

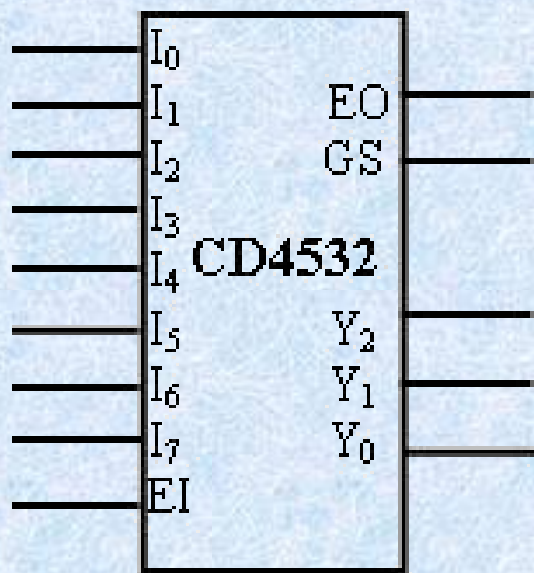
第4章 数字逻辑电路

- 4.1 逻辑电路的分析与设计
- 4.2 中规模集成逻辑电路及应用
- 4.3 硬件描述语言和可编程逻辑器件

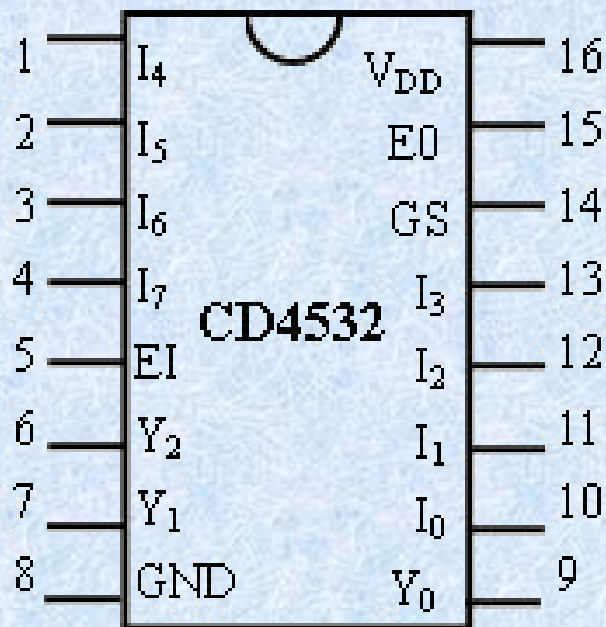
4.2.1 中规模编码器和译码器

中规模集成组合逻辑电路功能比较完善，除基本功能外，还具有控制、功能扩展等，能方便地实现多片集成芯片的电路连接，应用广泛。

1、CMOS中规模集成优先编码器CD4532



(a) 简化逻辑符号



(b) 引脚排列图



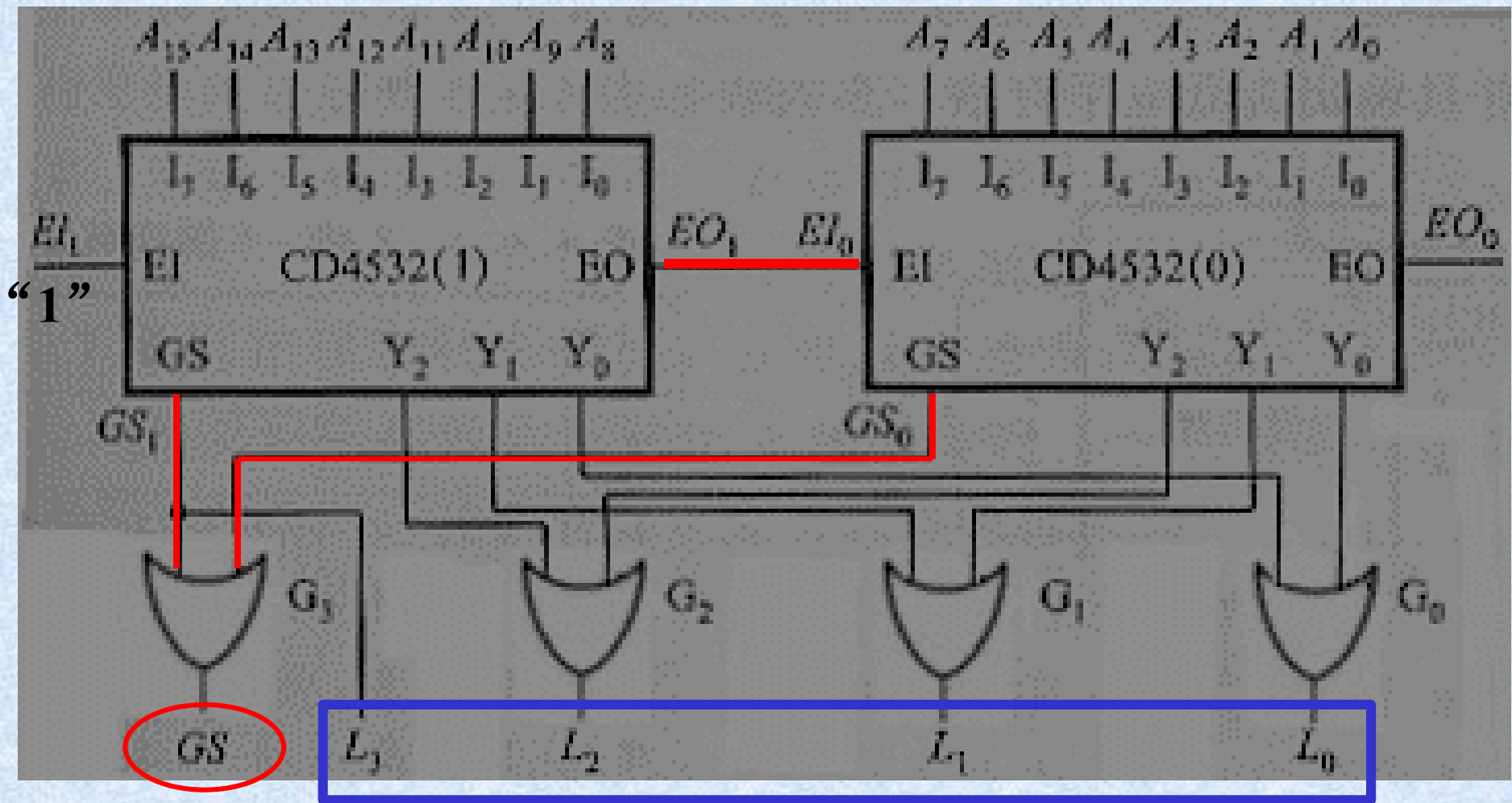
8线—3线优先编码器CD4532功能表

编码器输入									代码和控制输出				
EI	I ₇	I ₆	I ₅	I ₄	I ₃	I ₂	I ₁	I ₀	Y ₂	Y ₁	Y ₀	GS	EO
0	×	×	×	×	×	×	×	×	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	1
1	1	×	×	×	×	×	×	×	1	1	1	1	0
1	0	1	×	×	×	×	×	×	1	1	0	1	0
1	0	0	1	×	×	×	×	×	1	0	1	1	0
1	0	0	0	1	×	×	×	×	1	0	0	1	0
1	0	0	0	0	1	×	×	×	0	1	1	1	0
1	0	0	0	0	0	1	×	×	0	1	0	1	0
1	0	0	0	0	0	0	1	×	0	0	1	1	0
1	0	0	0	0	0	0	0	1	0	0	0	1	0

EO只有在EI=1，而且无编码对象输入时，输出为1，它可以与相同芯片的EI相连，**实现多片优先编码器的连接**（芯片扩展）

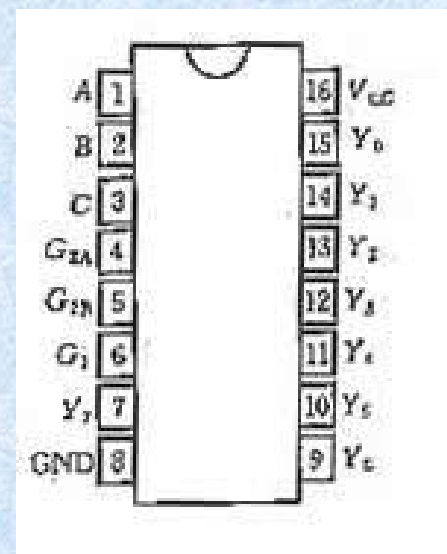
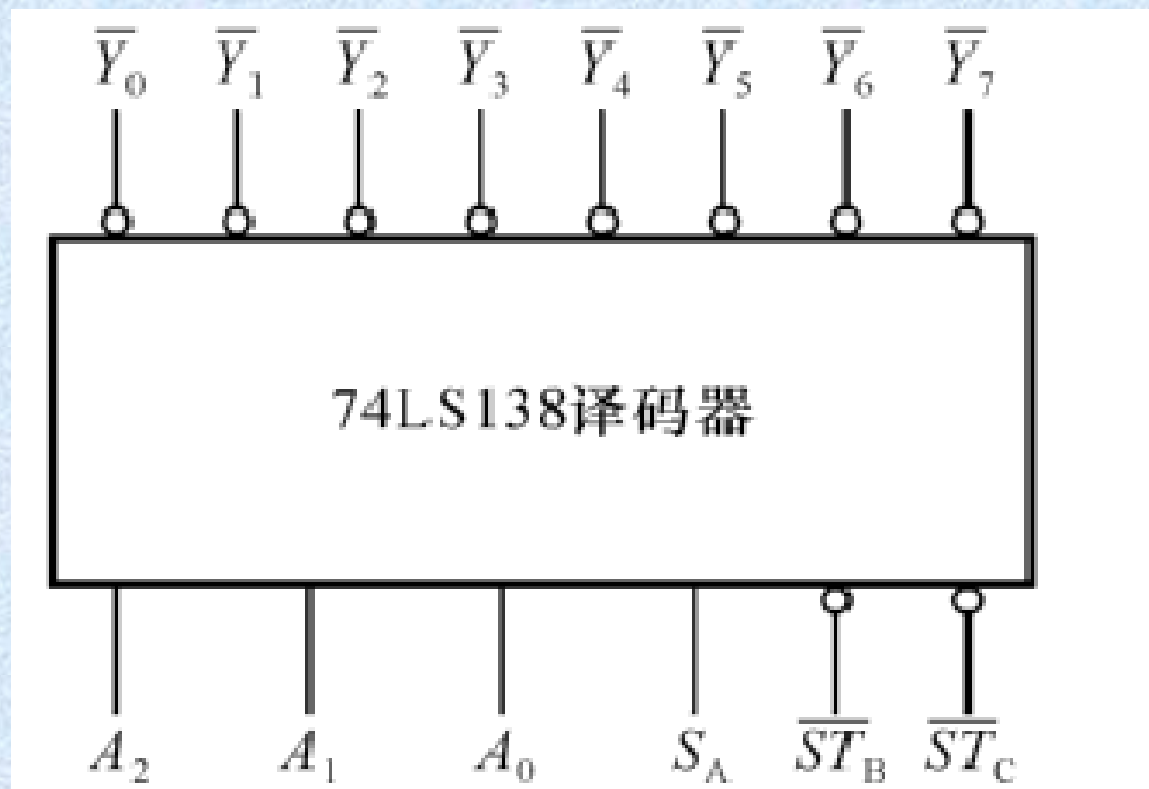
GS端当EI=1，且有编码输入时才为1，**表示编码器处于工作状态**，用于**区分**当编码器无输入和只有I₀输入的情况。

◆ 芯片的扩展使用



两片CD4532组成16线/4线优先编码器

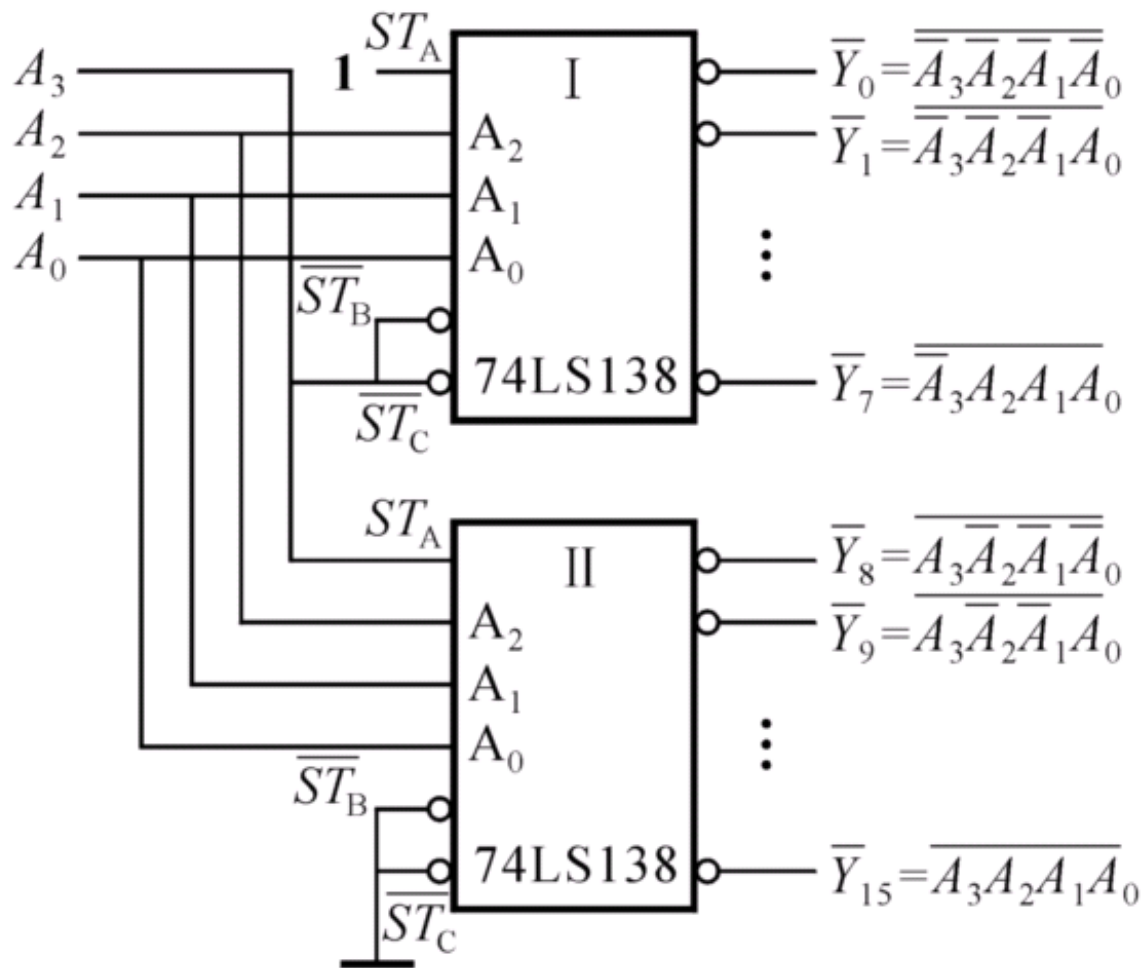
2、 中规模集成3线—8线译码器 74LS138



3/8译码器真值表

[illegible]

(1) 用3/8译码器采用分时制工作方式实现的4/16线译码



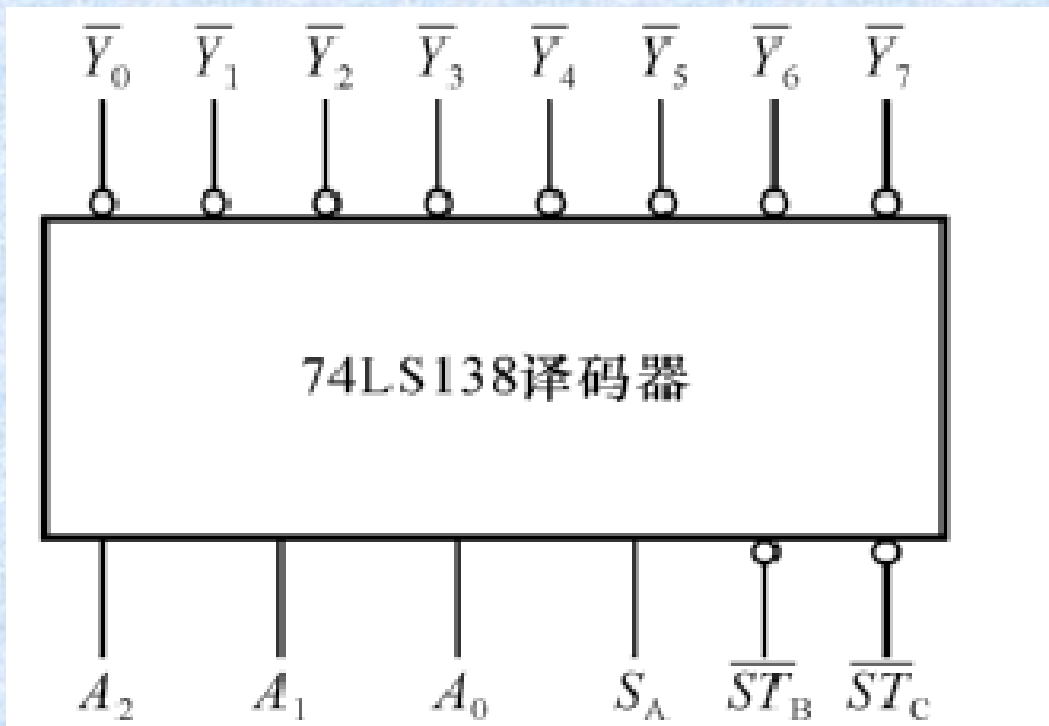
(2) 译码器应用举例

二进制译码器的每一个输出就是一个最小项，而任何一个输出函数都可以表示为最小项之和表达式，所以，译码器配上适当的逻辑门电路就可实现各种组合电路。

【例】：用3线-8线译码器74LS138及最少量的与非门实现下列逻辑函数。

$$Z = AB + AC + BC$$

$$Z = ABC + A\overline{B}\overline{C} + \overline{A}BC + \overline{A}\overline{B}C = \overline{\overline{ABC} \overline{ABC} \overline{ABC} \overline{ABC}} = \overline{Y_3 Y_5 Y_6 Y_7}$$



$$Z = \overline{\overline{Y_3 Y_5 Y_6 Y_7}}$$

$$Z = AB + AC + BC$$

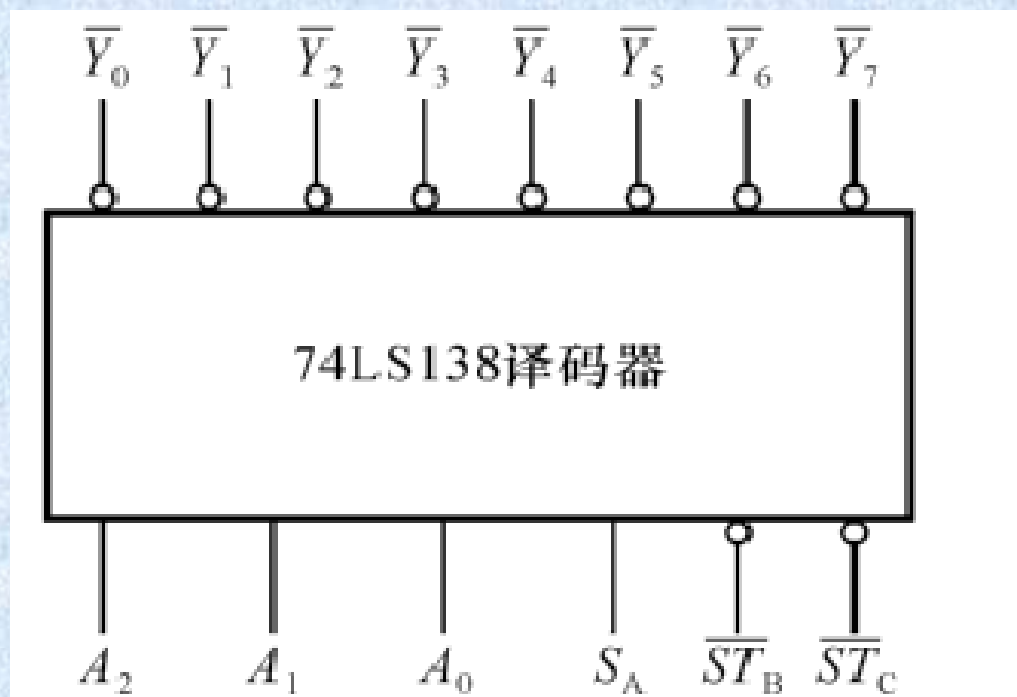
$$Z = ABC + A\overline{B}\overline{C} + \overline{A}BC + \overline{A}\overline{B}C$$

$$= \overline{\overline{ABC} \overline{ABC} \overline{ABC} \overline{ABC}}$$



例：试用3/8译码器设计一个能判别四位二进制码中1的位数是奇数还是偶数的奇偶识别电路

可用与非和与或非两种门电路。



解：令四位二进制码为 $A_3A_2A_1A_0$ ，输出 Y_{OD} 表示1的位数为奇数， ~~Y_E 为偶数~~。则卡诺图为：

$A_3A_2 \backslash A_1A_0$					
		00	01	11	10
A_3A_2	00	0	1	0	1
	01	1	0	1	0
	11	0	1	0	1
	10	1	0	1	0

