

*Tyler Valkavich*  
CSE 341  
Person #: 50183135

# *16-Bit*

# *ALU*

# *Write Up*

# *Table of Contents*

**1) Circuit Diagrams of the ALU down to the gate level**

*Pages 3 - 6*

**2) Critical path and delay of the ALU**

*Page 7*

**3) Gate cost of the ALU**

*Page 8*

**4) Functional (zero delay) simulation results for the ALU**

*Pages 9 - 16*

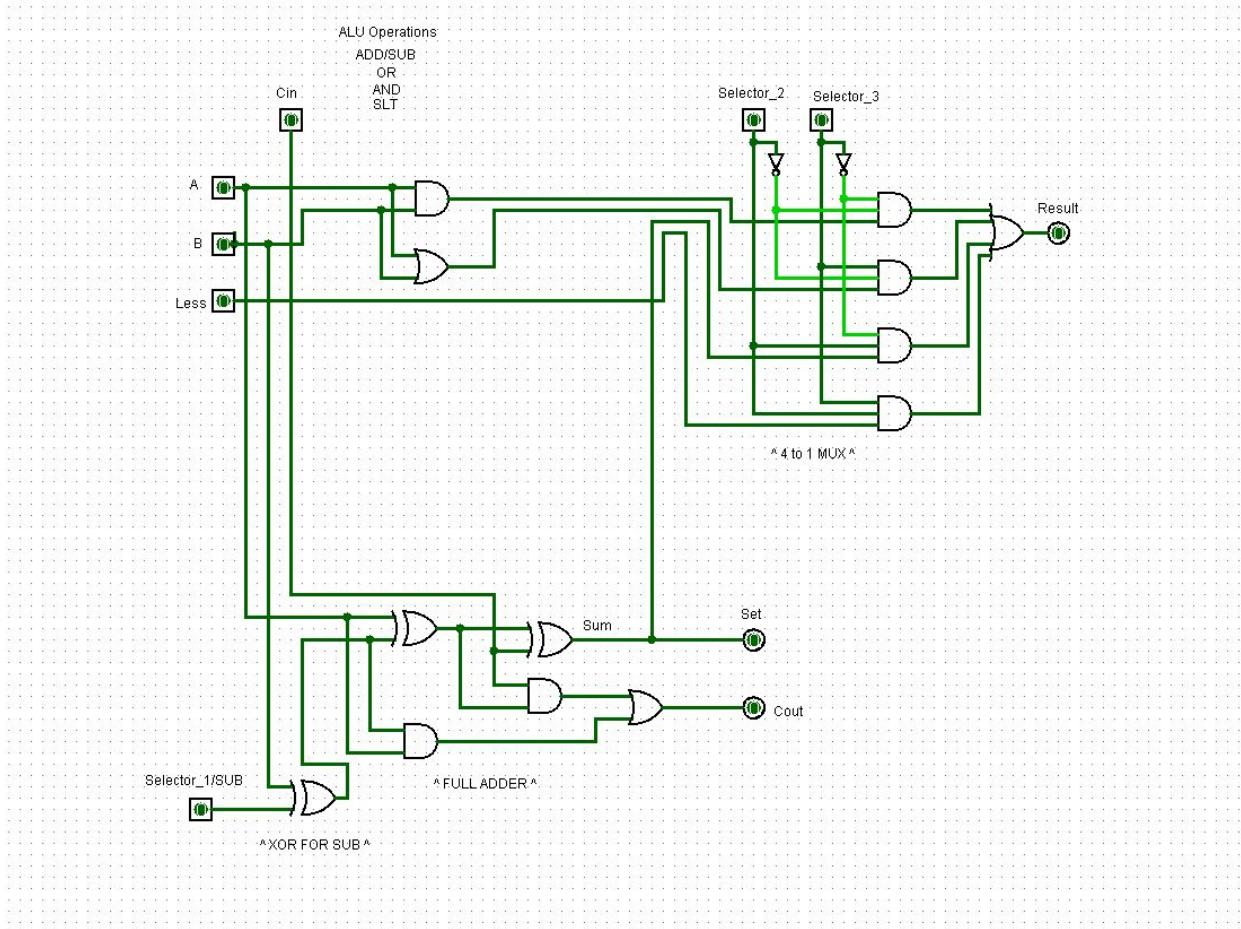
**5) Analysis of the temporal dependence of the delay on the previous set of inputs:**

*Pages 17 - 24*

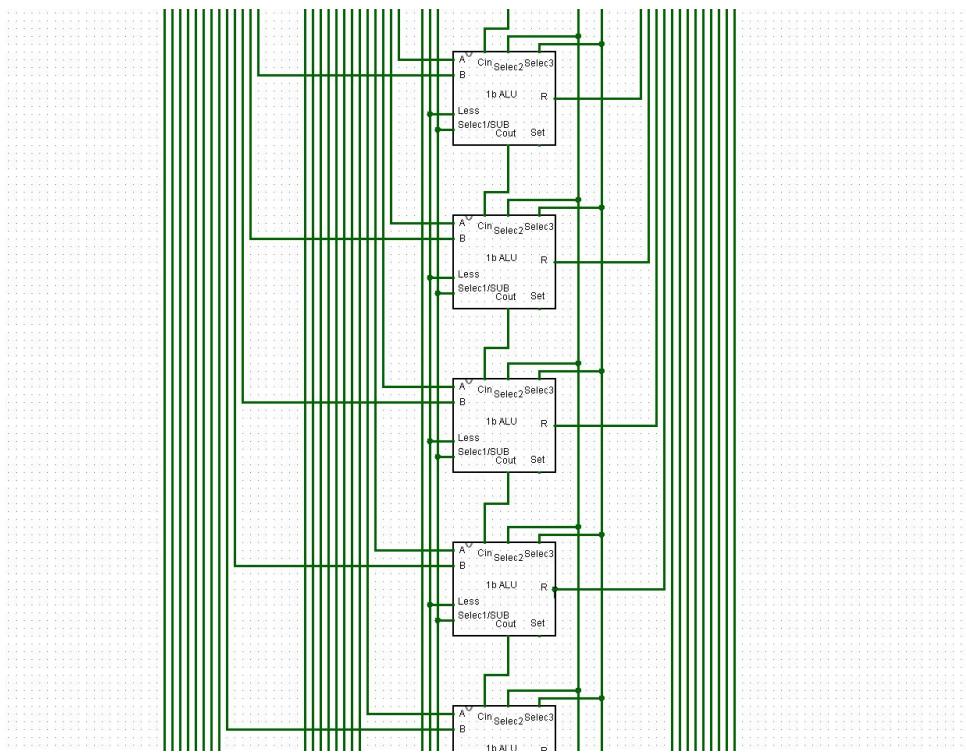
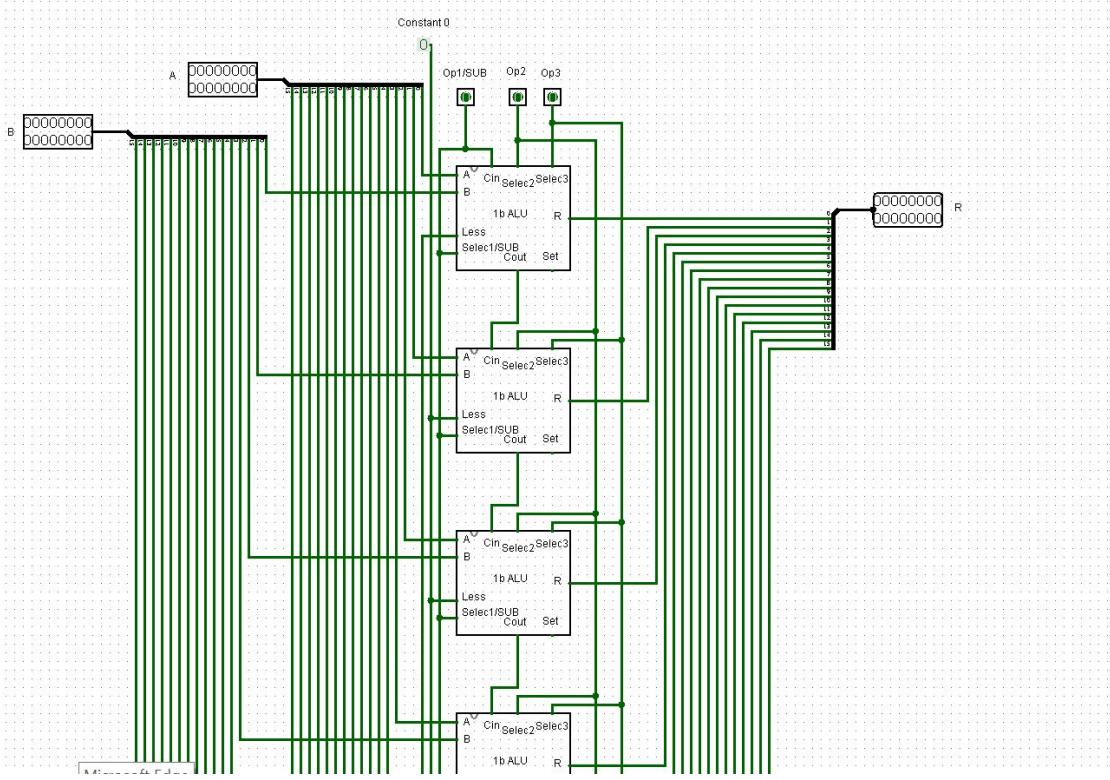
**6) Analysis of the average delay across 1,000 random input patterns:**

