

第四章 组合逻辑电路

主讲教师：何云峰



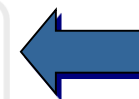
本章知识要点

- 组合逻辑电路分析和设计的基本方法
- 组合逻辑电路设计中几个常见的实际问题及其处理
- 组合逻辑电路中的竞争与险象问题

提 纲

1

组合逻辑电路分析



2

组合逻辑电路设计

3

组合逻辑电路的险象

逻辑电路的分类

□ 组合逻辑电路

- Combinational Logic Circuit
- 电路输出仅取决于当时的输入，而与过去的输入情况无关
- 可完成各种复杂的逻辑功能
- 是时序逻辑电路的组成部分

□ 时序逻辑电路

- Sequential Logic Circuits
- 电路输出不仅取决于当时的输入，而且也与过去的输入情况有关，即与过去的电路状态有关

组合逻辑电路

□ 定义

- 若逻辑电路在任何时刻产生的稳定输出值仅仅取决于该时刻各输入值的组合，而与过去的输入值无关



- 输入信号: X_0, X_1, \dots, X_n
- 输出信号: y_0, y_1, \dots, y_m
- $y = f(x)$
 - $y_i = f_i(X_0, X_1, \dots, X_n)$

组合逻辑电路

□特点

- 由逻辑门电路组成
- 不包含任何记忆元件
- 信号是单向传输的，不存在反馈回路

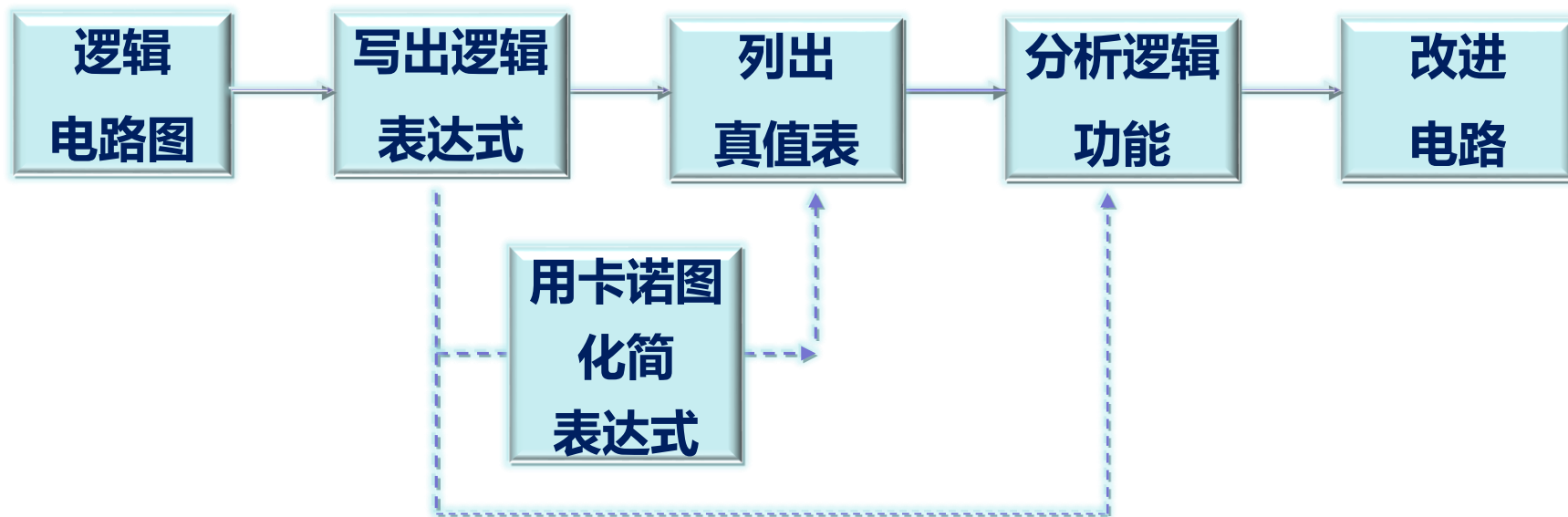
组合逻辑电路的分析

□ 目的

- 根据给定电路，分析输出与输入之间的逻辑关系
- 得出电路的逻辑功能的描述
- 评估此电路的性能
- 进一步改进电路

组合电路的分析

□ 步骤



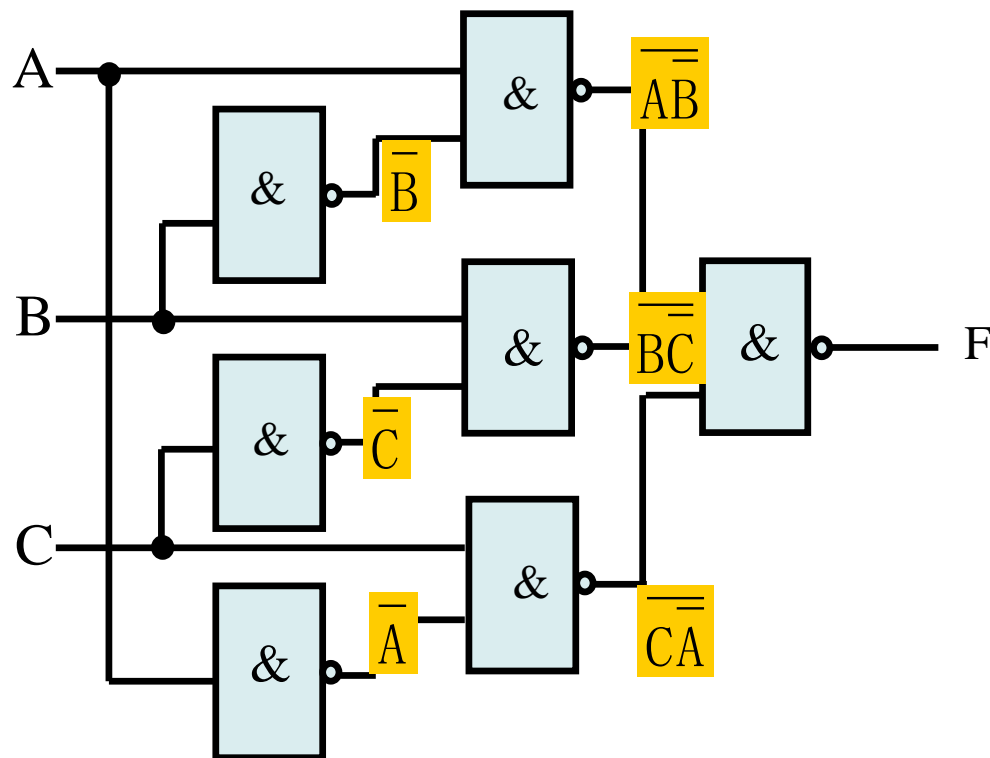
组合逻辑电路的分析

□代数法

- 根据电路逐级写出各门的输出表达式，直至写出整个电路的输出逻辑表达式

组合逻辑电路的分析

□ 例 1 分析下图所示组合逻辑电路



$$\bar{A}B \cdot B\bar{C} \cdot C\bar{A}$$

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

组合逻辑电路的分析

□ 真值表

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

组合逻辑电路的分析

□ 真值表

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

组合逻辑电路的分析

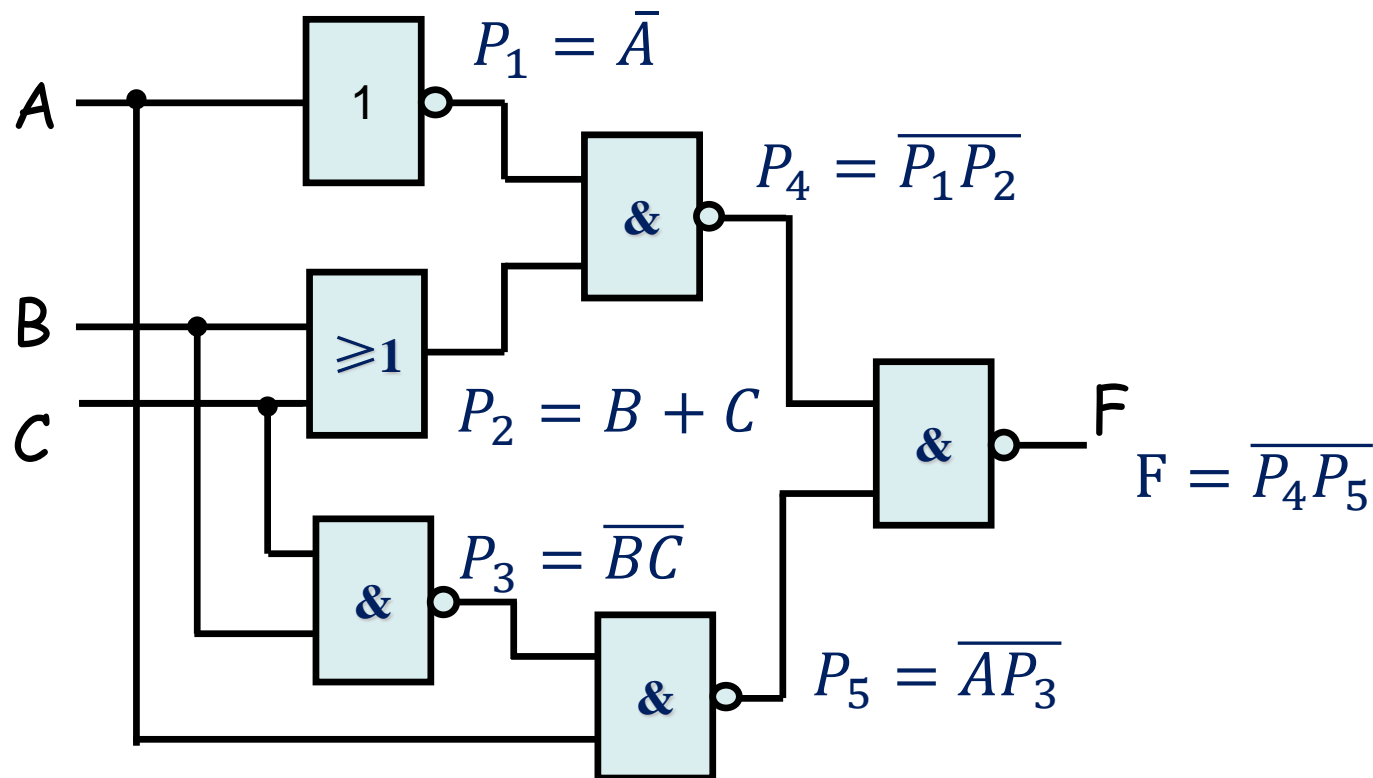
□ 逻辑功能

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

三变量非一致电路

组合逻辑电路的分析

□ 例 2 分析下图所示组合逻辑电路。



组合逻辑电路的分析

□ 逻辑表达式

$$\begin{aligned} F &= \overline{P_4 P_5} \\ &= \overline{P_4} + \overline{P_5} \\ &= P_1 P_2 + AP_3 \\ &= \bar{A}(B + C) + A\bar{B}\bar{C} \\ &= \bar{A}(B + C) + A(\bar{B} + \bar{C}) \\ &= A \oplus B + A \oplus C \end{aligned}$$

$$* \quad P_4 = \overline{P_1 P_2}$$

$$* \quad P_5 = \overline{AP_3}$$

$$* \quad P_1 = \bar{A}$$

$$* \quad P_2 = B + C$$

$$* \quad P_3 = \bar{B}\bar{C}$$

组合逻辑电路的分析

□ 真值表

□ 逻辑功能

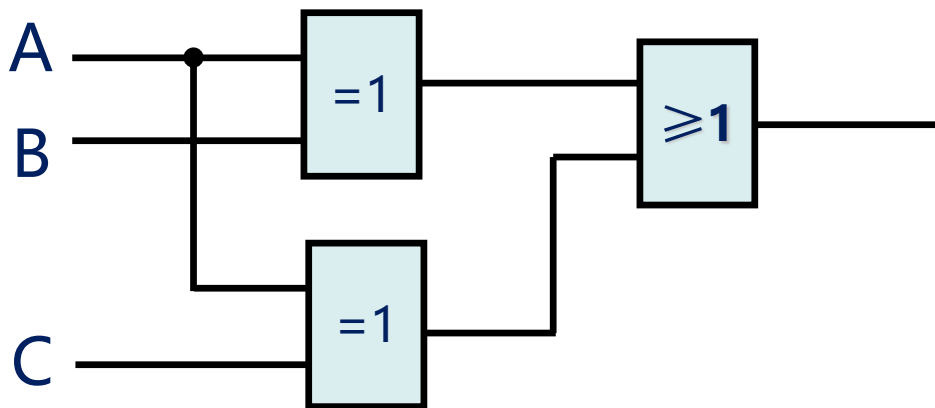
– 三变量不一致电路

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

组合逻辑电路的分析

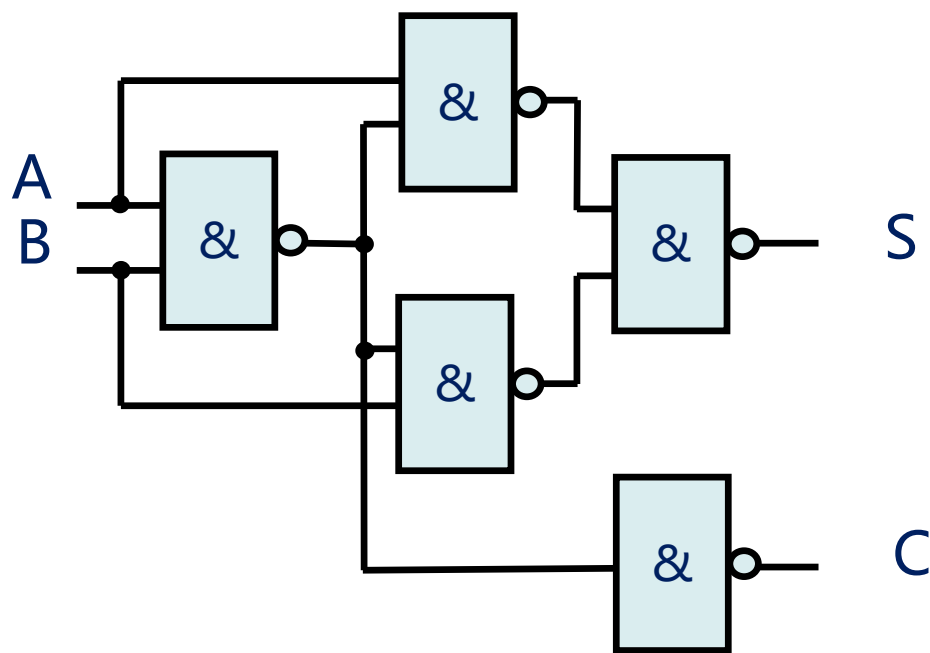
□ 改进电路

$$- F = A \oplus B + A \oplus C$$



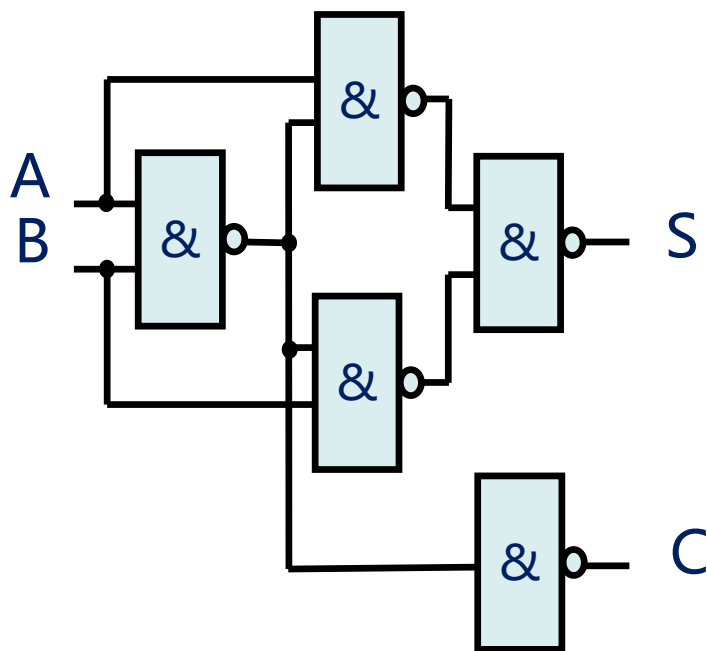
组合逻辑电路的分析

□ 例 3 分析下图所示逻辑电路



组合逻辑电路的分析

□ 写出逻辑表达式



$$\begin{aligned} S &= \overline{\overline{AB} \cdot A \cdot \overline{AB} \cdot B} \\ &= \overline{AB} \cdot A + \overline{AB} \cdot B \\ &= (\bar{A} + \bar{B}) \cdot A + (\bar{A} + \bar{B}) \cdot B \\ &= \bar{A}B + A\bar{B} \\ &= A \oplus B \end{aligned}$$

$$C = AB$$

组合逻辑电路的分析

□ 真值表

$$S = A \oplus B \quad C = AB$$

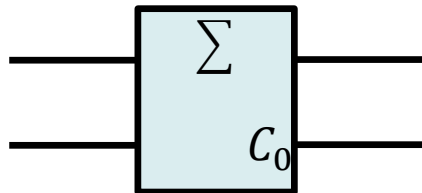
A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

