

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

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。

1、编码器 (Encoder)的定义与分类

编码器的逻辑功能：

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。

如BCD编码器：将10个编码输入信号分别编成10个4位码输出。

如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。

1、编码器 (Encoder)的定义与分类

编码器的分类：普通编码器和优先编码器。

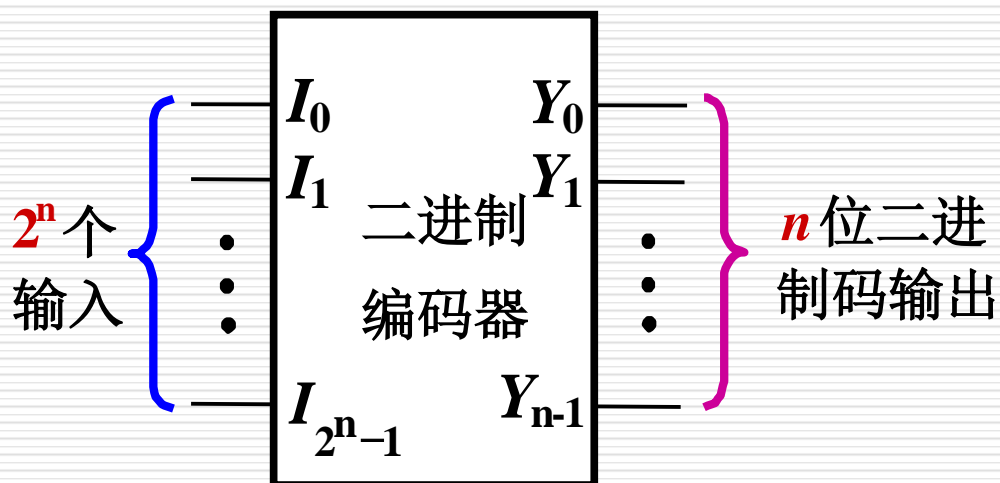
普通编码器：任何时候只允许输入一个有效编码信号，否则输出就会发生混乱。

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。

2、编码器的工作原理

普通二进制编码器

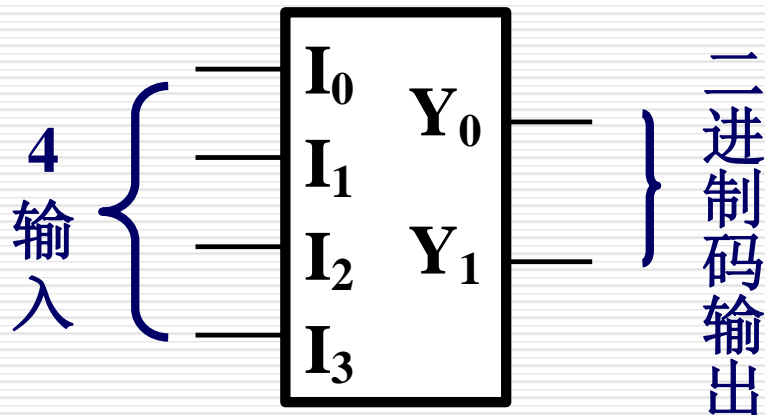
二进制编码器的结构框图



2、编码器的工作原理

(1) 4线—2线普通二进制编码器 (设计)

(a) 逻辑框图



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

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

该表达式是否可以再简化？

(2) 逻辑功能表

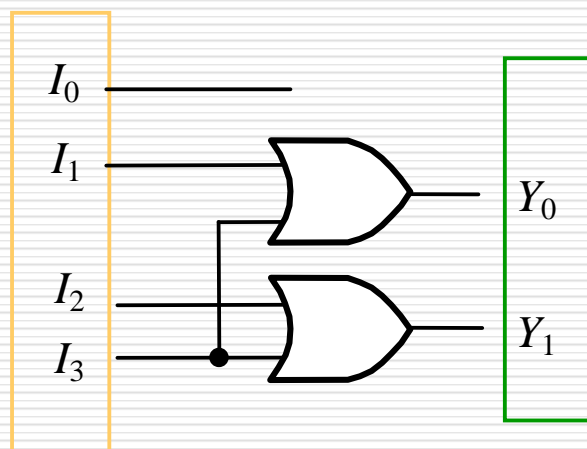
I_0	I_1	I_2	I_3	Y_1	Y_0
1	0	0	0	0	0
0	1	0	0	0	1
0	0	1	0	1	0
0	0	0	1	1	1

编码器的输入为高电平有效。

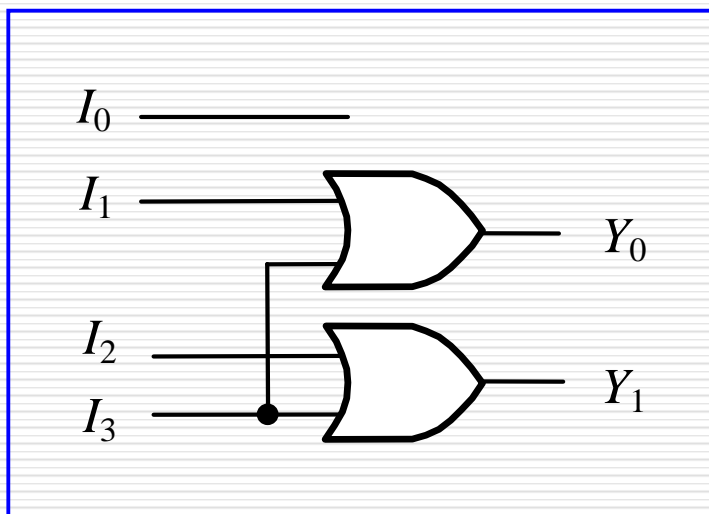
上述是将输入的其它12种组合对应的输出看做0。如果看做无关项，则表达式为

$$Y_1 = I_2 + I_3$$

$$Y_0 = I_1 + I_3$$



若有2个以上的输入为有效信号？



当只有 I_3 为1时，

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

$I_1 = I_2 = 1$, $I_0 = I_3 = 0$ 时，

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

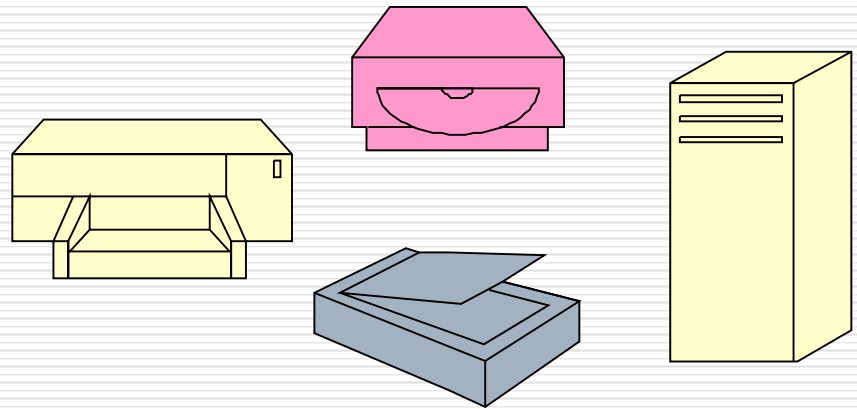
无法输出有效编码。

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

优先编码器的提出：

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，即优先级别。

识别多个编码请求信号的优先级别，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 $I_3 \sim I_0$

