

2.3.2 逻辑函数表达式的标准形式

一、最小项

如果一个具有 n 个变量的函数的"积"项包含全部 n 个变量, 每个变量都以原变量或反变量形式出现, 且仅出现一次, 则这个"积"项被称为最小项。假如一个函数完全由最小项所组成, 那么该函数表达式称为标准"积之和"表达式, 即"最小项之和".

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

表2.5

变量的各组取值			对应的最小项及其编号		对应的最大项及其编号	
A	B	C	最小项	编 号	最大项	编 号
0	0	0	$\overline{A}\overline{B}\overline{C}$	m_0	$A + B + C$	M_0
0	0	1	$\overline{A}\overline{B}C$	m_1	$A + B + \overline{C}$	M_1
0	1	0	$\overline{A}B\overline{C}$	m_2	$A + \overline{B} + C$	M_2
0	1	1	$\overline{A}BC$	m_3	$A + \overline{B} + \overline{C}$	M_3
1	0	0	$A\overline{B}\overline{C}$	m_4	$\overline{A} + B + C$	M_4
1	0	1	$A\overline{B}C$	m_5	$\overline{A} + B + \overline{C}$	M_5
1	1	0	$AB\overline{C}$	m_6	$\overline{A} + \overline{B} + C$	M_6
1	1	1	ABC	m_7	$\overline{A} + \overline{B} + \overline{C}$	M_7

$$\begin{aligned}\text{因此, } F(A, B, C) &= \overline{A}\overline{B}\overline{C} + \overline{A}B\overline{C} + A\overline{B}\overline{C} + ABC \\ &= m_2 + m_3 + m_6 + m_7 \\ &= \Sigma m(2, 3, 6, 7)\end{aligned}$$

注意：变量的顺序.

$$\text{因此, } f(A_1, A_2, \dots, A_n) + \overline{f}(A_1, A_2, \dots, A_n) = 1$$

$$\text{而 } f(A_1, A_2, \dots, A_n) + \overline{f}(A_1, A_2, \dots, A_n) = \sum_{i=0}^{2^n-1} m_i$$

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

即 n 个变量的所有最小项之和恒等于 1。

二、最大项

如果一个具有 n 个变量的函数的"和"项包含全部 n 个变量，每个变量都以原变量或反变量形式出现，且仅出现一次，则这个"和"项称为最大项。假如一个函数完全由最大项组成，那么这个函数表达式称为标准"和之积"表达式。

例如:

$$\begin{aligned} F(A,B,C) &= (A+B+C)(A+B+\bar{C})(\bar{A}+B+C)(\bar{A}+B+\bar{C}) \\ &= M_0 M_1 M_4 M_5 = \prod M(0,1,4,5) \end{aligned}$$

注意: 变量顺序.

与最小项类似, 有 $\prod_{i=0}^{2^n-1} M_i = 0$

且有 $M_i = \bar{m}_i$ 或 $\bar{M}_i = m_i$

类似地，用代数法求一个函数"最大项之积"的形式，也可分为两步：

第一步：将函数表达式转换成一般"或与"式；

第二步：反复使用 $A = (A + B)(A + \bar{B})$ 将非最大项的"或项"扩展成为最大项

如果给出的函数已经是"与或"式或者是"或与"式，则可直接进行第二步。

二、真值表转换法

一个逻辑函数的真值表与它的最小项表达式和最大项表达式均存在一一对应的关系。

函数 F 的最小项表达式由使 F 取值为1的全部最小项之和组成。函数 F 的最大项表达式由使 F 取值为0的全部最大项之积组成。

例：将 $F = \overline{A}C + A\overline{B}C$ 表示成“最小项之和”
和“最大项之积”的形式。

解：

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

$$\begin{aligned} F(A,B,C) &= \overline{A}\overline{B}C + \overline{A}BC + A\overline{B}C \\ &= \Sigma m(1,3,4) \end{aligned}$$

$$\begin{aligned} F(A,B,C) &= (A + B + C) \cdot (A + \overline{B} + C) \\ &\quad \cdot (\overline{A} + B + \overline{C}) \cdot (\overline{A} + \overline{B} + C) \\ &\quad \cdot (\overline{A} + \overline{B} + \overline{C}) \\ &= \Pi M(0,2,5,6,7) \end{aligned}$$

注意：任何一个逻辑函数的两种标准形式唯一。

2.4 逻辑函数的简化

一般来说, 逻辑函数表达式越简单, 设计出来的电路也就越简单。把逻辑函数简化成最简形式称为逻辑函数的最小化, 有三种常用的方法, 即代数化简法、卡诺图化简法和列表化简法。