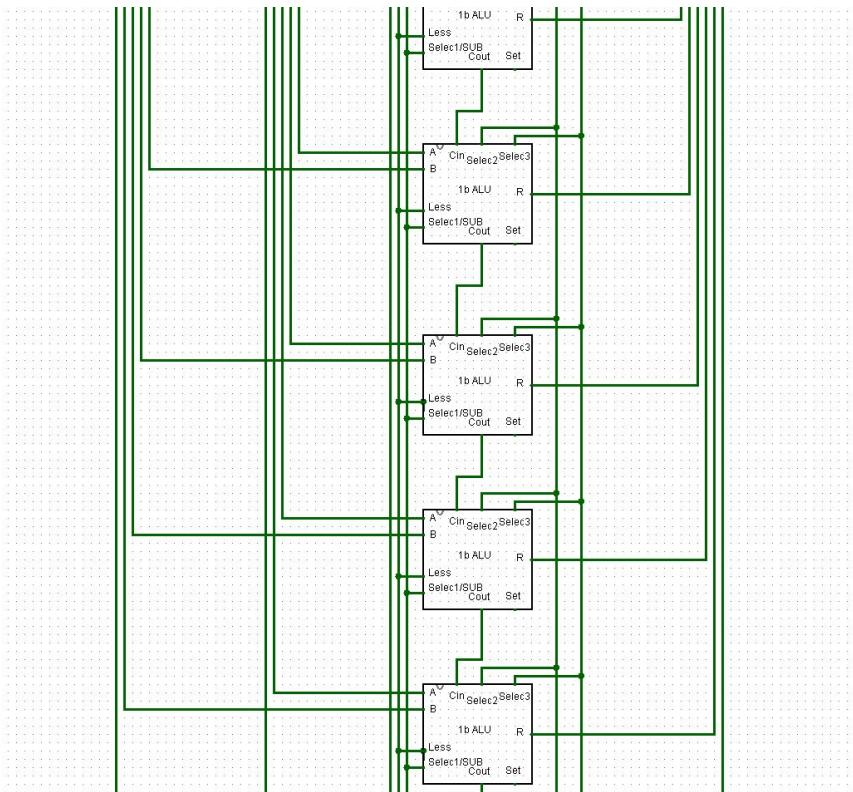
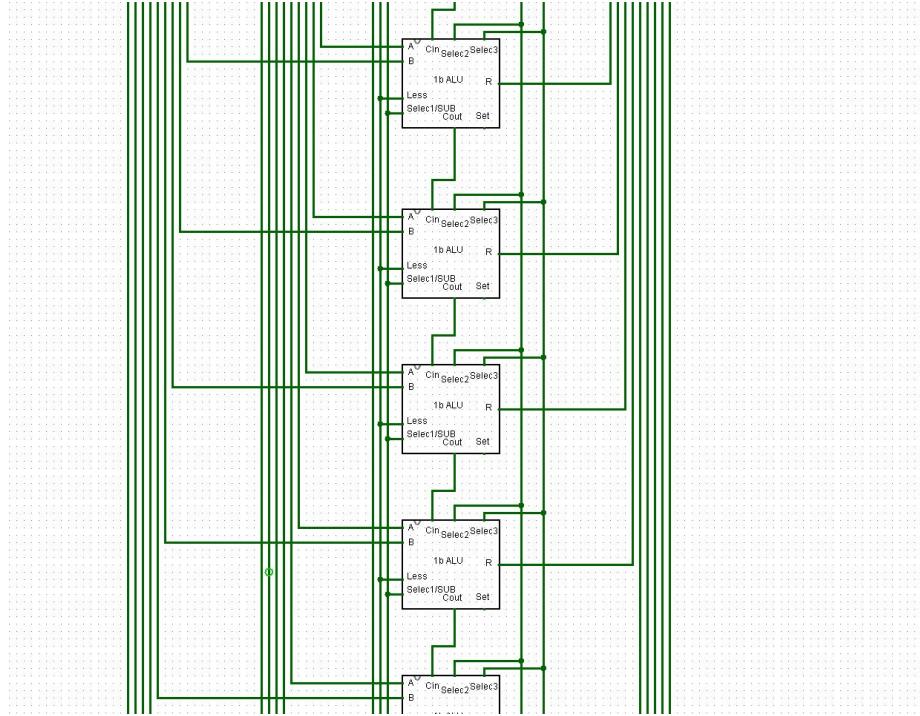
*Page 25*

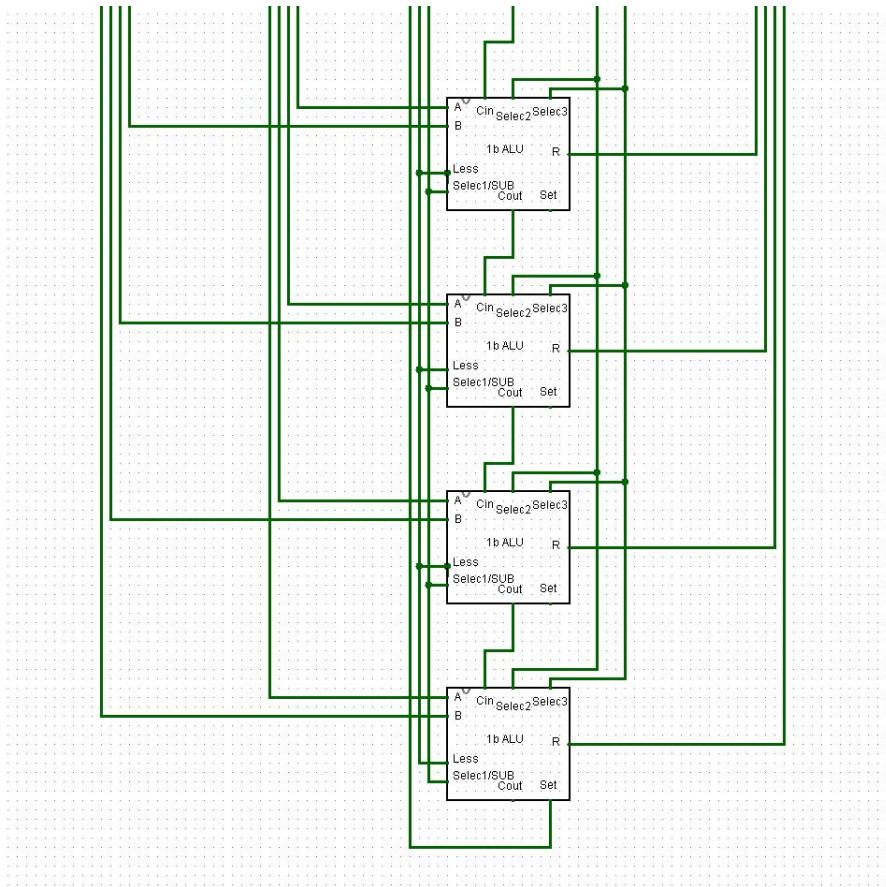
### 1) Circuit Diagrams of the ALU down to the gate level:



The diagram above is a circuit diagram of a 1-bit ALU down to the gate level used to make a 16-bit ALU.

