

实验项目

实验一 门电路和全加器实验	2
实验二 组合逻辑电路分析与设计	1
实验三 数值比较器/超前进位加法器的应用	3
实验四 触发器及其应用	3
实验五 时序电路测试及研究	8
实验六 集成计数器芯片的应用	13
实验七 组合电路的 VERILOG 设计和仿真	18
实验八 时序电路的 VERILOG 设计和仿真	22

实验一 门电路和全加器实验

一、实验目的

1. 熟悉门电路的逻辑功能、逻辑表达式、逻辑符号、等效逻辑图。
2. 掌握数字电路实验箱及示波器的使用方法。
3. 掌握组合逻辑电路的功能测试。
4. 验证半加器和全加器的逻辑功能。
5. 学会二进制数的运算规律。

二、实验仪器及材料

1. 仪器设备：双踪示波器、数字万用表、数字电路实验箱

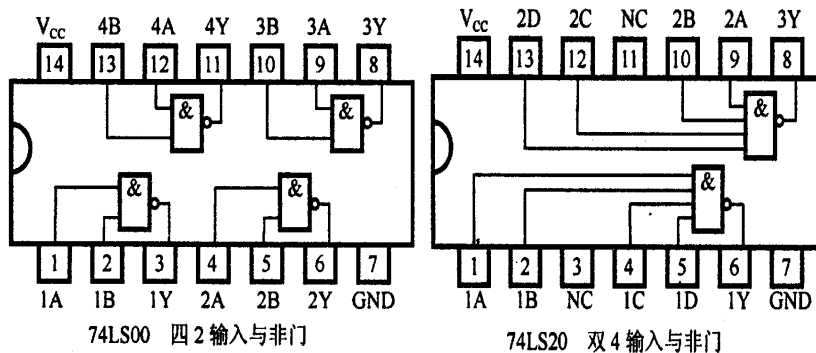
2. 器件：

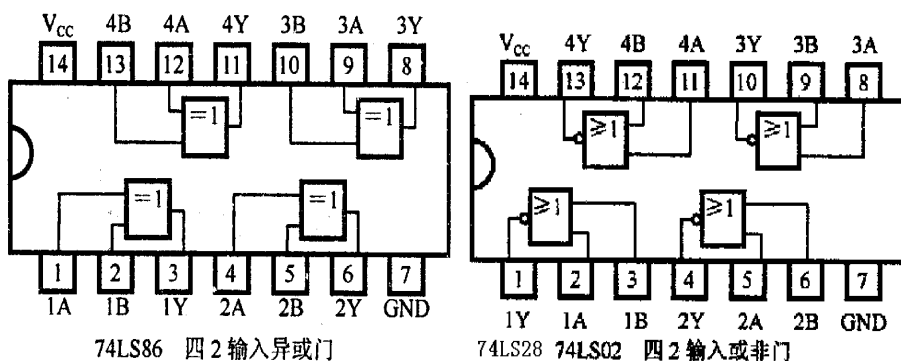
74LS00 二输入端四与非门 3 片

74LS20 四输入端双与非门 1 片

74LS86 二输入端四异或门 1 片

74LS02 二输入端四或非门 1 片（选做）





三、预习要求

1. 预习门电路相应的逻辑表达式。
2. 熟悉所用集成电路的引脚排列及用途。
3. 预习组合逻辑电路的分析方法。
4. 预习用与非门和异或门构成的半加器、全加器的工作原理。
5. 预习二进制数的运算。

四、实验内容及步骤

实验前按数字电路实验箱使用说明书先检查电源是否正常，然后选择实验用的集成块芯片插入实验箱中对应的 IC 座，按自己设计的实验接线图接好连线。注意集成块芯片不能插反。实验中改动接线须先断开电源，接好线后再通电实验。每个芯片的电源和 GND 引脚，分别和实验台的+5V 和“地 (GND)”连接。

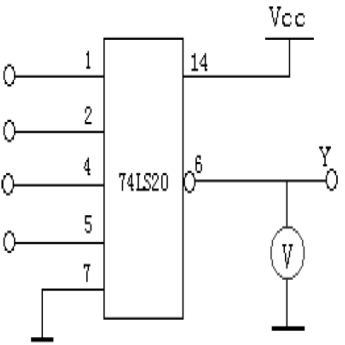
芯片不给它供电，芯片是不工作的。用实验台的逻辑开关作为被测器件的输入。拨动开关，则改变器件的输入电平。开关向上，输入为 1，开关向下，输入为 0。

将被测器件的输出引脚与实验台上的电平指示灯连接。指示灯亮表示输出电平为 1，指示灯灭表示输出电平为 0。

1. 与非门电路逻辑功能的测试

(1) 选用双四输入与非门 74LS20 一片，插入数字电路实验箱中对应的 IC 座，按图 1.1 接线、输入端 1、2、4、5、分别接到 $K_1 \sim K_4$ 的逻辑开关输出插口，输出端接电平显示发光二极管 $D_1 \sim D_4$ 中任意一个。注意：芯片 74LS20 的 14 号引脚要接试验箱下方的 +5V 电源，7 号引脚要接试验箱下方的地 (GND)。用万用表测电压时，万用表要调到直流 20V 档位，因为芯片接的电源是直流 +5V。

表 1.1



输入			输出		
1 (k1)	2 (k2)	4 (k3)	5 (k4)	Y	电压值
H	H	H	H		
L	H	H	H		
L	L	H	H		
L	L	L	H		
L	L	L	L		

图 1.1

(2) 按表 1.1 的状态改变逻辑开关的值，分别测出逻辑状态及输出电压。

2. 异或门逻辑功能的测试

(1) 选二输入四异或门电路 74LS86，按图 1.2 接线，输入端 1、2、4、5 接逻辑开关 ($K_1 \sim K_4$)，输出端 A、B、Y 接电平显示发光二极管。注意：芯片 74LS86 的 14 号引脚要接试验箱下方的 +5V 电源，7 号引脚要接试验箱下方的地 (GND)。

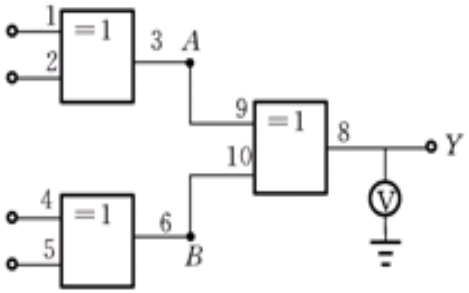


图 1.2

(2) 按表 1.2 的状态，拨动逻辑开关，将输出值的结果填入表中。

表 1.2

输入				输出			
1(K1)	2(K2)	4(K3)	5(K4)	A	B	Y	电压 (V)
L	L	L	L				
H	L	L	L				
H	H	L	L				
H	H	H	L				
H	H	H	H				
L	H	L	H				

3. 组合逻辑电路功能测试

(1) 用 2 片 74LS00 芯片组成图 1.3 所示逻辑电路。其中虚线框内是第一片 74LS00 芯片部分，虚线框外的是第二片 74LS00 芯片部分。注意：两片 74LS00 芯片的 14 号引脚都要接实验箱下方的+5V 电源，7 号引脚都要接实验箱下方的地 (GND)。为便于接线和检查，在图中要注明芯片编号及各引脚对应的编号。

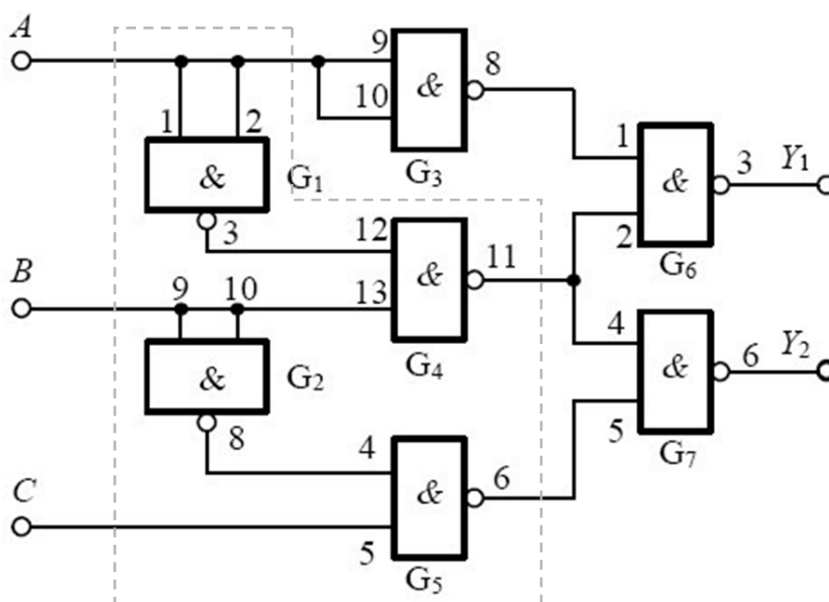


图 1.3

(2) 先按图 1.3 写出 Y2 的逻辑表达式并化简。

(3) 图中 A、B、C 接实验箱下方的逻辑开关，Y1，Y2 接实验箱上方的电平显示发光管。

(4) 按表 1.3 要求，拨动开关，改变 A、B、C 输入的状态，填表写出 Y1，Y2 的输出状态。

表 1.3

输入			输出	
A	B	C	Y1	Y2
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

(5) 将运算结果与实验结果进行比较。

4. 用异或门（74LS86）和与非门组成的半加器电路

根据半加器的逻辑表达式可知，半加器 Y 是 A、B 的异或，而进位 Z 是 A、B 相与，即半加器可用一个异或门和二一个与非门组成一个电路。如图 1.4。

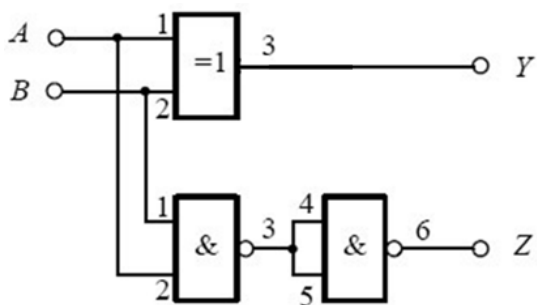


图 1.4

(1) 在数字电路实验箱上插入异或门和与非门芯片。输入端 A、B 接逻辑开关，

Y、Z 接电平显示发光管。

(2) 按表 1.4 要求，拨动开关，改变 A、B 输入的状态，填表写出 Y、Z 的输出状态，并根据真值表写出 Y、Z 逻辑表达式。

表 1.4

输入		输出	
A	B	Y	Z
0	0		
0	1		
1	0		
1	1		

5. 全加器组合电路的逻辑功能测试(选做)

