64-bit Integer ALU Design.

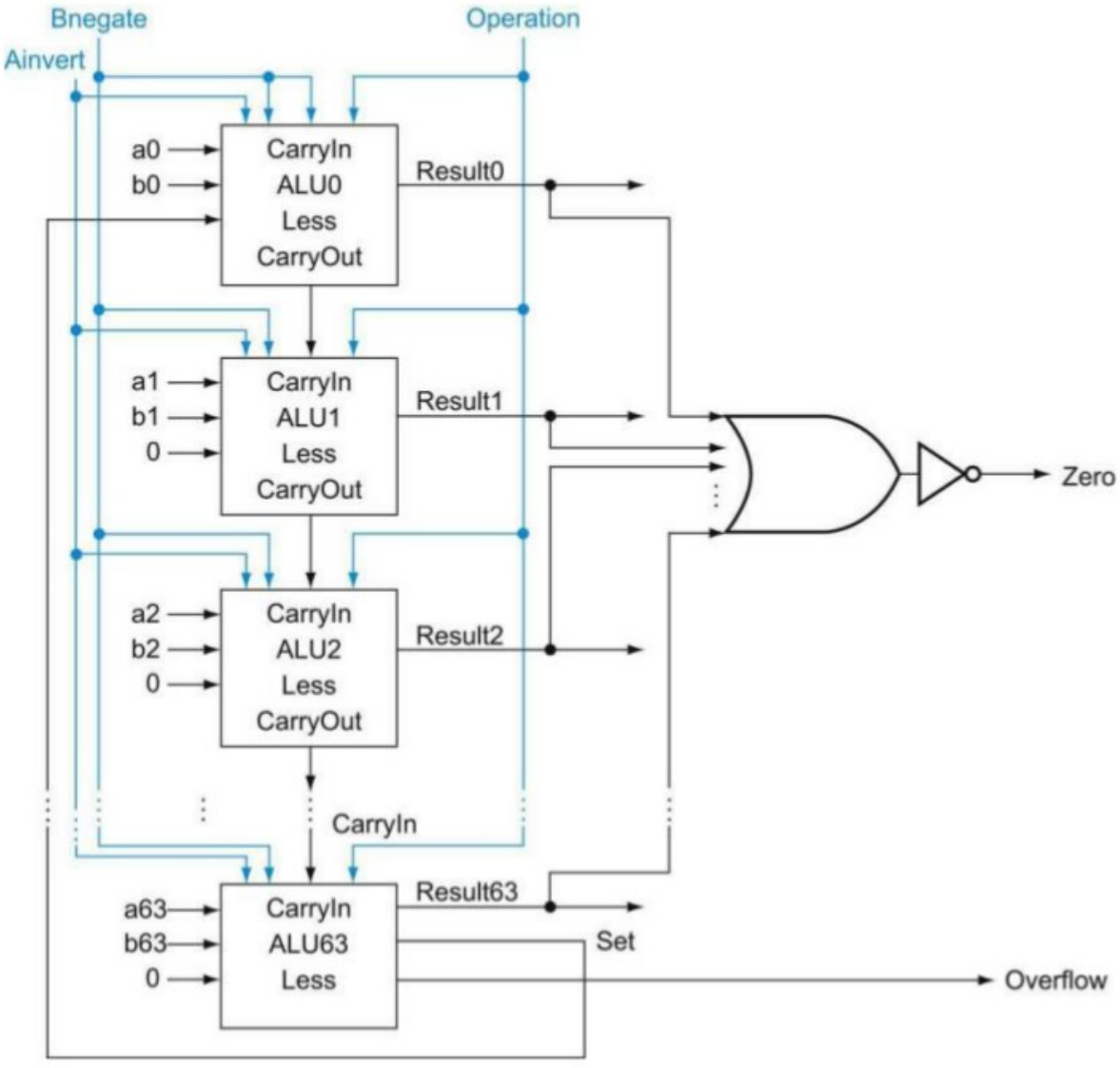
Extending the 1-bit ALU to support the same tasks on 64-bits. One change done is the passing the carry bit generated during addition/subtraction to the next 1-bit ALU cycle.

OP Codes:

4-bits
bit 1 : neg_A
bit 2 : neg_B
bit 3,4 : indicate component
00 => and
01 => or
10 => add

The main module accepts 4 inputs from the user. The input 1, input 2, carry_in and the opcode.

BLOCK DIAGRAM :



VERILOG CODE :

```
t View Search Terminal Help
        Sushruth V      17CO148
        24 October 2018

        Op-Code : 4-bits
        bit 1 : neg_A
        bit 2 : neg_B
        bit 3,4 : indicate component
                                00 => and
                                01 => or
                                10 => add

 */

 module ALU(output reg[63:0] Result, output reg Zero, output reg CarryOut, input[63:0] A, input[63:0] B, input CarryIn, input[3:0] Op);
     reg[63:0] a;
        reg[63:0] b;
     integer i;
        always@(A, B, CarryIn, Op) begin
               for(i = 0; i < 64; i++) begin
            a[i] = (~A[i] & Op[3]) | (A[i] & ~Op[3]);
                  b[i] = (~B[i] & Op[2]) | (B[i] & ~Op[2]);
         end
               if (Op[1:0] == 2'b00) begin
                   for(i = 0; i < 64; i++)
                Result[i] = a[i] & b[i];
               end
               if (Op[1:0] == 2'b01) begin
                      for(i = 0; i < 64; i++)
                Result[i] = a[i] | b[i];
               end
               if (Op[1:0] == 2'b10) begin
                      {CarryOut, Result} = Op[2] + a + b;
               end
               if(Result != 0)
            Zero = 0;
         if(Result == 0)
            Zero = 1;
        end
 endmodule
```

SAMPLE OUTPUT :

```
Time =    8
A =           1           B =           2
Op Code = 0100            Carry In =    0

Result = 0000000000000003                 Zero =    0               Carry Out = 0

Time =    9
A =           1           B =           2
Op Code = 0100            Carry In =    1

Result = 0000000000000001                 Zero =    0               Carry Out = 0

Time =   10
A =           1           B =           2
Op Code = 0101            Carry In =    0

Result = 0000000000000001                 Zero =    0               Carry Out = 0

Time =   11
A =           1           B =           2
Op Code = 0101            Carry In =    1

Result = fffffffffffffffd                 Zero =    0               Carry Out = 0

Time =   12
A =           1           B =           2
Op Code = 0110            Carry In =    0

Result = fffffffffffffffd                 Zero =    0               Carry Out = 0

Time =   13
A =           1           B =           2
Op Code = 0110            Carry In =    1

Result = ffffffffffffffff                 Zero =    0               Carry Out = 0

Time =   14
A =           1           B =           2
Op Code = 0111            Carry In =    0

Result = ffffffffffffffff                 Zero =    0               Carry Out = 0
```