



三、组合逻辑电路

黄慎宜

2025年11月16日

School of Microelectronics, University of Science and Technology of China



◆ 考点总结

◆ 习题讲解

◆ 补充题

考点总结--组合逻辑电路分析

1. 电路功能分析

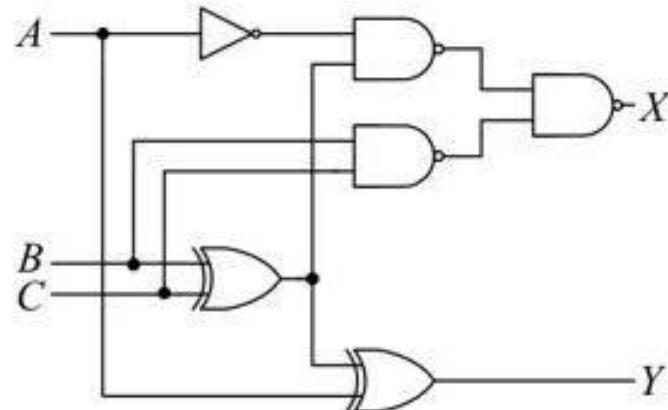
✓ 分析步骤

- A. 根据电路图，写出逻辑函数式
- B. 然后根据逻辑函数式列出真值表
- C. 观察真值表，判断电路的功能

➤ 常见的电路功能有以下4类：

- A. 代码转换
- B. 补码运算
- C. 二进制算数运算
- D. 其他：如比较器、数据选择器等等

Q1: (2017·期末) 分析下
图中电路的逻辑功能



1. 电路功能分析

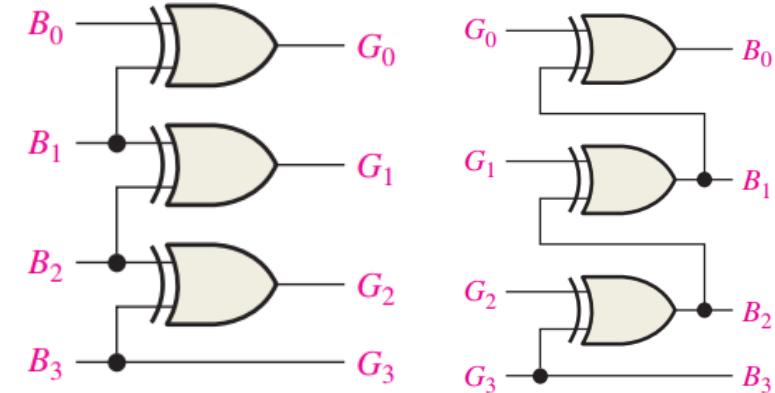
✓ 分析步骤

- A. 根据电路图，写出逻辑函数式
- B. 然后根据逻辑函数式列出真值表
- C. 观察真值表，判断电路的功能

A. 代码转换

- 一般输入和输出个数相同
- 不存在不同输入对应相同输出

Q2：分析下图中电路的逻辑功能



考点总结--组合逻辑电路分析

1. 电路功能分析

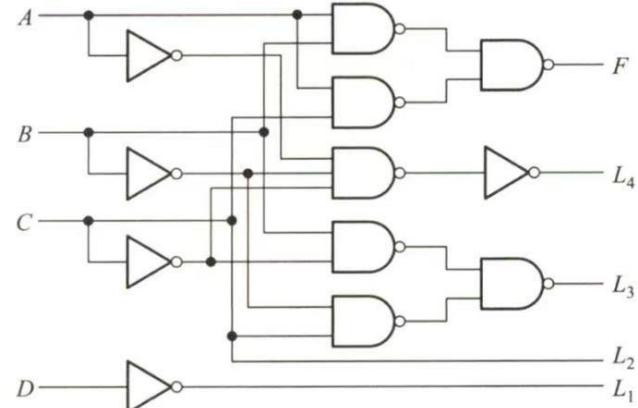
✓ 分析步骤

- 根据电路图，写出逻辑函数式
- 然后根据逻辑函数式列出真值表
- 观察真值表，判断电路的功能

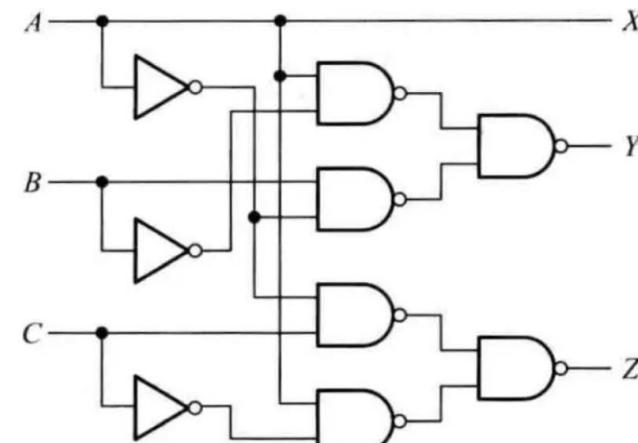
B. 补码运算

- 一般输入和输出个数相同
- 输入和输出的和为定值
- 注意：有时候可能是带符号数运算

Q3：分析下图电路的逻辑功能



Q4：分析下图电路的逻辑功能



1. 电路功能分析

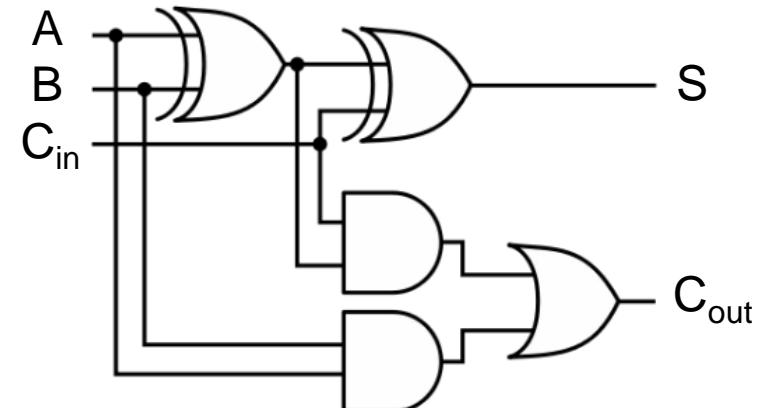
✓ 分析步骤

- A. 根据电路图，写出逻辑函数式
- B. 然后根据逻辑函数式列出真值表
- C. 观察真值表，判断电路的功能

C. 二进制算数运算

- 输入输出个数上没有固定规律
- 分别对加减乘三种运算进行分析
- 注意可能需要考虑进位
 - 如半加器、全加器等

Q5：分析下图中电路的逻辑功能



考点总结--组合逻辑电路分析

1. 电路功能分析

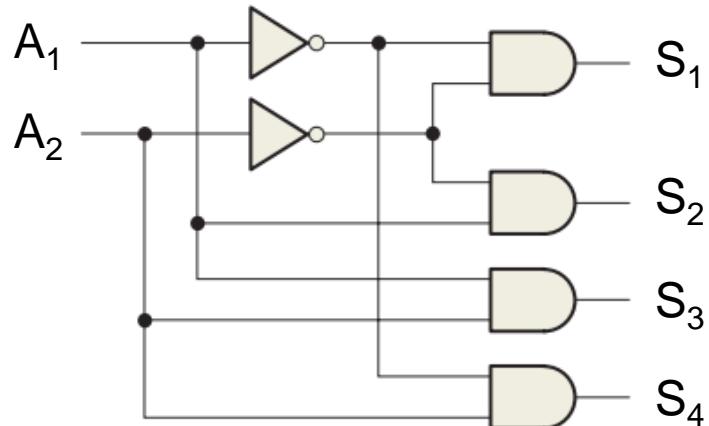
✓ 分析步骤

- A. 根据电路图，写出逻辑函数式
- B. 然后根据逻辑函数式列出真值表
- C. 观察真值表，判断电路的功能
- D. 其他

常见类型：数据选择器、比较器、
编码器/译码器等

- 输出通常只有一个
- 直接观察真值表，然后用自然语言描述其功能即可
- 注意“伪码”的处理

Q6：分析下图中电路的逻辑功能



考点总结--组合逻辑电路分析

2. 竞争-冒险现象

- 竞争：门电路两个输入信号同时向相反的逻辑电平跳变的现象
- 竞争-冒险：因竞争而在电路输出端可能产生尖峰脉冲的现象
- 消除方法：
 - ✓ 接入滤波电容
 - ✓ 引入选通脉冲
 - ✓ 修改逻辑设计

判断是否存在竞争-冒险：
如果输出端逻辑函数在一定条件下可以简化为：

- ✓ $Y = A + A'$
- ✓ 或者 $Y = A \cdot A'$

则可判定存在竞争-冒险

Q7: (2025·复旦) 逻辑函数 $F = X'_2X_3X'_4 + X_1X_3X_4 + X_2X_4$, 当输入信号____的逻辑值从__跳变到__时, 会发生冒险。逻辑表达式再加一项_____可消除冒险。



考点总结--组合逻辑电路设计

3. 常用的组合逻辑电路模块总结

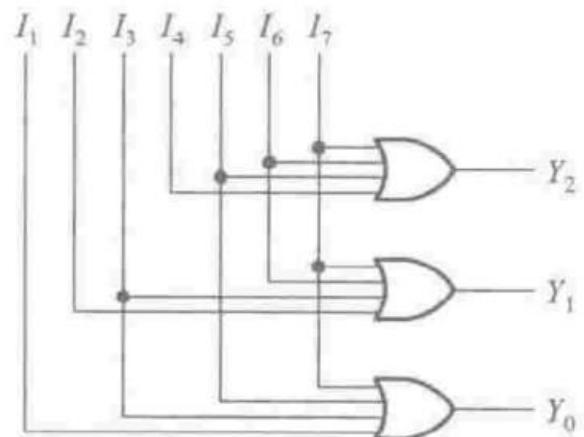
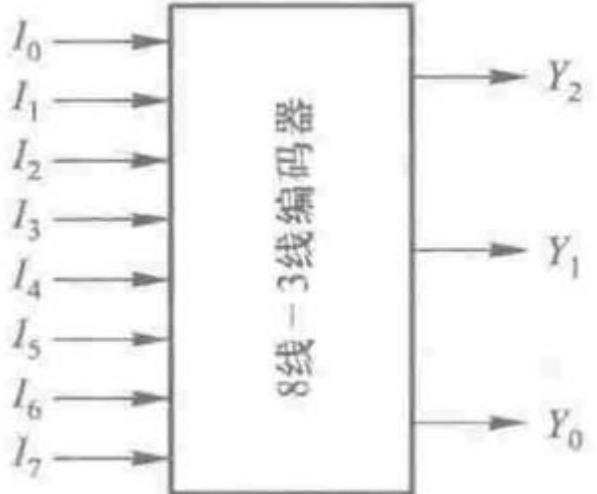
普通编码器：

