



# 逻辑与计算机设计基础课 程实验（实践）指导书

易丛琴 编著

上海海洋大学海洋智能信息实验教学示范中心

# 实验一 编码器设计

## 一、实验目的

- 1、学习译码器的原理
- 2、学会使用 logisim 制作译码器
- 3、8-3 译码器的实现

## 二、实验环境

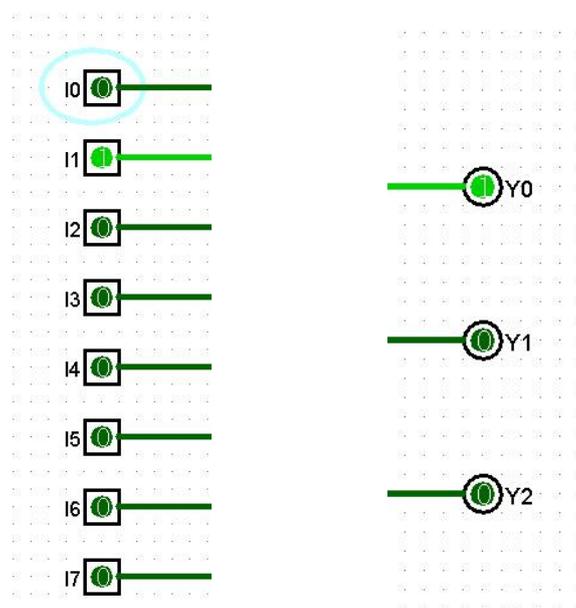
Logisim 软件

## 三、实验内容

实现从十进制的 0, 1, 2, 3, 4, 5, 6, 7 编码为三位二进制，将一位十进制 0-7 输入转化为 000-111 二进制输出。

## 四、实验步骤

- 1、设计 8-3 编码器输入输出端，用 I0-I7 分别表示输入的 8 个十进制数 0 到 7，用 Y2Y1Y0 表示输出的 3 位二进制数



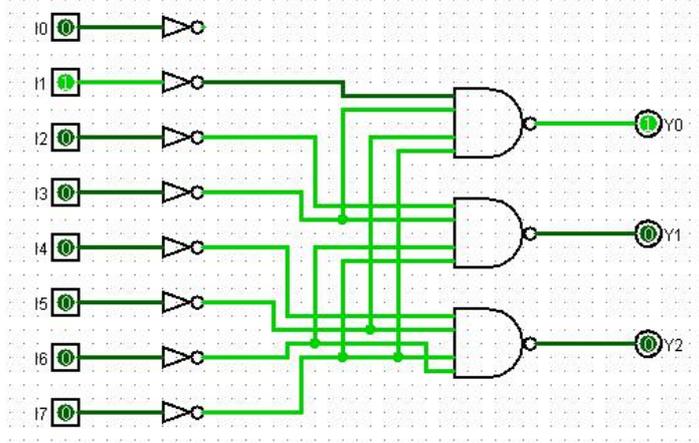
- 2、根据实现功能的真值表，写出逻辑表达式并用卡诺图化简

Output: Y0  
 $\overline{\overline{I1 I3 I5 I7}}$

Output: Y1  
 $\overline{\overline{I2 I3 I6 I7}}$

Output: Y2  
 $\overline{\overline{I4 I5 I7 I6}}$

3、根据逻辑表达式，用 logisim 画出电路图



五、实验报告要求：

实验报告参考模板如下：



实验报告

题目： \_\_\_\_\_

学院：信息学院

专业：

班级：

学号：

姓名：

年 月 日

### 一、实验目的（宋体四号加粗）

正文（正文 宋体小四，1.5倍行距）

### 二、实验环境（宋体四号加粗）

### 三、实验内容（宋体四号加粗）

四、实验步骤（图文方式叙述）（宋体四号加粗）

五、实验结果及分析（遇到的问题与解决）（宋体四号加粗）

六、实验体会（宋体四号加粗）

## 实验二 38 译码器设计

### 一、实验目的

理解译码器的原理，使用 logisim 设计实现 38 译码器

### 二、实验环境

Logisim 软件

### 三、实验内容（宋体四号加粗）

实验一完成了 83 编码器，本实验需完成 38 译码器，从三位二进制数易码为 0-7 十进制自然数。

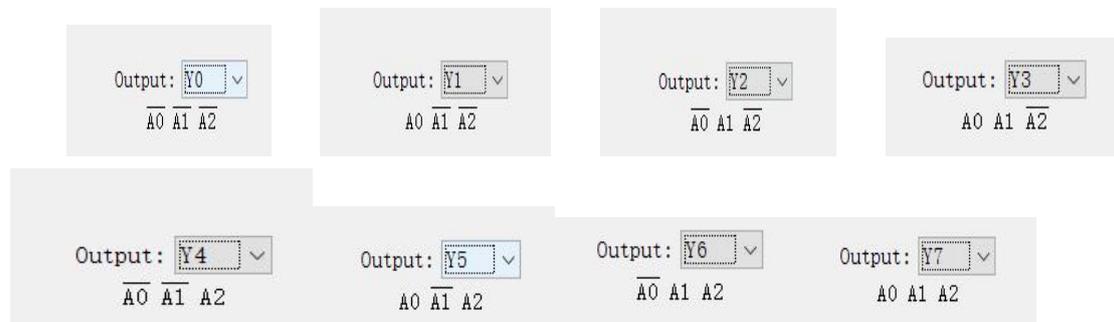
#### 四、实验步骤

1.确定输入输出：输入三位二进制数，用 A2,A1,A0 表示，输出八个十进制数使用 Y7-Y0 表示 7-0.

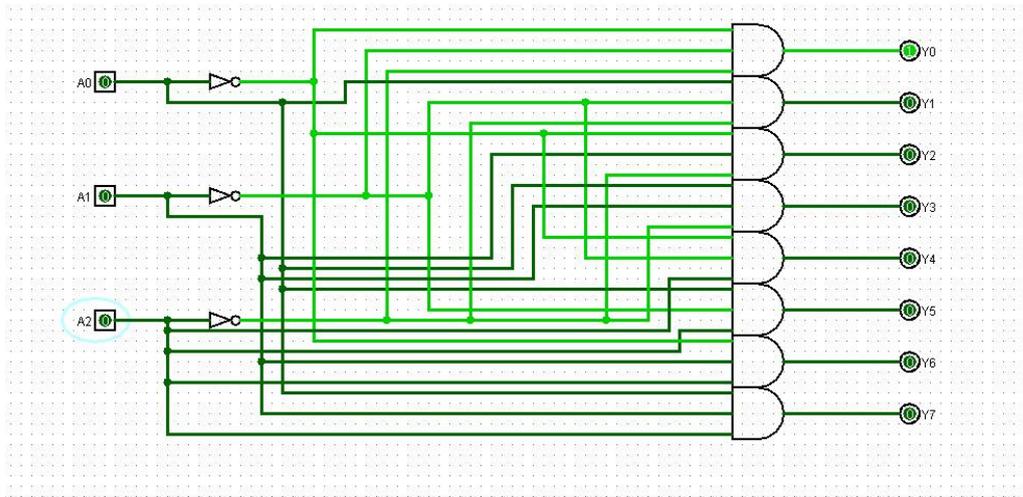
2.写出最简逻辑表达式

A0	A1	A2	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	0	0	0	1	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	0	0	0	1	0
1	0	0	0	1	0	0	0	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	1	0	0	0	0
1	1	1	0	0	0	0	0	0	0	1

