

桂林电子科技大学 2023-2024 学年 第 1 学期

计算机组成原理 B 实验报告

实验名称 **32 位 ALU 设计**

院 系 **计算机与信息安全学院** 专业 **软件工程**

学 号 **2200350204** 姓名 **李禹佳**

同 作 者

实验日期 **2023** 年 **11** 月 **25** 日

辅导教师意见:

成绩 教师签名:

一、实验目的和要求

1. 实验目的

- (1) 掌握算术逻辑运算单元 (ALU) 的基本构成;
- (2) 掌握 Logisim 中各种运算组件的使用方法, 熟悉多路选择器的使用;
- (3) 掌握 ALU 的设计和仿真方法。

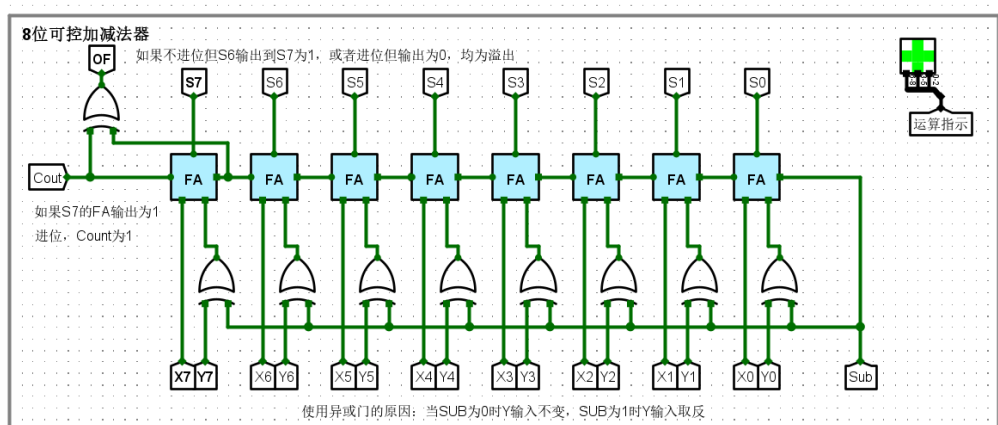
2. 实验要求

- (1) 实验前, 完成 Logisim 软件使用学习, 并预习实验内容, 准备好 ALU 设计方案;
- (2) 独立完成 ALU 设计, 在头歌平台完成指定闯关任务;
- (3) 如实记录实验设计步骤, 并对实验过程及结果进行分析总结, 撰写实验报告。

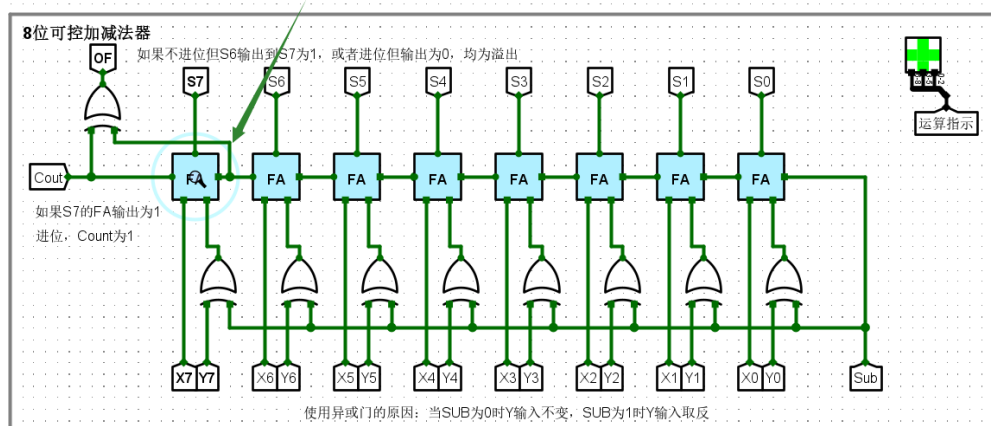
二、实验步骤

实验一：运算器设计 (HUST)

