

数字电路

Digital Circuits and System

李文明

liwenming@ict.ac.cn



组合逻辑电路



组合逻辑电路重点内容

- 组合逻辑电路的描述方法
- 构成组合逻辑电路的基本组件
- 组合逻辑电路的分析与设计方法
- 常用的组合逻辑电路模块及其应用实例
 - 编码器、译码器、数据选择器、加法器、数据比较器
- 竞争-冒险现象及其避免方法



组合逻辑电路重点内容

- 组合逻辑电路的描述方法
- 构成组合逻辑电路的基本组件
- 组合逻辑电路的分析与设计方法
- 常用的组合逻辑电路模块及其应用实例
 - 编码器、译码器、数据选择器、加法器、数据比较器
- 竞争-冒险现象及其避免方法



用译码器设计组合逻辑电路

- 3位二进制译码器给出3变量的全部最小项, n 位二进制译码器给出 n 个变量的全部最小项;
- 任意函数: 将 n 位二进制译码输出的最小项组合起来, 可获得任何变量不大于 n 的组合函数

$$Y = \sum m_i$$



用译码器设计组合逻辑电路举例

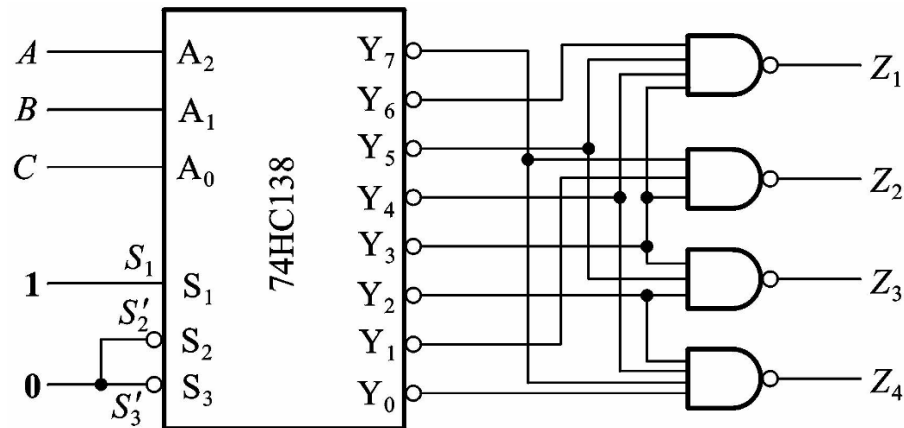
- 例：利用74HC138设计一个多输出的组合逻辑电路，输出逻辑函数式为：

$$Z_1 = AC' + A'BC + AB'C$$

$$Z_2 = BC + A'B'C$$

$$Z_3 = A'B + AB'C$$

$$Z_4 = A'BC' + B'C' + ABC$$



$$Z_1 = AC' + A'BC + AB'C = \sum m(3, 4, 5, 6) = (m'_3 m'_4 m'_5 m'_6)'$$

$$Z_2 = BC + A'B'C = \sum m(1, 3, 7) = (m'_1 m'_3 m'_7)'$$

$$Z_3 = A'B + AB'C = \sum m(2, 3, 5) = (m'_2 m'_3 m'_5)'$$

$$Z_4 = A'BC' + B'C' + ABC = \sum m(0, 2, 4, 7) = (m'_0 m'_2 m'_4 m'_7)'$$

用数据选择器实现逻辑函数方法

- 4选1数据选择器的逻辑式

$$Y_1 = S_1(D_{10}(A_1'A_0') + D_{11}(A_1'A_0) + D_{12}(A_1A_0') + D_{13}(A_1A_0))$$

- 当 $S_1 = 1$ 时，输出与输入间的逻辑关系为

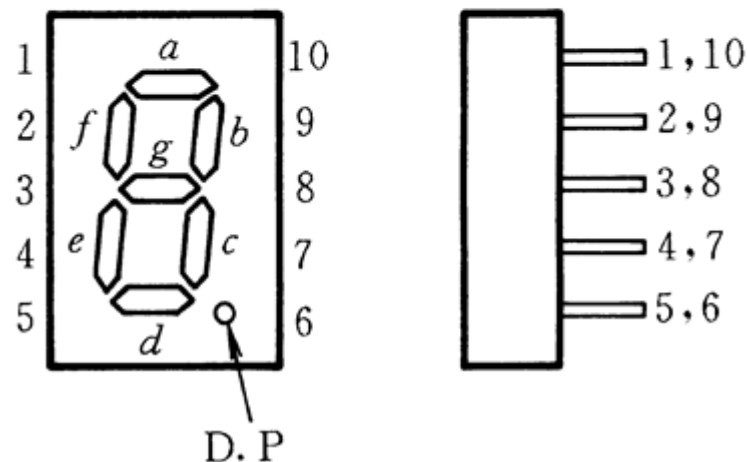
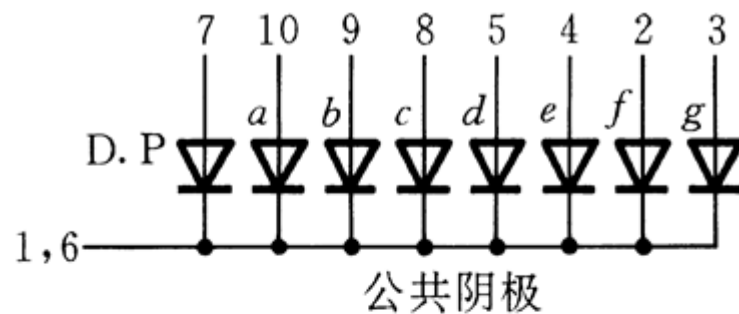
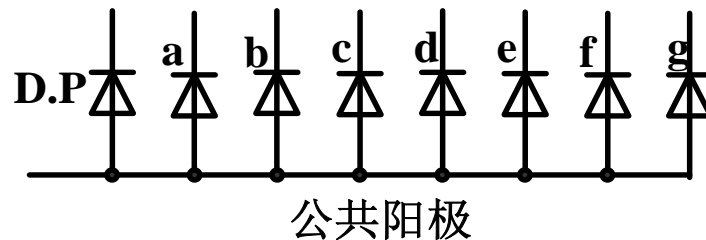
$$Y_1 = D_0(A_1'A_0') + D_1(A_1'A_0) + D_2(A_1A_0') + D_3(A_1A_0)$$

- A_1 、 A_0 作为两输入变量，输出端可以产生任何形式的三变量逻辑组合函数
- 同理，用具有n位地址输入的数据选择器，可以产生任何形式输入变量数不大于n+1的组合逻辑函数

