第 3 章　布尔代数基础

布尔代数得名于英国数学家乔治·布尔(George Boole,1815—1864)。为了研究人的逻辑思维规律,他在 1847 年发表的《逻辑的数学分析》和 1854 年发表的《思维规律的研究》两部著作中,首先提出了这种代数的基本概念和性质。此后,大约经过近 100 年之久,于 1938 年才由克劳德·香农(Claude E. Shannon)将布尔代数应用于电话继电器的开关电路中。至今,布尔代数已成为分析和设计开关电路的重要数学工具。

本章不是从数学的角度去研究布尔代数,而是从应用的角度介绍布尔代数的一些基本概念、基本定理、布尔函数的基本形式以及布尔函数的化简方法,以使读者掌握分析和设计数字逻辑网络所需的数学工具。

3.1　布尔代数的基本概念

计算机或其他数字系统无论多么复杂,它们都是由若干种最简单的、最基本的电路(如门电路、触发器等)所组成的。这些电路的工作具有下列基本特点:从电路内部看,或是管子导通,或是管子截止;从电路的输入输出看,或是电平的高低,或是脉冲的有无。由于这种电路工作在开关状态,故称为开关电路。开关电路的工作状态可以用二元布尔代数来描述,故二元布尔代数通常称为开关代数,或称为逻辑代数。因此,逻辑代数只是布尔代数的一种特例。在本书中,如无特别说明,布尔代数均指逻辑代数。

3.1.1　布尔变量及其基本运算

布尔代数和普通代数一样,用字母代表变量,布尔代数的变量称为布尔变量。和普通代数不同的是,布尔变量只有两种取值,即 0 或 1。并且,常量 0 和 1 没有普通代数中的 0 和 1 的意义,它只表示两种可能,即命题的“假”和“真”,信号的“无”和“有”等。

布尔代数中的变量运算只有“或”、“与”、“非” 3 种基本运算,任何复杂的逻辑运算都可以通过这 3 种基本运算来实现。

1. “或”运算

“或”运算又称为逻辑加。两个变量“或”运算的逻辑关系可表示为

$$F = A + B$$

式中,“+”号是“或”运算符。上式读作“F 等于 A 或 B”,或者“F 等于 A 加 B”,其意思是变量 A 和 B 中只要有一者取值为 1,则 F 就为 1;若 A 和 B 全为 0,则 F 为 0。其逻辑关系可以用真值表来描述,如表 3.1 所示。

2. “与”运算

“与”运算又称为逻辑乘。两个变量的“与”运算的逻辑关系可表示为

$$F = A \cdot B$$

式中“·”号表示“与”运算符。通常,“与”运算符可以省略。上式读作“F 等于 A 与 B”,或者“F 等于 A 乘 B”。其含义是只有当变量 A 与 B 都为 1 时;F 才为 1;否则,F 就为 0。其逻辑关系可以用真值表来描述,如表 3.2 所示。

表 3.1 “或”运算

A	B	F
0	0	0
0	1	1
1	0	1
1	1	1

表 3.2 “与”运算

A	B	F
0	0	0
0	1	0
1	0	0
1	1	1

3. “非”运算

“非”运算又称为逻辑取反。对一个变量的“非”运算的逻辑关系可表示为:

$$F = \overline{A}$$

式中“—”号表示“非”运算符。上式读作“F 等于 A 的非”,其意思是若 A 为 1,则 F 为 0;反之,若 A 为 0,则 F 为 1。“非”运算的逻辑关系可以用表 3.3 所示的真值表来描述。

表 3.3 “非”运算

A	F
0	1
1	0

综合上述对布尔变量及其 3 个基本运算的定义,我们可以对布尔代数下个定义:

布尔代数是一个由布尔变量集 K,常量 0、1 以及“或”、“与”、“非” 3 种运算符所构成的代数系统,记为

$$B = (K, +, \cdot, -, 0, 1)$$

其中,布尔变量集 K 是指布尔代数中的所有可能变量的集合,它可用任何字母表示,但每一个变量的取值只可能为常量 0 或 1,而且布尔代数中的变量只有“或”、“与”、“非” 3 种运算。

3.1.2 布尔函数及其表示方法

布尔代数中的函数定义与普通代数中函数定义十分相似,可以叙述如下。

设$(x_1, x_2, \cdots, x_n)$为布尔代数的一组布尔变量,其中每个变量取值为 0 或 1,则当把 n 序列$(x_1, x_2, \cdots, x_n)$映射到 $B=\{0,1\}$时,这个映射就是一个布尔函数。

从另一个角度,把布尔函数与逻辑网络联系起来,布尔函数可以这样叙述:

设某一逻辑网络的输入变量为 x_1、x_2、$\cdots$、x_n,输出变量为 F,如图 3.1 所示。对应于变量 x_1、x_2、$\cdots$、x_n的每一组确定值,F 就有唯一确定的值,则称 F 是变量 x_1、x_2、$\cdots$、x_n的布尔函数。记为

$$F = f(x_1, x_2, \cdots, x_n)$$

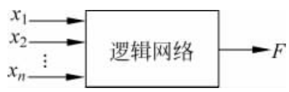


图 3.1　布尔函数

注意，布尔代数中函数的取值也只可能是 0 或 1，这与普通代数是不同的。

布尔函数的表示方法有 3 种形式：布尔表达式、真值表和卡诺图。这与普通代数中用公式、表格和图解这 3 种方法来表示函数十分类似。

1. 布尔表达式

布尔表达式是由布尔变量和“或”、“与”、“非” 3 种运算符所构成的式子，这是一种用公式表示布尔函数的方法。例如，要表示这样一个函数关系：当两个变量 A 和 B 取值相同时，函数取值为 0；否则，函数取值为 1。此函数称为异或函数，可以用下列布尔表达式来表示：

$$F = f(A, B) = \overline{A}B + A\overline{B}$$

显然，只要将 A 和 B 的 4 种可能取值代入这表达式，验证是正确的。

与异或函数相反，当两个变量 A 和 B 取值相同时，函数取值为 1；否则，函数取值为 0。此函数称为同或函数。通常，异或运算用符号$\oplus$表示；同或运算用$\odot$表示。因此，异或函数、同或函数可分别表示成：

$$F = \overline{A}B + A\overline{B} = A \oplus B;\ F = \overline{A}\,\overline{B} + A\,B = A \odot B$$

2. 真值表

真值表是由输入变量的所有可能取值组合及其对应的输出函数值所构成的表格，这是一种用表格表示布尔函数的方法。例如，对于前面的异或函数，可以用表 3.4 所示的真值表来表示。

真值表中的变量为两个，共有 2^2 种取值组合，所以该表由 4 行组成。当变量为 n 个时，真值表就由 2^n 行组成。显然，随着变量数目的增加，真值表的行数将急剧增加。因此，一般当变量数目不超过 4 个时，用真值表表示函数比较方便。

表 3.4　异或函数的真值表

A	B	F
0	0	0
0	1	1
1	0	1
1	1	0

3. 卡诺图

卡诺图是由表示逻辑变量的所有可能取值组合的小方格所构成的图形，如图 3.2 所示。图中分别表示了二变量及三变量的卡诺图。

利用卡诺图，可以很方便地表示一个函数。只要在那些使函数值为 1 的变量取值组合

所对应的小方格上标记 1,便得该函数的卡诺图。例如,对于异或函数,可以用图 3.3 所示的卡诺图来表示。

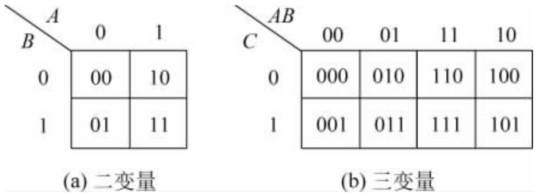


图 3.2 卡诺图

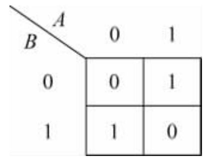


图 3.3 异或函数的卡诺图

卡诺图可以看成是真值表的重新排列,真值表的每一行用一个小方格来表示。当变量为两个时,真值表有 4 行,相应的卡诺图有 4 个方格;当变量为 n 个时,卡诺图有 2^n 个方格。卡诺图的这种方格排列方式比真值表更紧凑,而且便于进行函数的简化。

3.1.3 布尔函数的“相等”概念

布尔函数和普通代数一样,也有函数相等的问题。两个函数相等的定义如下。

设有两个布尔函数

$$F = f(x_1, x_2, \cdots, x_n)$$
$$G = g(x_1, x_2, \cdots, x_n)$$

其变量都为 $x_1, x_2, \cdots, x_n$。如果对应于变量 $x_1, x_2, \cdots, x_n$ 的任何一组变量取值,F 和 G 的值都相同,则称 F 和 G 是相等的,记为 $F=G$。

显然,若两个布尔函数相等,则它们的真值表一定相同;反之,若两个布尔函数的真值表完全相同,则此两个函数相等。因此,要证明两个布尔函数是否相等,只要分别列出它们的真值表,看其是否相同。

例如,已知下列两个函数

$$F = \overline{xy} \qquad G = \bar{x} + \bar{y}$$

列出 F 和 G 的真值表,如表 3.5 所示。由表可知,它们的真值表完全相同,故 F 和 G 是相等的,即有

$$\overline{xy} = \bar{x} + \bar{y}$$

表 3.5 $F=\overline{xy}$ 和 $G=\bar{x}+\bar{y}$ 的真值表

x	y	xy	$F=\overline{xy}$	$\bar{x}$	$\bar{y}$	$G=\bar{x}+\bar{y}$
0	0	0	1	1	1	1
0	1	0	1	1	0	1
1	0	0	1	1	1	1
1	1	1	0	0	0	0

3.2　布尔代数的公式、定理和规则

3.2.1　布尔代数的基本公式

根据布尔变量的取值只有 0 和 1，以及布尔变量仅有的 3 种运算的定义，不难推出下列基本公式。

(1) 交换律

$$A + B = B + A$$
$$A \cdot B = B \cdot A$$

(2) 结合律

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

(3) 分配律

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

(4) 0-1 律

$$\begin{cases} A + 0 = A \\ A + 1 = 1 \end{cases}$$
$$\begin{cases} A \cdot 0 = 0 \\ A \cdot 1 = A \end{cases}$$

(5) 互补律

$$A + \overline{A} = 1$$
$$A \cdot \overline{A} = 0$$

(6) 等幂律

$$A + A = A$$
$$A \cdot A = A$$

(7) 吸收律

$$\begin{cases} A + AB = A \\ A + \overline{A}B = A + B \end{cases}$$
$$\begin{cases} A(A + B) = A \\ A(\overline{A} + B) = AB \end{cases}$$

(8) 对合律(双重否定律)

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

以上是布尔代数的基本公式。其中交换律、结合律、分配律、0-1 律、互补律和对合律可以作为布尔代数的公理。公理是代数系统的基本出发点，是客观存在的抽象，它无须证明，

但它可以用客观存在来验证。以此为基础，可以推得布尔代数的等幂律与吸收律。例如，等幂律的证明如下：

$$
\begin{aligned}
A+A &= (A+A)\cdot 1 && (0\text{-}1\ 律)\\
&= (A+A)(A+\overline{A}) && (互补律)\\
&= A+A\cdot\overline{A} && (分配律)\\
&= A+0 && (互补律)\\
&= A && (0\text{-}1\ 律)
\end{aligned}
$$

又如，吸收律的证明如下：

$$
\begin{aligned}
A+\overline{A}B &= (A+\overline{A})(A+B) && (分配律)\\
&= 1\cdot(A+B) && (互补律)\\
&= A+B && (0\text{-}1\ 律)
\end{aligned}
$$

必须指出，上述基本公式中，有些公式与普通代数中的相同，如交换律、结合律，但有些公式却是布尔代数中所特有的，如分配律。

3.2.2 布尔代数的主要定理

定理 1 德·摩根(De Morgan)定理。

(1) $\overline{x_1+x_2+\cdots+x_n}=\overline{x_1}\cdot\overline{x_2}\cdot\cdots\cdot\overline{x_n}$

