中国科学技术大学计算机学院

《数字电路实验》报告

实验题目：简单组合逻辑电路

学生姓名：陈鸿绪

学生学号：PB21000224

完成日期：10.16

计算机实验教学中心制

2020 年 09 月

【实验题目】

简单组合逻辑电路

【实验目的】

掌握 Logisim 基本用法；

进一步熟悉 Logisim 更多功能；

用 Logisim 设计组合逻辑电路并进行仿真；

学习 Verilog 初级语法；

【实验环境】

vlab.ustc.edu.cn(创建虚拟机，实验基本配置齐全)

Logisim 仿真工具、Verilog IDE

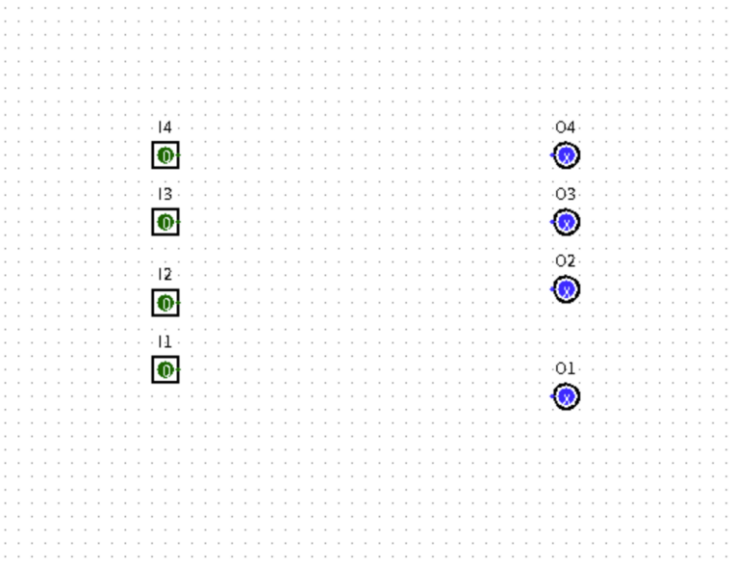
【实验过程】

Step1：用真值表自动生成电路

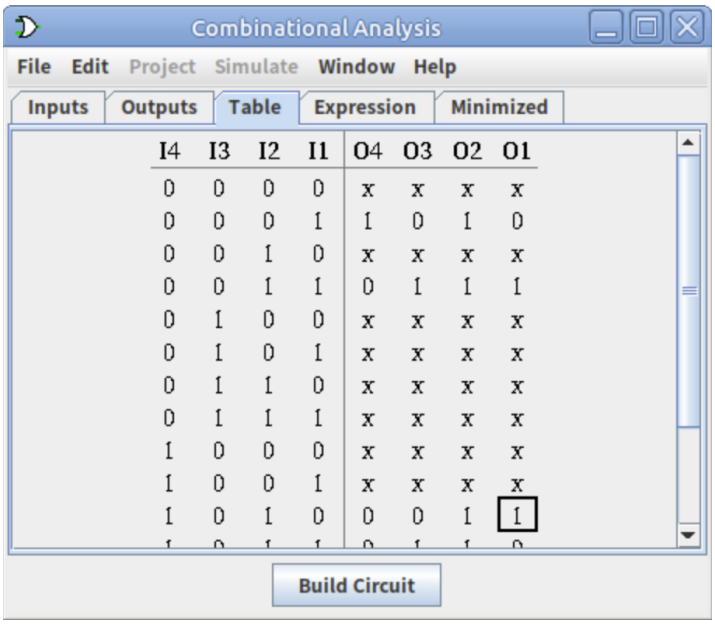
以下面真值表为例，利用 Logisim 自动生成逻辑电路

Input	output
0001	1010
0011	0111
1010	0011
1011	0110
1111	0101

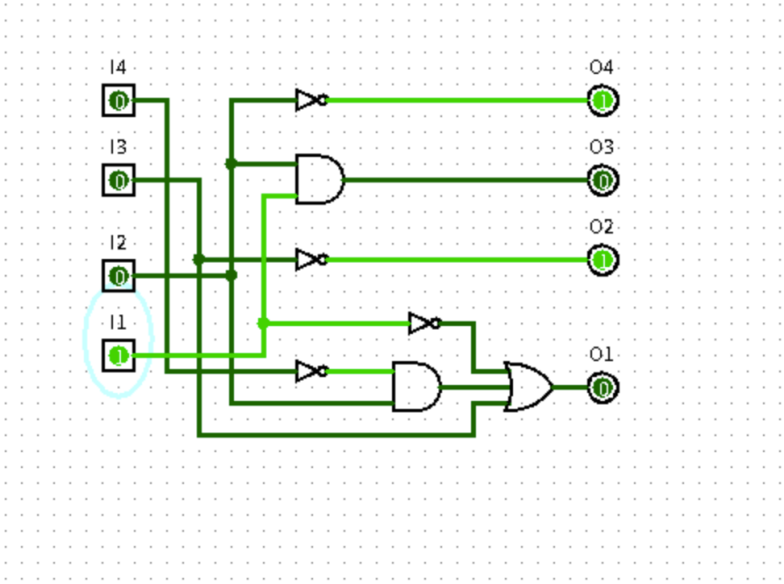
首先在 Logisim 中新建“生成电路”电路图，在电路图中放置输入引脚，输入数目与引脚数目相同，按同样的方式放置输出引脚。给所有引脚标上标号，并按高低位顺序排列。如下图所示：



编辑真值表：



生成电路后：



Step2：用表达式生成电路图

由下面的真值表：

输入								输出		
i7	i6	i5	i4	i3	i2	i1	i0	y2	y1	y0
1	x	x	x	x	x	x	x	1	1	1
0	1	x	x	x	x	x	x	1	1	0
0	0	1	x	x	x	x	x	1	0	1
0	0	0	1	x	x	x	x	1	0	0
0	0	0	0	1	x	x	x	0	1	1
0	0	0	0	0	1	x	x	0	1	0
0	0	0	0	0	0	1	x	0	0	1
0	0	0	0	0	0	0	1	0	0	0

