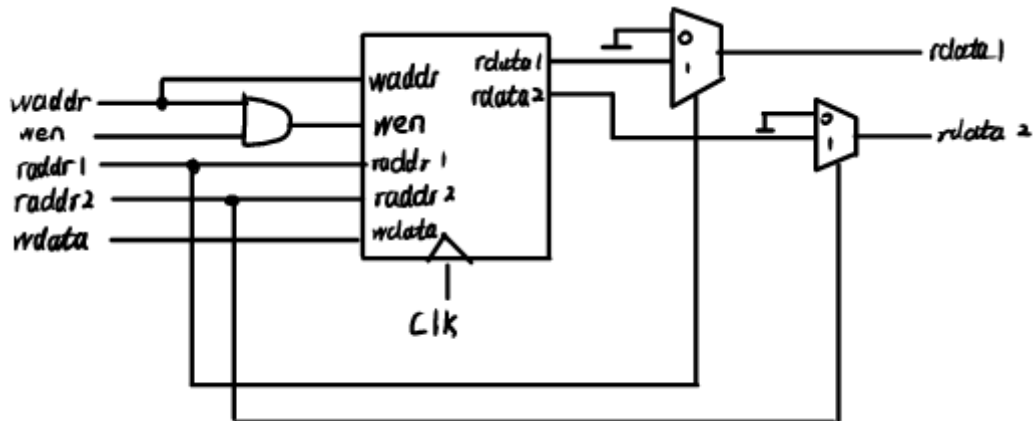


3) 逻辑电路结构图



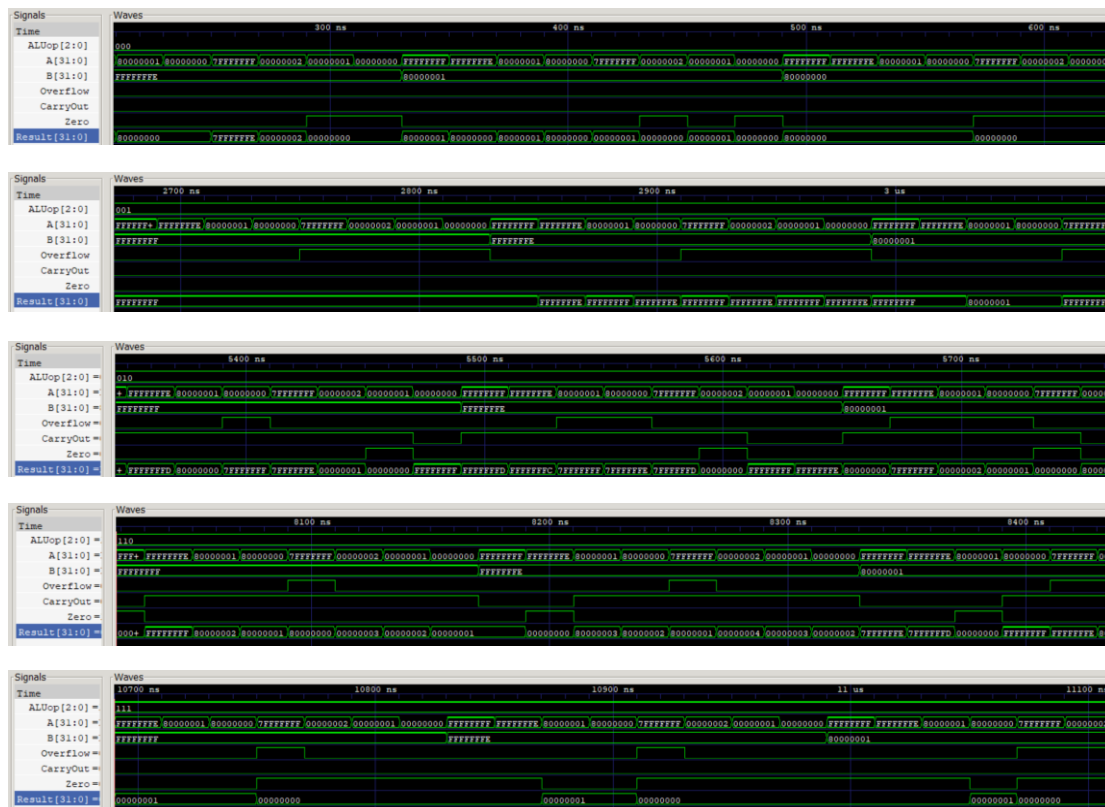
2. ALU 部分

1) 关键 RTL 代码

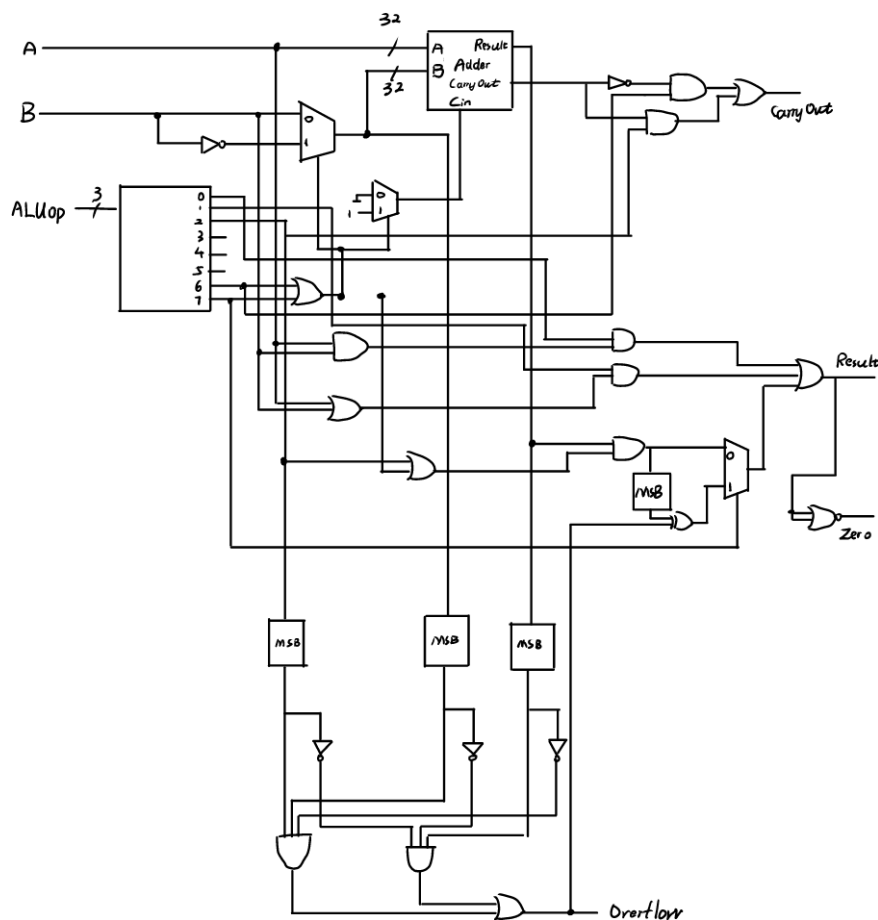
```
wire [`DATA_WIDTH - 1:0]oper_num = {`DATA_WIDTH{op_add}} & B |  
{`DATA_WIDTH{op_sub| op_slt}} & (~B);  
wire Cin = op_sub| op_slt;  
assign {CarryOut_init, add_res} = A + oper_num + Cin;  
assign Overflow = A[`DATA_WIDTH - 1] & oper_num[`DATA_WIDTH - 1] &  
~Result_tmp[`DATA_WIDTH - 1] |  
~A[`DATA_WIDTH - 1] & ~oper_num[`DATA_WIDTH - 1] &  
Result_tmp[`DATA_WIDTH - 1];  
assign CarryOut = {op_add & CarryOut_init} | {op_sub & ~CarryOut_init};
```

这一部分实现的是在创建一个 32 位 wire 型变量 `oper_num`, 它在加法时等于 `B`, 减法或比较时是 `-B` 的反码, 这是为了节省加法器, 同时方便 `overflow` 的赋值。另外一个临时变量 `Cin` 在减法或比较时为 1, 其他时为 0。则 `add_res` 为 `A+oper_num+Cin`, 由于 `Cin` 是 1 位 wire 型变量, 因此使用一个加法器就可以实现。由于取的是反码, 故只要 `A`, `oper_num` 符号位相同, 且与 `add_res` 的符号位相反, 则 `overflow` 为 1, 否则为 0。对于 `carry_out`, 则是在加法时与 `CarryOut_init` 相同, 而在减法时与之相反。具体原因在第三部分阐述。

2) 仿真波形截图



3) 逻辑电路结构图



二、 实验过程中遇到的问题、对问题的思考过程及解决方法