输入为编码信号 $I_3 \sim I_0$ 输出为 $Y_1 Y_0$

(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

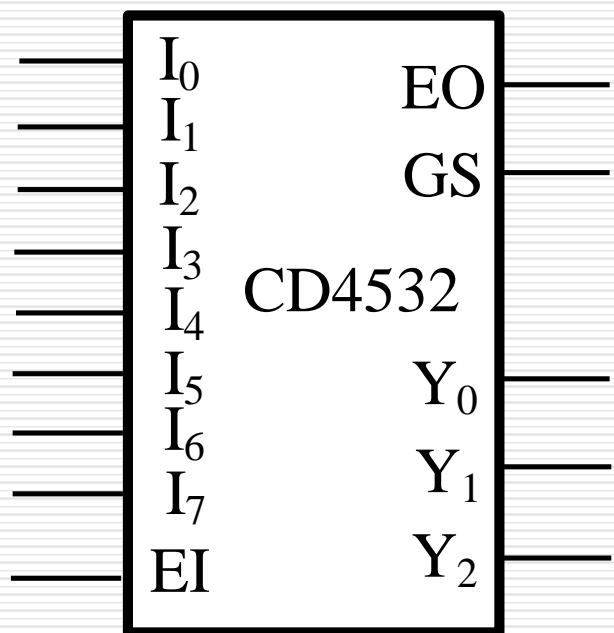
$$Y_1 = I_2 \overline{I_3} + I_3$$

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

(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



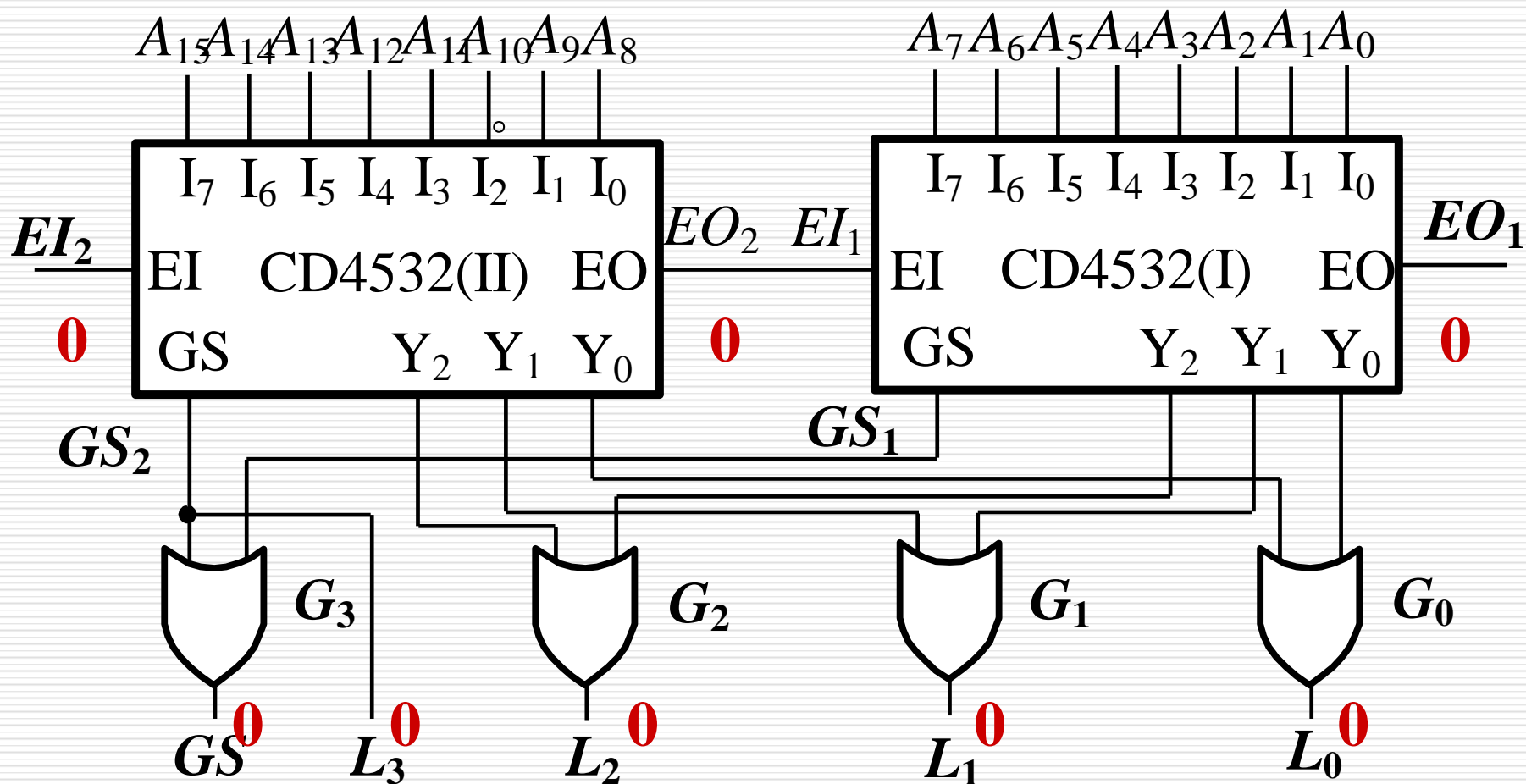
优先编码器CD4532功能表

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>GS</i>	<i>EO</i>
0	×	×	×	×	×	×	×	×	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	1
1	1	×	×	×	×	×	×	×	1	1	1	1	0
1	0	1	×	×	×	×	×	×	1	1	0	1	0
1	0	0	1	×	×	×	×	×	1	0	1	1	0
1	0	0	0	1	×	×	×	×	1	0	0	1	0
1	0	0	0	0	1	×	×	×	0	1	1	1	0
1	0	0	0	0	0	1	×	×	0	1	0	1	0
1	0	0	0	0	0	0	1	×	0	0	1	1	0
1	0	0	0	0	0	0	0	1	0	0	0	1	0

