Stephanie Doan (604981556)
Rohan Surve
CS M152A Lab 6
28 February 2021

Lab 3: Vending Machine

I. Introduction

In this lab, I design a classic vending machine as a finite state machine in Verilog HDL, to explore how these machines can model a common real-world system. I work with a larger set of states, transitions, inputs and outputs than seen before in this class, so it is necessary to document and plan my vending machine module.

Inputs to vending machine:

Input	Size	Behavior
CLK	1 bit	System clock
RESET	1 bit	Synchronous reset. Makes outputs and item counters go to 0, go to IDLE state.
RELOAD	1 bit	Reload machine by setting item counters to 10.
CARD_IN	1 bit	Stays high as long as card is inserted.
ITEM_CODE [3:0]	4 bits	Item code to select snack, which is two digits inputted one at a time.
KEY_PRESS	1 bit	Value in ITEM_CODE is read when this signal is high.
VALID_TRAN	1 bit	Transaction using card is valid.
DOOR_OPEN	1 bit	Vending machine door is open.

Outputs of vending machine:

Output	Size	Behavior
VEND	1 bit	High when transaction is deemed valid. Low when DOOR_OPEN goes high then low or if door does not open in 5 clock cycles.
INVALID_SEL	1 bit	HIGH if (1) Only 1 digit of ITEM_CODE is entered and there is no 2nd digit after 5 clock cycles or no digit is entered for clock cycles (2) Two-digit ITEM_CODE is invalid (3) Counter fo one of items is 0.
COST [2:0]	3 bits	Set to cost of item when code is entered and remains until new transaction begins.

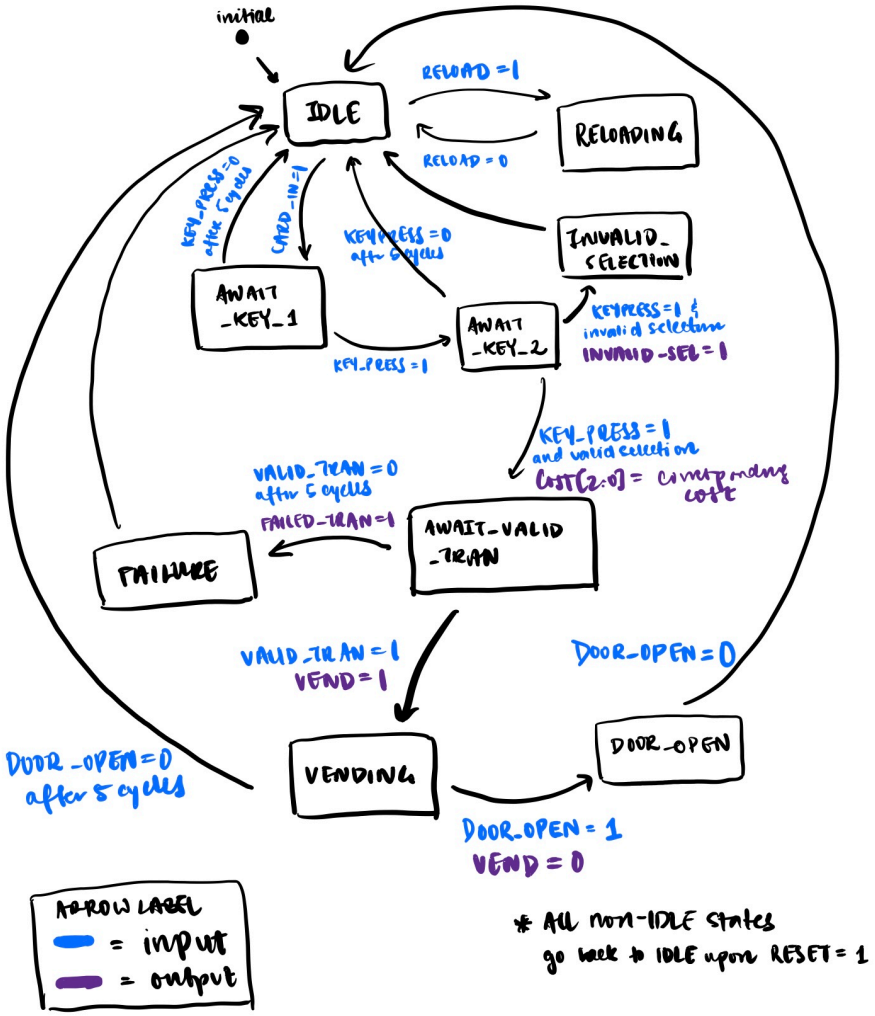
Output	Size	Behavior
FAILED_TRAN	1 bit	Set to 1 if VALID_TRAN signal does not go to high within 5 clock cycles of determining ITEM_CODE

