
第3章 组合逻辑电路

第一讲 组合逻辑电路概述

第二讲 典型组合逻辑部件设计

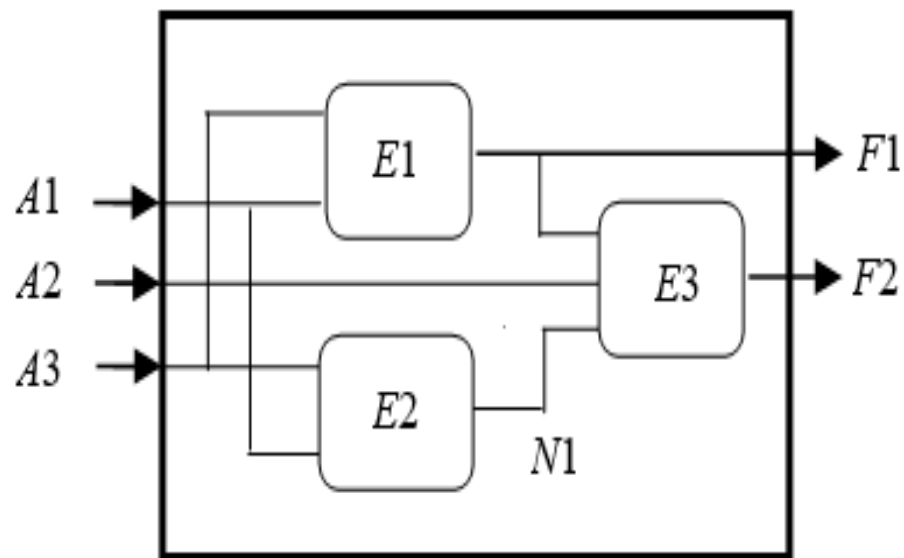
第三讲 组合逻辑电路时序分析

第一讲 组合逻辑电路概述

- ◆ 组合逻辑电路构成规则
- ◆ 逻辑电路图
- ◆ 两级与多级组合逻辑电路
- ◆ 组合逻辑电路设计
- ◆ 无关项、非法值和高阻态

1 组合逻辑电路概述

- ◆ **数字逻辑电路**可被看成是带有若干输入端和若干输出端的黑盒子，每个输入端和输出端只有高电平、低电平两种状态，对应1或0。
- ◆ 分为**组合 (combinational) 逻辑电路**和**时序 (sequential) 逻辑电路**两种类型。
 - **组合逻辑电路的输出值仅依赖于当前输入值 (本章)**
 - 时序逻辑电路的输出值不仅依赖输入值，还与当前状态 (现态) 有关。电路中存在存储部件或反馈结构 (**第4章**)



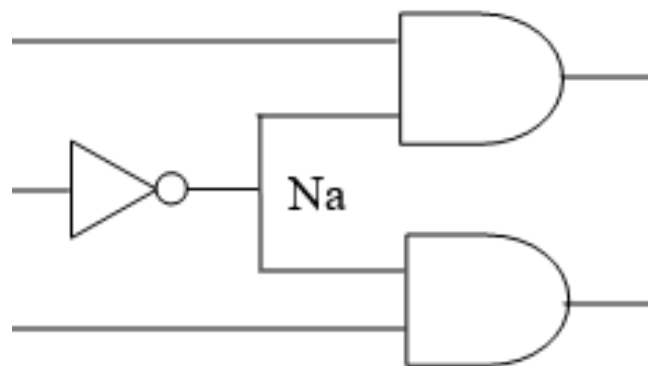
- ◆ 黑盒内部可被看成由若干**元件**和若干**结点**互连而成。
 - **元件本身又可以是一个数字逻辑电路**
 - **结点可以是输入结点(如 A1)、内部结点(如 N1)和输出结点(如 F1)**

1.1 组合逻辑电路构成规则

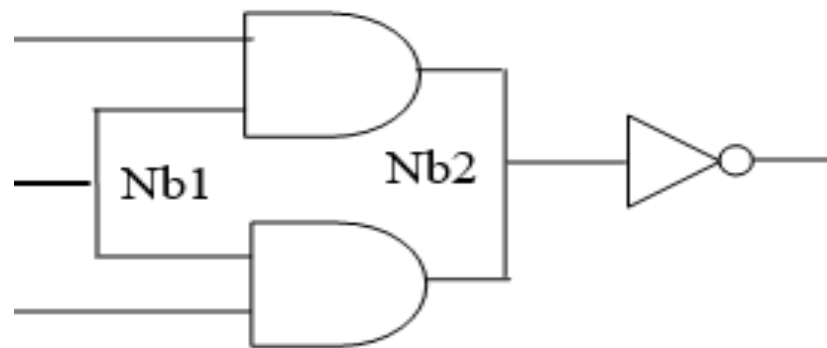
◆ 最简单的组合逻辑电路是**逻辑门**电路，用于实现基本逻辑运算

◆ 组合逻辑电路**构成规则**

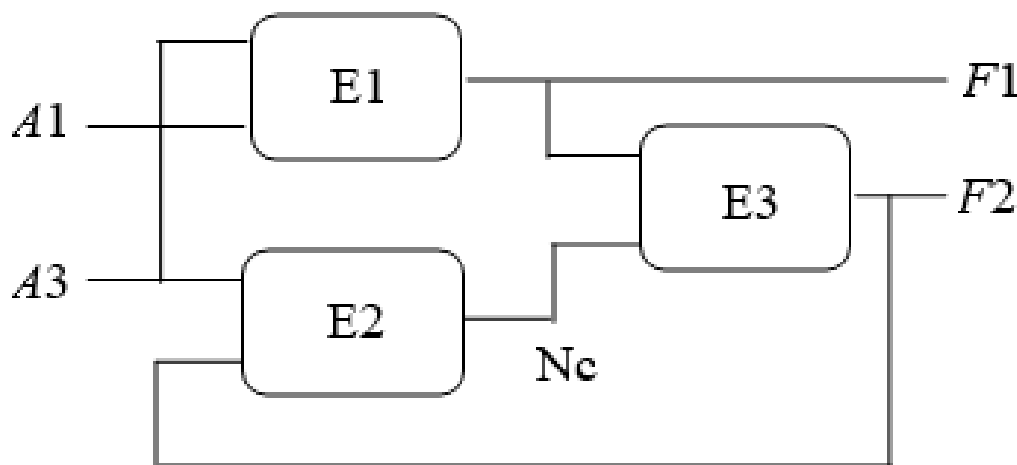
- 每个元件本身是组合逻辑电路
- 输出结点不能互连
- 输出结点不能反馈到输入端