- 写出图 1.5 电路的逻辑表达式。
- 根据逻辑表达式列真值表。
- 根据真值表画出逻辑输出函数 S_i 和 C_i 的卡诺图。

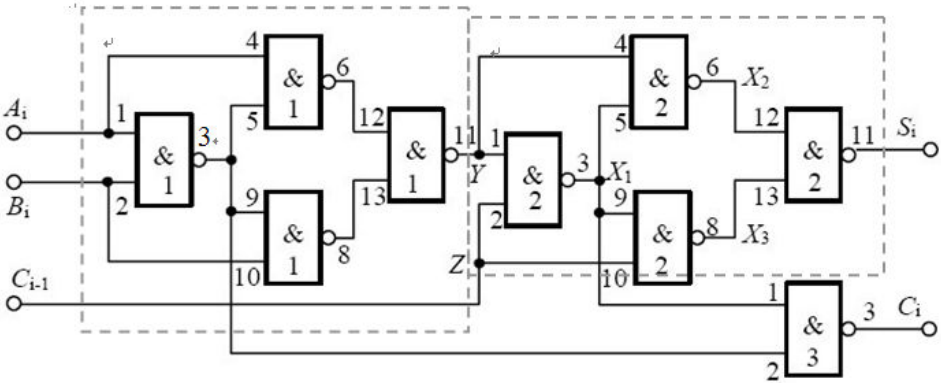


图 1.5

A_i	$B_i \ C_{i-1}$			
	00	01	11	10
0				
1				

$S_i =$

A_i	$B_i \ C_{i-1}$			
	00	01	11	10
0				
1				

$C_i =$

(4) 根据表达式直接填写表 1.5 各点状态

表 1.5

A_i	B_i	C_{i-1}	Y	Z	X_1	X_2	X_3	S_i	C_i
0	0	0							
0	0	1							
0	1	0							
0	1	1							
1	0	0							
1	0	1							
1	1	0							
1	1	1							

(5) 用 3 片 74LS00 芯片组成图 1.6 所示逻辑电路。其中第一个虚线框内是第一片 74LS00 芯片部分，第二个虚线框内的是第二片 74LS00 芯片部分。余下的是第三片 74LS00 芯片部分。注意：三片 74LS00 芯片的 14 号引脚都要接实验箱下方的+5V 电源，7 号引脚都要接实验箱下方的地（GND）。按原理图选择与非门并接线进行测试，将测试结果记入表 1.6，并与上表 1.5 进行比较看逻辑功能是否一致。

表 1.6

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

6. 逻辑电路的逻辑关系测试(选做)

(1)图 1.7 中需要用到两片 74LS00 芯片。图 1.8 中也需要用到两片 74LS00 芯片。其中虚线框内是第一片 74LS00 芯片部分，虚线框外的是第二片 74LS00 芯片部分。用两片 74LS00 芯片，分别按图 1.7，1.8 接线，将输入输出逻辑关系分别填入表 1.7、表 1.8 中。**A、B 接实验箱下方的逻辑开关。输出 Y 接实验箱上方的电平显示发光二极管。** 注意：两片 74LS00 芯片的 14 号引脚都要接实验箱下方的+5V 电源，7 号引脚都要接实验箱下方的地（GND）。

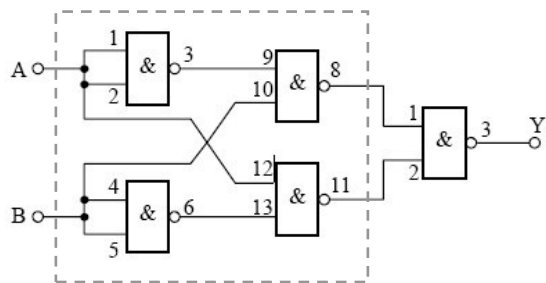


图 1.7

表 1.7

输入		输出
A	B	Y
L	L	
L	H	
H	L	
H	H	

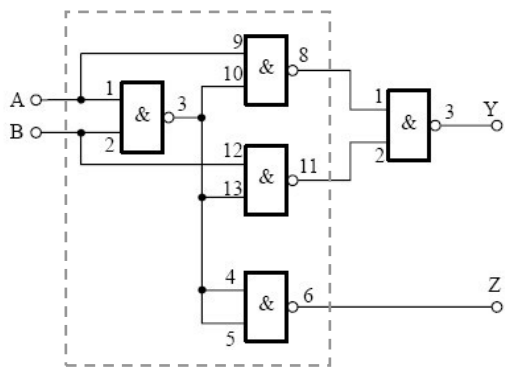


图 1.8

表 1.8

输入		输出	
A	B	Y	Z
L	L		
L	H		
H	L		
H	H		

(2) 写出上面两个电路对应的输出量的逻辑表达式，并画出等效逻辑图。

7. 利用与非门控制输出(选做)

(1) 用一片 74LS00 按图 1.9 接线，1 号引脚接实验箱左下方的**脉冲源输出孔**，将**频率档位调到 1KHZ 位置**。2 号引脚接一逻辑开关，将 Y 输出（3 号引脚）送到示波器上，用示波器观察 Y 输出脉冲的波形。理解**S 对 Y 的控制作用**。记录下，当 S=1 时，波形的特点是：_____。

当 $S=0$ 时，波形的特点是：_____。

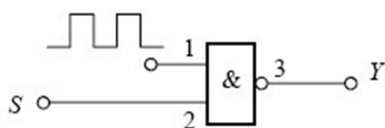


图 1.9

注意：芯片 74LS00 的 14 号引脚要接试验箱下方的+5V 电源，7 号引脚要接试验箱下方的地（GND）。示波器的电源开关在左下方，信号送到 CH1 通道上，对应的示波器上 CH1 通道按键要按下。也可以同时将两个信号分别送到 CH1 和 CH2 上，这时 CH1 和 CH2 通道按键都要按下。最后按下示波器右上方的“autoset”按键，稍等片刻就可以在示波器的屏幕上看到对应的脉冲波形了。通过旋转频率调节旋钮，可以增大或者减小实际输出脉冲的频率值。

(2) 将 1 号引脚和 Y 输出用两个探头分别连接到示波器的 CH1 和 CH2 通道上，用示波器进行观察两个波形的高低变化特点。记录下示波器上的两种波形。如果 1 号引脚的波形如下图所示，请画出与之对应的 Y 输出端的波形图。

1 号引脚的波形：



Y 输出端的波形：

根据图 1.9 写出输出 Y 的函数表达式，分析根据表达式得到的结果和示波器显示的结果是否一致。

8. 用异或门、或非门、与非门组成的全加器电路的逻辑功能测试（选做）

全加器电路可以用两个半加器和两个与门、一个或门组成。在实验中，也常用异或门、或非门和与非门来实现。

(1) 画出用异或门、或非门和与非门实现全加器的逻辑电路图，写出逻辑表达式。

(2) 找出异或门、或非门和与非门器件，按自己设计画出的电路图接线，注意：接线时，或非门中不用的输入端应该接地。与非门中不用的输入端应该接 V_{CC} 。

(3) 当输入端 A_i B_i C_{i-1} 为下列情况时，测量 S_i 和 C_i 的逻辑状态并填入表 1.9。

表 1.9

输入			输出	
A_i	B_i	C_{i-1}	S_i	C_i
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

五、实验报告

1. 按各步聚要求填表并画逻辑图。

2. 回答问题。

(1) 怎样判断门电路逻辑功能是否正常？

(2) 与非门一个输入接连续脉冲，其余端什么状态时允许脉冲通过？什么状态时禁止脉冲通过？

3. 整理实验数据、图表并对实验结果进行分析讨论。

4. 总结全加器卡诺图的分析方法。

5. 总结实验中出现的问题和解决的办法。

6. 预习实验二

实验二 组合逻辑电路分析与设计

一. 实验目的

1. 熟悉集成数据选择器、译码器的逻辑功能及测试方法。
2. 学会用集成数据选择器、译码器进行逻辑设计。
3. 熟悉组合逻辑电路的分析和验证方法。
4. 初步掌握利用 MSI 器件设计组合逻辑电路的方法。

二. 实验预习要求

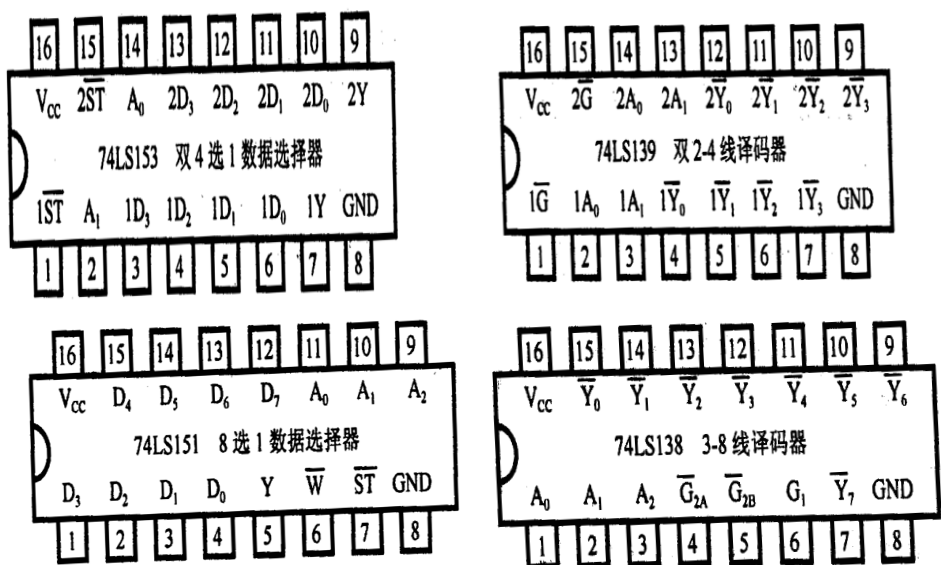
1. 复习实验芯片的逻辑功能及逻辑函数表达式。
2. 复习实验所用各芯片的结构图、管脚图和功能表。
3. 复习实验所用的相关原理。
4. 按要求设计实验中的各电路，给出原理图。

三. 实验原理

1. 组合逻辑电路的设计：就是按照具体逻辑命题按要求设计出最简的组合电路。经典的组合逻辑设计步骤如下：
 - (1) 根据给定事件的因果系列写函数式；
 - (2) 对函数式进行化简或变换；
 - (3) 画出逻辑图，并测试逻辑功能。
2. 数据选择器又称多路选择开关。数据选择器的主要作用是在地址码的控制下，从多个输入数据中选择其中一个送至输出端。通常把数据输入端的个数称为通道数。它除了具有选择信息的功能外，还可以用来形成各种逻辑函数。

四. 实验仪器及设备

1. 数字电路实验箱
2. 双踪示波器、万用表
3. 器件：74LS153、74LS151、74LS139、74LS138 和 74LS00 各一片。



五. 实验内容

1. 测试 74LS153 双4选1数据选择器的逻辑功能。

(1) 首先根据表 2.1, 检测 74LS153 双4选1数据选择器的逻辑功能和芯片的完好情况。74LS153 芯片的电源和地, 先要连接上。

(2) 按图 2.1 接线, 4 个数据输入引脚 C_0 — C_3 分别接函数信号发生器和试验箱上的 10KHz、1KHz、 V_{CC} 和 GND。按照真值表 2.1 来改变引脚 B、A 和使能引脚 G 的电平(K3、K2 和 K1), 产生不同的组合, 观测每种组合下数据选择器的输出波形。Output 输出送到示波器