The machine dispenses 20 snacks labeled with selection codes from 00 to 19 inclusive. Each snack is reloaded with 10 of its items, and only one transaction may occur at a time via credit card insertion. Prices corresponding to the snacks are outlined in the following table:

Item Code	Cost ($)
00, 01, 02, 03	1
04, 05, 06, 07	2
08, 09, 10, 11	3
12, 13, 14, 15	4
16, 17	5
18, 19	6

II. Design Description (FSM diagram)

The first step of writing the vending_machine module was mapping out the states and transitions of the FSM. My main state diagram is shown below, with the inputs and outputs color coded accompanying each arrowed transition between the boxed states. Note that all non-IDLE states go back to IDLE when the RESET input signal is 1. More detail about the states will follow.



My chosen **states** are outlined as follows:

1. [4'b0000] **IDLE**
   - Outputs are set to 0
   - Waits for CARD_IN signal to go to AWAIT_KEY_1 state or RELOAD signal to go to RELOADING state
2. [4'b0001] **RELOADING**
   - Snack counters are set to 10, indicating machine is fully reloaded
   - Can only be reached from the IDLE state
   - No transaction may begin in this state

3. [4'b0010] **AWAIT_KEY_1**
    - Waits for KEY_PRESS signal to be 1 to go to AWAIT_KEY_2 state, during which value in ITEM_CODE input is stored
    - Goes back to IDLE state if KEY_PRESS signal does not go to 1 within 5 clock cycles
4. [4'b0011] **AWAIT_KEY_2**
    - Waits for KEY_PRESS signal to be 1, to check if inputted digits are valid (within range of 00 to 19) and the value at that snack counter is greater than 0 (snack is still stocked and can be dispensed). If valid, goes to AWAIT_VALID_TRAN during which COST output is set
    - Goes to IDLE if KEY_PRESS signal does not go to 1 within 5 clock cycles
    - Goes to INVALID_SELECTION if invalid digits are inputted or the snack is out of stock
5. [4'b0100] **INVALID_SELECTION**
    - Reached from AWAIT_KEY_2 if selection is invalid
    - Immediately goes to IDLE after one clock cycle
6. [4'b0101] **AWAIT_VALID_TRAN**
    - Waits for VALID_TRAN signal to go to 1, to go to VENDING state while setting VEND output to 1
    - Goes to FAILURE if VALID_TRAN signal does not go to 1 after 5 clock cycles, sets FAILED_TRAN output to 1
7. [4'b0110] **FAILURE**
    - Reached from AWAIT_VALID_TRAN when validation from bank via VALID_TRAN input signal is not provided within 5 clock cycles
    - Immediately goes to IDLE after 1 clock cycle
8. [4'b0111] **VENDING**
    - Waits for DOOR_OPEN signal to go to 1 to transition to DOOR_OPENED state, while setting VEND output to 1
    - Goes to IDLE if DOOR_OPEN does not go to 1 within 5 clock cycles
9. [4'b1000] **DOOR_OPENED**
    - Waits for DOOR_OPEN to go to 0 to transition back to IDLE

At a high level, my **design** for this vending_machine module is based on the separation of updates to the current_state, next_state, and output values in three distinct always blocks, as done in the example in the spec. I had a set of current *state_counters* (an array of 20 three-bit registers) to hold the count of each snack item as a vending machine would hold, as well as a *current_state* register holding one of nine values representing each state. I also had an array of *state_counters_next* register and a *next_state* register as copies of the previous data structures, to be updated in the always blocks that are not to update current state. Then, these "next" values are written to the current registers in the first always block.

