University at Buffalo

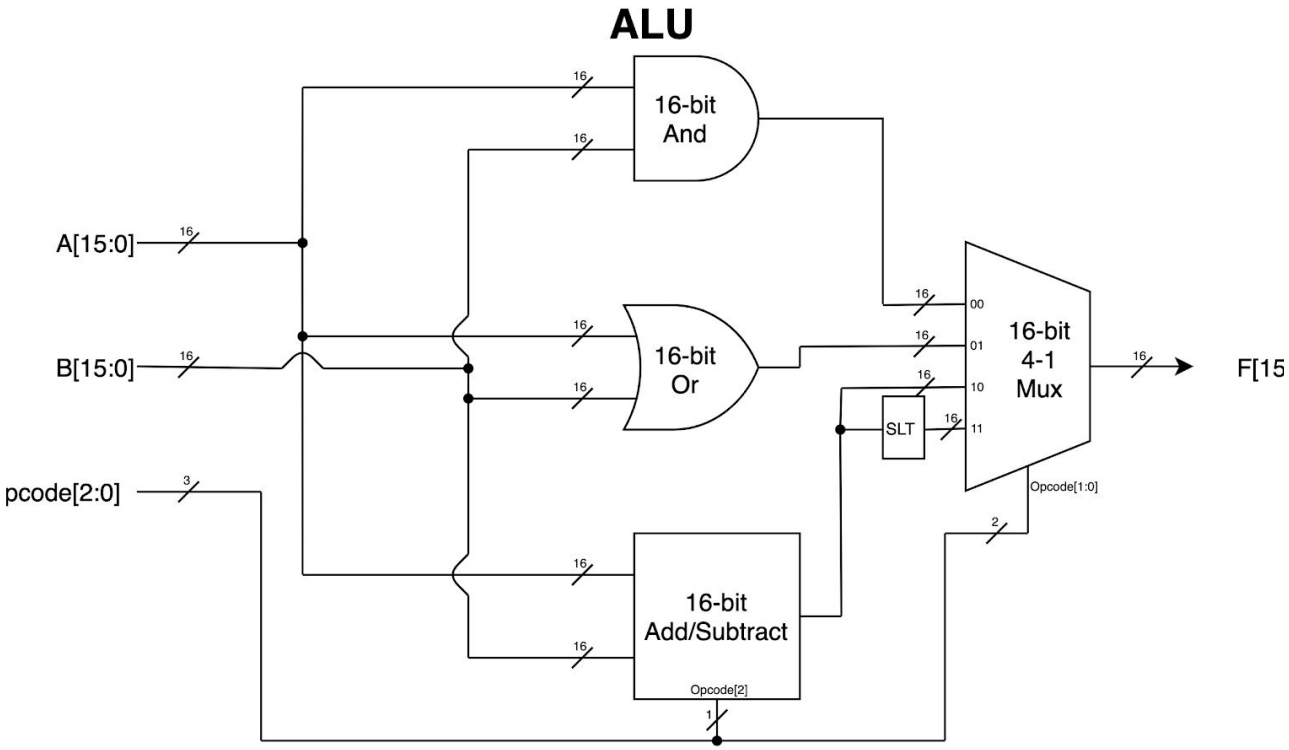
Project 3

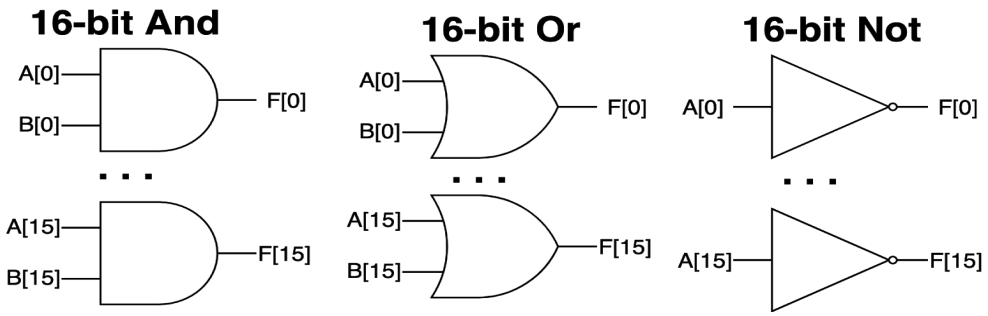
CSE 341

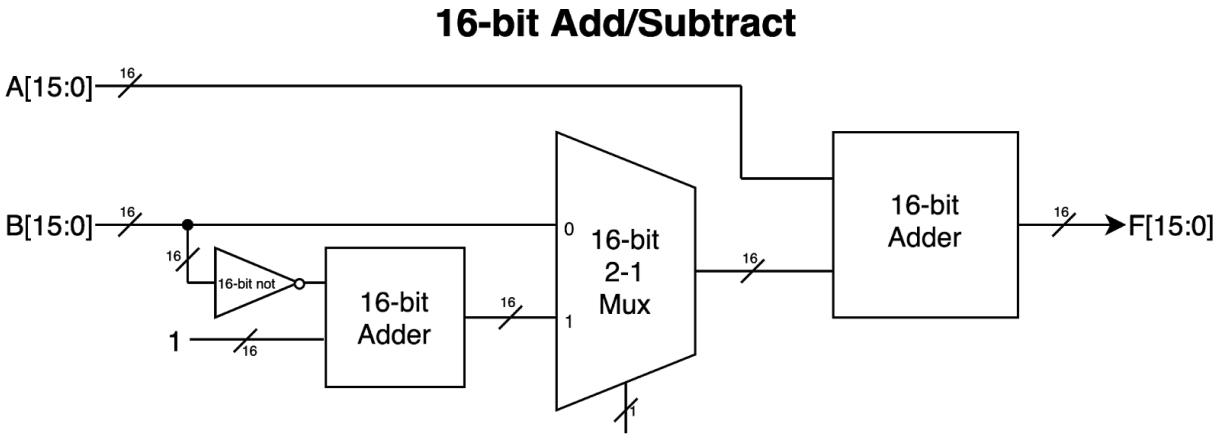
Brendan Meehan
50226243
Dr. Kris Schindler
22 November, 2019