是组合逻辑电路



不是组合逻辑电路



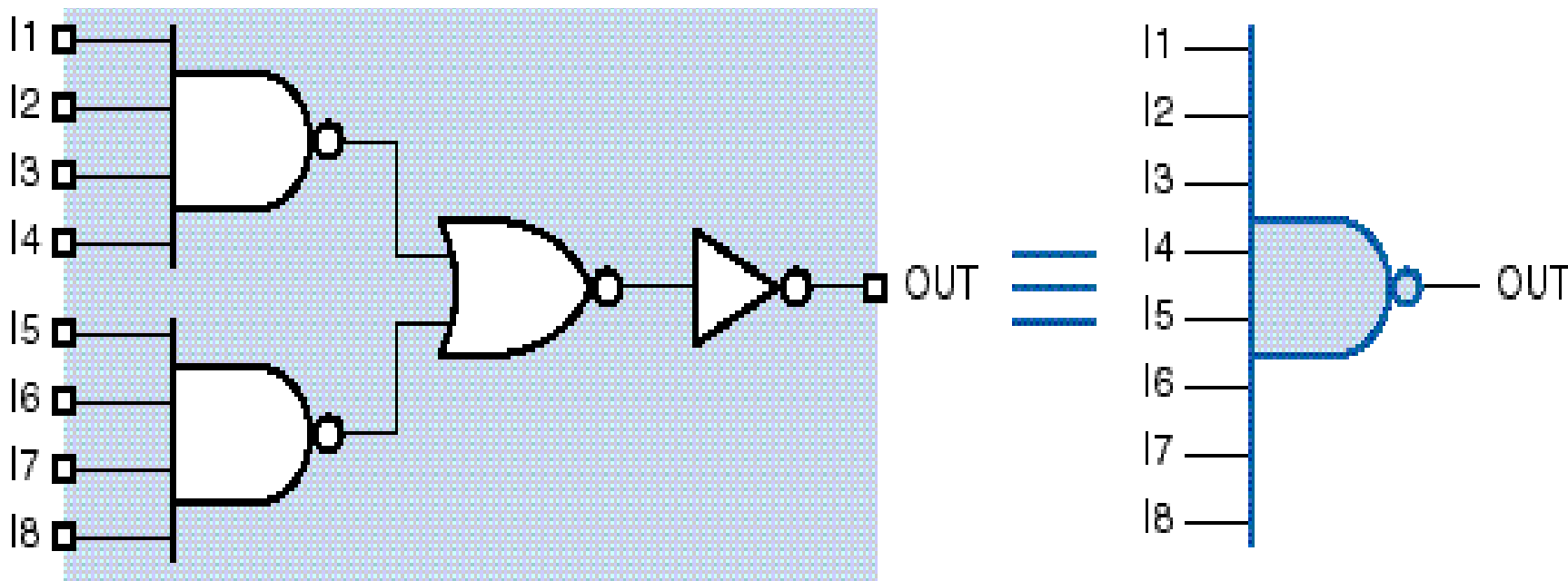
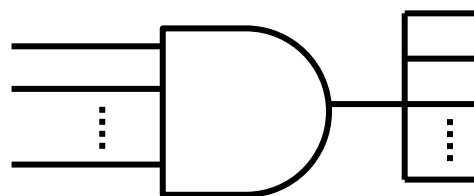
不是组合逻辑电路

1.2 逻辑电路图

- ◆ **逻辑电路图**描述数字电路内部元件的结构及其相互连接关系
- ◆ 每个逻辑电路图对应**一个**逻辑表达式
- ◆ 一个真值表可能对应**多个**不同的逻辑表达式，从而对应**多个**不同的逻辑电路图，因而可以有**多个不同的**实现方式
- ◆ 任何逻辑表达式都可写成与、或、非三种基本运算的逻辑组合
- ◆ 一个逻辑门的输出可作为另一个逻辑门的输入

1.2 逻辑电路图

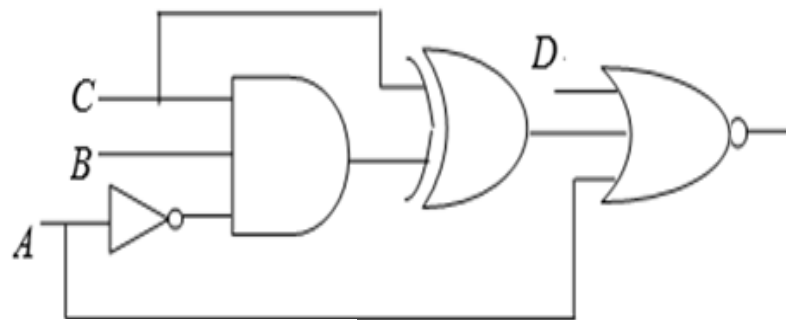
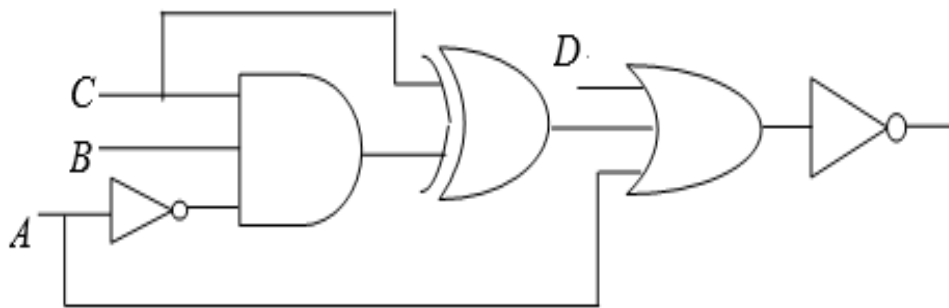
- ◆ **扇入系数**：一个逻辑门所允许的输入端的最大数目
- ◆ **扇出系数**：一个逻辑门输出端信号所能驱动的下一级输入端的最大数目



