Ridge Tejuco

Fall 2019

California State University, Northridge

Department of Electrical and Computer Engineering

ECE 526L

Experiment # 1
Familiarization with Linux and the Synopsys VCS Simulator

Sept. 12, 2019

Authors: Ridge Tejuco
Professor: Ronald Mehler Ph.D

I hereby attest that this lab report is entirely my own work. I have not copied either code or text from anyone, nor have I allowed or will I allow anyone to copy my work.

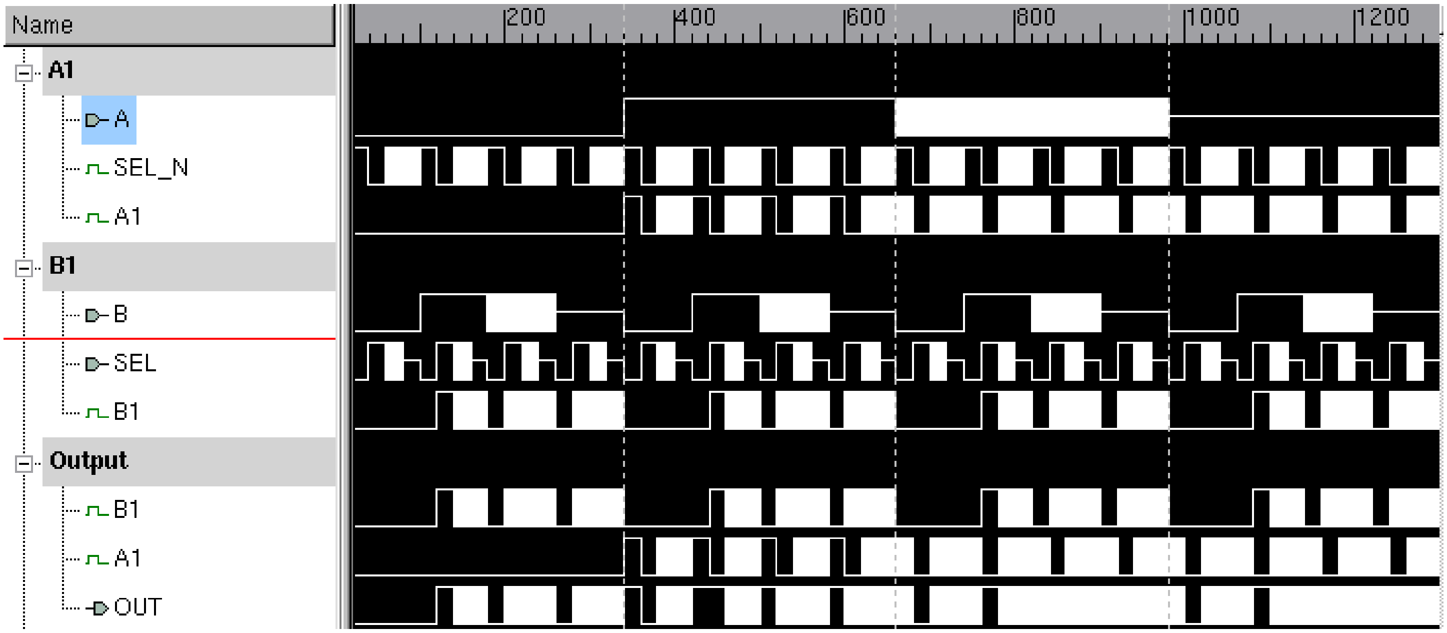
Name (printed) ________________________________

Name (signed) ________________________________

Date _________________

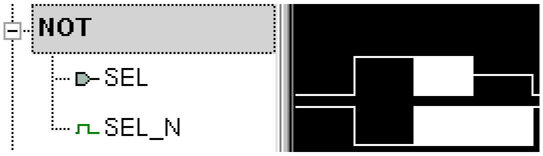
Analysis

Figure 1. Multiplexer wave view



The above waveform shows all possible inputs and outputs for the internal gates and overall module. The low value represents a logic value of 0. The high value represents a logic value of 1. The white blocks represent an unknown input, and the medium height values represent High impedance.

