用公式方法化简

［例］ 化简逻辑式 $Y = AD + A\overline{D} + AB + \overline{A}C + \overline{C}D + A\overline{B}EF$

解： $Y = A + AB + \overline{A}C + \overline{C}D + A\overline{B}EF$

$= A + \overline{A}C + \overline{C}D$

$\boxed{\text{应用}A + \overline{A}B = A + B}$

$= A + C + \overline{C}D = A + C + D$

［例］　化简逻辑式 $Y = \overline{A+B} \cdot \overline{ABC} \cdot \overline{\bar{A}\bar{C}}$

解：$\bar{Y} = \overline{\overline{A+B} \cdot \overline{ABC} \cdot \overline{\bar{A}\bar{C}}}$　　用摩根定律

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

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

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

$\Rightarrow$ $Y = \overline{A + B + \bar{C}} = \bar{A}\bar{B}C$

用卡诺图方法化简

最小项的概念与性质

1. 最小项的定义和编号

n 个变量有 2^n 种组合，可对应写出 2^n 个乘积项，这些乘积项均具有下列特点：包含全部变量，且每个变量在该乘积项中（以原变量或反变量）只出现一次。这样的乘积项称为这 *n* 个变量的最小项，也称为 *n* 变量逻辑函数的最小项。

- 最小项定义

  逻辑函数中，如果一个与项（乘积项）包含该逻辑函数的全部变量，且每个变量或以原变量或以反变量只出现一次，则该与项称为最小项。对于n个变量的逻辑函数共有2^n个最小项。

- 最小项的表示

  最小项用m表示，通常用十进制数作最小项编号。把最小项中的原变量当作1，反变量当作0，所得的二进制数所对应的十进制数即为最小项的编号。

- 若干最小项之和构成最小项表达式（也叫标准与-或表达式）：

例如

3 变量逻辑函数的最小项有 $2^3 = 8$ 个

将输入变量取值为 1 的代以原变量，取值为 0 的代以反变量，则得相应最小项。

A	B	C	最小项	简记符号	输入组合对应的十进制数
0	0	0	$\bar{A}\bar{B}\bar{C}$	m_0	0
0	0	1	$\bar{A}\bar{B}C$	m_1	1
0	1	0	$\bar{A}B\bar{C}$	m_2	2
0	1	1	$\bar{A}BC$	m_3	3
1	0	0	$A\bar{B}\bar{C}$	m_4	4
1	0	1	$A\bar{B}C$	m_5	5
1	1	0	$AB\bar{C}$	m_6	6
1	1	1	ABC	m_7	7

例如 $A\bar{B}C \Rightarrow 101 \Rightarrow 5 \Rightarrow m_5$

$m_4 \Rightarrow 4 \Rightarrow 100 \Rightarrow A\bar{B}\bar{C}$

2. 最小项的基本性质

（1）对任意一最小项，只有一组变量取值使它的值为 1，而其余各种变量取值均使其值为 0。

（2）不同的最小项，使其值为 1 的那组变量取值也不同。

（3）对于变量的任一组取值，任意两个最小项的乘积为 0。

（4）对于变量的任一组取值，全体最小项的和为 1。

三变量最小项表

A B C	m_0 $\bar{A}\bar{B}\bar{C}$	m_1 $\bar{A}\bar{B}C$	m_2 $\bar{A}B\bar{C}$	m_3 $\bar{A}BC$	m_4 $A\bar{B}\bar{C}$	m_5 $A\bar{B}C$	m_6 $AB\bar{C}$	m_7 ABC	$F=\sum_{i=0}^{2^n-1} m_i$
0 0 0	1	0	0	0	0	0	0	0	1
0 0 1	0	1	0	0	0	0	0	0	1
0 1 0	0	0	1	0	0	0	0	0	1
0 1 1	0	0	0	1	0	0	0	0	1
1 0 0	0	0	0	0	1	0	0	0	1
1 0 1	0	0	0	0	0	1	0	0	1
1 1 0	0	0	0	0	0	0	1	0	1
1 1 1	0	0	0	0	0	0	0	1	1