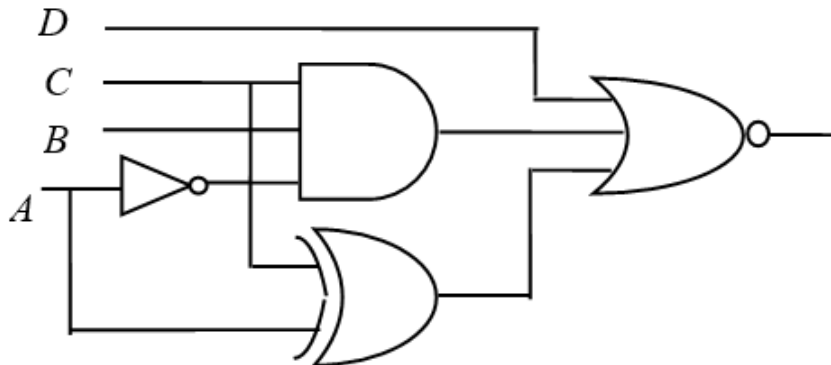
1.2 逻辑电路图

- ◆ 画逻辑电路图时，须依据逻辑运算的**优先级**确定逻辑门间的**连接关系**
 - 优先级**高**的运算对应的逻辑门的**输出**，是优先级**低**的运算对应逻辑门的**输入**
 - 优先级顺序如下：**非** > **与和与非** > **异或和同或** > **或和或非**

例：画出 $\overline{A \cdot B \cdot C \oplus C + A + D}$ 对应的逻辑电路图



$\overline{A \cdot B \cdot C \oplus C + A + D}$?



1.2 逻辑电路图

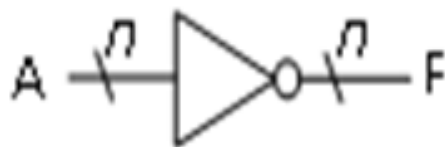
◆ n位逻辑运算在输入端和输出端标注位数即可



$$F=A \bullet B$$



$$F=A+B$$



$$F=\overline{A}$$



$$F=A \oplus B$$

1.3 两级和多级组合逻辑电路

- ◆ 信号通过逻辑门时在时间上存在延迟。从输入信号改变开始，到输出信号发生改变所用的时间称为**门延迟 (gate delay)**
- ◆ 任何逻辑表达式都可以转换成**与-或表达式**和**或-与表达式**，因此，任何组合逻辑电路都可以是一个**两级电路**
- ◆ 与-或表达式对应电路：**第一级是若干个与门，第二级是一个或门**

1.3 两级和多级组合逻辑电路

- ◆ 信号通过逻辑门时在时间上存在延迟。从输入信号改变开始，到输出信号发生改变所用的时间称为**门延迟 (gate delay)**
- ◆ 任何逻辑表达式都可以转换成**与-或表达式**和**或-与表达式**，因此，任何组合逻辑电路都可以是一个**两级电路**
- ◆ 与-或表达式对应电路：**第一级是若干个与门，第二级是一个或门**

例： $\overline{\overline{A} \cdot B \cdot C \oplus C + A + D}$ 可转换为 $\overline{A} \cdot B \cdot \overline{D} + \overline{A} \cdot \overline{C} \cdot \overline{D}$

转换前：最长路径从A输入端到输出经过了非门、与门、异或门、或门和非门；**转换后：**最长路径只经过非门、与门和或门

设与、或门延迟时间都是2ns，异或门延迟为3ns，不考虑非门延迟，则**转换前传输时间为7ns，转换后为4ns**

1.3 两级和多级组合逻辑电路

- ◆ 信号通过逻辑门时在时间上存在延迟。从输入信号改变开始，到输出信号发生改变所用的时间称为**门延迟 (gate delay)**
- ◆ 任何逻辑表达式都可以转换成**与-或表达式**和**或-与表达式**，因此，任何组合逻辑电路都可以是一个**两级电路**
- ◆ 与-或表达式对应电路：**第一级是若干个与门，第二级是一个或门**

例： $\overline{A} \cdot B \cdot C \oplus C + A + D$ 可转换为 $\overline{A} \cdot B \cdot \overline{D} + \overline{A} \cdot \overline{C} \cdot \overline{D}$