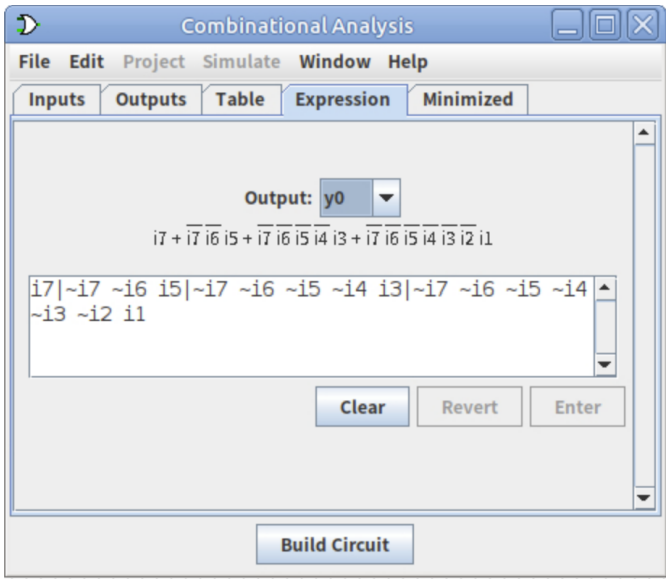
根据真值表，我们可以很快的写出各输出信号的表达式：

y2 = i7|~i7 i6|~i7 ~i6 i5|~i7 ~i6 ~i5 i4

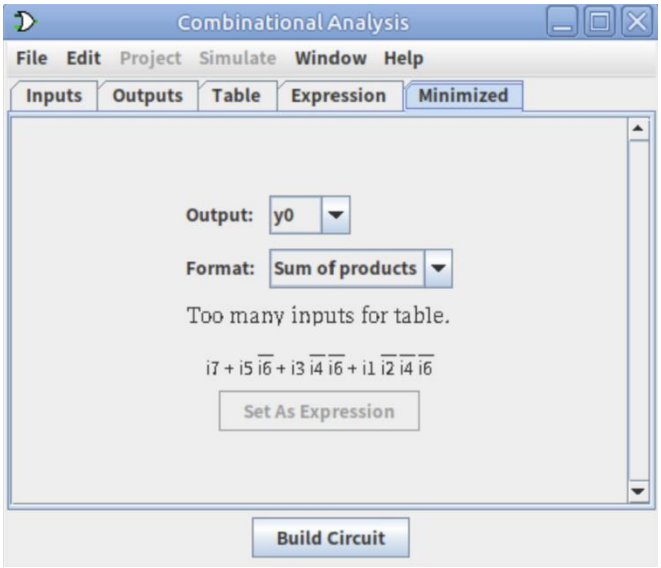
y1 = i7|~i7 i6 |~i7 ~i6 ~i5 ~i4 i3 |~i7 ~i6 ~i5 ~i4 ~i3 i2

