和項) 之變數數目 (Number of the Variables) 最少**等兩個條件，即可認定所得之布林函數為最簡表示式。

- 對布林函數之化簡方法 (Minimization Method)，本章將分三個小節，分別來討論 (1) **代數運算法**；(2) **卡諾圖法** (Karnaugh Map Method) 與 (3) **列表法** (Tabulation Method) 等三種常用化簡方法。

布林代數運算化簡法

- 使用代數運算化簡法來化簡布林函數表示式，必須對布林代數之基本定理與假說相當熟悉，即在化簡過程中，必須觀察每個布林代數式之積項與和項之關係，再重複使用各種布林定理與假設後，方可進行化簡之工作。
- 進行代數運算化簡時，沒有任何規則可遵循，而需依靠經驗來進行化簡工作，且所得之結果亦無法保證是最簡之布林函數表示式，故一般皆不建議採用此種方法來進行化簡之工作。

例題 3-1

試利用**代數運算化簡法**來簡化下列布林函數為最簡表示式。

解

1. $f_1(x,y,z)=\bar{x}\cdot y+\bar{x}\cdot y\cdot z=\bar{x}\cdot y\cdot(1+z)=\bar{x}\cdot y\cdot 1=\bar{x}\cdot y$

2. $f_2(x,y,z)=\bar{x}\cdot y+x\cdot y\cdot\bar{z}+x\cdot y=\bar{x}\cdot y+x\cdot y\cdot(1+\bar{z})=\bar{x}\cdot y+x\cdot y=y\cdot(1+\bar{x})=y\cdot 1=y$

3.
$$
\begin{aligned}
f_3(x,y,z)&=\sum(1,3,5,6,7)=\bar{x}\cdot\bar{y}\cdot z+\bar{x}\cdot y\cdot z+x\cdot\bar{y}\cdot z+x\cdot y\cdot\bar{z}+x\cdot y\cdot z\\
&=\bar{x}\cdot\bar{y}\cdot z+\bar{x}\cdot y\cdot z+x\cdot\bar{y}\cdot z+x\cdot y\cdot z+x\cdot y\cdot\bar{z}+x\cdot y\cdot z\\
&=\bar{x}\cdot z\cdot(\bar{y}+y)+x\cdot z\cdot(\bar{y}+y)+x\cdot y\cdot(\bar{z}+z)\\
&=\bar{x}\cdot z+x\cdot z+x\cdot y=(\bar{x}+x)\cdot z+x\cdot y=z+x\cdot y
\end{aligned}
$$

卡諾圖化簡法(概述)

- 將真值表 (True Table) 轉換為相對應圖形 (Graph) 之概念，以進行布林函數之化簡方法，稱為卡諾圖法 (Karnaugh Map Method)。
- 卡諾圖是由許多小方格 (Cell) 所組合成之圖形，而每個小方格代表一個最小項 (Minterms) 或最大項 (Maxterms)，若將這些小方格位置做適當安排後，便可利用位於相鄰方格之幾何位置上的最小項（最大項）之關係，以圖示之方法，輕易來進行化簡工作。
- 具有 n 個變數 (Variables) 的布林函數式，需使用 2^n 方格數目之卡諾圖來表示，故一個完整卡諾圖為 2^n 小方格所排列而成之一個大矩形或正方形圖形。
- 因卡諾圖之方格數與變數數目成指數函數 (Exponential Function) 之正比關係，若欲化簡之布林函數式的變數數目過多時，則卡諾圖之方格數會變成相當多，當進行化簡工作時，會顯得較為複雜，故採用卡諾圖來化簡布林函數式時，一般皆以在 6 個變數以內較為實用。因此對六個變數以上之布林函數，則建議採用適合計算機語言之列表法 (Tabulation Method) 來進行化簡工作。

卡諾圖之表示法

- 在卡諾圖之每個方格中，當其值為 1 時，則此方格稱為 1 方格 (1 Cell)；而其值為 0 時，則此方格稱為 0 方格 (0 Cell)。
- 為了方便利用圖形 (Graph) 的方法來進行布林函數之化簡工作，卡諾圖中之方格排列需遵循一定之規則，即兩相鄰 (Adjacency) 方格間之行 (Column) 與列 (Row)，所對應之二進位組合僅允許有一個位元改變（即相鄰方格所代表之最小項或最大項，僅允許有一個變數為互補之型態）。
- 將卡諾圖之所有方格值設定完成後，若有兩個 1 方格 (0 方格）相鄰時，則表示此兩方格所對應之最小項（最大項）有一個變數是以補數和非補數之型態出現，依布林函數之代數運算法則可知，即可將此變數去掉。若有四個 1 方格 (0 方格）相鄰時，則表示此四方格所對應之最小項（最大項）有兩個變數同時以補數和非補數之型態出現，依布林函數之代數運算法則可知，可將此兩個變數去掉。依此類推，只要有 2^n 方格數相鄰，則可消去 n 個變數。
- 只要圈選相鄰之方格與化簡方法正確，利用卡諾圖來化簡布林函數式，不但簡單、方便，且確定可得到最簡之布林函數表示式。

二變數之卡諾圖表示法

◆ 兩個變數 (分別以 x 與 y 來表示) 所組成之布林函數，必須用 4 $(2^2 = 4)$ 個方格來組成一個大正方形之卡諾圖來表示，而每一個方格分別代表變數 x 與 y 所組成之最小項 (Minterms) 或最大項 (Maxterms)。為了符合兩相鄰 (Adjacency) 方格間，最小項或最大項所對應之二進位數組合，僅允許有一個位元改變之原則，每個方格所表示之布林函數式與二進位數字，如下圖所示。

十進位數	二進位數	最小項	最大項
0	0 0	$\bar{x}\cdot\bar{y}\ (m_0)$	$x+y\ (M_0)$
1	0 1	$\bar{x}\cdot y\ (m_1)$	$x+\bar{y}\ (M_1)$
2	1 0	$x\cdot\bar{y}\ (m_2)$	$\bar{x}+y\ (M_2)$
3	1 1	$x\cdot y\ (m_3)$	$\bar{x}+\bar{y}\ (M_3)$

(a) 真值表

$x \backslash y$	0	1
0	$\overline{x} \cdot \overline{y}$ (m_0)	$\overline{x} \cdot y$ (m_1)
1	$x \cdot \overline{y}$ (m_2)	$x \cdot y$ (m_3)

(b) **最小項**

$x \backslash y$	0	1
0	$x + y$ (M_0)	$x + \overline{y}$ (M_1)
1	$\overline{x} + y$ (M_2)	$\overline{x} + \overline{y}$ (M_3)

(c) **最大項**

$x \backslash y$	0	1
0	0 0 (0)	0 1 (1)
1	1 0 (2)	1 1 (3)

(d) **二進位數**

三變數之卡諾圖表示法

◆ 參個變數（分別以 x、y 與 z 來代表）所組成之布林函數，必須用 8 ($2^3=8$) 個方格來組成一個大矩形之卡諾圖來表示，而每一個方格分別代表變數 x、y 與 z 所組成之最小項 (Minterms) 或最大項 (Maxterms)。為了符合兩相鄰 (Adjacency) 方格間，最小項或最大項所對應之二進位組合，僅允許有一個位元改變之原則，每個方格所表示之布林函數式與二進位數字，如下圖所示。

$x \backslash yz$	0 0	0 1	1 1	1 0
0	$\bar{x}\cdot\bar{y}\cdot\bar{z}$ (m_0)	$\bar{x}\cdot\bar{y}\cdot z$ (m_1)	$\bar{x}\cdot y\cdot z$ (m_3)	$\bar{x}\cdot y\cdot\bar{z}$ (m_2)
1	$x\cdot\bar{y}\cdot\bar{z}$ (m_4)	$x\cdot\bar{y}\cdot z$ (m_5)	$x\cdot y\cdot z$ (m_7)	$x\cdot y\cdot\bar{z}$ (m_6)

(a) 最小項

$x \backslash yz$	0 0	0 1	1 1	1 0
0	$x+y+z$ (M_0)	$x+y+\bar{z}$ (M_1)	$x+\bar{y}+\bar{z}$ (M_3)	$x+\bar{y}+z$ (M_2)
1	$\bar{x}+y+z$ (M_4)	$\bar{x}+y+\bar{z}$ (M_5)	$\bar{x}+\bar{y}+\bar{z}$ (M_7)	$\bar{x}+\bar{y}+z$ (M_6)

(b) 最大項

x \ yz	0 0	0 1	1 1	1 0
0	000 (0)	001 (1)	011 (3)	011 (2)
1	100 (4)	101 (5)	111 (7)	110 (6)

(c) 二進位數字

肆變數卡諾圖表示法

◆ 肆個變數 (分別以 w、x、y 與 z 來代表) 所組成之布林函數，必須用 16 ($2^4=16$) 個方格來組成一個大正方形之卡諾圖，而每一個方格分別代表變數 w、x、y 與 z 所組成之最小項 (Minterms) 或最大項 (Maxterms)。為了符合兩相鄰 (Adjacency) 方格間，最小項或最大項所對應之二進位組合，僅允許有一個位元改變之原則，每個方格所表示之布林函數式與二進位數字，如下圖所示。

