Part Ⅰ 逻辑代数*OK*

2023Lecture 03 逻辑函数化简

重点内容：最简与或式、卡诺图、最简或与式

“与-或”式和“或-与”式可以很方便地转换成任何其他所要求的形式。

逻辑函数化简有 2 **种常用方法**：代数法、卡诺图法。

一、代数化简法

代数化简法就是运用逻辑代数的公理、定律和规则对逻辑函数进行化简的方法。

1、与-或式的化简

最简“与-或”式应同时满足以下两个条件：

①表达式中的“与”项个数最少；

②每个“与”项中的变量个数最少。

几种**常用方法**如下：

（1）并项法

利用 $A\overline{B}+AB=A$，将两个“与”项合并成一个并消去一个变量。

例如：$\overline{A}B\overline{C}+\overline{A}BC=\overline{A}B$

（2）吸收法

利用$A+AB=A$，吸收多余的项并消去一个量。

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

（3）消去法

利用$A+\bar{A}B=A+B$，消去多余变量。

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

$=AB+\overline{AB}\bar{C}=AB+\bar{C}$

（4）配项法

利用$A=A\cdot 1$和$A+\bar{A}=1$，先从函数式中适当选择某些“与”项，并配上其所缺的一个合适的变量，然后再利用并项、吸收和消去等方法进行化简。

例如：

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

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

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

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

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

实际应用中遇到的逻辑函数往往比较复杂，化简时应灵活使用所学的公理、定律及规则，综合运用各种方法。

例如：

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

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

$$= BC + D + (\overline{B} + \overline{C})B\overline{C}\overline{D}$$

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

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

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

$$= BC\bar{D} + BCD + D + B\bar{C}\bar{D}$$

$$= BC\bar{D} + B\bar{C}\bar{D} + BCD + D$$

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

再如：$F = BC + D + (\bar{B} + \bar{C})(AD + B\bar{C})$

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

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

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

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

$$= B + D$$

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

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

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

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

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

2、“或-与”式的化简

最简“或-与”式应同时满足两个条件：

(1)表达式中的“或”项个数最少；

(2)每个“或”项中的变量个数最少。

用代数法化简“或-与”表达式可直接运用公理、定律中的“或-与”形式，并综合运用前面介绍“与-或”表达式化简时提出的各种方法。

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

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

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

此外，可以采用**两次对偶法**：

第一步：对“或-与”表达式表示的函数F求对偶，得到“与-或”表达式F'；

第二步：求出F'的最简“与-或”表达式；

第三步：对F'再次求对偶，即可得到F的最简“或-与”表达式。

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

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

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

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

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

还可以**两次取反**：

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

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

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

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

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

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

代数化简法的**优点**：不受变量数目的约束；当对公理、定律和规则十分熟练时，化简比较方便。

缺点：没有一定的规律和步骤，技巧性很强，而且在很多情况下难以判断化简结果是否最简。

二、卡诺图化简法

1、卡诺图的构成

卡诺图是一种平面方格图，每个小方格代表一个最小项，故又称为**最小项方格图**。

结构特点：

①n个变量的卡诺图由2^n个小方格构成，每个小方格代表一个最小项；

②几何上处在**相邻**、**相对**、**相重**位置的小方格所代表的最小项为相邻最小项。

2 变量、3 变量、4 变量的卡诺图分别如图(a)、(b)、(c)所示：

B \ A	0	1
0	m_0	m_2
1	m_1	m_3

(a)

C \ AB	00	01	11	10
0	m_0	m_2	m_6	m_4
1	m_1	m_3	m_7	m_5

(b)

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

(c)

从各卡诺图可以看出，在n个变量的卡诺图中，能从图形上直观、方便地找到每个最小项的n个相邻最小项。

例如：四变量卡诺图中，m_5的 4 个相邻最小项分别是和它几何相邻的m_1, m_4, m_7, m_{13}。

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

m_2的 4 个相邻最小项除了与之**几何相邻**的m_3, m_6之外，另外两个是处在“**相对**”位置的m_0(同一列的另一端)和m_{10}(同一行的另一端)。这种相邻称为**相对相邻**。

