

# 第4章

## 组合逻辑电路

### 本章知识结构

- 4.1 组合逻辑电路分析与设计
  - 4.1.1 组合逻辑电路的分析
  - 4.1.2 组合逻辑电路的设计
- 4.2 编码器
  - 4.2.1 普通编码器
  - 4.2.2 优先编码器
- 4.3 译码器
  - 4.3.1 二进制译码器
  - 4.3.2 二-十进制译码器
  - 4.3.3 数码显示器与显示译码器
- 4.4 数码管译码控制器的电路原理与实验
  - 4.4.1 电路原理
  - 4.4.2 实验操作
- 4.5 加法器
  - 4.5.1 半加器
  - 4.5.2 全加器
  - 4.5.3 多位加法器
- 4.6 数值比较器
  - 4.6.1 等值比较器
  - 4.6.2 数值比较器
- 4.7 数据选择器
  - 4.7.1 结构与原理
  - 4.7.2 常用数据选择器芯片
- 4.8 奇偶校验器
  - 4.8.1 奇偶校验原理
  - 4.8.2 奇偶校验器

组合逻辑电路又称组合电路，它任何时刻的输出只由当时的输入决定，而与电路的原始状态（以前的状态）无关，电路没有记忆功能。

常见的组合逻辑电路有编码器、译码器、加法器、数值比较器、数据选择器和奇偶校验器等。

## 4.1 组合逻辑电路分析与设计

组合逻辑电路的分析是指根据逻辑电路分析出它具有的功能；而设计则是指为了完成某些功能而设计出具体的逻辑电路来执行。

### 4.1.1 组合逻辑电路的分析

#### 1. 分析步骤

组合逻辑电路的分析一般按以下步骤进行：

- ① 根据逻辑电路写出逻辑表达式；
- ② 对逻辑表达式进行化简；
- ③ 根据化简后的表达式列出真值表；
- ④ 描述逻辑电路的功能（若功能较复杂，难以描述，该步骤可省略）。

#### 2. 分析举例

下面以图 4-1 所示电路为例来说明组合逻辑电路分析过程。

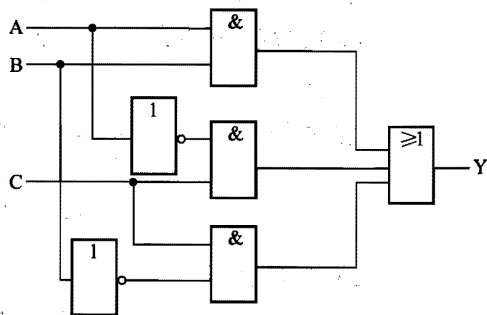


图 4-1 组合逻辑电路

分析过程如下：

（1）根据逻辑电路写出逻辑表达式

$$Y = AB + \bar{A}C + \bar{B}C$$



(2) 对逻辑表达式进行化简

$$\begin{aligned}
 Y &= AB + \bar{A}C + \bar{B}C \\
 &= AB + C(\bar{A} + \bar{B}) \quad (\text{根据公式 } \bar{A} + \bar{B} = \overline{AB}) \\
 &= AB + \overline{AB}C \quad (\text{根据公式 } A + \bar{A}B = A + B) \\
 &= AB + C
 \end{aligned}$$

(3) 根据化简后的表达式列出真值表。

真值表见表 4-1。

表 4-1  $Y=AB+C$  的真值表

A	B	C	Y	A	B	C	Y
0	0	0	0	1	0	0	0
0	0	1	1	1	0	1	1
0	1	0	0	1	1	0	1
0	1	1	1	1	1	1	1

(4) 描述逻辑电路的功能

从表 4-1 真值表可以看出, 图 4-1 所示电路的逻辑功能是: 当输入端 C 为 1 时, 输出端一定为 1; 当输入端 C 为 0 时, 只有 A、B 同时输入为 1, 输出端才会输出 1。

## 4.1.2 组合逻辑电路的设计

### 1. 设计步骤

组合逻辑电路的设计步骤如下:

- ① 根据实际问题需要实现的功能, 列出相应的真值表;
- ② 依据真值表写出逻辑表达式;
- ③ 化简逻辑表达式;
- ④ 根据化简后的逻辑表达式画出逻辑电路图。

### 2. 设计举例

下面举例来说明组合逻辑电路的设计。

某个运动会举行举重比赛, 比赛有 3 个裁判, A 为主裁判, B、C 为副裁判。举重是否成功由每个裁判按面前的按键来决定, 只有两个以上裁判 (其中必须有主裁判) 按下按钮确定成功时, 表明“成功”的灯才亮。请设计一个逻辑电路来实现上述功能。

设计过程如下:

(1) 根据实际问题需要实现的功能, 列出相应的真值表

根据上述问题，设 Y 为指示灯，1 表示灯亮，0 表示灯不亮；A 表示主裁判，B、C 表示两个副裁判，1 表示按键按下，0 表示按键未按下。列出的真值表见表 4-2。

表 4-2 举重裁判判定问题的真值表

A	B	C	Y	A	B	C	Y
0	0	0	0	1	0	0	0
0	0	1	0	1	0	1	1
0	1	0	0	1	1	0	1
0	1	1	0	1	1	1	1

## （2）根据真值表写出逻辑表达式

根据真值表写逻辑表达式的方法是：①从真值表上找出输出为 1 的各行，再把这些行的输入变量写成乘积的形式，如果变量值为 0，要在变量上加非；②把以上各行的乘积项相加，写出的逻辑表达式为

$$Y=A\bar{B}C+AB\bar{C}+ABC$$

## （3）化简逻辑表达式

$$\begin{aligned}
 Y &= A\bar{B}C+AB\bar{C}+ABC \\
 &= A\bar{B}C+AB(\bar{C}+C) \quad (\text{根据 } A+\bar{A}=1) \\
 &= A\bar{B}C+AB \\
 &= A(\bar{B}C+B) \quad (\text{根据 } A+\bar{A}B=A+B) \\
 &= A(B+C)
 \end{aligned}$$

## （4）根据化简后的逻辑表达式画出逻辑电路图

图 4-2 所示的逻辑电路能满足裁判判决的逻辑关系，但还不是一个可以实际应用的电路，在图 4-2 所示电路中再增加一些电路就可以构成具有实用价值的电路。举重比赛裁判裁决实用电路如图 4-3 所示。

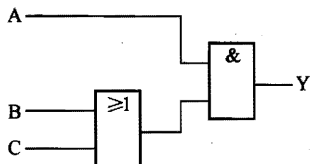


图 4-2  $Y=A(B+C)$  逻辑电路

在图 4-3 所示电路中，当按下按键  $S_A$  和  $S_B$  时，A、B 端分别输入高电平（即为“1”），C 端为低电平，结果逻辑电路 Y 端输出高电平（也即为“1”），高电平加到晶体管 VT 的基极，VT 导通，有电流流过灯，灯亮，表明判决成功。其他各种情况请读者自行分析。

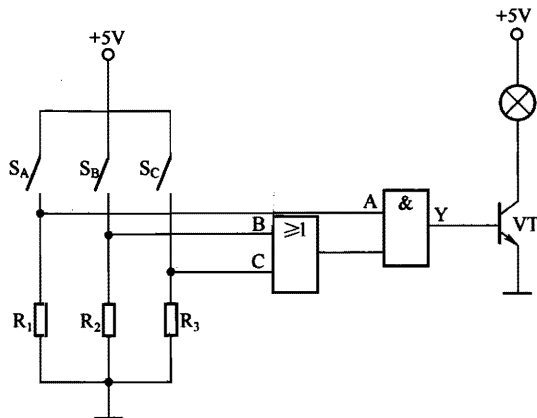


图 4-3 举重比赛裁判裁决实用电路

## 4.2 编码器

在数字电路中，将输入信号转换成一组二进制代码的过程称为编码。编码器是指能实现编码功能的电路。计算机键盘内部就用到编码器，当按下某个按键时，会给编码器输入一个信号，编码器会将该信号转换成一串由 1、0 组成的二进制代码送入计算机，按压不同的按键时，编码器转换成的二进制代码不同，计算机根据代码不同就能识别按下哪个按键。编码器的种类很多，主要分为两类：普通编码器和优先编码器。

### 4.2.1 普通编码器

普通编码器任何时刻只允许输入一个信号，若同时输入多个信号，编码输出就会产生混乱。

图 4-4 所示是一个典型普通编码器的电路结构。

工作原理说明如下：

图 4-4 中的  $S_0 \sim S_7$  8 个按键分别代表  $a \sim h$  8 个字母（各个按键上刻有相应的字母），当按下不同的按键时，编码器  $Y_0 \sim Y_2$  端会输出不同的二进制代码。

当按下代表字母“a”的按键  $S_0$  时，A 端为 1（高电平），但 A 端不与 3 个或门电路相连，又因为  $S_1 \sim S_7$  的按键都未按下，故 3 个或门输入都为 0，结果编码器输出  $Y_2 Y_1 Y_0 = 000$ 。即字母“a”经编码器编码后转换成二进制代码 000。

按下代表字母“f”的按键  $S_5$  时，F 端为 1，F=1 加到门  $G_1$  和门  $G_3$  的输入端，门  $G_1$  输出  $Y_0=1$ ，门  $G_3$  输出  $Y_2=1$ ，而门  $G_2$  输出  $Y_1=0$ ，结果编码器输出  $Y_2 Y_1 Y_0 = 101$ 。即字母“f”



经编码器编码后转换成二进制代码 101。

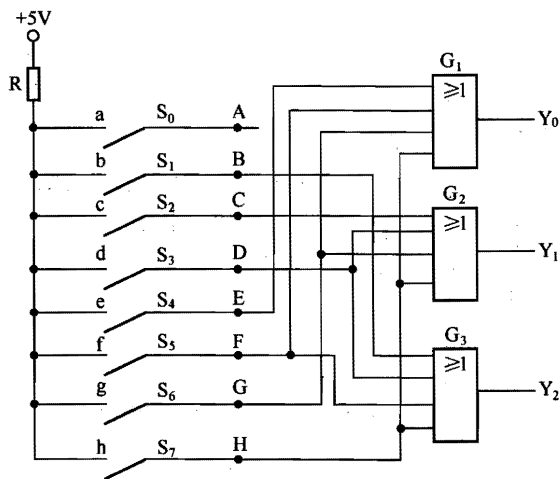


图 4-4 典型普通编码器的电路结构

当按下其他代表不同字母的按键时，编码器会输出相应的二进制代码，具体见表 4-3。

表 4-3 普通编码器的真值表

代表符号	输入变量	编码输出代码			代表符号	输入变量	编码输出代码		
		Y <sub>0</sub>	Y <sub>1</sub>	Y <sub>2</sub>			Y <sub>0</sub>	Y <sub>1</sub>	Y <sub>2</sub>
a	A=1	0	0	0	e	E=1	1	0	0
b	B=1	0	0	1	f	F=1	1	0	1
c	C=1	0	1	0	g	G=1	1	1	0
d	D=1	0	1	1	h	H=1	1	1	1

在图 4-4 所示的编码器中，如果同时按下多个按键，如同时按下“b”、“c”键，编码输出的代码为 Y<sub>2</sub>Y<sub>1</sub>Y<sub>0</sub>=110，它与按下“d”键时的编码输出相同。因此普通编码器在任意时刻只允许输入一个信号。

## 4.2.2 优先编码器

普通编码器在任意时刻只允许输入一个信号，而优先编码器同一时刻允许输入多个信号，但仅对输入信号中优先级别最高的一个信号进行编码输出。

### 1. 8 线-3 线优先编码器芯片

74LS148 是一种常用的 8 线-3 线优先编码器芯片，其各引脚功能如图 4-5 所示。



74LS148 有 8 个编码输入端 (0~7)、3 个编码输出端 ( $A_0 \sim A_2$ )、一个输入使能端 (EI)、一个输出使能端 (EO) 和一个片扩展输出端 (GS)。由于该编码器芯片有 8 个输入端和 3 个输出端, 故称为 8 线-3 线编码器。

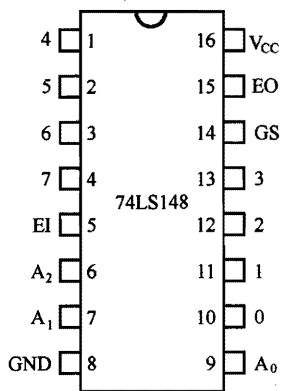


图 4-5 8 线-3 线优先编码器芯片

表 4-4 为 74LS148 的真值表, 表中的  $\times$  表示无论输入何值, 均不影响输出。

表 4-4 74LS148 的真值表

输 入									输 出				
EI	0	1	2	3	4	5	6	7	$A_2$	$A_1$	$A_0$	GS	EO
H	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	H	H	H	H	H
L	H	H	H	H	H	H	H	H	H	H	H	H	L
L	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	L	L	L	L	L	H
L	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	L	H	L	L	H	L	H
L	$\times$	$\times$	$\times$	$\times$	$\times$	L	H	H	L	H	L	L	H
L	$\times$	$\times$	$\times$	L	H	H	H	H	L	H	H	L	H
L	$\times$	$\times$	L	H	H	H	H	H	H	L	H	L	H
L	$\times$	L	H	H	H	H	H	H	H	H	L	L	H
L	L	H	H	H	H	H	H	H	H	H	H	L	H