$wx \backslash yz$	0 0	0 1	1 1	1 0
0 0	$\overline{w}\cdot\overline{x}\cdot\overline{y}\cdot\overline{z}$ (m_0)	$\overline{w}\cdot\overline{x}\cdot\overline{y}\cdot z$ (m_1)	$\overline{w}\cdot\overline{x}\cdot y\cdot z$ (m_3)	$\overline{w}\cdot\overline{x}\cdot y\cdot\overline{z}$ (m_2)
0 1	$w\cdot\overline{x}\cdot\overline{y}\cdot\overline{z}$ (m_4)	$\overline{w}\cdot x\cdot\overline{y}\cdot z$ (m_5)	$\overline{w}\cdot x\cdot y\cdot z$ (m_7)	$\overline{w}\cdot x\cdot y\cdot\overline{z}$ (m_6)
1 1	$w\cdot x\cdot\overline{y}\cdot\overline{z}$ (m_{12})	$w\cdot x\cdot\overline{y}\cdot z$ (m_{13})	$w\cdot x\cdot y\cdot z$ (m_{15})	$w\cdot x\cdot y\cdot\overline{z}$ (m_{14})
1 0	$w\cdot\overline{x}\cdot\overline{y}\cdot\overline{z}$ (m_8)	$w\cdot\overline{x}\cdot\overline{y}\cdot z$ (m_9)	$w\cdot\overline{x}\cdot y\cdot z$ (m_{11})	$w\cdot\overline{x}\cdot y\cdot\overline{z}$ (m_{10})

(a) 最小項

wx \ yz	00	01	11	10
00	$w+x+y+z$ (M_0)	$w+x+y+\bar{z}$ (M_1)	$w+x+\bar{y}+\bar{z}$ (M_3)	$w+x+\bar{y}+z$ (M_2)
01	$w+\bar{x}+y+z$ (M_4)	$w+\bar{x}+y+\bar{z}$ (M_5)	$w+\bar{x}+\bar{y}+\bar{z}$ (M_7)	$w+\bar{x}+\bar{y}+z$ (M_6)
11	$\bar{w}+\bar{x}+y+z$ (M_{12})	$\bar{w}+\bar{x}+y+\bar{z}$ (M_{13})	$\bar{w}+\bar{x}+\bar{y}+\bar{z}$ (M_{15})	$\bar{w}+\bar{x}+\bar{y}+z$ (M_{14})
10	$\bar{w}+x+y+z$ (M_8)	$\bar{w}+x+y+\bar{z}$ (M_9)	$\bar{w}+x+\bar{y}+\bar{z}$ (M_{11})	$\bar{w}+x+\bar{y}+z$ (M_{10})

(b) 最大項

wx \ yz	00	01	11	10
00	0000 (0)	0001 (1)	0011 (3)	0010 (2)
01	0100 (4)	0101 (5)	0111 (7)	0110 (6)
11	1100 (12)	1101 (13)	1111 (15)	1110 (14)
10	1000 (8)	1001 (9)	1011 (11)	1010 (10)

(c) 二進位數

伍變數卡諾圖表示法

◆ 由五個變數（u、v、w、x 與 y 來代表）所組成之布林代數，故必須用 32（$2^5 = 32$）個方格來組成一個大矩形之卡諾圖來表示，而每一個方格分別代表變數 u、v、w、x 與 y 所組成之最小項 (Minterms) 或最大項 (Maxterms)。為了符合兩相鄰 (Adjacency) 方格間，最小項或最大項所對應之二進位組合，僅允許有一個位元改變之原則，每個方格所表示之布林函數式與二進位數字，如下圖所示。

uv \ wxy	0 0 0	0 0 1	0 1 1	0 1 0	1 1 0	1 1 1	1 0 1	1 0 0
0 0	m_0 (00000)	m_1 (00001)	m_3 (00011)	m_2 (00010)	m_6 (00110)	m_7 (00111)	m_5 (00101)	m_4 (00100)
0 1	m_8 (01000)	m_9 (01001)	m_{11} (01011)	m_{10} (01010)	m_{14} (01110)	m_{15} (00111)	m_{13} (01101)	m_{12} (01100)
1 1	m_{24} (11000)	m_{25} (11001)	m_{27} (11011)	m_{26} (11010)	m_{30} (11110)	m_{31} (11111)	m_{29} (11101)	m_{28} (11100)
1 0	m_{16} (10000)	m_{17} (10001)	m_{19} (10011)	m_{18} (10010)	m_{22} (10110)	m_{23} (10111)	m_{21} (10101)	m_{20} (10100)

布林函數表示為卡諾圖之方法

◆ 利用卡諾圖來化簡布林函數時，首先需將所求之布林函數式展開為「標準積項之和」或「標準和項之積」後，再將布林函數式中之最小項 (Minterms) 或最大項 (Maxterms)，依序填入卡諾圖方格如下：

1、 若布林函數式以「標準積項之和 (SOP)」來表示時，則將所有最小項所對應之方格設定為「1 方格」；而剩下之方格設定為 0 方格。

2、 若布林函數式以「標準和項之積 (POS)」來表示時，則將所有最大項所對應之方格設定為「0 方格」；而剩下之方格設定為 1 方格。

卡諾圖之化簡程序

◆ 當將布林函數式表為卡諾圖後，再依照下列化簡程序，即可得到最簡之布林函數表示式。

1. 圈選所有 2^n 個相鄰 1 方格(求取布林函數最簡式)或 0 方格(求取布林函數最簡式之補數)之組合群(Cluster)。因卡諾圖之排列方式在相鄰之方格間，僅允許有一個變數為互補之型態出現，故圈選 2^n 個相鄰 1 方格(0 方格)，可以消去 n 個變數，而得到一個較所求布林函數簡單之積項(和項)，即每一個圈選可得到一個積項(和項)。
2. 在圈選 2^n 個相鄰 1 方格（0 方格）時，以最多相鄰 1 方格（0 方格）開始圈選為原則，再依次圈選次多相鄰之 1 方格（0 方格），直到將卡諾圖中之所有 1 方格（0 方格）圈選完畢為止。
3. 在圈選相鄰 1 方格（0 方格）時，卡諾圖中之 1 方格（0 方格）可重複使用，但每一個圈選中，至少必須有一個不與其它圈選共用的 1 方格（0 方格）。
4. 將所有圈選所得之積項(和項)用 OR 連接起來，即可得到最簡之布林函數式。

例題 3-2(兩變數卡諾圖之化簡程序)

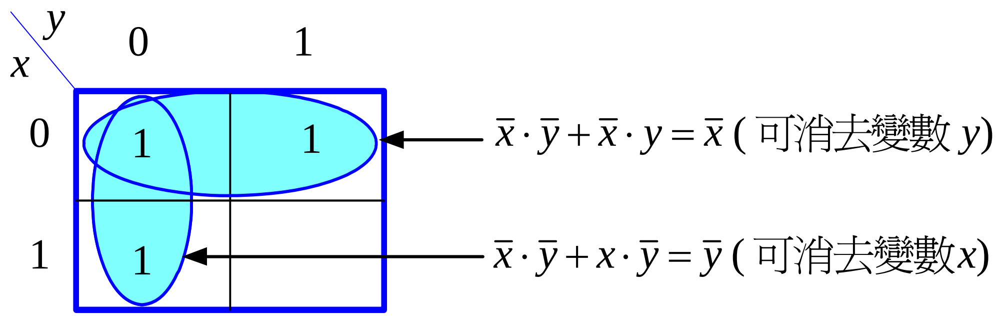
試利用卡諾圖來化簡下列布林函數式。

(a) $f_1(x, y) = \overline{x} + x \cdot \overline{y}$

(b) $f_2(x, y) = \overline{x} \cdot (x + \overline{y})$

解

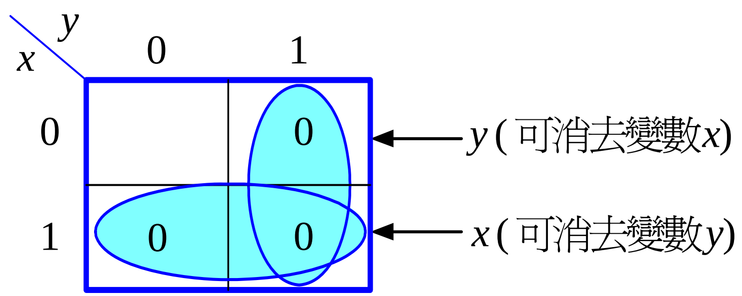
(a) 展開已知布林函數式為標準積項之和，即 $f_1(x, y) = \overline{x} + x \cdot \overline{y} = \overline{x} \cdot \overline{y} + \overline{x} \cdot y + x \cdot \overline{y} = \sum(0, 1, 2)$，接著將上式所有積項所對應之方格設定為 1 方格(其它應標示為 0 方格空白)，如下圖所示。



將兩個圈選中，各消去 1 個變數後，再用 OR 連結起來，即可得化簡後之結果為

$$f_1(x, y) = \overline{x} + x \cdot \overline{y} = \overline{x} + \overline{y}$$

(b) 展開已知布林函數式為標準和項之積，即 $f_2(x,y)=\bar{x}\cdot(x+\bar{y})=(x+\bar{y})\cdot(\bar{x}+y)\cdot(\bar{x}+\bar{y})=\prod(1,2,3)$，再將所有和項所對應之方格設定為 0 方格(其它應標示為 1 方格空白)，如下圖所示。