表 2.1 74LS153 真值表

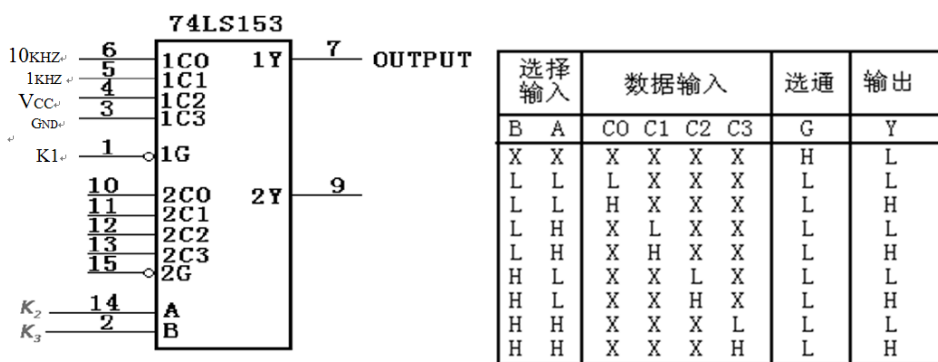


图 2.1 74LS153 实验接线图

上的红色线端, 示波器上的黑色线端接地。记录下示波器右下方与输出波形对

应的频率值。K1、K2 和 K3 是实验箱上 3 个不同的逻辑开关。

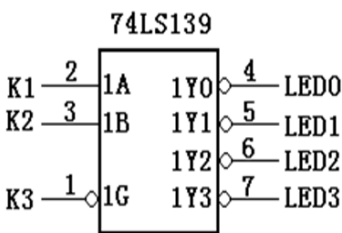
对实验结果进行分析，将输出 Y 的特点填入下表中：

输入		输出
B	A	Y
0	0	
0	1	
1	0	
1	1	

2. 测试 74LS139 双 2-4 译码器的逻辑功能。

(1) 按图 2.2 接线，芯片 74LS139 的电源和接地引脚要先连接上。4 个译码输出引脚 Y3—Y0 接实验箱上的电平指示灯。G、B、A 接实验箱上的逻辑开关 K3—K1。改变逻辑开关的值，观测并记录电平指示灯的显示状态，将结果填入表格 2.2 中。并总结译码器输出的特点。

表 2.2 74LS139 真值表



输入端			输出端			
允许 G	选择					
	B	A	Y0	Y1	Y2	Y3
H	X	X				
L	L	L				
L	L	H				
L	H	L				
L	H	H				

图 2.2 74LS139 实验接线图

3. 用 8 选 1 数据选择器 74LS151 实现逻辑函数：

$$F = A \cdot \overline{B} + \overline{A} \cdot C + B \cdot \overline{C}$$

画出逻辑电路连接图（参考课本中例题），并进行实际连线调试。记录结果填入下表 2.3 中。

调试芯片 74LS151 的电源和接地引脚要先连接上。

将这种设计方式与实验二的用与非门实现的方式进行对比，说明两种方式的优缺点。

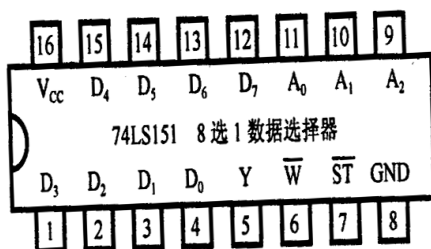


表 2.3 真值表

输入			输出
A	B	C	F
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

4. 用 3-8 译码器 74LS138 芯片和一片 74LS00 芯片实现逻辑函数:

$$F = \overline{A} \cdot \overline{B} \cdot C + A \cdot \overline{B} \cdot \overline{C} + A \cdot \overline{B} \cdot C$$

按要求画出逻辑电路连接图, 并进行实际连线调试, 记录结果填入表 2.4 中。调试时芯片 74LS138 和 74LS00 的电源和接地引脚要先连接上。

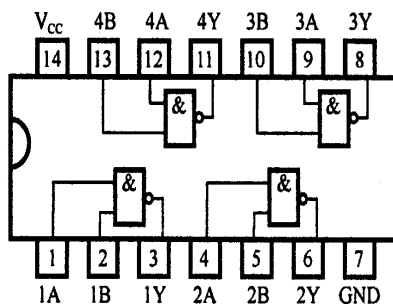
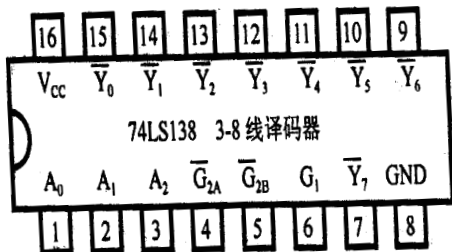


表 2.4 真

值表

输入			输出
A	B	C	F
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

六．实验报告要求与思考题

1. 按要求整理有关实验数据，分析问题要写出分析过程，检测方案及检测结果。
逻辑设计问题写出设计过程，画出逻辑图，给出调试方案和调试结果。
2. 总结利用 MSI 器件设计组合逻辑电路的方法。
3. 预习实验三

实验三 数值比较器/超前进位加法器的应用

一. 实验目的

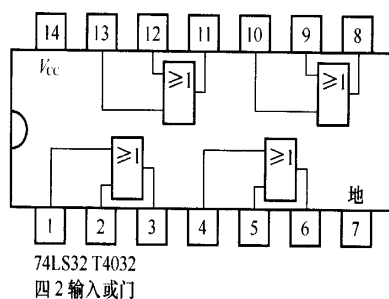
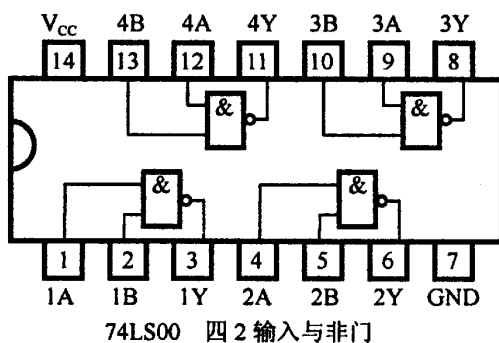
1. 熟悉集成数值比较器/超前进位加法器的逻辑功能及测试方法。
2. 学会用集成数值比较器/超前进位加法器进行逻辑设计。
3. 熟悉组合逻辑电路的验证和设计方法。
4. 初步掌握利用 MSI 器件设计组合逻辑电路的方法。

二. 实验预习要求

1. 复习实验芯片的逻辑功能及逻辑函数表达式。
2. 复习实验所用各芯片的结构图、管脚图和功能表。
3. 复习实验所用的相关原理。
4. 按要求设计实验中的各电路，给出原理图。

三. 实验仪器及设备

1. 数字电路实验箱
2. 双踪示波器、万用表
3. 器件：74LS00、74LS85 各一片
74LS32、74LS83 各两片。



四. 实验原理

1. 数值比较器的原理

在数字系统中，常常要比较两个数的大小。数值比较器就是对两数 A、B 进行比较，以判断其大小的逻辑电路。比较结果有 $A > B$ 、 $A < B$ 、 $A = B$ 三种情况。下表 3.1 和图 3.1 是最简单的一位数值比较器的真值表和逻辑电路图。

表 3.1 一位数值比较器的真值表

输入		输出		
A	B	$F_{A>B}$	$F_{A<B}$	$F_{A=B}$
0	0	0	0	1
0	1	0	1	0
1	0	1	0	0
1	1	0	0	1

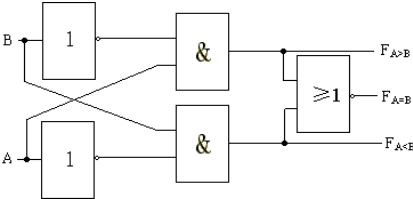


图 3.1 一位数值比较器的逻辑电路图

对于多位的情况，一般说来，先比较高位，当高位不等时，两个数的比较结果就是高位的比较结果。当高位相等时，两数的比较结果由低位决定。

a) 集成数值比较器 74LS85

集成数值比较器 74LS85 是四位数值比较器，它的管脚图和真值表如图 3.2 和表 3.2 所示：

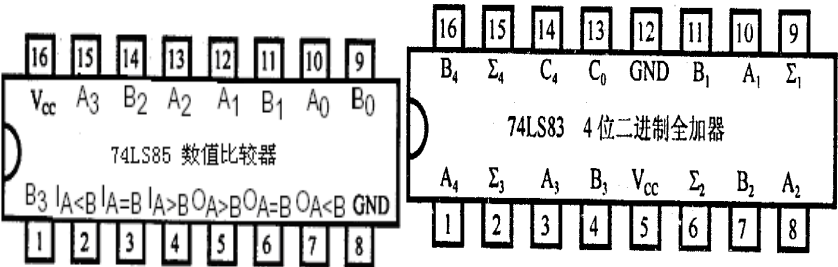


图 3.2 74LS85 的管脚图

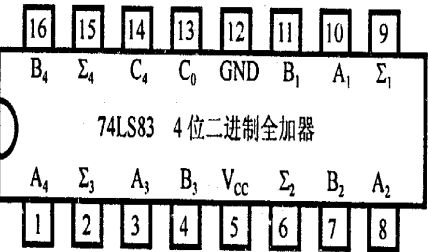


图 3.3 74LS83 的管脚图

在图 3.2 中，引脚 10、12、13、15 和 1、9、11、14 脚是输入端，分别对应两个 4 位的输入数 A 和 B。引脚 5、6、7 脚为输出端，对应于比较后的结果。

另外引脚 2、3、4 为级联输入端。没有级联时，接 010。引脚 8 接地，引脚 16 接电源。

表 3-2 74LS85 的真值表

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}	O _{A>B}	O _{A<B}	O _{A=B}
A ₃ >B ₃	X	X	X	X	X	X	H	L	L
A ₃ <B ₃	X	X	X	X	X	X	L	H	L
A ₃ =B ₃	A ₂ >B ₂	X	X	X	X	X	H	L	L
A ₃ =B ₃	A ₂ <B ₂	X	X	X	X	X	L	H	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ >B ₁	X	X	X	X	H	L	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ <B ₁	X	X	X	X	L	H	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ >B ₀	X	X	X	H	L	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ <B ₀	X	X	X	L	H	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	H	L	L	H	L	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	L	H	L	L	H	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	X	X	H	L	L	H
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	H	H	L	L	L	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	L	L	L	H	H	L

2. 74LS83 是超前进位加法器，可以实现两个 4 位的二进制数进行加法运算，引脚图如图 3.3 所示。该芯片在使用时，电源和地引脚要相应的连接上。
3. 七段显示器：用来显示数字、文字或符号的器件。

七段数码显示器是由 a~g 等七段可发光的线段拼合而成，控制各段的亮或灭可以显示不同的字符或数字。

七段数码显示器有发光二极管（LED）数码管和液晶显示器（LCD）两种。

LED 数码管分为共阴管和共阳管，目前使用最广泛。

五．实验内容

1. 测试四位数值比较器 74LS85 的逻辑功能。

按照图 3.4 所示，将输入信号 A₃、A₂、A₁、A₀、B₃、B₂、B₁、B₀ 和低位级联输入端 A<B、A=B、A>B 分别连接逻辑开关 K₈、K₇、K₆、K₅、K₄、K₃、K₂、K₁ 和 K₁₂、K₁₁、K₁₀。输出 L、G、M 连接三个电平指示灯。

根据表 3.3 中输入量的取值的不同，控制逻辑开关取不同的值，观察 L、G、M 输出值，填入表 3.3 中。

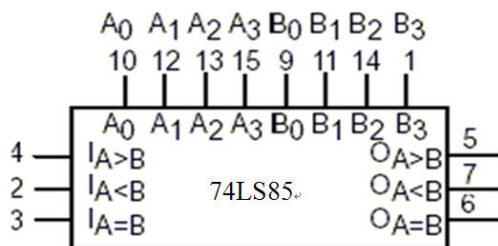


图 3-4

表 3-3 四位数值比较器逻辑功能验证实验数据表

输 入								输 出		
比较输入端								级联输入端		
A3	A2	A1	A0	B3	B2	B1	B0	A<B	A=B	A>B
0	0	0	0	1	0	1	0	0	1	0
0	0	0	1	1	0	1	0	0	1	0
0	0	1	0	1	0	1	0	0	1	0
0	0	1	1	1	0	1	0	0	1	0
0	1	0	0	1	0	1	0	0	1	0
0	1	0	1	1	0	1	0	0	1	0
0	1	1	0	1	0	1	0	0	1	0
0	1	1	1	1	0	1	0	0	1	0
1	0	0	0	1	0	1	0	0	1	0
1	1	1	1	1	0	1	0	0	1	0

