

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：\_宋玮\_\_

学生学号：PB20151793\_\_\_\_\_

完成日期：\_2021.10.28\_

计算机实验教学中心制

## 【实验题目】

### 实验二：简单组合逻辑电路

## 【实验目的】

1. 熟悉掌握 Logisim 的基本用法
2. 进一步熟悉 Logisim 更多功能
3. 用 Logisim 设计组合逻辑电路并进行仿真
4. 初步学习 Verilog 语法

总：在本次实验中，进一步熟悉 Logisim 用真值表，表达式快速搭建电路，verilog 的初步学习。

## 【实验环境】

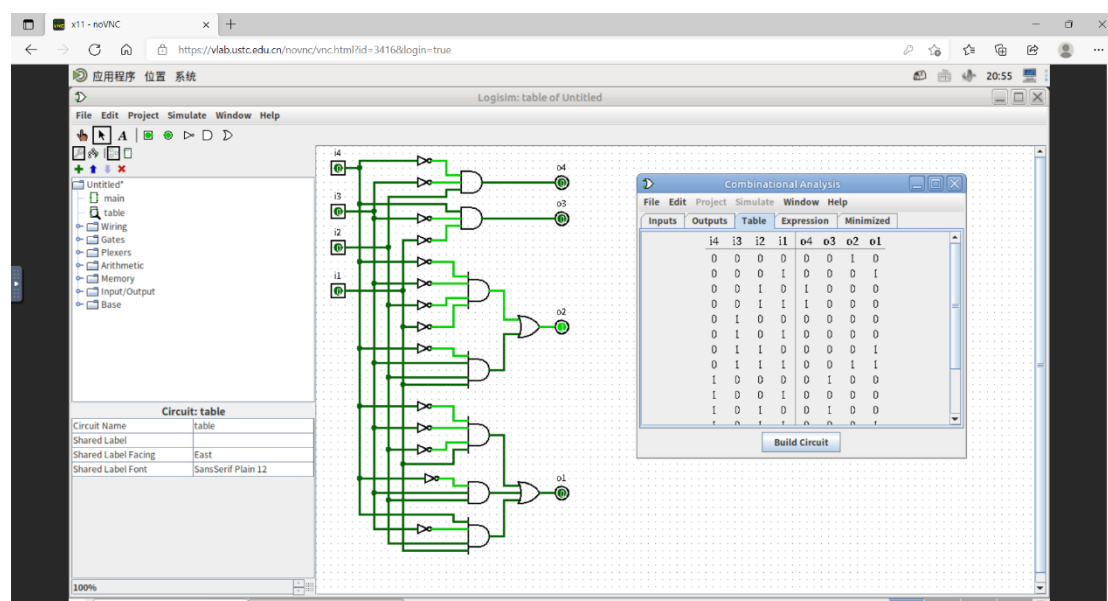
实验平台：vlab.ustc.edu.cn

实验设备：电脑（上的虚拟机）

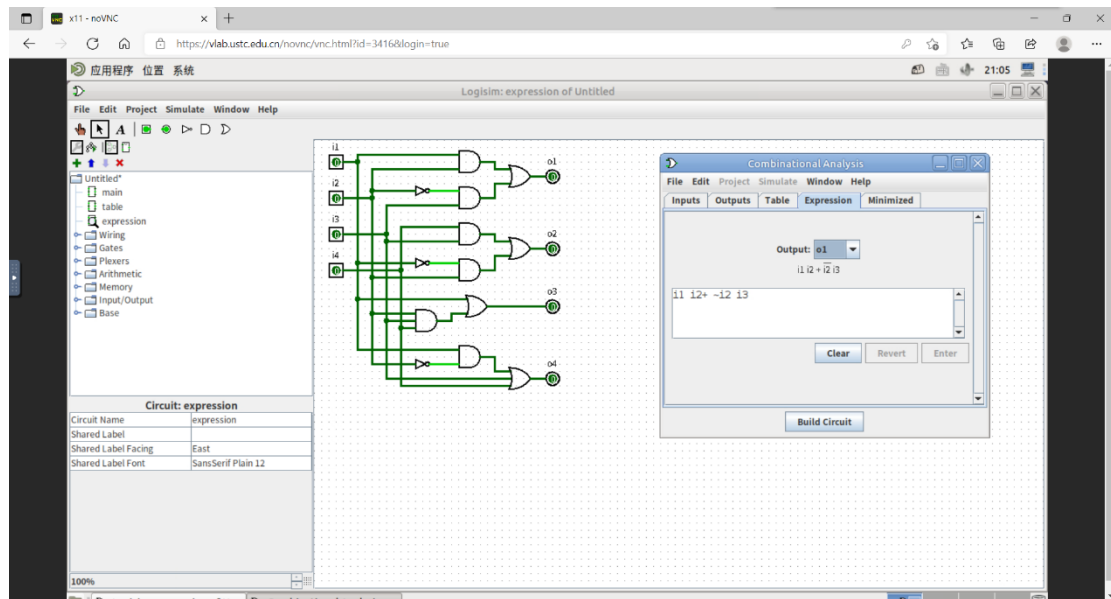