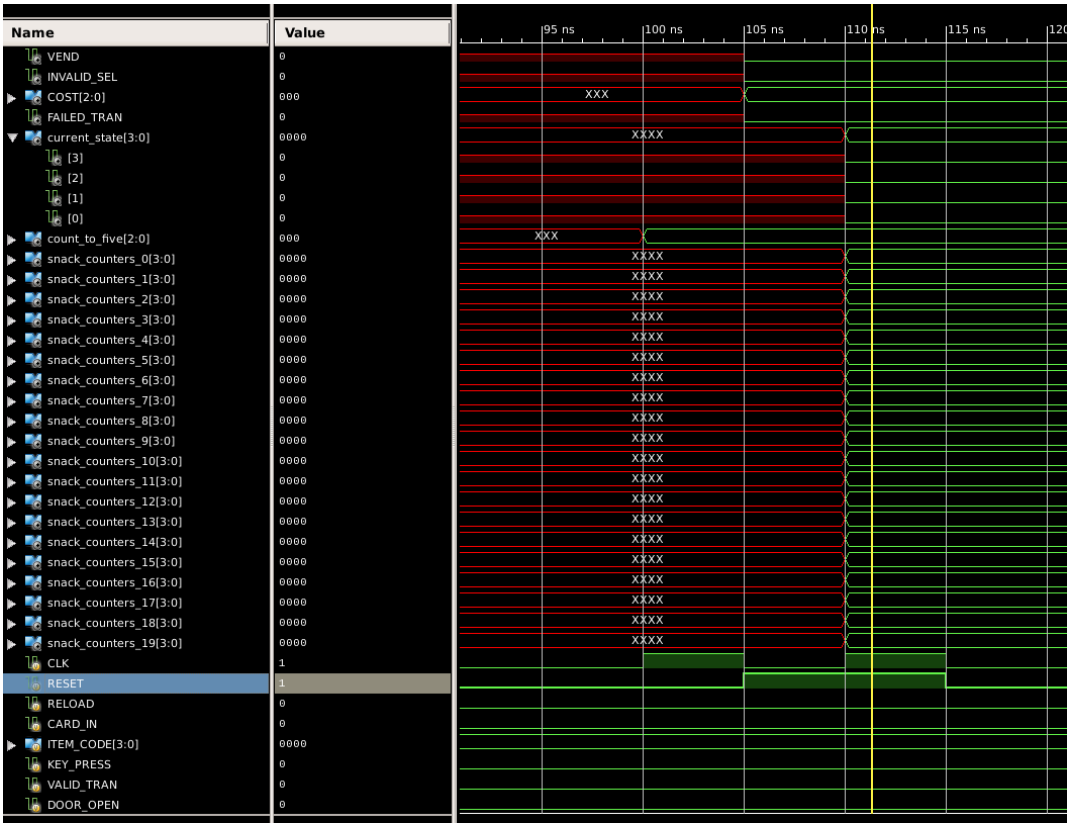
In addition, I assumed that KEY_PRESS must go to high and then low and then high again to count as two separate digits entered, as that is how vending machines operate in real life. I use a *hold* variable to keep track of this sequence of events.

To count to 5 when waiting for specific inputs in the AWAIT_KEY_1, AWAIT_KEY_2, and AWAIT_VALID_TRAN states, I kept a counter called *count_to_five* to be incremented and checked for the passing of 5 clock cycles.