Figure 2. NOT gate (SEL_N = not SEL)



Looking closer at the SEL input of the not gate, the gate functions as intended.
A logic 0 results in logic 1, a logic 1 results in logic 0. Most notably both unknown and high impedance result in an unknown output. This will be the same for other primitives.

```
not 0 = 1       not x = x
not 1 = 0       not z = x
```

Figure 3. A1 AND gate (A1 = A and SEL_N)

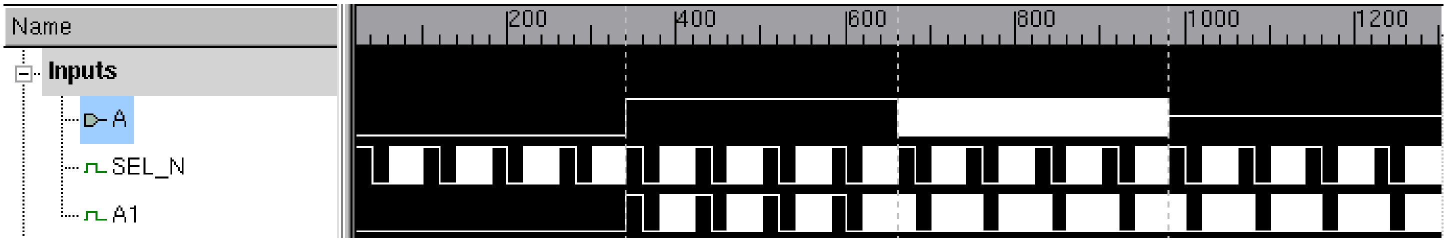
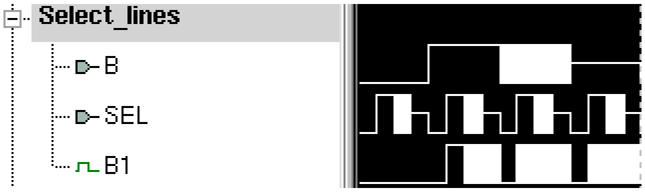


Figure 4. B1 AND gate (B1 = SEL and B)



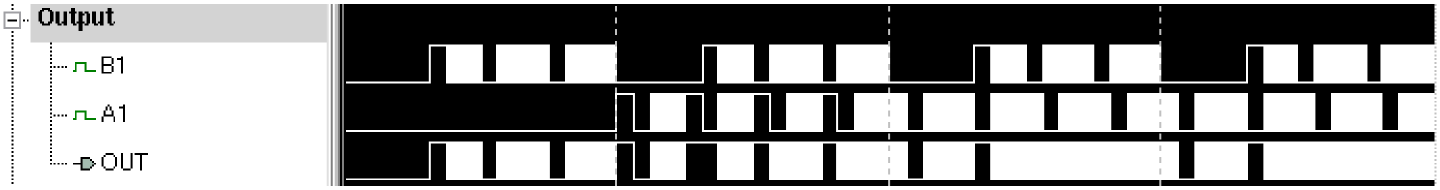
Looking close at the Figures 3 and 4, the following equations are true by evaluating the wave from left to right. These are the expected output of AND gates.

0 and 0 = 0
0 and 1 = 0
0 and X = 0
0 and Z = 0

1 and 1 = 1
1 and x = x
1 and z = x

x and x = x
x and z = x

Figure 5. OUT OR gate (OUT = B1 or A1)



Looking closer at figure five, the following cases are true. These are expected values of OR gates.

0 or 0 = 0
0 or 1 = 1
0 or x = x

x or x = x

A few possible cases of OR gates are omitted. For example, there is no case involving the OR of high impedance. There are also no cases in which are input are 1 and X at the same time. The following theoretical cases were filtered out by the A1 and B1 AND gates.