將兩個圈選中，各消去 1 個變數後，再用 OR 連結起來，即可得化簡後之結果為 $\overline{f_2(x,\,y)}=x+y$。接著將 $f_2(x,\,y)$ 取反相，即

$$f_2(x,\,y)=\overline{\overline{f_2(x,\,y)}}=\overline{x+y}=\bar{x}\cdot\bar{y}。$$

例題 3-3(三變數卡諾圖之化簡程序)

試利用卡諾圖來化簡下列布林函數式。

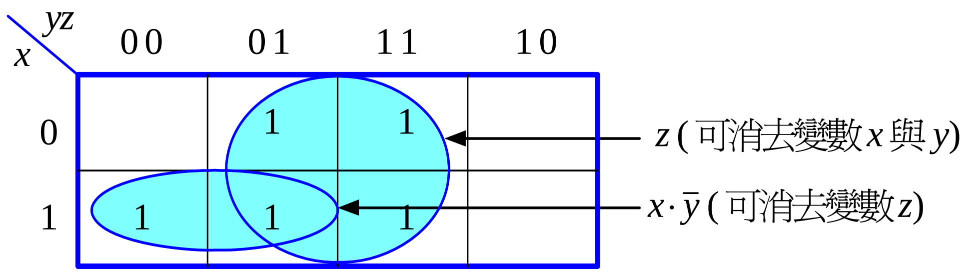
(a) $f_1(x, y, z) = \overline{x} \cdot z + x \cdot \overline{y} + x \cdot y \cdot z$

(b) $f_2(x, y, z) = (\overline{x} + \overline{z}) \cdot (x + \overline{y} + z) \cdot (x + y + z)$

解

(a) $f_1(x, y, z) = \overline{x} \cdot z + x \cdot \overline{y} + x \cdot y \cdot z = \overline{x} \cdot \overline{y} \cdot z + \overline{x} \cdot y \cdot z + x \cdot \overline{y} \cdot \overline{z} + x \cdot \overline{y} \cdot z + x \cdot y \cdot z = \sum(1, 3, 4, 5, 7)$ ，

接著將所有積項所對應之方格設定為 1 方格，如下圖所示。

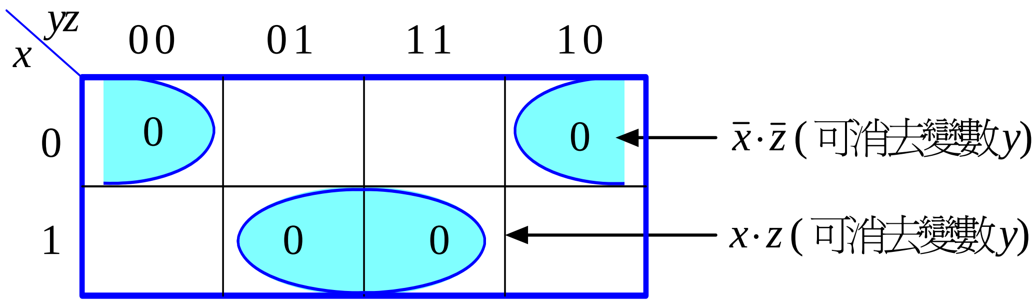


將兩個圈選中，分別消去 1 與 2 個變數後，再用 OR 連結起來，即可得化簡後之結果為

$$f_1(x, y, z) = z + x \cdot \overline{y}$$

(b) $f_2(x, y, z) = (\bar{x}+\bar{z})\cdot(x+\bar{y}+z)\cdot(x+y+z) = (x+y+z)\cdot(x+\bar{y}+z)\cdot(\bar{x}+y+\bar{z})\cdot(\bar{x}+\bar{y}+\bar{z}) = \prod(0, 2, 5, 7)$，

再將所有和項所對應之方格設定為 0 方格，如下圖所示。



將兩個圈選中，各消去 1 個變數後，再用 OR 連結起來，即可得化簡後之結果為 $\overline{f_2(x, y, z)} = \bar{x}\cdot\bar{z}+x\cdot z$ **。接著將** $f_2(x, y, z)$ **取反相，即**

$$f_2(x, y, z) = \overline{\overline{f_2(x, y, z)}} = \overline{\bar{x}\cdot\bar{z}+x\cdot z} = (x+z)\cdot(\bar{x}+\bar{z})$$ 。

例題 3-4(四變數卡諾圖之化簡程序)

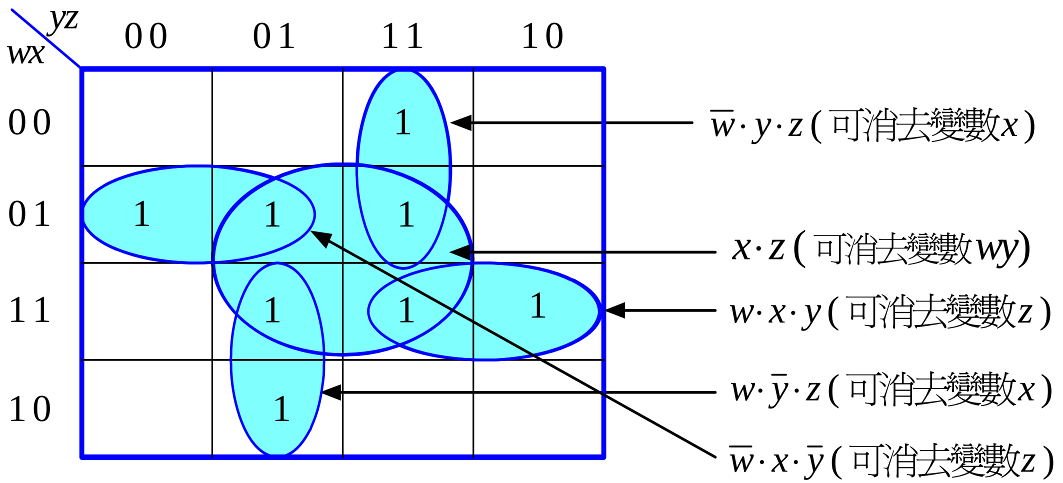
試利用卡諾圖來化簡下列布林函數式。

(a) $f_1(w, x, y, z) = \overline{w}\cdot\overline{x}\cdot y\cdot z + \overline{w}\cdot x\cdot\overline{y}\cdot\overline{z} + \overline{w}\cdot x\cdot\overline{y}\cdot z + \overline{w}\cdot x\cdot y\cdot z + w\cdot x\cdot\overline{y}\cdot z + w\cdot x\cdot y\cdot z + w\cdot x\cdot y\cdot\overline{z} + w\cdot\overline{x}\cdot\overline{y}\cdot z$

(b) $\boldsymbol{f_2(w, x, y, z) = (w + \overline{x} + \overline{y})\cdot(x + \overline{y} + z)\cdot(\overline{w} + \overline{x} + \overline{y})}$

解

(a) $\boldsymbol{f_1(w, x, y, z)} = \sum(3,4,5,7,9,13,14,15)$

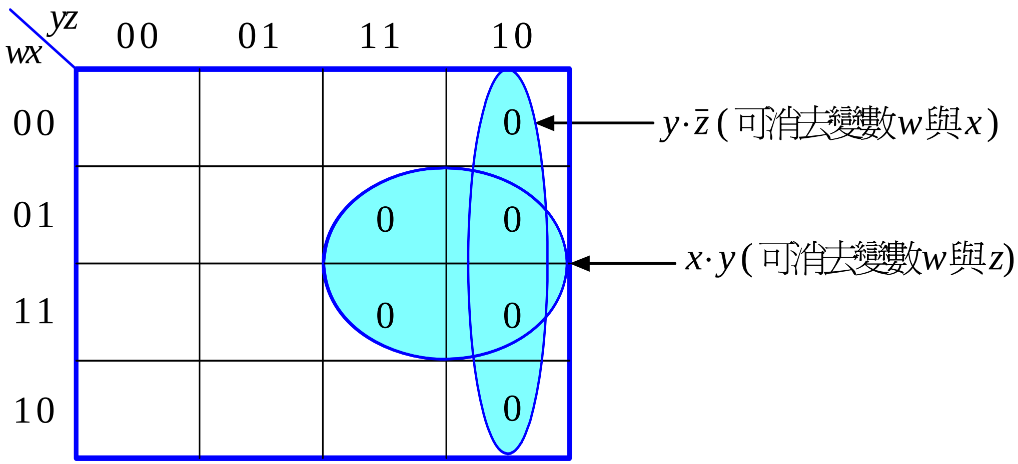


將 5 個圈選中，分別消去 1 個與 2 個變數後，再用 OR 連結起來，即可得化簡後之結果為

$$\boldsymbol{f_1(w, x, y, z) = \overline{w}\cdot x\cdot\overline{y} + \overline{w}\cdot y\cdot z + w\cdot x\cdot y + w\cdot\overline{y}\cdot z}$$

註：4 個相鄰之 1 方格之圈選，並沒有一個獨立的 1 方格，此圈選是屬於多餘的，故應予刪除。

(b) $f_2(w, x, y, z) = (w + \overline{x} + \overline{y}) \cdot (x + \overline{y} + z) \cdot (\overline{w} + \overline{x} + \overline{y}) = \prod(2, 6, 7, 10, 14, 15)$



