4、组合逻辑电路



4.1 概述



4.2 组合逻辑电路的分析和设计方法



4.3 若干常用中规模组合逻辑电路



4.4 组合逻辑电路中的竞争- 冒险现象



概述

4.1

根据逻辑功能的不同，可把数字电路分为组合逻 辑电路 (**Combinational** **Logic** **Circuit**)和 时序逻辑电 路 (**Sequential** **Logic** **Circuit**)两大类。

一、组合逻辑电路的特点

功能上：任意时刻的输出仅仅取决于该时刻的输 入，而与电路原来的状态无关，即无记忆功能。

电路结构上：只由逻辑门组成，不包含记忆元件， 输出和输入之间无反馈。

二、逻辑功能的描述方式

① 逻辑函数表达式



组合逻辑电路的框图

由框图可知，输入与输出之间的逻辑关系可用一 组逻辑函数表示：

(*y*1 = *f*1 (*a*1 , *a*2 , … , *an* )

|

〈： ：

| *ym* = *fm* (*a*1 , *a*2 , … , *an* )

即*Y* = *F*(*A*)

组合逻辑电路的其它描述方式**——**

② 逻辑电路图

③ 真值表

④ 卡诺图

⑤ 波形图

⑥ 语言描述



组合逻辑电路的分析和设计方法



4.2

分析和设计是一对逆过程。



**4.2.1** 组合逻辑电路的分析方法

所谓 “分析” ，即根据逻辑电路找出电路的逻辑功能。

分析的目的：求出逻辑功能或证明给定的逻辑功能正确

与否。

一、分析步骤：

(**1**)分别用符号标注各级门的输出端。

(**2**)从输入端到输出端逐级写出输出变量对输入变量的

逻辑表达式，最后得到输入变量表示的输出函数表达式。 需 要时用卡诺图或公式化简法化简逻辑函数成最简形式。

(**3**)列真值表。

(**4**)根据真值表或函数表达式确定电路的逻辑功能。 有 时功能难以用简练的语言描述，此时列真值表即可。



|  |
| --- |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

二、分析举例

例分析图示电路的逻辑功能。



> 1

>1

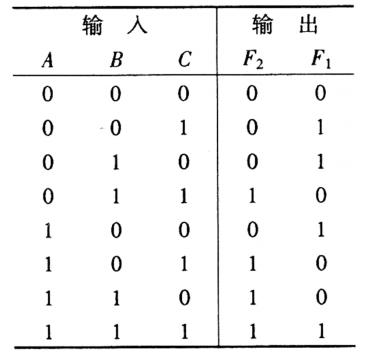


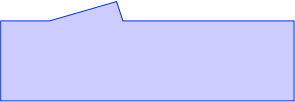
>1





真值表







**0**

**0**



**1**

**1**



**0**

**1**



**0**



**1**

**0**



**0**



**1**

**1**



**0**

**1**

**0**



**1**



**1**



**1**



**1**

**001** **010** **100** **111**

根据异或功能可列出真值表如右表； 也可先求标准与或式，然后得真值表。后 者是分析电路的常用方法，下面介绍之。

***A*** 、***B*** 、***C*** 三个输入变量中，有奇数个 **1** 时，输出为 **1**，否则输出为 **0**。因此，图示 电路为三位判奇电路，又称奇校验电路。

通过分析真值表 特点来说明功能。

例 分析下图所示逻辑

(**2**)列逻辑函数真值表

|  |  |  |  |
| --- | --- | --- | --- |
| 输 入 | | | 输 出 |
| ***A*** ***B*** ***C*** | | | ***Y*** |
| **0**  **0** | **0**  **1**  **0** | **0**  **1**  **1** | **0** |
| **1** | **1** | **0** | **0** |
|  |  |  |  |

电路的功能。



***Y*1**

***B***

***Y***

***C***

***A***

(**1**)写出输出逻辑函数式

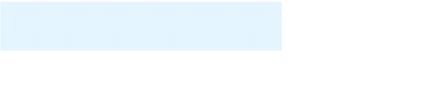
解：

***Y*1** = ***A***  ***B***

***Y*** =  ***C***

= **(*A***  ***B*)*C*** + ***A***  ***B*** . ***C***

= ***ABC*** + ***ABC*** + ***ABC*** + ***ABC***



(**3**)分析逻辑功能



**4.2.2** 组合逻辑电路的设计方法

所谓 “设计” ：即根据给出的实际逻辑问题，求出实 现这个逻辑功能的最简逻辑电路。

所谓 “最简” ：是指所用器件最少，器件种类最少， 而且器件之间的连线也最少。

一、设计步骤

(**1**)进行逻辑抽象

① 分析事件的因果关系，确定输入和输出变量；

② 定义逻辑状态的含意；

③ 根据因果关系列出真值表；

例、 三人裁判举重比赛，一个主裁判，两个副裁判。认为杠铃 举上时，各裁判按自己前面的电键(为1)，否则不按(为0)； 裁判结果用红绿灯表示，红绿灯均亮(为1)表示“完全举上”， 只红灯亮表示“需研究录像决定”，其余为未举上。

(1)三个裁判均按下自己的电键，红绿灯全亮；

(2)两个裁判(其中一个为主裁判)按下自己的电键，红绿灯全亮；

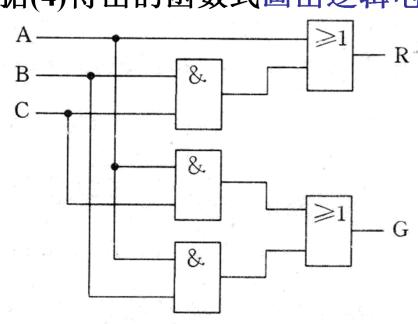
(3)两个副裁判或一个主裁判按下自己的电键，只红灯亮；

(4)其余情况红绿灯全灭。

试用两级与或电路实现满足上述四种要求的逻辑控制电路。

(**3**)选定器件的类型：**SSI** 、**MSI**或**PLD**等；

(**4**) 根据器件类型将逻辑函数化简或变换成适 当的形式。

(**5**) 根据**(4)**得出的函数式画出逻辑电路图。

举重裁判电路逻辑图

二、完整设计举例

例 :试用与非门设计一个将8421-BCD码转换为余3码的码制 转换电路。

|  |  |  |
| --- | --- | --- |
| **BCD**码十 进制数码 | **8421**码 | 余**3**码 |
| **0** | **0000** | **0011** |
| **1** | **0001** | **0100** |
| **2** | **0010** | **0101** |
| **3** | **0011** | **0110** |
| **4** | **0100** | **0111** |
| **5** | **0101** | **1000** |
| **6** | **0110** | **1001** |
| **7** | **0111** | **1010** |
| **8** | **1000** | **1011** |
| **9** | **1001** | **1100** |



B1B0

B3B2 00 01 11 10

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  | |  |  | | |
|  |  | 1 |  | 1 |  | 1 | |
|  |  |  | |
| 1 |  | |  | | |  | |
|  |  | |  | | |  | |
|  |  | 1 |  |  |  |  |  |
|  | |  |  |
|  | | | | | | |  |

00

01

11

10

00

01

11

10

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 00  01  11  10 | 1 |  |  | 1 |
| 1 |  |  | 1 |
|  |  |  |  |
| 1 |  |  |  |

00

01

11

10

B1B0

B3B2 00 01 11 10

B1B0 B1B0

B3B2 00 01 11 10 B3B2 00 01 11 10

Y

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  | 1 | 1 | 1 |
|  |  |  |  |
| 1 | 1 |  |  |

3

2

Y

|  |  |  |  |
| --- | --- | --- | --- |
| 1 |  | 1 |  |
| 1 |  | 1 |  |
|  |  |  |  |
| 1 |  |  |  |

Y1 Y0

|  |
| --- |
| 4.3 若干常用中规模组合逻辑电路 |

目前，一些常用的逻辑电路已经制成了中、小规 模集成化电路产品。



**4.3.1** 编码器(Encoder)

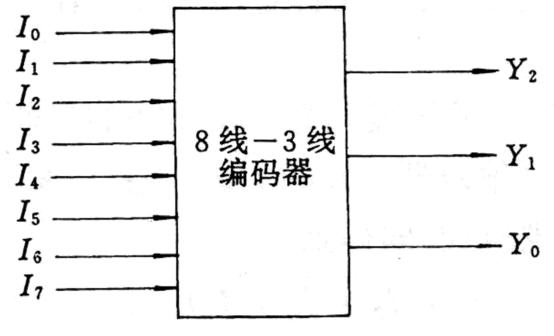
“编码”：即为了区分一系列不同的事物，将其 中的每个事物用一个二值代码表示。

编码器的逻辑功能：把输入的每一个高、低电平 信号变成一个对应的二进制代码。

一、普通编码器(Common Encoder)

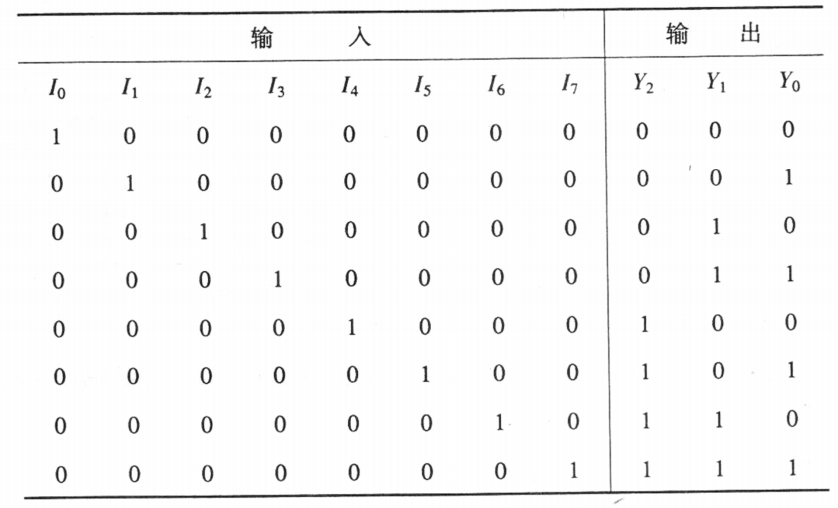
特点： 任何时刻只允许输入一个编码信号，否则将 发生混乱。

**3**位二进制普通编码器示例：



3位二进制普通编码器框图

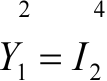
3位二进制普通编码器真值表



由于普通编码器在任何时刻 *I*0 ~ *I*7 当中仅有一个取值为**1**，

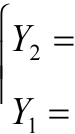
即只有真值表中所列的**8**种状态，而且它的( 28  8 )种状

态均为约束项。因此，由真值表可得到逻辑式：

〈|  + *I* + *I* + *I* | *Y*0 = *I*1 + *I*3 + *I*5 + *I*7

(*Y* = *I* + *I* + *I* + *I*

或门实现

〈 .. .. ..