0 or z = x

1 or x = 1
1 or z = 1

x or z = x

Ridge Tejuco

Additional Question

The 2:1 Multiplexer on pages 9 and 10 are not the same as the Multiplexer used in Experiment 1. The example on page 10 has an ENABLE input with a function where when ENABLE = 0, the output MUX_OUT = z. Otherwise, if ENABLE = 1, Multiplexer will function almost similarly to the design in Experiment 1. There is another issue, in the design of Experiment 1, the high impedance of inputs A,B, and SEL are eventually filtered out by AND gates. However, in the example design, High impedance may be output from the entire module because IF-ELSE statements might not filter them.

```
//======================================================
// Ridge Tejuco - CSUN
//======================================================
// LAB 1
// MUX2_1.v
// September 5, 2019
//======================================================
`timescale 1 ns / 1 ns

module MUX2_1(OUT, A, B, SEL);
    //Port declarations
    output OUT;
    input A, B, SEL;

    //Internal variable declarations
    wire A1, B1, SEL_N;

    //The netlist===================================
    not (SEL_N, SEL);

    and (A1, A, SEL_N);     //A1 = A and SEL_N;

    and (B1,B,SEL);              //B1 = B and SEL;

    or (OUT,A1,B1);         //OUT = A1 or B1;
endmodule
//======================================================
//==========================================================
// Ridge Tejuco - CSUN
//==========================================================
// LAB 1
// TB_MUX2_1.v
// September 5, 2019
//==========================================================
`timescale 1 ns / 1 ns
module TB_MUX2_1();
    reg A, B, SEL;
    wire OUT;

    MUX2_1 UUT(OUT, A, B, SEL);

    initial
    $monitorb ("%d out = %b a = %b b = %b sel = %b", $time, OUT, A, B, SEL);

    initial begin
      $vcdpluson;
      A = 1;B = 1; SEL = 1'bx;
      //==================================================
      // test for A =0; B = 0; then SEL = (0,1,1'bx,1'bz)
      // repeat for B = 0;B = 1; B = 1'bx; B= 1'bz;
      //==========================
      #20 A = 0;B = 0; SEL = 0;
      #20         SEL = 1;
      #20         SEL = 1'bx;
      #20         SEL = 1'bz;

      #20       B = 1; SEL = 0;
      #20         SEL = 1;
      #20              SEL = 1'bx;
      #20         SEL = 1'bz;

      #20       B = 1'bx; SEL = 0;
      #20            SEL = 1;
      #20            SEL = 1'bx;
      #20            SEL = 1'bz;

      #20       B = 1'bz; SEL = 0;
      #20            SEL = 1;
      #20            SEL = 1'bx;
      #20            SEL = 1'bz;
      //==================================================
      // Repeat for A = 1, then A = 1'bx, then A = 1'bz
```

```
//=====================================================
#20 A = 1;B = 0; SEL = 0;
#20         SEL = 1;
#20         SEL = 1'bx;
#20         SEL = 1'bz;

#20       B = 1; SEL = 0;
#20         SEL = 1;
#20              SEL = 1'bx;
#20         SEL = 1'bz;

#20       B = 1'bx; SEL = 0;
#20            SEL = 1;
#20            SEL = 1'bx;
#20            SEL = 1'bz;

#20       B = 1'bz; SEL = 0;
#20            SEL = 1;
#20            SEL = 1'bx;
#20            SEL = 1'bz;
//=====================================================
#20 A = 1'bx;B = 0; SEL = 0;
#20         SEL = 1;
#20         SEL = 1'bx;
#20         SEL = 1'bz;

#20       B = 1; SEL = 0;
#20         SEL = 1;
#20              SEL = 1'bx;
#20         SEL = 1'bz;

#20       B = 1'bx; SEL = 0;
#20            SEL = 1;
#20            SEL = 1'bx;
#20            SEL = 1'bz;

#20       B = 1'bz; SEL = 0;
#20            SEL = 1;
#20            SEL = 1'bx;
#20            SEL = 1'bz;
//=====================================================
#20 A = 1'bz;B = 0; SEL = 0;
#20         SEL = 1;
#20         SEL = 1'bx;
#20         SEL = 1'bz;
```

```
      #20       B = 1; SEL = 0;
      #20         SEL = 1;
      #20              SEL = 1'bx;
      #20         SEL = 1'bz;

      #20       B = 1'bx; SEL = 0;
      #20            SEL = 1;
      #20            SEL = 1'bx;
      #20            SEL = 1'bz;

