

logisim计组实验七 32位ALU

原创

[Treblez*](#) 于 2020-05-05 19:43:44 发布 8442 收藏 70

分类专栏: [logisim与计算机组成](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: <https://blog.csdn.net/treblez/article/details/105925234>

版权



[logisim与计算机组成](#) 专栏收录该内容

14 篇文章 23 订阅

订阅专栏

文章目录

[描述](#)

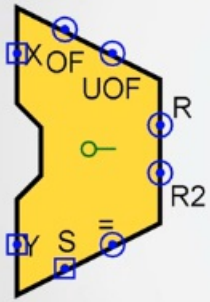
[注意](#)

[电路图](#)

[测试图](#)

电路文件已经托管至Github, 欢迎star: [点这里](#)

描述



引脚	输入/输出	位宽	功能描述
X	输入	32	操作数X
Y	输入	32	操作数Y
S	输入	4	运算操作码 ALU_OP
R	输出	32	ALU运算结果
R2	输出	32	ALU结果第二部分，用于乘法运算结果高位或除法运算的余数位，其它运算时值为零
OF	输出	1	有符号加减运算溢出标记，其他运算为0
UOF	输出	1	无符号加减运算溢出标记，其它运算为0
Equal	输出	1	Equal=(x==y)?1:0, 对所有运算均有效

<https://blog.csdn.net/treblez>

ALU OP	十进制	运算功能
0000	0	$R = X \ll Y$ 逻辑左移 (Y取低五位) $R_2=0$
0001	1	$R = X \gg Y$ 算术右移 (Y取低五位) $R_2=0$
0010	2	$R = X \ggg Y$ 逻辑右移 (Y取低五位) $R_2=0$
0011	3	$R = (X * Y)_{[31:0]}$ $R_2 = (X * Y)_{[63:32]}$ 无符号乘法
0100	4	$R = X/Y$ $R_2 = X\%Y$ 无符号除法
0101	5	$R = X + Y$ (Set OF/UOF)
0110	6	$R = X - Y$ (Set OF/UOF)
0111	7	$R = X \& Y$ 按位与
1000	8	$R = X Y$ 按位或
1001	9	$R = X \oplus Y$ 按位异或
1010	10	$R = \sim(X Y)$ 按位或非
1011	11	$R = (X < Y) ? 1 : 0$ 有符号比较
1100	12	$R = (X < Y) ? 1 : 0$ 无符号比较