二、逻辑函数的“**最小项之和**”形式——标准“与或”表达式（积之和 形式）（**SOP**）

利用基本公式 $\boxed{A+\overline{A}=1}$，可将任何一个逻辑函数化为最小项之和的标准形式。这种标准形式在逻辑函数的化简以及计算机辅助分析和设计中得到了广泛的应用。

例：试将逻辑函数 $Y=A\overline{B}\overline{C}\overline{D}+\overline{A}CD+AC$ **展为最小项之和的形式。**

任何函数 f 都可以用真值表中使 f =1 的各行所对应的最小项的逻辑和来表示。

若逻辑表达式由乘积项（与）的逻辑和（或）的组成，则该逻辑表达式被称为积之和形式。若每个乘积项都为最小项，则该式被称为函数 f 的正则积之和表达式。

- 最大项定义

  逻辑函数中，如果一个或项（和项）包含该逻辑函数的全部变量，且每个变量或以原变量或以反变量只出现一次，则该或项称为最大项。对于n个变量的逻辑函数共有2^n个最大项。

- 最大项的表示

  最大项用M表示，通常用十进制数作最大项编号。把最大项中的原变量当作0，反变量当作1，所得的二进制数所对应的十进制数即为最大项的编号。

- 若干最大项之积构成最大项表达式（也叫标准或-与表达式）：
- 变量数相同时，下标编号相同的最大项和最小项应为互补。

三变量最大项编号表

最大项	使最大项为 0 的变量取值			对应的十进制数	编　号
	A	B	C		
$A+B+C$	0	0	0	0	M_0
$A+B+\overline{C}$	0	0	1	1	M_1
$A+\overline{B}+C$	0	1	0	2	M_2
$A+\overline{B}+\overline{C}$	0	1	1	3	M_3
$\overline{A}+B+C$	1	0	0	4	M_4
$\overline{A}+B+\overline{C}$	1	0	1	5	M_5
$\overline{A}+\overline{B}+C$	1	1	0	6	M_6
$\overline{A}+\overline{B}+\overline{C}$	1	1	1	7	M_7

思考：最大项与最小项之间存在什么关系？

$$M_i=\overline{m_i},\quad m_i=\overline{M_i}$$

Row number	x_1	x_2	x_3	Minterm	Maxterm
0	0	0	0	$m_0 = \overline{x}_1\overline{x}_2\overline{x}_3$	$M_0 = x_1 + x_2 + x_3$
1	0	0	1	$m_1 = \overline{x}_1\overline{x}_2x_3$	$M_1 = x_1 + x_2 + \overline{x}_3$
2	0	1	0	$m_2 = \overline{x}_1x_2\overline{x}_3$	$M_2 = x_1 + \overline{x}_2 + x_3$
3	0	1	1	$m_3 = \overline{x}_1x_2x_3$	$M_3 = x_1 + \overline{x}_2 + \overline{x}_3$
4	1	0	0	$m_4 = x_1\overline{x}_2\overline{x}_3$	$M_4 = \overline{x}_1 + x_2 + x_3$
5	1	0	1	$m_5 = x_1\overline{x}_2x_3$	$M_5 = \overline{x}_1 + x_2 + \overline{x}_3$
6	1	1	0	$m_6 = x_1x_2\overline{x}_3$	$M_6 = \overline{x}_1 + \overline{x}_2 + x_3$
7	1	1	1	$m_7 = x_1x_2x_3$	$M_7 = \overline{x}_1 + \overline{x}_2 + \overline{x}_3$