1. 第 1 关：8 位可控加减法电路设计



线路中使用一位全加器 FA



FA 接收两位输入端和一位进位端，输出一位向前进位端，一位结果端

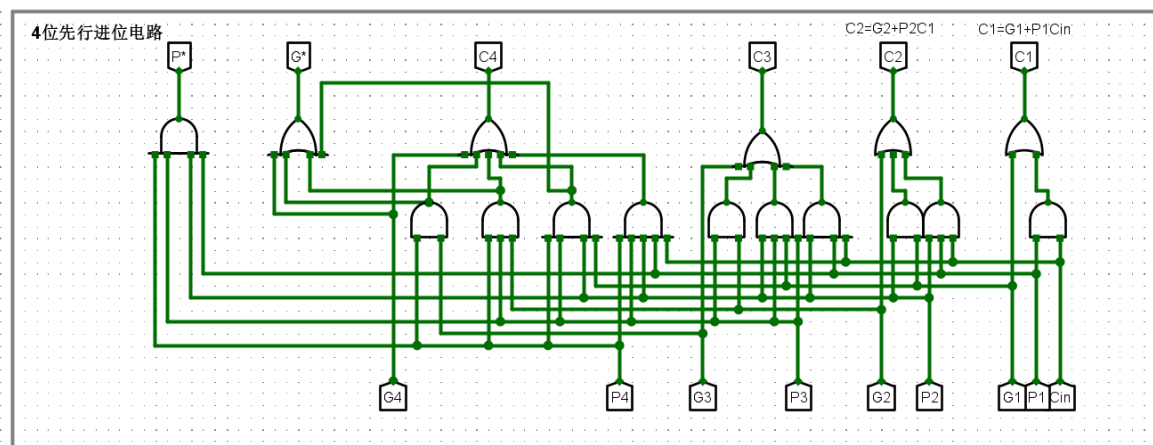
实现 8 位可控加减法器，可以用 8 个 FA 串行来实现加减法

其中 Sub 为控制加减法的端口，即 Sub=0 时，实现 $x+y$ ，无进位；Sub=1 时，实现 $x+(\sim y)+1$ (Sub 进位) 是否取反通过对 Sub 和单位减数的异或来判断，如果 sub=0,y=0,输出 0;sub=0,y=1,输出 1;sub=1,y=0,输出 1 (与 y 相反); sub=1,y=1,输出 0 (与 y 相反)

其中 OF 为溢出检测，通过比较最高进位和最高位的值是否相同来判断是否溢出，如果是 01 或者 10，则溢出 (使用异或门来判断)

2. 第 2 关：CLA182 四位先行进位电路设计、

为了降低运算的延迟，这里将所有线路并行连接



由全加器

$$C(i) = X(i-1)Y(i-1) + [X(i-1) \oplus Y(i-1)]C(i-1)$$

将 $G(i) = X(i-1)Y(i-1)$, $P(i) = X(i-1) \oplus Y(i-1)$ 带入上式得

$$C1 = P1 Cin + G1 \quad \text{又} \quad C2 = P2C1 + G2$$

$$C2 = P2 C1 + G2$$

$$C3 = P3 C2 + G3$$

$$C4 = P4 C3 + G4$$

由此，电路逻辑表达式为：

$$P = P_4 P_3 P_2 P_1$$

$$G = P_4 P_3 P_2 G_1 + P_4 P_3 G_2 + P_4 G_3 + G_4$$

$$C_1 = P_1 C_{in} + G_1$$

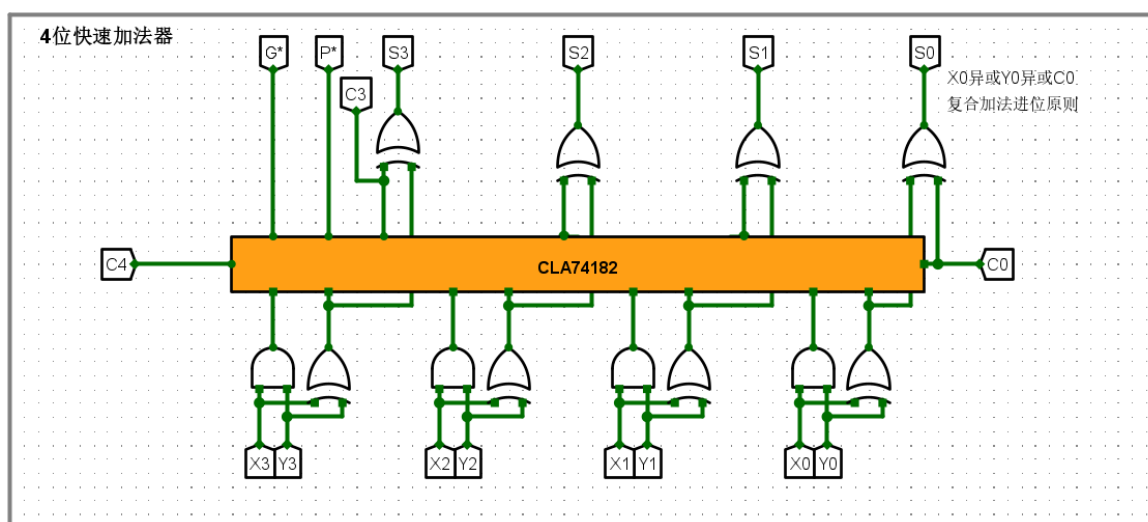
$$C_2 = P_2 P_1 C_{in} + P_2 G_1 + G_2$$

