**ECE 4250/ 7250: VHDL and Programmable Logic Devices  
Laboratory**

**Lab #4   
Lab Title: SIMULATION THE DESIGN USING TEST BENCH**

**Group #2  
Group Names: Chris Smith, Benjarit Hotrabhavananda**

**Teaching Assistant Use Only:**

**Points Earned Reasons for Deduction**

**Pre-lab:**

**Post Lab report:**

**Demonstration:**

**Final Lab Grade:**

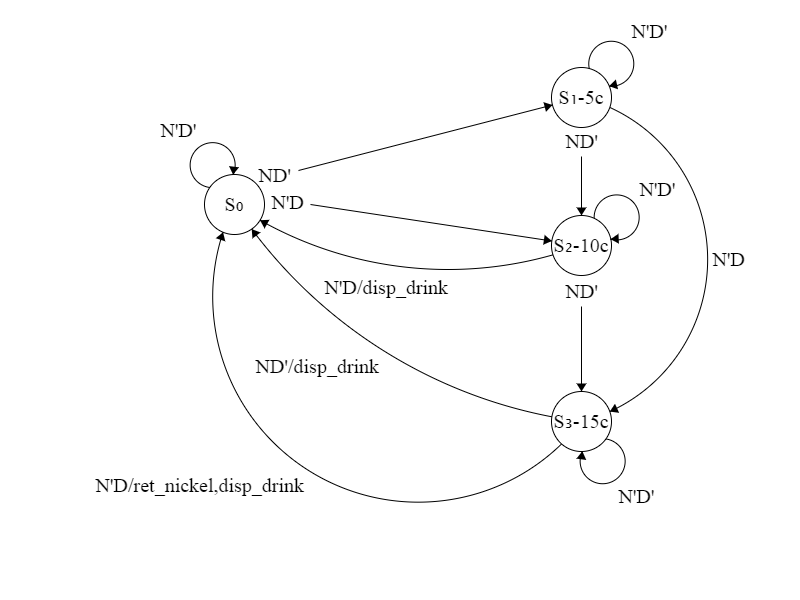
**Comments to students:**

1. **Objective:**

**To learn in this lab how to make a test bench for a soda vending machine. We used several different methods for implementing the vending machine, such as a behavioral model, and a ROM model. We also write a test bench to ensure that all combinations of inputs will lead to valid outputs.**

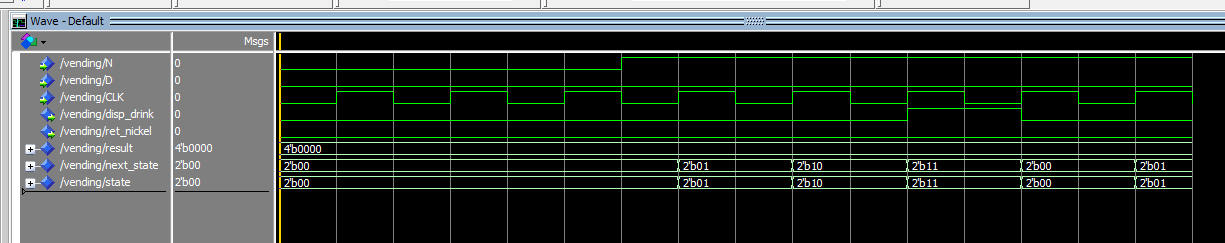
1. **Lab work:**

**Assert and report statements are useful for creation of test benches. A test bench is a program in VHDL that can provide input combinations to test a VHDL model for the system under test. The ROM model uses our state transition table to gather its inputs, then using array indexing, it finds the correct output. The behavioral model was much simpler to implement as we could simply assign our outputs using a series of case statements and other conditionals to find both the correct next state, and the correct output. The state transition diagram can be observed below.**