转换前：最长路径从A输入端到输出经过了非门、与门、异或门、或门和非门；**转换后：**最长路径只经过非门、与门和或门

设与、或门延迟时间都是2ns，异或门延迟为3ns，不考虑非门延迟，则**转换前传输时间为7ns，转换后为4ns**

- ◆ 两级组合逻辑电路**好处**：比多级组合逻辑电路的传输时间更短，速度更快；**坏处**：使用两级组合电路所需的硬件数量会成倍增长

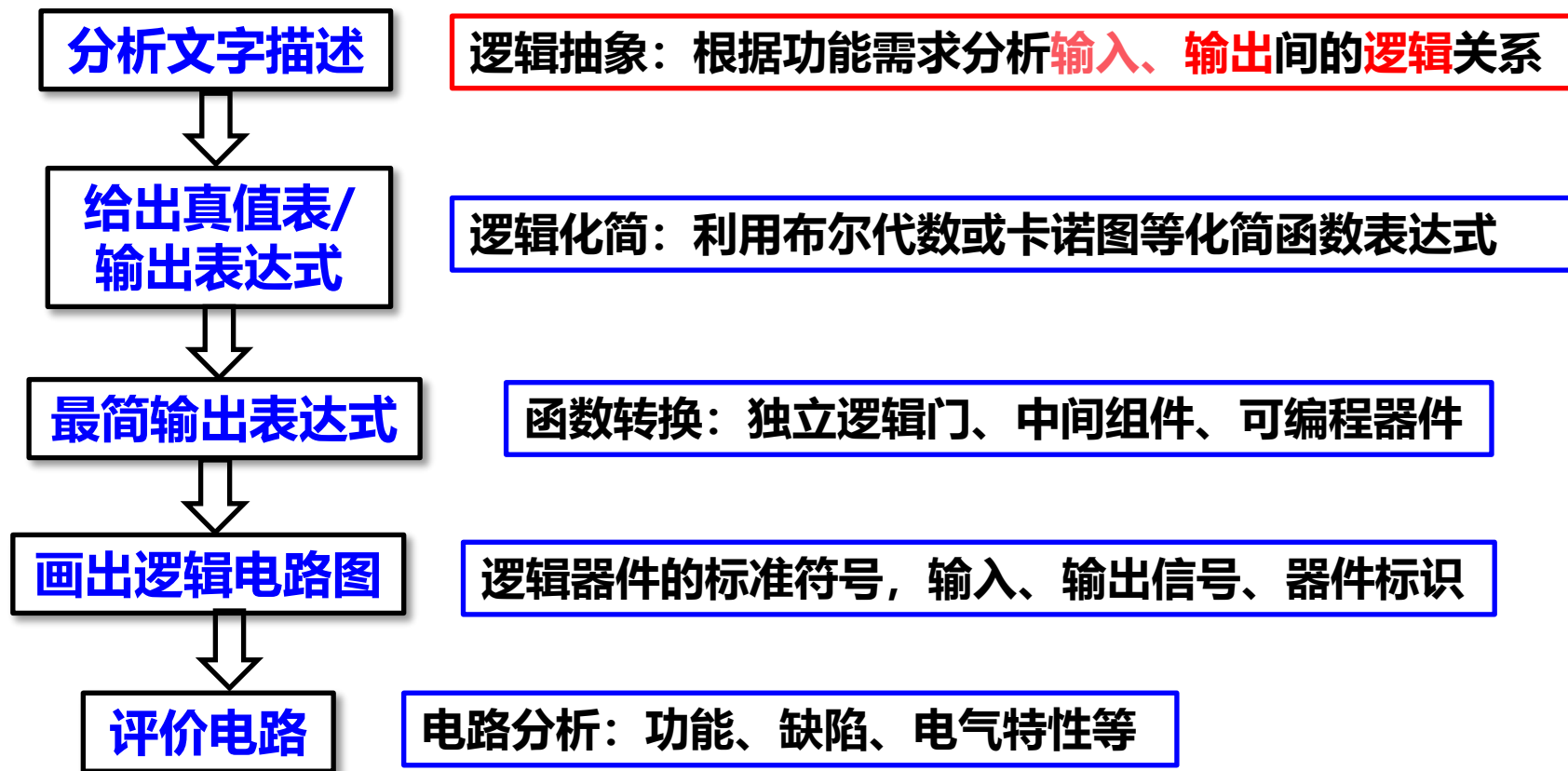
如 $F = A \oplus B \oplus C = \overline{A} \cdot \overline{B} \cdot \overline{C} + \overline{A} \cdot \overline{B} \cdot C + \overline{A} \cdot B \cdot \overline{C} + \overline{A} \cdot B \cdot C$ (8输入端异或门?)

128个8输入端与门和一个128输入端或门！7个2输入端异或门

- ◆ 采用两级还是多级需要在**速度和成本**之间进行权衡

1.4 组合逻辑电路设计

从文字描述到逻辑电路或系统设计的整个过程如下：

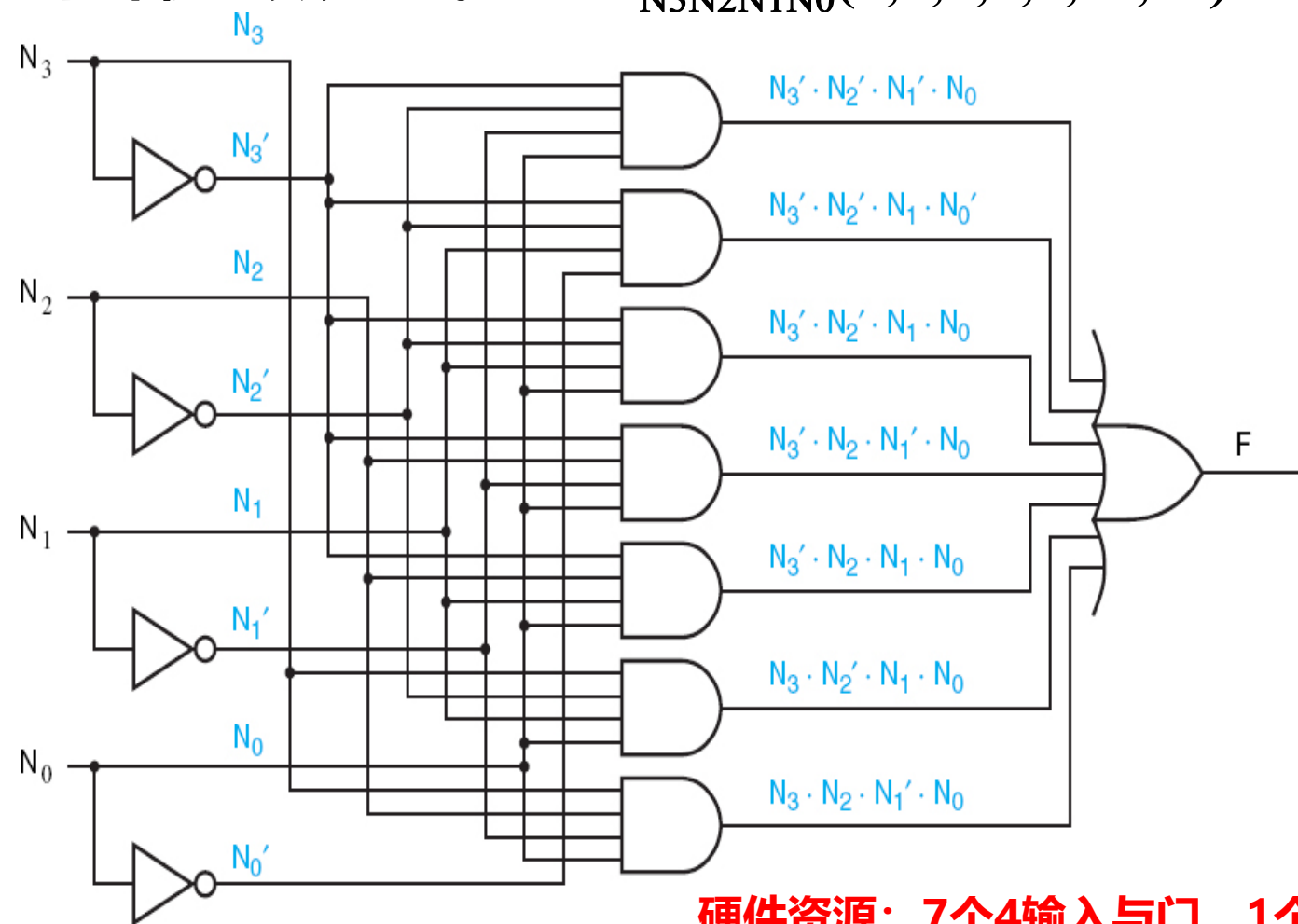


1.4 组合逻辑电路设计

例1：素数检测器的设计

• 4-bit input, $N_3N_2N_1N_0$

写出最小项表达式 $F = \sum_{N_3N_2N_1N_0}(1,2,3,5,7,11,13)$



列出真值表

row	N_3	N_2	N_1	N_0	F
0	0	0	0	0	0
1	0	0	0	1	1
2	0	0	1	0	1
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	0
10	1	0	1	0	0
11	1	0	1	1	1
12	1	1	0	0	0
13	1	1	0	1	1
14	1	1	1	0	0
15	1	1	1	1	0