为什么要设计GS、EO输出信号？

用二片**CD4532**构成16线-4线优先编码器,其逻辑图如下图所示,试分析其工作原理。

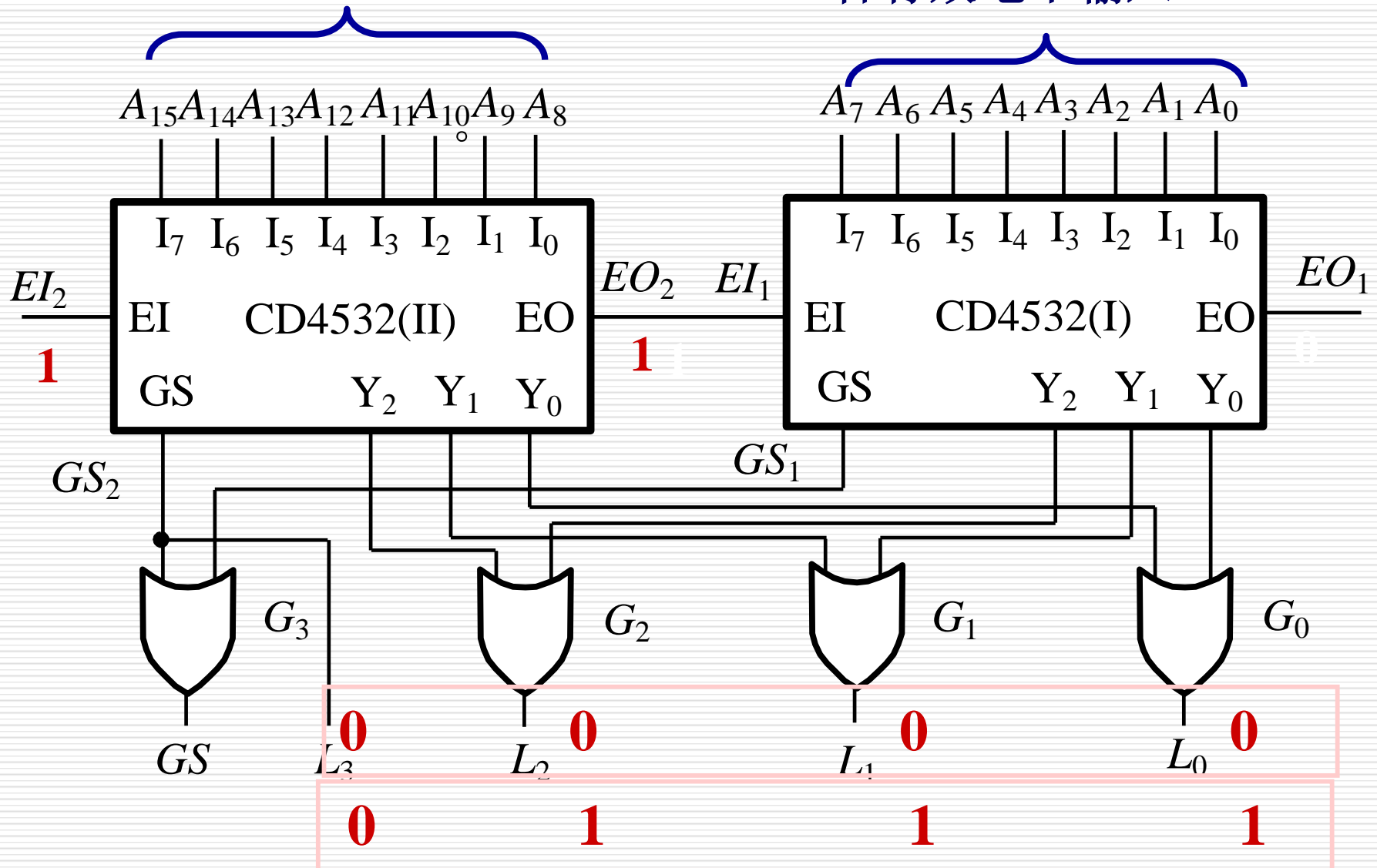
当使能端EI=0时，无编码输出。



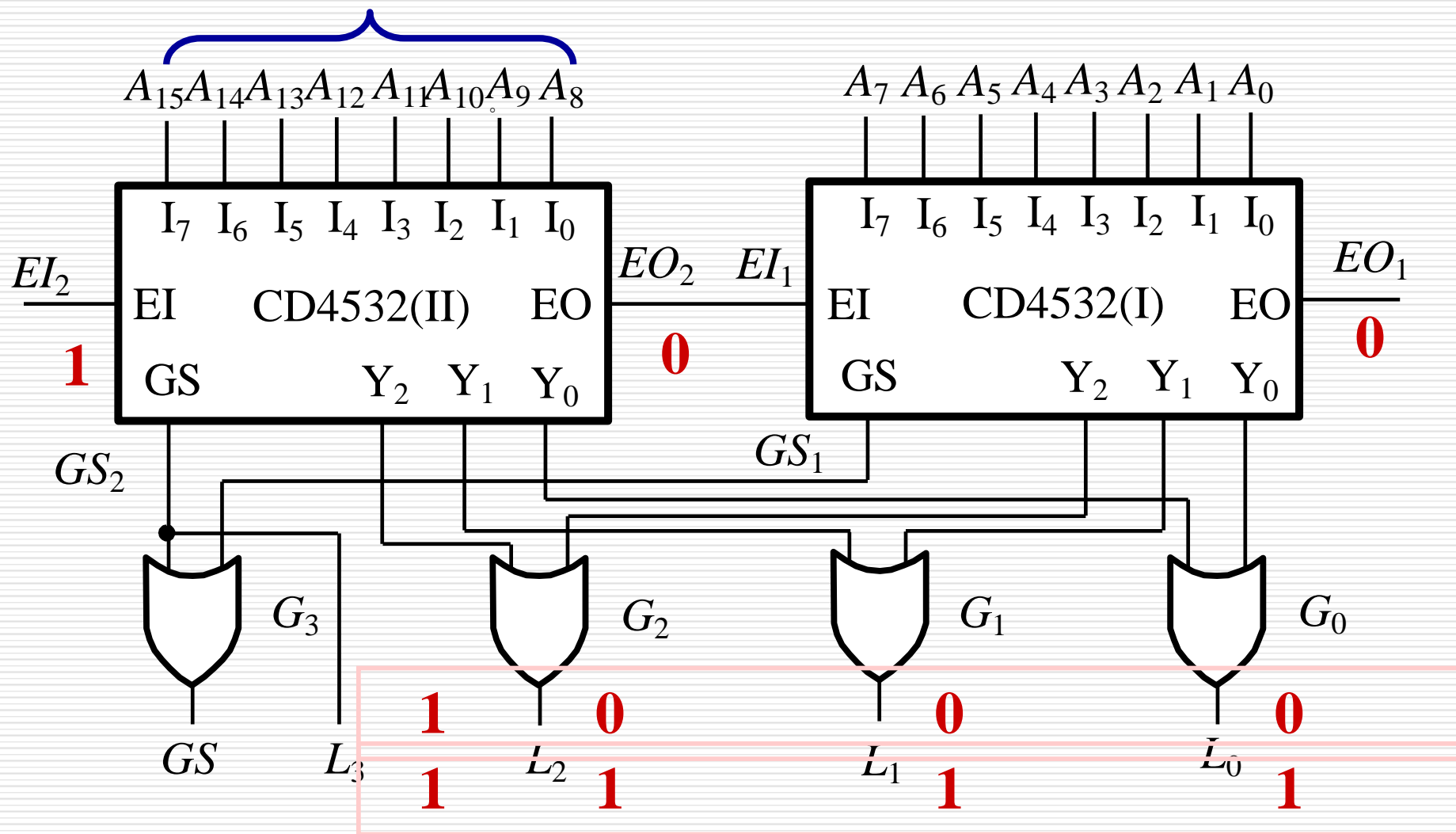
那块芯片的优先级高?

