School of Engineering

Grand Valley State University

EGR 224 – Final Project Professor Ekin

Vending Machine

By

Chase Mulder

Table of Contents

Objectives 3

Introduction 3

Fund Maintenance 4

* Figure 1: Fund Maintenance Schematic
* Figure 2: Fund Maintenance STD
* Figure 3: Fund Maintenance Simulation
* Figure 4: Fund Maintenance Test Bench

Coin Return 5

* Figure 4: Coin Return Simulation
* Figure 5: Coin Return STD

Purchase Selection 7

* Figure 6: Purchase Selection Simulation
* Figure 7: Purchase Selection STD

Seven Segment 8

* Figure 9: Seven Segment Decoder Simulation
* Figure 10: Seven Segment Decoder Verilog Task

Vending Machine 9

1. Figure 11: Vending Machine Schematic
2. Figure 12: Vending Machine I/O
3. Figure 13: Vending Machine Simulation

Conclusion 10

Objectives

* Design a vending machine FSM with inputs of different coins and outputs different coins and items as a normal vending machine would
* Fully simulate the vending machine walking through each step and all possible combinations of coins entered, items selected, and resetting
* End goal is that the design could be placed on the Basys3 board

Introduction

The Vending Machine will be capable of taking three different coin inputs: a Nickel, Dime, and a Quarter. Three items will be available for user selection and dispensing: Gum ($0.15), Apple ($0.25), and Pretzels ($0.30). A “Return Change” option should be available to the user to cancel their attempt and return their funds. The 7-Segment LED Displays and on-board LED’s will be used for user feedback. While usage of the Basys3 board will not be required, you should develop your system under the assumption that it could be placed on the board.

Your design should incorporate the following inputs:

• Switch-0: Nickel input

• Switch-1: Dime input

• Switch-2: Quarter input

• BTNL: Gum selection

• BTNC: Apple selection

• BTNR: Pretzel selection

• BTNU: System Reset

• BTND: Change Return

Your design should utilize the following outputs:

• 7-Segment LED Leftmost 2 Displays: Current amount deposited

• 7-Segment LED Rightmost 2 Displays: User Selection/Error Feedback

• LED0: Nickel Return (change request)

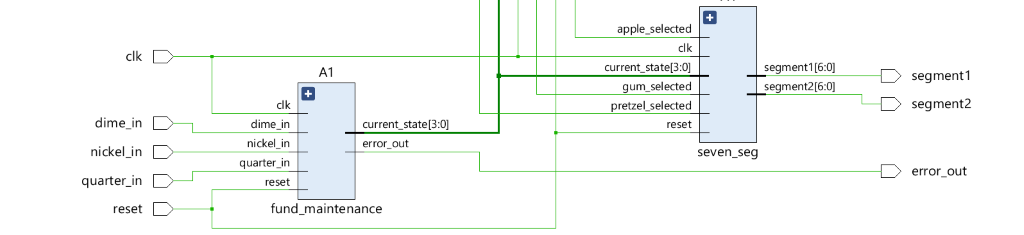
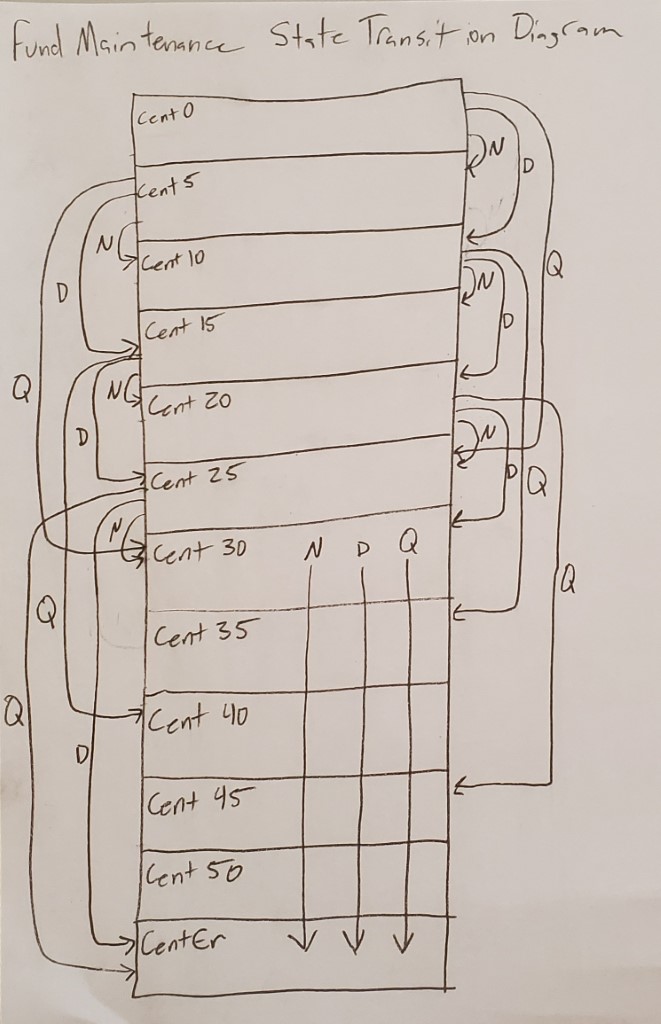
Fund Maintenance