2.4.1 代数化简法

该方法运用逻辑代数的公理、定理和规则对逻辑函数进行推导、变换而进行化简，没有固定的步骤可以遵循，主要取决于对公理、定理和规则的熟练掌握及灵活运用程度。有时很难判定结果是否为最简。

一、“与或”式的化简

化简应满足的两个条件：

表达式中“与项”的个数最少；

在满足 的前提下，每个“与项”中的变量个数

最少。
例：化简 $F = A\bar{C} + ABC + AC\bar{D} + CD$

解：

$$\begin{aligned} F &= A(\bar{C} + BC) + C(A\bar{D} + D) \\ &= A[(\bar{C} + B)(\bar{C} + C)] + C[(A + D)(\bar{D} + D)] \\ &= A\bar{C} + AB + AC + CD \\ &= A(\bar{C} + C) + AB + CD \\ &= A(1 + B) + CD = A + CD \end{aligned}$$

例：化简 $F = AB + A\bar{C} + \bar{B}C + B\bar{C} + \bar{B}D + B\bar{D}$
 $+ ADE(F + G)$

解：

$$\begin{aligned} F &= \underline{A\bar{B}\bar{C}} + \bar{B}C + B\bar{C} + \bar{B}D + B\bar{D} + ADE(F + G) \\ &= \underline{A} + \bar{B}C + B\bar{C} + \bar{B}D + B\bar{D} + \underline{ADE(F + G)} \\ &= A + \bar{B}C + B\bar{C} + \bar{B}D + B\bar{D} \\ &= A + \bar{B}C(\bar{D} + D) + B\bar{C} + \bar{B}D + B\bar{D}(C + \bar{C}) \\ &= A + \underline{\bar{B}C\bar{D}} + \underline{\bar{B}CD} + \underline{B\bar{C}} + \underline{\bar{B}D} + \underline{BC\bar{D}} + \underline{B\bar{C}\bar{D}} \\ &= A + C\bar{D} + \bar{B}D + B\bar{C} \end{aligned}$$