根据真值表画卡诺图化简，得到：



3. 用 logisim 画出电路图



## 五、实验报告要求

同实验一

## 实验三 显示译码器设计

对编码器的理解更进一步，能够使用字符编码器连接 7 段数码管显示器显示 0-9。

### 一、实验目的

## 二、实验环境

Logisim 软件

## 三、实验内容

1、使用 logisim 绘制 a, b, c, d, e, f, g 的编码器，输入 4 位 8421BCD 码，因为后面需要连接显示器显示 0-9，所以需要 4 位。

2、将 a, b, c, d, e, f, g 的编码器合为一个 seg 编码器，用于输出 8421BCD 码对应 0-9 的段码。

## 四、实验步骤（描述详细过程）

1.确定输入输出：A0,A1,A2,A3，输出 a

2.写出逻辑表达式

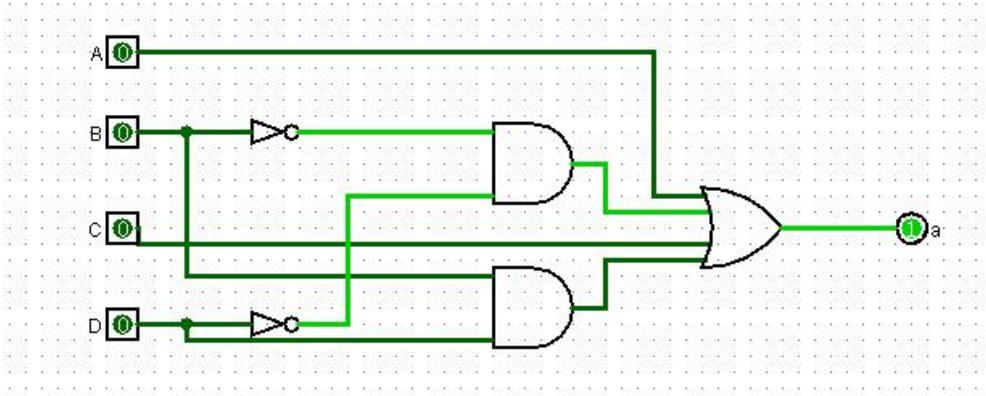
A	B	C	D	a
0	0	0	0	1
0	0	0	1	0
0	0	1	0	1
0	0	1	1	1
0	1	0	0	0
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

从 1010 开始后面 a 是 x 不定形式

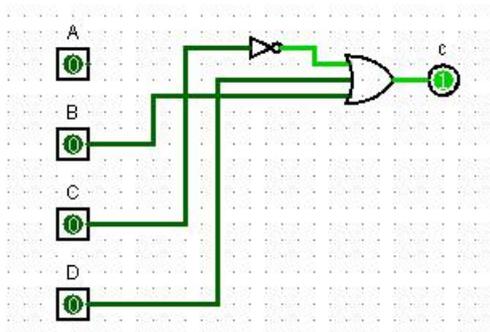
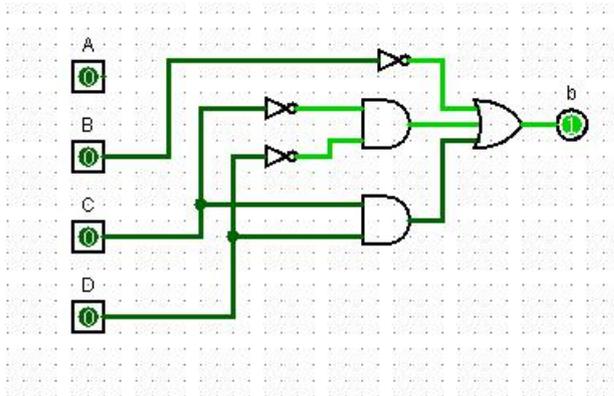
		C, D			
		00	01	11	10
A, B	00	1	0	1	1
	01	0	1	1	1
	11	1	1	1	1
	10	1	1	1	1

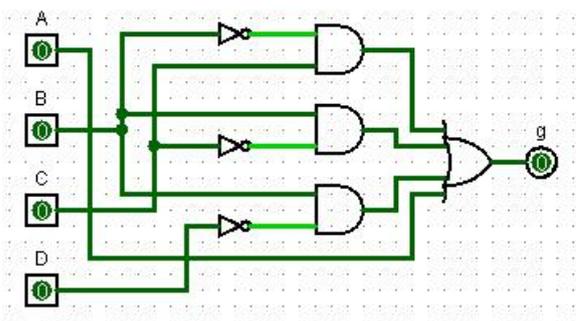
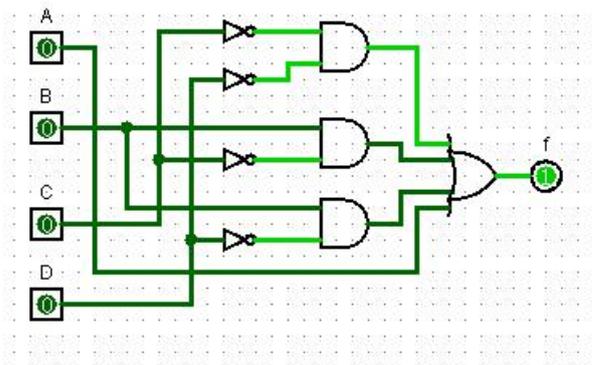
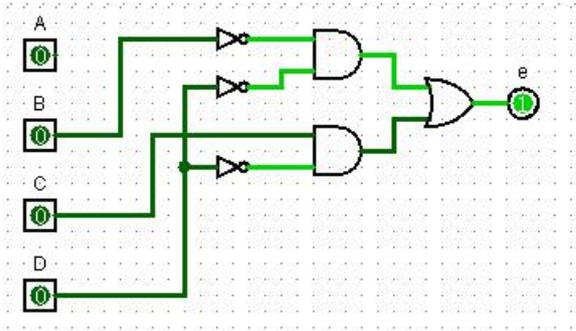
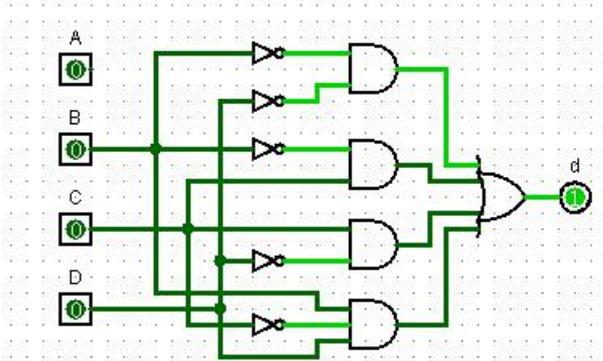
$\bar{B} \bar{D} + C + B D + A$