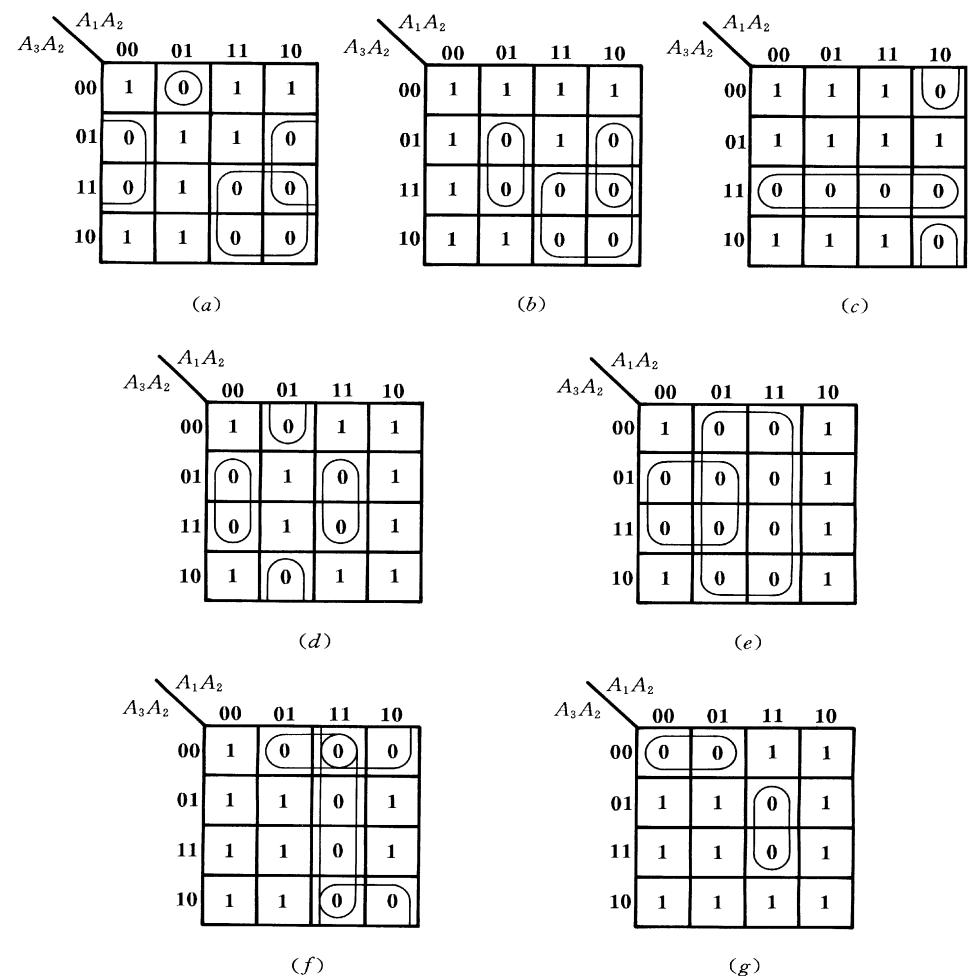


七段字符显示译码器(1)

- 功能：把二进制码转换成控制数码管显示相应字符的编码
- 七段数码管
 - 7个显示控制信号
 - 共阳：7个显示发光管的“正”极连在一起，控制信号为“低”点亮
 - 共阴：7个显示发光管的“负”极连在一起，控制信号为“高”点亮



七段字符显示译码器(2)



从真值表画出 $Y_a \sim Y_g$ 的卡诺图，圈“0”然后求反可得各输出端的逻辑式

数字	输 入				输 出						
	A_3	A_2	A_1	A_0	Y_a	Y_b	Y_c	Y_d	Y_e	Y_f	Y_g
0	0	0	0	0	1	1	1	1	1	1	0
1	0	0	0	1	0	1	1	0	0	0	0
2	0	0	1	0	1	1	0	1	1	0	1
3	0	0	1	1	1	1	1	1	0	0	1
4	0	1	0	0	0	1	1	0	0	1	1
5	0	1	0	1	1	0	1	1	0	1	1
6	0	1	1	0	0	0	1	1	1	1	1
7	0	1	1	1	1	1	1	0	0	0	0
8	1	0	0	0	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	0	0	1	1
10	1	0	1	0	0	0	0	1	1	0	1
11	1	0	1	1	0	0	1	1	0	0	1
12	1	1	0	0	0	1	0	0	0	1	1
13	1	1	0	1	1	0	1	1	0	1	1
14	1	1	1	0	0	0	0	1	1	1	1
15	1	1	1	1	0	0	0	0	0	0	0

七段字符显示译码器(3)

- 译码器输出逻辑函数

$$Y_a = (A'_3 A'_2 A'_1 A_0 + A_3 A_1 + A_2 A'_0)'$$

$$Y_b = (A_3 A_1 + A_2 A_1 A'_0 + A_2 A'_1 A_0)'$$

$$Y_c = (A_3 A_2 + A'_2 A_1 A'_0)'$$

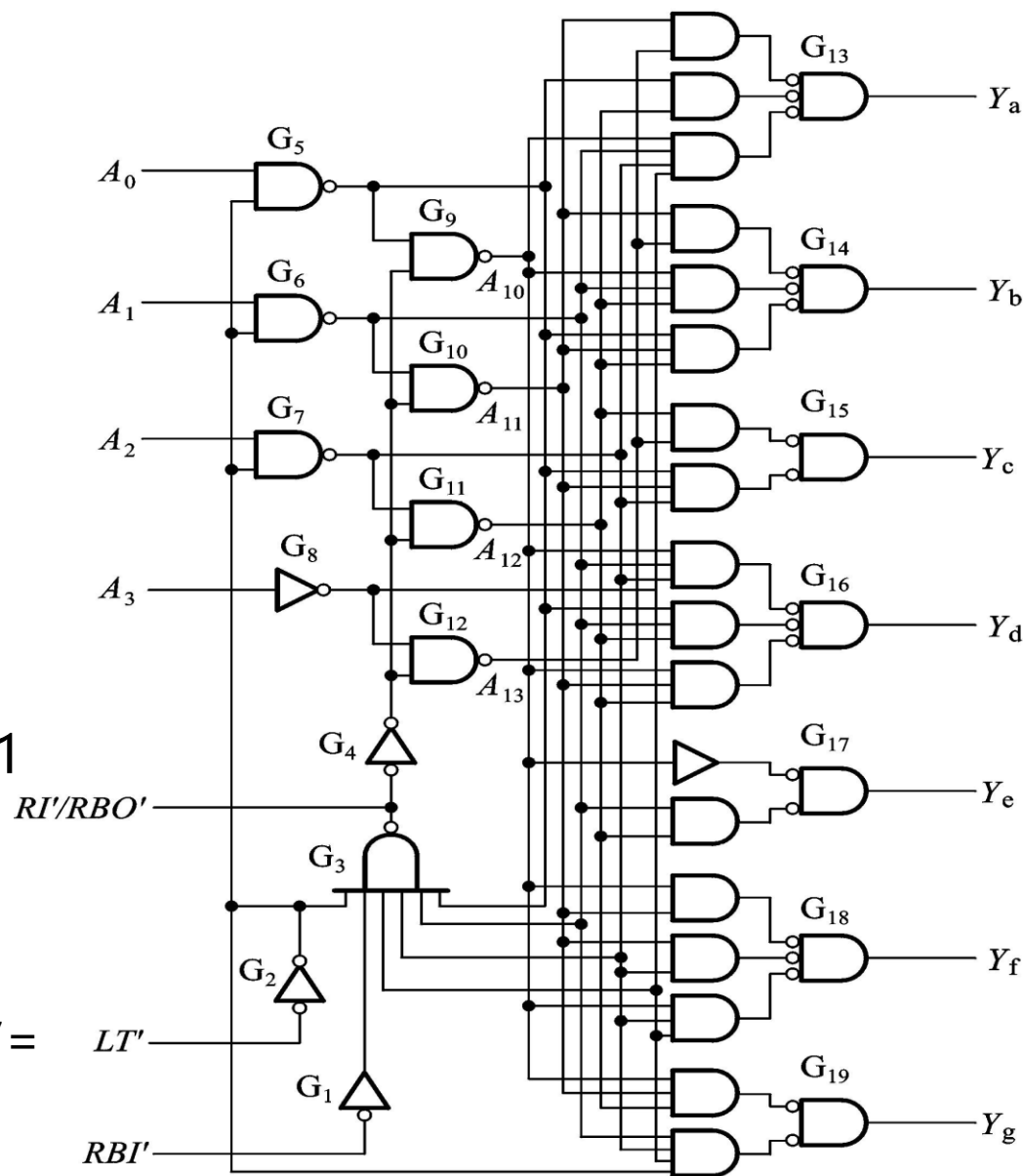
$$Y_d = (A_2 A_1 A_0 + A_2 A'_1 A'_0 + A'_2 A'_1 A_0)'$$