图中1方格表示奇数，0格为偶数。
所以结合1格得到 Y_{OD} 函数，~~结合0~~
~~格得 Y_E 函数。~~

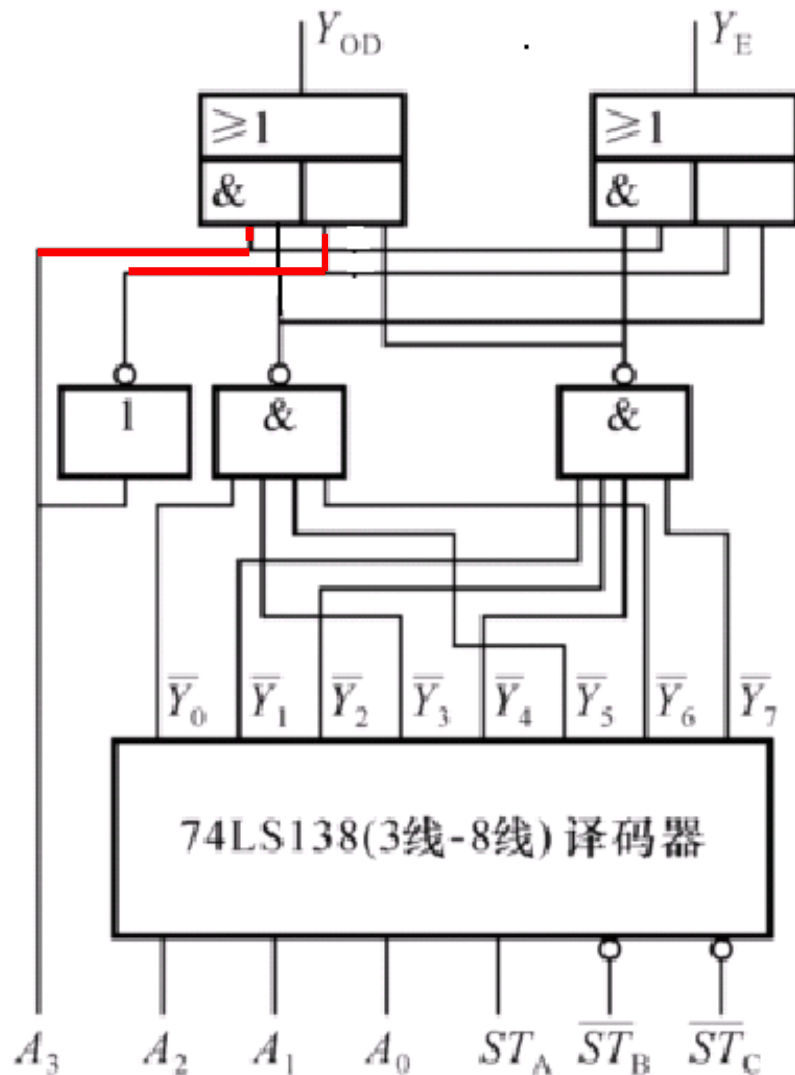
$$\begin{aligned}
 Y_{OD} &= \overline{A_3}(\overline{A_2}\overline{A_1}A_0 + \overline{A_2}A_1\overline{A_0} + A_2\overline{A_1}A_0 + A_2A_1\overline{A_0}) \\
 &+ A_3(\overline{A_2}\overline{A_1}A_0 + \overline{A_2}A_1A_0 + A_2\overline{A_1}A_0 + A_2A_1\overline{A_0}) \\
 &= \overline{A_3}(\overline{Y_1Y_2Y_4Y_7}) + A_3(\overline{Y_0Y_3Y_5Y_6})
 \end{aligned}$$

$$Y_{OD} = \overline{A_3}(\overline{Y_1 Y_2 Y_4 Y_7}) + A_3(\overline{Y_0 Y_3 Y_5 Y_6})$$

同理将有

$$\begin{aligned} Y_E &= \overline{A_3}(A_2 A_1 A_0 + \overline{A_2} A_1 A_0 + \\ &\quad A_2 \overline{A_1} A_0 + A_2 A_1 \overline{A_0}) \\ &\quad + A_3(\overline{A_2} \overline{A_1} A_0 + \overline{A_2} A_1 \overline{A_0} + \\ &\quad A_2 \overline{A_1} A_0 + A_2 A_1 A_0) \\ &= \overline{A_3}(\overline{Y_1 Y_2 Y_4 Y_7}) + A_3(\overline{Y_0 Y_3 Y_5 Y_6}) \end{aligned}$$

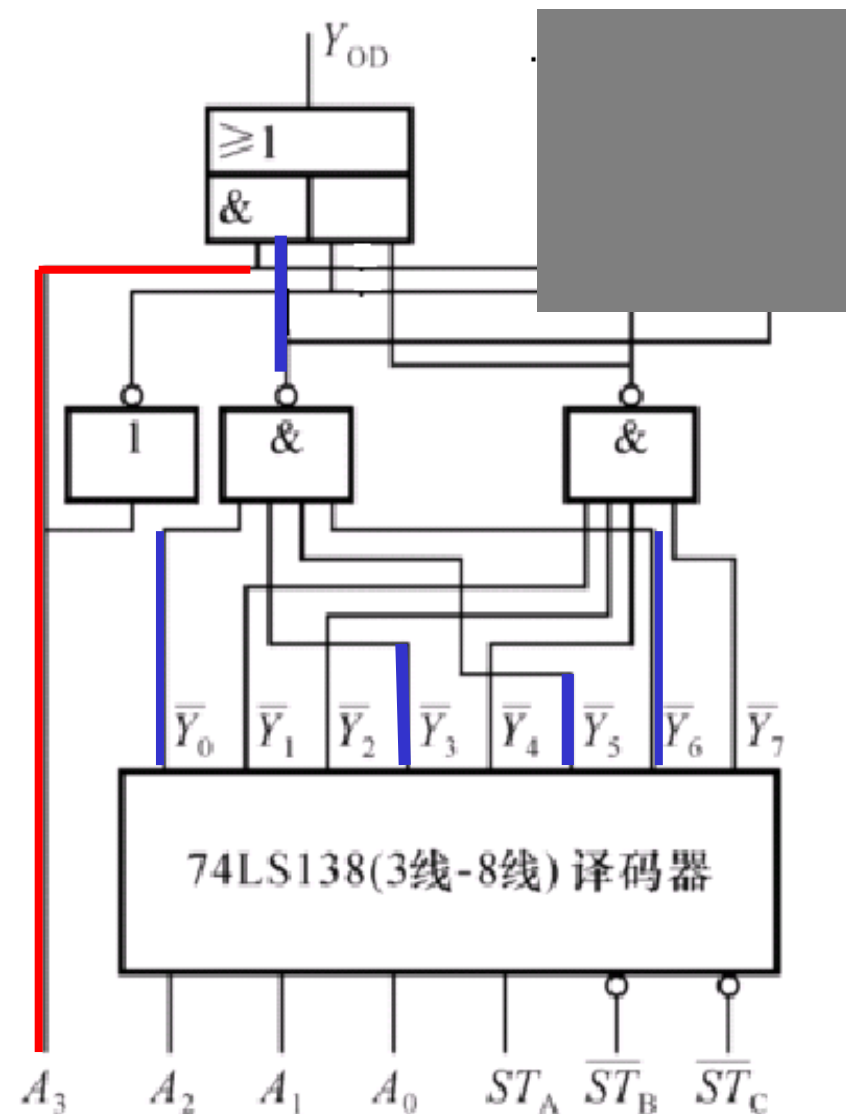
其电路为右图所示：



$$Y_{OD} = \overline{A_3}(\overline{Y_1 Y_2 Y_4 Y_7}) + A_3(\overline{Y_0 Y_3 Y_5 Y_6})$$



◆ 与或非门可以作为
二选一开关



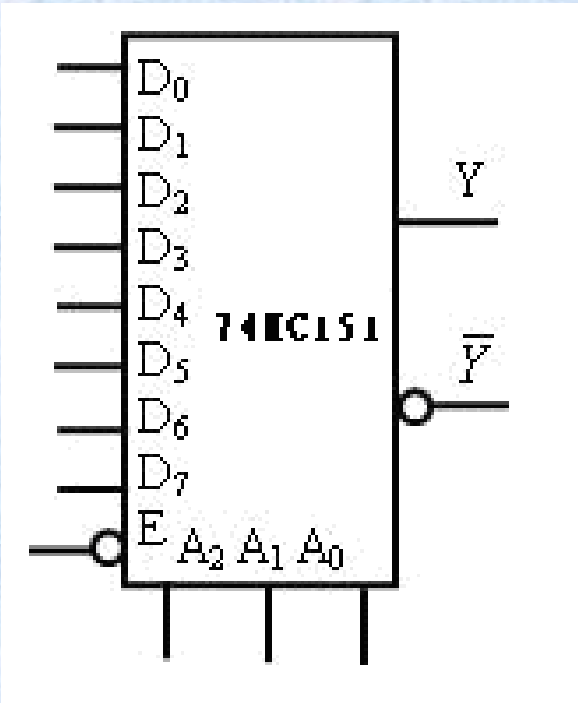
4.2.2 中规模集成数据选择器

从数据的传输方式讲，它是一个并行/串行的传输转换电路。

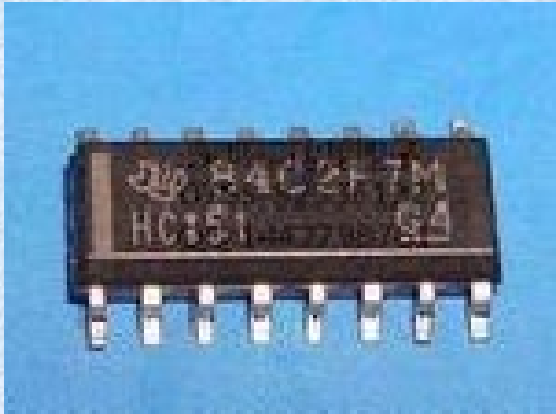
从电路的输出函数可知，它是一个与—或表达式，而电路的结构又是一个与或逻辑结构，因此，用该电路同样可以用来产生各种各样的组合逻辑电路。

74HC151数据选择器功能表

输 入				输 出	
使能输入	选择地址输入			原码输出	反码输出
\overline{E}	A_2	A_1	A_0	Y	\overline{Y}
1	×	×	×	0	1
0	0	0	0	D_0	$\overline{D_0}$
0	0	0	1	D_1	$\overline{D_1}$
0	0	1	0	D_2	$\overline{D_2}$
0	0	1	1	D_3	$\overline{D_3}$
0	1	0	0	D_4	$\overline{D_4}$
0	1	0	1	D_5	$\overline{D_5}$
0	1	1	0	D_6	$\overline{D_6}$
0	1	1	1	D_7	$\overline{D_7}$



