

数 字 逻 辑

Digital Logic Circuit

丁 贤 庆

ahhfdxq@163.com

Home work (P218)

1、下周三（四）的课上会有一次**10**分钟的随堂测验。

2、今天的作业：

4.4.14

4.4.20

4.4.26

4.4.36

4.4.37

第4章 组合逻辑电路

Combinational Logic Circuit

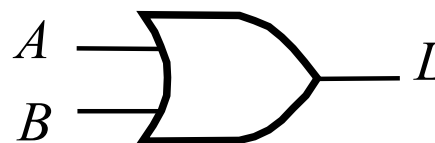
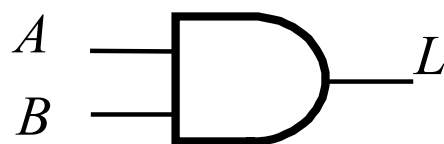
4.3 组合逻辑电路中的竞争冒险

4.3.1 产生的竞争冒险的原因

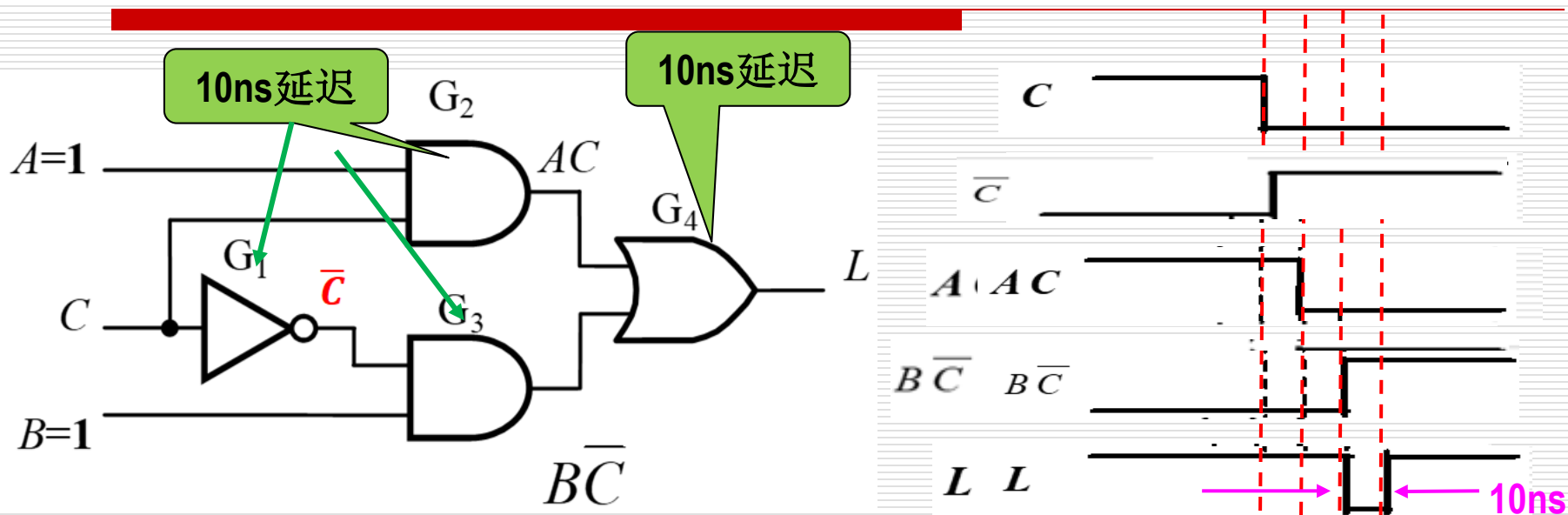
不考虑门的延时时间，且 $B=\bar{A}$

$$L = AB = 0$$

$$L = A + B = 1$$



考虑门的延时时间，且用非门实现 $B=\bar{A}$ 时



若不考虑10ns的门延时: $L = AC + B\bar{C} = 1 \cdot C + 1 \cdot \bar{C} = 1$

竞争: 当一个逻辑门的两个输入端的信号同时向相反方向变化, 而变化的时间有差异的现象。

冒险: 两个输入端的信号取值的变化方向是相反时, 如门电路输出端的逻辑表达式简化成两个互补信号相乘或者相加, 由竞争而可能产生输出干扰脉冲的现象。

竞争-冒险的检查方法

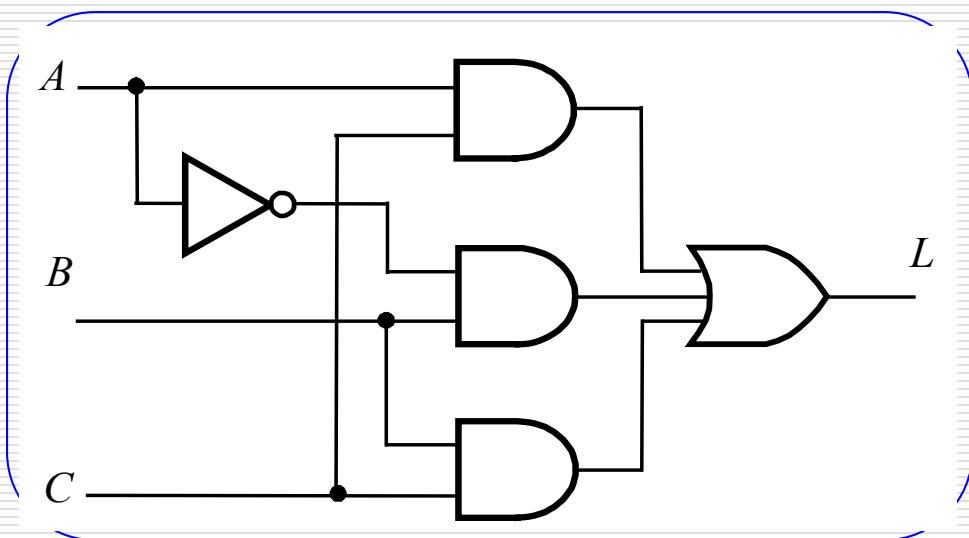
在输入变量每次只有一个改变状态的简单情况下，如果函数表达式中同时存在有 A 和 \bar{A} ，那么我们称 A 为具有竞争能力的变量。对于具有竞争能力的变量，若将其余变量任意取值，函数表达式能够转化成 $Y=A\bar{A}$ 或者 $Y=A+\bar{A}$ 形式之一的，会发生竞争-冒险。

由于竞争-冒险产生的尖峰脉冲持续的时间很短，包含的能量很小，所以大多数竞争-冒险并不会对电路造成危害。但是，如果负载是对尖峰脉冲敏感的存储电路时，竞争-冒险就有可能使存储电路发生误动作而产生错误，因此设计数字系统时应尽量避免竞争-冒险现象的发生。

4.3.2 消去竞争冒险的方法

1. 发现并消除互补变量

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



$B = C = 0$ 时

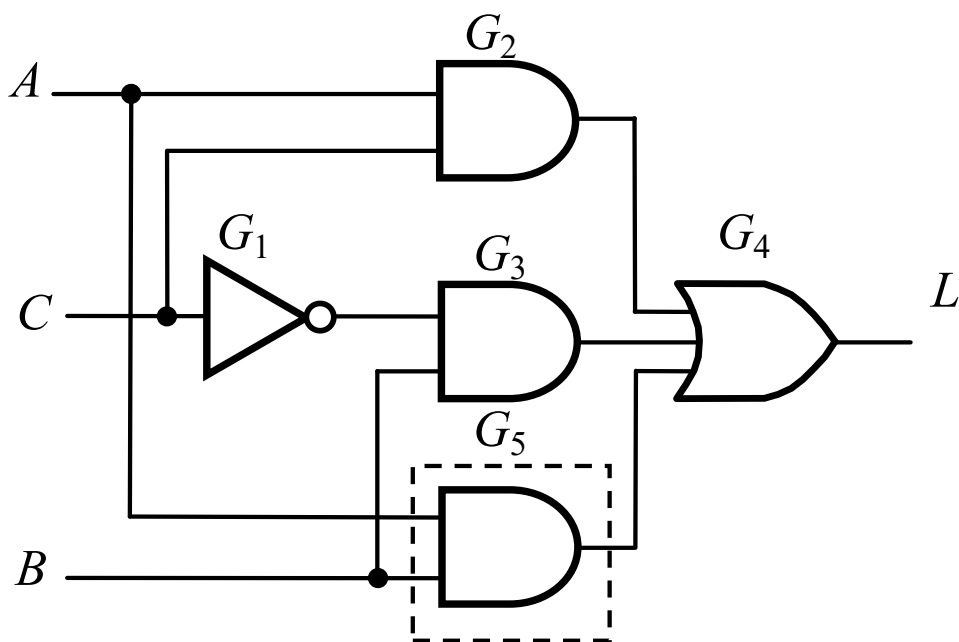
$$L = A\overline{A}$$

可能出现竞争冒险。

为消掉 $A\overline{A}$ ，变换逻辑函数式为

$$L = AC + \overline{A}B + BC$$

2. 增加乘积项, 避免互补项相加



$$L = AC + B\bar{C}$$

当 $A=B=1$ 时

$$L = C + \bar{C}$$

$$L = AC + B\bar{C}$$

$$L = AC + B\bar{C} + AB$$

L	B	C	00	01	11	10
A	0	0	0	0	0	1
1	0	0	1	1	1	1

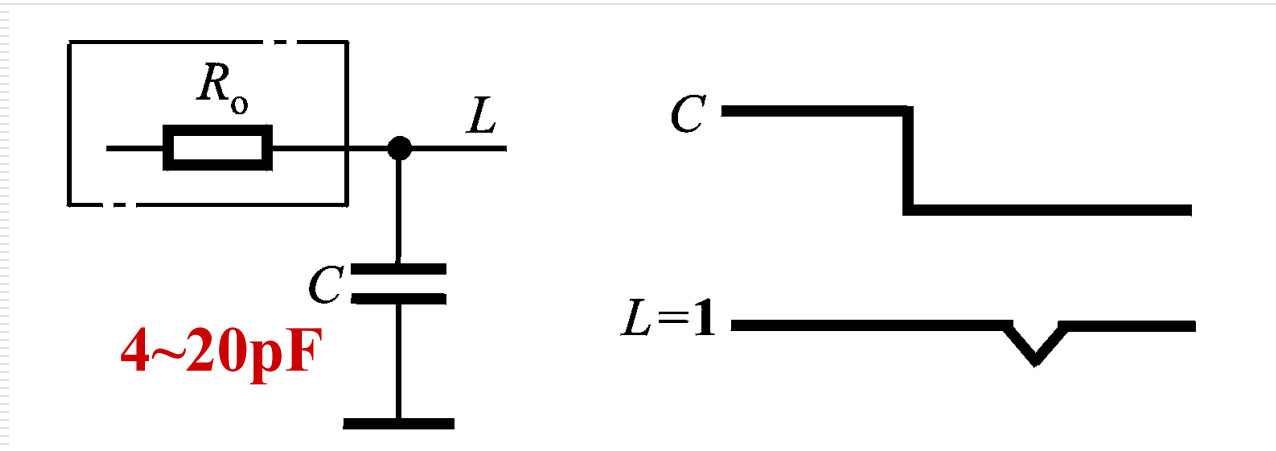
AB

当 $A=B=1$ 时, 根据逻辑表达式有

$$L = C + \bar{C} + 1$$

3. 输出端并联电容器

如果逻辑电路在较慢速度下工作，为了消去竞争冒险，可以在输出端并联一电容器，致使输出波形上升沿和下降沿变化比较缓慢，可对于很窄的负跳变脉冲起到平波的作用。



4.4 若干典型的组合逻辑电路

4.4.1 编码器

4.4.2 译码器/数据分配器

4.4.3 数据选择器

4.4.4 数值比较器

4.4.5 算术运算电路

4.4 若干典型的组合逻辑集成电路

4.4.1 编码器 Encoders

1、编码器 (Encoder)的定义与分类

编码：赋予二进制代码特定含义的过程称为**编码**。

如：8421BCD码中，用1000表示数字8

如：ASCII码中，用1000001表示字母A等

编码器：具有**编码功能的逻辑电路**。

编码器能将每一个编码输入信号变换为不同的二进制的代码输出。

1、编码器 (Encoder)的分类

按照编码器的**编码进制**来分, 有:

BCD编码器: 将10个编码输入信号分别编成10个4位码输出。

8线-3线编码器: 将8个输入的信号分别编成 8个3位二进制数码输出。

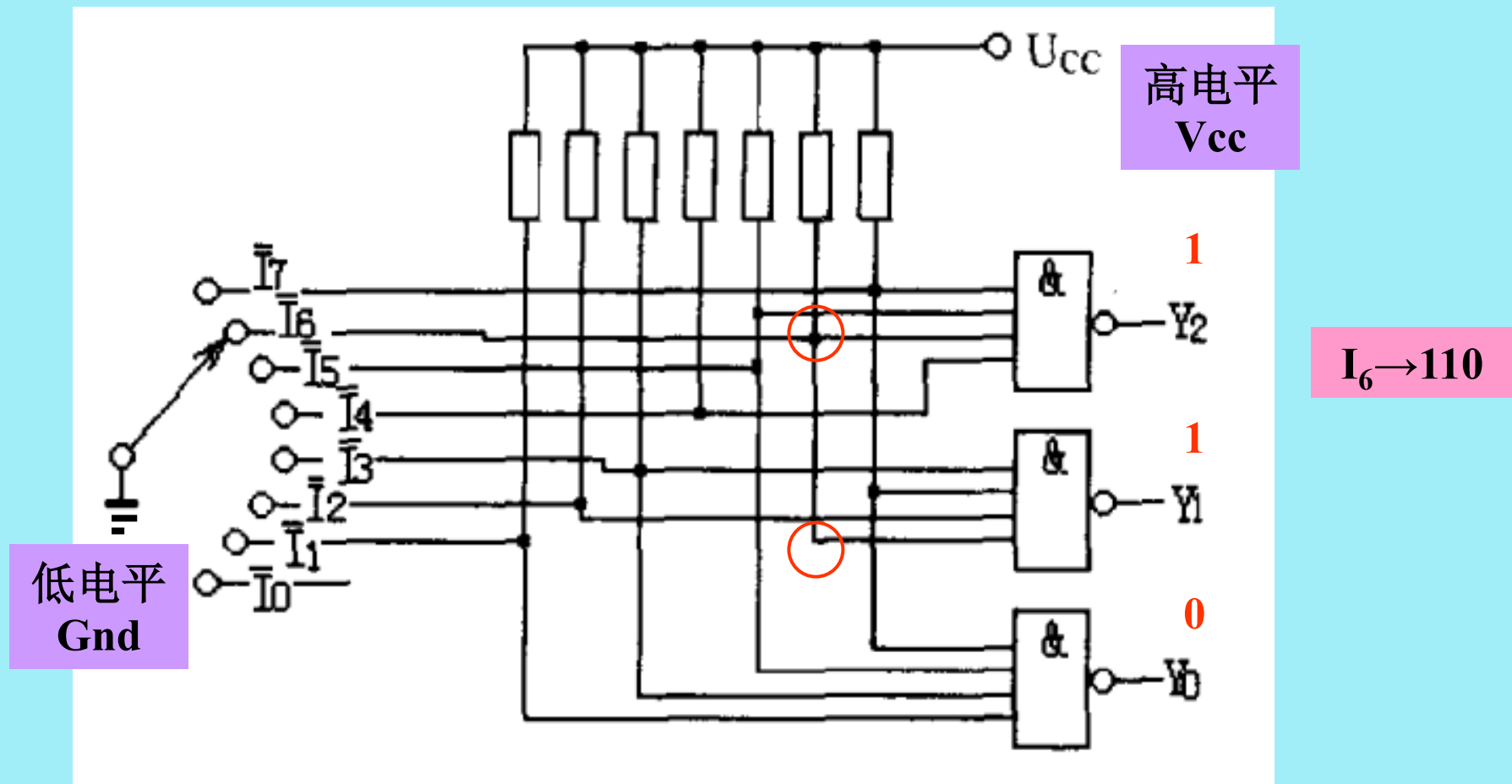
按照**优先级**来分: **普通编码器**和**优先编码器**。

普通编码器和优先编码器的区别：

普通编码器：任何时候只允许输入一个有效编码信号，否则输出就会发生混乱。

优先编码器：允许同时输入两个以上的有效编码信号。当同时输入几个有效编码信号时，优先编码器能按预先设定的优先级别，只对其中优先权最高的一个进行编码。

普通编码器电路



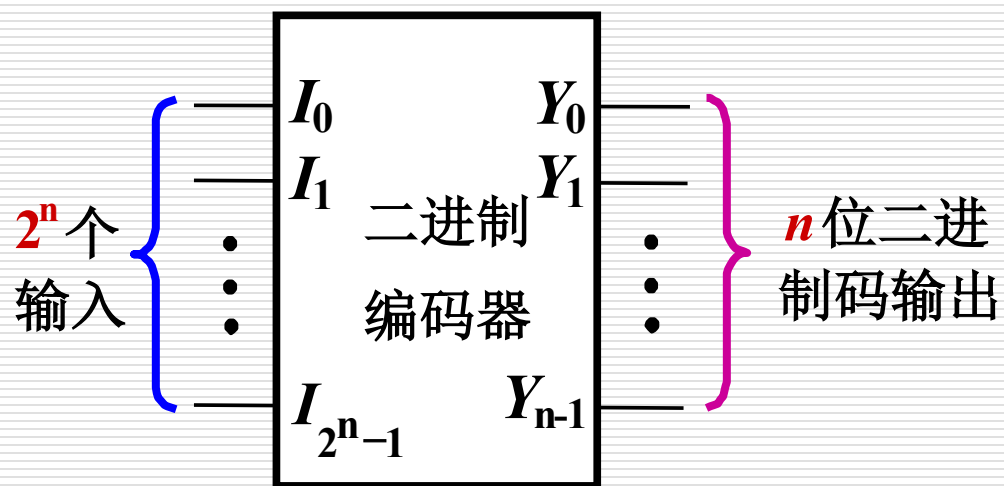
注意：此电路任意时刻只允许一个输入信号为有效信号，输出对这个输入信号编码。

问题：同时为低，编码混乱。

2、编码器的工作原理

普通二进制编码器

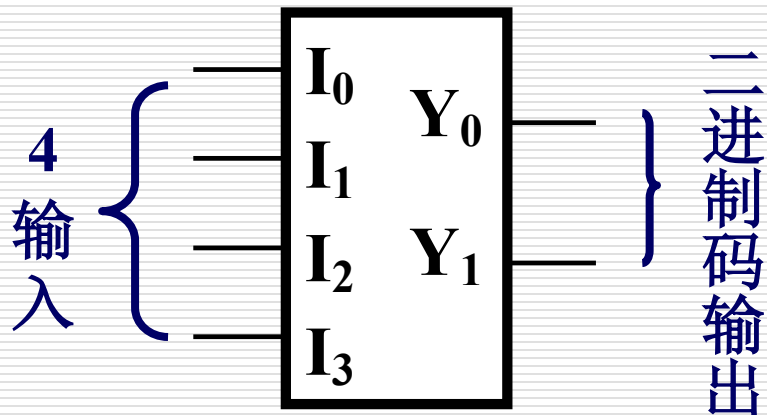
二进制编码器的结构框图



2、编码器的工作原理

(1) 4线—2线普通二进制编码器 (设计) (2) 逻辑功能表

(a) 逻辑框图



$$Y_1 = \bar{I}_0 \bar{I}_1 I_2 \bar{I}_3 + \bar{I}_0 \bar{I}_1 \bar{I}_2 I_3$$

$$Y_0 = \bar{I}_0 I_1 \bar{I}_2 \bar{I}_3 + \bar{I}_0 \bar{I}_1 \bar{I}_2 I_3$$

其它12种组合

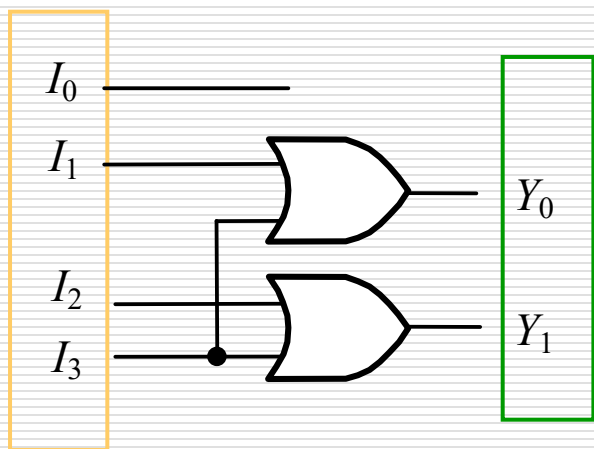
I_0	I_1	I_2	I_3	Y_1	Y_0
1	0	0	0		
0	1	0	0		
0	0	1	0		
0	0	0	1		
0	0	0	0	0	0
...	...			0	0
1	1	1	1	0	0

编码器的输入为高电平有效。

上述是将输入的其它12种组合对应的输出看做0。如果看做无关项，则表达式为

$$Y_1 = I_2 + I_3$$

$$Y_0 = I_1 + I_3$$

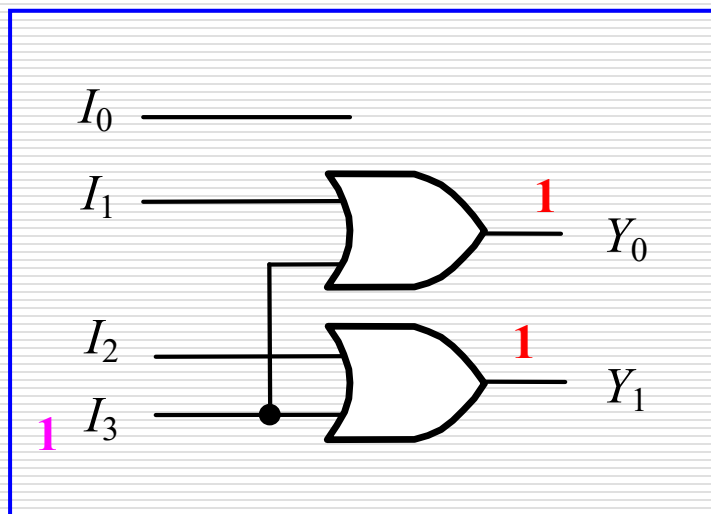


(2) 逻辑功能表

I_0	I_1	I_2	I_3	Y_1	Y_0
1	0	0	0		
0	1	0	0		
0	0	1	0		
0	0	0	1		
0	0	0	0		
			
1	1	1	1		

其它12种组合

若有2个以上的输入为有效信号？



当只有 I_3 为1时，

$$Y_1 Y_0 = ? \quad Y_1 Y_0 = 11$$

若 $I_1 = I_2 = 1$ ， $I_0 = I_3 = 0$ 时，

$$Y_1 Y_0 = ? \quad Y_1 Y_0 = 11$$

编码相同

无法输出有效编码。

结论：普通编码器不能同时输入两个以上的有效编码信号

3. 优先编码器

实际应用中，经常有两个或更多输入编码信号同时有效。



必须根据轻重缓急，规定好这些外设允许操作的先后次序，即优先级。

识别多个编码请求信号的优先级，并进行相应编码的逻辑部件称为优先编码器。

(2) 优先编码器线(4—2 线优先编码器) (设计)

输入为编码信号 $I_3 \sim I_0$ 输出为 $Y_1 Y_0$

输入编码信号高电平有效，输出为二进制代码

输入编码信号优先级从高到低为 $I_3 \sim I_0$

(1) 列出功能表

输 入				输 出	
I_0	I_1	I_2	I_3	Y_1	Y_0
1	0	0	0	0	0
×	1	0	0	0	1
×	×	1	0	1	0
×	×	×	1	1	1

低 ← 高

(2) 写出逻辑表达式

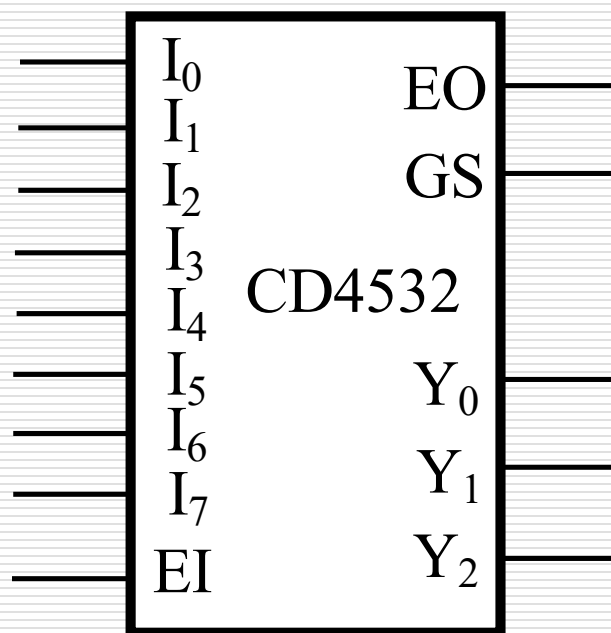
$$Y_1 = \overline{I_2} I_3 + I_3$$

$$Y_0 = \overline{I_1} \overline{I_2} I_3 + I_3$$

(3) 画出逻辑电路 (略)

