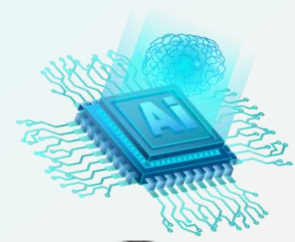


第5章 组合逻辑电路



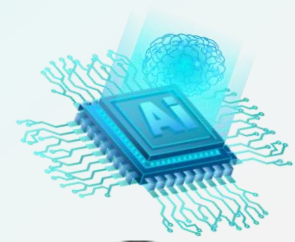
本章导读：

由门电路组成的逻辑电路称为**组合逻辑电路**。在组合逻辑电路中，任意时刻的输出仅仅取决于该时刻的输入，与电路原来的状态无关，这就是组合逻辑电路在逻辑功能上的共同特点。本章主要讲述组合逻辑电路的分析和设计方法，以及常用的组合逻辑器件。



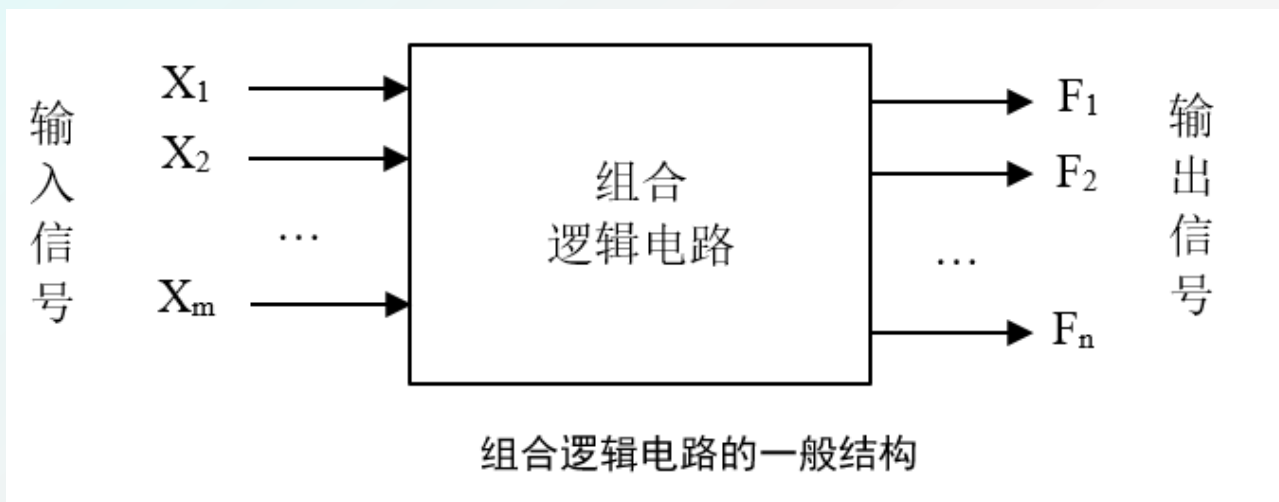
5.1 组合逻辑电路分析和设计方法

组合逻辑电路分析和设计是一对互逆的过程，前者对给定的组合电路图进行分析，确定其逻辑功能；后者根据给定的功能要求，设计最简的组合电路图，构建实现功能的逻辑电路。

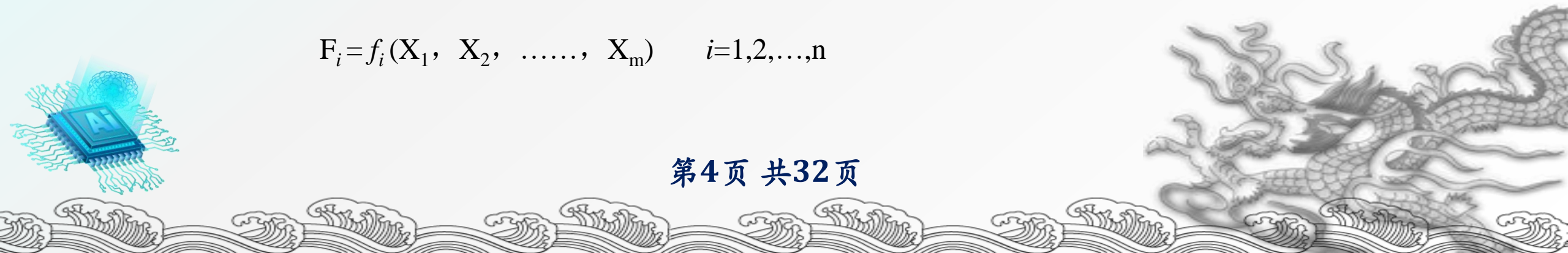


5.1.1 组合逻辑电路模型

组合逻辑电路是指在任何时刻产生的稳定输出值仅仅取决于该时刻各输入值的组合，与过去的输入值无关



$$F_i = f_i(X_1, X_2, \dots, X_m) \quad i=1,2,\dots,n$$



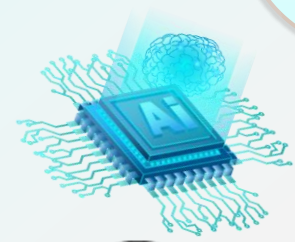
5.1.2 组合逻辑电路的分析方法

组合逻辑电路的分析的含义与步骤

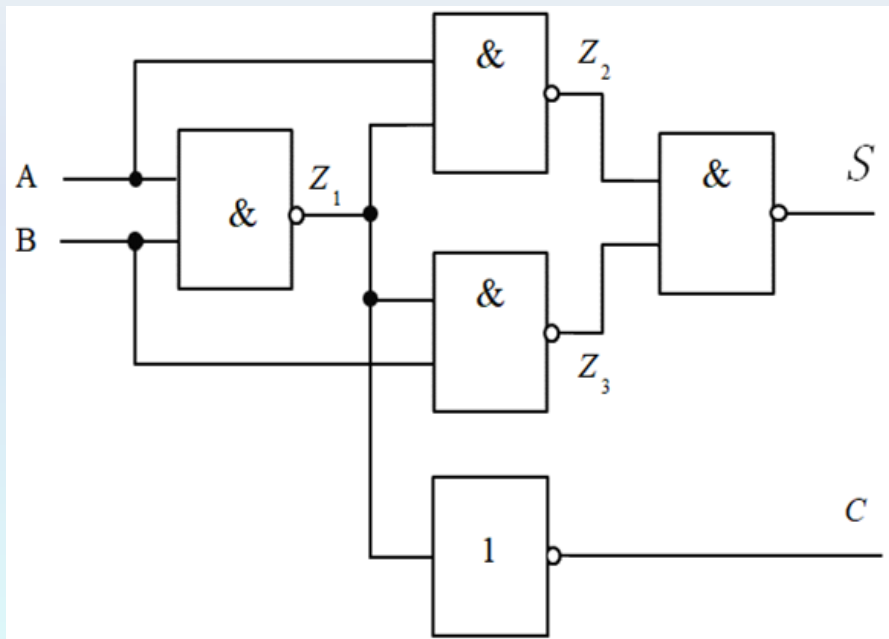
组合逻辑电路的分析，就是根据给定的逻辑电路图，确定其逻辑功能。分析组合逻辑电路的目的是确定已知电路的逻辑功能或者检查电路设计是否合理。

步骤如下：

- (1) 根据给定逻辑电路图，写出逻辑函数表达式；
- (2) 化简逻辑函数表达式；
- (3) 根据最简逻辑表达式列真值表；
- (4) 观察真值表中输出与输入的关系，描述电路逻辑功能。



【例5-1】 试分析下图所示组合逻辑电路的功能。



解：

(1) 写出中间连接端及输出端的逻辑函数表达式。

$$Z_1 = \overline{AB}$$

$$Z_2 = \overline{A \cdot \overline{AB}}$$

$$C = \overline{Z_1} = AB$$

$$Z_3 = \overline{B \cdot \overline{AB}}$$

$$S = \overline{Z_2 Z_3}$$

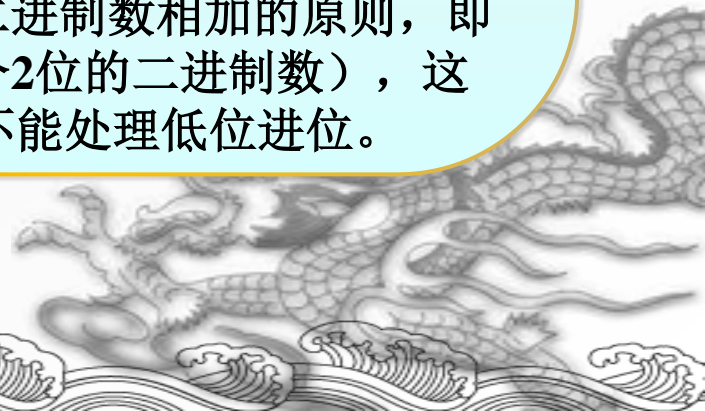
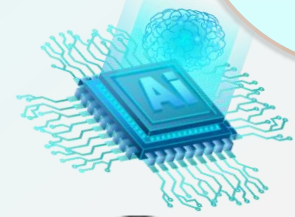
(2) 化简逻辑函数表达式。