Table of Contents

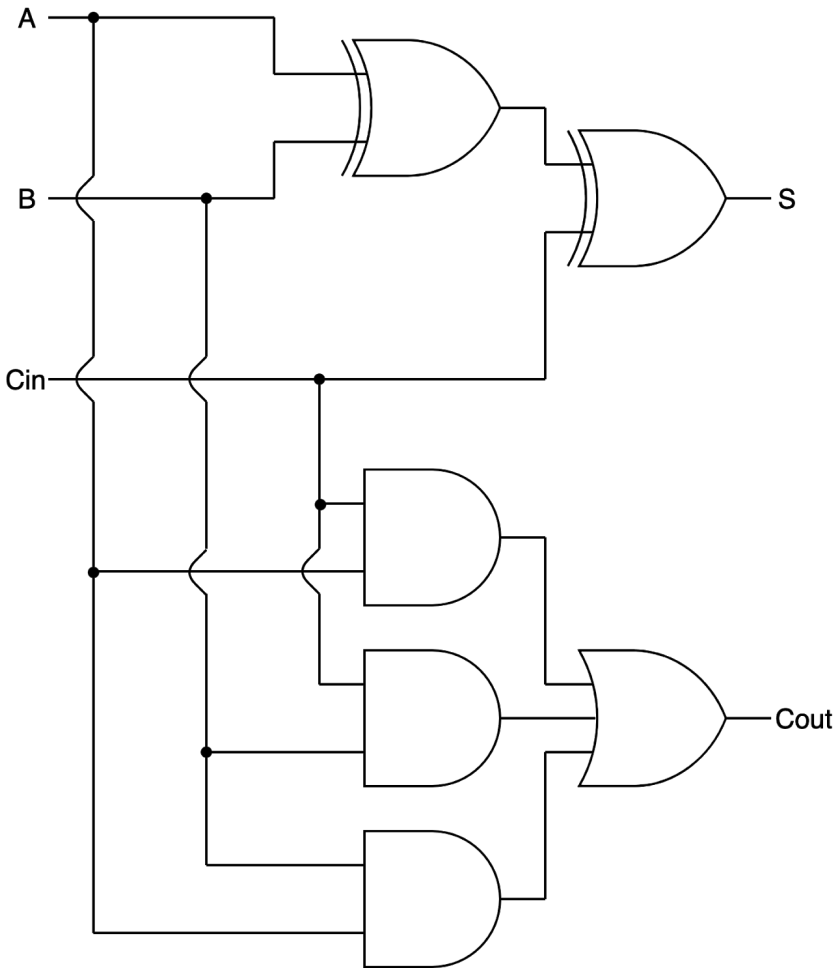
Circuit Diagrams



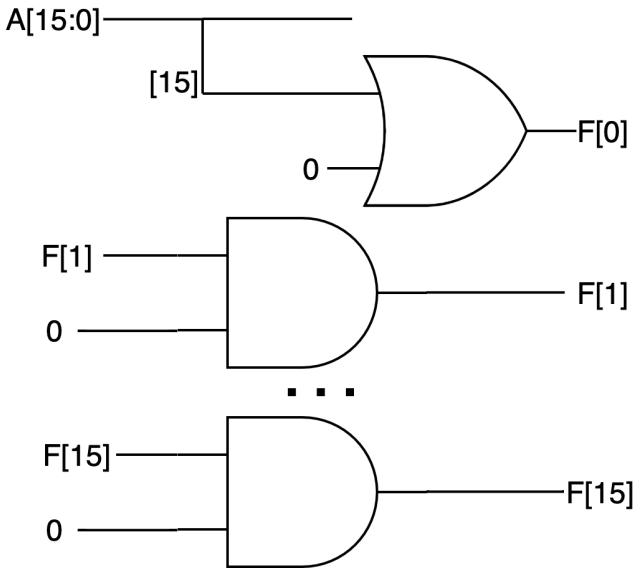




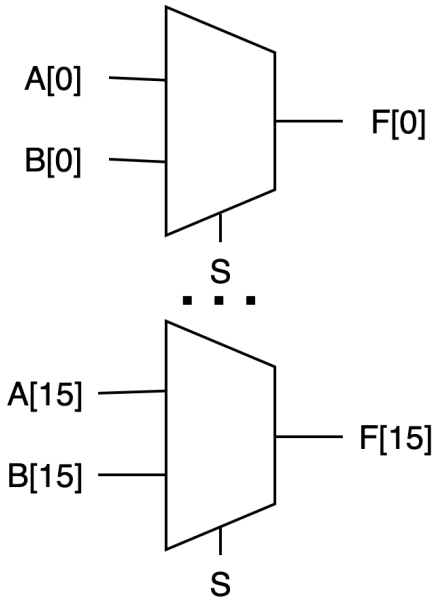
Full Adder



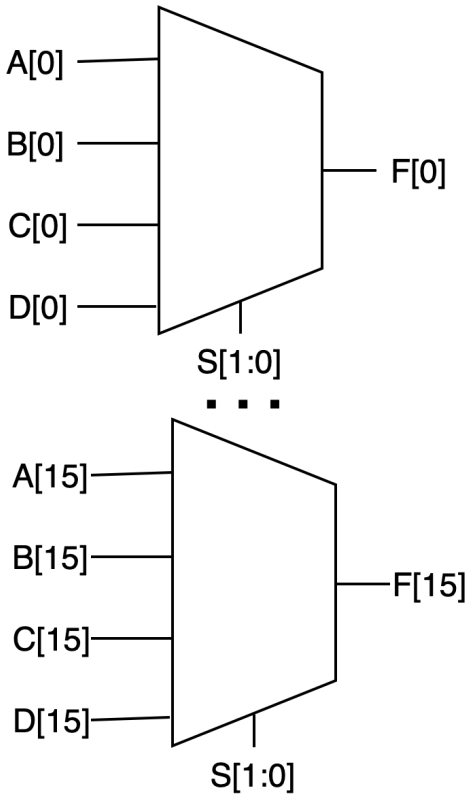
SLT



