**ECE324 Homework1: Gate circuit model**

Name: \_\_Alex Blake\_\_\_\_\_\_\_\_\_

|  |  |  |  |
| --- | --- | --- | --- |
| **Exercise** | **Course outcome** | | **Grade** |
| Homework1 | | 2.a, 7.b | /30 |

2.a. Define engineering problems from specified needs for digital systems including implementation on FPGAs using HDL programming.

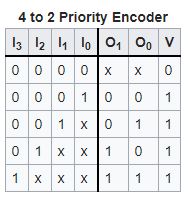
7.b. Employ appropriate learning strategies such as communicating with an expert, using external resources, experimentation, simulation, etc.

**Introduction:**

In this assignment, students were asked to create a new SystemVerilog project in Vivado to code a model of a 4-to-2-bit priority encoder. This includes creating a test bench to verify the encoder works as expected under every possible input.

**priorityEncoder Module:**

This module is setup with 3 ports: input [3:0]x and outputs [1:0]y and z. Using the truth table provided from the assignment (Figure 1). This was done with three lines corresponding to each bit of y and z.



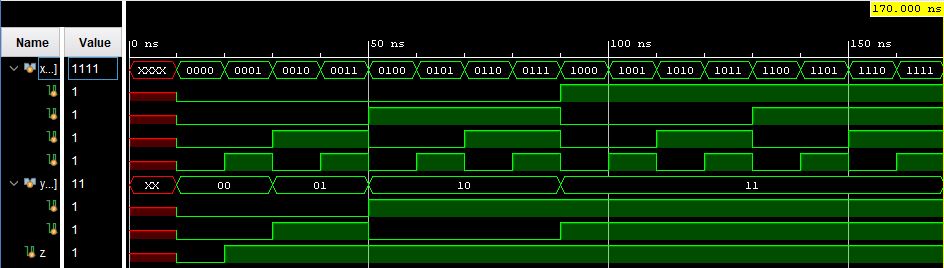
**Figure 1:** Truth table of a 4 to 2 Priority Encoder

After coding the module with the correct logic, the next step was to create a test bench simulation file. This file creates an instance of the priorityEncoder module and uses the same variables corresponding to the module to simulate it.

The simulation increments the input bus ([3:0]x) by one bit in binary by every 10ns until x=0b1111. This simulated every possible input the encoder can see to test and verify it matches the truth table.

**Results:**

After running the test bench simulation, the output shows that the encoder module works exactly as it should and matches the truth table (see Figure 2).



**Figure 2:** Output from test bench simulating the priorityEncoder module

As seen in the output, the valid bit (z) is 0 when x=0b0000 as it should and is 1 whenever x is not 0. When x=0b0001, y=0b01. When x=0b001X, y=0b01. When x=-b01XX, y=0b10. When x=0b1XXX, y=0b11. These results show the output is equal to what the truth table states.

**Conclusion:**

This assignment was a great way to test out the basics of creating a simple digital logic device in SystemVerilog with Vivado and to test it in a test bench written in Vivado to verify the module works correctly. These basic skills will be the building blocks of further assignments and are crucial to performing the basics of FPGA development in Vivado with SystemVerilog.