二、“或”与“与”式的化简

化简应满足的两个条件：

表达式中“或项”的个数最少；

在满足 的前提下, 每个“或项”
中的变量个数最少。

2.4.2 卡诺图化简法

该方法简单、直观、容易掌握，
当变量个数小于等于6时非常有效，在
逻辑设计中得到广泛应用。

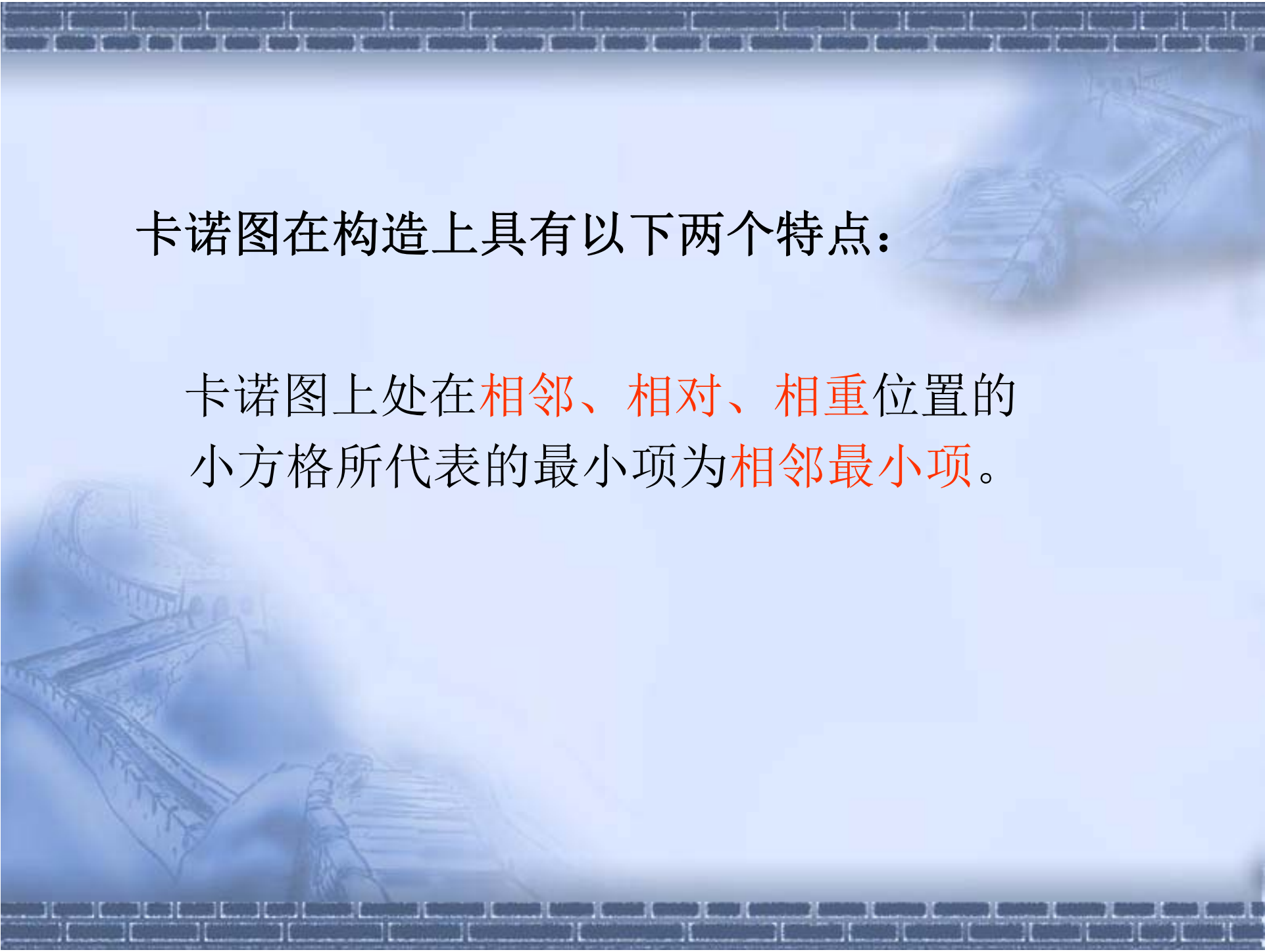
1、卡诺图的性质

根据定理7有 $AB + A\bar{B} = A$, 它表明两个相邻"与项"或"最小项"可以合并为一项, 这一项由两个"与项"中相同的变量组成, 可以消去两个"与项"中不同的变量。

在卡诺图上把相邻最小项所对应的小方格"圈"在一起可进行合并, 以达到用一个简单"与项"代替若干最小项的目的。这样的"圈"称为"卡诺圈"。

2、卡诺图的构成

n 个变量的卡诺图是一种由 2^n 个方格构成的图形, 每一个方格表示逻辑函数的一个最小项, 所有的最小项巧妙地排列成一种能清楚地反映它们相邻关系的方格阵列。因为任意一个逻辑函数都可表示成“最小项之和”的形式, 所以一个函数可用图形中若干小方格(最小项)构成。



卡诺图在构造上具有以下两个特点：

卡诺图上处在相邻、相对、相重位置的小方格所代表的最小项为相邻最小项。

3, 逻辑函数的卡诺图表示

将逻辑函数所对应的最小项在卡诺图的相应方格中标以1，
剩余方格标以0或不标。

二,三,四变量卡诺图

图中每个最小项与相邻(即几何相邻、相对相邻和重叠相邻)的最小项的变量仅且一个相反,其余都相同

二变量卡 诺

	\overline{A}	A
\overline{B}	m0	m2
B	m1	m3

三变量卡诺图

	\overline{A}	A
$\overline{B}\overline{C}$	m0	m4
$\overline{B}C$	m1	m5
BC	m3	m7
$B\overline{C}$	m2	m6

四变量卡诺图

	$\overline{A}\overline{B}$	$\overline{A}B$	AB	$A\overline{B}$
$\overline{C}\overline{D}$	m0	m4	m12	m8
$\overline{C}D$	m1	m5	m13	m9
CD	m3	m7	m15	m11
$C\overline{D}$	m2	m6	m14	m10

B \ A	0	1
0	m_0	m_2
1	m_1	m_3

B \ A	\overline{A} 0	A 1
\overline{B} { 0	$\overline{A} \overline{B}$	$A \overline{B}$
B { 1	$\overline{A} B$	AB

二变量卡诺图

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

Diagram illustrating a Karnaugh map for three variables (A, B, C). The map is a 2x4 grid. The columns are labeled with AB combinations: 00, 01, 11, 10. The rows are labeled with C-bar and C. The cells contain the minterms:

	00	01	11	10
\bar{C}	$\bar{A} \bar{B} \bar{C}$	$\bar{A} B \bar{C}$	$A B \bar{C}$	$A \bar{B} \bar{C}$
C	$\bar{A} \bar{B} C$	$\bar{A} B C$	$A B C$	$A \bar{B} C$

The map is used to illustrate the simplification of the expression $\bar{A} \bar{B} \bar{C} + \bar{A} B \bar{C} + A B \bar{C} + A \bar{B} \bar{C}$.