(2) $\overline{x_1\cdot x_2\cdot\cdots\cdot x_n}=\overline{x_1}+\overline{x_2}+\cdots+\overline{x_n}$

这就是说，n 个变量的“或”的“非”等于各变量的“非”的“与”，n 个变量的“与”的“非”等于各变量的“非”的“或”。

当变量数目较少时，该定理可很容易用真值表证明。当变量为 n 个时，则可以用数学归纳法证明。

德·摩根定理是布尔代数中一个很重要、且经常使用的定理，它提供了一种变换布尔表达式的简便方法。由于它具有反演特性，即把变量的与运算改成或运算，或运算改成与运算，所以又称为反演律。

定理 2 香农(Shannon)定理。

$$\overline{f(x_1,x_2,\cdots,x_n,0,1,+,\cdot)}=f(\overline{x_1},\overline{x_2},\cdots,\overline{x_n},1,0,\cdot,+)$$

这就是说，任何函数的反函数(或称补函数)，可以通过对该函数的所有变量取反，并将常量 1 换为 0，0 换为 1，运算符“+”换为“·”，“·”换为“+”而得到。

证明 根据德·摩根定理，任何函数的反函数可写成

$$
\begin{aligned}
&\overline{f(x_1,x_2,\cdots,x_n,0,1,+,\cdot)}\\
&=\overline{f_1(x_1,x_2,\cdots,x_n,0,1,+,\cdot)+f_2(x_1,x_2,\cdots,x_n,0,1,+,\cdot)}\\
&=\overline{f_1(x_1,x_2,\cdots,x_n,0,1,+,\cdot)}\cdot\overline{f_2(x_1,x_2,\cdots,x_n,0,1,+,\cdot)}
\end{aligned}
$$

或写成：

$$
\begin{aligned}
&\overline{f(x_1,x_2,\cdots,x_n,0,1,+,\cdot)}\\
&=\overline{f_1(x_1,x_2,\cdots,x_n,0,1,+,\cdot)\cdot f_2(x_1,x_2,\cdots,x_n,0,1,+,\cdot)}\\
&=\overline{f_1(x_1,x_2,\cdots,x_n,0,1,+,\cdot)}+\overline{f_2(x_1,x_2,\cdots,x_n,0,1,+,\cdot)}
\end{aligned}
$$

其中 f_1 和 f_2 是 f 的两个部分函数。对 f_1 和 f_2 重复上述过程，直到使 f 中的每个变量都用德・摩根定理。由于每对 f(或 f 的部分函数)应用一次德・摩根定理，就将部分函数(或子部分函数)取反，并将“与”、“或”运算变换一次，以求得函数 f(或部分函数)的反函数 $\bar{f}$，因此，当对 f 的每个变量进行德・摩根变换后，其结果必然是 $f(\overline{x_1}, \overline{x_2}, \cdots, \overline{x_n}, 1, 0, \cdot, +)$，证毕。

香农定理实际上是德・摩根定理的推广，它可以用在任何复杂函数。

【例 3.1】 已知函数 $F=\bar{A}B+A\bar{B}(C+\bar{D})$，求其反函数 $\bar{F}$。

$$
\begin{aligned}
\bar{F} &= \overline{\bar{A}B + A\bar{B}(C+\bar{D})} = \overline{\bar{A}B} \cdot \overline{A\bar{B}(C+\bar{D})} \\
&= (A+\bar{B}) \cdot (\overline{A\bar{B}} + \overline{C+\bar{D}}) = (A+\bar{B})((\bar{A}+B)+\bar{C}D)
\end{aligned}
$$

利用香农定理，可以直接写出

$$\bar{F} = (A+\bar{B}) \cdot (\bar{A}+B+\bar{C}D)$$

定理 3 展开定理。

(1) $f(x_1, x_2, \cdots, x_i, \cdots, x_n)$

$=x_i f(x_1, x_2, \cdots, 1, \cdots, x_n)+\overline{x_i} f(x_1, x_2, \cdots, 0, \cdots, x_n)$

(2) $f(x_1, x_2, \cdots, x_i, \cdots, x_n)$

$=(x_i+f(x_1, x_2, \cdots, 0, \cdots, x_n)) \cdot (\overline{x_i}+f(x_1, x_2, \cdots, 1, \cdots, x_n))$

这就是说，任何布尔函数都可以对它的某一变量 x_i 展开，或展开成(1) 所示的“与-或”形式，或展开为(2) 所示的“或-与”形式。

证明 将 $x_i=1, \overline{x_i}=0$ 代入上式，再将 $x_i=0, \overline{x_i}=1$ 代入上式，则两种情况下等式均成立。证毕。

由展开定理可得下列两个推理。

推理 1

(a) $x_i f(x_1, \cdots, x_i, \cdots, x_n)=x_i f(x_1, \cdots, 1, \cdots, x_n)$

(b) $x_i+f(x_1, \cdots, x_i, \cdots, x_n)=x_i+f(x_1, \cdots, 0, \cdots, x_n)$

推理 2

(a) $\overline{x_i} f(x_1, \cdots, x_i, \cdots, x_n)=\overline{x_i} f(x_1, \cdots, 0, \cdots, x_n)$

(b) $\overline{x_i}+f(x_1, \cdots, x_i, \cdots, x_n)=\overline{x_i}+f(x_1, \cdots, 1, \cdots, x_n)$

下面举例说明展开定理的应用。

【例 3.2】 证明公式 $AB+\bar{A}C+BC=AB+\bar{A}C$(包含律)。

用展开定理，等号左边可展开成

$$
\begin{aligned}
\text{左边} &= A(1 \cdot B + 0 \cdot C + BC) + \bar{A}(0 \cdot B + 1 \cdot C + BC) \\
&= A(B+BC) + \bar{A}(C+BC) = AB + \bar{A}C = \text{右边}
\end{aligned}
$$

证毕

【例 3.3】 将函数 $F=\bar{A}\bar{B}+AC$ 表示成“或-与”形式。

由展开定理，可得

$$F = [A+(1 \cdot \bar{B}+0 \cdot C)] \cdot [\bar{A}+(0 \cdot \bar{B}+1 \cdot C)] = (A+\bar{B})(\bar{A}+C)$$

3.2.3 布尔代数的重要规则

布尔代数有 3 个重要规则，即代入规则、反演规则和对偶规则，现分别叙述如下。

1. 代入规则

任何一个含有变量 x 的等式,如果将所有出现 x 的位置,都代之以一个布尔函数 F,则等式仍然成立。这个规则称为代入规则。

由于任何一个布尔函数也和任何一个变量一样,只有 0 或 1 两种取值,显然,以上规则是成立的。

【例 3.4】 已知等式$\overline{A+B}=\bar{A}\cdot\bar{B}$,函数 $F=B+C$,若用 F 代入此等式中的 B,则有

$$\overline{A+(B+C)}=\bar{A}\cdot\overline{B+C}$$

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

据此可以证明 n 变量的德·摩根定理的成立。

2. 对偶规则

任何一个布尔函数表达式 F,如果将表达式中的所有的“+”改成“·”,“·”改成“+”,“1”改成“0”,“0”改成“1”,而变量保持不变,则可得到一个新的函数表达式 F_d,我们称 F_d为 F 的对偶函数,这一规则称为对偶规则。例如,下列为几个原函数及其对偶函数:

$$F=\bar{A}B+A\bar{B}\bar{C} \qquad F_d=(\bar{A}+B)(A+\bar{B}+\bar{C})$$

$$F=A(\bar{B}+CD)+E \qquad F_d=[A+\bar{B}(C+D)]\cdot E$$

$$F=(A+0)\cdot(B+C\cdot 1) \qquad F_d=A\cdot 1+B\cdot(C+0)$$

$$F=A+\overline{B+\bar{C}+\overline{D+\bar{E}}} \qquad F_d=A\cdot\overline{B\cdot\bar{C}\cdot\overline{D\cdot\bar{E}}}$$

需要注意的是,在运用对偶规则求对偶函数时,必须按照先“与”后“或”的顺序,否则容易写错,如 $F=\bar{A}B+A\bar{B}\bar{C}$,若求出对偶函数 $F_d=\bar{A}+B\cdot A+\bar{B}+\bar{C}$,是错误的。因此,要特别注意原来函数中的“与”项,当这些“与”项变为“或”项时,应加括号。

从上面这些例子可以看出,如果 F 的对偶函数为 F_d,则 F_d的对偶函数就是 F。也就是说,F 和 F_d互为对偶函数,即$(F_d)_d= F$。

由布尔代数的基本公式可以看出,它们都是成对出现的互为对偶的等式。由此证明一个规则:如果两个函数的表达式相等,则它们的对偶函数也相等,即如果函数 $F=G$,则其对偶函数 $F_d=G_d$。

3. 反演规则

任何一个布尔函数表达式 F,如果将表达式中的所有的“+”改成“·”,“·”改成“+”,“1”改成“0”,“0”改成“1”,原变量改成反变量,反变量改成原变量,则可得函数 F 的反函数(或称补函数)$\bar{F}$,这个规则称为反演规则。

实际上,反演规则就是香农定理。运用反演规则可以很方便地求一个函数的补函数。例如,下列为几个原函数及其补函数:

$$F=\bar{A}B+A\bar{B}\bar{C} \qquad \bar{F}=(A+\bar{B})(\bar{A}+B+C)$$

$$F=A(\bar{B}+CD)+E \qquad \bar{F}=[\bar{A}+B(\bar{C}+\bar{D})]\cdot\bar{E}$$

$$F=(A+0)\cdot(B+C\cdot 1) \qquad \bar{F}=\bar{A}\cdot 1+\bar{B}\cdot(\bar{C}+0)$$

$$F=A+\overline{B+\bar{C}+\overline{D+\bar{E}}} \qquad \bar{F}=\bar{A}\cdot\overline{\bar{B}\cdot C\cdot\overline{\bar{D}\cdot E}}$$

与求对偶函数一样，求补函数需要注意的是，在运用反演规则时，必须按照先“与”后“或”的顺序进行变换。因此，特别要注意原来函数中的“与”项，当这些“与”项变换为“或”项时，应加括号。

把上述补函数的例子与前面对偶函数的例子对照一下，可以看出，补函数和对偶函数之间在形式上只差变量的“非”。因此，若已求得一函数的对偶函数，只要将所有变量取反便得该函数的补函数；反之亦然。

3.3　布尔函数的基本形式

一个布尔函数的表达式可以有多种表示形式，本节讨论几种基本形式，它们是进行布尔函数化简的基础。

3.3.1　函数的“积之和”与“和之积”形式

根据展开定理，任何一个 n 变量函数总可以展开成“与-或”形式，或者“或-与”形式。其中“与-或”形式又称为“积之和”形式，而“或-与”形式又称为“和之积”形式。

所谓“积之和”，是指一个函数表达式中包含若干个“积”项，其中每个“积”项可有一个或多个以原变量或反变量形式出现的字母，这些“积”项的“和”就表示了一个函数。例如，一个三变量函数为

$$F(A,B,C)=\overline{A}+B\overline{C}+A\overline{B}C$$

其中，$\overline{A}$、$B\overline{C}$、$A\overline{B}C$ 均为“积”项。这些积项的和，就表示了函数的“积之和”形式。

所谓“和之积”，是指一个函数表达式中包含若干个“和”项，其中每个“和”项可有一个或多个以原变量或反变量形式出现的字母，这些“和”项的“积”就表示一个函数。例如，一个四变量函数为

$$F(A,B,C,D)=(A+B)(C+\overline{D})(\overline{A}+B+C)$$

其中，$(A+B)$、$(C+\overline{D})$、$(\overline{A}+B+C)$均为“和”项，这些“和”项的“积”就构成了函数的“和之积”形式。