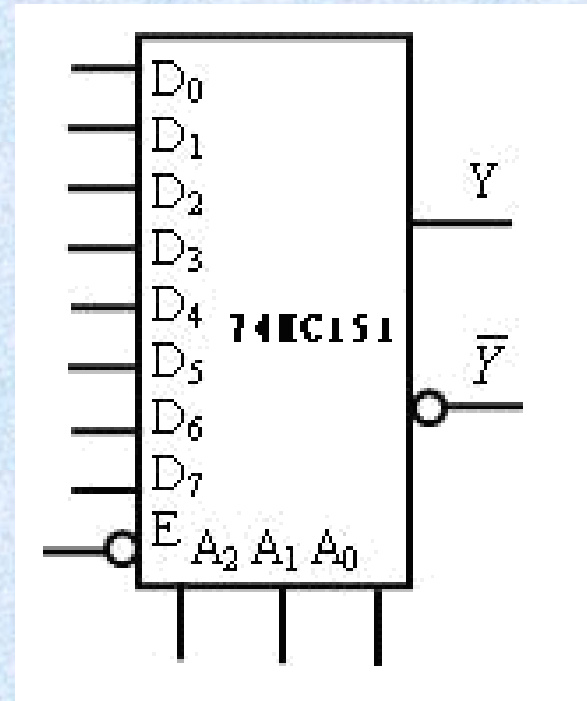
74HC151是一片CMOS集成的8选1数据选择器

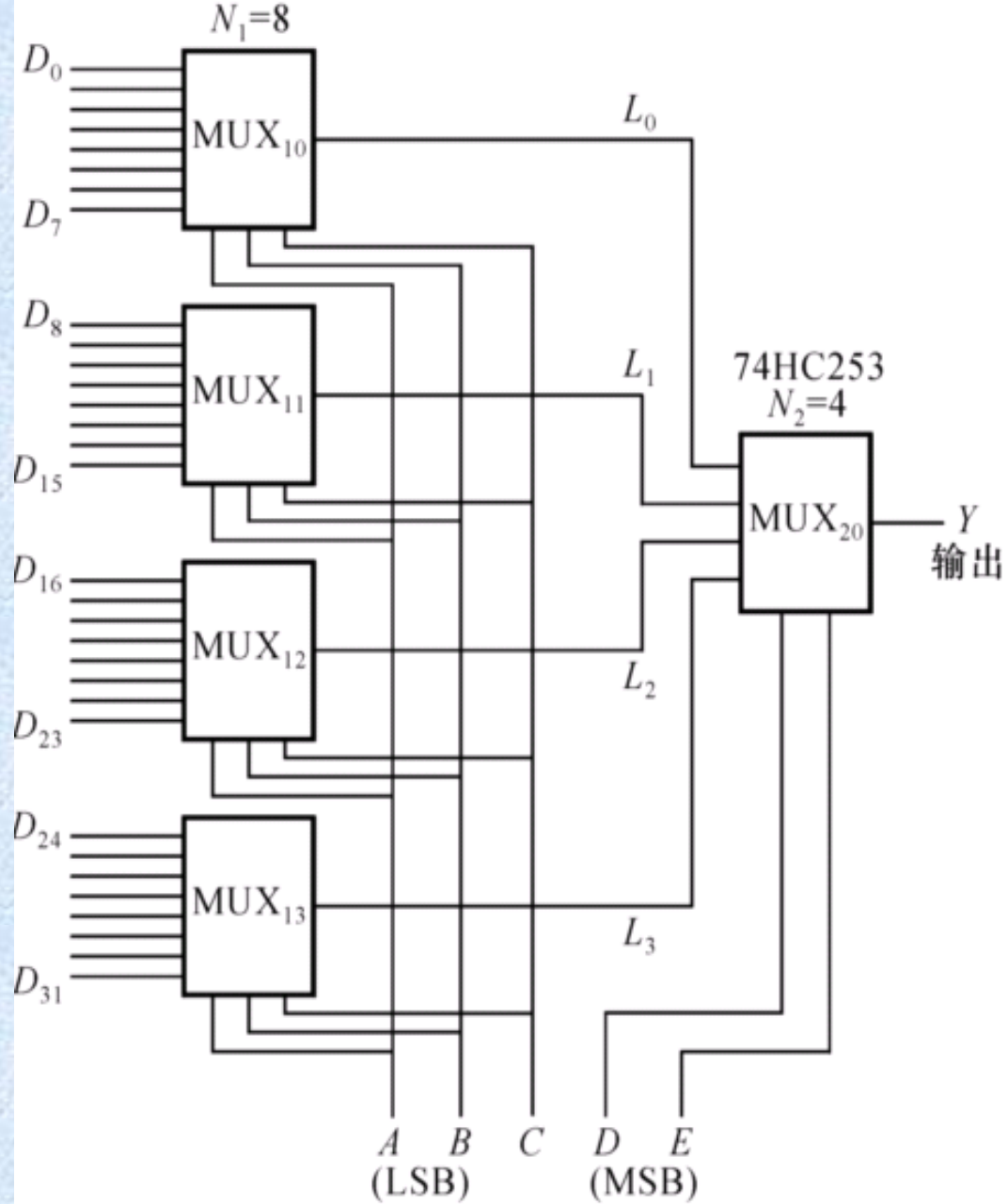


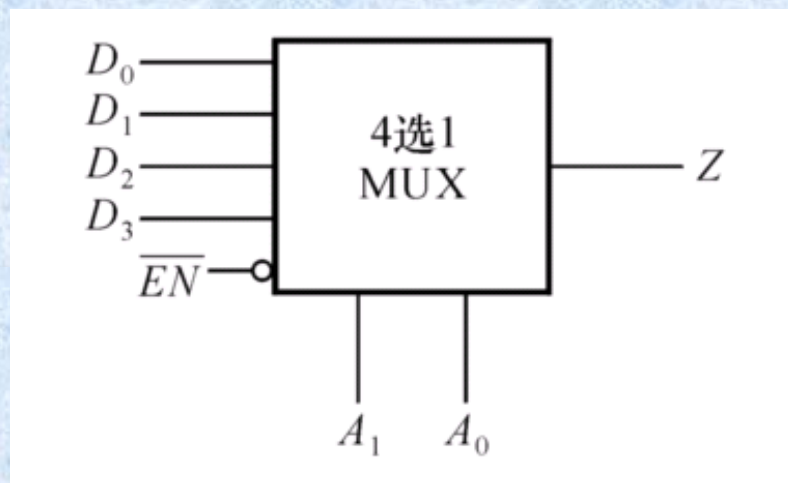
1、集成芯片的扩展使用

若输入的并行数据位增加时，
应采用扩展的方法。

如何用8/1和4/1扩展
为32选1数据选择器？







数据选择器输出：

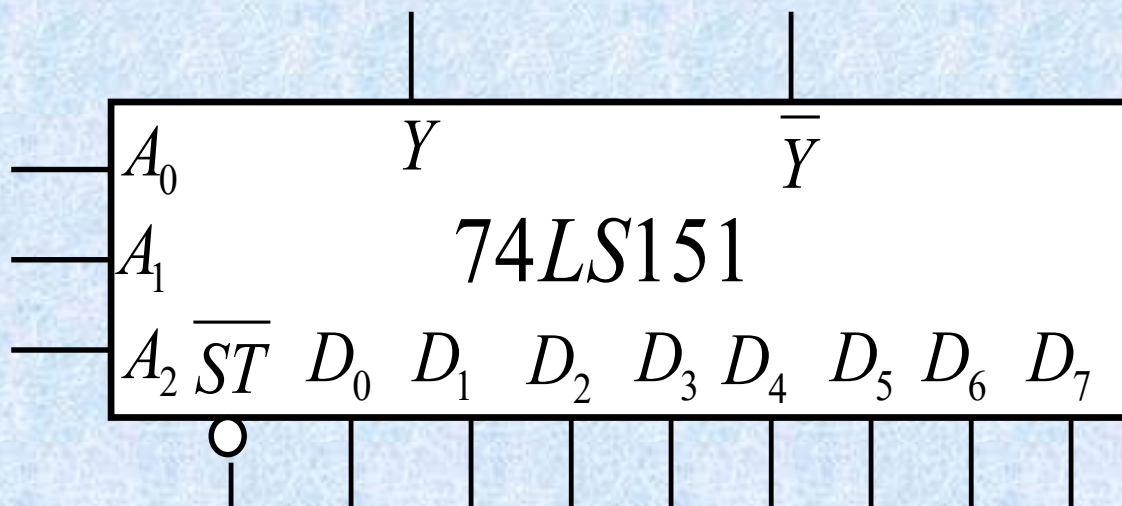
$$Z = D_3 A_1 A_0 + D_2 A_1 \overline{A_0} + D_1 \overline{A_1} A_0 + D_0 \overline{A_1} \overline{A_0}$$
$$= \sum_{i=0}^{2^n-1} D_i m_i$$

什么启发？

2、利用数据选择器实现组合逻辑函数

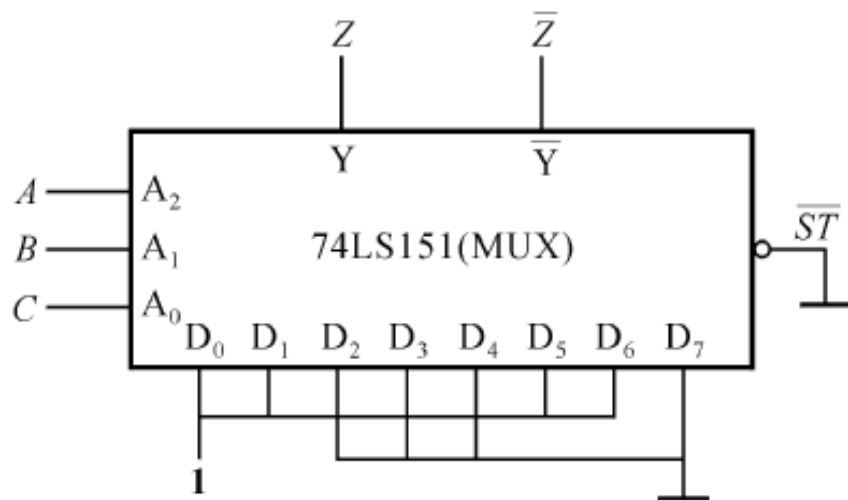
例：试用一片74LS151型8选1数据选择器实现函数

$$Z = f(A, B, C) = \overline{A}\overline{B} + \overline{B}C + A\overline{B}\overline{C}$$



解：选函数变量A, B, C从选择器地址 A_2, A_1, A_0 输入，根据多路选择器的原理，数据输入端必定是常量0或1。为找出地址最小项对应数据输入端的关系，画出卡诺图，图中直接求出了各数据端的值。

据此画出电路图：



$(A_2 A_1 A_0)$ ABC	(m_0)	(m_1)	(m_3)	(m_2)	(m_6)	(m_7)	(m_5)	(m_4)
	000	001	011	010	110	111	101	100
	1	1	0	0	1	0	1	0
	$D_0=1$	$D_1=1$	$D_3=0$	$D_2=0$	$D_6=1$	$D_7=0$	$D_5=1$	$D_4=0$

$$\begin{aligned}
 Z = f(A, B, C) &= \overline{A}\overline{B} + \overline{B}C + A\overline{B}\overline{C} \\
 &= \overline{A}\overline{B}C + \overline{A}\overline{B}\overline{C} + A\overline{B}C + \overline{A}\overline{B}\overline{C} + A\overline{B}\overline{C} \\
 &= \overline{A}\overline{B}C + \overline{A}\overline{B}\overline{C} + A\overline{B}C + A\overline{B}\overline{C}
 \end{aligned}$$

以上例子为采用具有 n 个地址端的数据选择器实现 n 变量的函数。

思考：当函数输入变量数小于或大于数据选择器的地址端（ A ）时，如何处理？

这类问题具体操作方法：

（1）在要实现的逻辑函数中选择适当的变量从选择器的地址输入。多余的变量从数据端加入。（当变量选得不同时，结果会不同）

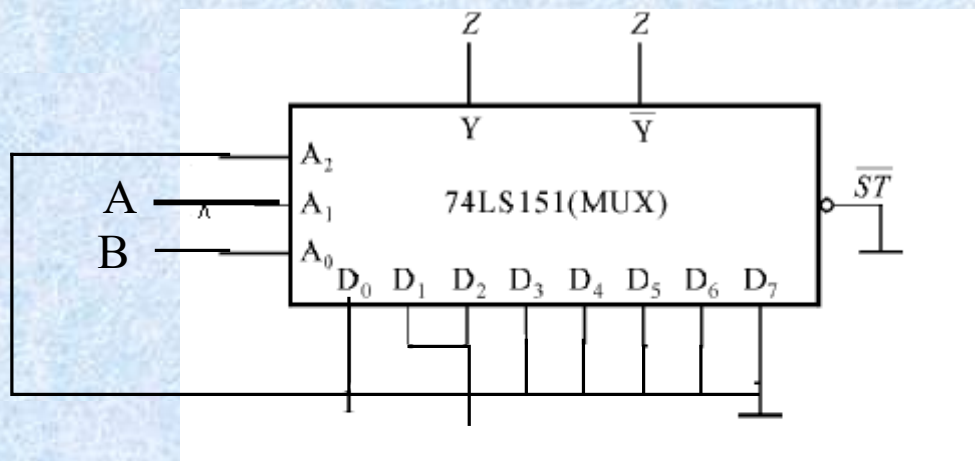
（2）画出给定函数的卡诺图，求出地址变量最小项所对应的数据输入端的逻辑关系。

（3）画出对应的逻辑图。

例：用8选1数据选择器实现二变量异或函数发生器

$$Z = A\bar{B} + \bar{A}B$$

另解：若把A2接高电平，
则D5和D6接“1”。

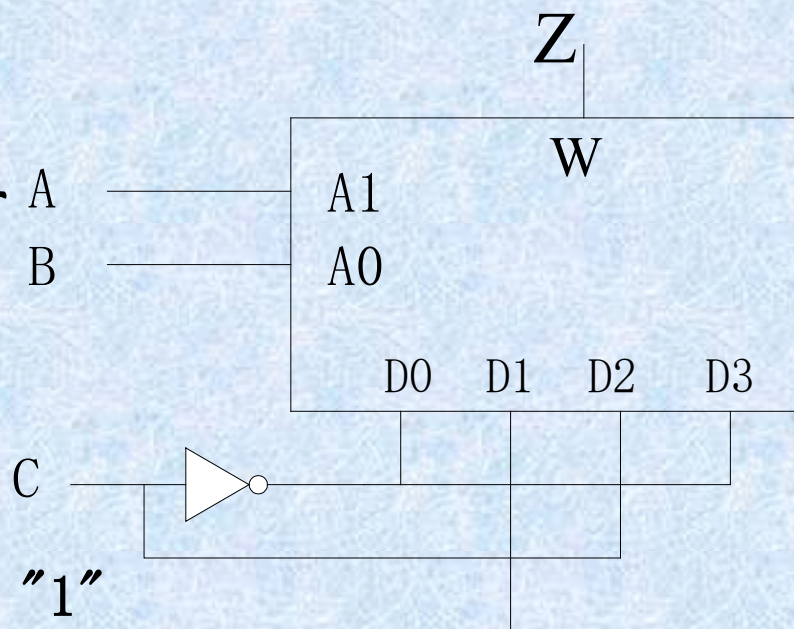


函数输入变量小于数据选择器的地址端（A）。将不用
的地址端和不用的数据输入端（D）都接地。

例：用4选1数据选择器实现：

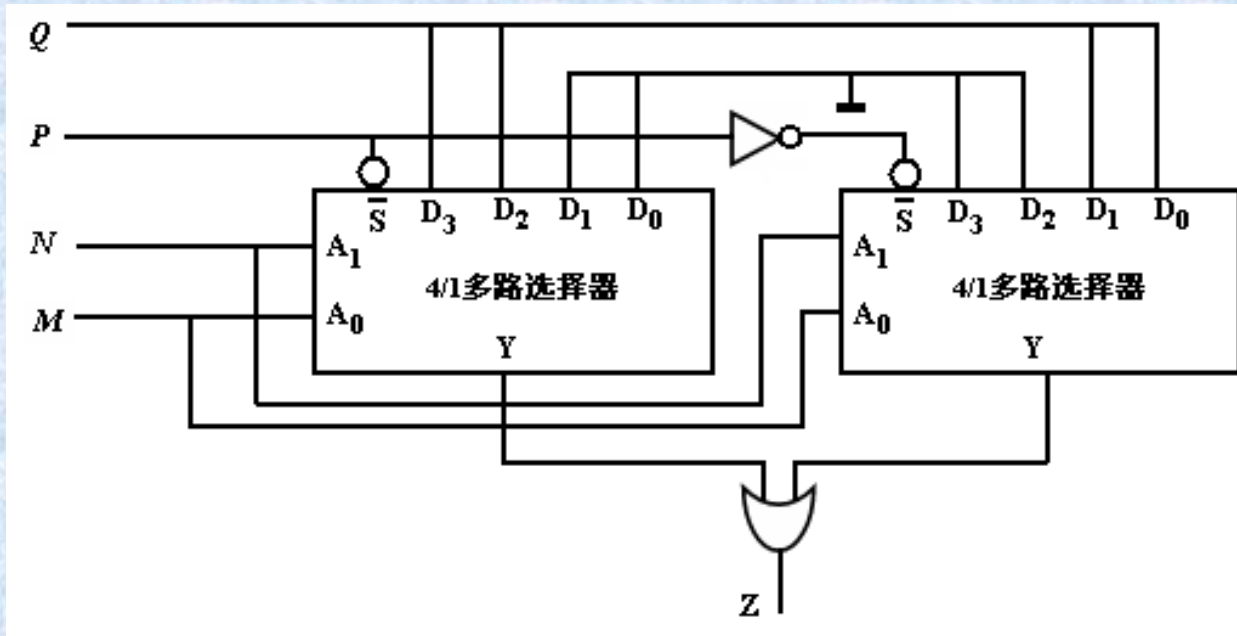
$$\begin{aligned} Z(A, B, C) &= \sum m(0, 2, 3, 5, 6) \\ &= \overline{A}\overline{B}\overline{C} + \overline{A}B\overline{C} + \overline{A}BC + A\overline{B}\overline{C} + ABC \\ &= \overline{A}\overline{B}\overline{C} + \overline{A}B + A\overline{B}\overline{C} + ABC \end{aligned}$$

当函数输入变量大于数据选择地址端（A）时，可能随着选用函数输入变量作地址的方案不同，设计结果不同。



例： 下图 是用两个4选1数据选择器组成的逻辑电路，试写出输出Z与输入M、N、P、Q之间的逻辑函数式。已知数据选择器的逻辑函数式是

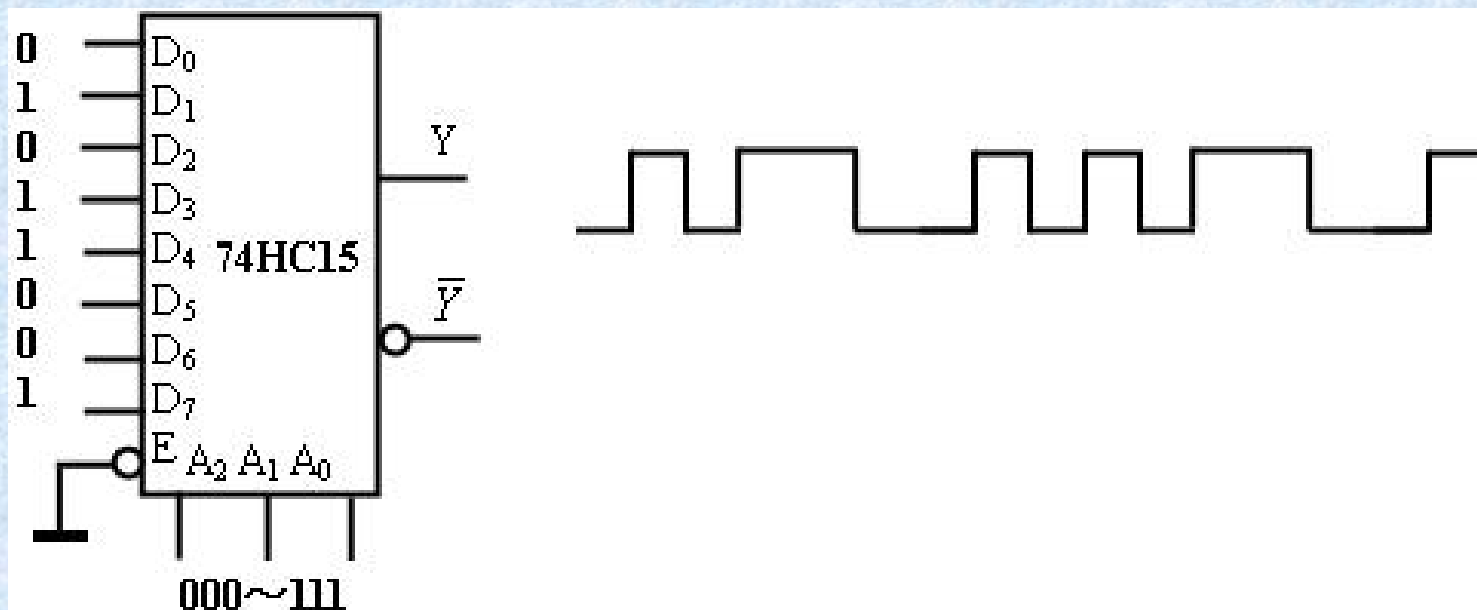
$$Y = (D_0 \bar{A}_1 \bar{A}_0 + D_1 \bar{A}_1 A_0 + D_2 A_1 \bar{A}_0 + D_3 A_1 A_0) \cdot S$$



$$\begin{aligned} Z &= (N\bar{M}Q + NMQ)\bar{P} + (\bar{N}\bar{M}Q + \bar{N}MQ)P \\ &= N\bar{P}Q + \bar{N}PQ = (N \oplus P)Q \end{aligned}$$

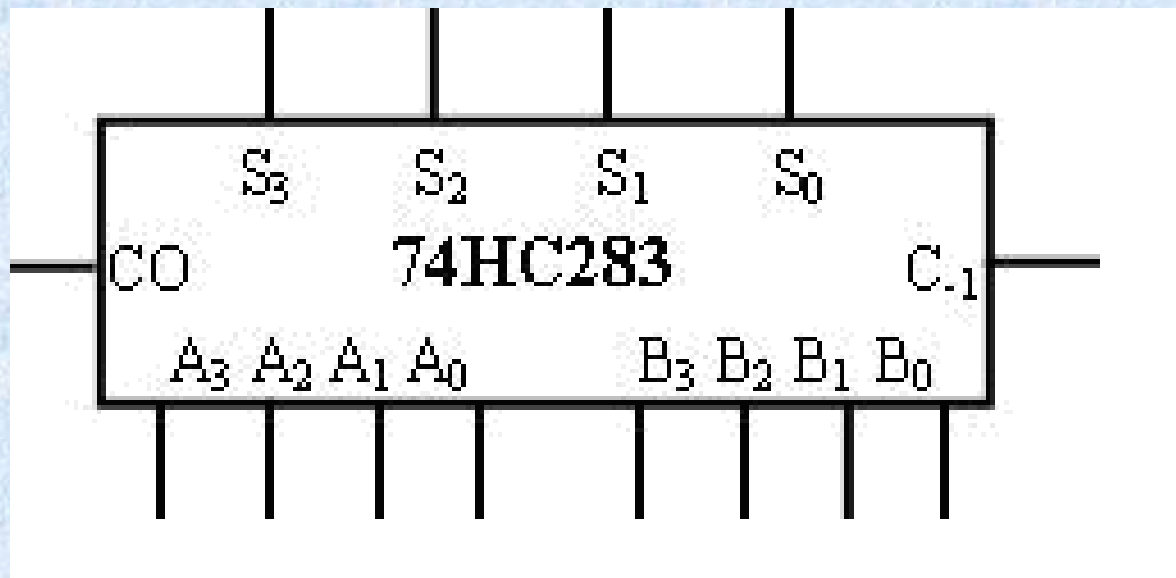
3、用于产生序列脉冲

如果把多路数据选择器的数据输入端接预先设计好的序列数据，而在地址控制端依次加上地址，则在选择器的输出端将可以输出序列脉冲。如图是产生01011001序列脉冲的电路。