硬件资源：7个4输入与门、1个7输入或门

1.4 组合逻辑电路设计

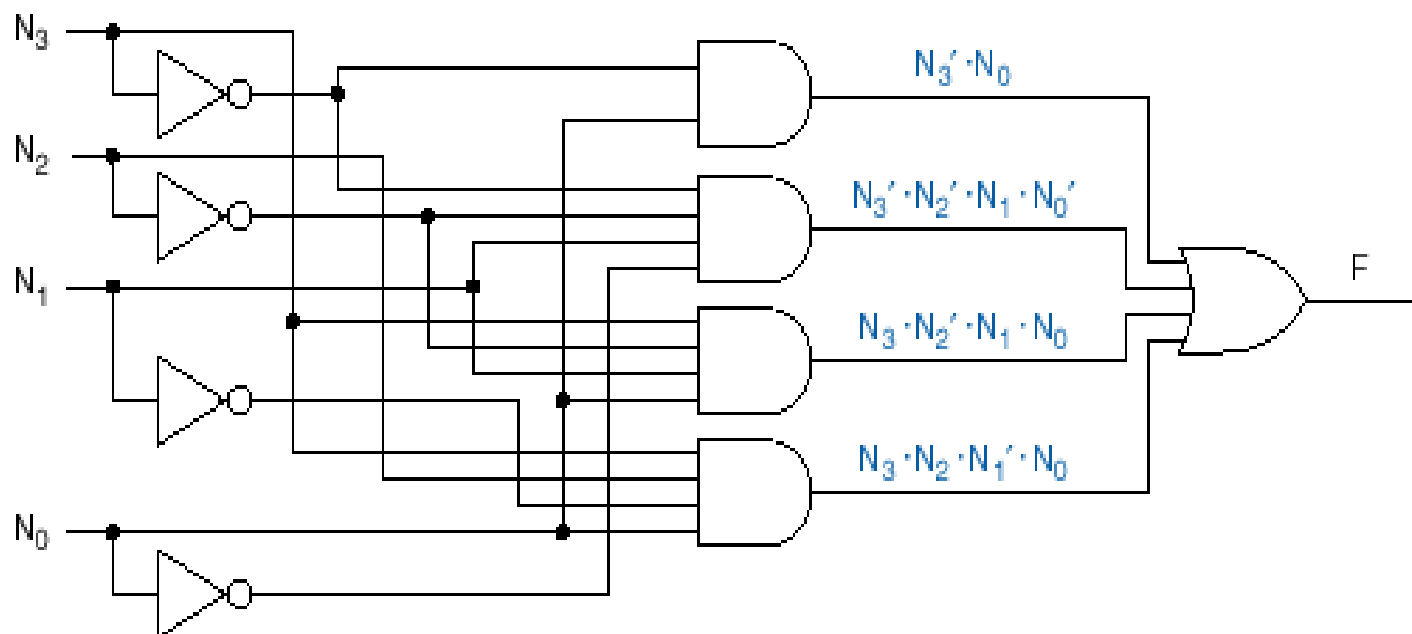
利用布尔代数化简, 以减少逻辑门数和输入端数 $X \cdot Y + X \cdot Y' = X$

$$F = \sum_{N_3 N_2 N_1 N_0} (1, 2, 3, 5, 7, 11, 13)$$

$$= N_3' \cdot N_2' \cdot \underline{N_1'} \cdot N_0 + N_3' \cdot N_2' \cdot \underline{N_1} \cdot N_0 + N_3' \cdot N_2 \cdot \underline{N_1'} \cdot N_0 + N_3' \cdot N_2 \cdot \underline{N_1} \cdot N_0 + \dots$$

$$= N_3' \cdot \underline{N_2'} \cdot N_0 + N_3' \cdot \underline{N_2} \cdot N_0 + \dots$$

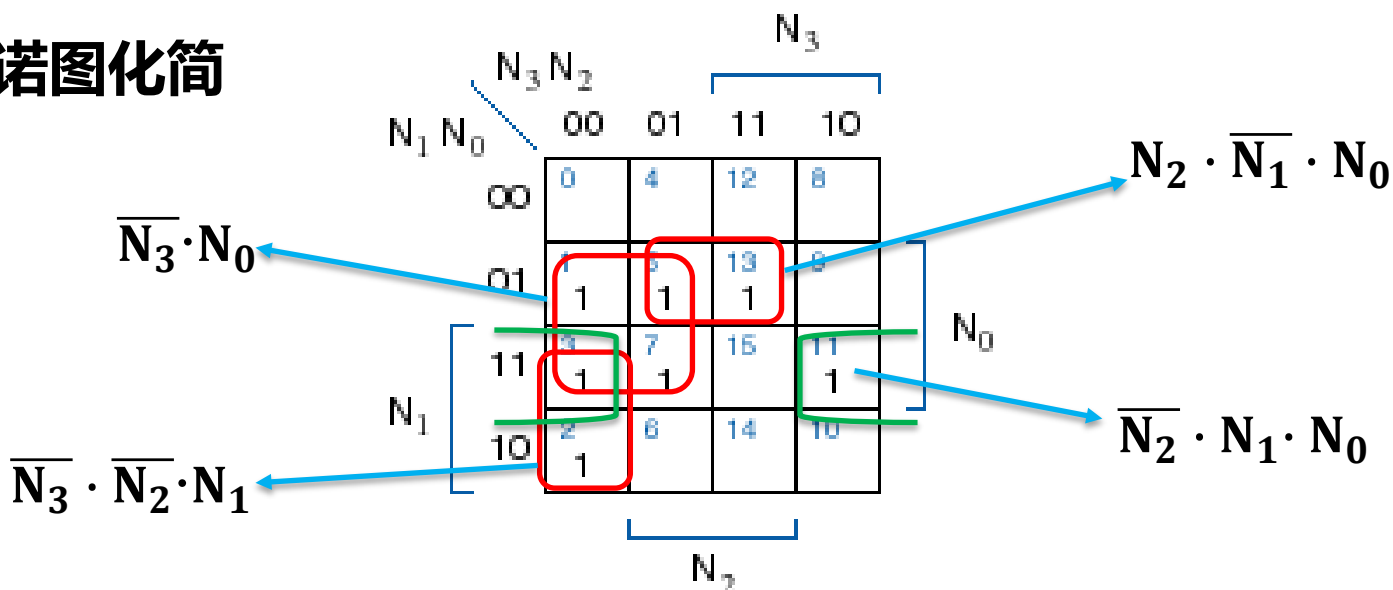
$$= N_3' \cdot N_0 + N_3' \cdot N_2' \cdot N_1 \cdot N_0' + N_3 \cdot N_2' \cdot N_1 \cdot N_0 + N_3 \cdot N_2 \cdot N_1' \cdot N_0$$