y0 = i7|~i7 ~i6 i5|~i7 ~i6 ~i5 ~i4 i3|~i7 ~i6 ~i5 ~i4 ~i3 ~i2 i1

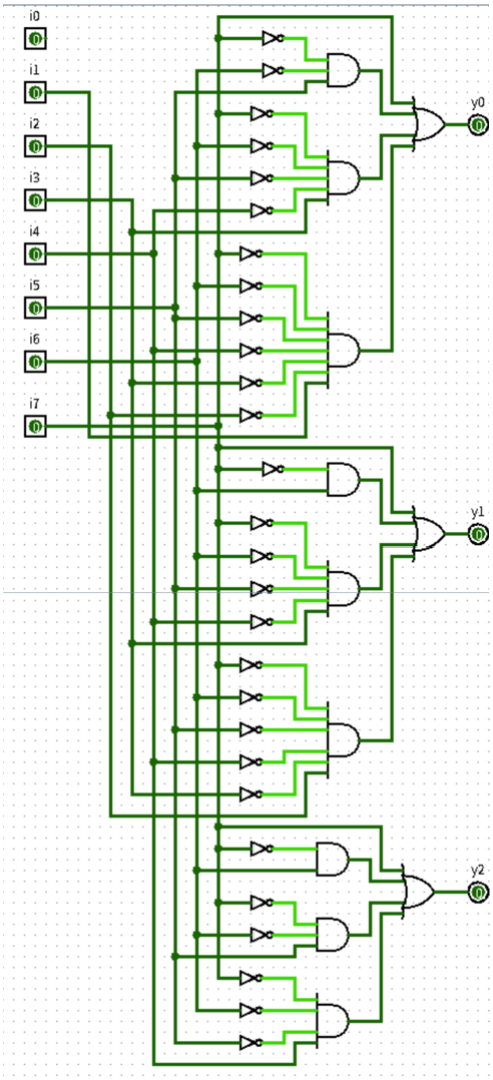
利用表达式生成



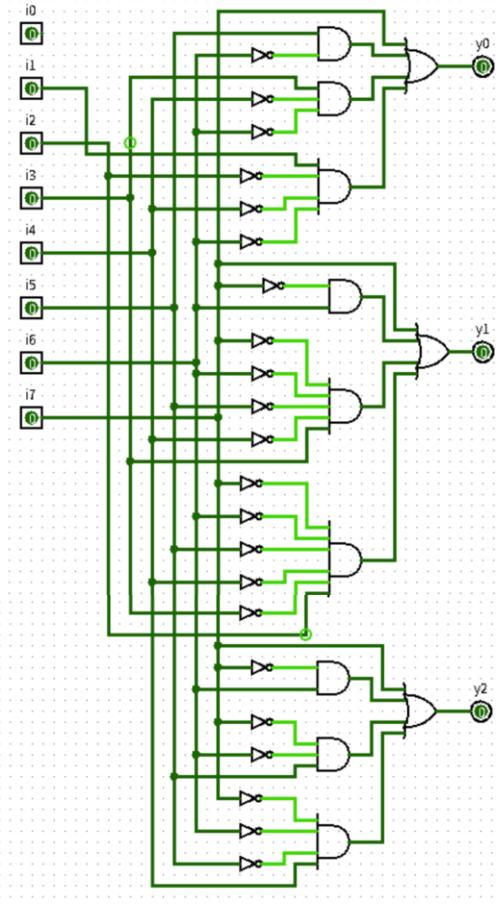
我们可以借助“Minimized”选项卡对表达式进行简化，进而减少电路使用的逻辑门数量