$$C_3 = P_3 P_2 P_1 C_{in} + P_3 P_2 G_1 + P_3 G_2 + G_3$$

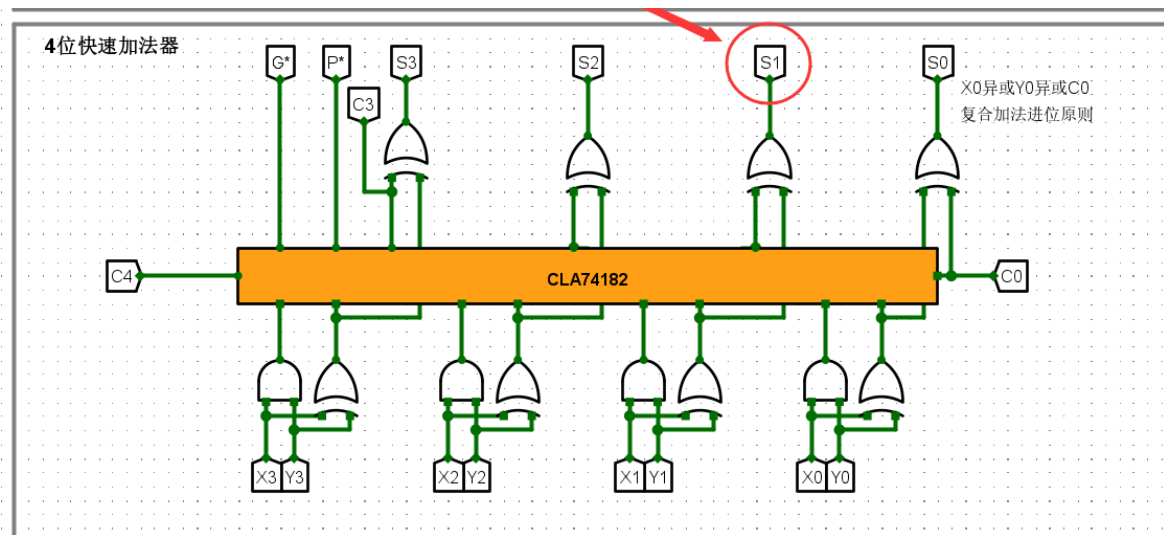
$$C_4 = P_4 P_3 P_2 P_1 C_{in} + P_4 P_3 P_2 G_1 + P_4 P_3 G_2 + P_4 G_3 + G_4$$