从表 4-4 中不难看出:

① 当输入使能端  $EI=H$  时, 0~7 端无论输入何值, 输出端均为 H。即  $EI=H$  时, 编码器无法编码。

② 当  $EI=L$  时, 编码器可以对输入信号进行编码。在 8 个输入端中, 优先级别由高到低依次是 7、6……1、0, 当优先级别高的端子有信号输入时 (端子为低电平 L 时表示有信

号输入），编码器仅对该端信号进行编码，而不理睬优先级别低的端子。例如端子 7 输入信号时，编码器仅对该端输入进行编码，输出  $A_2A_1A_0=000$ ，若这时 0~6 端子有信号输入，编码器不予理睬。

另外，在编码器有编码输入时，会使  $GS=L$ 、 $EO=H$ ，无编码输入时， $GS=H$ 、 $EO=L$ 。

## 2. 8 线-3 线优先编码器

图 4-6 所示是一个由 74LS148 芯片组成的 8 线-3 线优先编码器，其输入使能端 EI 接地（ $EI=L$ ），让芯片能进行编码，GS、EO 端悬空未用。

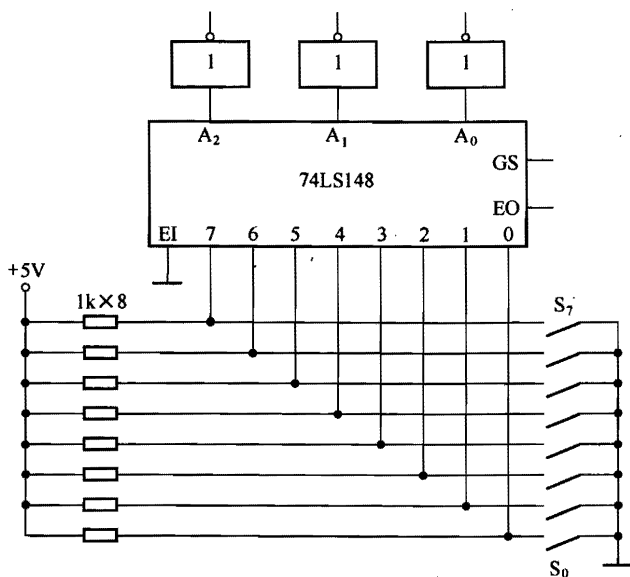


图 4-6 由 74LS148 芯片组成的 8 线-3 线优先编码器

当按键  $S_0 \sim S_7$  均未按下时，编码器 0~7 端子均为高电平，编码器无输入。

当  $S_6$  按下时，编码器 6 端变为低电平，表示 6 端有编码输入，编码器编码输出  $A_2A_1A_0=001$ ，经非门反相后变为 110。

当  $S_6$ 、 $S_5$  同时按下时，编码器 6、5 端均为低电平，但编码器仅对 6 端输入进行编码，编码输出  $A_2A_1A_0$  仍为 001。

## 3. 16 线-4 线优先编码器

图 4-7 所示是由两片 74LS148 芯片组成的 16 线-4 线优先编码器，它可以将  $D_{15} \sim D_0$  分别编码成 1111~0000 4 位代码输出。在两片 74LS148 中（2）为高位片，（1）为低位片，高位片的优先级别高，低位片的优先级别低。

该优先编码器的工作原理说明如下：





当高位片  $EI=1$  时，高位片禁止编码，高位片所有输出均为 1，高位片的  $EO$  也为 1，它使低位片的  $EI$  为 1，低位片也被禁止编码，低位片所有输出均为 1。

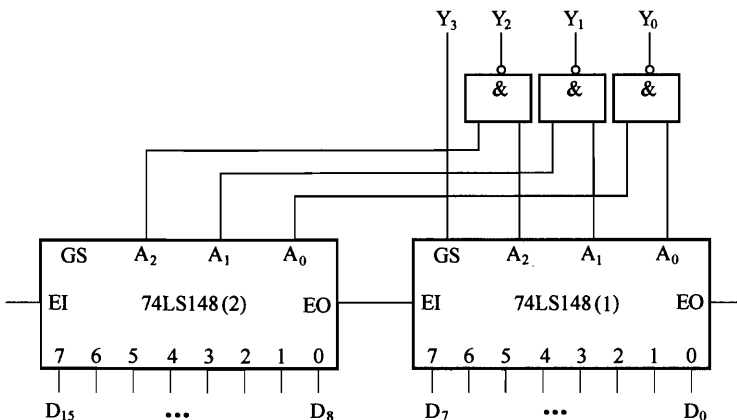


图 4-7 由两片 74LS148 芯片组成的 16 线-4 线优先编码器

当高位片  $EI=0$  时，高位片允许编码。若此时高位片有编码输入 ( $D_{15} \sim D_8$  端有低电平输入)，高位片的  $EO$  为 1，它使低位片的  $EI$  为 1，优先级别低的低位片被禁止编码。若高位片无编码输入，高位片的  $EO$  为 0，它使低位片  $EI$  为 0，低位片允许编码。

在高位片  $EI=0$  时，若  $D_{15}=0$ ，高位片的  $A_2A_1A_0=000$ ，高位片有编码输入，其  $EO=1$  使低位片禁止编码，低位片的  $A_2A_1A_0=111$ ，高、低位片输出经与非门后  $Y_2Y_1Y_0=111$ ，由于低位片  $GS=1$ ，故  $Y_3Y_2Y_1Y_0=1111$ 。

在高位片  $EI=0$  时，若  $D_6=0$ ，高位片无编码输入，其  $A_2A_1A_0=111$ ，高位片的  $EO=0$  使低位片允许编码，低位片的  $A_2A_1A_0=001$ ，高、低位片输出经与非门后  $Y_2Y_1Y_0=110$ ，由于低位片  $GS=0$ ，故  $Y_3Y_2Y_1Y_0=0110$ 。

## 4.3 译 码 器

“译码”是编码的逆过程，编码是将输入信号转换成二进制代码，而译码是将二进制代码翻译成特定输出信号的过程。能完成译码功能的电路称为译码器。常见的译码器有二进制译码器、二-十进制译码器和显示译码器等。

### 4.3.1 二进制译码器

#### 1. 二进制译码器工作原理

二进制译码器是一种能将不同组合的二进制代码译成相应输出信号的电路。下面以 2

位二进制译码器为例来说明二进制译码器的工作原理。

2 位二进制译码器框图如图 4-8 所示，其真值表见表 4-5。

当  $AB=00$  时，译码器  $Y_0$  端输出 “1”， $Y_1$ 、 $Y_2$ 、 $Y_3$  均为 “0”；

当  $AB=01$  时，译码器  $Y_1$  端输出 “1”， $Y_0$ 、 $Y_2$ 、 $Y_3$  均为 “0”；

当  $AB=10$  时，译码器  $Y_2$  端输出 “1”， $Y_0$ 、 $Y_1$ 、 $Y_3$  均为 “0”；

当  $AB=11$  时，译码器  $Y_3$  端输出 “1”， $Y_0$ 、 $Y_1$ 、 $Y_2$  均为 “0”。

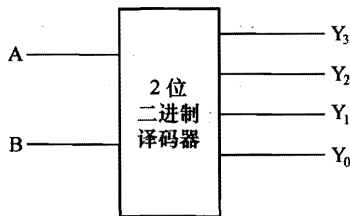


图 4-8 2 位二进制译码器框图

表 4-5 2 位二进制译码器真值表

输 入		输 出				输 入		输 出			
A	B	$Y_3$	$Y_2$	$Y_1$	$Y_0$	A	B	$Y_3$	$Y_2$	$Y_1$	$Y_0$
0	0	0	0	0	1	1	0	0	1	0	0
0	1	0	0	1	0	1	1	1	0	0	0

通过上面的过程了解二进制译码器后，下面再来分析 2 位二进制编码器的电路工作原理。2 位二进制译码器的电路结构如图 4-9 所示。

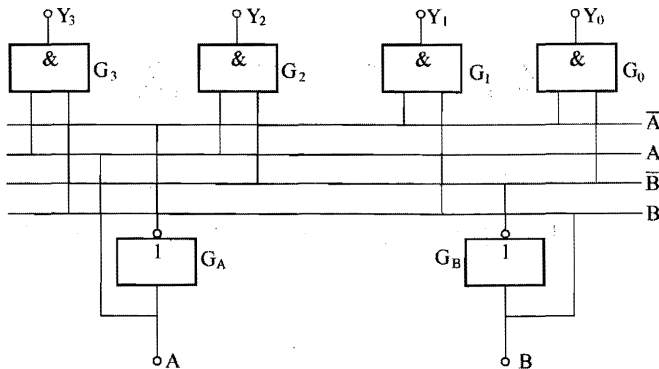


图 4-9 2 位二进制译码器电路结构

当  $A=0$ 、 $B=0$  时，非门  $G_A$  输出 “1”，非门  $G_B$  输出 “1”，与门  $G_3$  两个输入端同时输入 “0”，故输出端  $Y_3=0$ ；与门  $G_2$  两个输入端一个为 “0”，另一个为 “1”，输出端  $Y_2=0$ ；



与门  $G_1$  两个输入端一个为“0”，另一个为“1”，输出端  $Y_1=0$ ；与门  $G_0$  两个输入端同时输入“1”，故输出端  $Y_0=1$ 。也就是说，当  $AB=00$  时，只有  $Y_0$  输出为“1”。

当  $A=0$ 、 $B=1$  时，非门  $G_A$  输出“1”，非门  $G_B$  输出“0”，与门  $G_3$  两个输入端一个为“0”，另一个为“1”，输出端  $Y_3=0$ ；与门  $G_2$  两个输入端同时输入“0”，输出端  $Y_2=0$ ；与门  $G_1$  两个输入端同时输入“1”，输出端  $Y_1=1$ ；与门  $G_0$  两个输入端一个为“0”，另一个为“1”，输出端  $Y_0=0$ 。也就是说，当  $AB=01$  时，只有  $Y_1$  输出为“1”。

当  $A=1$ 、 $B=0$  时，只有  $Y_2=1$ ；当  $A=1$ 、 $B=1$  时，只有  $Y_3=1$ ；分析过程与上述过程相同，这里不再叙述。

2 位二进制译码器可以将 2 位代码译成 4 种输出状态，故又称 2 线-4 线译码器，而  $n$  位二进制译码器可以译成  $2^n$  种输出状态。

### 2. 3 线-8 线译码器芯片

74LS138 是一种常用的 3 线-8 线译码器芯片，其各引脚功能如图 4-10 所示。74LS138 有 3 个译码输入端 ( $A$ 、 $B$ 、 $C$ )、8 个译码输出端 ( $Y_0 \sim Y_7$ ) 和 3 个使能端 ( $G_{2A}$ 、 $G_{2B}$ 、 $G_1$ )。74LS138 的真值表见表 4-6。

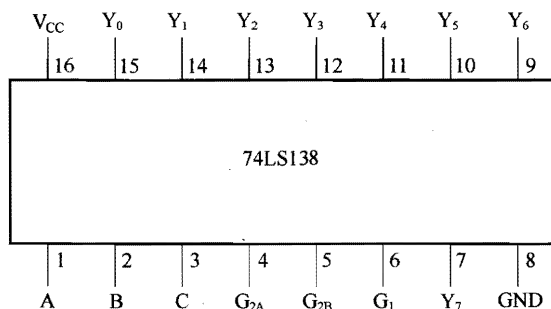


图 4-10 3 线-8 线译码器芯片 74LS138 的各引脚功能

表 4-6 74LS138 的真值表

输 入					输 出							
使 能		选 择										
$G_1$	$G_2$	$C$	$B$	$A$	$Y_0$	$Y_1$	$Y_2$	$Y_3$	$Y_4$	$Y_5$	$Y_6$	$Y_7$
×	H	×	×	×	H	H	H	H	H	H	H	H
L	×	×	×	×	H	H	H	H	H	H	H	H
H	L	L	L	L	L	H	H	H	H	H	H	H
H	L	L	L	H	H	L	H	H	H	H	H	H
H	L	L	H	L	H	H	L	H	H	H	H	H

续表

输 入					输 出							
使 能		选 择										
$G_1$	$G_2^*$	C	B	A	$Y_0$	$Y_1$	$Y_2$	$Y_3$	$Y_4$	$Y_5$	$Y_6$	$Y_7$
H	L	L	H	H	H	H	H	L	H	H	H	H
H	L	H	L	L	H	H	H	H	L	H	H	H
H	L	H	L	H	H	H	H	H	H	L	H	H
H	L	H	H	L	H	H	H	H	H	H	L	H
H	L	H	H	H	H	H	H	H	H	H	H	L

$$*G_2=G_{2A}+G_{2B}$$

从表不难看出：

① 当  $G_1=L$  或  $G_2=H$  ( $G_2=G_{2A}+G_{2B}$ ) 时，C、B、A 端无论输入何值，输出端均为 H。即  $G_1=L$  或  $G_2=H$  时，译码器无法译码。

② 当  $G_1=H$ 、 $G_2=L$  时，译码器允许译码，当 C、B、A 端输入不同的代码时，相应的输出端会输出低电平，如 CBA=001 时， $Y_1$  端会输出低电平（其他输出端均为高电平）。