实验软件：Logisim，编辑器

## 【实验过程】

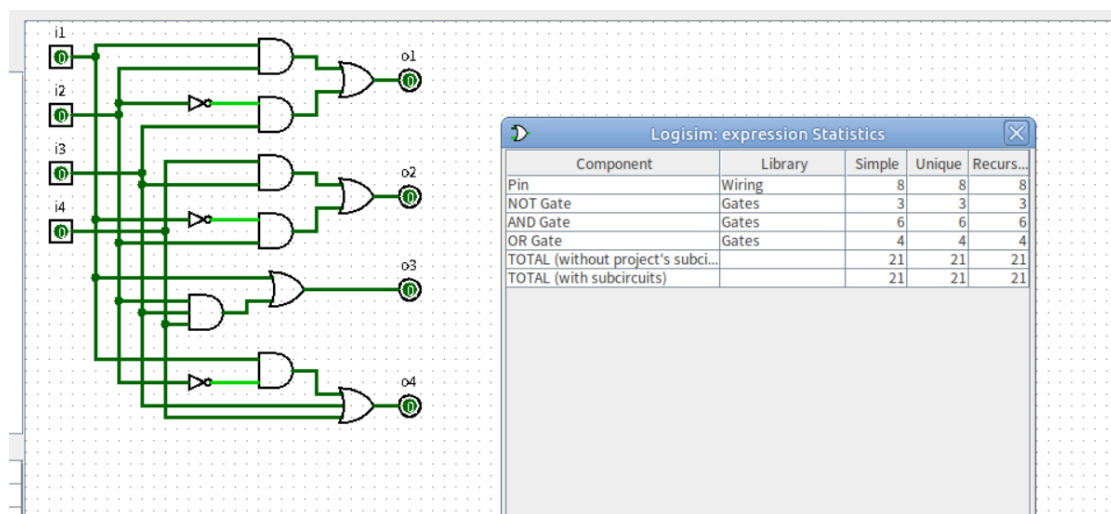
### 1. 用真值表自动生成电路



## 2. 用表达式生成电路图



通过“Project”--> “Get Circuit Statistics”选项统计电路的基本信息。



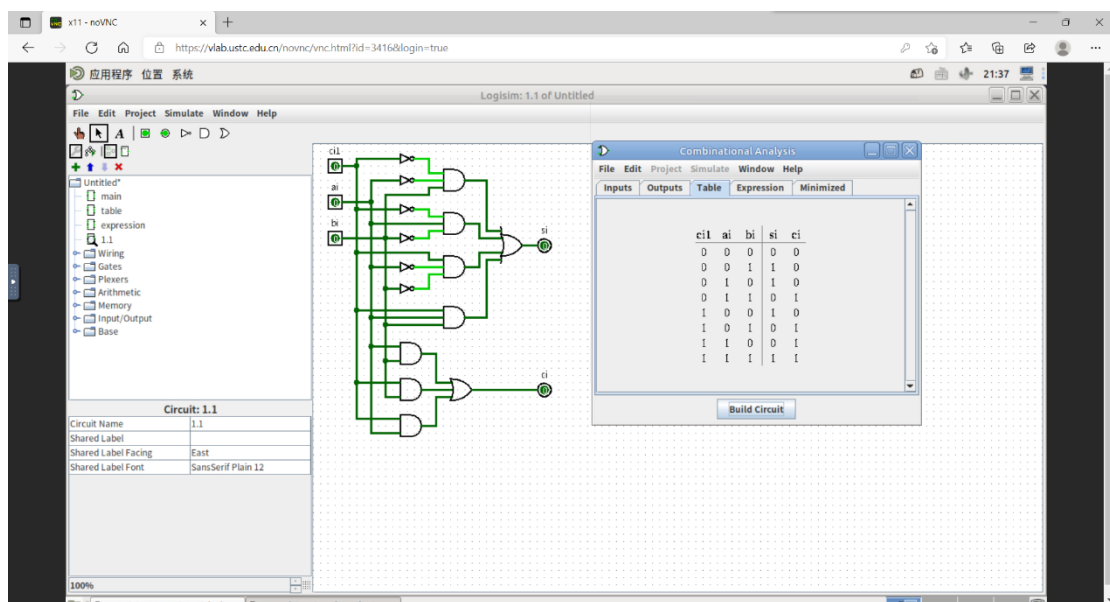
## 3. Verilog HDL 语法入门

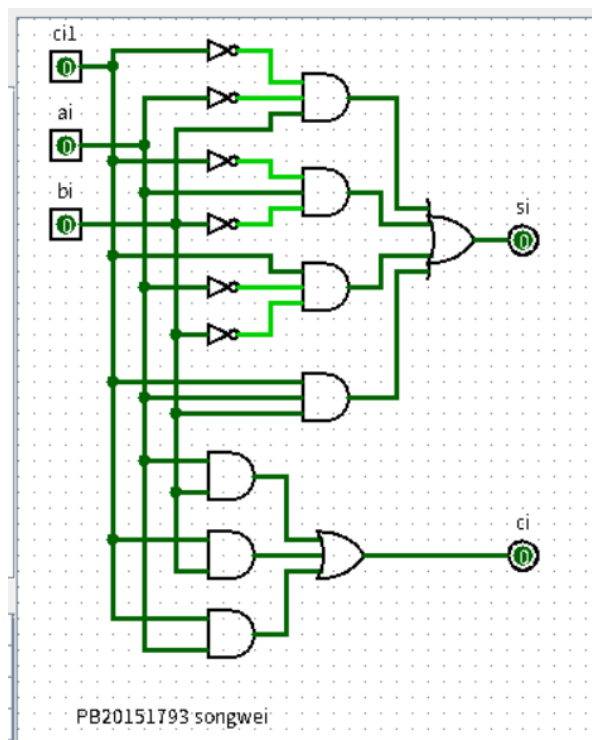
## 【实验练习】

题目 1. 依据如下真值表，通过 Logisim 编辑真值表功能，完成电路设计。电路下方需标注姓名学号。

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

该电路是一个 1 位全加器：