组合逻辑电路的分析

□ 逻辑功能

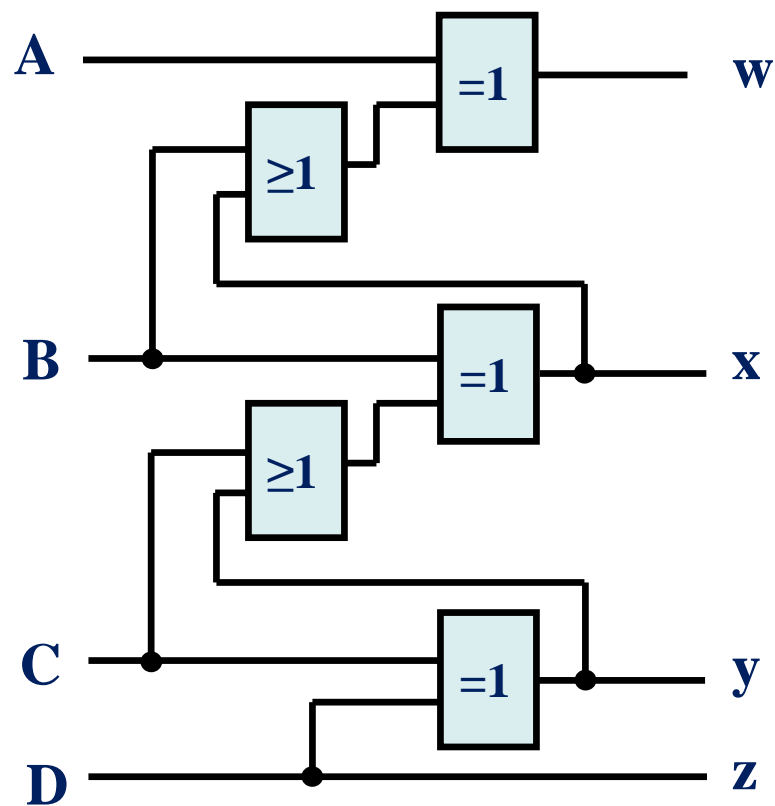
- 若将A、B分别作为一位二进制数
 - S是 A、B 相加的 “和”
 - C是相加产生的 “进位”
- 该电路称作 “半加器”
 - 实现两个一位二进制数加法运算
 - 集成芯片

A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1



组合逻辑电路的分析

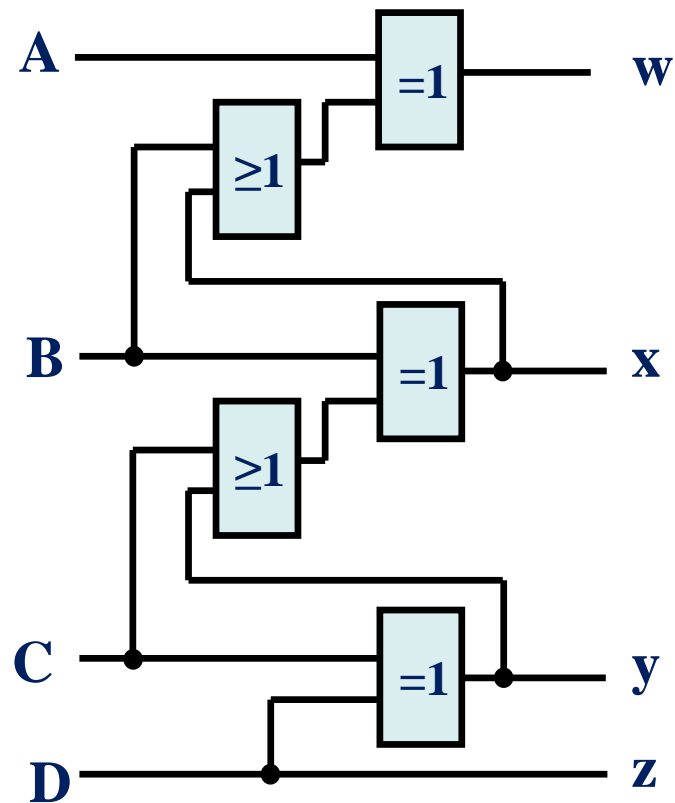
□ 例4：分析如下逻辑电路。



组合逻辑电路的分析

□ 写出逻辑表达式

- $w = A \oplus (B + x)$
- $x = B \oplus (C + y)$
- $y = C \oplus D$
- $z = D$



组合逻辑电路的分析

□ 逻辑表达式的变换

$$\begin{aligned} - w &= A \oplus (B + x) \\ &= A \oplus (B + B \oplus (C + D)) \\ &= A \oplus (B + C + D) \end{aligned}$$

$$\begin{aligned} \leftarrow x &= B \oplus (C + y) \\ y &= C \oplus D \end{aligned}$$

$$\begin{aligned} - x &= B \oplus (C + y) \\ &= B \oplus (C + C \oplus D) \\ &= B \oplus (C + D) \end{aligned}$$

$$- y = C \oplus D$$

$$- z = D$$

包含无关条件的组合逻辑电路设计

□ 真值表

$$- w = A \oplus (B + C + D)$$

$$x = B \oplus (C + D)$$

$$- y = C \oplus D$$

$$z = D$$

<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>w</i>	<i>x</i>	<i>y</i>	<i>z</i>
0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	1
0	0	1	0	1	1	1	0
0	0	1	1	1	1	0	1
0	1	0	0	1	1	0	0
0	1	0	1	1	0	1	1
0	1	1	0	1	0	1	0
0	1	1	1	1	0	0	1

<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>w</i>	<i>x</i>	<i>y</i>	<i>z</i>
1	0	0	0	1	0	0	0
1	0	0	1	0	1	1	1
1	0	1	0	0	1	1	0
1	0	1	1	0	1	0	1
1	1	0	0	0	1	0	0
1	1	0	1	0	0	1	1
1	1	1	0	0	0	1	0
1	1	1	1	0	0	0	1

包含无关条件的组合逻辑电路设计

□ 逻辑功能

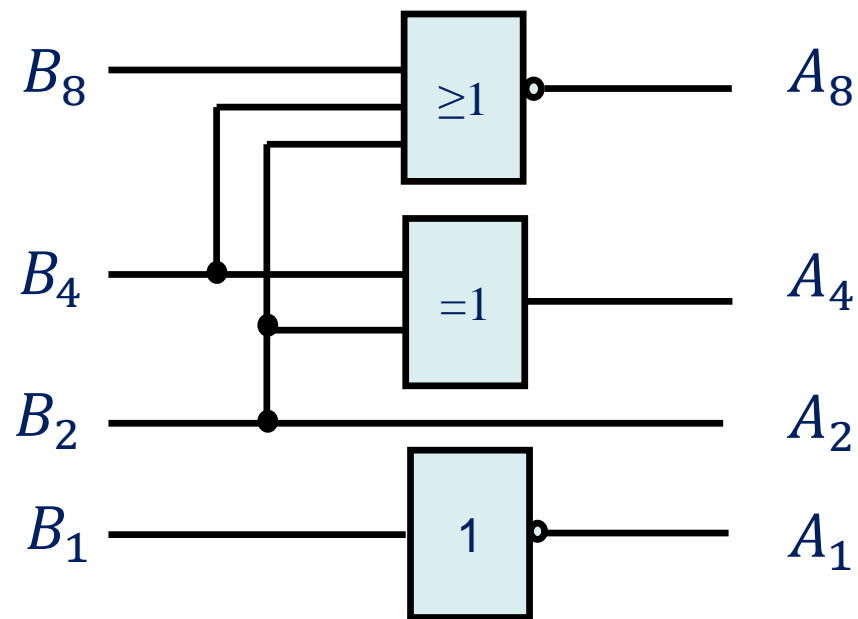
- 二进制变补器
- 16变补器

<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>w</i>	<i>x</i>	<i>y</i>	<i>z</i>
0	0	0	0	0	0	0	0
0	0	0	1	1	1	1	1
0	0	1	0	1	1	1	0
0	0	1	1	1	1	0	1
0	1	0	0	1	1	0	0
0	1	0	1	1	0	1	1
0	1	1	0	1	0	1	0
0	1	1	1	1	0	0	1

<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>w</i>	<i>x</i>	<i>y</i>	<i>z</i>
1	0	0	0	1	0	0	0
1	0	0	1	0	1	1	1
1	0	1	0	0	1	1	0
1	0	1	1	0	1	0	1
1	1	0	0	0	1	0	0
1	1	0	1	0	0	1	1
1	1	1	0	0	0	1	0
1	1	1	1	0	0	0	1

组合逻辑电路的分析

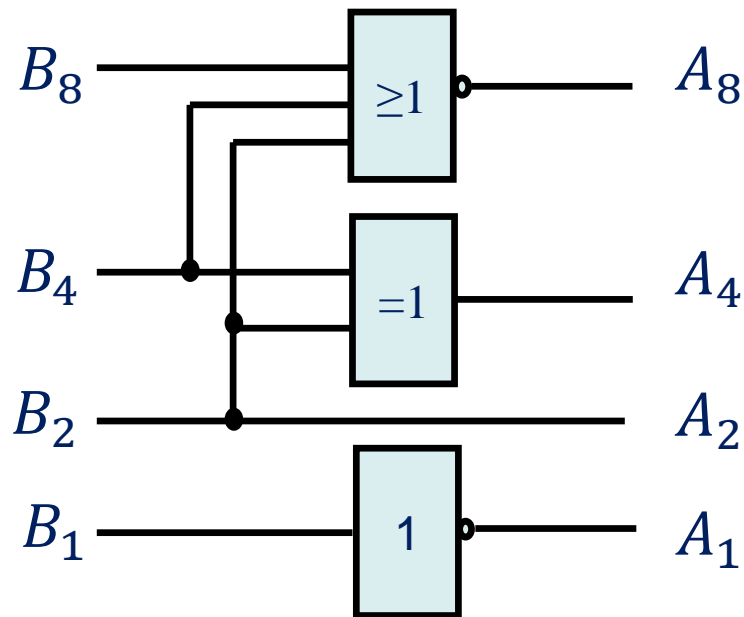
□ 例5：分析如下逻辑电路。



组合逻辑电路的分析

□ 逻辑表达式

- $A_8 = \overline{B_8 + B_4 + B_2}$
 $= \overline{B_8} \overline{B_4} \overline{B_2}$
- $A_4 = B_4 \oplus B_2$
- $A_2 = B_2$
- $A_1 = \overline{B_1}$



组合逻辑电路的分析

□ 真值表

$$- A_8 = \overline{B_8} \overline{B_4} \overline{B_2} \quad A_4 = B_4 \oplus B_2 \quad A_2 = B_2 \quad A_1 = \overline{B_1}$$

B_8 B_4 B_2 B_1	A_8 A_4 A_2 A_1	B_8 B_4 B_2 B_1	A_8 A_4 A_2 A_1
0 0 0 0	1 0 0 1	1 0 0 0	0 0 0 1
0 0 0 1	1 0 0 0	1 0 0 1	0 0 0 0
0 0 1 0	0 1 1 1	1 0 1 0	0 1 1 1
0 0 1 1	0 1 1 0	1 0 1 1	0 1 1 0
0 1 0 0	0 1 0 1	1 1 0 0	0 1 0 1
0 1 0 1	0 1 0 0	1 1 0 1	0 1 0 0
0 1 1 0	0 0 1 1	1 1 1 0	0 0 1 1
0 1 1 1	0 0 1 0	1 1 1 1	0 0 1 0

组合逻辑电路的分析

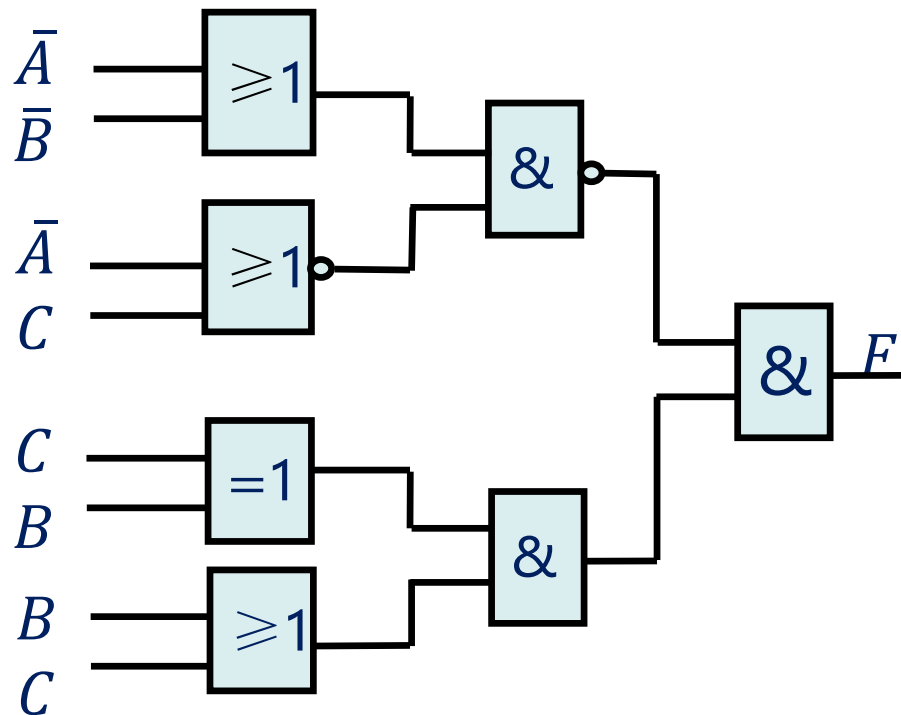
□ 真值表

– BCD码对9变补器

$B_8 B_4 B_2 B_1$	$A_8 A_4 A_2 A_1$	$B_8 B_4 B_2 B_1$	$A_8 A_4 A_2 A_1$
0 0 0 0	1 0 0 1	1 0 0 0	0 0 0 1
0 0 0 1	1 0 0 0	1 0 0 1	0 0 0 0
0 0 1 0	0 1 1 1	1 0 1 0	0 1 1 1
0 0 1 1	0 1 1 0	1 0 1 1	0 1 1 0
0 1 0 0	0 1 0 1	1 1 0 0	0 1 0 1
0 1 0 1	0 1 0 0	1 1 0 1	0 1 0 0
0 1 1 0	0 0 1 1	1 1 1 0	0 0 1 1
0 1 1 1	0 0 1 0	1 1 1 1	0 0 1 0