三变量卡诺图

四变量卡诺图1

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

四变量卡诺图

四变量卡诺图2

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

四变量卡诺图

4、"与或"式的卡诺图表示.

直接将表达式的"与项"或"最小项"所对应的方格标以1.

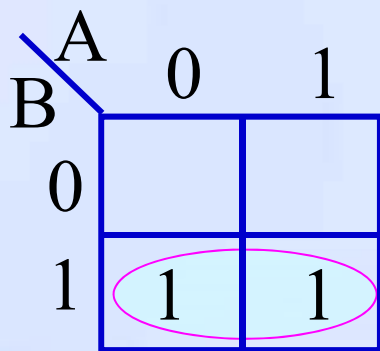
例如: $F(A, B, C) = \bar{A}C + \bar{A}B + BC + A\bar{B}C$

可表示为:

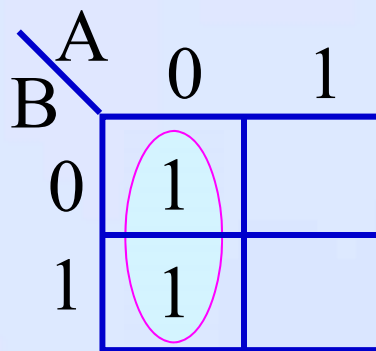
$\begin{array}{c} AB \\ \swarrow \searrow \\ C \end{array}$	00	01	11	10
0		1		
1	1	1	1	1