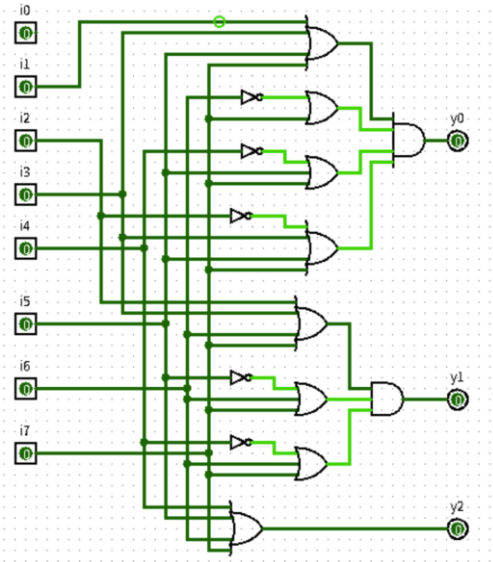
下图对比了同一功能电路未化简的电路结构和化简后的与或式和或与式结构，可以看出占用逻辑门数量有明显差异



未化简逻辑电路



化简与或逻辑电路



化简或与逻辑电路

我们还可以通过统计电路的基本信息：

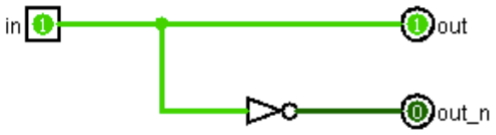
Logisim: 表达式生成 Statistics

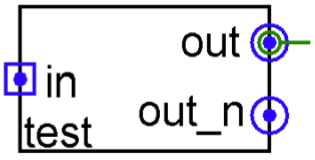
Component	Library	Simple	Unique	Recurs...
Pin	Wiring	11	11	11
NOT Gate	Gates	28	28	28
AND Gate	Gates	9	9	9
OR Gate	Gates	3	3	3
TOTAL (without project's subci...		51	51	51
TOTAL (with subcircuits)		51	51	51

Step3：Verilog HDL 语法入门

这里只显示例一的具体实现

例一：





代码如下：

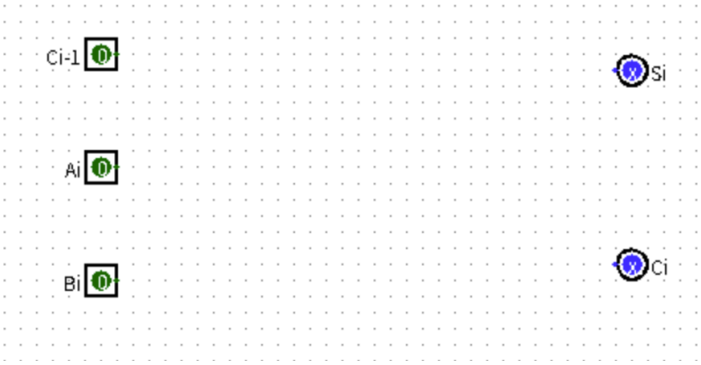
```
module test(
 input in,
 output out,
 output out_n,);
     assign out=in;
     assign out_n=~in;
endmodule
```