將兩個圈選中，各消去 2 個變數後，再用 OR 連結起來，即可得化簡後之結果為

$\overline{f_2(w, x, y, z)} = y \cdot \bar{z} + x \cdot y$ **，故可得**

$$f_2(w, x, y, z) = \overline{\overline{f_2(w, x, y, z)}} = \overline{y \cdot \bar{z} + x \cdot y} = (\bar{y} + z) \cdot (\bar{x} + \bar{y})$$

五變數卡諾圖之另一種表示方法

- 伍變數之卡諾圖有 32 個方格，因所能圈選之最多方格數為 32 個，若將 32 個方格一起進行化簡，則會因方格數目太多，而造成一些難以避免之疏忽。
- 若將 32 格方格分成兩組 16 方格之卡諾圖後，再來進行化簡工作，以減少一些不必要的錯誤發生，而更改後之伍變數卡諾圖，每個方格所代表之最小項如下圖所示。

$vw \backslash xy$	0 0	0 1	1 1	1 0
0 0	m_0	m_1	m_3	m_2
0 1	m_4	m_5	m_7	m_6
1 1	m_{12}	m_{13}	m_{15}	m_{14}
1 0	m_8	m_9	m_{11}	m_{10}

$u = 0\ (\bar{u})$

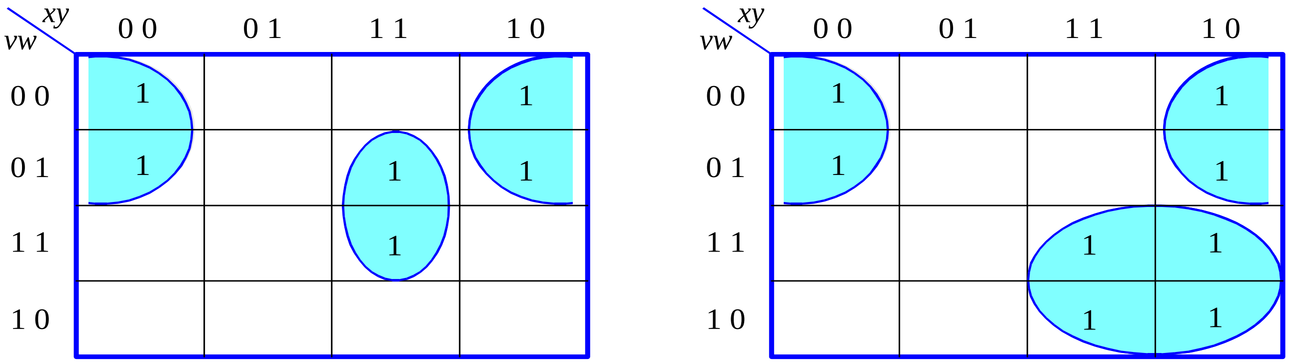
$vw \backslash xy$	0 0	0 1	1 1	1 0
0 0	m_{16}	m_{17}	m_{19}	m_{18}
0 1	m_{20}	m_{21}	m_{23}	m_{22}
1 1	m_{28}	m_{29}	m_{31}	m_{30}
1 0	m_{24}	m_{25}	m_{27}	m_{26}

$u = 1\ (u)$

例題 3-5(五變數卡諾圖之化簡程序)

利用**卡諾圖**來化簡 $f(u, v, w, x, y)=\sum(0, 2, 4, 6, 7, 15, 16, 18, 20, 22, 26, 27, 30, 31)$ 之布林函數式。

解



$\bar{u}$　　　　u

$$f_{\bar{u}}(u, v, w, x, y)=\bar{u}\cdot\bar{v}\cdot\bar{y}+\bar{u}\cdot w\cdot x\cdot y$$

$$f_{u}(u, v, w, x, y)=u\cdot\bar{v}\cdot\bar{y}+u\cdot v\cdot x$$

$$\begin{aligned} f(u, v, w, x, y) &= f_{\bar{u}}(u, v, w, x, y)+f_{u}(u, v, w, x, y) \\ &= \bar{u}\cdot\bar{v}\cdot\bar{y}+\bar{u}\cdot w\cdot x\cdot y+u\cdot\bar{v}\cdot\bar{y}+u\cdot v\cdot x \\ &= \bar{v}\cdot\bar{y}\cdot(\bar{u}+u)+\bar{u}\cdot w\cdot x\cdot y+u\cdot v\cdot x \\ &= \bar{v}\cdot\bar{y}+\bar{u}\cdot w\cdot x\cdot y+u\cdot v\cdot x \end{aligned}$$

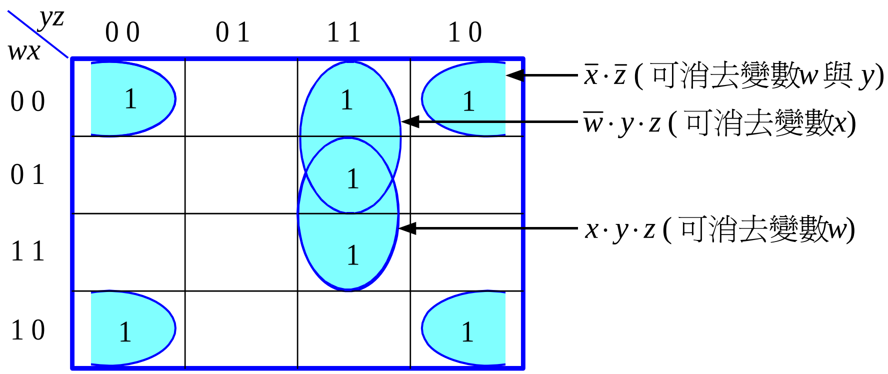
積項之和(SOP)與和項之積(POS)布林函數式化簡之比較

◆ 使用**積項之和 (SOP)** 與**和項之積 (POS)** 等兩種形式，均可表示**相同功能**之布林函數式，因此使用卡諾圖來化簡時，採用**任何一種形式所得之結果應會完全相等**，當利用卡諾圖來化簡布林函數時，需檢視使用 **1 方格**或 **0 方格**來化簡，所得之布林函數式**較為簡單**，以確保所得之結果為**最簡的布林函數表示式**，以節省**實現硬體電路所需之成本**。

例題 3-6：請利用卡諾圖來化簡 $f(w,x,y,z)=\sum(0,2,3,7,8,10,15)$ 之布林函數式，並指出採用 SOP 與 POS 表示法來進行化簡工作後，所得之布林函數式來**實現數位電路**時，**何者較為經濟**？

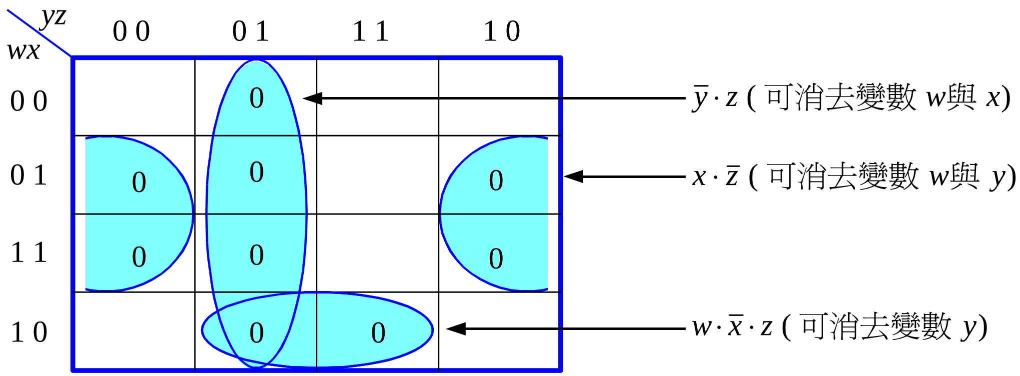
解

(1)、 $f(w,x,y,z)=\sum(0,2,3,7,8,10,15)$



$$f(w,x,y,z)=\bar{x}\cdot\bar{z}+\bar{w}\cdot y\cdot z+x\cdot y\cdot z$$

(2)、 $\overline{f(w, x, y, z)} = \prod(1, 4, 5, 6, 9, 11, 12, 13, 14)$



$$\overline{f(w, x, y, z)} = \bar{y} \cdot z + x \cdot \bar{z} + w \cdot \bar{x} \cdot z$$

$$\begin{aligned} f(w, x, y, z) &= \overline{\overline{f(w, x, y, z)}} = \overline{\bar{y} \cdot z + x \cdot \bar{z} + w \cdot \bar{x} \cdot z} \\ &= (y + \bar{z}) \cdot (\bar{x} + z) \cdot (\bar{w} + x + \bar{z}) \end{aligned}$$