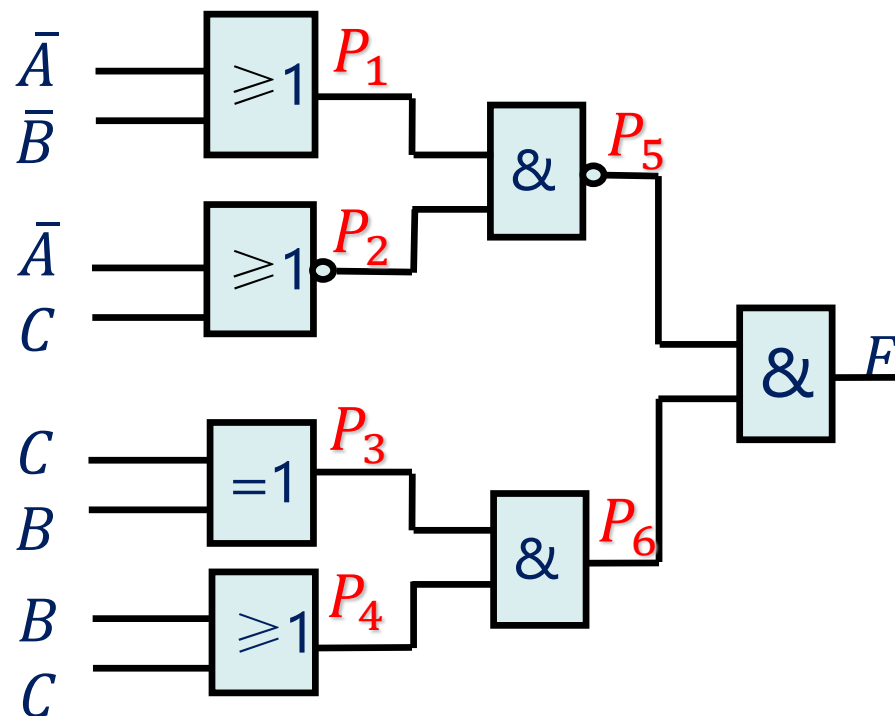
组合逻辑电路的分析

□ 例6：分析如下逻辑电路。



组合逻辑电路的分析

□ 例6：分析如下逻辑电路。



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

组合逻辑电路的分析

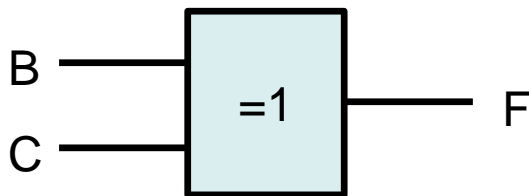
□ 逻辑表达式化简

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

组合逻辑电路的分析

□ 逻辑功能：异或

□ 电路优化



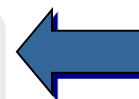
提 纲

1

组合逻辑电路分析

2

组合逻辑电路设计



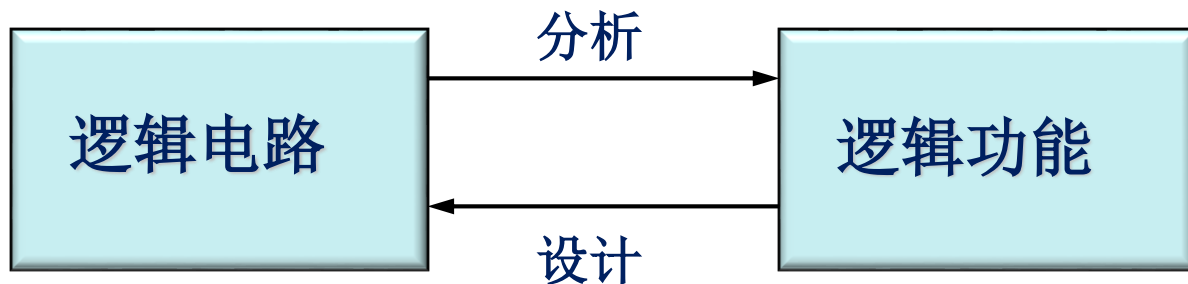
3

组合逻辑电路的险象

组合逻辑电路设计

□ 逻辑设计

- 根据问题要求完成的逻辑功能，求出在特定条件下实现给定功能的逻辑电路
- 逻辑综合



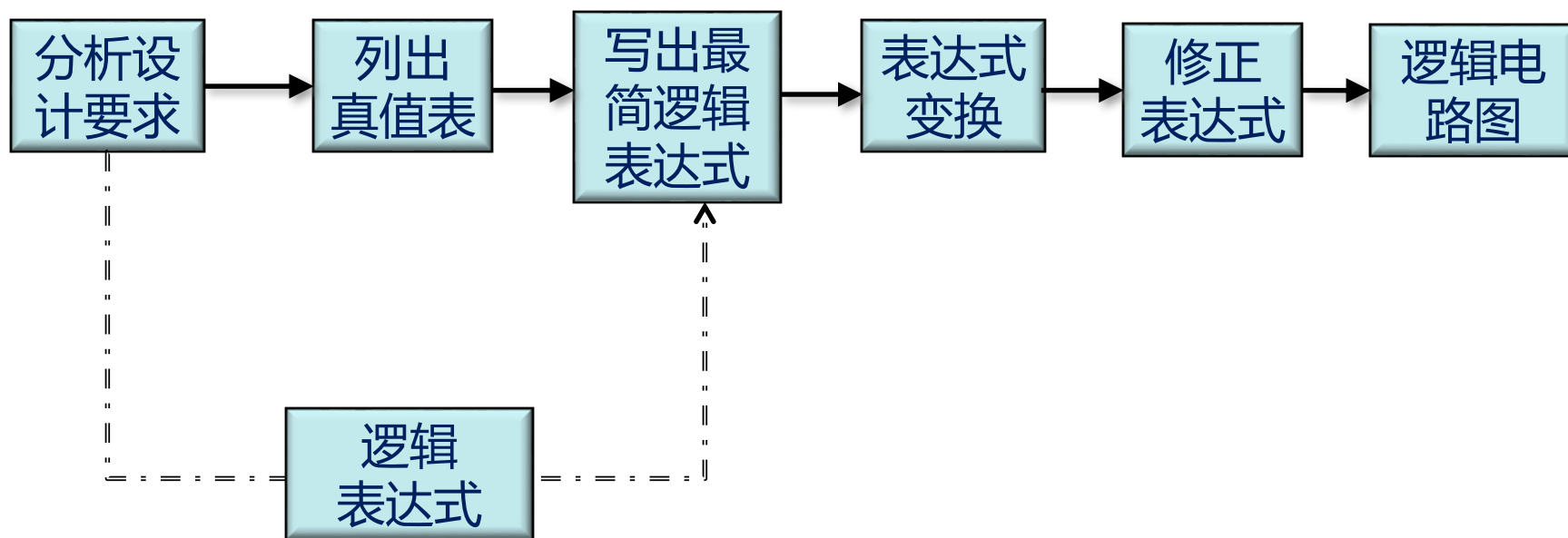
组合逻辑电路设计

□任务

- 将文字描述的设计要求抽象为一种逻辑关系
 - 实际应用中提出的各种设计要求一般是用文字形式描述的
 - 对于组合逻辑电路，即抽象出描述问题的逻辑表达式

组合逻辑电路设计

□ 步骤



分析设计要求

□ 建立给定问题的逻辑描述

□ 关键：正确理解设计要求

□ 确定

- 电路的输入和输出
- 函数与变量之间的逻辑关系
 - 逻辑表达式

□ 方法

- 真值表法
- 分析法

组合逻辑电路设计

□ 例1.设计一个三变量的“多数表决电路”。

– 分析

– 输入

- A 、 B 、 C --- 分别代表参加表决的3个成员
- 0表示反对，1表示赞成

– 输出

- F --- 表示表决结果
- 0表示决议被否定，1表示决议通过

– 逻辑关系：当3个变量 A 、 B 、 C 中有2个或2个以上取值为1时，函数 F 的值为1，其他情况下函数 F 的值为0。

组合逻辑电路设计

□ 建立给定问题的逻辑描述

– 真值表法

– 逻辑表达式

- $F(A, B, C) = \sum m(3, 5, 6, 7)$

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

组合逻辑电路设计

□ $F(A, B, C) = \sum m(3, 5, 6, 7)$

□ 最简表达式

- $F = AB + AC + BC$

□ 选择器件

- 与非门

- $F = \overline{\overline{AB + AC + BC}}$

$= \overline{\overline{AB} \overline{AC} \overline{BC}}$

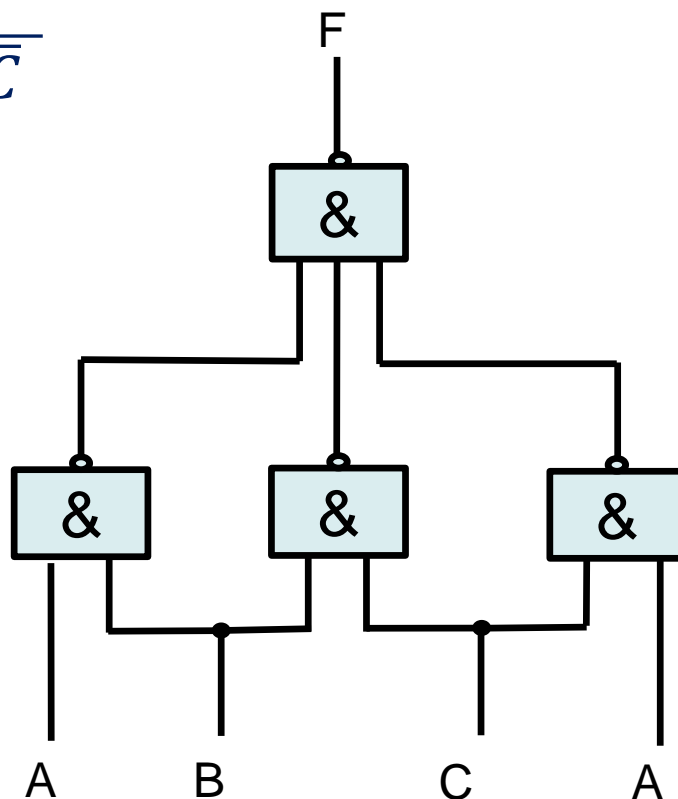
AB		00	01	11	10
C	0	0	0	1	0
	1	0	1	1	1

(b)

组合逻辑电路设计

□ 画出逻辑电路图

$$- F = \overline{AB} \overline{AC} \overline{BC}$$



组合逻辑电路设计

□ 真值表法

- 优点：规整、清晰
- 缺点：不方便，尤其当变量较多时十分麻烦

□ “分析法”

- 通过对设计要求的分析、理解，直接写出逻辑表达式

组合逻辑电路设计

□ 例2.设计一个比较两个三位二进制数是否相等的数值比较器。

□ 分析

– 输入：

- $A = a_0a_1a_2, B = b_0b_1b_2$
- 6个输入变量

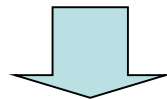
– 输出：

- 比较结果为函数 F
- 当 $A = B$ 时， F 为1，否则 F 为0

组合逻辑电路设计

□ 分析

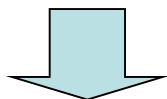
$$A=B$$



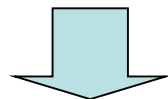
$$\begin{array}{c} a_0 = b_0 \\ f_0 \end{array}$$

$$\begin{array}{c} a_1 = b_1 \\ f_1 \end{array}$$

$$\begin{array}{c} a_2 = b_2 \\ f_2 \end{array}$$



$$\begin{array}{llll} f_0 = 1 & \Rightarrow & a_0 = b_0 & \Rightarrow & f_0 = \overline{a_0 \oplus b_0} \\ f_1 = 1 & \Rightarrow & a_1 = b_1 & \Rightarrow & f_1 = \overline{a_1 \oplus b_1} \\ f_2 = 1 & \Rightarrow & a_2 = b_2 & \Rightarrow & f_2 = \overline{a_2 \oplus b_2} \end{array}$$

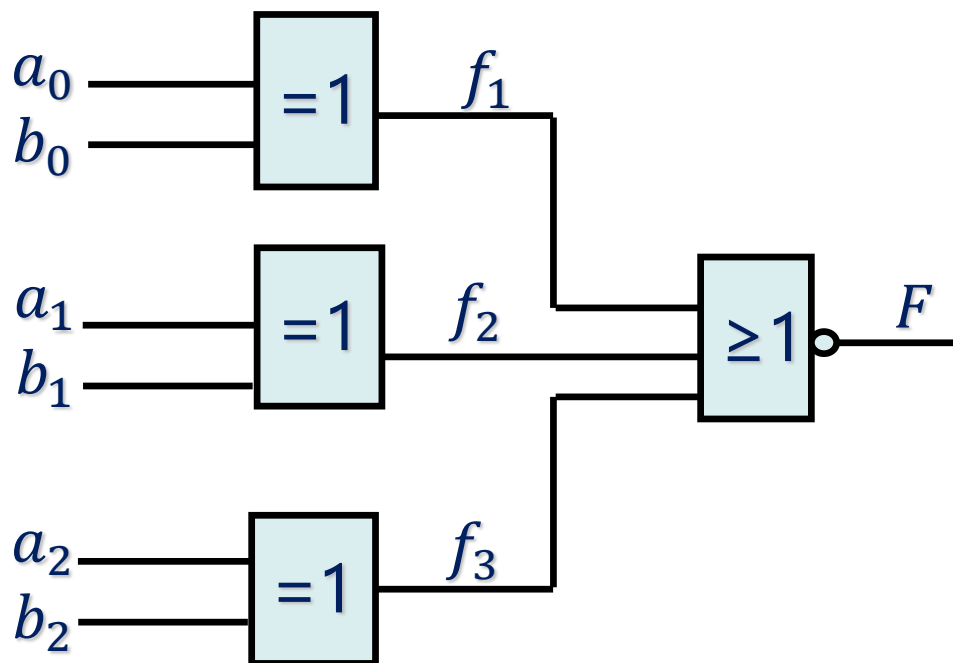


$$F = f_0 f_1 f_2 = \overline{(a_0 \oplus b_0) + (a_1 \oplus b_1) + (a_2 \oplus b_2)}$$

组合逻辑电路设计

□ 电路图

$$F = \overline{(a_0 \oplus b_0) + (a_1 \oplus b_1) + (a_2 \oplus b_2)}$$



多输出函数的组合逻辑电路设计

□ 多输出函数的组合逻辑电路

- 由同一组输入变量产生多个输出函数
- 应该将多个输出函数当作一个整体考虑，而不应该将其截然分开
- 关键：
 - 在函数化简时找出各输出函数的公用项，实现对逻辑门的“共享”

多输出函数的组合逻辑电路设计

□例3. 设计一个全加器。

- 要求：能对两个1位二进制数及来自低位的“进位”进行相加，产生本位“和”及向高位“进位”的逻辑电路
- 全加器可用于实现两个n位数相加

$$\begin{array}{r} A_{n-1} A_{n-2} \cdots A_i \cdots A_1 A_0 \\ + \quad B_{n-1} B_{n-2} \cdots B_i \cdots B_1 B_0 \\ \hline C_{i-1} \qquad \qquad \qquad C_i \qquad \qquad \qquad S_i \end{array}$$

多输出函数的组合逻辑电路设计

□分析：

— 输入端

- A_i ：被加数
- B_i ：加数
- C_{i-1} ：来自低位的进位输入

— 输出端

- S_i ：本位和
- C_i ：向高位的进位

$$\begin{array}{r} A_{n-1} A_{n-2} \cdots A_i \cdots A_1 A_0 \\ + \quad B_{n-1} B_{n-2} \cdots B_i \cdots B_1 B_0 \\ \hline C_i \qquad \qquad \qquad C_{i-1} \qquad \qquad \qquad S_i \end{array}$$

多输出函数的组合逻辑电路设计

□ 真值表

- $S_i = \sum m (1,2,4,7)$

- $C_i = \sum m (3,5,6,7)$

A_i	B_i	C_{i-1}	S_i	C_i
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

多输出函数的组合逻辑电路设计

□ 函数化简

$$- S_i = \sum m (1,2,4,7)$$

$$= \overline{A_i}\overline{B_i}C_{i-1} + \overline{A_i}B_i\overline{C_{i-1}} + A_i\overline{B_i}\overline{C_{i-1}} + A_iB_iC_{i-1}$$

$$= (\overline{A_i}B_i + A_i\overline{B_i}) \overline{C_{i-1}} + (\overline{A_i}\overline{B_i} + A_iB_i) C_{i-1}$$

$$= (A_i \oplus B_i) \overline{C_{i-1}} + \overline{A_i \oplus B_i} C_{i-1}$$

$$= A_i \oplus B_i \oplus C_{i-1}$$

		S_i			
		A_iB_i			
C_{i-1}		00	01	11	10
	0		1		1
	1	1		1	

多输出函数的组合逻辑电路设计

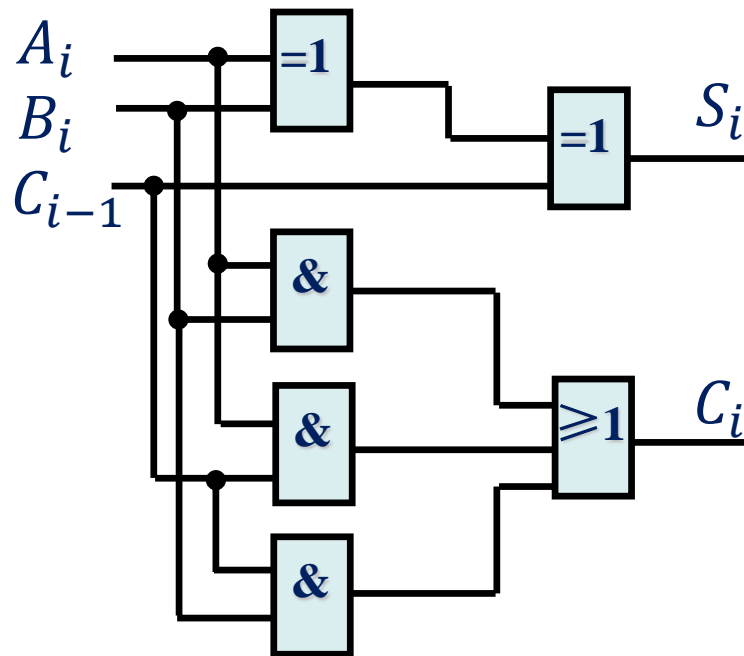
□ 函数化简

$$\begin{aligned} - C_i &= \sum m(3,5,6,7) \\ &= A_i B_i + A_i C_{i-1} + B_i C_{i-1} \end{aligned}$$

