

实验报告一 运算器

姓名	王磊	学号	202231060435
专业	计算机科学与技术	年级	2022 级

任务一

1、全加器的设计

首先，我们先要确定全加器的输入和输出变量。全加器就是在半加器的基础上引入一个进位输入，总共三输入两输出：输入 A,B,cin，输出 sum,cout。全加器真值表如下：

输入			输出	
A	B	cin	sum	cout
0	0	0	0	0
0	1	0	1	0
1	0	0	1	0
1	1	0	0	1
0	0	1	1	0
0	1	1	0	1
1	0	1	0	1
1	1	1	1	1

$$\text{sum} = A \oplus B \oplus \text{cin}, \text{cout} = A \cdot B + B \cdot \text{cin} + A \cdot \text{cin}$$

然后，在 Logisim 中绘制全加器的电路图，并进行仿真验证。

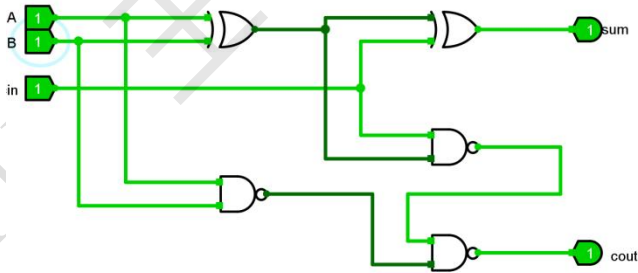


图 1 全加器的电路图

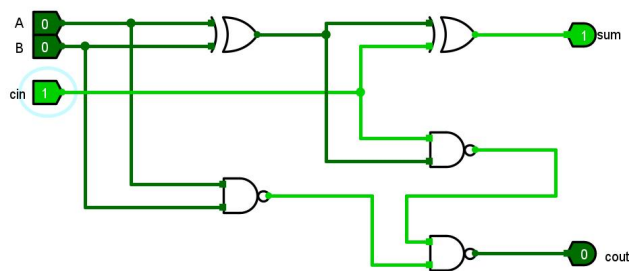


图 2 输入 001，结果正确显示 10

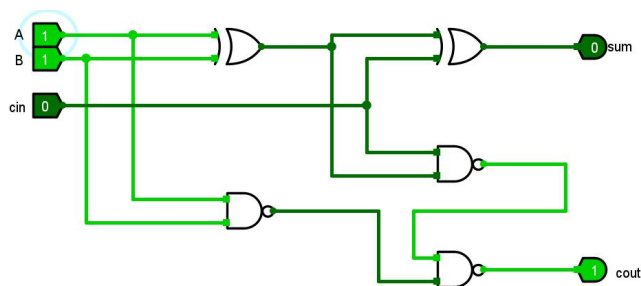


图 3 输入 110，结果正确显示 01

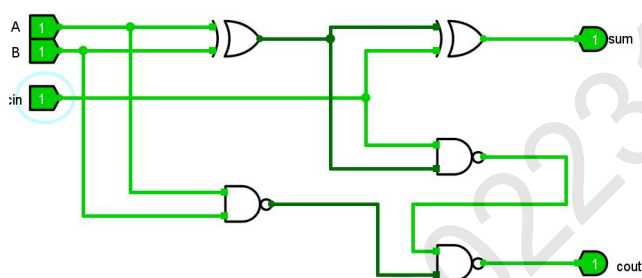


图 4 输入 111，结果正确显示 11

仿真结果显示，对于三种可能的输入组合（001，110，111），全加器都能正确输出与进位。

2、半加器的设计

首先，我们需要确定半加器的输入和输出变量。半加器，就是不考虑进位的加法器，只有两个输入或输出：输入 A 和 B，输出和数 sum 和进位 cout，半加器真值表如下：

A	B	sum	cout
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$$S=A \oplus B, C=AB$$