其中 C 为进位端，C_{in} 为低位进位。要使电路降低延迟，所以将线路接为并行，进位端 C₂ 不用等待 C₁ 的结果而直接使用 C₁ 的接线运算。

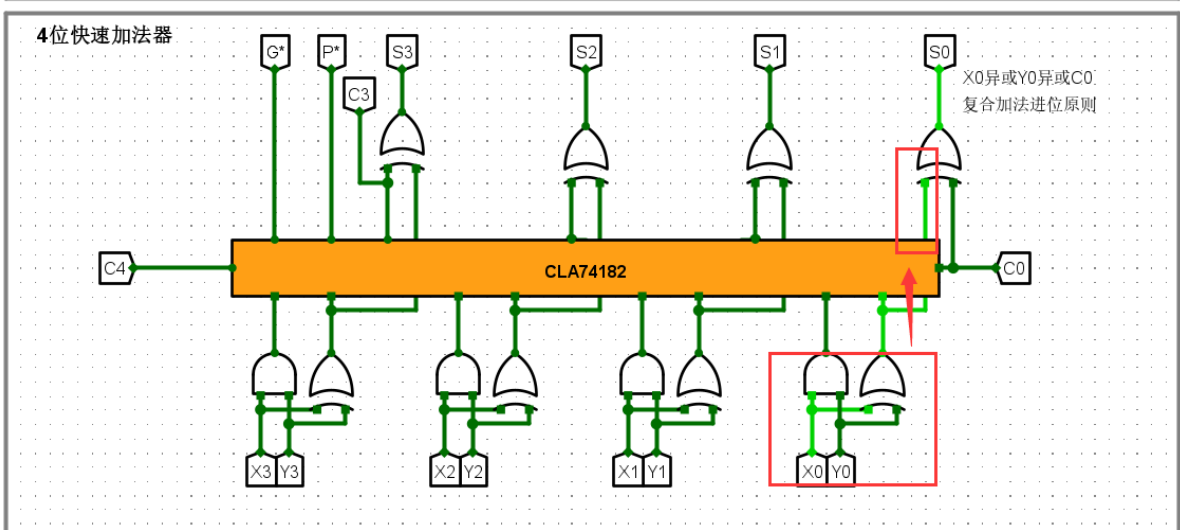
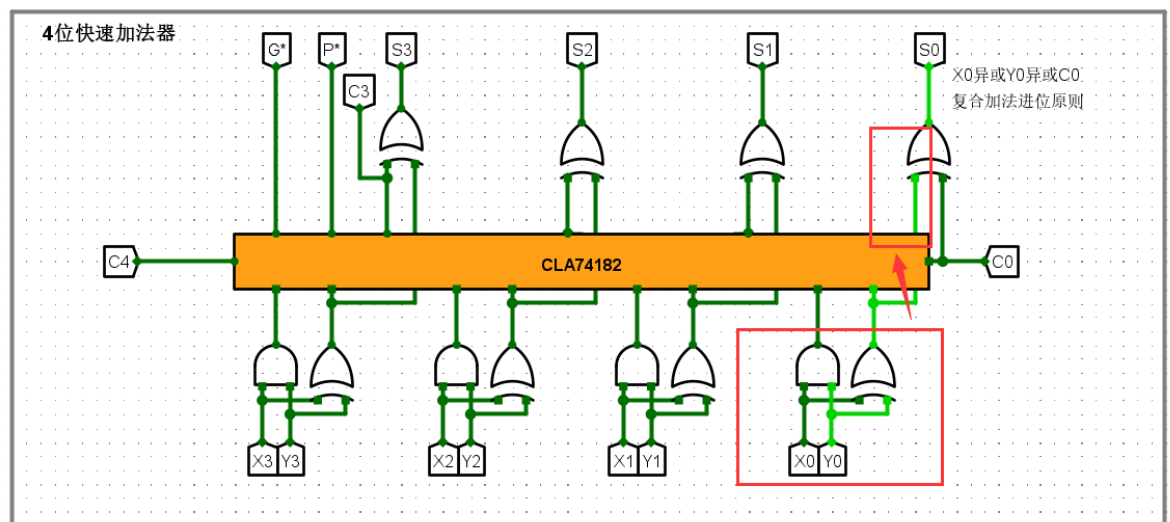
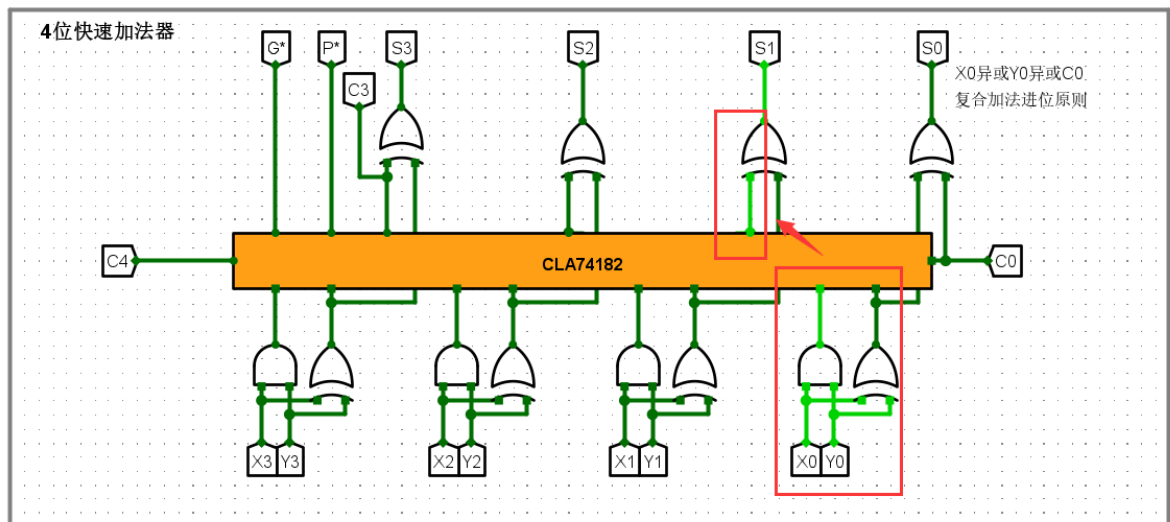
3. 第 3 关：4 位快速加法器设计