2. 图 3.5 为 2421BCD 码转换为 8421BCD 码的变换电路。试分析其功能的实现方法，并验证该电路是否能完成上述功能。记录结果填入表 3.4 中。其中 74LS48 芯片和七段显示器在实验箱上已经连接在一起了。

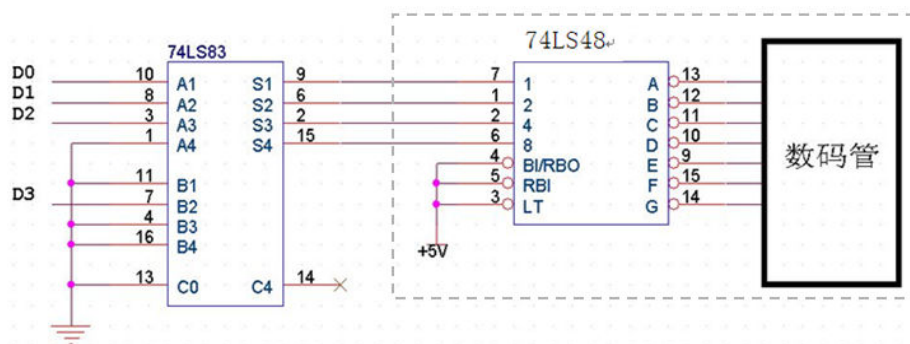


图 3.5 2421BCD—8421BCD 转换电路

表 3.4 真值表

输入				输出			
D3	D2	D1	D0	S4	S3	S2	S1
0	0	0	0				
0	0	0	1				
0	0	1	0				
0	0	1	1				
0	1	0	0				
0	1	0	1				
0	1	1	0				
0	1	1	1				
1	1	1	0				
1	1	1	1				

3. 采用两片 74LS83 芯片和其他门电路芯片设计一个 8421BCD 码全加器电路, 其要求输入两个 BCD 码数 A 和 B, 出来的是 BCD 码数 C_0 和 Y, 如图 3.6 所示。

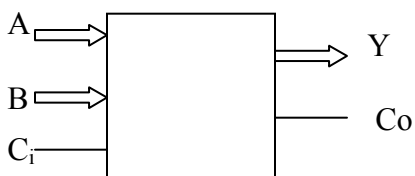


图 3.6 8421BCD 码全加器方框图

- 写出逻辑设计过程及相关表达式; (参考课本中的例题)
- 画出逻辑电路图; $Y_3Y_2Y_1Y_0$ 连接到 LED 灯的同时, 并行连接到一个七段显示器上。 C_o 连接到一个七段显示器上。 C_o 只有一位, 送到七段显示器上时, 高位要补上 3 个零。记录结果, 填入下表 3.5 中。

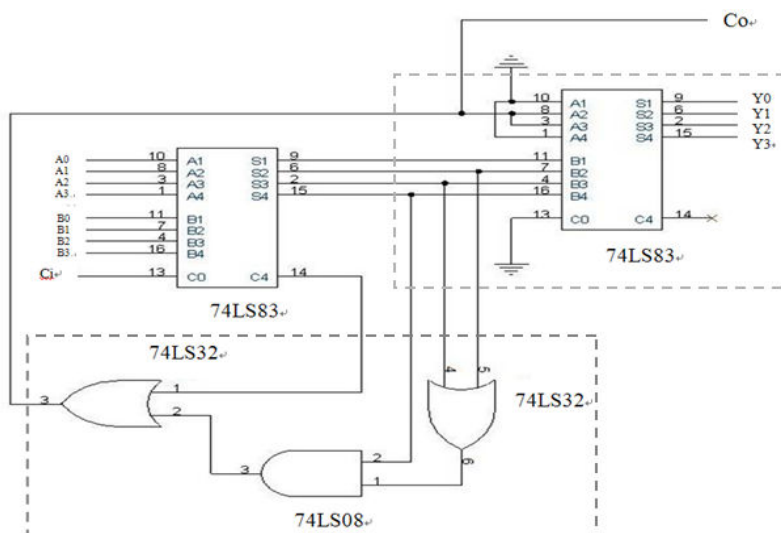


表 3.5

输 入								输 出					两个七段显示器上的字型
A3	A2	A1	A0	B3	B2	B1	B0	Co	Y3	Y2	Y1	Y0	
0	0	0	0	0	1	0	1						
0	0	0	1	0	1	0	1						
0	0	1	0	0	1	0	1						
0	0	1	1	0	1	0	1						
0	1	0	0	0	1	0	1						
0	1	0	1	0	1	0	1						
0	1	1	0	0	1	0	1						
0	1	1	1	0	1	0	1						
1	0	0	0	0	1	0	1						
1	0	0	1	0	1	0	1						

六．实验报告要求与思考题

1. 按要求整理有关实验数据，分析问题要写出分析过程，检测方案及检测结果。逻辑设计问题写出设计过程，画出逻辑图，给出调试方案和调试结果。
2. 总结利用 MSI 器件设计组合逻辑电路的方法。
3. 预习实验四

实验四 触发器及其应用

一、 实验目的

- 1、掌握基本 RS 触发器、D 触发器、JK 触发器的工作原理。
- 2、学会正确测试 RS 触发器、D 触发器、JK 触发器的逻辑功能。
- 3、熟悉触发器之间相互转换的方法。
- 4、了解用触发器构成自循环寄存器的电路结构及工作过程。

二、 实验所用器件和仪表

- | | |
|-------------------|-----|
| 1、与非门 74LS00 | 1 片 |
| 2、双 D 触发器 74LS74 | 2 片 |
| 3、双 JK 触发器 74LS73 | 1 片 |
| 4、万用表和示波器 | |
| 5、实验箱 | |

三、 实验内容

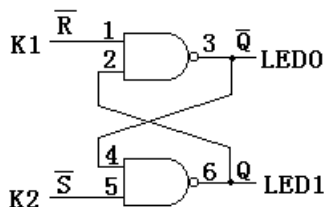
- 1、设计基本 RS 触发器并验证其功能。
- 2、验证 D 触发器功能。
- 3、验证 JK 触发器功能。
- 4、将 JK 触发器转换成 D 触发器和 T 触发器。
- 5、用 D 触发器构成自循环寄存器（又称环形计数器）。

四、实验接线图和测试步骤

- 1、设计基本 RS 触发器并验证其功能。（每个芯片的电源和地端要连接）

图 4.1 是基本 RS 触发器接线图。图中，K1、K2 是电平开关输出，LED0、LED1 是电平指示灯。基本 SR 触发器的测试步骤及结果如下：

- (1) $\overline{R}=0$, $\overline{S}=1$, 测得 $\overline{Q}=$, $Q=$ 。
- (2) $\overline{R}=1$, $\overline{S}=1$, 测得 $\overline{Q}=$, $Q=$ 。
- (3) $\overline{R}=1$, $\overline{S}=0$, 测得 $\overline{Q}=$, $Q=$ 。
- (4) $\overline{R}=1$, $\overline{S}=1$, 测得 $\overline{Q}=$, $Q=$ 。



4.1 基本 RS 触发器接线图

(5) $\overline{R}=0$, $\overline{S}=0$, 测得 $\overline{Q}=$, $Q=$ 。

根据触发器的定义，Q 和 \overline{Q} 应互补，因此 $\overline{R}=0$, $\overline{S}=0$ 是非法输入，当它们同时回到 1 时，Q 的状态不能确定。SR 触发器功能表如下：

输 入		输 出	
\overline{R}	\overline{S}	\overline{Q}	Q
0	0	1*	1*
0	1	1	0
1	0	0	1
1	1	\overline{Q}	Q

2、验证 D 触发器功能。

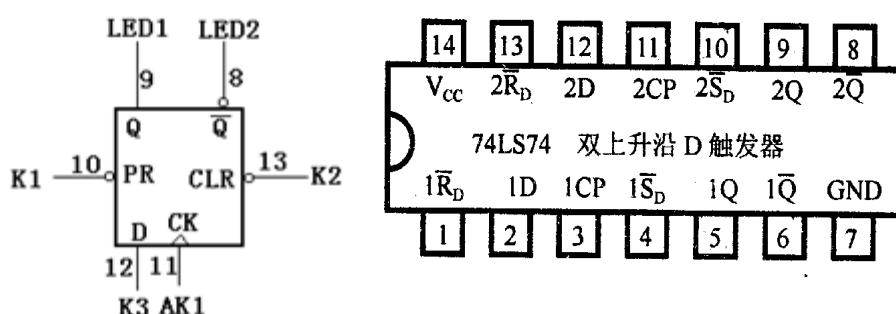


图 4.2 测试 D 触发器的接线图

注: $PR = \overline{S_D}$, $CLR = \overline{R_D}$

图 4.2 是测试 D 触发器的接线图, K1、K2、K3 接逻辑开关, LED0、LED1 是电平指示灯, AK1 是单次脉冲,接逻辑开关 K4。

测试步骤如下:

- (1) $CLR = 0$, $PR = 1$, 测得 $\overline{Q} =$, $Q =$ 。
- (2) $CLR = 1$, $PR = 0$, 测得 $\overline{Q} =$, $Q =$ 。
- (3) $CLR = 1$, $PR = 1$, $D = 1$, AK1 接逻辑开关 K4, 拨动 K4, 测得 $\overline{Q} =$, $Q =$ 。
- (4) $CLR = 1$, $PR = 1$, $D = 0$, AK1 接逻辑开关 K4, 拨动 K4, 测得 $\overline{Q} =$, $Q =$ 。将测试结果与下面 74LS74 功能表进行比较, 看是否一致。

D 触发器 74LS74 功能表

输 入				输 出	
PR	CLR	CK	D	Q	\overline{Q}
L	H	X	X	H	L
H	L	X	X	L	H
H	H	\uparrow	H	H	L
H	H	\uparrow	L	L	H
H	H	L	X	Q	\overline{Q}

3、验证 JK 触发器功能。（每个芯片的电源和地端要连接。K2-K4 输入来源于逻辑开关，AK1 输入来源于实验箱左下方的连续脉冲源，脉冲档位选中 1HZ（或者接 K1 逻辑开关），LED0 和 LED1 输出送到两个实验箱上不同的电平指示灯上，观察在不同的输入时 LED 灯的亮灭情况。）

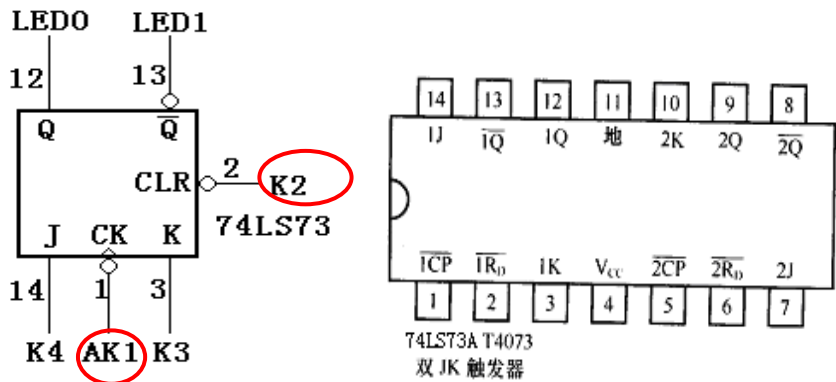


图 4.3 测试 JK 触发器的接线图

图 4.3 是测试 JK 触发器的接线图。K2、K3、K4 是电平开关输出，LED0、LED1 是电平指示灯，AK1 是单次脉冲。**74LS73 引脚 4 接+5V，引脚 11 接地。74LS73 只有复位端 CLR。**