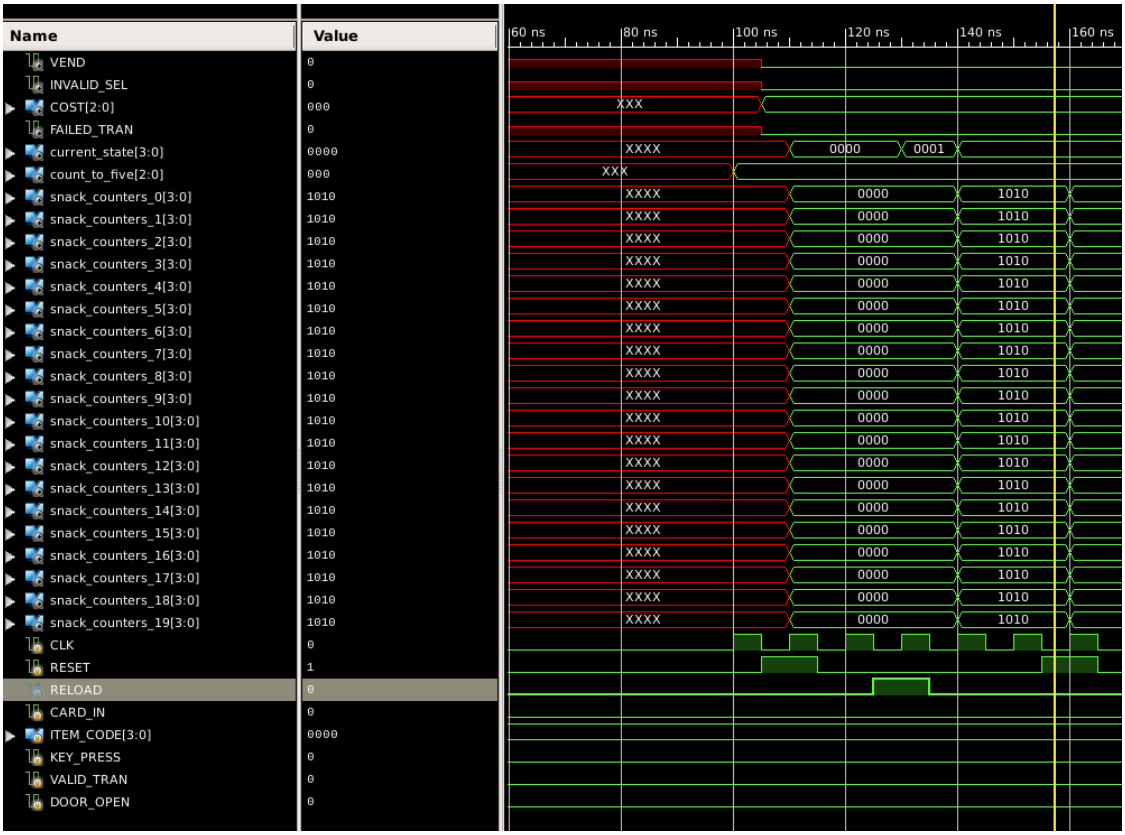
III. Simulation tests

In order to verify the correctness of my vending_machine module based on my understanding of the problem, I wrote test cases in my test bench, with input signals set at every negative clock edge of CLK, to make sure inputs are stable at the positive clock edge, which is what the vending_machine module acts on. Note that I also make my snack counters, *current_state*, and *count_to_five* output variables for the sake of testing and displaying my module's internal state, but the final code does not include these extra outputs.

1. Reset — We see here that having the RESET input be 1 makes all the snack_counters_x and outputs VEND, INVALID_SEL, COST, FAILED_TRAN hold the value 0, as expected. We also see that current_state is at 0 = 4'b0000 for IDLE state.



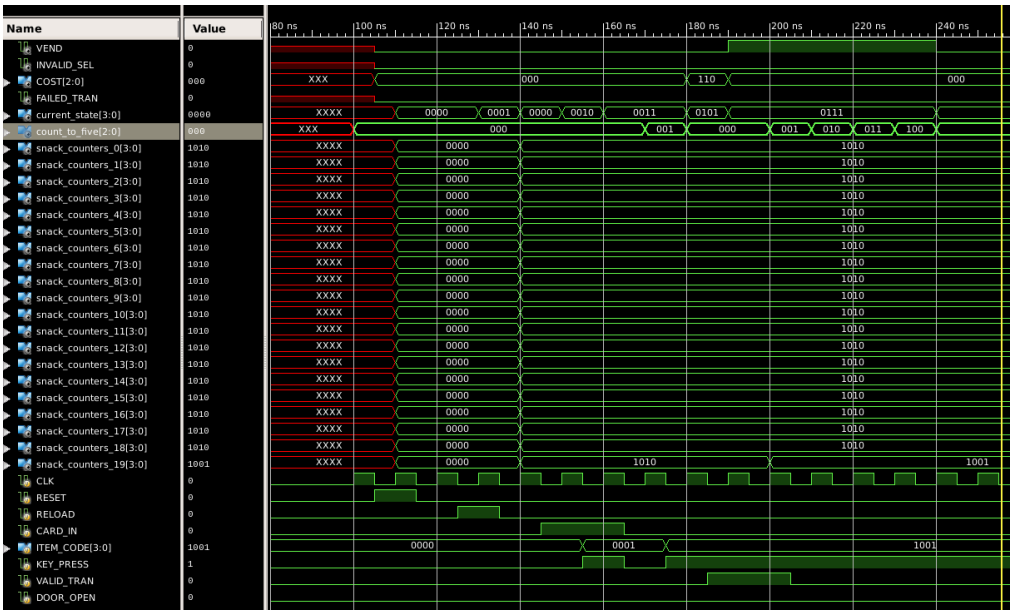
2. Reload — We see that the RELOAD input signal being set to 1 for one clock cycle results in the snack_counters to all hold the value 10 (fully restocked), as well as current_state moving to 4'b0001 = RELOADING state as well.