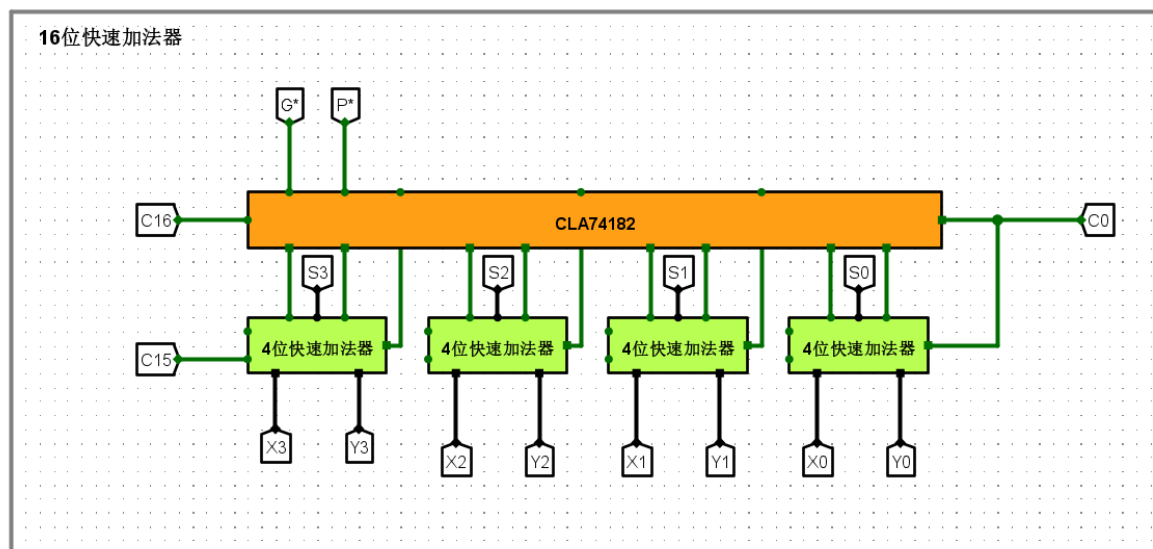
由第二关所封装的 CLA74182 可知输入两个四位二进制数可以得到四位加和
这里的 x,y 对应着公式 $G(i)=X(i-1)Y(i-1)$, $P(i)=X(i-1) \oplus Y(i-1)$ 中的 $X(i-1)$ 和 $Y(i-1)$



这里的 x0 和 y0 对应着 S1，将 x0 和 y0 引入后通过公式的计算输入至 74182 中，如果 $x_0=y_0=1$ ，则 74182 连接 S1 的引脚置 1，如果 x0 与 y0 相反，则对应的 S0(非 74182 接线端)置 1，故使用与非门。