$$Y_e = (A_2 A'_1 + A_0)'$$

$$Y_f = (A'_3 A'_2 A_0 + A'_2 A_1 + A_1 A_0)'$$

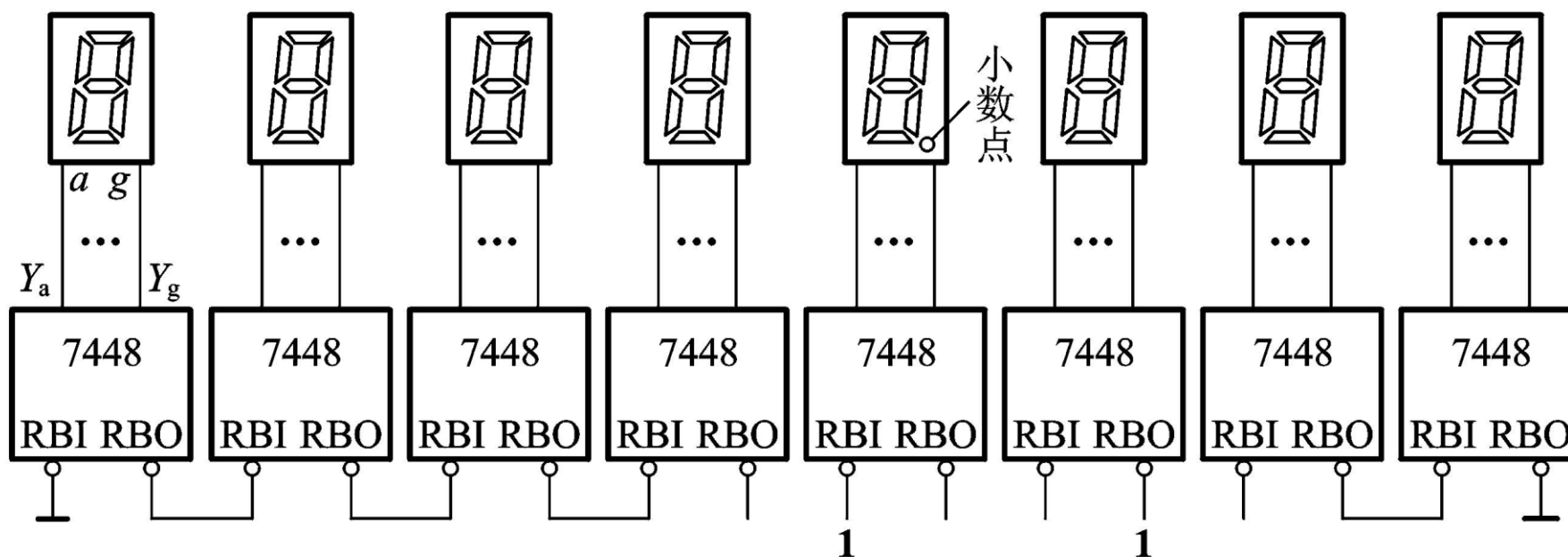
$$Y_g = (A'_3 A'_2 A'_1 + A_2 A_1 A_0)'$$

- 测试端, $LT' = 0$, 七段同时点亮; 工作时 $LT' = 1$
- 灭“0”端, $RBI' = 0$, 显示的“0”熄灭
- 灭灯输入/输出端, RI'/RBO'
 - ✓ 灭灯输入端, 当 $RI' = 0$, 熄灭相应数码段
 - ✓ 作为输出端使用时, 当 $A_3 A_2 A_1 A_0 = 0$, 并且 $RBI' = 0$ 时, 可使本该显示的零熄灭



七段字符显示译码器(4)

- 例:利用 RBI' 和 RBO' 的配合, 实现多位显示系统的灭零控制
 - 整数部分: 最高位是0, 而且灭掉以后, 输出 RBO' 作为次高位的输入信号 RBI'
 - 小数部分: 最低位是0, 而且灭掉以后, 输出 RBO' 作为次低位的输入信号 RBI'



组合逻辑电路重点内容

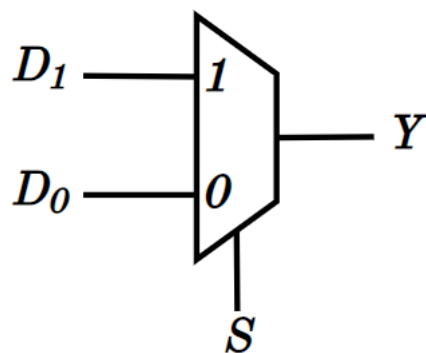
- 组合逻辑电路的描述方法
- 构成组合逻辑电路的基本组件
- 组合逻辑电路的分析与设计方法
- 常用的组合逻辑电路模块及其应用实例
 - 编码器、译码器、数据选择器、加法器、数据比较器
- 竞争-冒险现象及其避免方法