当然，布尔函数还可表示成其他形式。例如，上述四变量函数还可表示成

$$F(A,B,C,D)=(AC+B)(CD+\overline{D})=(AC+B)CD+(AC+B)\overline{D}$$

这种表示形式既不是“积之和”形式，又不是“和之积”形式，但它可以转换成后两种形式。

一个函数可以有多种表示形式，那么能不能找到统一的表示形式呢？有两种统一的标准形式。

3.3.2　函数的“标准积之和”与“标准和之积”形式

1. 标准积之和

所谓标准积，是指函数的积项包含了函数的全部变量，其中每个变量都以原变量或反变

量的形式出现，且仅出现一次。标准积项通常称为最小项。

一个函数可以用最小项之和的形式来表示，称为函数的“标准积之和”形式。例如，一个三变量函数为

$$F(A,B,C)=\overline{A}B\overline{C}+\overline{A}BC+A\overline{B}\overline{C}+ABC$$

它由 4 个最小项组成，这是函数的“标准积之和”形式。

由最小项的定义可知，3 个变量最多可组成 8 个最小项：$\overline{A}\overline{B}\overline{C}$、$\overline{A}\overline{B}C$、$\overline{A}B\overline{C}$、$\overline{A}BC$、$A\overline{B}\overline{C}$、$A\overline{B}C$、$AB\overline{C}$、$ABC$。为了叙述和书写方便，通常用 m_i 表示最小项，其下标 i 是这样确定的：把最小项中原变量记为 1，反变量记为 0，当变量顺序确定后，可以按顺序排列成一个二进制数。那么，与这个二进制数相对应的十进制数就是最小项的下标 i。表 3.6 列出了 3 个变量的全部最小项。

表 3.6 三变量的所有最小项和最大项

$A\quad B\quad C$	最小项	最大项
0 0 0	$m_0=\overline{A}\overline{B}\overline{C}$	$M_0=A+B+C$
0 0 1	$m_1=\overline{A}\overline{B}C$	$M_1=A+B+\overline{C}$
0 1 0	$m_2=\overline{A}B\overline{C}$	$M_2=A+\overline{B}+C$
0 1 1	$m_3=\overline{A}BC$	$M_3=A+\overline{B}+\overline{C}$
1 0 0	$m_4=A\overline{B}\overline{C}$	$M_4=\overline{A}+B+C$
1 0 1	$m_5=A\overline{B}C$	$M_5=\overline{A}+B+\overline{C}$
1 1 0	$m_6=AB\overline{C}$	$M_6=\overline{A}+\overline{B}+C$
1 1 1	$m_7=ABC$	$M_7=\overline{A}+\overline{B}+\overline{C}$

因此，上述函数 F(A,B,C)可以写成

$$F(A,B,C)=\overline{A}B\overline{C}+\overline{A}BC+A\overline{B}\overline{C}+ABC=m_2+m_3+m_4+m_7=\sum m(2,3,4,7)$$

其中，符号“$\sum$”表示各最小项求“或”，括号内的十进制数字表示各最小项的下标。

最小项有下列 3 个主要性质。

(1) 对于任意一个最小项，只有一组变量取值使其值为 1。

(2) 任意两个不同的最小项之积必为 0，即

$$m_i\cdot m_j=0\qquad(i\neq j)$$

(3) n 变量的所有 2^n 个最小项之和必为 1，即

$$\sum_{i=0}^{2^n-1}m_i=1$$

用展开定理可以证明，任一个 n 变量的函数都有一个且仅有一个最小项表达式，即“标准积之和”形式。下面介绍求函数的“标准积之和”的两种常用的方法。

方法一：代数演算法，即通过反复使用公式 $x+\overline{x}=1$ 和 $x(y+z)=xy+xz$ 而求得“标准积之和”的方法。例如，设 $F(A,B,C)=\overline{A}B+A\overline{B}\overline{C}+BC$，则得

$$\begin{aligned}F(A,B,C)&=\overline{A}B(C+\overline{C})+A\overline{B}\overline{C}+BC(A+\overline{A})\\&=\overline{A}BC+\overline{A}B\overline{C}+A\overline{B}\overline{C}+ABC+\overline{A}BC\end{aligned}$$

$$= \overline{A}B\overline{C} + \overline{A}BC + A\overline{B}\overline{C} + ABC = m_2 + m_3 + m_4 + m_7$$
$$= \sum m(2,3,4,7)$$

方法二：列表法，即列出函数的真值表，使函数取值为 1 的那些最小项，就构成了函数的“标准积之和”形式。例如，函数 $F(A,B,C)=\overline{A}B+A\overline{B}\overline{C}+BC$ 的真值表列于表 3.7。根据真值表可以很方便地写出函数的表达式为

$$F(A,B,C) = m_2 + m_3 + m_4 + m_7 = \sum m(2,3,4,7)$$

式中，m_2、m_3、m_4 和 m_7 是相应于真值表中使函数取值为 1 的那些最小项。

表 3.7　函数的真值表与最小项

A　B　C	$F(A,B,C)$	最小项
0　0　0	0	
0　0　1	0	
0　1　0	1	$m_2=\overline{A}B\overline{C}$
0　1　1	1	$m_3=\overline{A}BC$
1　0　0	1	$m_4=A\overline{B}\overline{C}$
1　0　1	0	
1　1　0	0	
1　1　1	1	$m_7=ABC$

2. 标准和之积

所谓标准和，是指函数的和项包含了全部变量，其中每个变量都以原变量或反变量形式出现，且仅出现一次。标准和项通常又称为最大项。

一个函数可以用最大项之积的形式表示，我们把这种形式称为函数的“标准和之积”形式。例如，一个三变量函数为

$$F(A,B,C) = (A+B+C)(A+B+\overline{C})(\overline{A}+B+\overline{C})(\overline{A}+\overline{B}+C)$$

它由 4 个最大项组成，这就是函数的“标准和之积”形式。

同样，3 个变量最多可组成 8 个最大项，如表 3.6 所示。通常，最大项用 M_i 来表示，其下标 i 是这样确定的：当最大项的各变量按一定次序排好后，把其中的原变量记为 0，反变量记为 1，便得到一个二进制数，与该二进制数相应的十进制数就是最大项的下标 i。

这样，上述函数 $F(A,B,C)$ 可以写成

$$F(A,B,C) = (A+B+C)(A+B+\overline{C})(\overline{A}+B+\overline{C})(\overline{A}+\overline{B}+C)$$
$$= M_0\,M_1\,M_5\,M_6 = \prod M(0,1,5,6)$$

其中，符号“Π”表示各最大项相“与”，括号内的十进制数表示各最大项的下标。

最大项具有下列 3 个主要性质：

(1) 对于任意一个最大项，只有一组变量取值可使其值为 0。

(2) 任意两个不同的最大项之和必为 1，即

$$M_i + M_j = 1 \quad (i \neq j)$$

(3) n 变量的所有 2^n 个最大项之积必为 0，即

$$\prod_{i=0}^{2^n-1} M_i = 0$$

同样地用展开定理可以证明,任何 n 变量的函数都有一个且仅有一个最大项表达式,即“标准和之积”形式。求函数的“标准和之积”的方法也有两种方法:

方法一:代数演算法,即通过反复地使用公式 $x \cdot \bar{x}=1$ 和 $x+yz=(x+y)(x+z)$ 而求得“标准和之积”的方法。例如

$$\begin{aligned}
F &= \bar{A}B + A\bar{B}\bar{C} + BC = (\bar{A}B + A)(\bar{A}B + \bar{B})(\bar{A}B + \bar{C}) + BC \\
&= (\bar{A} + A)(B + A)(\bar{A} + \bar{B})(B + \bar{B})(\bar{A} + \bar{C})(B + \bar{C}) + BC \\
&= (A + B)(\bar{A} + \bar{B})(\bar{A} + \bar{C})(B + \bar{C}) + BC \\
&= ((A + B)(\bar{A} + \bar{B})(\bar{A} + \bar{C})(B + \bar{C}) + B)((A + B)(\bar{A} + \bar{B})(\bar{A} + \bar{C})(B + \bar{C}) + C) \\
&= ((A + B + B)(\bar{A} + \bar{B} + B)(\bar{A} + \bar{C} + B)(B + \bar{C} + B)) \cdot \\
&\quad ((A + B + C)(\bar{A} + \bar{B} + C)(\bar{A} + \bar{C} + C)(B + \bar{C} + C)) \\
&= (A + B)(\bar{A} + B + \bar{C})(B + \bar{C})(A + B + C)(\bar{A} + \bar{B} + C)
\end{aligned}$$

由于表达式中第一项缺少变量 C,所以要加上 $C \cdot \bar{C}$;第三项缺少变量 A,所以要加 $A \cdot \bar{A}$,即

$$\begin{aligned}
F &= (A + B + C \cdot \bar{C})(\bar{A} + B + \bar{C})(A \cdot \bar{A} + B + \bar{C})(A + B + C)(\bar{A} + \bar{B} + C) \\
&= (A + B + C)(A + B + \bar{C})(\bar{A} + B + \bar{C})(\bar{A} + \bar{B} + C) \\
&= M_0 M_1 M_5 M_6 = \prod M(0,1,5,6)
\end{aligned}$$

可见,用这种方法是比较麻烦的。

方法二:列表法,即列出函数的真值表,那些使函数取值为 0 的最大项,就构成了函数的“标准和之积”形式。例如,上述函数的真值表列于表 3.8,根据真值表可以很方便地写出表达式为

$$F(A,B,C) = M_0 M_1 M_5 M_6 = \prod M(0,1,5,6)$$

表 3.8　函数的真值表与最大项

$A\ \ B\ \ C$	$F(A,B,C)$	最大项
0　0　0	0	$M_0=A+B+C$
0　0　1	0	$M_1=A+B+\bar{C}$
0　1　0	1	
0　1　1	1	
1　0　0	1	
1　0　1	0	$M_5=\bar{A}+B+\bar{C}$
1　1　0	0	$M_6=\bar{A}+\bar{B}+C$
1　1　1	1	

比较表 3.7 和表 3.8,可以得出两点结论:

(1) 同一个函数既可以表示成“标准积之和”的形式,又可表示成“标准和之积”的形式。

对于本例，有

$$F(A,B,C)=\bar{A}B+A\bar{B}\bar{C}+BC=\sum m(2,3,4,7)=\prod M(0,1,5,6)$$

（2）同一函数的最大项与最小项是互斥的，即如果真值表中的某一行作为函数的最小项，那么它就不可能是同一函数的最大项；反之亦然。一般有 $m_i=\overline{M_i}$ 或者 $M_i=\overline{m_i}$。换句话说，一个布尔函数的最小项的集合与它的最大项的集合，互为补集。因此，若已知一布尔函数的“标准积之和”形式，就可以很容易写出该函数的“标准和之积”形式。例如，已知函数的“标准积之和”形式为

$$F(A,B,C)=\sum m(1,3,4,6,7)$$

则该函数的“标准和之积”形式为

$$F(A,B,C)=\prod M(0,2,5)$$

3.4　不完全确定的布尔函数

前面所讨论的布尔函数都是属于完全确定的布尔函数。也就是说，它们的每一组输入变量的取值，都能得到一个完全确定的函数值(0 或 1)。如果布尔函数有 n 个变量，函数就有 2^n 个最小项(或最大项)，其中每一项都有确定的值。

在实际应用中，有时只要求某些最小项(或最大项)有确定的值，而对其余最小项(或最大项)的取值不感兴趣，它们既可为 0，也可为 1，即随意取值。通常，把这种可以随意取值的最小项(或最大项)称为随意项，或无关项，记为 d 。这种具有随意项的布尔函数称为不完全确定的布尔函数。

在数字系统的设计中，这种不完全确定的布尔函数是经常遇到的。例如，如果逻辑电路的输入是二进制编码的十进制数，4 位二进制输入共有 16 种不同的状态，其中只有 10 种是允许的，有确定的输出；而其余 6 种是不允许的，因此它们的输出结果是人们不关心的，换句话说，结果可以是任意的。在设计中，可以充分利用这些任意项，使设计得到简化。