2 典型编码器电路

优先编码器CD4532的示意框图



8—3 线优先编码器,
真值表如下页所示。

当EI为0时，不论I7~I0怎样变化，GS=0,EO=0

当EI为1时，如I7~I0中有一个1，则GS=1,EO=0

优先编码器CD4532功能表

输 入									输 出			
<i>EI</i>	<i>I</i> ₇	<i>I</i> ₆	<i>I</i> ₅	<i>I</i> ₄	<i>I</i> ₃	<i>I</i> ₂	<i>I</i> ₁	<i>I</i> ₀	<i>Y</i> ₂	<i>Y</i> ₁	<i>Y</i> ₀	<i>EO</i>
0	×	×	×	×	×	×	×	×	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	1
1	1	×	×	×	×	×	×	×	1	1	1	0
1	0	1	×	×	×	×	×	×	1	1	0	0
1	0	0	1	×	×	×	×	×	1	0	1	0
1	0	0	0	1	×	×	×	×	1	0	0	0
1	0	0	0	0	1	×	×	×	0	1	1	0
1	0	0	0	0	0	1	×	×	0	1	0	0
1	0	0	0	0	0	0	1	×	0	0	1	0
1	0	0	0	0	0	0	0	1	0	0	0	0

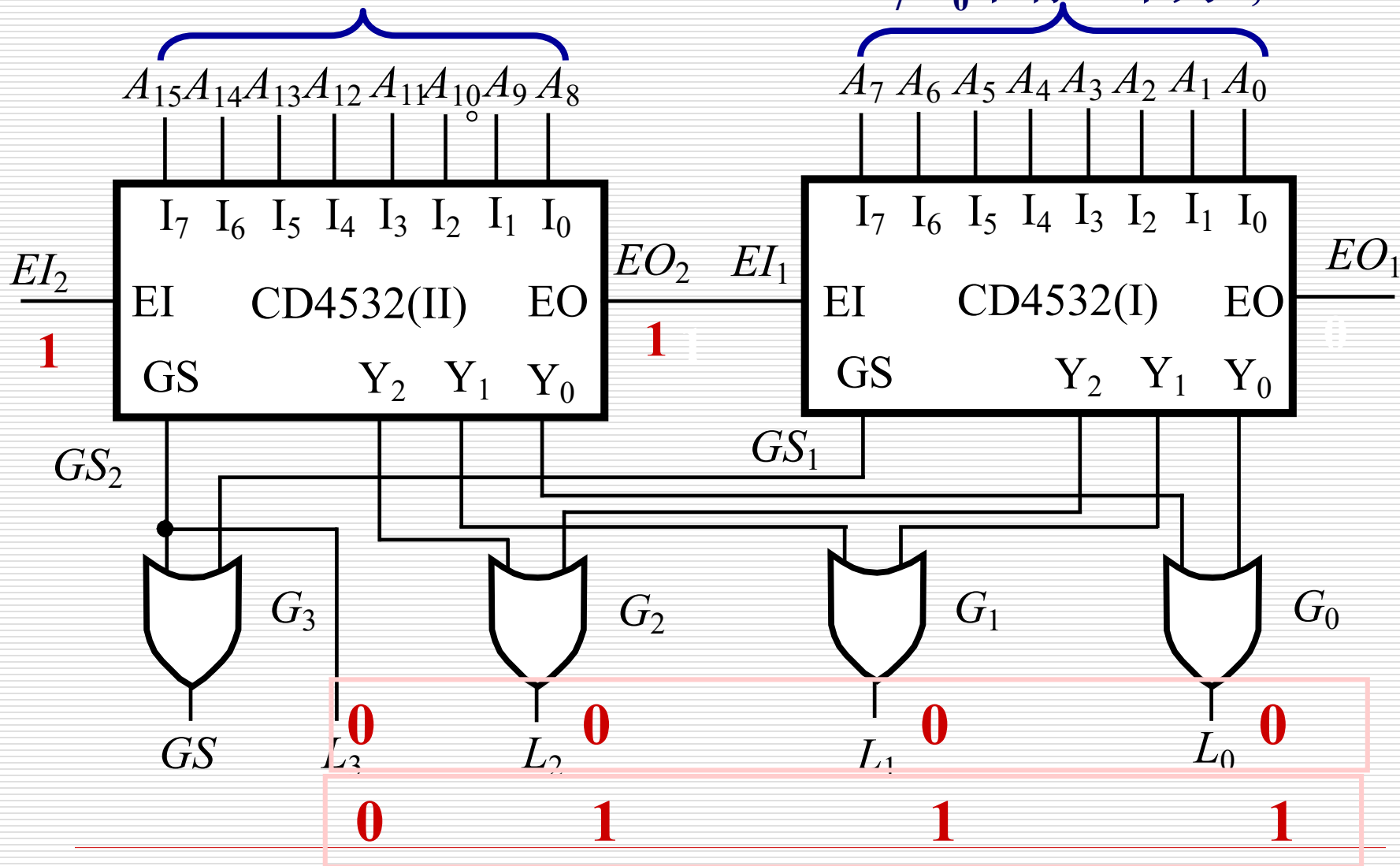
为什么要设计GS、EO输出信号？

GS、EO输出信号常用多个芯片的级联

(了解)

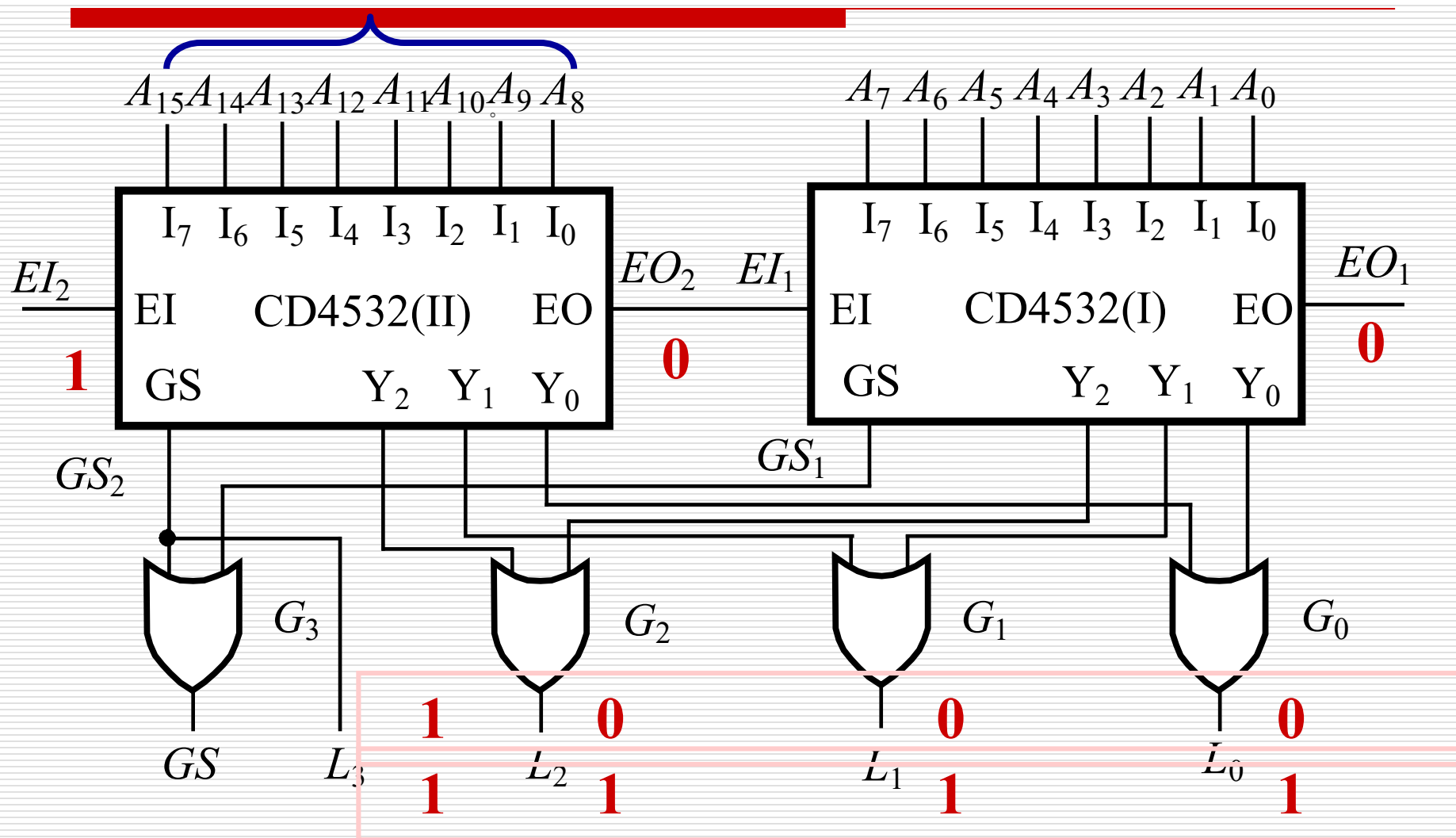
若无有效电平输入,即
 $I_{15} \sim I_8 = 0 \sim 0$;

若有效电平输入,即
 $I_7 \sim I_0$ 中有一个为1;



(了解) 若有效电平输入即 $I_{15} \sim I_8$ 中有一个为1;

哪块芯片的优先级高?



2. 8421BCD码编码器功能表

输 入										输 出					
$\overline{S_0}$	$\overline{S_1}$	$\overline{S_2}$	$\overline{S_3}$	$\overline{S_4}$	$\overline{S_5}$	$\overline{S_6}$	$\overline{S_7}$	$\overline{S_8}$	$\overline{S_9}$	A	B	C	D	GS	
1	1	1	1	1	1	1	1	1	1						
1	1	1	1	1	1	1	1	1	0						
1	1	1	1	1	1	1	1	0	1						
1	1	1	1	1	1	1	0	1	1						
1	1	1	1	1	1	0	1	1	1						
1	1	1	1	1	0	1	1	1	1						
1	1	1	1	0	1	1	1	1	1						
1	1	1	0	1	1	1	1	1	1						
1	1	0	1	1	1	1	1	1	1						
1	0	1	1	1	1	1	1	1	1						
0	1	1	1	1	1	1	1	1	1						

该编码器为输入低电平有效，输出高电平有效，**GS**为标志位。

1 译码器的定义与分类

译码：译码是编码的逆过程，它可将二进制码翻译成代表某一特定含义的信号。(即电路的某种状态)