然后，在 Logisim 中绘制半加器的电路图，并进行仿真验证。

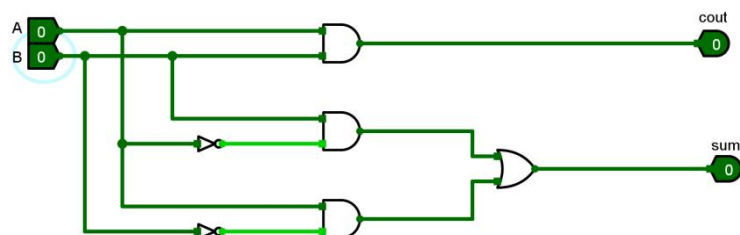


图 5 半加器的电路图

当 A=0, B=0 时, 实验输出结果为 $cout=0$, $sum=0$, 结果一致

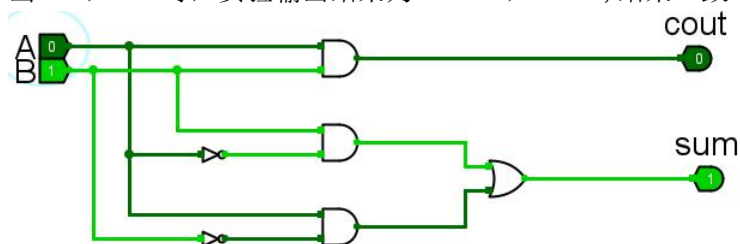


图 6 半加器的电路图

当 A=0, B=1 时, 实验输出结果为 $cout=0$, $sum=1$ 结果一致

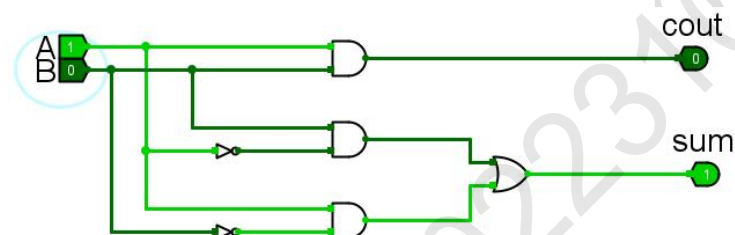


图 7 半加器的电路图

当 A=1, B=0 时, 实验输出结果为 $cout=0$, $sum=1$, 结果一致

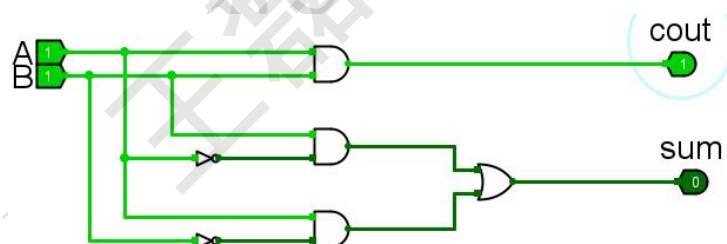


图 8 半加器的电路图

当 A=1, B=1 时, 实验输出结果为 $cout=1$, $sum=0$, 结果一致

仿真结果显示, 对于四种可能的输入组合 (00, 01, 10, 11), 半加器都能正确输出和与进位。

3、带进位输出的 4 位二进制加法计数器: 三个全加器一个半加器

首先, 将两个四位二进制数从低位到高位依次相加。对于最低位, 我们使用一个半加器

进行相加；对于其余位，我们使用全加器进行相加，并将前一位的进位作为当前位的 Cin 输入到全加器中。这样，通过三个全加器和一个半加器的级联，我们可以实现两个四位二进制数的加法运算。