题目 2：根据下列真值表，通过 Logisim 的编辑表达式功能完成电路设计，电路下方需标注姓名学号。

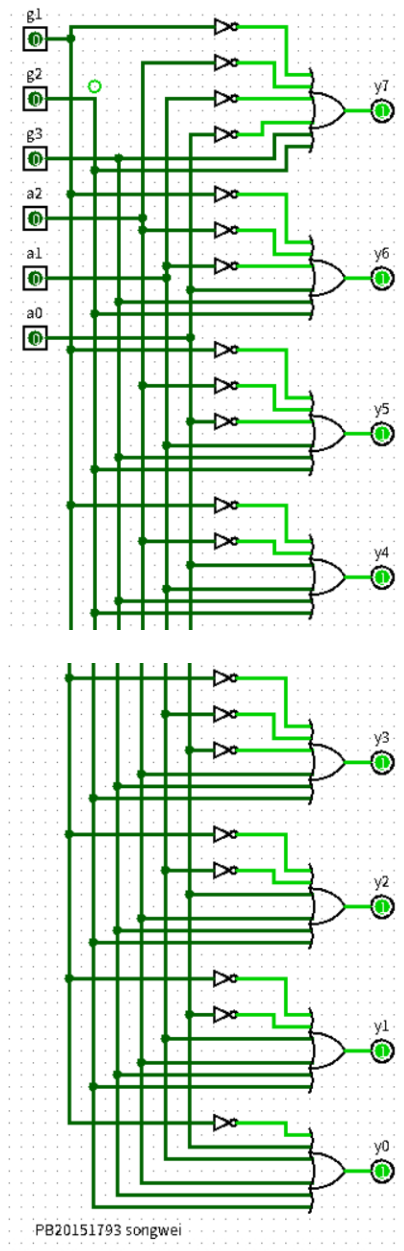
输入						输出							
G1	G2	G3	A2	A1	A0	Y7	Y6	Y5	Y4	Y3	Y2	Y1	Y0
X	1	X	X	X	X	1	1	1	1	1	1	1	1
X	X	1	X	X	X	1	1	1	1	1	1	1	1
0	X	X	X	X	X	1	1	1	1	1	1	1	1
1	0	0	0	0	0	1	1	1	1	1	1	1	0
1	0	0	0	0	1	1	1	1	1	1	1	0	1
1	0	0	0	1	0	1	1	1	1	1	0	1	1
1	0	0	0	1	1	1	1	1	1	0	1	1	1
1	0	0	1	0	0	1	1	1	0	1	1	1	1
1	0	0	1	0	1	1	1	0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