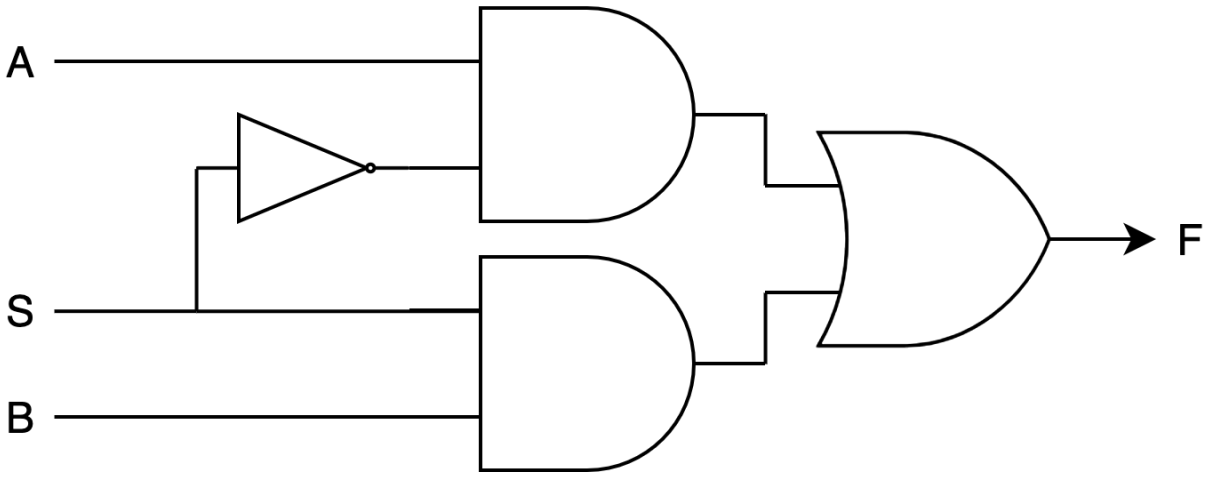
16-bit 2-1 Mux



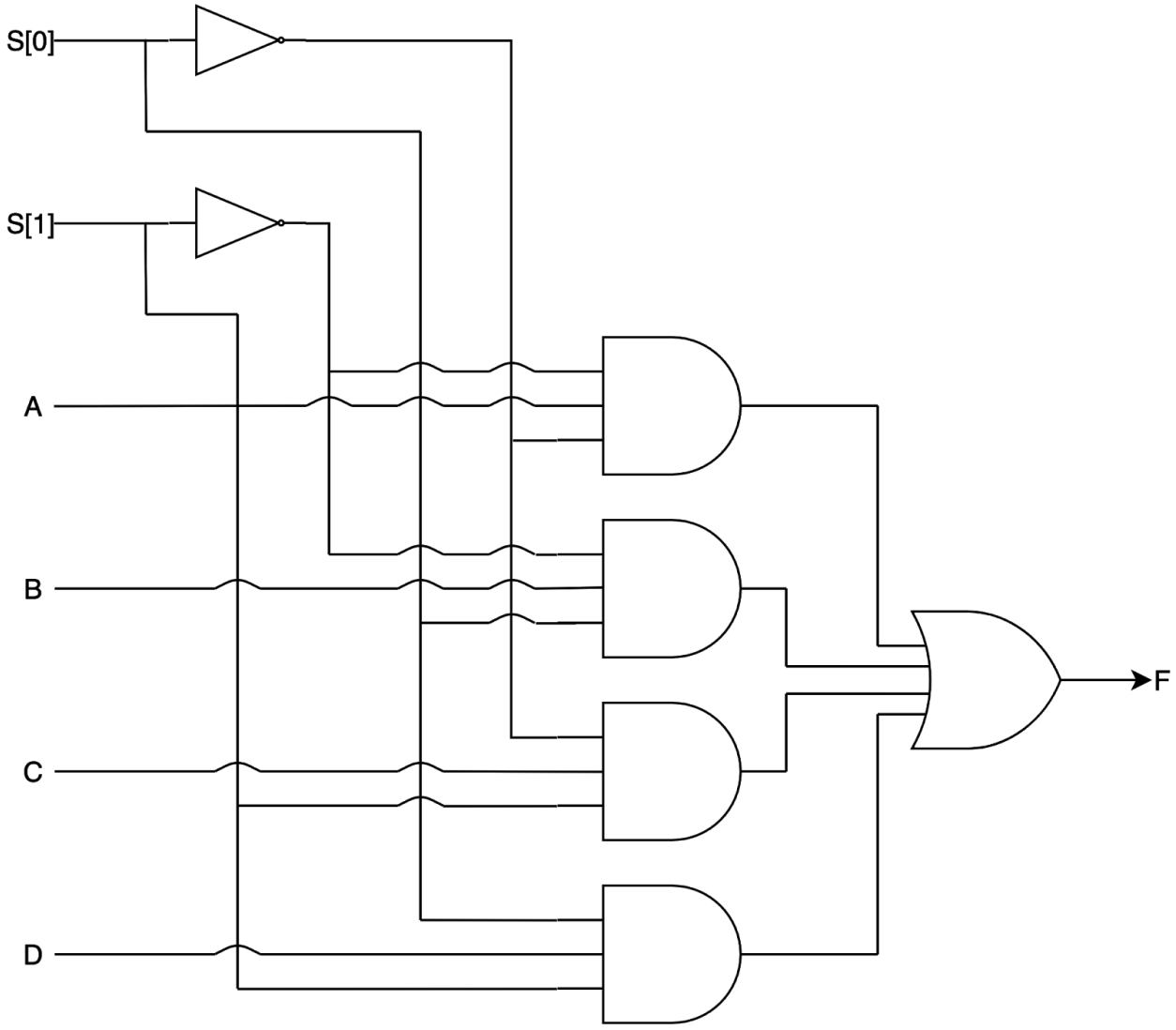
16-bit 4-1 Mux



2-1 Multiplexer

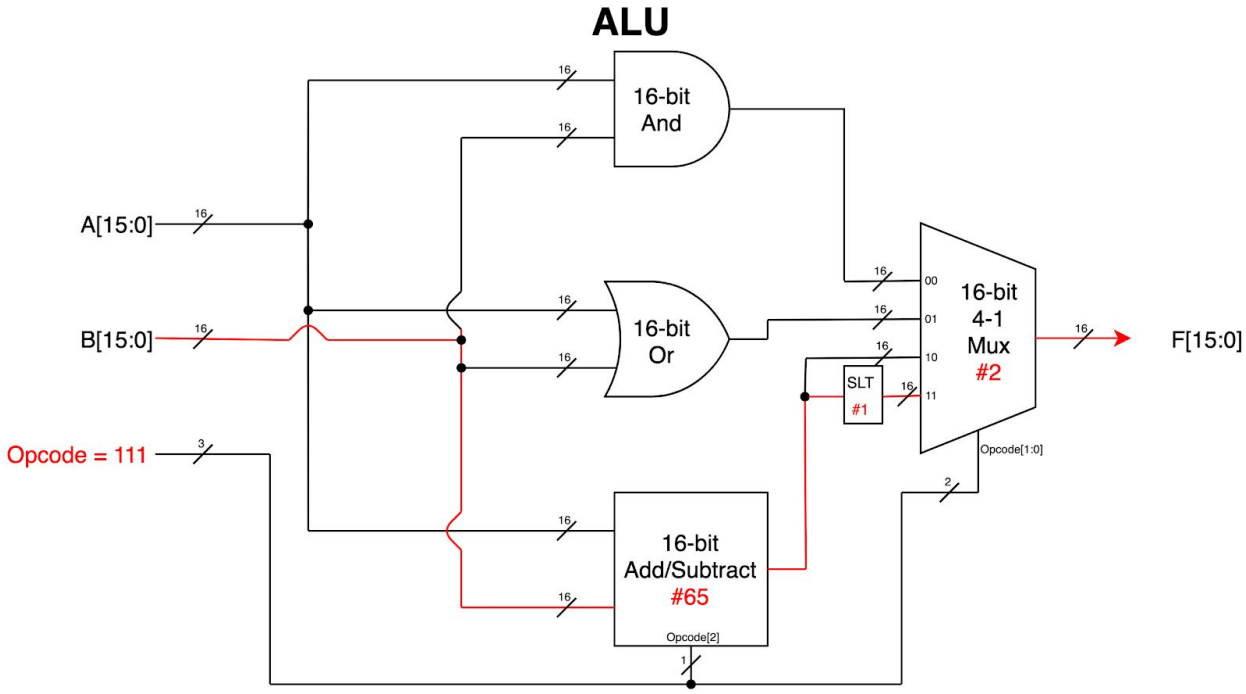


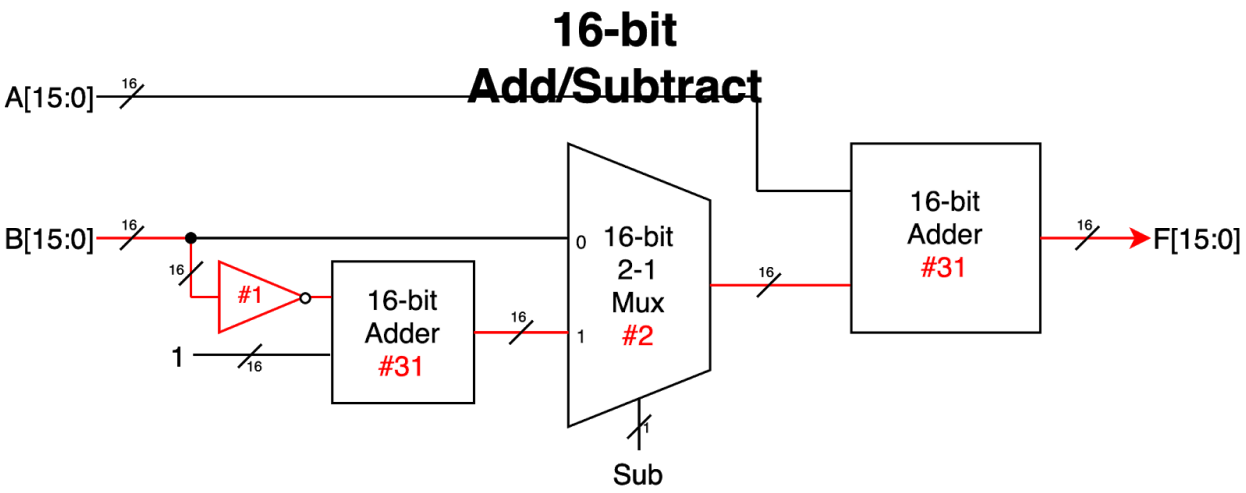
4-1 Multiplexer

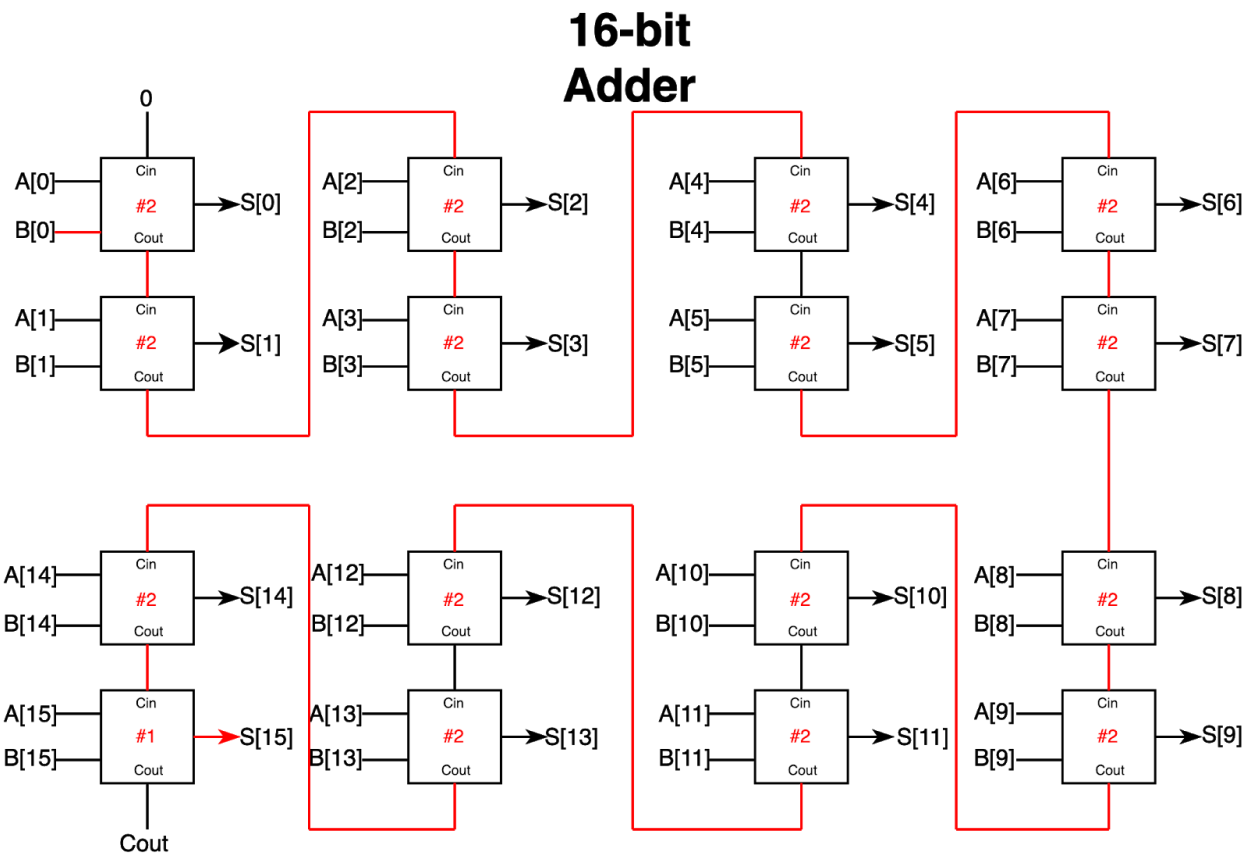