课堂练习：头上两横直接去除；反演率；异或基本式

(3) 列出真值表

课堂练习：S是“异或”；C是“与”

(4) 描述电路逻辑功能。由表5-1可知，A、B都为0时，C、S都为0；A、B有一个为1时，C为0，S为1；A、B都为1时，C为1，S为0。这个电路的逻辑功能符合两个1位二进制数相加的原则，即A、B为加数，S是它们的和，C是向高位的进位，即 $A+B=CS$ （CS是一个2位的二进制数），这个加法器电路称之为半加器（Half Adder）。半加器能产生进位，但是不能处理低位进位。

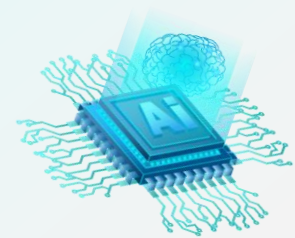


5.1.3 组合逻辑电路的设计方法

与分析过程相反，**组合逻辑电路的设计**是根据给定的实际逻辑问题，求出实现其逻辑功能的最简逻辑电路。

组合逻辑电路的设计步骤如下：

- (1) 分析设计要求，设置输入变量和输出变量并逻辑赋值；
- (2) 列真值表，根据上述分析和赋值情况，将输入变量的所有取值组合和与之相对应的输出函数值列表，即得真值表；
- (3) 写出逻辑表达式并化简；
- (4) 画逻辑电路图。



【例5-2】 用门电路设计一位全加器电路。一位全加器 (Full Adder) 是两个一位的二进制数相加，得到相加的和及进位的组合逻辑电路，全加器与半加器的差异在于，全加器可以处理低位的进位。

解：

(1) 分析设计要求，设输入、输出变量。

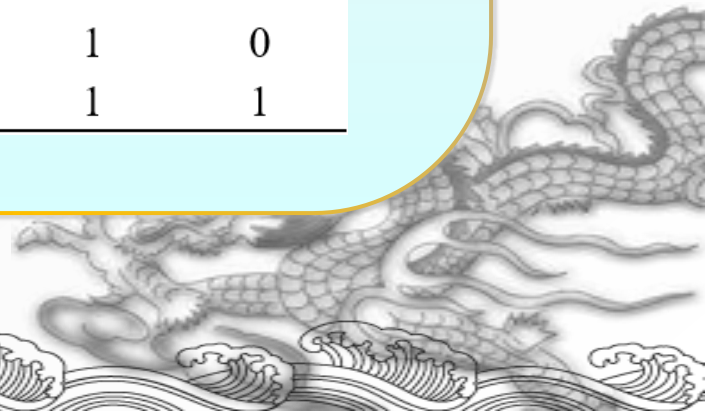
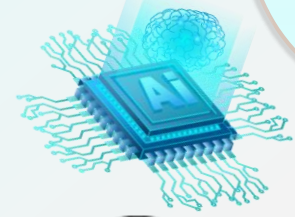
输入变量： A_i 、 B_i 为两个加数，低位来的进位数为 C_{i-1} ；

输出变量：当前位相加的和为 S_i ，向高位进位数为 C_i ；

逻辑功能： $A_i + B_i + C_{i-1} = C_i S_i$ ($C_i S_i$ 是一个二位的二进制数)，即：3个1位二进制数相加得到一个二位的二进制数。

(2) 列真值表。

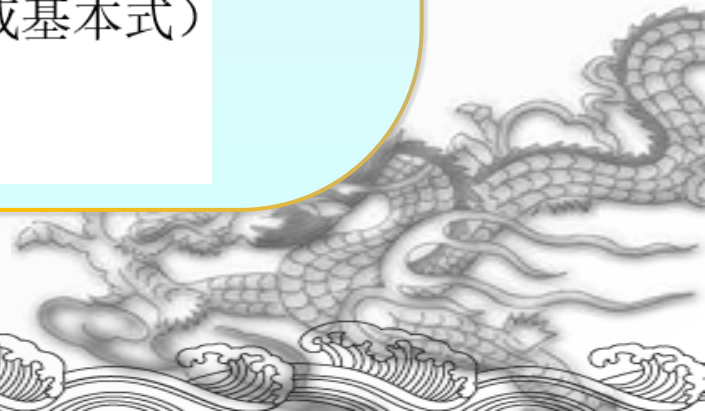
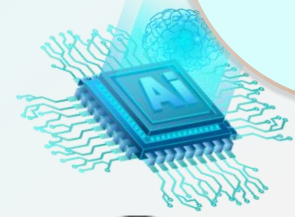
A_i	B_i	C_{i-1}	C_i	S_i	A	B	C_{i-1}	C_i	S_i
0	0	0	0	0	1	0	0	0	1
0	0	1	0	1	1	0	1	1	0
0	1	0	0	1	1	1	0	1	0
0	1	1	1	0	1	1	1	1	1



(3) 由真值表写出逻辑函数表达式并化简。

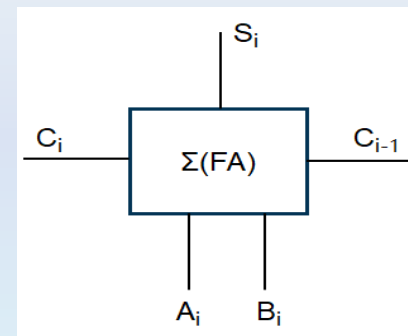
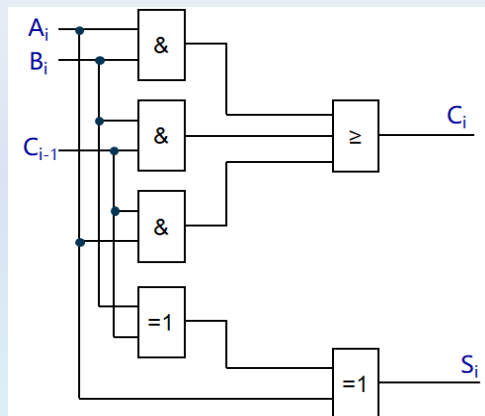
$$\begin{aligned}C_i &= \bar{A}_i B_i C_{i-1} + A_i \bar{B}_i C_{i-1} + A_i B_i \bar{C}_{i-1} + A_i B_i C_{i-1} \quad (\text{由真值表中 } C_i \text{ 为 1 的项得出}) \\&= \bar{A}_i B_i C_{i-1} + A_i \bar{B}_i C_{i-1} + A_i B_i \bar{C}_{i-1} + A_i B_i C_{i-1} + A_i B_i C_{i-1} + A_i B_i C_{i-1} \quad (\text{根据重叠律}) \\&= (A_i B_i \bar{C}_{i-1} + A_i B_i C_{i-1}) + (\bar{A}_i B_i C_{i-1} + A_i B_i C_{i-1}) + (A_i \bar{B}_i C_{i-1} + A_i B_i C_{i-1}) \quad (\text{根据交换律}) \\&= A_i B_i (\bar{C}_{i-1} + C_{i-1}) + B_i C_{i-1} (\bar{A}_i + A_i) + A_i C_{i-1} (\bar{B}_i + B_i) \quad (\text{根据分配律}) \\&= A_i B_i + B_i C_{i-1} + A_i C_{i-1} \quad (\text{根据互补律})\end{aligned}$$

$$\begin{aligned}S_i &= \bar{A}_i \bar{B}_i C_{i-1} + \bar{A}_i B_i \bar{C}_{i-1} + A_i \bar{B}_i \bar{C}_{i-1} + A_i B_i C_{i-1} \quad (\text{由真值表中 } S_i \text{ 为 1 的项得出}) \\&= \bar{A}_i B_i \bar{C}_{i-1} + A_i \bar{B}_i \bar{C}_{i-1} + \bar{A}_i \bar{B}_i C_{i-1} + A_i B_i C_{i-1} \quad (\text{根据交换律}) \\&= (\bar{A}_i B_i + A_i \bar{B}_i) \bar{C}_{i-1} + (\bar{A}_i \bar{B}_i + A_i B_i) C_{i-1} \quad (\text{根据分配律}) \\&= X \bar{C}_{i-1} + \bar{X} C_{i-1} \quad (\text{令 } X = \bar{A}_i B_i + A_i \bar{B}_i = A_i \oplus B_i, \text{ 根据异或基本式}) \\&= X \oplus C_{i-1} \\&= A_i \oplus B_i \oplus C_{i-1}\end{aligned}$$