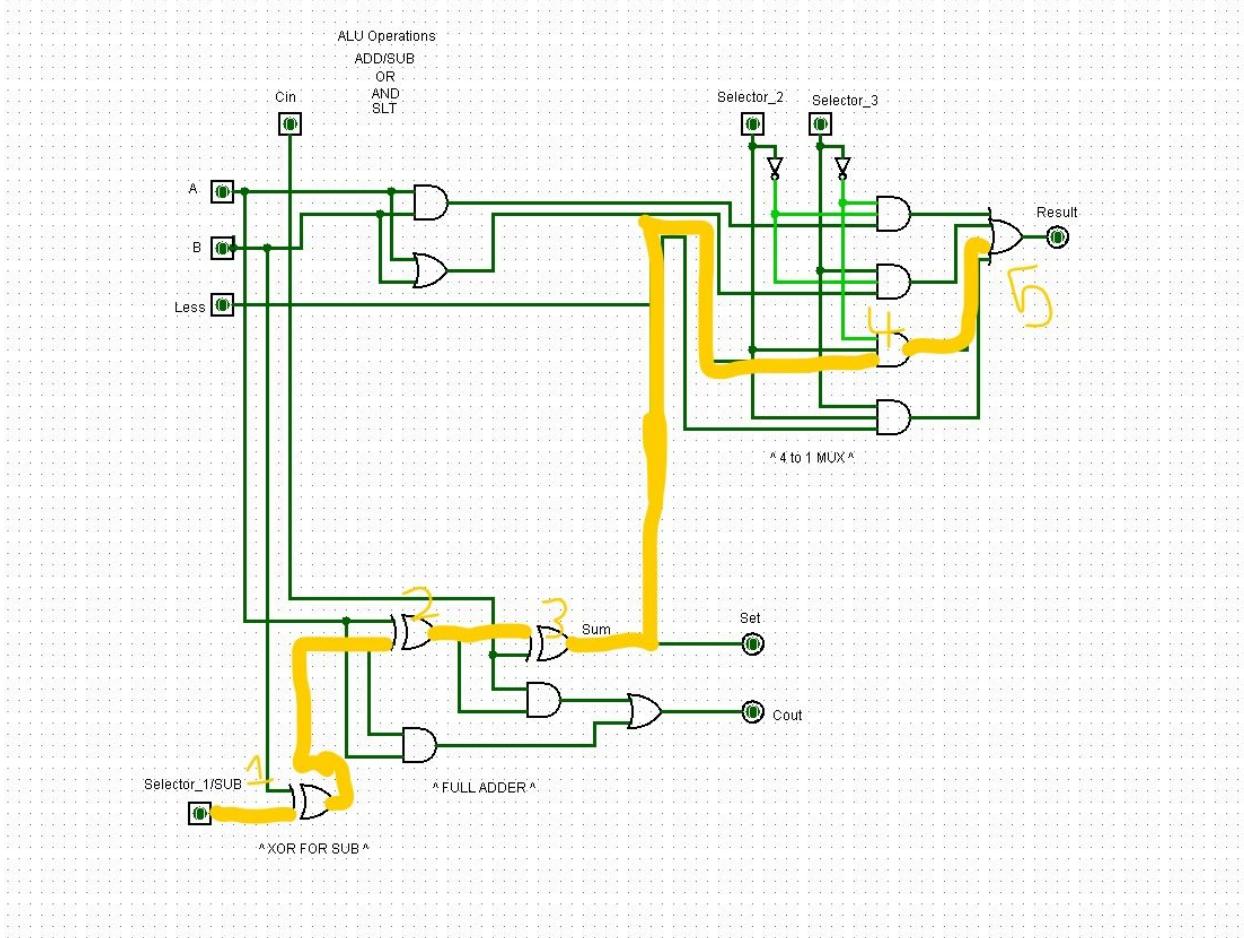






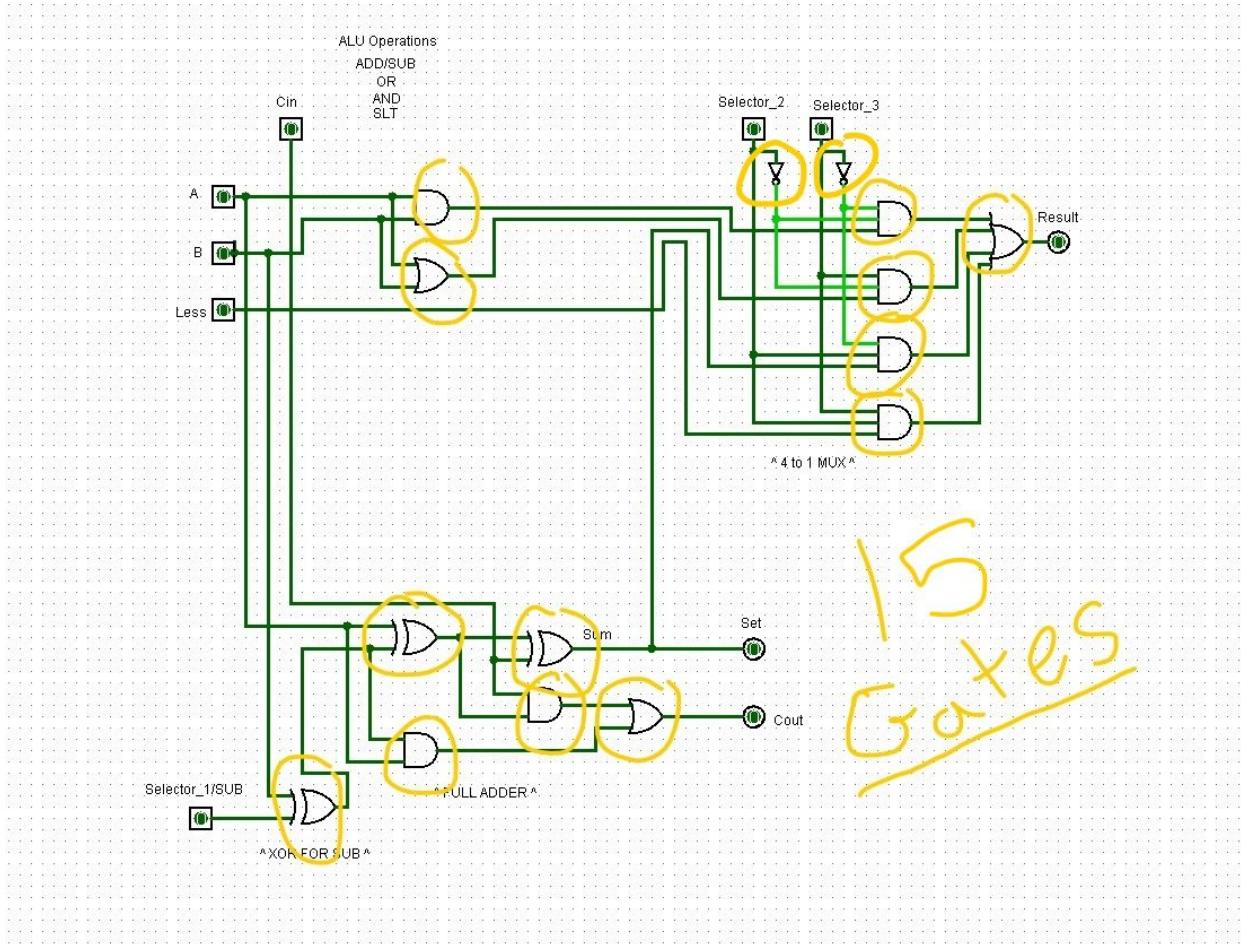
The diagrams above are sixteen 1-bit ALUs wired together to form the entire 16-bit ALU

## 2) Critical path and delay of the ALU under the unit gate delay model:



The diagram above is the critical path for a 1-bit ALU within a 16-bit ALU. The delay in the path for a 1-bit ALU is 5 nanoseconds therefore the longest delay within a 16-bit ALU would be 80 nanoseconds if a user were to input a number that would occupy all 16-bits of the ALU.

### 3) Gate cost of the ALU:



The above diagram of a 1-bit ALU contains 15 gates, therefore, the total gate cost of the 16-bit ALU is 240 gates because there is 15 gates in each of the 16 individual ALUs that make up a single 16-bit ALU.

## 4) Functional (zero delay) simulation results for the ALU:

And:

Positive Inputs:

```

timberlake {~/Documents} > ./a.out
0 A=xxxxxxxxxxxxxxxxxx, B=xxxxxxxxxxxxxxxxxx, Cin=x, F=xxxxxxxxxxxxxxxxzzz, Cout=zxxxxxxxxxxxxxzzz, Set=zxxxxxxxxxxxxxzzz, Sub=x, Op1=x, Op2=x, Less=x
0 A=0001001001001100, B=0111010011010110, Cin=0, F=zxxxxxxxxxxxxxzzz, Cout=xxxxxxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxxxxxx, Sub=0, Op1=0, Op2=0, Less=0
0 A=0001001001001100, B=0111010011010110, Cin=0, F=zxxxxxxxxxxxxxzzz, Cout=xxxxxxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxxxxxx, Sub=0, Op1=0, Op2=0, Less=0
1 A=0001001001001100, B=0111010011010110, Cin=0, F=xxxxxxxxxxxxxxxxxx, Cout=xxxxxxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxxxxxx, Sub=0, Op1=0, Op2=0, Less=0
2 A=0001001001001100, B=0111010011010110, Cin=0, F=xxxxxxxxxxxxxxxxxx, Cout=xxxxxxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxxxxxx, Sub=0, Op1=0, Op2=0, Less=0
3 A=0001001001001100, B=0111010011010110, Cin=0, F=00010000000000110, Cout=0xx1lx0xxxxxu110, Set=xxxxxxxxxxxxxx00, Sub=0, Op1=0, Op2=0, Less=0
4 A=0001001001001100, B=0111010011010110, Cin=0, F=00010000000000110, Cout=0xx1lx0xxxxxu110, Set=xx0x1xxxxx1100, Sub=0, Op1=0, Op2=0, Less=0
5 A=0001001001001100, B=0111010011010110, Cin=0, F=00010000000000110, Cout=0x1lx0xxxxx00110, Set=xx0x1xxxxx1100, Sub=0, Op1=0, Op2=0, Less=0
6 A=0001001001001100, B=0111010011010110, Cin=0, F=00010000000000110, Cout=0x1lx0xxxxx00110, Set=xx0x1xxxxx1100, Sub=0, Op1=0, Op2=0, Less=0
7 A=0001001001001100, B=0111010011010110, Cin=0, F=00010000000000110, Cout=011100xxx000110, Set=100011xxxx111100, Sub=0, Op1=0, Op2=0, Less=0
8 A=0001001001001100, B=0111010011010110, Cin=0, F=00010000000000110, Cout=011100xxx000110, Set=100011xxxx111100, Sub=0, Op1=0, Op2=0, Less=0
9 A=0001001001001100, B=0111010011010110, Cin=0, F=00010000000000110, Cout=011100xxx000110, Set=100011xxxx111100, Sub=0, Op1=0, Op2=0, Less=0
10 A=0001001001001100, B=0111010011010110, Cin=0, F=00010000000000110, Cout=011100xxx000110, Set=100011xxxx111100, Sub=0, Op1=0, Op2=0, Less=0
11 A=0001001001001100, B=0111010011010110, Cin=0, F=00010000000000110, Cout=011100x0xxxx00110, Set=100011xxxx111100, Sub=0, Op1=0, Op2=0, Less=0
12 A=0001001001001100, B=0111010011010110, Cin=0, F=00010000000000110, Cout=011100x0xxxx00110, Set=100011xxxx111100, Sub=0, Op1=0, Op2=0, Less=0
13 A=0001001001001100, B=0111010011010110, Cin=0, F=00010000000000110, Cout=01110000000000110, Set=100011lx11111100, Sub=0, Op1=0, Op2=0, Less=0
14 A=0001001001001100, B=0111010011010110, Cin=0, F=00010000000000110, Cout=01110000000000110, Set=1000110111111100, Sub=0, Op1=0, Op2=0, Less=0
timberlake {~/Documents} >

```

Negative Inputs:

```

timberlake {~/Documents} > iverilog project2.v
timberlake {~/Documents} > ./a.out
0 A= x, B= x, Cin=x, F=xxxxxxxxxxxxxxxx, Cout=xxxxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxxxx, Sub=x, Op1=x, Op2=x, Less=x
0 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxxxx, Sub=0, Op1=1, Op2=0, Less=0
0 A=65486, B=65516, Cin=0, F= x, Cout=xxxxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxxxx, Sub=0, Op1=1, Op2=0, Less=0
1 A=65486, B=65516, Cin=0, F= x, Cout=xxxxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxxxx, Sub=0, Op1=1, Op2=0, Less=0
2 A=65486, B=65516, Cin=0, F= x, Cout=xxxxxxxxxxxxxx0, Set=xxxxxxxxxxxxxx0, Sub=0, Op1=1, Op2=0, Less=0
3 A=65486, B=65516, Cin=0, F= x, Cout=1111111111011x0, Set=xxxxxxxxxxxxxx10, Sub=0, Op1=1, Op2=0, Less=0
4 A=65486, B=65516, Cin=0, F= x, Cout=1111111111011x01, Set=111111111111110, Sub=0, Op1=1, Op2=0, Less=0
5 A=65486, B=65516, Cin=0, F= x, Cout=111111111101100, Set=111111111111110, Sub=0, Op1=1, Op2=0, Less=0
6 A=65486, B=65516, Cin=0, F= x, Cout=111111111101100, Set=111111111111110, Sub=0, Op1=1, Op2=0, Less=0
7 A=65486, B=65516, Cin=0, F= x, Cout=111111111101101, Set=111111111111110, Sub=0, Op1=1, Op2=0, Less=0
8 A=65486, B=65516, Cin=0, F=65466, Cout=111111111101101, Set=111111111111110, Sub=0, Op1=1, Op2=0, Less=0
timberlake {~/Documents} >

```

The tests above display that, for the inputs;

**Positives: A = 0, B = 1**

A = 1, B = 1

A = 637, B = 120

A = 3,555, B = 1,245

A = 10,545, B = 5,600

**Negatives: A = 50, B = -20**

A = -50, B = -20

Neither of the two input types output the desired results for the AND operation due to the lack of a delay between execution of instructions. Although 5 or 2 inputs were put into a test bench, the output always appears to concatenate into a single, incorrect result.

