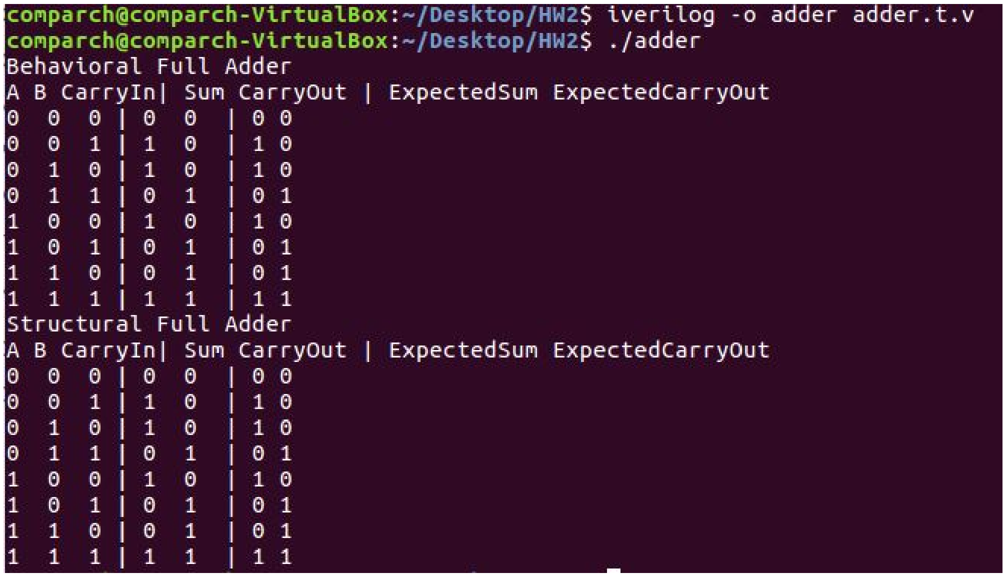
CompArch HW2

Maggie Jakus

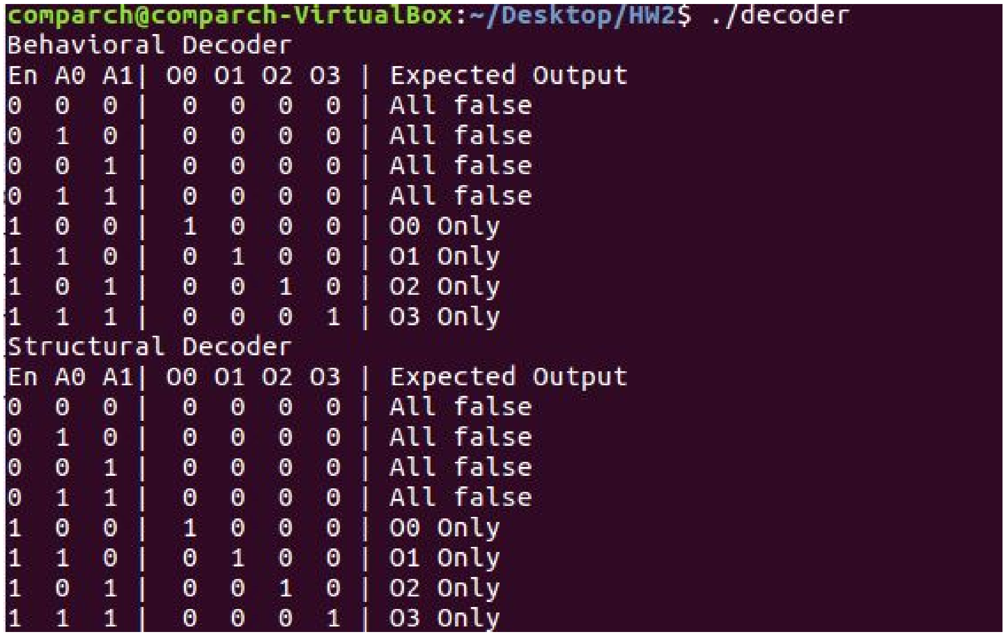
September 19, 2017

1 Test Bench Results



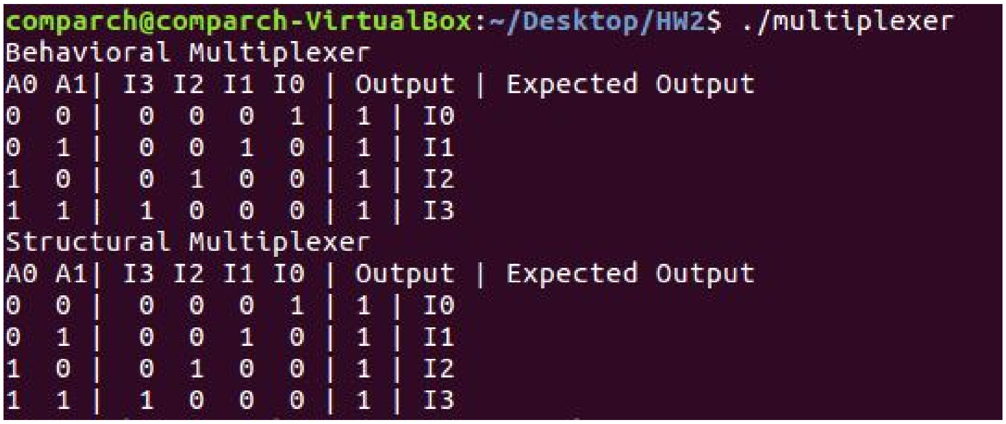
```
comparch@comparch-VirtualBox:~/Desktop/HW2$ iverilog -o adder adder.t.v
comparch@comparch-VirtualBox:~/Desktop/HW2$ ./adder
Behavioral Full Adder
A B CarryIn| Sum CarryOut | ExpectedSum ExpectedCarryOut
0  0  0 | 0  0  | 0 0
0  0  1 | 1  0  | 1 0
0  1  0 | 1  0  | 1 0
0  1  1 | 0  1  | 0 1
1  0  0 | 1  0  | 1 0
1  0  1 | 0  1  | 0 1
1  1  0 | 0  1  | 0 1
1  1  1 | 1  1  | 1 1
Structural Full Adder
A B CarryIn| Sum CarryOut | ExpectedSum ExpectedCarryOut
0  0  0 | 0  0  | 0 0
0  0  1 | 1  0  | 1 0
0  1  0 | 1  0  | 1 0
0  1  1 | 0  1  | 0 1
1  0  0 | 1  0  | 1 0
1  0  1 | 0  1  | 0 1
1  1  0 | 0  1  | 0 1
1  1  1 | 1  1  | 1 1
```

Figure 1: The structural adder was created with simple gates. The results from the Structural Full Adder match the expected truth table.