### 3. 4 线-16 线译码器

图 4-11 所示是由两片 74LS138 芯片组成的 4 线-16 线译码器，当  $D_3 \sim D_0$  端输入不同的 4 位二进制代码时，经译码后，会从  $Z_{15} \sim Z_0$  相应端输出低电平。

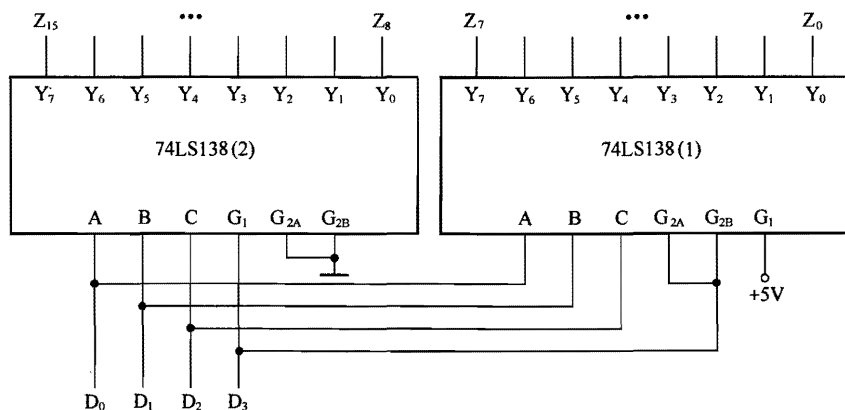


图 4-11 由两片 74LS138 芯片组成的 4 线-16 线译码器

该译码器的工作原理说明如下：

当  $D_3=0$  时，第 2 片 74LS138 的  $G_1=0$ ，该片禁止译码， $Z_{15} \sim Z_8$  端全为 1，第 1 片 74LS138 的  $G_2=0$  ( $G_2=G_{2A}+G_{2B}=0+0=0$ )、 $G_1=1$ ，该片允许译码。





例如在  $D_3D_2D_1D_0=0101$  时, 第2片 74LS148 禁止译码, 第1片 74LS148 的  $ABC=101$ ,  $Y_5$  端输出低电平, 即  $Z_5=0$ 。

当  $D_3=1$  时, 第2片 74LS138 的  $G_1=1$ 、 $G_2=0$ , 该片允许译码, 第1片 74LS138 的  $G_2=1$ , 该片禁止译码。

例如在  $D_3D_2D_1D_0=1101$  时, 第1片 74LS138 禁止译码, 第2片 74LS138 的  $ABC=101$ , 该片的  $Y_5$  端输出低电平, 即  $Z_{13}=0$ 。

### 4.3.2 二-十进制译码器

二-十进制译码器的功能是将 8421BCD 码中的 10 个代码译成 10 个相应的输出信号。

#### 1. 结构与原理

二-十进制译码器电路结构如图 4-12 所示, 其真值表见表 4-7。

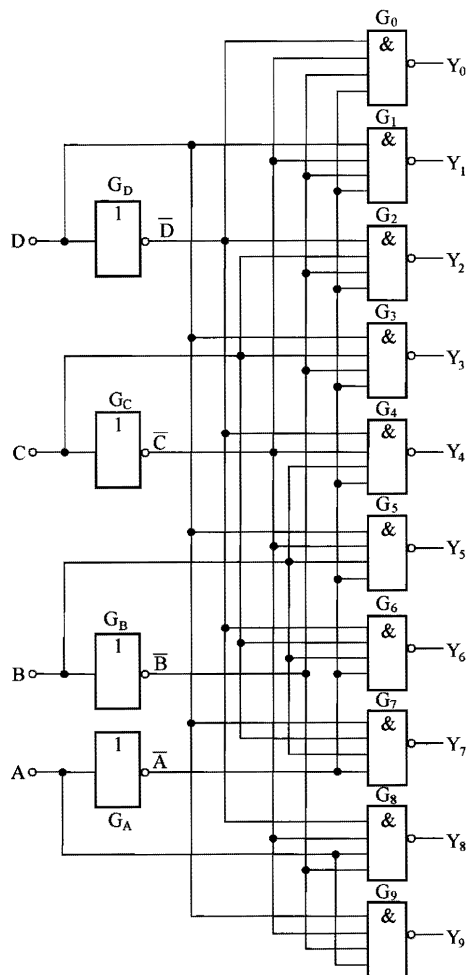


图 4-12 二-十进制译码器



表 4-7 二-十进制译码器的真值表

输 入				输 出										十进制数
A	B	C	D	Y <sub>0</sub>	Y <sub>1</sub>	Y <sub>2</sub>	Y <sub>3</sub>	Y <sub>4</sub>	Y <sub>5</sub>	Y <sub>6</sub>	Y <sub>7</sub>	Y <sub>8</sub>	Y <sub>9</sub>	
0	0	0	0	0	1	1	1	1	1	1	1	1	1	0
0	0	0	1	1	0	1	1	1	1	1	1	1	1	1
0	0	1	0	1	1	0	1	1	1	1	1	1	1	2
0	0	1	1	1	1	1	0	1	1	1	1	1	1	3
0	1	0	0	1	1	1	1	0	1	1	1	1	1	4
0	1	0	1	1	1	1	1	1	0	1	1	1	1	5
0	1	1	0	1	1	1	1	1	1	0	1	1	1	6
0	1	1	1	1	1	1	1	1	1	1	0	1	1	7
1	0	0	0	1	1	1	1	1	1	1	1	0	1	8
1	0	0	1	1	1	1	1	1	1	1	1	1	0	9
1	0	1	0	1	1	1	1	1	1	1	1	1	1	伪码
1	0	1	1	1	1	1	1	1	1	1	1	1	1	伪码
1	1	0	0	1	1	1	1	1	1	1	1	1	1	伪码
1	1	0	1	1	1	1	1	1	1	1	1	1	1	伪码
1	1	1	0	1	1	1	1	1	1	1	1	1	1	伪码
1	1	1	1	1	1	1	1	1	1	1	1	1	1	伪码

工作原理说明如下：

当输入二进制代码  $ABCD=0000$  时，非门  $G_A$ 、 $G_B$ 、 $G_C$ 、 $G_D$  输出都为“1”，与非门  $G_0$  4 个输入端都为“1”，故  $G_0$  输出端  $Y_0=0$ ，该端代表十进制数“0”，其他的与非门  $G_1\sim G_9$  都至少有一个输入为“0”，所以  $G_1\sim G_9$  都输出“1”。注：该译码器输出端为“1”表示无输出，而输出端为“0”表示有输出。

当输入二进制代码  $ABCD=0011$  时，非门  $G_A$ 、 $G_B$  输出都为“1”，非门  $G_C$ 、 $G_D$  输出都为“0”，与非门  $G_3$  4 个输入端都为“1”，故  $G_3$  输出端  $Y_3=0$ ，该端代表十进制数 3，其他的与非门  $G_0$ 、 $G_1$ 、 $G_2$ 、 $G_4\sim G_9$  都至少有 1 个输入为“0”，所以  $G_0$ 、 $G_1$ 、 $G_2$ 、 $G_4\sim G_9$  都输出“1”。

当输入二进制代码  $ABCD=1010$  时，非门  $G_A$ 、 $G_C$  输出都为“0”，非门  $G_B$ 、 $G_D$  输出都为“1”，与非门  $G_0\sim G_9$  都至少有 1 个输入为“0”， $G_0\sim G_9$  都输出“1”。也就是说，当二-十进制译码器输入 1010 时，译码器无输出。实际上，当  $ABCD$  为 1010、1011、1100、1101、



1110、1111 时，译码器都无输出，这些代码称之为伪码。

2. 常用的二-十进制译码器芯片

74LS42 是一种常用的二-十进制译码器芯片，其各引脚功能如图 4-13 所示，其真值表见表 4-8。

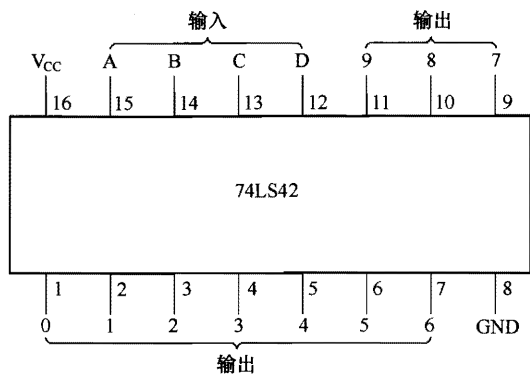


图 4-13 二-十进制译码器芯片 74LS42 的各引脚功能

表 4-8 74LS42 的真值表

BCD 码输入				译码输出										对应十进制数
D	C	B	A	0	1	2	3	4	5	6	7	8	9	
L	L	L	L	L	H	H	H	H	H	H	H	H	H	0
L	L	L	H	H	L	H	H	H	H	H	H	H	H	1
L	L	H	L	H	H	L	H	H	H	H	H	H	H	2
L	L	H	H	H	H	H	L	H	H	H	H	H	H	3
L	H	L	L	H	H	H	H	L	H	H	H	H	H	4
L	H	L	H	H	H	H	H	H	L	H	H	H	H	5
L	H	H	L	H	H	H	H	H	H	L	H	H	H	6
L	H	H	H	H	H	H	H	H	H	H	L	H	H	7
H	L	L	L	H	H	H	H	H	H	H	H	L	H	8
H	L	L	H	H	H	H	H	H	H	H	H	H	L	9
H	L	H	L	H	H	H	H	H	H	H	H	H	H	伪码
H	L	H	H	H	H	H	H	H	H	H	H	H	H	
H	H	L	L	H	H	H	H	H	H	H	H	H	H	
H	H	L	H	H	H	H	H	H	H	H	H	H	H	
H	H	H	L	H	H	H	H	H	H	H	H	H	H	
H	H	H	H	H	H	H	H	H	H	H	H	H	H	



### 4.3.3 数码显示器与显示译码器

数码显示器的功能是在显示译码器送来的信号驱动下直观显示十进制数码。显示译码器的功能是将输入二进制代码译成一定的输出信号，让该信号驱动显示器显示与输入代码相对应的字符。

#### 1. 数码显示器

数码显示器用来显示十进制数码。七段数码显示器是一种最常见的数码显示器，它可分为半导体数码显示器、荧光数码显示器和液晶数码显示器等。

##### (1) 七段半导体数码显示器

① 结构与原理。七段半导体数码显示器又称七段数码管，它采用 7 个半导体发光二极管（LED），它将 a、b、c、d、e、f、g 共 7 个发光二极管排成图 4-14 所示的“E”字形，这种显示器采用七段组合来显示 0~9 数字。七段半导体数码显示器外形如图 4-15 所示。

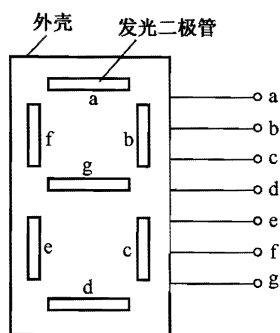


图 4-14 七段数码显示器七段排列图

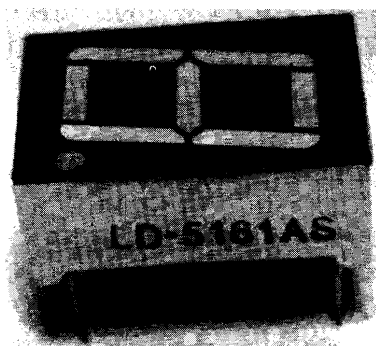


图 4-15 七段半导体数码显示器外形

由于 7 个发光二极管共有 14 个引脚，为了减少显示器的引脚数，在显示器内部将



7个发光二极管正极或负极引脚连接起来, 接成一个公共端, 根据公共端是发光二极管正极还是负极, 可分为共阳极接法(正极相连)和共阴极接法(负极相连), 如图4-16所示。

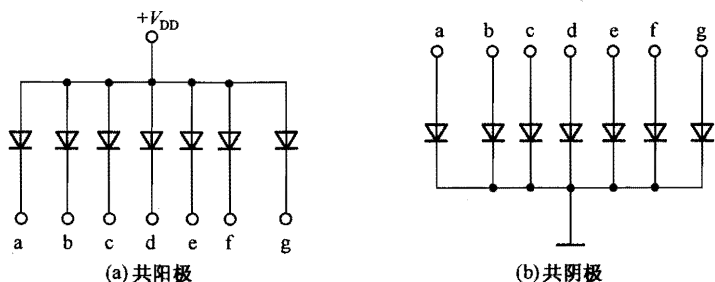


图 4-16 半导体数码显示器内部发光二极管的连接方式

对于共阳极接法的显示器, 需要给发光二极管加低电平才能发光; 而对于共阴极接法的显示器, 需要给发光二极管加高电平才能发光。假设图 4-14 是一个共阴极接法的显示器, 如果让它显示一个“5”字, 那么需要给 a、c、d、f、g 引脚加高电平(即这些引脚为 1), b、e 引脚加低电平(即这些引脚为 0), 这样 a、c、d、f、g 段的发光二极管有电流通过而发光, b、e 段的发光二极管不发光, 显示器就会显示出数字“5”。