Or:

**Positive Inputs:**

```
timberlake ~/Documents) > ./a.out
0 A=xxxxxxxxxxxxxx, B=xxxxxxxxxxxxxx, Cin=x, F=zzzzzzzzzzzzzzz, Cout=zzzzzzzzzzzzzzz, Set=zzzzzzzzzzzzzzz, Sub=x, Opl=x, Op2=x, Less=x
0 A=0001100100100110, B=0110010011010110, Cin=0, F=zzzzzzzzzzzzzzz, Cout=xxxxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxxxx, Sub=0, Opl=0, Op2=1, Less=0
0 A=0001100100100110, B=0110010011010110, Cin=0, F=zzzzzzzzzzzzzzz, Cout=xxxxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxxxx, Sub=0, Opl=0, Op2=1, Less=0
1 A=0001100100100110, B=0110010011010110, Cin=0, F=xxxxxxxxxxxxxxxx, Cout=xxxxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxxxx, Sub=0, Opl=0, Op2=1, Less=0
2 A=0001100100100110, B=0110010011010110, Cin=0, F=xxxxxxxxxxxxxxxx, Cout=xxxxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxxxx, Sub=0, Opl=0, Op2=1, Less=0
3 A=0001100100100110, B=0110010011010110, Cin=0, F=011110111110110, Cout=0x1lx0xxxx0110, Set=xxxxxxxxxxxxxx00, Sub=0, Opl=0, Op2=1, Less=0
4 A=0001100100100110, B=0110010011010110, Cin=0, F=011110111110110, Cout=0x1lx0xxxx0110, Set=x0xxlxxxx11100, Sub=0, Opl=0, Op2=1, Less=0
5 A=0001100100100110, B=0110010011010110, Cin=0, F=011110111110110, Cout=0x1lx0xxxx00110, Set=x0xxlxxxx11100, Sub=0, Opl=0, Op2=1, Less=0
6 A=0001100100100110, B=0110010011010110, Cin=0, F=011110111110110, Cout=0x1lx0xxxx00110, Set=x0xxlxxxx11100, Sub=0, Opl=0, Op2=1, Less=0
7 A=0001100100100110, B=0110010011010110, Cin=0, F=011110111110110, Cout=011100xxxx00110, Set=x0xxlxxxx11100, Sub=0, Opl=0, Op2=1, Less=0
8 A=0001100100100110, B=0110010011010110, Cin=0, F=011110111110110, Cout=011100xxxx00110, Set=10001lx11111100, Sub=0, Opl=0, Op2=1, Less=0
9 A=0001100100100110, B=0110010011010110, Cin=0, F=011110111110110, Cout=011100xxxx00110, Set=10001lx11111100, Sub=0, Opl=0, Op2=1, Less=0
10 A=0001100100100110, B=0110010011010110, Cin=0, F=011110111110110, Cout=011100xxxx00110, Set=10001lx11111100, Sub=0, Opl=0, Op2=1, Less=0
11 A=0001100100100110, B=0110010011010110, Cin=0, F=011110111110110, Cout=011100xxxx00110, Set=10001lx11111100, Sub=0, Opl=0, Op2=1, Less=0
12 A=0001100100100110, B=0110010011010110, Cin=0, F=011110111110110, Cout=011100xxxx00110, Set=10001lx11111100, Sub=0, Opl=0, Op2=1, Less=0
13 A=0001100100100110, B=0110010011010110, Cin=0, F=011110111110110, Cout=011100000000110, Set=10001lx11111100, Sub=0, Opl=0, Op2=1, Less=0
14 A=0001100100100110, B=0110010011010110, Cin=0, F=011110111110110, Cout=011100000000110, Set=10001lx11111100, Sub=0, Opl=0, Op2=1, Less=0
timberlake ~/Documents) >
```

## Negative Inputs:

```

timberlake {~/Documents} > ./a.out
0 A=x, B=x, Cin=x, F=xxxxxxxxxxxxxx, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=x, Op1=x, Op2=x, Less=x
0 A=65486, B=65516, Cin=0, F=xxxxxxxxxxxxxx, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Op1=0, Op2=1, Less=0
0 A=65486, B=65516, Cin=0, F=xxxxxxxxxxxxxx, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Op1=0, Op2=1, Less=0
1 A=65486, B=65516, Cin=0, F=xxxxxxxxxxxxxx, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Op1=0, Op2=1, Less=0
2 A=65486, B=65516, Cin=0, F=xxxxxxxxxxxxxx, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Op1=0, Op2=1, Less=0
3 A=65486, B=65516, Cin=0, F=xxxxxxxxxxxxxx, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Op1=0, Op2=1, Less=0
4 A=65486, B=65516, Cin=0, F=xxxxxxxxxxxxxx, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Op1=0, Op2=1, Less=0
5 A=65486, B=65516, Cin=0, F=xxxxxxxxxxxxxx, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Op1=0, Op2=1, Less=0
6 A=65486, B=65516, Cin=0, F=xxxxxxxxxxxxxx, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Op1=0, Op2=1, Less=0
timberlake {~/Documents} >

```

The tests above display that, for the inputs;

Positives: A = 0, B = 1  
A = 1, B = 1  
A = 637, B = 120  
A = 3,555, B = 1,245  
A = 10,545, B = 5,600

Negatives: A = 50, B = -20  
A = -50, B = -20

Neither of the two input types output the desired results for the OR operation due to the lack of a delay between execution of instructions. Although 5 or 2 inputs were put into a test bench, the output always appears to concatenate into a single, incorrect result.

Add:

Positive Inputs:

```

timberlake {~/Documents} > ./a.out
0 A=    x, B=    x, Cin=x, F=    z, Cout=zzzzzzzzzzzzzzzz, Set=zzzzzzzzzzzzzzzz, Sub=x, Opl=x, Op2=x, Less=x
0 A=21982, B= 4750, Cin=0, F=    z, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Opl=1, Op2=0, Less=0
0 A=21982, B= 4750, Cin=0, F=    z, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Opl=1, Op2=0, Less=0
1 A=21982, B= 4750, Cin=0, F=    x, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Opl=1, Op2=0, Less=0
2 A=21982, B= 4750, Cin=0, F=    x, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Opl=1, Op2=0, Less=0
3 A=21982, B= 4750, Cin=0, F=    x, Cout=0x010xxx1x0x1110, Set=xxxxxxxxxxxxxx, Sub=0, Opl=1, Op2=0, Less=0
4 A=21982, B= 4750, Cin=0, F=    x, Cout=0x010xxx1x0x1110, Set=xxxxxxxxxxxxxx, Sub=0, Opl=1, Op2=0, Less=0
5 A=21982, B= 4750, Cin=0, F=    x, Cout=00010xx1l0011110, Set=xxxxxxxxxxxxxx, Sub=0, Opl=1, Op2=0, Less=0
6 A=21982, B= 4750, Cin=0, F=    x, Cout=00010xx1l0011110, Set=xxxxxxxxxxxxxx, Sub=0, Opl=1, Op2=0, Less=0
7 A=21982, B= 4750, Cin=0, F=    x, Cout=00010x1110011110, Set=0110wx001101100, Sub=0, Opl=1, Op2=0, Less=0
8 A=21982, B= 4750, Cin=0, F=    x, Cout=00010x1110011110, Set=0110wx00001101100, Sub=0, Opl=1, Op2=0, Less=0
9 A=21982, B= 4750, Cin=0, F=    x, Cout=0001011110011110, Set=0110wx00001101100, Sub=0, Opl=1, Op2=0, Less=0
10 A=21982, B= 4750, Cin=0, F=    x, Cout=0001011110011110, Set=0110100001101100, Sub=0, Opl=1, Op2=0, Less=0
12 A=21982, B= 4750, Cin=0, F=26732, Cout=0001011110011110, Set=0110100001101100, Sub=0, Opl=1, Op2=0, Less=0
timberlake {~/Documents} >

```

Negative Inputs:

```

timberlake {~/Documents} > iverilog project2.v
timberlake {~/Documents} > ./a.out
0 A=    x, B=    x, Cin=x, F=zzzzzzzzzzzzzzzz, Cout=zzzzzzzzzzzzzzzz, Set=zzzzzzzzzzzzzzzz, Sub=x, Opl=x, Op2=x, Less=x
0 A=65486, B=65516, Cin=0, F=    z, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Opl=1, Op2=0, Less=0
0 A=65486, B=65516, Cin=0, F=    z, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Opl=1, Op2=0, Less=0
1 A=65486, B=65516, Cin=0, F=    x, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Opl=1, Op2=0, Less=0
2 A=65486, B=65516, Cin=0, F=    x, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Opl=1, Op2=0, Less=0
3 A=65486, B=65516, Cin=0, F=    x, Cout=111111111x011x0, Set=xxxxxxxxxxxxxx, Sub=0, Opl=1, Op2=0, Less=0
4 A=65486, B=65516, Cin=0, F=    x, Cout=1111111111x01100, Set=111111111x111x10, Sub=0, Opl=1, Op2=0, Less=0
5 A=65486, B=65516, Cin=0, F=    x, Cout=11111111111001100, Set=111111111x111010, Sub=0, Opl=1, Op2=0, Less=0
6 A=65486, B=65516, Cin=0, F=    x, Cout=11111111111001100, Set=1111111110111010, Sub=0, Opl=1, Op2=0, Less=0
7 A=65486, B=65516, Cin=0, F=    x, Cout=11111111111001100, Set=1111111110111010, Sub=0, Opl=1, Op2=0, Less=0
8 A=65486, B=65516, Cin=0, F=65466, Cout=11111111111001100, Sub=0, Opl=1, Op2=0, Less=0
timberlake {~/Documents} >

```

The tests above display that, for the inputs;

**Positives: A = 0, B = 1**

A = 1, B = 1

A = 637, B = 120

A = 3,555, B = 1,245

A = 10,545, B = 5,600

**Negatives: A = 50, B = -20**

A = -50, B = -20

Neither of the two input types output the desired results for the ADD operation due to the lack of a delay between execution of instructions. Although 5 or 2 inputs were put into a test bench, the output always appears to concatenate into a single, incorrect result.

### Subtract:

**Positives:**

```
timberlake ~/Documents > ./a.out
0 A=      x, B=      x, Cin=x, F=      z, Cout=zzzzzzzzzzzzzzzz, Set=zzzzzzzzzzzzzzzz, Sub=x, Op1=x, Op2=x, Less=x
0 A=21982, B= 4750, Cin=0, F=      z, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=1, Op1=1, Op2=0, Less=0
0 A=21982, B= 4750, Cin=0, F=      z, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=1, Op1=1, Op2=0, Less=0
1 A=21982, B= 4750, Cin=0, F=      x, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=1, Op1=1, Op2=0, Less=0
3 A=21982, B= 4750, Cin=0, F=      x, Cout=xlxxx10lxlxlxx1, Set=xxxxxxxxxxxxxx, Sub=1, Op1=1, Op2=0, Less=0
4 A=21982, B= 4750, Cin=0, F=      x, Cout=xlxxx10lxlxlxx1, Set=0xxx001x0x0xxx0, Sub=1, Op1=1, Op2=0, Less=0
5 A=21982, B= 4750, Cin=0, F=      X, Cout=llxx1l0l1l1l1lx11, Set=0xx001lx0x0xxx0, Sub=1, Op1=1, Op2=0, Less=0
6 A=21982, B= 4750, Cin=0, F=      X, Cout=llxx1l0l1l1l1lx11, Set=0xx0001l010x0x00, Sub=1, Op1=1, Op2=0, Less=0
7 A=21982, B= 4750, Cin=0, F=      X, Cout=llxl1l0l1l1l1lx11, Set=0x00001l010x0x00, Sub=1, Op1=1, Op2=0, Less=0
8 A=21982, B= 4750, Cin=0, F=      X, Cout=llxl1l0l1l1l1lx11, Set=0x00001l010x0000, Sub=1, Op1=1, Op2=0, Less=0
9 A=21982, B= 4750, Cin=0, F=      X, Cout=llll1l0l1l1l1l1l1, Set=0x00001l010x0000, Sub=1, Op1=1, Op2=0, Less=0
10 A=21982, B= 4750, Cin=0, F=      X, Cout=llll1l0l1l1l1l1l1, Set=0100001l01010000, Sub=1, Op1=1, Op2=0, Less=0
12 A=21982, B= 4750, Cin=0, F=17232, Cout=llll1l0l1l1l1l1l1, Set=0100001l01010000, Sub=1, Op1=1, Op2=0, Less=0
timberlake ~/Documents >
```