2、其它形式函数的卡诺图表示要转换成"与或"式再在卡诺图上表示。

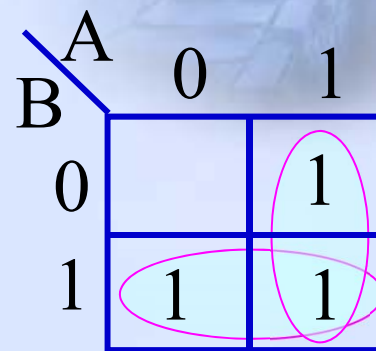
卡诺图的合并情况



$$A + \bar{A} = 1$$



$$B + \bar{B} = 1$$



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

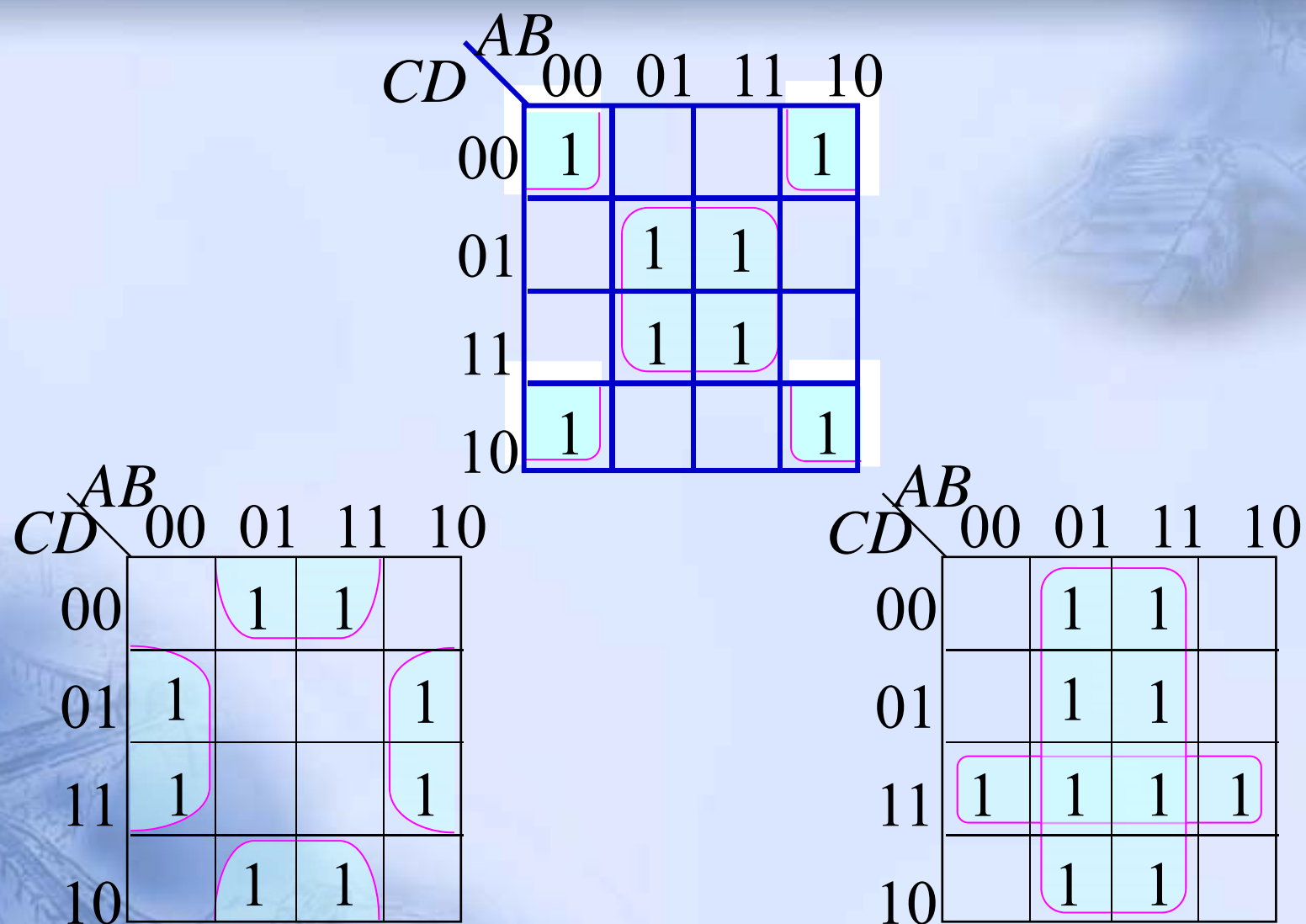
二变量卡诺图的典型合并情况

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

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

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

三变量卡诺图的典型合并情况



四变量卡诺图的典型合并情况

4、卡诺图化简逻辑函数的步骤：

蕴涵项定义：

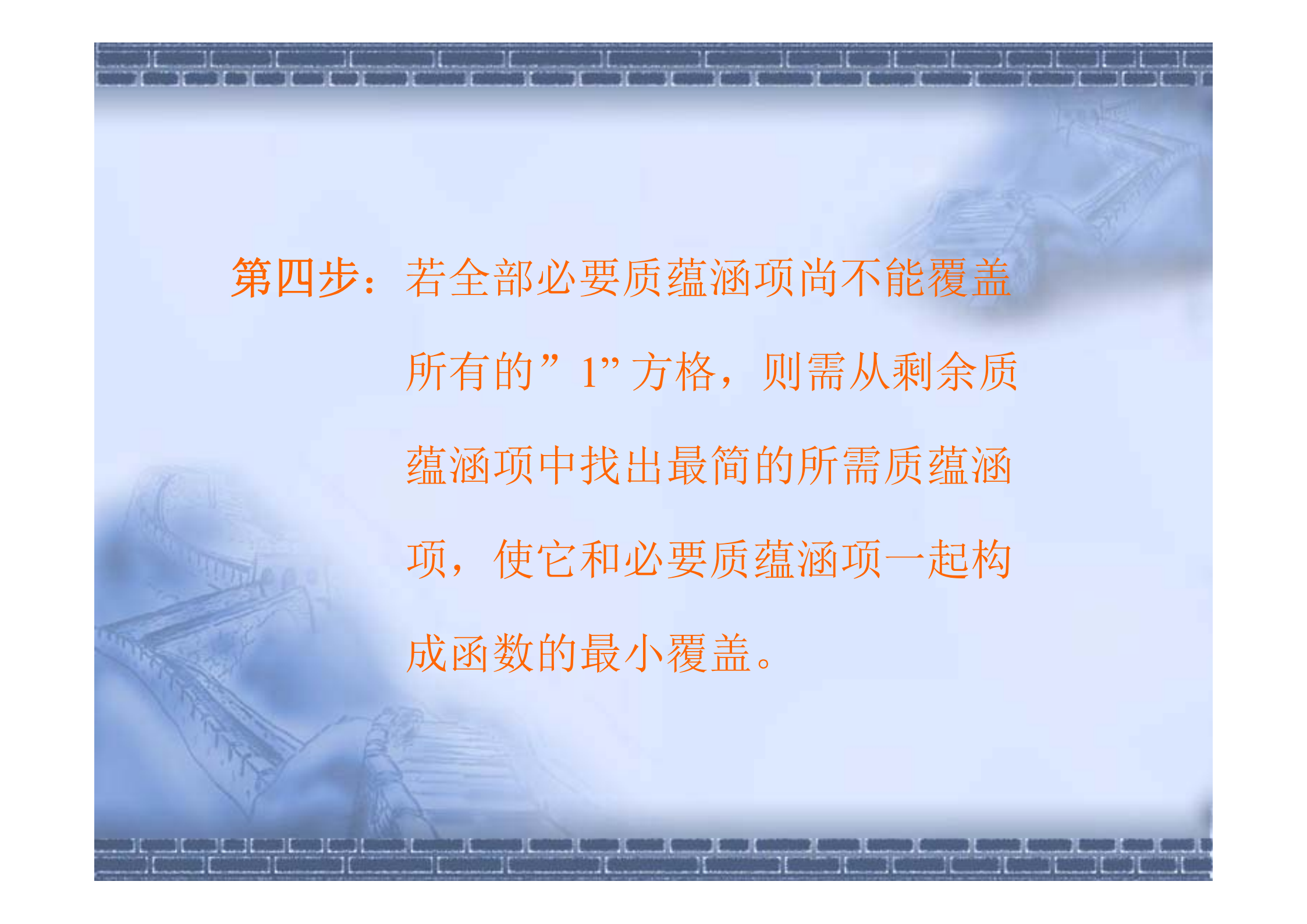
- 1, 蕴涵项： "与或"式中的每一个"与项"称为函数的蕴涵项；
- 2, 质蕴涵项： 不被其它蕴涵项所包含的蕴涵项；
- 3, 必要质蕴涵项： 质蕴涵项中至少有一个最小项不被其它蕴涵项所包含。