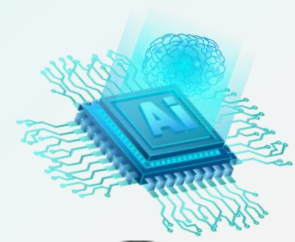
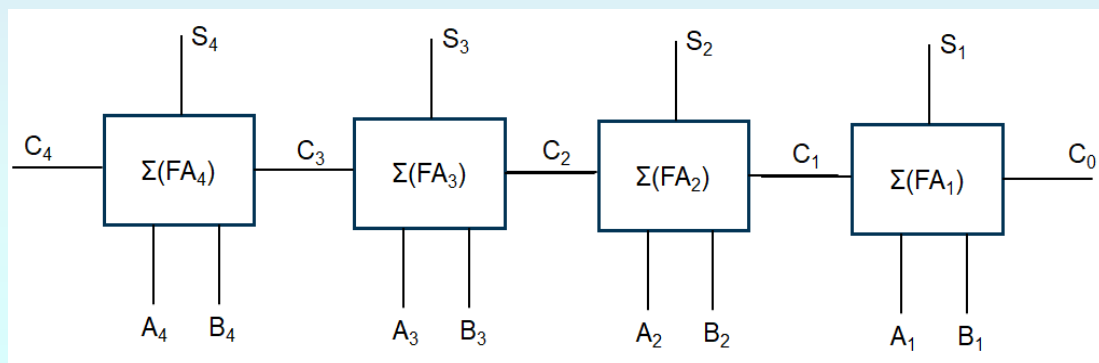


(4) 画逻辑电路图

1位全加器的逻辑电路图及逻辑符号



4位串行进位加法器逻辑电路图

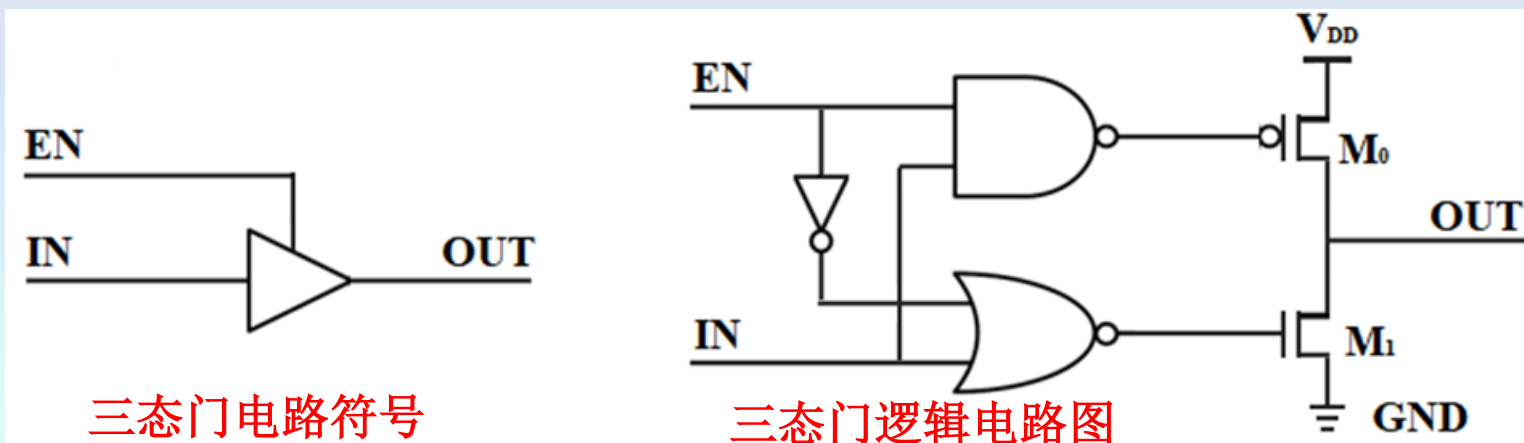


5.2 三态门、编码器与译码器

三态门、编码器与译码器是计算机硬件组成中的重要组合逻辑电路。

5.2.1 三态门

三态门指可以输出高电平、低电平，有可保存高阻态的逻辑电路。高电平一般使用逻辑1表达，低电平一般使用逻辑0表达，高阻态相当于隔断状态（电阻很大，相当于开路）。

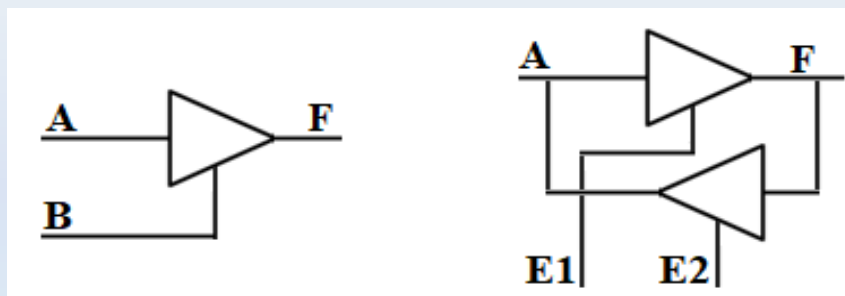


三态门电路符号

三态门逻辑电路图

原理分析：当 $EN=1$ 且 $IN=1$ ，MOS管 M_0 导通、 M_1 断开， OUT 和 V_{DD} 相连，输出1；其他类比

单向三态门与双向三态门：

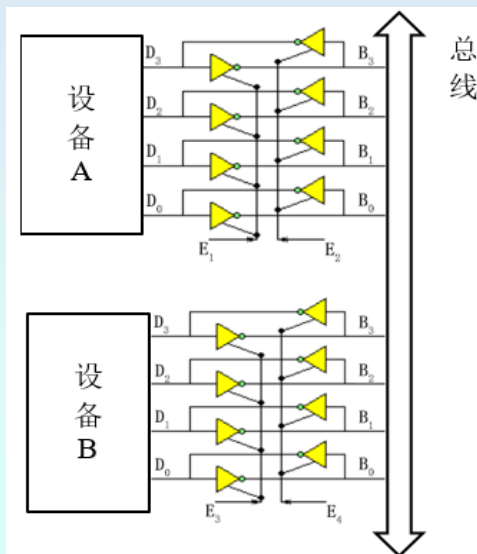


$E1=1 \ E2=0 \ A \rightarrow F$

$E1=0 \ E2=1 \ F \rightarrow A$

三态门应用举例：

实现总线的数据传输



$E1=1$ 、 $E2=0$ 、 $E3=0$ 、 $E4=1$ 时，
数据设备A→总线→设备B，
实现设备A通过总线将数据传输给设备B

$E1=0$ 、 $E2=1$ 、 $E3=1$ 、 $E4=0$ 时，
数据设备B→总线→设备A，
实现设备B通过总线将数据传输给设备A

5.2.2 编码器

1. 编码器的定义

广义地说，编码器是指可将特定物理信号转换成文字、符号或者数码表示的逻辑电路器件。

狭义地说，编码器是指将特定的物理信号转换成二进制数字信息的逻辑电路器件。

举个例子来说，设有4个按键，表面文字标识分别“开始、前进、后退、停止”，希望按下某一个按键，对应输出二进制信息，设分别按下“开始、前进、后退、停止”键，对分别应输出二进制00~11，给出一个电路完成这样的功能，这就是编码器。

2. 编码器的分类

按照编码方式不同，编码器可分为普通编码器和优先编码器。普通编码器是指任何时刻只允许一个输入信号有效的编码器；优先编码器是指允许多个输入信号同时有效，输出是对优先级别高的输入信号进行编码的编码器。

普通编码器又可分二进制编码器和非二进制编码器。若输入信号的个数 N 与输出变量的位数 n 满足 $N=2^n$ ，此电路称为二进制编码器；若输入信号的个数 N 与输出变量的位数 n 不满足 $N=2^n$ ，此电路称为非二进制编码器。普通编码器任何时刻只能对其中一个输入信息进行编码，即输入的 N 个信号是互相排斥的。例如，编码器的输入为4个信号，输出为两位代码，则称为4-2编码器。

3. 编码器举例

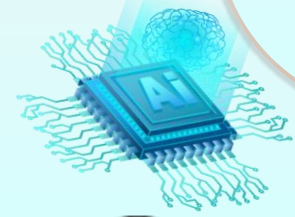
1) 键控8421BCD码编码器

BCD的含义：Binary-Coded Decimal，二进制码的十进数，用4位二进制数表示1位十进制数中的0~9这10个数码。

8421BCD码是最基本和最常用的BCD码，它和四位自然二进制码相似，各位的权值为8、4、2、1，称为有权BCD码。它只选用了四位二进制码中前10组代码，即用0000~1001分别代表它所对应的十进制数，余下的六组代码不用。

例：10个按键 $S_0 \sim S_9$ 代表输入的10个十进制数0~9，输入为低电平有效，即某一按键按下，对应的输入信号为0，输出对应的8421码，输出为4位码，所以有4个输出端A、B、C、D。