*Y*0 = *I*1 . *I*3 . *I*5 . *I*7

与非门实现

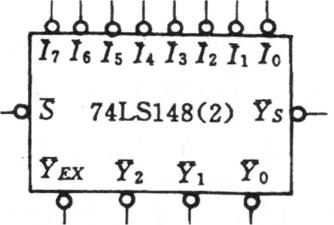
思考**1**：如何用与非门实现**8421-BCD**码普通编码器？



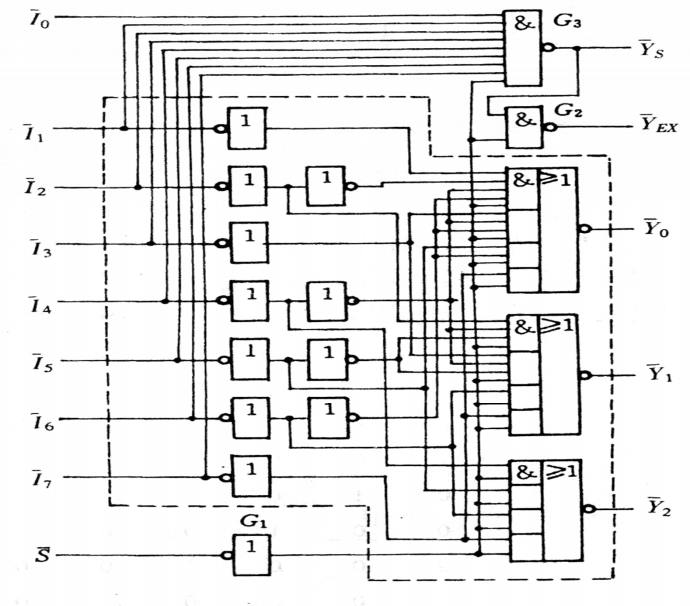
二、优先编码器(Priority Encoder)

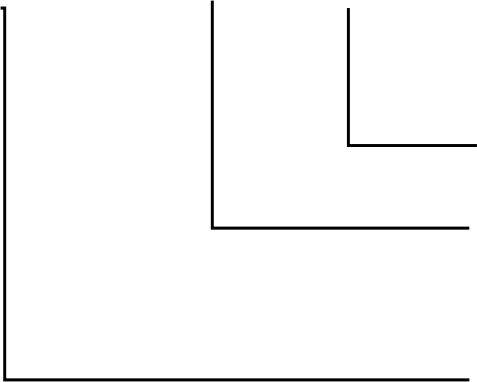
特点：允许同时输入两个以上编码信号。不过在设计 优先编码器时已经将所有的输入信号按优先顺序排了队， 当几个输入信号同时出现时， 只对其中优先权最高的一个 进行编码。

下面以**8**线**-3**线优先编码器**74LS148**为例分析优先编 码器的工作原理。 **74LS148**框图(俯视图)如下：



74LS148的俯视图





封装形式

工作温度范围

器件系列及品种代号

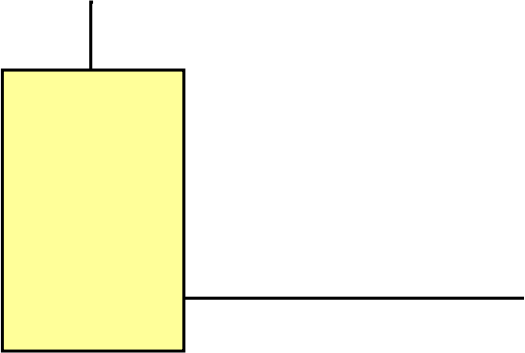
器件类型

|  |
| --- |
| E:-40~85ºC  C:0~70ºC  R: -55~85ºC  M: -55~125ºC |

|  |
| --- |
| **D:**陶瓷直插 **P:**塑料直插 **W:**陶瓷扁平 **B:**塑料扁平  **T:**金属圆形 |

国产半导体集成电路型号命名法

C T 74LS148 E D



T:TTL

H:HTL

E:ECL

C:CMOS

M:Memory

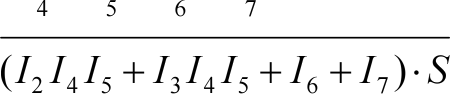
符合中国标准

74LS148功能表



由逻辑图和功能表可知：

(**1**) *S*为选通输入端，只有在*S* = 0 (即**S=1**时) 编码器才能正常工作；而在 *S* = 1 (即**S=0**时) 所有的输出端均被封锁在高电平。且此时，输出

〈| = 

(*Y* = (*I* + *I* + *I* + *I* ) . *S*

*Y*0 = (*I*1*I*2 *I*4 *I*6 + *I*3 *I*4 *I*6 + *I*5 *I*6 + *I*7 ) . *S*

(由功能表第一行体现) 。

时， *Y* 才为低电平。 *Y* 为低电平表示 “电路工

(**2**) *YS* 为选通输出端，其表达式为：

*YS* = *I*0 *I*1 *I*2 *I*3 *I*4 *I*5 *I*6 *I*7  *S*

此式表明：只有当所有的编码输入端均为高 电平(即没有编码输入)，且**S=1** ( *S* = 0 )

作，但无编码输入”。

*S* *S*

(功能表第二行所示) 。

(**3**) *YEX* 为扩展端，用于扩展编码功能，其表达 式为：

*YEX* = (*I*0 + *I*1 + *I*2 + *I*3 + *I*4 + *I*5 + *I*6 +*I*7) . *S*

此式表明：只要任何一个编码输入端有低电 平信号输入(即有编码信号)，且**S=1** (即



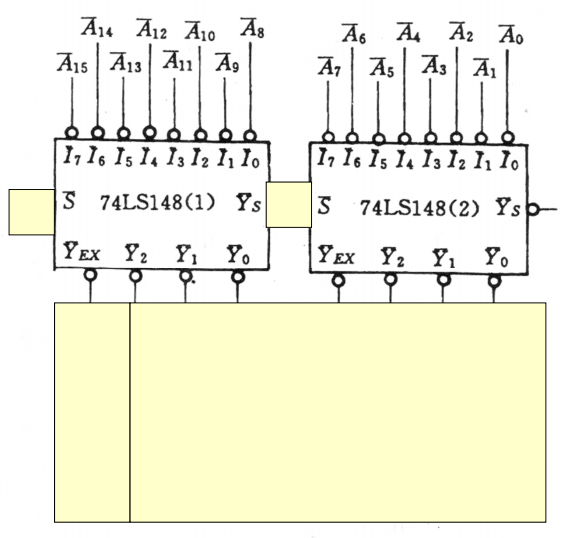
*S* = 0 )，*YEX* 即为低电平。所以， *YEX* 低电平输 出信号表示 “电路工作，且有编码输入”。

(功能表第**3——10**行所示) 。

(**4**) 在 *S* = 0 ，且有编码输入的工作状态下，允 许 *I*0 ~ *I*7 当中同时有几个输入端低电平，且其 中 *I*7 优先权最高， *I*0 优先权最低。

例、 试用两片74LS148接成16线-4线优先编码器，将 16个低电平输入信号 *A*0 ~ *A*15 编为‘0000－1111 **’** 16 个4位二进制代码，其中 *A*15 的优先权最高， *A*0 的优先 权最低。

接成的电路图如图所示：



用两片74LS148接成的16线－4线优先编码器逻辑图

思考**2**：如何用一片**74LS148**实现**8421-BCD**码优先编码器？



**4.3.2** 译码器(Decoder)

逻辑功能：将每个输入的二进制代码对应的输出为高、 低电平信号。

译码是编码的反操作。

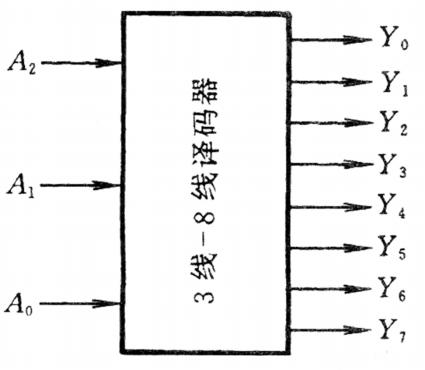
常用的译码器有二进制译码器 (**binary** **decoder**)、 二**—**十进制译码器 (**binary-coded** **decimal** **decoder**)、 显 示译码器 (**display** **decoder**)等。

一、二进制译码器(最小项译码器)