3. 画出逻辑电路图

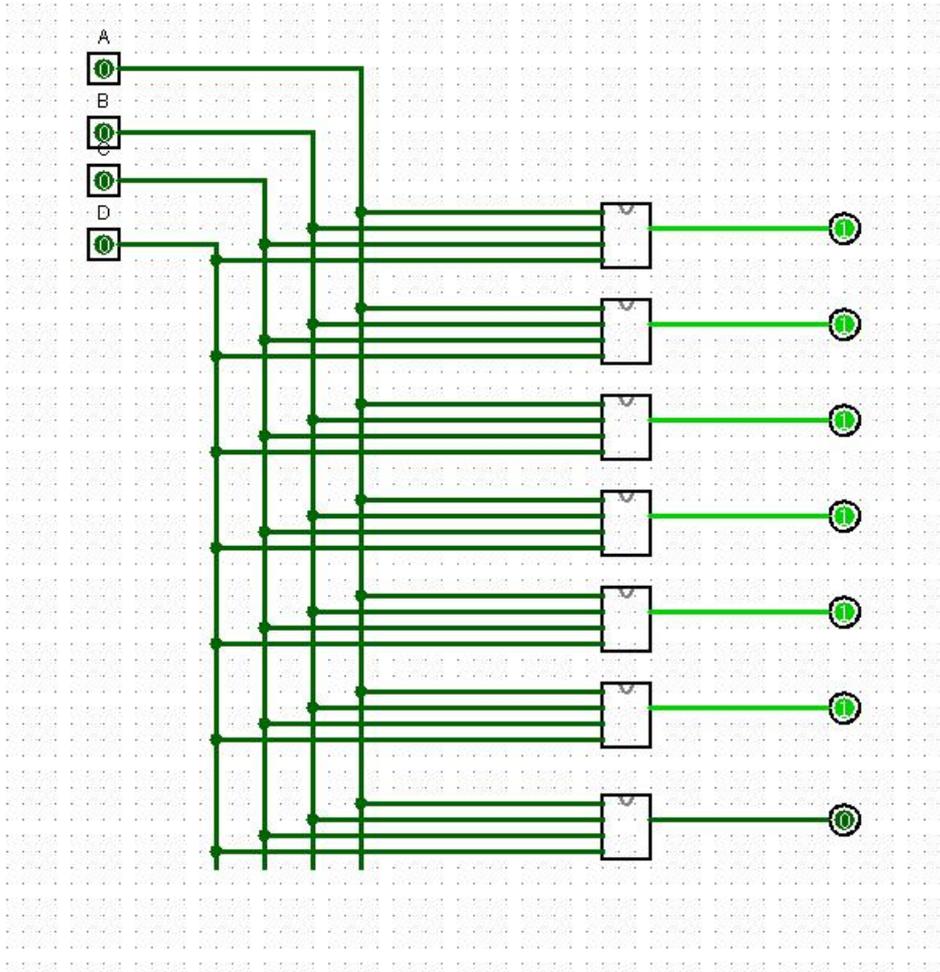


b,c,d,e,f,g 类似，逻辑电路图如下：

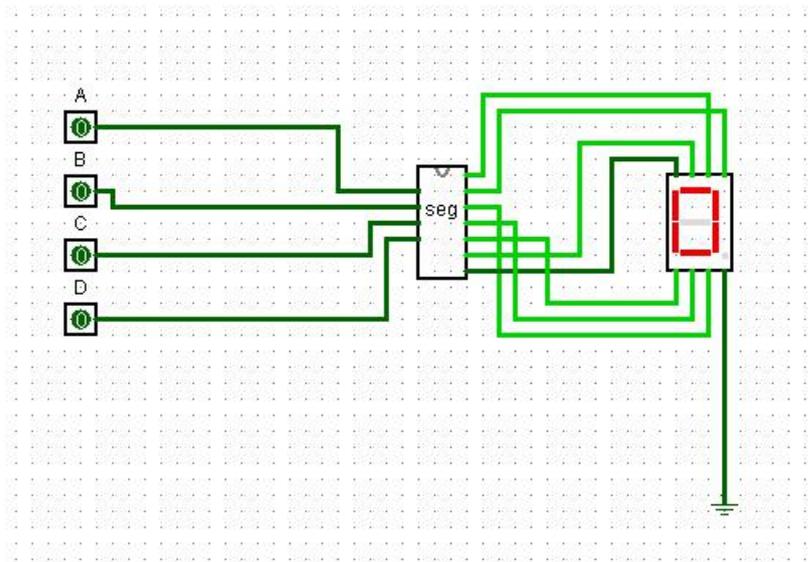




4.将字符编码器结合在一起，包装为 seg



5.用 seg 驱动 7 段数码管



## 五、实验报告要求

同实验一

## 实验四 总线控制设计

### 一、实验目的

理解总线的作用，实现控制器的设计，将编码器、译码器、显示器通过总线连接起来。

### 四、实验环境

Logisim 软件