译码器：具有译码功能的逻辑电路称为译码器。

译码器的分类：

地址译码器 将一系列代码转换成与之一一对应的有效信号。

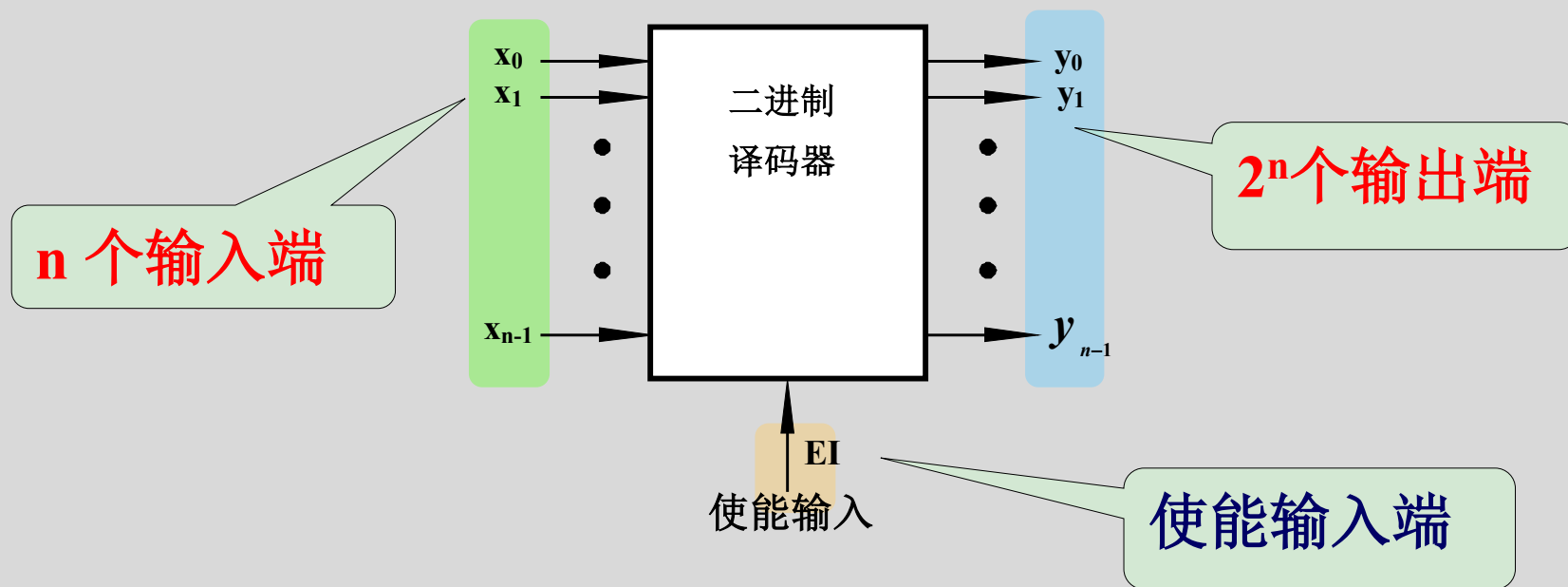
代码变换器 将一种代码转换成另一种代码。

常见的地址译码器：

- 二进制译码器
- 二—十进制译码器
- 显示译码器

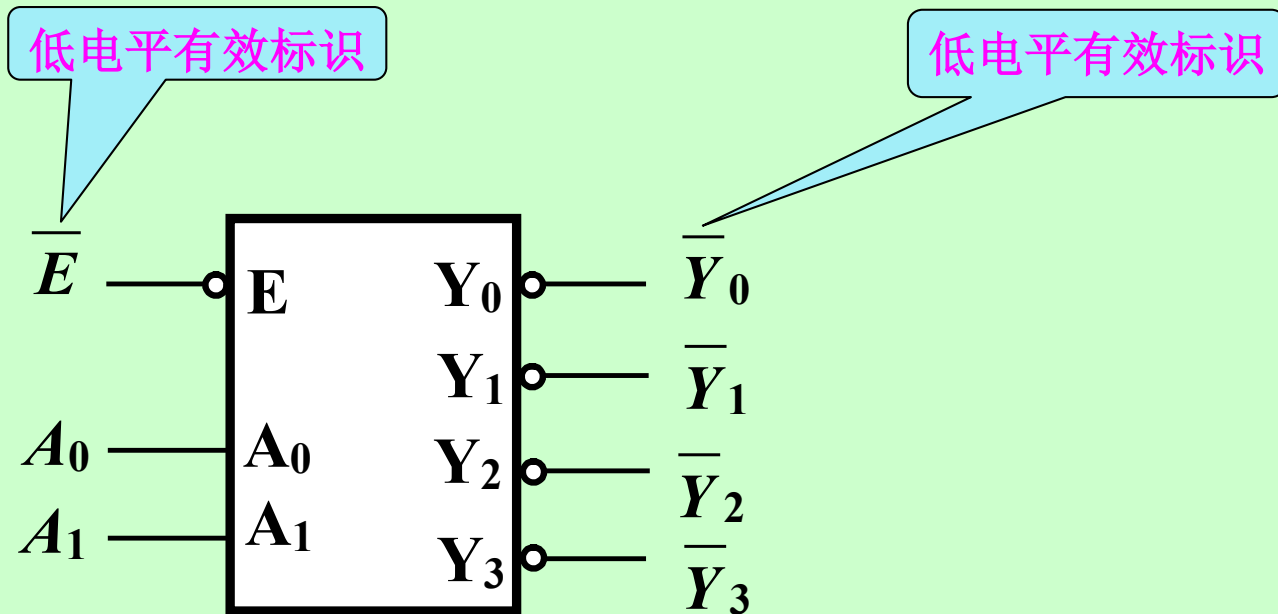
2. 典型译码器电路及应用

(1) 二进制译码器



设输入端的个数为 n ，输出端的个数为 M
则有 $M=2^n$

(a) 2线-4线译码器



在使能信号 $E=1$ 的情况下, 四个输出全部为1。(静态电流比较小, 功耗比较小)

在使能信号有效 ($E=0$) 的情况下, 对应的输出中, 只有1个为0, 其它输出全部为1。

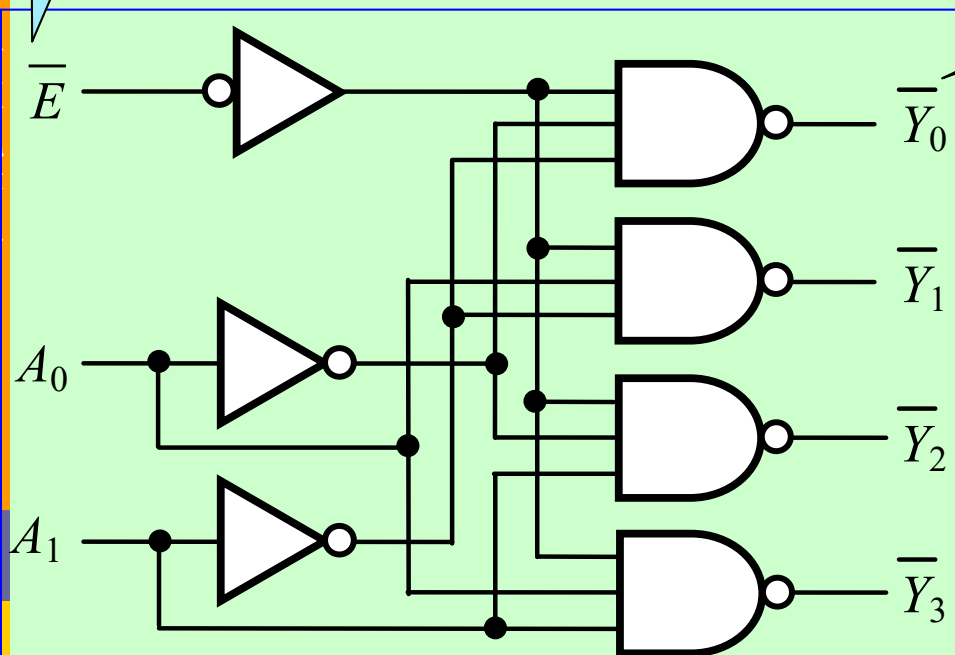
说明: 当电路的输出状态变化时, 静态电流比较大, 漏电流比较大, 功耗比较大。

低电平有效标识

在使能信号有效的情况下，每种输入对应的输出，只有1个为0。

2线 - 4线译码器的逻辑电路(分析)

低电平有效标识



功能表

输入			输出			
\overline{E}	A_1	A_0	\overline{Y}_0	\overline{Y}_1	\overline{Y}_2	\overline{Y}_3
1	×	×	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0

根据真值表中输出F=1的情况
能够写出F的最小项表达式。

$$\overline{Y}_0 = \overline{\overline{E} \overline{A_1} \overline{A_0}}$$

$$\overline{Y}_1 = \overline{\overline{E} \overline{A_1} A_0}$$

$$\overline{Y}_2 = \overline{\overline{E} A_1 \overline{A_0}}$$

$$\overline{Y}_3 = \overline{\overline{E} A_1 A_0}$$

根据真值表中输出F=0的情况
能够写出 \overline{F} 的最小项表达式。

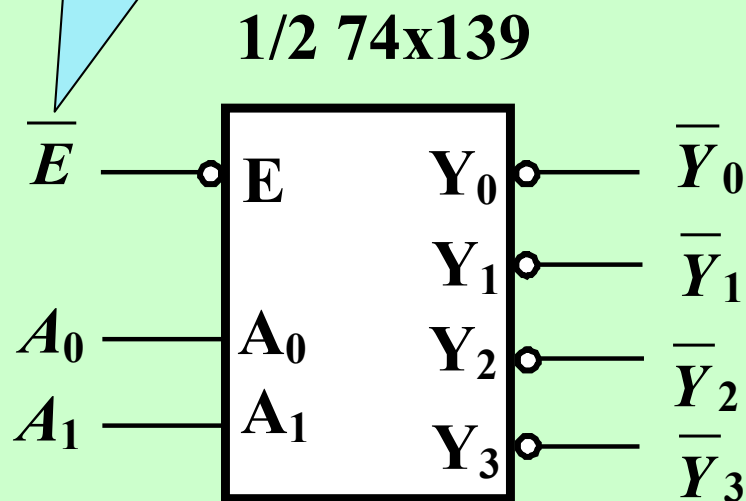
不考虑使能信号E的情况下，译码器的每个输出是一个最小项的非。

(a) 2线-4线译码器 (74HC139芯片)

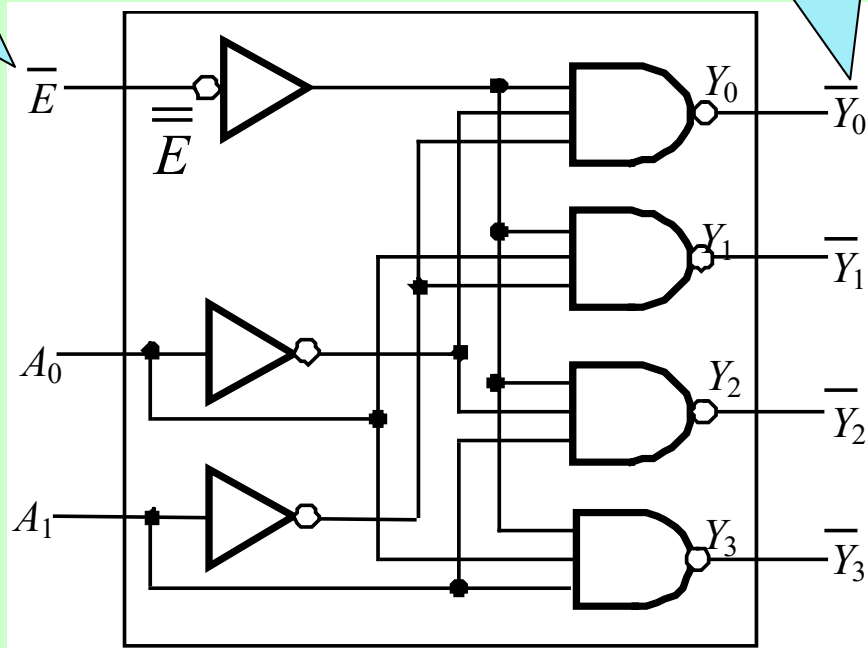
逻辑符号框外部的符号，表示外部输入或输出信号名称，字母上面的“—”号说明该输入或输出是低电平有效。在推导表达式的过程中，表示低有效的输入或输出变量(如)上面的“—”号只是一个提示符，不是“非”，不能参与运算。

低电平有效标识

低电平有效标识



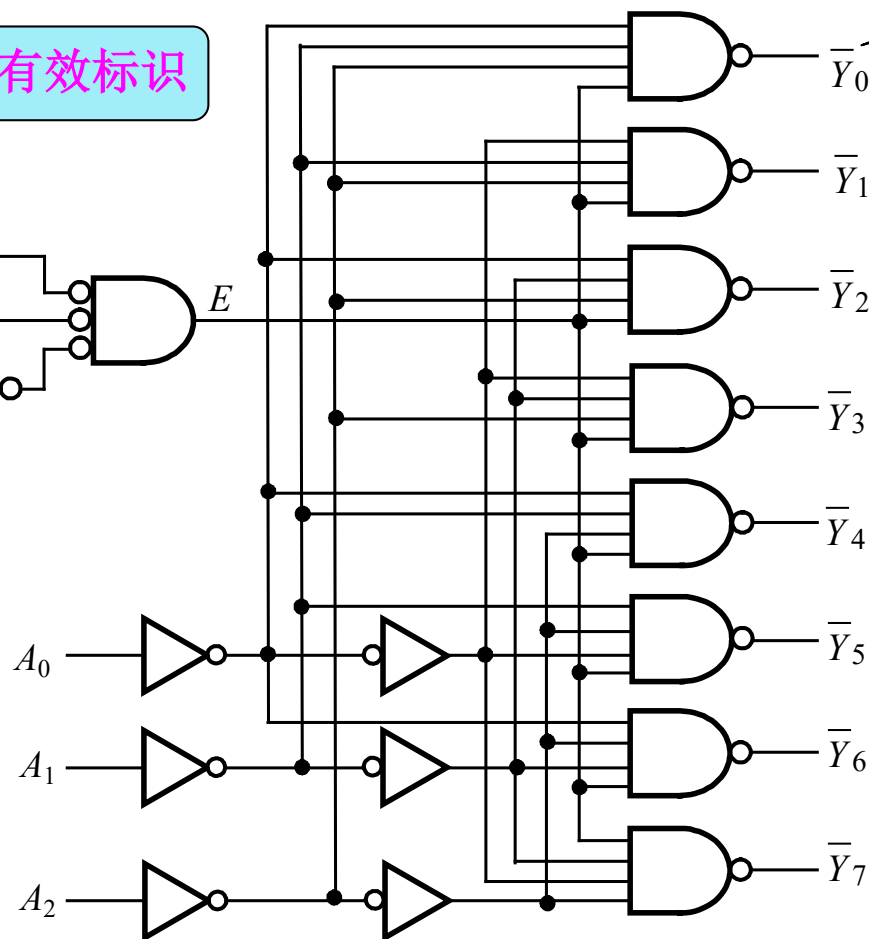
低电平有效标识



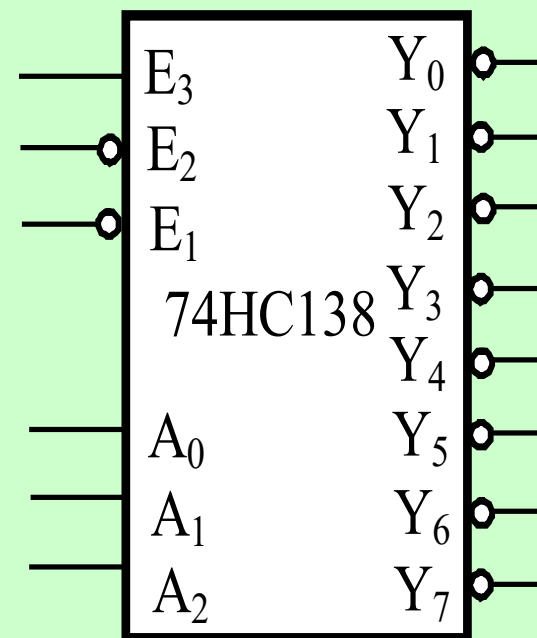
(b) 3线-8线译码器 (74HC138芯片)