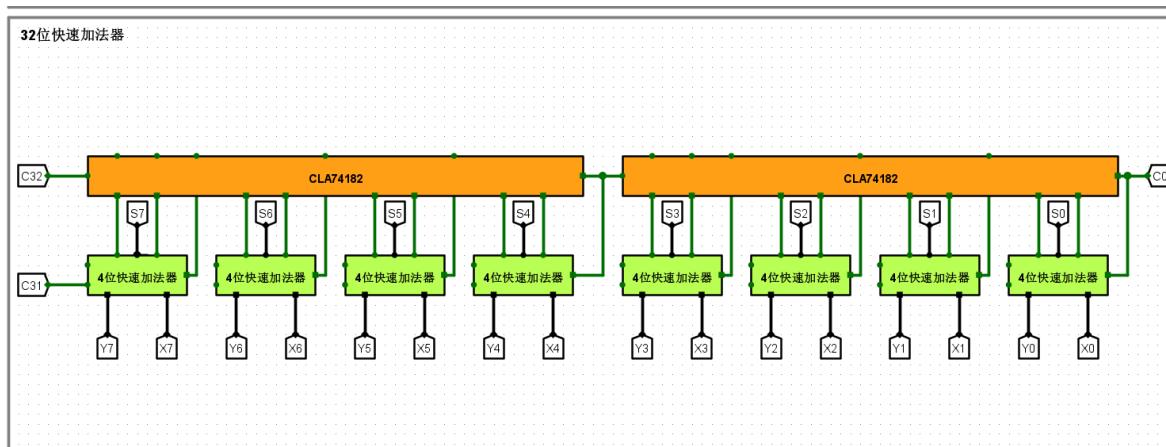
4. 第4关：16位快速加法器设计



在十六进制快速加法器中，将十六位分成 4×4 的四个四位快速加法器，图中显示的封装输出为 G,S,P

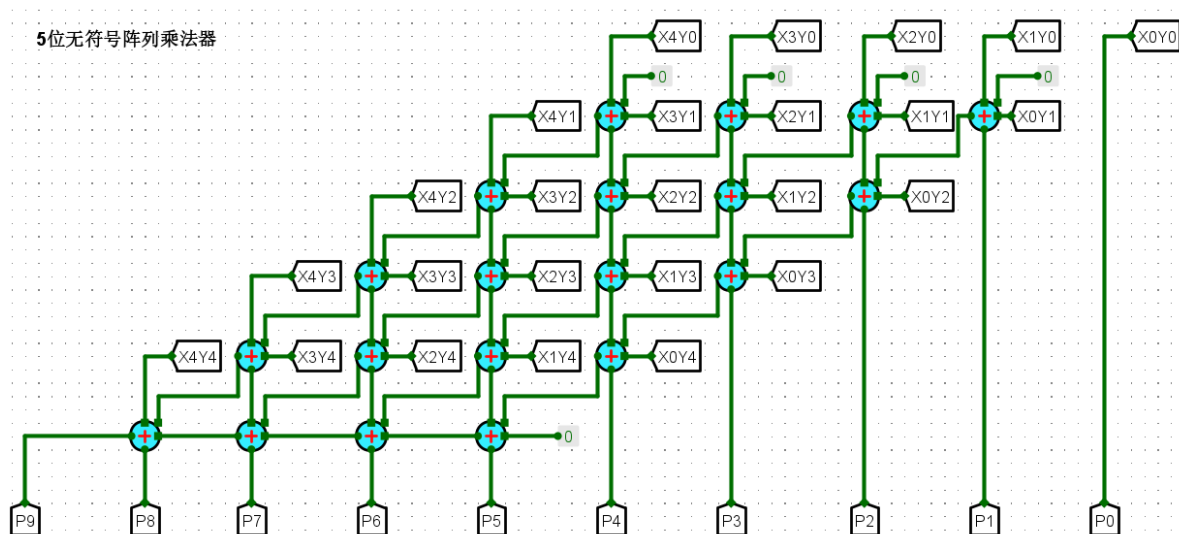
其中 $x_0+y_0=1111$ 时 P0 有效，当 P0~P3 均有效时 P 有效。当输出 >1111 时 G0 有效，只要有一个 G()有效，则 G 有效。其中加法的结果存在 S0 中。

5. 第5关：32位快速加法器设计



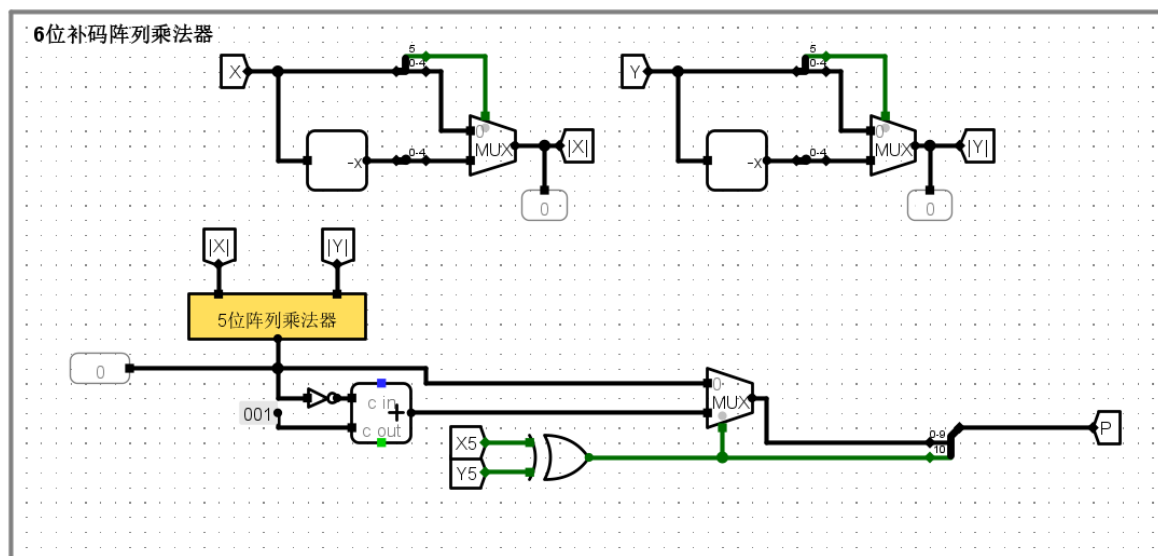
32 进制与 16 进制类似，只是将第一部分 16 进制的向高进位接为低位向第二部分进位。

6. 第 6 关：5 位无符号阵列乘法器设计



使用阵列乘法器模拟人工计算，将 x 中每一位去乘上 Y 中的每一位，再将每一位的权值对应相加，其中依次向左移一位的方式摆放可以实现上一步进位输入，在实验中，先使得 x 与 y 分别进行与操作，分别为 $x_0y_0, x_0y_1, x_0y_2, x_0y_3, x_0y_4 \dots$ 再通过加法器运算，得到输出结果 $P_0 \sim P_4$