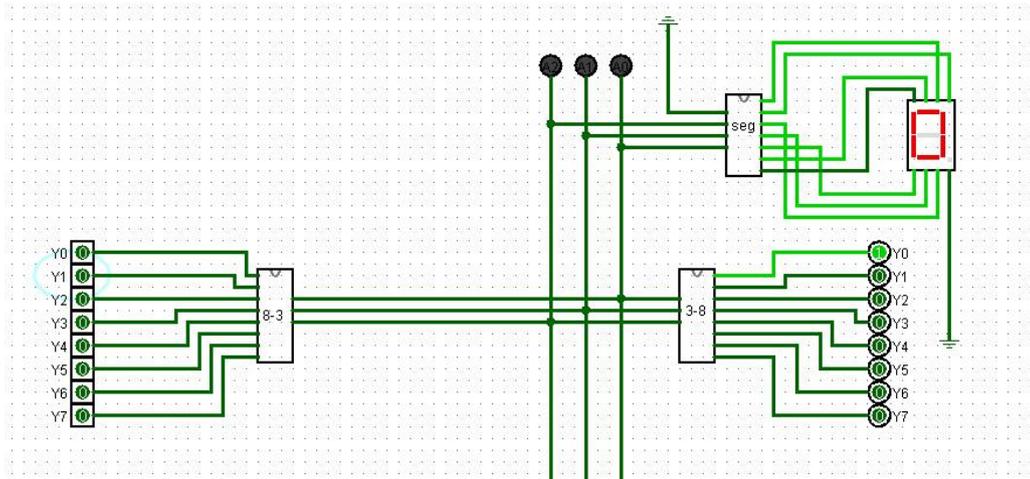
### 三、实验内容

1、将实验一 83 编码器，实验二 38 译码器，实验三数字显示器，用总线结合在一起。

2、设计用两个 83 编码器连接到总线上，怎么才能控制显示哪一个编码器的结果。

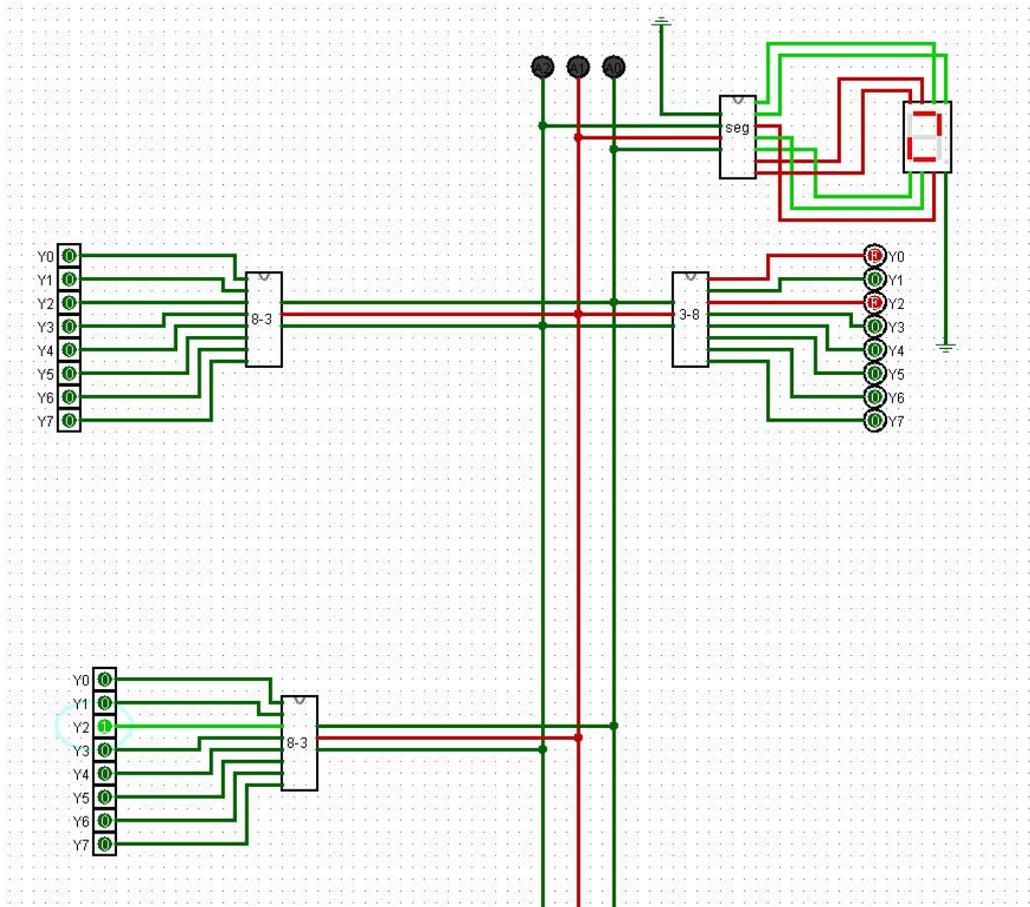
### 四、实验步骤

1.连接编码译码和数字显示器



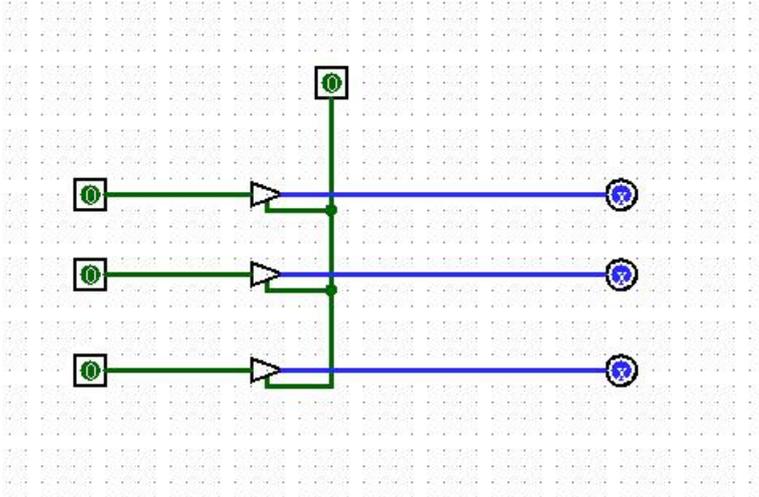
因为之前做的编码器是 0-7，所以 seg 的最高位接地就可以了，数码管显示 0-7