输 入										输 出				
S9	S8	S7	S6	S5	S4	S3	S2	S1	S0	A	B	C	D	GS
1	1	1	1	1	1	1	1	1	1	0	0	0	0	0
1	1	1	1	1	1	1	1	1	0	0	0	0	0	1
1	1	1	1	1	1	1	1	0	1	0	0	0	1	1
1	1	1	1	1	1	1	0	1	1	0	0	1	0	1
1	1	1	1	1	1	0	1	1	1	0	0	1	1	1
1	1	1	1	1	0	1	1	1	1	0	1	0	0	1
1	1	1	1	0	1	1	1	1	1	0	1	0	1	1
1	1	1	0	1	1	1	1	1	1	0	1	1	0	1
1	1	0	1	1	1	1	1	1	1	0	1	1	1	1
1	0	1	1	1	1	1	1	1	1	1	0	0	0	1
0	1	1	1	1	1	1	1	1	1	1	0	0	1	1



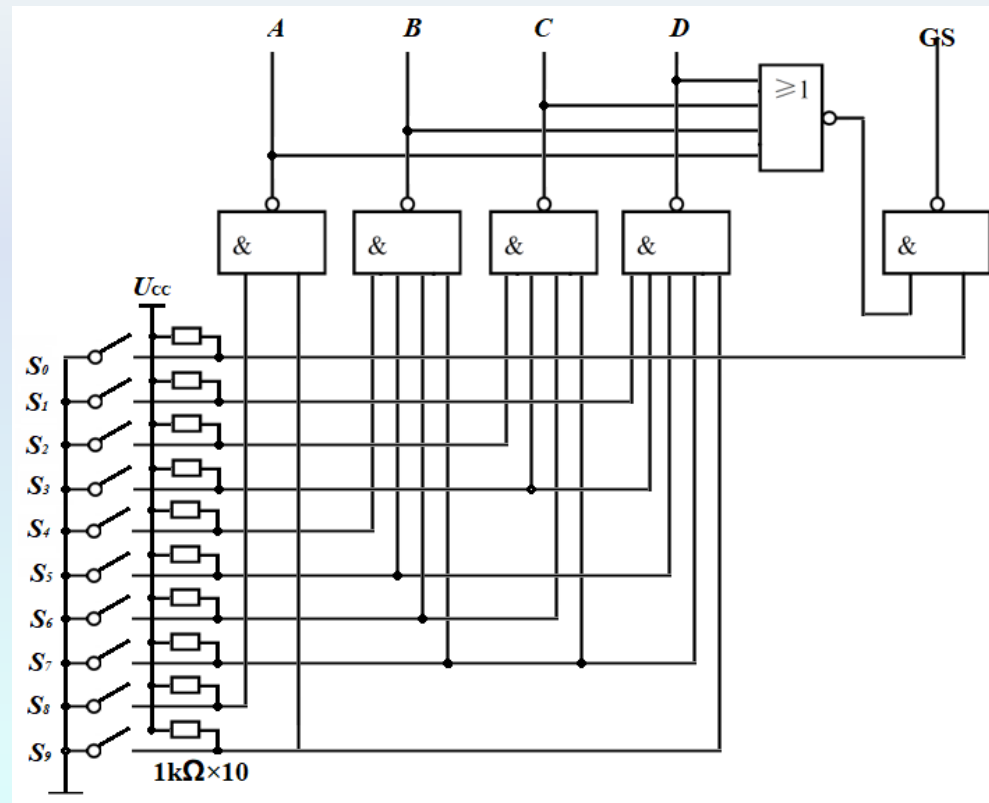
由真值表写出各输出的逻辑表达式为

$$A = \overline{S_8} + \overline{S_9} = \overline{S_8 S_9}$$

$$B = \overline{S_4} + \overline{S_5} + \overline{S_6} + \overline{S_7} = \overline{S_4 S_5 S_6 S_7}$$

$$C = \overline{S_2} + \overline{S_3} + \overline{S_6} + \overline{S_7} = \overline{S_2 S_3 S_6 S_7}$$

$$D = \overline{S_1} + \overline{S_3} + \overline{S_5} + \overline{S_7} + \overline{S_9} = \overline{S_1 S_3 S_5 S_7 S_9}$$



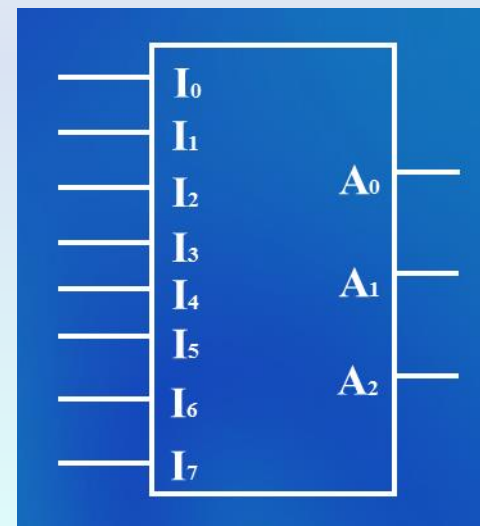
逻辑图

2) 二进制编码器

用 n 位二进制代码对 2^n 个信号进行编码的电路称为二进制编码器。3位二进制编码器有8个输入端3个输出端，所以常称为8线—3线编码器，其功能真值表见表5-4，输入为高电平有效。

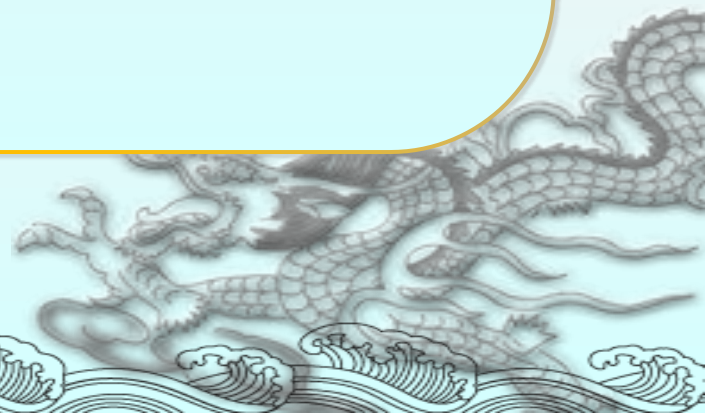
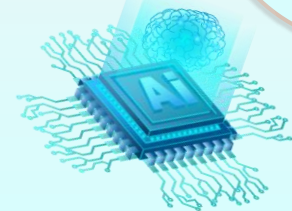
表5-4 编码器真值表

输 入								输 出		
I_0	I_1	I_2	I_3	I_4	I_5	I_6	I_7	A_2	A_1	A_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