$A_i B_i$ C_{i-1}		C_i			
		00	01	11	10
0				1	
1			1	1	1

多输出函数的组合逻辑电路设计

□ 电路图



- 该电路就单个函数而言， S_i 、 C_i 均已达到最简
- 从整体考虑则并非最简

多输出函数的组合逻辑电路设计

□ 函数化简

		S_i			
$C_{i-1} \backslash A_i B_i$		00	01	11	10
	0		1		1
1		1		1	

		C_i			
$C_{i-1} \backslash A_i B_i$		00	01	11	10
	0			1	
1			1	1	1

$$S_i = A_i \oplus B_i \oplus C_{i-1}$$

$$C_i = \sum m(3,5,6,7)$$

$$= A_i B_i + \bar{A}_i B_i C_{i-1} + A_i \bar{B}_i C_{i-1}$$

$$= A_i B_i + (\bar{A}_i B_i + A_i \bar{B}_i) C_{i-1}$$

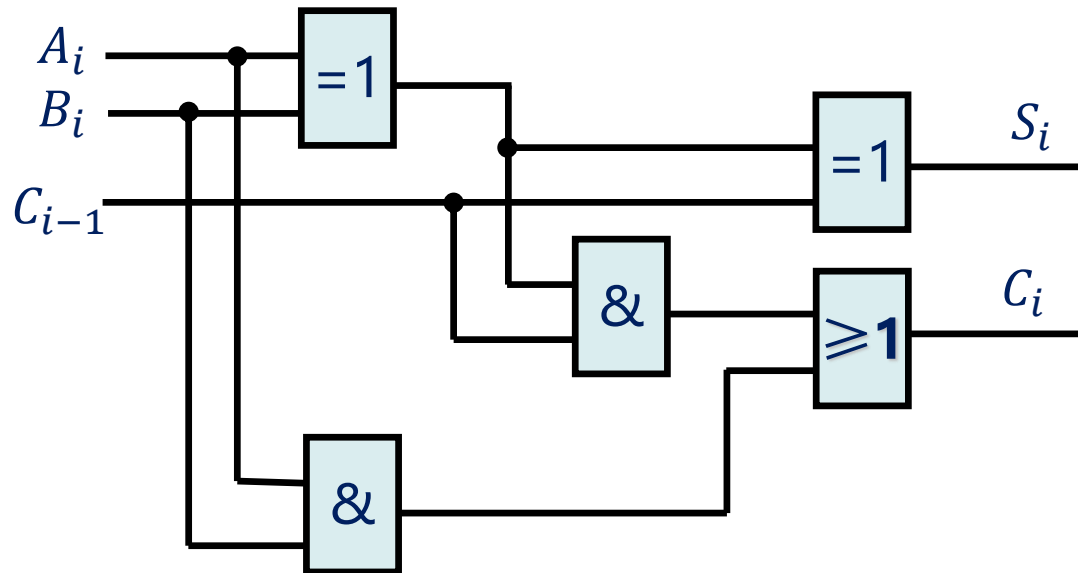
$$= A_i B_i + (A_i \oplus B_i) C_{i-1}$$

多输出函数的组合逻辑电路设计

□ 电路图

$$S_i = A_i \oplus B_i \oplus C_{i-1}$$

$$C_i = A_i B_i + (A_i \oplus B_i) C_{i-1}$$



多输出函数的组合逻辑电路设计

□例4.设计一个乘法器，用于产生两个2位二进制数相乘的积

□分析

– 输入： A_1A_0 和 B_1B_0

– 输出函数

- $\text{Max}(A_1A_0) = 11$ $\text{Max}(B_1B_0) = 11$
- $\text{Max}(A_1A_0 \times B_1B_0) = 11$
- 相乘的积为 $M_3M_2M_1M_0$

多输出函数的组合逻辑电路设计

$$\begin{array}{rcccl}
 & & A_1 & A_0 & \\
 & & \times B_1 & B_0 & \\
 \hline
 & & A_1 \times B_0 & A_0 \times B_0 & \\
 + A_1 \times B_1 & & A_0 \times B_1 & & \\
 \hline
 C_2 & A_1 \times B_1 + C_1 & A_1 \times B_0 + A_0 \times B_1 & A_0 \times B_0 & \\
 M_3 & M_2 & M_1 & M_0 &
 \end{array}$$

多输出函数的组合逻辑电路设计

- $M_0 = A_0 \times B_0$
- $M_1 = A_1 \times B_0 + A_0 \times B_1$
- $M_2 = A_1 \times B_1 + C_1$
- $M_3 = C_2$

			A_1	A_0
			$\times B_1$	B_0
			<hr/>	
			$A_1 \times B_0$	$A_0 \times B_0$
	$+ A_1 \times B_1$		$A_0 \times B_1$	
	<hr/>			
C_2	$A_1 \times B_1 + C_1$	$A_1 \times B_0 + A_0 \times B_1$	$A_0 \times B_0$	
M_3	M_2	M_1	M_0	

多输出函数的组合逻辑电路设计

□ 积项：两输入与门

□ 和项：异或门

□ C_1

- A_1B_0 和 A_0B_1 相加产生的进位

- $C_1 = 1 \Leftrightarrow A_1B_0 = 1 \ \& \ A_0B_1 = 1$

- $C_1 = A_0B_1A_1B_0$

多输出函数的组合逻辑电路设计

□ C_2

□ C_1 和 A_1B_1 相加产生的进位

$$- C_2 = 1 \Leftrightarrow C_1 = 1 \quad \& \quad A_1B_1 = 1$$

$$- C_2 = C_1A_1B_1 = A_0B_1A_1B_0$$

多输出函数的组合逻辑电路设计

$$\square M_0 = A_0 \times B_0$$

$$- M_0 = A_0 B_0$$

$$\square M_1 = A_1 \times B_0 + A_0 \times B_1$$

$$- M_1 = A_0 B_1 \oplus A_1 B_0$$

$$\square M_2 = A_1 \times B_1 + C_1$$

$$- M_2 = C_1 \oplus A_1 B_1 = A_0 B_1 A_1 B_0 \oplus A_1 B_1$$

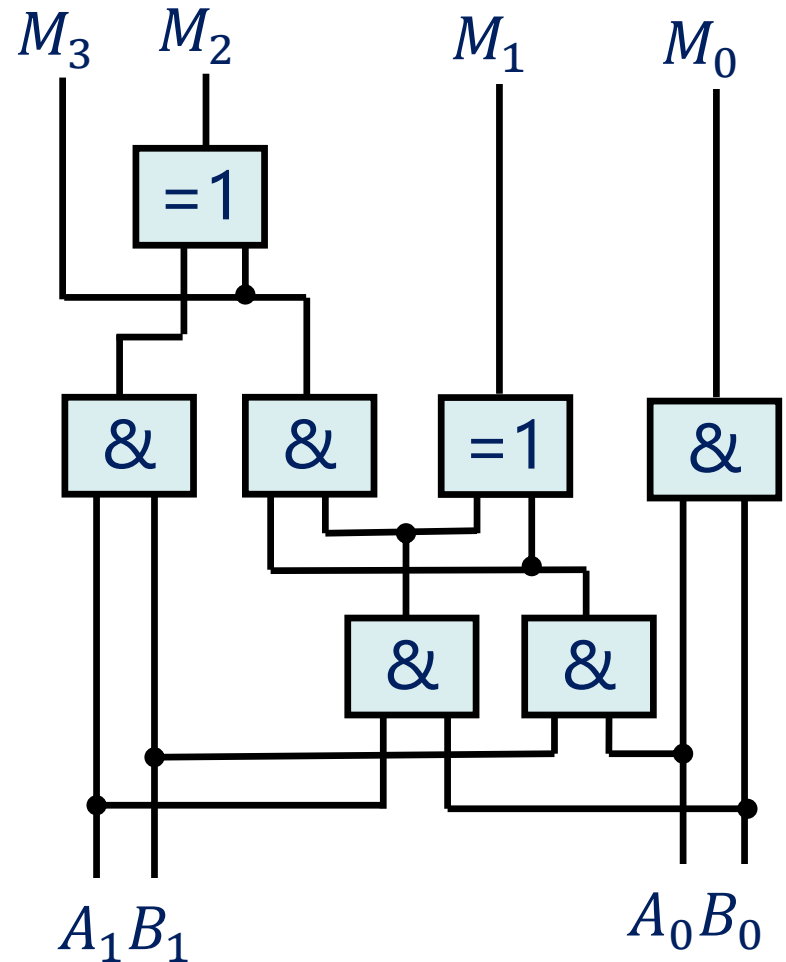
$$\square M_3 = C_2$$

$$- M_3 = A_0 B_1 A_1 B_0$$

多输出函数的组合逻辑电路设计

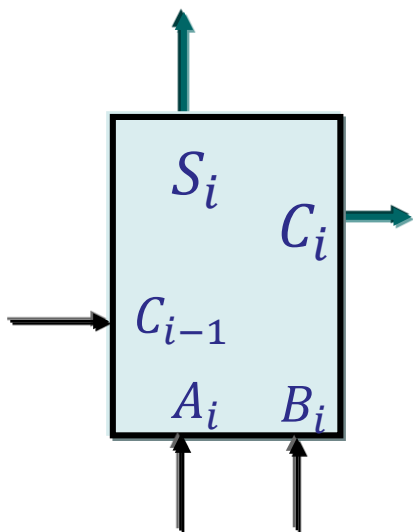
□ 电路图

- $M_0 = A_0 B_0$
- $M_1 = A_0 B_1 \oplus A_1 B_0$
- $M_2 = A_0 B_1 A_1 B_0 \oplus A_1 B_1$
- $M_3 = A_0 B_1 A_1 B_0$



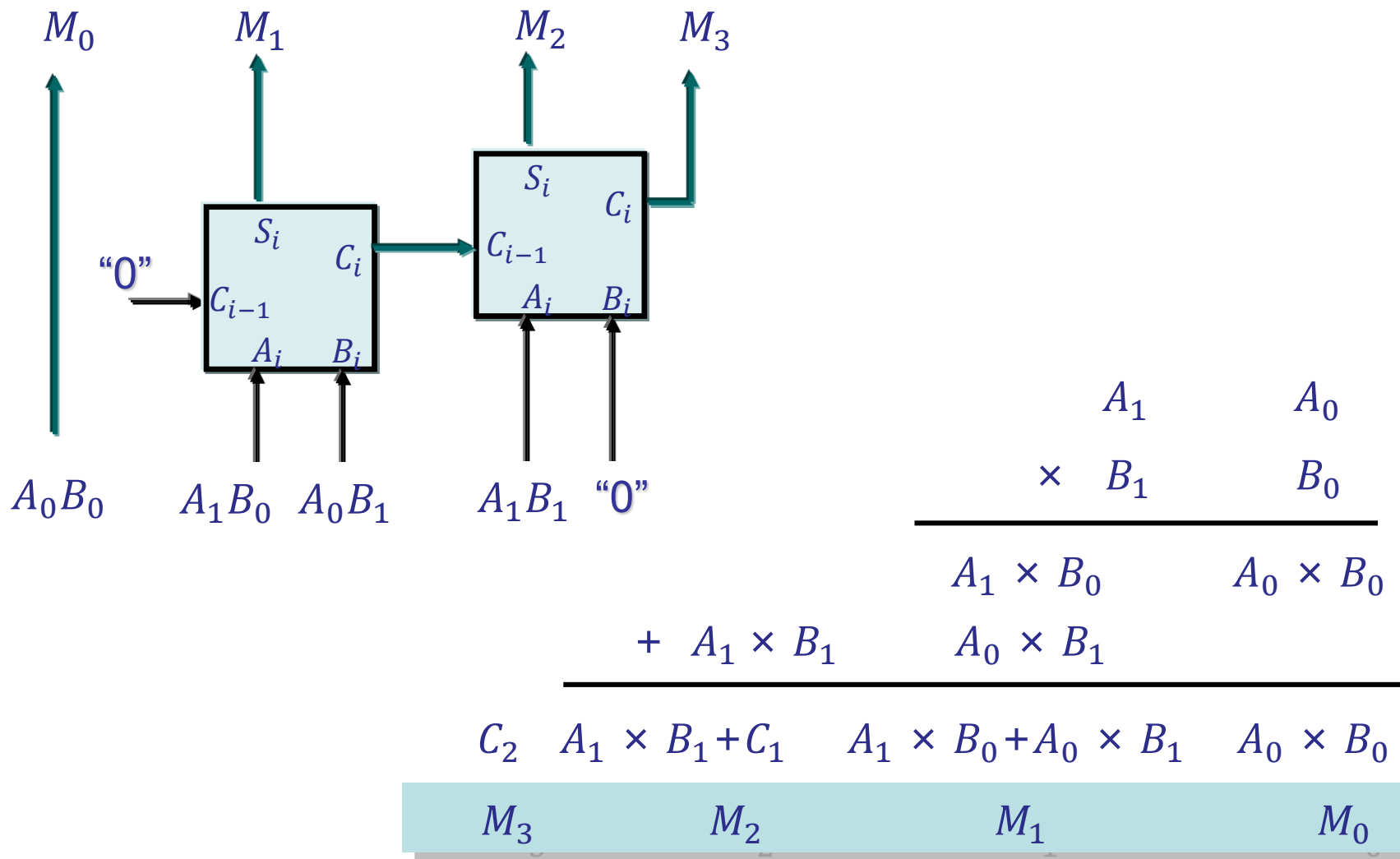
多输出函数的组合逻辑电路设计

□ 用全加器如何实现？



$$\begin{array}{r}
 \begin{array}{cc}
 & A_1 & A_0 \\
 & \times B_1 & B_0 \\
 \hline
 & A_1 \times B_0 & A_0 \times B_0 \\
 + & A_1 \times B_1 & A_0 \times B_1 \\
 \hline
 C_2 & A_1 \times B_1 + C_1 & A_1 \times B_0 + A_0 \times B_1 & A_0 \times B_0 \\
 \hline
 M_3 & M_2 & M_1 & M_0
 \end{array}
 \end{array}$$

多输出函数的组合逻辑电路设计



包含无关条件的组合逻辑电路设计

□ 包含无关条件的逻辑问题

- 逻辑函数与输入变量的某些取值组合无关
 - 输入变量之间存在的相互制约
 - 问题的某种特殊限定
 - ...

□ 包含无关条件的逻辑函数

- 描述包含无关条件的逻辑问题的逻辑函数

包含无关条件的组合逻辑电路设计

□无关最小项

- 和输出无关的输入取值组合对应的最小项
- 简称：
 - 无关项
 - 任意项

包含无关条件的组合逻辑电路设计

□例：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A、B、C三个变量不允许两个或两个以上同时为1。

□分析

- 只能取值：000, 001, 010, 100
- 不允许出现：011, 101, 110, 111
- 无关最小项 $\bar{A}BC$, $A\bar{B}C$, $AB\bar{C}$, ABC
- 包含无关条件的逻辑函数

包含无关条件的组合逻辑电路设计

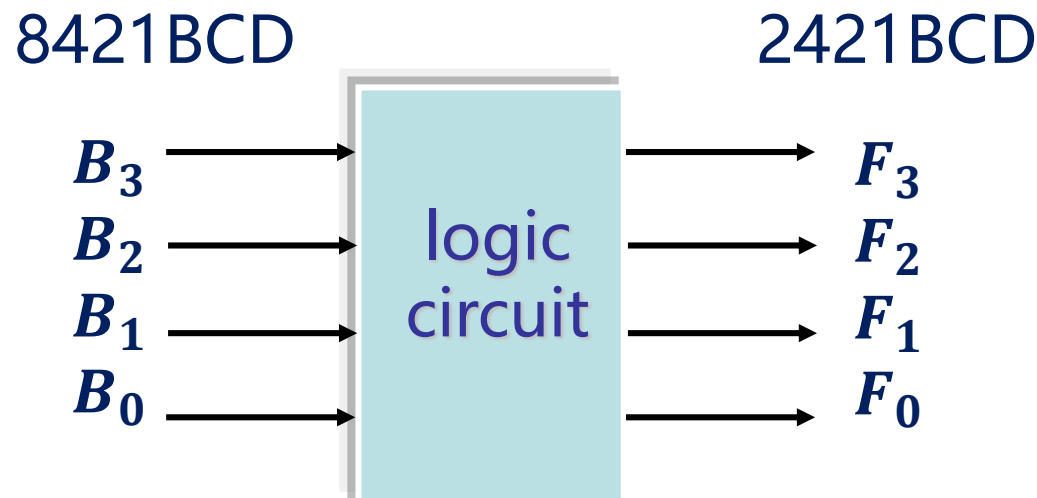
- 当采用“最小项之和”表达式描述一个包含无关条件的逻辑问题时，函数表达式中是否包含无关项，以及对无关项是令其值为1还是为0，并不影响函数的实际逻辑功能
 - 利用无关项的随意性往往可以使逻辑函数得到更好地简化，从而使设计的电路达到更简

包含无关条件的组合逻辑电路设计

□例5.代码转换器，将 8421 BCD 转换为 2421 BCD

□分析：

- 输入：4个
- 输出：4个



包含无关条件的组合逻辑电路设计

□ 真值表

$B_3B_2B_1B_0$	$F_3F_2F_1F_0$

包含无关条件的组合逻辑电路设计

□ 真值表

$B_3 B_2 B_1 B_0$	$F_3 F_2 F_1 F_0$
0 0 0 0	0 0 0 0
0 0 0 1	0 0 0 1
0 0 1 0	0 0 1 0
0 0 1 1	0 0 1 1
0 1 0 0	0 1 0 0
0 1 0 1	1 0 1 1
0 1 1 0	1 1 0 0
0 1 1 1	1 1 0 1
1 0 0 0	1 1 1 0
1 0 0 1	1 1 1 1

包含无关条件的组合逻辑电路设计

□ 真值表

- $F_3 = \sum m(5,6,7,8,9)$
- $F_2 = \sum m(4,6,7,8,9)$
- $F_1 = \sum m(2,3,5,8,9)$
- $F_0 = \sum m(1,3,5,7,9)$