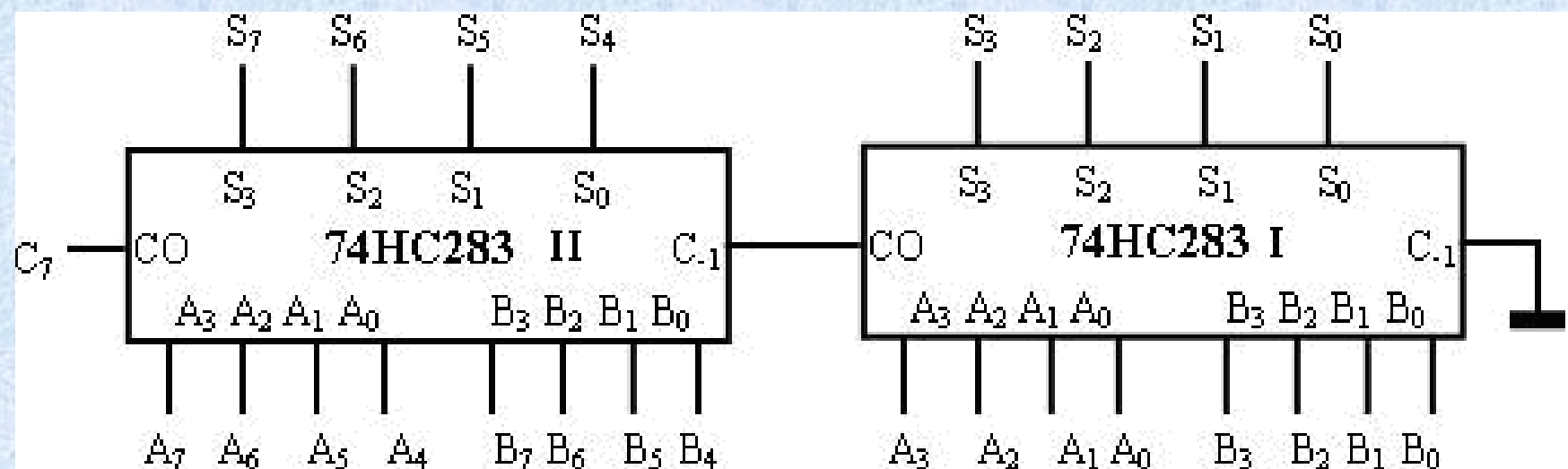


4.2.3 中规模集成加法器与数值比较器

一、加法器 74HC283



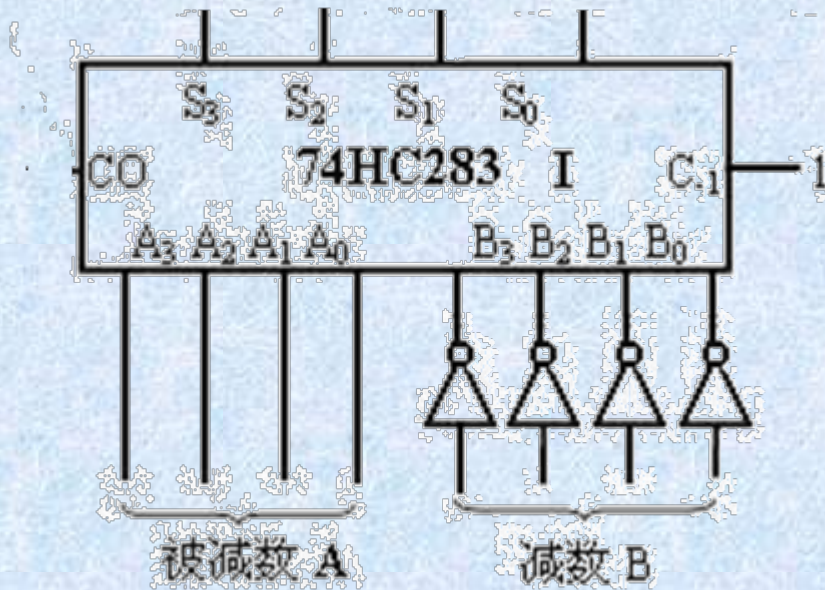
1. 实现多位二进制数相加

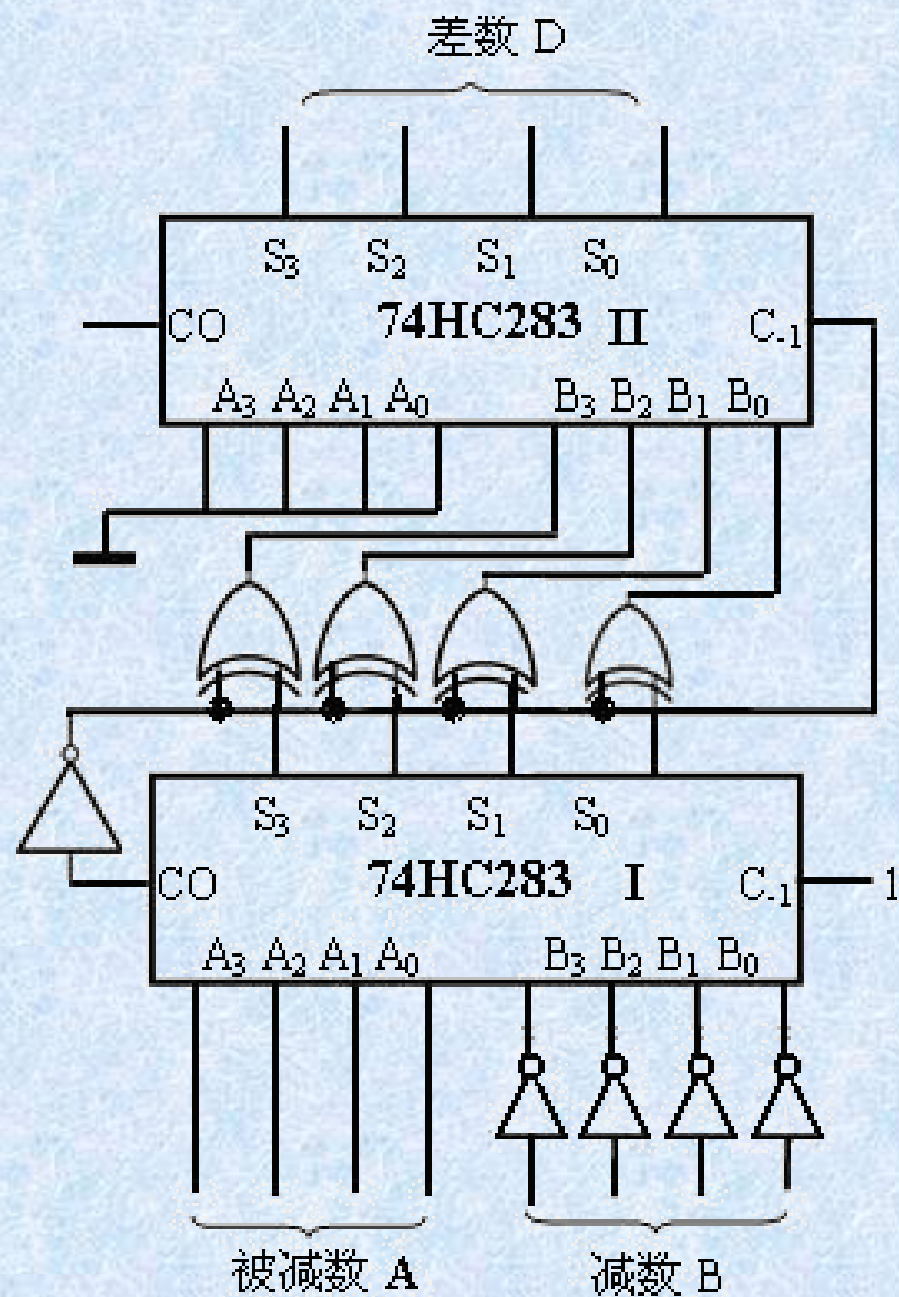


2、用74HC283实现减法运算

被减数大于减数？

可逆加法？





3、实现代码转换

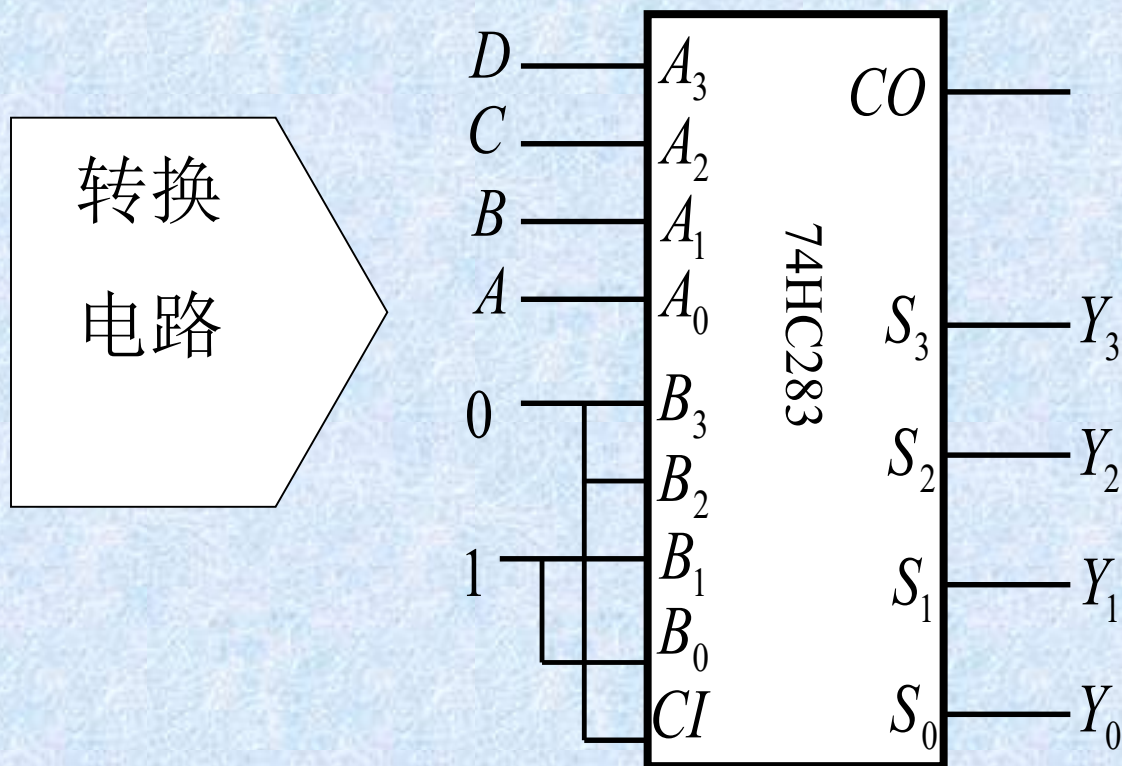
用74HC283加法器将BCD代码的8421码转换成余三码

以8421码为输入、
余三码为输出列出
真值表

输 入				输 出			
D	C	B	A	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0

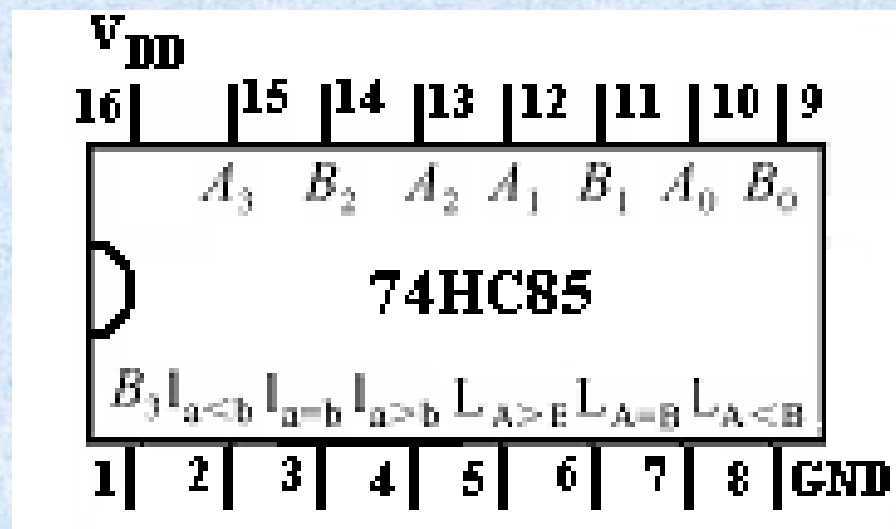
由表可知：输出即为输入加3（即0011）

所以有： $Y_3Y_2Y_1Y_0 = DCBA + 0011$

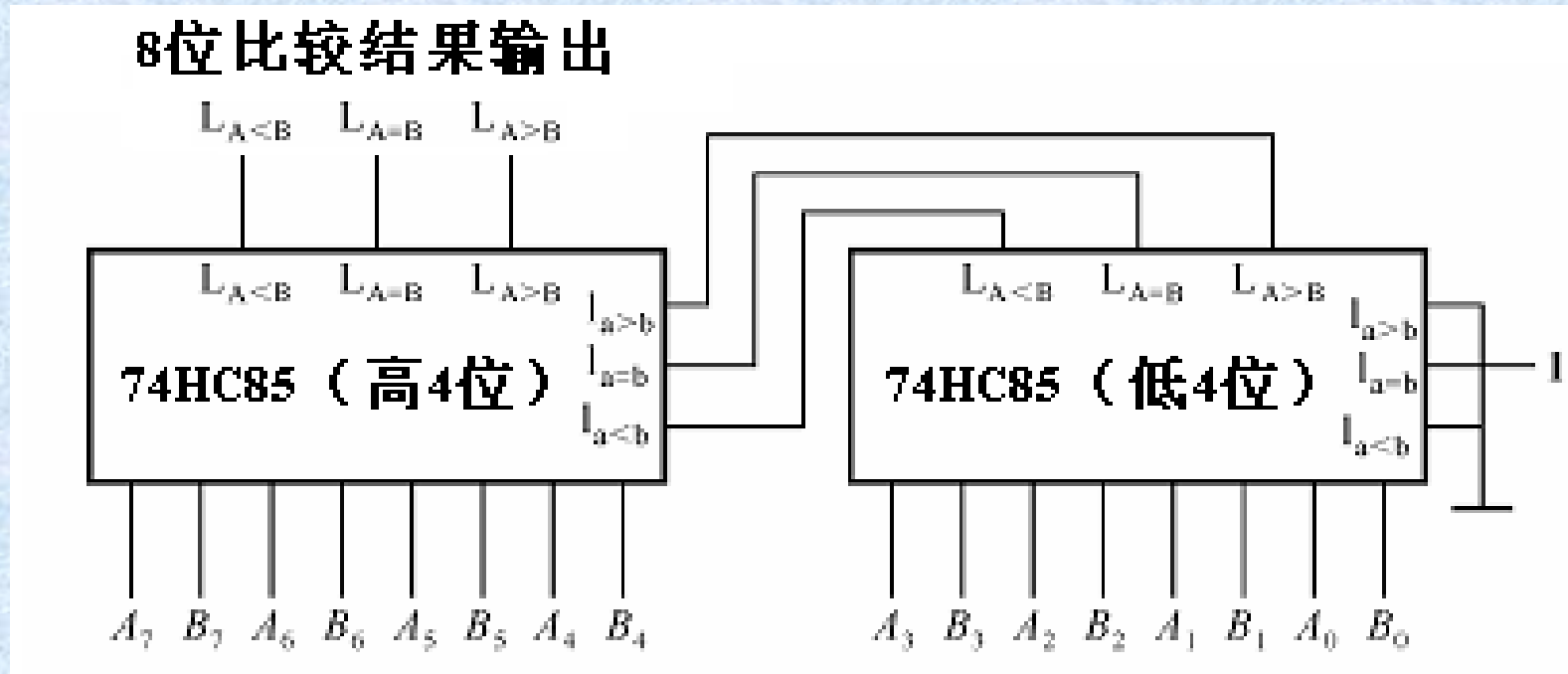


二、数值比较器74HC85

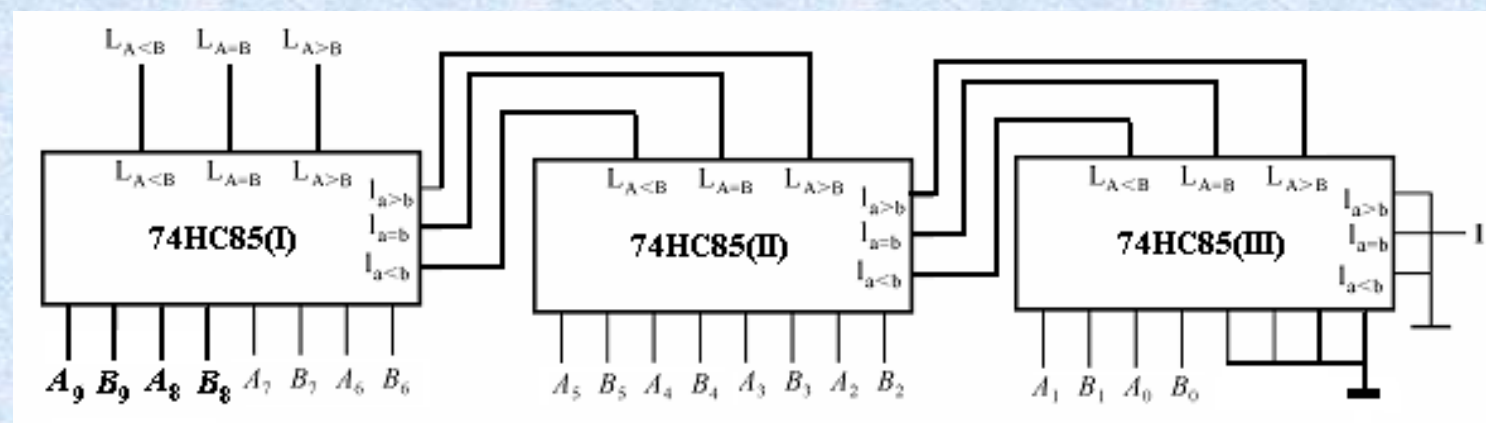
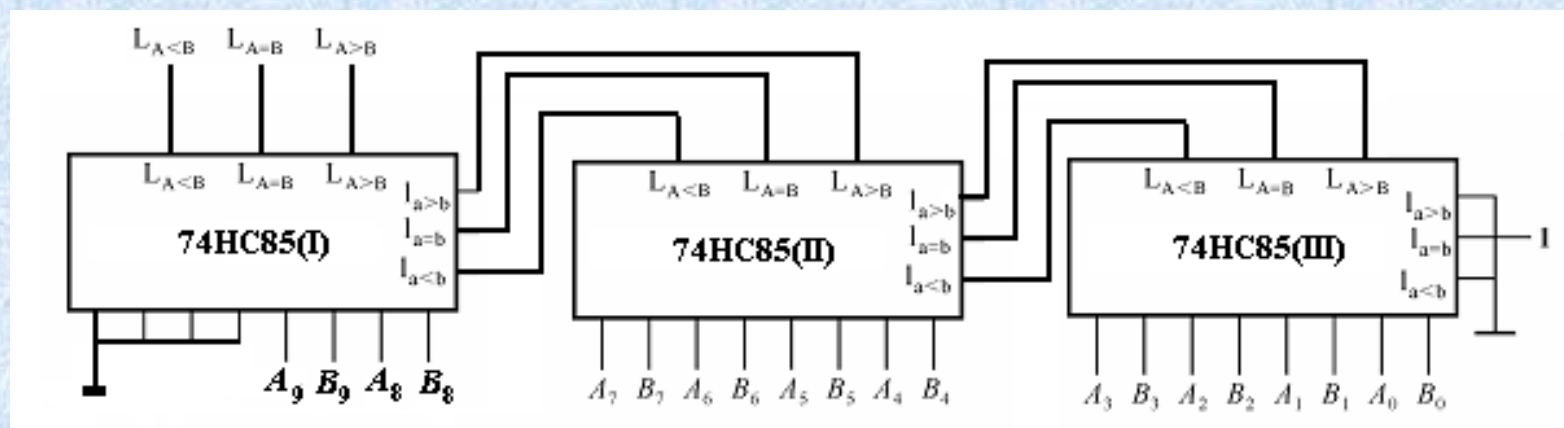
74HC85型中规模集成四位数值比较器的引脚排列图