从最大项的定义出发可以证明它具有如下的重要性质：

（1）在输入变量的任何取值下必有一个且仅有一个最大项的值为0；

（2）全体最大项之积为0；

（3）某一最大项若不包含在F中，则必在$\overline{F}$中；

（4）任意两个最大项之和为1；

（5）只有一个变量不同的两个最大项的乘积等于各相同变量之和。

三、逻辑函数的“**最大项之积**”形式——标准“或与”表达式（和之积形式**POS**）

证明：任何一个逻辑函数都可以化成最大项之积的标准形式。

例：试将逻辑函数 $Y = AB\overline{C} + BC$

化为最大项之积的标准形式。

对偶原理指出：若考虑真值表中使 f = 1 的各行可以综合出一个函数 f ，则考虑使 f = 0 的各行也可以综合出函数 f

任何函数 f 可以通过找到它的正则和之积来实现综合，即从真值表中的每一行中取出使 f ＝ 0 的最大项，把这些最大项组成乘积。

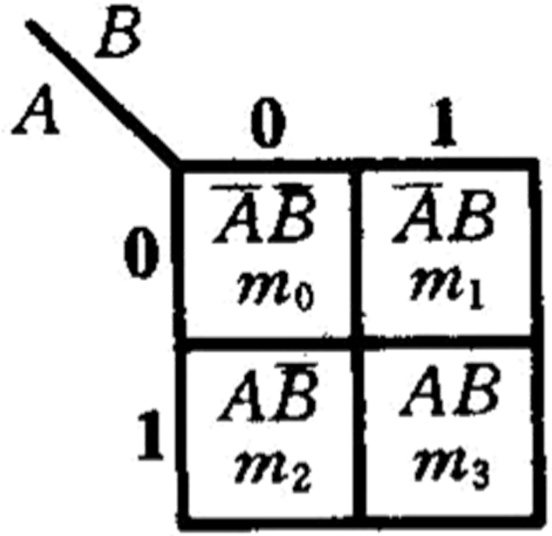
逻辑函数的卡诺图化简

一、逻辑函数的卡诺图表示法

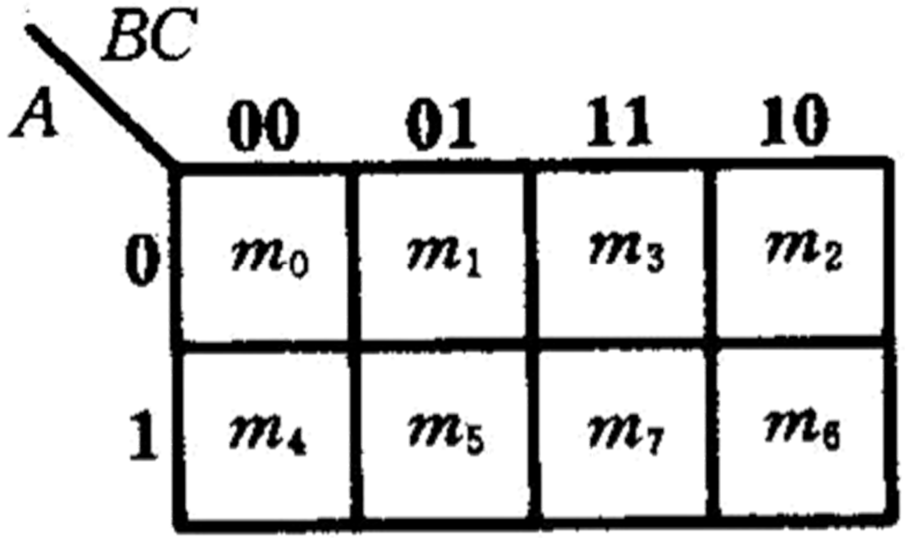
（一）卡诺图（Karnaugh Diagram)

卡诺图是由美国工程师卡诺首先提出的一种用来描述逻辑函数的特殊方格图。在这个方格图中，每个小方格代表逻辑函数的一个最小项，而且几何相邻的小方格具有逻辑相邻性，即两相邻小方格所代表的最小项只有一个变量取值不同。

二、三、四、五变量卡诺图为：



二变量卡诺图



三变量卡诺图

$AB \backslash CD$	00	01	11	10
00	m_0	m_1	m_3	m_2
01	m_4	m_5	m_7	m_6
11	m_{12}	m_{13}	m_{15}	m_{14}
10	m_8	m_9	m_{11}	m_{10}