输入								输出		
I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7	Y_2	Y_1	Y_0
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

$$\left\{ \begin{array}{l} Y_2 = I'_0 I'_1 I'_2 I'_3 I'_4 I'_5 I'_6 I'_7 + I'_0 I'_1 I'_2 I'_3 I'_4 I'_5 I'_6 I'_7 \\ \quad + I'_0 I'_1 I'_2 I'_3 I'_4 I'_5 I'_6 I'_7 + I'_0 I'_1 I'_2 I'_3 I'_4 I'_5 I'_6 I'_7 \\ Y_1 = I'_0 I'_1 I'_2 I'_3 I'_4 I'_5 I'_6 I'_7 + I'_0 I'_1 I'_2 I'_3 I'_4 I'_5 I'_6 I'_7 \\ \quad + I'_0 I'_1 I'_2 I'_3 I'_4 I'_5 I'_6 I'_7 + I'_0 I'_1 I'_2 I'_3 I'_4 I'_5 I'_6 I'_7 \\ Y_0 = I'_0 I'_1 I'_2 I'_3 I'_4 I'_5 I'_6 I'_7 + I'_0 I'_1 I'_2 I'_3 I'_4 I'_5 I'_6 I'_7 \\ \quad + I'_0 I'_1 I'_2 I'_3 I'_4 I'_5 I'_6 I'_7 + I'_0 I'_1 I'_2 I'_3 I'_4 I'_5 I'_6 I'_7 \end{array} \right.$$

仅有一个输入
取“1”

$$\left\{ \begin{array}{l} Y_2 = I_4 + I_5 + I_6 + I_7 \\ Y_1 = I_2 + I_3 + I_6 + I_7 \\ Y_0 = I_1 + I_3 + I_5 + I_7 \end{array} \right.$$



考点总结--组合逻辑电路设计

3. 常用的组合逻辑电路模块总结

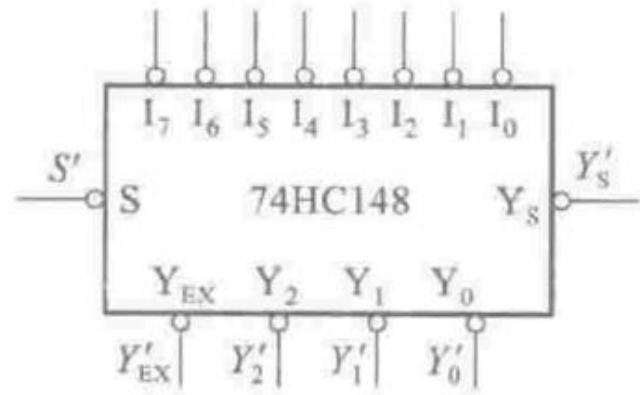
优先编码器 (74HC148) :

- ✓ 允许同时输入多个编码信号
- ✓ 只对其中优先权最高的一个进行编码

功能表:

输入									输出				
S'	I'_0	I'_1	I'_2	I'_3	I'_4	I'_5	I'_6	I'_7	Y'_2	Y'_1	Y'_0	Y'_S	Y'_{EX}
1	x	x	x	x	x	x	x	x	1	1	1	1	1
0	1	1	1	1	1	1	1	1	1	1	1	0	1
0	x	x	x	x	x	x	x	0	0	0	0	1	0
0	x	x	x	x	x	x	0	1	0	0	1	1	0
0	x	x	x	x	x	0	1	1	0	1	0	1	0
0	x	x	x	x	0	1	1	1	0	1	1	1	0
0	x	x	x	0	1	1	1	1	1	0	0	1	0
0	x	x	0	1	1	1	1	1	1	0	1	1	0
0	x	0	1	1	1	1	1	1	1	1	0	1	0
0	0	1	1	1	1	1	1	1	1	1	1	1	0

I'_7 优先级最高; I'_0 优先级最低



	0	1
S' 选通输入端	编码器正常工作	不工作
Y'_S	电路工作 但无编码输入	
Y'_{EX}	电路工作 且有编码输入	

考点总结--组合逻辑电路设计

3. 常用的组合逻辑电路模块总结

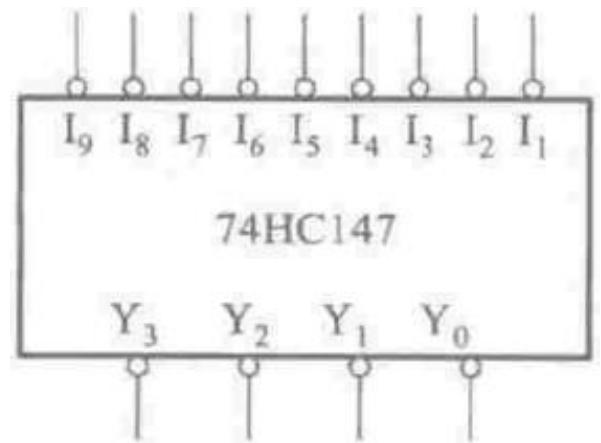
二十进制优先编码器 (74HC147) :

- ✓ 和74HC148功能大致相同
- ✓ 同样是低电平有效

功能表:

输入										输出			
I'_1	I'_2	I'_3	I'_4	I'_5	I'_6	I'_7	I'_8	I'_9		Y'_3	Y'_2	Y'_1	Y'_0
1	1	1	1	1	1	1	1	1	1	1	1	1	1
x	x	x	x	x	x	x	x	x	0	0	1	1	0
x	x	x	x	x	x	x	x	0	1	0	1	1	1
x	x	x	x	x	x	0	1	1	1	1	0	0	0
x	x	x	x	x	0	1	1	1	1	1	0	0	1
x	x	x	x	0	1	1	1	1	1	1	0	1	0
x	x	x	0	1	1	1	1	1	1	1	0	1	1
x	0	1	1	1	1	1	1	1	1	1	1	0	0
0	1	1	1	1	1	1	1	1	1	1	1	1	0

I'_9 优先级最高; I'_0 优先级最低



注: 当 $I'_1 \sim I'_9$ 均为无效输入时, 隐含表示了 I'_0 为有效输入, 编码输出为1111

考点总结--组合逻辑电路设计

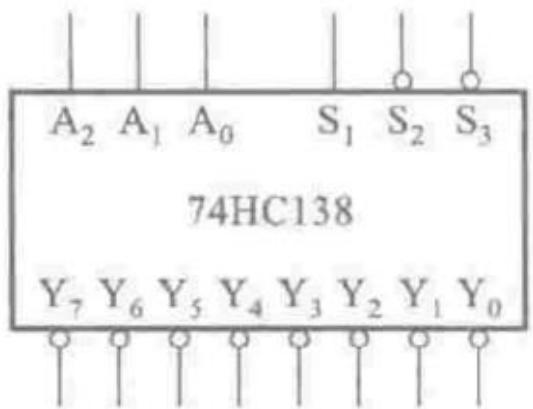
3. 常用的组合逻辑电路模块总结

译码器 (74HC138) :

功能表:

输入					输出							
S_1	$S'_2 + S'_3$	A_2	A_1	A_0	Y'_0	Y'_1	Y'_2	Y'_3	Y'_4	Y'_5	Y'_6	Y'_7
0	x	x	x	x	1	1	1	1	1	1	1	1
x	1	x	x	x	1	1	1	1	1	1	1	1
1	0	0	0	0	0	1	1	1	1	1	1	1
1	0	0	0	1	1	0	1	1	1	1	1	1

输入					输出							
S_1	$S'_2 + S'_3$	A_2	A_1	A_0	Y'_0	Y'_1	Y'_2	Y'_3	Y'_4	Y'_5	Y'_6	Y'_7
1	0	0	1	0	1	1	0	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1	1	1
1	0	1	0	0	1	1	1	1	0	1	1	1
1	0	1	0	1	1	1	1	1	1	0	1	1
1	0	1	1	0	1	1	1	1	1	1	0	1
1	0	1	1	1	1	1	1	1	1	1	1	0



当 S_1 、 S'_2 和 S'_3 同时满足：

✓ $S'_1 = 1$

✓ $S'_2 + S'_3 = 0$

译码器处于工作状态；

否则，译码器被禁止

S_1 、 S'_2 和 S'_3 被称为片选输入端，用于控制芯片是否工作，可以用来实现芯片的功能扩展

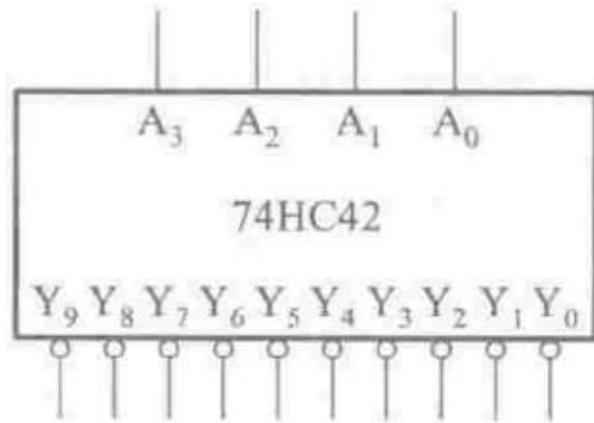
考点总结--组合逻辑电路设计

3. 常用的组合逻辑电路模块总结

二十进制译码器 (74HC42) :

功能表:

序号	输入				输出									
	A_3	A_2	A_1	A_0	Y'_0	Y'_1	Y'_2	Y'_3	Y'_4	Y'_5	Y'_6	Y'_7	Y'_8	Y'_9
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
伪 码	1	0	1	0	1	1	1	1	1	1	1	1	1	1
	1	0	1	1	1	1	1	1	1	1	1	1	1	1
	1	1	0	0	1	1	1	1	1	1	1	1	1	1
	1	1	0	1	1	1	1	1	1	1	1	1	1	1
	1	1	1	0	1	1	1	1	1	1	1	1	1	1
	1	1	1	1	1	1	1	1	1	1	1	1	1	1



$$\begin{cases} Y'_0 = (A'_3 A'_2 A'_1 A'_0)' & Y'_5 = (A'_3 A_2 A'_1 A_0)' \\ Y'_1 = (A'_3 A'_2 A'_1 A_0)' & Y'_6 = (A'_3 A_2 A_1 A'_0)' \\ Y'_2 = (A'_3 A'_2 A_1 A'_0)' & Y'_7 = (A'_3 A_2 A_1 A_0)' \\ Y'_3 = (A'_3 A'_2 A_1 A_0)' & Y'_8 = (A_3 A'_2 A'_1 A'_0)' \\ Y'_4 = (A'_3 A_2 A'_1 A'_0)' & Y'_9 = (A_3 A'_2 A'_1 A_0)' \end{cases}$$



对于BCD代码以外的伪码，输出均无低电平信号产生，译码器拒绝翻译，所以这个电路有**拒绝伪码**的功能

考点总结--组合逻辑电路设计

3. 常用的组合逻辑电路模块总结

显示译码器（7448）：

- 数字显示器的原理



LT'	灯测试信号	=0时全亮
RBI'	灭零输入	=0且输入为0时可以让显示的0熄灭
BI'	灭灯输入	=0时所有数码管熄灭
RBO'	灭零输出端	=0表示已将本该显示的0熄灭

功能表：

数字	输入					输出							字形
	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	1	0	0
1	0	0	0	1	0	1	1	0	0	0	0	0	1
2	0	0	1	0	1	1	0	1	1	0	0	1	1
3	0	0	1	1	1	1	1	1	0	0	0	1	1
4	0	1	0	0	0	1	1	0	0	0	1	1	1
5	0	1	0	1	1	0	1	1	0	1	1	1	1
6	0	1	1	0	0	0	1	1	1	1	1	1	1
7	0	1	1	1	1	1	1	0	0	0	0	0	0
8	1	0	0	0	1	1	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	0	0	1	1	1	1
10	1	0	1	0	0	0	0	1	1	0	1	1	1
11	1	0	1	1	0	0	1	1	0	0	1	1	1
12	1	1	0	0	0	1	0	0	0	1	1	1	1
13	1	1	0	1	1	0	0	1	0	1	1	1	1
14	1	1	1	0	0	0	0	1	1	1	1	1	1
15	1	1	1	1	0	0	0	0	0	0	0	0	0

3. 常用的组合逻辑电路模块总结

数据选择器（二选一数据选择器）：

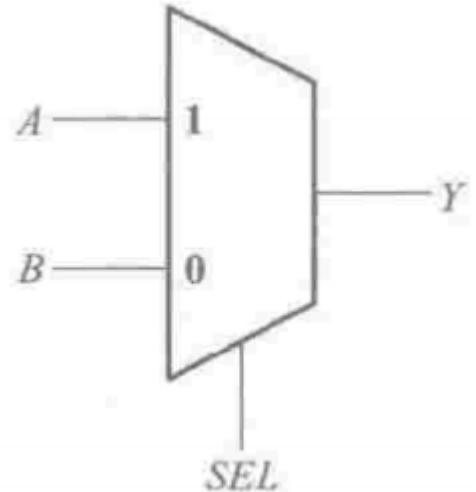
真值表：

SEL	A	B	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

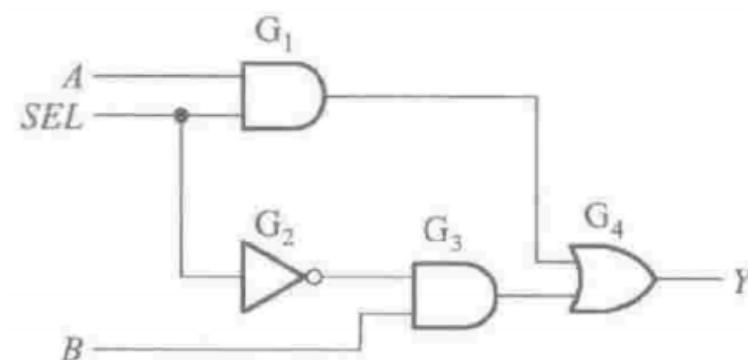


$$Y = SEL \cdot A + SEL' \cdot B$$

逻辑图像符号：



内部电路图：

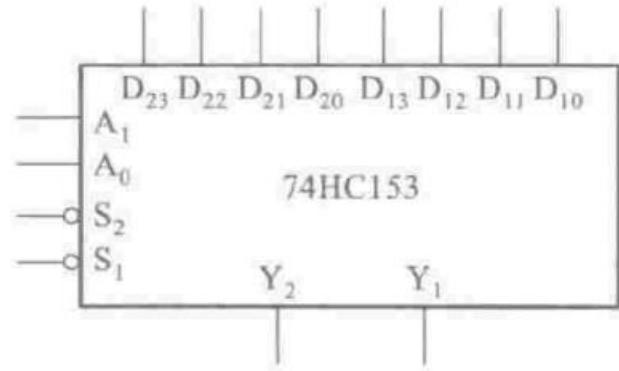
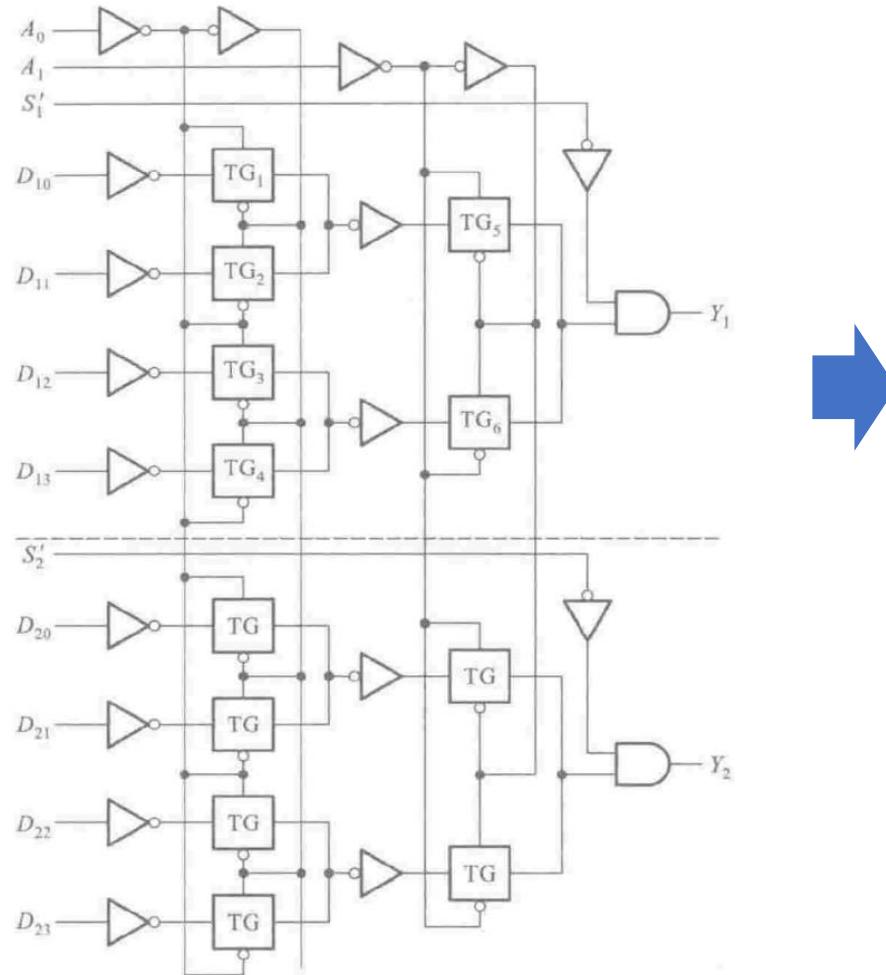