② 检测。实际的七段数码管有 10 个引脚, 分作两排, 每排中间的一个引脚为公共引脚 com, 其他 8 个引脚分别为 a、b、c、d、e、f、g 和小数点。在安装数码管前, 先要检测该数码管极性(共阳极或共阴极), 再检测各引脚对应的段位。

在检测七段数码管极性时, 万用表选择  $\times 10k\Omega$  挡, 黑表笔接 com 引脚(公共引脚), 红表笔接 com 引脚外的任意一脚, 如图 4-17 所示, 若测得阻值小, 则该数码管为共阳极, 若测得阻值接近无穷大则为共阴极。

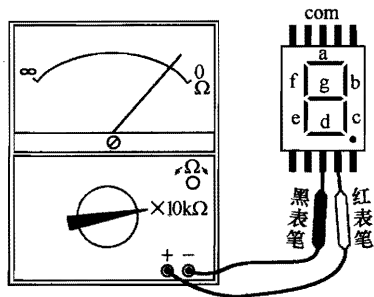


图 4-17 七段数码管的检测

七段数码管引脚与内部段位对应关系检测与极性检测基本相同, 对于共阳极数码管, 万用表选择  $\times 10k\Omega$  挡, 黑表笔接 com 引脚, 红表笔接其他某个引脚, 这时会发现数码管

某段会有微弱的亮光，如 a 段有亮光，表明红表笔接的引脚与 a 段负极连接；对于共阴极数码管，万用表仍选择  $\times 10\text{k}\Omega$  挡，红表笔接 com 引脚，黑表笔接其他某个引脚，会发现数码管某段会有微弱的亮光，则黑表笔接的引脚与该段正极连接。

## （2）荧光数码显示器

荧光数码显示器常用在一些家用电器中（如影碟机、录像机和音响设备），用来显示机器的状态和时间等。荧光数码显示器有 1 位荧光数码显示器和多位荧光数码显示器。

① 1 位荧光数码显示器。荧光数码显示器是一种真空器件，1 位荧光数码显示器的结构示意图如图 4-18 所示。它内部有灯丝、栅极（控制极）和 a、b、c、d、e、f、g 7 个阳极，这 7 个阳极上都涂有荧光粉并排列成“8”字样，灯丝的作用是发射电子，栅极处于灯丝和阳极之间，灯丝发射出来的电子能否到达阳极受栅极的控制，阳极上涂有荧光粉，当电子轰击荧光粉时，阳极上的荧光粉发光。

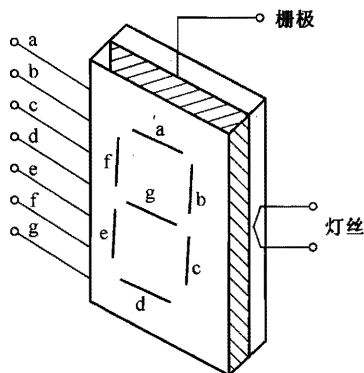


图 4-18 1 位荧光数码显示器结构示意图

在荧光数码显示器工作时，要给灯丝提供约 6.3V 的交流电压，灯丝发热后才能发射电子，栅极要加上较高的电压才能吸引电子，让它穿过栅极并往阳极方向运动。电子要轰击某个阳极，该阳极必须有高电压。

当要显示“3”字样时，译码器给荧光数码显示器的 a、b、c、d、e、f、g 7 个阳极分别送 1、1、1、1、0、0、1，即给 a、b、c、d、g 5 个阳极送高电压，另外给栅极也加上高电压，于是灯丝发射的电子穿过栅极后轰击加有高电平的 a、b、c、d、g 阳极，由于这些阳极上涂有荧光粉，在电子的轰击下，这些阳极发光，显示器显示“3”的字样。

② 多位荧光数码显示器。1 位荧光数码显示器能显示 1 位数字，当需要同时显示多位数字时就要用到多位荧光数码显示器。下面以 4 位荧光数码显示器为例来说明其工作原理。4 位荧光数码显示器的结构示意图如图 4-19 所示。

4 位荧光数码显示器有 A、B、C、D 4 个位区，每个位区可以看成是 1 位荧光数码显



示器，每个位区都有单独的栅极、灯丝和 a、b、c、d、e、f、g 7 个阳极。4 个位区的栅极引出脚分别为  $G_1$ 、 $G_2$ 、 $G_3$ 、 $G_4$ ；每个位区的灯丝在内部以并联的形式连接起来，对外只引出两个引脚；每个位区相应各段的阳极都连接在一起，再与外面的引脚相连，例如 D 位区的阳极 a 与 C、B、A 位内的阳极 a 都连接起来，再与显示器外引脚 a 连接。

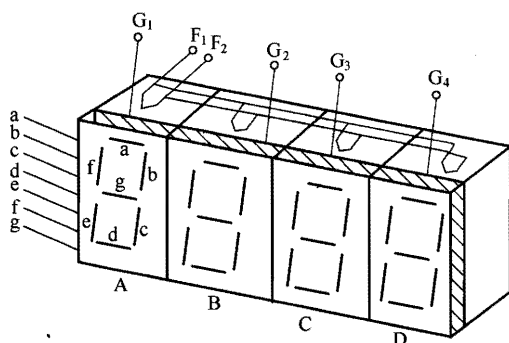


图 4-19 4 位荧光数码显示器结构示意图

**多位荧光数码显示器采用了扫描显示原理。**为了让大家理解这种显示原理，这里以在图 4-19 所示显示器上显示“1278”这 4 位数为例来说明。

首先给灯丝引脚  $F_1$ 、 $F_2$  通电，再给  $G_1$  引脚加一个高电平，此时  $G_2$ 、 $G_3$ 、 $G_4$  均为低电平，然后分别给 b、c 引脚加高电平。灯丝通电发热后发射电子，电子穿过  $G_1$  栅极轰击 A 位阳极 b、c，这两个电极的荧光粉发光，在 A 位显示“1”字样，这时虽然 b、c 引脚的电压也会加到 B、C、D 位的阳极 b、c 上，但因为 B、C、D 位的栅极为低电平，这些位的灯丝发射的电子无法穿过栅极轰击阳极，故 B、C、D 位无显示；接着给  $G_2$  脚加高电平，此时  $G_1$ 、 $G_3$ 、 $G_4$  引脚均为低电平，再给阳极 a、b、d、e、g 加高电平，灯丝发射的电子轰击 B 位阳极 a、b、d、e、g，这些阳极发光，在 B 位显示“2”字样。同样原理，在 C 位和 D 位分别显示“7”、“8”字样。

显示器的数字虽然是一位一位地显示出来的，但由于荧光粉的余辉效应（所谓余辉效应是指荧光粉发光后，即使无电子轰击光亮还保持一定时间）和人眼视觉暂留特性（所谓视觉暂留特性是指当人眼看见一个物体后，如果物体消失，人眼还会觉得物体仍在原位置，这种感觉约保留 0.04s 的时间），当显示器显示最后 1 位数字“8”时，人眼会感觉前面 3 位数字还在显示，故看起来好像是一下子显示“1278”4 位数。

### （3）液晶数码显示器

液晶数码显示器的主要材料是液态晶体，简称液晶，它是一种有机材料，这种材料在一个特定的温度范围内既有液体的流动性，又有晶体的某些光学特性，其透明度和颜色随电场、磁场、光和温度等外界条件变化而变化。**液晶数码显示器是利用液晶在电场作用下光学性能变化的特性制成的。**

液晶数码显示器的结构如图 4-20 所示，它是将液晶材料封装在两块玻璃之间，在上玻璃内表面涂上“日”字形的 7 段透明导电层，在下玻璃内表面整个涂上导电层（反射层）。

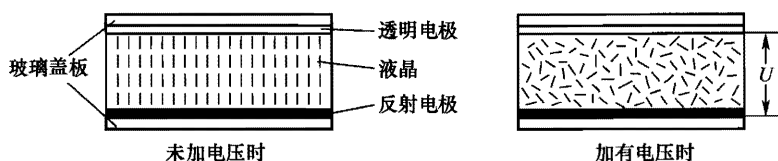


图 4-20 液晶显示器结构

当给液晶显示器正面（也即上面）玻璃板上的某段电极与下面玻璃的导电层之间加上适当大小的电压时，该段电极与下玻璃导电层所夹持的液晶会产生“散射效应”，夹持的液晶不透明，就会显示出该段形状。例如给下玻璃层的导电层加一个低电压，而给上玻璃层的 a、b 段透明导电电极加高电压，这两段电极与下玻璃上的导电层存在电压差，它们中间夹持的液晶特性改变，a、b 段下面液晶变为不透明，显示“1”字样。

液晶显示器工作时不需要电流，耗电很少，但由于本身不发光，所以需借助外界光源照射显示数码。

半导体数码显示器工作电压低、字形清晰、体积小、寿命长；荧光数码显示器字形清晰，工作电压较低且驱动电流不大，但工作时由于需要灯丝发热，故功耗很大；液晶数码显示器工作电压和电流都很小，制作工艺简单、体积小，但清晰度较低。

## 2. 显示译码器

显示译码器的功能是将输入的二进制代码译成一定的输出信号，让输出信号驱动显示器来显示与输入代码相对应的字符。显示译码器种类很多，这里介绍 BCD-七段显示译码器，它可以将 BCD 码译成一定的输出信号，该信号能驱动七段数码显示器显示与 BCD 码对应的十进制数。

### （1）常用的 BCD-七段显示译码器芯片

74LS48 是一种常用的 BCD-七段显示译码器芯片，其各引脚功能如图 4-21 所示，其真值表见表 4-9。

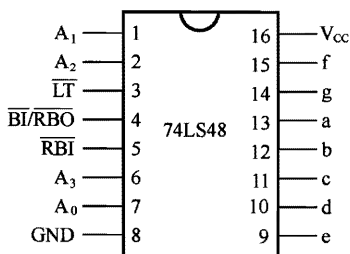


图 4-21 74LS48 芯片的各引脚功能





表 4-9 74LS48 的真值表

十进制数	控 制			输 入				输 出							字形
	$\overline{\text{LT}}$	$\overline{\text{RBI}}$	$\overline{\text{BI/RBO}}$	$A_3$	$A_2$	$A_1$	$A_0$	a	b	c	d	e	f	g	
0	H	H	H	L	L	L	L	H	H	H	H	H	H	L	0
1	H	x	H	L	L	L	H	L	H	H	L	L	L	L	1
2	H	x	H	L	L	H	L	H	H	L	H	H	L	H	2
3	H	x	H	L	L	H	H	H	H	H	H	L	L	H	3
4	H	x	H	L	H	L	L	L	H	H	L	L	H	H	4
5	H	x	H	L	H	L	H	H	L	H	H	L	H	H	5
6	H	x	H	L	H	H	L	L	L	H	H	H	H	H	6
7	H	x	H	L	H	H	H	H	H	H	L	L	L	L	7
8	H	x	H	H	L	L	L	H	H	H	H	H	H	H	8
9	H	x	H	H	L	L	H	H	H	H	L	L	H	H	9
10	H	x	H	H	L	H	L	L	L	L	H	H	L	H	c
11	H	x	H	H	L	H	H	L	L	H	H	L	L	H	>
12	H	x	H	H	H	L	L	L	H	L	L	L	H	H	u
13	H	x	H	H	H	L	H	H	L	L	H	L	H	H	E
14	H	x	H	H	H	H	L	L	L	L	H	H	H	H	z
15	H	x	H	H	H	H	H	L	L	L	L	L	L	L	全暗
	x	x	L	x	x	x	x	L	L	L	L	L	L	L	全暗
	H	L	L	L	L	L	L	L	L	L	L	L	L	L	全暗
	L	x	H	x	x	x	x	H	H	H	H	H	H	H	8

74LS48 有 3 类端子：输入端、输出端和控制端。 $A_3 \sim A_0$  为输入端，用来输入 8421BCD 码； $a \sim g$  为输出端，芯片对输入的 BCD 码译码后，会从  $a \sim g$  端输出相应的信号，来驱动七段显示器显示与 BCD 码对应的十进制数。 $\overline{\text{LT}}$ 、 $\overline{\text{RBI}}$  和  $\overline{\text{BI/RBO}}$  为控制端。

$\overline{\text{LT}}$  端为灯测试输入端。只要  $\overline{\text{LT}}=0$ ，就可以使  $a \sim g$  端输出全为高电平，将七段显示器所有段全部点亮，以检查显示器各段显示是否正常。

$\overline{\text{RBI}}$  端为灭零输入端。当多位七段显示器显示多位数字时，利用该端  $\overline{\text{RBI}}=0$  可以将不希望显示的“0”熄灭，如 8 位七段显示器显示数字“12.3”，如果不灭零，会显示“0012.3000”，



灭零后则显示“12.3”，使显示更醒目。

$\overline{\text{BI}}/\overline{\text{RBO}}$  端为灭灯输入/灭零输出端，它是一个双功能端子。当  $\overline{\text{BI}}/\overline{\text{RBO}}$  端用作输入端使用时，称灭灯输入控制端，只要  $\overline{\text{BI}}/\overline{\text{RBO}}=0$ ，无论  $A_3$ 、 $A_2$ 、 $A_1$ 、 $A_0$  输入什么， $a\sim g$  端输出全为低电平，使七段显示器的各段同时熄灭。当  $\overline{\text{BI}}/\overline{\text{RBO}}$  作为输出端使用时，称灭零输出端。当  $A_3A_2A_1A_0=0000$  且有灭零信号输入（ $\overline{\text{RBI}}=0$ ）时，该端会输出低电平，表示译码器已进行了灭零操作。