在 Logisim 中进行仿真验证时，我们可以将八个一位二进制数作为输入，通过一个半加器和三个全加器的级联实现加法运算：

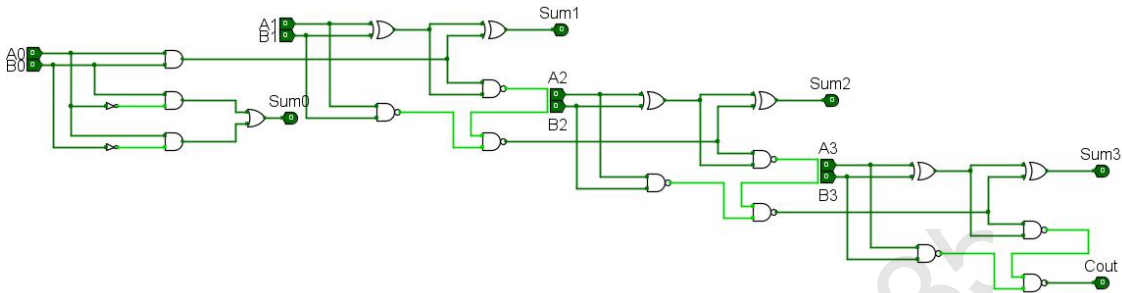


图 9 四位二进制数串行加法器的电路图

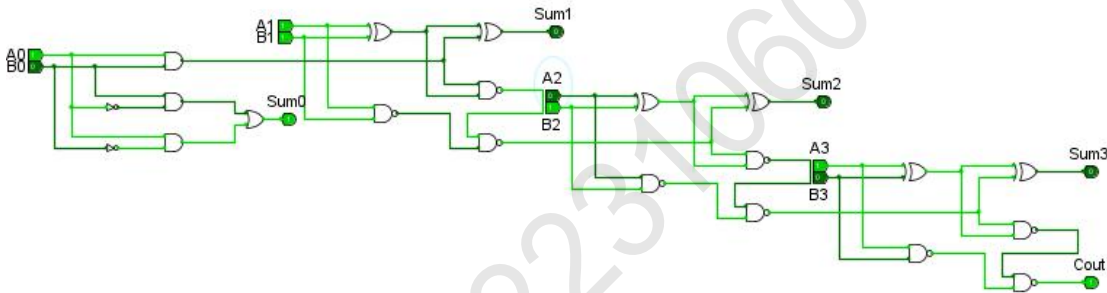


图 10 四位二进制数串行加法器的电路图

当 A=1011, B=0110 时，实验输出结果为 cout=1, sum=0001, 即为 10001，结果一致

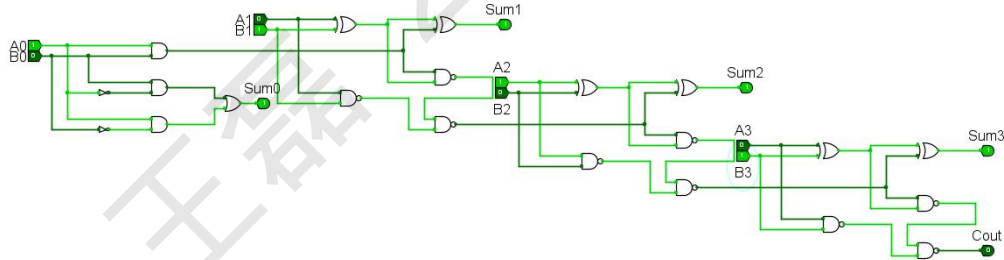


图 11 四位二进制数串行加法器的电路图

当 A=0101, B=1010 时，实验输出结果为 cout=0, sum=1111，结果一致

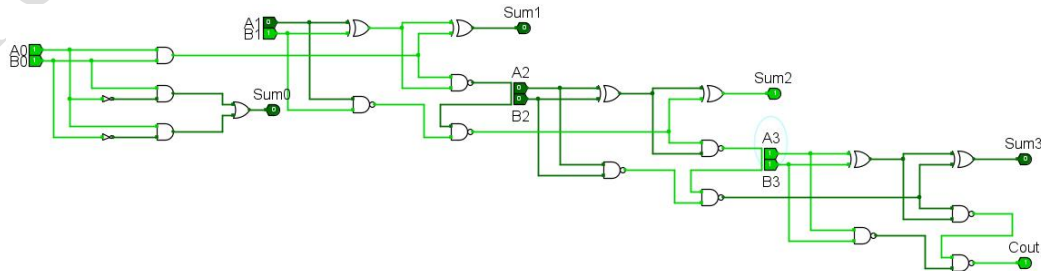


图 12 四位二进制数串行加法器的电路图

当A=1001, B=1011 时, 实验输出结果为 cout=1, sum=0100, 即为 10100, 结果一致
对于三种可能的输入组合 (1011 0110, 0101 1010, 1001 1011), 加法计数器都能正确输出和进位。

任务二

一、可控 8 位的加减法运算器:

要实现 8 位可控加减法器, 可以通过实验所给的 8 个一位全加器串行来实现加法, 然后通过可控反向异或门按位取反, 使 Sub 通过置 0 置 1 来实现做减法时减数的补码 即 $X + Y = X + Y + 0$, $X - Y = X + \sim Y + 1$ 。异或门既能实现 Y 和 $\sim Y$, 同时还能作为 Cin 的 0、1 信号的输入。Overflow (OF) 为运算的有符号溢出检测, 通过最高位进位 Cn 与符号位进位 Cf 异或: 若相同, 则无溢出为 0; 若不同, 则有溢出为 1。