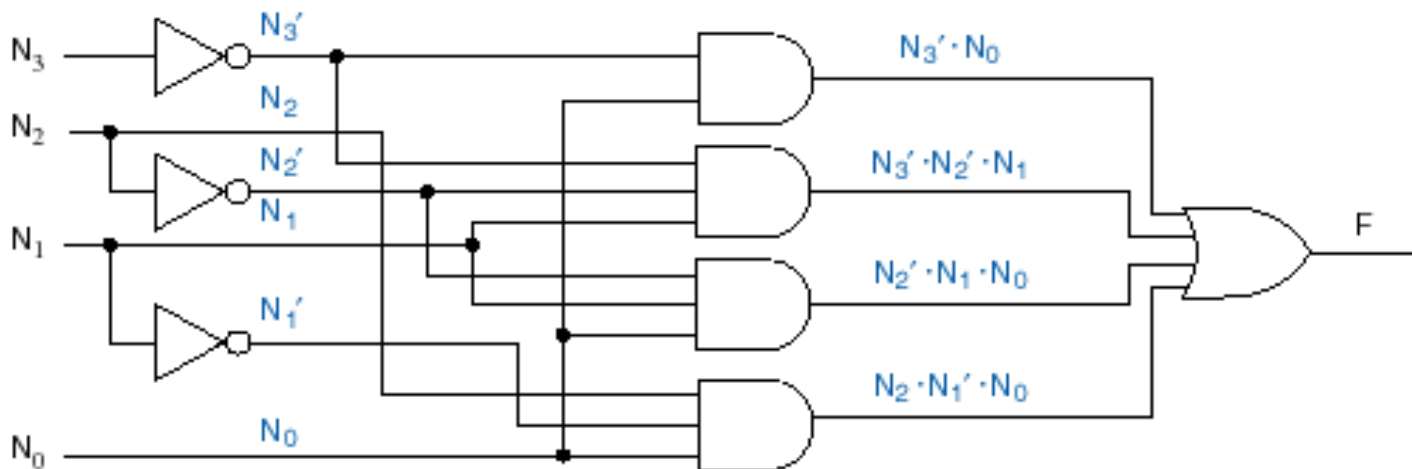
减少: 3个与门和17个输入端

1.4 组合逻辑电路设计

按卡诺图化简



$$F = \overline{N_3} \cdot N_0 + \overline{N_3} \cdot \overline{N_2} \cdot N_1 + N_2 \cdot \overline{N_1} \cdot N_0 + N_2 \cdot N_1 \cdot N_0$$



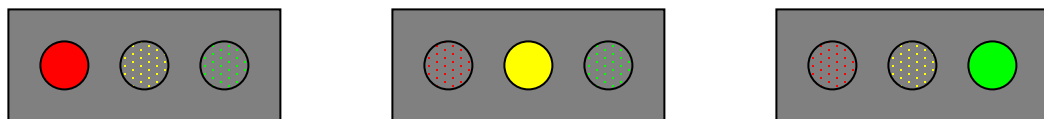
减少：3个与门和20个输入端

1.4 组合逻辑电路设计

例2：设计一个监视交通信号灯工作状态的逻辑电路

1、进行逻辑抽象

正常工作状态



输入变量：红R 黄Y 绿G 三盏灯的状态

灯亮为1，不亮为0

输出变量：故障信号F

正常工作为0，发生故障为1

故障状态	真 值 表			
	R	Y	G	F
	0	0	0	1
	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