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

此外，处在“**相重**”位置的最小项相邻，如五变量卡诺图中的m_3，除了几何相邻的m_1, m_2, m_7和相对相邻的m_{11}外，还与m_{19}相邻。这种相邻称为**重叠相邻**。

DE \ ABC	000	001	011	010	100	101	111	110
00	0	4	12	8	16	20	28	24
01	**1**	5	13	9	17	21	29	25
11	**3**	**7**	15	**11**	**19**	23	31	27
10	**2**	6	14	10	18	22	30	26

(d)

好处：可以从图形上直观地找出相邻最小项进行合并，合并的理论依据是$A\overline{B}+AB=A$。

基本原理：把卡诺图上代表相邻最小项的相邻小方格“圈”在一起进行合并，达到用一个简单“与”项代替若干最小项的目的。

CD \ AB	00	01	11	10
00				
01		m_5	m_{13}	
11		m_7	m_{15}	
10				

$$\underline{\bar{A}B\bar{C}D + \bar{A}BCD} + \underline{AB\bar{C}D + ABCD}$$

$$\underline{\bar{A}BD} + \underline{ABD}$$

$$BD$$

通常把用来包围那些能由一个简单“与”项代替的若干最小项的“圈”称为**卡诺圈**。

卡诺图化简的关键就是找合理的卡诺圈！

2、用卡诺图表示逻辑函数

（1）逻辑函数为标准“与-或”式

当逻辑函数为标准“与-或”式时，只需在卡诺图上找出表达式中最小项对应的小方格填上 1，其余小方格填上 0，即可得到该函数的卡诺图。

例如：函数$F(A,B,C)=\sum m(1,2,3,7)$的卡诺图如下所示：

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

$F(A, B, C)=\sum m(1, 2, 3, 7)$的卡诺图

（2）逻辑函数为一般“与-或”式

当逻辑函数为一般“与-或”式时，可根据“与”的公共性和“或”的叠加性作出相应的卡诺图。

例如：函数 $F(A,B,C,D)=AB+CD+\overline{A}\overline{B}C$ 的卡诺图如下所示：

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

为了叙述的方便，通常将卡诺图上填 1 的小方格称为 **1 方格**，填 0 的小方格称为 **0 方格**。0 方格有时用空格表示。

问题：函数为或与式时又该如何呢？