数据选择器概念

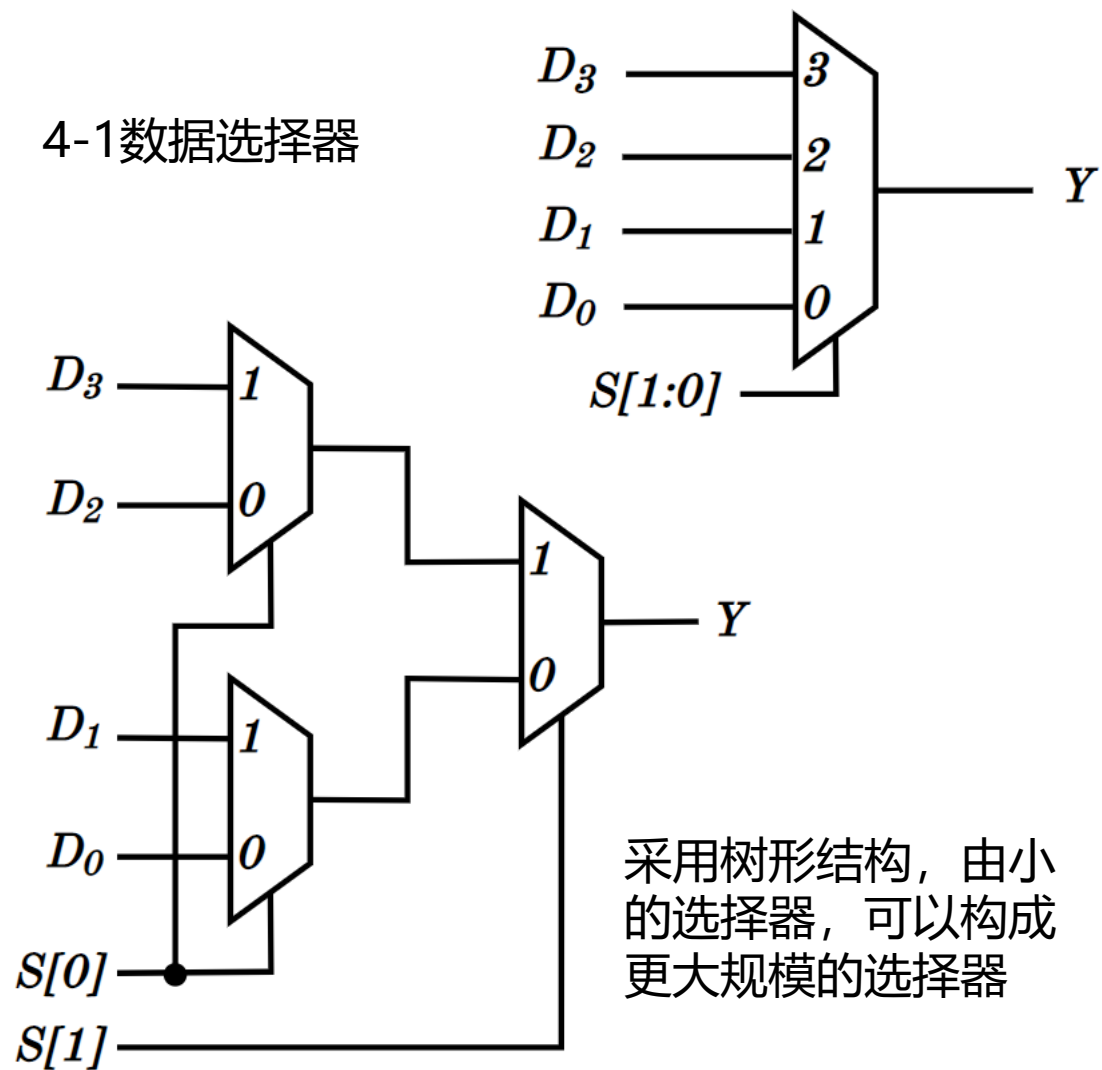
- 多个输入，一个输出，通过选择信号控制哪个输入连到输出

2-1数据选择器
(Multiplexer, MUX)



S	D_1	D_0	Y
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

4-1数据选择器



数据选择器实例

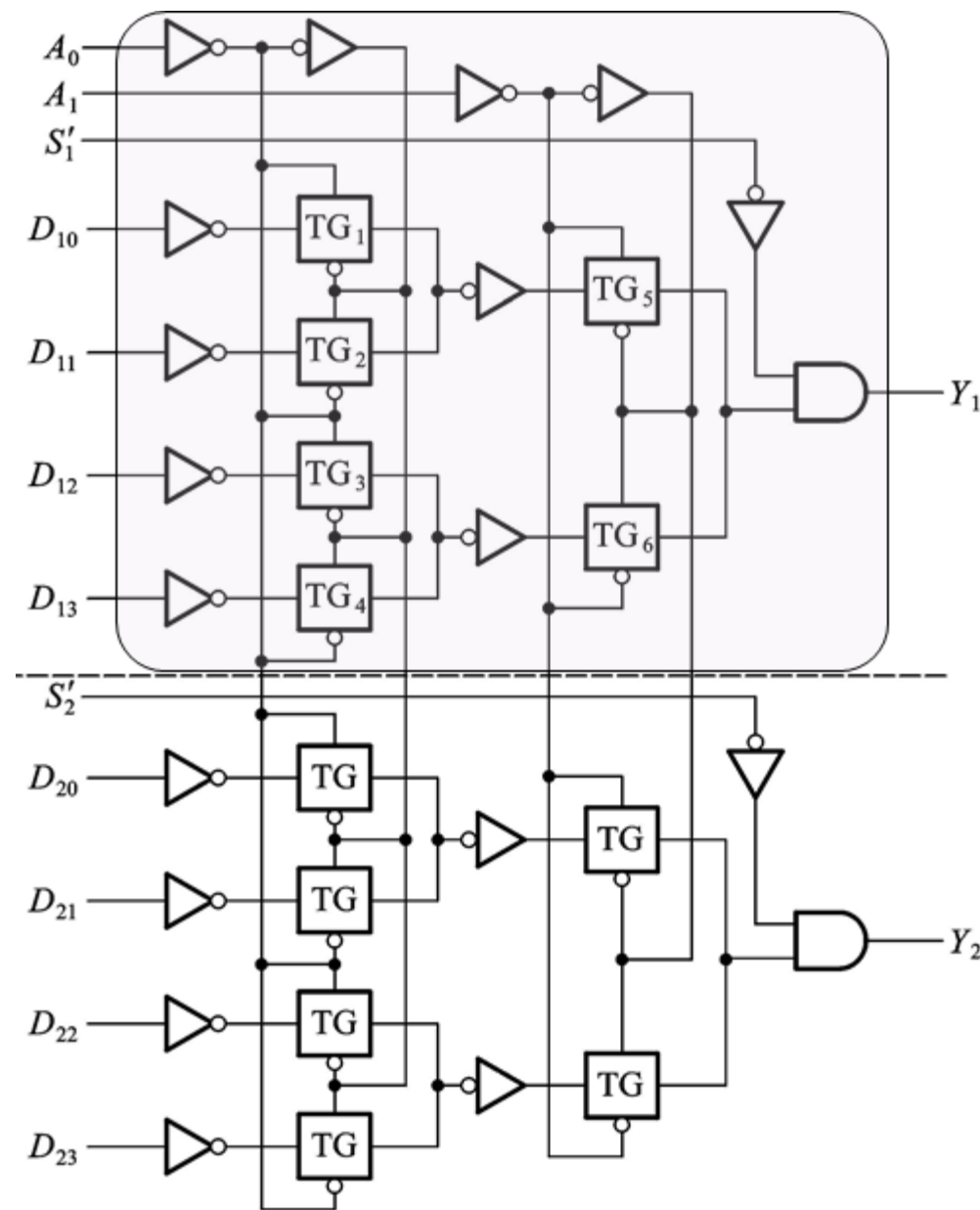
- 数据选择器
 - Data Selector,也称多路开关 (Multiplexer)
 - 功能: 数字信号传输过程中, 从多个数据中选出某一个送到输出端
- 例: 双4选1数据选择器74HC153