1.4 组合逻辑电路设计

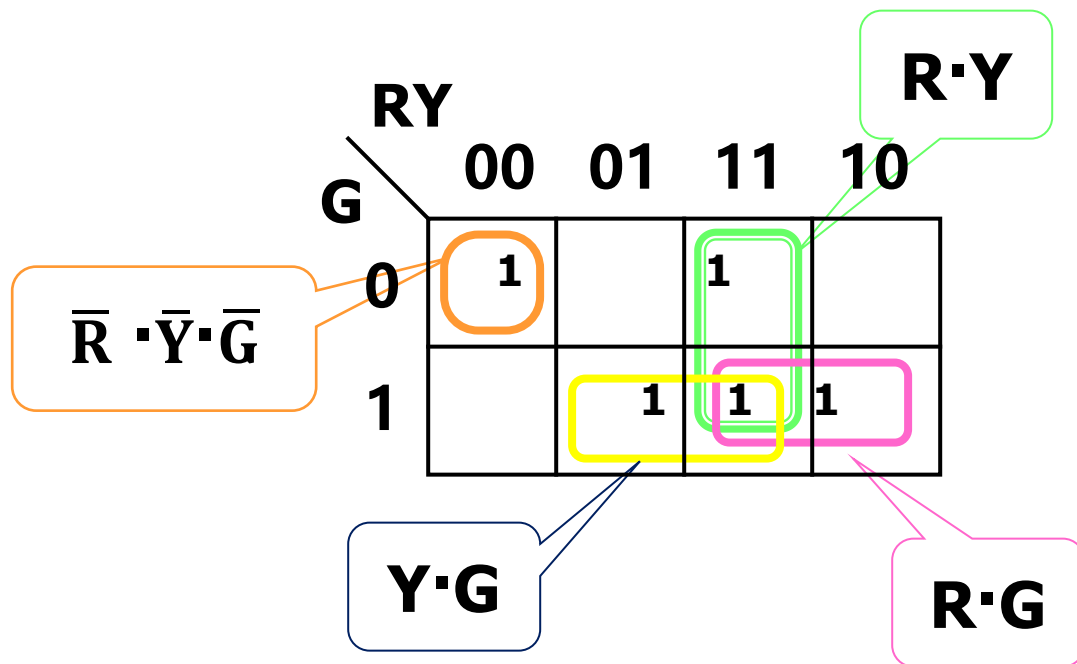
逻辑抽象结果

真值表

R	Y	G	F
0	0	0	1
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

2、逻辑化简

写出逻辑函数式并化简

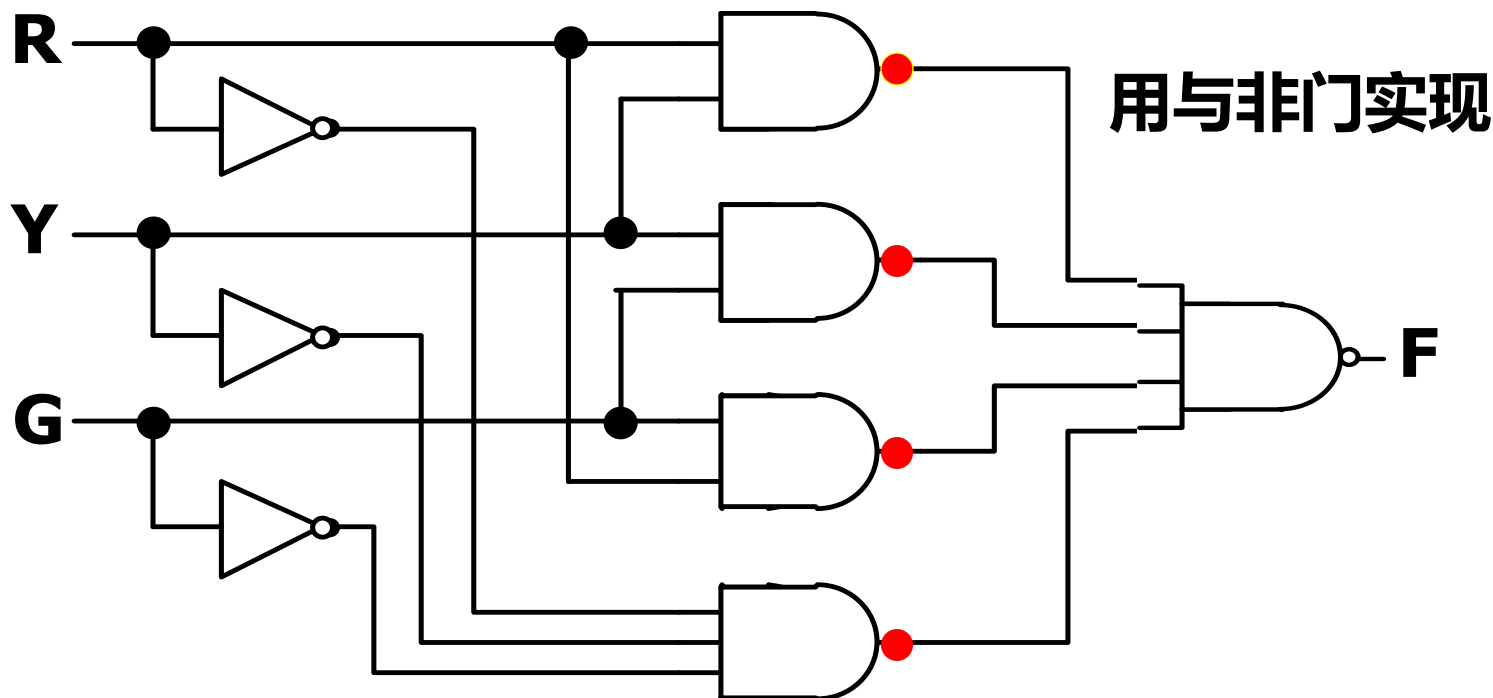


$$F = \bar{R} \cdot \bar{Y} \cdot \bar{G} + R \cdot Y + R \cdot G + Y \cdot G$$

1.4 组合逻辑电路设计

3、电路设计

$$F = \bar{R} \cdot \bar{Y} \cdot \bar{G} + R \cdot Y + R \cdot G + Y \cdot G$$



以上：正常工作为0，发生故障为1

如果：正常工作为1，发生故障为0？

1.5 无关项、非法值和高阻态

◆ 无关项

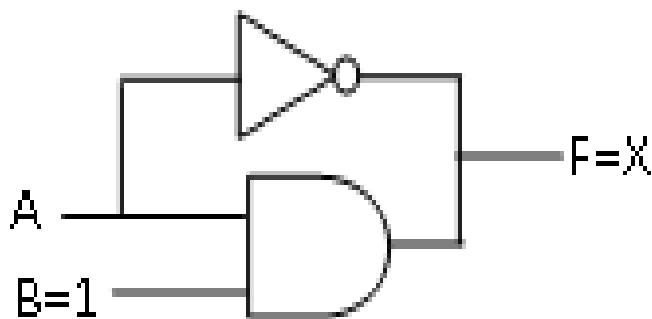
- 某些输入组合对应的输出值可以是任意值，某些输入组合不可能出现。这些输入组合对应的输出值在化简时可标识为**d**，表示可以取值**0**或**1**，具体数值根据化简的需要而定
- 可简化电路从而降低成本，但也更易受干扰