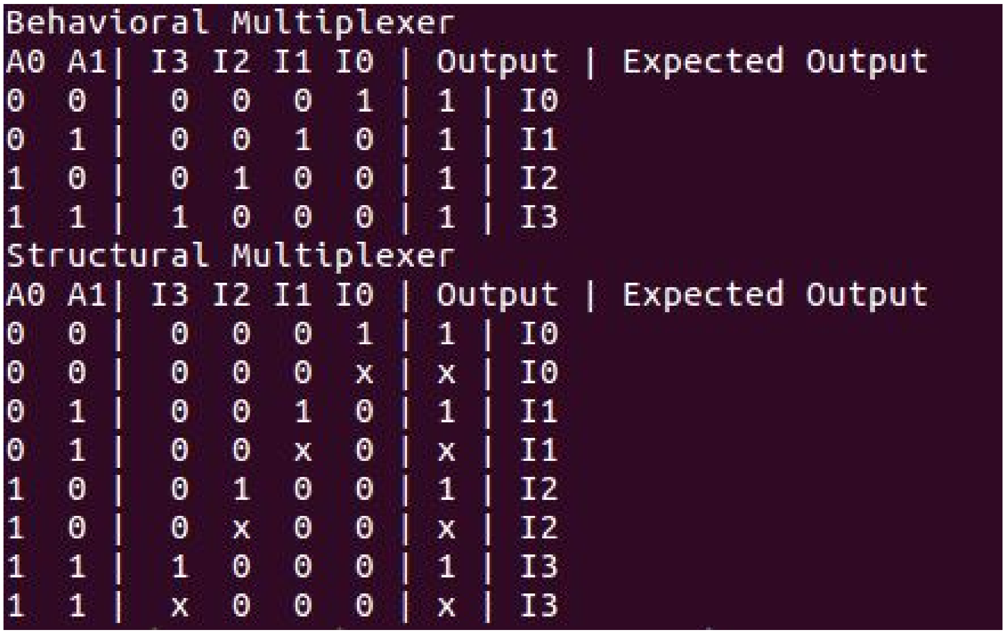
```
comparch@comparch-VirtualBox:~/Desktop/HW2$ ./decoder
Behavioral Decoder
En A0 A1| O0 O1 O2 O3 | Expected Output
0  0  0 |  0  0  0  0 | All false
0  1  0 |  0  0  0  0 | All false
0  0  1 |  0  0  0  0 | All false
0  1  1 |  0  0  0  0 | All false
1  0  0 |  1  0  0  0 | O0 Only
1  1  0 |  0  1  0  0 | O1 Only
1  0  1 |  0  0  1  0 | O2 Only
1  1  1 |  0  0  0  1 | O3 Only
Structural Decoder
En A0 A1| O0 O1 O2 O3 | Expected Output
0  0  0 |  0  0  0  0 | All false
0  1  0 |  0  0  0  0 | All false
0  0  1 |  0  0  0  0 | All false
0  1  1 |  0  0  0  0 | All false
1  0  0 |  1  0  0  0 | O0 Only
1  1  0 |  0  1  0  0 | O1 Only
1  0  1 |  0  0  1  0 | O2 Only
1  1  1 |  0  0  0  1 | O3 Only
```

Figure 2: The structural decoder was created with simple gates. The results from the Structural Decoder match the expected truth table.



```
comparch@comparch-VirtualBox:~/Desktop/HW2$ ./multiplexer
Behavioral Multiplexer
A0 A1| I3 I2 I1 I0 | Output | Expected Output
0  0 |  0  0  0  1 | 1 | I0
0  1 |  0  0  1  0 | 1 | I1
1  0 |  0  1  0  0 | 1 | I2
1  1 |  1  0  0  0 | 1 | I3
Structural Multiplexer
A0 A1| I3 I2 I1 I0 | Output | Expected Output
0  0 |  0  0  0  1 | 1 | I0
0  1 |  0  0  1  0 | 1 | I1
1  0 |  0  1  0  0 | 1 | I2
1  1 |  1  0  0  0 | 1 | I3
```

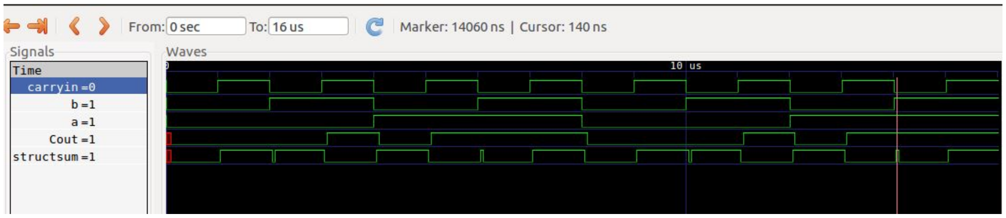
Figure 3: The structural multiplexer was created with simple gates. The results from the Structural Multiplexer match the expected truth table.



```
Behavioral Multiplexer
A0 A1| I3 I2 I1 I0 | Output | Expected Output
0  0 |  0  0  0  1 | 1 | I0
0  1 |  0  0  1  0 | 1 | I1
1  0 |  0  1  0  0 | 1 | I2
1  1 |  1  0  0  0 | 1 | I3
Structural Multiplexer
A0 A1| I3 I2 I1 I0 | Output | Expected Output
0  0 |  0  0  0  1 | 1 | I0
0  0 |  0  0  0  x | x | I0
0  1 |  0  0  1  0 | 1 | I1
0  1 |  0  0  x  0 | x | I1
1  0 |  0  1  0  0 | 1 | I2
1  0 |  0  x  0  0 | x | I2
1  1 |  1  0  0  0 | 1 | I3
1  1 |  x  0  0  0 | x | I3
```