Figure 1: Fund Maintenance Schematic

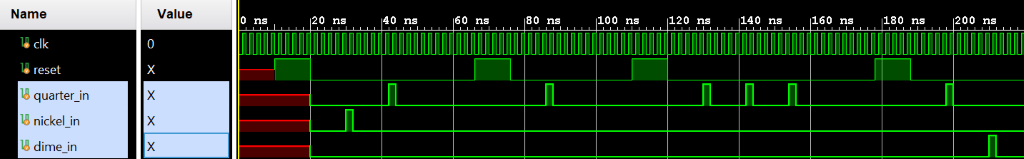
Figure 2: Fund Maintenance STD



Description of Fund Maintenance

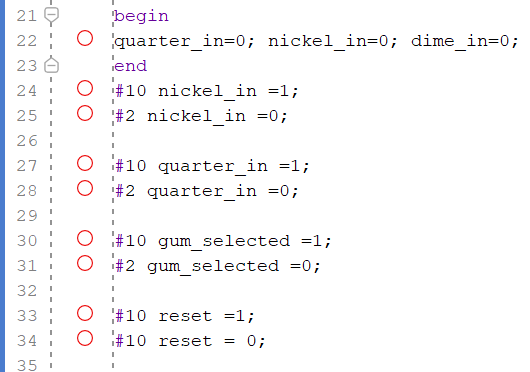
Fund maintenance keeps track of the state the user is in depending on how many coins they entered. The states have been labeled according to the number of cents the user has; cent0, cent5, cent10… Once the user has reached cent 30, if the user attempts to enter another coin it will result in a coin error and will display “Lo” on the seven-segment display. Below is the simulation showing the fund maintenance module working.

Figure 3: Fund Maintenance Simulation



In this simulation there are four tests, one is nickel quarter, two is quarter, three is quarter quarter quarter, and fourth is quarter dime. This simulation shows that the module is correctly inputting the coins. Below is the testbench used to simulate the fund maintenance module.