四变量卡诺图

$AB \backslash CDE$	000	001	011	010	110	111	101	100
00	m_0	m_1	m_3	m_2	m_6	m_7	m_5	m_4
01	m_8	m_9	m_{11}	m_{10}	m_{14}	m_{15}	m_{13}	m_{12}
11	m_{24}	m_{25}	m_{27}	m_{26}	m_{30}	m_{31}	m_{29}	m_{28}
10	m_{16}	m_{17}	m_{19}	m_{18}	m_{22}	m_{23}	m_{21}	m_{20}

五变量卡诺图

（二）卡诺图的特点：

1、卡诺图中的小方格数等于最小项总数，若逻辑函数的变量数为 **n**，则小方格数为 2^n 个。

2、卡诺图行列两侧标注的**0**和**1**表示使对应方格内最小项为**1**的变量取值。同时，这些**0**和**1**组成的二进制数大小就是对应最小项的编号。此外，在卡诺图中，几何相邻的最小项具有逻辑相邻性，因此，变量的取值不能按照二进制数的顺序排列，必须按循环码排列。

3、卡诺图是一个上下、左右闭合的图形，即不但紧挨着的方格是相邻的，而且上下、左右相对应的方格也是相邻的。

（三）用卡诺图表示逻辑函数

1、已知函数式 $\longrightarrow$ 化成最小项之和形式 $\longrightarrow$ 卡诺图中对应最小项格填入“1”$\longrightarrow$ 得到卡诺图。