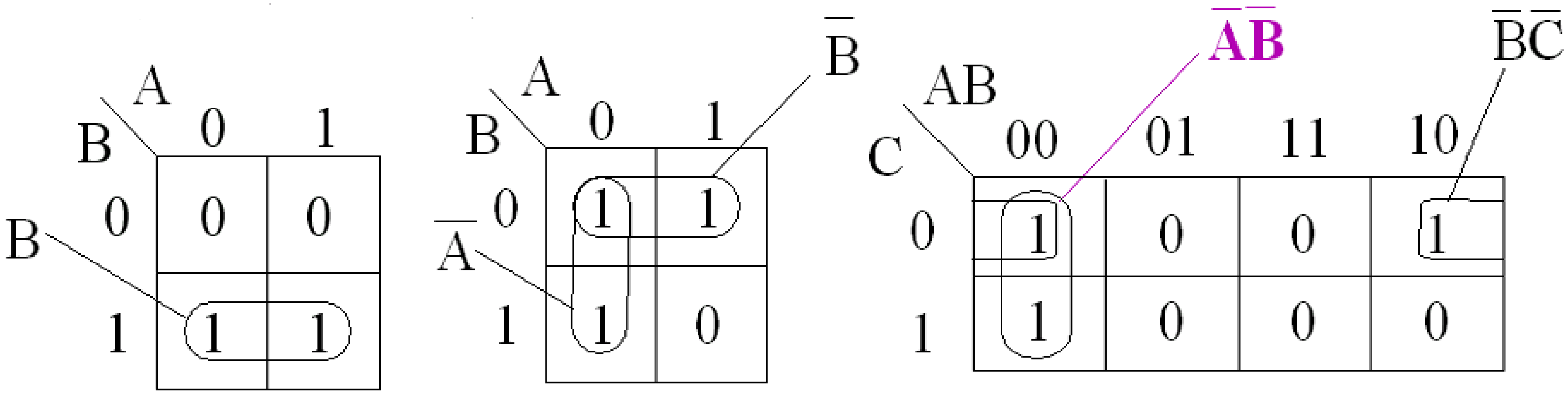
3、利用卡诺图化简逻辑函数

当一个函数用卡诺图表示出来后，究竟哪些最小项可以合并呢？

下面以 2、3、4 变量卡诺图为例予以说明。

1）两个小方格相邻，或处于某行(列)两端时，所代表的最小项可以合并，合并后可消去一个变量。

例如，下图给出了2、3变量卡诺图上两个相邻最小项合并的典型情况的。

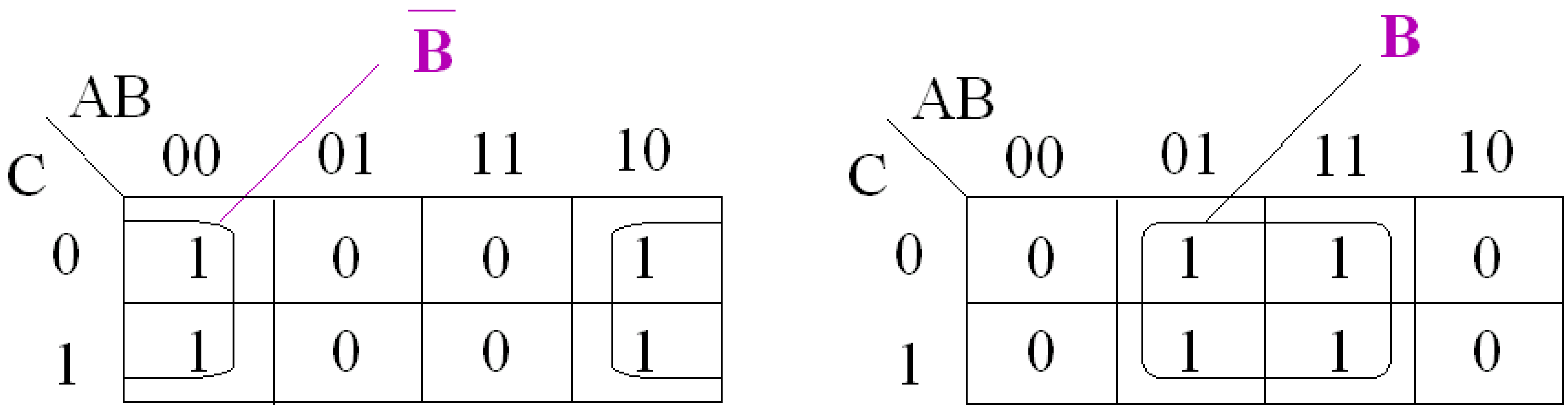


两个相邻最小项合并的情况

2）四个小方格组成一个大方格、或组成一行（列）、或处于相邻两行（列）的两端、或处于四

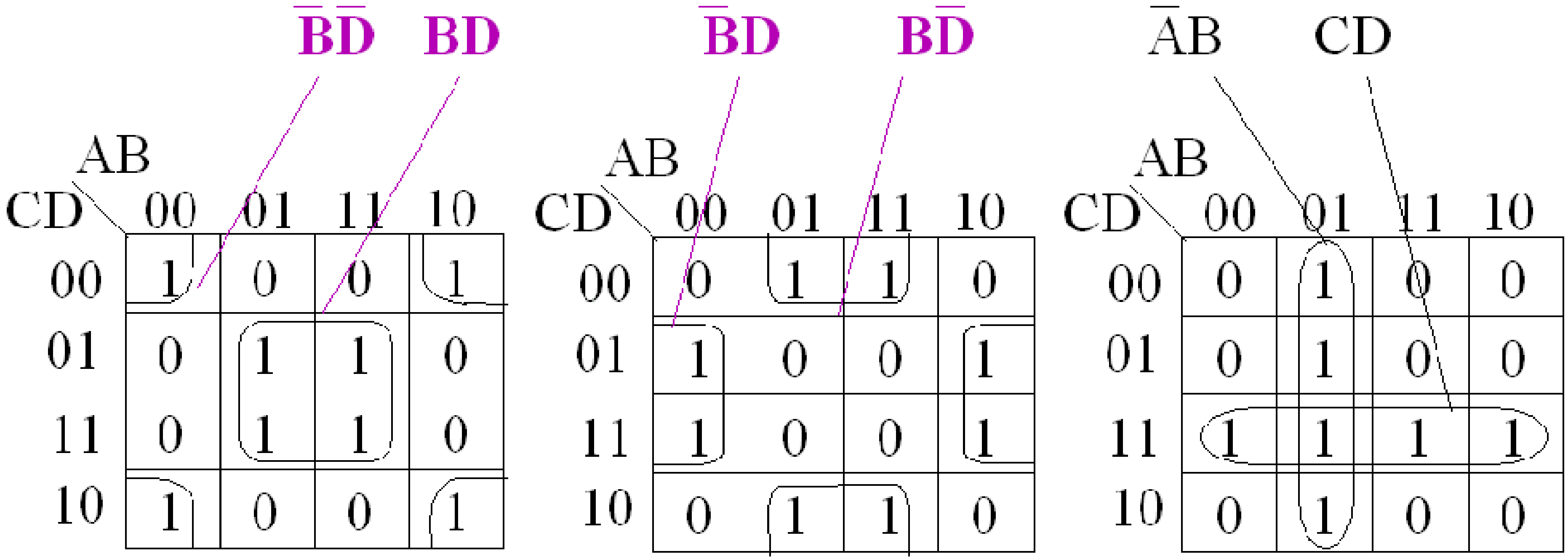
角时，所代表的最小项可以合并，合并后可消去两个变量。

例如，下图给出了 3 变量卡诺图上四个相邻最小项合并的典型情况的。



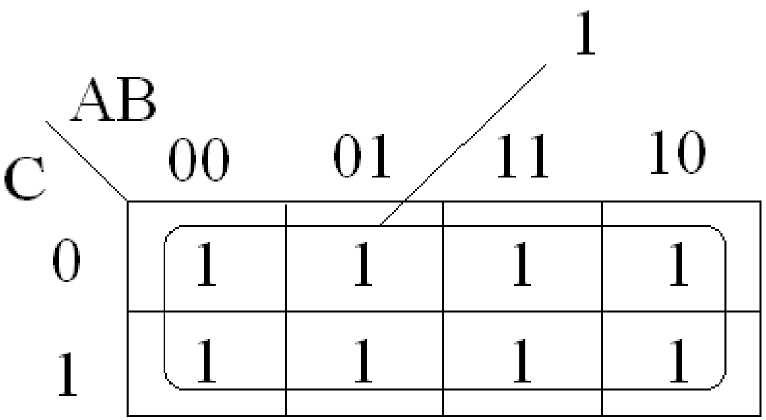
四个相邻最小项合并的情况

4 变量卡诺图上四个相邻最小项合并的典型情况：

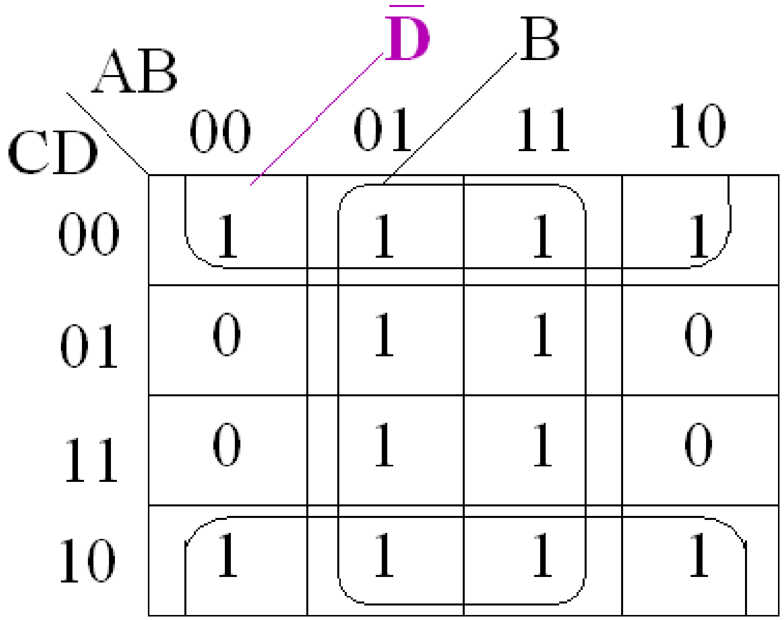


四个相邻最小项合并的几种情况

3）八个小方格组成一个大方格、或组成相邻的两行(列)、或处于两个边行(列)时，所代表的最小项可以合并，合并后可消去三个变量。



(a)



(b)