## Negatives:

```

timberlake {~/Documents} > iverilog project2.v
timberlake {~/Documents} > ./a.out
      0 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxxxxxx, Set=xxxxxxxxxxxx, Sub=x, Op1=x, Op2=x, Less=x
      0 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxxxxxx, Set=xxxxxxxxxxxx, Sub=1, Op1=1, Op2=0, Less=0
      1 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxxxxxx, Set=xxxxxxxxxxxx, Sub=1, Op1=1, Op2=0, Less=0
      2 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxxxxxx, Set=xxxxxxxxxxxx, Sub=1, Op1=1, Op2=0, Less=0
      3 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxxxxxx0xxx1, Set=xxxxxxxxxxxxxx, Sub=1, Op1=1, Op2=0, Less=0
      4 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxxxxxx0xxx1, Set=xxxxxxxxx1xxx010, Sub=1, Op1=1, Op2=0, Less=0
      5 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxxxxxx0xxx1, Set=xxxxxxxxx1xxx010, Sub=1, Op1=1, Op2=0, Less=0
      6 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxxxxxx0xxx1, Set=xxxxxxxxx1xxx010, Sub=1, Op1=1, Op2=0, Less=0
      7 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxxxxxx0xxx1, Set=xxxxxxxxx1xxx010, Sub=1, Op1=1, Op2=0, Less=0
      8 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxx000xxx11, Set=xxxxxxxx111x0010, Sub=1, Op1=1, Op2=0, Less=0
      9 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxx000xxx11, Set=xxxxxxxx111x0010, Sub=1, Op1=1, Op2=0, Less=0
     10 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxx00001111, Set=xxxxxxxx111x0010, Sub=1, Op1=1, Op2=0, Less=0
     11 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxx00001111, Set=xxxxxxxx1111111100010, Sub=1, Op1=1, Op2=0, Less=0
     12 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxx00001111, Set=xxxxxxxx1111111100010, Sub=1, Op1=1, Op2=0, Less=0
     13 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxx00001111, Set=xxxxxxxx1111111100010, Sub=1, Op1=1, Op2=0, Less=0
     14 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxx00001111, Set=xxxxxxxx1111111100010, Sub=1, Op1=1, Op2=0, Less=0
     15 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxx00001111, Set=xxxxxxxx1111111100010, Sub=1, Op1=1, Op2=0, Less=0
     16 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxx00001111, Set=xxxxxxxx1111111100010, Sub=1, Op1=1, Op2=0, Less=0
     17 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxx00001111, Set=xxxxxxxx1111111100010, Sub=1, Op1=1, Op2=0, Less=0
     18 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxx00001111, Set=xxxxxxxx1111111100010, Sub=1, Op1=1, Op2=0, Less=0
     19 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxx00001111, Set=xxxxxxxx1111111100010, Sub=1, Op1=1, Op2=0, Less=0
     20 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxx00001111, Set=xxxxxxxx1111111100010, Sub=1, Op1=1, Op2=0, Less=0
     21 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxx00001111, Set=xxxxxxxx1111111100010, Sub=1, Op1=1, Op2=0, Less=0
     22 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxx00001111, Set=xxxxxxxx1111111100010, Sub=1, Op1=1, Op2=0, Less=0
     23 A=65486, B=65516, Cin=0, F= z, Cout=xxxxxxxx00001111, Set=xxxxxxxx1111111100010, Sub=1, Op1=1, Op2=0, Less=0
     24 A=65486, B=65516, Cin=0, F=65506, Cout=00000000001111, Set=11111111111100010, Sub=1, Op1=1, Op2=0, Less=0
timberlake {~/Documents} >

```

The tests above display that, for the inputs;

**Positives:  $A = 0, B = 1$**

$A = 1, B = 1$

$A = 637, B = 120$

$A = 3,555, B = 1,245$

$A = 10,545, B = 5,600$

**Negatives:  $A = 50, B = -20$**

$A = -50, B = -20$

Neither of the two input types output the desired results for the SUB operation due to the lack of a delay between execution of instructions. Although 5 or 2 inputs were put into a test bench, the output always appears to concatenate into a single, incorrect result.

## Set Less Than:

## Positive Inputs:

```
timberlake (~/Documents) > ./a.out
      0 A=    x, B=    x, Cin=x, F=    z, Cout=zzzzzzzzzzzzzzz, Set=zzzzzzzzzzzzzzz, Sub=x, Opl=x, Op2=x, Less=x
      0 A=21982, B= 4750, Cin=0, F=    z, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=1, Opl=1, Op2=1, Less=0
      0 A=21982, B= 4750, Cin=0, F=    z, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=1, Opl=1, Op2=1, Less=0
      1 A=21982, B= 4750, Cin=0, F=    X, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=1, Opl=1, Op2=1, Less=0
      2 A=21982, B= 4750, Cin=0, F=    x, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=1, Opl=1, Op2=1, Less=0
      3 A=21982, B= 4750, Cin=0, F=    X, Cout=xxxxx10lx1k1xxx1, Set=xxxxxxxxxxxxxx, Sub=1, Opl=1, Op2=1, Less=0
      4 A=21982, B= 4750, Cin=0, F=    X, Cout=xxxxx10lx1k1xxx1, Set=xxxxxxxxxxxxxx, Sub=1, Opl=1, Op2=1, Less=0
      5 A=21982, B= 4750, Cin=0, F=    X, Cout=1lx11011111111111, Set=0xxx001lx0x0xxx0, Sub=1, Opl=1, Op2=1, Less=0
      6 A=21982, B= 4750, Cin=0, F=    0, Cout=1lx11011111111111, Set=0x00011010x0xxx0, Sub=1, Opl=1, Op2=1, Less=0
      7 A=21982, B= 4750, Cin=0, F=    0, Cout=1lx11101111111111, Set=0x00011010x0xxx0, Sub=1, Opl=1, Op2=1, Less=0
      8 A=21982, B= 4750, Cin=0, F=    0, Cout=1lx11101111111111, Set=0x0000011010x0xxx0, Sub=1, Opl=1, Op2=1, Less=0
      9 A=21982, B= 4750, Cin=0, F=    0, Cout=1111101111111111, Set=0x0000011010x0xxx0, Sub=1, Opl=1, Op2=1, Less=0
     10 A=21982, B= 4750, Cin=0, F=    0, Cout=1111101111111111, Set=01000001101010000, Sub=1, Opl=1, Op2=1, Less=0

timberlake (~/Documents) > 
```

## Negative Inputs:

```
timberlake ~/Documents > iverilog project2.v
timberlake ~/Documents > ./a.out
   0 A=      x, B=      x, Cin=x, F=      z, Cout=zxxxxxxxxxxxxx, Set=zxxxxxxxxxxxxx, Sub=x, Opl=x, Op2=x, Less=x
   0 A=65486, B=65516, Cin=0, F=      z, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=1, Opl=1, Op2=1, Less=0
   0 A=65486, B=65516, Cin=0, F=      z, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=1, Opl=1, Op2=1, Less=0
   1 A=65486, B=65516, Cin=0, F=      x, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=1, Opl=1, Op2=1, Less=0
   2 A=65486, B=65516, Cin=0, F=      x, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=1, Opl=1, Op2=1, Less=0
   3 A=65486, B=65516, Cin=0, F=      x, Cout=xxxxxxxxxxxx0xxx1, Set=xxxxxxxxxxxx0xxx0, Sub=1, Opl=1, Op2=1, Less=0
   4 A=65486, B=65516, Cin=0, F=      x, Cout=xxxxxxxxxxxx0xx11, Set=xxxxxxxxxx1lx010, Sub=1, Opl=1, Op2=1, Less=0
   5 A=65486, B=65516, Cin=0, F=      x, Cout=xxxxxxxxxxxx0xx111, Set=xxxxxxxxxx1lx010, Sub=1, Opl=1, Op2=1, Less=0
   6 A=65486, B=65516, Cin=0, F=      x, Cout=xxxxxxxxxxxx0xx1111, Set=xxxxxxxxxx1lx0010, Sub=1, Opl=1, Op2=1, Less=0
   7 A=65486, B=65516, Cin=0, F=      x, Cout=xxxxxxxxxxxx00xx111, Set=xxxxxxxxxx1lx0010, Sub=1, Opl=1, Op2=1, Less=0
   8 A=65486, B=65516, Cin=0, F=      x, Cout=xxxxxxxxxxxx00xx1111, Set=xxxxxxxxxx11x00010, Sub=1, Opl=1, Op2=1, Less=0
   9 A=65486, B=65516, Cin=0, F=      x, Cout=xxxxxxxx000011111, Set=xxxxxxxx111x00010, Sub=1, Opl=1, Op2=1, Less=0
  10 A=65486, B=65516, Cin=0, F=      x, Cout=xxxxxxxx000011111, Set=xxxxxxxx111100010, Sub=1, Opl=1, Op2=1, Less=0
  11 A=65486, B=65516, Cin=0, F=      x, Cout=xxxxxxxx000001111, Set=xxxxxxxx1111100010, Sub=1, Opl=1, Op2=1, Less=0
  12 A=65486, B=65516, Cin=0, F=      x, Cout=xxxxxxxx000001111, Set=xxxxxxxx1111100010, Sub=1, Opl=1, Op2=1, Less=0
  13 A=65486, B=65516, Cin=0, F=      x, Cout=xxxxxx0000001111, Set=xxxxxxxx11111100010, Sub=1, Opl=1, Op2=1, Less=0
  14 A=65486, B=65516, Cin=0, F=      x, Cout=xxxxxx0000001111, Set=xxxxxxxx11111100010, Sub=1, Opl=1, Op2=1, Less=0
  15 A=65486, B=65516, Cin=0, F=      x, Cout=xxxxxx0000001111, Set=xxxxxxxx111111100010, Sub=1, Opl=1, Op2=1, Less=0
  16 A=65486, B=65516, Cin=0, F=      x, Cout=xxxxxx0000001111, Set=xxxxxxxx111111100010, Sub=1, Opl=1, Op2=1, Less=0
  17 A=65486, B=65516, Cin=0, F=      x, Cout=xxx0000000011111, Set=xxx11111111100010, Sub=1, Opl=1, Op2=1, Less=0
  18 A=65486, B=65516, Cin=0, F=      x, Cout=xxx0000000011111, Set=xxx11111111100010, Sub=1, Opl=1, Op2=1, Less=0
  19 A=65486, B=65516, Cin=0, F=      x, Cout=x000000000011111, Set=x111111111100010, Sub=1, Opl=1, Op2=1, Less=0
  20 A=65486, B=65516, Cin=0, F=      x, Cout=x000000000011111, Set=x111111111100010, Sub=1, Opl=1, Op2=1, Less=0
  21 A=65486, B=65516, Cin=0, F=      x, Cout=x000000000011111, Set=x111111111100010, Sub=1, Opl=1, Op2=1, Less=0
  22 A=65486, B=65516, Cin=0, F=      x, Cout=x000000000011111, Set=x111111111100010, Sub=1, Opl=1, Op2=1, Less=0
  23 A=65486, B=65516, Cin=0, F=      x, Cout=000000000011111, Set=11111111111100010, Sub=1, Opl=1, Op2=1, Less=0
  24 A=65486, B=65516, Cin=0, F=      1, Cout=000000000011111, Set=11111111111100010, Sub=1, Opl=1, Op2=1, Less=0
timberlake ~/Documents > █
```

The tests above display that, for the inputs;

Positives: A = 0, B = 1  
A = 1, B = 1  
A = 637, B = 120  
A = 3,555, B = 1,245  
A = 10,545, B = 5,600

Negatives: A = 50, B = -20  
A = -50, B = -20

Neither of the two input types output the desired results for the SLT operation due to the lack of a delay between execution of instructions. Although 5 or 2 inputs were put into a test bench, the output always appears to concatenate into a single, incorrect result.