$B_3B_2B_1B_0$	$F_3F_2F_1F_0$
0 0 0 0	0 0 0 0
0 0 0 1	0 0 0 1
0 0 1 0	0 0 1 0
0 0 1 1	0 0 1 1
0 1 0 0	0 1 0 0
0 1 0 1	1 0 1 1
0 1 1 0	1 1 0 0
0 1 1 1	1 1 0 1
1 0 0 0	1 1 1 0
1 0 0 1	1 1 1 1

包含无关条件的组合逻辑电路设计

$$\square F_3 = \sum m(5,6,7,8,9)$$

B_3B_2		F_3			
		00	01	11	10
B_1B_0	00				1
	01		1		1
	11		1		
	10		1		

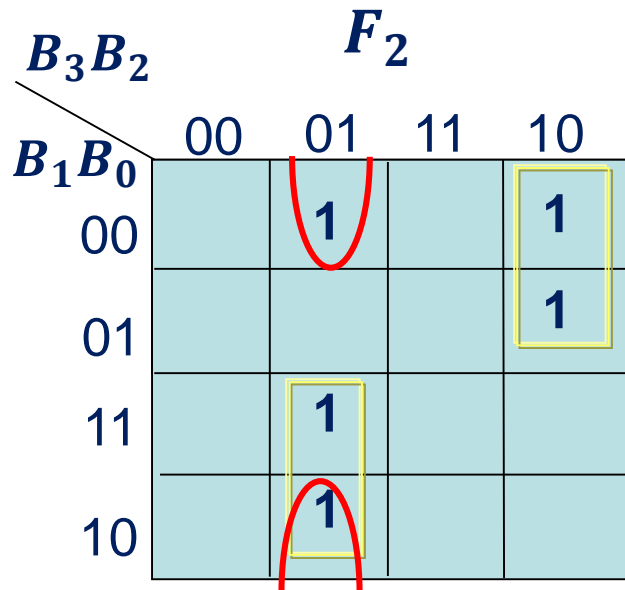
$$F_3 = B_3 \overline{B_2} \overline{B_1} + \overline{B_3} B_2 B_1 + \overline{B_3} B_2 B_0$$

B_3B_2		F_3			
		00	01	11	10
B_1B_0	00			d	1
	01		1	d	1
	11		1	d	d
	10		1	d	d

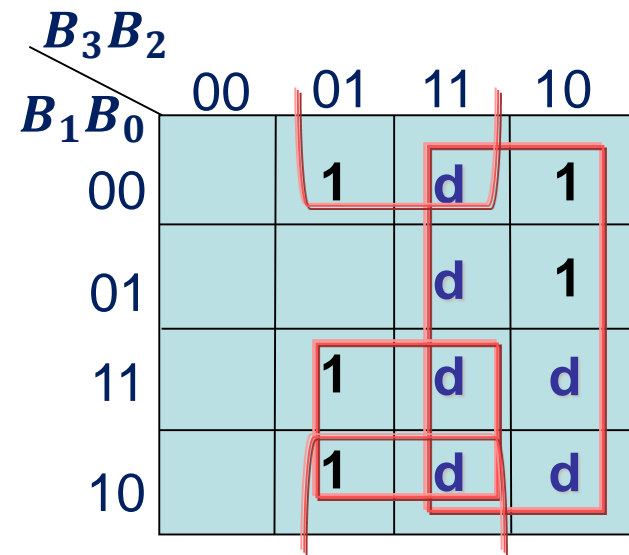
$$F_3 = B_3 + B_2 B_1 + B_2 B_0$$

包含无关条件的组合逻辑电路设计

$$\square F_2 = \sum m(4,6,7,8,9)$$



$$F_2 = B_3 \overline{B_2} \overline{B_1} + \overline{B_3} B_2 B_1 + \overline{B_3} B_2 \overline{B_0}$$



$$F_2 = B_3 + B_2 B_1 + B_2 \overline{B_0}$$

包含无关条件的组合逻辑电路设计

$$\square F_1 = \sum m(2,3,5,8,9)$$

B_3B_2		F_1			
		00	01	11	10
B_1B_0	00				1
	01		1		1
	11	1			
	10	1			

B_3B_2					
		00	01	11	10
B_1B_0	00			d	1
	01		1	d	1
	11	1		d	d
	10	1		d	d

$$F_1 = B_3 \overline{B_2} \overline{B_1} + \overline{B_3} \overline{B_2} B_1 + \overline{B_3} B_2 \overline{B_1} B_0$$

$$F_1 = B_3 + \overline{B_2} B_1 + B_2 \overline{B_1} B_0$$

包含无关条件的组合逻辑电路设计

$$\square F_0 = \sum m(1,3,5,7,9)$$

F_0

B_3B_2				
B_1B_0	00	01	11	10
00				
01	1	1		1
11	1	1		
10				

$$F_0 = \overline{B_3}B_1 + \overline{B_2}\overline{B_1}B_0$$

B_3B_2				
B_1B_0	00	01	11	10
00			d	
01	1	1	d	1
11	1	1	d	d
10			d	d

$$F_0 = B_0$$

包含无关条件的组合逻辑电路设计

□例7 设计一个组合逻辑电路，用于判别以余3码表示的1位十进制数是否为合数。

□分析：

- 输入变量：ABCD
- 输出函数：F
- 逻辑关系：
 - 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。
- 无关项：0000、0001、0010、1101、1110、1111

包含无关条件的组合逻辑电路设计

□ 真值表

<i>ABCD</i>	<i>F</i>	<i>ABCD</i>	<i>F</i>
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

$$F(A,B,C,D) = \sum m(7,9,11,12) + \sum d(0,1,2,13,14,15)$$

包含无关条件的组合逻辑电路设计

□ 表达式化简

– 若不考虑无关项

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

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

包含无关条件的组合逻辑电路设计

□ 表达式化简

– 若考虑无关项

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

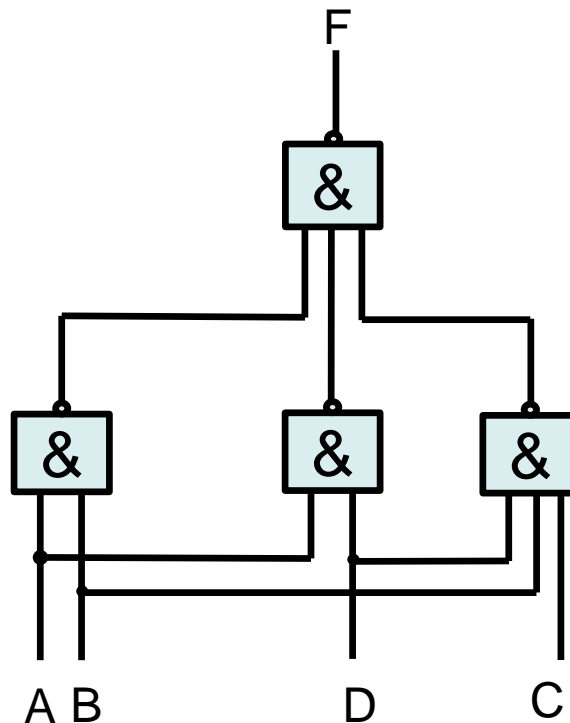
$$F(A, B, C, D) = AD + AB + BCD$$

包含无关条件的组合逻辑电路设计

□ 器件选择：与非门

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

□ 逻辑电路图



无反变量提供的组合逻辑电路设计

- 在某些问题的设计中，不提供反变量
 - 为了减少各部件之间的连线
- 若直接用非门将原变量转换成相应的反变量，则处理结果往往是不经济的
 - 通常进行适当的变换，以便尽可能减少非门数量。

无反变量提供的组合逻辑电路设计

□例7.输入不提供反变量时，用与非门实现逻辑函数

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

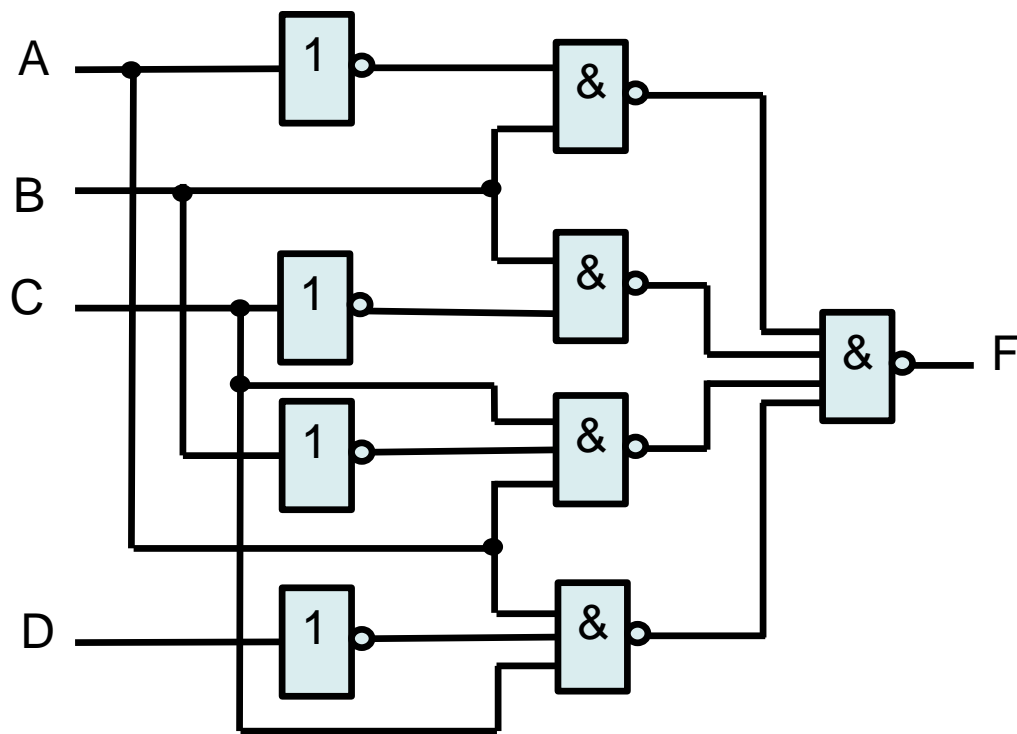
□分析

- F已经是 “与-或” 表达式，是否最简？
- 故可直接变换成 “与非-与非” 表达式

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

无反变量提供的组合逻辑电路设计

$$\square F(A, B, C, D) = \overline{\overline{A}B} \cdot \overline{B\overline{C}} \cdot \overline{A\overline{B}C} \cdot \overline{AC\overline{D}}$$



无反变量提供的组合逻辑电路设计

□ 对函数F的表达式作整理

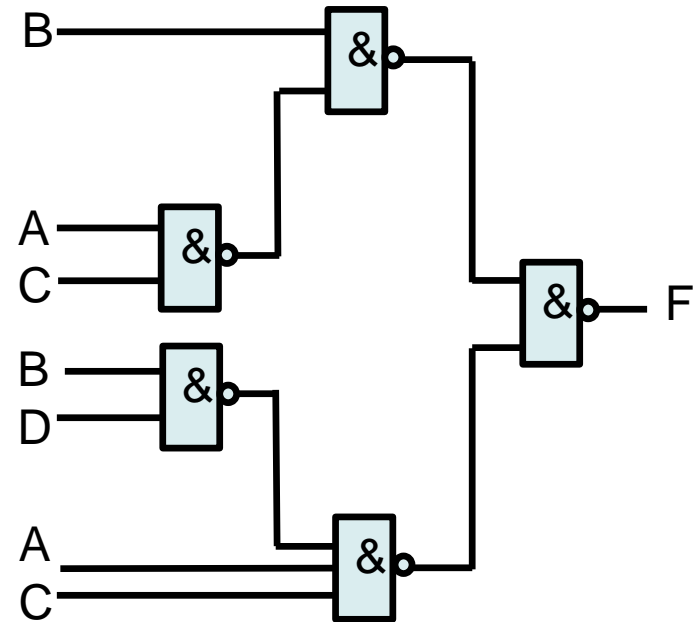
$$F(A, B, C, D)$$

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

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

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

$$= \overline{\overline{B\bar{A}\bar{C}}} \cdot \overline{\overline{AC\bar{B}\bar{D}}}$$



无反变量提供的组合逻辑电路设计

□例8.设计一个组合逻辑电路，用来判断献血者与受血者血型是否相容。

献 血	受 血			
	A	B	AB	O
A	√		√	
B		√	√	
AB			√	
O	√	√	√	√

无反变量提供的组合逻辑电路设计

□ 分析：

- 电路输入变量为献血者血型和受血者血型。
- 血型共4种
 - 可用两个变量的4种编码进行区分
 - WX表示献血者血型，YZ表示受血者血型

□ 血型编码

血型	WX (献血)	YZ (受血)
A	00	00
B	01	01
AB	10	10
O	11	11

无反变量提供的组合逻辑电路设计

□ 设电路输出用F表示

- 则当输血者与受血者血型相容时，F为1，否则F为0

献 血	受 血			
	A	B	AB	O
A	✓		✓	
B		✓	✓	
AB			✓	
O	✓	✓	✓	✓

血型	WX (献血)	YZ (受血)
A	00	00
B	01	01
AB	10	10
O	11	11

无反变量提供的组合逻辑电路设计

□ 设电路输出用F表示

- 则当输血者与受血者血型相容时，F为1，否则F为0

献 血	受 血			
	A	B	AB	O
A	1		1	
B		1	1	
AB			1	
O	1	1	1	1

血型	WX (献血)	YZ (受血)
A	00	00
B	01	01
AB	10	10
O	11	11

无反变量提供的组合逻辑电路设计

□ 设电路输出用F表示

- 则当输血者与受血者血型相容时，F为1，否则F为0

献 血	受 血			
	00	01	10	11
00	1		1	
01		1	1	
10			1	
11	1	1	1	1

血型	WX (献血)	YZ (受血)
A	00	00
B	01	01
AB	10	10
O	11	11

无反变量提供的组合逻辑电路设计

□ 设电路输出用F表示

- 则当输血者与受血者血型相容时，F为1，否则F为0

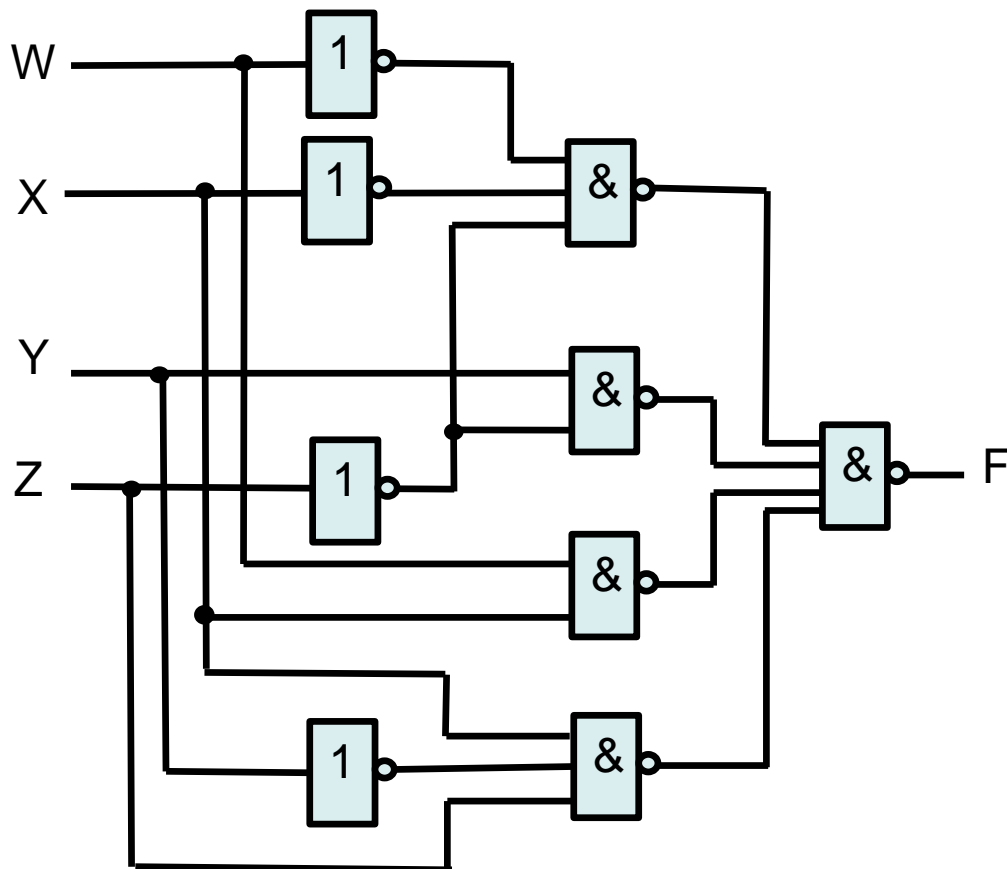
献 血	受 血			
	00	01	10	11
00	1		1	
01		1	1	
10			1	
11	1	1	1	1

		YZ			
		00	01	11	10
WX	00	1			1
	01		1		1
	11	1	1	1	1
	10				1

$$\begin{aligned}F(W, X, Y, Z) &= WX + \bar{W}\bar{X}\bar{Z} + X\bar{Y}Z + Y\bar{Z} \\ &= \overline{\bar{W}\bar{X}} \cdot \overline{\bar{W}\bar{X}\bar{Z}} \cdot \overline{X\bar{Y}Z} \cdot \overline{Y\bar{Z}}\end{aligned}$$