8个相邻最小项合并的两种情况

例如，上图给出了 3、4 变量卡诺图上八个相邻最小项合并的典型情况的。

n变量卡诺图中最小项的合并规律归纳如下：

①卡诺圈中小方格的个数必须为2^m个，m为小于或等于n的整数。

②卡诺圈中的2^m个小方格有一定的排列规律，具体地说，它们含有m个不同量，$n-m$个相同量。

③卡诺圈中的2^m个小方格对应的所有最小项可用$n-m$个变量的“与”项表示，该“与”项由这些最小项中的相同变量构成。

④当$m=n$时，卡诺圈包含了整个卡诺图，可用 1 表示，即n个变量的全部最小项之和为 1。

总结：卡诺圈越少越好，越大越好；全部卡诺圈要覆盖函数所涉及的所有最小项；任何一个卡诺圈至少有一个最小项未出现在其它卡诺圈中。

过程总结：

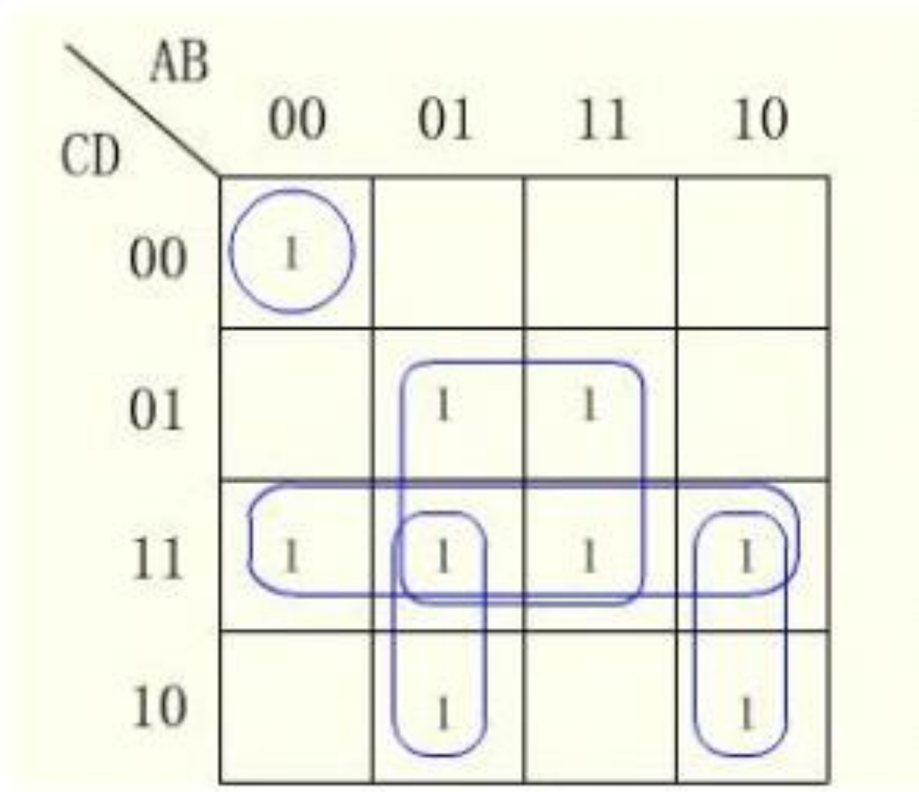
1）正确画出卡诺图

2）正确画出卡诺圈

3）根据卡诺圈写出最简表达式

例如：化简

$$F(A,B,C,D)=\sum m(0,3,5,6,7,10,11,13,15)$$



解答：函数F的最简"与-或"表达式为：

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

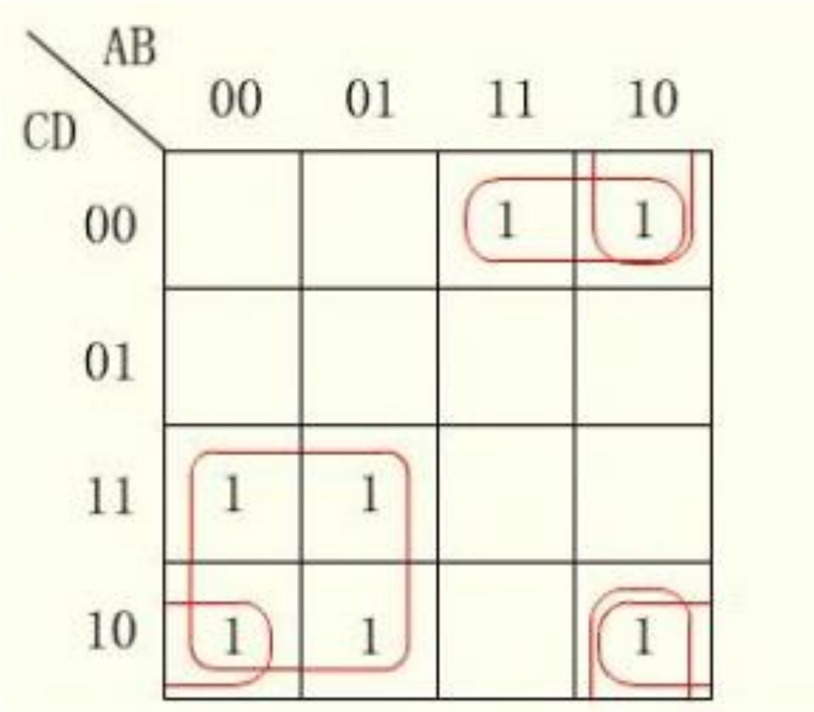
$$=\bar{A}\bar{B}\bar{C}\bar{D}+\bar{A}BC$$