用二片74HC85画出的一个8位数值比较器的电路连接



二个10位数值比较器的级连



4.2.4 中规模集成计数器

中规模集成计数器有总清零端、置数端、数据输入端、进借位输出端，扩展控制端等，利用这些端可以把中规模集成计数器连接成各种进制的计数器

对中规模集成计数器，主要能读懂真值表（功能表），引脚排列，就可以使用它。

一、典型的中规模集成计数器

1. 四位二进制加法计数器74HC163和74HC161

功能说明

集成计数器 74HC163 功能表

输入									触发器状态			
CP	\overline{CR}	\overline{LD}	CT_P	CT_T	D_3	D_2	D_1	D_0	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}	Q_0^{n+1}
↑	0	x	x	x	x	x	x	x	0	0	0	0
↑	1	0	x	x	A_3	A_2	A_1	A_0	A_3	A_2	A_1	A_0
↑	1	1	1	1	x	x	x	x	四位二进制加法计数			
↑	1	1	0	x	x	x	x	x	同步保持			
x	1	1	x	0	x	x	x	x	异步保持			

进位输出端

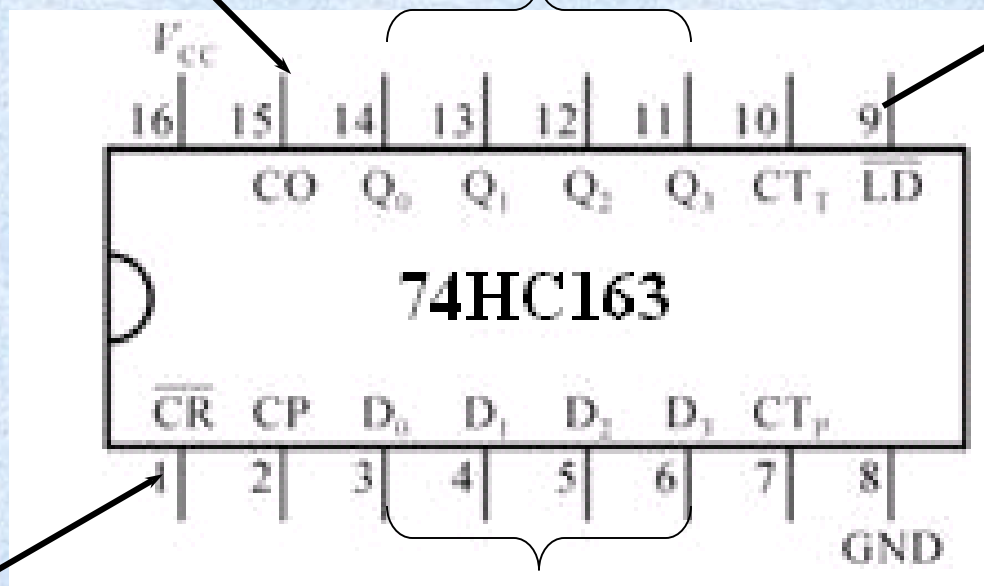
四位二进制数输出

同步置
数控制
端

CT_P 、 CT_T 用于
功能扩展控制

同步清零（总
复位端）

四位数据输入端



集成计数器74HC161引脚排列与74HC163完全相同，但它采用了“**异步清零**”和“**同步置数**”方式。

集成计数器 74HC161 功能表

输入									触发器状态			
CP	\overline{CR}	\overline{LD}	CT_P	CT_T	D_3	D_2	D_1	D_0	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}	Q_0^{n+1}
X	0	X	X	X	X	X	X	X	0	0	0	0
\uparrow	1	0	X	X	A_3	A_2	A_1	A_0	A_3	A_2	A_1	A_0
\uparrow	1	1	1	1	X	X	X	X	四位二进制加法计数			
\uparrow	1	1	0	X	X	X	X	X	保持			
X	1	1	X	0	X	X	X	X	不变			



2. 十进制可逆计数器

◆十进制可逆计数器74HC217（目前讲义上无）

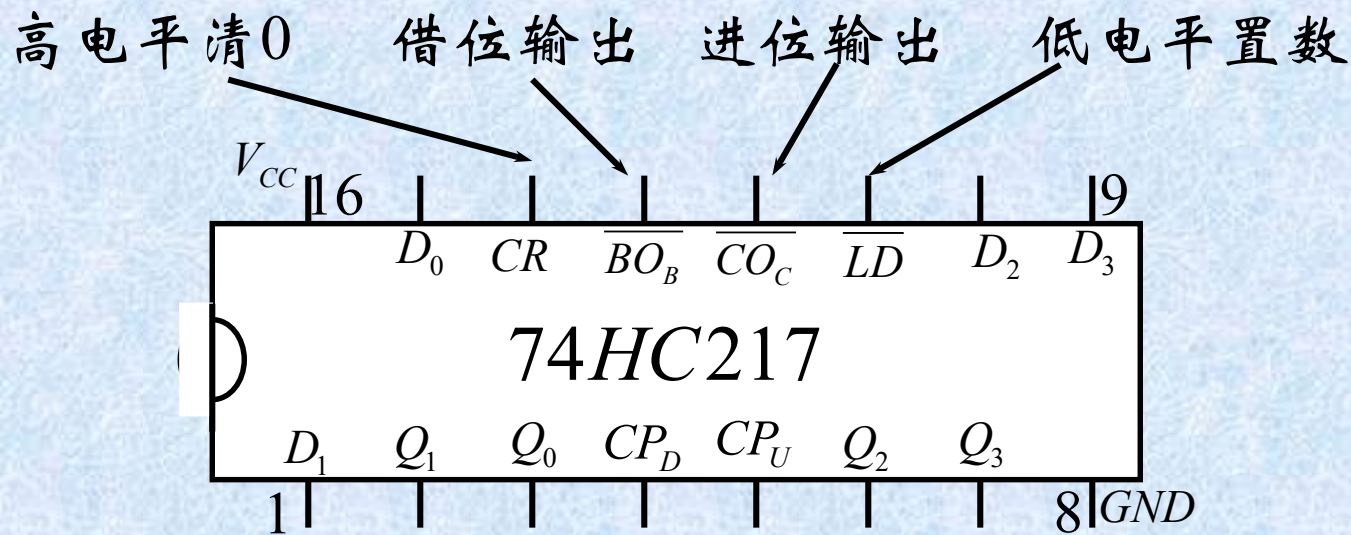
74HC217型 BCD码双时钟可逆计数器功能表

输 入								触发器状态				说明
CR	\overline{LD}	CP_U	CP_D	D_3	D_2	D_1	D_0	Q_3	Q_2	Q_1	Q_0	
1	×	×	×	×	×	×	×	0	0	0	0	异步清0
0	0	×	×	D	C	B	A	D	C	B	A	异步置数
0	1	↑	1	×	×	×	×	8421加计数				
0	1	1	↑	×	×	×	×	8421减计数				
0	1	1	1	×	×	×	×	保持不变				

双时钟触发 $\left\{ \begin{array}{l} \text{减法计数时钟输入} \rightarrow \overline{CP_-} (CP_D) \\ \text{加法计数时钟输入} \rightarrow CP_+ (CP_U) \end{array} \right.$

$$\overline{BO} = \overline{Q_3 Q_2 Q_1 Q_0 CP_D}$$

$$\overline{CO} = \overline{Q_3 Q_0 CP_U}$$



74HC217型BCD码双时钟可逆计数器引脚图

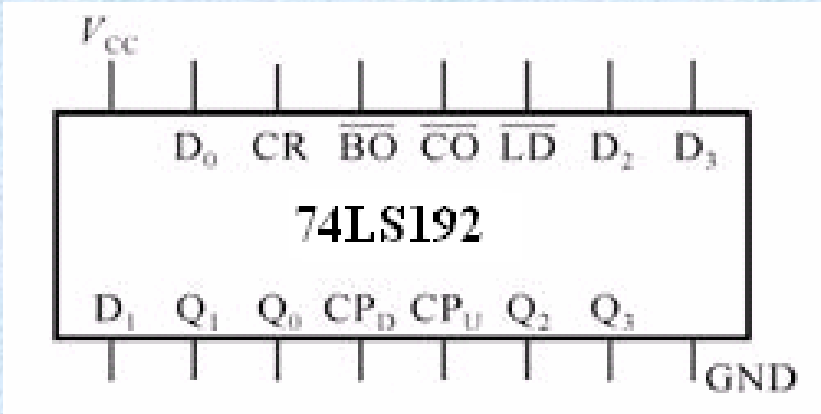
◆十进制可逆计数器74LS192

74LS192 功能表

输入								触发器状态			
CR	\overline{LD}	CP_U	CP_D	D_3	D_2	D_1	D_0	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}	Q_0^{n+1}
1	X	X	X	X	X	X	X	0	0	0	0
0	0	X	X	A_3	A_2	A_1	A_0	A_3	A_2	A_1	A_0
0	1	\uparrow	1	X	X	X	X	8421 十进制加法计数			
0	1	1	\uparrow	X	X	X	X	8421 十进制减法计数			
0	1	1	1	X	X	X	X	保持			

$$\overline{CO} = \overline{Q_3 Q_0 CP_U}$$

$$\overline{BO} = \overline{Q_3 Q_2 Q_1 Q_0 CP_D}$$

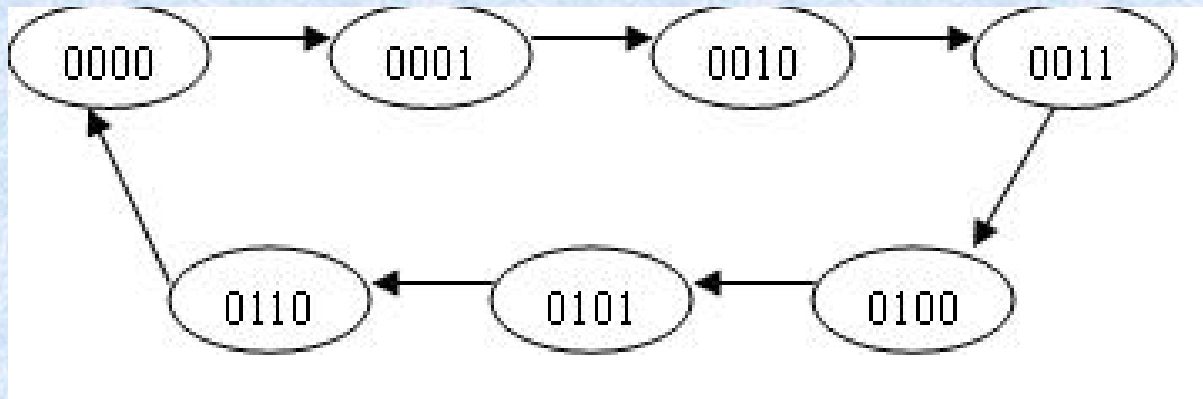


二、利用单片中规模集成计数器实现N进制计数

1. 用反馈清零法实现功能扩展

基本想法：在正常计数时，清零端 \overline{CR} 或 CR 应在高电平（或低电平），当计到某个数时（人为设定），清零端变低电平，然后又回到高电平，计数器重新开始计数

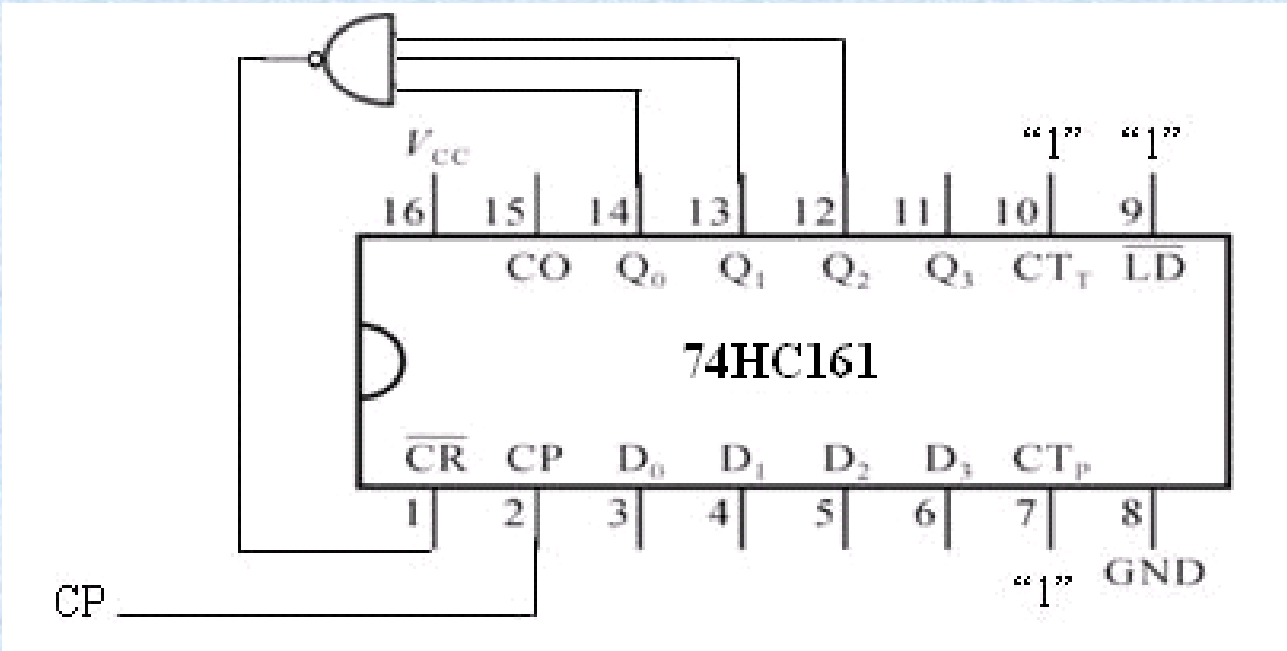
【例】 试分别用74HC161和74HC163型中规模集成计数器设计一个8421BCD编码的七进制加法计数器。



当状态“0110”出现后可采用“反馈清零”法将下一个状态强制置成“0000”

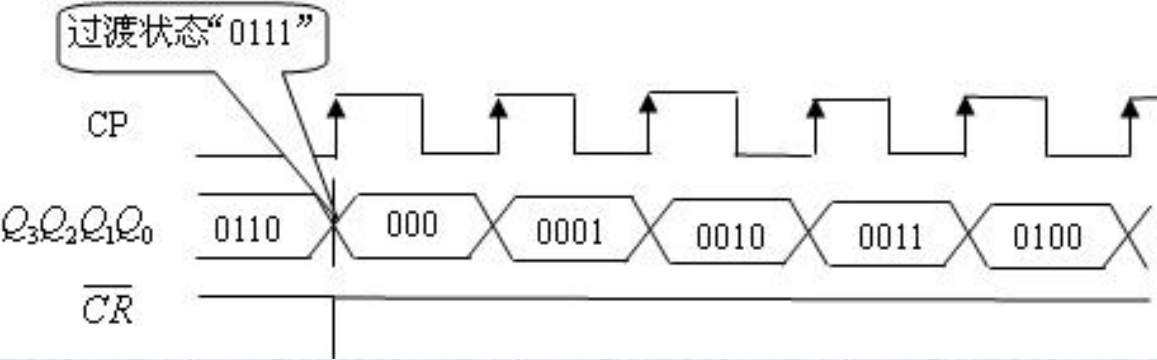
在**异步清零**时，必须借助0110的下一个状态0111，利用0111状态作为反馈清零控制

$$\overline{CR} = \overline{Q_2^n Q_1^n Q_0^n}$$

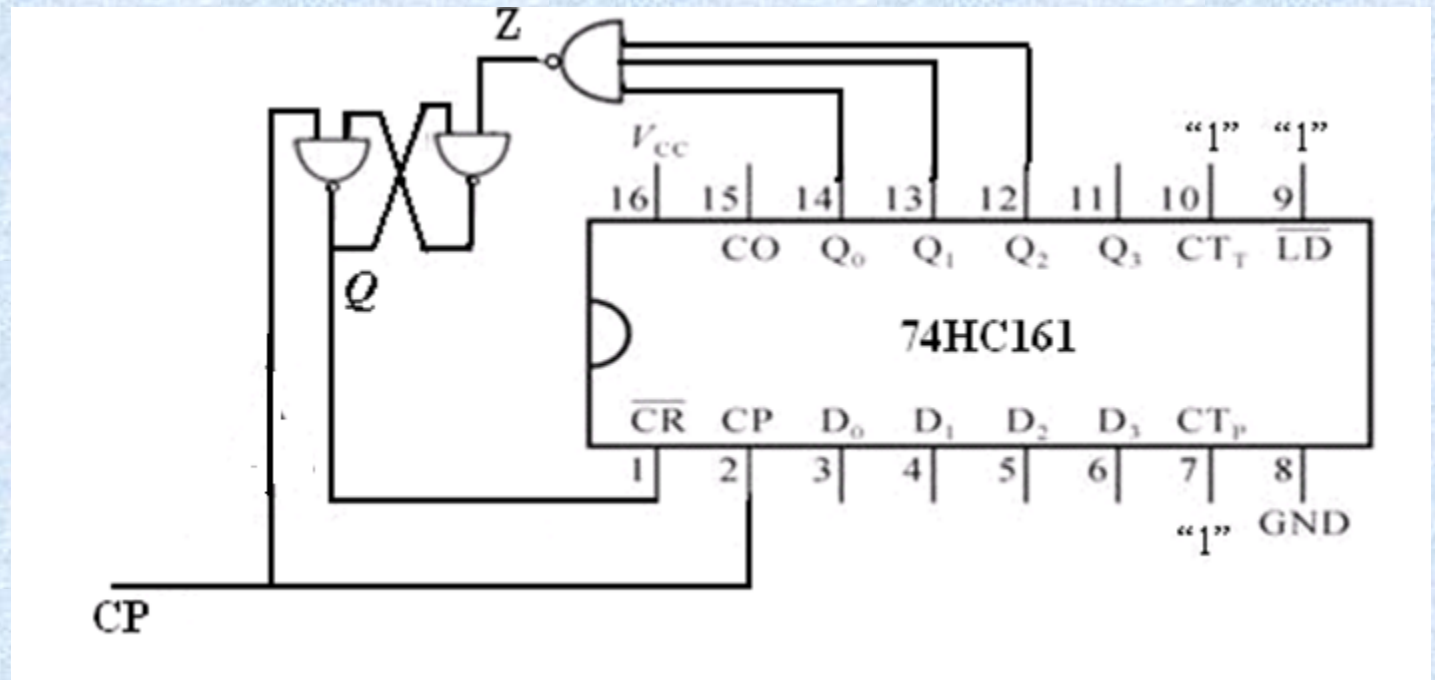


集成计数器 74HC161 功能表

输入									触发器状态			
CP	\overline{CR}	\overline{LD}	CT_P	CT_T	D_3	D_2	D_1	D_0	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}	Q_0^{n+1}
X	0	X	X	X	X	X	X	X	0	0	0	0
↑	1	0	X	X	A_3	A_2	A_1	A_0	A_3	A_2	A_1	A_0
↑	1	1	1	1	X	X	X	X	四位二进制加法计数			
↑	1	1	0	X	X	X	X	X	保持			
X	1	1	X	0	X	X	X	X	不变			