无反变量提供的组合逻辑电路设计

$$\square F(W, X, Y, Z) = \overline{\overline{W}X} \cdot \overline{\overline{W}\overline{X}\overline{Z}} \cdot \overline{X\overline{Y}Z} \cdot \overline{Y\overline{Z}}$$



无反变量提供的组合逻辑电路设计

□分析：

- 对该问题的逻辑描述与血型编码是直接相关的

献 血	受 血			
	A	B	AB	O
A	1		1	
B		1	1	
AB			1	
O	1	1	1	1

血型	WX (献血)	YZ (受血)
A	00	00
AB	01	01
O	10	10
B	11	11

无反变量提供的组合逻辑电路设计

□ A:00

B:11

AB:01

O:10

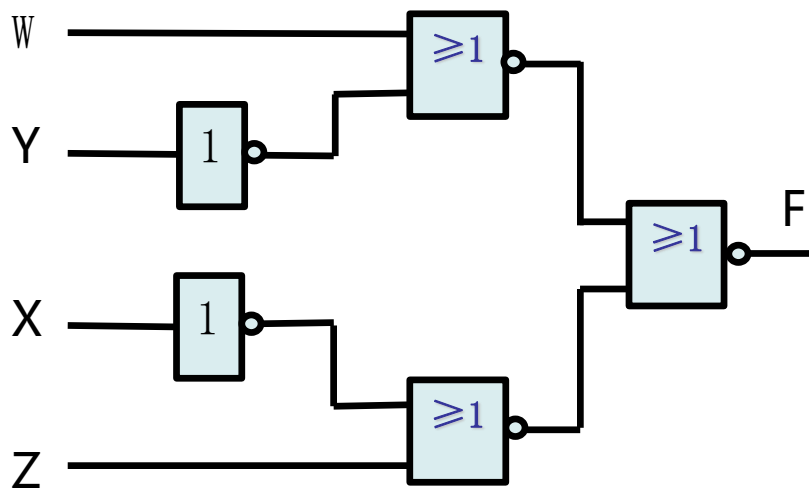
献 血	受 血			
	00	11	01	10
00	1		1	
11		1	1	
01			1	
10	1	1	1	1

		YZ			
		00	01	11	10
WX	00	1	1		
	01		1		
	11		1	1	
	10	1	1	1	1

$$\begin{aligned}
 F(W, X, Y, Z) &= (W + \bar{Y})(\bar{X} + Z) \\
 &= \overline{W + \bar{Y}} \cdot \overline{\bar{X} + Z}
 \end{aligned}$$

无反变量提供的组合逻辑电路设计

$$\square F(W, X, Y, Z) = \overline{W + \bar{Y} \cdot \bar{X} + Z}$$



提 纲

1

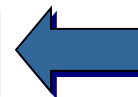
组合逻辑电路分析

2

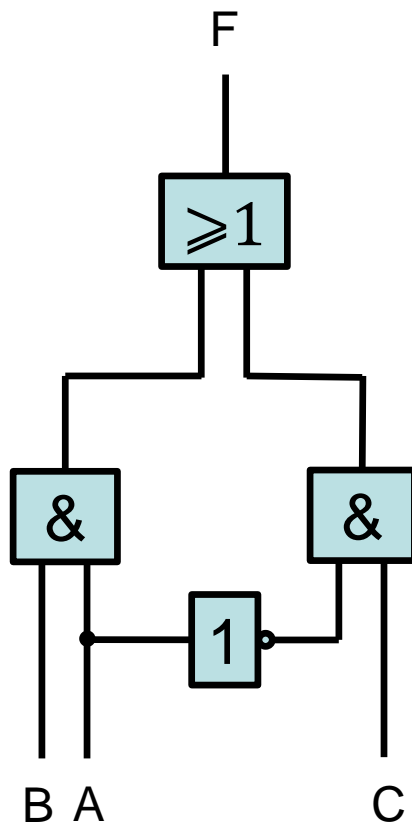
组合逻辑电路设计

3

组合逻辑电路的险象



组合逻辑电路中的险象



组合逻辑电路中的险象

□ 信号经过任何逻辑门和导线都会产生时间延迟

- 当电路所有输入达到稳定状态时，输出并不是立即达到稳定状态

□ 延迟时间对数字系统是一个有害的因素

- 使得系统操作速度下降
- 引起电路中信号的波形参数变坏
- 以及产生竞争险象等问题

竞争现象与险象的产生

□ 延迟时间

- 与信号经过的门的级数有关
- 与具体逻辑门的时延大小有关
- 与导线的长短有关

□ 竞争

- 由于延迟时间的影响，输入信号经过不同路径到达输出端的时间有先有后的现象
- 广义理解：多个信号到达某一点有时差的现象

组合逻辑电路中的险象

□ 竞争的类型

- 非临界竞争
 - 不产生错误输出的竞争
- 临界竞争
 - 导致错误输出的竞争

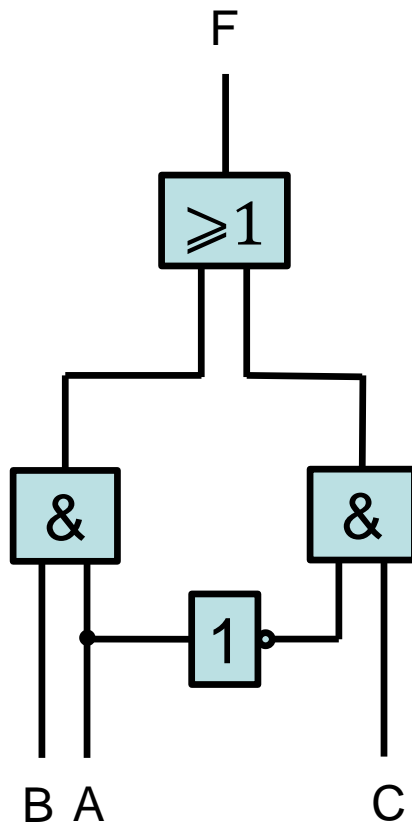
组合逻辑电路中的险象

□ 险象：由竞争导致的错误输出信号

- 组合电路中的险象是一种瞬态现象
- 它表现为在输出端产生不应有的尖脉冲，暂时地破坏正常逻辑关系
- 一旦瞬态过程结束，即可恢复正常逻辑关系

组合逻辑电路的险象

□例：分析下图所示电路是否会产生竞争和险象。



组合逻辑电路的险象

□分析：

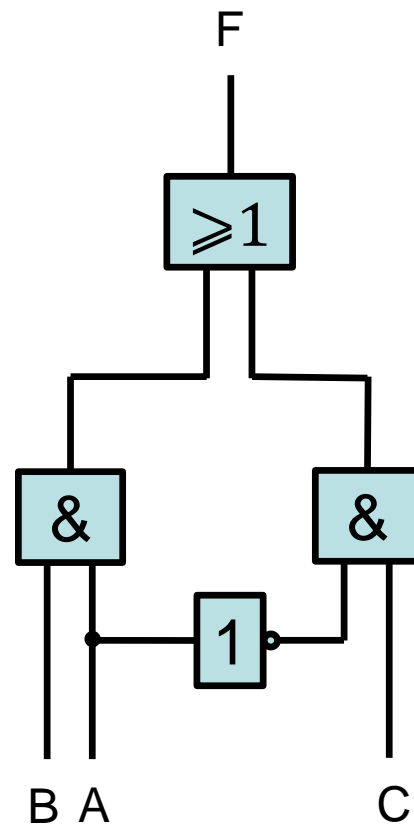
– $F = AB + \bar{A}C$

- 假设输入变量 $B = C = 1$

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

– 当考虑时间延迟时

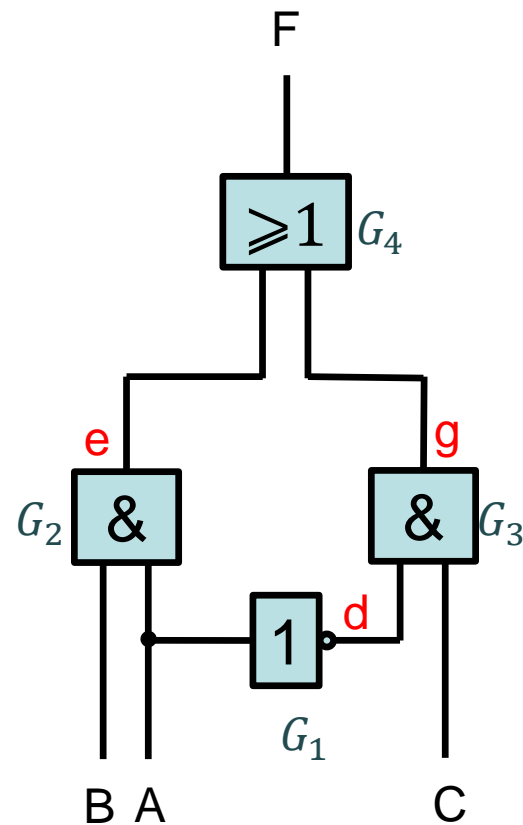
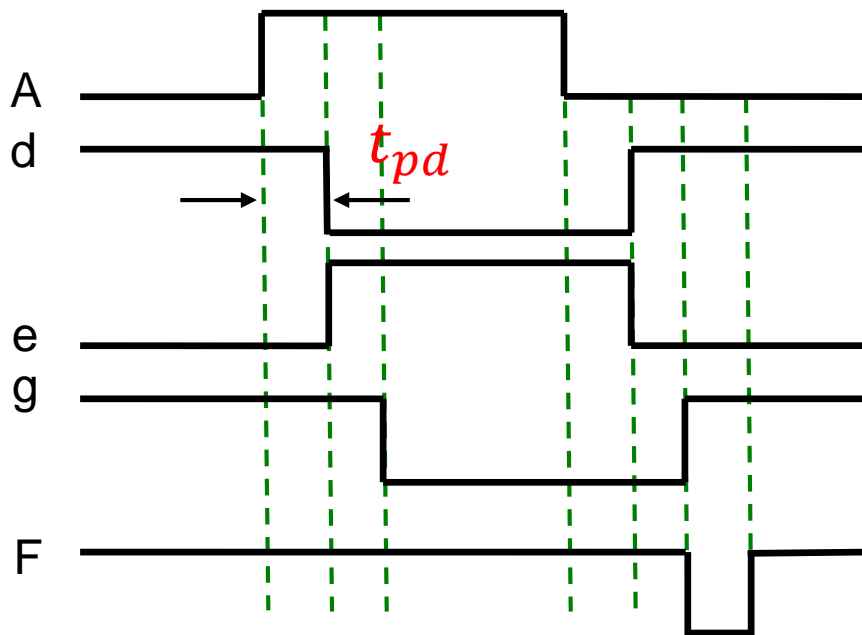
- 实际输入、输出关系又将怎样呢？



组合逻辑电路的险象

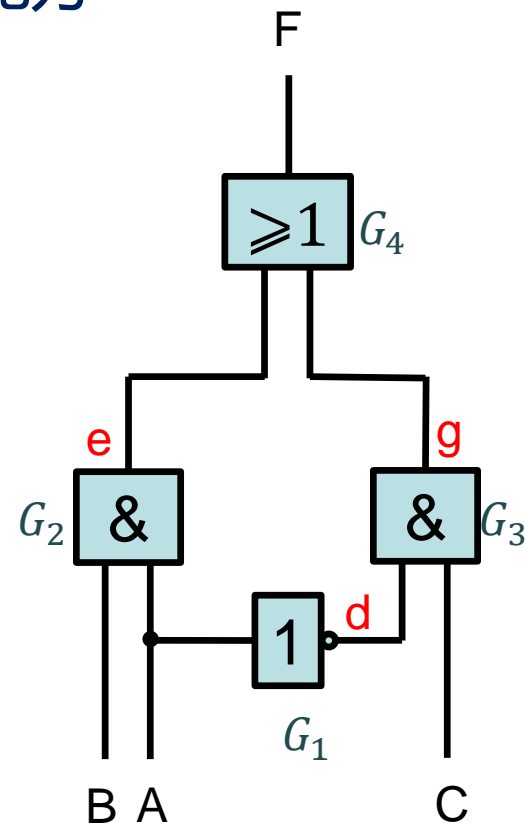
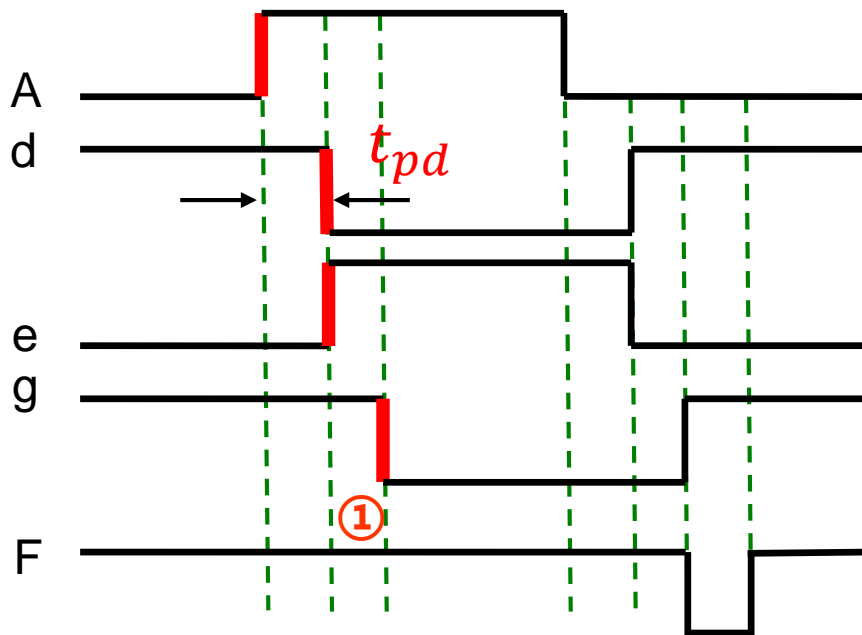
□ 分析

- 假定每个门的延迟时间为 t_{pd}



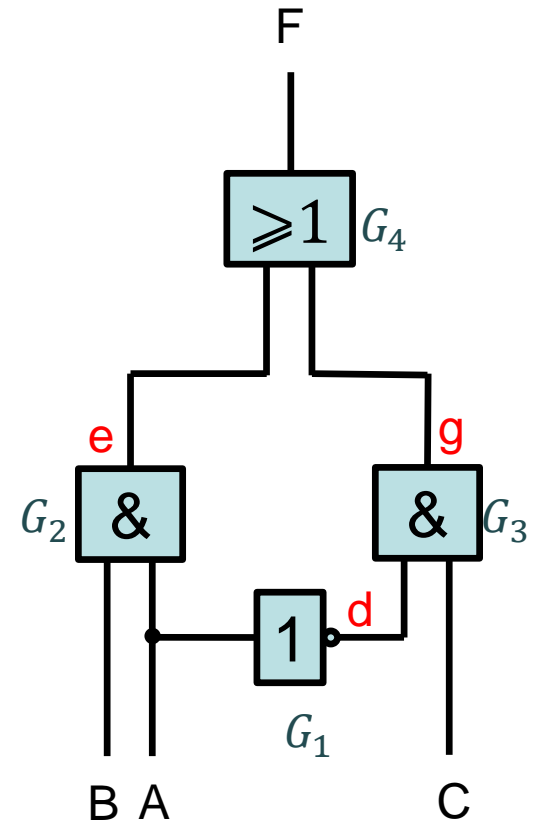
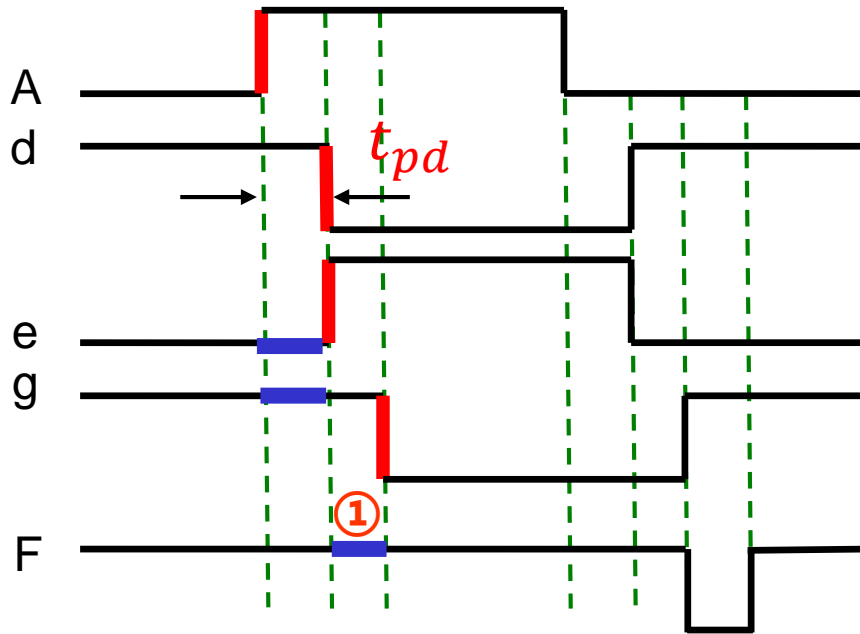
组合逻辑电路的险象