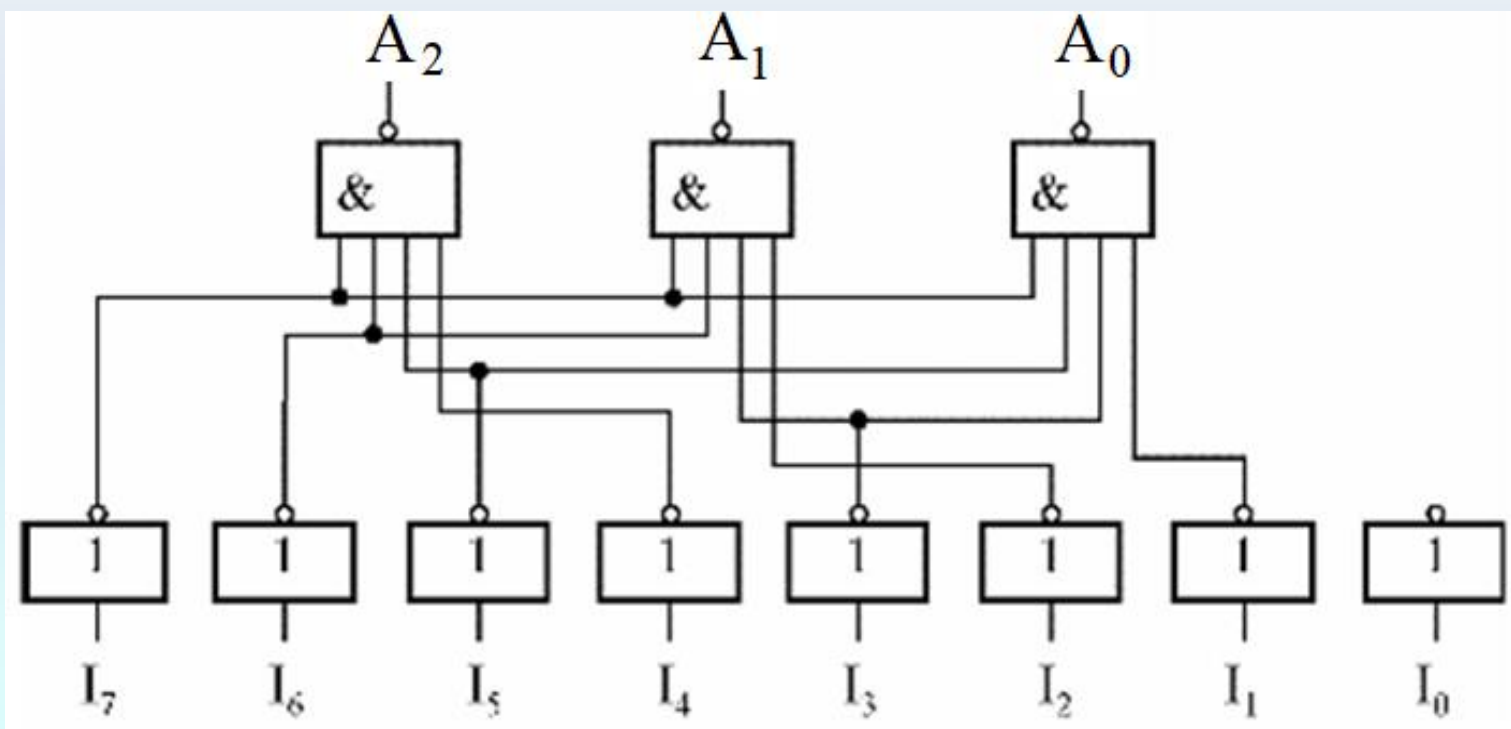


由真值表写出各输出的逻辑表达式为

$$A_2 = \overline{\overline{I_4 I_5 I_6 I_7}} \quad A_1 = \overline{\overline{I_2 I_3 I_6 I_7}} \quad A_0 = \overline{\overline{I_1 I_3 I_5 I_7}}$$



用门电路实现逻辑电路



3) 集成编码器 74LS148

74LS148是一种常用的8线—3线优先编码器。优先编码器：允许同时输入两个以上信号，并按优先级输出。优先编码器常用于优先中断系统和键盘编码。与普通编码器不同，优先编码器允许多个输入信号同时有效，但它只按其中优先级别最高的有效输入信号编码，对级别较低的输入信号不予理睬。



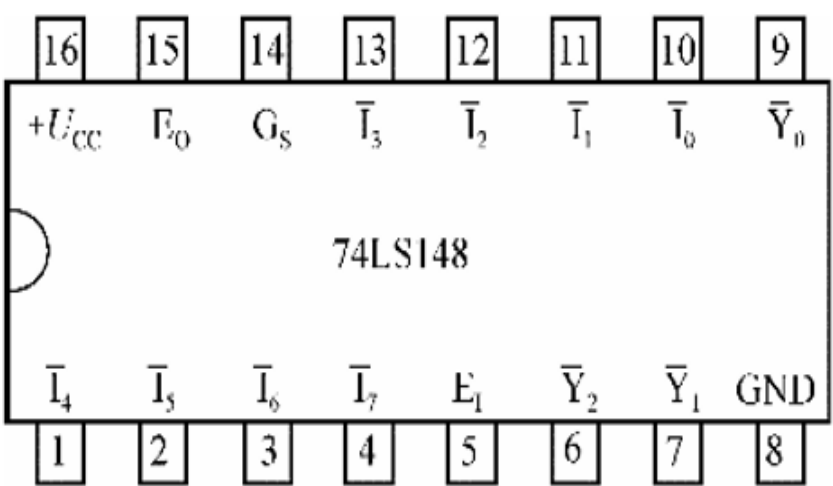
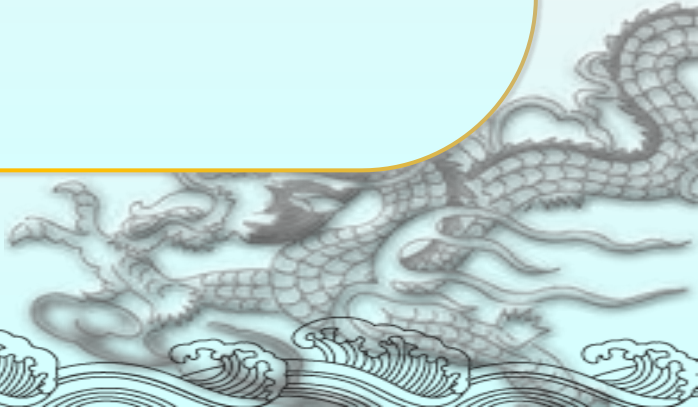
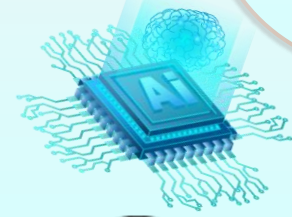


表5-5 编码器真值表

输 入									输 出				
E _I	\bar{I}_0	\bar{I}_1	\bar{I}_2	\bar{I}_3	\bar{I}_4	\bar{I}_5	\bar{I}_6	\bar{I}_7	\bar{A}_2	\bar{A}_1	\bar{A}_0	G _s	E _O
1	×	×	×	×	×	×	×	×	1	1	1	1	1
0	1	1	1	1	1	1	1	1	1	1	1	1	0
0	×	×	×	×	×	×	×	0	0	0	0	0	1
0	×	×	×	×	×	×	0	1	0	0	1	0	1
0	×	×	×	×	×	0	1	1	0	1	0	0	1
0	×	×	×	×	0	1	1	1	0	1	1	0	1
0	×	×	×	0	1	1	1	1	1	0	0	0	1
0	×	×	0	1	1	1	1	1	1	1	0	0	1
0	×	0	1	1	1	1	1	1	1	1	1	0	1
0	1	1	1	1	1	1	1	1	1	1	1	0	1

其逻辑功能见表5-5，其中 $\bar{I}_0 \sim \bar{I}_7$ 为编码输入端，低电平有效。 $\bar{A}_0 \sim \bar{A}_2$ 为编码输出端，也为低电平有效，即反码输出。其他功能如下。

- (1) E_I 为使能输入端，低电平有效；
- (2) 优先顺序从低到高依次为 $\bar{I}_0 \rightarrow \bar{I}_7$ ，即 \bar{I}_7 的优先级最高，然后是 \bar{I}_6 、 \bar{I}_5 、... \bar{I}_0 ；
- (3) G_s 为编码器的工作标志，低电平有效
- (4) E_O 为使能输出端，高电平有效



5.2.3 译码器

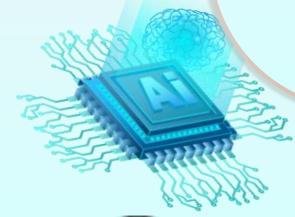
译码是编码的逆过程，即将每一组输入二进制代码“翻译”成为一个特定的输出信号。实现译码功能的数字电路称为译码器。集成译码器分为二进制译码器、二—十进制译码器和显示译码器3种。

1. 二进制译码器

集成二进制译码器由于其输入、输出端的数目满足 $2^N=M$ ，属完全译码器，常见的二进制译码器有2-4线译码器（74LS139）、3-8线译码器（**74LS138**）和4-16线译码器（74LS154）等。表5-6所示为3-8线译码器74LS138的逻辑功能表。

表5-6 74LS138逻辑功能表

输 入			输 出							
A ₂	A ₁	A ₀	\bar{Y}_7	\bar{Y}_6	\bar{Y}_5	\bar{Y}_4	\bar{Y}_3	\bar{Y}_2	\bar{Y}_1	\bar{Y}_0
0	0	0	1	1	1	1	1	1	1	0
0	0	1	1	1	1	1	1	1	0	1
0	1	0	1	1	1	1	1	0	1	1
0	1	1	1	1	1	1	0	1	1	1
1	0	0	1	1	1	0	1	1	1	1
1	0	1	1	1	0	1	1	1	1	1
1	1	0	1	0	1	1	1	1	1	1
1	1	1	0	1	1	1	1	1	1	1