考点总结--组合逻辑电路设计

3. 常用的组合逻辑电路模块总结

数据选择器（双四选一数据选择器，74HC153）：

内部电路图：



功能表（对于 Y_2 同理）：

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

$S'_1 = 0$ 电路正常工作；

$S'_1 = 1$ 禁止工作，输出被封锁在高电平

考点总结--组合逻辑电路设计

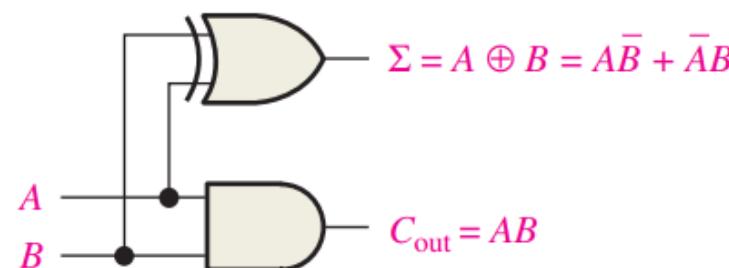
3. 常用的组合逻辑电路模块总结

1位半加器（不考虑进位输入）：

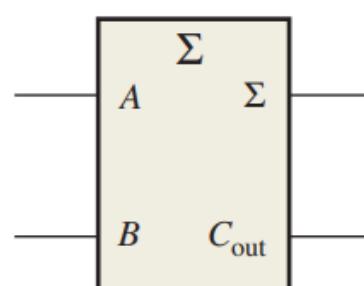
真值表：

A	B	C _{out}	Σ
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

逻辑图：



符号：

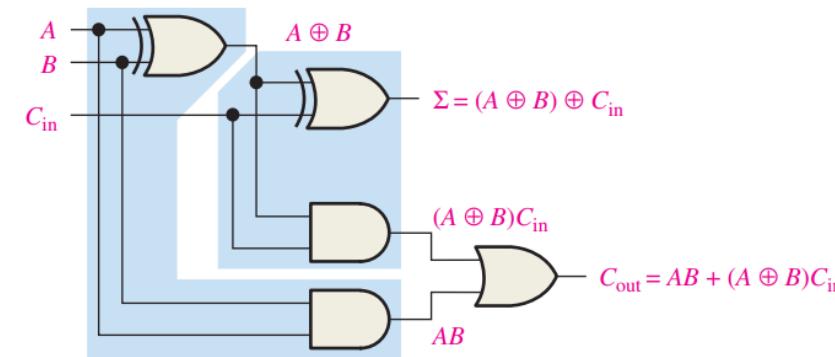


1位全加器（考虑进位输入）：

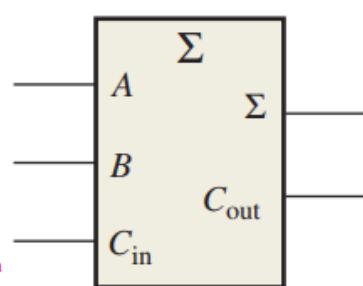
真值表：

A	B	C _{in}	C _{out}	Σ
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

逻辑图：



符号：

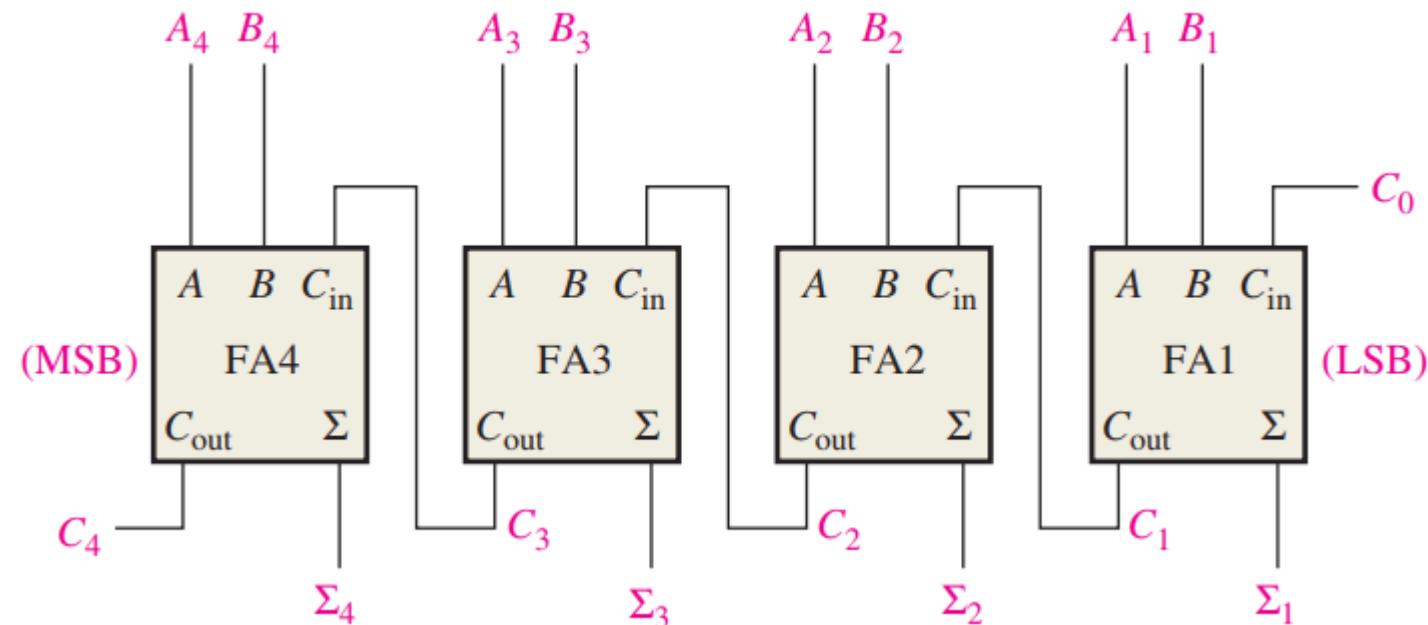


考点总结--组合逻辑电路设计

3. 常用的组合逻辑电路模块总结

串行进位加法器：

电路图：



真值表：

C_{n-1}	A_n	B_n	Σ_n	C_n
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

缺点：速度慢

- 做一次加法运算需要经过4个全加器的传输延迟时间

考点总结--组合逻辑电路设计

3. 常用的组合逻辑电路模块总结

超前进位加法器 (74HC283) :

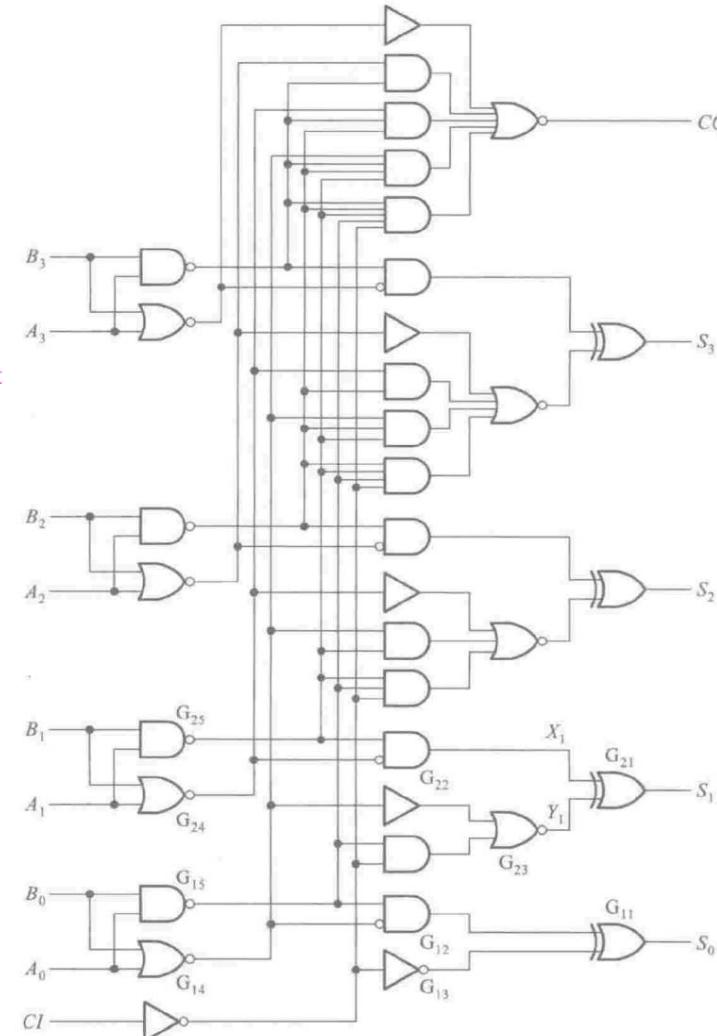
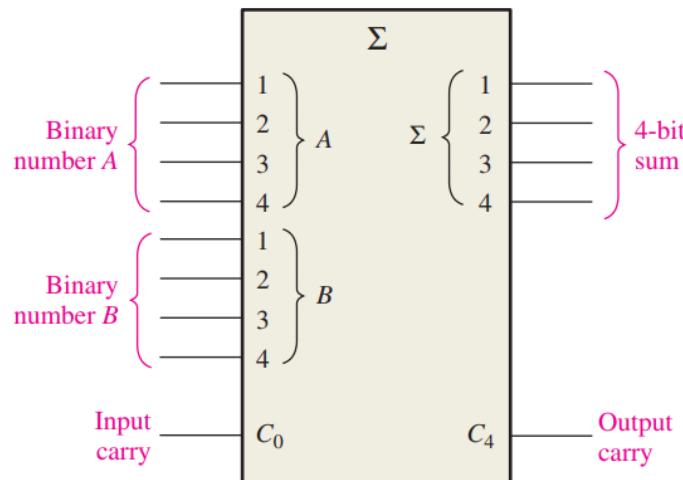
原理:

- ① 加到第*i*位的进位信号是两个加数第*i*位一下各位状态的函数
- ② 第*i*位进位输入信号一定能由 $A_{i-1}A_{i-2} \dots A_0$ 和 $B_{i-1}B_{i-2} \dots B_0$ 唯一确定
- ③ 可以通过逻辑电路事先得出每一位全加器的进位输入信号，而无需逐位传递进位信号

✓ 优点: 延迟小, 计算速度快

➤ 缺点: 电路的复杂程度高

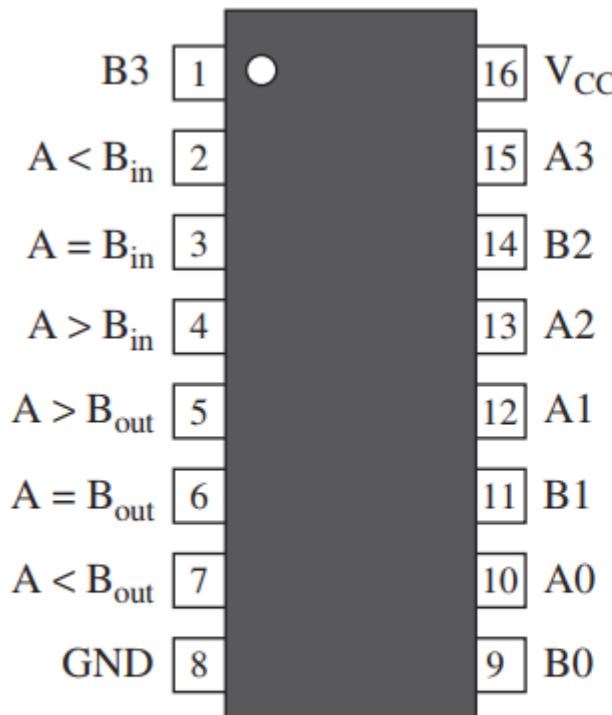
电路图:



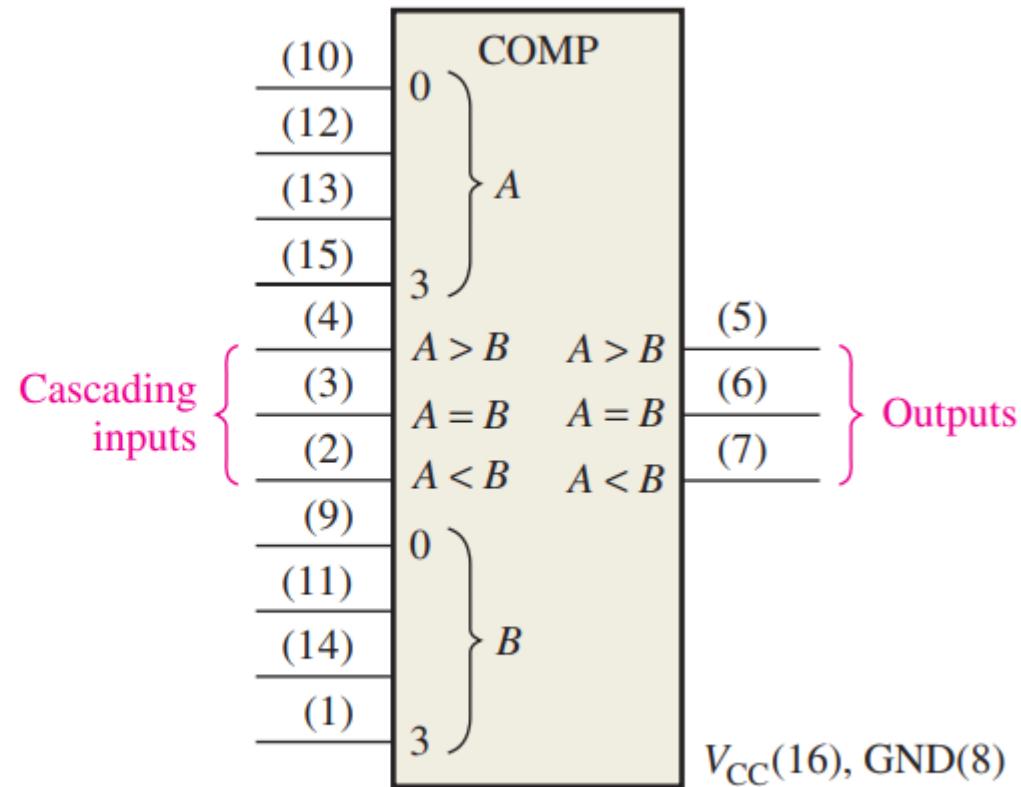
考点总结--组合逻辑电路设计

3. 常用的组合逻辑电路模块总结

数值比较器 (74HC85/74LS85) :



(a) Pin diagram



(b) Logic symbol

考点总结--组合逻辑电路设计

4. 门电路级组合逻辑电路设计

✓ 分析步骤

A. 逻辑抽象

- 确定逻辑变量数及其描述

B. 自然语言描述

C. 真值表

D. 逻辑函数式

E. 逻辑电路图

- ✓ 该部分在前面章节已经有所涉及
- ✓ 需注意关注器件的选型，然后给出适当的表达

Q8：试用三个3输入端与门和一个或门及两个非门实现“ $A > B$ ”的比较电路，A和B均为两位二进制数。



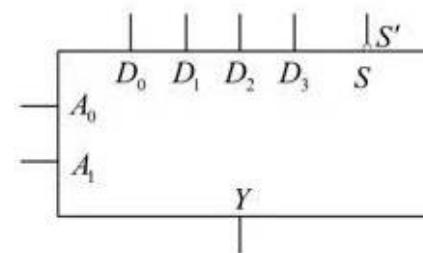
考点总结--组合逻辑电路设计

5. 基于中规模电路的设计问题

- ✓ 分析步骤和门电路级设计问题类似
- 可用于设计任何逻辑电路的模块（芯片）
只有译码器和数据选择器
 - 输出恰好可以看成输入的所有最小项的集合
- 译码器
 - n 位输入译码器，可以用于生成变量数不超过 **n** 的任意逻辑函数
- 数据选择器
 - n 位地址输入的数据选择器，可以用于生成变量数不超过(**$n+1$**)的任意逻辑函数

Q9：(2017·期末) 用4选1数据选择器设计一个组合电路。输入ABC为三位二进制数，当输入能被3整除的时候输出Z=1，否则Z=0。数据选择器的功能表和框图如下。（注：0可被任何数整除）

S'	A_1	A_0	Y
1	×	×	0
0	0	0	D_0
0	0	1	D_1
0	1	0	D_2
0	1	1	D_3





◆ 知识点总结

◆ 习题讲解

◆ 补充题

1. (2020·期末) 分析下图电路逻辑功能

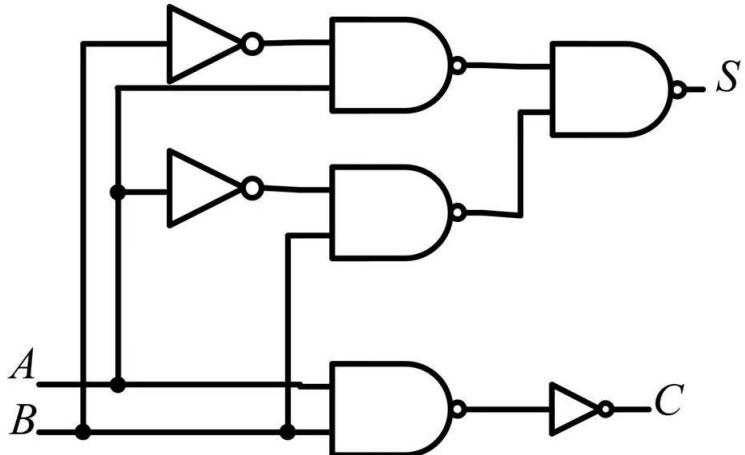
① 先写出输出表达式：

$$C = A \cdot B$$

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

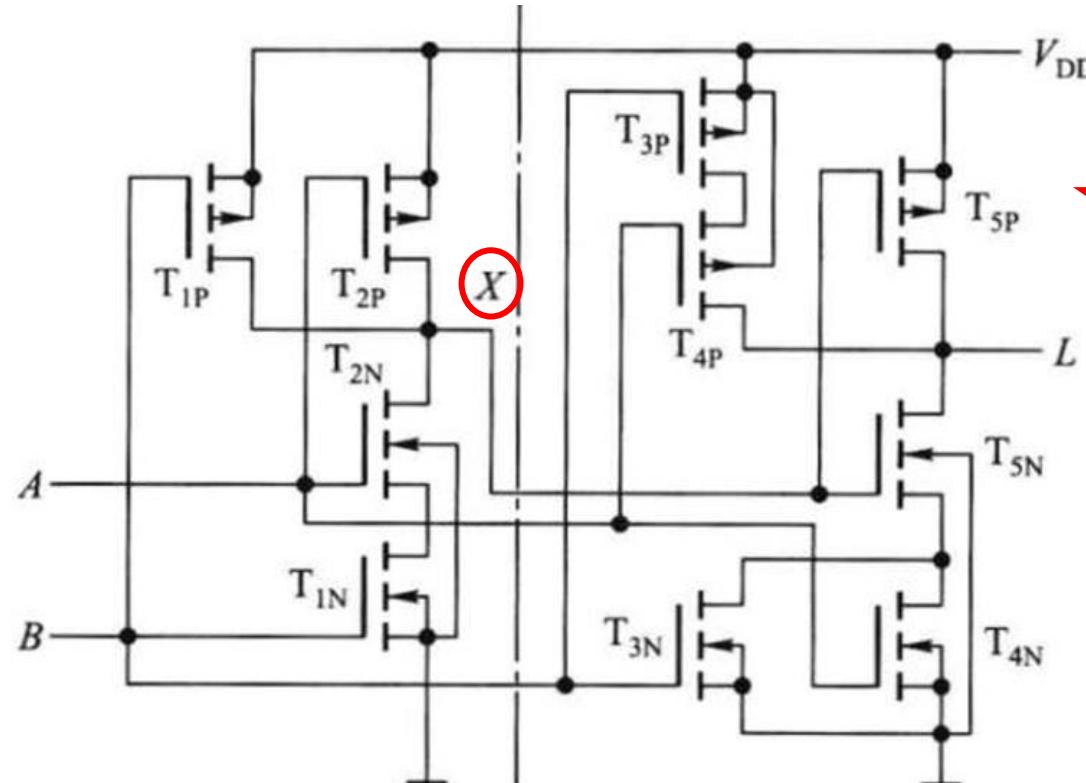
② 列出真值表

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



- ③ 可以发现，S表示A与B的和，C代表A和B相加时产生的进位
- 故该电路为一个半加器

1. (2020·期末) 分析下图电路逻辑功能



CMOS电路横向分级

★ 分析方法：

可以对电路横向分级之后，对每个基本模块分别进行分析

很明显，本电路可以沿虚线左右分为两级

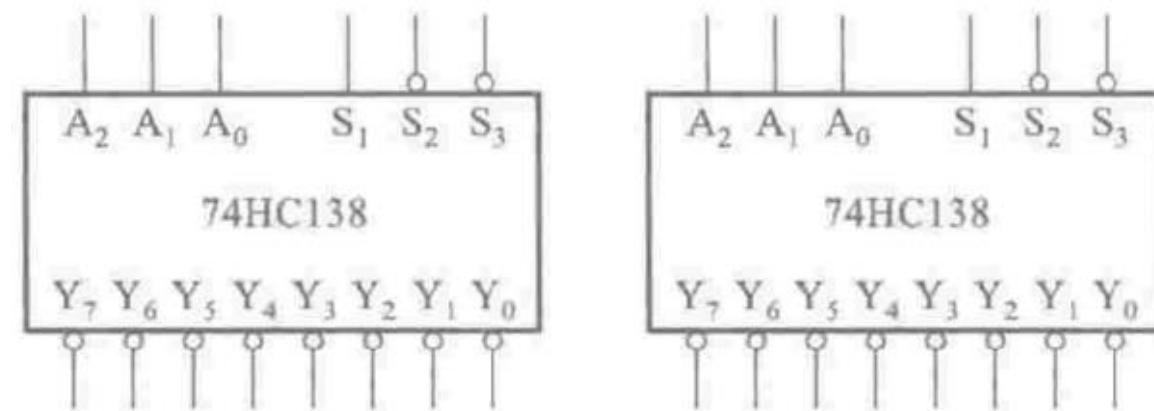
左侧电路很好分析：

根据所学知识，它是一个二输入与非门，故：

$$X = (AB)'$$

2. (2020·期末) 试用2片3线-8线译码器(74HC138)扩展成4线-16线译码器，将输入的二进制代码 $D_3D_2D_1D_0$ 译成低电平信号 $Z'_0 \sim Z'_{15}$ ，74HC138功能表与框图如下。