【实验练习】

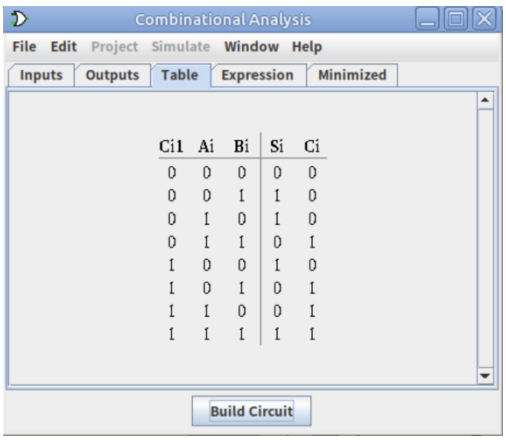
T1. 真值表如下：

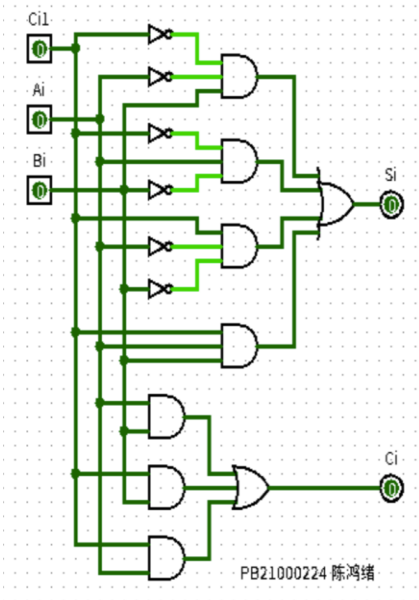
输入			输出	
Ci-1	Ai	Bi	Si	Ci
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

放置输入输出引脚，并编辑相应的 label：



编辑好真值表，并生成相应的逻辑电路：





T2.由真值表：

输入						输出							
G1	G2	G3	A2	A1	A0	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
X	1	X	X	X	X	1	1	1	1	1	1	1	1
X	X	1	X	X	X	1	1	1	1	1	1	1	1
0	X	X	X	X	X	1	1	1	1	1	1	1	1
1	0	0	0	0	0	1	1	1	1	1	1	1	0
1	0	0	0	0	1	1	1	1	1	1	1	0	1
1	0	0	0	1	0	1	1	1	1	1	0	1	1
1	0	0	0	1	1	1	1	1	1	0	1	1	1
1	0	0	1	0	0	1	1	1	0	1	1	1	1
1	0	0	1	0	1	1	1	0	1	1	1	1	1
1	0	0	1	1	0	1	0	1	1	1	1	1	1
1	0	0	1	1	1	0	1	1	1	1	1	1	1

写出相应的逻辑表达式：

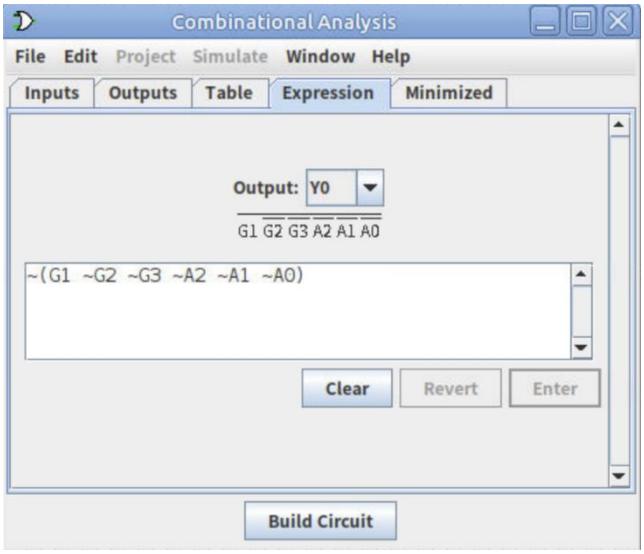
Y7=~(G1 ~G2 ~G3 A2 A1 A0) Y6=~(G1 ~G2 ~G3 A2 A1 ~A0)

Y5=~(G1 ~G2 ~G3 A2 ~A1 A0) Y4=~(G1 ~G2 ~G3 A2 ~A1 ~A0)