【例 3.5】　设计一个奇偶判别电路，其输入为一位十进制的 BCD 码。当输入为偶数时，电路输出为 0；当输入为奇数时，电路输出为 1，如图 3.4 所示。

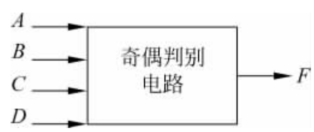


图 3.4　奇偶判别电路

根据设计要求可以列出描述该电路的布尔函数真值表，如表 3.9 所示。其中，第 10～15 行是不确定的，所以这是一个不完全确定的布尔函数，函数的表达式为：

$$F(A,B,C,D)=\sum m(1,3,5,7,9)+\sum d(10,11,12,13,14,15)$$

式中，d 表示随意项，可取 0，也可取 1。

表 3.9 BCD 码奇偶判别电路真值表

十进制数 r	BCD A	BCD B	BCD C	BCD D	$F(A,B,C,D)$
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	1	0	0
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	1
6	0	1	1	0	0
7	0	1	1	1	1
8	1	0	0	0	0
9	1	0	0	1	1
10	1	0	1	0	d
11	1	0	1	1	d
12	1	1	0	0	d
13	1	1	0	1	d
14	1	1	1	0	d
15	1	1	1	1	d

为了使设计的电路简单，可以将函数化简。下面分两种情况来考虑。

(1) 如果不考虑随意项(即取 $d=0$)，则有

$$
\begin{aligned}
F(A,B,C,D) &= \sum m(1,3,5,7,9) \\
&= \overline{A}\overline{B}\overline{C}D + \overline{A}\overline{B}CD + \overline{A}B\overline{C}D + \overline{A}BCD + A\overline{B}\overline{C}D \\
&= \overline{A}\overline{B}D(C+\overline{C}) + \overline{A}BD(C+\overline{C}) + \overline{B}\overline{C}D(\overline{A}+A) \\
&= \overline{A}\overline{B}D + \overline{A}BD + \overline{B}\overline{C}D \\
&= \overline{A}D + \overline{B}\overline{C}D \qquad (3.1)
\end{aligned}
$$

(2) 如果考虑随意项，且取随意项 11、13、15 的值为 1，其余随意项取值为 0，则有

$$
\begin{aligned}
F(A,B,C,D) &= \sum m(1,3,5,7,9) + \sum_{d=1} d(11,\ 13,\ 15) \\
&= \sum m(1,3,5,7,9,11,13,15) = D \qquad (3.2)
\end{aligned}
$$

式(3.1)比式(3.2)要简单得多。由此可见，对随意项合理赋值，可以使函数大为简化。因而其相应的逻辑电路也简单了。

3.5 布尔函数的化简

由 3.3 节讨论知道，同一个布尔函数可以有多种表示形式。一种形式的函数表达式相应于一种逻辑电路，尽管它们的形式不同，但其逻辑功能是相同的。函数表达式有简有繁，相应的逻辑电路也有简有繁，人们希望用尽可能少的逻辑门来完成同样的逻辑功能，这就要求函数表达式是最简单的。因此，如何使函数的表达式最简单，即函数的化简，成为逻辑设

计的一个关键问题。因为函数越简单,所设计的电路就不仅简单、经济,而且出现故障的可能性也越少,可靠性就越好。

在函数的各种不同形式的表达式中,“与或”表达式是最基本的,其他形式的表达式都可由它变换而得。例如

$$
\begin{aligned}
F(A,B,C) &= \bar{A}\bar{B} + AC && \text{(“与或”形式)} \\
&= (\bar{A}+C)(A+\bar{B}) && \text{(“或与”形式)} \\
&= \overline{\overline{\bar{A}\bar{B}} \cdot \overline{AC}} && \text{(“与非”形式)} \\
&= \overline{\overline{\bar{A}+C} + \overline{A+\bar{B}}} && \text{(“或非”形式)} \\
&= \overline{A\bar{C}+\bar{A}B} && \text{“与或非”形式}
\end{aligned}
$$

因此,将从“与或”表达式出发来讨论函数的化简方法。

什么是函数的最简“与或”式呢？一个最简“与或”式应同时满足以下两个条件。

(1) 该式中的与项最少。

(2) 该式中的每个与项的变量也最少。

这样,用逻辑门来实现布尔函数时,所需的与门数目最少,而且每个与门的输入端数目也最少。

本节讨论布尔函数的几种常用的化简方法,以及用不同门电路来实现布尔函数的方法。这些方法是组合逻辑网络设计的基础,应熟练掌握。

3.5.1 代数化简法

所谓代数化简法,就是运用布尔代数的基本公式、定理和规则来化简布尔函数的一种方法。这种方法没有固定的步骤可以遵循,主要是凭对布尔代数的公式、定理和规则的熟练运用程度。尽管如此,我们还是可以总结出一些适用于大多数情况的方法。这些方法的基本思想是对布尔函数进行等式变换,使表达式的与项减少,或使与项中的变量减少,以达到简化函数的目的。下面介绍化简“与或”表达式的几种常用的方法。

1. 并项法

利用公式 $AB+A\bar{B}=A$,将两项合并为一项,并消去一个变量。例如

$$A\bar{B}C + A\bar{B}\bar{C} = A\bar{B}$$

$$AB\bar{C} + A\,\overline{B\bar{C}} = A$$

2. 吸收法

利用吸收律 $A+AB=A$,消去多余的项。例如

$$\bar{B} + A\bar{B}D = \bar{B}$$

$$A\bar{B} + A\bar{B}CD(E+F) = A\bar{B}$$

还可以利用吸收律 $A+\bar{A}B=A+B$,消去多余的变量。例如

$$\bar{A} + AB + DE = \bar{A} + B + DE$$

$$AB + \bar{A}C + \bar{B}C = AB + (\bar{A}+\bar{B})C = AB + \overline{AB}C = AB + C$$

3. 配项法

利用 $A\cdot 1=A$ 和 $A+\overline{A}=1$，为某一项配上其所缺的一个变量，以便用其他方法进行化简。例如

$$AB+\overline{A}C+BC=AB+\overline{A}C+(A+\overline{A})BC=AB+\overline{A}C+ABC+\overline{A}BC$$
$$=(AB+ABC)+(\overline{A}C+\overline{A}BC)=AB+\overline{A}C \qquad \text{（吸收法）}$$

还可以利用公式 $A+A=A$，为某项配上其所能合并的项。例如

$$ABC+AB\overline{C}+A\overline{B}C+\overline{A}BC$$
$$=(ABC+AB\overline{C})+(ABC+A\overline{B}C)+(ABC+\overline{A}BC)$$
$$=AB+AC+BC$$

4. 消去冗余项法

等式 $AB+\overline{A}C+BC=AB+AC$ 可以作为一个基本公式使用，它称为包含律，其中 BC 称为冗余项。利用这个公式，去寻找一个表达式的冗余项，然后消去冗余项。例如

$$A\overline{B}+AC+ADE+\overline{C}D=A\overline{B}+(AC+\overline{C}D+ADE)=A\overline{B}+AC+\overline{C}D$$
$$AB+\overline{B}C+AC(D+E)=AB+\overline{B}C$$

以上介绍了几种常用的方法。在实际应用中可能遇到比较复杂的函数，只要熟练掌握布尔代数的公式和定理，灵活运用上述方法，总能找到化简的办法。下面是几个综合运用上述方法化简布尔函数的实例。

【例 3.6】 化简 $F=AD+A\overline{D}+AB+\overline{A}C+BD+ACEF+\overline{B}EF+DEFG$

$$F=AD+A\overline{D}+AB+\overline{A}C+BD+ACEF+\overline{B}EF+DEFG$$
$$=A+AB+\overline{A}C+BD+ACEF+\overline{B}EF+DEFG \qquad \text{（并项法）}$$
$$=A+\overline{A}C+BD+\overline{B}EF+DEFG \qquad \text{（吸收法）}$$
$$=A+C+BD+\overline{B}EF+DEFG \qquad \text{（吸收法）}$$
$$=A+C+BD+\overline{B}EF \qquad \text{（消去冗余项）}$$

【例 3.7】 化简 $F=A\overline{B}+B\overline{C}+\overline{B}C+\overline{A}B$

$$F=A\overline{B}+B\overline{C}+\overline{B}C+\overline{A}B$$
$$=A\overline{B}+B\overline{C}+(A+\overline{A})\overline{B}C+\overline{A}B(C+\overline{C}) \qquad \text{（配项法）}$$
$$=(A\overline{B}+A\overline{B}C)+(B\overline{C}+\overline{A}B\overline{C})+(\overline{A}\overline{B}C+\overline{A}BC)$$
$$=A\overline{B}+B\overline{C}+\overline{A}C \qquad \text{（吸收法）}$$

【例 3.8】 化简或与表达式

$$F=(\overline{B}+D)(\overline{B}+D+A+G)(C+E)(\overline{C}+G)(A+E+G)$$

可以利用对偶规则，先求出 F 的对偶式：

$$F_d=\overline{B}D+\overline{B}DAG+CE+\overline{C}G+AEG$$

然后，利用与或式的化简方法进行化简，则得

$$F_d=\overline{B}D+CE+\overline{C}G$$

最后，对 F_d 再求对偶式，则得

$$F=(F_d)_d=(\overline{B}+D)(C+E)(\overline{C}+G)$$

由本例可见，与或表达式的化简是基础，其他形式的表达式都可先变换为与或表达式进

行化简，然后再变换成所需的形式。

从以上的例子可以看出，代数化简法不仅使用不便，而且难以判断所得之结果是否为最简。因此，代数化简法一般适用于函数表达式较为简单的情况。当函数较为复杂时，往往采用比较方便的、有规则的卡诺图法。

3.5.2　卡诺图化简法

卡诺图化简法是将布尔函数用卡诺图来表示，在卡诺图上进行函数的化简的方法。这是一种很简单、直观的方法。

1. 卡诺图的构成

卡诺图是真值表的图形化。我们可以把真值表看成是由最小项构成的一个纵列。一个函数由若干个最小项构成，则在相应的最小项上填 1，其余填 0。

例如，设有函数 $F(A,B,C)=\sum m(2,3,5,7)$，该函数的真值表如表 3.10 所示。

该函数可以进行化简，即

$$\begin{aligned} F(A,B,C) &= \overline{A}B\overline{C}+\overline{A}BC+A\overline{B}C+ABC \\ &= \overline{A}B(\overline{C}+C)+AC(\overline{B}+B) \\ &= \overline{A}B+AC \end{aligned}$$

其中，最小项 $\overline{A}B\overline{C}$ 和 $\overline{A}BC$（或者 $A\overline{B}C$ 和 ABC）只有一个变量互补，其余变量相同，称这样的最小项为相邻最小项，它们可以合并消去一个变量。因此，化简的实质是两个相邻最小项的合并。

表 3.10　$F(A,B,C)=\sum m(2,3,5,7)$ 的真值表

$A\ \ B\ \ C$	$F(A,B,C)$	最小项
0　0　0	0	m_0
0　0　1	0	m_1
0　1　0	1	m_2
0　1　1	1	m_3
1　0　0	0	m_4
1　0　1	1	m_5
1　1　0	0	m_6
1　1　1	1	m_7

从真值表上很难直观地看出最小项的相邻关系的。例如，相邻最小项 m_5 和 m_7 在位置上并不相邻。那么，能不能把最小项排成一种图，可以直观地从图上看出最小项的相邻关系呢？可以的。从 Gray 码的特点知道，两个相邻代码之间只有一位不同。因此，只要把真值表中的最小项重新排列，把它们排列成矩阵形式，并且使矩阵的横方向和纵方向的布尔变量的取值按 Gray 码的顺序排列，这样构成的图形就是卡诺图。图 3.5 表示了二变量、三变量和四变量的卡诺图的构成。