$$+A\bar{B}C+BD+CD$$

再如：用卡诺图化简

$$F(A,B,C,D)=\sum m(2,3,6,7,8,10,12)$$

解答：



结果： $F(A,B,C,D)$

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

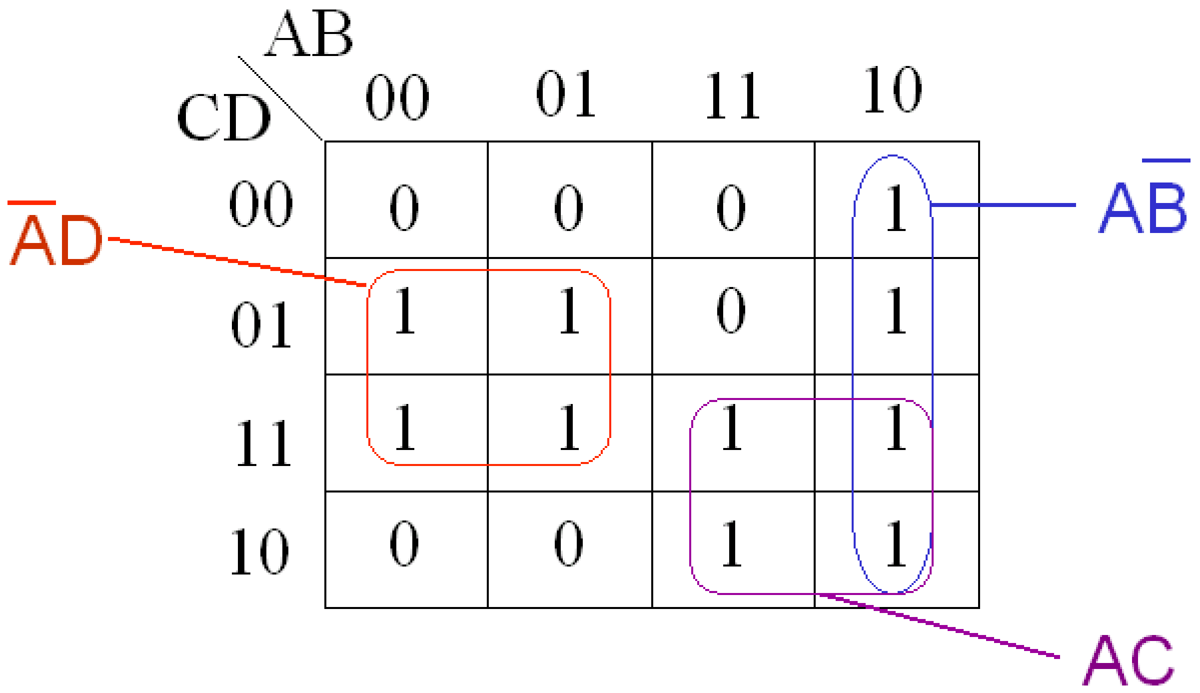
或者： $F(A,B,C,D)$

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

再如：化简

$$F(A,B,C,D)=\sum m(1,3,5,7,8,9,10,11,14,15)$$

解答：$F(A,B,C,D)=AC+A\bar{B}+\bar{A}D$



用卡诺图化简总的原则是：

(1)在覆盖函数中所有最小项的前提下，卡诺圈的个数应达到最少；

(2)在满足合并规律的前题下卡诺圈应达到最大；

(3)根据合并的需要，每个最小项可以被多个卡诺圈包含。

4、求逻辑函数最简“或-与”式

（1）两次取反法

当给定逻辑函数为“与-或”式或标准“与-或”式时，通常采用“两次取反法”，具体如下：

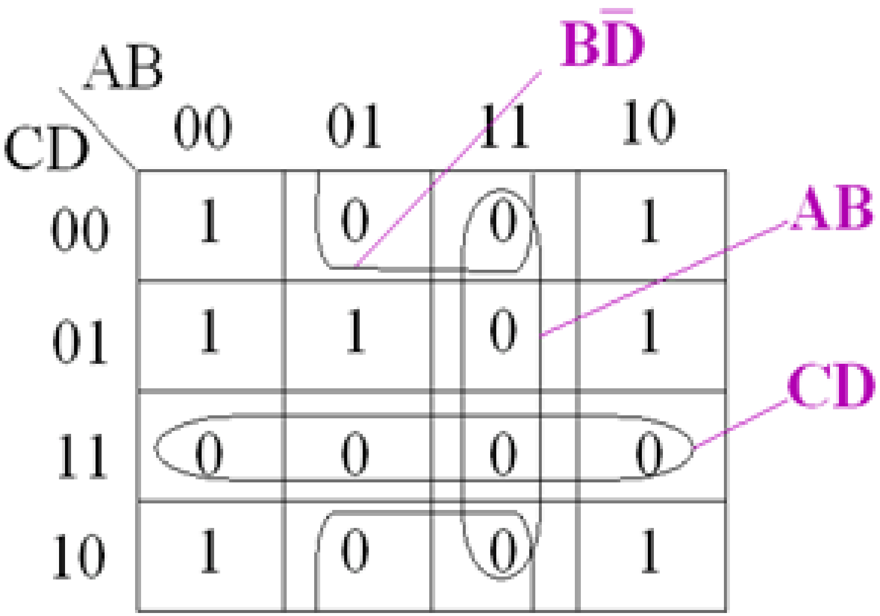
①作出F的卡诺图，求出$\overline{F}$的最简“与-或”式(合并卡诺图上的0方格)；