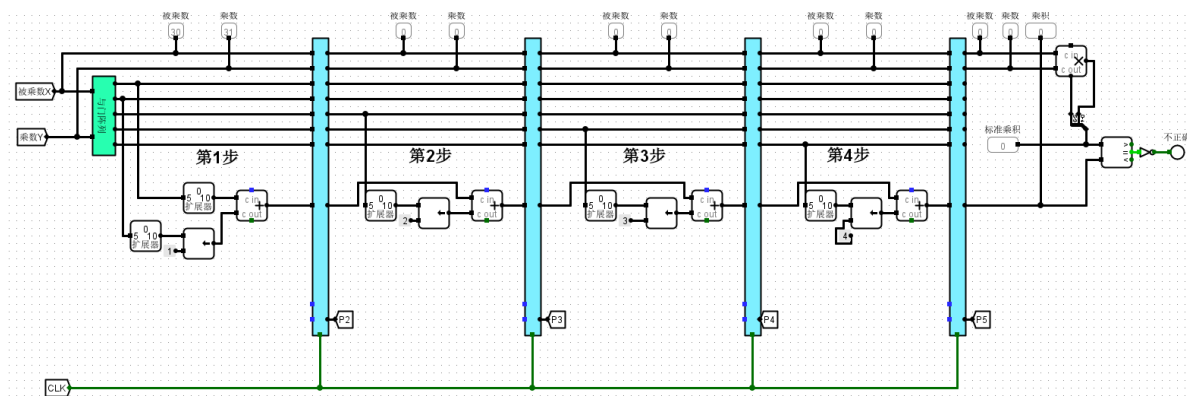
7. 第 7 关：6 位有符号补码阵列乘法器



首先，补码的运算可依据首位（符号位）的不妨进行不同的处理，如果首位相同且都是正数，可以先将符号位去掉，直接输入 5 位阵列乘法器，如果都是负数，将 x 与 y 输入求补器得到两个数补码再输入五位阵列乘法器。如果两个数符号位相同，则乘积的补码位 p 输出值（补全位数）如果符号位不同，补全位数为 1

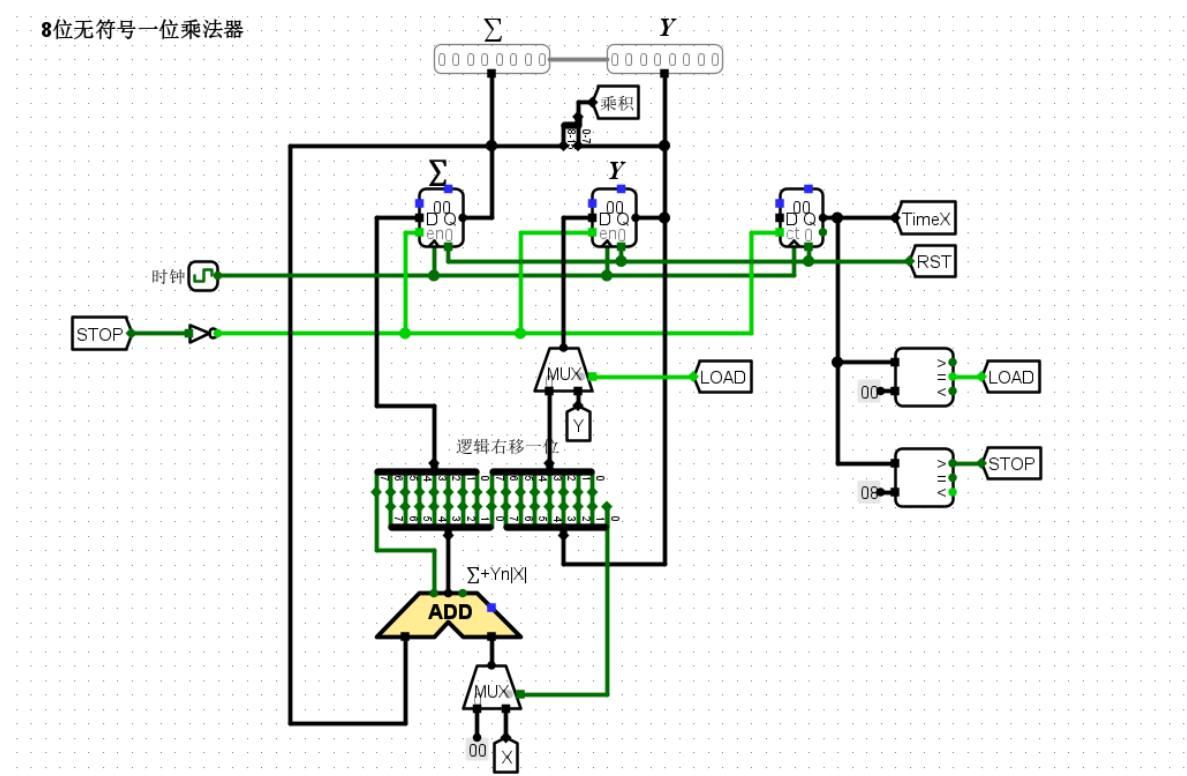
这里的求补器先是将负数的补码转化为原码，再将符号位数改为 0，如果是正数，则先变为负数再求补码，在输入至五位阵列乘法器前，先计算 x 与 y 的绝对值在输入。x 与 y 的接线电路相同，先将 x 的首位与剩余位分开，得到数值位与符号位。再将符号位接入选择端，如果是 1，则通过求补器得到绝对值