Figure 4: Fund Maintenance Test Bench

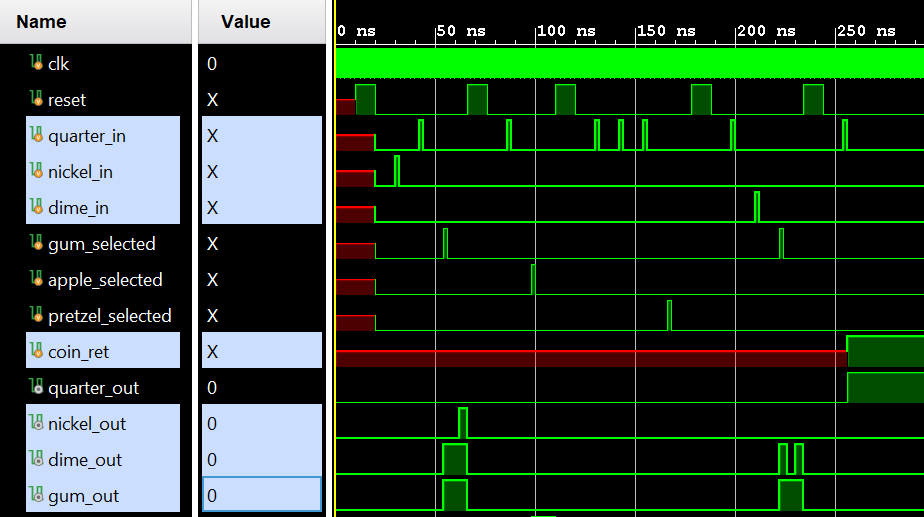


In the testbench we set all initial coin values to 0 and then we use the #10 and #2 commands to tell the system to wait 10 time units, then turn the coin on, then wait 2 time units, then turn the coin off. Then the reset is set, so the module starts over by resetting the initial values.

Coin Return

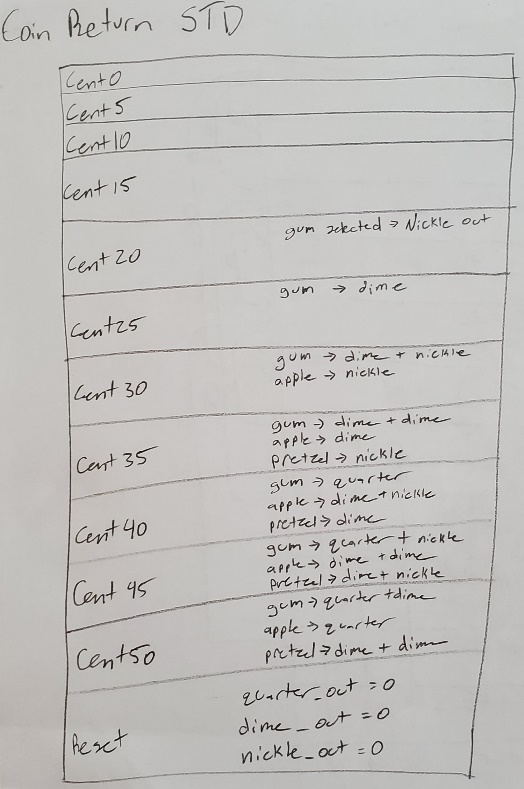
Coin return is a module that takes the input of the current state and the user selected item and calculates how much coins to return. The simulation will show the output of the coins separated by 4 clock cycles.

Figure 4: Coin Return Simulation



This simulation shows 5 tests to test the coin return functionality. The key thing to note about the coin return is that it’s been fabricated to make extra profits for the vending machine owner if the user’s total is 30 cents, or more and the user inputs coins. This is great for the vending machine owner, but not great for the hangry user who doesn’t bother to read the sign on the vending machine that says, “Vending Machine coin capacity capped at 30 cents”. The simulation above shows in test one the inputs of a quarter, +25, a nickel, +5, and the user selecting the gum option. The output at 50ns shows the gum being outputted and a dime and a nickel being returned to the user because they had 30 cents, spent 15 cents, and get 15 cents back. Test 5 at 250ns shows the coin return button being pressed when the user has input a quarter, so the machine outputs a quarter. Below is the state transition diagram of the coin return module.