2.测试将两个编码器连接到总线上



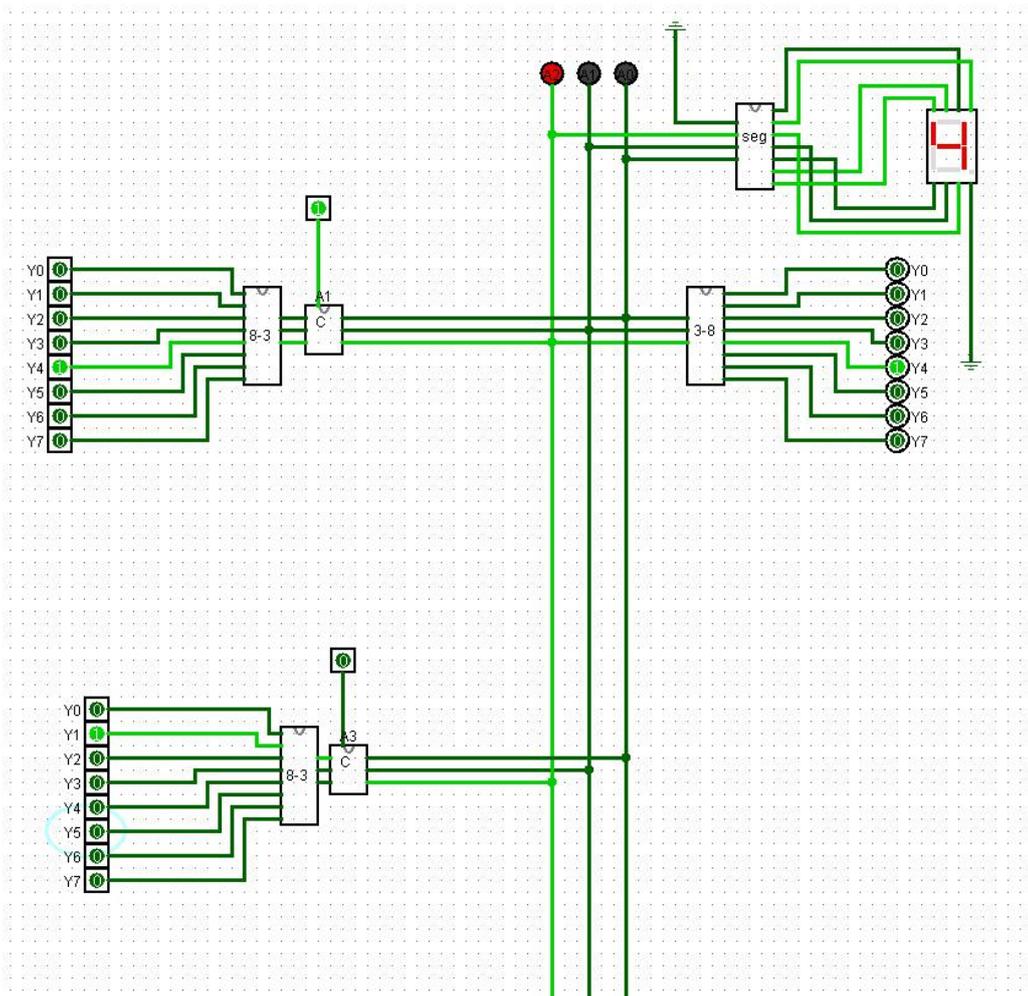
发现数码管不能正常显示，因为两个编码器的电路信号都接到总线上，会引起混乱，接下在设计控制器。

3.控制器



用三态门进行设计，因为总线是三位二进制，这里也只需要控制三位即可，当控制按钮为 1 时，数据才能通过。

#### 4.将控制器用于总线电路中



## 五、实验报告要求：

同实验一

## 实验五 寄存器设计

### 一、实验目的

理解寄存器的构成  
完成寄存器的设计

### 六、实验环境

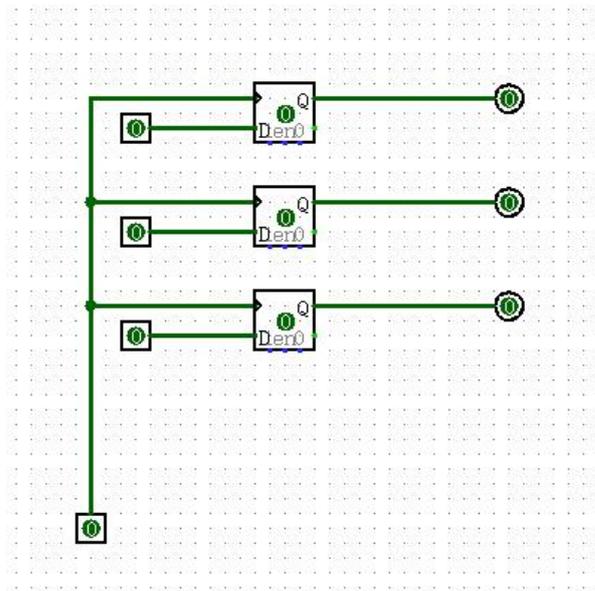
Logisim 软件

### 三、实验内容

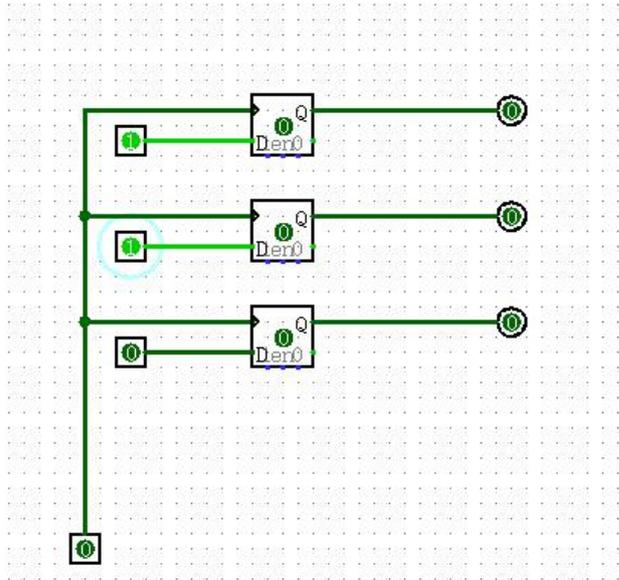
使用 logisim 利用 3 个 D 触发器制作一个 3 位的寄存器。

### 四、实验步骤

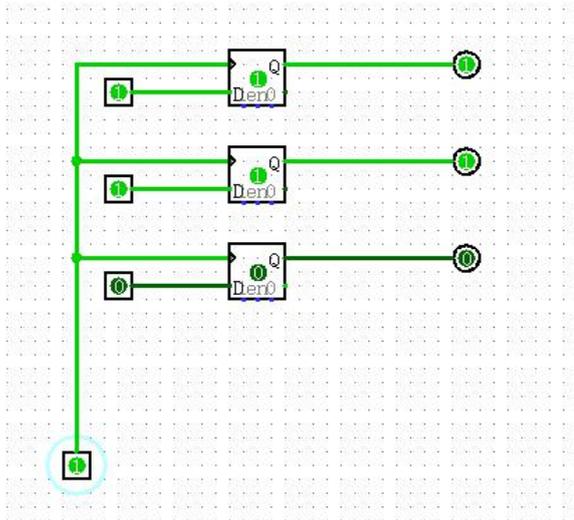
1、理解 D 触发器原理：CP 上升沿有效，并且结果能够暂存在寄存器中，并使用 logisim 完成 3 位寄存器电路