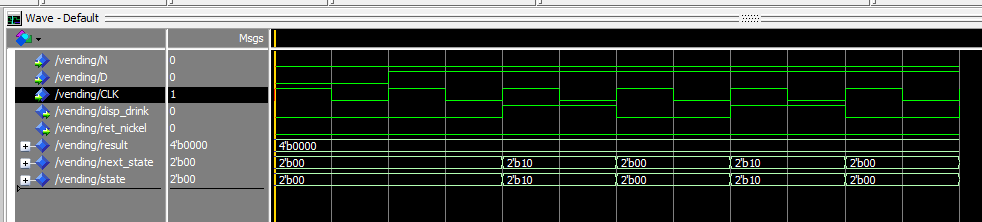
****

1. **Conclusion:**

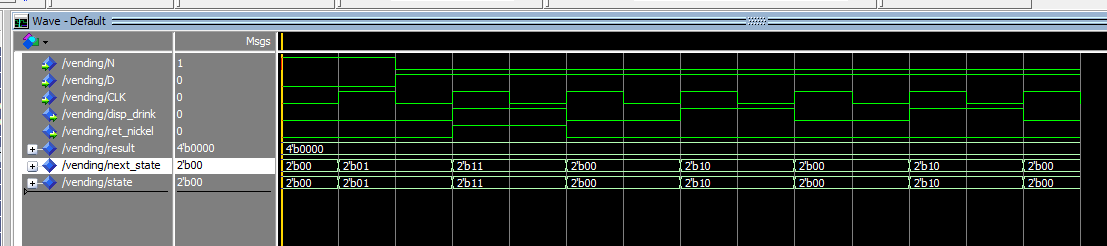
**In this lab, we designed a test Bench for soda vending machine to check whether the program work correctly or not, the test Bench will have the combination of inputs and outputs and will compare the results with the results which are resulted from the behavioral model and structural model, if any result is wrong a report will alert. We had some difficulties implementing the ROM model, as we had to define our own table to be checked for all inputs and outputs. We had several problems implementing our design, many of which were solved by carefully checking our data types and array indexing. The last issue was the disp\_drink and ret\_nickel not updating when they should, we quickly learned to move our output assignments outside of the process, and use intermediate signals within the process.**

****

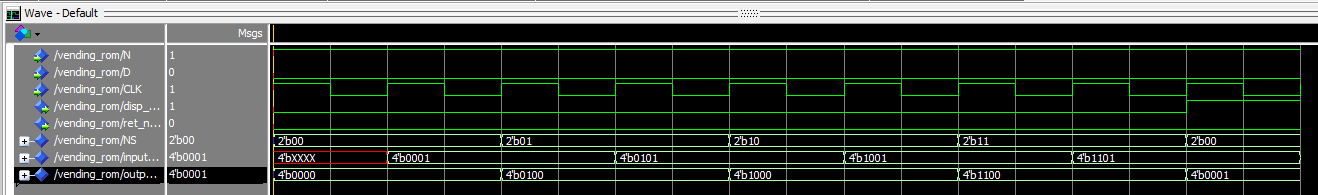
**Figure: Behavioral model waveform for Nickels only**

****

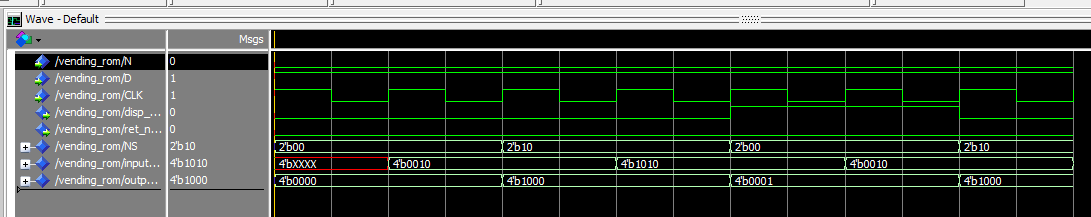
**Figure: Behavioral model waveform for Dimes only**

****

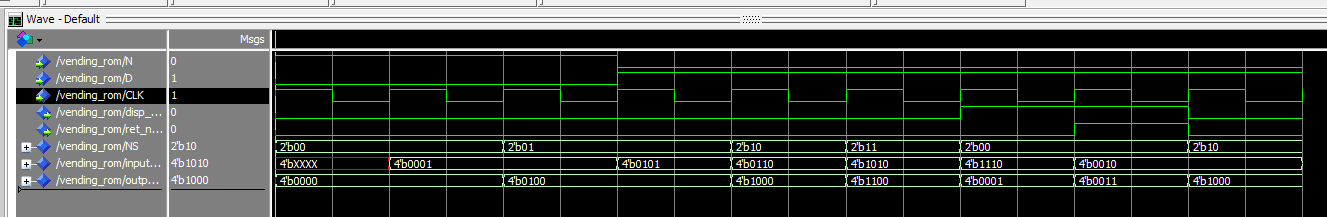
**Figure: Behavioral model waveform for nickel, then all dimes (demonstrates the return nickel)**

****

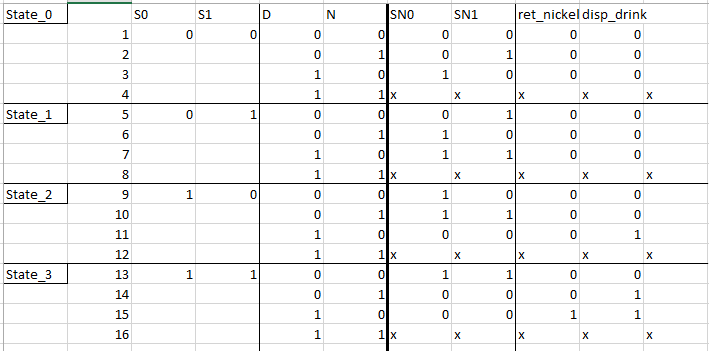
**Figure: ROM model waveform for nickels only**

****

**Figure: ROM model waveform for dimes only**

****

**Figure: ROM model waveform for nickel, then all dimes (demonstrates the return nickel)**

****

**Figure: State transition table used to implement the behavioral model, as well as the ROM implementation**