---

*All of these tests show that functionally, this code simply does not work as intended purely because of the lack of time delay between each instruction executed. Even the simplest of inputs for the simplest of operations did not produce the desired results.*

## **5) Analysis of the temporal dependence of the delay on the previous set of inputs:**

I will be analyzing the 7 inputs I utilized for my test bench in various operations to show that my ALU does indeed function as it should for the inputs provided and how, in some cases, it does not produce the desired result. The time delay between all these instructions was set to 100 nanoseconds.

*Example 1) AND: A = 0, B = 1*

The above test bench displays that the AND operation is indeed working properly with inputs A = 0 and B = 1 as the result is 0.

*Example 2) OR: A = 1, B = 1*

The above test bench displays that the OR operation is indeed working properly with inputs A = 1 and B = 1 as the result is 1.

**Example 3) ADD: A = 637, B = 120**

```

timberlake.cse.buffalo.edu - PuTTY
0 A= x, B= x, Cin=x, F= z, Cout=zzzzzzzzzzzzzzz, Set=zzzzzzzzzzzzzzz, Sub=x, Opl=x, Op2=x, Less=x
0 A= x, B= x, Cin=x, F= z, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=x, Opl=x, Op2=x, Less=x
1 A= x, B= x, Cin=x, F= X, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=x, Opl=x, Op2=x, Less=x
100 A= 0, B= 1, Cin=0, F= X, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Opl=1, Op2=0, Less=0
101 A= 0, B= 1, Cin=0, F= x, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Opl=1, Op2=0, Less=0
102 A= 0, B= 1, Cin=0, F= x, Cout=xxxxxxxxxxxxxx0, Set=xxxxxxxxxxxxxx, Sub=0, Opl=1, Op2=0, Less=0
103 A= 0, B= 1, Cin=0, F= x, Cout=0000000000000000, Set=xxxxxxxxxxxxxx0, Sub=0, Opl=1, Op2=0, Less=0
104 A= 0, B= 1, Cin=0, F= x, Cout=0000000000000000, Set=0000000000000001, Sub=0, Opl=1, Op2=0, Less=0
105 A= 0, B= 1, Cin=0, F= X, Cout=0000000000000000, Set=0000000000000001, Sub=0, Opl=1, Op2=0, Less=0
106 A= 0, B= 1, Cin=0, F= 1, Cout=0000000000000000, Set=0000000000000001, Sub=0, Opl=1, Op2=0, Less=0
200 A= 1, B= 1, Cin=0, F= 1, Cout=0000000000000000, Set=0000000000000001, Sub=0, Opl=1, Op2=0, Less=0
202 A= 1, B= 1, Cin=0, F= 1, Cout=0000000000000001, Set=0000000000000000, Sub=0, Opl=1, Op2=0, Less=0
203 A= 1, B= 1, Cin=0, F= 1, Cout=0000000000000001, Set=0000000000000010, Sub=0, Opl=1, Op2=0, Less=0
204 A= 1, B= 1, Cin=0, F= 0, Cout=0000000000000001, Set=0000000000000010, Sub=0, Opl=1, Op2=0, Less=0
300 A= 637, B= 120, Cin=0, F= 2, Cout=0000000000000001, Set=0000000000000010, Sub=0, Opl=1, Op2=0, Less=0
302 A= 637, B= 120, Cin=0, F= 2, Cout=0000001001111110, Sub=0, Opl=1, Op2=0, Less=0
303 A= 637, B= 120, Cin=0, F= 2, Cout=000000001111000, Set=0000000000111, Sub=0, Opl=1, Op2=0, Less=0
304 A= 637, B= 120, Cin=0, F= 638, Cout=000000001111000, Set=000001011110101, Sub=0, Opl=1, Op2=0, Less=0
305 A= 637, B= 120, Cin=0, F= 614, Cout=000000001111000, Set=000001011110101, Sub=0, Opl=1, Op2=0, Less=0
306 A= 637, B= 120, Cin=0, F= 757, Cout=000000001111000, Set=000001011110101, Sub=0, Opl=1, Op2=0, Less=0
402 A= 3555, B= 1245, Cin=0, F= 757, Cout=0000000011110000, Set=000110101010111, Sub=0, Opl=1, Op2=0, Less=0
403 A= 3555, B= 1245, Cin=0, F= 757, Cout=0000010011110001, Set=000100011011110, Sub=0, Opl=1, Op2=0, Less=0
404 A= 3555, B= 1245, Cin=0, F= 3435, Cout=00000010111100001, Set=0000000011011100, Sub=0, Opl=1, Op2=0, Less=0
405 A= 3555, B= 1245, Cin=0, F= 2270, Cout=000011011110001, Set=0000001011111100, Sub=0, Opl=1, Op2=0, Less=0
406 A= 3555, B= 1245, Cin=0, F= 220, Cout=0000110111100001, Set=000100101111000, Sub=0, Opl=1, Op2=0, Less=0
407 A= 3555, B= 1245, Cin=0, F= 764, Cout=000011011100011, Set=000100101111000, Sub=0, Opl=1, Op2=0, Less=0
408 A= 3555, B= 1245, Cin=0, F= 4856, Cout=000011011100011, Set=001001010110000, Sub=0, Opl=1, Op2=0, Less=0
409 A= 3555, B= 1245, Cin=0, F= 4792, Cout=000011011100011, Set=000100101011000, Sub=0, Opl=1, Op2=0, Less=0
410 A= 3555, B= 1245, Cin=0, F= 4784, Cout=000011011100111, Set=0001001010100000, Sub=0, Opl=1, Op2=0, Less=0
411 A= 3555, B= 1245, Cin=0, F= 4784, Cout=000011011101111, Set=0001001010100000, Sub=0, Opl=1, Op2=0, Less=0
412 A= 3555, B= 1245, Cin=0, F= 4768, Cout=000011011101111, Set=0001001010100000, Sub=0, Opl=1, Op2=0, Less=0
413 A= 3555, B= 1245, Cin=0, F= 4768, Cout=000011011111111, Set=0001001010000000, Sub=0, Opl=1, Op2=0, Less=0
414 A= 3555, B= 1245, Cin=0, F= 4736, Cout=000011011111111, Set=0001001011000000, Sub=0, Opl=1, Op2=0, Less=0
416 A= 3555, B= 1245, Cin=0, F= 4800, Cout=000011011111111, Set=0001001011000000, Sub=0, Opl=1, Op2=0, Less=0
500 A=21982, B= 4750, Cin=0, F= 4800, Cout=000101011101110, Set=0100101011111101, Sub=0, Opl=1, Op2=0, Less=0
502 A=21982, B= 4750, Cin=0, F= 4800, Cout=0001001111011110, Set=010101001111011100, Sub=0, Opl=1, Op2=0, Less=0
503 A=21982, B= 4750, Cin=0, F= 4800, Cout=0001001111011110, Set=010101001111011100, Sub=0, Opl=1, Op2=0, Less=0
504 A=21982, B= 4750, Cin=0, F=19197, Cout=0001001111001110, Set=0110000011101100, Sub=0, Opl=1, Op2=0, Less=0
505 A=21982, B= 4750, Cin=0, F=27884, Cout=0001011110011110, Set=011000001101100, Sub=0, Opl=1, Op2=0, Less=0
506 A=21982, B= 4750, Cin=0, F=24812, Cout=0001011110011110, Set=0110100001101100, Sub=0, Opl=1, Op2=0, Less=0
507 A=21982, B= 4750, Cin=0, F=24684, Cout=0001011110011110, Set=0110100001101100, Sub=0, Opl=1, Op2=0, Less=0
508 A=21982, B= 4750, Cin=0, F=26732, Cout=0001011110011110, Set=0110100001101100, Sub=0, Opl=1, Op2=0, Less=0

```

The above test bench displays that the ADD operation is indeed working properly with inputs A = 637 and B = 120 as the result is 757.