- (1) $\overline{\text{CLR}} = 0$ ，测得 $\overline{\text{Q}} = 1$ ， $\text{Q} = 0$ 。
- (2) $\overline{\text{CLR}} = 1$ ， $\text{J} = 0$ ， $\text{K} = 0$ ，按单次脉冲按钮 AK1，测得 $\overline{\text{Q}} =$ ， $\text{Q} =$ 。
- (3) $\overline{\text{CLR}} = 1$ ， $\text{J} = 1$ ， $\text{K} = 0$ ，按单次脉冲按钮 AK1，测得 $\overline{\text{Q}} =$ ， $\text{Q} =$ 。
- (4) $\overline{\text{CLR}} = 1$ ， $\text{J} = 0$ ， $\text{K} = 0$ ，按单次脉冲按钮 AK1，测得 $\overline{\text{Q}} =$ ， $\text{Q} =$ 。
- (5) $\overline{\text{CLR}} = 1$ ， $\text{J} = 0$ ， $\text{K} = 1$ ，按单次脉冲按钮 AK1，测得 $\overline{\text{Q}} =$ ， $\text{Q} =$ 。
- (6) $\overline{\text{CLR}} = 1$ ， $\text{J} = 0$ ， $\text{K} = 0$ ，按单次脉冲按钮 AK1，测得 $\overline{\text{Q}} =$ ， $\text{Q} =$ 。
- (7) $\overline{\text{CLR}} = 1$ ， $\text{J} = 1$ ， $\text{K} = 1$ ，按单次脉冲按钮 AK1，测得 $\text{Q} =$ ， $\text{Q} =$ ；再按单次脉冲按钮 AK1，测得 $\overline{\text{Q}} =$ ， $\text{Q} =$ 。

根据以上的测试，请填写出 74LS73 功能表。

JK 触发器 74LS73 功能表

输 入					输 出	
PR	CLR	CK	J	K	Q	$\overline{\text{Q}}$
L	H	X	X	X		
H	L	X	X	X		
H	H	\uparrow	L	L		
H	H	\uparrow	L	H		
H	H	\uparrow	H	L		
H	H	\uparrow	H	H		

4、将 JK 触发器转换成 D 触发器和 T 触发器。

a) 按图 4.4 连线，将 JK 触发器转换成为 D 触发器。

测试该 D 触发器的逻辑功能，并画出功能表将实验结果填入表内。

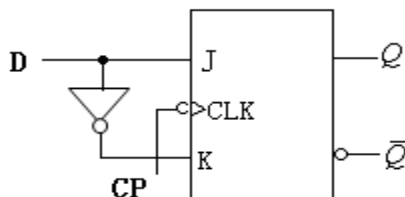


图 4.4 JK 触发器转换为 D 触发器

b) 将 JK 触发器的 J、K 端连在一起，构成 T 触发器

分别当 $T=J=K=0$ 和 $T=J=K=1$ 时，在 CP 端输入连续脉冲，连接实验箱左下方的连续脉冲源，脉冲档位选中 1HZ（或者接 K1 逻辑开关）观察 Q 端的变化，用双踪示波器测量 CP、 Q 和 \bar{Q} 的波形，注意相位关系，描绘之。

5、用 D 触发器构成自循环寄存器（又称环形计数器）。

a) 用双 D 触发器 74LS74 构成一个四位自循环寄存器。方法是第一级的 Q 端接第二级的 D 端，依次类推，最后第四级的 Q 端接第一级的 D 端。四个 D 触发器的 CLK 端连接在一起，然后接单脉冲时钟。接线电路如图 4.5。

b) 将触发器 Q_0 置 1， Q_1 、 Q_2 、 Q_3 清 0。接单脉冲按钮，观察并记录 Q_0 、 Q_1 、 Q_2 、 Q_3 的值。

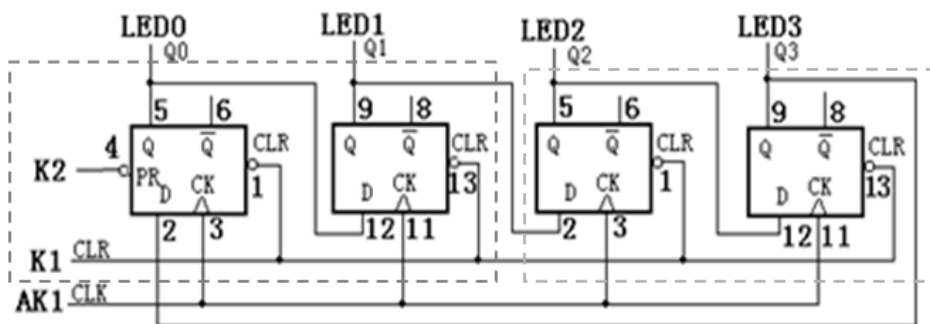


图 4.5 自循环寄存器接线图

图 4.5 中，K1、K2 连接两个逻辑开关，AK1 是单脉冲，连接实验箱左下方的连续脉冲源，脉冲档位选中 1HZ（或者接 K3 逻辑开关）LED0、LED1、LED2 和 LED3 是电平指示灯。测试步骤：

- (1) 置 K1 为低电平，K2 为高电平，四个电平指示灯灭，表示 $Q_3Q_2Q_1Q_0$ 为 0000。
- (2) 置 K1 为高电平，K2 为低电平，LED0 指示灯亮，表示 $Q_3Q_2Q_1Q_0$ 为

0001。

(3) 置 K1、K2 为高电平。按单脉冲按钮 AK1，观察 Q₃Q₂Q₁Q₀ 的值变化如下：

环形计数器状态转移表

Q ₃	Q ₂	Q ₁	Q ₀
0	0	0	1
0	0	1	0
0	1	0	0
1	0	0	0
0	0	0	1

(4) 将接单脉冲 AK1 的线 (CLK) 改接 1KHz 连续脉冲，用示波器观测 Q₀、Q₁、Q₂、Q₃。画出连续计数时钟下 Q₀、Q₁、Q₂ 和 Q₃ 的波形如图 4.6。

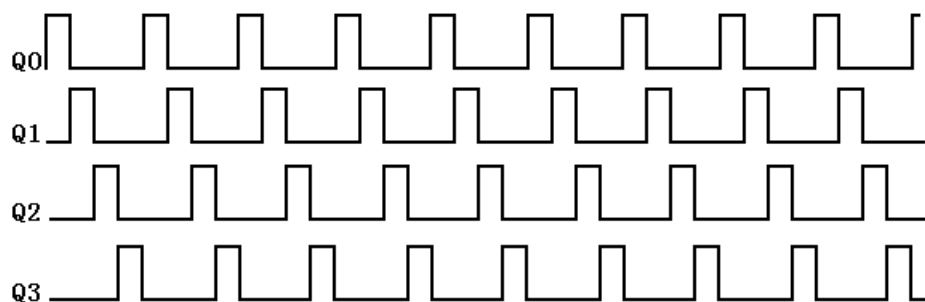


图 4.6 自循环寄存器器波形图

五．实验报告要求与思考题

1. 按要求整理有关实验数据，分析问题要写出分析过程，检测方案及检测结果。逻辑设计问题写出设计过程，画出逻辑图，给出调试方案和调试结果。
2. 总结各种触发器的工作特性以及相关转换的方法。
3. 预习实验五

实验五 时序电路测试及研究

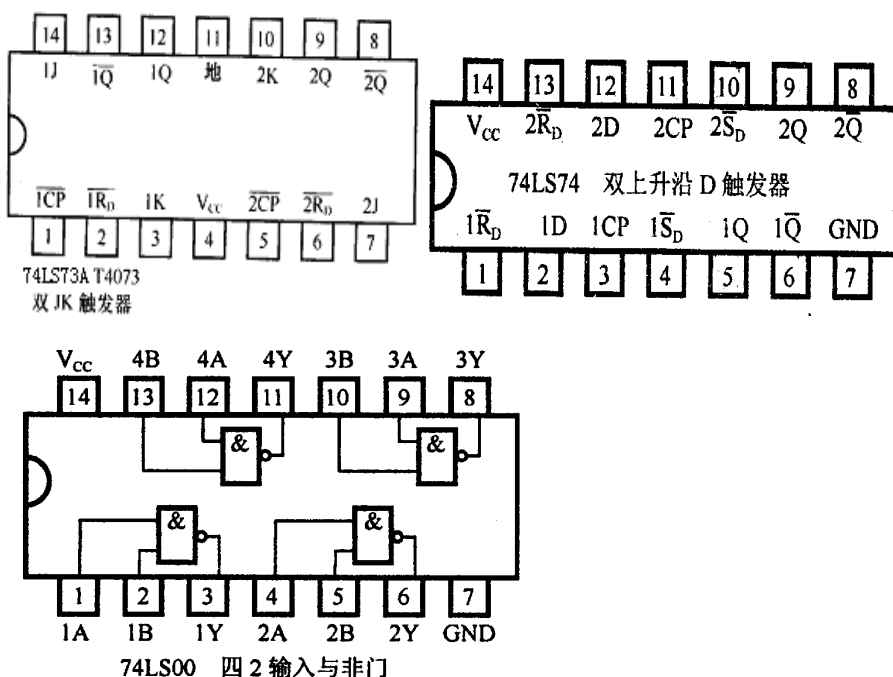
一、实验目的

- 1、掌握常用时序电路分析、设计及测试方法。
- 2、训练独立进行实验的技能。

二、实验仪器及材料

- 1、双踪示波器
- 2、器件

74LS73	双 J-K 触发器	2 片
74LS74	双 D 触发器	2 片
74LS00	二输入端四与非门	1 片



三、实验原理

1、计数器是最典型的时序电路之一。它可对脉冲的个数进行计数。

(1) 计数器的种类繁多,分类方法也有多种,例如,按进位数值来分类,可分为二进制计数器、二十进制器等;按计数器中触发器翻转的次序来分类,可以分为同步计数器和异步计数器;按计数过程中计数器数字的增减来分类,可以分为加法计数器、减法计数器和可逆计数器等。

(2) 图 5-1 为异步二进制加法计数器,由 D 触发器构成。除第一级触发器由计

数脉冲 CP 直接驱动外,其它各级触发器的动作都要由其前一级触发器 Q 的状态变化来确定,可见这些触发器的动作时间各异。当 $\overline{R_D} = 0$ 时,计数器置零 ($Q_3Q_2Q_1Q_0=0000$)。

(3) 然后将 $\overline{R_D} = 1$ 。这时计数脉冲从 CP 端输入,第一个计数脉冲输入后,计数器 $Q_3Q_2Q_1Q_0$ 状态从 0000 变成了 0001,随着计数脉冲的继续输入,计数器的状态根据二进制码顺序依次递增,第十五个脉冲输入后,计数器状态为 1111。第十六个脉冲输入后,计数器恢复起始状态 0000。如果继续输入脉冲,则重复上述过程。