分析如下：

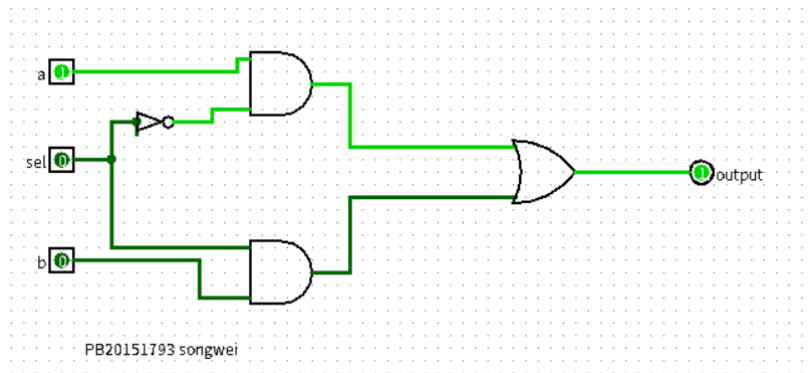
$$\begin{aligned}
 Y7 &= g2 + g3 + \sim g1 + g1 \sim g2 \sim g3 \sim a2 \sim a1 \sim a0 + g1 \sim g2 \sim g3 \sim a2 \sim a1 \ a0 + \\
 &g1 \sim g2 \sim g3 \sim a2 \ a1 \sim a0 + g1 \sim g2 \sim g3 \sim a2 \ a1 \ a0 + g1 \sim g2 \sim g3 \ a2 \sim a1 \sim a0 + \\
 &g1 \sim g2 \sim g3 \ a2 \sim a1 \ a0 + g1 \sim g2 \sim g3 \ a2 \ a1 \sim a0
 \end{aligned}$$