例如：8421 BCD码输入时，大于1001的编码为无关项

◆ 非法值

- 信号值不能被有效识别为高电平或低电平，处于不确定状态。

例如：下图中的信号X

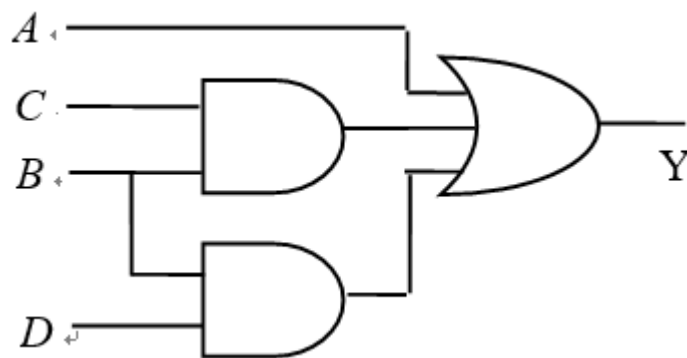
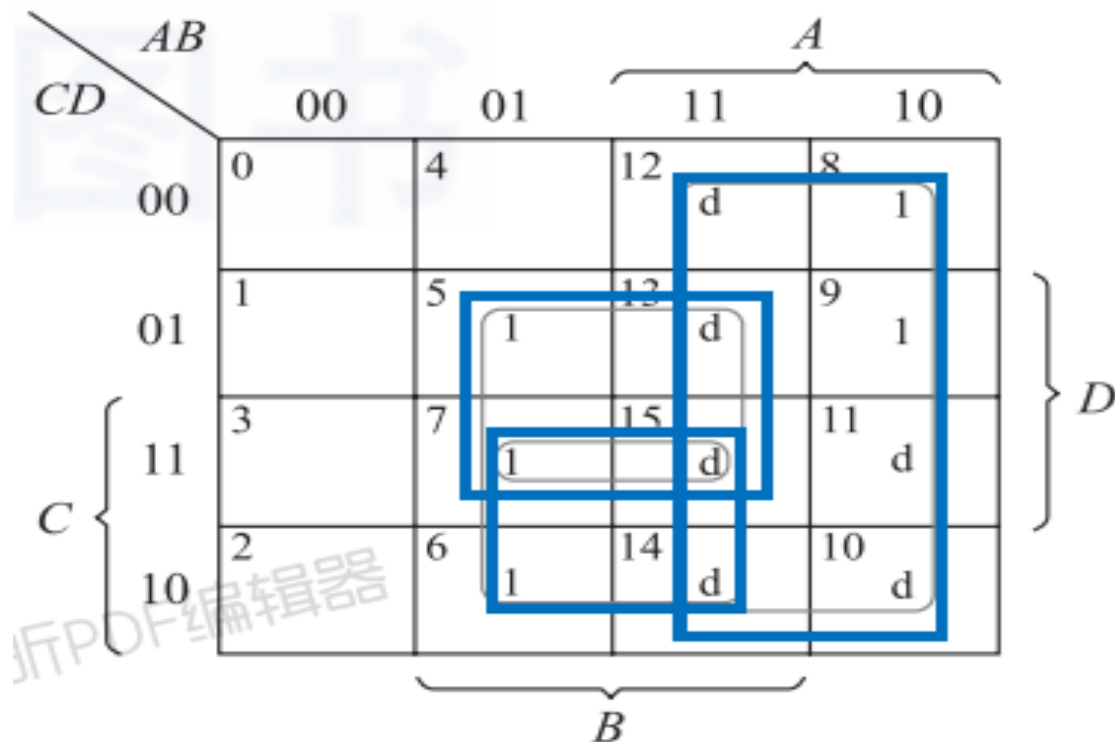


不管A是0还是1，F结点都会同时被高电平和低电平驱动，可能导致在F结点处之间有较大电流流动，使电路发热而被损坏

1.5 无关项、非法值和高阻态

例：设计一个检测电路，当NBCD（8421）码数值大于等于5时，输出为1

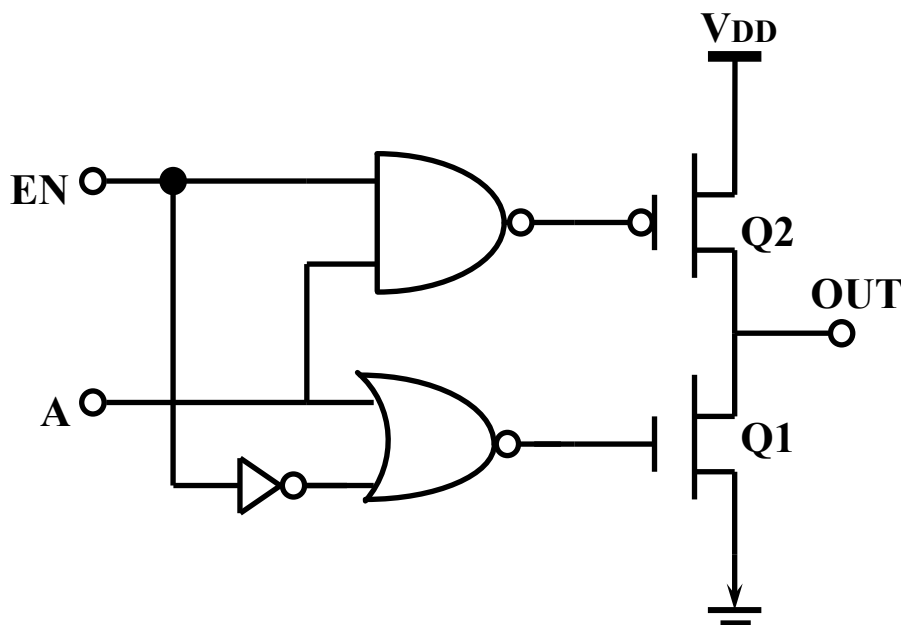
<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>Y</i>
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	d
1	0	1	1	d
1	1	0	0	d
1	1	0	1	d
1	1	1	0	d
1	1	1	1	d



$$Y=A+B\cdot C+B\cdot D$$

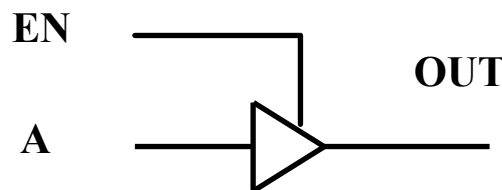
1.5 无关项、非法值和高阻态

- ◆ **高阻态Hi-Z**: 输出处于非正常逻辑态的第三种电气态, 好像和电路断开一样
- ◆ **三态门 (three-state gate)** 是一种重要的总线接口电路, 也称三态缓冲器, 其输出既可以是通常的逻辑值1 或 0, 又可以是高阻态
- ◆ 三态门有一个额外的输出**使能控制端EN**



使能控制端EN高电平有效

EN	A	Q1	Q2	OUT
L	L	off	off	Hi-Z
L	H	off	off	Hi-Z
H	L	on	off	L
H	H	off	on	H



第2章的
传输门是
双向传输

1.5 无关项、非法值和高阻态

三态门用途：可用于连接总线，多个三态输出连在一起等