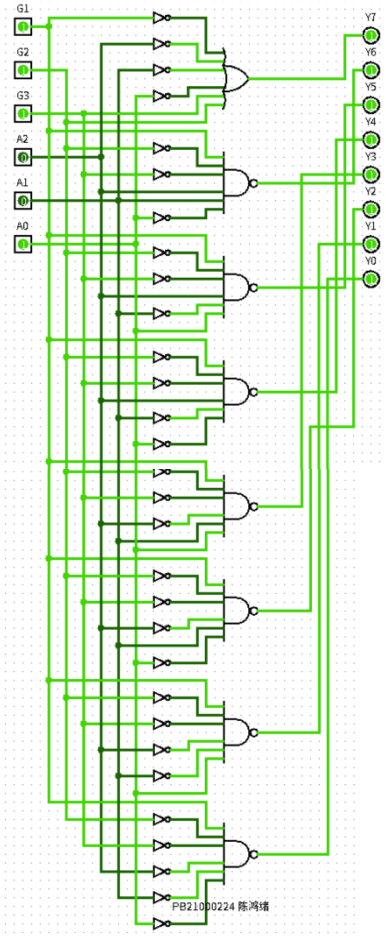
Y3=~(G1 ~G2 ~G3 ~A2 A1 A0) Y2=~(G1 ~G2 ~G3 ~A2 A1 ~A0)

Y1=~(G1 ~G2 ~G3 ~A2 ~A1 A0) Y0=~(G1 ~G2 ~G3 ~A2 ~A1 ~A0)

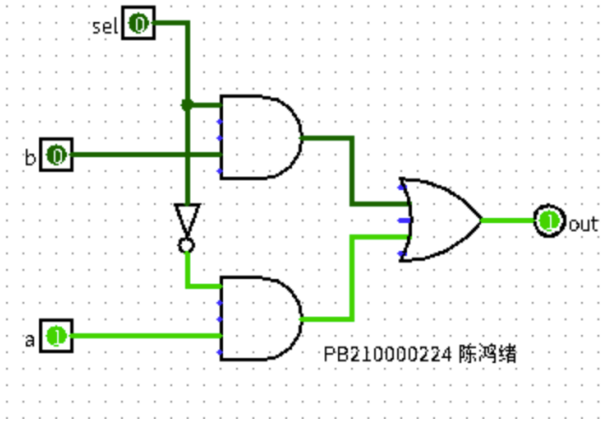
将逻辑表达式输入



得到如下的逻辑电路：



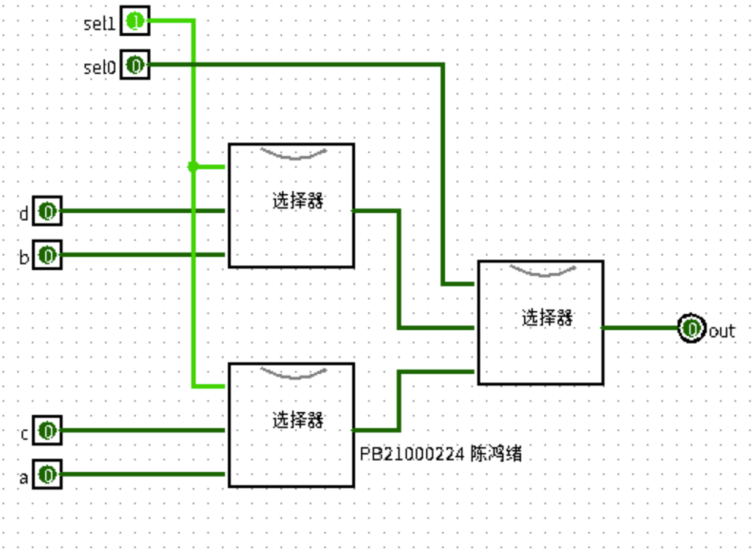
T3. 使用 Logisim 绘制 1bit 位宽的二选一选择器电路图,如下图所示:



编写 Verilog 代码,代码如下:

```
module MUX1(
  input a,
  input b,
  input sel,
  output out);
     assign out=(~sel&a)|(sel&b);
  endmodule
```

T4.首先封装T3中实现的二选一选择器,再使用MUX2-1实现四选一选择器,Logisim实现的逻辑电路如下:



通过例化题目 3 中的二选一选择器,以下为Verilog代码:

```
module MUX4(
  input a,b,c,d,sel0,sel1,
```

```
  output out);
    wire L1,L2;
    MUX1 M1(.a(a), .b(c), .sel(sel1), .out(L1));
    MUX1 M2(.a(b), .b(d), .sel(sel1), .out(L2));
    MUX1 M3(.a(L1), .b(L2), .sel(sel0), .out(out));
endmodule
```

T5. 由逻辑表达式:

y2 = i7|~i7 i6|~i7 ~i6 i5|~i7 ~i6 ~i5 i4

y1 = i7|~i7 i6 |~i7 ~i6 ~i5 ~i4 i3 |~i7 ~i6 ~i5 ~i4 ~i3 i2

y0 = i7|~i7 ~i6 i5|~i7 ~i6 ~i5 ~i4 i3|~i7 ~i6 ~i5 ~i4 ~i3 ~i2 i1

我们得到 Verilog 代码如下:

```
module encode(
  input i7,i6,i5,i4,i3,i2,i1,i0,
  output y2,y1,y0);
    assign y2 = i7|(~i7&i6)|(~i7&~i6&i5)|(~i7&~i6&~i5&i4);
    assign y1 = i7|(~i7&i6)|(~i7&~i6&~i5&i4&i3)|(~i7&~i6&
                ~i5&~i4&~i3&i2);
    assign y0 = i7|(~i7&~i6&i5)|(~i7&~i6&~i5&~i4&i3)|(~i7&
                ~i6&~i5&~i4&~i3&~i2&i1);
endmodule
```

T6.

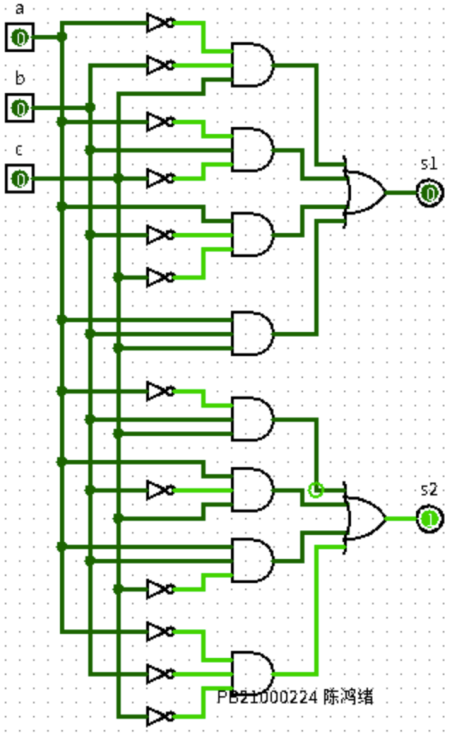
```
module test(
  input a,b,c,
  output s1,s2);
  assign s1= ~a &~b & c | ~a & b &~c | a &~b &~c | a & b & c;
  assign s2= ~a & b & c | a &~b & c | a & b &~c | ~a &~b &~c;
  endmodule
```

将逻辑表达式:

s1= ~a ~b c | ~a b ~c | a ~b ~c | a b c

s2= ~a b c | a ~b c | a b ~c | ~a ~b ~c

输入 Logisim 中,得出如下图所示的逻辑电路



分析其如下的真值表:

a	b	c	s1	s2
0	0	0	0	1
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	0

功能:由真值表,注意 a,b,c 中有偶数个取 1 时,s2 取 1,s1 取 0;a,b,c 中有奇数个取 1 时,s1 取 1,s2 取 0;故该逻辑电路可以判断 a,b,c 中取真值 1 的个数的奇偶性.

【总结与思考】

1. 收获:我更好地掌握了 Logisim 基本用法;进一步熟悉 Logisim 更多功能;学会了用 Logisim 设计组合逻辑电路并进行仿真;学习掌握了 Verilog 初级语法;
2. 本次实验的难易程度: 难度适中,适合初学者.
3. 本次实验的任务量: 任务量合理适中
4. 建议和意见: 可以适当增加实验难度,提高挑战.