输入：一组二进制代码

输出：一组与输入代码一一对应的高、低电平信号。

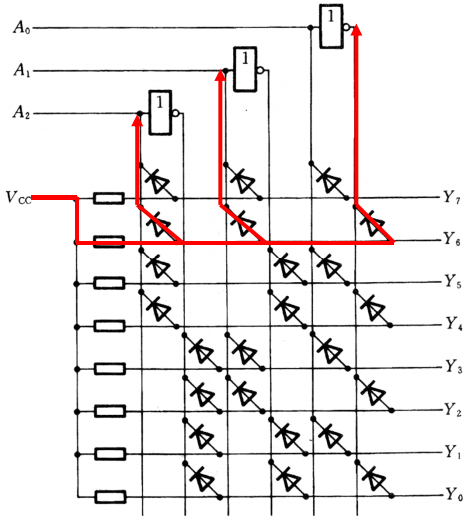
例： **3**位二进制(**3**线－**8**线)译码器框图如下所示：



图**3.3.5** **3**线－**8**线译码器框图

二进制译码器可采用二极管与门阵列或三极管集 成门电路等构成。

(**1**)二极管与门阵列译码器电路

**0(0V)**

**1(3V)**

**1(3V)**

**+5V**

二极管与门阵列构 成的3位二进制译码

器电路图

用二极管与门阵列构成的译码器：

优点： 结构简单

缺点： **1** 、电路的输入电阻较低而输出电阻较高；

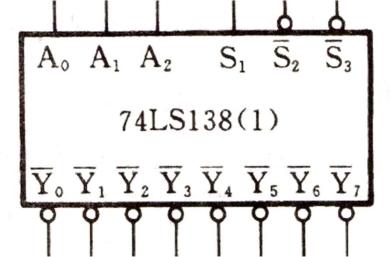
**2**、输出的高、低电平信号发生偏移(**0.7V**)。

因此，二极管门阵列译码器通常用于大规模 (**LSI**)集成电路中。

(**2**)三极管集成门译码器电路

中规模(**MSI**)集成电路通常采用三极管集成门 (如**TTL**)电路。

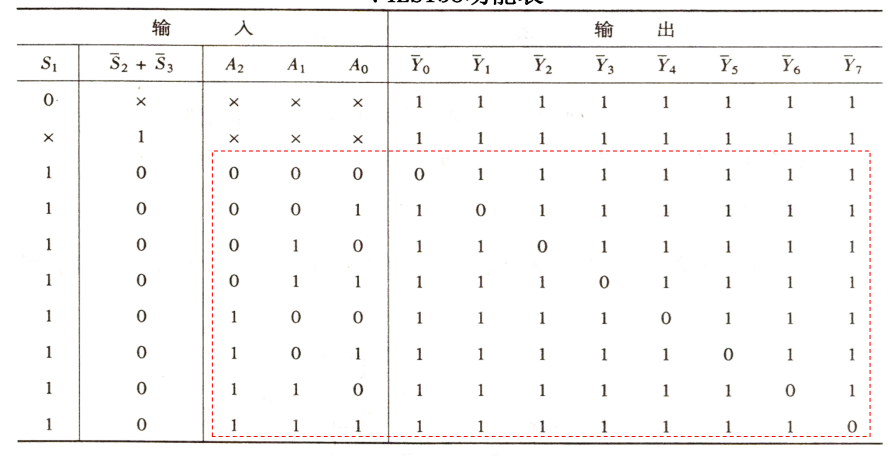
下面以**74LS138** **3**线－**8**线译码器为例来分析译码 器的工作原理：





74LS138框图及实物图



74LS138功能表

*S*

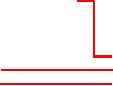
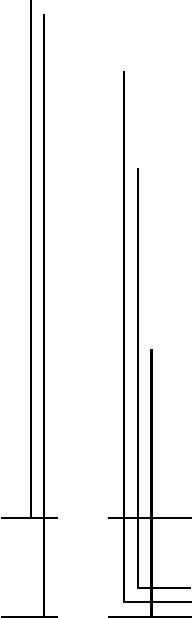
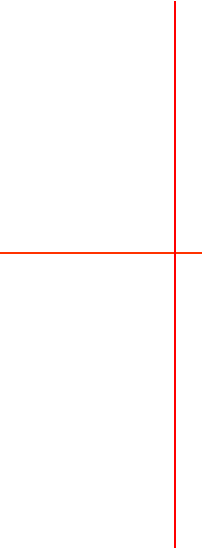
1

|  |
| --- |
|  |

|  |
| --- |
|  |

*S*2

*S*3



|  |
| --- |
|  |

***Y*0** (**15**) ***Y*1** (**14**) ***Y*2** (**13**) ***Y*3** (**12**) ***Y*4** (**11**) ***Y*5** (**10**) ***Y*6** (**9**)

***Y*7** (**7**)

**&** **&** **&** **&** **&** **&** **&**

**&**

**3-8**译码器**74xx138** 内部逻辑图

(**6**) ***G*1**

(**4**) ***G*2*A***

(**5**) ***G*2*B***

(**1**) ***A*0**

(**2**) ***A*1**

(**3**) ***A*2**

|  |
| --- |
|  |

|  |
| --- |
|  |



**1**

|  |
| --- |
|  |



**&**

|  |
| --- |
|  |

|  |
| --- |
|  |



**1**

|  |
| --- |
|  |



**1**

|  |
| --- |
|  |



**1**

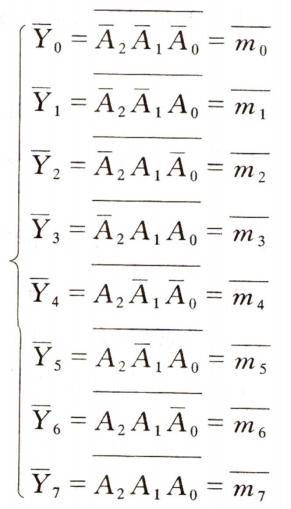
|  |
| --- |
|  |
|  |
| |  | | --- | | **1** | |

**1**

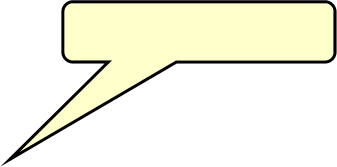
|  |
| --- |
|  |



|  |
| --- |
| **1** |



**74LS138**框图及其各输出函数表达式如下：

最小项译码器

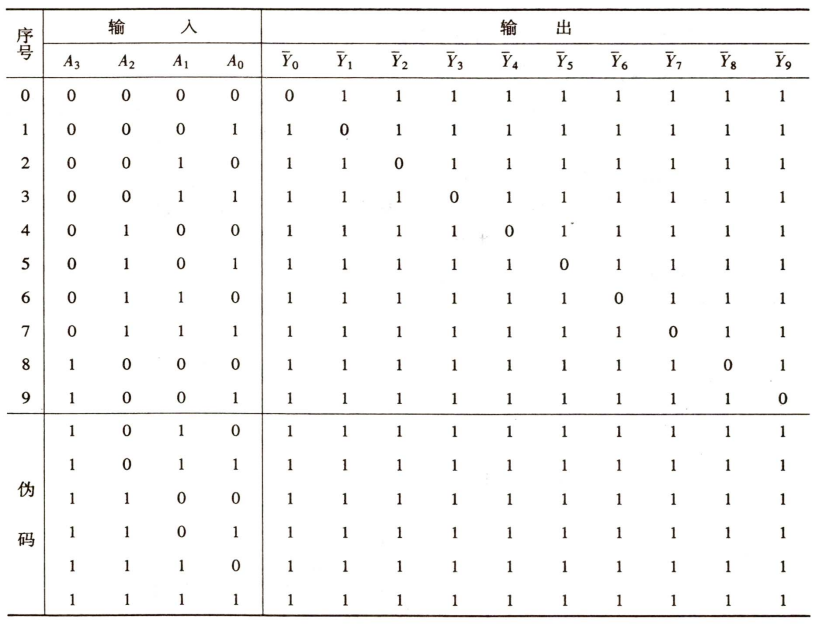
二、二－十进制译码器

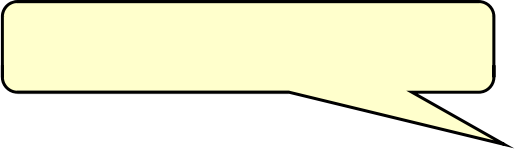
逻辑功能：将输入的**BCD**码的**10**个代码译成**10**个 高、低电平输出信号。它属于码制变换译码器中的一 种。

**4**线－**10**线译码器**74LS42**是二－十进制译码器的 一个典型例子，它将所输入的**8421**－**BCD**码二进制代 码译成十进制代码**0**~**9**。

**74LS42**的功能表(真值表)如下所示：

74LS42功能表



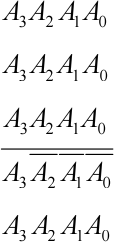


(

|

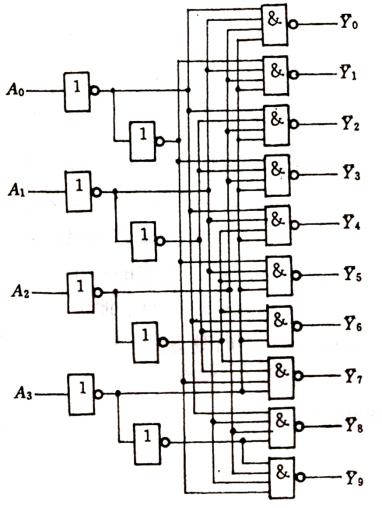
|*Y*4 = *A*3 *A*2 *A*1 *A*0

〈

思考：如何实现**5421** 、**2421**、余 **3**码等**BCD**码的译码转换？

**74LS42**逻辑电路图及各输出表达式如下所示：



74LS42逻辑电路图

三、显示译码器

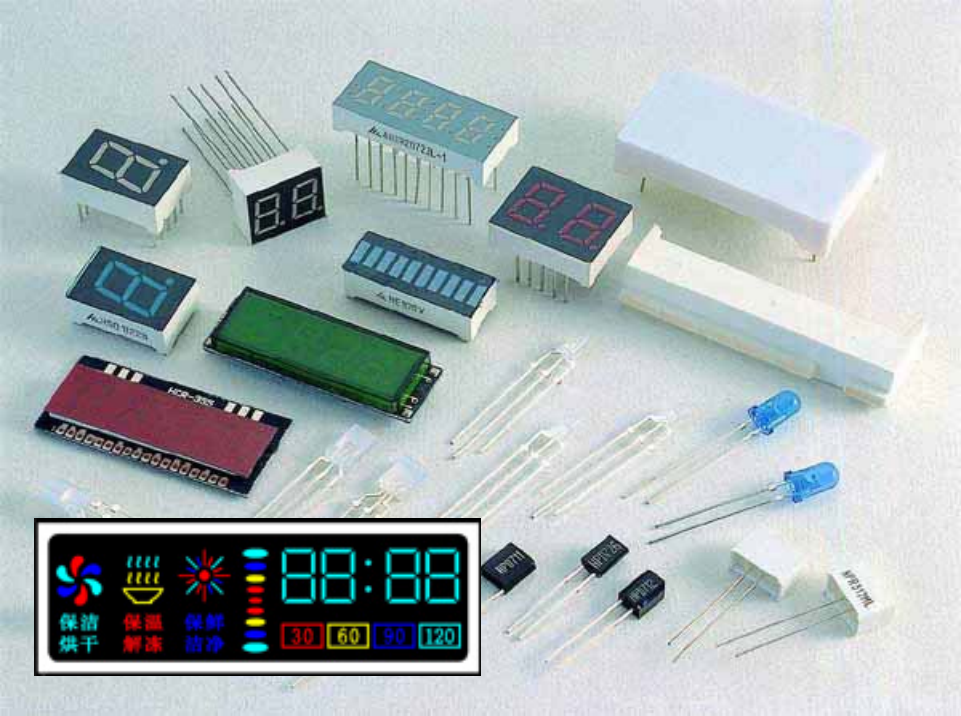
逻辑功能：将数字(**0**~**9**)、文字、符号(**A~F)**等的二 进制代码翻译并显示出来的电路叫显示译码器。它包括译码 驱动电路和数码显示器两部分。