比較(1) 部分與(2)部分化簡所得之布林函數可知，POS 比 SOP 少一個邏輯閘的輸入，故使用和項之積 (POS) 來化簡較積項之和 (SOP)經濟。

不完全指定函數

- 在設計數位電路時，有些輸入變數之最小項 (最大項)，會因某些原因可能不會出現或可以任意被指定為邏輯 1 或邏輯 0，亦即這些最小項 (最大項) 可存在，亦可不存在，這種函數被稱為不完全指定函數 (Incompletely Specified Function)。
- 對這些不完全指定函數是否有出現，皆不會影響布林函數輸出的結果，因此又可稱為不在意項 (Don’t Care Terms)。
- 對這不在意項之函數值，在卡諾圖中可以用利用「×」來標示。而對標示為「×」之方格，在利用卡諾圖來化簡布林函數時，若有利於形成更多相鄰之 1 方格 (0 方格)，以消去更多之變數時，就使用此方格來幫助得到更簡之布林函數式；若無發法幫助形成更多相鄰之 1 方格 (0 方格) 時，則在進行化簡時，就可將標示為「×」之方格視同不存在。

例題 3-7

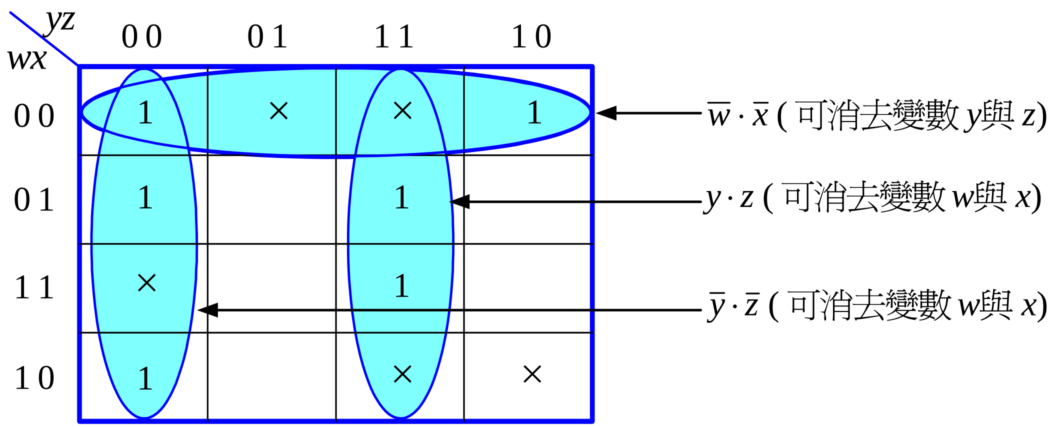
試利用卡諾圖來化簡下列布林函數式。

(a) $f_1(w, x, y, z)=\sum(0, 2, 4, 7, 8, 15)+\sum_{\times}(1, 3, 10, 11, 12)$

(b) $f_2(w, x, y, z)=\prod(1, 2, 3, 4, 6, 12)+\prod_{\times}(0, 9, 10, 14)$

解

(a) $f_1(w, x, y, z)=\sum(0, 2, 4, 7, 8, 15)+\sum_{\times}(1, 3, 10, 11, 12)$

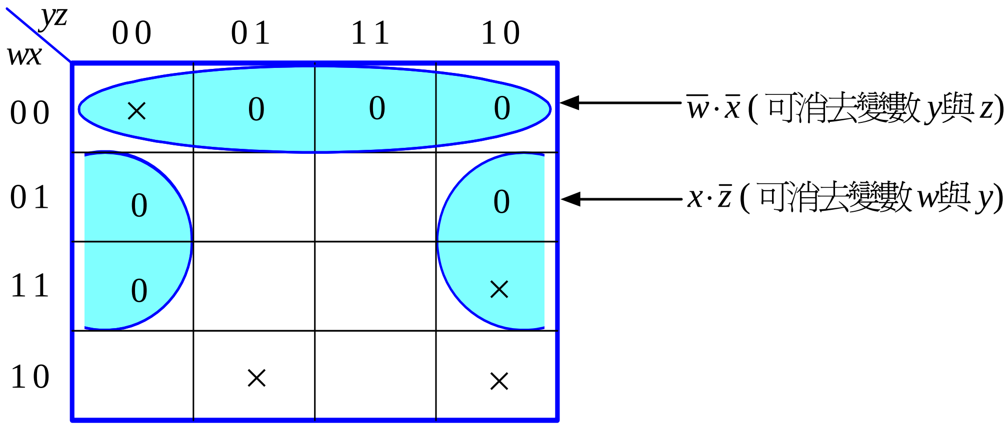


將此三組 4 個相鄰之圈選，分別消去 2 個變數後，再用 OR 連結起來，即可得化簡後之結果為

$$f_1(w, x, y, z)=\overline{w}\cdot\overline{x}+y\cdot z+\overline{y}\cdot\overline{z}$$

註：標記為「×」的「1010」方格未圈選，因此「×」方格無法幫助得到更簡之結果，故可將此方格視為 0 方格。

(b) $f_2(w,\ x,\ y,\ z) = \prod(1,2,3,4,6,12) + \prod_{\times}(0,9,10,14)$



將 2 個圈選中，各消去 2 個變數後，再用 OR 連結起來，即可得化簡後之結果為

$$f_2(w,\ x,\ y,\ z) = \overline{\overline{f_2(w,\ x,\ y,\ z)}} = \overline{\overline{w}\cdot\overline{x} + x\cdot\overline{z}} = (w + x)\cdot(\overline{x} + z)$$

註：標記為「×」方格，分別為「1001」與「1010」方格未被圈選，因這些方格無法幫助得到更簡之結果，故可將此方格視為 1 方格。

變數引入圖法(Variable Enter Map Method)

◆ 藉由一些運算程序後，可適當降低變數數目，以降低使用卡諾圖化簡布林函數之複雜度，即可用較少方格數之卡諾圖來化簡較多變數之布林函數式，稱為變數引入圖法。

◆ 接著以積項之和 (SOP) 表示式為例，說明使用變數引入圖法來對 n 個變數之布林函數進行化簡步驟如下：

1. 使用餘式圖法 (Residue Map Method) 選取一些適當之變數當作外部變數 (External Variable)，若選取 m 個變數當外部變數，接著將剩下 $n-m$ 個變數製作一個餘式圖，以將原來需 2^n 個方格之卡諾圖降低為 2^{n-m} 個方格之變數引入圖。
2. 將所得之變數引入圖，首先將設定為外部變數名稱之方格，視為 0 方格後，再由所得之新卡諾圖，求出最簡之布林函數式。
3. 再將步驟 (2) 所得變數引入圖中選取一個外部變數 (注意：變數 x 與 $\overline{x}$ 應視為不同之外部變數) 設定為 1 方格，且將其它之外部變數方格設為 0 方格，且將 1 方格視為不在意項 (Don’t Care Term)，再由所得之新卡諾圖，求出最簡單之布林函數式。
4. 重複步驟 (3) 之程序，直到將所有外部變數逐一處理完畢為止，最後再將上述步驟所得之最簡布林函數式，用 OR 連結起來，即可得到所求之最簡布林函數表示式。

餘 式 圖

◆ 對一個 n 個變數布林函數 $f(x_{n-1}, x_n, ..., x_1, x_0)$ 而言，此布林函數最多可用 2^n 個最小項 (Minterms) 來表示，即 $f(x_{n-1}, x_n, ..., x_1, x_0) = \sum_{j=0}^{2^n-1} C_j \cdot m_j$ 。

其中 m_j 為 2^n 個最小項中之第 j 個，而 C_j 為 0 或 1 之常數，即當 m_j 存在所求之布林函數時，則 C_j 值為 1，否則 C_j 值為 0。

◆ 當選取 m 個變數當作變數引入圖法之外部變數 (External variable) 時，則表示原布林函數最小項減少為 2^{n-m} 個，故可用 2^{n-m} 個方格之卡諾圖來化簡所求之布林函數。而選取餘式之方法，將 m 個與 $n-m$ 個變數以二進位循環碼之方式，繪出如下圖所示之組合圖，稱為二進位組合餘式圖 (Residue Map)。

$x_{n-1} \cdots x_{n-m}$ \ $x_{n-m-1} \cdots x_0$	$0\cdots00$	$0\cdots01$	$\cdots$	$1\cdots10$	$1\cdots11$
$0\cdots00$	0	1	$\cdots$	$n-m-1$	$n-m$
$0\cdots01$					
$\vdots$		$\vdots$		$\vdots$	
$1\cdots10$			$\cdots$		
$1\cdots11$	2^n-n+1	2^n-n+2		2^n-2	2^n-1

例題 3-8 (變數引入圖化簡法)

試利用變數引入圖法來化簡 $f(w,x,y,z)=w\cdot x\cdot y\cdot\bar{z}+w\cdot\bar{x}\cdot y\cdot\bar{z}+x\cdot\bar{y}\cdot\bar{z}+\bar{x}\cdot\bar{y}\cdot\bar{z}$ 之布林函數式。