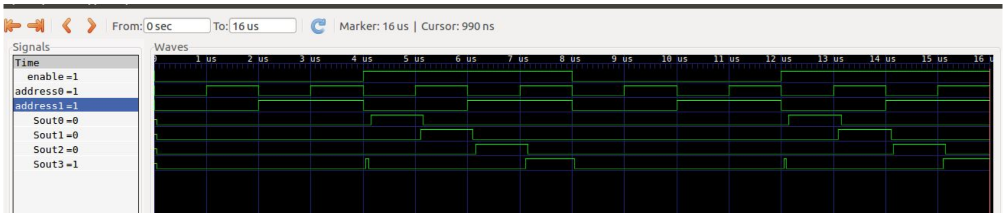
Figure 4: The multiplexer table seen in figure 3 but with additional tests. While there are far too many input combinations to reasonably input, I used the variable "x" to highlight the relationship between inputs and outputs.

2 Waveforms



From: 0 sec To: 16 us Marker: 14060 ns | Cursor: 140 ns

Figure 5: The waveforms generated from the adder table. There are blips caused by the delays that cause things to be 1 too early or too late, triggering other outputs.



From: 0 sec To: 16 us Marker: 16 us | Cursor: 990 ns

Figure 6: The waveforms generated from the decoder table. The blips are once again caused by the delays that cause certain inputs to be triggered longer than they should be.

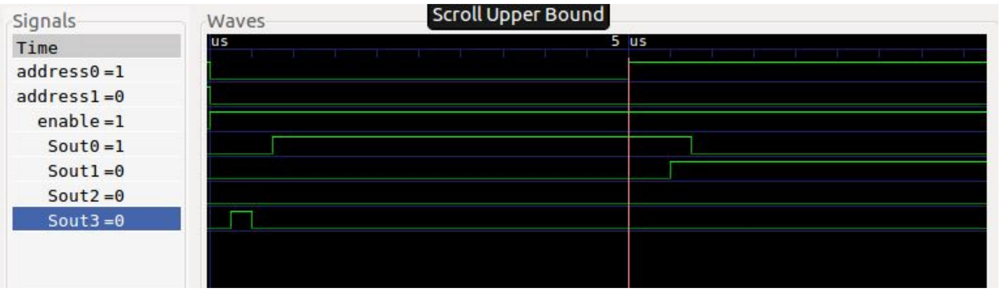


Figure 7: A zoomed in version of figure 6. As you can see, address0 switches from 0 to 1, but Sout1 does not switch from 0 to 1 until a noticeable amount of time later.

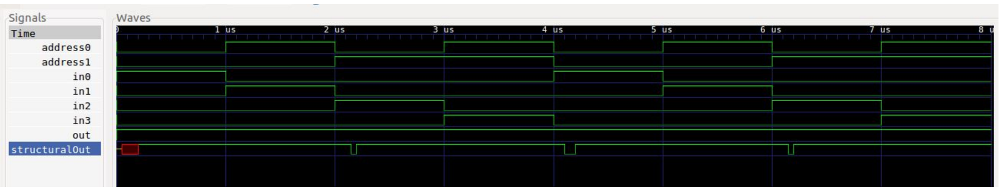


Figure 8: The waveforms generated from the multiplexer table. Here I show two outputs, one from the Behavioral Multiplexer (“out”) and one from the Structural Multiplexer (“structuralOut”). As you can see, the “out” is perfectly smooth, while the “structuralOut” suffers blips, caused by the delays in its system.