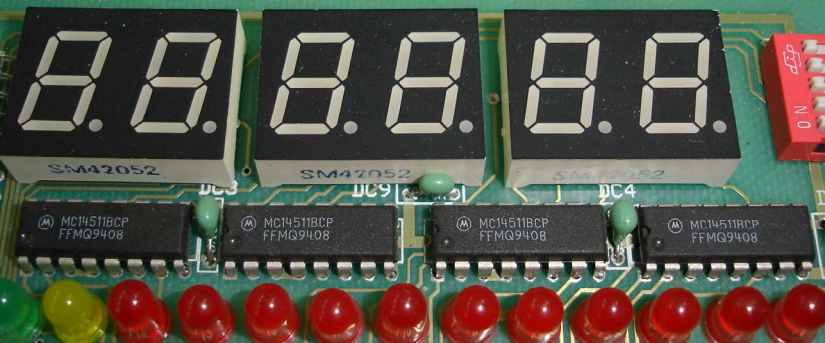
按发光物质分，数码显示器可以分为以下四种类型： **1**)半导体显示器，亦称发光二极管(**LED**)显示器；

**2**)荧光数字显示器，如荧光数码管、场效发光数字板等；

**3**)液晶数字显示器，如液晶显示器(**LCD)**、电泳显示 器等；

**4**)气体放电显示器，如辉光数码管、等离子显示板等。





数码显示实物图



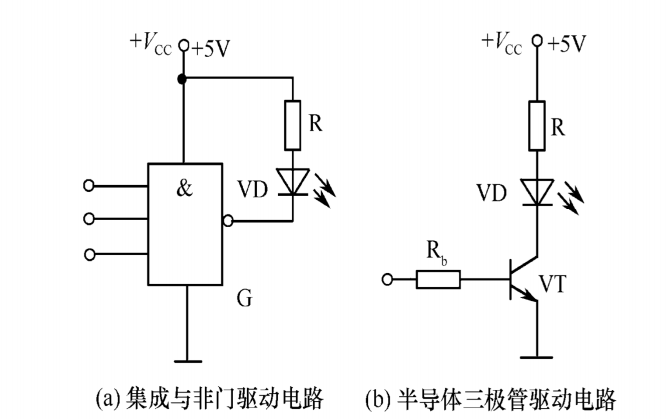
半导体数码管(八段)外形图及等效电路

优点**:** 工作电压低、体积小、寿命长、可靠性高，响应时 间短(< 0. 1*s*)，亮度较高。

缺点**:** 工作电流较大，每一段工作电流在**10mA**左右。

LED的驱动电路——

既可以用半导体三极管驱动，也可以用**TTL**与非门驱动。





|  |
| --- |
| 数码显示电路 |

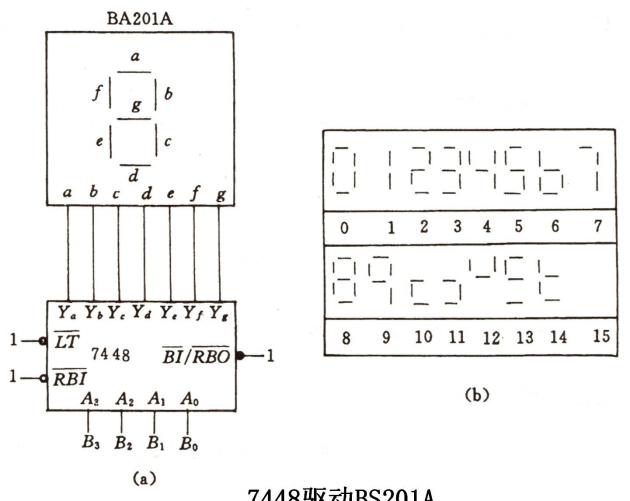
**CC**

|  |
| --- |
| 译码驱动电路 |

7448驱动BS201A

**V**

例： **BCD**七段字符译码显示电路





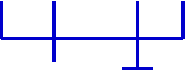
**R**并＝**1K**×**7**





**…**

(a)接线图 (b)七段显示字型



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | | |  |
|  | |  |  | |
| *RBI* *RBO* | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | | |  |
|  | |  |  | |
| *RBI* *RBO* | | | | |

|  |
| --- |
|  |
| *RBI* *RBO* |

|  |
| --- |
|  |
| *RBI* *RBO* |

|  |
| --- |
|  |
| *RBI* *RBO* |

|  |
| --- |
|  |
| *RBI* *RBO* |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | | |  |
|  | |  |  | |
| *RBI* *RBO* | | | | |



1 1

有灭零控制的数码显示系统实例图

74LS48功能表

• 74LS48除了有实现7段显示译码器基本功能的输入(DCBA)和输出(Ya~Yg )端外，还引入了灯测试输入端(LT)和动态灭零输入端(RBI)，以及既有 输入功能又有输出功能的消隐输入/动态灭零输出(BI/RBO)端

• 逻辑功能：

• 1 ) 7段译码功能 (LT=1 ，RBI=1)在灯测试输入端(LT)和动态灭零输入端 (RBI)都接无效电平时，输入DCBA经7448译码，输出高电平有效的

• 7段字符显示器的驱动信号，显示相应字符。除DCBA= 0000外， RBI也可以接 低电平，见表中1~16行。

• 2)消隐功能 (BI=0)此时BI/RBO端作为输入端，该端输入低电平信号时，表 倒数第3行，无论LT 和RBI输入什么电平信号，不管输入DCBA是什么状态，

输出全为“0”， 7段显示器熄灭。该功能主要用于多显示器的动态显示。

• 3)灯测试功能 (LT = 0)此时BI/RBO端作为输出端，端输入低电平信号时， 表最后一行，与及DCBA输入无关，输出全为“1”，显示器7个字段都点亮。 该功能用于7段显示器测试，判别是否有损坏的字段。

• 4)动态灭零功能 (LT=1 ，RBI=1)此时BI/RBO端也作为输出端， LT端输入高 电平信号， RBI 端输入低电平信号，若此时DCBA=0000，表1倒数第2行，输 出全为“0”，显示器熄灭，不显示这个零。 DCBA≠0，则对显示无影响。该 功能主要用于多个7段显示器同时显示时熄灭高位的零。

四、译码器的应用

(**1**)在存储器中的应用

用作地址译码器或指令译码器，译码器输入地 址码，输出为存储单元地址。如**n**位地址线可寻址2*n* 个单元。

(**2**)扩展应用

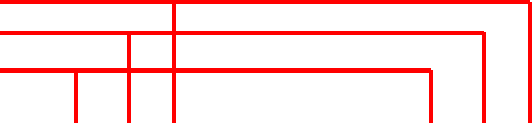
在需进行大容量译码时，可将芯片进行扩展。

例、 试用两片**74LS138**组成**4**线－**16**线译码器，

将输入的**4**位二进制代码 *D* *D* *D* *D*译成**16**个独立的

3 2 1 0

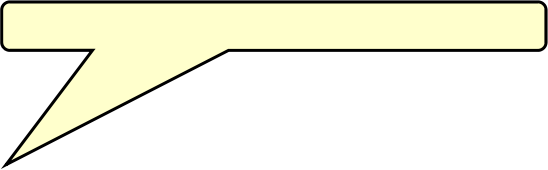
低电平信号 *Z*0 ~ *Z*15 。



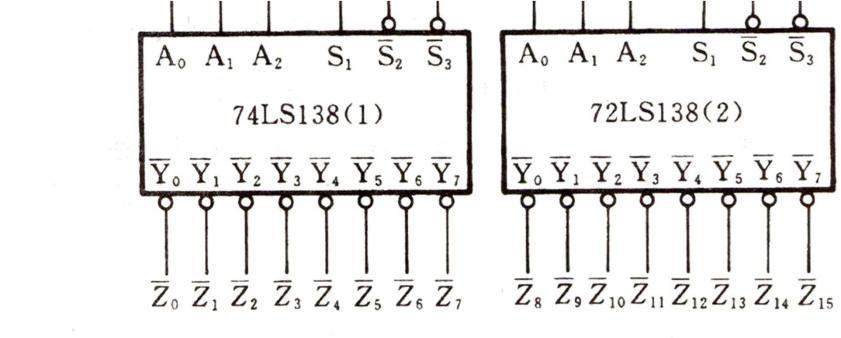
D3 D2 D1

D0

+5v



如何用**74LS138**实现**5**线－**32**线译码器？



用两片74LS138接成的4线－16线译码器

(**3**)实现逻辑函数

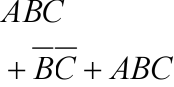
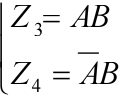
由于**n**变量二进制译码器可以提供变量的2*n* 个最小 项非的输出，而任何逻辑函数均可化为最小项之和的标 准形式，所以利用二进制译码器和一些必要的逻辑门可 以实现逻辑函数。

例、 试用**74LS138**设计一个多输出的组合逻辑电路。

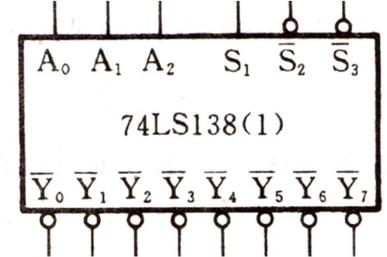
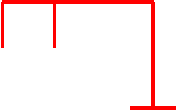
输出的逻辑函数为