②对$\overline{F}$的最简“与-或”式取反，得到F的最

简“或–与”式。

例如：化简$F(A,B,C,D)=\sum m(0,1,2,5,8,9,10)$

解答：$\bar{F}$的卡诺图和卡诺圈如下：



$$\bar{F}=AB+CD+B\bar{D}$$

$$F=\overline{AB+CD+B\bar{D}}$$

$$=\overline{AB}\,\overline{CD}\,\overline{B\bar{D}}$$

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

（2）两次对偶法

当给定逻辑函数为“或-与”式或标准“或-与”式时，通常采用“两次对偶法”，具体如下：

①作出F的对偶式F'的卡诺图，并求出F'的最简“与-或”式；

②对F'的最简“与-或”式取对偶，得到F的最简“或-与”式。

例如：求下列函数的最简“或–与”式：

$$F(A,B,C,D)=\prod M(2,4,5,10)$$

解答：$F(A,B,C,D)=\prod M(2,4,5,10)$

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

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

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

$$=\sum m(5,10,11,13)$$

利用卡诺图求出F'的最简“与-或”式：

AB / CD	00	01	11	10
00				
01		1	1	
11				1
10				1

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

So：

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

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

卡诺图化简逻辑函数具有方便、直观、容易掌握等优点。但受到变量个数的约束，当变量个数大

于 6 时，画图以及对图形的识别都变得相当复杂。

问题：如何直接求 F 的最简或与式？

问题：为什么变量超过 6 个时，卡诺图也不好使？

三、含任意项的逻辑函数的化简

任意项：由于输入变量之间存在的相互制约或问题的某种特殊限定，输出函数与输出变量的某些取值无关，这些输入组合对应的**最小项**称为**任意项**或**无关项**。

例如，假定用 A、B、C 表示计算机中的＋、－、×运算，并令变量取值 1 执行相应运算，则这三个变量不允许两个或两个以上同时为 1。

换言之，A、B、C 只允许出现 000、001、010、100 这四种取值组合，而不允许出现其余的四种组合：011、101、110、111。

此问题包含四个任意项：$\bar{A}BC$、$A\bar{B}C$、$AB\bar{C}$、ABC。

问题：任意项对应的函数值如何处理？

回答：在保证不出现无效组合的条件下，爱咋

处理咋处理！不管怎么处理都不影响函数的实际逻辑功能。

问题：你会如何处理？

回答：按有利于化简的方式处理！

【例如】设计一个组合逻辑电路，用来判断以余 3 码表示的 1 位十进制数是否为合数。

【分析】设输入变量为 A、B、C、D，输出函

数为F，当$ABCD$表示的十进制数为合数（4、6、8、9）时，输出F为 1，否则F为 0。

按照余 3 码的编码规则，$ABCD$的取值组合不允许为 0000、0001、0010、1101、1110、1111，这 6 种取值组合对应的最小项为任意项，即它们对应的输出可以随意指定为 1 或者为 0，通常记为“d”。

根据以上分析，可建立描述该问题的真值表如右：

由真值表可写出F的逻辑表达式：

A	B	C	D	F	A	B	C	D	F
0	0	0	0	d	1	0	0	0	0
0	0	0	1	d	1	0	0	1	1
0	0	1	0	d	1	0	1	0	0
0	0	1	1	0	1	0	1	1	1
0	1	0	0	0	1	1	0	0	1
0	1	0	1	0	1	1	0	1	d
0	1	1	0	0	1	1	1	0	d
0	1	1	1	1	1	1	1	1	d

$$F(A,B,C,D)=\sum m(7,9,11,12)$$

$$+\sum d(0,1,2,13,14,15)$$

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

若无关项都处理成 0，则F的卡诺图如下：

化简后的逻辑表达式为

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

$$=A\bar{B}D+AB\bar{C}\bar{D}+\bar{A}BCD$$

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

若根据化简的需要进行处理，则函数F的卡诺图如左，化简结果为

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

$$=AB+AD+BCD$$

问题：大家看懂了吗？

假定采用**与非门**实现给定逻辑功能的电路，

可将F的最简表达式变换成“与非”式：

$$F(A,B,C,D)=AB+AD+BCD$$

$$=\overline{\overline{AB}\cdot\overline{AD}\cdot\overline{BCD}}$$

相应的逻辑电路图如下：