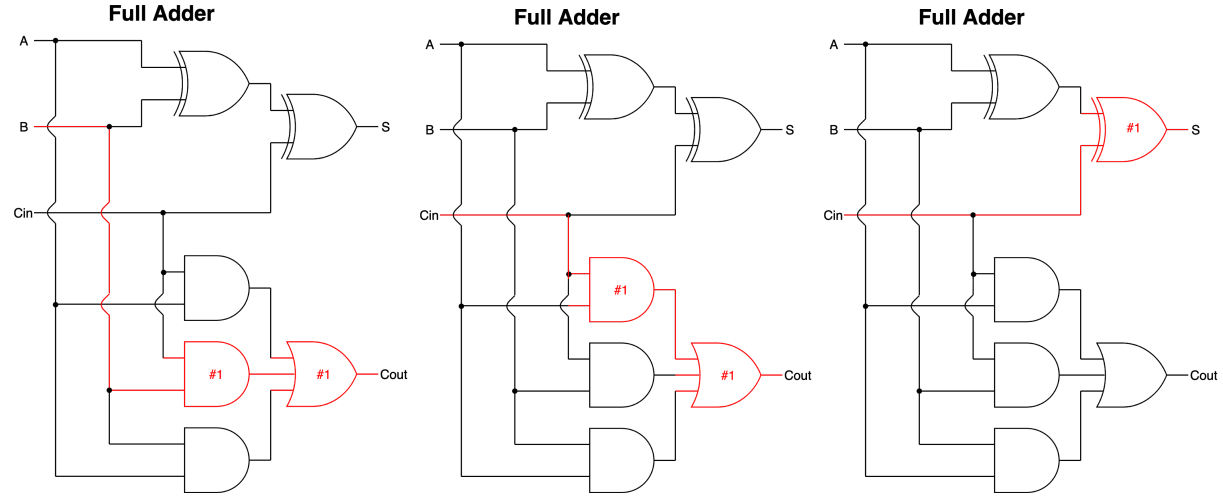
Critical Path & Delay

The critical path of the ALU is shown highlighted in red in the diagrams below. The delay of the critical path is 68 delay units, and the circuit uses a total of 432 gates.