$S'_1 = 0$ 工作模式

$S'_1 = 1$ 锁定模式输出为低电平

S'_1	A_1	A_0	Y_1
1	X	X	0
0	0	0	D_{10}
0	0	1	D_{11}
0	1	0	D_{12}
0	1	1	D_{13}

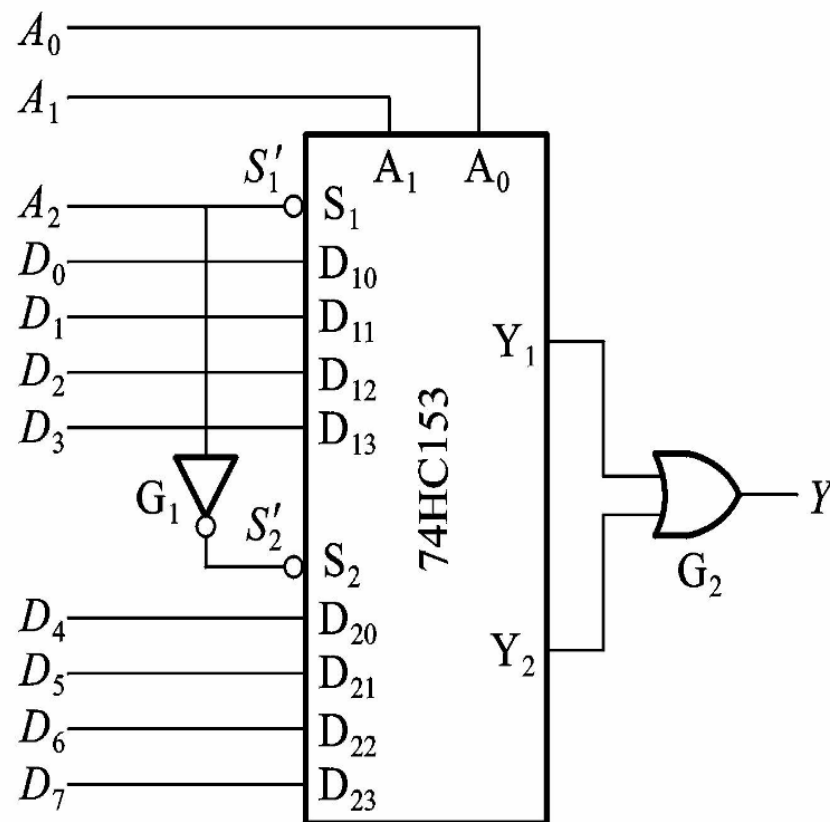
$$Y_1 = S_1(D_{10}(A'_1 A'_0) + D_{11}(A'_1 A_0) + D_{12}(A_1 A'_0) + D_{13}(A_1 A_0))$$



数据选择器功能扩展

- 例，用两个带附加控制端的“4-1数据选择器”组成一个“8-1数据选择器”
- 扩展方法：
 - 数据输入：2个“4选1”有8个数据输入
 - 地址选择：“8选1”需3位地址选择输入，用“4选1”的2位地址入，附加控制信号 S' 用作第3位地址
- 逻辑表达式：

$$Y = (A_2' A_1' A_0') D_0 + (A_2' A_1' A_0) D_1 + (A_2' A_1 A_0') D_2 + (A_2' A_1 A_0) D_3 \\ + (A_2 A_1' A_0') D_4 + (A_2 A_1' A_0) D_5 + (A_2 A_1 A_0') D_6 + (A_2 A_1 A_0) D_7$$



用数据选择器实现逻辑函数方法

- 4选1数据选择器的逻辑式

$$Y_1 = S_1(D_{10}(A_1'A_0') + D_{11}(A_1'A_0) + D_{12}(A_1A_0') + D_{13}(A_1A_0))$$

- 当 $S_1 = 1$ 时，输出与输入间的逻辑关系为

$$Y_1 = D_0(A_1'A_0') + D_1(A_1'A_0) + D_2(A_1A_0') + D_3(A_1A_0)$$

- A_1 、 A_0 作为两输入变量，输出端可以产生任何形式的三变量逻辑组合函数
- 同理，用具有n位地址输入的数据选择器，可以产生任何形式输入变量数不大于n+1的组合逻辑函数



用数据选择器实现交通信号灯监视电路

- 交通信号灯监视电路的逻辑函数式：

$$Z = R'A'G' + R'AG + RA'G + RAG' + RAG$$

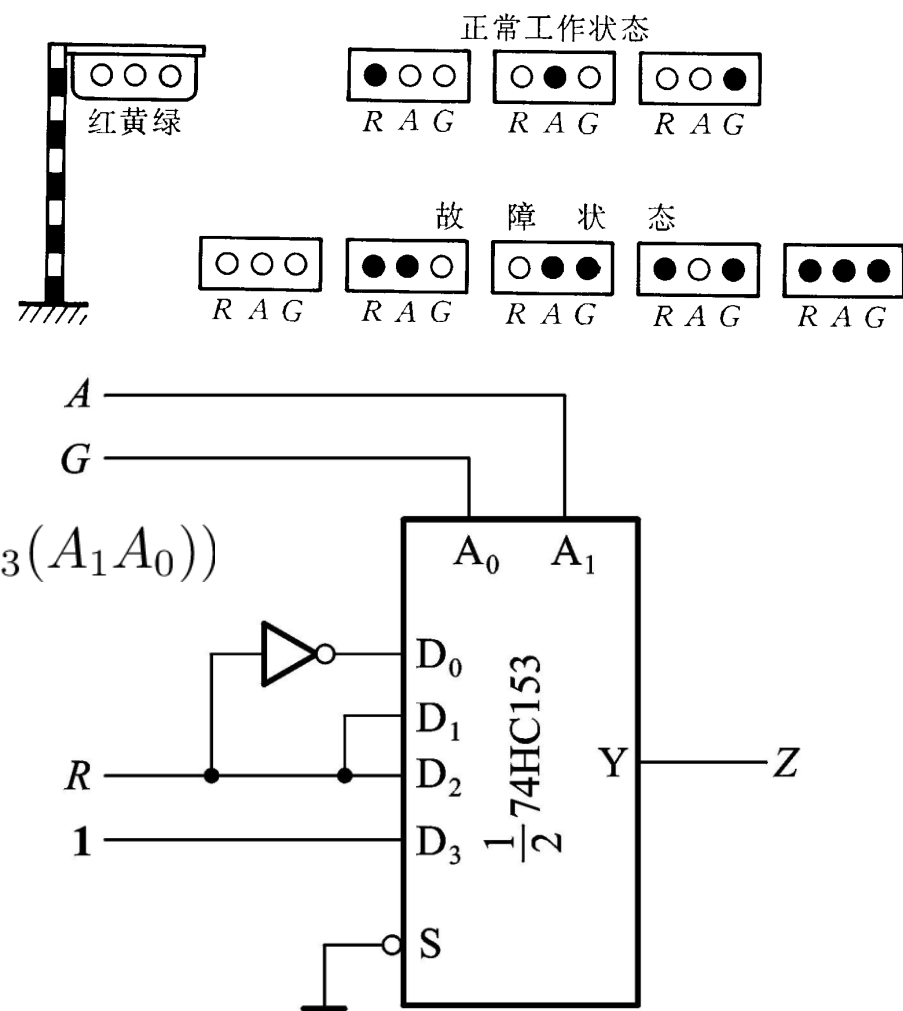
$$= R'(A'G') + R(A'G) + R(AG') + 1 \cdot (AG)$$

