

# 组合逻辑部件设计——实验报告

孙启翔 241220098

## 1 译码器实验

### 1.1 整体方案设计

G为使能端，A，B，C为传入信号，进行 $2^3 = 8$ 的运算

### 1.2 logisim电路图

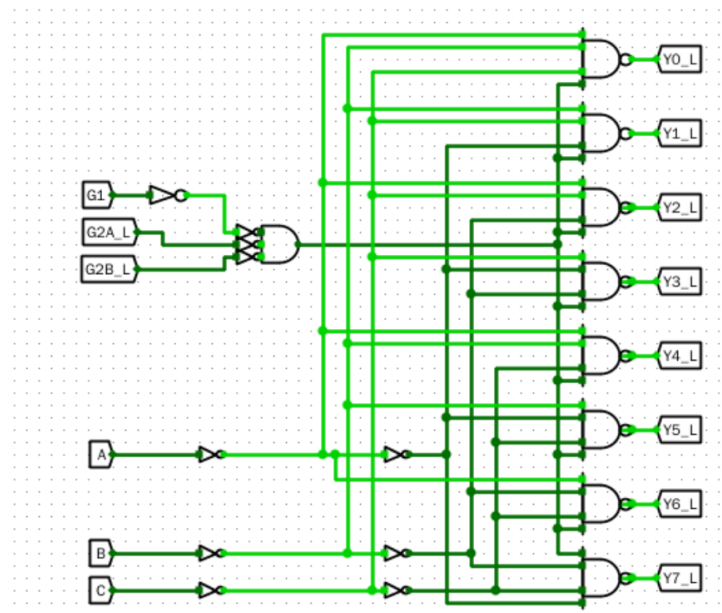


Figure 1: 译码器实验

图 1-1-1 所示为 74LS148 的引脚图。图中展示了 16 个引脚的排列，包括输入引脚（G1, G2, G2B, G1B, G1A, G2A, G2B, G1B, G1A, G2A, G2B, G1B, G1A, G2A, G2B, G1B）、输出引脚（Y0, Y1, Y2, Y3, Y4, Y5, Y6, Y7, Y7L, Y7H, Y7L, Y7H, Y7L, Y7H, Y7L, Y7H）以及电源引脚（VCC, GND）。图中还标注了“输入引脚”和“输出引脚”。

图 1 为 12 路输入、12 路输出、12 路地址的 128K 静态 CMOS 存储器芯片的管脚图。图中各管脚的功能如下：

Figure 2: 测试结果

需要注意看线不要连错了，线很多很杂很容易错（

## 2.1 整体方案设计

其实是译码器的反过来做一遍

## 2.2 logisim电路图

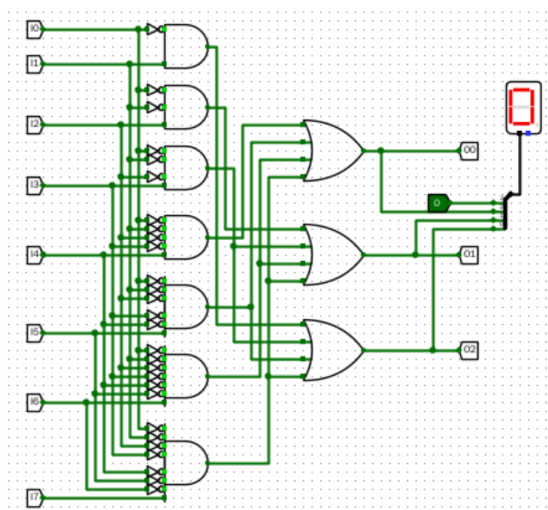
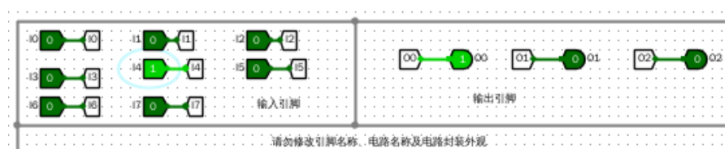
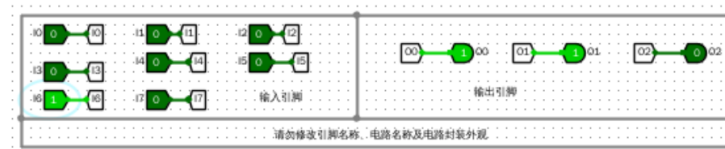