### （2）1 位译码显示电路

图 4-22 所示是一个由 74LS48 芯片和 BS202 型共阴极半导体数码管组成的 1 位译码显示电路。

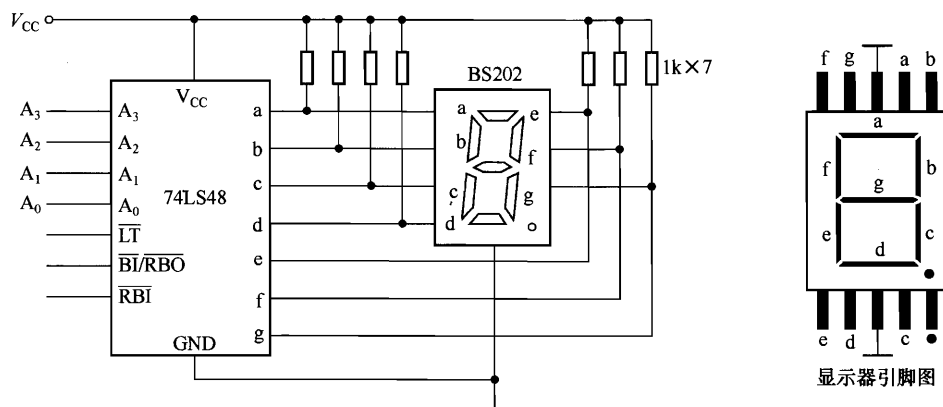


图 4-22 由 74LS48 芯片和共阴极数码管组成的 1 位译码显示电路

如果要检测数码管各段是否显示正常，可让  $\overline{\text{LT}}=0$ ，74LS48 芯片的  $a\sim g$  端输出全为高电平，数码管各段同时点亮。若某段不显示，而芯片相应输出端又为高电平，则为数码管该段有故障。

当  $A_3A_2A_1A_0=0000$  时，74LS48 芯片的  $a\sim f$  端输出为高电平， $g$  端为低电平，数码管显示“0”，如果要将该“0”熄灭，可让  $\overline{\text{RBI}}=0$ ，芯片  $a\sim g$  端输出全为低电平。

在数码管显示任何数字时，若让  $\overline{\text{BI}}/\overline{\text{RBO}}=0$ ，74LS48 芯片  $a\sim g$  端输出全变为低电平，数码管原先显示的数字将消失。

当  $A_3A_2A_1A_0=0000$  且  $\overline{\text{RBI}}=0$ （有灭零信号输入）时，74LS48 芯片  $a\sim g$  端输出全为低电平，同时  $\overline{\text{BI}}/\overline{\text{RBO}}$  会输出低电平，表示译码器已进行了灭零操作。

在正常工作时，可将  $\overline{\text{LT}}$ 、 $\overline{\text{RBI}}$  和  $\overline{\text{BI}}/\overline{\text{RBO}}$  三端连接在一起，并接高电平，数码管的显示会随  $A_3A_2A_1A_0$  的变化而变化。

### （3）多位译码显示电路

图 4-23 所示是一个由 74LS48 芯片和半导体数码管组成的 8 位译码显示电路。该电路



将 74LS48 的灭零输入端与灭零输出端配合使用, 来实现多位数码显示控制。

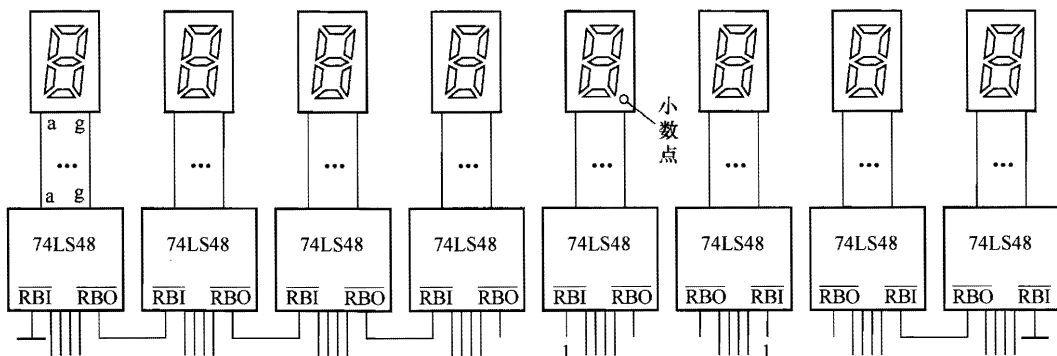


图 4-23 由 74LS48 芯片和半导体数码管组成的 8 位译码显示电路

在使用时, 只需在整数部分将高位的  $\overline{\text{RBO}}$  与低位的  $\overline{\text{RBI}}$  相连, 而在小数部分将低位的  $\overline{\text{RBO}}$  与高位的  $\overline{\text{RBI}}$  相连, 就可以把前后多余的零熄灭。下面以显示“00381.560”为例进行说明。

在整数部分, 最高位 74LS48 输入为 0000 且灭零端  $\overline{\text{RBI}}=0$  ( $\overline{\text{RBI}}$  接地), 最高位数码管灭零, 同时最高位 74LS48 的灭零输出端  $\overline{\text{RBO}}=0$ , 它使次高位 74LS48 的  $\overline{\text{RBI}}=0$ , 因为次高位 74LS48 的输入也为 0000, 故次高位数码管也被灭零, 次高位 74LS48 的灭零输出端  $\overline{\text{RBO}}=0$ , 它使第 3 高位 74LS48 的  $\overline{\text{RBI}}=0$ , 但因第 3 高位 74LS48 输入不为 0000 (为 0011), 故第 3 高位数码管正常显示“3”。

在小数部分, 最低位 74LS48 输入为 0000 且灭零端  $\overline{\text{RBI}}=0$  ( $\overline{\text{RBI}}$  接地), 最低位数码管灭零, 同时最低位 74LS48 的灭零输出端  $\overline{\text{RBO}}=0$ , 它使次低位 74LS48 的  $\overline{\text{RBI}}=0$ , 但因次低位 74LS48 输入不为 0000 (为 0110), 故次低位数码管正常显示“6”。

## 4.4 数码管译码控制器的电路原理与实验

数码管译码控制器是一种将 8421BCD 码进行译码并驱动七段数码管显示数字 0~9 的电路, 该控制器还能对数码管进行试灯、灭灯和灭零控制。

### 4.4.1 电路原理

图 4-24 所示为数码管译码控制器的电路原理图。在电路中, 5161BS 为共阳极七段数码管, 74LS47 为 BCD-七段显示译码器芯片, 表 4-10 为 74LS47 的真值表。S<sub>RBI</sub> 为灭零按钮, S<sub>LT</sub> 为试灯按钮, S<sub>BI/RBO</sub> 为灭灯输入/灭零输出按钮, 这 3 个按钮在未按下时,

74LS47 的  $\overline{\text{LT}}$ 、 $\overline{\text{RBI}}$  和  $\overline{\text{BI/RBO}}$  引脚均为高电平； $S_0 \sim S_3$  按钮分别为 74LS47 的  $A_0 \sim A_3$  引脚提供输入信号，按钮未按下时，输入为低电平，按下时输入为高电平。

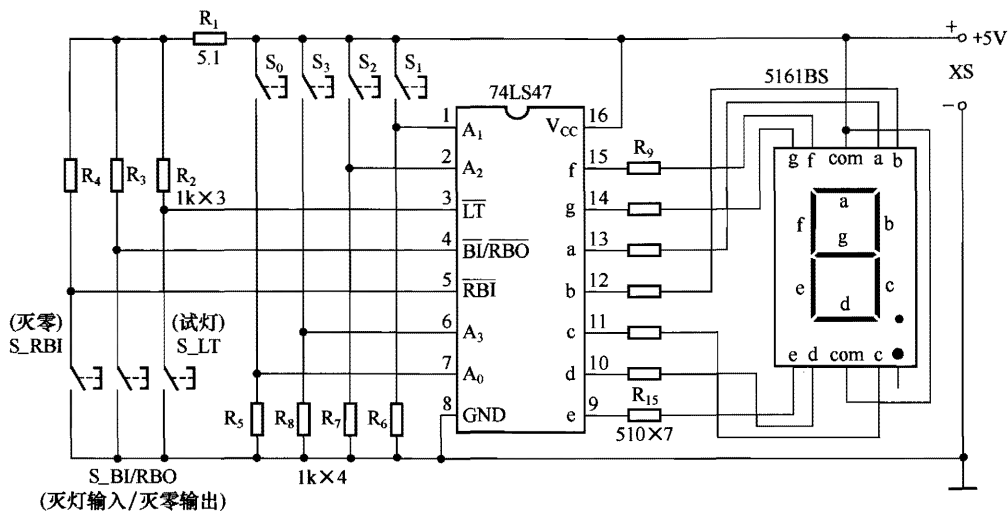


图 4-24 数码管译码控制器的电路原理图

表 4-10 74LS47 真值表

十进制数	输入及控制							输出						
	$\overline{\text{LT}}$	$\overline{\text{RBI}}$	$A_3$	$A_2$	$A_1$	$A_0$	$\overline{\text{BI/RBO}}$	a	b	c	d	e	f	g
0	H	H	L	L	L	L	H	L	L	L	L	L	L	H
1	H	×	L	L	L	H	H	H	L	L	H	H	H	H
2	H	×	L	L	H	L	H	L	L	H	L	L	H	L
3	H	×	L	L	H	H	H	L	L	L	L	H	H	L
4	H	×	L	H	L	L	H	H	L	L	H	H	L	L
5	H	×	L	H	L	H	H	L	H	L	L	H	L	L
6	H	×	L	H	H	L	H	H	H	L	L	L	L	L
7	H	×	L	H	H	H	H	L	L	L	H	H	H	H
8	H	×	H	L	L	L	H	L	L	L	L	L	L	L
9	H	×	H	L	L	H	H	L	L	L	H	H	L	L
10	H	×	H	L	H	L	H	H	H	H	L	L	H	L
11	H	×	H	L	H	H	H	H	H	L	L	H	H	L
12	H	×	H	H	L	L	H	H	L	H	H	H	L	L
13	H	×	H	H	L	H	H	L	H	H	L	H	L	L





续表

十进制数	输入及控制							输出						
	$\overline{\text{LT}}$	$\overline{\text{RBI}}$	$A_3$	$A_2$	$A_1$	$A_0$	$\overline{\text{BI}}/\overline{\text{RBO}}$	a	b	c	d	e	f	g
14	H	×	H	H	H	L	H	H	H	H	L	L	L	L
15	H	×	H	H	H	H	H	H	H	H	H	H	H	H
$\overline{\text{BI}}$	×	×	×	×	×	×	L	H	H	H	H	H	H	H
$\overline{\text{RBI}}$	H	L	L	L	L	L	L	H	H	H	H	H	H	H
$\overline{\text{LT}}$	L	×	×	×	×	×	H	L	L	L	L	L	L	L

4.4.2 实验操作

- 根据数码管译码控制器电路原理图和 74LS47 真值表分析下面的实验操作结果。
- 第一步：将数码管译码控制器与 5V 电源连接好，数码管显示的字形为\_\_\_\_\_。
- 第二步：按下按键  $S_0$ ，数码管显示字形为\_\_\_\_\_。
- 第三步：按下按键  $S_1$ 、 $S_0$ ，数码管显示字形为\_\_\_\_\_。
- 第四步：按下按键  $S_2$ ，数码管显示字形为\_\_\_\_\_。
- 第五步：按下按键  $S_2$ 、 $S_0$ ，数码管显示字形为\_\_\_\_\_。
- 第六步：按下按键  $S_3$ 、 $S_1$ ，数码管显示字形为\_\_\_\_\_。
- 第七步：按下按键  $S_3$ 、 $S_2$ 、 $S_1$ 、 $S_0$ ，数码管显示字形为\_\_\_\_\_。
- 第八步：按下按键  $S_{\overline{\text{LT}}}$ ，数码管显示字形为\_\_\_\_\_。
- 第九步：按下按键  $S_{\overline{\text{RBI}}}$ ，数码管显示字形为\_\_\_\_\_。
- 第十步：按下按键  $S_{\overline{\text{BI}}/\overline{\text{RBO}}}$ ，数码管显示字形为\_\_\_\_\_。

4.5 加 法 器

计算机等数字电子设备最基本的任务是进行算术运算，数字电子设备中的加、减、乘、除四则运算都是分解成加法运算进行的，所以加法器是数字电子设备中最基本的运算单元。加法器又分半加器和全加器。

4.5.1 半加器

两个 1 位二进制数相加运算，称为半加，实现半加运算功能的电路称为半加器。半加



器可以由一个异或门和一个与门组成，如图 4-25（a）所示；也可以由一个异或门、一个与非门及一个非门组成，如图 4-25（b）所示。半加器的图形符号如图 4-25（c）所示，其中 A、B 表示加数，S 表示半加和，C 表示进位数。