例： 试用卡诺图表示逻辑函数

$$Y=\bar{A}\bar{B}\bar{C}D+\bar{A}B\bar{D}+ACD+A\bar{B}$$

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

2、已知真值表 → 每组变量（即最小项）所对应的函数值 → 填入卡诺图中相应方格 → 化简得到函数式。

3、已知逻辑函数的卡诺图 → 该函数的真值表 → 写出该函数的逻辑函数式。

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

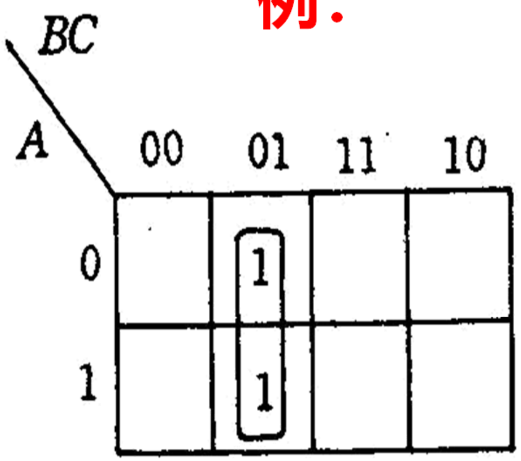
二、用卡诺图化简逻辑函数

（一）用卡诺图化简逻辑函数的依据（基本性质）

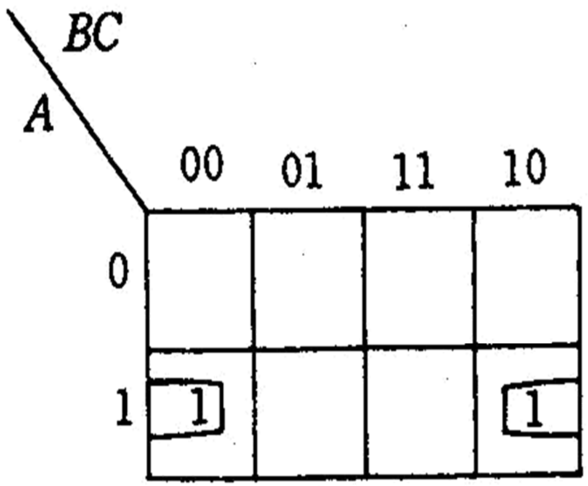
基本原理：各几何相邻方格的逻辑相邻性。

性质1：卡诺图中两个相邻“1”格的最小项可以合并成一个与项，并消去一个变量。

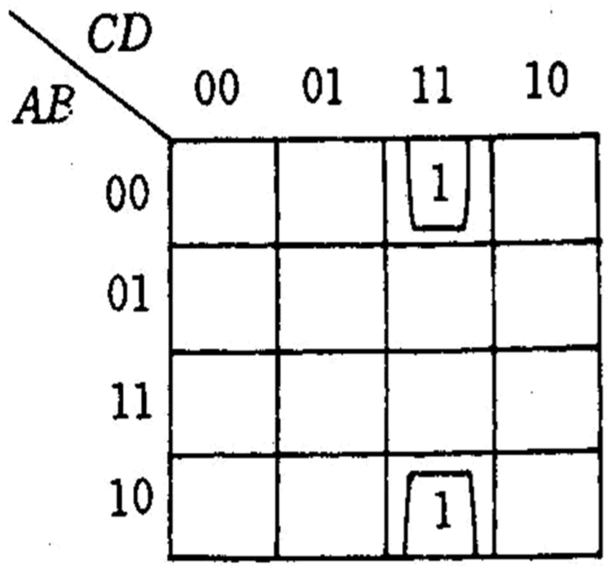
例：