从卡诺图可以看出，任意两个相邻的最小项在图上是相邻的。并且，图中最左列的最小

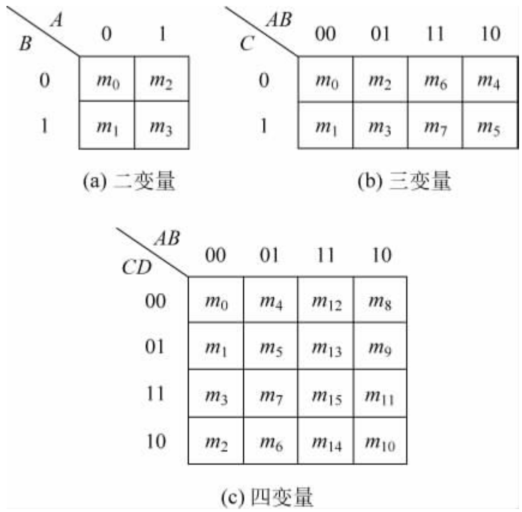


图 3.5 卡诺图的构成

项与最右列的相应最小项也是相邻的；位于最上面一行的最小项与最下面一行的相应最小项也是相邻的。因此，每个二变量的最小项有两个最小项与它相邻；每个三变量的最小项有 3 个最小项与它相邻；每个四变量的最小项有 4 个最小项与它相邻。可以证明每个 n 变量的最小项有 n 个最小项与它相邻。

五变量的卡诺图如图 3.6 所示。图中方格内的数字为最小项的下标。它可以看成是由两个四变量的卡诺图构成的，只要把右边部分的四变量卡诺图重合到左边部分的四变量卡诺图上来，就可找出各最小项的相邻关系。例如，最小项 3 与最小项 1、2、7、11 及 19 是相邻的。

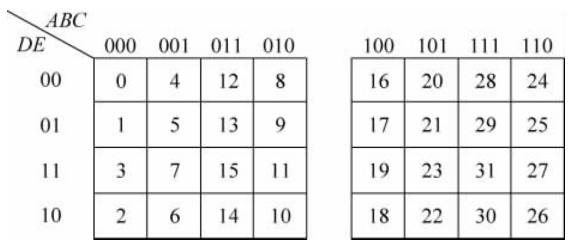


图 3.6 五变量卡诺图

六变量卡诺图如图 3.7 所示，它可以看成是由 4 个四变量卡诺图构成的。在寻找各最小项的相邻关系时，除了要注意左、右两个四变量卡诺图的重合关系外，还要注意上、下两个四变量卡诺图的重合关系。例如，最小项 3 除与最小项 1、2、7、11、19 相邻外，还与最小项 35 相邻。

变量多于 6 个时，卡诺图就显得很庞大，在实际应用中已失去它的优越性，一般就很少用它了。

2. 布尔函数在卡诺图上的表示

从卡诺图的构成方法可知，卡诺图实际是真值表的重新排列，使最小项排列得更紧凑、

DEF \ ABC	000	001	011	010	100	101	111	110
000	0	4	12	8	16	20	28	24
001	1	5	13	9	17	21	29	25
011	3	7	15	11	19	23	31	27
010	2	6	14	10	18	22	30	26
100	32	36	44	40	48	52	60	56
101	33	37	45	41	49	53	61	57
111	35	39	47	43	51	55	63	59
110	34	38	46	42	50	54	62	58

图 3.7　六变量卡诺图

更便于化简。因此，根据卡诺图与真值表的对应关系，就可以知道布尔函数在卡诺图上的表示方法。

如果布尔函数是以真值表的形式或者以“标准积之和”的形式给出的，我们只要在卡诺图上找出那些与给定布尔函数的最小项相对应的方格，并标以 1，就得到该函数的卡诺图。例如，三变量函数

$$F(A,B,C)=\sum m(2,3,5,7)$$

其卡诺图如图 3.8 所示。

如果布尔函数是“与或”表达式，则要将各“与项”分别标在卡诺图上。例如，给定函数的表达式为

$$F(A,B,C)=AB+A\overline{C}$$

只要在 AB 为 11 的列标以 1，并在 A 为 1、C 为 0 的对应方格中标上 1，便可得到如图 3.9 所示的卡诺图。

C \ AB	00	01	11	10
0	0	1	0	0
1	0	1	1	1

图 3.8　$F(A,B,C)=\sum m(2,3,5,7)$ 的卡诺图

C \ AB	00	01	11	10
0	0	0	1	1
1	0	0	1	0

图 3.9　$F(A,B,C)=AB+A\overline{C}$ 的卡诺图

又如，给定四变量函数为

$$F(A,B,C,D)=\overline{A}BD+\overline{B}\overline{D}+\overline{A}\overline{B}D$$

其卡诺图如图 3.10 所示。

CD \ AB	00	01	11	10
00	1	0	0	1
01	1	1	0	0
11	1	1	0	0
10	1	0	0	1

图 3.10　$F(A,B,C,D)=\overline{A}BD+\overline{B}\overline{D}+\overline{A}\overline{B}D$ 的卡诺图

3. 卡诺图的性质

卡诺图化简布尔函数的基本原理是基于卡诺图的性质。前面已指出，化简的实质是相邻最小项的合并。卡诺图的一个明显优点是它能利用人的直观的阅图能力，方便

地表示出所有相邻最小项。卡诺图具有下列性质。

性质 1　卡诺图上任何两个(2^1个)标 1 的相邻最小项,可以合并为一项,并消去一个变量。

例如,在图 3.10 中,最小项 $m_1=\overline{A}\overline{B}\overline{C}D$ 和 $m_3=\overline{A}\overline{B}CD$ 相邻,所以它们可以合并;最小项 $m_5=\overline{A}B\overline{C}D$ 和 $m_7=\overline{A}BCD$ 相邻,所以它们也可以合并。即有

$$\overline{A}\overline{B}\overline{C}D+\overline{A}\overline{B}CD=\overline{A}\overline{B}D,\quad \overline{A}B\overline{C}D+\overline{A}BCD=\overline{A}BD$$

性质 2　卡诺图上任何 4 个(2^2个)标 1 的相邻最小项,可以合并为一项,并消去两个变量。

例如,在图 3.10 中,最小项 m_1、m_3、m_5和 m_7彼此相邻,这 4 个最小项可以合并。即有

$$(m_1+m_3)+(m_5+m_7)=\overline{A}\overline{B}D+\overline{A}BD=\overline{A}D$$

这种合并,在卡诺图中表示为把 4 个 1 圈在一起。

在四变量卡诺图中,4 个标 1 的小方格相邻的典型情况如图 3.11 所示。

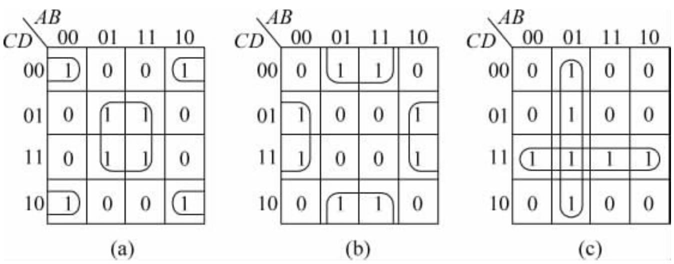


图 3.11　4 个相邻最小项合并的情况

在图 3.11(a)中,最小项 m_0、m_2、m_8、m_{10}相邻,而最小项 m_5、m_7、m_{12}、m_{15}也相邻。它们合并后,可得函数表达式为

$$F(A,B,C,D)=\overline{B}\overline{D}+BD$$

根据同样道理,将图 3.11(b)中的相邻最小项合并后,可得函数表达式为

$$F(A,B,C,D)=B\overline{D}+\overline{B}D$$

将图 3.11(c)中的相邻最小项合并后,结果为

$$F(A,B,C,D)=\overline{A}B+CD$$

性质 3　任何 8 个(2^3个)标 1 的相邻最小项,可合并为一项,并消去 3 个变量。

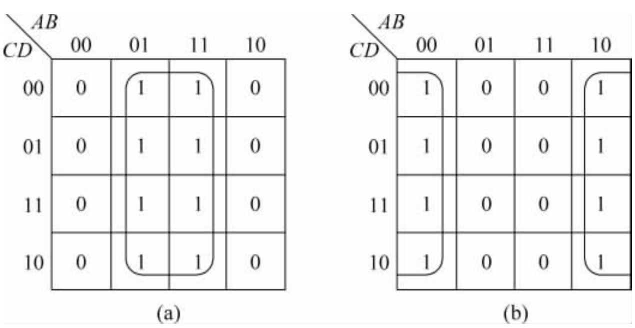


图 3.12　8 个相邻最小项合并的情况

图 3.12 表示了 8 个相邻最小项合并的情况。其中，图 3.12(a)所示的 8 个最小项合并后，结果为

$$F(A,B,C,D)=B$$

图 3.12(b)中的 8 个最小项合并后，结果为

$$F(A,B,C,D)=\bar{B}$$

由上述性质可知，相邻最小项的数目必须为 2^i 个才能合并成一项，并消去 i 个变量。包含的最小项数目越多，消去的变量也越多，从而所得到的逻辑表达式就越简单。

4. 卡诺图化简的基本步骤

在讨论用卡诺图化简的具体方法之前，先定义几个概念。

蕴涵项　在函数的任何与或表达式中，每个与项称为该函数的蕴涵项(Implicant)。显然，在函数的卡诺图中，任一标 1 的最小项以及由 2^i 个相邻最小项所形成的圈都是函数的蕴涵项。

质蕴涵项　如果函数的某一蕴涵项不是该函数中其他蕴涵项的一个子集，则此蕴涵项称为质蕴涵项(Prime Implicant)。从卡诺图上看，所谓质蕴涵项，就是大得不能再大的圈。例如，在图 3.13 中，BD、$A\bar{C}\bar{D}$ 和 $AB\bar{C}$ 都是质蕴涵项，而 BCD、$\bar{A}B\bar{C}D$ 等都不是质蕴涵项。

必要质蕴涵项　如果函数的一个质蕴涵项，至少包含了一个其他任何质蕴涵项都不包含的标 1 最小项，则此质蕴涵项称为必要质蕴涵项(Essential Prime Implicant). 例如，在图 3.13 中，BD、$A\bar{C}\bar{D}$ 是必要质蕴涵项，而 $AB\bar{C}$ 就不是必要质蕴涵项。

根据以上定义，可以给出用卡诺图化简布尔函数的基本步骤如下。

(1) 将布尔函数正确地标到卡诺图上，并在图上找出所有质蕴涵项。

(2) 求出所有必要质蕴涵项。

(3) 求函数的最小覆盖(即函数的最简表达式)。

下面通过具体例子来说明上述步骤。

【例 3.9】　用卡诺图法化简函数

$$F(A,B,C,D)=\sum m(0,3,4,5,7,11,13,15)$$

(1) 作 F 的卡诺图，并求得所有质蕴涵项为 BD、CD、$\bar{A}\bar{C}\bar{D}$、$\bar{A}B\bar{C}$，如图 3.14 所示。

(2) 求出所有必要质蕴涵项为 CD、BD、$\bar{A}\bar{C}\bar{D}$。

(3) 由于必要质蕴涵项的集合已覆盖了函数的所有最小项，因此函数的最简与或式为

$$F(A,B,C,D)=BD+CD+\bar{A}\bar{C}\bar{D}$$

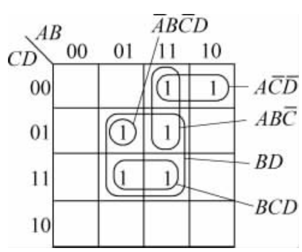


图 3.13　卡诺图中的质蕴涵项

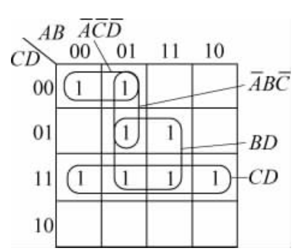


图 3.14　例 3.9 的卡诺图

【例 3.10】 用卡诺图化简布尔函数

$$F(A,B,C,D)=\sum m(0,5,6,8,15)+\sum d(1,2,3,7,10,12,13)$$

这是含有随意项 d 的情况。利用随意项求质蕴涵时，如果它对化简有利，则取 $d=1$；如果它对化简不利，则取 $d=0$。

(1) 作函数的卡诺图，求出所有质蕴涵项为 $\overline{A}\overline{B}$、$\overline{A}D$、$\overline{A}C$、BD、$\overline{B}\overline{D}$。如图 3.15 所示。