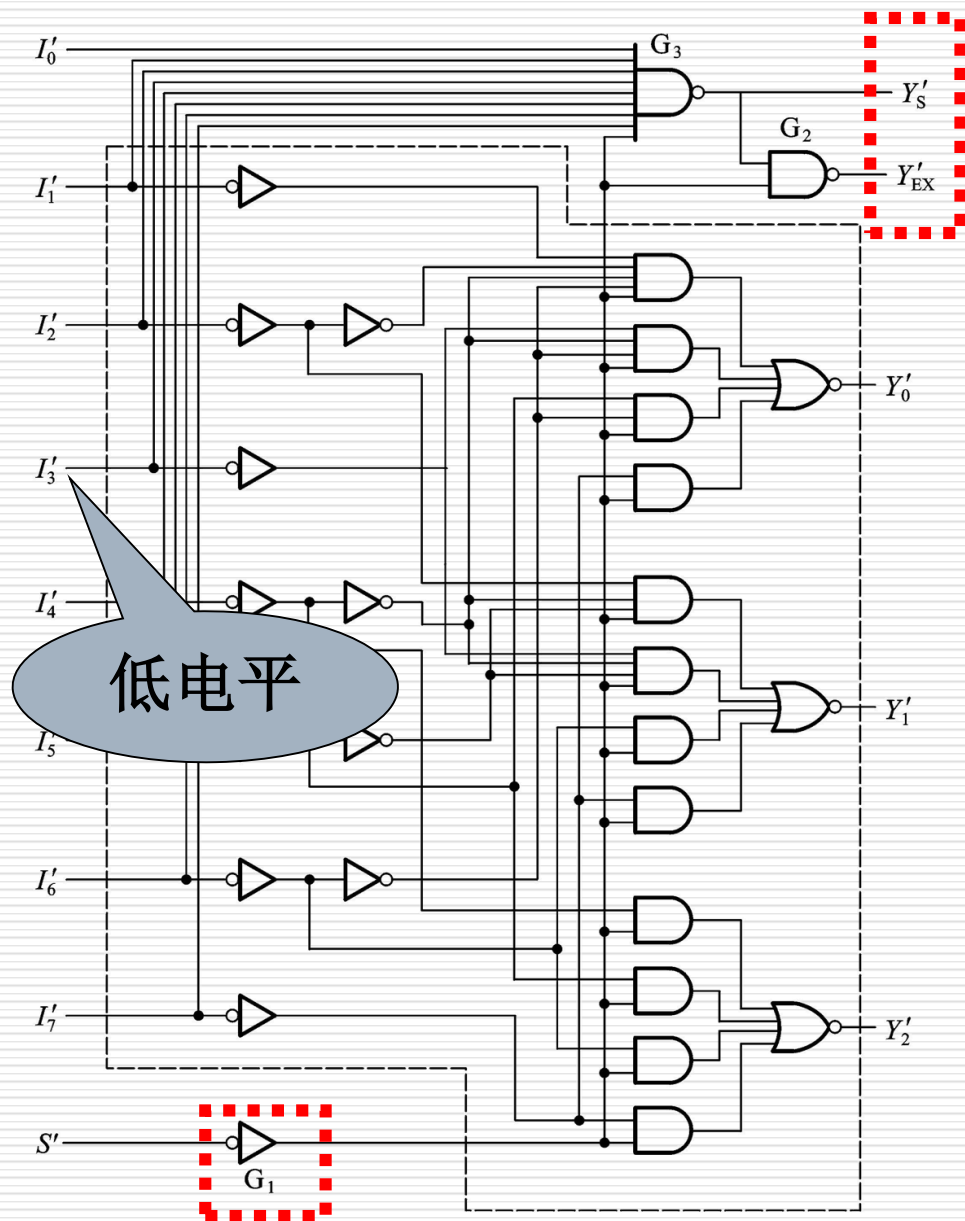
若有效电平输入

若无有效电平输入



若有效电平输入





实例：
74HC148

$$Y_2' = (I_7 + I_6 + I_5 + I_4)'$$

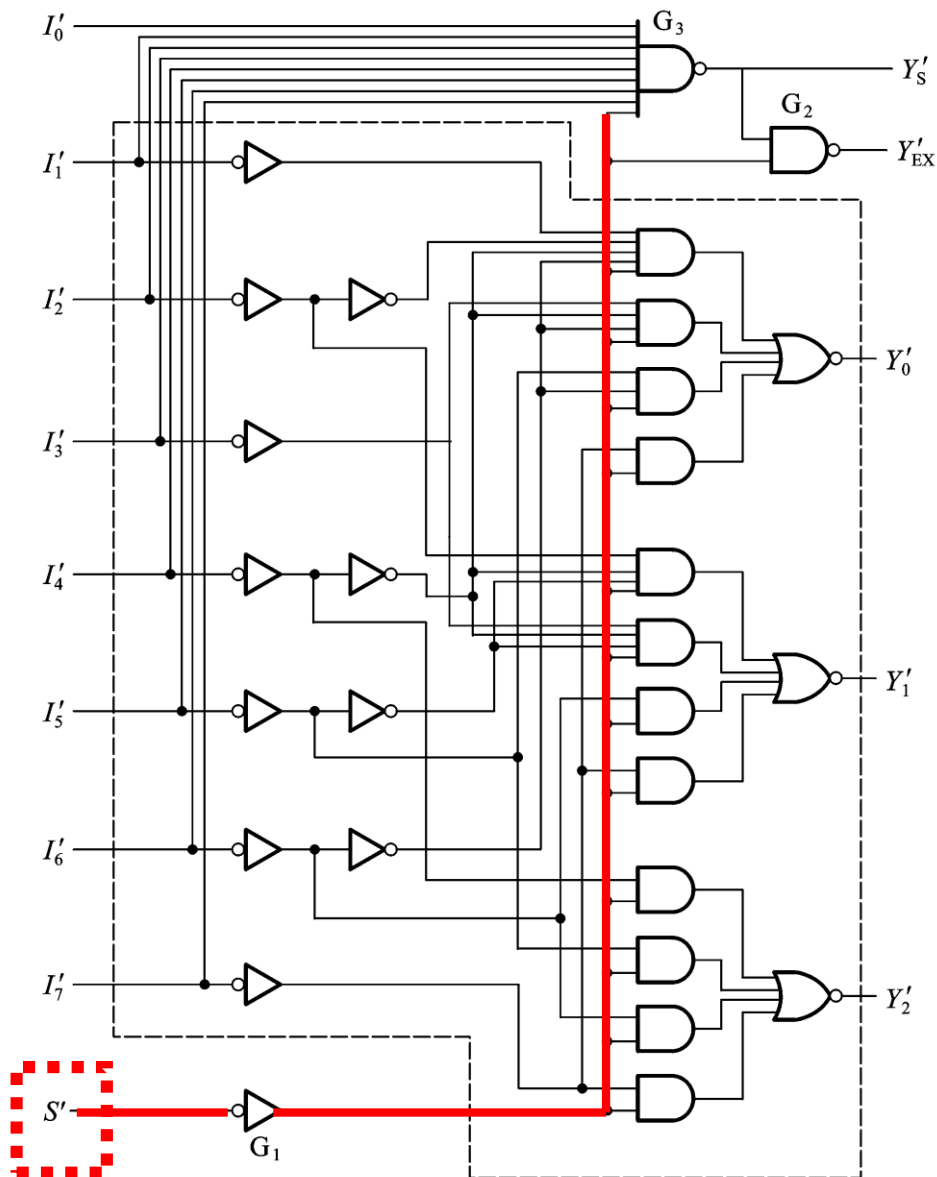
选通信号

$$Y_2' = [(I_7 + I_6 + I_5 + I_4)S]'$$

$$Y_2' = [(I_7 + I_6 + I_5 + I_4)S]'$$

$$Y_1' = [(I_7 + I_6 + I_5 I_4' I_3' + I_2 I_4' I_5')S]'$$

$$Y_0' = [(I_7 + I_6 I_5 + I_3 I_4' I_6' + I_1 I_2 I_4' I_6')S]'$$



选通信号

附加输出信号

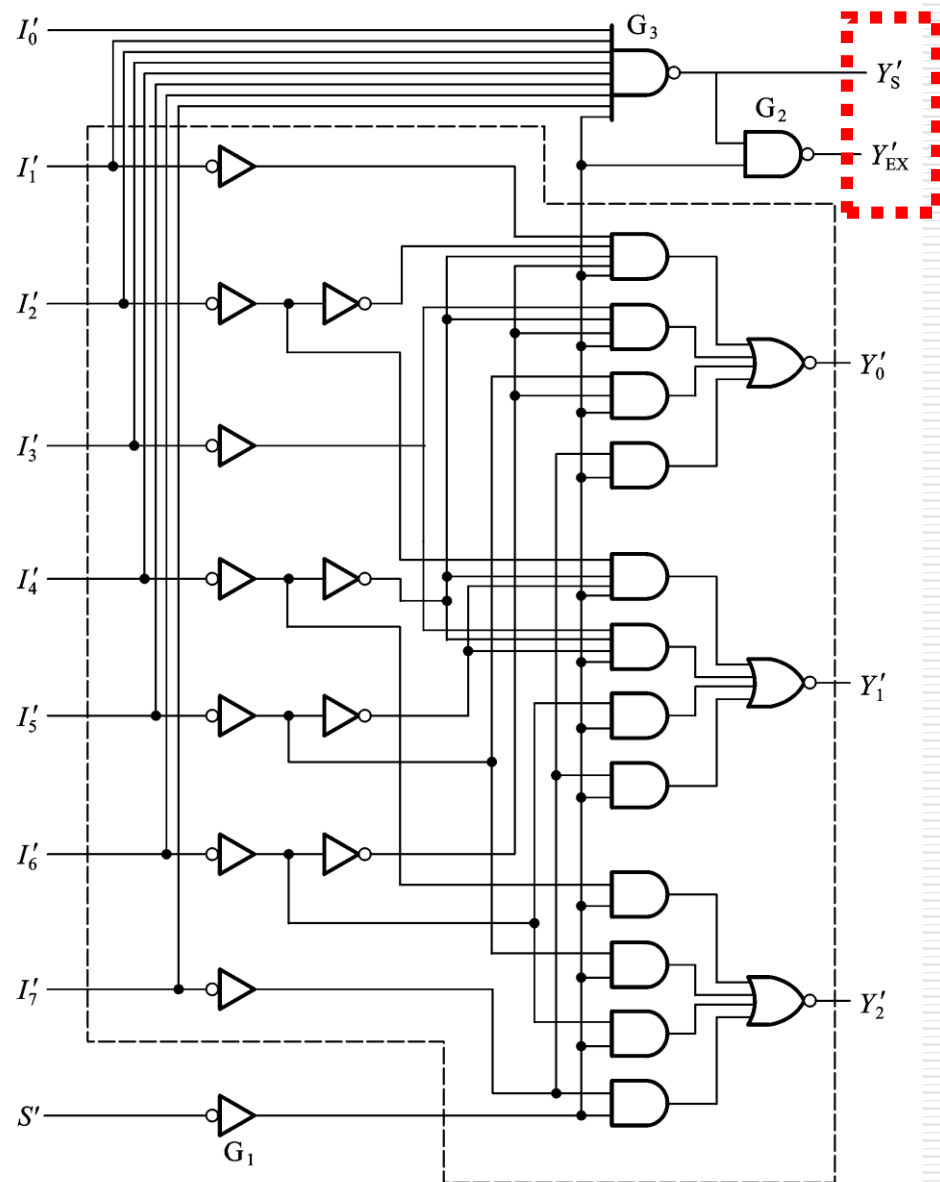
$$Y'_S = (I'_7 I'_6 I'_5 I'_4 I'_3 I'_2 I'_1 I'_0 S)'$$

$$Y'_{EX} = [(I'_7 I'_6 I'_5 I'_4 I'_3 I'_2 I'_1 I'_0 S)' S']$$

$$= [(I_7 + I_6 + I_5 + I_4 + I_3 + I_2 + I_1 + I_0) S']$$

Y'_S	Y'_{EX}	状态
1	1	不工作($S'=1$)
0	1	工作，但无输入
1	0	工作，且有输入
0	0	不可能出现

解决了当输出全为零时，是否有编码输入的问题。



[illegible]