(a) $\overline{B}C$



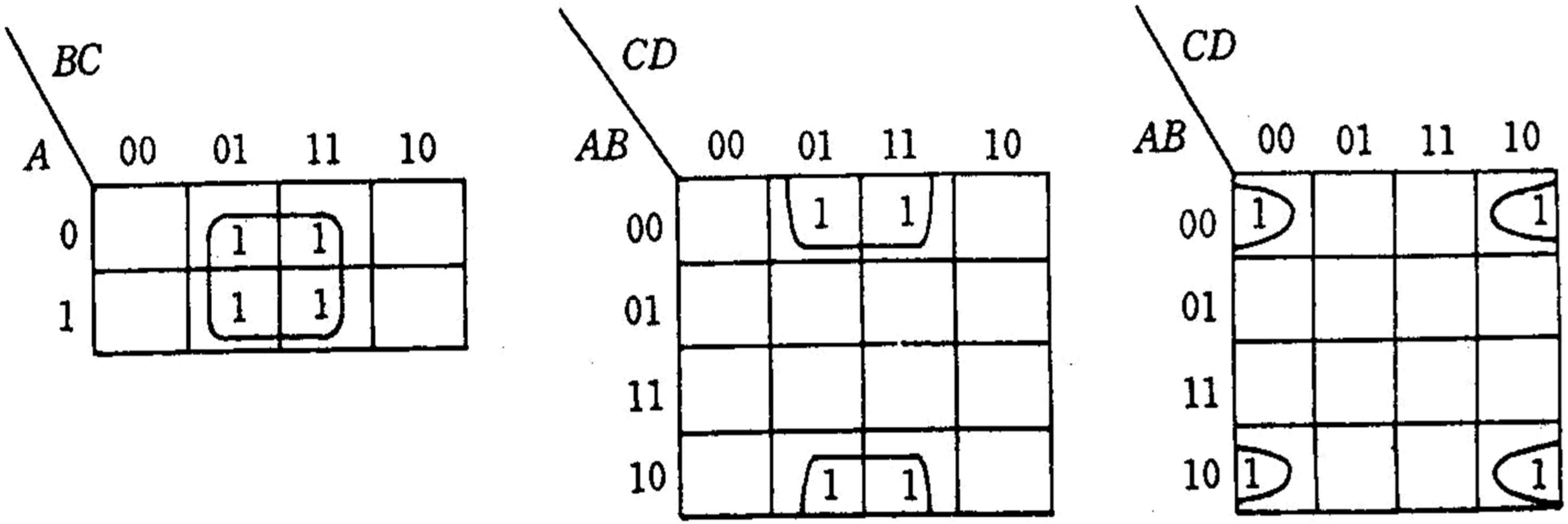
(b) $A\overline{C}$



(c) $\overline{B}CD$

性质2：卡诺图中**四个相邻“1”格**的最小项可以合并成一个与项，并消去**两个**变量。

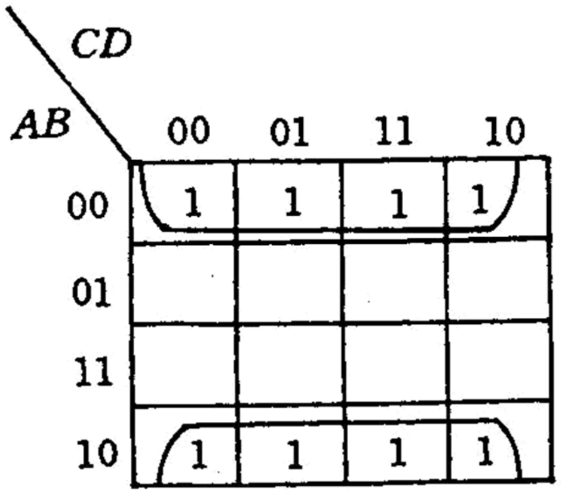
例:



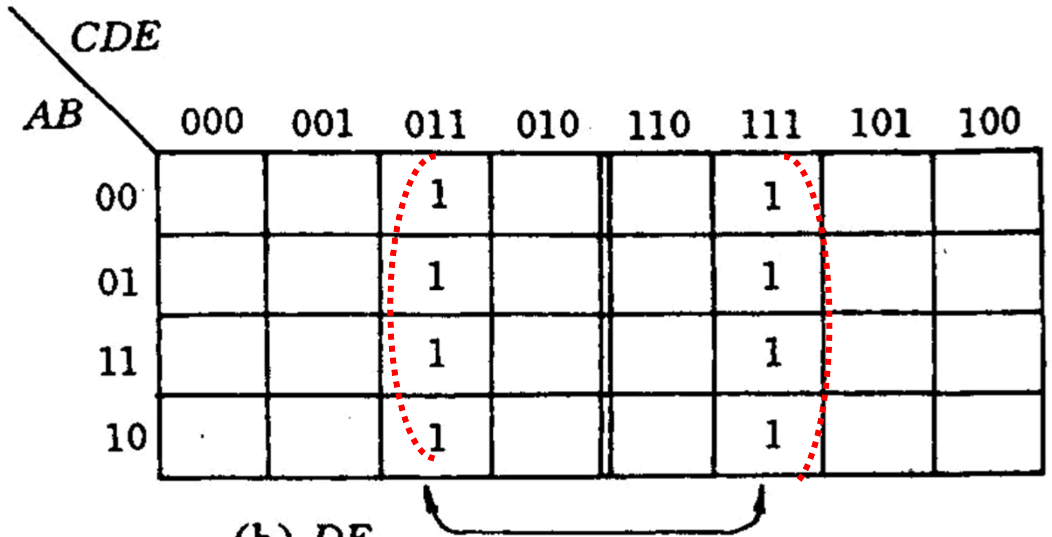
(a) C　　(b) $\bar{B}D$　　(c) $\bar{B}\bar{D}$

性质3：卡诺图中**八个相邻“1”格**的最小项可以合并成一个与项，并消去**三个**变量。

例：



(a) $\overline{B}$



(b) DE

推论：在n个变量的卡诺图中，若有2^k 个 “1”格相邻（k=0，1，2，3，…，n），它们可以圈在一起加以合并，合并时可以消去**k个**不同的变量，简化为一个具有（**n-k**）个变量的与项。若**k=n**，则合并时可消去全部变量，结果为**1**。