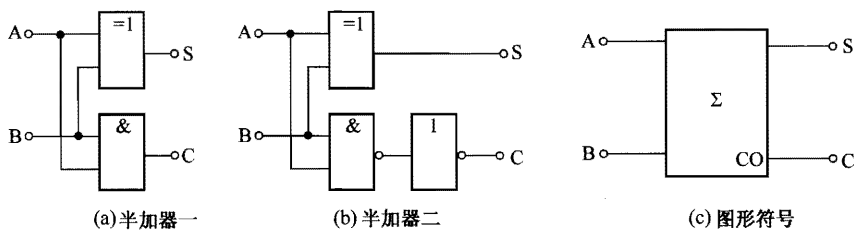


图 4-25 半加器

下面以图 4-25（a）所示的半加器为例来说明其工作原理。

当 A 端输入“0”，B 端输入“1”时，异或门的 S 端输出“1”（异或门的功能是输入相同时输出为“0”，输入相异时输出为“1”），而与门的 C 端输出“0”，即“0+1=1”。

当 A、B 端都输入“1”时，异或门的 S 端输出“0”，与门的 C 端输出“1”，即“1+1=10”。A、B 端其他的输入情况不再叙述，请读者自己分析。半加器的真值表见表 4-11。

表 4-11 半加器的真值表

输 入		输 出		输 入		输 出	
A	B	S	C	A	B	S	C
0	0	0	0	1	0	1	0
0	1	1	0	1	1	0	1

## 4.5.2 全加器

在实际的二进制加法运算中，经常会遇到多位数相加的情况，例如两位数 11+01 的运算，两个数的低位 1 和 1 相加时会产生进位 1，而两个数的高位除了要进行 1+0 外，还要加上低位的进位数 1，这是半加器无法完成的，需要由全加器来完成。

全加是带进位的加法运算，它除了要将两个同位数相加外，还要加上低位送来的进位数。全加器是用来实现全加运算的电路。全加器具有 3 个输入端：加数 A、B 和低位来的进位数  $C_{n-1}$ ；两个输出端：和数  $S_n$  和向高位进位数  $C_n$ 。全加器由两个半加器和一个或门组成，如图 4-26（a）所示，全加器的图形符号如图 4-26（b）所示。

下面来分析图 4-26（a）所示全加器的工作原理。

A、B 为两个加数， $C_{n-1}$  为低位来的进位数， $S_n$  为和数， $C_n$  为高位进位数， $\Sigma_1$  和  $\Sigma_2$



均为半加器。

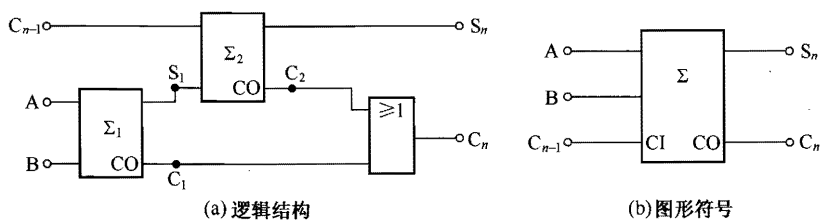


图 4-26 全加器

当 A 端输入“1”、B 端输入“0”、 $C_{n-1}$  端输入“0”（即低位无进位）时，半加器  $\Sigma_1$  的进位  $C_1$  端输出“0”去或门，和数  $S_1$  端输出“1”去半加器  $\Sigma_2$  的一个输入端，同时低位进位数  $C_{n-1}$  的“0”送到半加器  $\Sigma_2$  的另一个输入端，结果半加器  $\Sigma_2$  的和数  $S_n$  端输出“1”，进位  $C_2$  端输出“0”。 $C_1=0$  和  $C_2=0$  送到或门的输入端，或门  $C_n$  端输出“0”。即当 A 端输入“1”、B 端输入“0”、 $C_{n-1}$  端输入“0”时，全加器的  $S_n=1$ ，高位进位数  $C_n=0$ 。

当  $A=1$ 、 $B=1$ 、低位进位数  $C_{n-1}=1$ （即低位有进位数）时，半加器  $\Sigma_1$  的和数端  $S_1=0$ ，进位输出端  $C_1=1$ 。 $S_1=0$  和  $C_{n-1}=1$  送到半加器  $\Sigma_2$  输入端，半加器  $\Sigma_2$  的和数端  $S_n=1$ ，进位数端  $C_2=0$ 。 $C_2=0$  和  $C_1=1$  去或门，或门输出端  $C_n$  为“1”。即当  $A=1$ 、 $B=1$ 、低位进位数  $C_{n-1}=1$  时，全加器的  $S_n=1$ ，高位进位数  $C_n=1$ 。

全加器的真值表见表 4-12。

表 4-12 全加器的真值表

输 入			输 出		输 入			输 出	
A	B	$C_{n-1}$	$S_n$	$C_n$	A	B	$C_{n-1}$	$S_n$	$C_n$
0	0	0	0	0	1	0	0	1	0
0	0	1	1	0	1	0	1	0	1
0	1	0	1	0	1	1	0	0	1
0	1	1	0	1	1	1	1	1	1

4.5.3 多位加法器

半加器和全加器只能实现 1 位二进制数相加，而实际更多的是多位二进制数进行相加，这就要用到多位加法器。多位加法器由多个全加器或者全加器与半加器混合组成。

1. 结构与原理

图 4-27 所示为 4 位串行二进制加法器的电路结构，它由 4 个全加器  $\Sigma_1 \sim \Sigma_4$  组成。

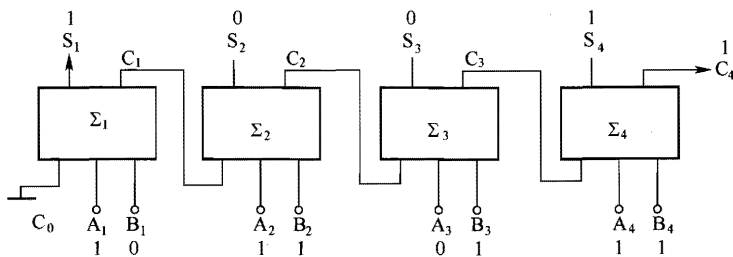


图 4-27 4 位串行二进制加法器

下面以“ $A_4A_3A_2A_1+B_4B_3B_2B_1$ ”为例来说明其工作过程，这里设  $A_4A_3A_2A_1=1011$ 、 $B_4B_3B_2B_1=1110$ 。

多位加法器的相加过程就像用竖式计算一样，先将低位数相加，得到和数，若有进位，则向高位进位，高位相加时则要考虑有无进位，1011 与 1110 相加的竖式计算过程如下

$$\begin{array}{r} 1011 \\ + 1110 \\ \hline 11001 \end{array}$$

在全加器  $\Sigma_1$  中进行“ $A_1+B_1(1+0)$ ”运算，其进位数  $C_1=0$ （无进位），和数  $S_1=1$ ；在全加器  $\Sigma_2$  中进行“ $A_2+B_2(1+1)$ ”运算，其进位数  $C_2=1$ （有进位），和数  $S_2=0$ ；在全加器  $\Sigma_3$  中进行“ $A_3+B_3(0+1)$ ”并加低位进位数  $C_2=1$  运算，得到和数  $S_3=0$ ，同时产生高位进位数  $C_3=1$ ；在全加器  $\Sigma_4$  中进行“ $A_4+B_4(1+1)$ ”并加  $\Sigma_3$  送来的进位数  $C_3=1$  运算，结果和数  $S_4=1$ ，高位进位数  $C_4=1$ 。

通过上述过程，4 位二进制加法器的输出端  $C_4S_4S_3S_2S_1=11001$ ，从而完成了“ $1011+1110=11001$ ”的运算。

## 2. 常用多位加法器芯片

74LS83 是一个常用的 4 位加法器芯片，内部由 4 个全加器组成。74LS83 各引脚功能如图 4-28 所示。

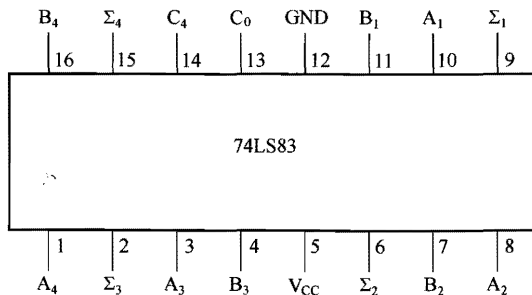


图 4-28 多位加法器芯片 74LS83

74LS83 的  $\Sigma_4 \sim \Sigma_1$  端分别为各全加器的和输出端，相当于图 4-27 中的  $S_4 \sim S_1$  端； $A_4 \sim$



$A_1$  端和  $B_4 \sim B_1$  端用于输入两组相加数； $C_0$  端用于接受低位进位数，不使用时接地， $C_4$  为最高位进位数。

使用举例：在使用 74LS83 进行 “1011+1110=11001” 运算时，可让  $A_4A_3A_2A_1=1011$ 、 $B_4B_3B_2B_1=1110$ ，并将  $C_0$  端接地（即让  $C_0=0$ ），芯片对两组数进行相加运算后， $\Sigma_4\Sigma_3\Sigma_2\Sigma_1=1001$ ，同时  $C_4=1$ 。

## 4.6 数值比较器

在数字电子设备中，经常需要比较两个数值的大小及是否相等，能完成数据比较功能的逻辑电路称为数值比较器。数值比较器有两类：一种是等值比较器；另一种是数值比较器。

### 4.6.1 等值比较器

等值比较器的功能是检验数据是否相等。等值比较器可分为一位等值比较器和多位等值比较器。

#### 1. 1 位等值比较器

1 位等值比较器如图 4-29 所示，其中图 4-29 (a) 所示为异或非门构成的 1 位等值比较器，图 4-29 (b) 所示为与或非门构成的 1 位等值比较器。

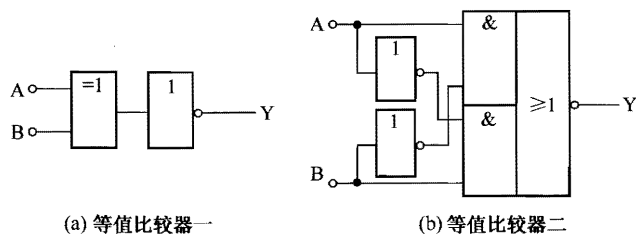


图 4-29 1 位等值比较器

异或非门又称同或门，在第 2 章已经介绍过，其逻辑功能是：当  $A$ 、 $B$  输入相同（相等）时，输出为 “1”，否则为 “0”。因此可以根据异或非门的输出来判断  $A$ 、 $B$  是否相等，在图 4-29 (a) 中，当输出为 “1” 时，表明  $A$ 、 $B$  相等；当输出为 “0” 时，表明  $A$ 、 $B$  不相等。

图 4-29 (b) 中的等值比较器由两个非门和一个与或非门构成。与或非门的逻辑功能是：两个与门中有一组全为 “1” 时，输出就为 “0”，否则为 “1”。在图 4-29 (b) 中，如果  $A$ 、 $B$  相同（等值）时，两个与门的两个输入值必不相同（即  $A$ 、 $B$  相同时， $A$  和  $\bar{B}$  必不相同，



B 和  $\bar{A}$  也不相同), 输出  $Y=1$ ; 如果 A、B 不相同, 两个与门的两个输入值必然相同, 输出  $Y=0$ 。

## 2. 多位等值比较器

在实际的数字电路中经常需要进行多位数值的比较, 这就要用到多位等值比较器。

图 4-30 所示为 4 位等值比较器, 它由 4 个同或门 (即异或非门) 和 1 个与门构成的。

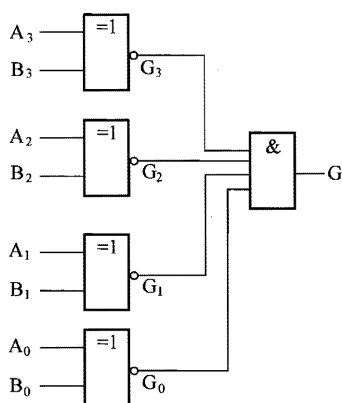


图 4-30 4 位等值比较器

这里以比较  $A_3A_2A_1A_0$  和  $B_3B_2B_1B_0$  两个数为例来说明比较器的工作过程。比较器采用逐位比较的方法来判断整个 4 位数是否相等。当  $A_0$ 、 $B_0$  相等时, 同或门  $G_0$  输出 “1” 到与门, 同样地, 只有  $A_1$  和  $B_1$ 、 $A_2$  和  $B_2$ 、 $A_3$  和  $B_3$  都相等, 同或门  $G_1$ 、 $G_2$ 、 $G_3$  都输出 “1” 到与门, 与门才会输出 “1”。如果  $A_0$  和  $B_0$ 、 $A_1$  和  $B_1$ 、 $A_2$  和  $B_2$ 、 $A_3$  和  $B_3$  中有一组不相同, 相应的同或门就会输出 “0” 到与门, 与门则输出 “0”。即当两个 4 位数各位数都相同时, 这两个 4 位数才相等, 比较器输出为 “1”; 否则, 比较器输出为 “0”。