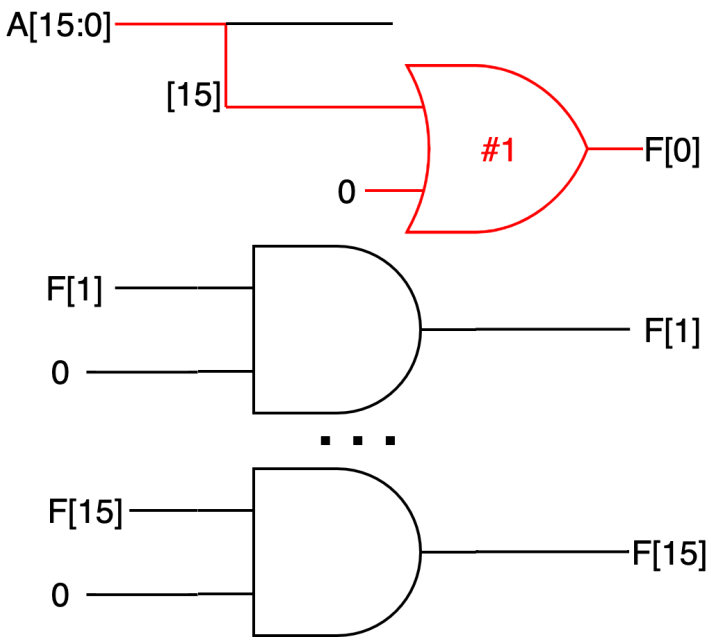




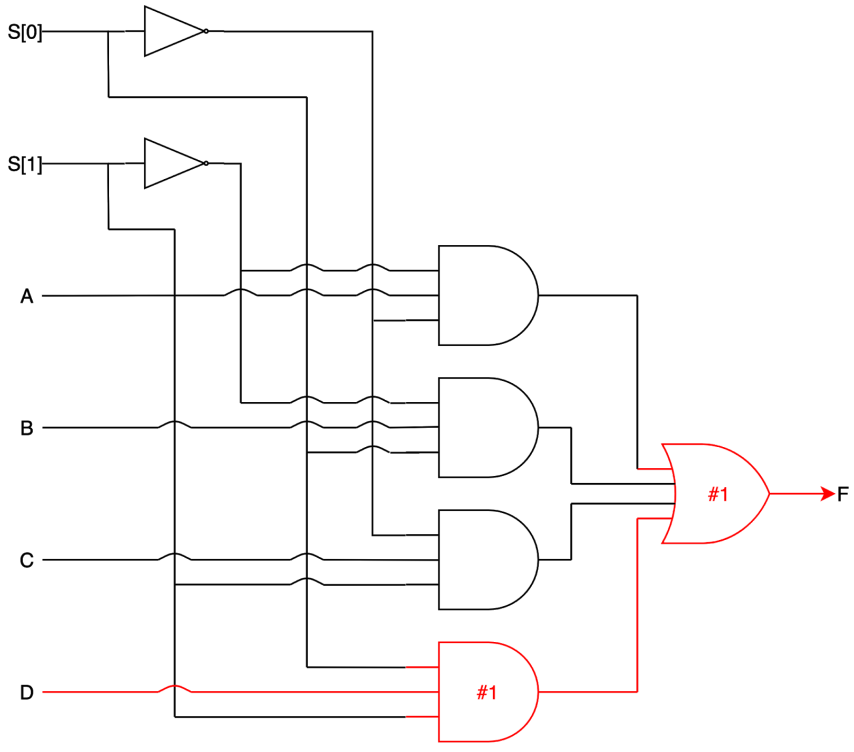

16-bit
Adder
0
A[0]
B[0]
A[1]
B[1]
A[2]
B[2]
A[3]
B[3]
A[4]
B[4]
A[5]
B[5]
A[6]
B[6]
A[7]
B[7]
A[8]
B[8]
A[9]
B[9]
A[10]
B[10]
A[11]
B[11]
A[12]
B[12]
A[13]
B[13]
A[14]
B[14]
A[15]
B[15]
Cin
#2
#1
Cout
S[0]
S[1]
S[2]
S[3]
S[4]
S[5]
S[6]
S[7]
S[8]
S[9]
S[10]
S[11]
S[12]
S[13]
S[14]
S[15]
Cout


Full Adder
A
B
Cin
S
#1
Cout
Full Adder
A
B
Cin
S
#1
Cout
Full Adder
A
B
Cin
#1
S
Cout

SLT



4-1 Multiplexer



Functional Simulation Results

```
0      opcode=xxx, a=xxxxxxxxxxxxxxxx, b=xxxxxxxxxxxxxxxx, f=xxxxxxxxxxxxxxxx
0      opcode=xxx, a=xxxxxxxxxxxxxxxx, b=xxxxxxxxxxxxxxxx, f=xxxxxxxxxxxxxxxx

100    opcode=000, a=0000000000000111, b=0000000000000100, f=0000000000000100
200    opcode=000, a=0000000000011001, b=1111111111011000, f=0000000000011000
300    opcode=000, a=1111111110110011, b=0000000000100001, f=0000000000100001
400    opcode=000, a=1111111001110011, b=1111110110011010, f=1111110000010010

500    opcode=001, a=0000000000000111, b=0000000000000100, f=0000000000000111
600    opcode=001, a=0000000000011001, b=1111111111011000, f=1111111111011001
700    opcode=001, a=1111111110110011, b=0000000000100001, f=1111111110110011
800    opcode=001, a=1111111001110011, b=1111110110011010, f=1111111111111011

900    opcode=010, a=0000000000000111, b=0000000000000100, f=0000000000001011
1000   opcode=010, a=0000000000011001, b=1111111111011000, f=1111111111110001
1100   opcode=010, a=1111111110110011, b=0000000000100001, f=1111111111010100
1200   opcode=010, a=1111111001110011, b=1111110110011010, f=1111110000001101

1300   opcode=110, a=0000000000000111, b=0000000000000100, f=0000000000000011
1400   opcode=110, a=0000000000011001, b=1111111111011000, f=0000000001000001
1500   opcode=110, a=1111111110110011, b=0000000000100001, f=1111111110010010
1600   opcode=110, a=1111111001110011, b=1111110110011010, f=0000000011011001

1700   opcode=111, a=0000000000000111, b=0000000000000100, f=0000000000000000
1800   opcode=111, a=0000000000011001, b=1111111111011000, f=0000000000000000
1900   opcode=111, a=1111111110110011, b=0000000000100001, f=0000000000000001
2000   opcode=111, a=1111111001110011, b=1111110110011010, f=0000000000000000

2100   opcode=111, a=1111111001110011, b=1111110110011010, f=0000000000000000
```