8 位全加器包含 8 个 1 位全加器, 将 8 个一位全加器串联, 低位进位输出 Cout 连接到高位进位输入 Cin 中, 实现逐位从右向左传递, 如下图所示。

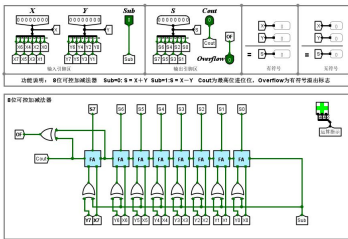


图 12 8 位可控加减运算器

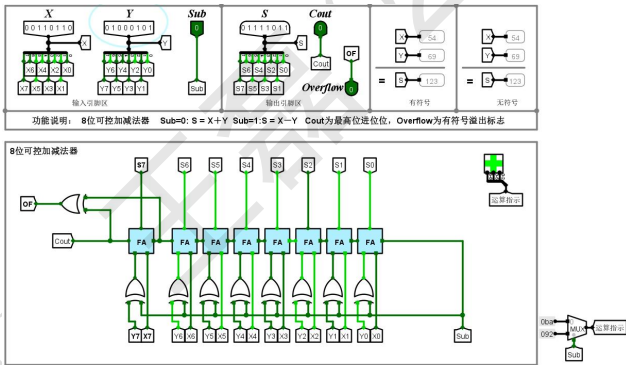


图 13 8 位可控加减运算器

当 X=00110110, Y=01000101, Sub=0 时, 实验输出结果为 S=01111011, Cout=0, OF=0, 结果一致

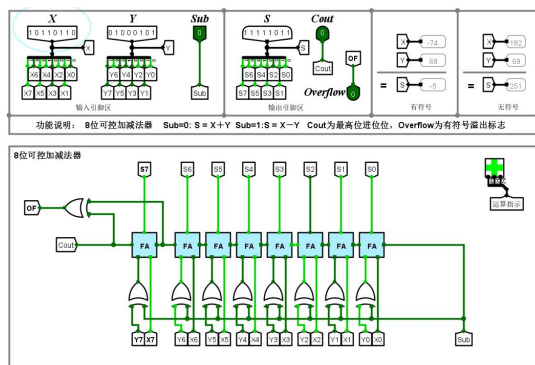


图 14 8 位可控加减运算器

当 $X=10110110$, $Y=01000101$, $Sub=0$ 时，做加法运算，实验输出结果为 $S=11111011$, $Cout=0$, $OF=0$ ，结果一致

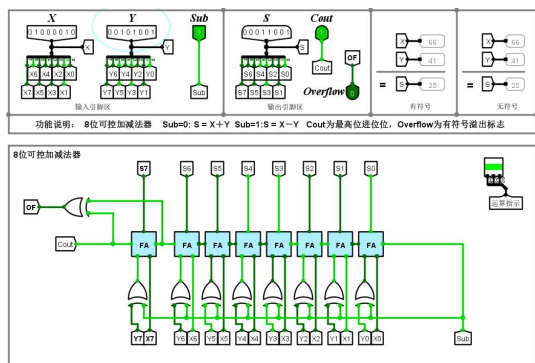


图 15 8 位可控加减运算器

当 $X=01000010$, $Y=00101001$, $Sub=1$ 时，做减法运算，实验输出结果为 $S=00011001$, $Cout=1$, $OF=0$ ，结果一致

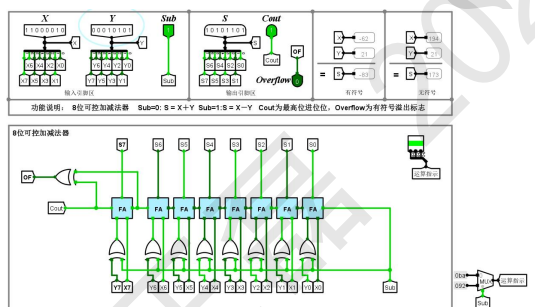


图 16 8 位可控加减运算器

当 $X=11000010$, $Y=00010101$, $Sub=1$ 时，做减法运算，实验输出结果为 $S=10101101$, $Cout=1$, $OF=0$ ，结果一致