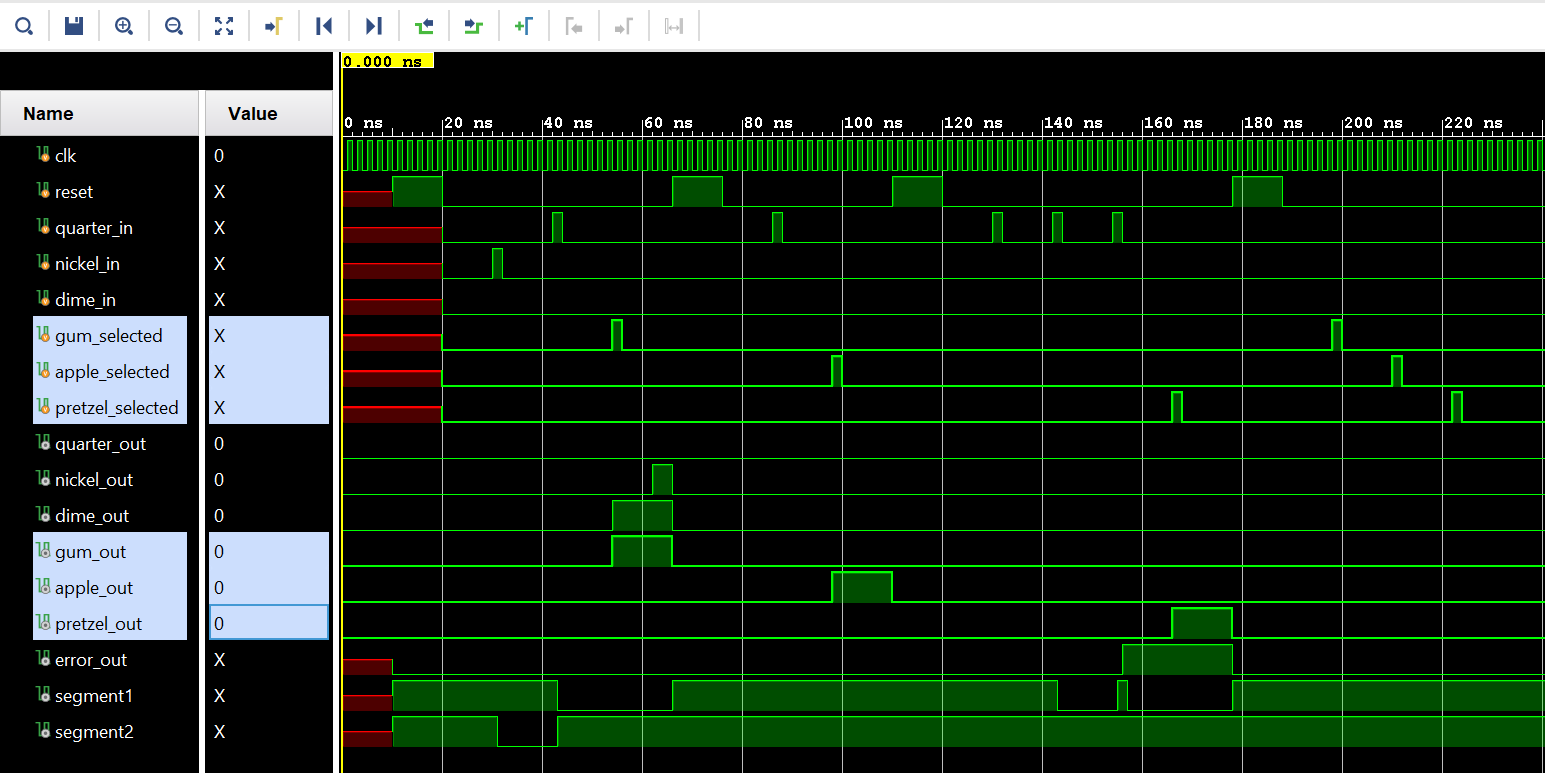
Figure 5: Coin Return STD



Purchase Selection

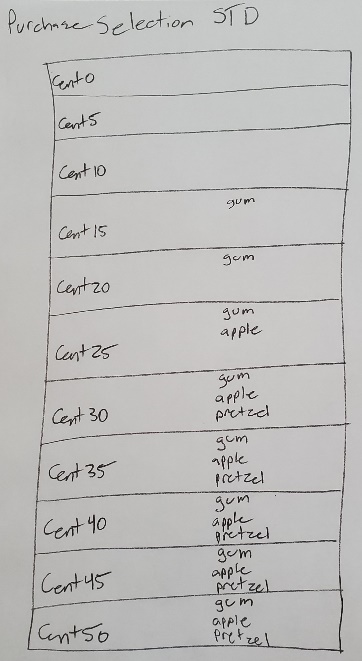
Purchase selection is a module that handles if the vending machine outputs an item based on the coin input. The simulation below tests the selections of gum, apple, and pretzel with proper funds and then with improper funds.