(*Z* = *AC*+ *ABC*+ *ABC*

|*Z* = *BC* + *ABC*



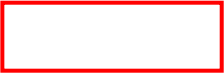
〈



例、 用**74LS138**实现函数**F(A,B,C)=AB+AC+BC**

**C** **B** **A**

**+5V**

&

 **F**

注：实现多变量译码输入的逻辑函数时，可以先 扩展再按上述方法实现。

思考： 如何用**74LS138**实现组合逻辑函数

*F*(*A*, *B*, *C*, *D*) = *ABC* + *ABD* + *BCD*

**(4)**有些二进制译码器还可作数据分配器使用。



**4.3.3** 数据分配器(Demultiplexer)

数据传输过程中，有时需要将数据分配到不同的 数据通道上，能够完成这种功能的电路称为数据分配 器，亦称多路分配器、多路调节器，简称**DEMUX** ， 其电路为单输入、多输出形式。

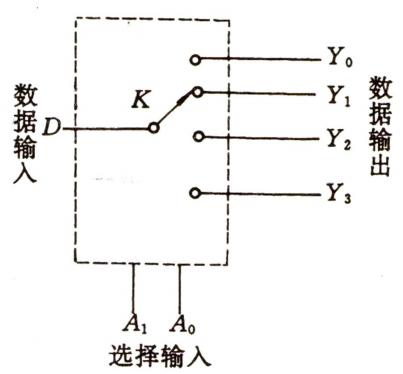
**1** 、**DEMUX**的应用

基本用途：有选择的将一个数据送到多路输出中的

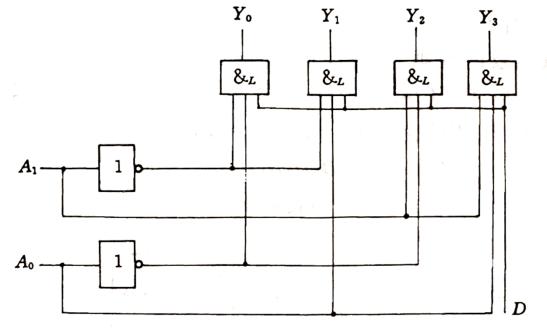
一路。

**2**、数据分配器的逻辑功能

**DEMUX**的功能如同多位开关一样，将输入**D**送 到选择输入指定的通道上(如图所示)。

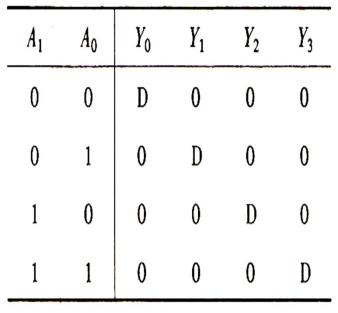


图所示为一个四路数据分配器的逻辑图， **D**为 被传输的数据， *A*0 ，*A*1 是选择输入端， *Y*0 ~ *Y*3为数据 输出端。



数据分配器示意图

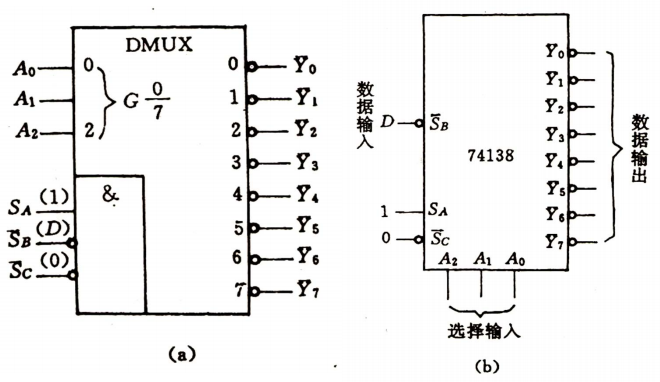
数据分配真值表



**3** 、**1**路－**8**路**DEMUX74138** (应用 (**4**) )

**74138**不仅可以作**3**线－**8**线译码器，而且还可用

作**1**路－**8**路数据分配器(如图所示)。



74138用作1路－8路数据分配器的逻辑符号 (a)国际逻辑符号 (b)惯用逻辑符号



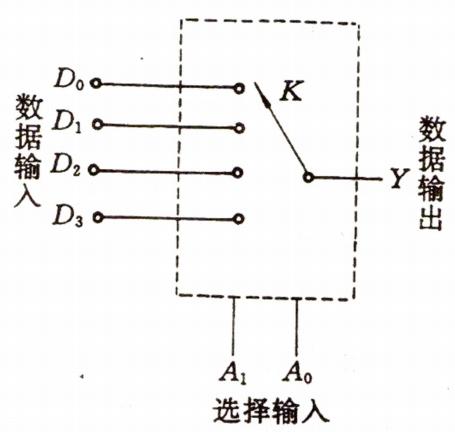
**4.3.4** 数据选择器( **Multiplexer** )

能够实现从多路数据中选择一路进行传输的电 路叫做数据选择器，亦称多路调制器、多路开关或 多路选择器(简称**MUX**)。

1、数据选择器的逻辑功能

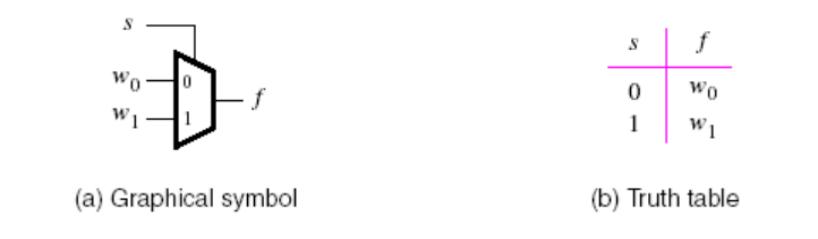
**MUX**的功能正好与**DEMUX**相反，为多输入、 单输出形式。

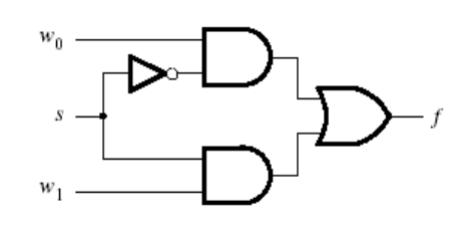
目前，常用的**MUX**有二选一、四选一、八选 一和十六选一等多种类型。

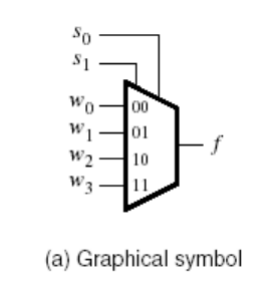
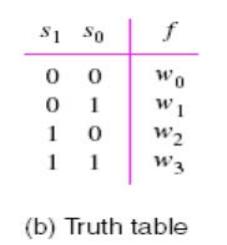