低电平有效标识

低电平有效标识



逻辑图



逻辑符号

3线-8线译码器（74HC138芯片）功能表

低电平有效标识

低电平有效标识

输			入			输 出							
E_3	$\overline{E_2}$	$\overline{E_1}$	A_2	A_1	A_0	$\overline{Y_0}$	$\overline{Y_1}$	$\overline{Y_2}$	$\overline{Y_3}$	$\overline{Y_4}$	$\overline{Y_5}$	$\overline{Y_6}$	$\overline{Y_7}$
×	1	×	×	×	×								
×	X	1	×	×	×								
0	×	×	×	×	×								
1	0	0	0	0	0								
1	0	0	0	0	1								
1	0	0	0	1	0								
1	0	0	0	1	1								
1	0	0	1	0	0								
1	0	0	1	0	1								
1	0	0	1	1	0								
1	0	0	1	1	1								

根据真值表中输出F=1的情况能够写出F的最小项表达式。

$$\overline{Y_0} = \overline{A_2} \cdot \overline{A_1} \cdot \overline{A_0};$$

根据真值表中输出F=0的情况能够写出F的最小项表达式。

$$\overline{Y_0} = \overline{A_2} \cdot \overline{A_1} \cdot \overline{A_0}; \quad \overline{Y_1} = \overline{A_2} \cdot \overline{A_1} \cdot A_0; \quad \overline{Y_2} = \overline{A_2} \cdot A_1 \cdot \overline{A_0}; \quad \overline{Y_3} = \overline{A_2} \cdot A_1 \cdot A_0;$$

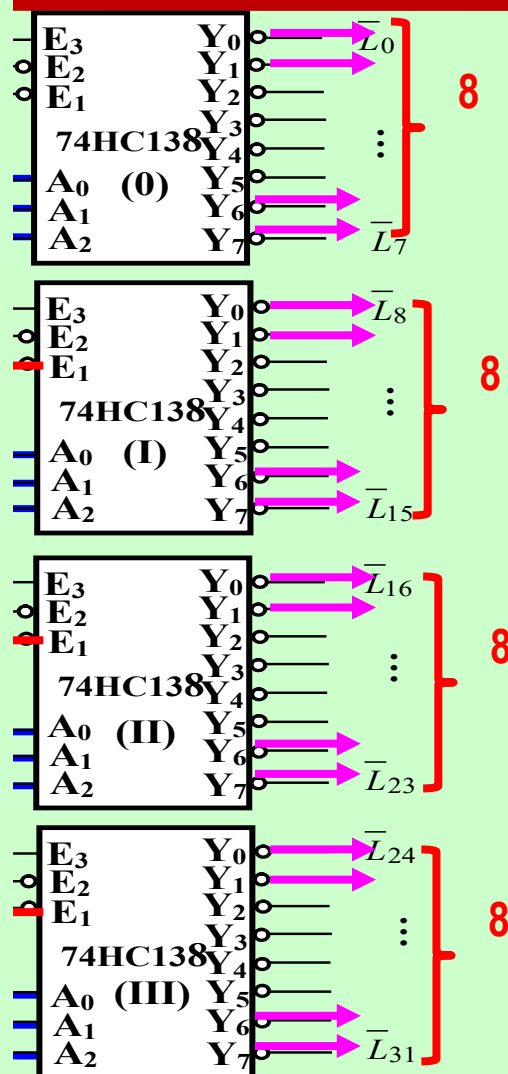
$$\overline{Y_4} = A_2 \cdot \overline{A_1} \cdot \overline{A_0}; \quad \overline{Y_5} = A_2 \cdot \overline{A_1} \cdot A_0; \quad \overline{Y_6} = A_2 \cdot A_1 \cdot \overline{A_0}; \quad \overline{Y_7} = A_2 \cdot A_1 \cdot A_0;$$

输 入						输 出							
E_3	$\overline{E_2}$	$\overline{E_1}$	A_2	A_1	A_0	$\overline{Y_0}$	$\overline{Y_1}$	$\overline{Y_2}$	$\overline{Y_3}$	$\overline{Y_4}$	$\overline{Y_5}$	$\overline{Y_6}$	$\overline{Y_7}$
×	1	×	×	×	×	1	1	1	1	1	1	1	1
×	0	1	×	×	×	1	1	1	1	1	1	1	1
0	×	×	×	×	×	1	1	1	1	1	1	1	1
1	0	0	0	0	0	0	1	1	1	1	1	1	1
1	0	0	0	0	1	1	0	1	1	1	1	1	1
1	0	0	0	1	0	1	1	0	1	1	1	1	1
1	0	0	0	1	1	1	1	1	0	1	1	1	1
1	0	0	1	0	0	1	1	1	1	0	1	1	1
1	0	0	1	0	1	1	1	1	1	1	0	1	1
1	0	0	1	1	0	1	1	1	1	1	1	0	1
1	0	0	1	1	1	1	1	1	1	1	1	1	0

不考虑使能信号E的情况下，译码器的每个输出是一个最小项的非。

1、译码器的扩展

用74HC139和74HC138构成5线-32线译码器



74HC139是2线-4线译码器

74HC138是3线-8线译码器

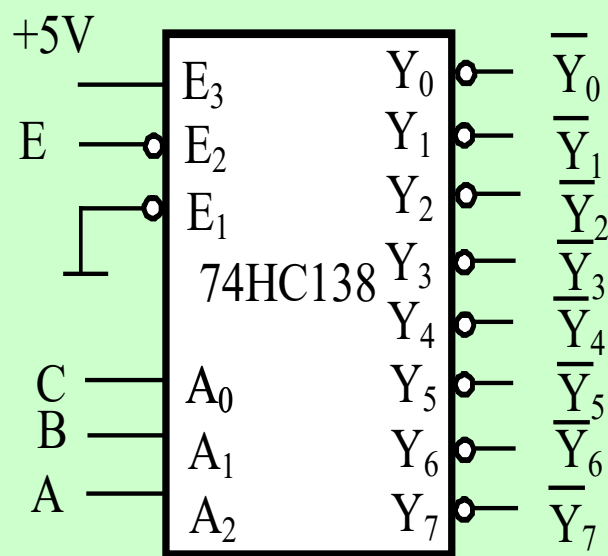
5线中的3线分给138芯片，
另外2线分给139芯片。

32线的输出可以看成
是由4个8线的输出组成。

B4	B3	B2	B1	B0	L0	L1	L2	L31
0	0	0	0	0	0	1	1		1
0	0	0	0	1	1	0	1	1

2、用译码器实现逻辑函数。

当 $E_3=1$ ， $E_2=E_1=0$ 时，如果 A_2 、 A_1 、 A_0 分别连接 A 、 B 、 C 三路信号。则：



$$\overline{Y}_0 = \overline{A_2} \cdot \overline{A_1} \cdot \overline{A_0} = \overline{A} \cdot \overline{B} \cdot \overline{C} = \overline{m_0}$$

$$\overline{Y}_1 = \overline{A_2} \cdot \overline{A_1} \cdot A_0 = \overline{m_1}$$

$$\overline{Y}_2 = \overline{A_2} \cdot A_1 \cdot \overline{A_0} = \overline{m_2}$$

⋮

⋮

⋮

$$\overline{Y}_7 = A_2 \cdot A_1 \cdot A_0 = \overline{m_7}$$

3线-8线译码器的 $\overline{Y}_0 \sim \overline{Y}_7$ 是三变量最小项的非。

基于这一点用该器件能够方便地实现三变量逻辑函数。

例：用一片74HC138和与非门实现函数

$$L = \overline{A}\overline{C} + AB$$

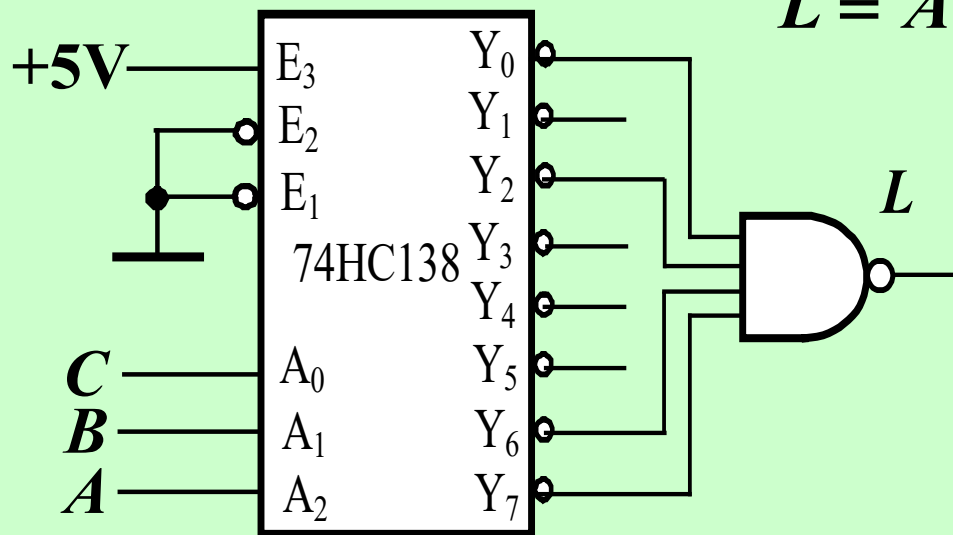
解答：首先将函数式变换为最小项之和的形式

$$L = \overline{A}\overline{B}\overline{C} + \overline{A}B\overline{C} + AB\overline{C} + ABC$$

$$= m_0 + m_2 + m_6 + m_7$$

$$= \overline{m_0} \cdot \overline{m_2} \cdot \overline{m_6} \cdot \overline{m_7}$$

$$= \overline{Y_0} \cdot \overline{Y_2} \cdot \overline{Y_6} \cdot \overline{Y_7}$$

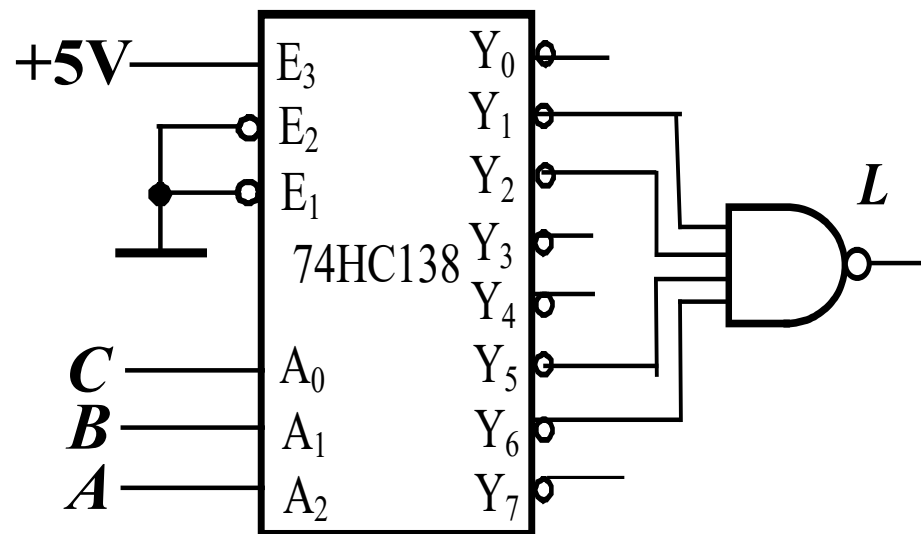


在译码器的输出端加一个与非门，即可实现给定的组合逻辑函数。

下图所示的电路，输出L对应的表达式，正确的是（ ）

- A $L = \overline{m_1} + \overline{m_2} + \overline{m_5} + \overline{m_6}$
- B $L = m_1 + m_2 + m_5 + m_6$**
- C $L = \overline{m_1 \cdot m_2 \cdot m_5 \cdot m_6}$
- D $L = \overline{m_1 + m_2 + m_5 + m_6}$