5,用卡诺图化简逻辑函数的一般步骤为:

第一步: 作出函数的卡诺图;

第二步: 在卡诺图上圈出函数的全部质蕴涵项 (或者最小项);

第三步: 从全部质蕴涵项中找出所有必要质蕴涵项(最简项);



第四步：若全部必要质蕴涵项尚不能覆盖所有的“1”方格，则需从剩余质蕴涵项中找出最简的所需质蕴涵项，使它和必要质蕴涵项一起构成函数的最小覆盖。

例：用卡诺图化简逻辑函数

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

解：

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

		<i>AB</i>			
		00	01	11	10
<i>CD</i>	00	1			
	01		1	1	
	11	1	1	1	1
	10		1		1

		<i>AB</i>			
		00	01	11	10
<i>CD</i>	00	1*			
	01		1*	1*	
	11	1*	1	1	1
	10		1*		1*

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

例：用卡诺图化简逻辑函数

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

解：

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

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

		AB			
		00	01	11	10
CD	00			1*	1
	01				
	11	1*	1*		
	10	1	1*		1

		AB			
		00	01	11	10
CD	00			1*	1
	01				
	11	1*	1*		
	10	1	1*		1

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

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

例：用卡诺图把逻辑函数

$$F(A, B, C, D) = \prod M(3, 4, 6, 7, 11, 12, 13, 14, 15)$$

化简成最简"或与"表达式。

解: $F(A, B, C, D) = \overline{\overline{\Pi M(3, 4, 6, 7, 11, 12, 13, 14, 15)}}$

$$= \overline{M_3} + \overline{M_4} + \overline{M_6} + \overline{M_7} + \overline{M_{11}} + \overline{M_{12}} + \overline{M_{13}} + \overline{M_{14}} + \overline{M_{15}}$$

$$= m_3 + m_4 + m_6 + m_7 + m_{11} + m_{12} + m_{13} + m_{14} + m_{15}$$

$$= \Sigma m(0,1,2,5,8,9,10)$$

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

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

$$\begin{aligned}
 F(A,B,C,D) &= \overline{CD + AB + B\overline{D}} \\
 &= (\overline{C} + \overline{D})(\overline{A} + \overline{B})(\overline{B} + D)
 \end{aligned}$$

学生练习：

【例】 求 $F = \overline{B}CD + \overline{A}B\overline{D} + \overline{B}C\overline{D} + AB\overline{C} + ABCD$

的最简与或式。

【例】 求 $F = \overline{B}CD + \overline{A}\overline{B}\overline{D} + \overline{B}\overline{C}\overline{D} + A\overline{B}\overline{C} + ABCD$

的最简与或式。

解：① 画出 F 的K图。给出的 F 为一般与或式，将每个与项所覆盖的最小项都填1，K图如图所示。

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

(a)

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

(b)