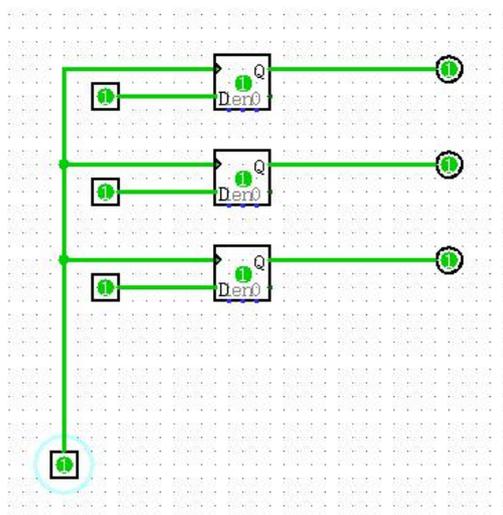
2、测试寄存器，若 D 不变时寄存器不工作



3、若 D 从 0 到 1 变化时，D 触发器工作，将输入的内容暂存起来



4、改变输入再次尝试，当且仅当上升沿到来时，暂存的数值才会变化



## 五、实验报告要求:

同实验一

## 实验六 加法器设计

### 一、实验目的

理解加法器的构成

能够设计多位全加器

### 七、实验环境

Logisim 软件

### 三、实验内容

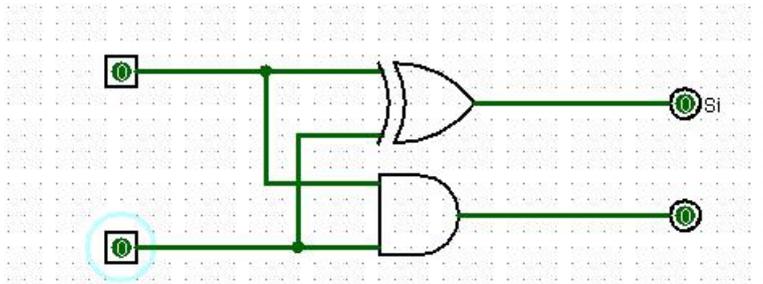
- 1、设计半加器、全加器
- 2、设计三位加法器
- 3、将加法器连接到电路中
- 4、将加法器连接到显示译码器上

### 四、实验步骤

#### 1.半加器

半加器不考虑低位来的进位， $S=A$  异或  $B$ ， $C=AB$

a	b	Si	x
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

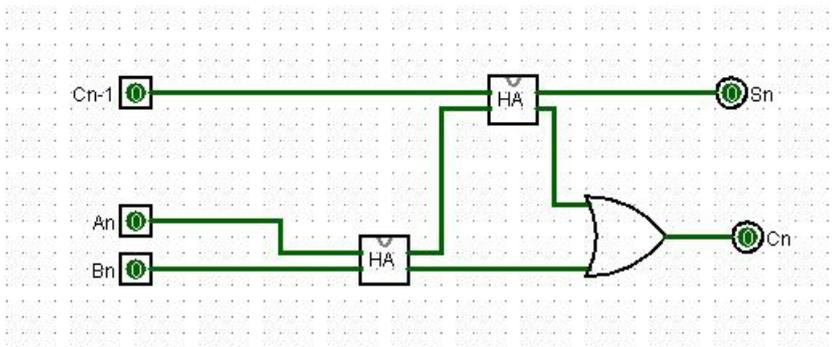


## 2.全加器

全加器考虑低位来的进位，可由两个半加器构成， $C_n = (A_n \text{ 异或 } B_n)C_{n-1} + A_n B_n$

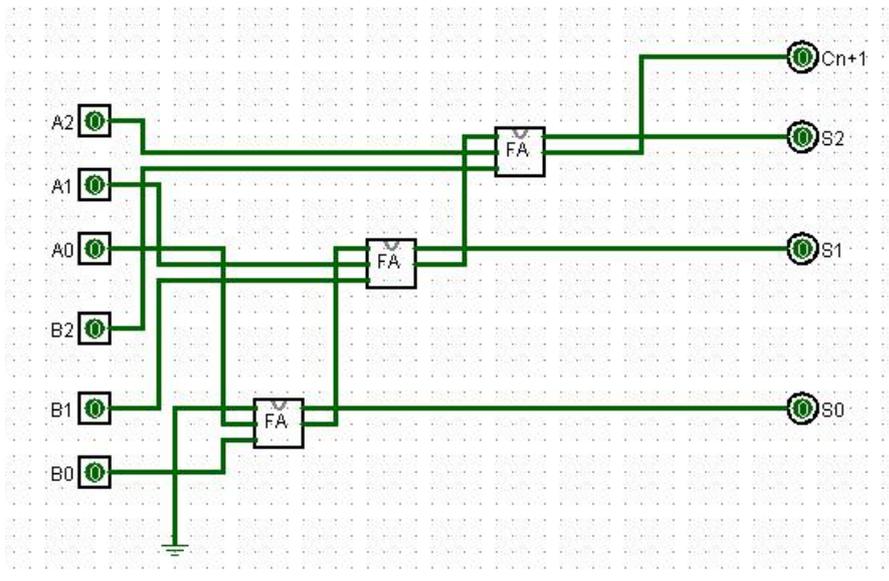
$S_n = A_n \text{ 异或 } B_n \text{ 异或 } C_{n-1}$ .

$C_{n-1}$	$A_n$	$B_n$	$S_n$	$C_n$
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