对于四种可能的输入组合，加减运算器都能正确输出和进位。

二、双控制端 8 位算术/逻辑运算器

1、首先与或以 S_0 一个量决定，于是我打算封装一个可以通过一个输入接口控制选择做 8 位与、或运算的电路部件，如图所示：

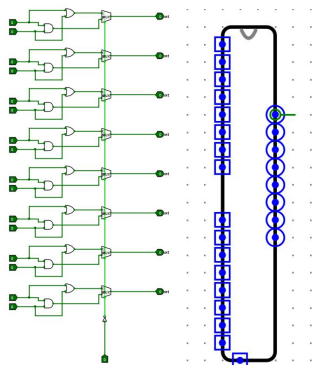


图 17 S0 选择器

2、 加减方面直接可以设计一个有进位、溢出输出的可控八位加减电路，同时可以通过一个变量 S0 同时控制与或里的筛选变量和加减里的变量 sub
加减电路如图：

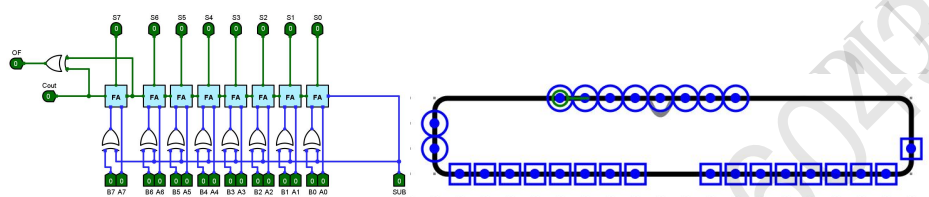


图 18 8 位可控加减运算器

3、最后由于要通过 S2 对与或、加减做进一步筛选，我选择设计一个 8 位选择器，并进行封装，Z 为判断结果是否为 0，如是则 Z 值为 1，否则为 0
如图：

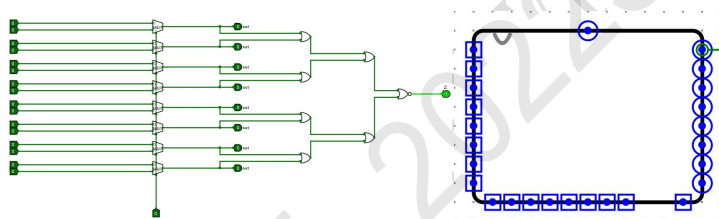


图 19 S1 选择器

4、将以上几部分组合起来：

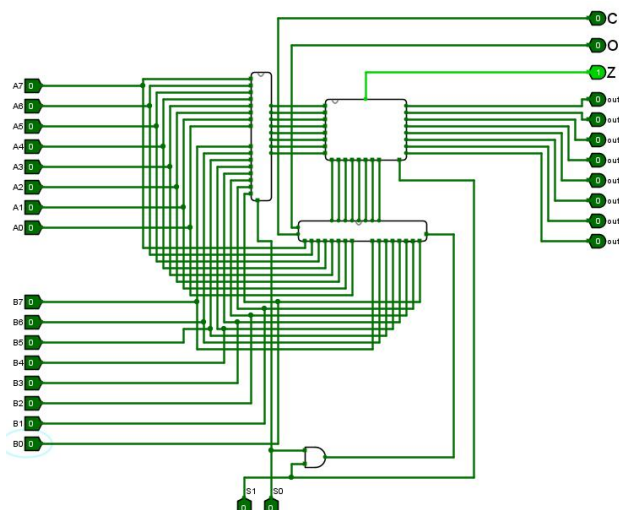
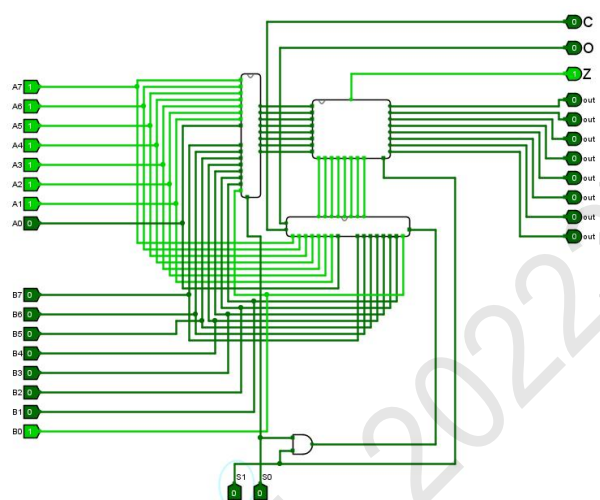
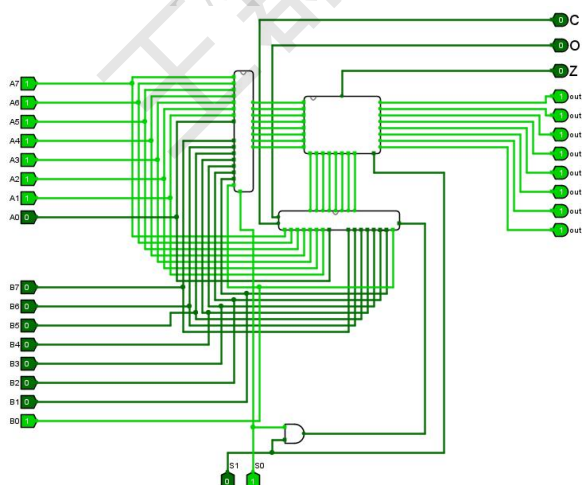