## 4.6.2 数值比较器

数值比较器又称为大小比较器, 它不但能检验两个数据是否相等, 还能比较它们的大小。

### 1. 1 位数值比较器

1 位数值比较器电路结构如图 4-31 所示, 它由一个异或非门、两个与门和两个非门构成的。

数值比较过程如下:

当  $A=B$ , 即 A、B 同时为 “1” 或 “0” 时, 与门  $G_3$  两个输入不同, 其输出  $Y_3=0$ ; 与门  $G_2$  两个输入也不同, 其输出  $Y_2=0$ ; 而与异或非门两输入相同, 其输出  $Y_1=1$ 。

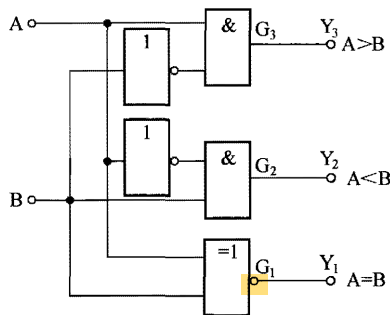


图 4-31 1 位数值比较器的电路结构

当  $A > B$ ，即  $A=1$ 、 $B=0$  时，与门  $G_3$  两个输入都为“1”，其输出  $Y_3=1$ ；与门  $G_2$  两个输入均为“0”，其输出  $Y_2=0$ ；异或非门两输入不同，其输出  $Y_1=0$ 。

当  $A < B$ ，即  $A=0$ 、 $B=1$  时，与门  $G_3$  两个输入都为“0”，其输出  $Y_3=0$ ；与门  $G_2$  两个输入均为“1”，其输出  $Y_2=1$ ；异或非门两输入不同，其输出  $Y_1=0$ 。

也就是说，当数值比较器的  $Y_1=1$  时，表明输入值  $A=B$ ；当数值比较器的  $Y_3=1$  时，表明输入值  $A > B$ ；当数值比较器的  $Y_2=1$  时，表明输入值  $A < B$ 。

2. 多位数值比较器

(1) 多位数值比较原理

多位数值比较器采用由高位到低位逐次比较的方式，当高位数值大时，则整个多位数数值都大，若高位相等，再比较下一位，下一位数值大的整个多位数数值大，这样依次逐位进行比较，当所有的位都相等时，则两个多位数相等。图 4-32 所示是一个 4 位数值比较器框图。

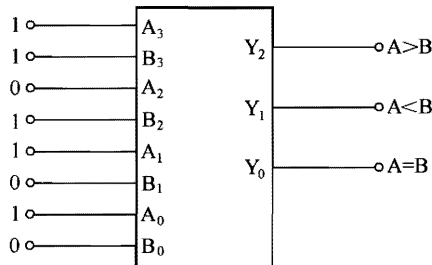


图 4-32 4 位数值比较器框图

4 位数值比较器内部逻辑电路比较复杂，这里只简单说明它的比较过程。设其中的一个 4 位数  $A_3A_2A_1A_0$  为 1011，另一个 4 位数  $B_3B_2B_1B_0$  为 1100，比较器首先比较  $A_3$  和  $B_3$  的大小，因为  $A_3$  和  $B_3$  相等，比较器接着比较  $A_2$  和  $B_2$ ，由于  $A_2=0$ ，而  $B_2=1$ ， $A_2 < B_2$ ，所以数  $A_3A_2A_1A_0$  (1011) 小于  $B_3B_2B_1B_0$  (1100)，比较器从  $Y_1$  端输出“1”，而  $Y_2$ 、 $Y_0$  均为“0”。





(2) 多位数值比较器芯片

74LS85 是一个常用的 4 位数值比较器芯片，如图 4-33 所示，其真值表见表 4-13。

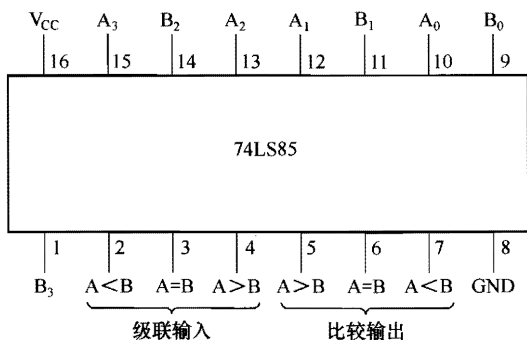


图 4-33 4 位数值比较器芯片 74LS85

表 4-13 74LS85 的真值表

比 较 输 入				级 联 输 入			比 较 输 出		
A <sub>3</sub> , B <sub>3</sub>	A <sub>2</sub> , B <sub>2</sub>	A <sub>1</sub> , B <sub>1</sub>	A <sub>0</sub> , B <sub>0</sub>	A>B	A<B	A=B	A>B	A<B	A=B
A <sub>3</sub> >B <sub>3</sub>	×	×	×	×	×	×	H	L	L
A <sub>3</sub> <B <sub>3</sub>	×	×	×	×	×	×	L	H	L
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> >B <sub>2</sub>	×	×	×	×	×	H	L	L
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> <B <sub>2</sub>	×	×	×	×	×	L	H	L
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> =B <sub>2</sub>	A <sub>1</sub> >B <sub>1</sub>	×	×	×	×	H	L	L
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> =B <sub>2</sub>	A <sub>1</sub> <B <sub>1</sub>	×	×	×	×	L	H	L
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> =B <sub>2</sub>	A <sub>1</sub> =B <sub>1</sub>	A <sub>0</sub> >B <sub>0</sub>	×	×	×	H	L	L
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> =B <sub>2</sub>	A <sub>1</sub> =B <sub>1</sub>	A <sub>0</sub> <B <sub>0</sub>	×	×	×	L	H	L
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> =B <sub>2</sub>	A <sub>1</sub> =B <sub>1</sub>	A <sub>0</sub> =B <sub>0</sub>	H	L	L	H	L	L
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> =B <sub>2</sub>	A <sub>1</sub> =B <sub>1</sub>	A <sub>0</sub> =B <sub>0</sub>	L	H	L	L	H	L
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> =B <sub>2</sub>	A <sub>1</sub> =B <sub>1</sub>	A <sub>0</sub> =B <sub>0</sub>	L	L	H	L	L	H
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> =B <sub>2</sub>	A <sub>1</sub> =B <sub>1</sub>	A <sub>0</sub> =B <sub>0</sub>	×	×	H	L	L	H
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> =B <sub>2</sub>	A <sub>1</sub> =B <sub>1</sub>	A <sub>0</sub> =B <sub>0</sub>	H	H	L	L	L	L
A <sub>3</sub> =B <sub>3</sub>	A <sub>2</sub> =B <sub>2</sub>	A <sub>1</sub> =B <sub>1</sub>	A <sub>0</sub> =B <sub>0</sub>	L	L	L	H	H	L

74LS85 的 A<sub>3</sub>~A<sub>0</sub> 和 B<sub>3</sub>~B<sub>0</sub> 为比较输入端，可同时输入两组 4 位二进制数；74LS85 的⑤、⑥、⑦脚为比较输出端，②、③、④脚为级联输入端，当使用多片 74LS85 组成 8 位或更高位数值比较器时，高位片 74LS85 级联输入端接低位片的比较输出端。



从真值表可以看出,当 74LS85 的  $A_3A_2A_1A_0 \neq B_3B_2B_1B_0$  时,级联输入端输入无效(即不管输入何值都不会影响比较输出),当 74LS85 的  $A_3A_2A_1A_0 = B_3B_2B_1B_0$  时,级联输入端输入会影响比较输出。

### (3) 数值比较器的扩展

在进行多位数值比较时,单个芯片常常无法胜任,采用多个芯片进行级联可以解决这个问题。图 4-34 所示是一个由两片 74LS85 级联构成的 8 位数值比较器,从图中可以看出,低位片的级联输入端均接地,而比较输出端接高位片的级联输入端。

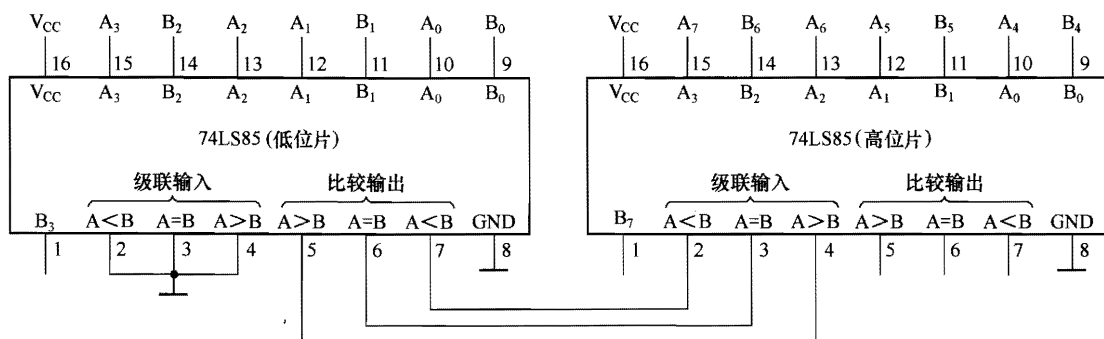


图 4-34 由两片 74LS85 级联构成的 8 位数值比较器

## 4.7 数据选择器

数据选择器又称为多路选择开关,它是一个多路输入、一路输出的电路,其功能是在选择控制信号的作用下,能从多路输入的数据中选择其中一路输出。数据选择器在音响设备、电视机、计算机和通信设备中广泛应用。

### 4.7.1 结构与原理

图 4-35 (a) 所示是典型的四选一数据选择器电路结构,图 4-35 (b) 所示为其等效图。

$A_0$ 、 $A_1$ 、 $A_2$ 、 $A_3$  为数据选择器的 4 个输入端,  $Y$  为数据选择器的输出端,  $S_0$ 、 $S_1$  为数据选择控制端,用来控制数据选择器选择四路数据中的某一路数据输出。为了分析更直观,假设数据选择器的四路输入端  $A_0$ 、 $A_1$ 、 $A_2$ 、 $A_3$  分别输入 1、1、1、1。

当  $S_0=0$ 、 $S_1=1$  时,  $S_1$  的“1”经非门后变成“0”送到与门  $G_0$  和  $G_1$  的输入端,与门  $G_0$  和  $G_1$  关闭(与门只要有一个输入为“0”,输出就为“0”), $A_0$  和  $A_1$  数据“1”均无法通过; $S_0$  的“0”一路直接送到与门  $G_3$  输入端,与门  $G_3$  关闭, $A_3$  数据“1”无法通过与门  $G_3$ ;而与门  $G_2$  两个输入端则输入由  $S_1$  直接送来的“1”和由  $S_0$  经非门转变成“1”,故与门



$G_2$  开通， $G_2$  输出“1”，该数据“1”送到或门  $G_4$ ， $G_4$  输出“1”。也就是说，当  $S_0=0$ 、 $S_1=1$  时， $A_2$  数据能通过与门  $G_2$  和或门  $G_4$  从  $Y$  端输出。

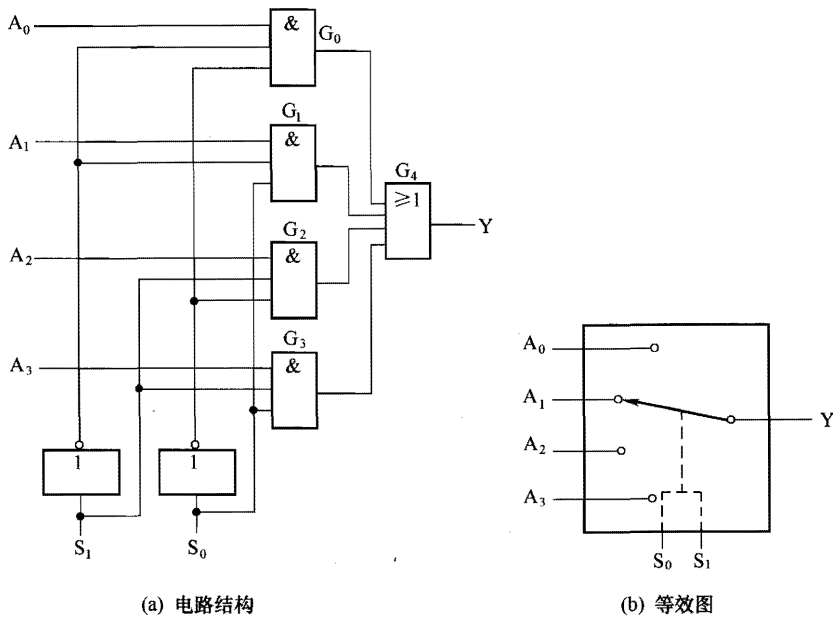


图 4-35 四选一数据选择器

当  $S_0=1$ 、 $S_1=1$  时，与门  $G_3$  开通， $A_3$  数据被选择输出。

当  $S_0=0$ 、 $S_1=0$  时，与门  $G_0$  开通， $A_0$  数据被选择输出。

当  $S_0=1$ 、 $S_1=0$  时，与门  $G_1$  开通， $A_1$  数据被选择输出。

四选一数据选择器的真值表见表 4-14。表中的“×”表示无论输入什么值（1 或 0）都不影响输出结果。

表 4-14 四选一数据选择器的真值表

选择控制输入		输 入				输 出
$S_1$	$S_0$	$A_0$	$A_1$	$A_2$	$A_3$	$Y$
0	0	$A_0$	×	×	×	$A_0$
0	1	×	$A_1$	×	×	$A_1$
1	0	×	×	$A_2$	×	$A_2$
1	1	×	×	×	$A_3$	$A_3$