② 画K圈化简函数 。

③ 写出最简与或式。

本例有两种圈法， 都可以得到最简式。

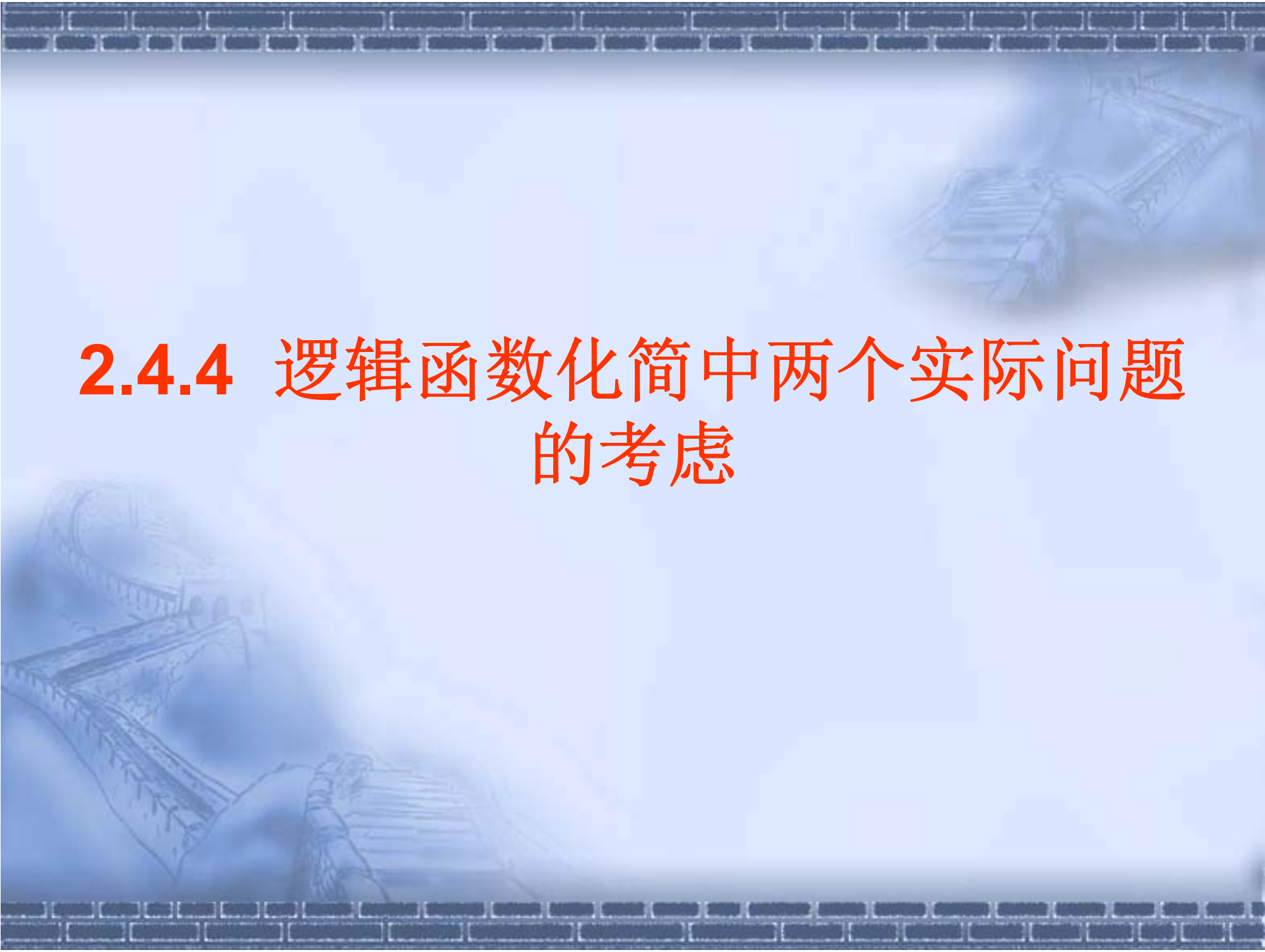
按图2-21(a)圈法：

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

按图2-21(b)圈法：

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

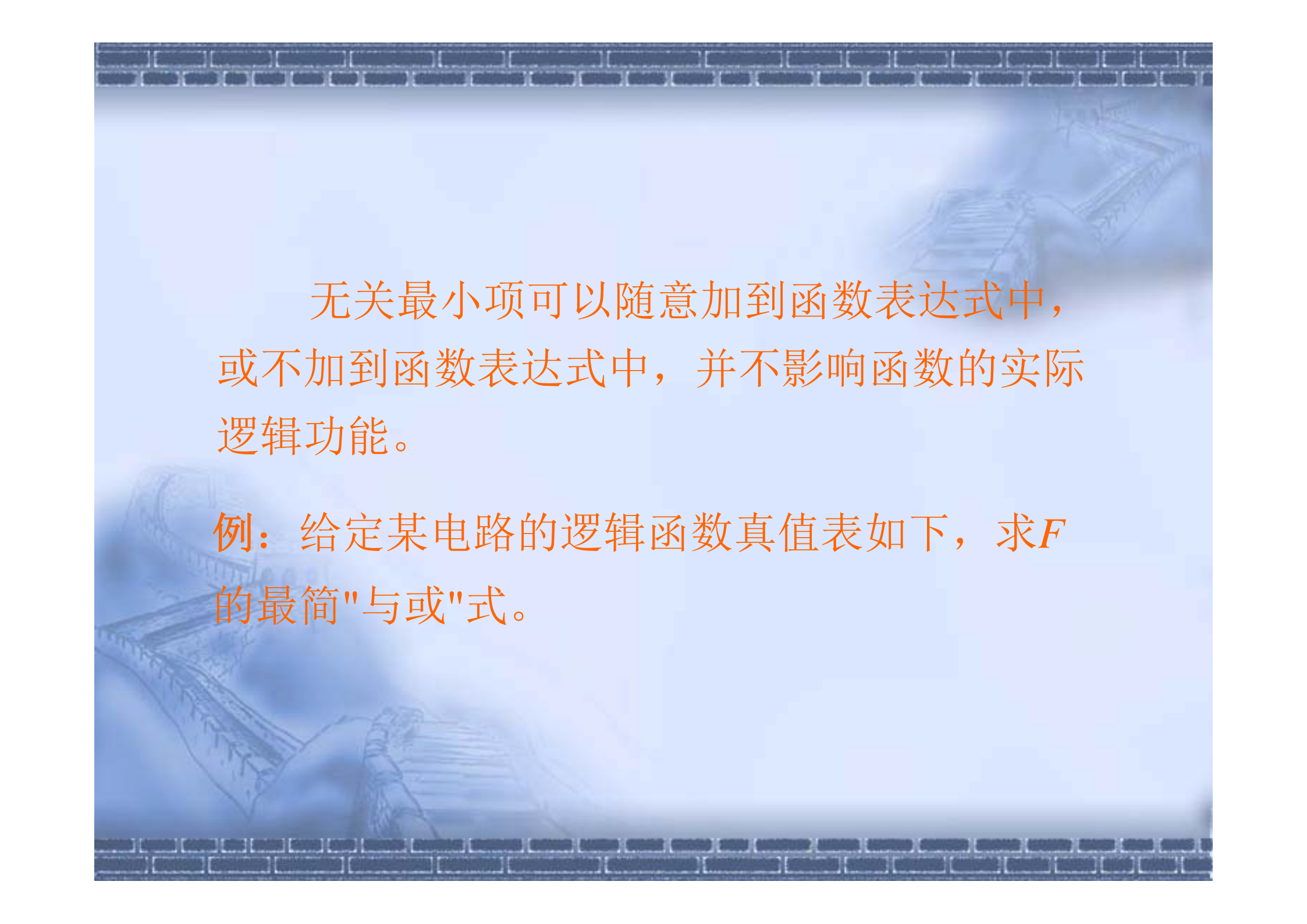
该例说明，逻辑函数的最简式不是惟一的。



2.4.4 逻辑函数化简中两个实际问题的考虑

一、包含无关最小项的逻辑函数的化简

无关最小项： 一个逻辑函数, 如果它的某些输入取值组合因受特殊原因制约而不会再现, 或者虽然每种输入取值组合都可能出现, 但此时函数取值为1还是为0无关紧要, 那么这些输入取值组合所对应的最小项称为无关最小项。



无关最小项可以随意加到函数表达式中，
或不加到函数表达式中，并不影响函数的实际
逻辑功能。

例：给定某电路的逻辑函数真值表如下，求 F
的最简"与或"式。

<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>F</i>
0	0	0	0	<i>d</i>
0	0	0	1	<i>d</i>
0	0	1	0	<i>d</i>
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	<i>d</i>
1	1	1	0	<i>d</i>
1	1	1	1	<i>d</i>

<i>AB</i> <i>CD</i>	00	01	11	10
00		1	1	
01		1		
11	1			1
10				1

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

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

		AB			
		00	01	11	10
CD	00	<i>d</i>	1	1	
	01	<i>d</i>	1	<i>d</i>	
	11	1		<i>d</i>	1
	10	<i>d</i>		<i>d</i>	1

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

二、多输出逻辑函数的化简.

对于多输出逻辑函数，如果孤立地将单个输出一一化简，然后直接拼在一起，通常并不能保证整个电路最简，因为各个输出函数之间往往存在可供共享的部分。

多输出逻辑函数化简的标准:

- 1: 所有逻辑表达式包含的不同"与项"总数最小;
- 2: 在满足上述条件的前提下, 各不同"与项"中所含的变量总数最少。