2. 非二进制译码器

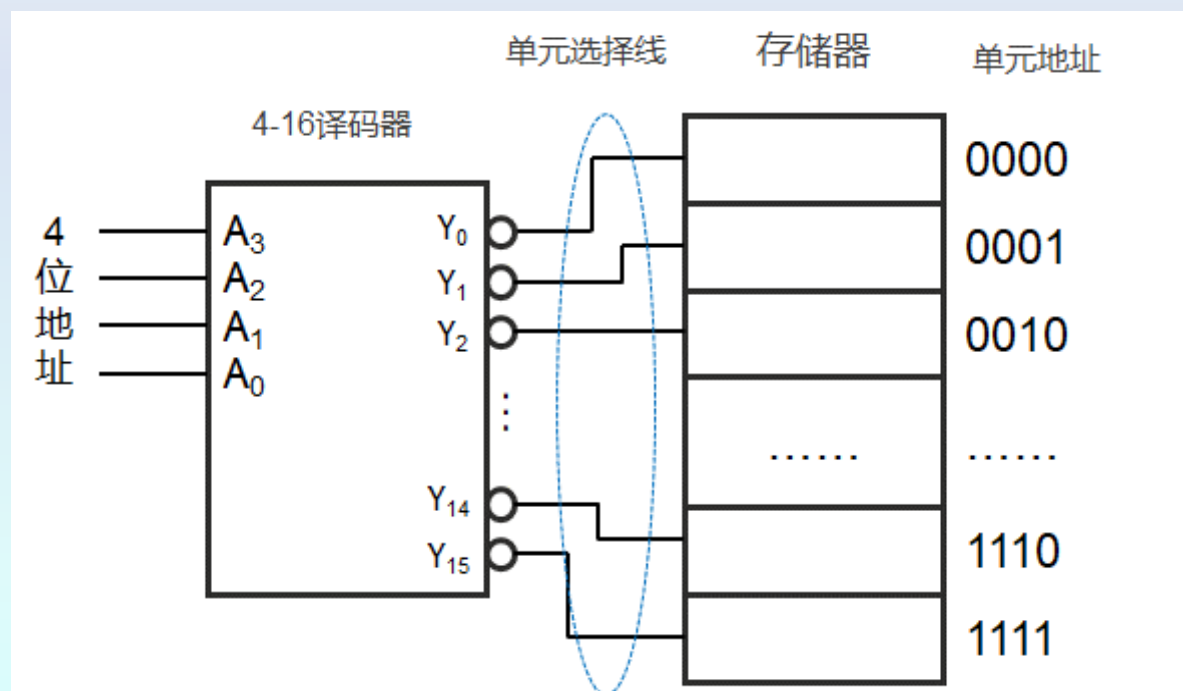
前面介绍过，若输入信号的个数 N 与输出变量的位数 n 不满足 $N=2^n$ ，此电路称为非二进制编码器。

非二进制译码器种类很多，其中二—十进制译码器应用较广泛。二—十进制译码器又称 4-10 线译码器，属不完全译码器。二—十进制译码器常用的型号有 TTL 系列的 54/7442、54/74LS42 和 CMOS 系列中的 54/74HC42、54/74HCT42 等。



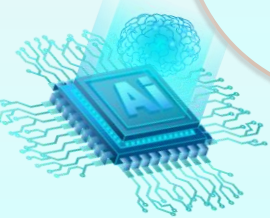
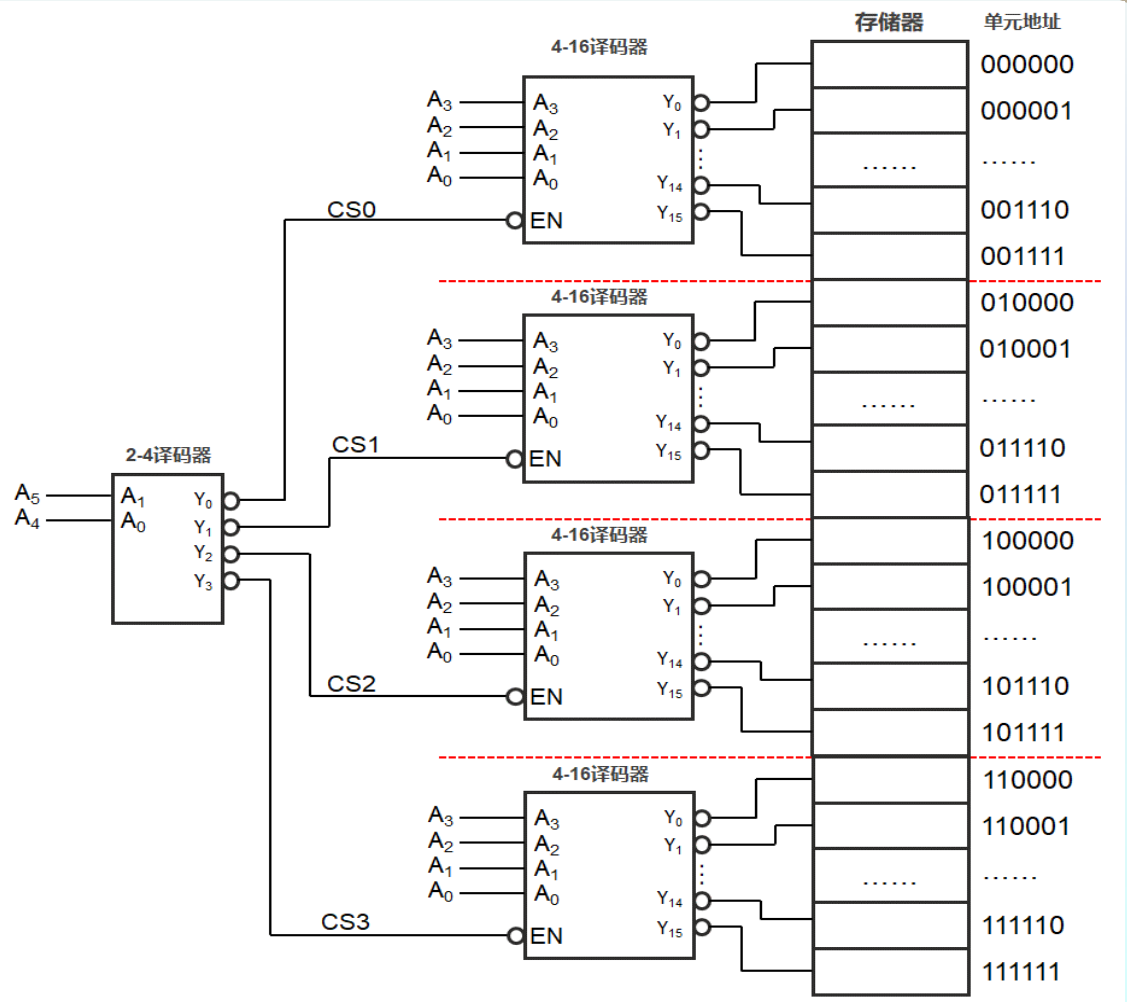
3. 译码器在存储器访问中的应用

4-16译码器实现16个存储单元控制逻辑电路





多级译码实现64个存储单元控制逻辑电路



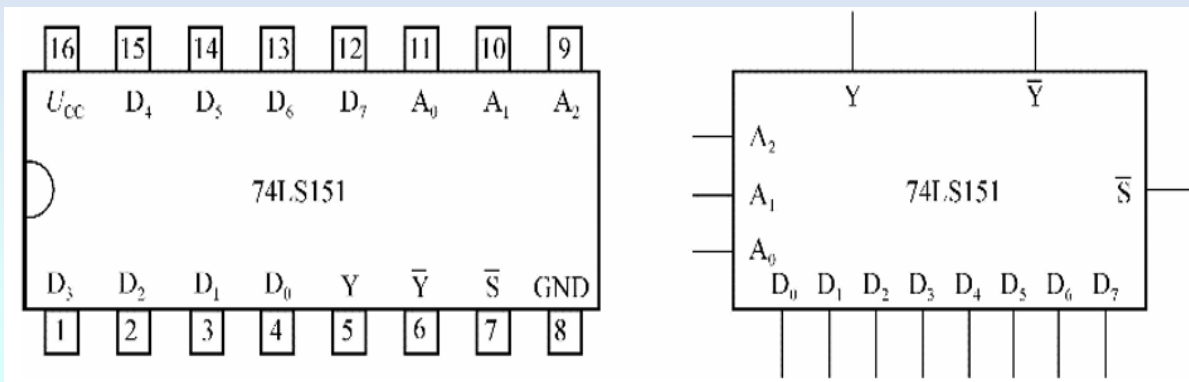
5.3 数据选择器与数值比较器

假如有多路信息需要通过一条线路传输或多路信息需要逐个处理，这时就要有一个电路，它能选择某个信息而排斥其他信息，这就称作**数据选择**。反之，把一路信息逐个安排到各输出端去，叫做**数据分配**。若从硬件角度比较两个数据的大小，则称为**数值比较器**。