4.4.2 译码器/数据分配器

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：

唯一地址译码器

将一系列代码转换成与之一一对应的有效信号。

常见的唯一地址译码器：

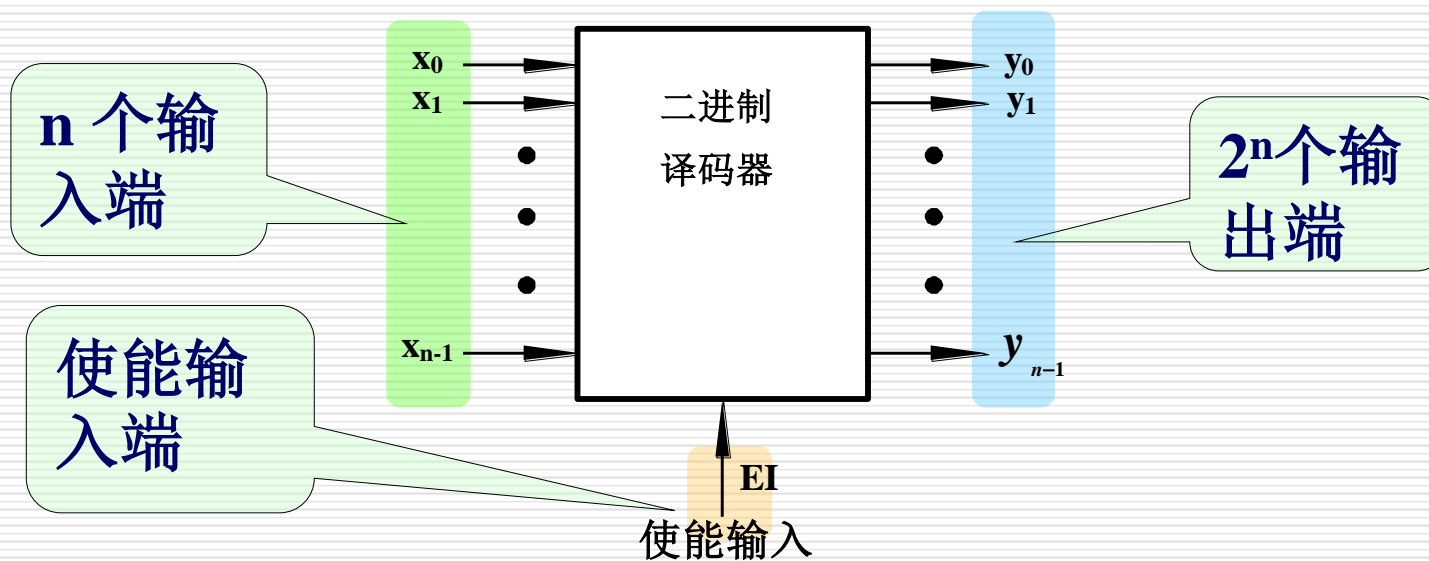
- 二进制译码器
- 二—十进制译码器
- 显示译码器

代码变换器

将一种代码转换成另一种代码。

2. 典型译码器电路及应用

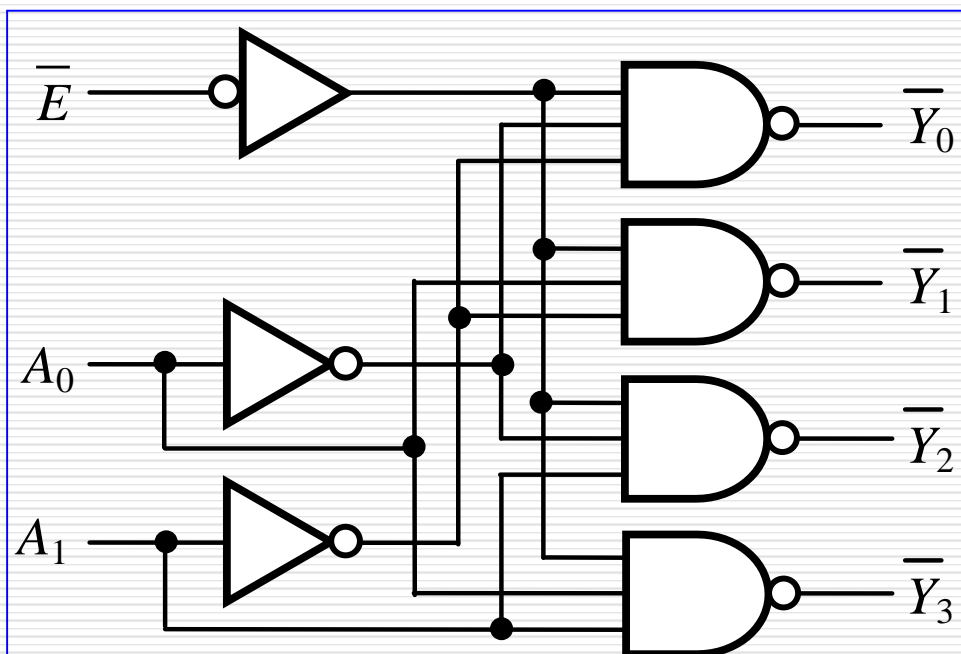
(1) 二进制译码器



设输入端的个数为 n ，输出端的个数为 M
则有

$$M=2^n$$

2线 - 4线译码器的逻辑电路(分析)



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

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

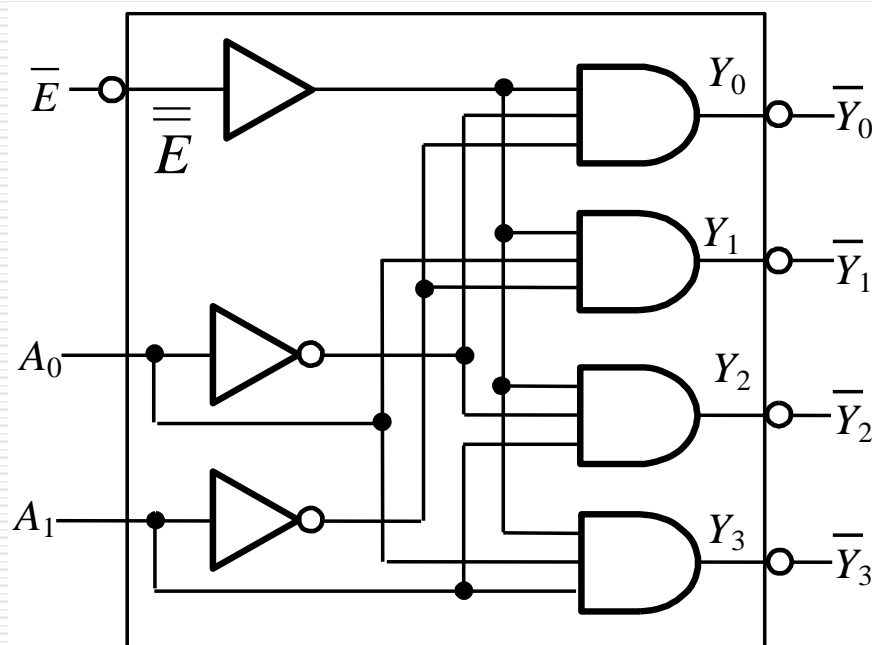
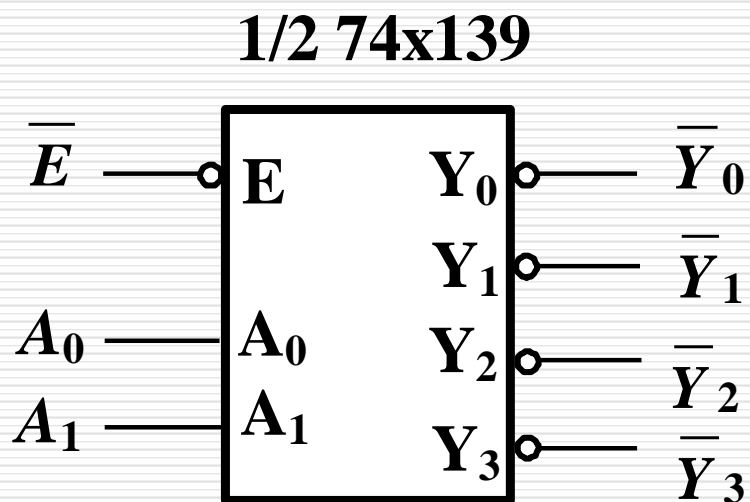
$$\overline{Y}_2 = \overline{\overline{\overline{E}} \overline{\overline{A_1}} \overline{\overline{A_0}}}$$