提交



(2) 二-十进制译码器的真值表

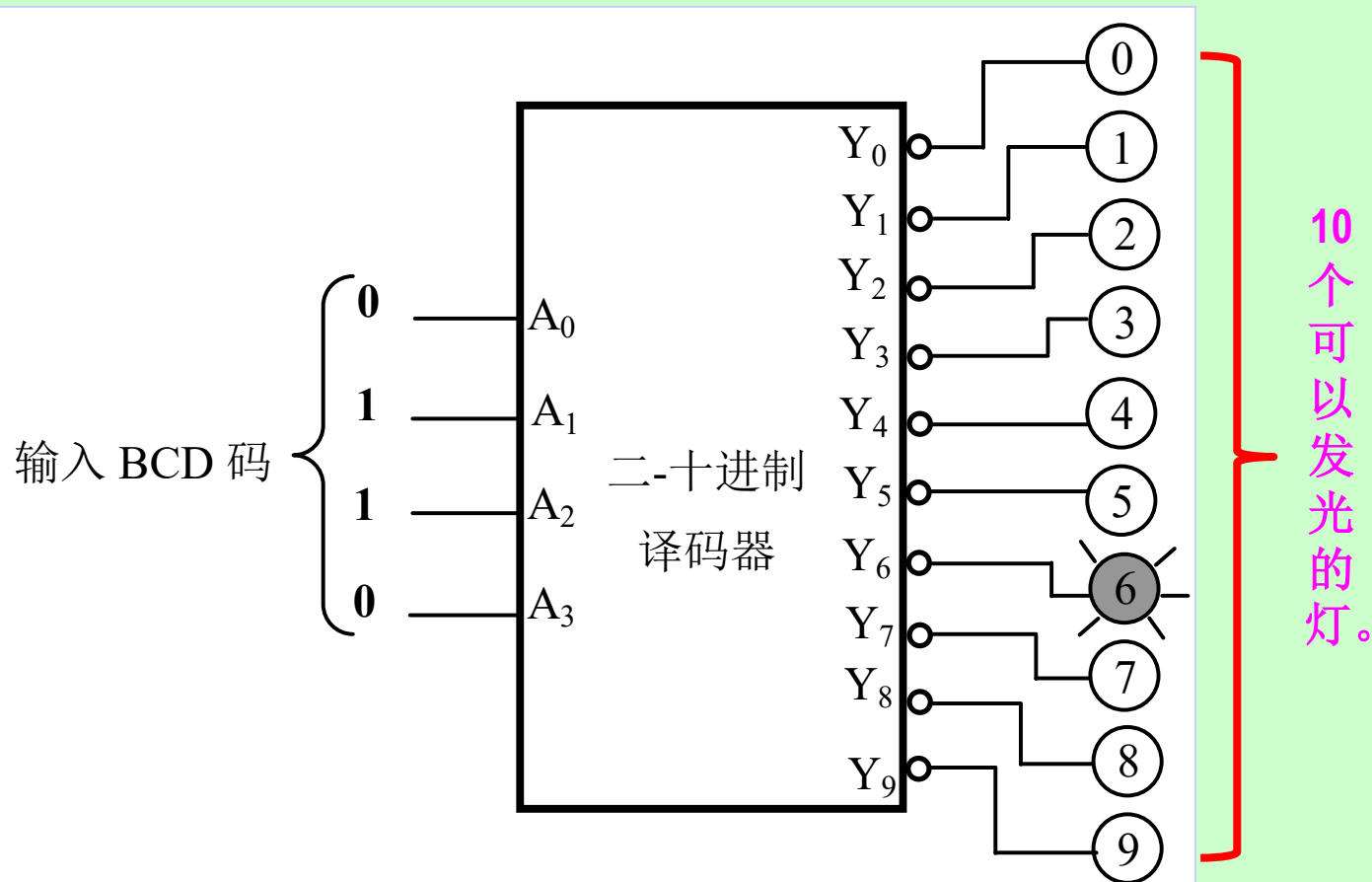
对于BCD代码以外的伪码（1010~1111这6个代码） $Y_0 \sim Y_9$ 均为高电平。

0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
1	0	0	0	1	1	0	1	1	1	1	1	1	1	1
2	0	0	1	0	1	1	0	1	1	1	1	1	1	1
3	0	0	1	1	1	1	1	0	1	1	1	1	1	1
4	0	1	0	0	1	1	1	1	0	1	1	1	1	1
5	0	1	0	1	1	1	1	1	1	0	1	1	1	1
6	0	1	1	0	1	1	1	1	1	1	0	1	1	1
7	0	1	1	1	1	1	1	1	1	1	1	0	1	1
8	1	0	0	0	1	1	1	1	1	1	1	1	0	1
9	1	0	0	1	1	1	1	1	1	1	1	1	1	0

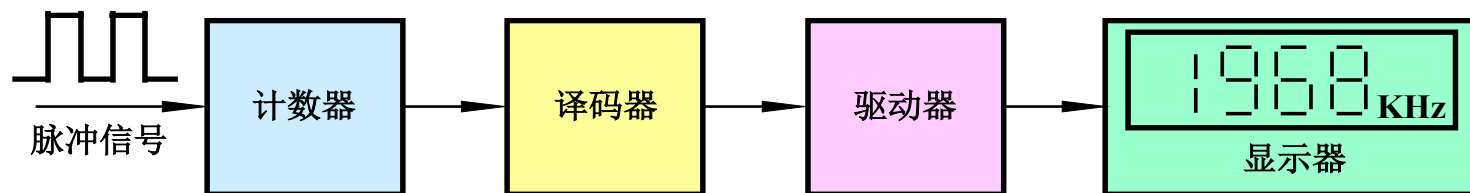
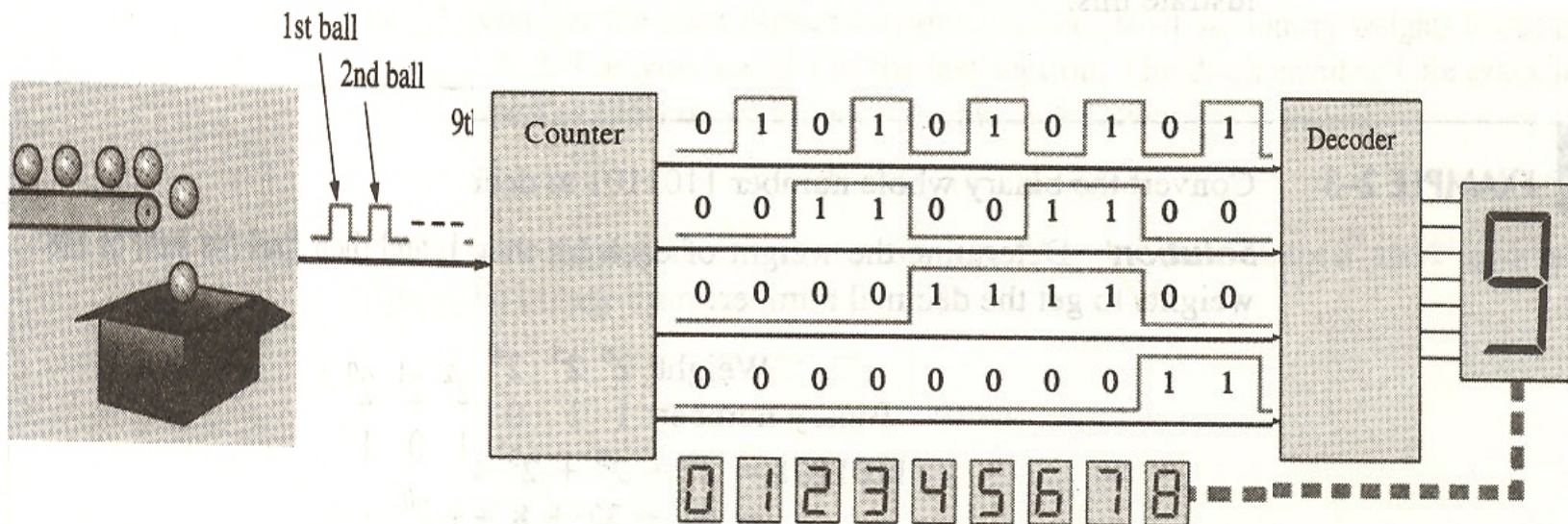
每种输入，对应的输出中，只有1位输出为0，其它位输出全部为1。