解

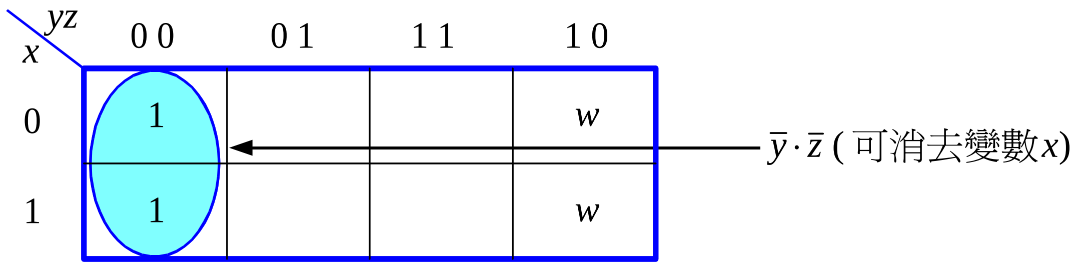
展開已知之布林函數式為標準積項之和

$$f(w,x,y,z)=w\cdot x\cdot y\cdot\bar{z}+w\cdot\bar{x}\cdot y\cdot\bar{z}+x\cdot\bar{y}\cdot\bar{z}+\bar{x}\cdot\bar{y}\cdot\bar{z}=\sum(0,4,8,10,12,14)\text{。}$$

(1)、選取 w 作外部變數，並繪出選取 w 當作外部變數之餘數圖如下圖所示，最後在餘式圖上求解每一行之最簡式。

w \ xyz	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
0	0	1	2	3	4	5	6	7
1	8	9	10	11	12	13	14	15
	1	0	w	0	1	0	w	0

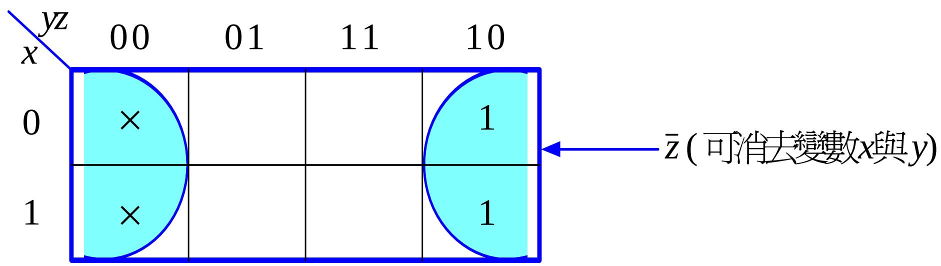
(2)、將每一行化簡所得之結果，以設定變數 x、y 與 z 所對應之變數引入圖之方格值，如下圖所示。



設定為外部變數 w 之方格視為 0 方格，再使用卡諾圖化簡法，可得化簡後之布林函數

$$f'(w, x, y, z) = \bar{y} \cdot \bar{z}$$

(3)、將外部變數 w 之方格設定為 1 方格，而將原有之 1 方格設定為不在意項後，如此又可得到新的卡諾圖，如下圖所示。再使用卡諾圖之化簡法，可得化簡後之布林函數表示式如下



$$f''(w, x, y, z) = \bar{z}$$

將上述步驟所得之最簡布林函數式，用 OR 連結起來(由外部變數 w 設定為 1 方格之卡諾圖化簡所得之積項需多加入外部變數 w，即可得最簡布林函數表示式

$$f(w, x, y, z) = \bar{y} \cdot \bar{z} + w \cdot \bar{z}$$

多輸出布林函數之卡諾圖化簡法

- 對大部分之數位電路而言，在同一組輸入變數之組合下，往往會有兩個或兩個以上之輸出函數，稱為多輸出布林函數 (Multiple-Output Boolean Function)。針對這種多輸出布林函數之化簡，若僅分別考慮每一個輸出函數之最簡表示式，雖求解問題之方法較為簡單，但對整體電路之考量而言，所得之化簡結果未必是最經濟情況，即實現化簡所得之最簡布林函數式，所需之邏輯閘數量未必為最少。
- 對使用同一組輸入變數之多輸出布林函數而言，為得到最經濟之結果，首先檢視所有布林函數間之關係，以尋找這些布林函數之公共項為最先考慮之化簡步驟，雖此考量方法對單一輸出函數而言，未必是最簡之結果，但對整體之考量而言，經此化簡後所得之布林函數式，當採用邏輯閘來實現硬體電路時，將可得到較經濟之結果。

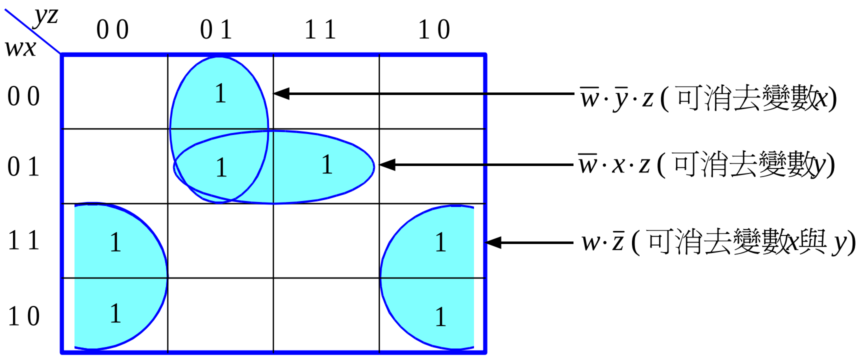
例題 3-9

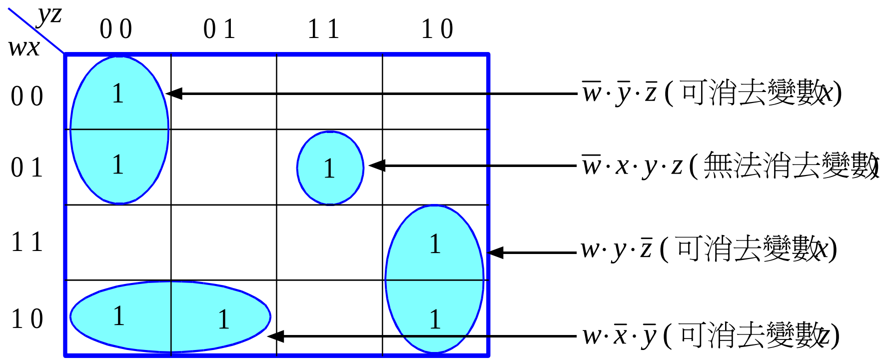
試利用卡諾圖化簡下列兩個使用同一組變數之布林函數式，並將考慮多輸出函數化簡所得之結果與分別考慮單一布林函數之化簡結果，使用邏輯閘來實現整體電路時，並說明何者較為經濟？

$$f_1(w, x, y, z) = \sum(1, 5, 7, 8, 10, 12, 14) \quad \text{與} \quad f_2(w, x, y, z) = \sum(0, 4, 7, 8, 9, 10, 14)$$

解

(1) 首先依照前述之方法，對每一個布林函數求最簡之表示式如下



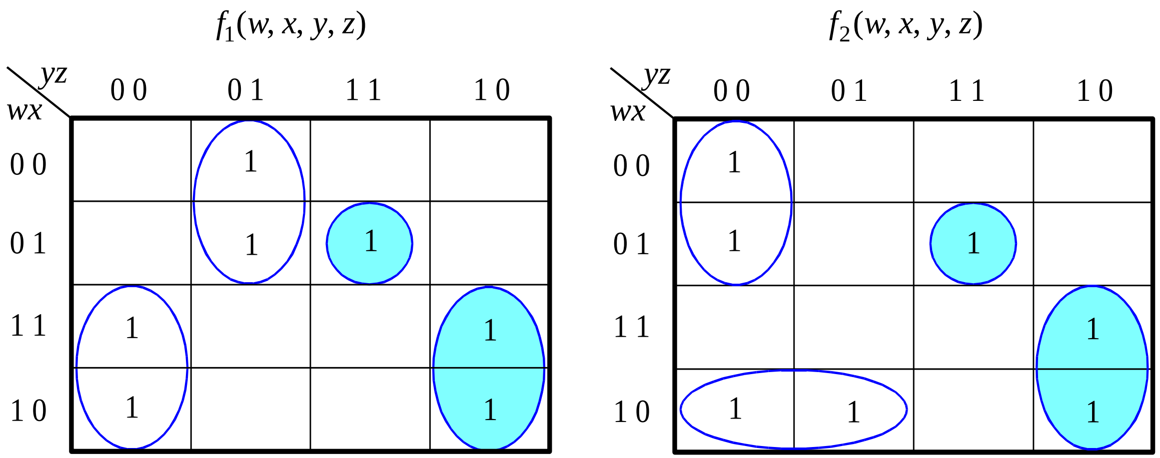


以上卡諾圖法之化簡，即可得兩個布林函數之最簡結果為

$$f_1(w, x, y, z) = \overline{w}\cdot\overline{y}\cdot z + \overline{w}\cdot x\cdot z + w\cdot\overline{z}$$

$$f_2(w, x, y, z) = \overline{w}\cdot\overline{y}\cdot\overline{z} + \overline{w}\cdot x\cdot y\cdot z + w\cdot y\cdot\overline{z} + w\cdot\overline{x}\cdot\overline{y}$$