（二）用卡诺图求最简与或表达式

例1：用卡诺图化简函数

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

步骤：

1、得到函数的真值表或将函数化为最小项之和的标准形式；

2、画出函数的卡诺图；

3、合并最小项（即“画圈”）；

“画圈”时应注意的问题：

①“1”格一个也不能漏，否则表达式与函数不等；

②“1”格允许被一个以上的圈包围，因为A+A=A；

③ 圈的个数应尽可能少，因为一个圈对应一个与项，即与项最少；

例：

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

$$F = BC + \bar{A}\bar{B}\bar{C} + \bar{A}BD + \bar{A}C\bar{D}$$

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

$$F = BC + \bar{A}\bar{B}\bar{D} + \bar{A}\bar{C}D$$

④ 圈的面积越大越好，但必须为 2^k 个方格。这是因为圈越大，消去的变量就越多，与项中的变量数就越少。

例：

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

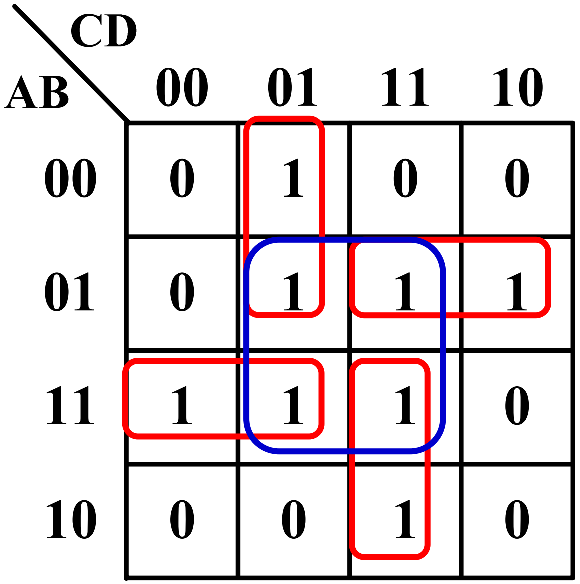
$$F = \bar{A} + A\bar{B}C$$

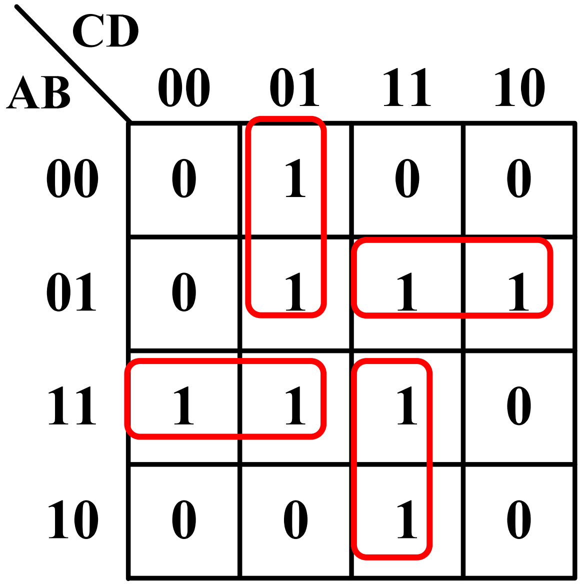
AB \ CD	00	01	11	10
00	1	1	1	1
01	1	1	1	1
11	0	0	0	0
10	0	0	1	1

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

⑤ 每个圈至少应包含一个新的“1”格，否则这个圈是多余的，即增加了冗余项；

例：





总之，即“可以重画，不能漏画，圈数要少，圈面要大，每个圈必有一个新‘1’”。

4、写出最简“与-或”表达式。

例2：试用图形化简法求逻辑函数F（A，B，C，D）=∑m (1，2，4，9，10，11，13，15）的最简与或表达式。

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

例3：用卡诺图化简函数

$$Y = \sum m(1,\ 3,\ 6,\ 7,\ 14)$$

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

$$Y = A\overline{B} + \overline{A}B + B\overline{C} + \overline{B}C$$

注：卡诺图化简得到的最简与或式不一定唯一。

具有无关项的逻辑函数及其化简

一、约束项、任意项和逻辑函数式中的无关项

约束项——在某些情况下，输入变量的取值不是任意的。当限制某些输入变量的取值不能出现时，可以用它们对应的最小项恒等于0来表示。这些恒等于0的最小项叫约束项。

任意项——有时输入变量的某些取值是1还是0皆可，并不影响电路的功能。在这些变量取值下，其值等于1的那些最小项称为任意项。

无关项——约束项和任意项统称为逻辑函数中的无关项。“无关”指是否将这些最小项写入逻辑函数式无关紧要，在卡诺图中用“×”表示无关项。在化简逻辑函数时，可认为它是1，也可认为它是0。