(4) 异步二进制减法计数的电路结构与加法计数器相似,不同的是级间改由前级的 Q 与后级的 CP 连接。异步二进制减法计数器的计数过程是每输入一个 CP 脉冲,计数器的数值减 1,例如设计数器原状态为 0000,则输入第一个 CP 脉冲后,变为 1111。输入第二个 CP 脉冲后,变为 1110,依次类推。

2、异步二 - 十进制加法计数器如图 5-2 所示。它由两片 74LS73 双 J-K 触发器和一片 74LS00 二输入端四与非门组成。前九个计数脉冲输入后计数器的状态变化与异步二进制数据相同;当第十个脉冲输入后,计数器状态恢复为 0000,并从 $\overline{Q_D}$ 端送出一个进位脉冲。

把移位寄存器的输出,以一定的方式反馈到串行输入端可构成寄存器型计数器,常用的寄存器型计数器有环形计数器。

3、图 5-3 是由 D 触发器 74LS74 组成的环形计数器。第四级的 Q_D 端与第一级的 1D 端相接(反馈)。这种电路,在输入计数脉冲 CP 操作下,其状态在 1000, 0100, 0010, 0001 (有效状态)中循环,但工作时,必须先用启动脉冲 ($\overline{S_D}$ 、 $\overline{R_D}$) 将计数器置入有效状态。由于不能自启动,倘若由于电源故障可信号干扰,使电路进入非使用状态 (无效状态),计数器就无法恢复正常工作。

4、图 5-4 电路是具有自启动功能的环形计数器。无论原状态如何,经数个 CP 脉冲作用后,电路总能进入有效循环计数。

四、实验内容

1、用两片 D 触发器 74LS74 芯片构成异步二进制加 / 减计数器

参照图 5-1,用四只 D 触发器构成四位二进制异步加法计数器,它的连接特点是将每只低位 D 触发器的 \overline{Q} 端和高一位的 CP 端相连接。输出 $Q_3Q_2Q_1Q_0$ 分别连接到四个电平指示灯,同时用四根线接到一个七段显示器的输入端 DCBA 上。输入 R_D 接逻辑开关, CP_0 接试验箱左下方的 1HZ 连续脉冲或者接逻辑开关。

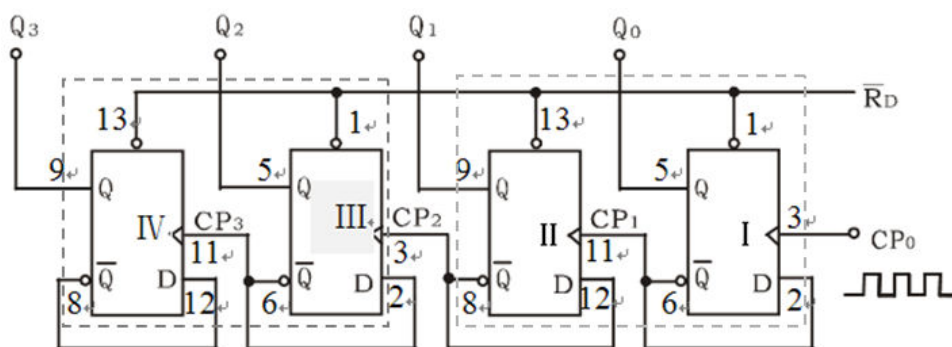


图 5-1 四位二进制异步加法计数器

先将 R_D 置 0，从而使 $Q_3Q_2Q_1Q_0=0000$ ，然后将 $R_D=1$ ，观察 CP_0 引脚进入的脉冲对输出 $Q_3Q_2Q_1Q_0$ 的影响，并将观察结果填入下表中。

CP	输 入				输 出				显示字型
	Q_3	Q_2	Q_1	Q_0	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}	Q_0^{n+1}	
1	0	0	0	0					
2	0	0	0	1					
3	0	0	1	0					
4	0	0	1	1					
5	0	1	0	0					
6	0	1	0	1					
7	0	1	1	0					
8	0	1	1	1					
9	1	0	0	0					
10	1	0	0	1					
11	1	0	1	0					

若将图 5-1 稍加改动，即将低位触发器的 Q 端与高一位的 CP 端相连接，去掉原来的 \bar{Q} 端与高一位的 CP 端连线，即构成了一个 4 位二进制减法计数器。如图 5-2 所示。先将 R_D 置 0，从而使 $Q_3Q_2Q_1Q_0=0000$ ，然后将 $R_D=1$ ，观察 CP_0 引脚进入的脉冲对输出 $Q_3Q_2Q_1Q_0$ 的影响，并将观察结果填入下表中。

CP	输 入				输 出				显示字型
	Q_3	Q_2	Q_1	Q_0	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}	Q_0^{n+1}	
1	1	0	0	1					
2	1	0	0	0					
3	0	1	1	1					

4	0	1	1	0					
5	0	1	0	1					
6	0	1	0	0					
7	0	0	1	1					
8	0	0	1	0					
9	0	0	0	1					
10	0	0	0	0					
11	1	1	1	1					

2、异步二—十进制加法计数器

(1) 采用两片 74LS73 芯片和一片 74LS00 芯片，如图 5-3 连接。其中 QD、QC、QB、QA 四个输出端分别接发光管二极管显示，同时用四根线接到一个七段显示器的输入端 DCBA 上。CP 端连接实验箱上的 1HZ 连续脉冲。

(2) 在 CP 端接连续脉冲，观察 CP、QD、QC、QB、QA 的波形。

(3) 画出 CP、QD、QC、QB、QA 的波形。

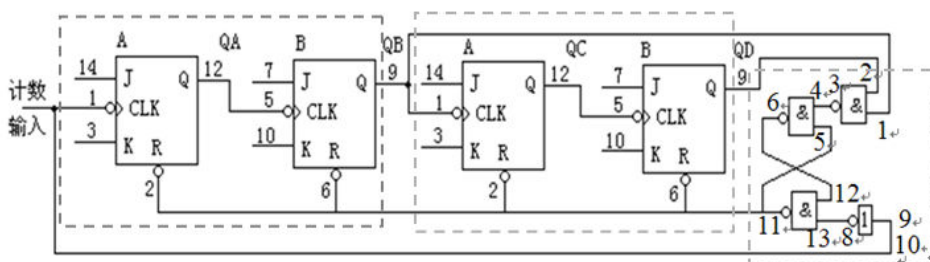


图 5-3 异步二 — 十进制加法计数器

3、自循环移位寄存器 — 环形计数器。

(1) 按图 5-4 接线，将 D、C、B、A 置为 0001，同时用四根线接到一个七段显示器的输入端 DCBA 上。CP 端连接实验箱上的 1HZ 连续脉冲。记录输出状态。

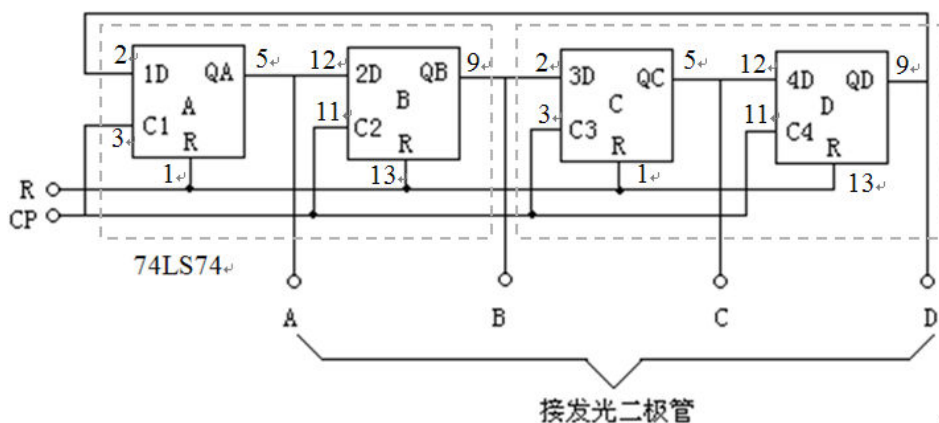


图 5-4 环形计数器（一）

改为连续脉冲计数，并将其中一个状态为“0”的触发器置为“1”（模拟干扰信号作用的结果）。观察计数器能否正常工作。分析原因。看看能否自启动。

(2)按图 5-5 接线，与非门可以用 74LS10 三输入端三与非门(也可以用 74LS00 芯片来实现该逻辑结构，去掉两个三输入的与非门。请大家提前画出用 74LS00 实现对应的连线图)，重复上述实验，对比实验结果，总结关于自启动的体会。

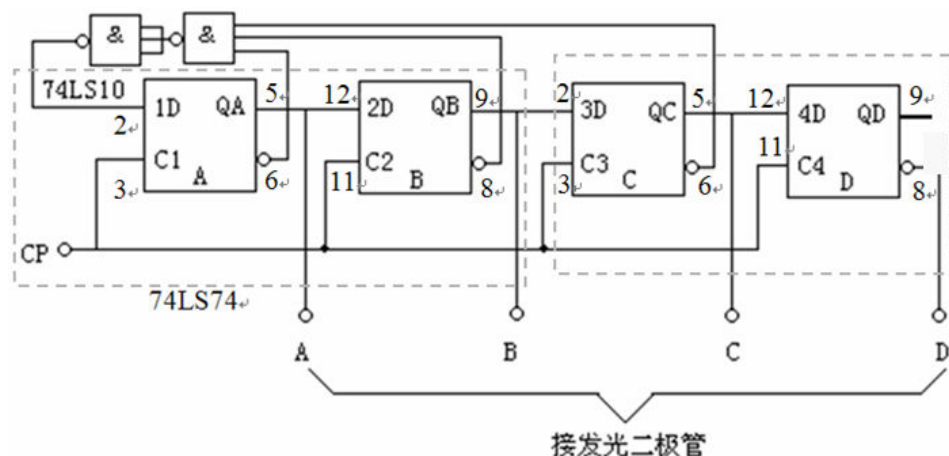


图 5-5 环形计数器 (二)

五、实验报告与分析

- 1、画出实验内容要求的波形及记录表格。
- 2、总结时序电路特点。
- 3、预习实验六

实验六 集成计数器芯片的应用

一、实验目的

- 1、掌握计数器芯片 74LS160 的功能。
- 2、掌握计数器的级联方法。
- 3、熟悉任意模计数器的构成方法。
- 4、熟悉数码管的使用。

二、实验说明

计数器器件是应用较广的器件之一，它有很多型号，各自完成不同的功能，供使用中根据不同的需要选用。本实验选用 74LS160 做实验用器件。74LS160 引脚图如图 7.1 所示。74LS160 是十进制 BCD 同步计数器。CP 是时钟输入端，上升沿触发计数器对应的输出翻转。允许端 CT_P 和 CT_T 都为高电平时允许计数，允许端 $CT_T \cdot CT_P$ 为低电平时禁止计数，保持原状态不变。同步预置端 LD 加低电平时，在下一个时钟的上升沿将计数器置为预置数据端的值，即将 D3D2D1D0 四个输入端的值打入输出端 Q3Q2Q1Q0 上。清除端 CR 为异步清除，低电平有效，使输出 Qd, Qc, Qb, Qa 清零。74LS160 在计数值等于 9 时，进位输出引脚 Co 输出为高电平，并且持续时间只有一个时钟周期，这个引脚可用于级联。