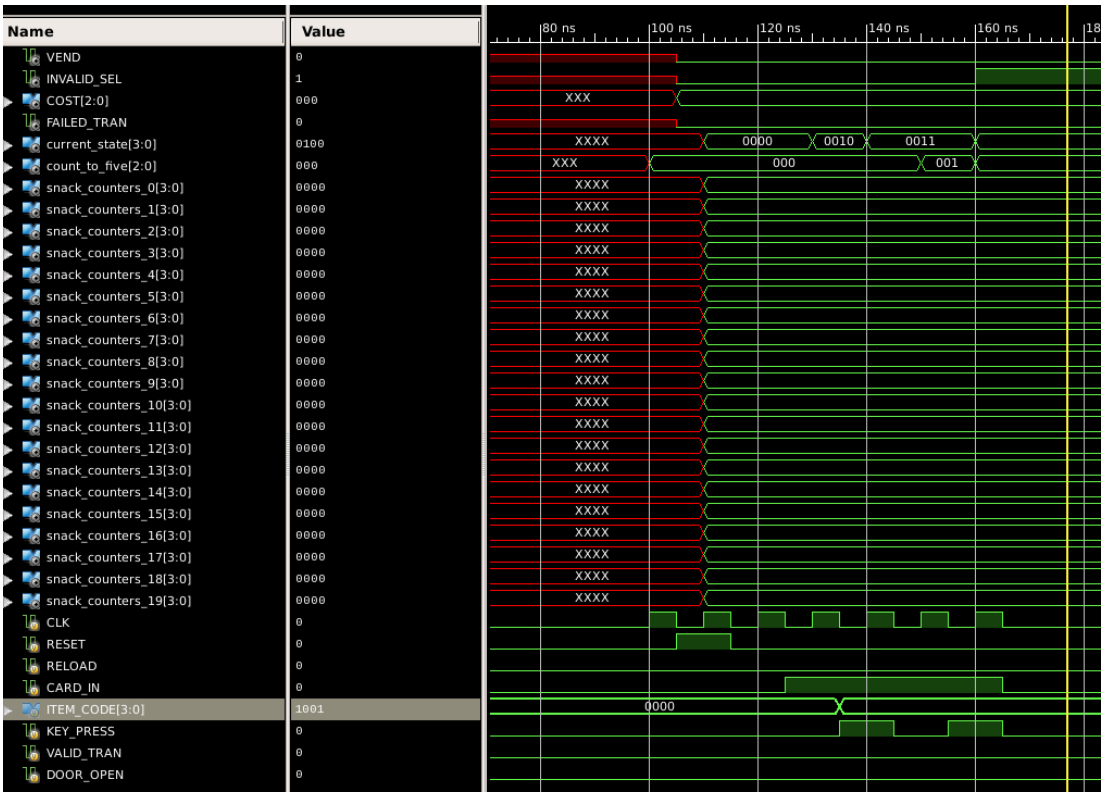
3. Timeout after no keypress in AWAIT_KEY_1 state — Here we see the current_state in 4'b0010 as AWAIT_KEY_1, as count_to_five begins to increment in value from 0 to 4 before being reset to 0 and the current_state holding 4'b0000 = IDLE again. This is what happens when the credit card is inserted without being followed with a timely key press.

4. Timeout after no keypress in AWAIT_KEY_2 state — Here we see the current state holding value 4'b0011 for AWAIT_KEY_2 and count_to_five being incremented to 4 and then reset to 0 when the 5 clock cycle limit is reached. The current_state goes back to 4'b0000 for IDLE if KEY_PRESS is not set to 1 within those five clock cycles.

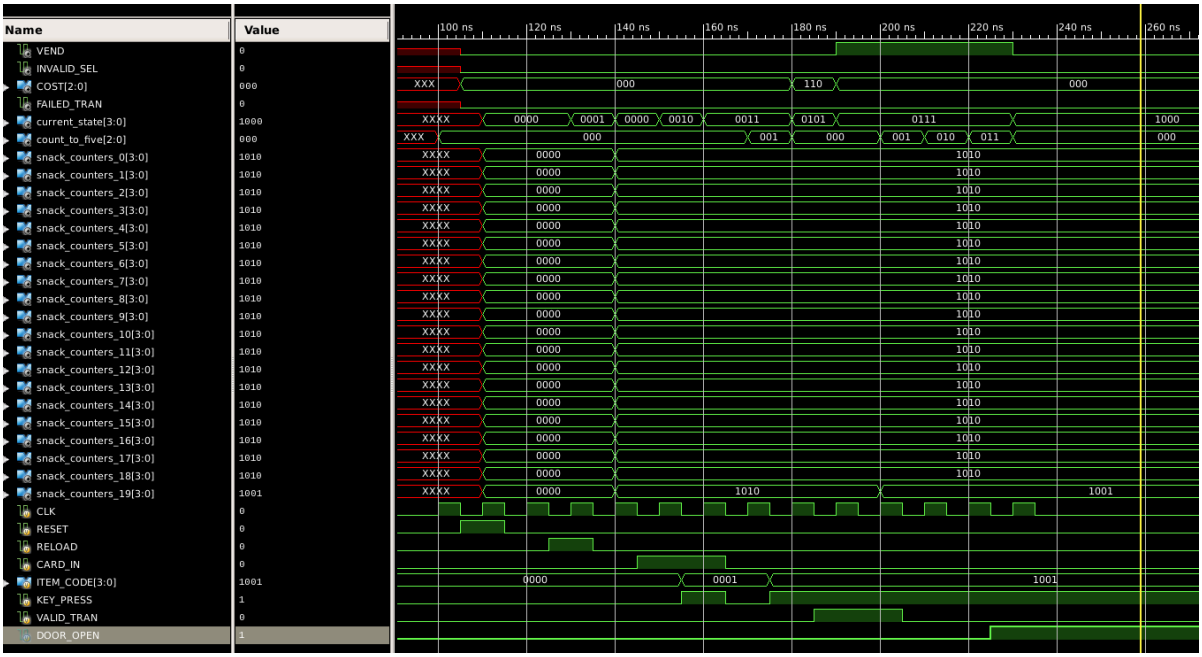
5. Timeout after door is not open in VENDING state — Here we see the current_state all the way at 4'b0111 = VENDING after going through the necessary transitions and states of a transaction. Here the count_to_five counter is start to wait for DOOR_OPEN to be set to 1, which never happens. Therefore, current_state goes back to 4'b0000 = IDLE after the five clock cycle limit is reached.