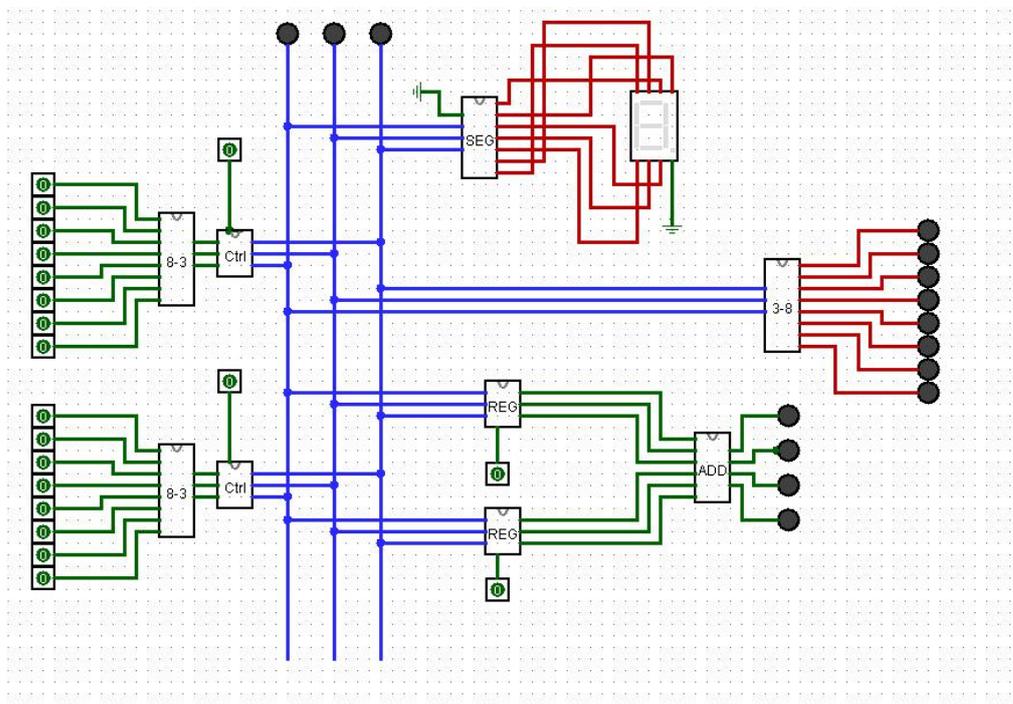


## 3.三位加法器

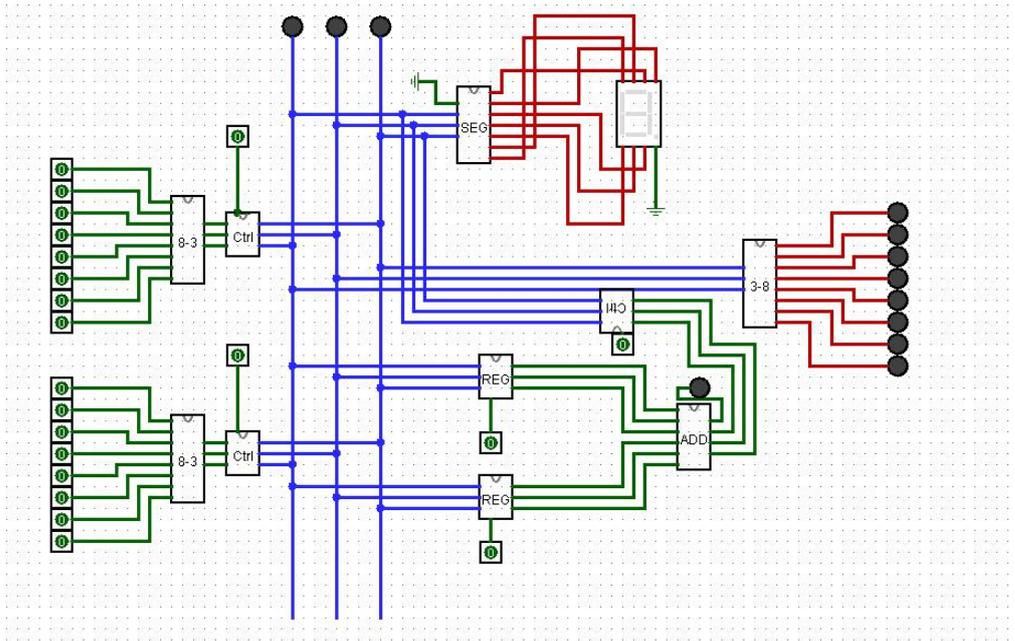
将三个全加器连接起来组成，默认最低位进位为 0



用全加器设计三位加法器



用 LED 灯验证加法器



加法器连接到显示译码器上

## 五、实验报告要求：

同实验一

## 实验七 手动加法验证

### 一、实验目的

利用寄存器，加法器，之前实验做的总线控制制作可以控制的三位加法器。

### 二、实验环境

Logisim 软件

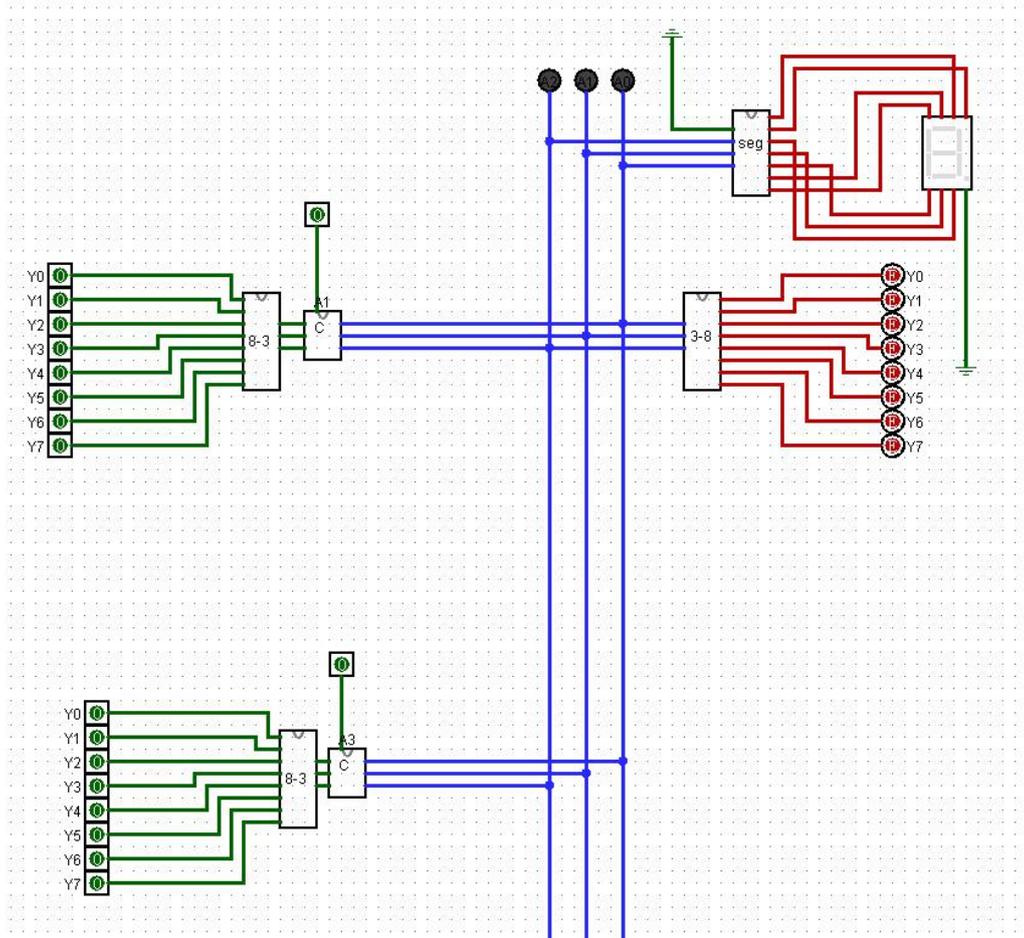
### 三、实验内容

将之前实验完成的编码器、译码器、寄存器，加法器融入到总线

控制电路中，手动验证计算机是否正确。

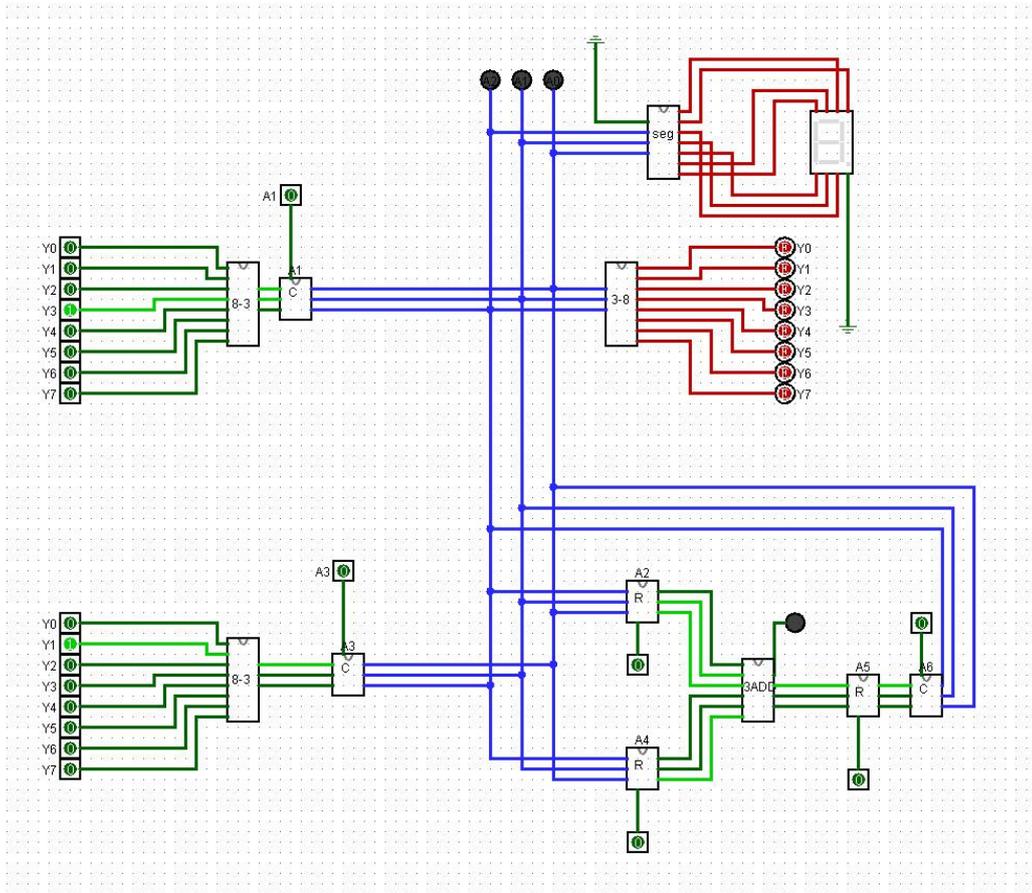
## 四、实验步骤

### 1.之前做的总线控制



这里连个 83 编码器的输出作为被加数和加数，但是电路转瞬即逝，需要寄存器保存下来，然后经过加法器运算求和，再存放在寄存器中，由控制器控制输出。

### 2.整体的结构



### 3.手动验:3+1=4

步骤：打开 A1-->打开 A2 存储被加数,关掉 A1---->打开 A3---->打开 A4 存储加数，关掉 A3---->打开 A5 存储计算结果---->打开 A6 输出和。