输入			输出									
S_1	$S'_2 + S'_3$	A_2	A_1	A_0	Y'_0	Y'_1	Y'_2	Y'_3	Y'_4	Y'_5	Y'_6	Y'_7
0	x	x	x	x	1	1	1	1	1	1	1	1
x	1	x	x	x	1	1	1	1	1	1	1	1
1	0	0	0	0	0	1	1	1	1	1	1	1
1	0	0	0	1	1	0	1	1	1	1	1	1
1	0	0	1	0	1	1	0	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1	1	1
1	0	1	0	0	1	1	1	1	0	1	1	1
1	0	1	0	1	1	1	1	1	1	0	1	1
1	0	1	1	0	1	1	1	1	1	1	0	1
1	0	1	1	1	1	1	1	1	1	1	1	0

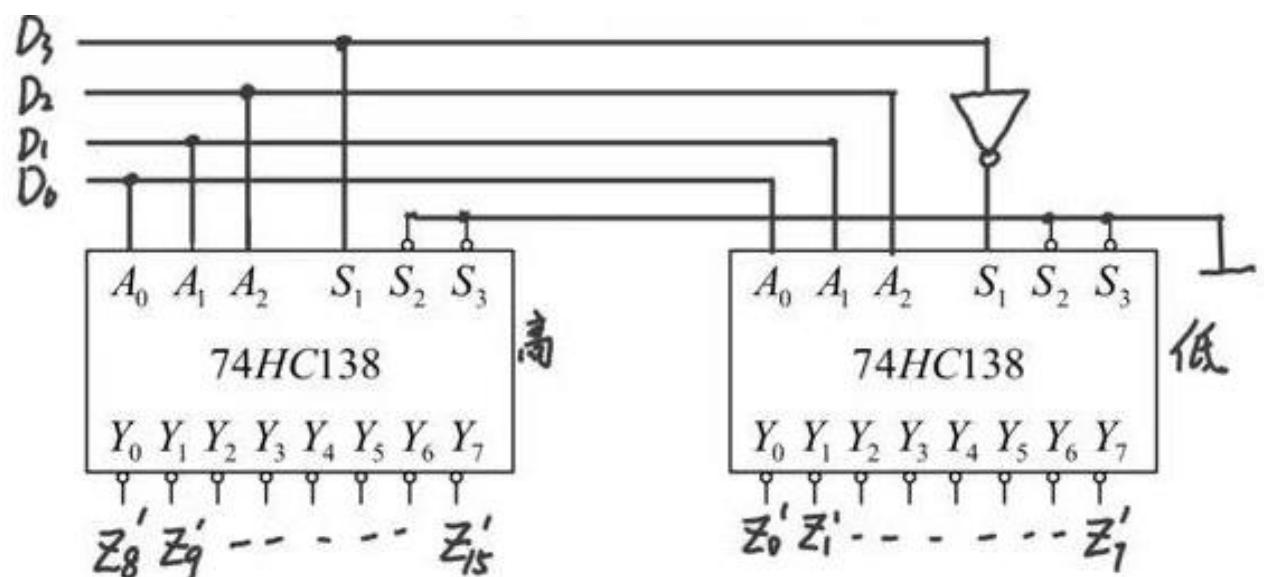


习题讲解

2. (2020·期末) 试用2片3线-8线译码器(74HC138)扩展成4线-16线译码器，将输入的二进制代码 $D_3D_2D_1D_0$ 译成低电平信号 $Z'_0 \sim Z'_{15}$ ，74HC138功能表与框图如下。

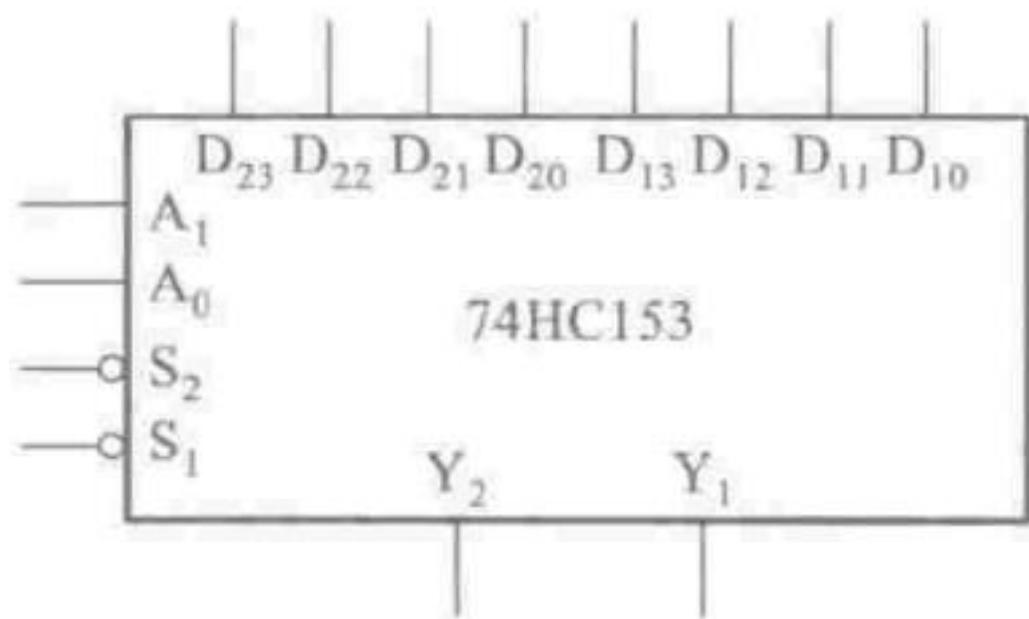
由于输入有4个变量，一片3-8译码器只有3个输入变量，我们需要将多出的一个输入变量作为片选信号

- ✓ 将 D_3 作为片选信号，使得 $D_3 = 0$ 时第一个译码器工作， $D_3 = 1$ 时第二个译码器工作即可



3. (2020·期末) 试用双4选1数据选择器74HC153实现逻辑函数 $Y = AC'D + A'B'CD + BC + A'B'C$, 在下图器件上画出完整电路图。4选1数据选择器功能表和双4选1器件框图如下。

S'	A_1	A_0	Y
1	x	x	0
0	0	0	D_0
0	0	1	D_1
0	1	0	D_2
0	1	1	D_3



3. (2020·期末) 试用双4选1数据选择器74HC153实现逻辑函数 $Y = AC'D + A'B'CD + BC + A'B'C$, 在下图器件上画出完整电路图。4选1数据选择器功能表和双4选1器件框图如下。

① 先将逻辑函数适当化简:

$$Y = AC'D + BC + A'C$$

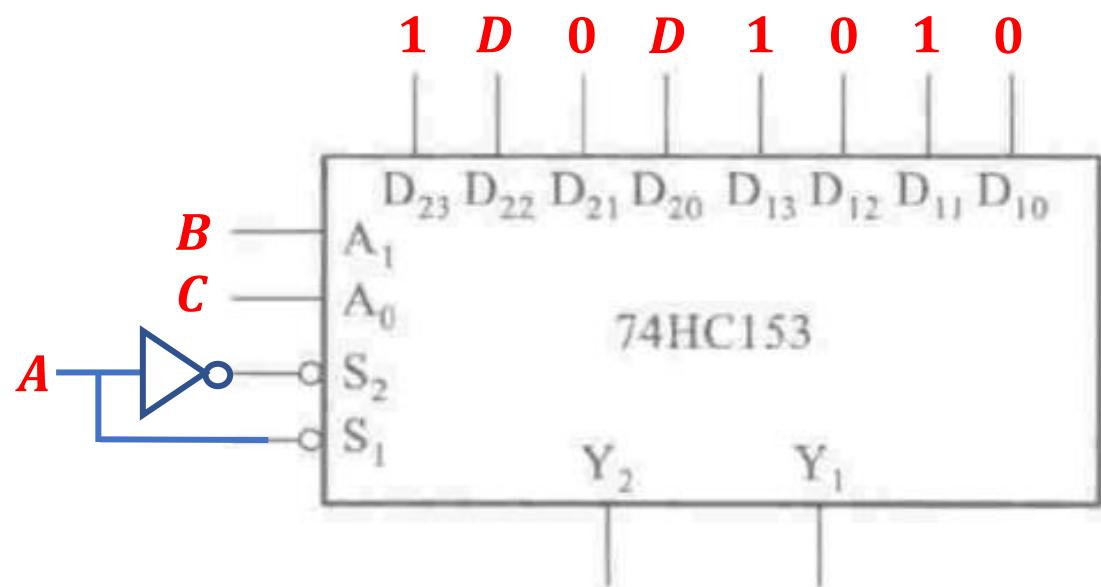
② 我们可以选 A 作为片选信号, 即:

通过 A 控制 S_1 , 通过 A' 控制 S_2

- $A = 0$ 时, $Y = C$

- $A = 1$ 时, $Y = C'D + BC$

③ 用 B 和 C 控制数据, 然后根据②的表达式填入对应变量即可





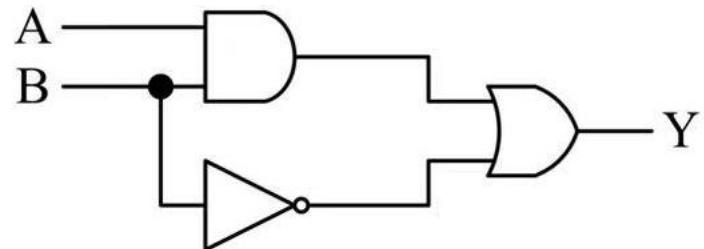
◆ 知识点总结

◆ 习题讲解

◆ 补充题

补充题

1. (2016·期末) 试分析下图中当A、B单独一个发生状态改变时，是否存在竞争-冒险现象？如果存在，那么发生在其他变量为何种取值的情况下？



2. (2020·期末) 某体育学校男生体能测试规定，3000米跑必须达标（10分钟以内），同时一下三项至少两项达标：引体向上20个、立定跳远2.5米、100米短跑12秒。针对该规则试对问题进行逻辑抽象并设计体测合格判定电路，给出：