6. Invalid selection — An incorrect selection value of 99 is inputted to the vending machine via the ITEM_CODE input. After the KEY_PRESS signal goes from high to low to high again for the vending machine to read in these two digits, the current_state goes from AWAIT_KEY_2 to INVALID_SELECTION and the INVALID_SEL output is set to 1.

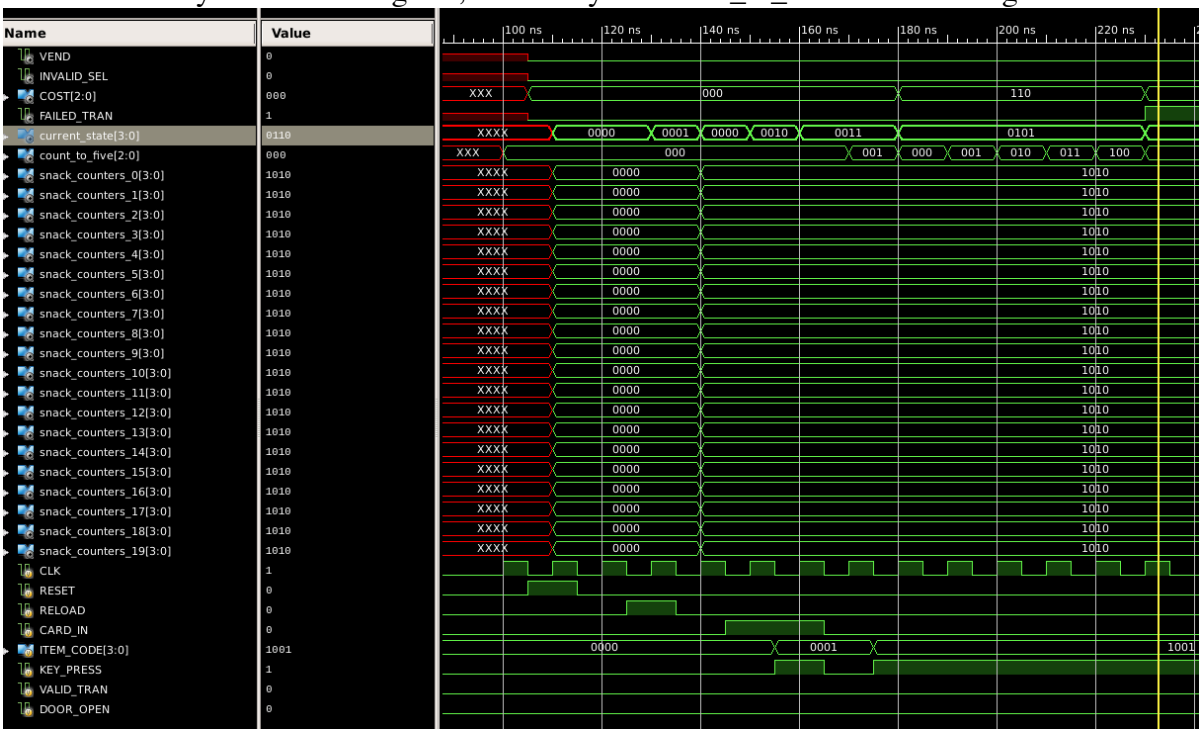


7. Door stays open — As seen, the current_state stays at 4'b1000 = DOOR_OPENED as long as the DOOR_OPEN signal stays high.