四选一数据选择示意图

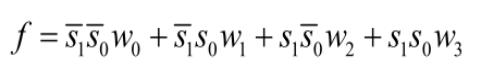
**2** 选 **1** 多路选择器

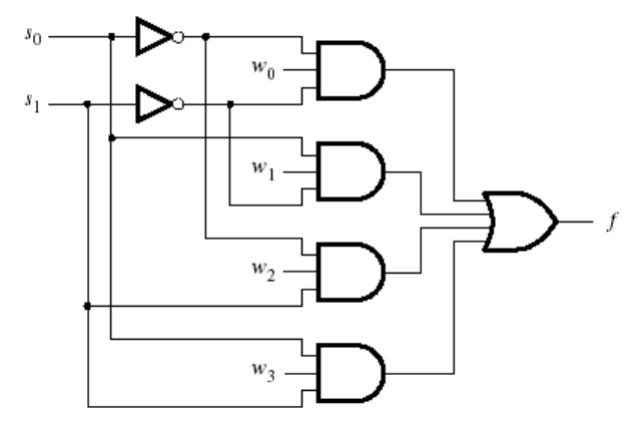


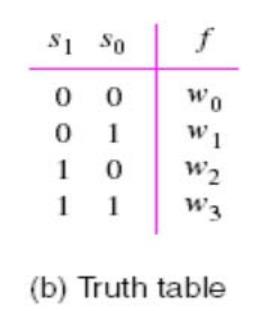




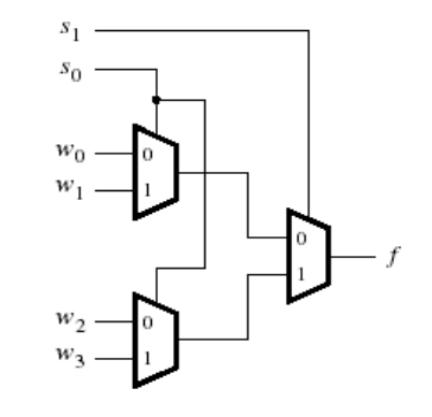
**4** 选 **1** 多路选择器





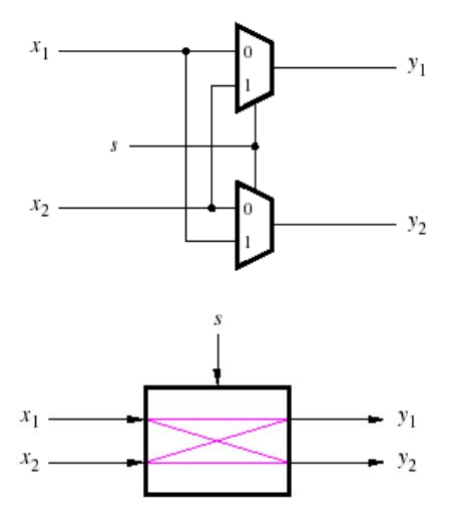


更大规模的多路选择器可以由较简单的多路选择器构成

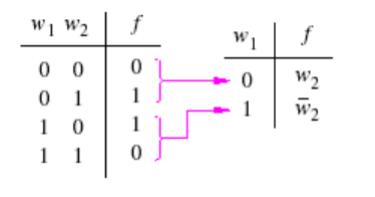


由 2 选 1 多路选择器构建的 4 选 1 多路选择器

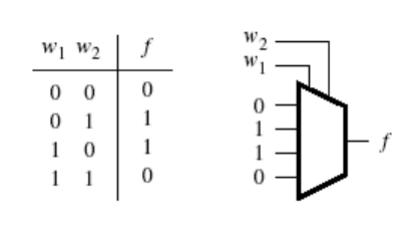
多路选择器的实际应用

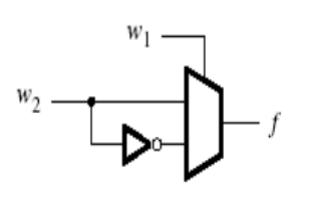


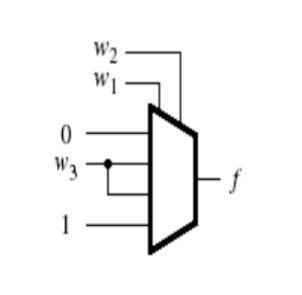
2×2 纵横开关



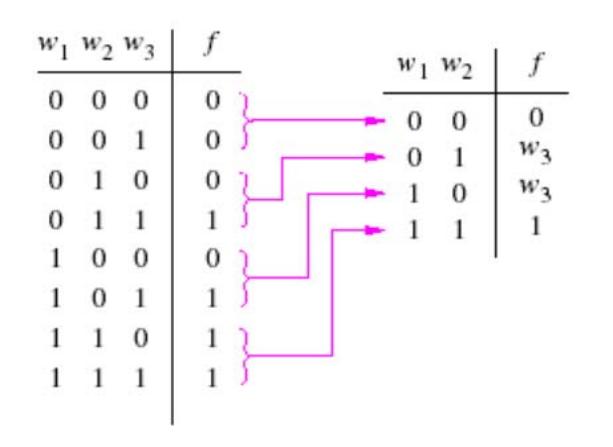
利用多路器的逻辑综合



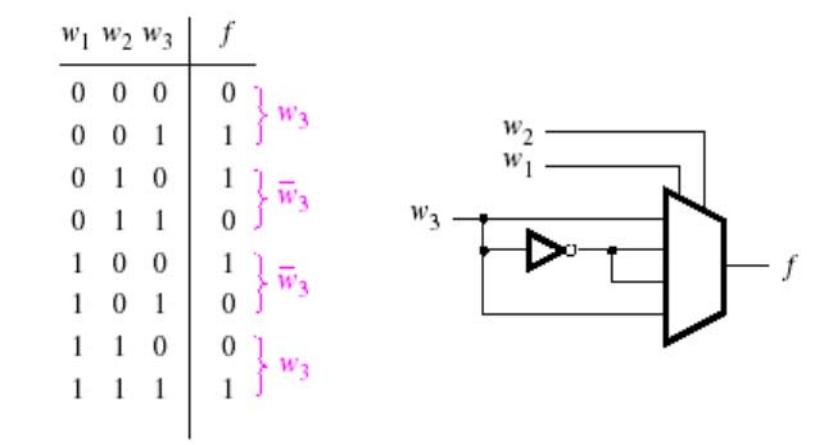




利用4选1多路器来实现3输入表决器



用4选1多路器来实现3位输入的异或逻辑

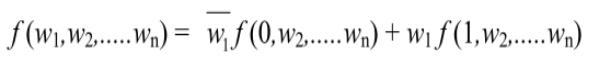


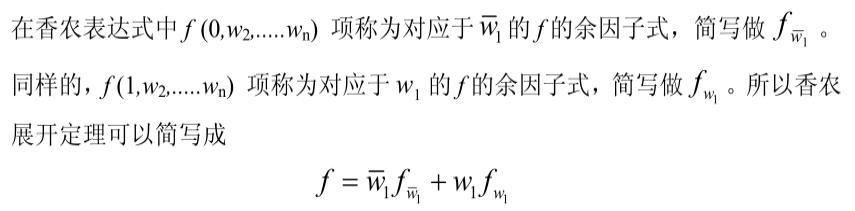
用香农(Shannon)展开的多路器综合

如何改写真值表以便用多路器来实现逻辑函数

香农展开定理

任何布尔函数 都可以表示成如下形式：

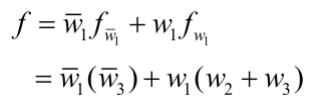


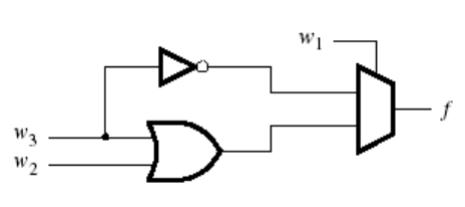


假设我们想要用2选1的多路器和其他必须的门电路来 实现以下函数：

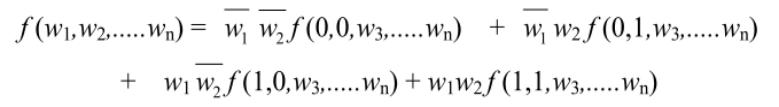


若以w 1 为基础来进行香农展开，则函数可以表示为：



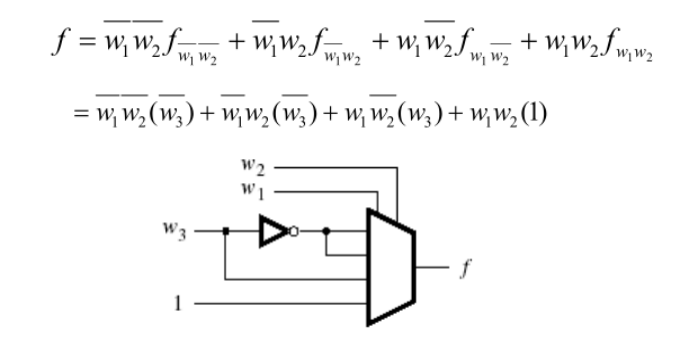


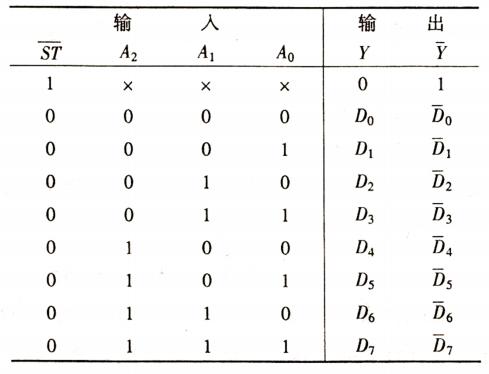
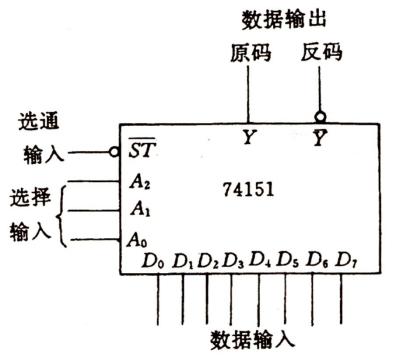
香农展开也能以多个变量进行分解，以w1 w2 例，所得结果如 下：



上述展开式可用 4 选 1 多路器实现







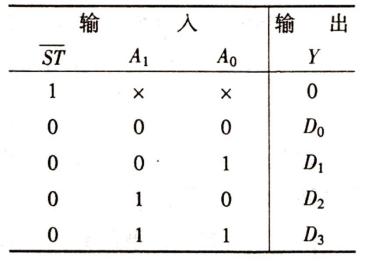
2、八选一数据选择器74151

八选一**MUX**需要**3**个选择输入端， **8**个数据输入端， 并有互补的原码和反码两种输出形式。

**74151**的惯用逻辑符号和真值表如下所示：

74151真值表

74151惯用逻辑符号

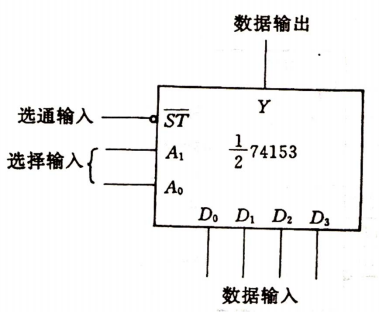


3、双四选一数据选择器74153

**74153**包含两个完全相同的**4**选一**MUX**，两个**MUX** 有公共的地址输入端，而数据输入和输出端各自独立。 通过给定不同的地址代码(*A*1*A*0 **),**即可从**4**个输入数据 中选出所需要的一个，并送至输出端**Y**。

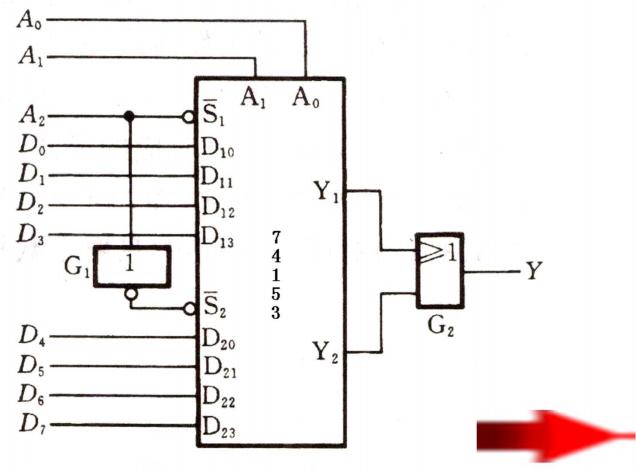
**74153**的惯用逻辑符号和真值表如下：

**74153**真值表

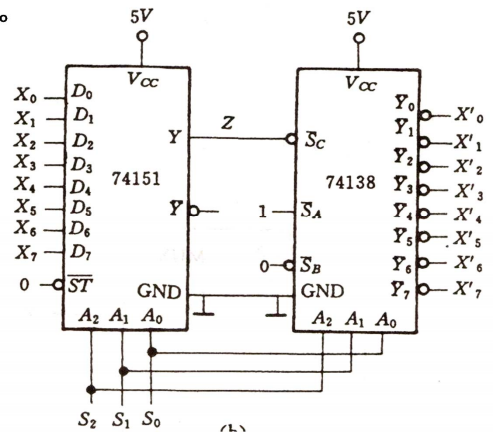


**74153**惯用逻辑符号

例： 试用双四选一**MUX74LS153**组成一个**8**选一**MUX**。



用双四选一MUX74153组成8选一MUX示意图



**SA**

4、数据选择器的应用

(**1**)与数据分配器(**DEMUX**)一起实现多路信号分时 传送。

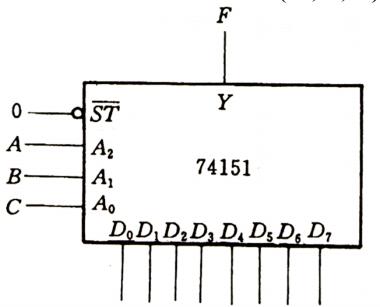
例

多路信号的分时传送

(**2**)实现组合逻辑函数

原理：若逻辑函数变量的数目与数据选择器选择输入端 的数目相等，则函数最小项的数目就与数据选择器输入端 的数目相同，这样可以直接用数据选择器实现逻辑函数。

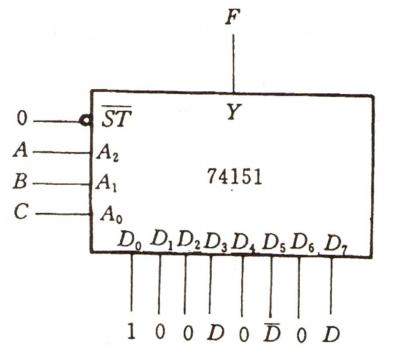
例、 试用**74151**实现逻辑函数**F(A,B,C)=AB+AC+BC**