- 当A由低电平变到高电平时，e和g的变化方向相反，并具有一个 t_{pd} 的时差
- 图中①处存在一次竞争



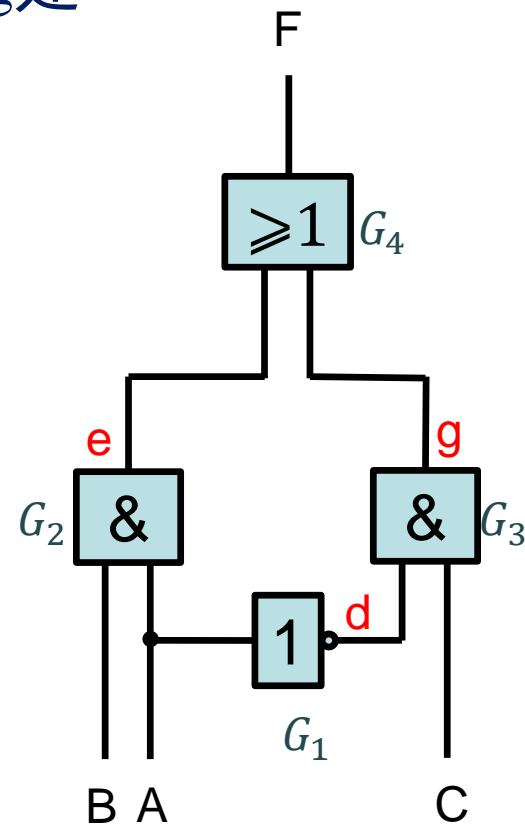
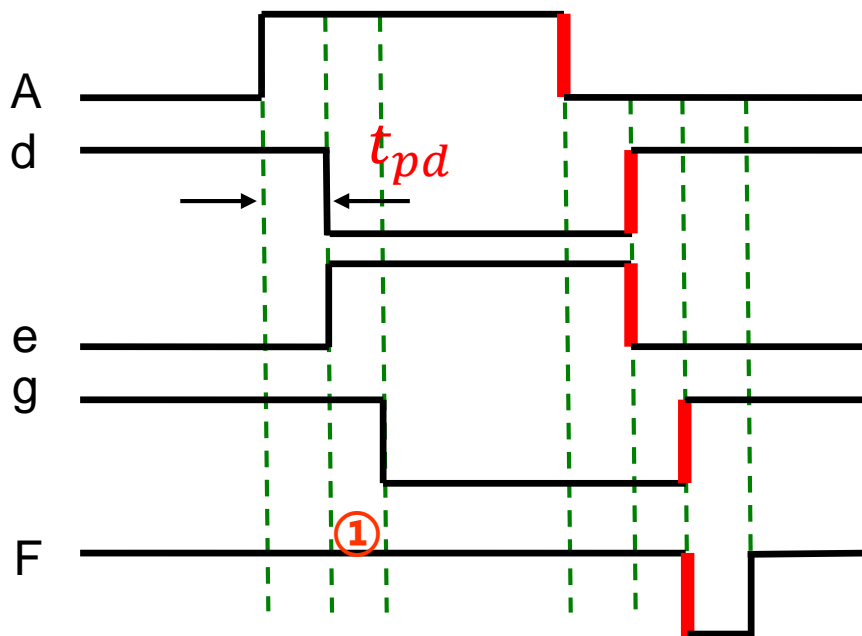
组合逻辑电路的险象

- e和g在一个 t_{pd} 的时间一个为高一个为低
- 输出保持为高电平
- 这次竞争是一次非临界竞争



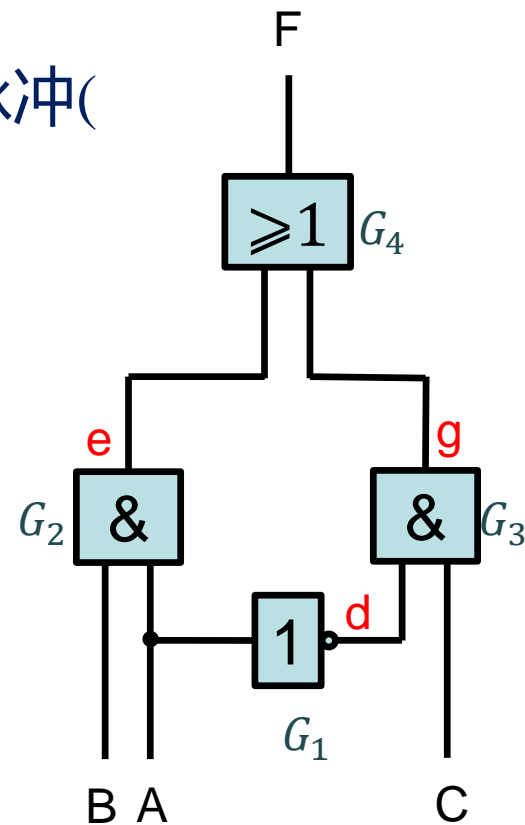
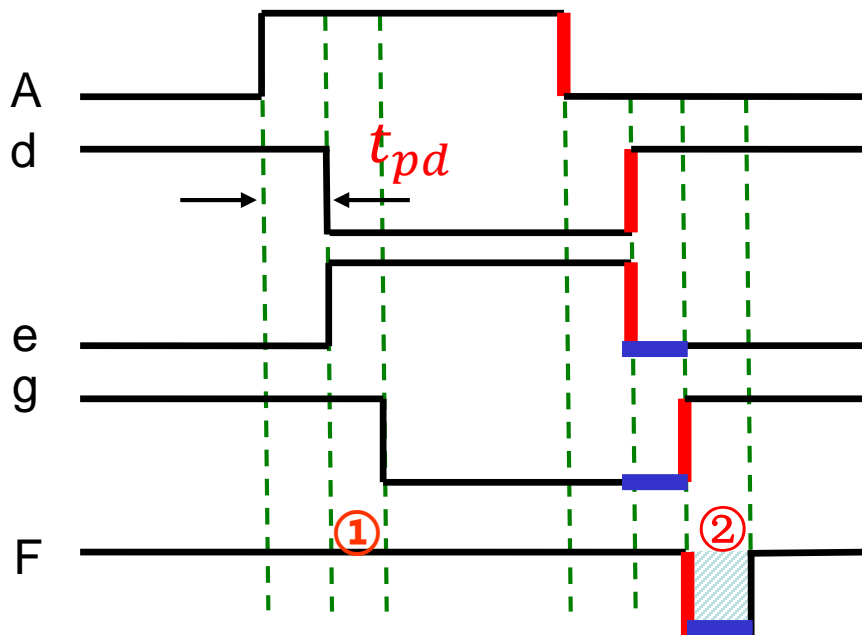
组合逻辑电路的险象

- 当A由高电平变为低电平时，同样在e和g处发生竞争



组合逻辑电路的险象

- e和g在一个 t_{pd} 的时间内同时为低电平
- 根据与逻辑特性，使输出F产生一个负脉冲(如图中②处所示)



组合逻辑电路的险象

□ 险象的分类

— 静态险象

- 在输入变化而输出不应发生变化的情况下，输出端产生的短暂的错误输出

— 动态险象

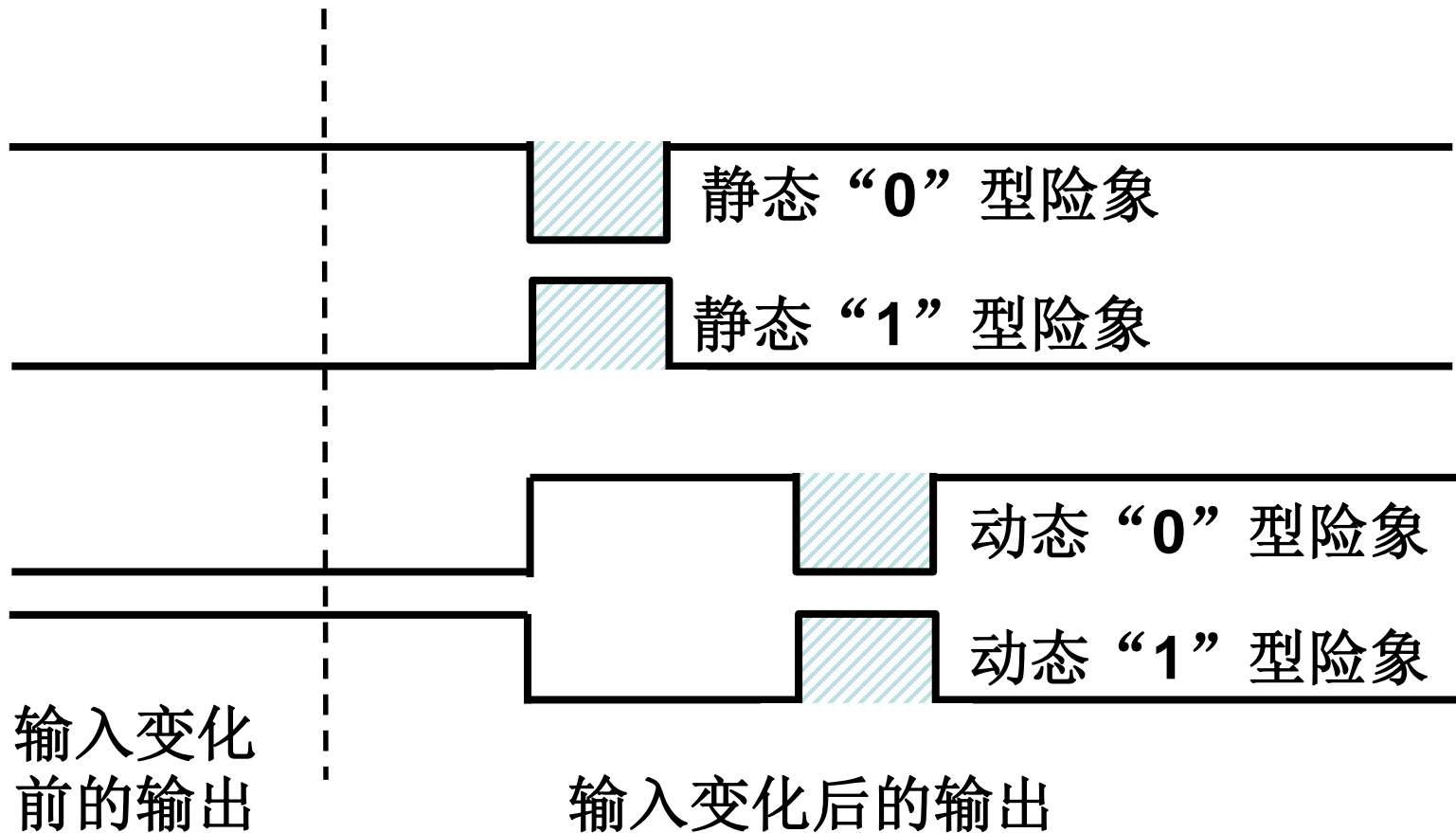
- 在输入变化而输出应该发生变化的情况下，输出在变化过程中产生的短暂的错误输出

险象的分类

□ 按错误输出脉冲信号的极性划分

- “0” 型险象
 - 错误输出信号为负脉冲。
- “1” 型险象
 - 错误输出信号为正脉冲。

险象的分类



组合逻辑电路的险象

□ 险象的判断

- 代数法
- 卡诺图法

险象的判断

□代数法

- 检查函数表达式中是否存在具备竞争条件的变量
 - 某个变量X同时以原变量和反变量的形式出现在函数表达式中
- 若存在具备竞争条件的变量X, 则消去函数式中的其他变量,看函数表达式是否会变为

$$X + \bar{X} \quad \text{或} \quad X \cdot \bar{X}$$

- 若会, 则说明对应的逻辑电路可能产生险象。

代数法

□例1 已知描述某组合电路的逻辑函数表达式为

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

试判断该逻辑电路是否可能产生险象。

□分析：

— 具备竞争条件的变量

- 变量A
- 变量C

代数法

□变量A

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

- $BC=00 \quad F = \bar{A}$
- $BC=01 \quad F = A$
- $BC=10 \quad F = \bar{A}$
- 当 $B=C=1$ 时, A的变化可能使电路产生险象

□变量C

- 同理, 可判断变量C发生变化时不会产生险象

代数法

□例2 试判断下面函数表达式描述的逻辑电路中是否可能产生险象。

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

□分析：

– 具备竞争条件的变量：变量A和B

代数法

□变量B

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

– $AC=00 \quad F = B\bar{B}$

– $AC=01 \quad F=B$

– $AC=10 \quad F=0$

– $AC=11 \quad F=1$

– 当 $A=C=0$ 时，B的变化可能使电路输出产生险象。

□变量A

– 同理，当 $B=C=0$ 时，A的变化也可能产生险象

险象的判断

□ 卡诺图法

- 适用范围：逻辑函数为“与-或”表达式时
- 作出函数卡诺图
- 并画出和函数表达式中各“与”项对应的卡诺圈
- 结论：
 - 若卡诺圈之间存在“相切”关系，即两卡诺圈之间存在不被同一卡诺圈包含的相邻最小项，则该电路可能产生险象

卡诺图法

□例 判断函数表达式 $F = \bar{A}D + \bar{A}C + ABC\bar{C}$ 对应的逻辑电路是否可能产生险象。

□分析：

- 做出卡诺图
- 画出卡诺圈
- 卡诺圈 $\bar{A}D$ 和 $ABC\bar{C}$ 相切
 - m_5 和 m_{13} 相邻，且不被同一卡诺圈包围
- 该电路存在险象
 - 当 $BCD = 101$ 时， A 的变换可能会产生险象

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

组合逻辑电路的险象

□ 险象的消除

- 增加冗余项
- 增加惯性延时环节
- 选通法

用增加冗余项的方法消除险象

□ 增加冗余项的方法

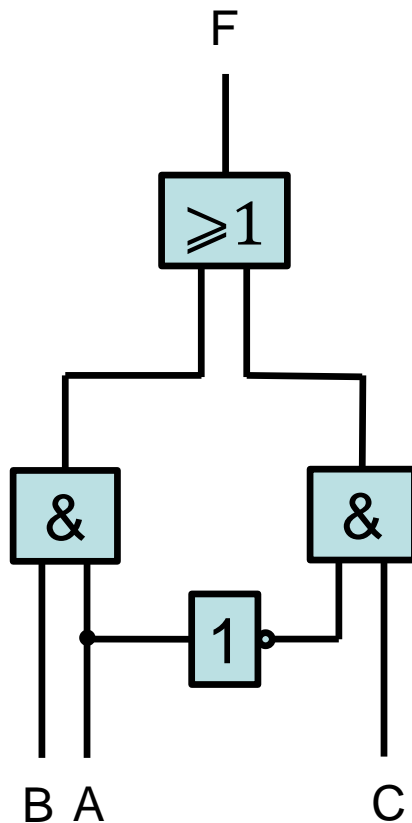
- “或” 上冗余的 “与” 项
- “与” 上冗余的 “或” 项

□ 冗余项的选择

- 代数法
- 卡诺图

用增加冗余项的方法消除险象

□ 例：消除下面电路中的险象。



用增加冗余项的方法消除险象

□分析：

– 逻辑表达式

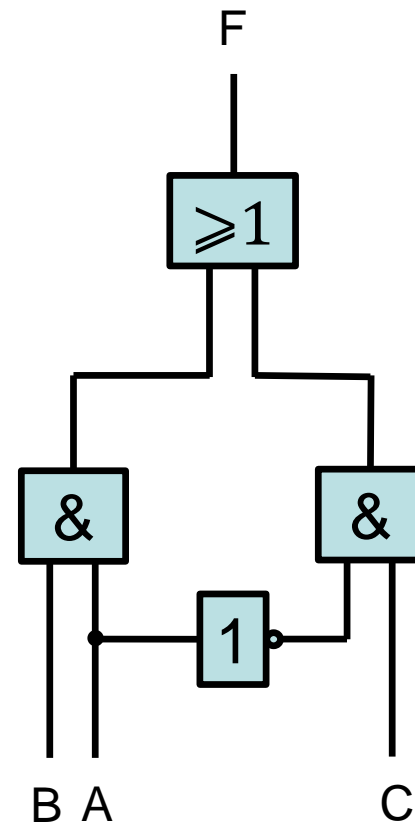
- $F = AB + \bar{A}C$

– 判断

- $B = C = 1$ 时, A 的变化可能使输出产生“0”型险象

– 增加冗余项 BC

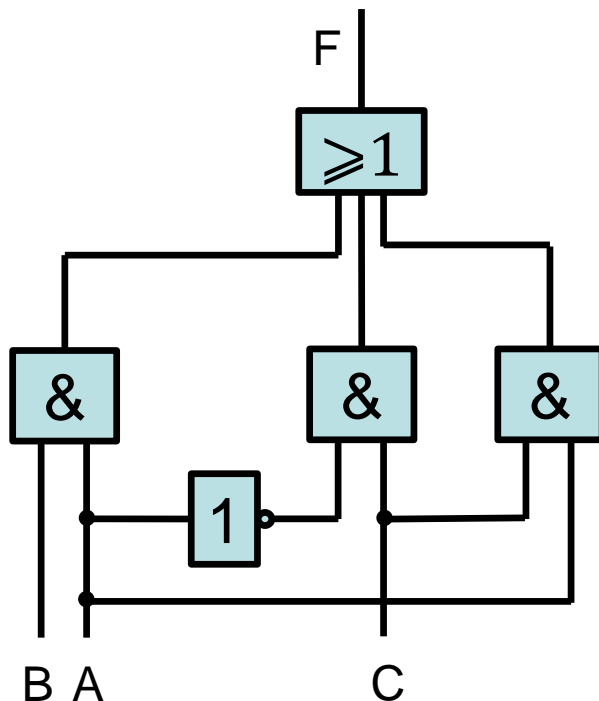
- $F = AB + \bar{A}C + BC$
- $BC = 11$ 时, $F = \bar{A} + A + 1 = 1$,
该电路不再产生险象