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

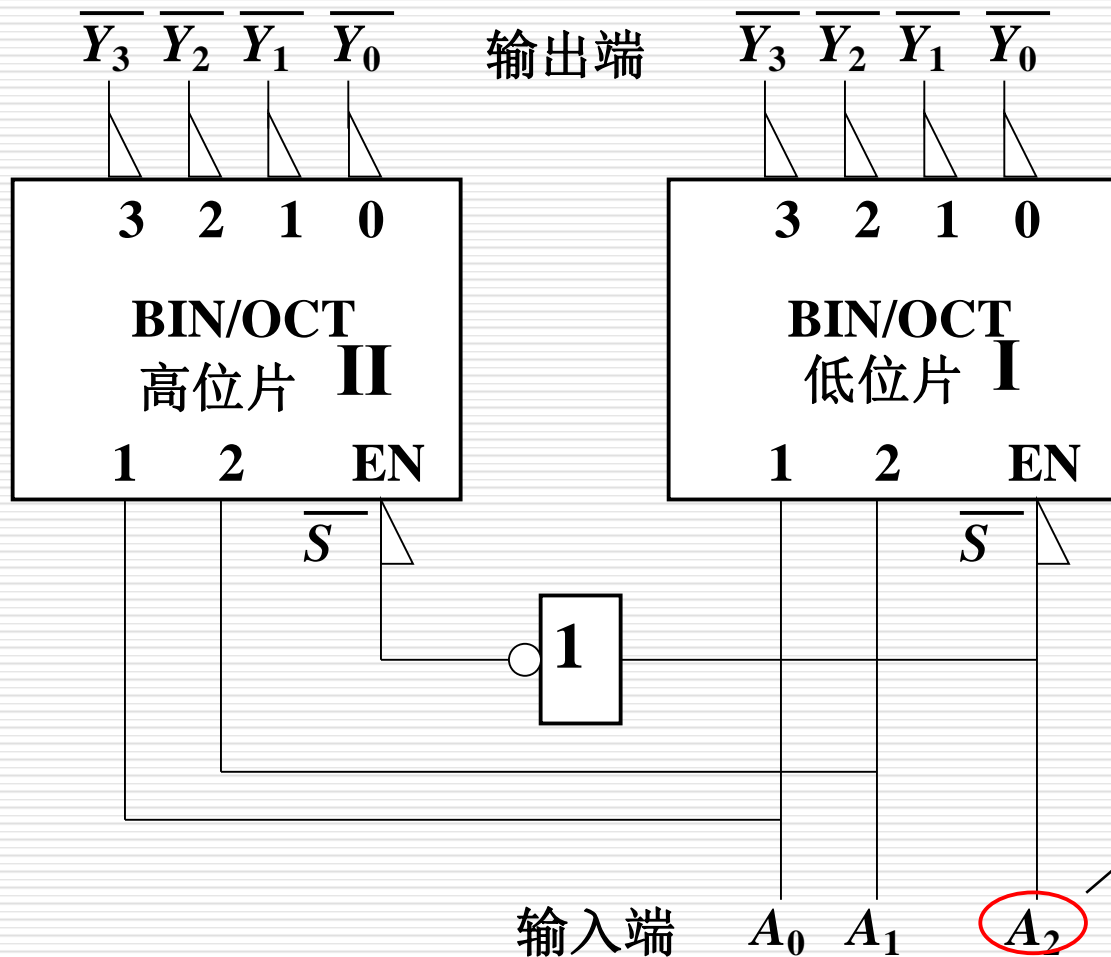
$$\overline{Y}_3 = \overline{\overline{\overline{E}} \overline{\overline{A_1}} A_0}$$

(a) 2线-4线译码器 (74HC139) ——逻辑符号说明

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。符号框内部的输入、输出变量表示其内部的逻辑关系。在推导表达式的过程中，如果低有效的输入或输出变量(如)上面的“—”号参与运算(如 \bar{E} 变为 E)，则在画逻辑图或验证真值表时，注意将其还原为低有效符号。



2线-4线译码器扩展应用

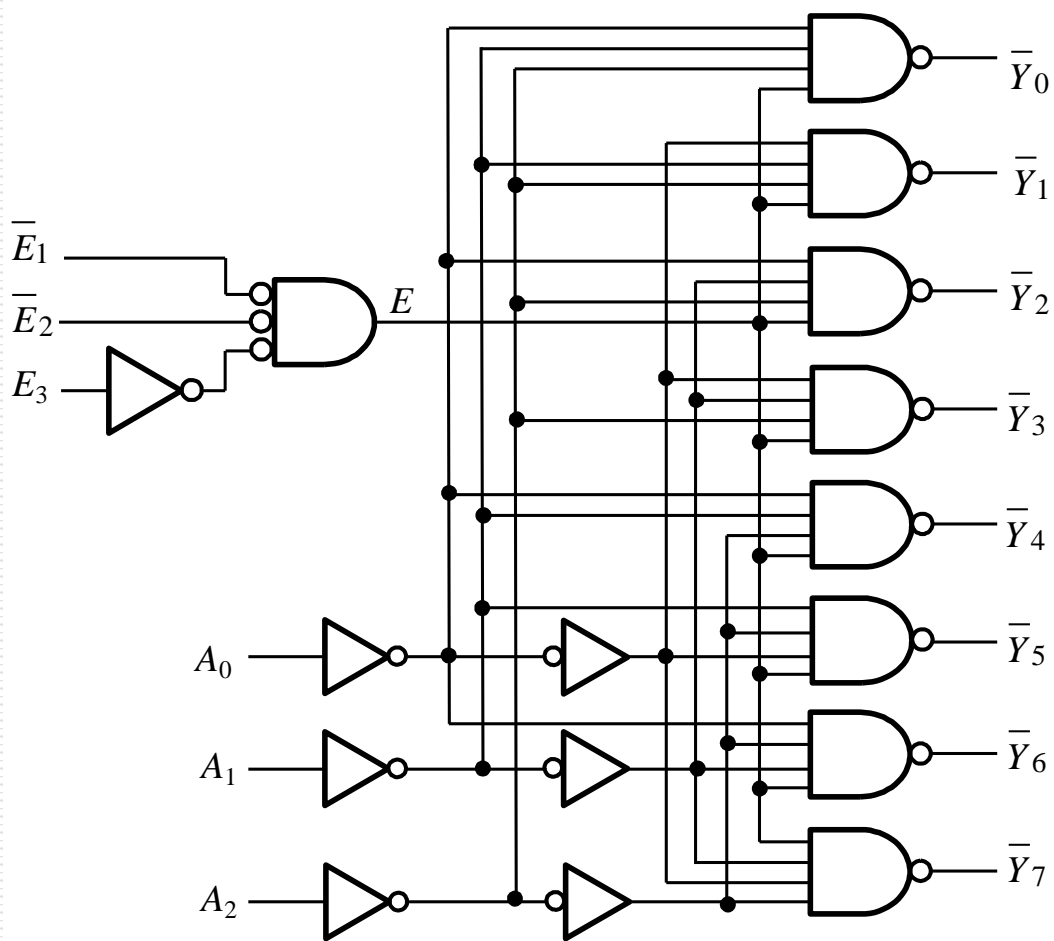


A2	A1	A0
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

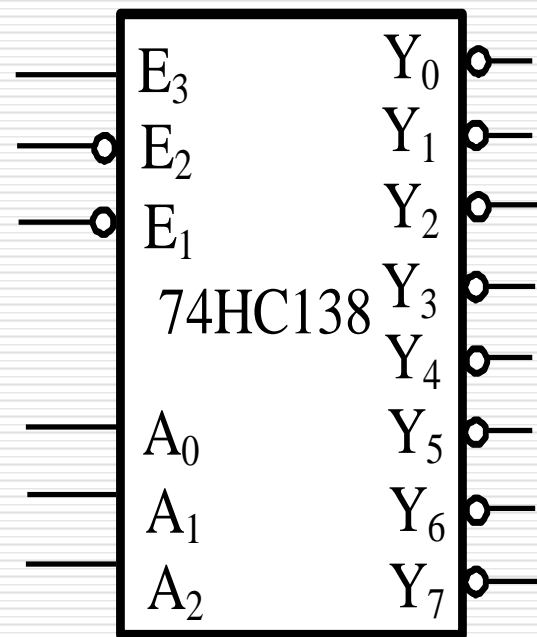
2线-4线译码器扩展构成3线-8线译码器

CT54/74138

(b) 3线-8线译码器 (74HC138)