二、无关项在化简逻辑函数中的应用

化简具有无关项的逻辑函数时，如果能合理利用这些无关项，一般都可以得到更加简单的化简结果。

合并最小项时，究竟把卡诺图上的“×”作为**1**还是**0**，应以得到的相邻最小项矩形组合最大，而且矩形组合数目最小为原则。

例：试化简逻辑函数

$$Y = \overline{A}C\overline{D} + \overline{A}B\overline{C}\,\overline{D} + A\overline{B}\,\overline{C}\,\overline{D}$$

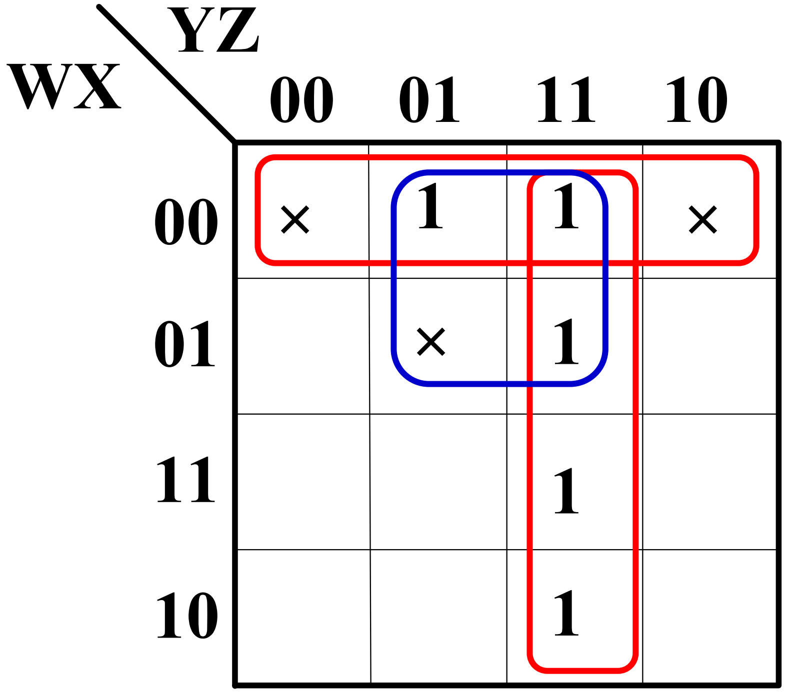
已知约束条件为：

$$A\overline{B}C\overline{D} + A\overline{B}CD + AB\overline{C}\,\overline{D} + AB\overline{C}D + ABC\overline{D} + ABCD = 0$$

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

例：试用卡诺图化简逻辑函数

$$F(W, X, Y, Z) = \sum_m (1,3,7,11,15) + \sum_d (0,2,5)$$



思考：由上例可得出什么结论和启示？

解答：此例有两种解法，从原理而言，两种解法均正确，但就“最简”原则而言，只有一种解法最简单、最可取。因此，**在考虑卡诺图化简不唯一性的同时，还应考虑“最简”原则**。

思考1：公式化简和卡诺图化简各有何优缺？

化简法	优点	缺点
公式法	化简不受输入变量数目的影响。	化简过程没有固定的、通用的步骤可循，不适用于计算机辅助化简。
卡诺图法	直观、简单	输入变量数目较多时（例如>5），不再直观，且化简需凭设计者的经验，不便于利用计算机完成化简工作。