$$\begin{aligned}
Y6 = & g_2 + g_3 + \sim g_1 + g_1 \sim g_2 \sim g_3 \sim a_2 \sim a_1 \sim a_0 + g_1 \sim g_2 \sim g_3 \sim a_2 \sim a_1 a_0 + \\
& g_1 \sim g_2 \sim g_3 \sim a_2 a_1 \sim a_0 + g_1 \sim g_2 \sim g_3 \sim a_2 a_1 a_0 + g_1 \sim g_2 \sim g_3 a_2 \sim a_1 \sim a_0 + \\
& g_1 \sim g_2 \sim g_3 a_2 \sim a_1 a_0 + g_1 \sim g_2 \sim g_3 a_2 a_1 a_0 \\
Y5 = & g_2 + g_3 + \sim g_1 + g_1 \sim g_2 \sim g_3 \sim a_2 \sim a_1 \sim a_0 + g_1 \sim g_2 \sim g_3 \sim a_2 \sim a_1 a_0 + \\
& g_1 \sim g_2 \sim g_3 \sim a_2 a_1 \sim a_0 + g_1 \sim g_2 \sim g_3 \sim a_2 a_1 a_0 + g_1 \sim g_2 \sim g_3 a_2 \sim a_1 \sim a_0 + \\
& g_1 \sim g_2 \sim g_3 a_2 a_1 \sim a_0 + g_1 \sim g_2 \sim g_3 a_2 a_1 a_0 \\
Y4 = & g_2 + g_3 + \sim g_1 + g_1 \sim g_2 \sim g_3 \sim a_2 \sim a_1 \sim a_0 + g_1 \sim g_2 \sim g_3 \sim a_2 \sim a_1 a_0 + \\
& g_1 \sim g_2 \sim g_3 \sim a_2 a_1 \sim a_0 + g_1 \sim g_2 \sim g_3 \sim a_2 a_1 a_0 + g_1 \sim g_2 \sim g_3 a_2 \sim a_1 a_0 + \\
& g_1 \sim g_2 \sim g_3 a_2 a_1 \sim a_0 + g_1 \sim g_2 \sim g_3 a_2 a_1 a_0 \\
Y3 = & g_2 + g_3 + \sim g_1 + g_1 \sim g_2 \sim g_3 \sim a_2 \sim a_1 \sim a_0 + g_1 \sim g_2 \sim g_3 \sim a_2 \sim a_1 a_0 + \\
& g_1 \sim g_2 \sim g_3 \sim a_2 a_1 \sim a_0 + g_1 \sim g_2 \sim g_3 a_2 \sim a_1 \sim a_0 + g_1 \sim g_2 \sim g_3 a_2 \sim a_1 a_0 + \\
& g_1 \sim g_2 \sim g_3 a_2 a_1 \sim a_0 + g_1 \sim g_2 \sim g_3 a_2 a_1 a_0 \\
Y2 = & g_2 + g_3 + \sim g_1 + g_1 \sim g_2 \sim g_3 \sim a_2 \sim a_1 \sim a_0 + g_1 \sim g_2 \sim g_3 \sim a_2 \sim a_1 a_0 + \\
& g_1 \sim g_2 \sim g_3 \sim a_2 a_1 a_0 + g_1 \sim g_2 \sim g_3 a_2 \sim a_1 \sim a_0 + g_1 \sim g_2 \sim g_3 a_2 \sim a_1 a_0 + \\
& g_1 \sim g_2 \sim g_3 a_2 a_1 \sim a_0 + g_1 \sim g_2 \sim g_3 a_2 a_1 a_0 \\
Y1 = & g_2 + g_3 + \sim g_1 + g_1 \sim g_2 \sim g_3 \sim a_2 \sim a_1 \sim a_0 + g_1 \sim g_2 \sim g_3 \sim a_2 a_1 \sim a_0 + \\
& g_1 \sim g_2 \sim g_3 \sim a_2 a_1 a_0 + g_1 \sim g_2 \sim g_3 a_2 \sim a_1 \sim a_0 + g_1 \sim g_2 \sim g_3 a_2 \sim a_1 a_0 + \\
& g_1 \sim g_2 \sim g_3 a_2 a_1 \sim a_0 + g_1 \sim g_2 \sim g_3 a_2 a_1 a_0 \\
Y0 = & g_2 + g_3 + \sim g_1 + g_1 \sim g_2 \sim g_3 \sim a_2 \sim a_1 a_0 + g_1 \sim g_2 \sim g_3 \sim a_2 a_1 \sim a_0 + \\
& g_1 \sim g_2 \sim g_3 \sim a_2 a_1 a_0 + g_1 \sim g_2 \sim g_3 a_2 \sim a_1 \sim a_0 + g_1 \sim g_2 \sim g_3 a_2 \sim a_1 a_0 + \\
& g_1 \sim g_2 \sim g_3 a_2 a_1 \sim a_0 + g_1 \sim g_2 \sim g_3 a_2 a_1 a_0
\end{aligned}$$

借助“Minimized”选项卡对表达式进行简化，然后生成电路；



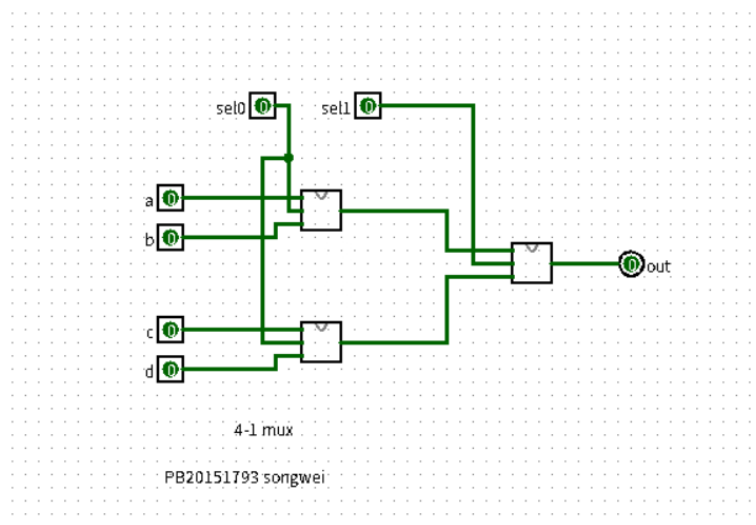
题目 3：使用 Logisim 绘制 1bit 位宽的二选一选择器电路图，并根据生成的电路图编写 Verilog 代码。输入信号为 a,b,sel，输出信号为 out,sel 为 0 时选通 a 信号。