逻辑图



逻辑符号

$A_0 \sim A_2$ — 地址输入端
 $\bar{Y}_0 \sim \bar{Y}_7$ — 译码输出端
 $\left. \begin{matrix} E_3 \\ \bar{E}_2 \\ \bar{E}_1 \end{matrix} \right\}$ — 选通输入端

3线-8线译码器（74HC138）功能表

$$\begin{cases} E_3 = 1 \\ \overline{E_1} + \overline{E_2} = 0 \end{cases}$$

[illegible]

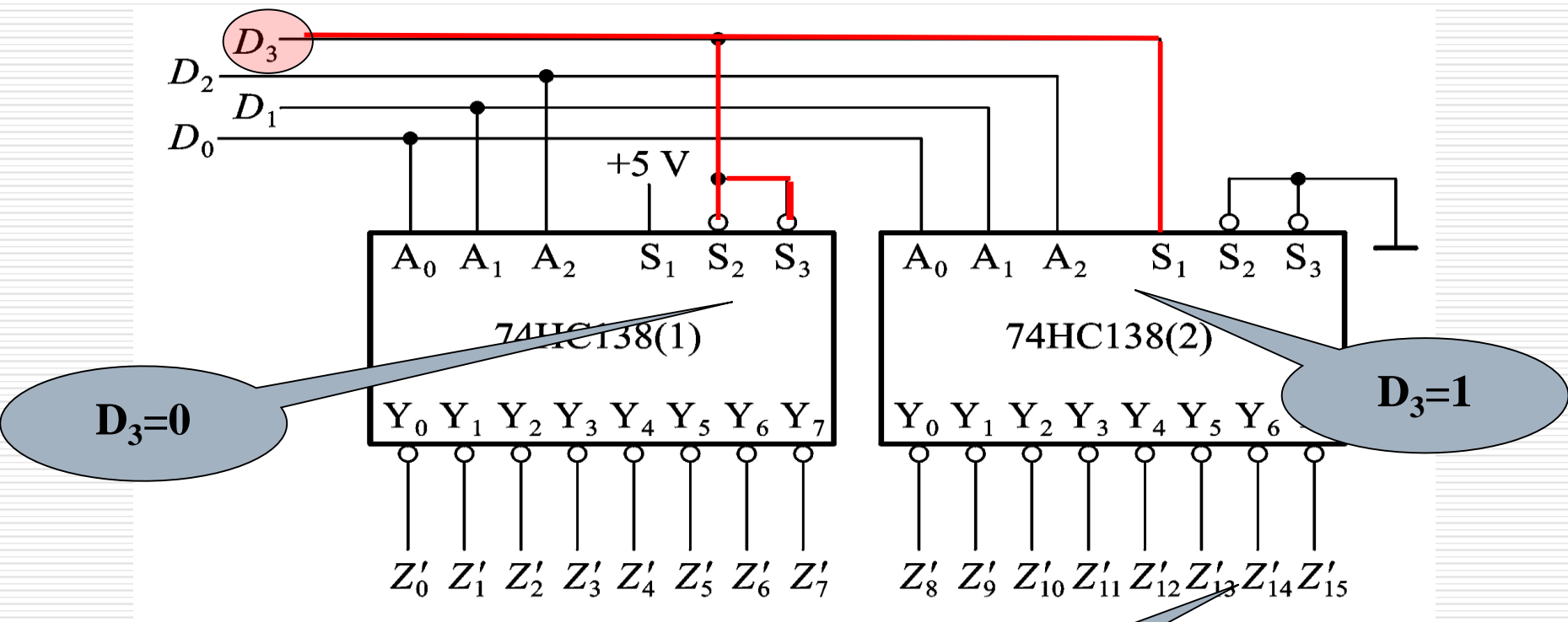
$$\overline{Y}_0 = \cdot \overline{A}_2 \cdot \overline{A}_1 \cdot \overline{A}_0 \quad \overline{Y}_1 = \cdot \overline{A}_2 \cdot \overline{A}_1 \cdot A_0 \quad \overline{Y}_2 = \cdot \overline{A}_2 \cdot A_1 \cdot \overline{A}_0 \quad \overline{Y}_3 = \cdot \overline{A}_2 \cdot A_1 \cdot A_0$$

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

[illegible]