必须保证清零
脉冲有足够的
宽度



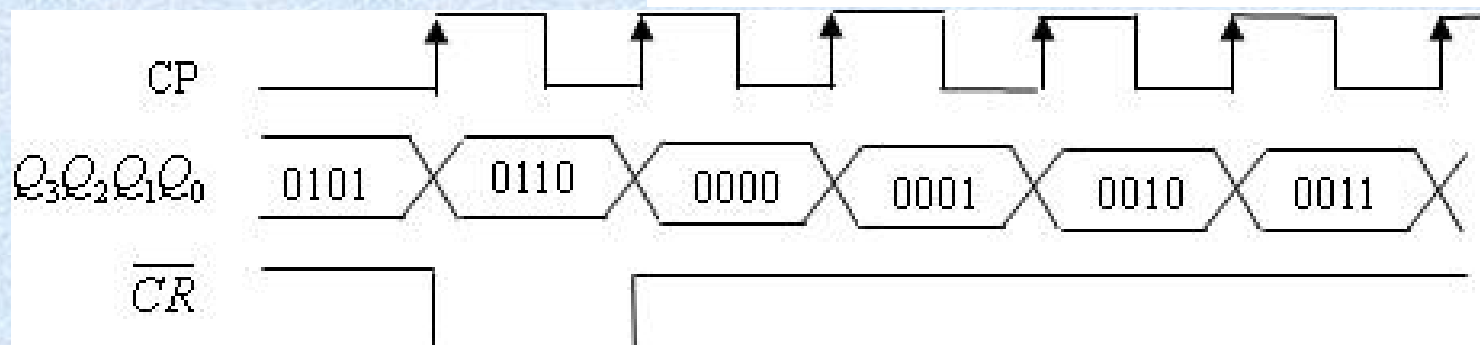
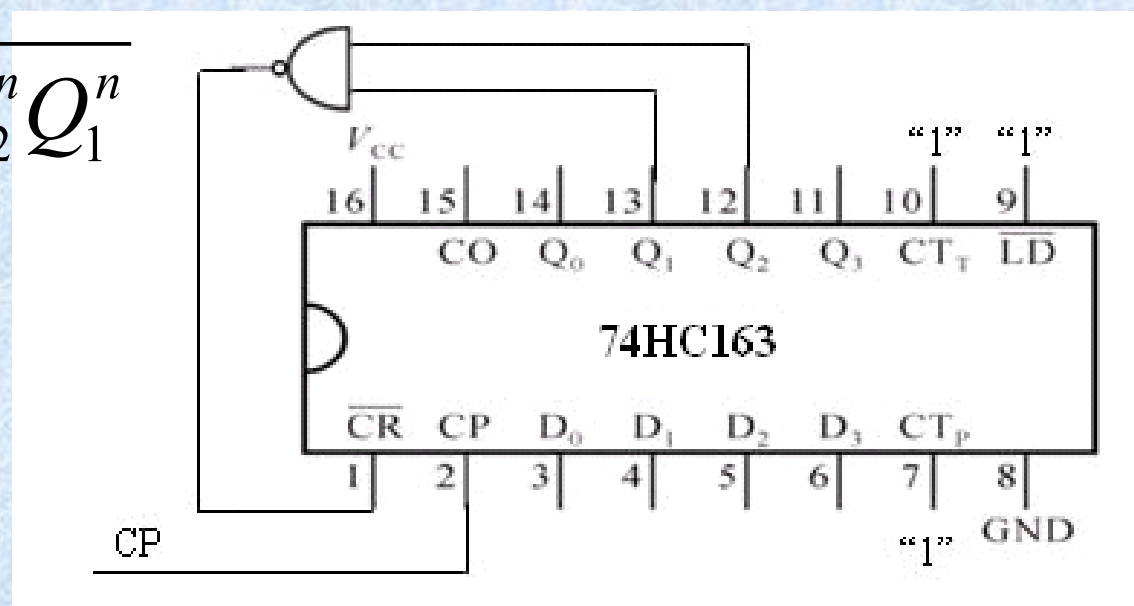
充分保证异步清零的有效性

集成计数器 74HC163 功能表

输入									触发器状态			
CP	\overline{CR}	\overline{LD}	CT_P	CT_T	D_3	D_2	D_1	D_0	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}	Q_0^{n+1}
↑	0	x	x	x	x	x	x	x	0	0	0	0
↑	1	0	x	x	A_3	A_2	A_1	A_0	A_3	A_2	A_1	A_0
↑	1	1	1	1	x	x	x	x	四位二进制加法计数			
↑	1	1	0	x	x	x	x	x	同步保持			
x	1	1	x	0	x	x	x	x	异步保持			

采用**同步清零**时，除了要求清零端 \overline{CR} 为低电平外，还要求有时钟CP。可以将最后一个状态“0110”作为反馈控制逻辑

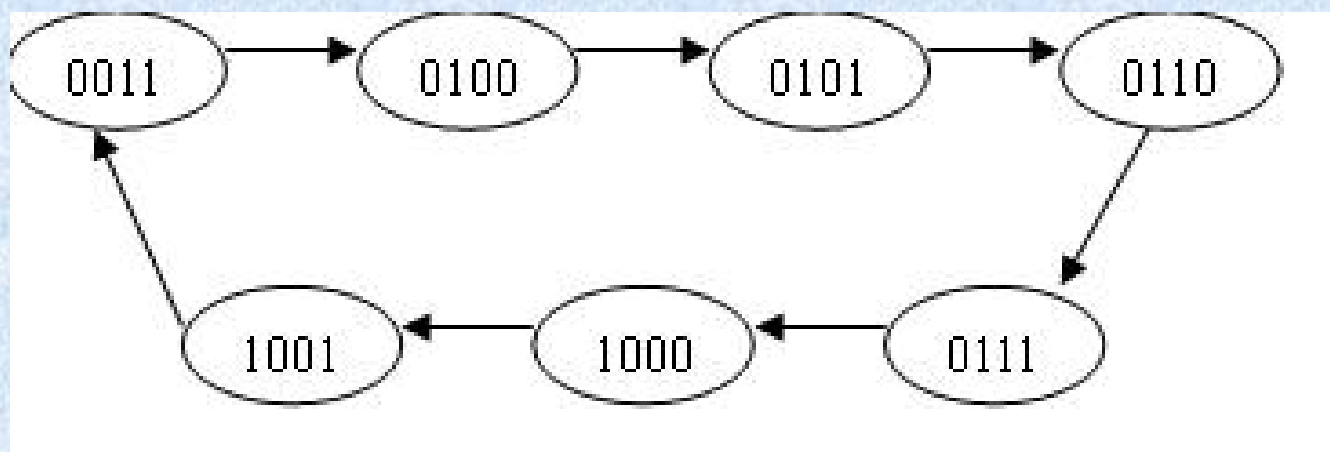
$$\overline{CR} = \overline{Q_2^n Q_1^n}$$



2. 用置数法实现功能扩展

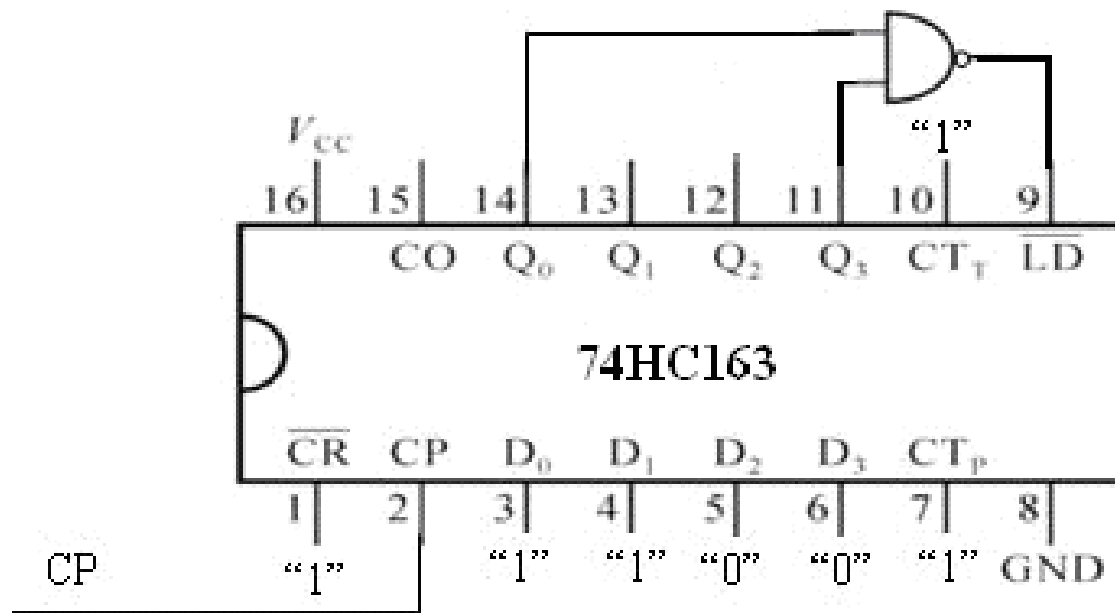
该方法的基本思路：计数器可以从0...0开始计数，也可从某一个数字开始计数，而0...0或某个数字可以从数据输入端预置入计数器，然后计数

【例】 试用中规模集成计数器74HC163设计一个余3编码的七进制加法计数器。



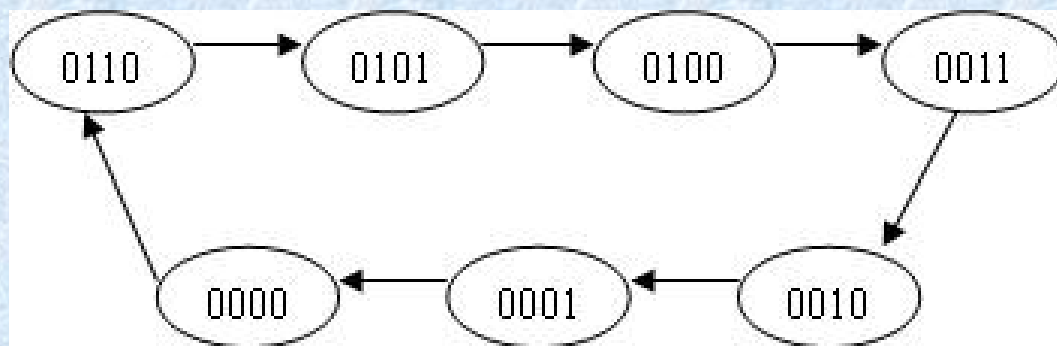
74HC163的置数控制属于**同步置数**

将状态转移图中最大状态“1001”作为置数控制逻辑，让计数器状态出现1001时，将计数器的状态置成0011



$$\overline{LD} = \overline{Q_3^n Q_0^n}$$

【例】 用中规模集成计数器74HC217设计一个8421BCD编码的七进制减法计数器。



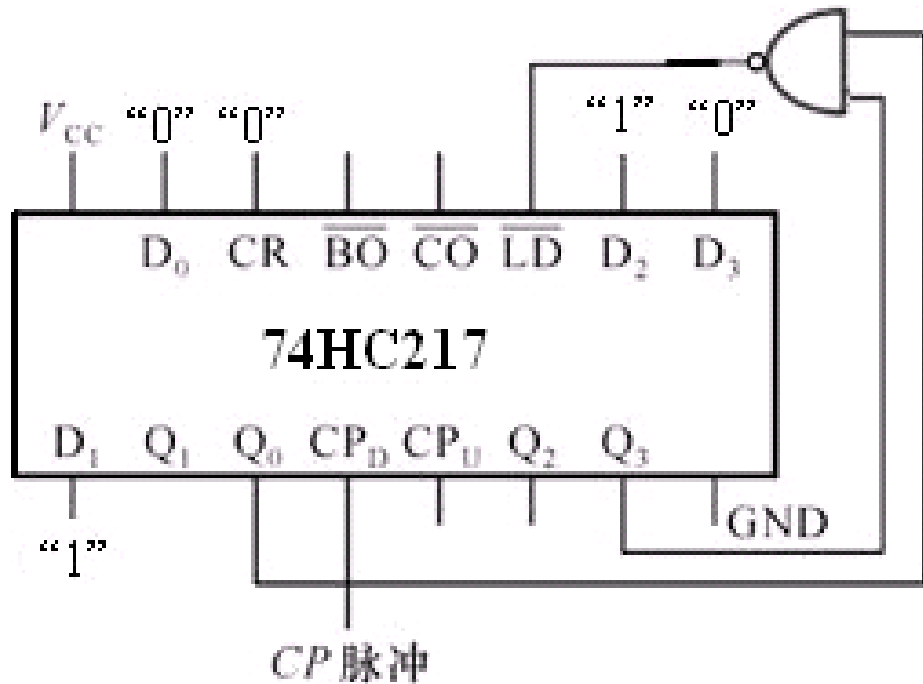
74HC217为异步方式置数

借助于“0000”的下一个状态“1001”（当74HC217连接成十进制减法计数功能时，

“0000”状态的下一个状态是“1001”）作为置数控制逻辑状态

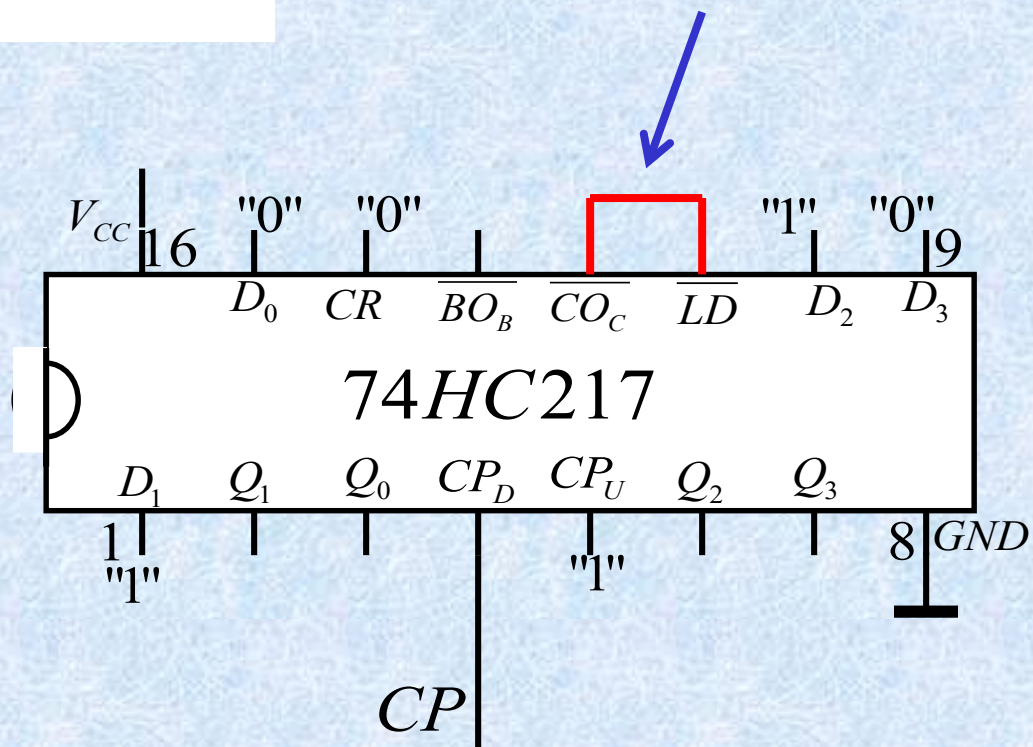
74HC217型 BCD码双时钟可逆计数器功能表

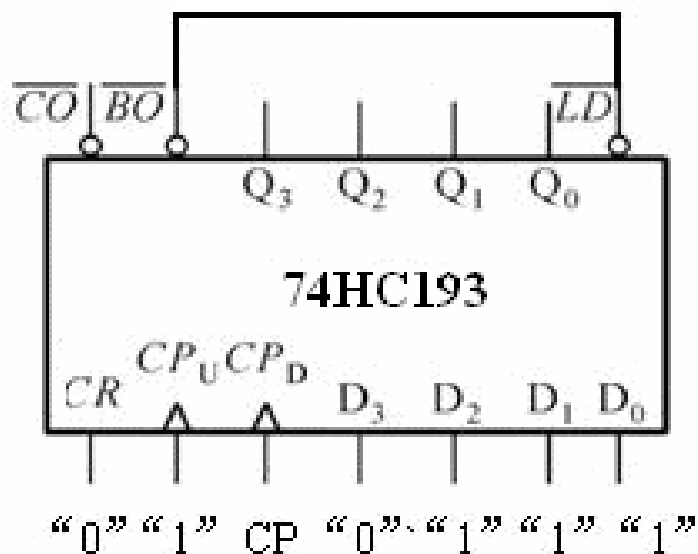
输 入								触发器状态				说明
CR	\overline{LD}	CP_U	CP_D	D_3	D_2	D_1	D_0	Q_3	Q_2	Q_1	Q_0	
1	×	×	×	×	×	×	×	0	0	0	0	异步清0
0	0	×	×	D	C	B	A	D	C	B	A	异步置数
0	1	↑	1	×	×	×	×	8421加计数				
0	1	1	↑	×	×	×	×	8421减计数				
0	1	1	1	×	×	×	×	保持不变				