5.3.1 数据选择器

数据选择器又称多路选择器或多路开关。它可以从多个输入信息中选择其中一个送至输出。常见的数据选择器有4选1、8选1、16选1等。

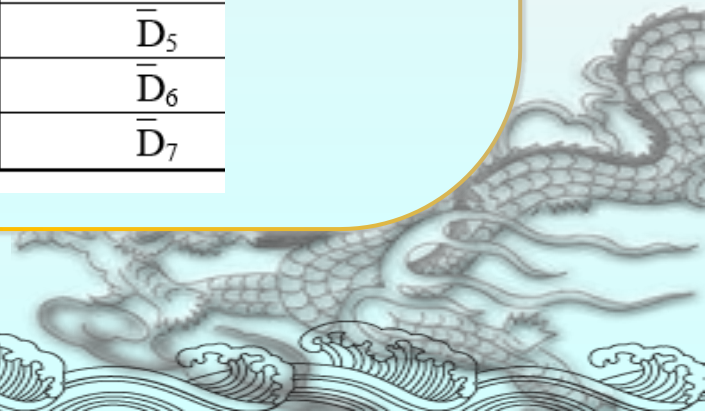
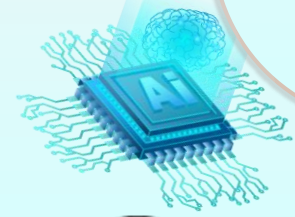
例：8选1数据选择器74LS151



它有 3 个地址输入端 A_2 、 A_1 、 A_0 ，8 个数据输入端 $D_0 \sim D_7$ ，2 个互补的输出端 Y 和 \bar{Y} ，1 个使能输入端 \bar{S} ， \bar{S} 为低电平有效。当 $\bar{S}=1$ 时，无论输入端 A_2 、 A_1 、 A_0 的状态如何，电路不工作，输出端 Y 为 0；当 $\bar{S}=0$ 时，电路根据输入端 A_2 、 A_1 、 A_0 的状态，在数 D_0 、 D_1 、 D_2 、 D_3 、 D_4 、 D_5 、 D_6 、 D_7 中选出对应的信号，并从输出端 Y 输出。

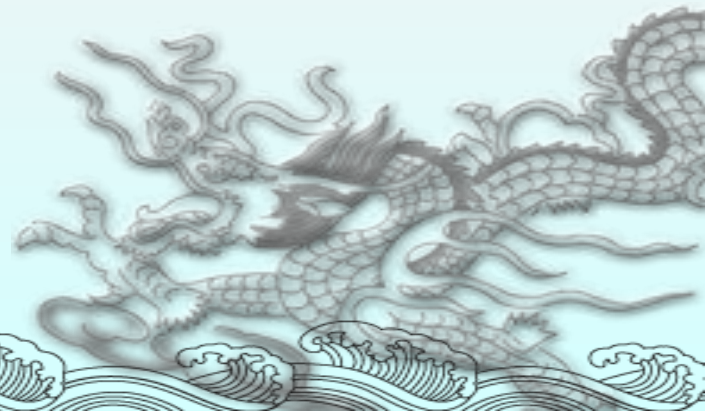
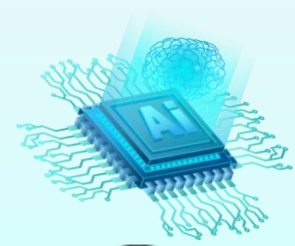
表5-7 8选1数据选择器74LS151功能表

输 入	地址输入			输 出	
\bar{S}	A_2	A_1	A_0	Y	\bar{Y}
1	×	×	×	0	1
0	0	0	0	D_0	\bar{D}_0
0	0	0	1	D_1	\bar{D}_1
0	0	1	0	D_2	\bar{D}_2
0	0	1	1	D_3	\bar{D}_3
0	1	0	0	D_4	\bar{D}_4
0	1	0	1	D_5	\bar{D}_5
0	1	1	0	D_6	\bar{D}_6
0	1	1	1	D_7	\bar{D}_7



5.3.2 数据分配器

在数据传输过程中，有时需要将某一路数据分配到多路装置中去，能够完成这种功能的电路称为**数据分配器**。根据输出的个数不同，数据分配器可分为4路分配器、8路分配器等。**数据分配器实际上是译码器的特殊应用**。带有使能端的译码器都具有数据分配器的功能。一般2-4线译码器可作为4路分配器，3-8线译码器作为8路分配器，4-16线译码器作为16路分配器。它们的使能端作为数据线，其扩展方法同译码器。



5.3.3 数值比较器

在数字系统中，特别是在计算机系统中经常要求比较两个数的大小。能对两个位数相同的二进制数进行比较，并判断其大小关系的逻辑电路称为数值比较器。数值比较器对两个位数相同的二进制数 A 、 B 进行比较，其比较的结果有 $A > B$ 、 $A < B$ 和 $A = B$ 三种可能性。

1. 一位数值比较器

当 A 和 B 都是1位二进制数时，它们只能取0或1两种值，由此可写出1位数值比较器的真值表

真值表

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

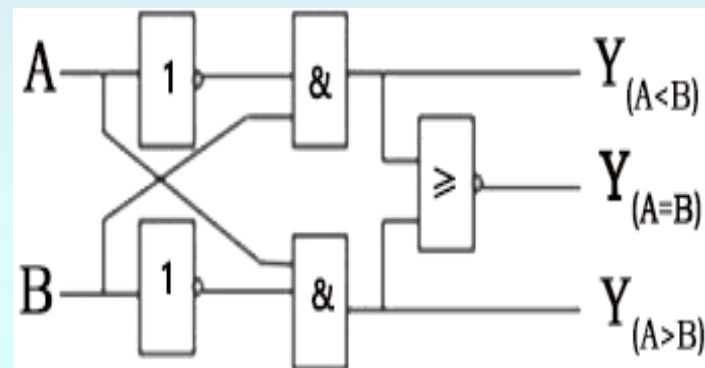
逻辑关系

$$Y_{(A > B)} = A\bar{B}$$

$$Y_{(A < B)} = \bar{A}B$$

$$Y_{(A = B)} = \bar{A}\bar{B} + AB$$

逻辑电路图



2. 二位数值比较器

当A和B都是2位二进制数时，分别用 A_1A_0 和 B_1B_0 表示，用 $Y_{(A>B)}$ 、 $Y_{(A<B)}$ 和 $Y_{(A=B)}$ ，表示比较的结果。当高位 A_1 和 B_1 不相等时， A_1 和 B_1 比较的结果就是两个数的比较结果，无需比较低位；当高位 A_1 和 B_1 相等时，两数的比较结果由低位比较的结果决定。利用1位数值的比较结果

真值表

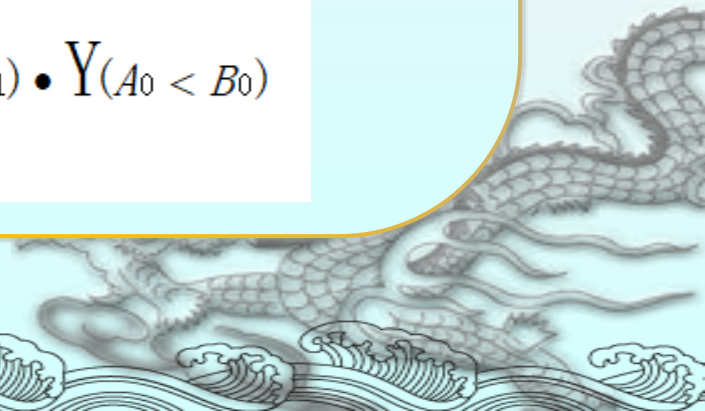
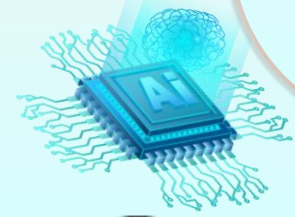
输 入				输 出		
A_1	B_1	A_0	B_0	$Y_{(A>B)}$	$Y_{(A<B)}$	$Y_{(A=B)}$
$A_1 > B_1$		X		1	0	0
$A_1 < B_1$		X		0	1	0
$A_1 = B_1$		$A_0 > B_0$		1	0	0
$A_1 = B_1$		$A_0 < B_0$		0	1	0
$A_1 = B_1$		$A_0 = B_0$		0	0	1