*Example 4) SUB: A = 3,555, B = 1,245*

```
timberlake.cse.buffalo.edu - PuTTY
215 A= 1, B= 1, Cin=0, F=65408, Cout=0000000011111111, Set=1111111000000000, Sub=1, Op1=1, Op2=0, Less=0
216 A= 1, B= 1, Cin=0, F=65280, Cout=0000000011111111, Set=1111111000000000, Sub=1, Op1=1, Op2=0, Less=0
217 A= 1, B= 1, Cin=0, F=65280, Cout=0000000011111111, Set=1111111000000000, Sub=1, Op1=1, Op2=0, Less=0
218 A= 1, B= 1, Cin=0, F=65280, Cout=0000000011111111, Set=1111111000000000, Sub=1, Op1=1, Op2=0, Less=0
219 A= 1, B= 1, Cin=0, F=65024, Cout=0000000011111111, Set=1111110000000000, Sub=1, Op1=1, Op2=0, Less=0
220 A= 1, B= 1, Cin=0, F=65024, Cout=0000000011111111, Set=1111110000000000, Sub=1, Op1=1, Op2=0, Less=0
221 A= 1, B= 1, Cin=0, F=64512, Cout=0000000011111111, Set=1111100000000000, Sub=1, Op1=1, Op2=0, Less=0
222 A= 1, B= 1, Cin=0, F=64512, Cout=0000000011111111, Set=1111100000000000, Sub=1, Op1=1, Op2=0, Less=0
223 A= 1, B= 1, Cin=0, F=63488, Cout=0000000011111111, Set=1110000000000000, Sub=1, Op1=1, Op2=0, Less=0
224 A= 1, B= 1, Cin=0, F=63488, Cout=0000111111111111, Set=1110000000000000, Sub=1, Op1=1, Op2=0, Less=0
225 A= 1, B= 1, Cin=0, F=61440, Cout=0000111111111111, Set=1110000000000000, Sub=1, Op1=1, Op2=0, Less=0
226 A= 1, B= 1, Cin=0, F=61440, Cout=0000111111111111, Set=1111110000000000, Sub=1, Op1=1, Op2=0, Less=0
227 A= 1, B= 1, Cin=0, F=57344, Cout=0001111111111111, Set=1100000000000000, Sub=1, Op1=1, Op2=0, Less=0
228 A= 1, B= 1, Cin=0, F=57344, Cout=0011111111111111, Set=1100000000000000, Sub=1, Op1=1, Op2=0, Less=0
229 A= 1, B= 1, Cin=0, F=49152, Cout=0011111111111111, Set=1000000000000000, Sub=1, Op1=1, Op2=0, Less=0
230 A= 1, B= 1, Cin=0, F=49152, Cout=0111111111111111, Set=1000000000000000, Sub=1, Op1=1, Op2=0, Less=0
231 A= 1, B= 1, Cin=0, F=32768, Cout=0111111111111111, Set=0000000000000000, Sub=1, Op1=1, Op2=0, Less=0
232 A= 1, B= 1, Cin=0, F=32768, Cout=1111111111111111, Set=0000000000000000, Sub=1, Op1=1, Op2=0, Less=0
233 A= 1, B= 1, Cin=0, F= 0, Cout=1111111111111111, Set=0000000000000000, Sub=1, Op1=1, Op2=0, Less=0
300 A= 637, B= 120, Cin=0, F= 0, Cout=1111111111111111, Set=0000000000000000, Sub=1, Op1=1, Op2=0, Less=0
302 A= 637, B= 120, Cin=0, F= 0, Cout=1111111111111111, Set=0000001001111100, Sub=1, Op1=1, Op2=0, Less=0
303 A= 637, B= 120, Cin=0, F= 0, Cout=1111111111111111, Set=0000001000000101, Sub=1, Op1=1, Op2=0, Less=0
304 A= 637, B= 120, Cin=0, F= 636, Cout=1111111111111111, Set=0000001000000101, Sub=1, Op1=1, Op2=0, Less=0
400 A= 3555, B= 1245, Cin=0, F= 517, Cout=1111111111111111, Set=0000001000000101, Sub=1, Op1=1, Op2=0, Less=0
402 A= 3555, B= 1245, Cin=0, F= 517, Cout=1111111111111111, Set=0000110110011011, Sub=1, Op1=1, Op2=0, Less=0
403 A= 3555, B= 1245, Cin=0, F= 517, Cout=1111111111111111, Set=0000100100001110, Sub=1, Op1=1, Op2=0, Less=0
404 A= 3555, B= 1245, Cin=0, F= 3483, Cout=1111111111111111, Set=0000100100000110, Sub=1, Op1=1, Op2=0, Less=0
405 A= 3555, B= 1245, Cin=0, F= 2310, Cout=1111111111111111, Set=0000100100000110, Sub=1, Op1=1, Op2=0, Less=0
406 A= 3555, B= 1245, Cin=0, F= 2310, Cout=1111111111111111, Set=0000100100000110, Sub=1, Op1=1, Op2=0, Less=0
502 A=21982, B= 4750, Cin=0, F= 2310, Cout=111111111000110, Set=0101000100111011, Sub=1, Op1=1, Op2=0, Less=0
503 A=21982, B= 4750, Cin=0, F= 2310, Cout=1111110111010111, Set=0100011100100010, Sub=1, Op1=1, Op2=0, Less=0
504 A=21982, B= 4750, Cin=0, F=20795, Cout=1111110111011101, Set=0100001100000000, Sub=1, Op1=1, Op2=0, Less=0
505 A=21982, B= 4750, Cin=0, F=18210, Cout=1111110111111111, Set=0100001100010100, Sub=1, Op1=1, Op2=0, Less=0
506 A=21982, B= 4750, Cin=0, F=17152, Cout=1111110111111101, Set=0100001101010000, Sub=1, Op1=1, Op2=0, Less=0
507 A=21982, B= 4750, Cin=0, F=17172, Cout=1111110111111111, Set=0100001101011000, Sub=1, Op1=1, Op2=0, Less=0
508 A=21982, B= 4750, Cin=0, F=17232, Cout=1111110111111011, Set=0100001101010000, Sub=1, Op1=1, Op2=0, Less=0
509 A=21982, B= 4750, Cin=0, F=17240, Cout=1111110111111111, Set=0100001101000000, Sub=1, Op1=1, Op2=0, Less=0
510 A=21982, B= 4750, Cin=0, F=17232, Cout=1111110111111111, Set=0100001101010000, Sub=1, Op1=1, Op2=0, Less=0
511 A=21982, B= 4750, Cin=0, F=17216, Cout=1111110111111111, Set=0100001101010000, Sub=1, Op1=1, Op2=0, Less=0
512 A=21982, B= 4750, Cin=0, F=17232, Cout=1111110111111111, Set=0100001101010000, Sub=1, Op1=1, Op2=0, Less=0
600 A=21982, B= 4750, Cin=0, F=17232, Cout=1111110111111111, Set=0100001101010000, Sub=1, Op1=1, Op2=0, Less=0
```

The above test bench displays that the SUB operation is indeed working properly with inputs A = 3,555 and B = 1,245 as the result is 2310.

**Example 5) SLT: A = 21,982, B = 4,750**

```

timberlake.cse.buffalo.edu - PuTTY
208 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
209 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
210 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
211 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
212 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
213 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
214 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
215 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
216 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
217 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
218 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
219 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
220 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
221 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
222 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
223 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
224 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
225 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
226 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
227 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
228 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
229 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
230 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
231 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
232 A= 1, B= 1, Cin=0, F= 1, Cout=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
233 A= 1, B= 1, Cin=0, F= 0, Cout=11111111111111111111111111111111, Set=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
300 A= 637, B= 120, Cin=0, F= 0, Cout=11111111111111111111111111111111, Set=00000000000000000000000000000000, Sub=1, Op1=1, Op2=1, Less=0
302 A= 637, B= 120, Cin=0, F= 0, Cout=11111111111111111111111111111111, Set=00000001001111100, Sub=1, Op1=1, Op2=1, Less=0
303 A= 637, B= 120, Cin=0, F= 0, Cout=11111111111111111111111111111111, Set=000000010000000101, Sub=1, Op1=1, Op2=1, Less=0
400 A= 3555, B= 1245, Cin=0, F= 0, Cout=11111111111111111111111111111111, Set=000000010000000101, Sub=1, Op1=1, Op2=1, Less=0
402 A= 3555, B= 1245, Cin=0, F= 0, Cout=11111111111111111111111111111111, Set=00000110110011011, Sub=1, Op1=1, Op2=1, Less=0
403 A= 3555, B= 1245, Cin=0, F= 0, Cout=11111111111111111111111111111111, Set=00000100100001110, Sub=1, Op1=1, Op2=1, Less=0
500 A=21982, B= 4750, Cin=0, F= 0, Cout=11111111111111111111111111111111, Set=00001001000000110, Sub=1, Op1=1, Op2=1, Less=0
502 A=21982, B= 4750, Cin=0, F= 0, Cout=11111111111111111111111111111111, Set=0101000100111011, Sub=1, Op1=1, Op2=1, Less=0
503 A=21982, B= 4750, Cin=0, F= 0, Cout=11111111111111111111111111111111, Set=01000011100100010, Sub=1, Op1=1, Op2=1, Less=0
504 A=21982, B= 4750, Cin=0, F= 0, Cout=111111011101101101, Set=0100001100000000, Sub=1, Op1=1, Op2=1, Less=0
505 A=21982, B= 4750, Cin=0, F= 0, Cout=11111101111111111111111111111111, Set=0100001100010100, Sub=1, Op1=1, Op2=1, Less=0
506 A=21982, B= 4750, Cin=0, F= 0, Cout=11111101111111111111111111111111, Set=0100001101010000, Sub=1, Op1=1, Op2=1, Less=0
507 A=21982, B= 4750, Cin=0, F= 0, Cout=11111101111111111111111111111111, Set=0100001101010000, Sub=1, Op1=1, Op2=1, Less=0
508 A=21982, B= 4750, Cin=0, F= 0, Cout=11111101111111111111111111111111, Set=0100001101010000, Sub=1, Op1=1, Op2=1, Less=0
509 A=21982, B= 4750, Cin=0, F= 0, Cout=11111101111111111111111111111111, Set=0100001101000000, Sub=1, Op1=1, Op2=1, Less=0
510 A=21982, B= 4750, Cin=0, F= 0, Cout=11111101111111111111111111111111, Set=0100001101010000, Sub=1, Op1=1, Op2=1, Less=0

```

The above test bench displays that the SLT operation is indeed working properly with inputs A = 21,982 and B = 4,750 as the result is 0.