Temporal Delay Results & Analysis

```
0       opcode=xxx, a=xxxxxxxxxxxxxxx, b=xxxxxxxxxxxxxxx, f=xxxxxxxxxxxxxxx
0       opcode=xxx, a=xxxxxxxxxxxxxxx, b=xxxxxxxxxxxxxxx, f=xxxxxxxxxxxxxxx

100     opcode=000, a=0000000000000111, b=0000000000000100, f=xxxxxxxxxxxxxxx
103     opcode=000, a=0000000000000111, b=0000000000000100, f=0000000000000100
200     opcode=000, a=0000000000011001, b=1111111111011000, f=0000000000000100
203     opcode=000, a=0000000000011001, b=1111111111011000, f=0000000000011000
300     opcode=000, a=1111111110110011, b=0000000000100001, f=0000000000011000
303     opcode=000, a=1111111110110011, b=0000000000100001, f=0000000000100001
400     opcode=000, a=1111111001110011, b=1111110110011010, f=0000000000100001
403     opcode=000, a=1111111001110011, b=1111110110011010, f=1111110000010010

500     opcode=001, a=0000000000000111, b=0000000000000100, f=1111110000010010
502     opcode=001, a=0000000000000111, b=0000000000000100, f=1111111111111011
503     opcode=001, a=0000000000000111, b=0000000000000100, f=0000000000000111
600     opcode=001, a=0000000000011001, b=1111111111011000, f=0000000000000111
603     opcode=001, a=0000000000011001, b=1111111111011000, f=1111111111011001
700     opcode=001, a=1111111110110011, b=0000000000100001, f=1111111111011001
703     opcode=001, a=1111111110110011, b=0000000000100001, f=1111111110110011
800     opcode=001, a=1111111001110011, b=1111110110011010, f=1111111110110011
803     opcode=001, a=1111111001110011, b=1111110110011010, f=1111111111111011

900     opcode=010, a=0000000000000111, b=0000000000000100, f=1111111111111011
902     opcode=010, a=0000000000000111, b=0000000000000100, f=0000000000000000
903     opcode=010, a=0000000000000111, b=0000000000000100, f=1111110000001101
904     opcode=010, a=0000000000000111, b=0000000000000100, f=0000001001111001
905     opcode=010, a=0000000000000111, b=0000000000000100, f=0000011010010001
906     opcode=010, a=0000000000000111, b=0000000000000100, f=1111101100001111
907     opcode=010, a=0000000000000111, b=0000000000000100, f=0000000000001011
1000    opcode=010, a=0000000000011001, b=1111111111011000, f=0000000000001011
1004    opcode=010, a=0000000000011001, b=1111111111011000, f=0000000000010101
1005    opcode=010, a=0000000000011001, b=1111111111011000, f=0000000000001101
1006    opcode=010, a=0000000000011001, b=1111111111011000, f=1111111111010001
1007    opcode=010, a=0000000000011001, b=1111111111011000, f=1111111111110001
1100    opcode=010, a=1111111110110011, b=0000000000100001, f=1111111111110001
1104    opcode=010, a=1111111110110011, b=0000000000100001, f=0000000001011011
1105    opcode=010, a=1111111110110011, b=0000000000100001, f=1111111100001011
1106    opcode=010, a=1111111110110011, b=0000000000100001, f=0000000011110010
1107    opcode=010, a=1111111110110011, b=0000000000100001, f=0000000111010000
1109    opcode=010, a=1111111110110011, b=0000000000100001, f=0000001111010100
1111    opcode=010, a=1111111110110011, b=0000000000100001, f=0000011111010100
1113    opcode=010, a=1111111110110011, b=0000000000100001, f=0000111111010100
1115    opcode=010, a=1111111110110011, b=0000000000100001, f=0001111111010100
1117    opcode=010, a=1111111110110011, b=0000000000100001, f=0011111111010100
```

```
1119   opcode=010, a=1111111110110011, b=0000000000100001, f=0111111111010100
1121   opcode=010, a=1111111110110011, b=0000000000100001, f=1111111111010100
1200   opcode=010, a=1111111001110011, b=1111110110011010, f=1111111111010100
1204   opcode=010, a=1111111001110011, b=1111110110011010, f=1111111000010100
1205   opcode=010, a=1111111001110011, b=1111110110011010, f=1111111010010100
1206   opcode=010, a=1111111001110011, b=1111110110011010, f=0000001100101111
1207   opcode=010, a=1111111001110011, b=1111110110011010, f=1111101001001101
1209   opcode=010, a=1111111001110011, b=1111110110011010, f=1111100010001101
1211   opcode=010, a=1111111001110011, b=1111110110011010, f=1111110100001101
1213   opcode=010, a=1111111001110011, b=1111110110011010, f=1111111000001101
1215   opcode=010, a=1111111001110011, b=1111110110011010, f=1111100000001101
1217   opcode=010, a=1111111001110011, b=1111110110011010, f=1111110000001101

1300   opcode=110, a=0000000000000111, b=0000000000000100, f=1111110000001101
1304   opcode=110, a=0000000000000111, b=0000000000000100, f=0000001001111001
1305   opcode=110, a=0000000000000111, b=0000000000000100, f=0000011010010001
1306   opcode=110, a=0000000000000111, b=0000000000000100, f=1111100101101101
1307   opcode=110, a=0000000000000111, b=0000000000000100, f=0000011001101101
1309   opcode=110, a=0000000000000111, b=0000000000000100, f=1111111111110011
1310   opcode=110, a=0000000000000111, b=0000000000000100, f=1111011111100011