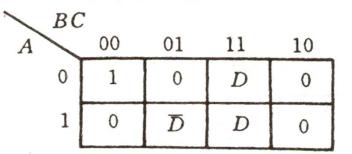


**0** **0** **0** **1** **0** **1** **1** **1**

用74151实现逻辑函数

例、 试用一片**74151**实现逻辑函数

*F*(*A*, *B*, *C*, *D*) = *ABCD* + *ABCD*+ *ABCD* + *ABC*



(a) 引入变量的卡诺图

(b) 逻辑图 用74151实现逻辑函数

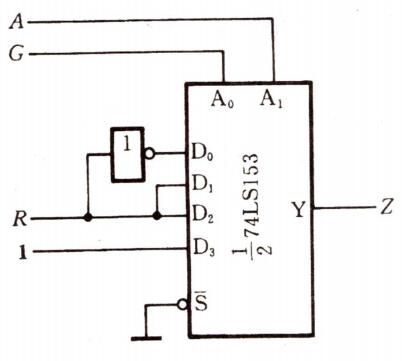


思考： 如何用一片**74151**实现逻辑函数

*F*(*A*, *B*, *C*, *D*, *E*) = *ABCDE* + *ABCDE* +

*ABCDE* + *ABCDE* + *ABCDE*

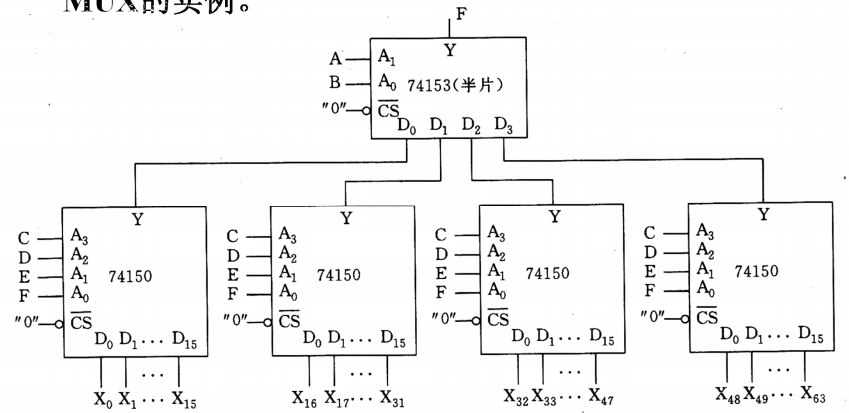
例、 试用半片双四选一**74153**实现交通信号灯故障

监视电路。 

74151实现交通信号灯监视电路图

(**3**)数据选择器的扩展

例、 下图为一个将十六选一**MUX**扩展为六十四选一

**MUX**的实例。

使用**MUX**和**DEMUX**实现组合逻辑函数**F**的共同点是： **1**)不用对**F**进行化简，可节省时间； **2**)检查和排除故障容易； **3**)可以使集成芯片的数目减到最少。



**4.3.5** 数值比较器( Digital Comparator )

在一些数字系统当中经常要求比较两个数字的大 小。为完成这一功能所设计的各种逻辑电路简称为数 值比较器。

一、一位数值比较器

两个**1**位二进制数**A,B**相比的情况有以下几种：



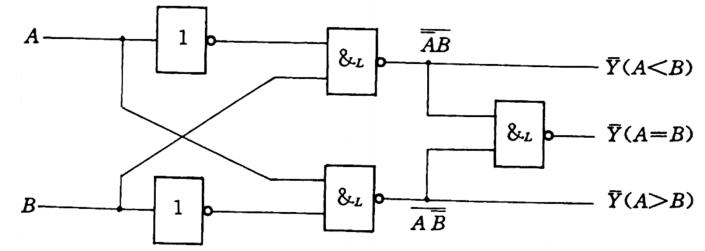
①**A>B** (即**A=1,B=0**)，则 *AB* = 1 ，所以可用 *AB*

作为**A>B**的输出信号 *Y*( *A*>*B* ) 。

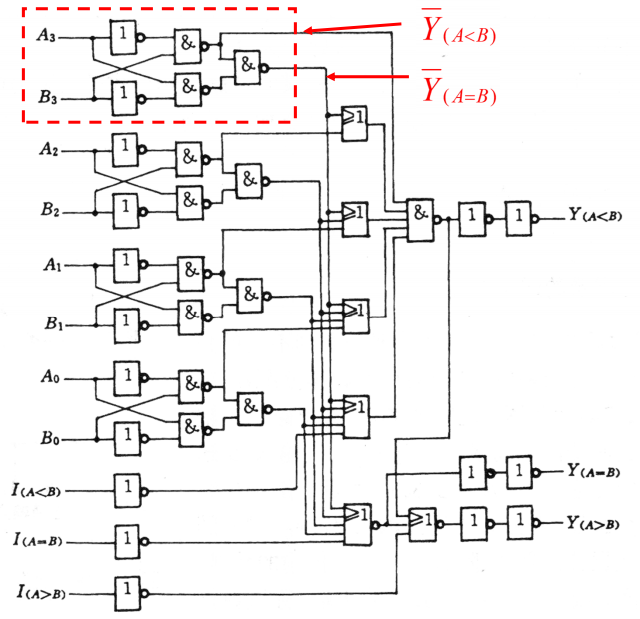
②同理可用 *AB* 作为**A<B**的输出信号 *Y*( *A*<*B*) 。

③同理可用**A**⊙**B** 作为**A=B**的输出信号 *Y*(*A*=*B*) 。

于是， **1**位数值比较器的电路图可如下设计：



1位数值比较器逻辑图



二、多位数值

比较器

在比较两

个多位数的大

小时，必须自

高而低的逐位

比较，而且只

有在高位相等

时，才需比较

较低位。

下图示出

了**4**位比较器

**CC14585**的逻

辑图。

在比较两个**4**位以上的二进制数时，应将两片以上 的**CC14585**级联，组合成位数更多的比较电路。

例： 试用两片**CC14585**组成一个**8**位比较器。



将两片CC14585接成8位数值比较器



**4.3.6** 加法器(Adder )

两个二进制数之间的算术运算无论是加、减、乘、除， 目前在数字计算机中都是化为若干步加法运算加移位进行的。 因此，加法器是构成算术运算器的基本单元。

一、1位加法器

**1**、半加器(**Half** **Adder**)

若不考虑有来自低位的进位将两个**1**位二进制数相加， 称为半加。实现半加运算的电路叫做半加器。

半加器的真值表、逻辑表达式、电路图和惯用符号如下

所示：

〈

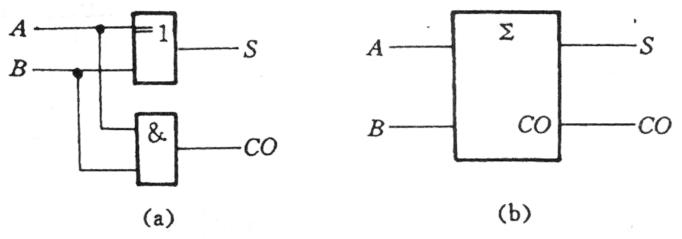
(*S* = *AB* + *AB* = *A*由 *B*

 *CO* = *AB*

半加器的真值表

|  |  |  |  |
| --- | --- | --- | --- |
| 输入 | | 输出 | |
| **A** | **B** | **S** | **CO** |
| **0** | **0** | **0** | **0** |
| **0** | **1** | **1** | **0** |
| **1** | **0** | **1** | **0** |
| **1** | **1** | **0** | **1** |

半加器的逻辑表达式



半加器的电路图和惯用逻辑符号

**2**、全加器(**Full** **Adder**)

将两个多位二进制数相加时，除了最低位以外，每一位都应考虑来 自低位的进位，即将两个对应的加数和来自低位的进位**3**个数相加。这 种运算称为全加，所用电路称为全加器。

**1**位全加器的真值表、逻辑表达式、 电路图和惯用符号如下所示：

全加器的真值表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 输入 | | | 输出 | |
| **CI** | **A** | **B** | **S** | **CO** |
| **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** |

全加器的逻辑表达式

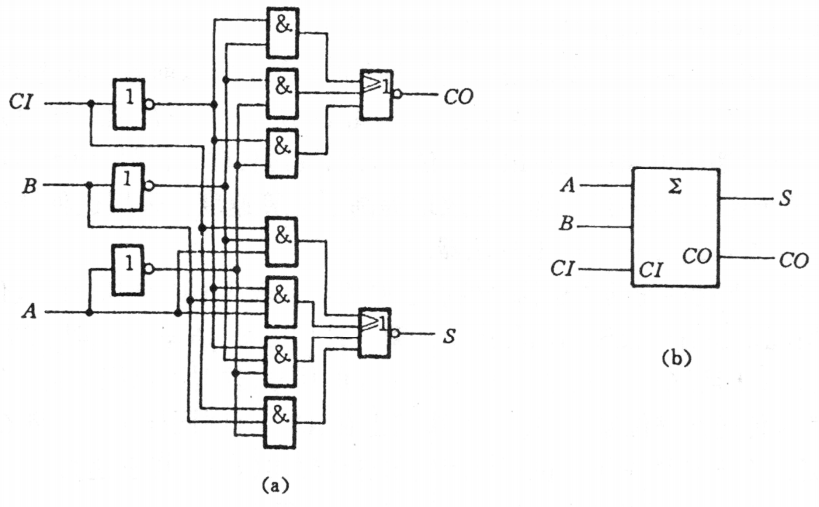
(|*S* = *AB* . *CI* + *AB* . *CI* + *AB* . *CI* + *AB* . *CI*