在设计 ALU 部件时,我认为 overflow 和 carry-out 的输出的设计是难点。首先说相对简单的 carry-out,由于 carry-out 面向的是无符号整数,在考虑设计如何判断有进位和借位时有困难。后来我的解决思路是在加法部分增添一位,得到一个 33 位的结果,其第一位时 carry-out,后面 32 位是 Result (注意,为了避免组合逻辑环,这里所述 carry-out 和 Result 分别代表另设的两个 wire 型变量,称为 carry-out-tem 和 Result-tem)。加法时显然当 carry-out-tem 和 carry-out 值相等,需要考虑的是减法,由于运算数都是正数,结合补码的计算规律,当 A 比 B 小时, A 加-B 的补码会得到比 33 位 1 还大的数,发生溢出,此时 carry-out-tem 是 0;相反 carry-out-tem 则为 1,由于当产生借位时 carry-out 才应为 1,故在减法或比较时, carry-out 应为 carry-out-tem 的取反。

在 overrrflow 部分,我一开始想的是另设一个 32 位 wire 型变量 oper_tem,它在加法时等于 B,减法时等于-B 的补码,overflow 则在 A, oper_tem 符号位相同,而与 Result_tem 符号位相反时为 1。但这个方法会在 B 为 80000000,即绝对值最大的负数时出错,这是因为他不存在对应的补码,对它的负数进行补码取值时,符号位是错误的 1。我的解决方法是咋 oper_num 赋值部分将求-B 的补码改为求反码,在加法时引入新 wire 型变量 Cin,加法时 Cin 为 0,减法时 Cin 为 1,加法时使用 $A + \text{oper_tem} + \text{Cin}$,这样减少了求补码时用的一个加法器,只用一个加法器就可以实现。在 overflow 判断部分,我们只关注符号位,此时 oper_num 一定会与 B (加法) 或-B (减法或比较) 的符号位相等,则 overflow 按照

以前的计算方法就正确了。

三、 在课后，你花费了大约 2 小时完成此次实验。

其中寄存器堆所用时间较少，出错主要是因为对没把握准设计要求，ALU 设计时间则较长。

四、 对于此次实验的心得、感受和建议

本次实验难度不大，但具有一定的挑战。如第二部分所述，设计 ALU 过程中也碰到了一些困难。ALU 的思想较为简单，但具体设计时也有一些特殊情况需要具体考虑。感谢我的舍友贾城昊在设计过程中与我的讨论，使得我们共同解决了一些 bug，提供了一些设计思路。感谢老师的悉心指导。