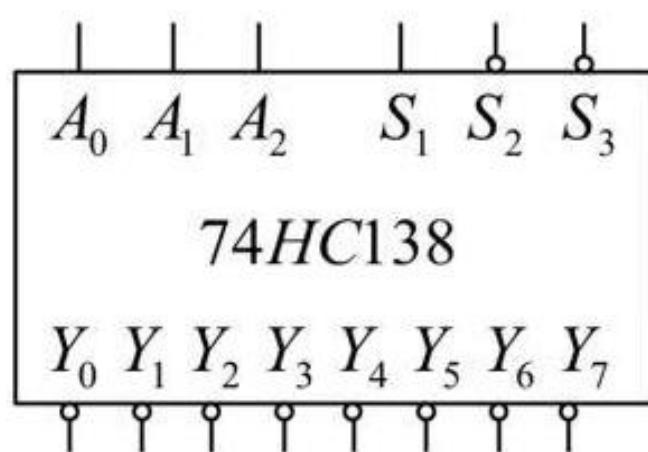
- ① 真值表
- ② 逻辑函数式（最简与或式）
- ③ 以与非门画出该逻辑电路

补充题

3. (2019·期末) 某实验室用两个灯显示三台设备 (A、B、C) 的故障情况, 当一台设备有故障时黄灯 (Z_1) 亮, 两台设备有故障时红灯 (Z_2) 亮, 三台设备同时故障时黄、红灯都亮。设计以逻辑电路实现以上功能。(注: 设备有故障用“1”表示, 无故障用“0”表示; 灯亮用“1”表示, 灯灭用“0”表示)

- ① 列出真值表, 求输出逻辑函数式
- ② 用3-8译码器74HC138和门电路实现该逻辑电路

输入			输出									
S_1	$S'_2 + S'_3$	A_2	A_1	A_0	Y'_0	Y'_1	Y'_2	Y'_3	Y'_4	Y'_5	Y'_6	Y'_7
0	x	x	x	x	1	1	1	1	1	1	1	1
x	1	x	x	x	1	1	1	1	1	1	1	1
1	0	0	0	0	0	1	1	1	1	1	1	1
1	0	0	0	1	1	0	1	1	1	1	1	1
1	0	0	1	0	1	1	0	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1	1	1
1	0	1	0	0	1	1	1	1	1	0	1	1
1	0	1	0	1	1	1	1	1	1	0	1	1
1	0	1	1	0	1	1	1	1	1	1	0	1
1	0	1	1	1	1	1	1	1	1	1	1	0
1	0	1	1	1	1	1	1	1	1	1	1	0