Verilog 代码如下：

```
module 2lmux(input a, b, sel,
output out);
assign out=(~sel)&a|sel&b;
endmodule
```

题目 4：通过例化题目 3 中的二选一选择器，用 Verilog 实现一个四选一选择器，并画出对应的电路图。输入信号为 a, b, c, d, sel1, sel0, out, sel1 和 sel0 都为 0 时选中 a 信号。



Verilog 代码如下：



```

module 4lmux(input a, b, c, d, sel0, sel1,
output out);
wire o1, o2;
2lmux 2lmux_1(
    .a(a),
    .b(b),
    .sel(sel0),
    .out(o1));
2lmux 2lmux_2(
    .a(c),
    .b(d),
    .sel(sel0),
    .out(o2));
2lmux 2lmux_3(
    .a(o1),
    .b(o2),
    .sel(sel1),
    .out(out));
endmodule

```

题目 5：根据前面用到的八位优先编码器真值表，编写 verilog 代码。

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

Verilog 代码如下：

```

module 8coder(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)
endmodule

```

题目 6：阅读如下 Verilog 代码，描述其功能，并画出其对应的电路图。

```

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

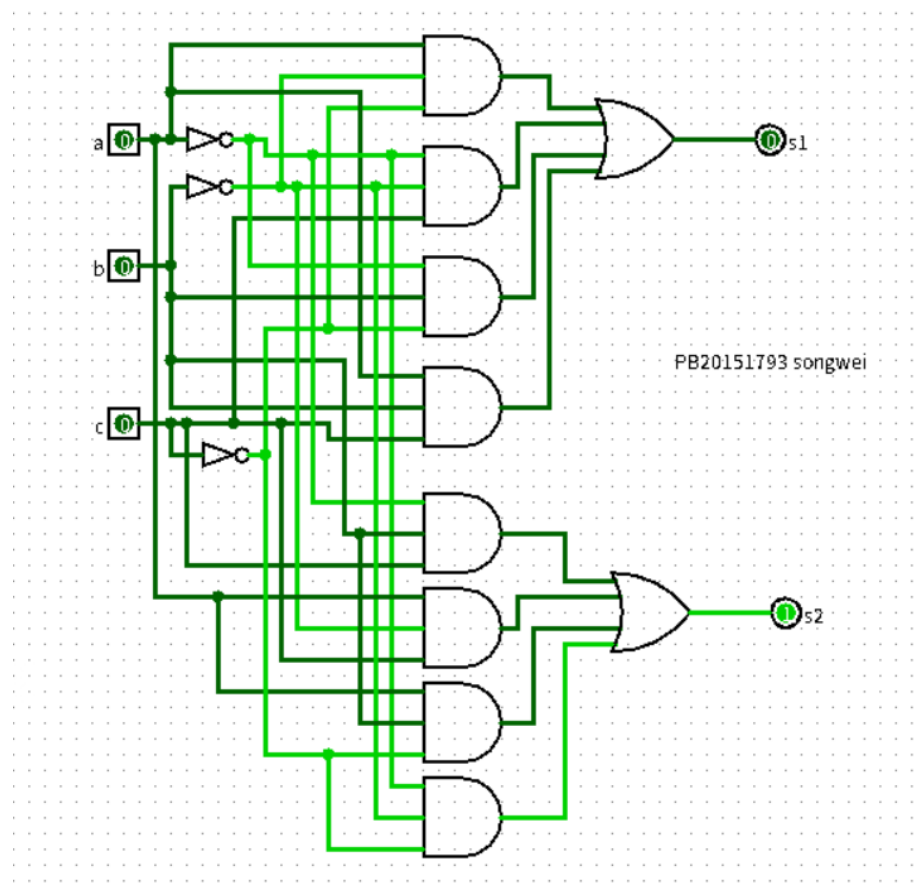
```

描述功能:

当 a, b, c 三个输入中 1 的个数为奇数时, s1 输出为 1, s2 输出为 0;

当 a, b, c 三个输入中 1 的个数为偶数时 (包括个数为 0), s2 输出为 1, s1 输出为 0;

电路图如下:



### 【总结与思考】

通过本次实验，了解了快速生成电路的几种方法；并且对 verilog 语法有了一个了解，能用 verilog 编写简单电路。

本次难度不高，任务量不大，无改进建议。