Figure 6: Purchase Selection Simulation



As you can see in the simulation above, the selection of gum, apple, and pretzel without funds did not output the item when the user did not input any coins. The selections when the used did have proper funds did output the right item and coin return handled the output of the change. Below is the STD.

Figure 7: Purchase Selection STD



Seven Segment Decoder

The seven-segment decoder module will take the input from fund maintenance of the current state and will also take the input of purchase selection if an item has been selected. The seven-segment decoder will display the total amount of funds on two of the seven segment displays and will also display the item in two letters if the item is output. The seven-segment display is made up of displaying the numbers 1 through 5 and the letters ‘G’, ‘u’, ‘A’, ‘P’, ‘r’, ‘L’ and ‘o’. Below is the simulation showing that the different states will produce the intended output of the right segments being on to make up the number or letter and the Verilog task verifying that the simulation is working taking inputs and what it’s outputting.

Figure 9: Seven Segment Decoder Simulation

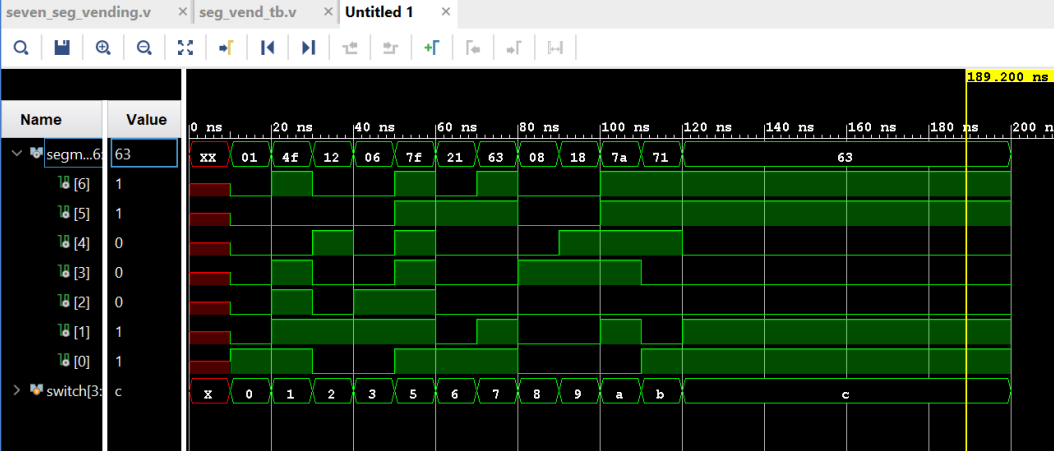
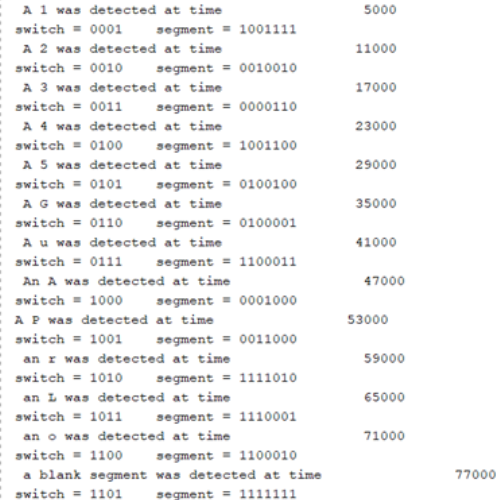