最终结果显示图：



## Logisim 软件

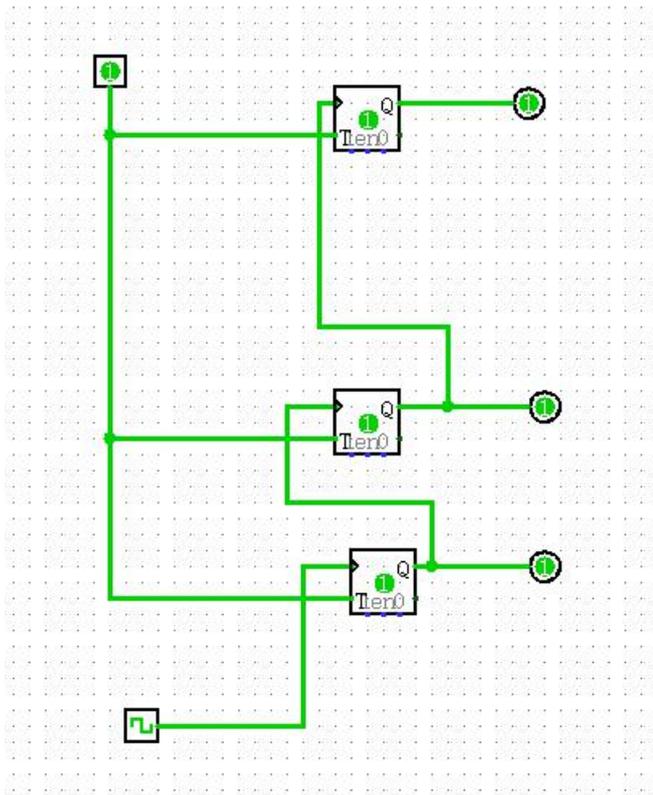
### 三、实验内容

- 1、设计三位二进制计数器
- 2、利用计数器设计 0-7 八个输出的跑马灯

### 四、实验步骤

#### 1.设计计数器

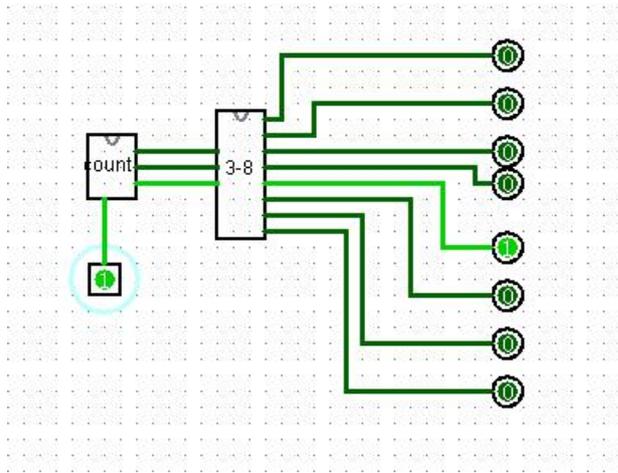
利用 T 触发器制作三位计数器



使用异步时序电路，CP 由 logisim 的时钟模拟，T0, T1,T2 输入为 1 时，循环计数开始。

#### 2. 利用计数器设计跑马灯

计数器输出的是三位二进制，因此需要连接一个 38 译码器



## 五、实验报告要求

同实验一

## 实验九 硬逻辑的自动实现

### 一、实验目的

实现自动加法控制电路

### 二、实验环境

Logisim 软件

### 三、实验内容

将跑马灯融入总线控制的手动加法系统中实现自动计算四、实验步骤

### 四、实验步骤

将手动加法的实现系统利用跑马灯，设计好控制逻辑，画出控制元件的真值表，根据真值表将跑马灯的输出电路按控制按钮的打开顺序相连，实现自动控制。

自主完成。

## 五、实验报告要求

同实验一