1311   opcode=110, a=0000000000000111, b=0000000000000100, f=1111111111100011
1312   opcode=110, a=0000000000000111, b=0000000000000100, f=1110111111000011
1313   opcode=110, a=0000000000000111, b=0000000000000100, f=1111111111000011
1314   opcode=110, a=0000000000000111, b=0000000000000100, f=1101111110000011
1315   opcode=110, a=0000000000000111, b=0000000000000100, f=1111111110000011
1316   opcode=110, a=0000000000000111, b=0000000000000100, f=1011111100000011
1317   opcode=110, a=0000000000000111, b=0000000000000100, f=1111111100000011
1318   opcode=110, a=0000000000000111, b=0000000000000100, f=0111111000000011
1319   opcode=110, a=0000000000000111, b=0000000000000100, f=1111111000000011
1320   opcode=110, a=0000000000000111, b=0000000000000100, f=1111110000000011
1322   opcode=110, a=0000000000000111, b=0000000000000100, f=1111100000000011
1324   opcode=110, a=0000000000000111, b=0000000000000100, f=1111000000000011
1326   opcode=110, a=0000000000000111, b=0000000000000100, f=1110000000000011
1328   opcode=110, a=0000000000000111, b=0000000000000100, f=1100000000000011
1330   opcode=110, a=0000000000000111, b=0000000000000100, f=1000000000000011
1332   opcode=110, a=0000000000000111, b=0000000000000100, f=0000000000000011
1400   opcode=110, a=0000000000011001, b=1111111111011000, f=0000000000000011
1404   opcode=110, a=0000000000011001, b=1111111111011000, f=0000000000011101
1405   opcode=110, a=0000000000011001, b=1111111111011000, f=0000000000010101
1409   opcode=110, a=0000000000011001, b=1111111111011000, f=1111111111001001
1410   opcode=110, a=0000000000011001, b=1111111111011000, f=0000000001010001
1411   opcode=110, a=0000000000011001, b=1111111111011000, f=0000000001000001
1412   opcode=110, a=0000000000011001, b=1111111111011000, f=0000000001100001
1413   opcode=110, a=0000000000011001, b=1111111111011000, f=0000000001000001
1414   opcode=110, a=0000000000011001, b=1111111111011000, f=0000000000000001
1415   opcode=110, a=0000000000011001, b=1111111111011000, f=0000000001000001
1500   opcode=110, a=1111111110110011, b=0000000000100001, f=0000000001000001
1504   opcode=110, a=1111111110110011, b=0000000000100001, f=1111111111101011
```

```
1505    opcode=110, a=1111111110110011, b=0000000000100001, f=1111111111111011
1507    opcode=110, a=1111111110110011, b=0000000000100001, f=1111111111011011
1509    opcode=110, a=1111111110110011, b=0000000000100001, f=0000000000100010
1510    opcode=110, a=1111111110110011, b=0000000000100001, f=1111111111010010
1511    opcode=110, a=1111111110110011, b=0000000000100001, f=1111111111110110
1512    opcode=110, a=1111111110110011, b=0000000000100001, f=1111111100010010
1513    opcode=110, a=1111111110110011, b=0000000000100001, f=1111111100011010
1514    opcode=110, a=1111111110110011, b=0000000000100001, f=1111111110010010
1515    opcode=110, a=1111111110110011, b=0000000000100001, f=1111111110000010
1516    opcode=110, a=1111111110110011, b=0000000000100001, f=1111111110010010
1517    opcode=110, a=1111111110110011, b=0000000000100001, f=1111111110110010
1518    opcode=110, a=1111111110110011, b=0000000000100001, f=1111111110010010
1600    opcode=110, a=1111111001110011, b=1111110110011010, f=1111111110010010
1604    opcode=110, a=1111111001110011, b=1111110110011010, f=1111111001010010
1609    opcode=110, a=1111111001110011, b=1111110110011010, f=0000001111101001
1610    opcode=110, a=1111111001110011, b=1111110110011010, f=0000000011111001
1612    opcode=110, a=1111111001110011, b=1111110110011010, f=0000000011011001

1700    opcode=111, a=0000000000000111, b=0000000000000100, f=0000000011011001
1703    opcode=111, a=0000000000000111, b=0000000000000100, f=0000000000000000
1705    opcode=111, a=0000000000000111, b=0000000000000100, f=0000000000000001
1706    opcode=111, a=0000000000000111, b=0000000000000100, f=0000000000000000
1710    opcode=111, a=0000000000000111, b=0000000000000100, f=0000000000000001
1733    opcode=111, a=0000000000000111, b=0000000000000100, f=0000000000000000
1800    opcode=111, a=0000000000011001, b=1111111111011000, f=0000000000000000
1810    opcode=111, a=0000000000011001, b=1111111111011000, f=0000000000000001
1811    opcode=111, a=0000000000011001, b=1111111111011000, f=0000000000000000
1900    opcode=111, a=1111111110110011, b=0000000000100001, f=0000000000000000
1905    opcode=111, a=1111111110110011, b=0000000000100001, f=0000000000000001
1910    opcode=111, a=1111111110110011, b=0000000000100001, f=0000000000000000
1911    opcode=111, a=1111111110110011, b=0000000000100001, f=0000000000000001
2000    opcode=111, a=1111111001110011, b=1111110110011010, f=0000000000000001
2010    opcode=111, a=1111111001110011, b=1111110110011010, f=0000000000000000
2100    opcode=111, a=1111111001110011, b=1111110110011010, f=0000000000000000
```