二-十进制译码器

功能：将8421BCD码译成为10个状态输出。



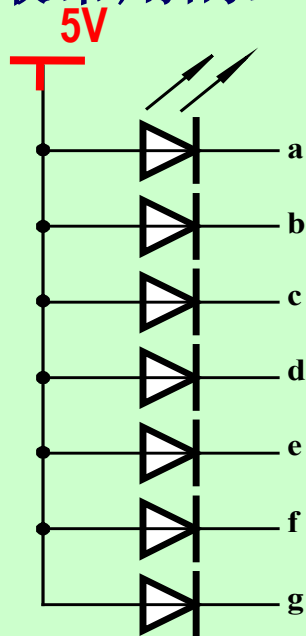
(3) 显示译码器



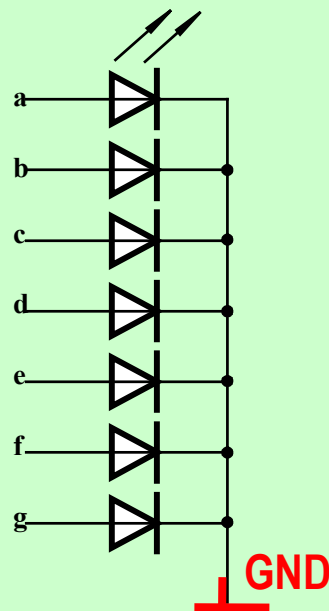
1. 七段显示译码器

The BCD-to-7-Segment Decoder

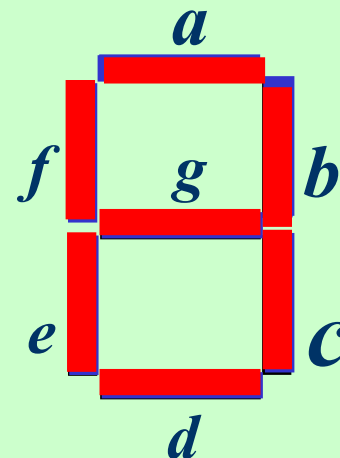
(1) 最常用的显示器有：半导体发光二极管和液晶显示器。



共阳极显示器



共阴极显示器



显示器分段布局图

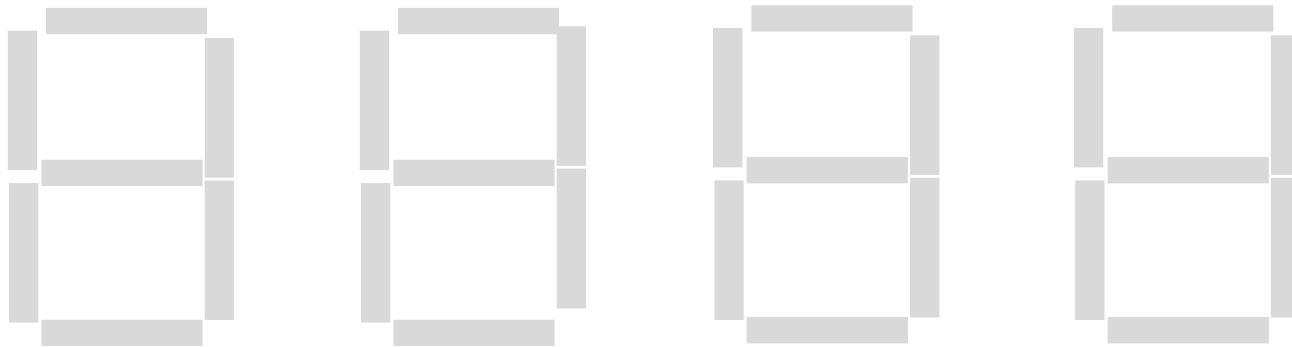
当共阳极接0V时，共阳极显示器不亮。

当共阴极接5V时，共阴极显示器不亮。

1. 七段显示译码器

The BCD-to-7-Segment Decoder

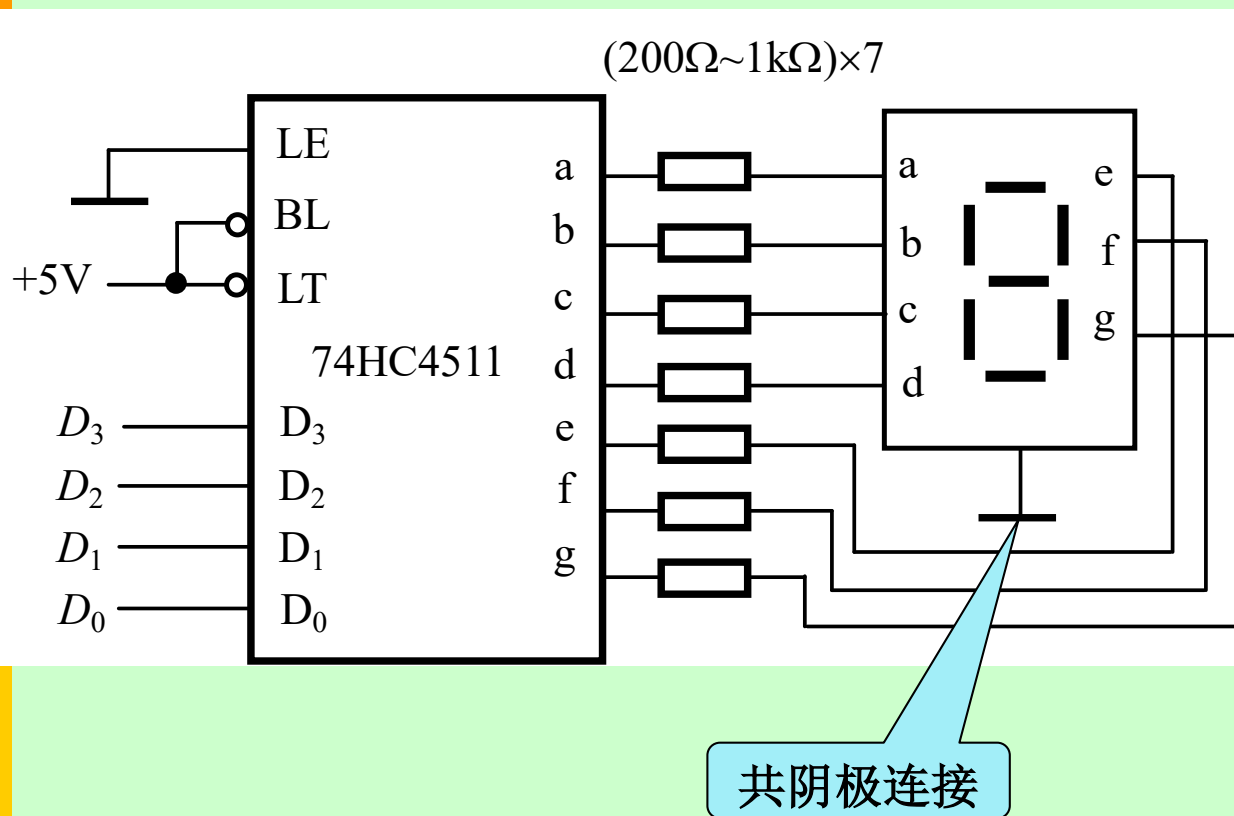
(1) 最常用的显示器有：半导体发光二极管和液晶显示器。



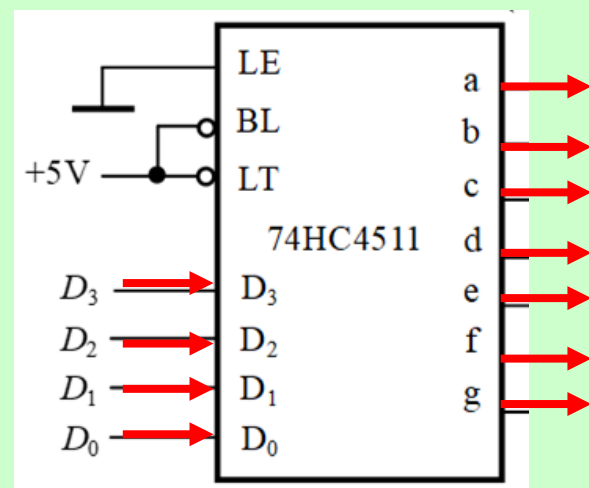
常用的集成七段显示译码器

-----CMOS七段显示译码器**74HC4511**

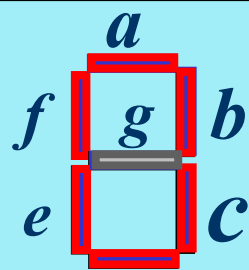
显示译码器与显示器的连接方式



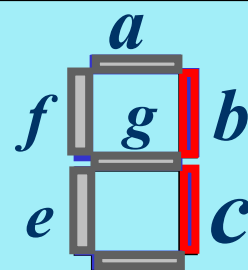
74HC4511输入与输出



CMOS七段显示译码器74HC4511功能表



显示“0”字形



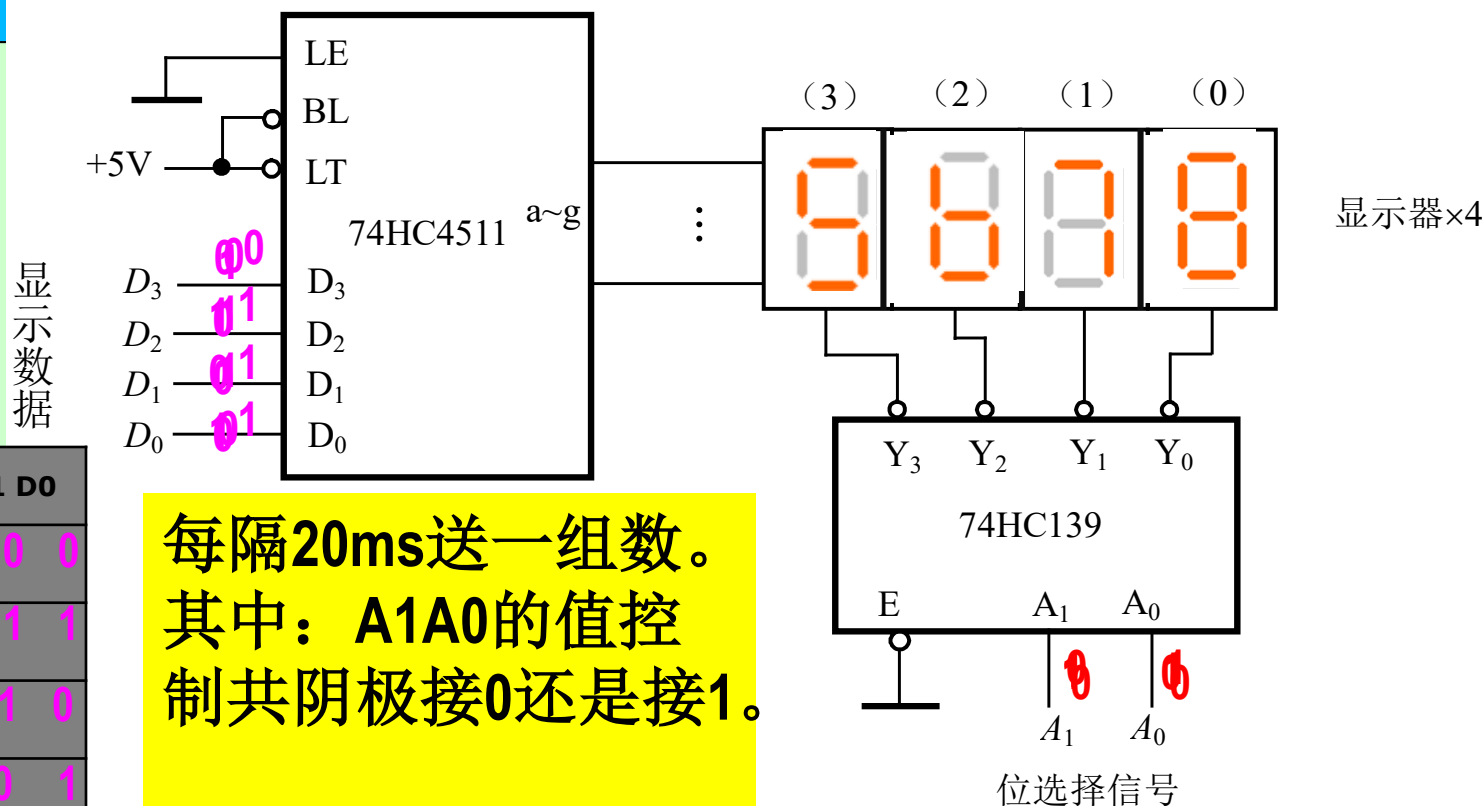
显示“1”字形

十进制或功能	输入							输出							字形
	LE	\overline{BL}	\overline{LT}	D_3	D_2	D_1	D_0	a	b	c	d	e	f	g	
0	0	1	1	0	0	0	0								0
1	0	1	1	0	0	0	1								1
2	0	1	1	0	0	1	0								2
3	0	1	1	0	0	1	1								3
4	0	1	1	0	1	0	0								4
5	0	1	1	0	1	0	1								5
6	0	1	1	0	1	1	0								6
7	0	1	1	0	1	1	1								7
8	0	1	1	1	0	0	0								8
9	0	1	1	1	0	0	1								9

计算机学院

数字逻辑电路

例 由译码器、显示译码及4个七段显示器构成的4位动态显示电路如图所示，试分析工作原理。

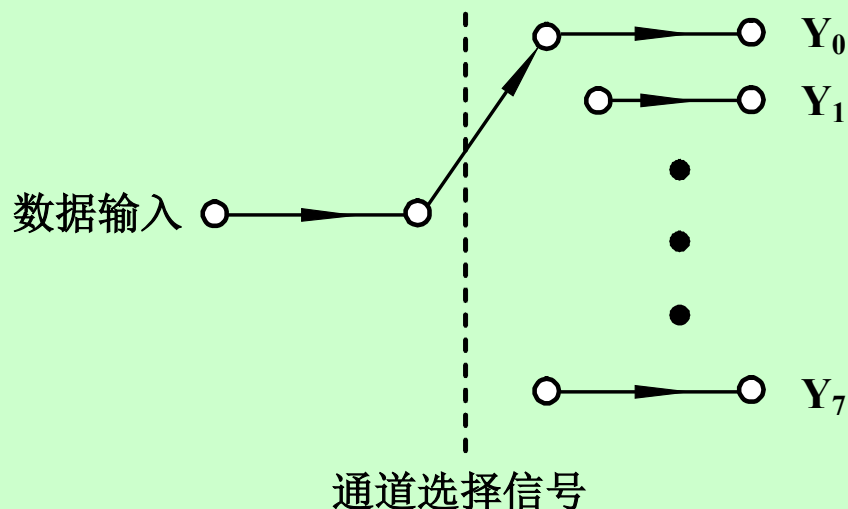


A1	A0	D3	D2	D1	D0
0	0	1	0	0	0
0	1	0	1	1	1
1	0	0	1	1	0
1	1	0	1	0	1

位选择信号A1、A0控制 $\overline{Y_3} \sim \overline{Y_0}$ 依次产生低电平，使4个显示器轮流显示。要显示的数据组依次送到 $D_3D_2D_1D_0$ 分别在4个显示器上显示。利用人的视觉暂留时间，可以看到稳定的数字。