Figure 10: Seven Segment Decoder Verilog Task



Final Vending Machine

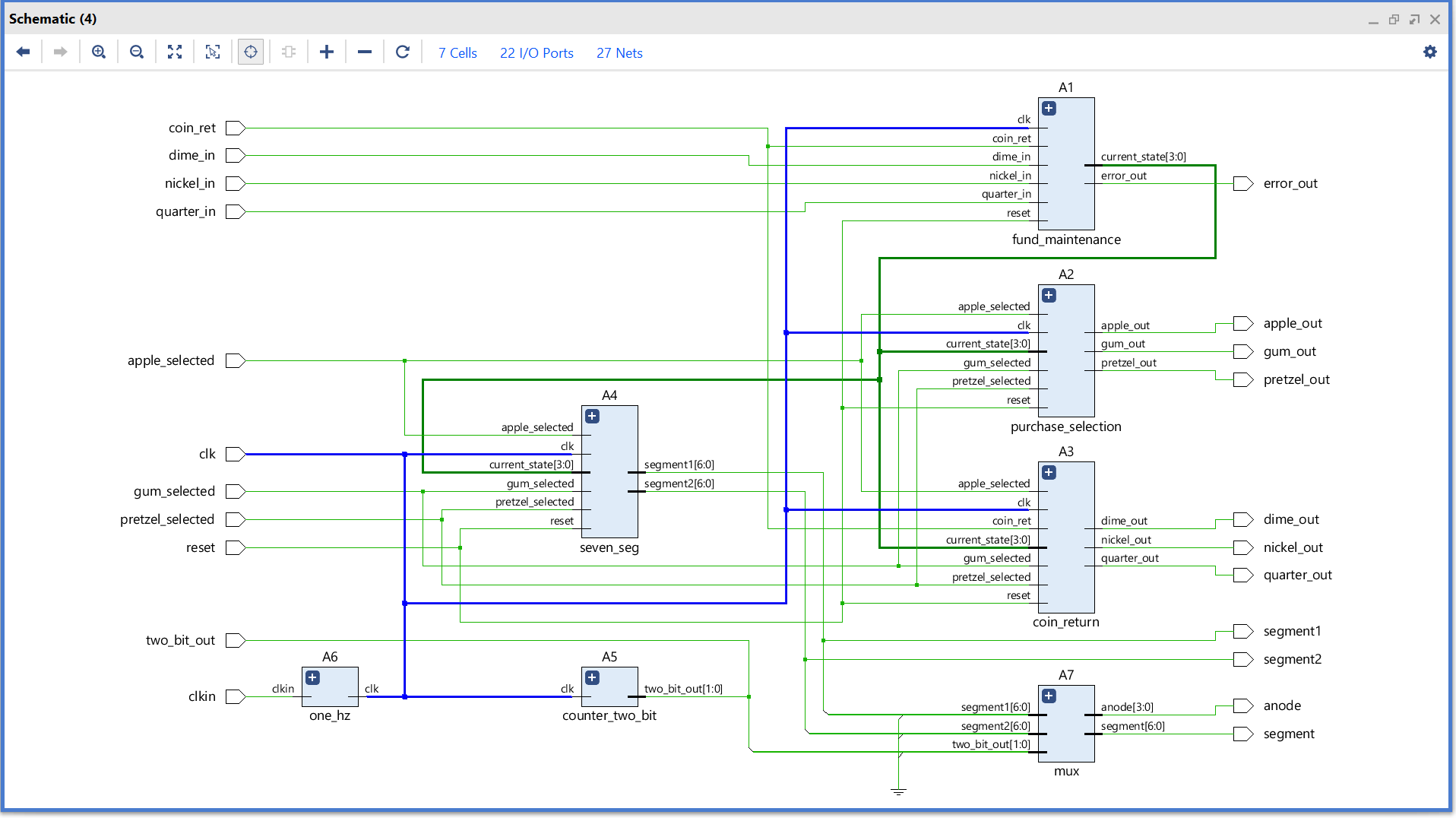
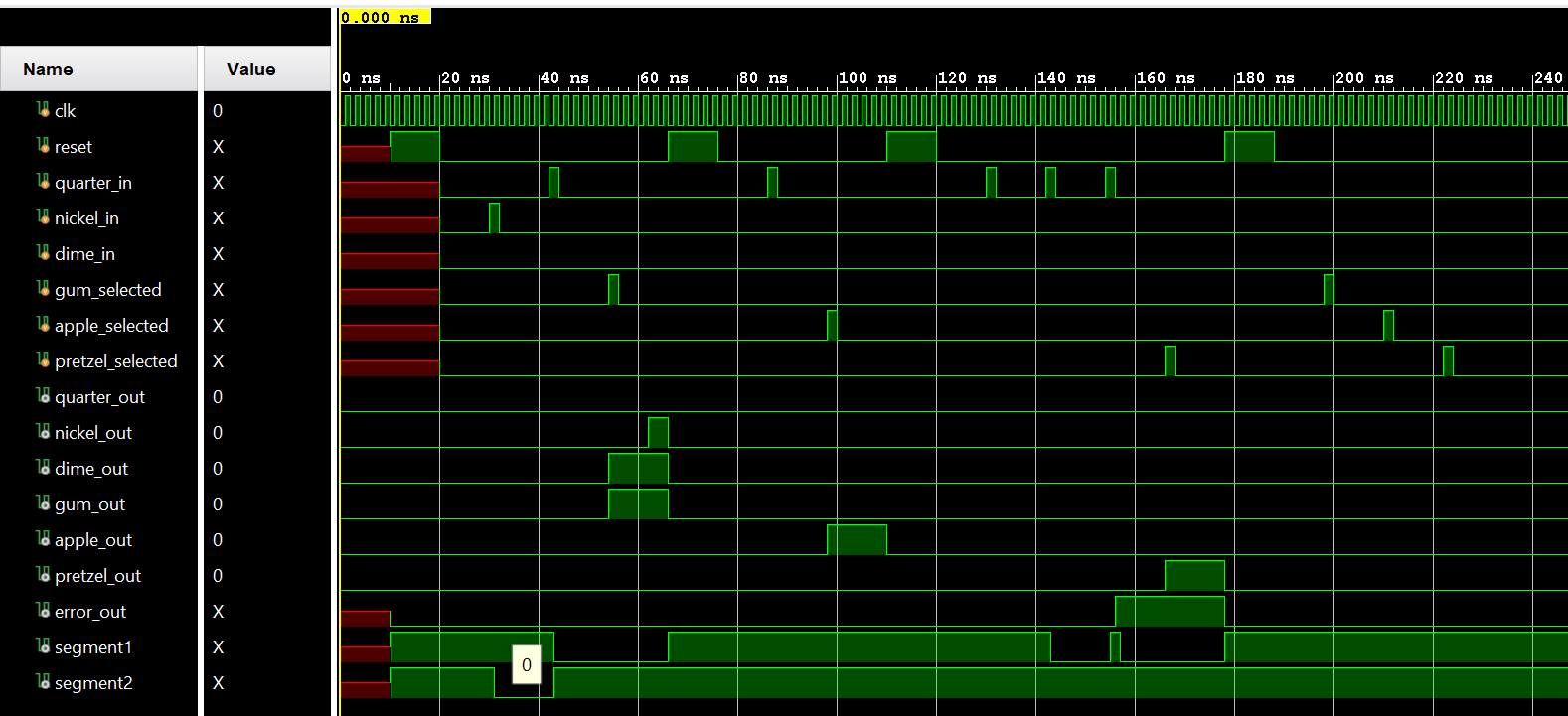
Figure 11: Vending Machine Schematic

Figure 12: Vending Machine I/O

|  |  |
| --- | --- |
| Inputs | Outputs |
| Dime | Dime |
| Nickle | Nickle |
| Quarter | Quarter |
| Coin return | Gum |
| Gum | Apple |
| Apple | Pretzel |
| Pretzel | Error |
| Reset | 2 Seven segment displays |

Figure 13: Vending Machine Simulation

Conclusion

If there were more time to design this vending machine, I’d make my fund maintenance a register with more bits, so the user could input more than 30 cents that the vending machine currently caps the input at. Also, I’d use all four seven segment displays, so the user could input more that 100 cents and I’d update the seven-segment display to display more of the word of the item the user is buying. Finally, I’d update my return coin module to not be greedy and reset the user’s fund total back to 30 cents, so it wouldn’t be capped.

Hardships that came up during this project were how to route wires to other modules and have that module input that wire. This was all solved by opening the elaborated design feature Vivado provides and using the generated schematic.