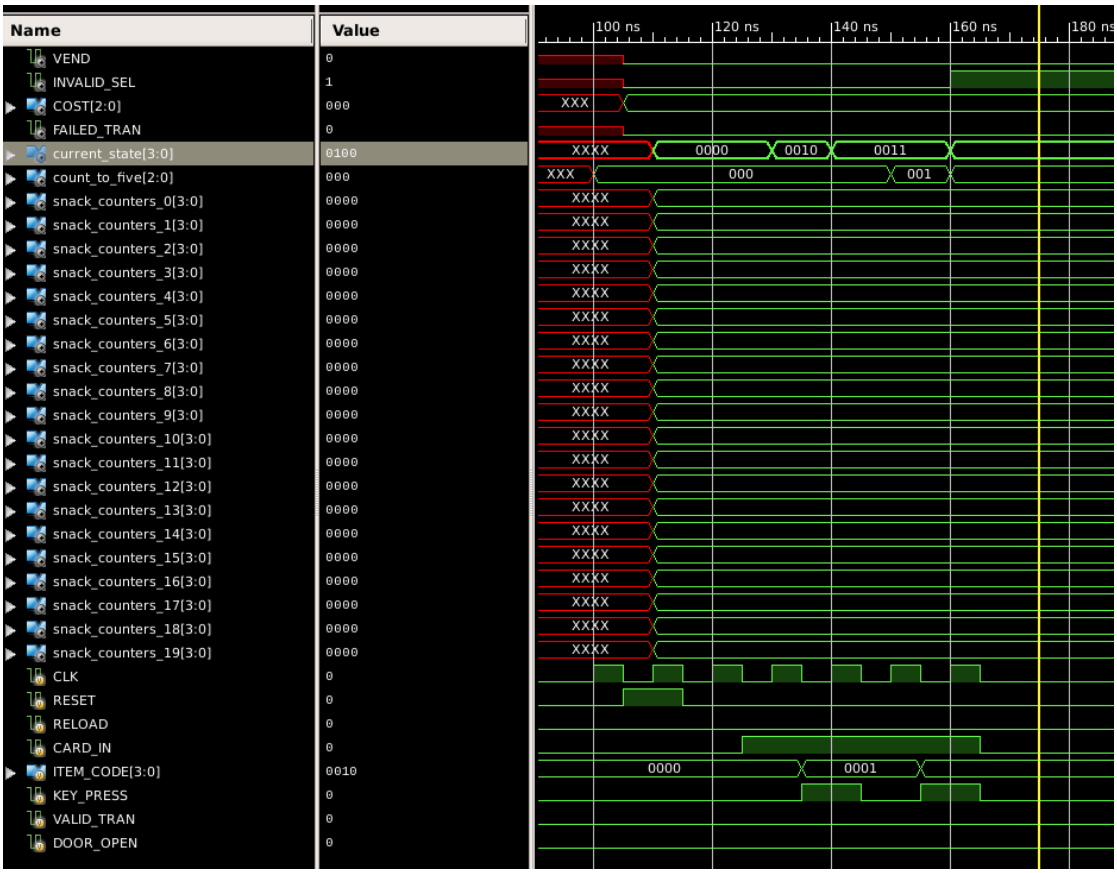


8. Invalid transaction due to absence of VALID_TRAN high signal — As seen, the current_state goes to 4'b0110 = FAILURE after being in 4'b0101 = AWAIT_VALID_TRAN

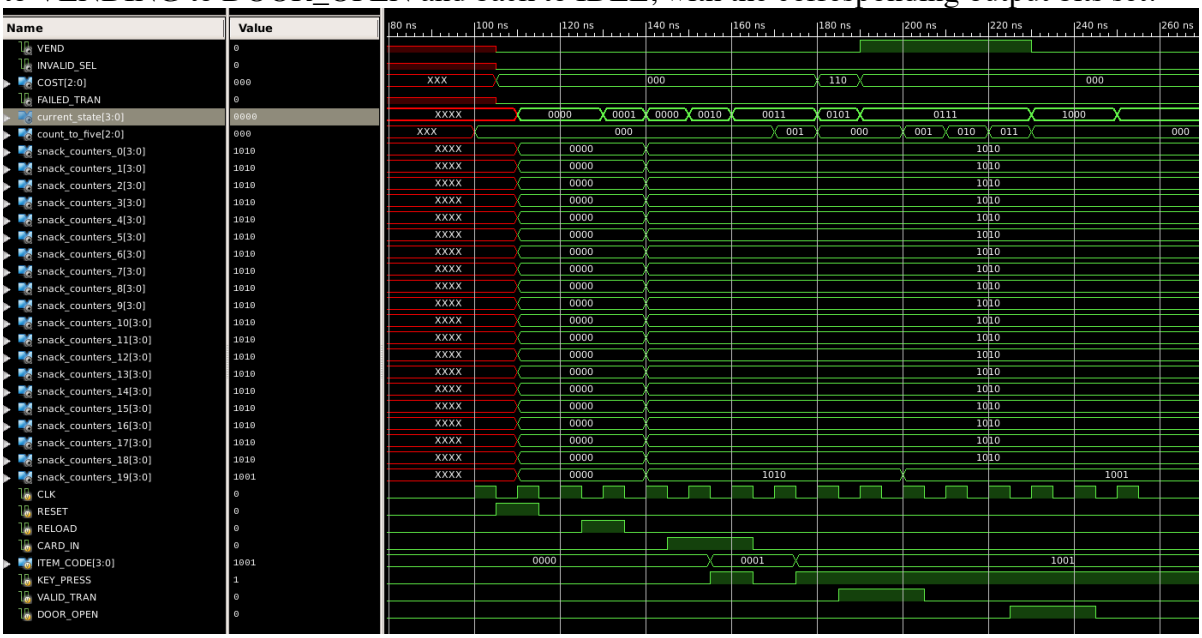
for five clock cycles and timing out, as seen by the count_to_five counter being incremented.