Example 1

```
100     opcode=000, a=0000000000000111, b=0000000000000100, f=xxxxxxxxxxxxxxxx
103     opcode=000, a=0000000000000111, b=0000000000000100, f=0000000000000100
```

In this example, 7 is ANDed with 4. The delay of this function is only 3 units, as the AND gate is 1 unit and the multiplexor is 2 units.

Example 2

```
600    opcode=001, a=0000000000011001, b=1111111111011000, f=0000000000000111
603    opcode=001, a=0000000000011001, b=1111111111011000, f=1111111111011001
```

25 is being ORed with 40 in this example. Once again, the delay is only 3 units because of the OR gate and multiplexor.

Example 3

```
1000   opcode=010, a=0000000000011001, b=1111111111011000, f=0000000000001011
1004   opcode=010, a=0000000000011001, b=1111111111011000, f=0000000000010101
1005   opcode=010, a=0000000000011001, b=1111111111011000, f=0000000000001101
1006   opcode=010, a=0000000000011001, b=1111111111011000, f=1111111111010001
1007   opcode=010, a=0000000000011001, b=1111111111011000, f=1111111111110001
```

This example is a bit different than the rest. The function is 25 + -40. As you can see, the circuit takes longer to complete because of the ripple carry adder. Adding and subtracting takes much longer to complete than a simple AND or OR.

Example 4

```
1400   opcode=110, a=0000000000011001, b=1111111111011000, f=0000000000000011
1404   opcode=110, a=0000000000011001, b=1111111111011000, f=0000000000011101
1405   opcode=110, a=0000000000011001, b=1111111111011000, f=0000000000010101
1409   opcode=110, a=0000000000011001, b=1111111111011000, f=1111111111001001
1410   opcode=110, a=0000000000011001, b=1111111111011000, f=0000000001010001
1411   opcode=110, a=0000000000011001, b=1111111111011000, f=0000000001000001
1412   opcode=110, a=0000000000011001, b=1111111111011000, f=0000000001100001
1413   opcode=110, a=0000000000011001, b=1111111111011000, f=0000000001000001
1414   opcode=110, a=0000000000011001, b=1111111111011000, f=0000000000000001
1415   opcode=110, a=0000000000011001, b=1111111111011000, f=0000000001000001
```

Example 4 is very similar to example 3, as subtracting is very similar to adding. The ALU doesn't actually subtract the second number from the first, it actually adds the two's complement version of the second number. In this case, we are subtracting 25 - -40. To convert from one sign to the other, the number must be NOTed then have one added to it. Given that the ripple carry adder is being run twice, it makes sense that the delay is essentially twice as long.

Example 5

```
1700   opcode=111, a=0000000000000111, b=0000000000000100, f=0000000011011001
1703   opcode=111, a=0000000000000111, b=0000000000000100, f=0000000000000000
1705   opcode=111, a=0000000000000111, b=0000000000000100, f=0000000000000001
1706   opcode=111, a=0000000000000111, b=0000000000000100, f=0000000000000000
1710   opcode=111, a=0000000000000111, b=0000000000000100, f=0000000000000001
1733   opcode=111, a=0000000000000111, b=0000000000000100, f=0000000000000000
```

In the final example, SLT is used to determine whether or not 7 is less than 4. SLT will return 1 if the first number is less than the second, and 0 if it isn't. The way it accomplishes this is by subtracting the second number from the first, and returning 0 if the result is positive and 1 if the result is negative. The ALU will essentially return the most significant bit of the result of 7 - 4. The subtraction is why SLT has such a long delay.

5000 Input Test

To test the ALU across 5000 randomly generated inputs, I first wrote a script that generated the inputs ranging from -32000 to 32000. After running the simulation, I then wrote another script that calculated the average delay for each test case. The program determined that the average delay for my ALU was 8 delay units. This is not too surprising, as since the opcodes were also randomly generated, roughly ⅜ or 40% of the tests were AND or OR operations, which appear to always have a delay of 3. Given that and the fact that the other operations usually had a delay between 10 and 25, an average delay of 8 is somewhat expected. This is obviously a far cry from the critical path of 68 delay units, meaning that the ALU was rarely hitting that much, if at all.