<https://blog.csdn.net/treblez>

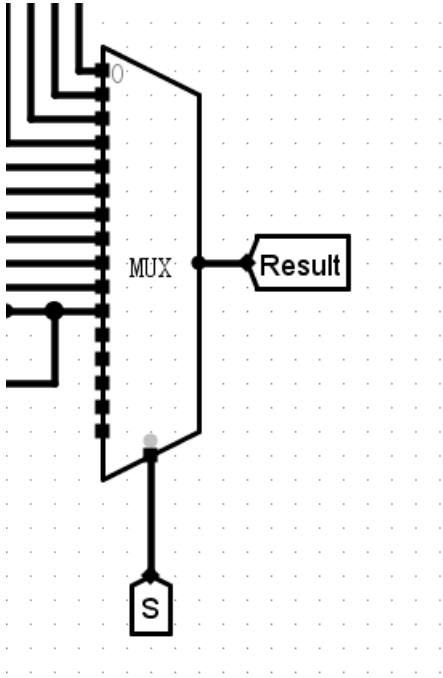
注意

0.当最高位向更高位有进位（或借位）时产生溢出。

1.有符号数溢出判断：

最高位进位状态 \oplus 次高位进位状态=1，则溢出

2.像下图这种地方容易连错，错了的话会比别的地方粗一点点，比较难找，注意一点吧

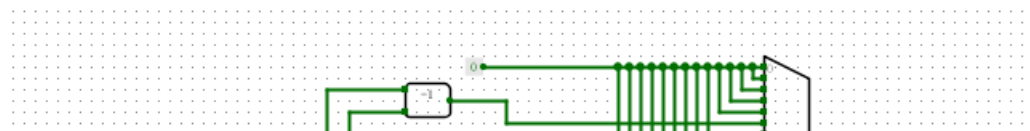
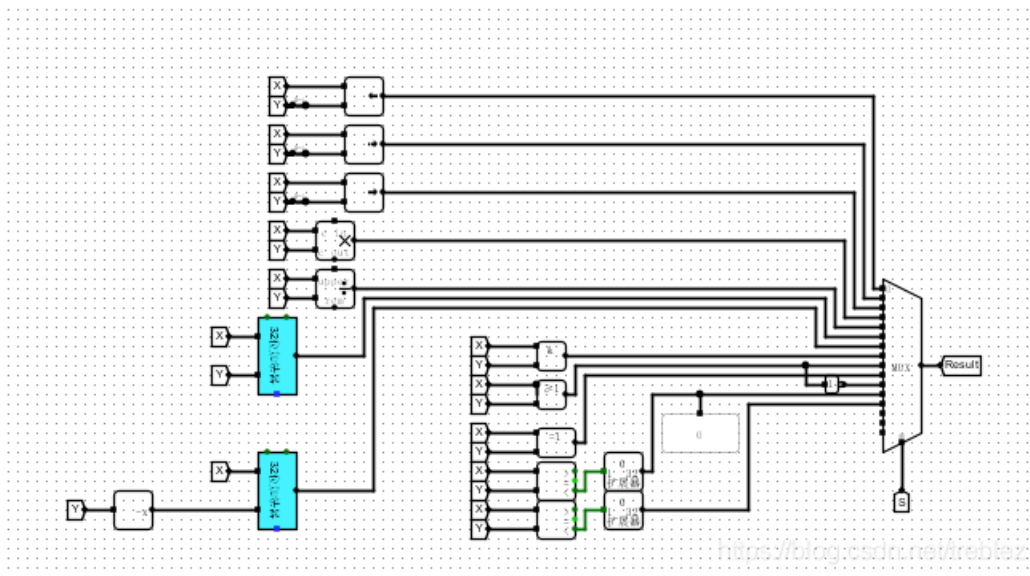


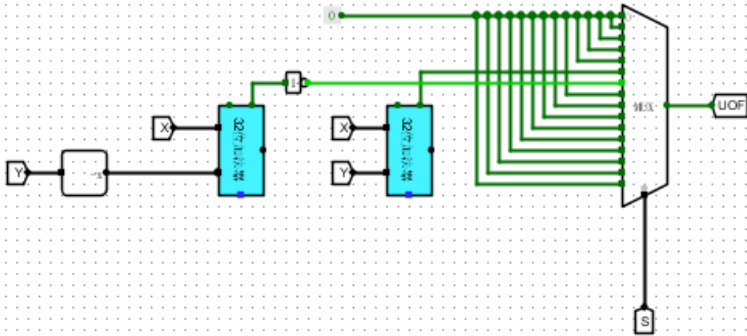
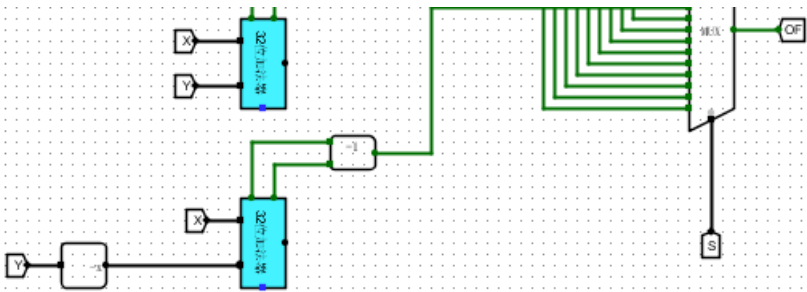
4.无符号溢出加法是最高位进位，减法取反

电路图

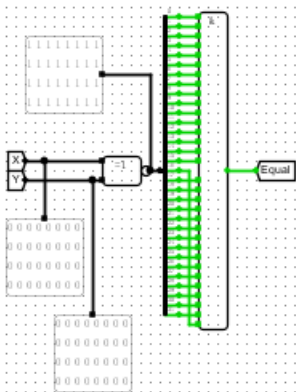
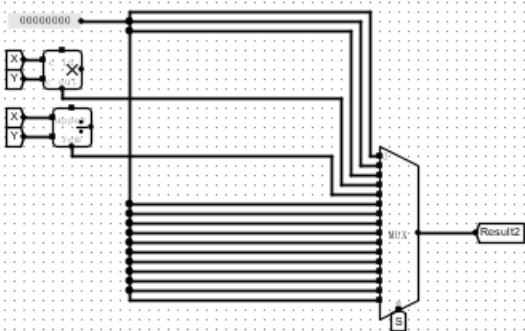


一包茶一根烟，一个电路连一天





<https://blog.csdn.net/febiez>



<https://blog.csdn.net/febiez>

文件 命令行 已保存

```
<a name="tristate" val="false"/>
<a name="pull" val="none"/>
<a name="label" val="a"/>
<a name="labelloc" val="north"/>
<a name="labelfont" val="Dialog plain 12"/>
<a name="labelcolor" val="#000000"/>
</comp>
<comp lib="a" loc="(40,80)" name="F">
  <a name="facing" val="east">
  <a name="output" val="1">
  <a name="width" val="1">
  <a name="tristate" val="false">
  <a name="pull" val="none">
  <a name="label" val="a">
  <a name="labelloc" val="north">
  <a name="labelfont" val="Dialog plain 12">
  <a name="labelcolor" val="#000000">
</comp>
</circuit>
/project>
```



完成