□ 利用附加控制端进行扩展

例：用74HC138（3线—8线译码器）扩展 4线—16线译码器



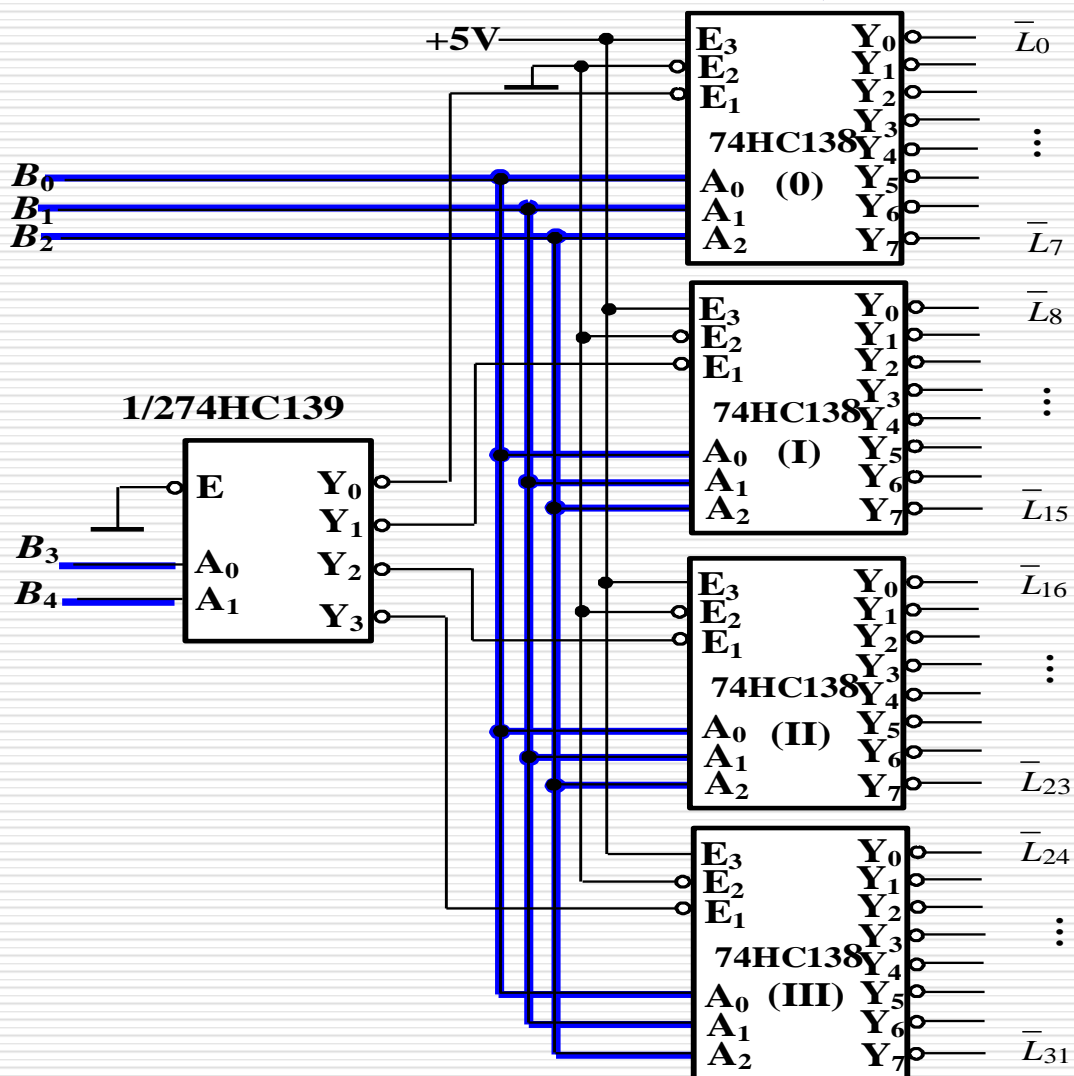
译码条件

$$\begin{cases} E_3 = 1 \\ \overline{E_1} + \overline{E_2} = 0 \end{cases}$$

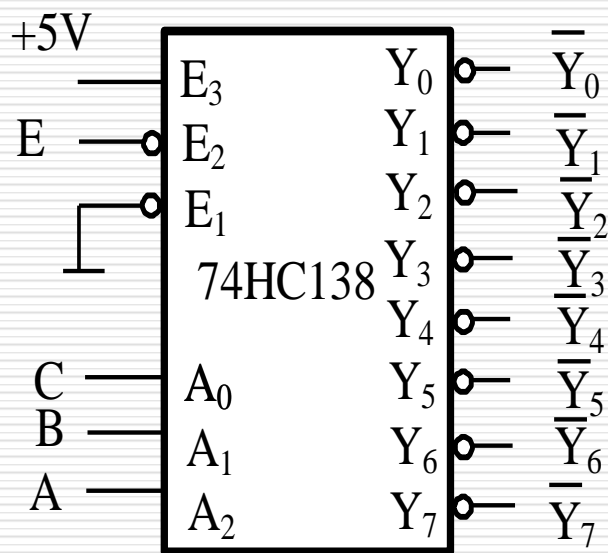
$$Z'_i = m'_i$$

1、译码器的扩展

用74X139和74X138构成5线-32线译码器



2、用译码器实现逻辑函数。当 $E_3=1$ ， $E_2=E_1=0$ 时



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

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

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

⋮

⋮

⋮

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

3线-8线译码器的 $Y_0 \sim Y_7$ 含三变量函数的全部最小项。

基于这一点用该器件能够方便地实现三变量逻辑函数。

用一片74HC138实现函数 $L = \overline{A}\overline{C} + AB$

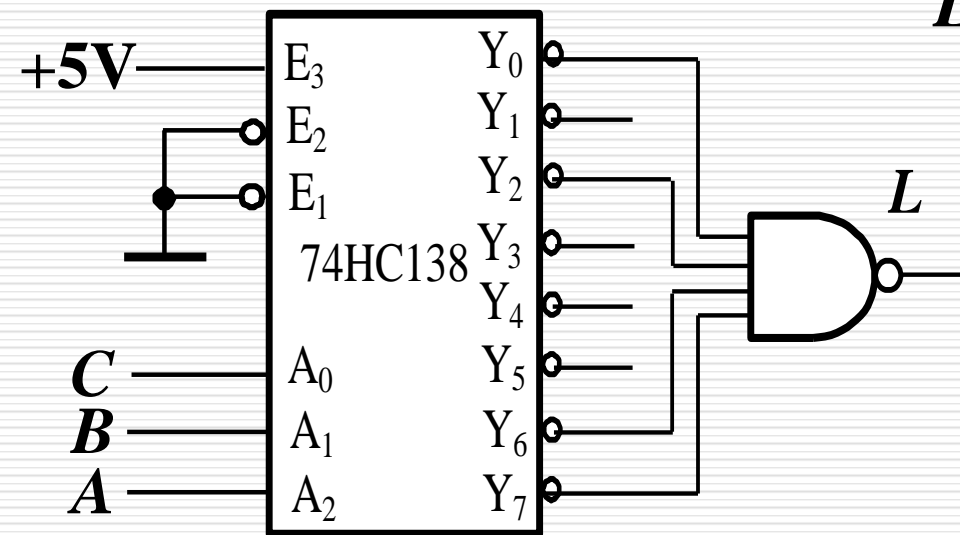
首先将函数式变换为最小项之和的形式

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

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

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

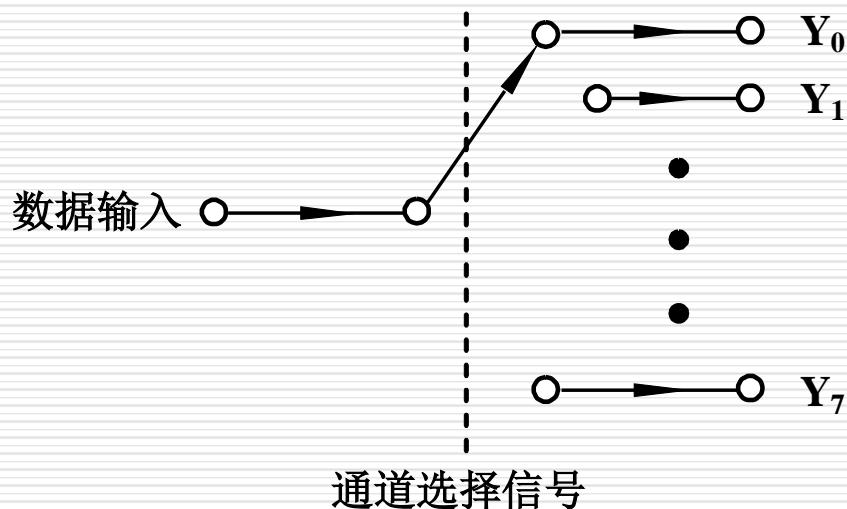
$$= \overline{Y_0 \cdot Y_2 \cdot Y_6 \cdot Y_7}$$



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。

用74HC138组成数据分配器

数据分配器示意图

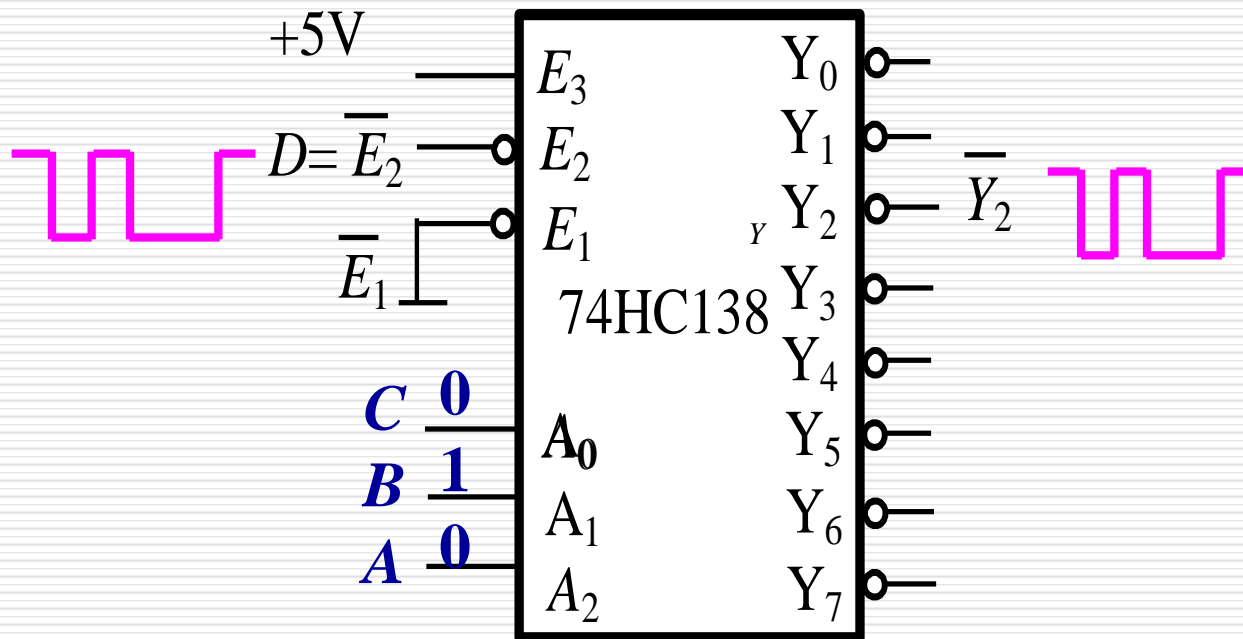


数据分配器：相当于多输出的单刀多掷开关，是将公共数据线上的数据按需要送到不同的通道上去的逻辑电路。

用译码器实现数据分配器

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

当 $ABC = 010$ 时, $\overline{Y_2} = D$



74HC138译码器作为数据分配器时的功能表

[illegible]