三、实验所用仪器和芯片

- | | |
|----------------------|-----|
| 1、数字逻辑实验箱 | 1 台 |
| 2、二输入四与非门 74LS00 | 1 片 |
| 3、同步十进制加法计数器 74LS160 | 2 片 |

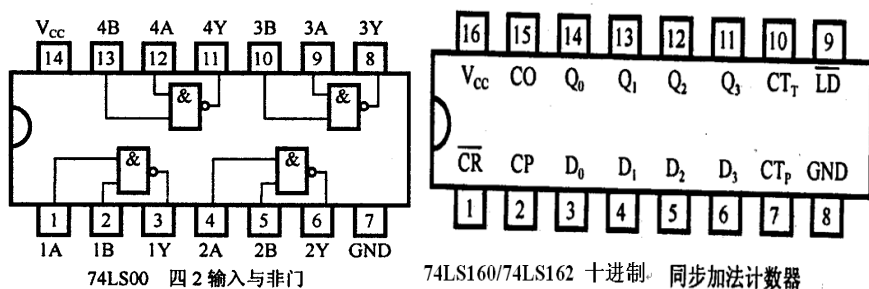


图 6.1

四、实验预习要求

- 1、复习有关计数器部分内容
- 2、绘出各实验内容的详细线路图
- 3、拟出各实验内容所需的测试记录表格
- 4、查手册、给出并熟悉实验所用各集成块的引脚排列图

五、实验原理和实验电路



1. 计数器

计数器不仅可用来计数，也可用于分频、定时和数字运算。在实际工程应用中，一般很少使用小规模触发器组成计数器，而是直接选用中规模集成计数器。

2. 四位 BCD 码计数器 74LS160 是十进制计数芯片。

74LS160是同步置数、异步清零的4位BCD码加法计数器，其功能表见表7.1。清零是异步的（直接清零），不管时钟输入CP、置数输入LD、使能输入CT_P和CT_T为何电平，清除输入端CR的低电平立即能把所有四个触发器的输出Q_d，Q_c，Q_b，Q_a直接置为低电平。预置是同步的，当CR为1时，如果置数输入端LD上输入一低电平，则禁止计数，并在下一个时钟CP的上升沿来临之后，不管使能输入CT_P和CT_T是何电平，输出Q_d，Q_c，Q_b，Q_a都与预置数据D₃D₂D₁D₀一致。这称为同步置数。外部引脚图如上图6.1所示。

表 6.1 74LS160 的功能表

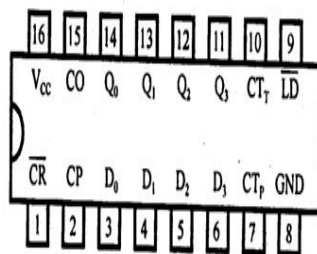
清 零 _⊗	预 置 _⊗	使能 _⊗		时钟 _⊗	预置数据输入 _⊗				输出 _⊗				工作模式 _⊗
R _D _⊗	LD _⊗	EP _⊗	ET _⊗	CP _⊗	A _⊗	B _⊗	C _⊗	D _⊗	Q _A _⊗	Q _B _⊗	Q _C _⊗	Q _D _⊗	
0 _⊗	× _⊗	× _⊗	× _⊗	× _⊗	× _⊗	× _⊗	× _⊗	× _⊗	0 _⊗	0 _⊗	0 _⊗	0 _⊗	异步清零 _⊗
1 _⊗	0 _⊗	× _⊗	× _⊗		D _A _⊗	D _B _⊗	D _C _⊗	D _D _⊗	D _A _⊗	D _B _⊗	D _C _⊗	D _D _⊗	同步置数 _⊗
1 _⊗	1 _⊗	0 _⊗	× _⊗	× _⊗	× _⊗	× _⊗	× _⊗	× _⊗	保 持 _⊗				数据保持 _⊗
1 _⊗	1 _⊗	× _⊗	0 _⊗	× _⊗	× _⊗	× _⊗	× _⊗	× _⊗	保 持 _⊗				数据保持 _⊗
1 _⊗	1 _⊗	1 _⊗	1 _⊗		× _⊗	× _⊗	× _⊗	× _⊗	计 数 _⊗				加 1 计数 _⊗

3、四位二进制计数器 74LS161 是十六进制芯片，共可以记录 16 个不同的状态。其功能表与表 7.1 相同。当输出 Q_d，Q_c，Q_b，Q_a 为 1111 时，进位输出引脚 Co 输出才为高电平，并且持续时间只有一个时钟周期，这个引脚可用于级联。

同步清零，含义是：清零信号 R_d 变为低电平后，要等 CP 下一个时钟的上升沿到来后，才能使输出 Q_d，Q_c，Q_b，Q_a 清零。也就是时钟同步。受同步时钟 CP 的控制。异步清零，含义是：清零信号 R_d 只要变为低电平，不需要等下一个时钟的上升沿到来，直接使输出 Q_d，Q_c，Q_b，Q_a 清零，不受同步时钟 CP 的控制，这种异步清零法会使清零前的上一个状态持续时间比较短。需要注意的是，74LS162 芯片是同步清零的芯片。而 74LS160 芯片是异步清零的芯片。

同步预置的含义是，在 CR=1 的条件下，当 LD=0 且有时钟脉冲 CP 的上升沿作用时，D₃，D₂，D₁，D₀ 输入端的数据将分别被 Q₃~Q₀ 所接收。由于置数操作必须有 CP 脉冲上升沿相配合，故称为同步置数。异步预置的含义是，在 CR=1 的条件下，当 LD=0 且不需要等待时钟脉冲 CP 的上升沿到来，D₃，D₂，D₁，D₀ 输入端的数据将分别被 Q₃~Q₀ 所接收。由于置数操作不需要 CP 脉冲上升沿配合，故称为异步置数。这种异步置数法会使置数前的上一个状态持续时间比较短。74LS191 芯片是异步置数的 16 进制的计数器芯片。

74LS161 外部引脚图和计数值的状态转换见图 6.2 相同。



74LS161 四位二进制同步加法计数器

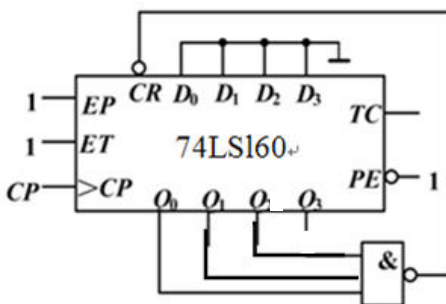
输入脉冲数	Q_3^n	Q_2^n	Q_1^n	Q_0^n	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}	Q_0^{n+1}
1	0	0	0	0	0	0	0	1
2	0	0	0	1	0	0	1	0
3	0	0	1	0	0	0	1	1
4	0	0	1	1	0	1	0	0
5	0	1	0	0	0	1	0	1
6	0	1	0	1	0	1	1	0
7	0	1	1	0	0	1	1	1
8	0	1	1	1	1	0	0	0
9	1	0	0	0	1	0	0	1
10	1	0	0	1	1	0	1	0
11	1	0	1	0	1	0	1	1
12	1	0	1	1	1	1	0	0
13	1	1	0	0	1	1	0	1
14	1	1	0	1	1	1	1	0
15	1	1	1	0	1	1	1	1
16	1	1	1	1	0	0	0	0

图 6.2

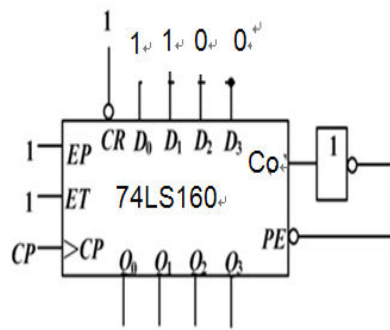
4. 集成计数器的应用——实现任意 M 进制计数器

一般情况任意 M 进制计数器的结构分为 3 类，第一类是由触发器构成的简单计数器。第二类是由集成二进制计数器构成计数器。第三类是由移位寄存器构成的移位寄存型计数器。第一类，可利用时序逻辑电路的设计方法步骤进行设计。第二类，当计数器的模 M 较小时用一片集成计数器即可以实现，当 M 较大时，可通过多片计数器级联实现。两种实现方法：反馈置位法和反馈复位法。具体参考实验电路如图 6.3 所示。

74LS160 芯片加上其它门电路，可以构成任意进制计数器。如下图图 6.3 所示。



复位法



置位法

图 6.3

五、实验内容

1. 采用 74LS160 芯片，按照下图 6.4 连线。将输出连接到七段显示器上，将 CLK 连接试验箱上的 1HZ 连续脉冲，观察并记录数码管显示的字型。注意 CR、LD、CTT 和 CTP 引脚都连接“1”信号。数码管的 VCC 要连接+5V。

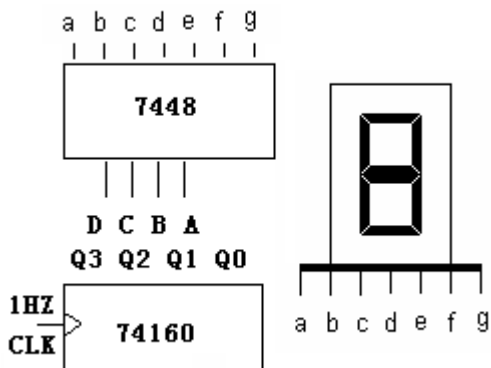


图 6.4

2、用 1 片 74LS160 和 1 片 74LS00 采用复位法构成一个模 7 计数器。画出逻辑电路连线图，用单脉冲做计数时钟，将输出连接到七段显示器上，观测并记录数码管显示的字型。

3、用 1 片 74LS160 和 1 片 74LS00 采用置位法构成一个模 7 计数器。画出逻辑电路连线图，用单脉冲做计数时钟，将输出连接到七段显示器上，观测并记录数码管显示的字型。

- 4、用 2 片 74LS160 和 1 片 74LS00 构成一个模 60 计数器。2 片 74LS160 的 Qd, Qc, Qb, Qa 分别接两个数码管的 D,C,B,A。画出逻辑电路连线图。用单脉冲做计数时钟，将输出分别连接到两个七段显示器上，检验接线是否正确，观测并记录两个数码管的数字变化。

六、实验报告

- 1、画出实验线路图，记录、整理实验现象及实验所得的有关波形。对实验结果进行分析。
- 2、比较 74LS160 和 74LS161 的不同。总结使用集成计数器芯片的体会。
- 3、预习实验七。

实验七 组合电路的 verilog 设计和仿真

一、实验目的

- (1) 熟悉 Modelsim 软件
- (2) 掌握 Modelsim 软件的编译、仿真方法
- (3) 熟练运用 Modelsim 软件进行 HDL 程序设计开发

二、实验内容

(一)、异或门的 verilog 设计和仿真

my_or, my_and 和 my_not 门构造一个双输入端的 xor 门, 其功能是计算 $z = x' y + xy'$, 其中 x 和 y 为输入, z 为输出; 编写激励模块对 x 和 y 的四种输入组合进行测试仿真