**Example 6) ADD: A = 50, B = -20**

```

timberlake ~/Documents > iverilog project2.v
timberlake ~/Documents > ./a.out
      0 A=   x, B=   x, Cin=x, F=   z, Cout=zzzzzzzzzzzzzzzz, Set=zzzzzzzzzzzzzzzz, Sub=x, Op1=x, Op2=x, Less=x
      0 A=   x, B=   x, Cin=x, F=   z, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=x, Op1=x, Op2=x, Less=x
      ...
      100 A=  50, B=65516, Cin=0, F=   X, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Op1=1, Op2=0, Less=0
      101 A=  50, B=65516, Cin=0, F=   x, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Op1=1, Op2=0, Less=0
      102 A=  50, B=65516, Cin=0, F=   x, Cout=xxxxxxxxxxxxxx0, Set=xxxxxxxxxxxxxx, Sub=0, Op1=1, Op2=0, Less=0
      103 A=  50, B=65516, Cin=0, F=   x, Cout=xxxxxxxxxx1xxx0, Set=xxxxxxxxxxxxx1, Sub=0, Op1=1, Op2=0, Less=0
      104 A=  50, B=65516, Cin=0, F=   x, Cout=xxxxxxxxxx1xxx00, Set=xxxxxxxxxx0xxx10, Sub=0, Op1=1, Op2=0, Less=0
      105 A=  50, B=65516, Cin=0, F=   X, Cout=xxxxxxxxxx1xxx00, Set=xxxxxxxxxx0xxx10, Sub=0, Op1=1, Op2=0, Less=0
      106 A=  50, B=65516, Cin=0, F=   X, Cout=xxxxxxxxxx1xxx00, Set=xxxxxxxxxx0xxx10, Sub=0, Op1=1, Op2=0, Less=0
      107 A=  50, B=65516, Cin=0, F=   X, Cout=xxxxxxxxxx11xx000, Set=xxxxxxxx0xx1110, Sub=0, Op1=1, Op2=0, Less=0
      108 A=  50, B=65516, Cin=0, F=   X, Cout=xxxxxxxx0xx1110, Sub=0, Op1=1, Op2=0, Less=0
      109 A=  50, B=65516, Cin=0, F=   X, Cout=xxxxxxxx1111x000, Set=xxxxxxxx000x11110, Sub=0, Op1=1, Op2=0, Less=0
      110 A=  50, B=65516, Cin=0, F=   X, Cout=xxxxxxxx111100000, Set=xxxxxxxx0000x11110, Sub=0, Op1=1, Op2=0, Less=0
      111 A=  50, B=65516, Cin=0, F=   X, Cout=xxxxxxxx111100000, Set=xxxxxxxx0000011110, Sub=0, Op1=1, Op2=0, Less=0
      112 A=  50, B=65516, Cin=0, F=   X, Cout=xxxxxxxx111100000, Set=xxxxxxxx0000011110, Sub=0, Op1=1, Op2=0, Less=0
      113 A=  50, B=65516, Cin=0, F=   X, Cout=xxxx11111100000, Set=xxxxx00000011110, Sub=0, Op1=1, Op2=0, Less=0
      114 A=  50, B=65516, Cin=0, F=   X, Cout=xxxx11111100000, Set=xxxx00000011110, Sub=0, Op1=1, Op2=0, Less=0
      115 A=  50, B=65516, Cin=0, F=   X, Cout=xxxx111111100000, Set=xxxx000000011110, Sub=0, Op1=1, Op2=0, Less=0
      116 A=  50, B=65516, Cin=0, F=   X, Cout=xxxx111111100000, Set=xxx000000011110, Sub=0, Op1=1, Op2=0, Less=0
      117 A=  50, B=65516, Cin=0, F=   X, Cout=xxx1111111100000, Set=xxx0000000011110, Sub=0, Op1=1, Op2=0, Less=0
      118 A=  50, B=65516, Cin=0, F=   X, Cout=xxx1111111100000, Set=xxx0000000011110, Sub=0, Op1=1, Op2=0, Less=0
      119 A=  50, B=65516, Cin=0, F=   X, Cout=xx11111111100000, Set=xx0000000011110, Sub=0, Op1=1, Op2=0, Less=0
      120 A=  50, B=65516, Cin=0, F=   X, Cout=xx11111111100000, Set=x00000000011110, Sub=0, Op1=1, Op2=0, Less=0
      121 A=  50, B=65516, Cin=0, F=   X, Cout=x111111111100000, Set=x000000000011110, Sub=0, Op1=1, Op2=0, Less=0
      122 A=  50, B=65516, Cin=0, F=   X, Cout=x1111111111100000, Set=0000000000011110, Sub=0, Op1=1, Op2=0, Less=0
      123 A=  50, B=65516, Cin=0, F=   X, Cout=1111111111100000, Set=0000000000011110, Sub=0, Op1=1, Op2=0, Less=0
      124 A=  50, B=65516, Cin=0, F=   30, Cout=1111111111100000, Set=0000000000011110, Sub=0, Op1=1, Op2=0, Less=0
      ...
      202 A=65486, B=65516, Cin=0, F=   30, Cout=1111111111001100, Set=111111111100010, Sub=0, Op1=1, Op2=0, Less=0
      203 A=65486, B=65516, Cin=0, F=   30, Cout=1111111111001100, Set=1111111110111010, Sub=0, Op1=1, Op2=0, Less=0
      204 A=65486, B=65516, Cin=0, F=65506, Cout=1111111111001100, Set=1111111110111010, Sub=0, Op1=1, Op2=0, Less=0
      205 A=65486, B=65516, Cin=0, F=65466, Cout=1111111111001100, Set=1111111110111010, Sub=0, Op1=1, Op2=0, Less=0
      300 A=65486, B=65516, Cin=0, F=65466, Cout=1111111111001100, Set=1111111110111010, Sub=0, Op1=1, Op2=0, Less=0

```

The above test bench displays that the ADD operation with a single negative input is indeed working properly with inputs A = 50 and B = -20 as the result is 30.

*Example 7) ADD: A = -50, B = -20*

```

timberlake ~/Documents > iverilog project2.v
timberlake ~/Documents > ./a.out
      0 A=  x, B=  x, Cin=x, F= z, Cout=zzzzzzzzzzzzzzzzzzzz, Sub=x, Op1=x, Op2=x, Less=x
      0 A=  x, B=  x, Cin=x, F= z, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=x, Op1=x, Op2=x, Less=x
      1 A=  x, B=  x, Cin=x, F= x, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=x, Op1=x, Op2=x, Less=x
100 A= 50, B=65516, Cin=0, F= x, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Op1=1, Op2=0, Less=0
101 A= 50, B=65516, Cin=0, F= x, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Op1=1, Op2=0, Less=0
102 A= 50, B=65516, Cin=0, F= x, Cout=xxxxxxxxxxxxxx, Set=xxxxxxxxxxxxxx, Sub=0, Op1=1, Op2=0, Less=0
103 A= 50, B=65516, Cin=0, F= x, Cout=xxxxxxxxxx1xxxx0, Set=xxxxxxxxxxxxxx10, Sub=0, Op1=1, Op2=0, Less=0
104 A= 50, B=65516, Cin=0, F= x, Cout=xxxxxxxxxx1xxxx0, Set=xxxxxxxxxxxxxx10, Sub=0, Op1=1, Op2=0, Less=0
105 A= 50, B=65516, Cin=0, F= x, Cout=xxxxxxxxxx1xxxx0, Set=xxxxxxxxxxxxxx0xxx110, Sub=0, Op1=1, Op2=0, Less=0
106 A= 50, B=65516, Cin=0, F= x, Cout=xxxxxxxxxx11xx00, Set=xxxxxxxxxx0xxx110, Sub=0, Op1=1, Op2=0, Less=0
107 A= 50, B=65516, Cin=0, F= x, Cout=xxxxxxxxxx11xx00, Set=xxxxxxxxxx0xxx110, Sub=0, Op1=1, Op2=0, Less=0
108 A= 50, B=65516, Cin=0, F= x, Cout=xxxxxxxxxx11xx000, Set=xxxxxxxxxx00xx1110, Sub=0, Op1=1, Op2=0, Less=0
109 A= 50, B=65516, Cin=0, F= x, Cout=xxxxxxxxxx111x0000, Set=xxxxxxxxxx000x11110, Sub=0, Op1=1, Op2=0, Less=0
110 A= 50, B=65516, Cin=0, F= x, Cout=xxxxxxxxxx11110000, Set=xxxxxxxxxx0000x11110, Sub=0, Op1=1, Op2=0, Less=0
111 A= 50, B=65516, Cin=0, F= x, Cout=xxxxxxxxxx111100000, Set=xxxxxxxxxx00000011110, Sub=0, Op1=1, Op2=0, Less=0
112 A= 50, B=65516, Cin=0, F= x, Cout=xxxxxxxxxx111100000, Set=xxxxxxxx00000011110, Sub=0, Op1=1, Op2=0, Less=0
113 A= 50, B=65516, Cin=0, F= x, Cout=xxxxxxxxxx111100000, Set=xxxxxxxx00000011110, Sub=0, Op1=1, Op2=0, Less=0
114 A= 50, B=65516, Cin=0, F= x, Cout=xxxxxxxx111110000, Set=xxxxxxxx000000011110, Sub=0, Op1=1, Op2=0, Less=0
115 A= 50, B=65516, Cin=0, F= x, Cout=xxxx11111100000, Set=xxxx0000000011110, Sub=0, Op1=1, Op2=0, Less=0
116 A= 50, B=65516, Cin=0, F= x, Cout=xxxx111111100000, Set=xxxx00000000011110, Sub=0, Op1=1, Op2=0, Less=0
117 A= 50, B=65516, Cin=0, F= x, Cout=xxxx1111111100000, Set=xxxx00000000011110, Sub=0, Op1=1, Op2=0, Less=0
118 A= 50, B=65516, Cin=0, F= x, Cout=xxxx11111111100000, Set=xxxx00000000011110, Sub=0, Op1=1, Op2=0, Less=0
119 A= 50, B=65516, Cin=0, F= x, Cout=xxxx111111111100000, Set=xxxx00000000011110, Sub=0, Op1=1, Op2=0, Less=0
120 A= 50, B=65516, Cin=0, F= x, Cout=xxxx1111111111100000, Set=x000000000011110, Sub=0, Op1=1, Op2=0, Less=0
121 A= 50, B=65516, Cin=0, F= x, Cout=xxxx11111111111100000, Set=x0000000000011110, Sub=0, Op1=1, Op2=0, Less=0
122 A= 50, B=65516, Cin=0, F= x, Cout=xxxx111111111111100000, Set=00000000000011110, Sub=0, Op1=1, Op2=0, Less=0
123 A= 50, B=65516, Cin=0, F= x, Cout=xxxx1111111111111100000, Set=000000000000011110, Sub=0, Op1=1, Op2=0, Less=0
      200 A=65486, B=65516, Cin=0, F= 30, Cout=111111111111111100000, Set=000000000011110, Sub=0, Op1=1, Op2=0, Less=0
202 A=65486, B=65516, Cin=0, F= 30, Cout=111111111111111100010, Set=111111111111111100010, Sub=0, Op1=1, Op2=0, Less=0
203 A=65486, B=65516, Cin=0, F= 30, Cout=111111111111111100110, Set=111111111111111100110, Sub=0, Op1=1, Op2=0, Less=0
204 A=65486, B=65516, Cin=0, F=65506, Cout=111111111111111100110, Sub=0, Op1=1, Op2=0, Less=0
205 A=65486, B=65516, Cin=0, F=65466, Cout=111111111111111100110, Sub=0, Op1=1, Op2=0, Less=0
300 A=65486, B=65516, Cin=0, F=65466 Cout=111111111111111100110, Sub=0, Op1=1, Op2=0, Less=0

```

The above test bench displays that the ADD operation with a double negative input is not working as desired though in a way, still producing the correct answer. With inputs  $A = -50$  and  $B = -20$ , the result is 65,466, this is not the desired result that we want which would be -70 in decimal, however, the ALU is actually computing -70, but only in 2's complement format. The user would have to convert the number 65,466 through 2's complement into the desired result of -70.

*The ALU is providing desired outputs to certain operations with certain inputs such as ADD or SUB with only positive numbers however, with double negative numbers it still provides a correct output in a certain way, it is just not the desired result the user wishes to see if one does not wish to pursue any additional work or code in order to convert that number into a truly correct output of the ALU.*

**6) Analysis of the average delay across 1,000 random input patterns:**

The analysis of 1000 random input patterns formed this data of average delay across the operations:

200 input patterns:  
AND = Avg Delay of 8.44311 nanoseconds  
OR = Avg Delay of 8.53535 nanoseconds  
ADD = Avg Delay of 10.66667 nanoseconds  
SUB = Avg Delay of 11.08333 nanoseconds  
SLT = Avg Delay of 10.53123 nanoseconds

---

*The average delay was greatest for the ADD and SUB operations because the critical path is located within the ADD/SUB functionality within the ALU therefore, while executing an ADD or SUB operation, inputs would pass through the most gates in order to achieve an output which leads to the highest average delay between all the operations.*