- 4-1数据选择器的逻辑函数式：

$$Y_1 = S_1(D_0(A'_1A'_0) + D_1(A'_1A_0) + D_2(A_1A'_0) + D_3(A_1A_0))$$

- 比较逻辑函数式 Z 与 Y_1

- $A_1 = A, A_0 = G$
- $D_0 = R', D_3 = 1$
- $D_1 = D_2 = R$



用数据选择器实现逻辑函数举例

- 用8选1数据选择器产生三变量逻辑函数

$$Z = A'B'C' + AC + A'BC$$

$$\begin{aligned} &= 1 \cdot (A'B'C') + 0 \cdot (A'B'C) + 0 \cdot (A'BC') + 1 \cdot (A'BC) + \\ &0 \cdot (AB'C') + 1 \cdot (AB'C) + 0 \cdot (ABC') + 1 \cdot (ABC) \end{aligned}$$

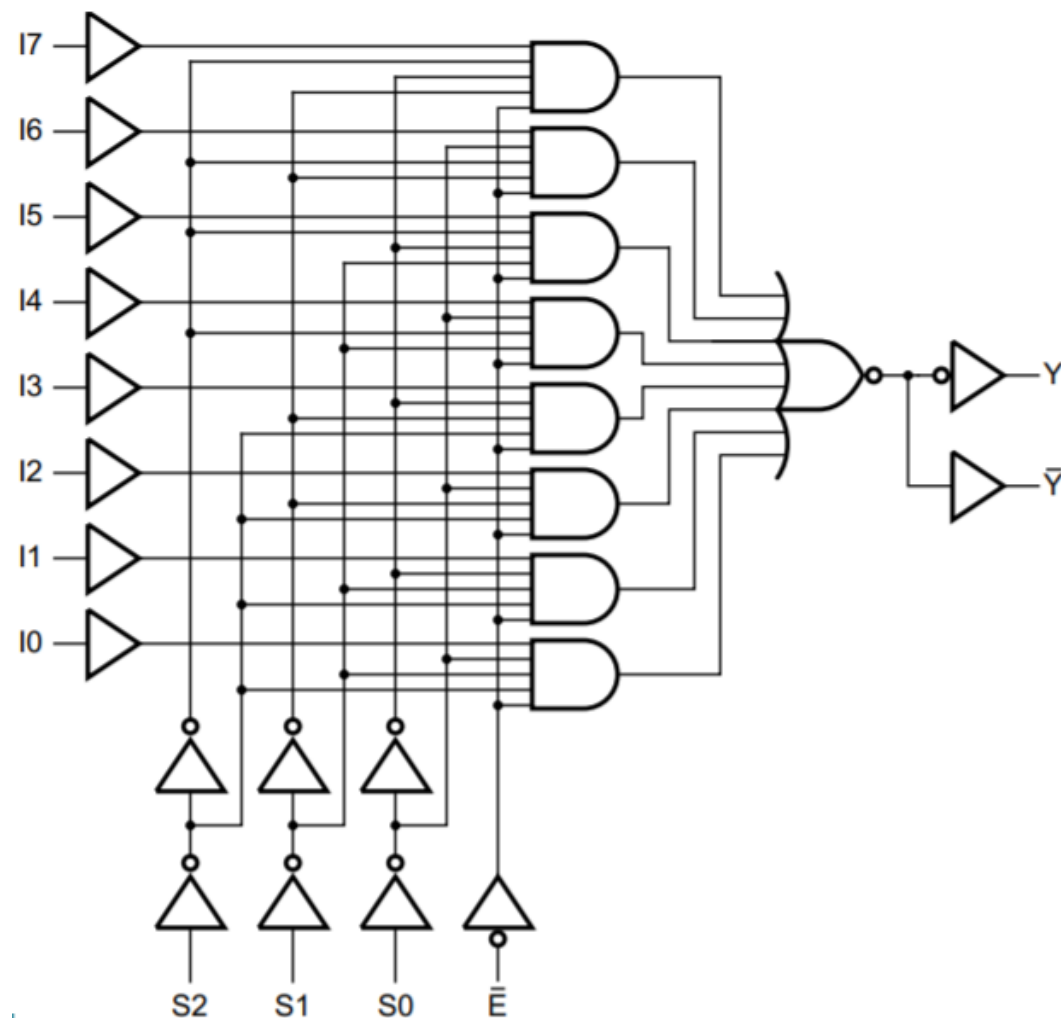
- 8选1数据选择器 **74HC151** 在 $E'=0$ ($E=1$) 时, 输出逻辑式为

$$\begin{aligned} Y = & I_0(S_2'S_1'S_0') + I_1(S_2'S_1'S_0) + I_2(S_2'S_1S_0') + I_3(S_2'S_1S_0) + \\ & I_4(S_2S_1'S_0') + I_5(S_2S_1'S_0) + I_6(S_2S_1S_0') + I_7(S_2S_1S_0) \end{aligned}$$

$$\bar{Y} = Y'$$

- 对照上述 Z 、 Y 表达式

- $I_0 = I_3 = I_5 = I_7 = 1$
- $I_1 = I_2 = I_4 = I_6 = 0$
- $S_2 = A, S_1 = B, S_0 = C$



用数据选择器实现1位全减器(1)

- 试用双4选1数据选择器74HC153构成全减器，设A为被减数，B为减数， C_I 为低位的借位，D为差， C_O 为向高位的借位
- 全减器真值表如右，输出逻辑函数式：

$$D = m_1 + m_2 + m_4 + m_7$$

$$= A'B'C_I + A'BC'_I + AB'C'_I + ABC_I$$

$$= A \cdot B'C'_I + A' \cdot B'C_I + A' \cdot BC'_I + A \cdot BC_I$$

$$C_O = m_1 + m_2 + m_3 + m_7$$

$$= A'B'C_I + A'BC'_I + A'BC_I + ABC_I$$

$$= 0 \cdot B'C'_I + A' \cdot B'C_I + A' \cdot BC'_I + 1 \cdot BC_I$$

A	B	C_I	D	C_O
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

用数据选择器实现1位全减器(2)