Figure 3: 编码器实验

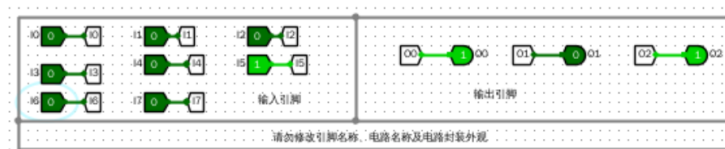
## 2.3 仿真测试图



(a) 图 1



(b) 图 2



(c) 图 3

Figure 4: 测试结果

## 2.4 错误现象即分析

最后计数板上的最高位直接给常量0就行，因为一共只有8位

# 3 4位加法器实验

## 3.1 整体方案设计

全加器就是看图连线，总体电路就是逐位计算，上一个全加器的Cout是下一个全加器的Cin

### 3.2 logisim电路图

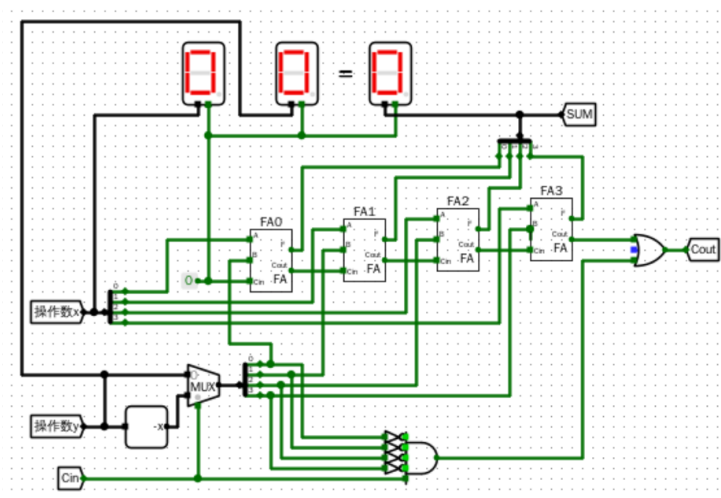
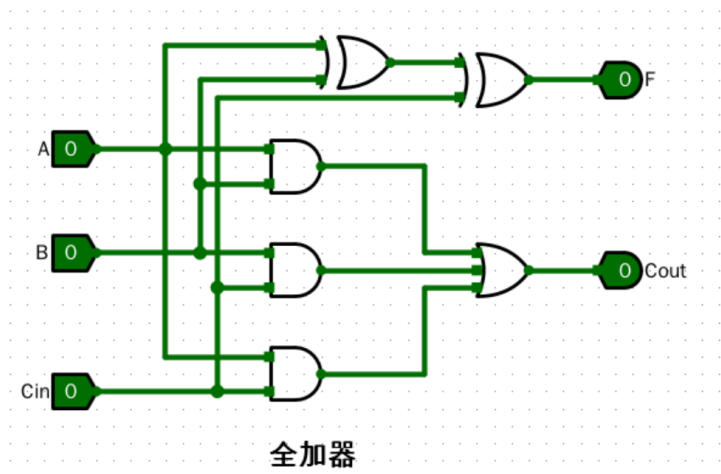


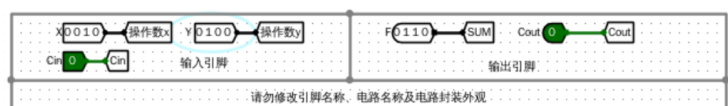
Figure 5: 4位全加器实验



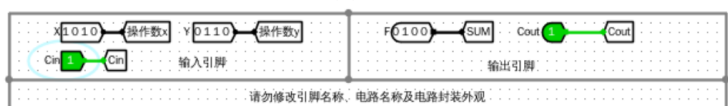
全加器

Figure 6: 全加器

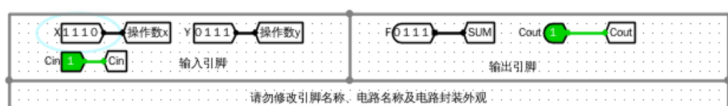
### 3.3 仿真测试图



(a) 图 1



(b) 图 2



(c) 图 3

Figure 7: 测试结果

### 3.4 错误现象即分析

在减法运算时，我的实现并不能处理像 $x-0$ 这种情况，因为0的补码实际上应该是10000而将第一个作为进位，但我的实现不能做到，所以在多少减0这里我做了一个特判