〈

|*CO* = *AB* + *B* .*CI* + *A* . *CI*

(*S* = *A*  *B*  *CI*

或〈 *CO* = *AB* + *CI*(*A* + *B*)



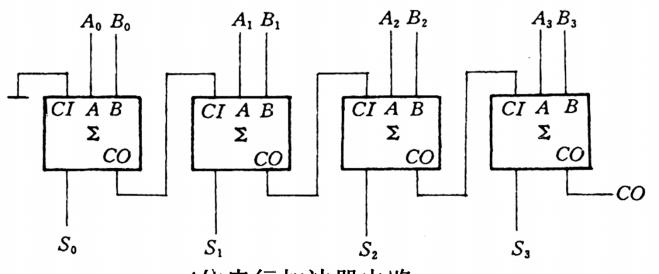
全加器的电路图和惯用逻辑符号

二、多位加法器

**1**、串行进位加法器

原理：依次将低位全加器的进位输出端**CO**接到高 位全加器的进位输入端**CI**即可构成多位串行加法器。

例： **4**位串行进位加法器电路如下：

4位串行加法器电路

应用举例：多人表决电路。

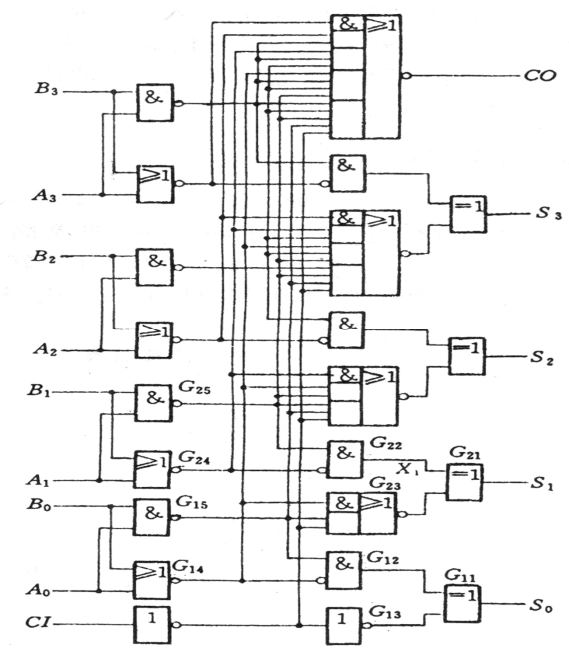
串行进位加法器的优点：电路结构比较简单；

缺点：运算速度慢。

**2**、超前进位加法器

为了提高运算速度，须减小或消除由于进位信号逐级 传递所耗费的时间。由于第**i**位的进位输入信号(*CI*)*i* 一定能 由 *Ai*1*Ai*2 … *A*0 和 *Bi*1*Bi*2 …*B*0 唯一确定，所以可先得出每 一位全加器的进位输入信号，而无需再从最低位开始向高 位逐级传递进位信号了，这就有效的提高了运算速度。

采用这种结构形式的加法器为超前进位(**Carry-** **Lookahead**)加法器。



下图示出了**4**位

超前进位加法

器**74LS283**的电

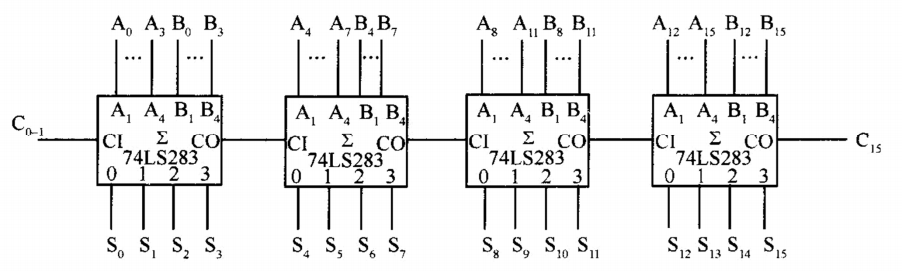
路图：

74LS283的逻辑图

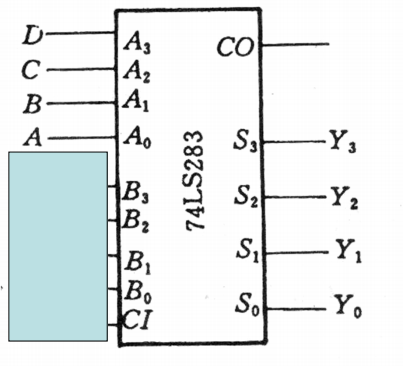
由图可知，两个加数送到输入端到完成加法运算只需三 级门电路的传输时间，这样， 运算速度大大提高了，但电路 复杂程度也随之急剧上升。

为扩充相加数的位数，可将多片低位加法器级联。

例： **4**片**74283**级联成**16**位二进制加法电路的电路为：



74283级联成16位二进制加法电路



三、用加法器设计组合逻辑电路

对“变量＋变量”或“变量＋常量”类型的逻辑函 数用加法器设计起来非常简单。

例： 试设计一个代码转换电路，将**BCD**代码的**8421** 码转换成余**3**码。

用加法器设计的代码转换电路



组合逻辑电路中的竞争－冒险现象



4.4

**1** 产生的竞争冒险的原因

不考虑门的延时时间

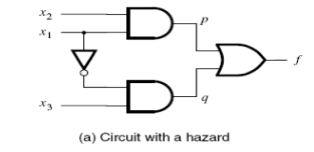
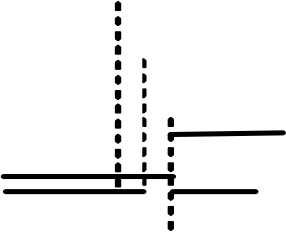
***L*** = ***AA*** = **0**

***L*** = ***A*** + ***B*** = **1**

考虑门的延时时间,当***A*=0** ***B*=1**



|  |
| --- |
|  |



***C***

|  |
| --- |
|  |

*C*

***AC***

*BC*

***L***

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。

成因：当两个输入信号同时向相反的逻辑电平跳变

时(一个从**1**变为**0**，一个从**0**变为**1**) ，由于存在时刻上

的差异，使两个信号在 *t* 的极短时间内同时为高电平

或低电平，从而产生尖峰脉冲，不符合门电路稳态下的 逻辑功能，产生内部噪声。

竞争：门电路两个输入信号同时向相反的逻辑电平 跳变(一个从**1**变为**0**，一个从**0**变为**1**)的现象叫竞争。

有竞争不一定产生尖峰脉冲。

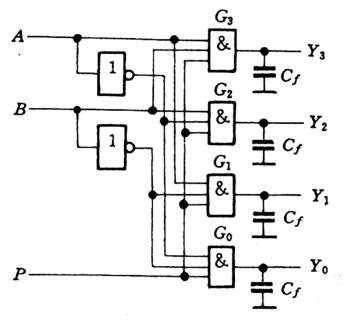
由于竞争而在电路输出端可能产生尖峰脉冲的现象 叫做竞争－冒险。

二、检查竞争－冒险现象的方法

**1**、可通过逻辑函数式判断组合逻辑电路中是否有竞 争－冒险存在。 只要输出端的逻辑函数在一定条件下能化 简成 *Y* = *A* + *A* 或 *Y* = *A* . *A*的形式，则可判定存在竞争 －冒险(此方法适用于任何瞬间只可能有一个输入变量改 变状态的情况)。

**2** 、用计算机辅助分析，运行数字电路的模拟程序。

**3** 、用实验检查。



三、消除竞争－冒险现象的方法

(一)接入滤波电容

尖峰脉冲一般都很窄(几十**ns** 以内)，只要在输出 端并接一个很小的滤波电容*Cf* (**TTL**电路中通常为几 十~几百皮法)，就足以将尖峰脉冲的幅度削弱至门电 路的阈值电压以下。

例：

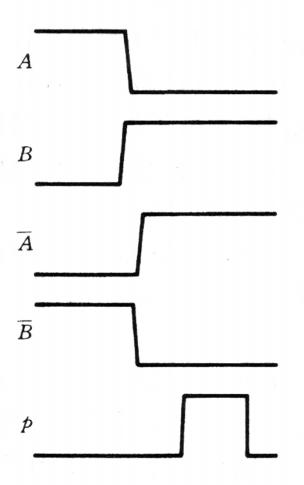
接入滤波电容消除竞

争－冒险现象的示意

图

优点： 简单易行。

缺点：增加了输出电压波形的上升和下降时间，使波形变坏。



(二)引入选通脉冲

例：

引入选通脉冲消除竞争－

冒险现象的示意图

优点：简单，不需增加电路元件。

缺点： 正常的输出信号也将变成脉冲信号，宽度与选通

脉冲相同，且此选通脉冲必须与输入信号同步。

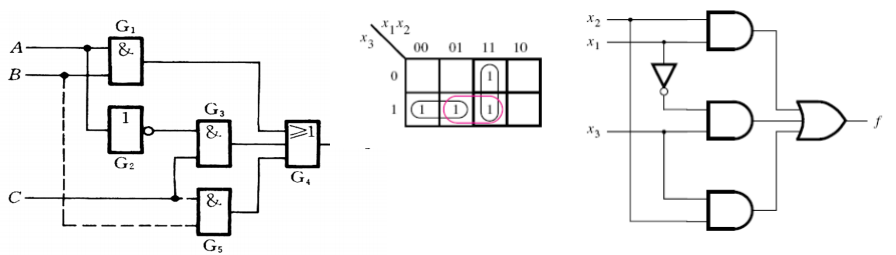


(三)修改逻辑设计

有时可用增加冗余项的方法消除竞争－冒险现象。

例：将 *Y* = *AB* + *AC* 化成 *Y* = *AB* + *AC*+ *BC* ，可

使电路功能不变，而消去**B=C=1**时的竞争－冒险现象。



修改逻辑设计消除竞争－冒险现象的示意图

优点：运用得当可收到令人满意的结果。

缺点：有利条件并不是任何时候都存在，其适用范围是有限的。