除了四选一数据选择器外，还有八选一数据选择器和十六选一数据选择器。八选一数据选择器需要 3 个数据选择控制端，而十六选一数据选择器需要 4 个数据选择控制端。



4.7.2 常用数据选择器芯片

74LS153 是一个常用的双四选一数据选择器芯片，如图 4-36 所示，其真值表见表 4-15。

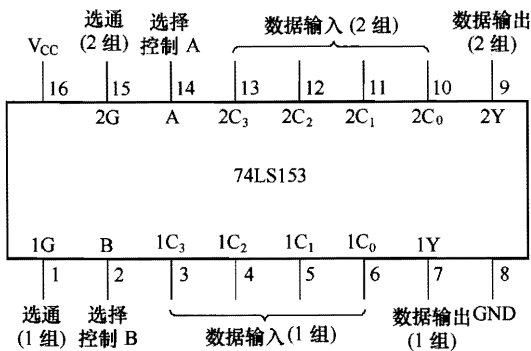


图 4-36 74LS153 的各引脚功能

表 4-15 74LS153 的真值表

选 择 控 制		数 据 输 入				选 通	数 据 输 出
B	A	C <sub>0</sub>	C <sub>1</sub>	C <sub>2</sub>	C <sub>3</sub>	G	Y
×	×	×	×	×	×	H	L
L	L	L	×	×	×	L	L
L	L	H	×	×	×	L	H
L	H	×	L	×	×	L	L
L	H	×	H	×	×	L	H
H	L	×	×	L	×	L	L
H	L	×	×	H	×	L	H
H	H	×	×	×	L	L	L
H	H	×	×	×	H	L	H

74LS153 内部有两个完全相同的四选一数据选择器，C<sub>3</sub>~C<sub>0</sub> 为数据输入端，Y 为数据输出端。1G、2G 分别是 1 组、2 组选通端，当 1G=0 时，第 1 组数据选择器工作，当 2G=0 时，第 2 组数据选择器工作，当 1G、2G 均为高电平时，1、2 组数据选择器均不工作。

A、B 为选择控制端，在 G 端为低电平时，可以选择某路输入数据并输出。例如当 1G=0 时，若 AB=10，1C<sub>1</sub> 端输入的数据会被选择并从 1Y 端输出。

## 4.8 奇偶校验器

在数字电子设备中，数字电路之间经常要进行数据传输，由于受一些因素的影响，数据在传送过程中可能会产生错误，从而会引起设备工作不正常。为了解决这个问题，常常在数据传送电路中设置奇偶校验器。

### 4.8.1 奇偶校验原理

奇偶校验是检验数据传输是否发生错误的方法之一。它是通过检验传递数据中“1”的个数是奇数还是偶数来判断传递数据是否有错误。

奇偶校验有奇校验和偶校验之分。对于奇校验，若数据中有奇数个“1”，则校验结果为0，若数据中有偶数个“1”，则校验结果为1；对于偶校验，若数据中有偶数个“1”，则校验结果为0，若数据中有奇数个“1”，则校验结果为1。

下面以图4-37所示的8位并行传递奇偶校验示意图为例来说明奇偶校验原理。

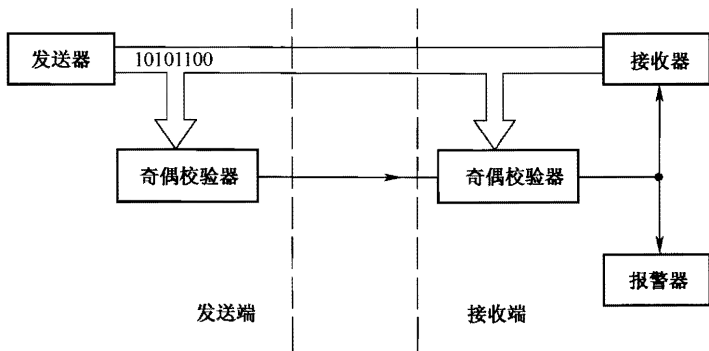


图 4-37 8 位并行传递奇偶校验示意图

在图4-37中，发送器通过8根数据线同时向接收器传递8位数据，这种通过多根数据线同时传递多位数的数据传递方式称为并行传递。发送器在往接收器传递数据的同时，也会把数据传递给发送端的奇偶校验器，假设发送端要传递的数据是10101100。

若图4-37中所示的奇偶校验器为奇校验，发送器的数据10101100送到奇偶校验器，由于数据中的“1”的个数是偶数个，奇偶校验器输出1，它送到接收端的奇偶校验器，与此同时，发送端的数据10101100也送到接收端的奇偶校验器，这样送到接收端的奇偶校验器的数据中“1”的个数为奇数个（含发送端奇偶校验器送来的“1”），接收端的奇偶校验器输出0，它去控制接收器工作，接收发送过来的数据。如果数据在传递过程



中发生了错误,数据由 10101100 变为 10101000,那么送到接收端奇偶校验器的数据中的“1”的个数是偶数个(含发送端奇偶校验器送来的“1”),校验器输出为 1,它一方面控制接收器,禁止接收器接收错误的数,同时还去触发报警器,让它发出数据错误报警。

若图 4-37 中的奇偶校验器为偶校验,发送器的数据为 10101100 时,发送端的奇偶校验器会输出 0。如果传递的数据没有发生错误,接收端的奇偶校验器会输出 0;如果传递的数据发生错误,10101100 变成了 10101000,接收端的奇偶校验器会输出 1。

### 4.8.2 奇偶校验器

奇偶校验器可采用异或门构成,2 位奇偶校验器和 3 位奇偶校验器分别如图 4-38 (a)、(b) 所示。

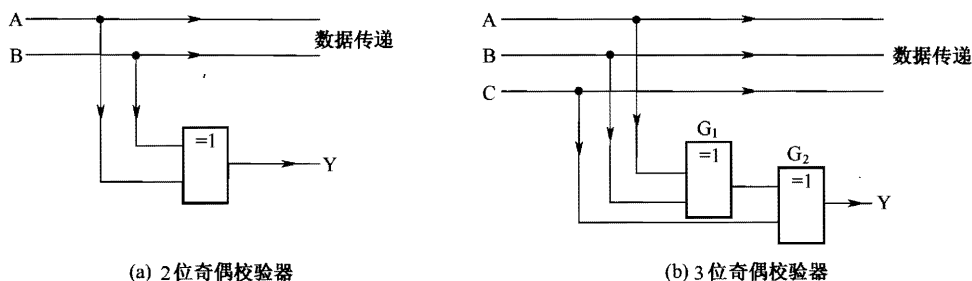


图 4-38 奇偶校验器

图 4-38 中所示的奇偶校验器是由异或门构成的,异或门具有的特点是:输入相同时输出为“0”,输入相异时输出为“1”。图 4-36 (a) 所示的 2 位奇偶校验器由一个异或门构成,当 A、B 都输入“1”,即输入的“1”为偶数个时,输出  $Y=0$ ;当 A、B 中只有一个为“1”,即输入的“1”为奇数个时,输出  $Y=1$ 。

图 4-38 (b) 所示的 3 位奇偶校验器由两个异或门构成,当  $A=1$ 、 $B=1$ 、 $C=1$  时,输出  $Y=1$ ;当  $A=1$ 、 $B=1$ ,而  $C=0$  时,异或门  $G_1$  输出为“0”,异或门  $G_2$  输出为“0”,即输入的“1”为偶数个时,输出  $Y=0$ 。

以上两种由异或门组成的奇偶校验器具有偶校验功能,如果将异或门换成异或非门组成奇偶校验器,它就具有奇校验功能。

从图 4-37 可以看出,由于接收端的奇偶校验器除了要接收传递的数据外,还要接收发送端奇偶校验器送来的校验位,所以接收端的奇偶校验器的位数较发送端的多 1 位。

下面以图 4-39 所示电路为例进一步说明奇偶校验器的实际应用。

图 4-39 中所示的发送器要送 2 位数  $AB=10$  到接收器, $A=1$ 、 $B=0$  一方面通过数据线往



接收器传递，另一方面送到发送端的奇偶校验器，该校验器为偶校验，它输出的校验位为1。校验位1与A=1、B=0送到接收端奇偶校验器，此校验器校验输出为“0”，该校验位0去控制接收器，让接收器接收数据线送到的正确数据。

如果数据在传递过程中，AB由10变为11（注：送到发送端奇偶校验器的数据AB是正确的，仍为10，只是数据传送到接收器的途中发生了错误，由10变成11），发送端的奇偶校验器输出的校验位仍为1，而由于传送到接收端的数据10变成了11，所以接收端的奇偶校验器输出校验位为1，它去禁止接收器接收错误的数

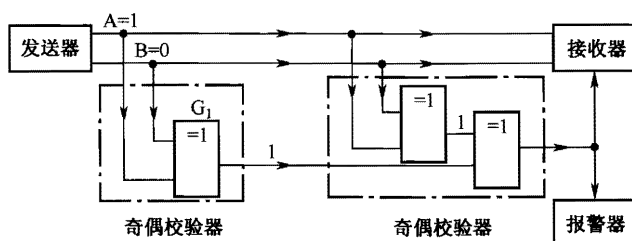


图 4-39 2 位并行传递奇偶校验电路

## 习题 4

### 一、填空题

- 组合逻辑电路在任何时刻的输出只由\_\_\_\_\_决定，而与\_\_\_\_\_无关，电路没有记忆功能。
- 组合逻辑电路分析的一般步骤为：①\_\_\_\_\_；②\_\_\_\_\_；③\_\_\_\_\_；④\_\_\_\_\_。
- 组合逻辑电路设计的一般步骤为：①\_\_\_\_\_；②\_\_\_\_\_；③\_\_\_\_\_；④\_\_\_\_\_。
- 在数字电路中，编码是指\_\_\_\_\_的过程。编码器是指\_\_\_\_\_。
- 普通编码器任何时刻允许输入\_\_\_\_\_，若输入\_\_\_\_\_，编码输出就会产生\_\_\_\_\_。优先编码器同一时刻允许输入\_\_\_\_\_，但仅对输入信号中\_\_\_\_\_信号进行编码输出。
- 译码是\_\_\_\_\_的过程，它是\_\_\_\_\_的逆过程，\_\_\_\_\_的电路称为译码器。
- 2 位二进制译码器可以将 2 位代码译成\_\_\_\_\_种输出状态，故又称\_\_\_\_\_译码器，而  $n$  位二进制译码器可以译成\_\_\_\_\_种输出状态。
- 显示译码器的功能是\_\_\_\_\_。
- 根据七段数码管内部发光二极管公共端不同，可分为\_\_\_\_\_接法和\_\_\_\_\_接法，对于\_\_\_\_\_接法的显示器，需要给发光二极管加低电平才能发光；而对于\_\_\_\_\_接法的显示器，需要给发光二极管加





高电平才能发光。

10. 多位荧光数码显示器采用了\_\_\_\_\_显示原理,它利用了荧光粉的\_\_\_\_\_和人眼\_\_\_\_\_。
11. \_\_\_\_\_运算称为半加,全加是指\_\_\_\_\_运算,多位加法器由\_\_\_\_\_组成,也可由\_\_\_\_\_与\_\_\_\_\_混合组成。
12. 等值比较器的功能是\_\_\_\_\_。数值比较器又称为\_\_\_\_\_,它不但能\_\_\_\_\_,还能\_\_\_\_\_。
13. 数据选择器又称为\_\_\_\_\_,其功能是\_\_\_\_\_。
14. 奇偶校验有奇校验和偶校验之分。对于奇校验,若数据中有奇数个“1”,则校验结果为\_\_\_\_\_;若数据中有偶数个“1”,则校验结果为\_\_\_\_\_。对于偶校验,若数据中有偶数个“1”,则校验结果为\_\_\_\_\_;若数据中有奇数个“1”,则校验结果为\_\_\_\_\_。
15. 组合逻辑电路种类很多,常见的有\_\_\_\_\_,\_\_\_\_\_,\_\_\_\_\_,\_\_\_\_\_,\_\_\_\_\_和\_\_\_\_\_等。

## 二、设计题

有 A、B 两个单位举行联谊娱乐活动,要求每个单位都派出数量相等的男女各组成一队(如 A 单位 9 男 9 女组成一队, B 单位也应 9 男 9 女组成一队),两队用隔板分开,相互之间无法看见对方,在隔板两边各有一个开关,上方有一只灯泡,两支队伍以隔板为中心排成两个纵队,纵队中男女排列随意。游戏开始时,每队同时各派出一人操作本方的开关,男士要将开关上拨,女士则应将开关下拨。如果隔板上方灯亮,说明隔板双方为男女,双方应离开队伍到指定地点会面;如果灯泡不亮,则双方队员退回本队伍末尾等待下一轮。

请按照组合逻辑电路一般设计步骤设计出逻辑电路来实现上述功能。