$$\overline{LD} = \overline{Q_3 Q_0}$$

$$\overline{CO} = \overline{Q_3 Q_0 CP_U}$$

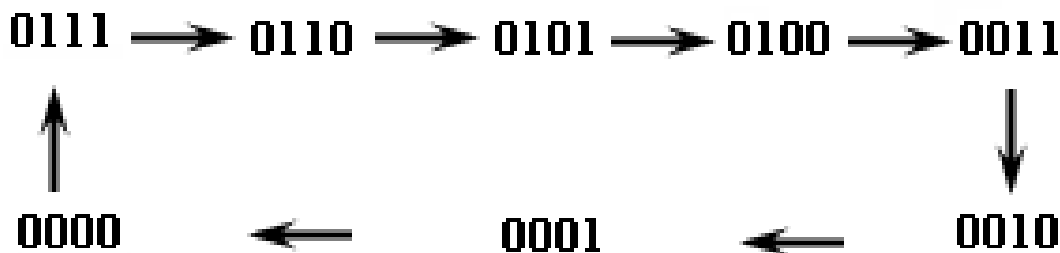




* 【题4.36 (b)】

该电路是连接成减法模式，
当有借位输出时，将数据输入端的数据置入计数器中，
计数器又重新开始新一轮计数，
状态转换图如图所示。

$$\overline{BO} = \overline{Q_3 Q_2 Q_1 Q_0 CP_D}$$



0000状态出现时是否就置数？ 虽然是异步置数方式
注意 \overline{BO} 还需要脉冲配合！
慎用！

74HC193功能表

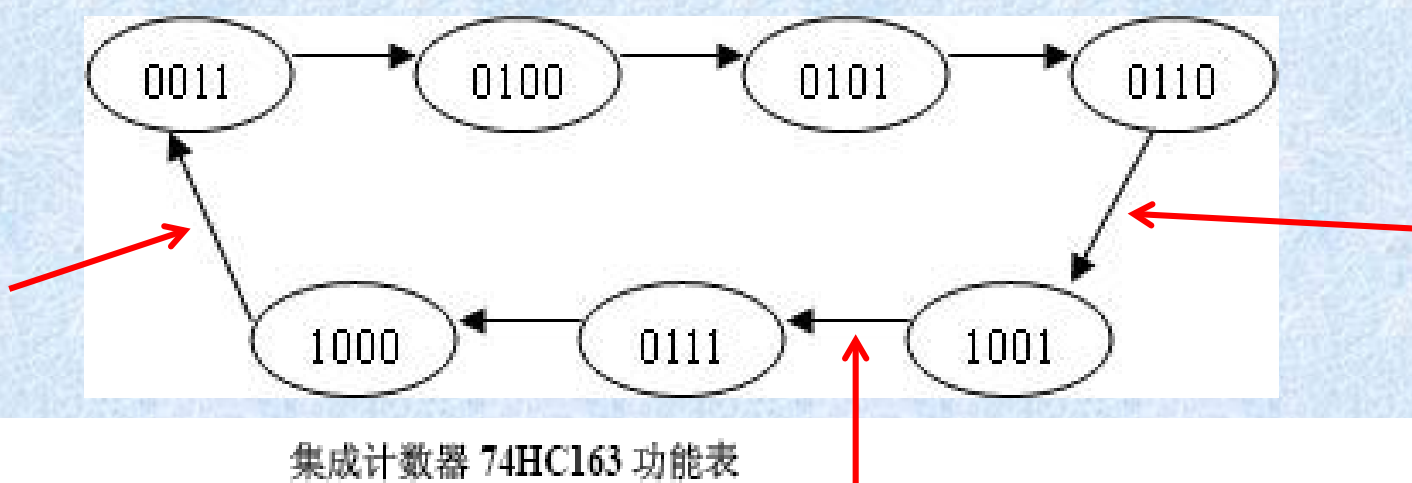
输 入								输 出			
CR	\overline{LD}	CP_U	CP_D	D	C	B	A	Q_3	Q_2	Q_1	Q_0
1	×	×	×	×	×	×	×	0	0	0	0
0	0	×	×	D_3	D_2	D_1	D_0	D_3	D_2	D_1	D_0
0	1	↑	1	×	×	×	×	四位二进制加法计数			
0	1	1	↑	×	×	×	×	四位二进制减法计数			

3. 多次反馈置数法

反馈清零法和反馈置数法局限于原计数器的计数状态以二进制规律连续变化。

如果状态转移不连续，则应采用置数和计数功能交替使用。状态不连续时采用置数法，在状态连续变化时采用计数法实现。

【例】 用中规模集成计数器74HC163设计一个如图所示状态转移图的七进制计数器。



集成计数器 74HC163 功能表

输入									触发器状态			
CP	\overline{CR}	\overline{LD}	CT_P	CT_T	D_3	D_2	D_1	D_0	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}	Q_0^{n+1}
↑	0	x	x	x	x	x	x	x	0	0	0	0
↑	1	0	x	x	A_3	A_2	A_1	A_0	A_3	A_2	A_1	A_0
↑	1	1	1	1	x	x	x	x	四位二进制加法计数			
↑	1	1	0	x	x	x	x	x	同步保持			
x	1	1	x	0	x	x	x	x	异步保持			

三、实现大容量计数器的连接

大容量是指：几十进制以上的计数器；

连接原则：用小容量计数器串联实现

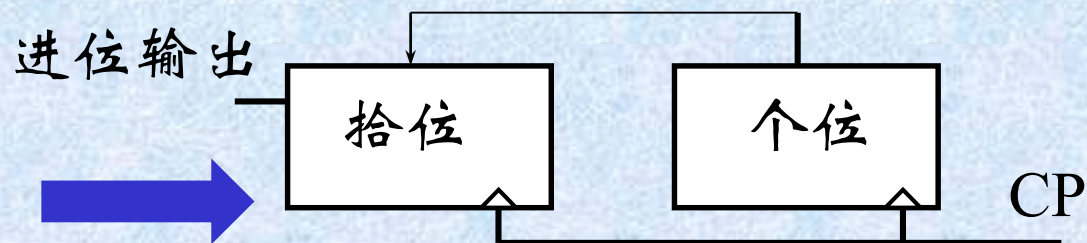
$$M(\text{大容量}) = M_1 \times M_2 \times \cdots$$

如一个60进制计数器可用一个6进和一个10进制计数器串联构成，即 $60 = 6 \times 10$ 。

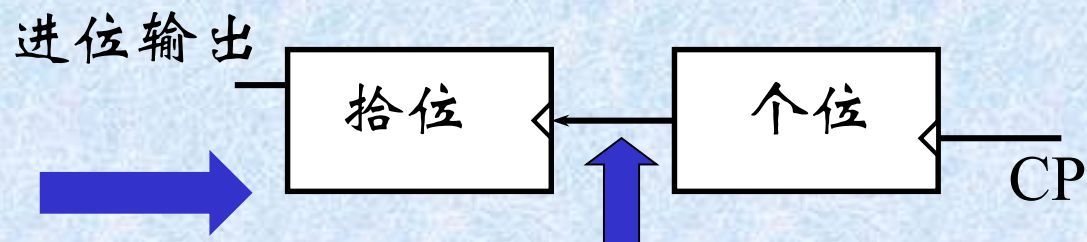
拾位计数器为6进制，个位计数器为10进制。100进制可用二个10进制计数器等。

连接方法：

同步方式



异步方式



个位进位输出作
拾位CP脉冲

【例】 试用中规模集成计数器74HC163设计一个8421BCD编码的60进制计数器。

60进制计数器必须分解成二级实现

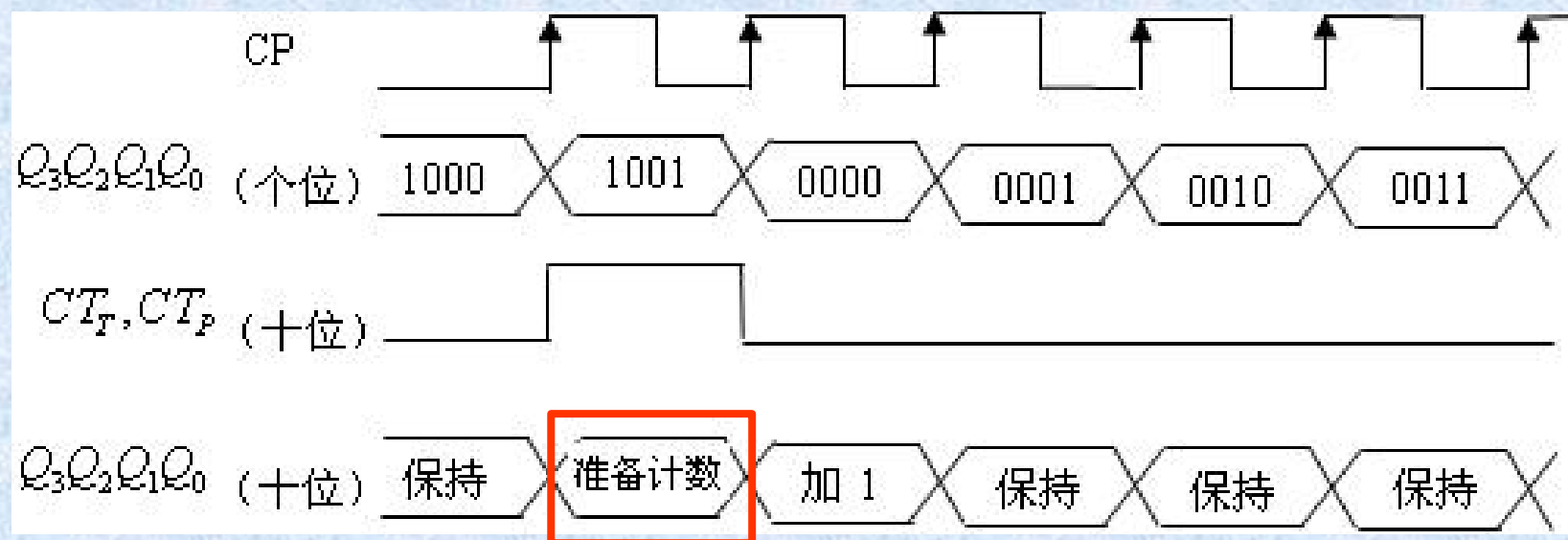
$$N = M_1 \times M_2 = 6 \times 10$$

拾位和个位各用一片集成计数器实现，本例中的个位10进制计数器采用反馈置数法实现，拾位6进制采用反馈清零法实现。

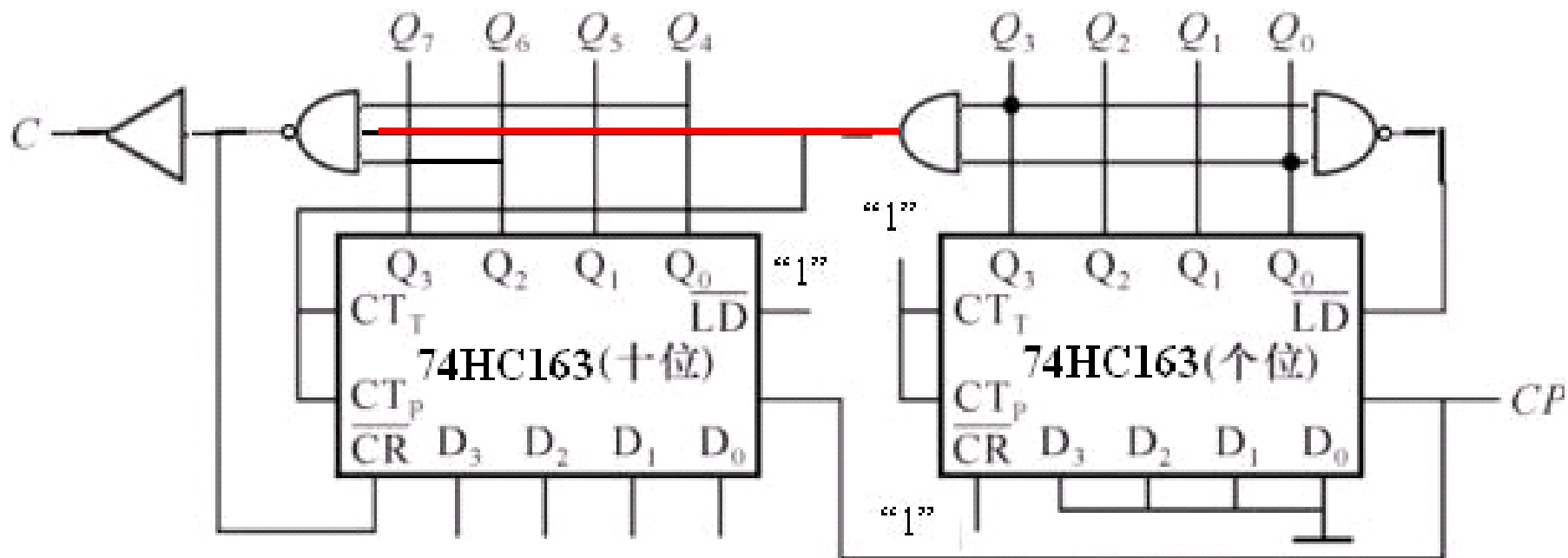
片间级联要求是：当个位计数器有进位输出时，拾位计数器加1。由于74HC163具有同步保持功能，因此片间级联可以采用同步级联方式。

集成计数器 74HC163 功能表

输入									触发器状态			
CP	\overline{CR}	\overline{LD}	CT_P	CT_T	D_3	D_2	D_1	D_0	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}	Q_0^{n+1}
\uparrow	0	x	x	x	x	x	x	x	0	0	0	0
\uparrow	1	0	x	x	A_3	A_2	A_1	A_0	A_3	A_2	A_1	A_0
\uparrow	1	1	1	1	x	x	x	x	四位二进制加法计数			
\uparrow	1	1	0	x	x	x	x	x	同步保持			
x	1	1	x	0	x	x	x	x	异步保持			



同步级联方式



当拾位计数器计到5、同时个位计数器计到9时，拾位的同步清零端 $\overline{CR}=0$ 作好了同步清零的准备。

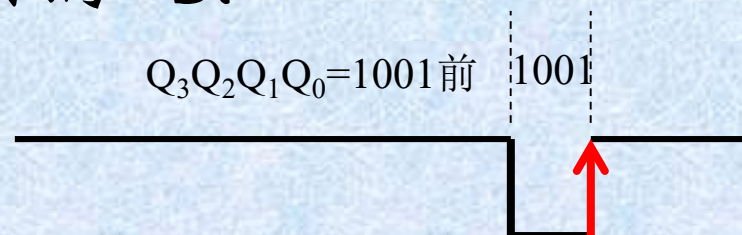
异步级联法

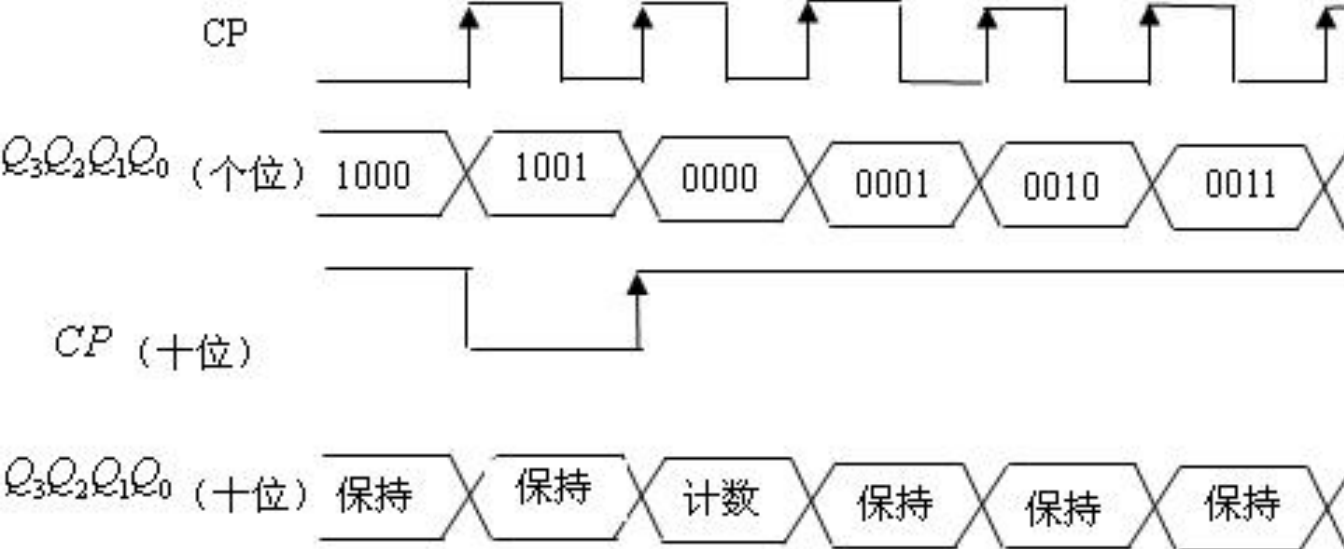
◆ 实用于某些不具有同步保持功能芯片

思路： 当个位计数产生进位输出时，该进位信号应该为拾位计数器提供一个计数时钟

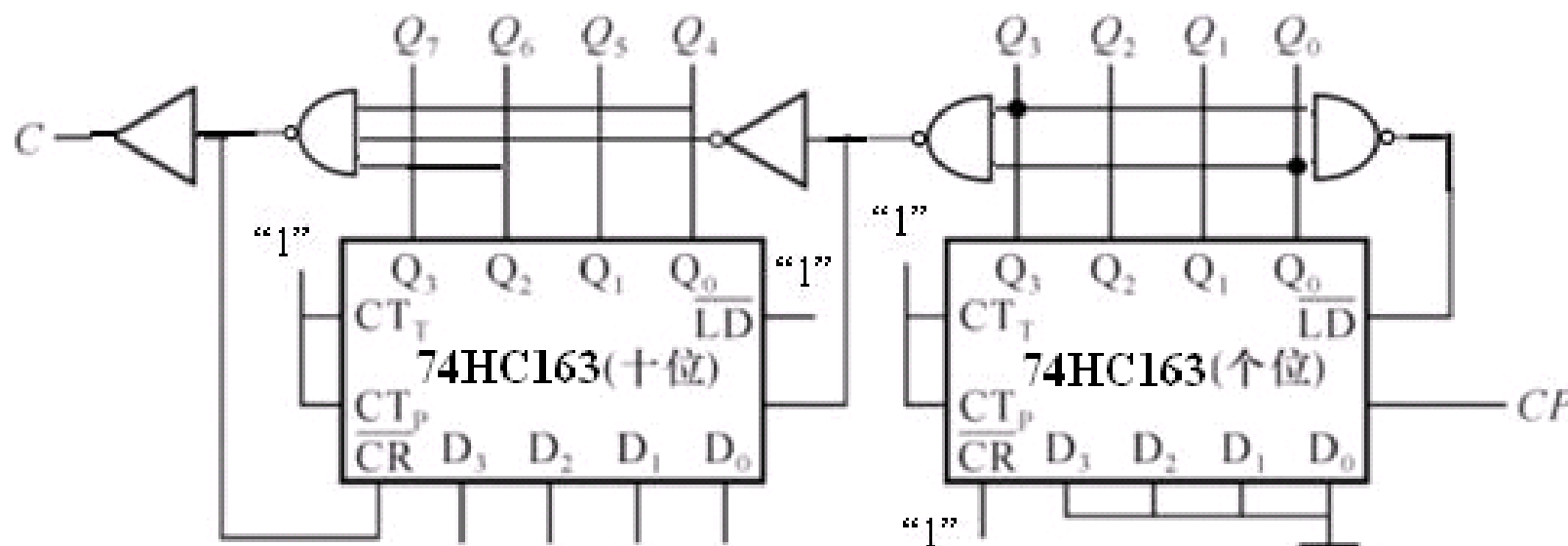
由于74HC163为时钟脉冲的上升沿触发，为保证在个位计数器由9变0时产生上升沿，拾位计数器的时钟方程必须满足：