4. (2021·期末) 设计一组合电路，该电路有三个输入端A、B、C和一个输出端Z。当 $A \oplus B = (BC)'$ 时， $Z = 1$ 。

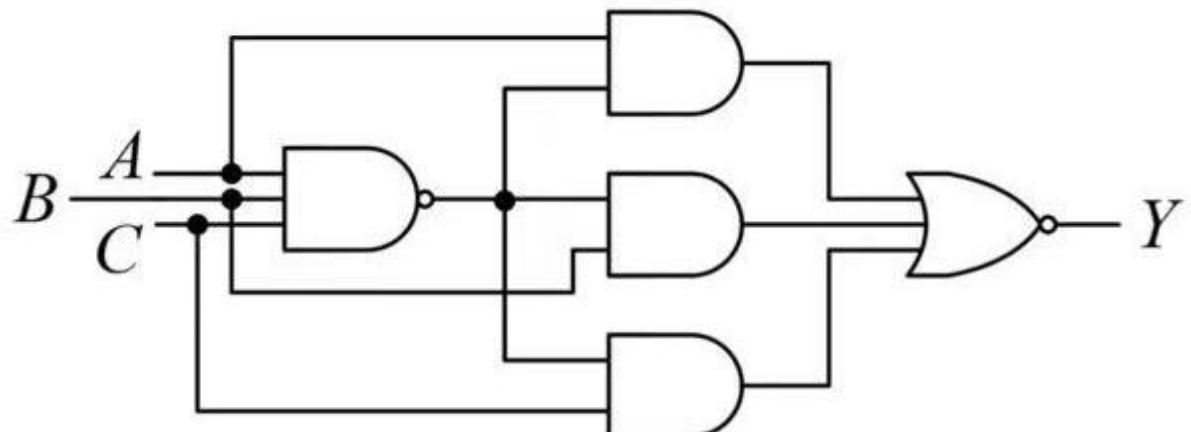
① 列出真值表

② 用4选1的数据选择器实现该电路，要求 $A_1A_0 = AB$

S'	A_1	A_0	Y
1	×	×	0
0	0	0	D_0
0	0	1	D_1
0	1	0	D_2
0	1	1	D_3

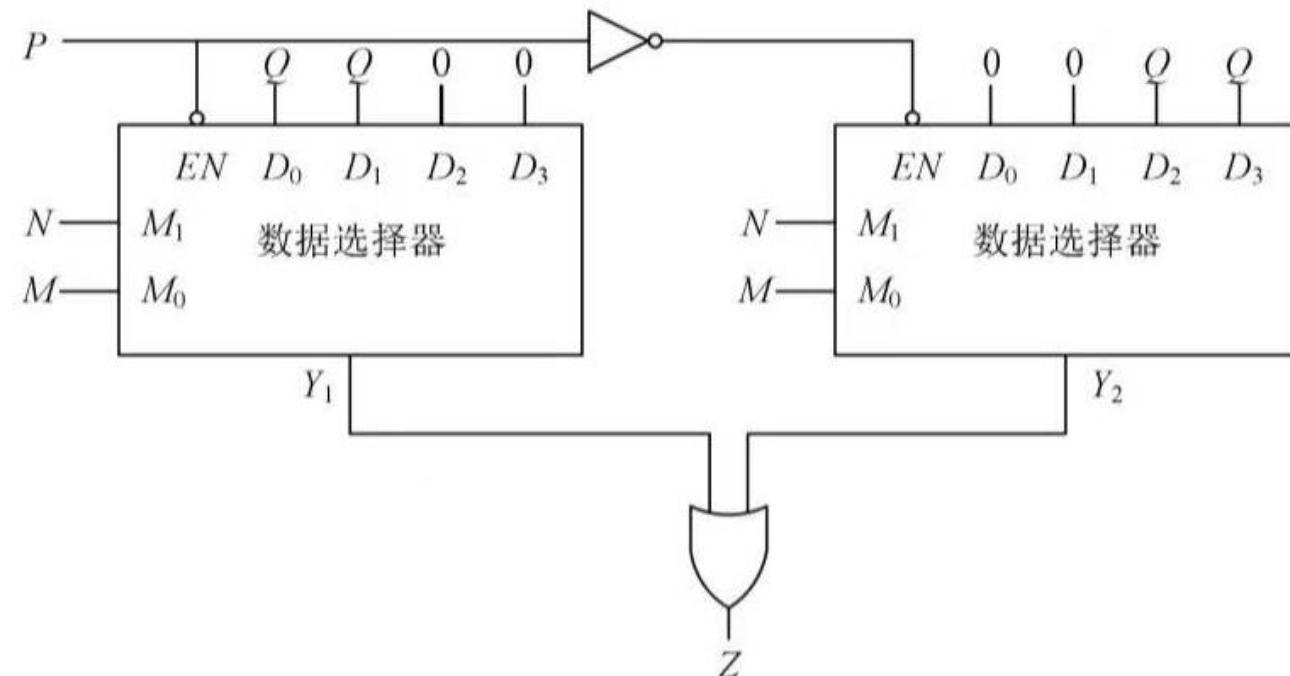
5. (2021·期末) 设分析如图所示组合逻辑电路。

- ① 写出输出逻辑函数式
- ② 画出波形图，说明电路功能



补充题

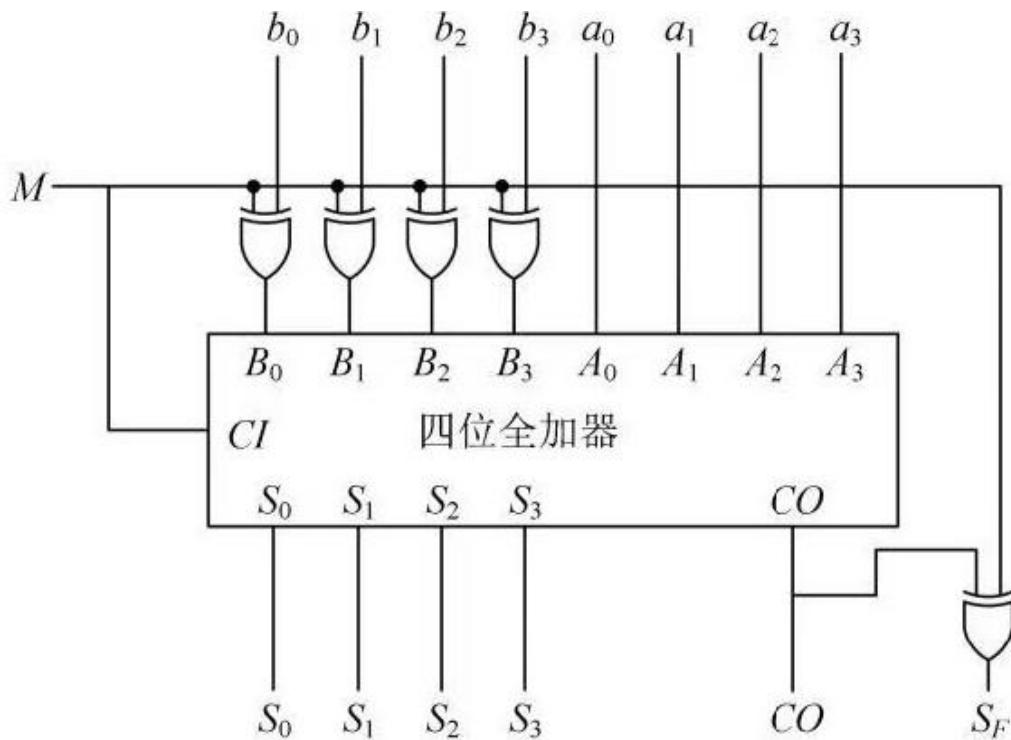
6. (2023·清华) 写出输出函数表达式并化简



7. (2023·清华) 根据如图所示电路回答以下问题

① 说明电路的作用

② 若 $a_3a_2a_1a_0 = 1001$, $b_3b_2b_1b_0 = 0011$, $M = 1$, 求输出数值



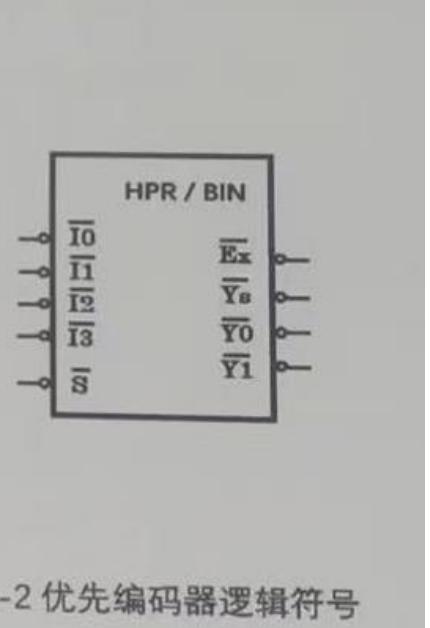
补充题



8. (2020·复旦) 4-2优先编码器的真值表和逻辑符号如下图所示。试用若干块4-2优先编码器和逻辑门，组成一个8-3优先编码器。要求8-3优先编码器的 $I'_7 \sim I'_0$ 位数据局输入端，其中 I'_7 优先级最高， I'_0 优先级最低； S' 为片选端， Y'_2 、 Y'_1 、 Y'_0 为数据输出端， Ex' 为扩展输出端， $Y_{s'}$ 为宣统输出端。请写出设计过程，并画出8-3优先编码器的真值表和电路图。

S'	$I0'$	$I1'$	$I2'$	$I3'$	$Y1'$	$Y0'$	Ex'	$Y_{s'}$
1	x	x	x	x	1	1	1	1
0	1	1	1	1	1	1	0	1
0	x	x	x	0	0	0	1	0
0	x	x	0	1	0	1	1	0
0	x	0	1	1	1	0	1	0
0	0	1	1	1	1	1	1	0

4-2 优先编码器真值表



4-2 优先编码器逻辑符号

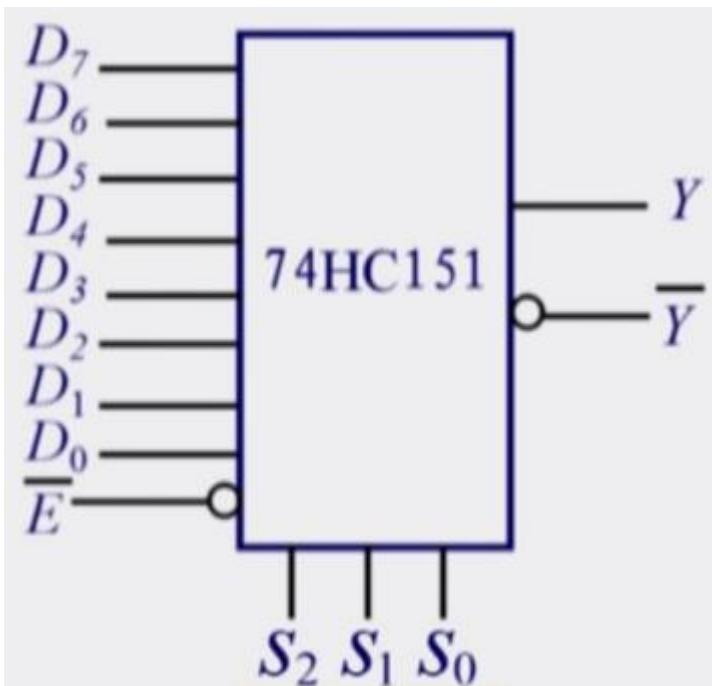
补充题



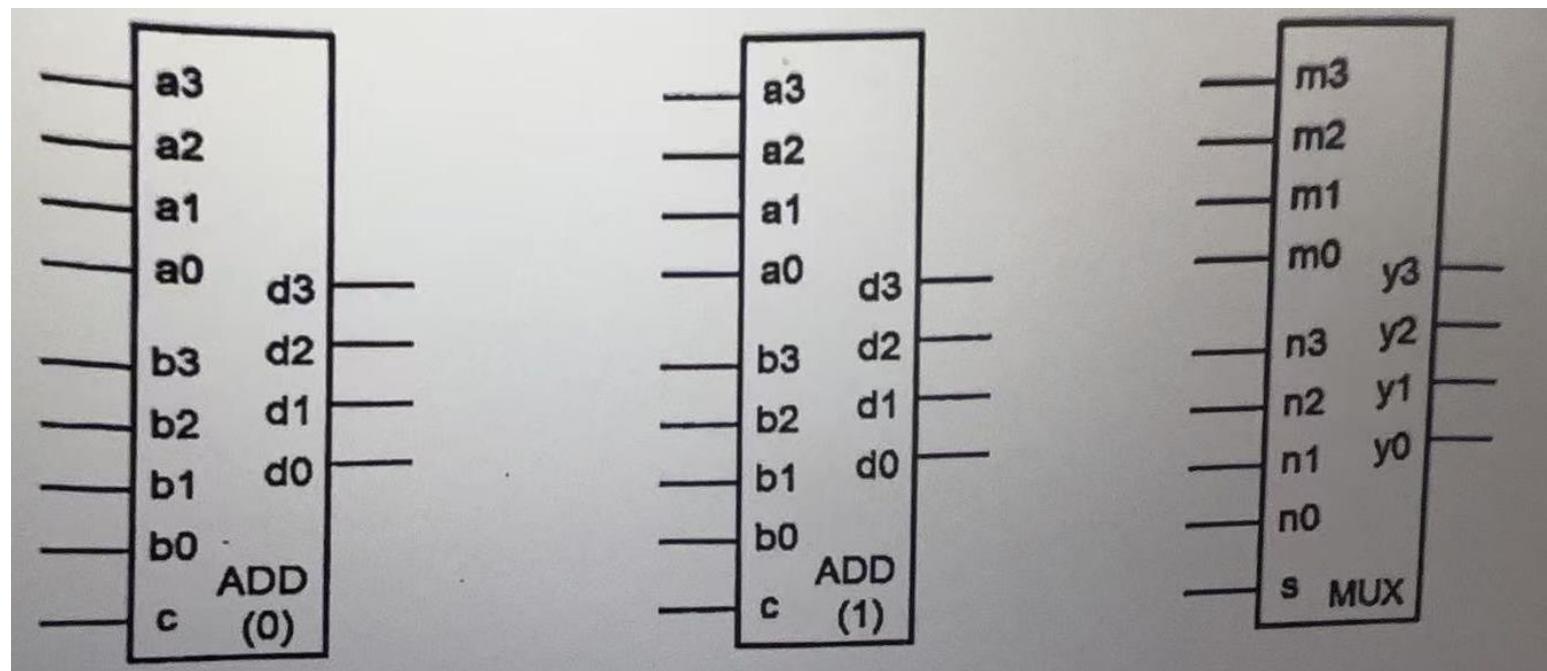
9. (2021·复旦) 全加器的输入为A、B、 C_{in} , 输出为S和 C_{out} 。用最简与或式表示S; 用最简与非-与非式表示 C_{out} 。

10. (2024·华科) 采用如图所示的低使能控制的8选1数据选择器(74HC151)设计一个组合逻辑电路，该电路有3个输入逻辑变量A、B、C和一个控制变量M。当M=0时，A、B、C三个输入端的状态完全相同时输出为1，否则输出为0。当M=1时，输出与A、B、C三个输入端占多数的状态保持一致。试求：

- ① 真值表
- ② 写出逻辑表达式
- ③ 画出逻辑电路图

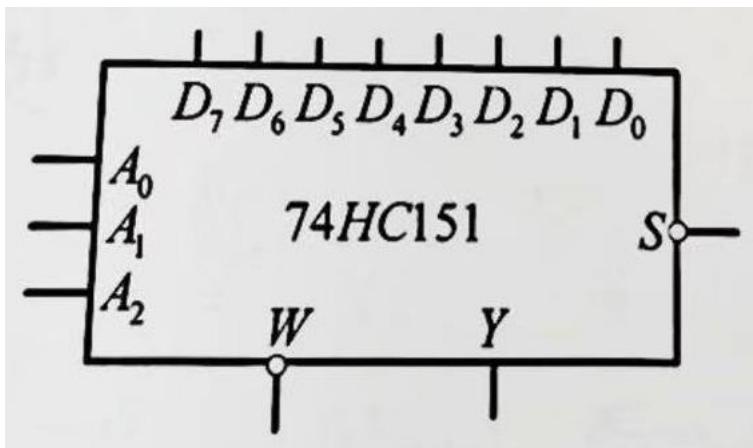


11. (2021·期末) 请使用最多 2 个 4 位加法器 (ADD) , 1 个 2 路 4 位数据选择器 (MUX) 和尽可能少的额外的逻辑门, 设计实现将 2 个 4 位原码有符号数 a 和 b 相加, 输出 4 位原码结果 y 和 1 位溢出标识 v 。其中, 部件 ADD 的功能: $d=a+b+c$; MUX 的功能: 当 $s=0$ 时, $y=m$; 当 $s=1$ 时, $y=n$ 。要求写出设计思路和涉及到的中间变量的逻辑式, 并画出电路图。



12. (2024·期末) 某图书馆上午8-12时，下午2-6时开馆，在开馆时间内图书馆门前的指示灯亮，试设计一个时钟控制指示灯亮灭的逻辑电路，允许输入到有反变量出现。**(提示：设输入信号ABCD为钟点变量，设T为区分午前、午后的标志变量， $T=0$ 表示1-12时， $T=1$ 表示13-24时，输出函数为F)。**要求：**用8选1数据选择器74HC151 (注： $W=Y'$) 来实现。**

输入				输出
S'	A_2	A_1	A_0	Y
1	x	x	x	0
0	0	0	0	D_0
0	0	0	1	D_1
0	0	1	0	D_2
0	0	1	1	D_3
0	1	0	0	D_4
0	1	0	1	D_5
0	1	1	0	D_6
0	1	1	1	D_7





感谢各位聆听！