## 4 汉明码纠错实验

### 4.1 整体方案设计

在汉明码中，第1，2，4位作为校验位用来检测传输的数据是否有错误，并根据实验手册给出的逻辑代码来完成电路

## 4.2 logisim电路图

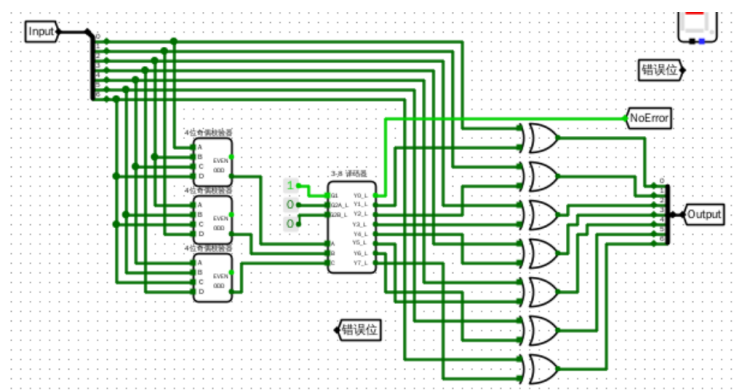


Figure 8: main

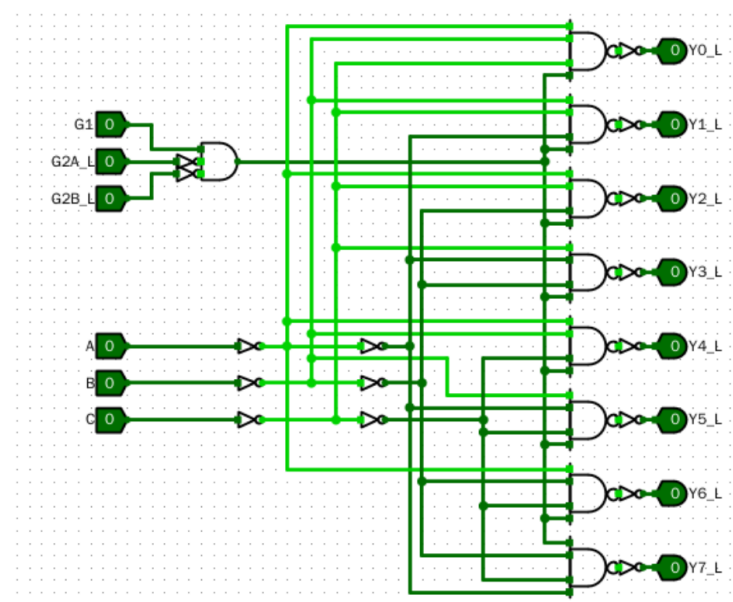


Figure 9: 3-8译码器

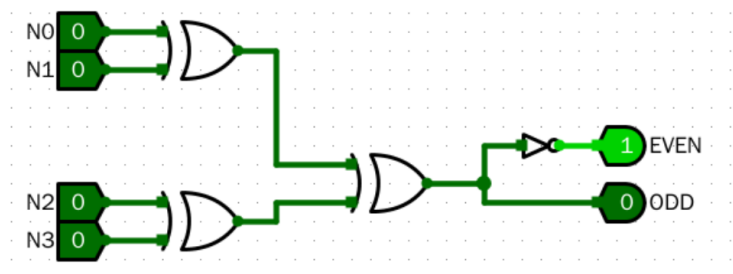
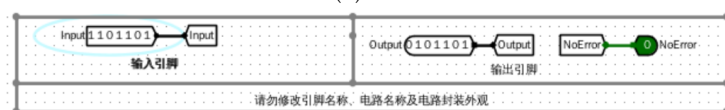