$$CP_{\text{拾位}} = \overline{Q_3 Q_0}$$





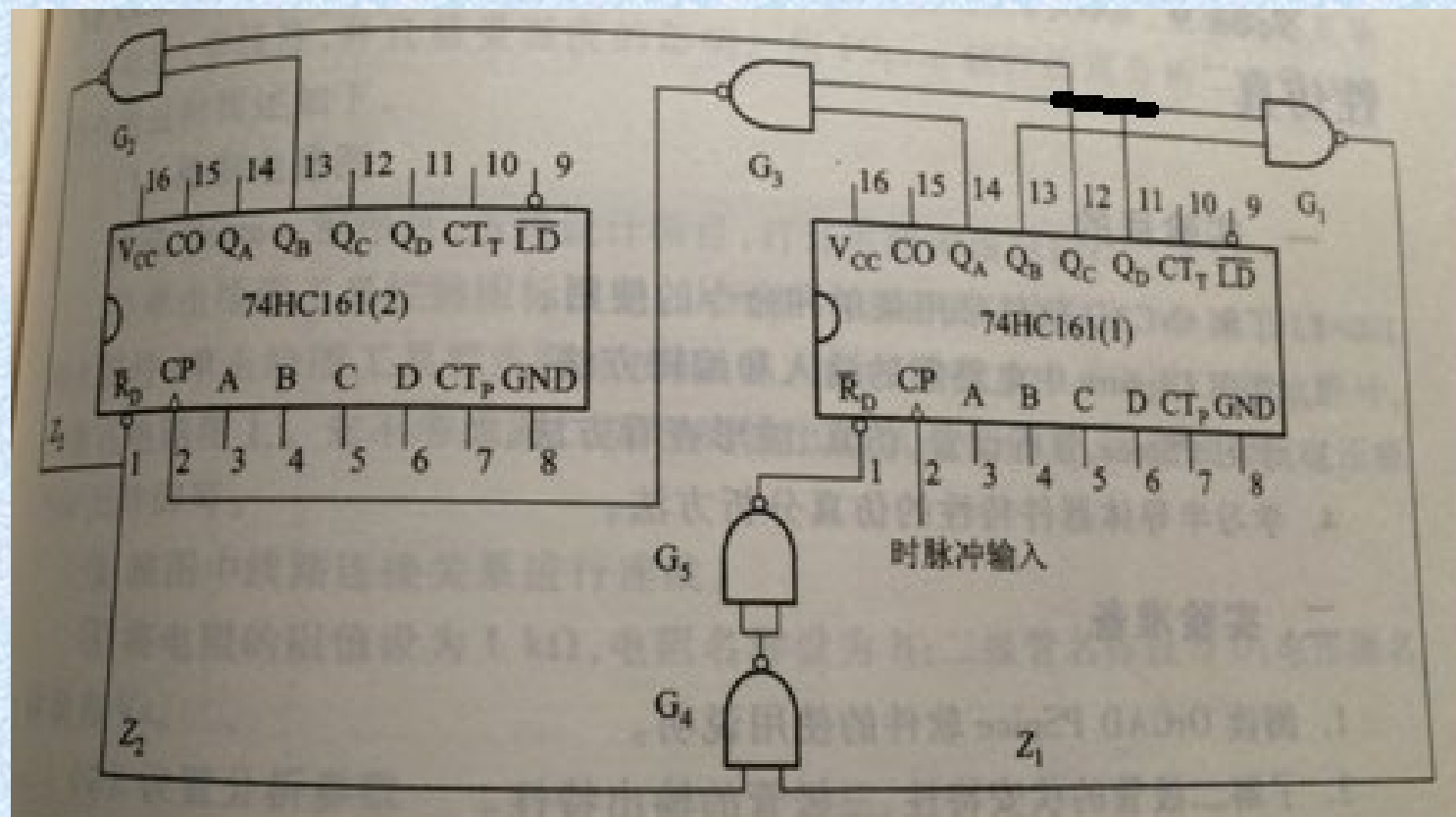
异步级联法



***24进制又该如何接法?**

集成计数器 74HC161 功能表

输入									触发器状态			
CP	\overline{CR}	\overline{LD}	CT_P	CT_T	D_3	D_2	D_1	D_0	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}	Q_0^{n+1}
X	0	X	X	X	X	X	X	X	0	0	0	0
\uparrow	1	0	X	X	A_3	A_2	A_1	A_0	A_3	A_2	A_1	A_0
\uparrow	1	1	1	1	X	X	X	X	四位二进制加法计数			
\uparrow	1	1	0	X	X	X	X	X	保持			
X	1	1	X	0	X	X	X	X	不变			



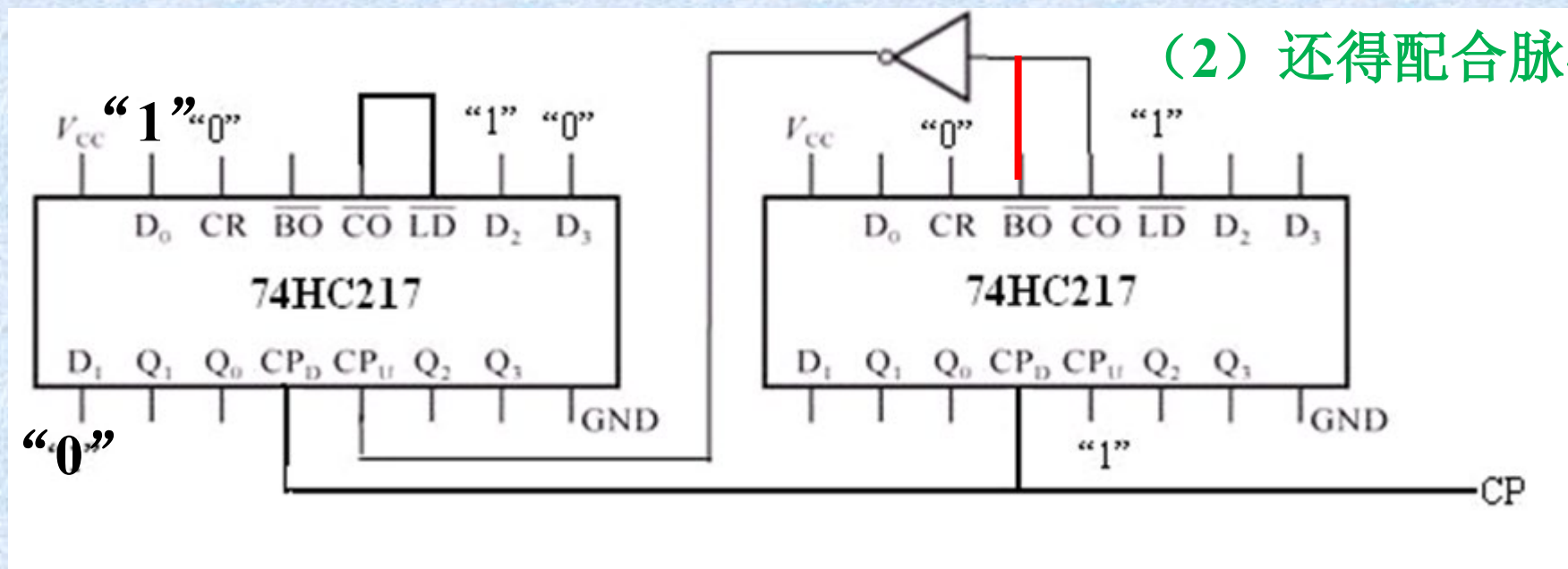
**** 【例】 用中规模集成计数器74HC217设计8421BCD编码的60进制减法计数器。**

$$\overline{CO} = \overline{Q_3 Q_0} CP_U$$

$$\overline{BO} = \overline{Q_3 Q_2 Q_1 Q_0} CP_D$$

(1) 有问题?? 应该从BO引出

（2）还得配合脉冲？

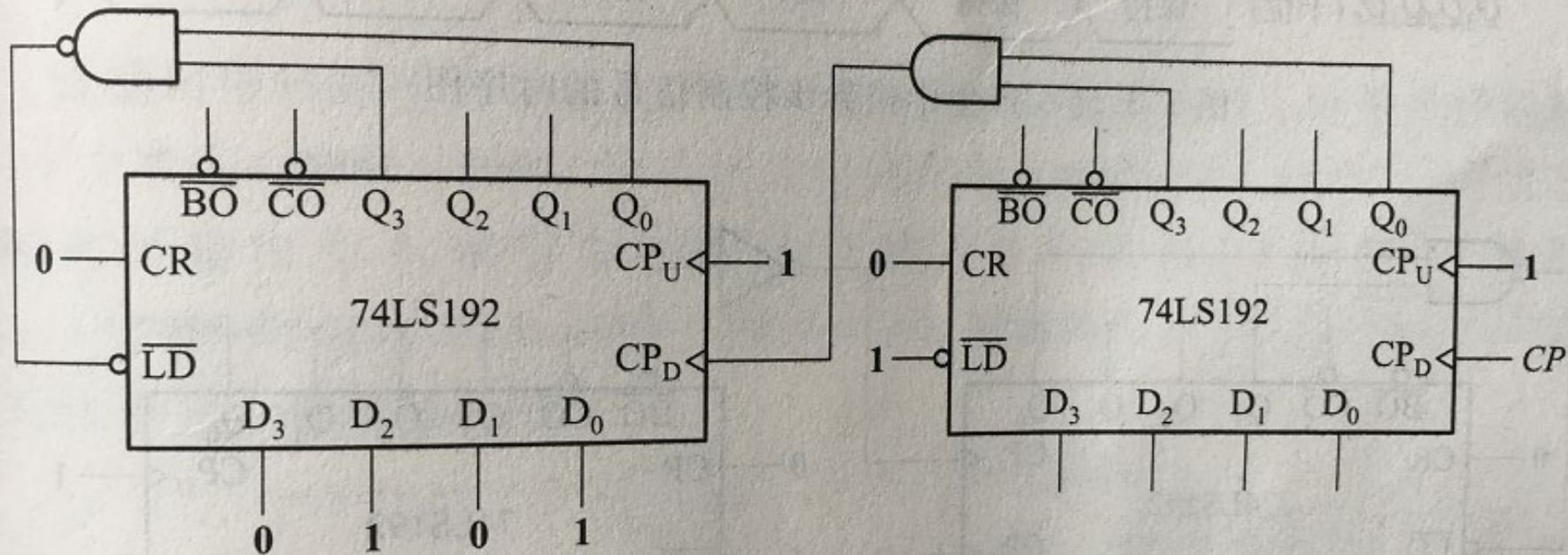


异步清0、异步置数

(3) 还不能考虑到门的延迟, 否则 CP_D 为“1”在 CP_U 一个上升沿, 则一个加法计数。

74HC217型 BCD码双时钟可逆计数器功能表

输 入								触发器状态				说明
CR	\overline{LD}	CP_U	CP_D	D_3	D_2	D_1	D_0	Q_3	Q_2	Q_1	Q_0	
1	×	×	×	×	×	×	×	0	0	0	0	异步清0
0	0	×	×	D	C	B	A	D	C	B	A	异步置数
0	1	↑	1	×	×	×	×	8421加计数				
0	1	1	↑	×	×	×	×	8421减计数				
0	1	1	1	×	×	×	×	保持不变				



74LS192 功能表

输入								触发器状态			
CR	\overline{LD}	CP_U	CP_D	D_3	D_2	D_1	D_0	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}	Q_0^{n+1}
1	X	X	X	X	X	X	X	0	0	0	0
0	0	X	X	A_3	A_2	A_1	A_0	A_3	A_2	A_1	A_0
0	1	\uparrow	1	X	X	X	X	8421 十进制加法计数			
0	1	1	\uparrow	X	X	X	X	8421 十进制减法计数			
0	1	1	1	X	X	X	X	保持			

◆ 这类芯片建议还是用异步级联的方式

四、应用中规模集成计数器实现一般时序逻辑电路

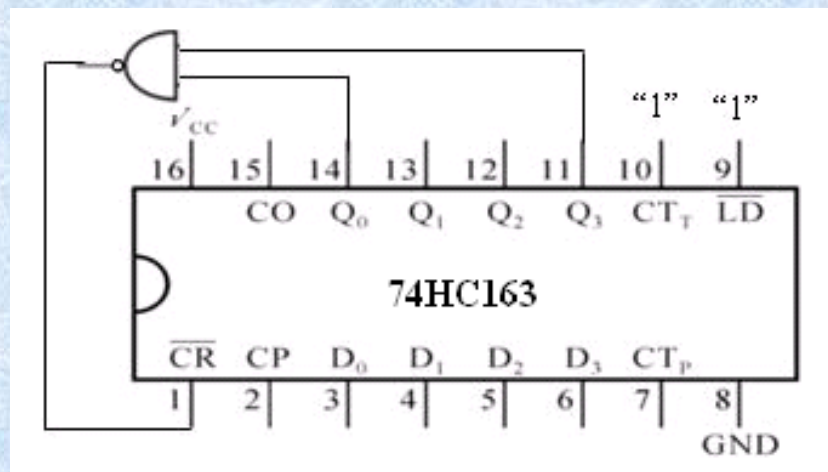
【例】试用74HC163设计一个“1100110001”序列脉冲发生器。

“1100110001”序列共有十个状态。设计过程可分成两步：

第一步利用中规模集成计数器设计一个十进制计数器；

第二步对计数器的状态进行译码，产生“1100110001”序列。

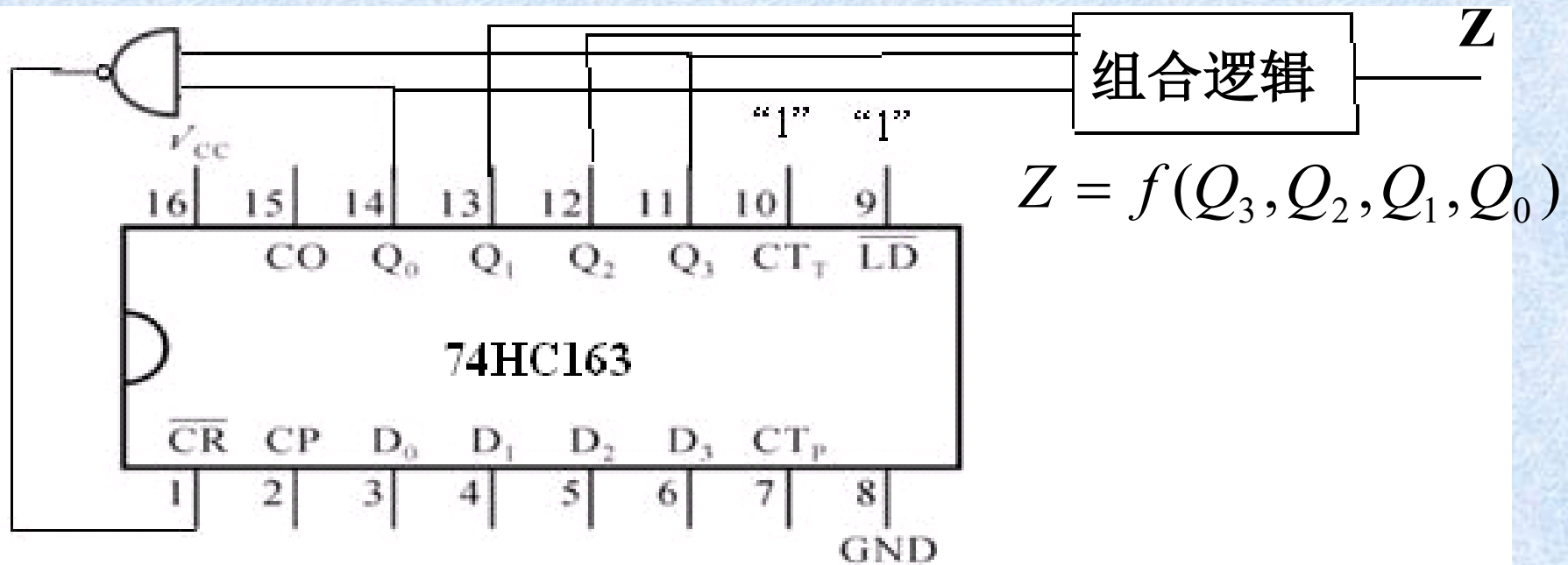
(1)反馈清零法设计的十进制计数器



(2)设输出序列脉冲信号为Z，Z由中规模集成计数器的状态变量经译码后得到，十个状态（0000~1001）对应序列信号“1100110001”的十位

		Q_3Q_2			
		00	01	11	10
Q_1Q_0	00	1	1	0	0
	01	1	1	0	0
	11	X	X	X	X
	10	0	1	X	X

$$Z = f(Q_3, Q_2, Q_1, Q_0)$$



第4章 数字逻辑电路

- 4.1 逻辑电路的分析与设计
- 4.2 中规模集成逻辑电路及应用
- 4.3 硬件描述语言和可编程逻辑器件