(2) 接著檢視所求之兩個布林函數間的關係，以尋找 $f_1(w, x, y, z)$ 與 $f_2(w, x, y, z)$ 之公共項，如下圖所示。



利用以上卡諾圖法之化簡，即可得兩個布林函數之最簡結果為

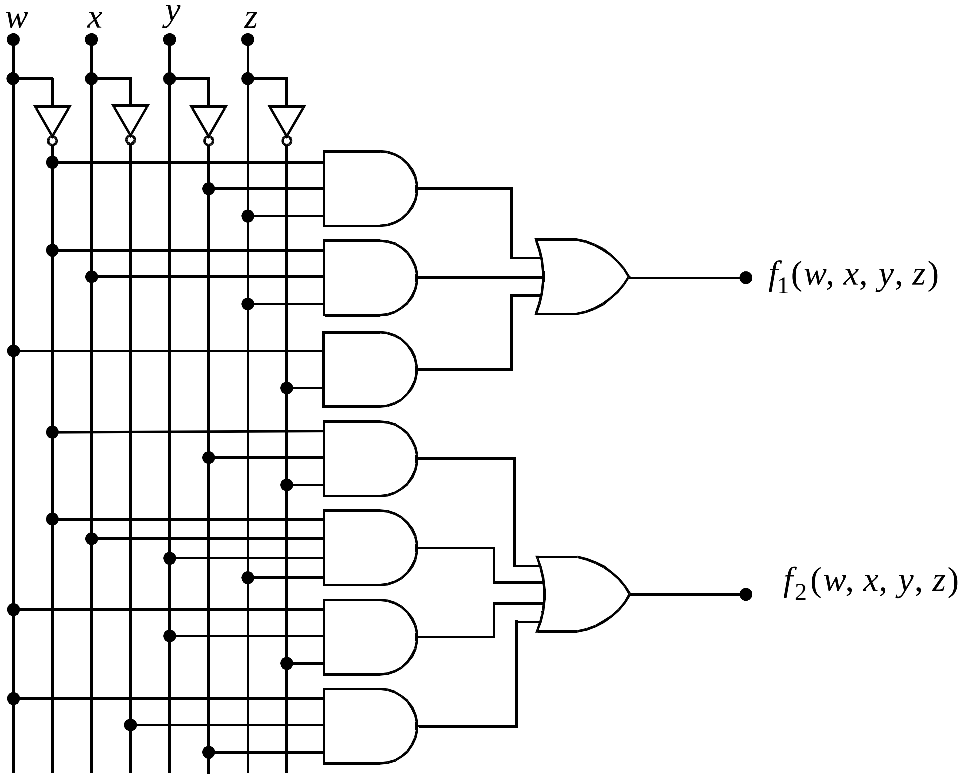
$$f_1(w, x, y, z) = \underline{\overline{w} \cdot x \cdot y \cdot z} + \underline{w \cdot y \cdot \overline{z}} + \overline{w} \cdot \overline{y} \cdot z + w \cdot \overline{y} \cdot \overline{z}$$

$$f_2(w, x, y, z) = \underline{\overline{w} \cdot x \cdot y \cdot z} + \underline{w \cdot y \cdot \overline{z}} + \overline{w} \cdot \overline{y} \cdot \overline{z} + w \cdot \overline{x} \cdot \overline{y}$$

◆ 使用**邏輯閘**來實現對**每一個布林函數求最簡之表示式**，使用邏輯電路來實現時，需使用 **9 個邏輯閘**，如下圖所示。

$$f_1(w, x, y, z) = \overline{w} \cdot \overline{y} \cdot z + \overline{w} \cdot x \cdot z + w \cdot \overline{z}$$

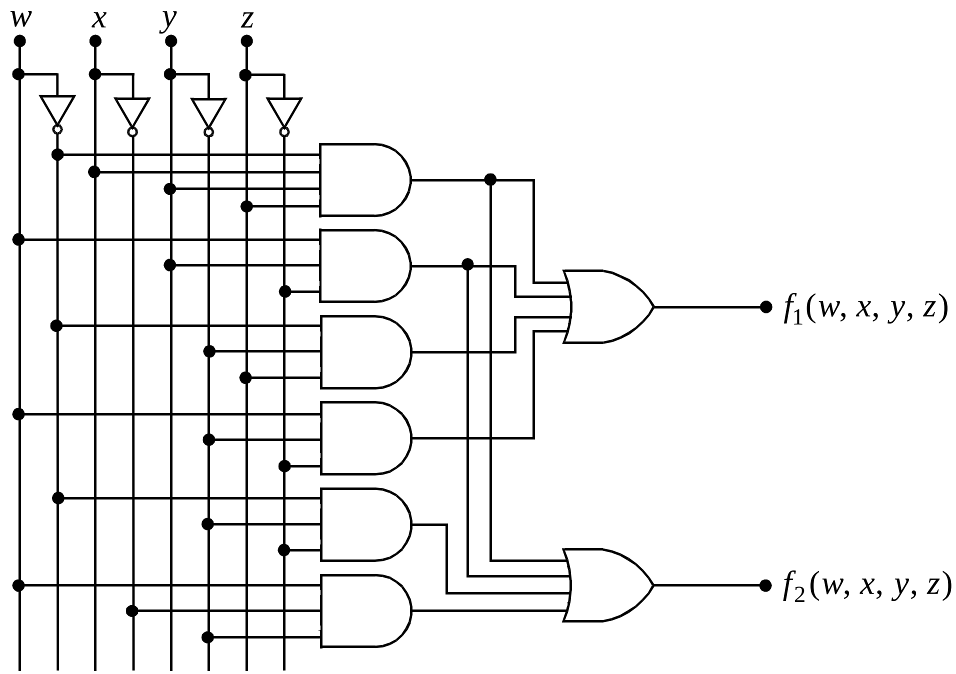
$$f_2(w, x, y, z) = \overline{w} \cdot \overline{y} \cdot \overline{z} + \overline{w} \cdot x \cdot y \cdot z + w \cdot y \cdot \overline{z} + w \cdot \overline{x} \cdot \overline{y}$$



◆ 檢視所求之兩個布林函數間的關係，以尋找 $f_1(w, x, y, z)$ 與 $f_2(w, x, y, z)$ 之公共項，兩個布林函數之最簡結果，使用邏輯電路來實現時，僅需使用 8 個邏輯閘，如下圖所示。

$$f_1(w, x, y, z) = \underline{\overline{w} \cdot x \cdot y \cdot z} + \underline{w \cdot y \cdot \overline{z}} + \overline{w} \cdot \overline{y} \cdot z + w \cdot \overline{y} \cdot \overline{z}$$

$$f_2(w, x, y, z) = \underline{\overline{w} \cdot x \cdot y \cdot z} + \underline{w \cdot y \cdot \overline{z}} + \overline{w} \cdot \overline{y} \cdot \overline{z} + w \cdot \overline{x} \cdot \overline{y}$$



◆ 使用邏輯閘來實現化簡所得之布林函數時，(2)部份(需 8 個邏輯閘)比(1)部份(需 9 個邏輯閘)之結果較為經濟。

列　表　法

◆ 一種適合使用計算機語言 (Computer Language) 之布林函數化簡法，可用來克服變數數目為六個以上布林函數之化簡問題，稱為列表法 (Tabulation Method)，以化簡變數數目增加至六個以上時，因卡諾圖之方格數目太多，導致化簡工作變得相當複雜難解之問題。

◆ 雖然使用列表法來化簡布林函數是一種比較有系統之方法，但其化簡過程相當之繁複且單調，故此種方法不適合採用人工計算，而比較適合採用計算機來輔助化簡。

◆ 使用列表法來化簡布林函數時，必須先將所求布林函數展開成標準積項之和 (SOP) 後，再分兩個步驟，即

1. 尋找所求布林函數之質隱項 (Prime Implicants) 集合。
2. 將步驟 (1) 所得之質隱項集合轉換成質隱項表後，再利用此質隱項表求取必要質隱項 (Essential Prime Implicant)，最後將這些必要質隱項用 OR 連結起來，即為所求之最簡布林函數式。

質隱項(Prime Implicant)集合之求解

◆ 將所求布林函數展開成標準積項之和(SOP)，再將所有之最小項(Minterms)，以特定方式列成一個表後，接著經過匹配程序來求解布林函數之質隱項(Prime Implicant)集合，而求解質隱項集合步驟列述於後：

1. 以所求布林函數式之最小項 (Minterms) 所相對應二進位數字之 1 的個數為指標 (Index) 個數，並依指標個數之多寡為順序，以排列出決定質隱項之表格

2. 依序比較指標個數為 i 與 $i+1$ 之所有最小項，若兩個最小項間所對應之二進位數字，僅有 1 個二進位數有 0 與 1 的變化，則表示此兩個最小項可合併成一項，並將可合併之兩個最小項作記號(＊)，並在下一步驟將此兩個最小項所對應之十進位數列出來(所能合併之最小項僅限於指標個數為 i 與 $i+1$ 之所有最小項)。

3. 重覆步驟 (2) 之合併程序，並將可能合併之最小項作記號 (＊)，直到所有指標個數為 i 與 $i+1$ 之所有最小項全部處理完畢為止。

4. 集合步驟 (2) 與 (3) 之最小項合併過程中，未被作記號 (＊) 之最小項，即為所求之質隱項集合。

例題 3-10

試求布林函數 $f(w,x,y,z)=\overline{x}\cdot y\cdot z+\overline{x}\cdot\overline{y}\cdot z+w\cdot x\cdot\overline{y}\cdot\overline{z}+w\cdot x\cdot\overline{y}\cdot z+x\cdot y\cdot\overline{z}$ 之質隱項集合。