(2) 求出所有必要质蕴涵项为 $\overline{A}C$、BD、$\overline{B}\overline{D}$。注意，在求必要质蕴涵时，只要考虑 1 的覆盖。

(3) 求函数的最小覆盖。本例中必要质蕴涵已覆盖了卡诺图中所有标 1 的最小项。因此，布尔函数的最简与或式为

$$F(A,B,C,D)=\overline{A}C+BD+\overline{B}\overline{D}$$

【例 3.11】 用卡诺图化简布尔函数

$$F(A,B,C,D)=\sum m(0,4,6,7,8,9,11,12,13,15)$$

(1) 求出所有质蕴涵项为 $\overline{C}\overline{D}$、$A\overline{C}$、AD、$\overline{A}B\overline{D}$、$\overline{A}BC$、$BCD$，如图 3.16 所示。

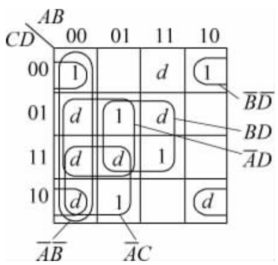


图 3.15　例 3.10 的卡诺图

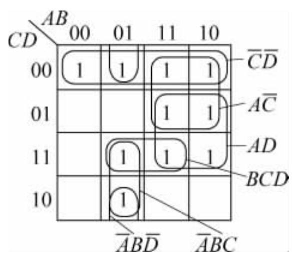


图 3.16　例 3.11 的卡诺图

(2) 求出所有必要质蕴涵项为 $\overline{C}\overline{D}$、AD。

(3) 求函数的最小覆盖。本例中必要质蕴涵只覆盖了函数的 8 个最小项，还剩下两个最小项 m_6、m_7 未被覆盖。由观察可知，在几个非必要质蕴涵中，选择 $\overline{A}BC$ 这一项即可覆盖 m_6、m_7。因此，该函数的最简与或式为

$$F(A,B,C,D)=AD+\overline{C}\overline{D}+\overline{A}BC$$

这说明，在一个最简与或式中，并非每个质蕴涵项都是必要的，其中可能包含所选择的质蕴涵项。这一点应引起注意。

必须指出，由于卡诺图化简法带有试凑性质，因此，当读者已对卡诺图应用自如时，就不必按上述步骤去做，可以在卡诺图上一次画出最小覆盖。

【例 3.12】 用卡诺图化简函数

$$F(A,B,C,D)=\sum m(0,1,3,4,7,12,13,15)$$

作出 F 的卡诺图如图 3.17(a)所示。该函数有 8 个质蕴涵项，它们相互交连，找不出哪个是必要质蕴涵项，这种情况通常称为循环结构。对于这类循环结构，通常可选取一个最大的质蕴涵圈作为必要质蕴涵，以打破循环结构。本例中，由于各个质蕴涵圈的大小相等，故可任选一个质蕴涵作为必要质蕴涵项。图 3.17(b)为其中的一种解，可得 F 的最简表达式为

$$F(A,B,C,D)=\overline{A}\overline{B}\overline{C}+\overline{A}CD+ABD+B\overline{C}\overline{D}$$

同理,可得另一个最简表达式为

$$F(A,B,C,D)=\overline{A}\overline{B}D+BCD+AB\overline{C}+\overline{A}\overline{C}D$$

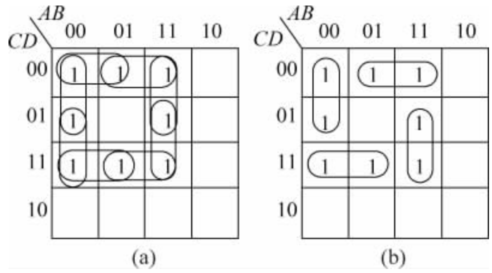


图 3.17　例 3.12 的卡诺图

【例 3.13】　化简五变量函数

$$F(A,B,C,D,E)=\sum m(2,4,5,6,7,12,13,18,20,21,22,23,24,25,28,29)$$

五变量布尔函数可用两个四变量卡诺图来表示,相重叠的最小项是相邻的,可以合并。画出函数 F 的卡诺图如图 3.18 所示。重叠部分的质蕴涵为 $\overline{B}D\overline{E}$、$\overline{B}C$ 和 $C\overline{D}$;不重叠的质蕴涵为 $AB\overline{D}$。这几个都为必要质蕴涵,且覆盖了函数的全部标 1 最小项。因此,函数的最简表达式为

$$F(A,B,C,D,E)=\overline{B}D\overline{E}+\overline{B}C+C\overline{D}+AB\overline{D}$$

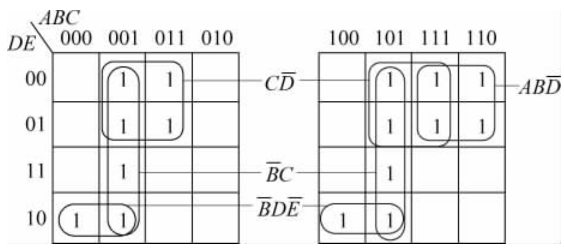


图 3.18　例 3.13 的卡诺图

由上述例子可见,用卡诺图化简布尔函数,简单明了,形象直观,容易掌握。

3.5.3　列表化简法

上节介绍的卡诺图化简法是手算常用的方法,它的缺点是:由于要靠人对图形的识别能力,因此不便于机器实现;而且,当函数的变量增多时(如多于 6 个),这种方法就逐渐失去它的优越性。

本节介绍一种更有规律的、系统的方法,即列表化简法。这一方法是由 W. V. Quine 和 E. J. McCluskey 在 1956 年研究发表的,故也称奎恩-麦克拉斯基法,简称 Q-M 法。这种方法的基本步骤与卡诺图法是相同的,即先找出布尔函数的所有质蕴涵,然后找出其中的必要质蕴涵,最后求函数的最小覆盖。所不同的是,列表化简法完成上述步骤是通过约定形式的表格,按照一定规则求得的。下面分别讨论列表法的这几个步骤。

1. 用列表法确定布尔函数的所有质蕴涵项

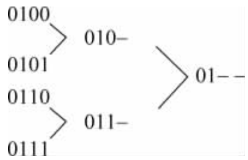
用列表法找布尔函数的所有质蕴涵项的基本思想是这样的：首先将布尔函数用最小项来表示，并把最小项表示成二进制数形式。例如

m_4：$\overline{A}B\overline{C}\,\overline{D}$ 表示成 0100。

m_5：$\overline{A}B\overline{C}D$ 表示成 0101。

然后进行两两比较，如果两个二进制数只有一位不同，就可合并。例如

其中，"－"表示该位置上的变量被消去，同理，对于带有"－"的两个二进制数，若只有一位不相同(0 与 1)，则此二进制数又可进一步合并，得到带有两个"－"的二进制数。例如：



就这样逐次合并只有一位数值不同的两个二进制数，所得到的不能再合并的二进制数，其对应的乘积项，即为质蕴涵项。

下面举例说明用列表法求质蕴涵的过程。

【例 3.14】 求下列函数的全部质蕴涵项：

$$F(A,B,C,D)=\sum m(1,3,4,5,10,11,12,13)$$

为便于对照，以弄懂合并的原理，不妨先给出卡诺图上合并的情况，如图 3.19 所示。

用 Q-M 法求质蕴涵过程如下：

第一步，列出最小项的二进制数形式，如表 3.11 所示。

表 3.11 最小项的二进制形式

m_i	A	B	C	D
1	0	0	0	1
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1

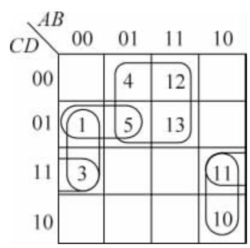


图 3.19 例 3.14 的卡诺图

可以看出，两个二进制数可以合并有如下规律：只有二进制数中 1 的个数相差 1 的两数才能合并，而 1 的个数相同或相差 2 和 2 以上的两数不能合并。这一规律启示我们，如果把上述二进制数按 1 的个数分组，那么，只有在相邻组之间才有合并的可能性，组内和隔组之间则不必考虑合并可能。这样可以大大地减少合并的工作量。

第二步,按 1 的个数分组列表,并在相邻组之间进行搜索合并。凡是能合并的两个项,则在其后面打"√",表示它们不是质蕴涵项,并将合并结果列于另一栏中。如此继续,直到无法合并为止。最后,那些没有打"√"的项,就是质蕴涵项 P_i。以上过程如表 3.12 所示。

最后,求得全部质蕴涵项为

$$P_1 = B\overline{C}, \quad P_2 = \overline{A}\overline{B}D, \quad P_3 = \overline{A}\overline{C}D, \quad P_4 = \overline{B}CD, \quad P_5 = A\overline{B}C$$

该结果与图 3.19 卡诺图上合并结果相同。

表 3.12　求质蕴涵项 P_i

m_i	A	B	C	D		m_i	A	B	C	D		m_i	A	B	C	D	
1	0	0	0	1	√	1,3	0	0	—	1	P_2	4,5	—	1	0	—	P_1
4	0	1	0	0	√	1,5	0	—	0	1	P_3	12,13					
3	0	0	1	1	√	4,5	0	1	0	—	√						
5	0	1	0	1	√	4,12	—	1	0	0	√						
10	1	0	1	0	√	3,11	—	0	1	1	P_4						
12	1	1	0	0	√	5,13	—	1	0	1	√						
11	1	0	1	1	√	10,11	1	0	1	—	P_5						
13	1	1	0	1	√	12,13	1	1	0	—	√						

2. 用质蕴涵表确定必要质蕴涵

所谓质蕴涵表,就是函数的各个质蕴涵覆盖函数的相应最小项的情况表。下面通过例子来说明如何列质蕴涵表,以及如何用质蕴涵表来确定必要质蕴涵。

【例 3.15】　已知函数

$$F(A,B,C,D) = \sum m(1,3,4,5,10,11,12,13)$$

求得它的全部质蕴涵为 $B\overline{C}$、$\overline{A}\overline{B}D$、$\overline{A}\overline{C}D$、$\overline{B}CD$、$A\overline{B}D$。列质蕴涵表如表 3.13 所示。表中,各纵列代表函数所包含的最小项,各横行是函数的质蕴涵项。每一个质蕴涵项所覆盖的最小项,都在表中行列交叉处用记号"×"标出,例如,质蕴涵项 $B\overline{C}$ 可覆盖最小项 4、5、12、13。因此,在 $B\overline{C}$ 这一行的最小项 4、5、12、13 列下标上"×"号。其他各行按同样方法标上记号。

表 3.13　质蕴涵表

P_j \ m_i	1	3	4	5	10	11	12	13
√$B\overline{C}$			$\otimes$	×			$\otimes$	$\otimes$
$\overline{A}\overline{B}D$	×	×						
$\overline{A}\overline{C}D$	×			×				
$\overline{B}CD$		×				×		
√$A\overline{B}D$					$\otimes$	×		
覆盖情况			×	×	×	×	×	×

必要质蕴涵是按下列步骤求得的:

(1) 逐列检查表 3.13 中标有"×"号的情况,凡只标有一个"×"号的列,则在该"×"的

外面打一个圈，即⊗。例如，表中最小项 4、10、12、13 各列都只有一个“× ”，故都加上圈。

（2）找出包含有⊗号的各行，这些行的质蕴涵项就是必要质蕴涵项，并在其前加上标记“√”。例如，表中 $B\overline{C}$ 和 $A\overline{B}D$ 就为必要质蕴涵项。

（3）在表的最后一行覆盖情况一栏中，标上必要质蕴涵项覆盖最小项的情况。凡能被必要质蕴涵项覆盖的最小项，在最后一行的该列上打“×”号。

由表 3.13 可知，本例中必要质蕴涵项 $B\overline{C}$ 和 $A\overline{B}D$，没有覆盖 F 的全部最小项。因此，接着要做的下一步是求 F 的最小覆盖。

3. 求函数的最小覆盖

一般来说，必要质蕴涵覆盖函数的最小项的情况有两种可能：一种是必要质蕴涵已覆盖了函数的所有最小项，对于这种情况，函数的简化工作就已完成，即函数的最小覆盖就是所有必要质蕴涵项。另一种是必要质蕴涵没有覆盖函数的所有最小项，对于这种情况，还需要选择适当的质蕴涵项，以覆盖剩下的最小项。那么，覆盖剩余的最小项所需的质蕴涵应该怎样选择才能使函数表达式最简呢？下面介绍两种常用的选取所需质蕴涵的方法。

1）行列消去法

在质蕴涵表中，去掉必要质蕴涵项和已被其覆盖的最小项，剩下的部分称为简化的质蕴涵表。下面的两个规则——优势行规则和优势列规则，能从简化的质蕴涵表中选取所需的质蕴涵。

① 优势行规则。设质蕴涵 P_i 和 P_j 是简化质蕴涵表中的两行，其中 P_j 行中的“×”完全包含在 P_i 行中，则称 P_i 为优势行，P_j 为劣势行，记作 $P_i \supset P_j$。这时，在简化质蕴涵表中可以消去劣势行 P_j（这是因为选取了优势行 P_i 后，不仅可覆盖劣势行 P_j 所覆盖的最小项，而且还可覆盖其他最小项）。

例如，由表 3.13 去掉必要质蕴涵项和被它所覆盖的最小项，得到简化的质蕴涵表如表 3.14 所示。表中，P_3 相对于 P_2 来说是劣势行，即 $P_2 \supset P_3$，可消去 P_3；同理，P_4 相对于 P_2 来说也是劣势行，即 $P_2 \supset P_4$，可消去 P_4。最后只剩下 P_2。

表 3.14 简化的质蕴涵表

P_j \ m_i	1	3
P_2	×	×
P_3	×	
P_4		×

表 3.15 优势列举例

P_j \ m_i	m_1	m_2	m_3
P_1	×	×	
P_2	×		×
P_3	×		

② 优势列规则。设最小项 m_i 和 m_j 是简化质蕴涵表中的两列，其中 m_j 列中的“×”完全包含在 m_i 列之中，则称 m_i 为优势列，m_j 为劣势列，记作 $m_i \supset m_j$。这时，在简化质蕴涵表中可以消去优势列 m_i（这是因为选取了覆盖劣势列的质蕴涵后，一定能覆盖优势列，反之则不一定）。

例如，设表 3.15 为简化质蕴涵表，表中的 m_2 和 m_3 相对于 m_1 来说是劣势列，即 $m_2 \subset m_1$，$m_3 \subset m_1$，故可消去优势列 m_1。

这种优势行规则和优势列规则可以反复交替使用，使简化的质蕴涵表进一步简化，以便

求得所需的质蕴涵项。

【例 3.16】 用行列消去法求所需质蕴涵项，设简化的质蕴涵表如表 3.16 所示。

表中，根据优势行规则，有 $P_4 \supset P_5$，$P_3 \supset P_6$，故可消去 P_5 和 P_6；又根据优势列规则，由于 $m_4 \subset m_6$，$m_{10} \subset m_2$，故可消去 m_2 和 m_6。最后，求得所需质蕴涵为 P_3 和 P_4，它们覆盖了简化质蕴涵表的所有最小项 m_2、m_4、m_6 和 m_{10}。

表 3.16　简化的质蕴涵表

P_j \ m_i	m_2	m_4	m_6	m_{10}
P_2	×		×	
P_3	×			×
P_4		×	×	
P_5		×		
P_6				×

2）布尔代数法

所谓布尔代数法，就是从简化质蕴涵表列出布尔表达式，从中选出最简的所需质蕴涵项。

例如，对于表 3.14 的简化质蕴涵表，要覆盖最小项 m_1，可选取质蕴涵项 P_2 或 P_3；要覆盖最小项 m_3，可选取质蕴涵项 P_2 或 P_4。因此，若要同时覆盖 m_1 和 m_3，可以用如下布尔表达式来表示：

$$(P_2 + P_3)(P_2 + P_4) = P_2 + P_2 P_4 + P_2 P_3 + P_3 P_4 = P_2 + P_3 P_4$$

该式表明，同时覆盖 m_1 和 m_3 的方案有两个，即选用 P_2 或选用 $P_3 P_4$，其中选用 P_2 最简单；该结果与行列消去法所得结果相同。

又如，对于表 3.16 的简化质蕴涵表，若要同时覆盖最小项 m_2、m_4、m_6 和 m_{10}，则有

$$\begin{aligned}(P_2 + P_3)(P_4 + P_5)(P_2 + P_4)(P_3 + P_6) &= (P_3 + P_2 P_6)(P_4 + P_2 P_5)\\ &= P_3 P_4 + P_2 P_3 P_5 + P_2 P_4 P_6 + P_2 P_5 P_6\end{aligned}$$

该式表明，可供选取的所需质蕴涵集有 4 个，其中只有 $P_3 P_4$ 最简单。所以，用布尔代数法求得所需质蕴涵项为 P_3 和 P_4，其结果与行列消去法所得结果相同。

最后，我们举两个综合性的例子，将前面所讲的连贯起来，以说明用 Q-M 列表法化简一个布尔函数的全过程。

【例 3.17】 化简布尔函数

$$F(A,B,C,D) = \sum m(2,4,6,8,9,10,12,13,15)$$

① 求函数的所有质蕴涵，如表 3.17 所示。

求得所有质蕴涵为

$$P_1 = A\overline{C}, P_2 = \overline{A}C\overline{D}, P_3 = \overline{B}C\overline{D}, P_4 = \overline{A}B\overline{D}, P_5 = B\overline{C}\,\overline{D}, P_6 = A\overline{B}\,\overline{D}, P_7 = ABD$$

② 求必要质蕴涵，如表 3.18 所示。

由表 3.18 可确定 P_1 和 P_7 为必要质蕴涵。

表 3.17　求质蕴涵项 P_i

m_i	A	B	C	D		m_i	A	B	C	D		m_i	A	B	C	D	
2	0	0	1	0	√	2,6	0	—	1	0	P_2	8,9 12,13	1	—	0	—	P_1
4	0	1	0	0	√	2,10	—	0	1	0	P_3						
8	1	0	0	0	√	4,6	0	1	—	0	P_4						
6	0	1	1	0	√	4,12	—	1	0	0	P_5						
9	1	0	0	1	√	8,9	1	0	0	—	√						
10	1	0	1	0	√	8,10	1	0	—	0	P_6						
12	1	1	0	0	√	8,12	1	—	0	0	√						
13	1	1	0	1	√	9,13	1	—	0	1	√						
15	1	1	1	1	√	12,13	1	1	0	—	√						
						13,15	1	1	—	1	P_7						

③ 求函数的最小覆盖。

去掉表 3.18 中的 P_1 和 P_7 行，以及已被 P_1 和 P_7 覆盖的最小项 m_8、m_9、m_{12}、m_{13} 和 m_{15}，得到简化的质蕴涵表，如表 3.16 所示。

利用前面讲的行列消去法或用代数法求得所需质蕴涵为 P_3 和 P_4，因此，函数的最简表达式为

$$F(A,B,C,D) = P_1 + P_7 + P_3 + P_4 = A\overline{C} + ABD + \overline{B}C\overline{D} + \overline{A}B\overline{D}$$

表 3.18　质蕴涵表

	2	4	6	8	9	10	12	13	15
√ P_1				×	⊗		×	×	
P_2	×		×						
P_3	×					×			
P_4		×	×						
P_5		×					×		
P_6				×		×			
√ P_7								×	⊗
覆盖情况				×	×		×	×	×

【例 3.18】　化简不完全确定的布尔函数

$$F(A,B,C,D) = \sum m(1,3,4,5,10,11,12) + \sum d(2,13)$$

对于不完全确定的布尔函数的化简，要注意两点：一是在列表求所有质蕴涵时，应该令 $d=1$，以尽量利用随意项进行合并；二是在列质蕴涵表时，应令 $d=0$，即随意项的覆盖问题可不必考虑，以有利于得到最简式。

① 求所有质蕴涵项。令所有随意项为 1，如表 3.19 所示。

表 3.19　求质蕴涵项 P_i

m_i	A	B	C	D	
1	0	0	0	1	√
2	0	0	1	0	√
4	0	1	0	0	√
3	0	0	1	1	√
5	0	1	0	1	√
10	1	0	1	0	√
12	1	1	0	0	√
11	1	0	1	1	√
13	1	1	0	1	√

m_i	A	B	C	D	
1,3	0	0	—	1	P_3
1,5	0	—	0	1	P_4
2,3	0	0	1	—	√
2,10	—	0	1	0	√
4,5	0	1	0	—	√
4,12	—	1	0	0	√
3,11	—	0	1	1	√
5,13	—	1	0	1	√
10,11	1	0	1	—	√
12,13	1	1	0	—	√

m_i	A	B	C	D	
2,3 10,11	—	0	1	—	P_1
4,5 12,13	—	1	0	—	P_2

求得所有质蕴涵项为

$$P_1 = \bar{B}C,\quad P_2 = B\bar{C},\quad P_3 = \bar{A}\bar{B}D,\quad P_4 = \bar{A}\bar{C}D$$

② 求必要质蕴涵。令所有的随意项为 0,即在质蕴涵表中不必列随意项,如表 3.20 所示。

由表 3.20 可确定必要质蕴涵为 P_1和 P_2。

③ 求函数的最小覆盖。

去掉质蕴涵表中被必要质蕴涵覆盖的最小项后,只剩下最小项 m_1未被覆盖,直接从表 3.20 可以选取 P_3或 P_4作为所需质蕴涵。因此,函数的最简表达式为

$$F(A,B,C,D) = P_1 + P_2 + P_3 = \bar{B}C + B\bar{C} + \bar{A}\bar{B}D$$

或者　$F(A,B,C,D)=P_1+P_2+P_3=\bar{B}C+B\bar{C}+\bar{A}\bar{C}D$

表 3.20　质蕴涵表

P_j \ m_i	1	3	4	5	10	11	12
√P_1		×			⊗	⊗	
√P_2			⊗	×			⊗
P_3	×	×					
P_4	×			×			
覆盖情况		×	×	×	×	×	×

以上介绍的都是单个布尔函数的化简,然而在逻辑设计中。经常会遇到多输出布尔函数。所谓多输出布尔函数,就是同样的布尔变量输入得到多个输出的函数。我们当然可以利用卡诺图或列表法单独地化简每个输出函数。然而有时这样得到的电路不一定是最简单的。有时被某个输出函数使用的一些项可以被其他输出函数共享,从而可以减少总的门数。因此我们在化简多输出函数时要考虑尽量多地找出所有的输出函数的公共项,而不是仅仅考虑单个输出函数的化简。