1、实验要求

用 Verilog HDL 程序实现一个异或门, Modelism 仿真, 观察效果。

2、步骤

- 1、建立工程
- 2、添加文件到工程
- 3、编译文件
- 4、查看编译后的设计单元
- 5、将信号加入波形窗口
- 6、运行仿真

实验描述如下:

```
module my_and(a_out,a1,a2);  
    output a_out;
```

```

        input  a1,a2;
        wire   s1;
        nand(s1,a1,a2);
        nand(a_out,s1,1'b1);
    endmodule

module my_not(n_out,b);

    output n_out;
    input  b;
    nand(n_out,b,1'b1);
endmodule


module my_or(o_out,c1,c2);
    output o_out;
    input  c1,c2;
    wire   s1,s2;
    nand(s1,c1,1'b1);
    nand(s2,c2,1'b1);
    nand(o_out,s1,s2);
endmodule


module MY_XOR(z,x,y);
    output z;
    input  x,y;
    wire   a1,a2,n1,n2;
    my_not STEP01(n1,x);
    my_not STEP02(n2,y);
    my_and STEP03(a1,n1,y);
    my_and STEP04(a2,n2,x);
    my_or  STEP05(z,a1,a2);
Endmodule

```

//激励部分代码

```

module stimulus;
    reg X,Y;
    wire OUTPUT;
    MY_XOR xor01(OUTPUT,X,Y);
    initial

```

```

begin
    $monitor($time,"X=%b,Y=%b --- OUTPUT=%b\n",X,Y,OUTPUT);
end
initial
begin
    X = 1'b0; Y = 1'b0;
    #5 X = 1'b1; Y = 1'b0;
    #5 X = 1'b1; Y = 1'b1;
    #5 X = 1'b0; Y = 1'b1;
end
endmodule

```

3、实验结果

观察波形图并记录之。

(二)、二进制全加器设计的 verilog 设计和仿真

一位全加器使用乘积项之和的形式可以表示为：

$$\begin{aligned} \text{sum} &= a \cdot b \cdot c_{in} + a' \cdot b \cdot c_{in}' + a' \cdot b' \cdot c_{in} + a \cdot b' \cdot c_{in}' \\ c_{out} &= a \cdot b + b \cdot c_{in} + a \cdot c_{in} \end{aligned}$$

其中 a, b 和 c_in 为输入，sum 和 c_out 为输出，只使用与门，或门，非门实现一个一位全加器，写出 Verilog 描述，限制是每个门最多只能有四个输入端。编写激励模块对其功能进行检查，并对全部的输入组合输入组合进行测试。

1、实验要求

用 Verilog HDL 语言描述一位全加器，并使用 modelsim 仿真验证结果。

```

module fulladd(sum,c_out,a,b,c_in);
    output sum,c_out;
    input a,b,c_in;
    wire s1,s2,s3,s4,a1,b1,c_in1,c1,c2,c3;

```



```

    and(s1,a,b,c_in);
    not(a1,a);
    not(b1,b);
    not(c_in1,c_in);
    and(s2,a1,b,c_in1);
    and(s3,a1,b1,c_in);
    and(s4,a,b1,c_in1);
    and(c1,a,b);
    and(c2,b,c_in);
    and(c3,a,c_in);
    or(sum,s1,s2,s3,s4);
    or(c_out,c1,c2,c3);
endmodule
//激励部分代码
module stimulus;
    reg A,B,C_IN;
    wire SUM,C_OUT;
    fulladd FA1(SUM,C_OUT,A,B,C_IN);
    initial
    begin
        $monitor($time,"A=%b,
B=%b,C_IN=%b,---C_OUT=%b,SUM=%b\n",A,B,C_IN,C_OUT,SUM);
    end
    initial
    begin
        A=1'd0;B=1'd0;C_IN=1'b0;
        #5 A=1'd0;B=1'd0;C_IN=1'b1;
        #5 A=1'd0;B=1'd1;C_IN=1'b0;
        #5 A=1'd0;B=1'd1;C_IN=1'b1;
        #5 A=1'd1;B=1'd0;C_IN=1'b0;
        #5 A=1'd1;B=1'd0;C_IN=1'b1;
        #5 A=1'd1;B=1'd1;C_IN=1'b0;
        #5 A=1'd1;B=1'd1;C_IN=1'b1;
    end
End

```

2、观察波形图并记录之。

实验八 时序电路的 verilog 设计和仿真

一、实验目的

- (1) 熟悉 Modelsim 软件
- (2) 掌握 Modelsim 软件的编译、仿真方法
- (3) 熟练运用 Modelsim 软件进行 HDL 程序设计开发

二、实验内容

(一)、计数器的 verilog 设计和仿真

一个同步计数器可以使用主从 JK 触发器来设计。设计一个同步计数器，其逻辑图和 JK 触发器的逻辑图如书中图所示。清零信号 clear 低电平有效，输入数据在时钟信号 clock 的上升沿被锁存，触发器在 clock 的下降沿输出；当 count_enable 信号为低电平时停止计数。写出同步计数器的 Verilog 描述和激励模块，在激励模块中使用 clear 和 count_enable 对计数器进行测试，并显示输出计数 Q[3:0]。

技术规范：

本次试验中共有三个模块，分别是 JK 主从触发器设计模块，计数器设计模块及激励模块。对于 JK 触发器，共有四个输入端 (j, k, clear, clock)，两个输出端 (q, qbar)，根据其实际电路结构做出相应的设计；对于计数器模块，有三个输入端 (clear, clock, counter_clock)，四个输出 (Q[3:0])，使用的是四个 JK 触发器和门电路组合，采用同步清零和同步脉冲构成四位同步

计数器；激励模块中根据技术模块输入端口进行相关赋值以便进行仿真观察；

1. 实验步骤：

1). 在 modulesim 软件中使用数据流建模进行四位计数器设计及输入；

2). 进行编译及仿真。

2. 源代码：

JK 触发器模块：

```
module m_c_jkff(q, qbar, J, K, clear, clock);  
output q, qbar;  
input J, K, clear, clock;  
wire a, b, c, d, y, ybar, cbar;  
assign cbar=~clock;  
assign #1 a=(J & qbar & clock & clear),  
        b=~(K & q & clock),  
        y=~(a & ybar),  
        ybar=~(y & b & clear),  
        c=~(y & cbar),  
        d=~(ybar & cbar);  
assign #1 q=~(c & qbar);  
assign #1 qbar=~(d & clear & q);  
endmodule
```

计数器模块:

```
module four_count_ff(Q, clear, clock, count_enable);  
output [3:0] Q;  
input clear, clock, count_enable;  
wire a1, a2, a3;  
assign a1=count_enable & Q[0],  
        a2=a1&Q[1], a3=a2&Q[2];  
    m_c_jkff  
m1(Q[0],, count_enable, count_enable, clear, clock);  
    m_c_jkff m2(Q[1],, a1, a1, clear, clock);  
    m_c_jkff m3(Q[2],, a2, a2, clear, clock);  
    m_c_jkff m4(Q[3],, a3, a3, clear, clock);  
endmodule
```

激励模块:

```
module stimulus;  
reg clock, clear, count_enable;  
wire [3:0] Q;  
initial  
$monitor($time, "Count Q = %b Clear = %b", Q[3:0], clear);  
    four_count_ff f1(Q, clear, clock, count_enable);  
initial  
begin  
    clear=1'b0;  
    count_enable=1'b1;  
    #10 clear=1'b1;
```

```

//#1 count_enable=1'b1;

//#50 clear=1'b0;

//#100 count_enable=1'b0;

//#100 clear=1'b0;

//#50 clear=1'b1;

//#50 count_enable=1'b1;

end

initial

begin

    clock=1'b0;

    forever #20 clock=~clock;

end

initial

    #1000 $finish;

Endmodule

```

3、观察波形图并记录之。

(二)、状态机的 verilog 设计和仿真

使用同步有限状态自动机方法设计一个电路，它的引脚 **in** 接受一位的输入流。每当检测到模式 10101 时，输出引脚 **reset** 被赋值为高电平引脚。**reset** 引脚以同步方式初始化电路。输入引脚 **clk** 用于给电路提供时钟信号。使用身边现有的任何工艺库综合该电路，优化电路，使其达到最快速度。把同样的激励应用到 RTL 和门级网表上，比较它们的输出

1. 实验步骤：

(1). 在 modulesim 软件中使用同步有限状态自动机方法设计一个电路。

(2) . 进行编译及仿真。

```
module select(in,clk,reset,match);
input in;
input clk;
input reset;
output match;
wire match;
wire [4:0] NEXT_STATE;
reg [4:0] PRES_STATE;
parameter s1=5'b00000;
parameter s2=5'b00001;
parameter s3=5'b00010;
parameter s4=5'b00101;
parameter s5=5'b01010;
parameter s6=5'b10101;

function [5:0] fsm;
input fsm_in;
input [4:0] fsm_PRES_STATE;

reg fsm_match;
reg [4:0] fsm_NEXT_STATE;

begin
    case(fsm_PRES_STATE)
    s1:
        begin
            if(fsm_in==1'b1)
                begin
                    fsm_match=1'b0;
                    fsm_NEXT_STATE=s2;
                end
            else if(fsm_in==1'b0)
                begin
                    fsm_match=1'b0;
                    fsm_NEXT_STATE=s1;
                end
        end
    end
end
```

```

    end
s2:
    begin
        if(fsm_in==1'b1)
            begin
                fsm_match=1'b0;
                fsm_NEXT_STATE=s2;
            end
        else if(fsm_in==1'b0)
            begin
                fsm_match=1'b0;
                fsm_NEXT_STATE=s3;
            end
        end
    end
s3:
    begin
        if(fsm_in==1'b1)
            begin
                fsm_match=1'b0;
                fsm_NEXT_STATE=s4;
            end
        else if(fsm_in==1'b0)
            begin
                fsm_match=1'b0;
                fsm_NEXT_STATE=s1;
            end
        end
    end
s4:
    begin
        if(fsm_in==1'b1)
            begin
                fsm_match=1'b0;
                fsm_NEXT_STATE=s2;
            end
        else if(fsm_in==1'b0)
            begin
                fsm_match=1'b0;
                fsm_NEXT_STATE=s5;
            end
        end
    end

```

```

        end
    end
s5:
    begin
        if(fsm_in==1'b1)
            begin
                fsm_match=1'b0;
                fsm_NEXT_STATE=s6;
            end
        else if(fsm_in==1'b0)
            begin
                fsm_match=1'b0;
                fsm_NEXT_STATE=s1;
            end
        end
    end
s6:
    begin
        if(fsm_in==1'b1)
            begin
                fsm_match=1'b0;
                fsm_NEXT_STATE=s2;
            end
        else if(fsm_in==1'b0)
            begin
                fsm_match=1'b0;
                fsm_NEXT_STATE=s5;
            end
        end
    endcase
    fsm={fsm_match,fsm_NEXT_STATE};
end
endfunction
assign {match,NEXT_STATE}=fsm(in,PRES_STATE);
always @(posedge clk)
begin
    if (reset==1'b1)
        PRES_STATE<=s1;
    else

```



```

        PRES_STATE<=NEXT_STATE;
    end
endmodule

```

激励模块：

```

module stimulus;
reg clk;
reg in;
reg reset;
wire match;
select se(in,clk,reset,match);
initial
begin
clk=0;
in=0;
reset=1;
#100 reset=0;
#30 in=1;
#40 in=1;
#40 in=0;
#40 in=1;
#40 in=0;
#40 in=1;
#40 in=1;
#40 in=1;
#40 in=0;
#40 in=0;
#40 in=1;
#40 in=1;
#40 in=0;
#40 in=0;
#40 in=1;
#40 in=0;
#40 in=1;
#40 in=0;
#40 in=1;

```

```
#40 in=1;  
#50 $finish;  
end  
  
always  
begin  
#20 clk=~clk;  
end  
endmodule
```

(3). 观察波形图并记录之。