用增加冗余项的方法消除险象

□分析：

$$- F = AB + \bar{A}C + BC$$



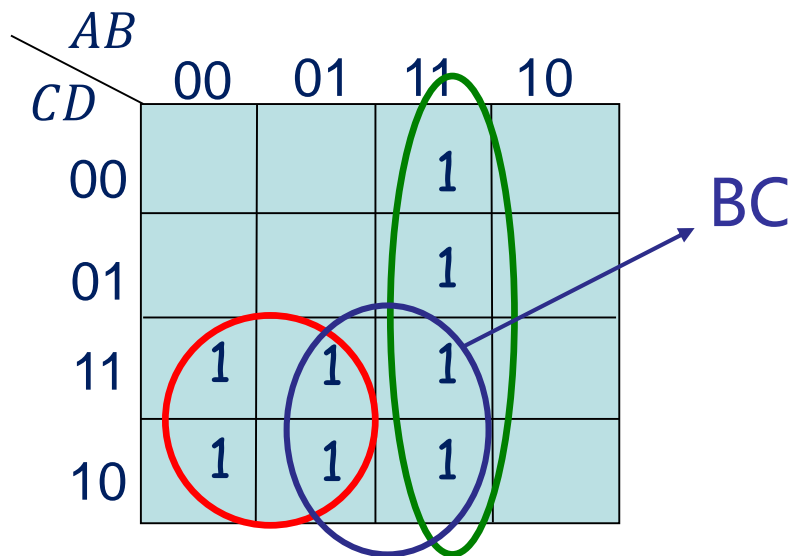
用增加冗余项的方法消除险象

□ 卡诺图法

- 在函数卡诺图上增加多余的卡诺圈
 - 若卡诺图上某两个卡诺圈“相切”，则用一个多余的卡诺圈将它们之间的相邻最小项圈起来，与多余卡诺圈对应的“与”项即为要加入函数表达式中的冗余项

用增加冗余项的方法消除险象

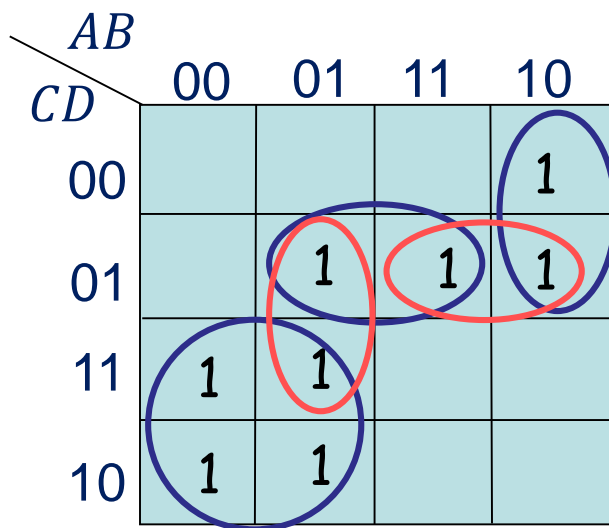
□例1: $F = AB + \bar{A}C$



$$F = AB + \bar{A}C + BC$$

用增加冗余项的方法消除险象

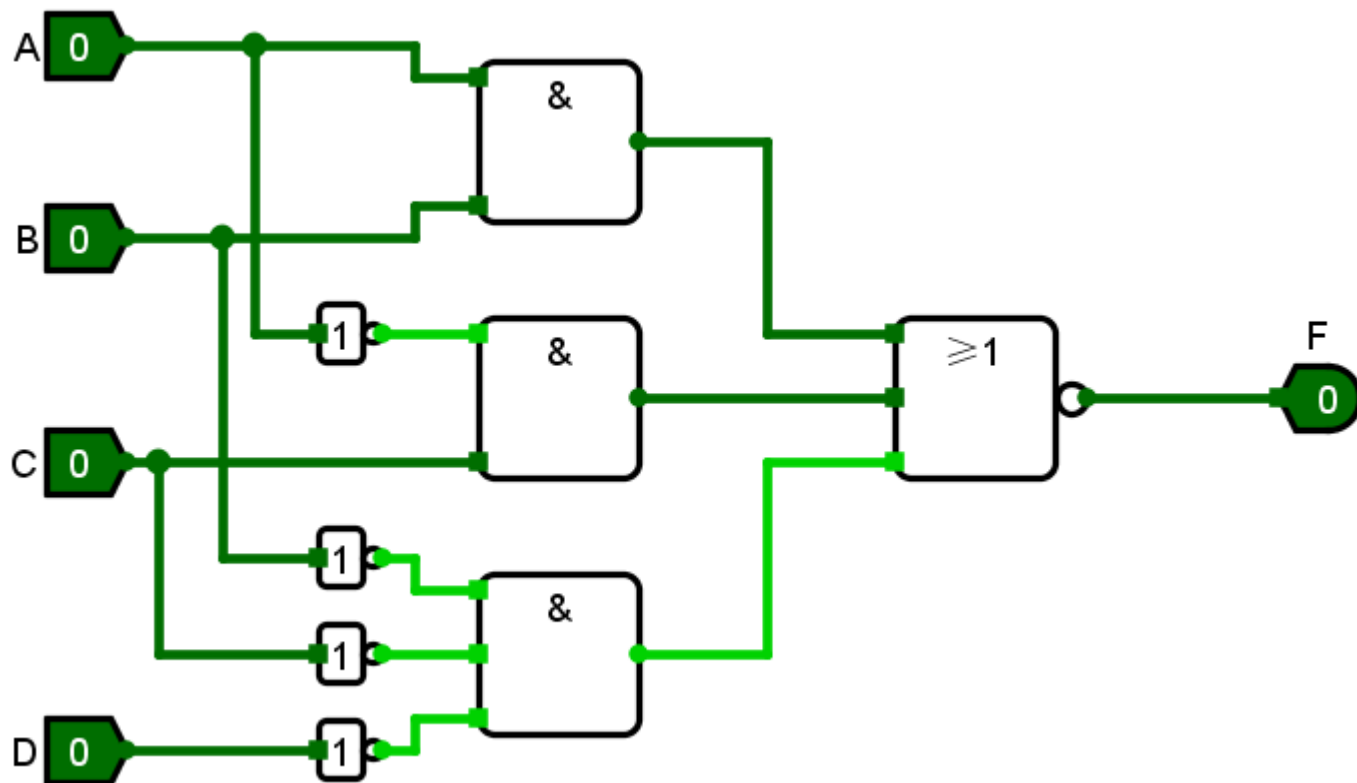
□例2：电路 $F = \bar{A}C + B\bar{C}D + A\bar{B}\bar{C}$ 可能产生险象，
试用增加冗余项的方法消除。



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

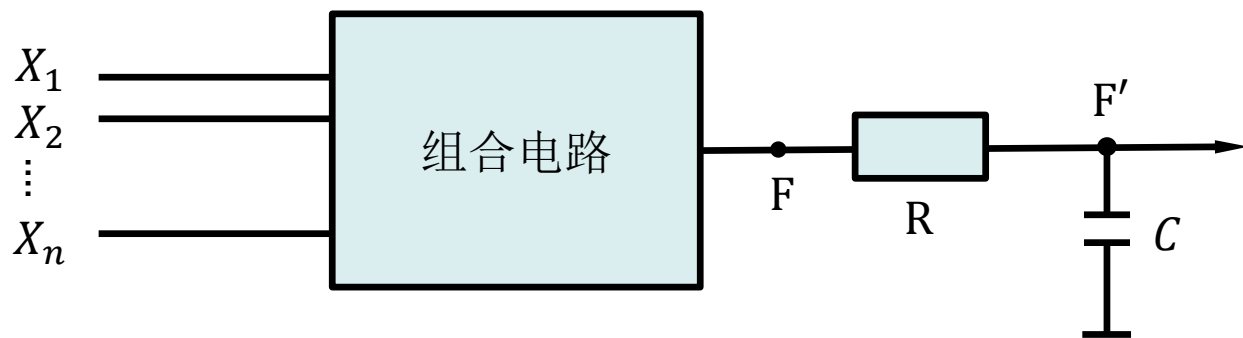
用增加冗余项的方法消除险象

□ 例3：判断下图所示电路竞争的产生情况（卡诺图法），并用增加冗余项的方法消除。



增加惯性延时环节

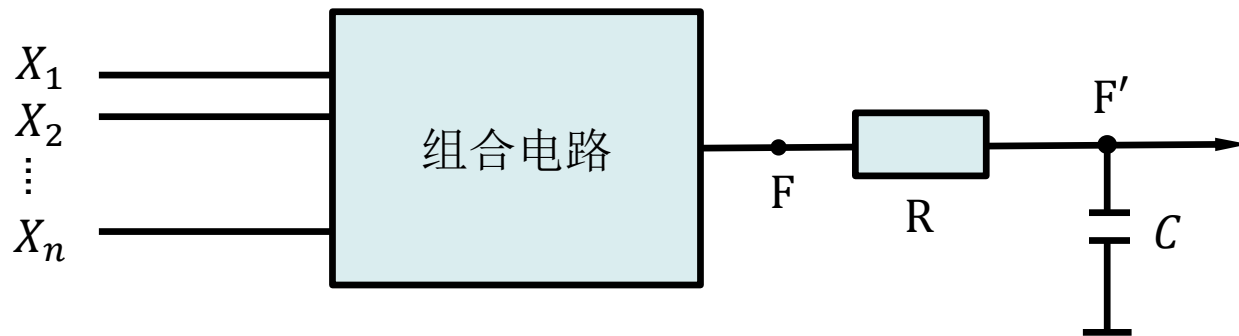
- 消除险象的另一种方法是在组合电路输出端连接一个惯性延时环节
- 通常采用RC电路作惯性延时环节



增加惯性延时环节

□ RC电路实际上是一个低通滤波器

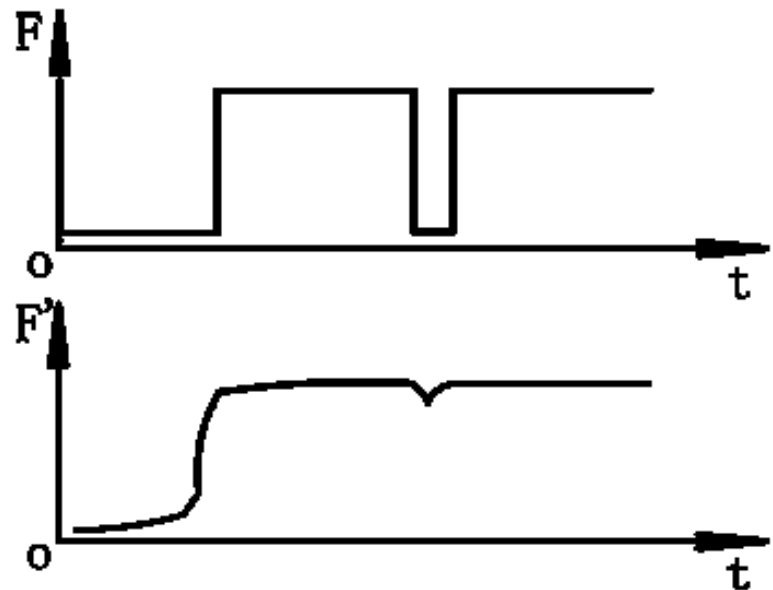
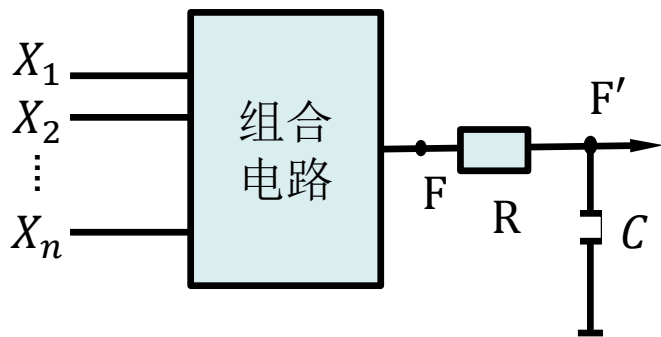
- 竞争引起的险象都是一些频率很高的尖脉冲信号
- 险象在通过RC电路后能基本被滤掉
- 保留下来的仅仅是一些幅度极小的毛刺，它们不再对电路的可靠性产生影响



增加惯性延时环节

□ 惯性环节的时间常数 $\tau = RC$

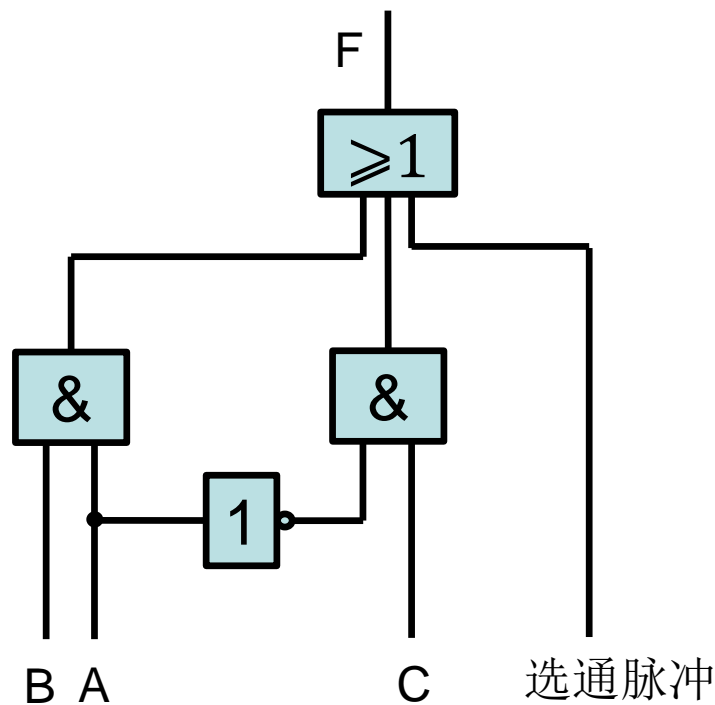
- 一般要求 τ 大于尖脉冲的宽度，以便能将尖脉冲“削平”
- 但 τ 太大将使正常输出信号产生不允许的畸变



选通法

□ 选通法不必增加任何器件，仅仅是利用选通脉冲的作用，从时间上加以控制，使输出避开险象脉冲。

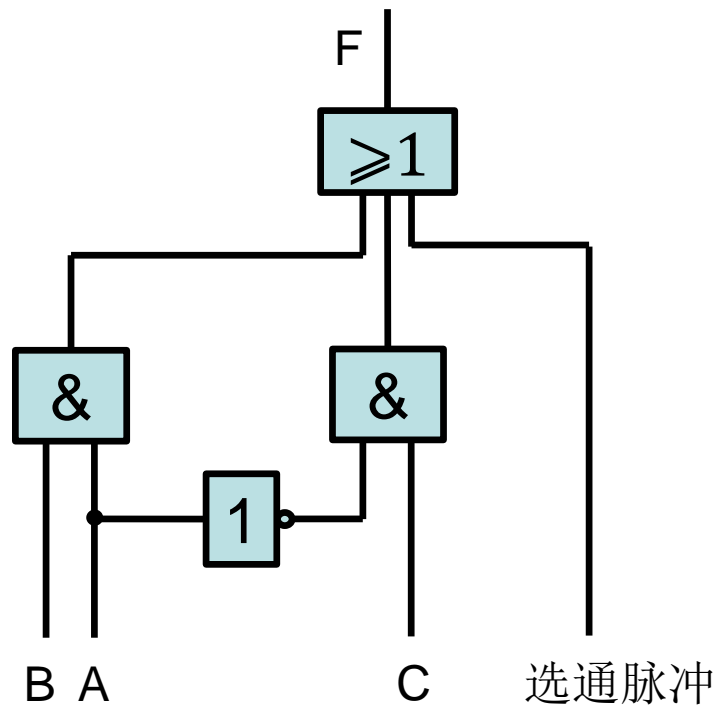
□ 例：



选通法

□分析：

- 该电路当A发生变化时，可能产生“0”型险象
- 通过选通脉冲对电路的输出门加以控制，令选通脉冲在电路稳定后出现，则可使输出避开险象脉冲，送出稳定输出信号。



问题解答



Thank You!