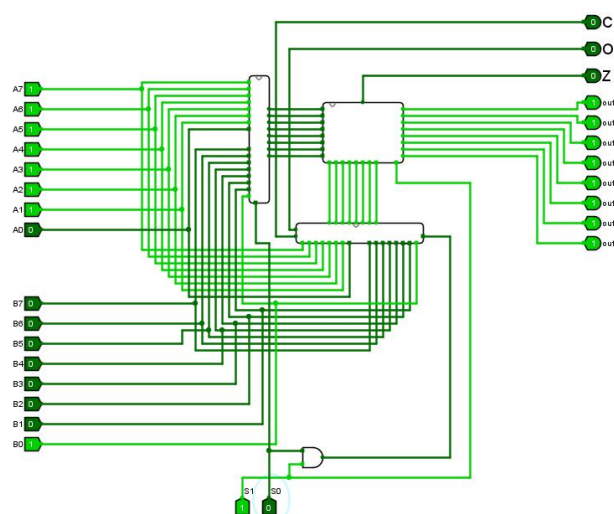
图 20 双控制端 8 位算术/逻辑运算器



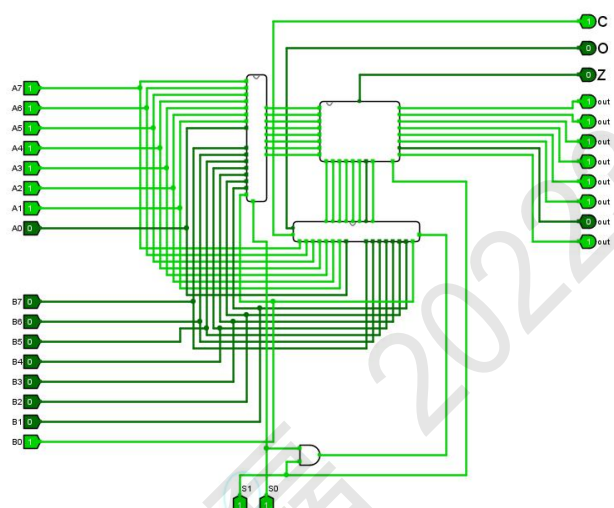
当 $A=11111110$, $B=00000001$, $S0=S1=0$ 时, 做与运算, 实验输出结果为 $out=00000000$, $ZF=1$, $OF=0$, $CF=0$, 结果一致



当 $A=11111110$, $B=00000001$, $S0=1$, $S1=0$ 时, 做或运算, 实验输出结果为 $out=11111111$, $ZF=0$, $OF=0$, $CF=0$, 结果一致



当 $A=11111110$, $B=00000001$, $S0=0$, $S1=1$ 时, 做加法运算, 实验输出结果为 $out=11111111$, $ZF=0$, $OF=0$, $CF=0$, 结果一致



当 $A=11111110$, $B=00000001$, $S0=0$, $S1=1$ 时, 做减法运算, 实验输出结果为 $out=11111101$, $ZF=0$, $OF=0$, $CF=1$, 结果一致

对于四种可能的输入组合, 算术/逻辑运算器都能正确输出和进位。