习题 3

3.1　下列函数当变量($A,B,C,\cdots$)取哪些 1 时,F 的值为 1：

(1) $F=AB+\bar{A}C$

(2) $F=A\bar{B}+\bar{A}B$

(3) $F=\bar{A}\bar{B}+AB$

(4) $F=ABC+AB\bar{C}+A\bar{B}C+\bar{A}BC$

(5) $F=(A+\bar{B}+\bar{A}B)(A+\bar{B})\bar{A}B$

(6) $F=(A+B+C)(A+B+\bar{C})(A+\bar{B}+C)(A+\bar{B}+\bar{C})$

(7) $F=(A+B\bar{C})\bar{D}+\overline{(A+\bar{B})CD}$

(8) $F=(A\oplus B)C+\bar{A}(B\oplus C)$

3.2　用真值表验证下列等式：

(1) $\overline{A+B}=\bar{A}\cdot\bar{B}$

(2) $A\bar{B}+\bar{A}B=(\bar{A}+\bar{B})(A+B)$

(3) $(\bar{A}+\bar{B})(A+B)=\overline{AB+\overline{AB}}$

(4) $AB+A\bar{B}+\bar{A}B+\bar{A}\bar{B}=1$

(5) $A\oplus B\oplus C=A\odot B\odot C$

(6) $(A\oplus B)C=A\oplus(B\oplus C)$

3.3　用基本公式和基本规则证明下列等式：

(1) $BC+D+\bar{D}(\bar{B}+\bar{C})(AD+B)=B+D$

(2) $AB+A\bar{B}+\bar{A}B+\bar{A}\bar{B}=1$

(3) $(A+B)(A+\bar{B})(\bar{A}+B)(\bar{A}+\bar{B})=0$

(4) $ABC+\bar{A}\bar{B}\bar{C}=\overline{A\bar{B}+B\bar{C}+\bar{A}C}$

(5) $A\bar{B}+B\bar{C}+C\bar{A}=\bar{A}B+\bar{B}C+\bar{C}A$

(6) $AB+BC+AC=(A+B)(B+C)(A+C)$

(7) $(AB+\bar{A}\bar{B})(BC+\bar{B}\bar{C})(CD+\bar{C}\bar{D})=\overline{A\bar{B}+B\bar{C}+C\bar{D}+D\bar{A}}$

(8) $A\oplus B\oplus C=A\odot B\odot C$

(9) $(x\oplus y)\oplus z=x\oplus(y\oplus z)$

3.4　用展开定理将下式各式简化成为 $A(\cdots)+\bar{A}(\cdots)$形式及$[A+(\cdots)][\bar{A}+(\cdots)]$形式,括号中 A 及 $\bar{A}$ 均不出现：

(1) $F=AG+(A+B)C+\bar{A}D+(\bar{A}+F)E$

(2) $F=(A+\bar{B})(\bar{A}+C)(\bar{D}+E+AF)(G+\bar{H}+\bar{A}J)$

3.5　写出下列表达式的对偶式：

(1) $F=(A+B)(\bar{A}+C)(C+DE)+F$

(2) $F=\overline{\overline{A\bar{B}}\cdot\overline{C\bar{D}}\cdot\overline{D\bar{A}\bar{B}}}$

(3) $F=\overline{\overline{\bar{A}+B+\overline{B+C}}+\overline{A+C+\overline{B+C}}}$

(4) $F=B\,\overline{(A\oplus B)}+B(A\oplus C)$

(5) $F=\overline{(C\odot A)\oplus(B\oplus\bar{D})}$

3.6　求下列函数的补函数：

(1) $F=[(\overline{x_1}x_2+\overline{x_3})x_4+\overline{x_5}]x_6$

(2) $F=S[\bar{W}+I(T+\bar{C})]+H$

(3) $F=A[\bar{B}+(C\bar{D}+\bar{E}F)G]$

(4) $F=A\bar{B}+B\bar{C}+C(\bar{A}+D)$

3.7　将下列函数转换为由“标准积之和”形式表示的函数：

(1) $F(A,B,C)=A\bar{B}C+\bar{A}B+AC+AB\bar{C}$

(2) $F(A,B,C)=\bar{A}+A\bar{C}+BC+A\bar{B}C$

(3) $F(A,B,C)=B(A+\bar{B}+\bar{C})(\bar{A}+\bar{C})C$

(4) $F(A,B,C)=ABC+\bar{A}\bar{B}\bar{C}$

(5) $F(A,B,C)=(\bar{A}B+C)[(\bar{A}\bar{B}+B)C+A]$

3.8　用“标准和之积”形式表示 3.7 题中的函数。

3.9　用代数运算法求下列各函数的“标准积之和”形式及“标准和之积”形式：

(1) $F(A,B,C,D)=AB\bar{C}\bar{D}+ABCD$

(2) $F(A,B,C,D)=\bar{A}BCD+A\bar{B}CD+ACD+\bar{B}CD$

(3) $F(A,B,C,D)=B\bar{C}\bar{D}+\bar{A}B+AB\bar{C}D+BC$

(4) $F(A,B,C)=\bar{C}+\bar{A}B+\bar{B}C+ABC$

(5) $F(A,B,C,D)=\bar{A}(\bar{B}+C)(A+\bar{C})(A+B+C+D)(A+C+\bar{D})$

3.10　设 $F(A,B,C,D,E)=\sum m(0,1,3,7,8,9,12,15,16,17,20,28,29,30,31)$，试用最大项表示这个函数。

3.11　求 3.10 题中所给函数的补函数，并以最大项表示之。

3.12　证明 $A\bar{B}\bar{C}+\bar{A}B\bar{C}+\bar{A}\bar{B}C+ABC=A\oplus B\oplus C$。

3.13　设 A、B、C 为逻辑变量，试回答：

(1) 若已知 $A+B=A+C$，则 $B=C$，对吗？

(2) 若已知 $AB=AC$，则 $B=C$，对吗？

(3) 若已知 $A+B=A+C$，且 $AB=AC$，则 $B=C$，对吗？

3.14　用代数化简法将下列函数化简为“与或”表达式：

(1) $F=\bar{A}\bar{B}C+\bar{A}BC+ABC+AB\bar{C}$

(2) $F=A\bar{B}+B+BCD$

(3) $F=ABC+\bar{A}+\bar{B}+\bar{C}$

(4) $F=\bar{A}\bar{B}+(AB+A\bar{B}+\bar{A}B)C$

(5) $F=\overline{A[B+\bar{C}(D+\bar{E})]}$

(6) $F=\overline{(A+B\bar{C})(\bar{A}+\bar{D}\bar{E})}$

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

(8) $F=\overline{\overline{ABC+\bar{A}\bar{B}}+BC}$

(9) $F=\overline{ABC(A+B+C)}$

(10) $F=\overline{(\bar{A}\bar{B}+ABC)(A\bar{B}C)}$

3.15　用卡诺图法将下列函数化为最简“与或”表达式：

(1) $F(A,B,C)=\sum m(0,1,2,4,5,7)$

(2) $F(A,B,C,D)=\sum m(0,1,2,3,4,6,7,8,9,11,15)$

(3) $F(A,B,C,D)=\sum m(3,4,5,7,9,13,14,15)$

(4) $F(A,B,C,D)=\sum m(0,1,2,5,6,7,8,9,13,14)$

(5) $F(A,B,C,D)=\sum m(0,2,3,5,7,8,10,11)+\sum d(14,15)$

(6) $F(A,B,C,D,E)=\sum m(0,3,4,6,7,8,11,15,16,17,20,22,25,27,29,30,31)$

(7) $F(A,B,C,D,E)=\sum m(0,1,2,3,4,5,6,7,16,17,20,21)+\sum d(24,25,27,28,30)$

(8) $F(A,B,C,D,E)=\prod M(0,1,2,3,8,9,16,17,20,21,24,25,28,29,30,31)\cdot\prod d(13,14,19)$

3.16　用 Q-M 法求下列函数的所有质蕴涵：

(1) $F(A,B,C,D)=\sum m(0,1,2,3,6,7,8,9,10,13,14,15)$

(2) $F(A,B,C,D)=\sum m(0,1,2,3,6,7,9,10,14,15)+\sum d(8,13)$

(3) $F(A,B,C,D,E)=\sum m(0,2,4,9,11,14,15,16,17,19,23,25,29,31)$

3.17　求下列函数的最简"或与"式：

(1) $F(A,B,C,D)=\sum m(4,5,6,13,14,15)$

(2) $F(A,B,C,D)=\sum m(4,5,6,13,14,15)+\sum d(8,9,10)$

3.18　某单位有 5 位外语人员：A 会英语和法语，B 会英语和俄语，C 会俄语和日语，D 会德语，E 会日语和法语。

(1) 外地有一外事活动，要求英、俄、日、德、法 5 种外语，求最经济的出差方案。

(2) 试写出这 5 人中两两进行外语会话的条件(这里指只有两人在场的会话)。

3.19　已知 $F=F(A,B,C,D)$的全部质蕴涵为 $\overline{A}\,\overline{B}\,\overline{C}$、$\overline{A}\,\overline{C}\,\overline{D}$、$\overline{B}D$、$CD$、$\overline{A}B\overline{D}$、$BC$、$AD$、$AC$。求 F 的最简与或式。要求：列质蕴涵表，找必要质蕴涵，列简化的质蕴涵表，找最小质蕴涵覆盖。

3.20　用 Q-M 法化简下列函数：

(1) $F(A,B,C,D,E)=\prod M(0,1,2,3,8,9,10,11,17,19,21,23,25,27,29,31)$

(2) $F(A,B,C,D,E)=\prod M(0,2,4,6,8,10,12,14,16,18,20,22,25,27,29,31)\cdot$
$$\prod d(5,7,13,15,24,26,28,30)$$

3.21　用公式和定理化简下列函数：

(1) $F(A,B,C)=ABC+\overline{A}\,\overline{B}C+\overline{A}B\overline{C}+A\overline{B}\,\overline{C}+\overline{A}\,\overline{B}\,\overline{C}$

(2) $F(A,B,C)=AC+ABC+A\overline{C}+\overline{A}\,\overline{B}\,\overline{C}+BC$

3.22　用卡诺图化简如下函数，并列出它们的质蕴涵项和必要质蕴涵项：

(1) $F(x_1,x_2,x_3,x_4)=\sum m(0,1,4,7,9,10,13)+\sum d(2,5,8,12,15)$

(2) $F(X_1,X_2,X_3,X_4)=\prod M(0,13,15)\cdot\prod d(3,7,9,10,12,14)$

3.23　用卡诺图化简如下四变量函数：

$$F(A,B,C,D)=F_1(A,B,C,D)\oplus F_2(A,B,C,D)$$

其中，$F_1=\overline{A}D+BC+\overline{B}\,\overline{C}\,\overline{D}+\sum d(2,11,13)$

$F_2=\prod M(0,2,4,8,9,10,14)\cdot\prod d(1,7,13,15)$

3.24　用两函数卡诺图之对应单元相加的方法求两函数的布尔和 F_1+F_2。其中

$$F_1(x_1,x_2,x_3,x_4)=\sum m(0,3,10,12,15)+\sum d(5,6,9)$$

$$F_2(x_1,x_2,x_3,x_4) = \sum m(0,2,5,10,15) + \sum d(1,6,12)$$

3.25　用两函数卡诺图的对应单元相乘的方法求两函数的布尔积 $F_1 \cdot F_2$。其中

$$F_1(x_1,x_2,x_3,x_4) = \sum m(0,2,12,14) + \sum d(3,5,9,15)$$

$$F_2(x_1,x_2,x_3,x_4) = \sum m(5,6,10,12) + \sum d(1,2,8,9,15)$$

3.26　化简下列多输出函数：

(1) $F_1(A,B,C,D) = A + D + \overline{A}\overline{C}D$

$F_2(A,B,C,D) = \overline{C}\overline{D} + ABD + \overline{B}\overline{C}D$

$F_3(A,B,C,D) = \overline{B}\overline{D} + ABCD + \overline{A}\overline{B}\overline{C}$

(2) $F_1(A,B,C,D) = \sum m(2,3,4,5,6,7,11,14) + \sum d(9,10,13,15)$

$F_2(A,B,C,D) = \sum m(0,1,3,4,5,7,11,14) + \sum d(8,10,12,13)$