Figure 10: 4位奇偶校验器

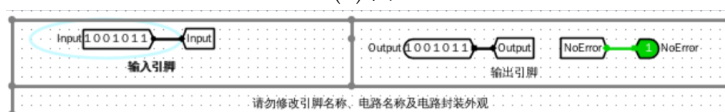
### 4.3 仿真测试图



(a) 图 1



(b) 图 2



(c) 图 3

Figure 11: 测试结果

### 4.4 错误现象即分析

连线是很难做到规整的这道题，尤其是异或门那一部分，要特别小心

## 5 8位桶形移位器实验

### 5.1 整体方案设计

通过L/R和A/L控制移动的方式（算数or逻辑）以及左移还是右移，通过shamt控制移动的位数



## 5.2 logisim电路图

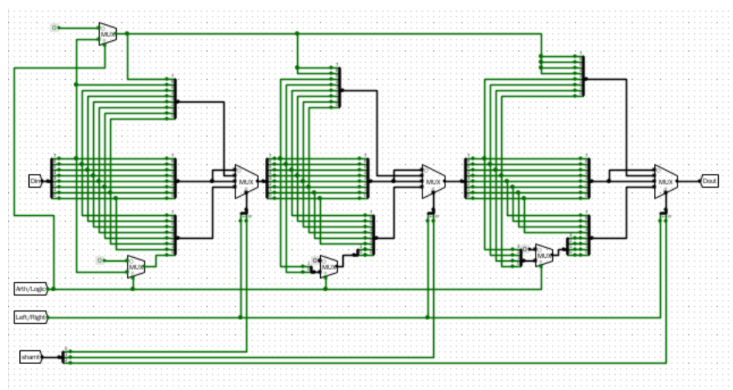
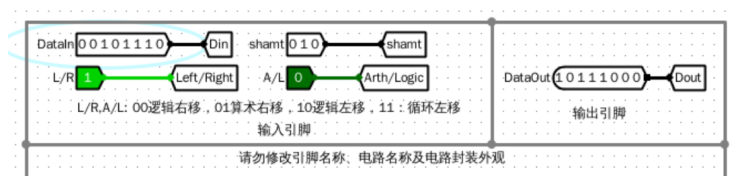
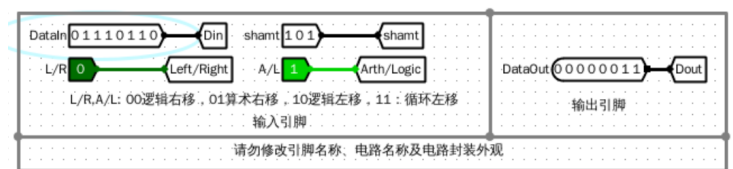


Figure 12: 整体实现

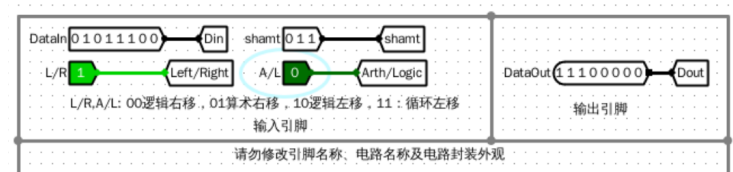
## 5.3 仿真测试图



(a) 图 1



(b) 图 2



(c) 图 3

Figure 13: 测试结果

## 5.4 错误现象即分析

分线器的连接情况特别复杂且繁琐，需要根据输出来逆推电路中的错误

## 6 思考题

### 6.1 修改实验中的加法器电路，生成进位标志CF、溢出标志OF、符号标志SF和结果为零标志ZF

$$CF = Cout$$

$$OF = \overline{Cin} \cdot Cout + Cin \cdot \overline{Cout}$$

$$SF = F_3$$

$$ZF = \overline{F_1} \cdot \overline{F_2} \cdot \overline{F_3} \cdot \overline{F_4}$$

### 6.2 通过带标志位加法器输出 (X>Y) 等于和(X<Y)小于的信号

1. 借助已完成的四位加法器，若输入X，Y满足 $Cin \cdot \overline{Cout}$ 为真，则X<Y
2.  $Cin \cdot Cout \cdot Sum$ 为真，则x>y
3.  $Cin \cdot Cout \cdot \overline{Sum}$ 为真，则X=Y

### 6.3 不使用加法器直接使用逻辑门电路实现4位无符号二进制数比较器，输出大于和小于二个结果

逐位进行比较，若第四位X与Y不相等则直接输出结果，若相等再比较第三位，依此类推，直至第一位，若仍相等则X=Y

### 6.4 如何使用8位桶形移位器扩展到32位桶形移位器

将shamt改为5位，将Din和Dout改为32位，电路中间再接2个类似结构的多路选择器用来处理 $2^3, 2^4$ 的位移