- 数据选择器函数式:

$$Y_1 = S_1(D_0(A_1' A_0') + D_1(A_1' A_0) + D_2(A_1 A_0') + D_3(A_1 A_0))$$

$$D = m_1 + m_2 + m_4 + m_7$$

$$= A' B' C_I + A' B C_I' + A B' C_I' + A B C_I$$

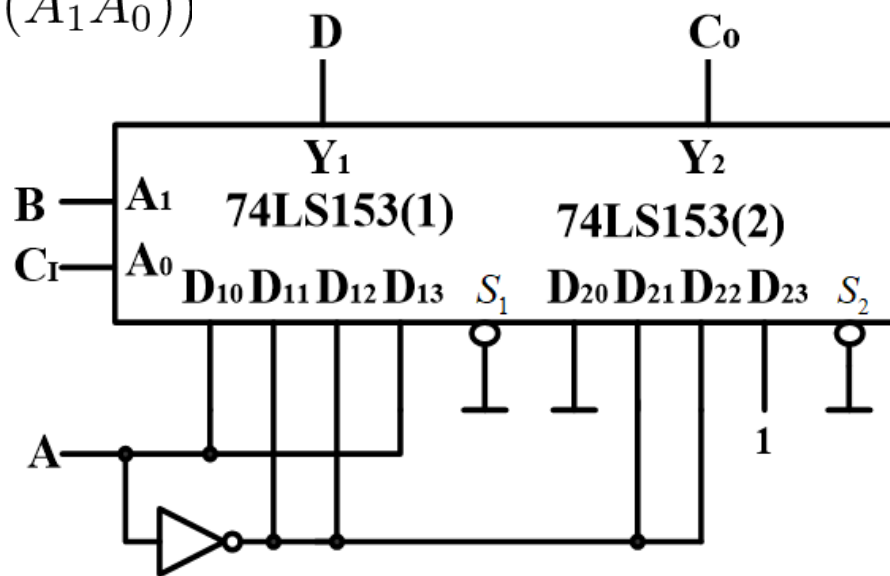
$$= A \cdot B' C_I' + A' \cdot B' C_I + A' \cdot B C_I' + A \cdot B C_I$$

$$C_O = m_1 + m_2 + m_3 + m_7$$

$$= A' B' C_I + A' B C_I' + A' B C_I + A B C_I$$

$$= 0 \cdot B' C_I' + A' \cdot B' C_I + A' \cdot B C_I' + 1 \cdot B C_I$$

- 比较上述各式 $A_1 = B, A_0 = C_I, Y_1 = D, Y_2 = C_O$
 $D_{10} = D_{13} = A, D_{11} = D_{12} = A'$
 $D_{20} = 0, D_{21} = D_{22} = A', D_{23} = 1$



组合逻辑电路重点内容

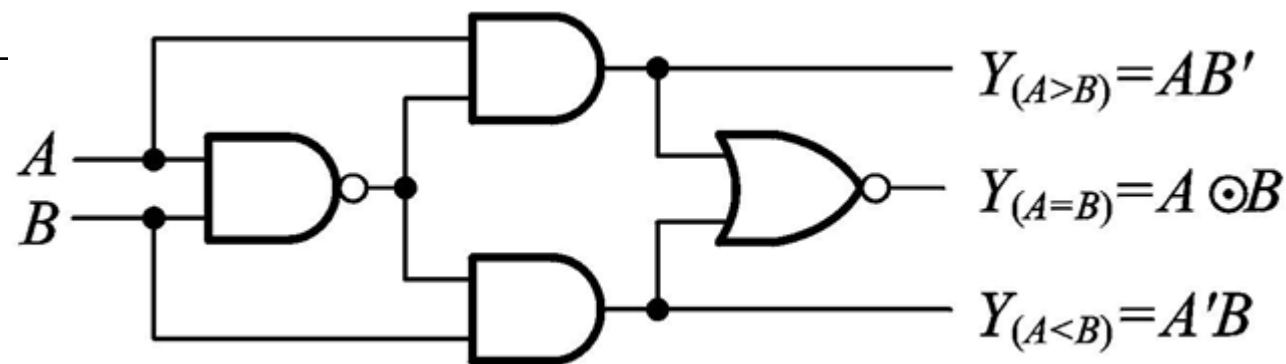
- 组合逻辑电路的描述方法
- 构成组合逻辑电路的基本组件
- 组合逻辑电路的分析与设计方法
- 常用的组合逻辑电路模块及其应用实例
 - 编码器、译码器、数据选择器、加法器、数据比较器
- 竞争-冒险现象及其避免方法



数值比较器的概念

- 用于比较两个数值大小的逻辑电路
- 1位数值比较器， A, B 两个 1 位二进制数比较有 3 种可能结果

输 入		输 出		
A	B	$Y_{A>B}$	$Y_{A=B}$	$Y_{A<B}$
0	0	0	1	0
0	1	0	0	1
1	0	1	0	0
1	1	0	1	0



多位数值比较器

- 原理：从高位比起。高位相等，再比较下一位
- 比较两个4为二进制数 $A_3 A_2 A_1 A_0$ 和 $B_3 B_2 B_1 B_0$ ，输出为 $Y_{(A>B)}$ 、 $Y_{(A=B)}$ 和 $Y_{(A<B)}$

$$Y_{(A<B)} = A'_3 B_3 + (A_3 \oplus B_3)' A'_2 B_2 + (A_3 \oplus B_3)' (A_2 \oplus B_2)' A'_1 B_1 + (A_3 \oplus B_3)' (A_2 \oplus B_2)' (A_1 \oplus B_1)' A'_0 B_0$$

$$Y_{(A=B)} = (A_3 \oplus B_3)' (A_2 \oplus B_2)' (A_1 \oplus B_1)' (A_0 \oplus B_0)'$$