解

展開已知布林函數為標準積項之和為

$$f(w,x,y,z)=\overline{x}\cdot y\cdot z+\overline{x}\cdot\overline{y}\cdot z+w\cdot x\cdot\overline{y}\cdot\overline{z}+w\cdot x\cdot\overline{y}\cdot z+x\cdot y\cdot\overline{z}=\sum(1,3,6,9,11,12,13,14)$$

將上述之最小項所對應的二進位數，具有相同指標個數整理在同一組，並由大至小依序排列，比較指標個數為 i 與 $i+1$ 之所有最小項，若兩個最小項間僅有 1 個二進位數有 0 與 1 的變化，則表示此兩個最小項可以合併成一項，並將可合併之兩個最小項作記號，如下表所示。

指標個數	等效十進位數	等效二進位數				
		w	x	y	z	
1	1	0	0	0	1	*
2	3	0	0	1	1	*
	6	0	1	1	0	*
	9	1	0	0	1	*
	12	1	1	0	0	*
3	11	1	0	1	1	*
	13	1	1	0	1	*
	14	1	1	1	0	*

處理上表中指標個數為 i 與 $i+1$ 之可合併之最小項，重複上述之方法，再繼續比較指標個數為 i 與 $i+1$ 之所有最小項，所得結果如下表所示。

指標個數	等效十進位數	等效二進位數				
		w	x	y	z	
1	1, 3	0	0	–	1	*
	1, 9	–	0	0	1	*
2	3, 11	–	0	1	1	*
	6, 14	–	1	1	0	← 質隱項 A
	9, 11	1	0	–	1	*
	9, 13	1	–	0	1	← 質隱項 B
	12, 13	1	1	0	–	← 質隱項 C
	12, 14	1	1	–	0	← 質隱項 D

接著處理上表中指標個數為 i 與 $i+1$ 之可合併之最小項，所得結果如下表所示。

指標個數	等效十進位	等效二進位數				
		w	x	y	z	
1	1, 3, 9, 11	–	0	–	1	← 質隱項 E

檢查指標個數為 i 與 $i+1$ 所得之三個表中，將標示有質隱項之最小項組合起來，即可得質隱項之集合 $I=\{x\cdot y\cdot \bar{z},\, w\cdot \bar{y}\cdot z,\, w\cdot x\cdot \bar{y},\, w\cdot x\cdot \bar{z},\, \bar{x}\cdot z\}$。

必要質隱項 (Essential Prime Implicant) 之求解

- 為有系統之求解必要質隱項，採用派屈克演算法 (Petrick Algorithm) 來求解，此演算法先列出質隱項與其所包含的最小項關係之質隱項表 (Prime Implicant Table)，接著使用包含消去法求解最少質隱項數目，且在每一個質隱項中字母最少之乘積項稱為必要質隱項，這些必要質隱項之集合，即為所求布林函數之最簡式。

- 必要質隱項是由質隱項表得到，故首先將初步化簡所得質隱項集合，以每一個質隱項成一列 (Row) 方式，依序組成一個質隱項表後，再將每一個質隱項所有可能對應之十進位數字（即將質隱項展開成標準積項之和後，所對應之十進位數字）上作記號，接著利用所得之質隱項表，觀察這些質隱項間彼此之包含關係，以消去被包含之列 (Row) 或行 (Column)，便可決定所求必要質隱項。而質隱項表選取必要質隱項，以求得最簡布林函數式之步驟如下：

1. 由質隱項表之各行 (Column) 中選取一個最少記號（*）之質隱項為必要質隱項（在質隱項表中，一個最小項只被一個質隱項所包含時，此質隱項即為必要質隱項），並用一小圓圈標示出該必要質隱項。接著使用包含消去法來求取必要質隱項，其作法為當被選取之行為C_j，則被C_i所包含有加註記號之行或列均可被消除。

2. 重複上述步驟，直到質隱項表中，所有列 (Column) 皆被加上標記 (×) 為止。

3. 最後在將所選取之必要質隱項，用 OR 連結起來，即可得到所求布林函數之最簡表示式。

例題 3-11

試以例題 3-10 所得之質隱項來求解所求布林函數之最簡表示式。

解

將例題 3-10 化簡所得質隱項集合，以每個質隱項成一列 (Row) 方式，依序組成一個質隱項表，接著將每個質隱項之所有可能對應十進位數字上作記號 (∗)，如下表所示。

質隱項對應之十進位數

	質隱項	1	3	6	9	11	12	13	14	
A	$x\cdot y\cdot \bar{z}$			∗					∗	必要質隱項
B	$w\cdot \bar{y}\cdot z$				∗			∗		
C	$w\cdot x\cdot \bar{y}$						∗	∗		必要質隱項
D	$w\cdot x\cdot \bar{z}$						∗		∗	
E	$\bar{x}\cdot z$	∗	∗		∗	∗				必要質隱項
		×	×	×	×	×	×	×	×	

1. 檢視「6」行可知，此行僅有一個「∗」，故首先選擇「質隱項 A」為必要質隱項，而此行同時包含「6」與「14」等兩行，故可在這兩行畫「×」之標記。
2. 考慮「3」行可知，此行亦僅有一個「∗」，故選擇「質隱項 E」為必要質隱項，而此行同時包含「1」、「3」、「9」與「11」等 4 行，故可在這 4 行畫「×」之標記。

3. 選擇「12」行（即選擇「質隱項 C」為必要質隱項），此行同時包含「12」與「13」等兩行，故可在這兩行畫「×」之標記。

最後將所有選取之必要質隱項所對應的積項，用 OR 連結起來，即可得到所求之最簡布林函數表示式為

$$f(w, x, y, z) = \bar{x} \cdot z + w \cdot x \cdot \bar{y} + x \cdot y \cdot \bar{z}$$

問題討論

1、試討論進行布林代數化簡之目的與布林化簡之原則。

解：

(1) 布林代數化簡之目的：在設計數位系統時，將所得之布林代數，依所設計數位電路之實際狀況，進行適當的化簡 (Minimization) 工作，以降低布林代數之複雜度，便可減少實現數位電路 (Digital Circuit) 所需之邏輯閘數量與邏輯閘間連接線 (Interconnections) 之複雜度，如此可進一步降低製作硬體 (Hardware) 電路所需之成本。

(2) 布林化簡之原則：將布林函數 (Boolean Function) 表示式採用某些簡化準則化簡後，變成另外一種與原函式等效，且較簡單之布林代數式來表示。

2、若利用積項之和(和項之積)來表示布林代數，試說明經過化簡後，認定所得布林代數為最簡表示式之原則。

解：

經化簡所得之布林函數表示式，若同時符合以下兩個條件，即可認定所得之布林函數為最簡表示式。

(1) 積項（或和項）之項數 (Number of the Terms) 最少。

(2) 積項（或和項）之變數數目 (Number of the Variables) 最少。

3、試討論卡諾圖來進行布林代數之化簡時，所使用圖形之方格式數目、方格代表之意義與方格排列應遵守的規則。

解：

(1) 卡諾圖 (Karnaugh Map) 之方格數目是依所化簡之布林函數式的變數來決定，即 n 個變數之布林函數式，需用 2^n 方格數目來表示，而每一個方格可用來表示 n 個變數之其中一個可能的最小項或最大項。

(2) 在卡諾圖之每個方格中，當其值為 1 時，則此方格稱為 1 方格 (1 Cell)；而其值為 0 時，則此方格稱為 0 方格 (0 Cell)。

(3) 卡諾圖中之方格排列需遵循一定之規則，即兩相鄰 (Adjacency) 方格間之行 (Column) 與列 (Row)，所對應之二進位組合僅可允許有一個位元改變（即相鄰方格所代表之最小項或最大項，僅可允許有一個變數為互補之型態）。

4、試說明決定使用積項之和或和項之積來進行布林代數化簡之差異與時機。

解：

(1) 使用積項之和 (SOP) 與和項之積 (POS) 等兩種形式，均可用來表示相同功能之布林函數式，因此使用卡諾圖來化簡時，採用任何一種形式所得之結果應會完全相等，但於形式上會有不同。而所不同之部分為

(a) 採用積項之和（標示為 1 方格）來化簡，所得之結果為原函數。

(b) 採用和項之積（標示為 0 方格）來化簡，則所得之結果為原函數之補數。

(1) 利用卡諾圖來化簡布林函數時，需檢視使用 1 方格或 0 方格來化簡，所得之布林函數式較為簡單，以確保所得之結果為最簡的布林代數表示式，以節省實現硬體電路 (Hardware Circuit) 所需之成本。

5、試說明不完全指定函數之定義。當利用卡諾圖來化簡布林代數時，處理被設定為不完全指定函數方格，以得到最簡布林代數之方法。

解：

(1) 在設計數位電路時，有些輸入變數之最小項（最大項），會因某些原因可能不會出現或可以任意被指定為邏輯 1 或邏輯 0，亦即這些最小項（最大項）可存在，亦可不存在，這種函數被稱為不完全指定函數 (Incompletely Specified Function)。

(2) 利用卡諾圖來化簡布林函數時，若有利於形成更多相鄰之 1 方格 (0 方格）時，以消去更多之變數時，就利用設定為不完全指定函數之方格來幫助得到更簡之布林函數式；若無發法幫助形成更多相鄰之 1 方格 (0 方格）時，則在進行化簡時，就可將設定為不完全指定函數之方格視同不存在。