      #20       B = 1'bz; SEL = 0;
      #20            SEL = 1;
      #20            SEL = 1'bx;
      #20            SEL = 1'bz;

      #20 $finish;
    end
Endmodule
```

```
//=======================================================
// Ridge Tejuco - CSUN
//=======================================================
// LAB 1
// lab1.log
// September 5, 2019
//=======================================================
Chronologic VCS simulator copyright 1991-2017
Contains Synopsys proprietary information.
Compiler version N-2017.12-SP2-2_Full64; Runtime version
N-2017.12-SP2-2_Full64;  Sep  5 19:44 2019
VCD+ Writer N-2017.12-SP2-2_Full64 Copyright (c) 1991-2017 by Synopsys Inc.
                0 out = x a = 1 b = 1 sel = x
                20 out = 0 a = 0 b = 0 sel = 0
                40 out = 0 a = 0 b = 0 sel = 1
                60 out = 0 a = 0 b = 0 sel = x
                80 out = 0 a = 0 b = 0 sel = z
                100 out = 0 a = 0 b = 1 sel = 0
                120 out = 1 a = 0 b = 1 sel = 1
                140 out = x a = 0 b = 1 sel = x
                160 out = x a = 0 b = 1 sel = z
                180 out = 0 a = 0 b = x sel = 0
                200 out = x a = 0 b = x sel = 1
                220 out = x a = 0 b = x sel = x
                240 out = x a = 0 b = x sel = z
                260 out = 0 a = 0 b = z sel = 0
                280 out = x a = 0 b = z sel = 1
                300 out = x a = 0 b = z sel = x
                320 out = x a = 0 b = z sel = z
                340 out = 1 a = 1 b = 0 sel = 0
                360 out = 0 a = 1 b = 0 sel = 1
                380 out = x a = 1 b = 0 sel = x
                400 out = x a = 1 b = 0 sel = z
                420 out = 1 a = 1 b = 1 sel = 0
                440 out = 1 a = 1 b = 1 sel = 1
                460 out = x a = 1 b = 1 sel = x
                480 out = x a = 1 b = 1 sel = z
                500 out = 1 a = 1 b = x sel = 0
                520 out = x a = 1 b = x sel = 1
                540 out = x a = 1 b = x sel = x
                560 out = x a = 1 b = x sel = z
                580 out = 1 a = 1 b = z sel = 0
                600 out = x a = 1 b = z sel = 1
                620 out = x a = 1 b = z sel = x
                640 out = x a = 1 b = z sel = z
                660 out = x a = x b = 0 sel = 0
                680 out = 0 a = x b = 0 sel = 1
                700 out = x a = x b = 0 sel = x
                720 out = x a = x b = 0 sel = z
                740 out = x a = x b = 1 sel = 0
                760 out = 1 a = x b = 1 sel = 1
                780 out = x a = x b = 1 sel = x
                800 out = x a = x b = 1 sel = z
                820 out = x a = x b = x sel = 0
                840 out = x a = x b = x sel = 1
                860 out = x a = x b = x sel = x
                880 out = x a = x b = x sel = z
                900 out = x a = x b = z sel = 0
                920 out = x a = x b = z sel = 1
                940 out = x a = x b = z sel = x
                960 out = x a = x b = z sel = z
                980 out = x a = z b = 0 sel = 0
                1000 out = 0 a = z b = 0 sel = 1
                1020 out = x a = z b = 0 sel = x
                1040 out = x a = z b = 0 sel = z
                1060 out = x a = z b = 1 sel = 0
                1080 out = 1 a = z b = 1 sel = 1
                1100 out = x a = z b = 1 sel = x
                1120 out = x a = z b = 1 sel = z
                1140 out = x a = z b = x sel = 0
                1160 out = x a = z b = x sel = 1
                1180 out = x a = z b = x sel = x
                1200 out = x a = z b = x sel = z
                1220 out = x a = z b = z sel = 0
                1240 out = x a = z b = z sel = 1
                1260 out = x a = z b = z sel = x
                1280 out = x a = z b = z sel = z
$finish called from file "TB_MUX2_1.v", line 96.
$finish at simulation time                 1300
           V C S   S i m u l a t i o n   R e p o r t
Time: 1300 ns
CPU Time:       0.250 seconds;        Data structure size:   0.0Mb
Thu Sep  5 19:44:32 2019
```