8. 第 8 关：乘法流水线设计



通过使用五位阵列乘法器和求补器进行乘法流水线操作。乘法操作的低位部分，使用乘法器的第 1 个部件计算最低位的乘积。乘法操作的中间部分，使用乘法器的第 2~5 个部件计算次低位到最高位的乘积。求和并取反，将结果取反并与之前的结果相加得到最终的积。最后将各个阶段通过流水线寄存器连接起来，传递数据和控制信号。

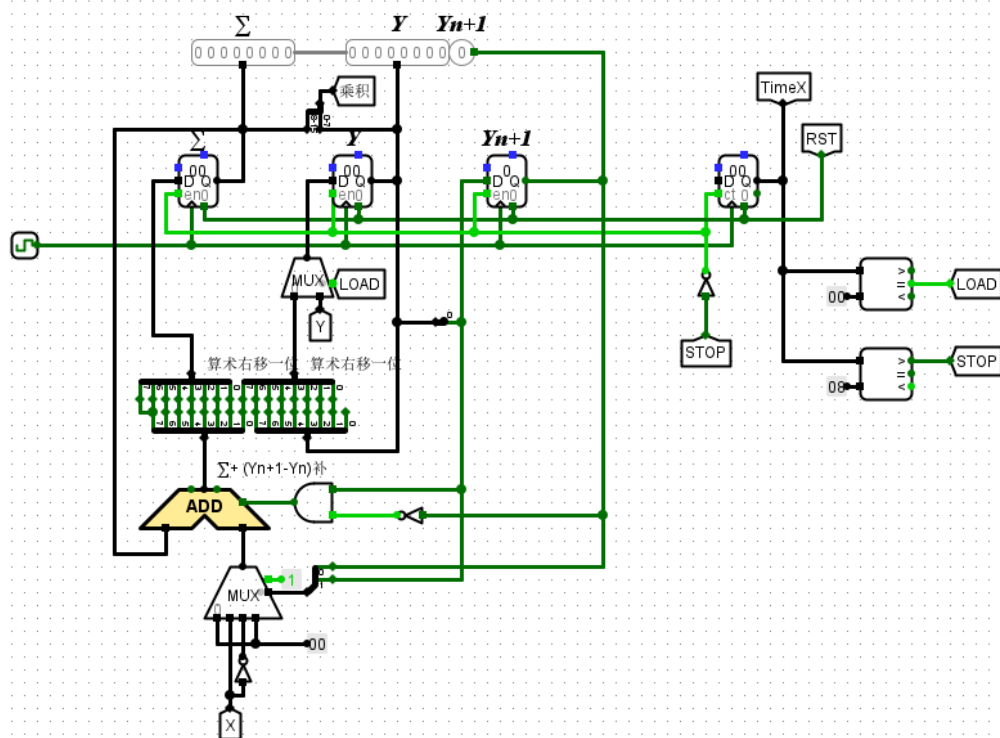
9. 第 9 关：原码一位乘法器设计



无符号原码一位乘，先设置一个部分积 x 使得 $x = x + \text{符号位} \times \text{乘数}$ ，然后将 x 右移一位（无进位则补 0）在最终的计算后将各个部分积相加，在电路中，首先用分线器分离符号位和数值位，再用多路选择器 \times 乘数 b 。最后的加法部分使用 8 位串行加法器实现，在右移部分使用四组八位分线器实现，使用寄存器保存部分积和乘数 b ，使用计数器统计脉冲次数时边界情况的特殊处理。

10. 第 10 关：补码一位乘法器设计

8位补码 Booth一位乘法器



补码一位乘法器在原码一位乘法器的基础上取数的补码再进行运算，最后将结果转换为补码形式即可

三、实验小结

1. 使用软件不熟练，需要先在草纸上画一些路线图再结合给出资料才能正确连线
2. 在第二关中将 p 与 G 错误的接在了每个 c 上，后改正
3. 在第四关中对封装器件引脚不熟悉，尝试多次后通关
4. 在第七关中对求补器运用不熟练