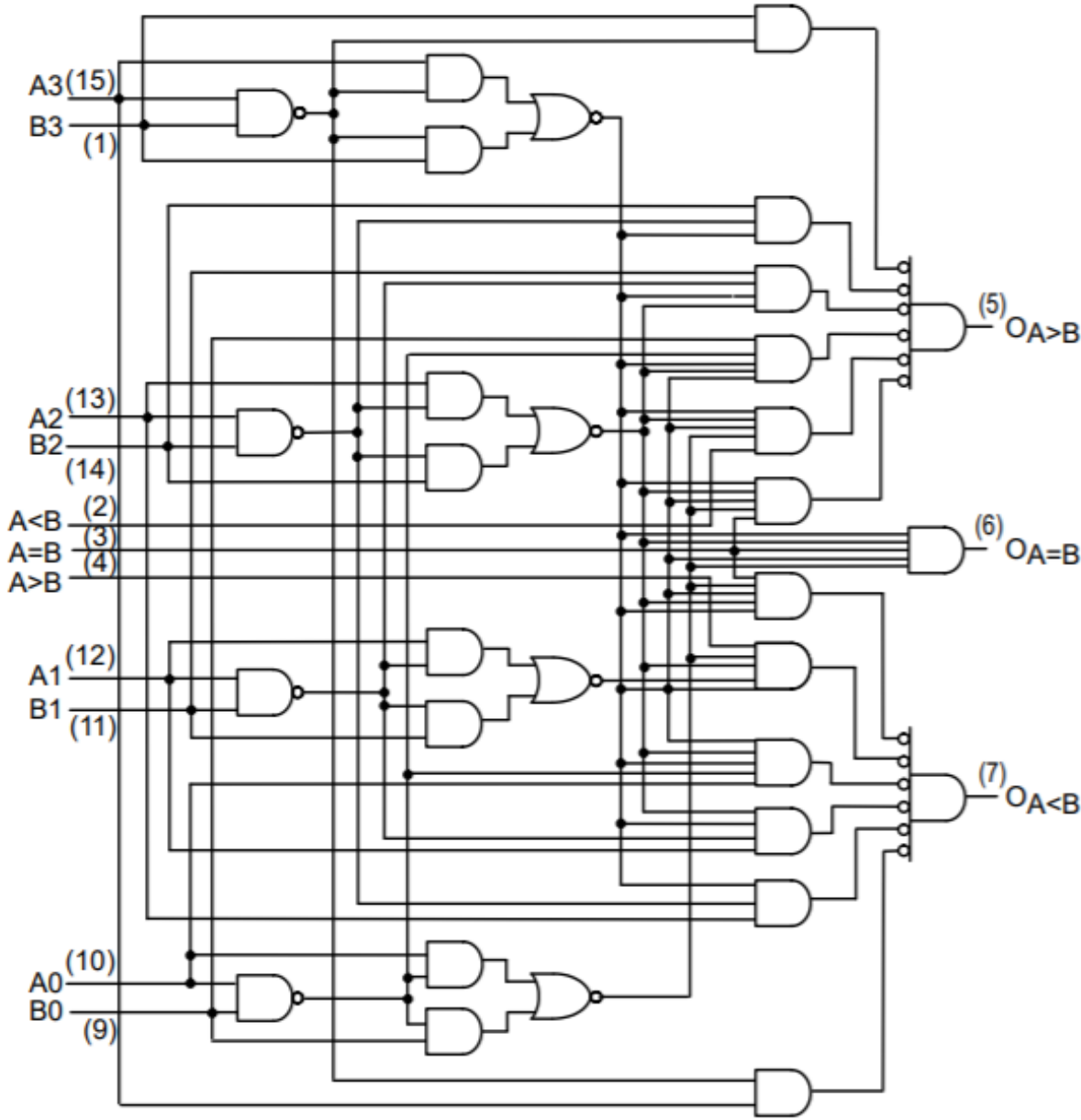
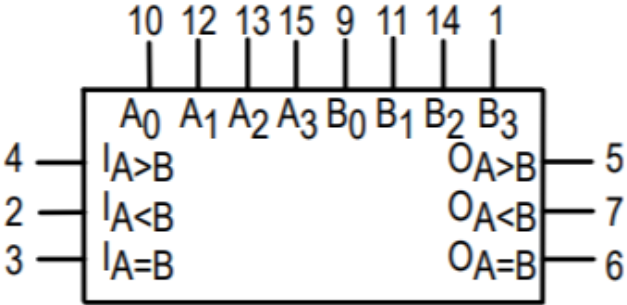
$$Y_{(A>B)} = (Y_{(A<B)} + Y_{(A=B)})'$$

可扩展4位数值较器逻辑图

真值表

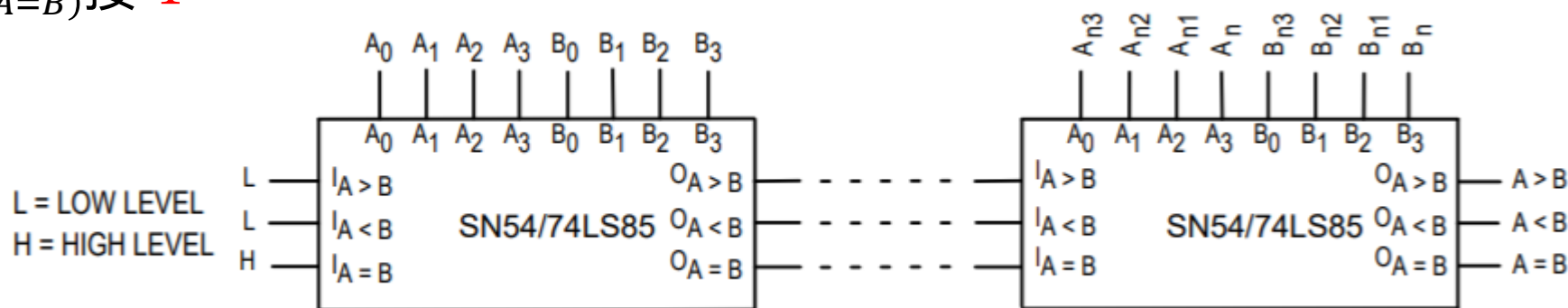
COMPARING INPUTS				CASCADING INPUTS			OUTPUTS		
A ₃ ,B ₃	A ₂ ,B ₂	A ₁ ,B ₁	A ₀ ,B ₀	I _{A>B}	I _{A<B}	I _{A=B}	O _{A>B}	O _{A<B}	O _{A=B}
A ₃ >B ₃	X	X	X	X	X	X	H	L	L
A ₃ <B ₃	X	X	X	X	X	X	L	H	L
A ₃ =B ₃	A ₂ >B ₂	X	X	X	X	X	H	L	L
A ₃ =B ₃	A ₂ <B ₂	X	X	X	X	X	L	H	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ >B ₁	X	X	X	X	H	L	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ <B ₁	X	X	X	X	L	H	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ >B ₀	X	X	X	H	L	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ <B ₀	X	X	X	L	H	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	H	L	L	H	L	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	L	H	L	L	H	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	X	X	H	L	L	H
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	H	H	L	L	L	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	L	L	L	H	H	L

74LS85比较器



多位比较器扩展举例

- 例，用两片74LS585组成一个8位数值比较器
- 根据多位比较的规则，在高位相等时取决于低位的比较结果
- 因此只要将两个数
 - 高4位 $A_7A_6A_5A_4$ 和 $B_7B_6B_5B_4$ 接到第2片74LS85上
 - 将低4位的 $A_3A_2A_1A_0$ 和 $B_3B_2B_1B_0$ 接到第1片74LS85上
 - 把第1片的 $Y_{(A>B)}$ 、 $Y_{(A<B)}$ 和 $Y_{(A=B)}$ 接到第2片 $I_{(A>B)}$ 、 $I_{(A<B)}$ 和 $I_{(A=B)}$ 即可
 - 因为第1片74LS85没有来自低位的比较信号输入，所以将它的 $I_{(A>B)}$ 和 $I_{(A<B)}$ 接"0"， $I_{(A=B)}$ 接"1"



问题和建议?