逻辑关系

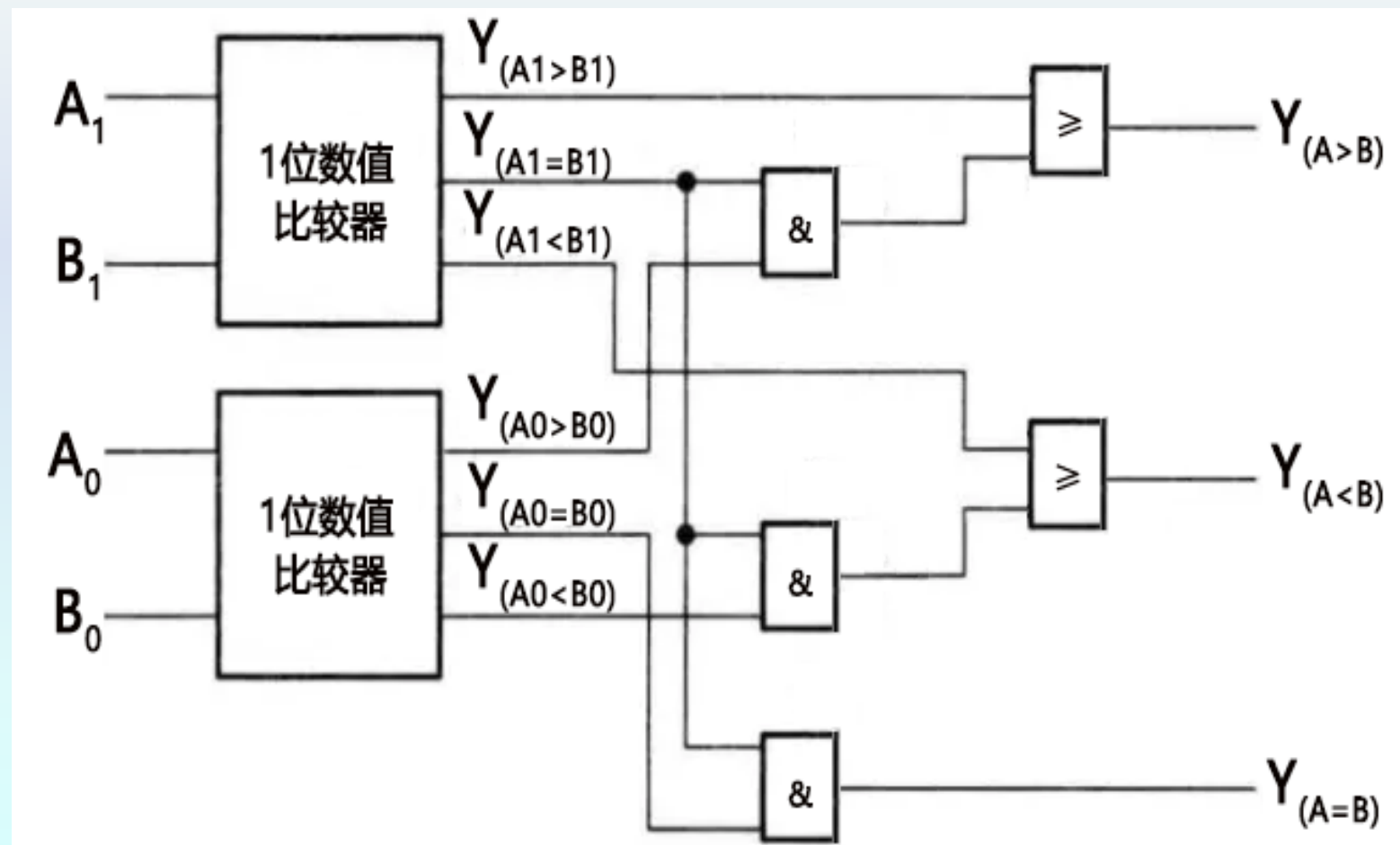
$$Y_{(A > B)} = A_1B_1 + (\overline{A_1}B_1 + A_1\overline{B_1})A_0B_0 = Y_{(A_1 > B_1)} + Y_{(A_1 = B_1)} \cdot Y_{(A_0 > B_0)}$$

$$Y_{(A < B)} = \overline{A_1}B_1 + (\overline{A_1}\overline{B_1} + A_1B_1)\overline{A_0}B_0 = Y_{(A_1 < B_1)} + Y_{(A_1 = B_1)} \cdot Y_{(A_0 < B_0)}$$

$$Y_{(A = B)} = Y_{(A_1 = B_1)} \cdot Y_{(A_0 = B_0)}$$



逻辑电路图

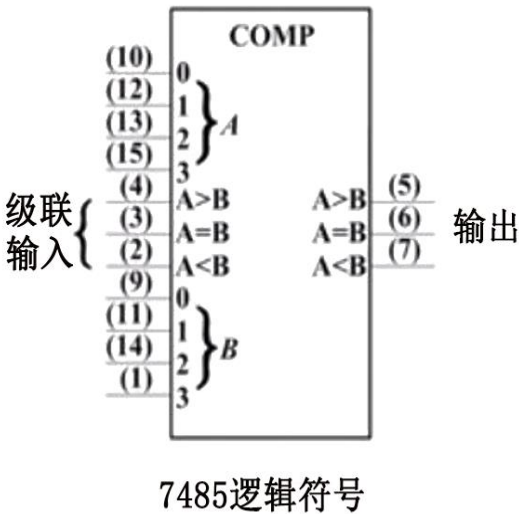




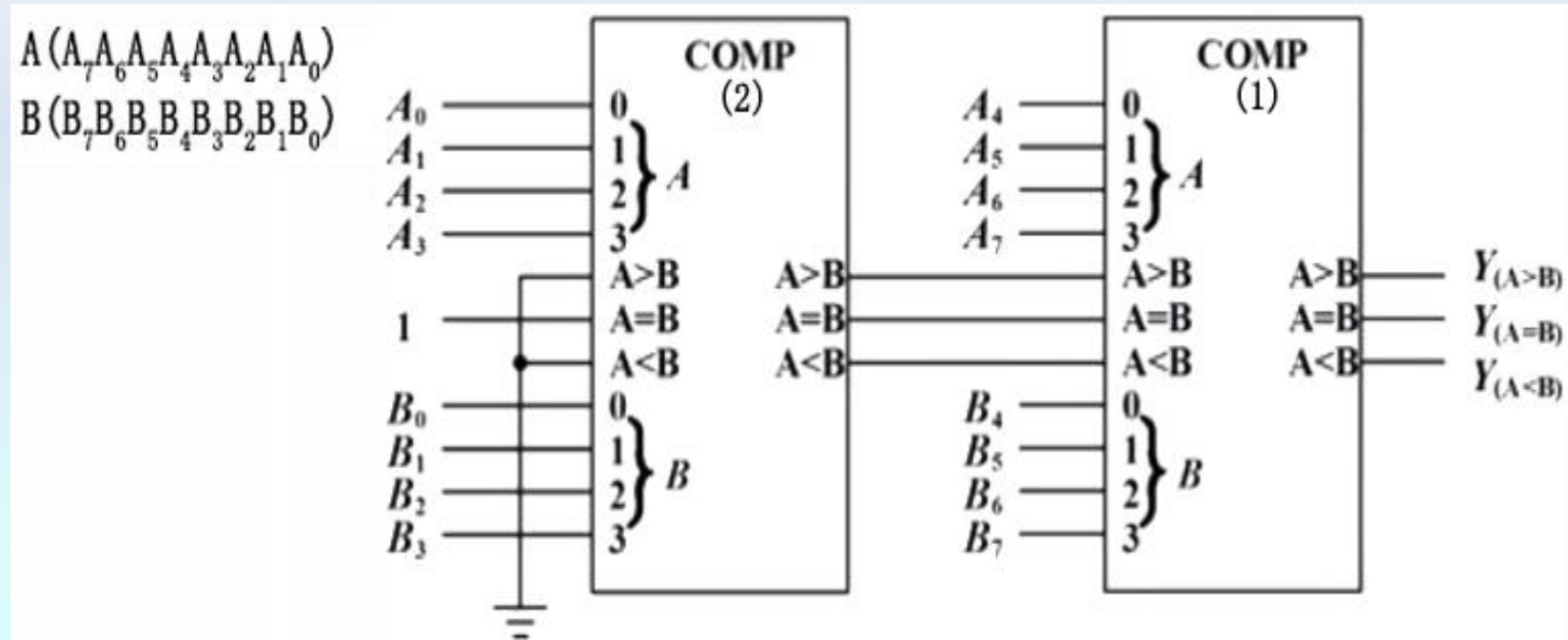
3. 4位集成数值比较器7485

7485实现4位二进制数A(A₃A₂A₁A₀)和B(B₃B₂B₁B₀)的数值大小的比较。

7485功能表									
比较输入				级联输入			输出		
A ₃ B ₃	A ₂ B ₂	A ₁ B ₁	A ₀ B ₀	I _(A>B)	I _(A<B)	I _(A=B)	Y _(A>B)	Y _(A<B)	Y _(A=B)
A ₃ >B ₃	×	×	×	×	×	×	1	0	0
A ₃ <B ₃	×	×	×	×	×	×	0	1	0
A ₃ =B ₃	A ₂ >B ₂	×	×	×	×	×	1	0	0
A ₃ =B ₃	A ₂ <B ₂	×	×	×	×	×	0	1	0
A ₃ =B ₃	A ₂ =B ₂	A ₁ >B ₁	×	×	×	×	1	0	0
A ₃ =B ₃	A ₂ =B ₂	A ₁ <B ₁	×	×	×	×	0	1	0
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ >B ₀	×	×	×	1	0	0
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ <B ₀	×	×	×	0	1	0
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	1	0	0	1	0	0
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	0	1	0	0	1	0
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	×	×	1	0	0	1
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	1	1	0	0	0	0
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	0	0	0	1	1	0



两片7485级联可以实现8位二进制数的比较





Thank you