$$25\text{Hz} < f_c < 100\text{Hz}$$

数据分配器示意图

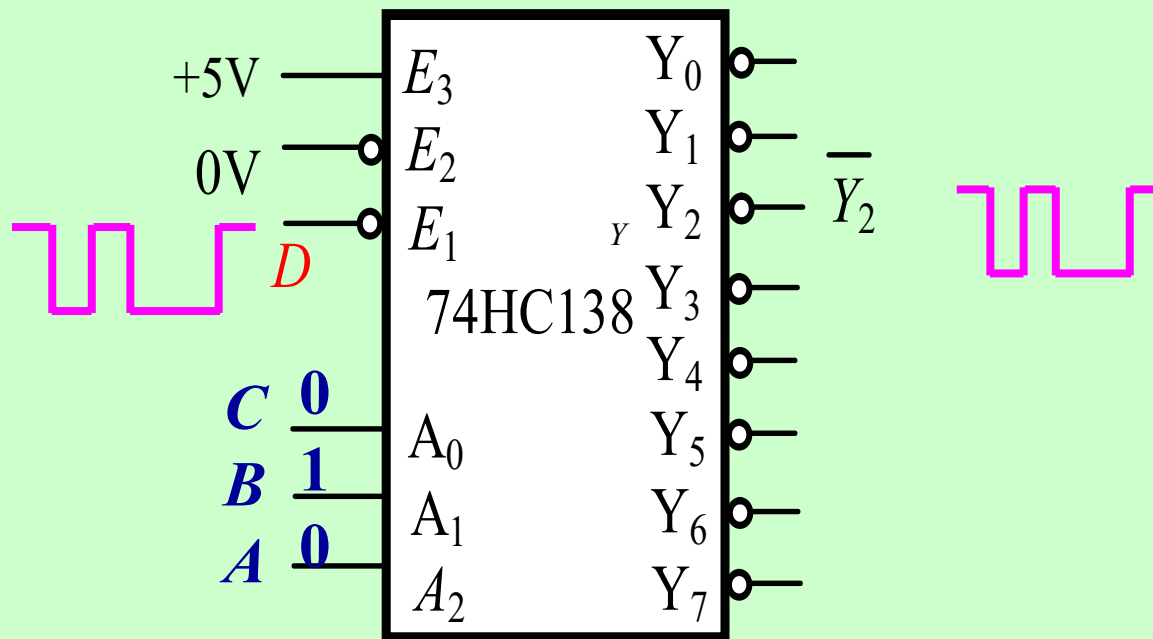


数据分配器：相当于多输出的单刀多掷开关，是将公共数据线上的数据按需要送到不同的通道上去的逻辑电路。

用译码器实现数据分配器

$$\overline{Y_2} = \overline{E_3 E_2 D A B C}$$

当 $ABC = 010$ 时, $\overline{Y_2} = D$



74HC138译码器作为数据分配器时的功能表

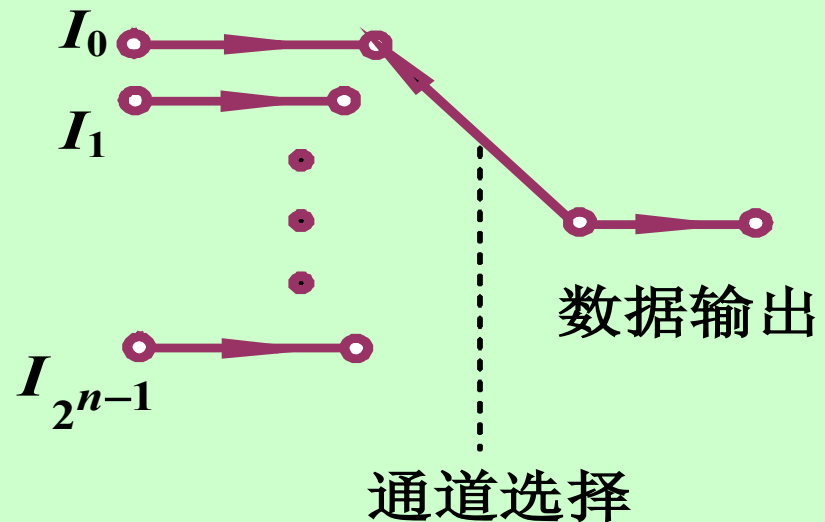
输 入						输 出							
E_3	\overline{E}_2	\overline{E}_1	A_2	A_1	A_0	\overline{Y}_0	\overline{Y}_1	\overline{Y}_2	\overline{Y}_3	\overline{Y}_4	\overline{Y}_5	\overline{Y}_6	\overline{Y}_7
0	0	X	X	X	X	1	1	1	1	1	1	1	1
1	0	D	0	0	0	D	1	1	1	1	1	1	1
1	0	D	0	0	1	1	D	1	1	1	1	1	1
1	0	D	0	1	0	1	1	D	1	1	1	1	1
1	0	D	0	1	1	1	1	1	D	1	1	1	1
1	0	D	1	0	0	1	1	1	1	D	1	1	1
1	0	D	1	0	1	1	1	1	1	1	D	1	1
1	0	D	1	1	0	1	1	1	1	1	1	D	1
1	0	D	1	1	1	1	1	1	1	H	1	1	D

4.4.3 数据选择器 Multiplexers (Data Selectors)

1、数据选择器的定义与功能

数据选择器：能实现**数据选择功能**的逻辑电路。它的作用相当于**多个输入**的**单刀多掷开关**，又称“**多路开关**”。

数据选择的功能：在通道选择信号的作用下，将**多个通道**的数据分时传送到**公共的数据通道**上去的。



2选1数据选择器

简化真值表

完整的真值表

1位地址码
输入端

数据
输入
端

1路数据输出端

逻辑符号

Y=1的情况共有两种情况：

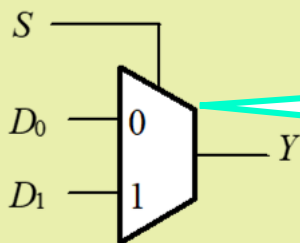
(1) $SD_0=01$

(2) $SD_1=11$

$$Y = \bar{S}D_0 + SD_1$$

选择 输入	输 出
S	Y
0	D_0
1	D_1

输入			输出
S	D_0	D_1	Y
0	1	x	1
0	0	x	0
1	x	1	1
1	x	0	0
...			



4选1数据选择器

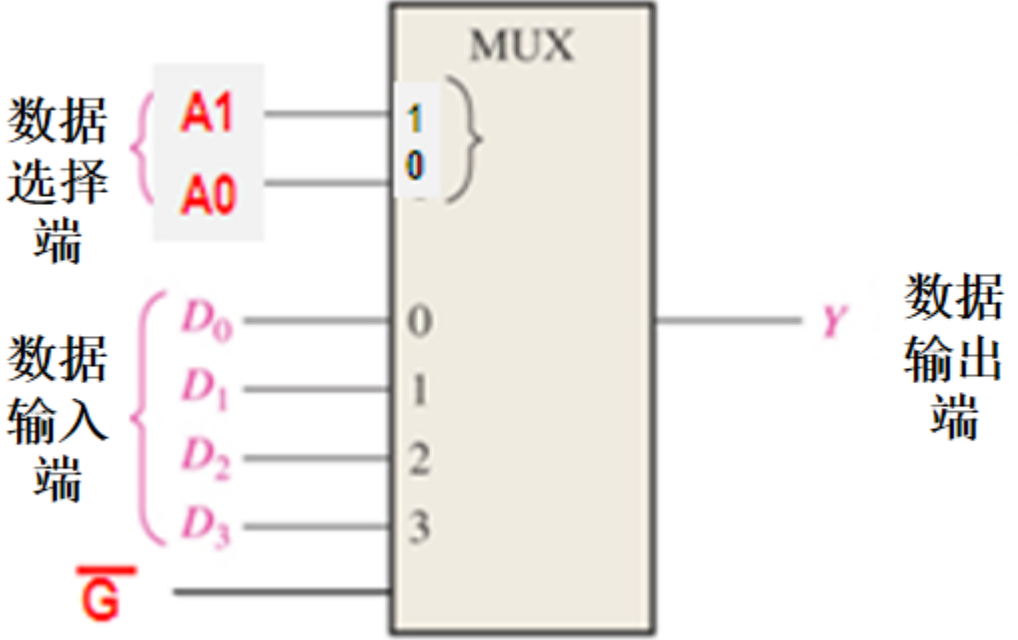
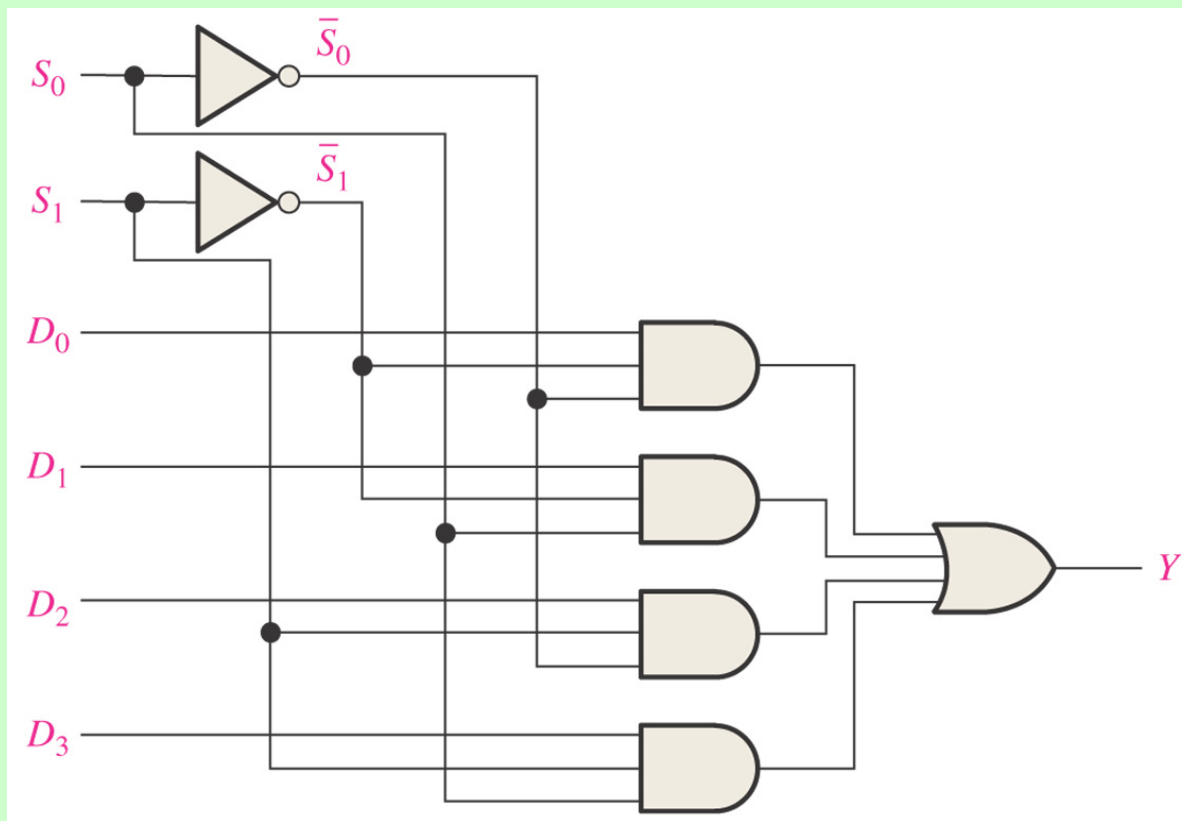


表 4.3.1 4 选 1 数据选择器功能表

输 入				输 出			
G	A_1	A_0	D_3	D_2	D_1	D_0	Y
1	\times	\times	\times	\times	\times	\times	0
0	0	0	\times	\times	\times	0	0
			\times	\times	\times	1	1
	0	1	\times	\times	0	\times	0
			\times	\times	1	\times	1
	1	0	\times	0	\times	\times	0
			\times	1	\times	\times	1
	1	1	0	\times	\times	\times	0
			1	\times	\times	\times	1

4选1数据选择器



$$Y = \overline{S_1}\overline{S_0}D_0 + \overline{S_1}S_0D_1 + S_1\overline{S_0}D_2 + S_1S_0D_3$$

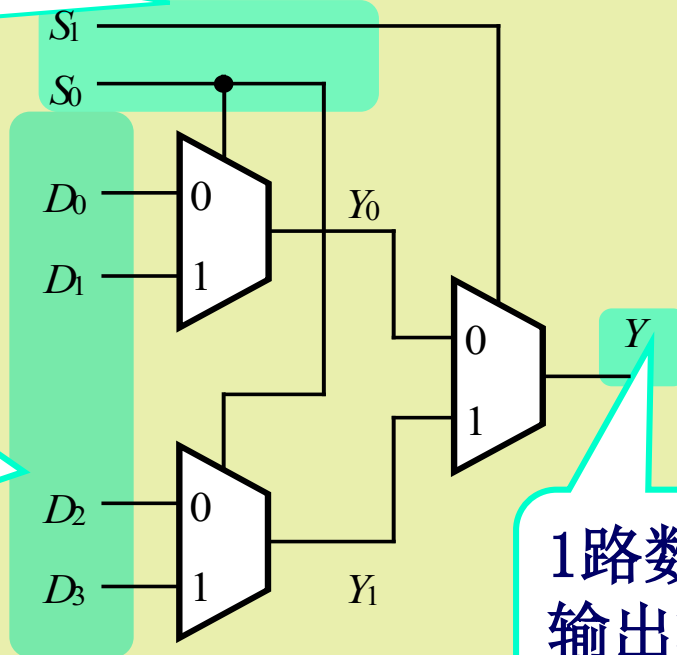
4选1数据选择器

(1) 逻辑电路

由3个2选1数据选择器构成4选1数据选择器。

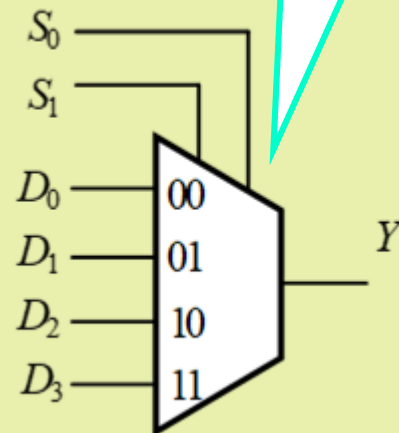
2 位地址
码输入端

数据
输入端

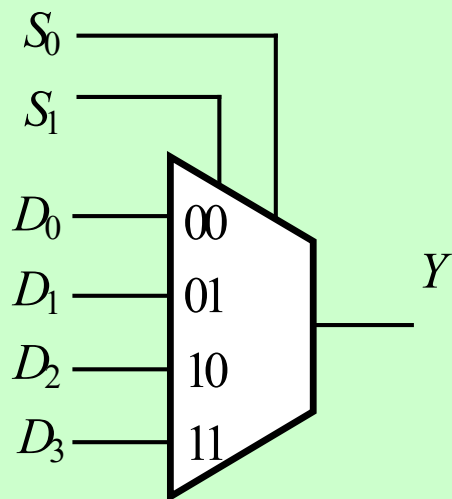


1路数据
输出端

逻辑符号



(2) 工作原理及逻辑功能



简化真值表

选择输入		输出
S_1	S_0	
0	0	
0	1	
1	0	
1	1	

$$Y = \overline{S_1} \overline{S_0} D_0 + \overline{S_1} S_0 D_1 + S_1 \overline{S_0} D_2 + S_1 S_0 D_3$$

$$Y = D_0 m_0 + D_1 m_1 + D_2 m_2 + D_3 m_3$$

这两个公式常用