9. Snack stock is empty — Without reloading the snack counter, the counter value of each snack remains at 0, so any attempt at purchasing an item would result in being in the INVALID_SELECTION state with INVALID_SEL output being set to 1.

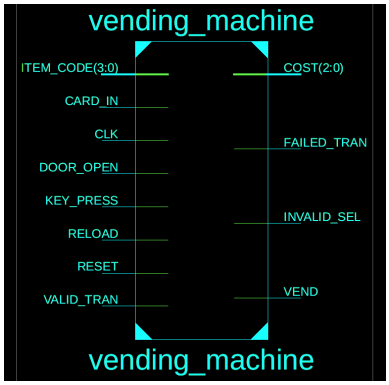


10. Successful vend — This is a smooth transaction with valid selection values, stocked snacks, transaction validation from the bank, and door open and close. We see the transition of current_state from IDLE to AWAIT_KEY_1 to AWAIT_KEY_2 to AWAIT_VALID_TRAN to VENDING to DOOR_OPEN and back to IDLE, with the corresponding output bits set.

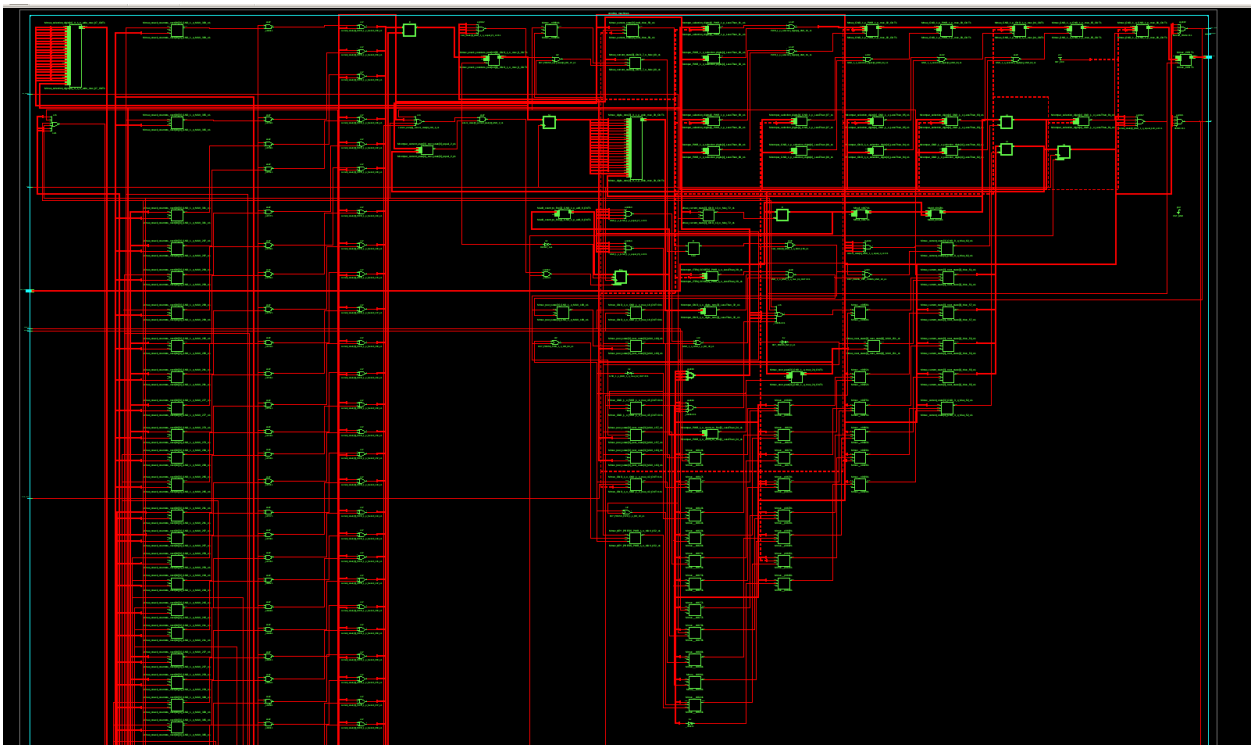


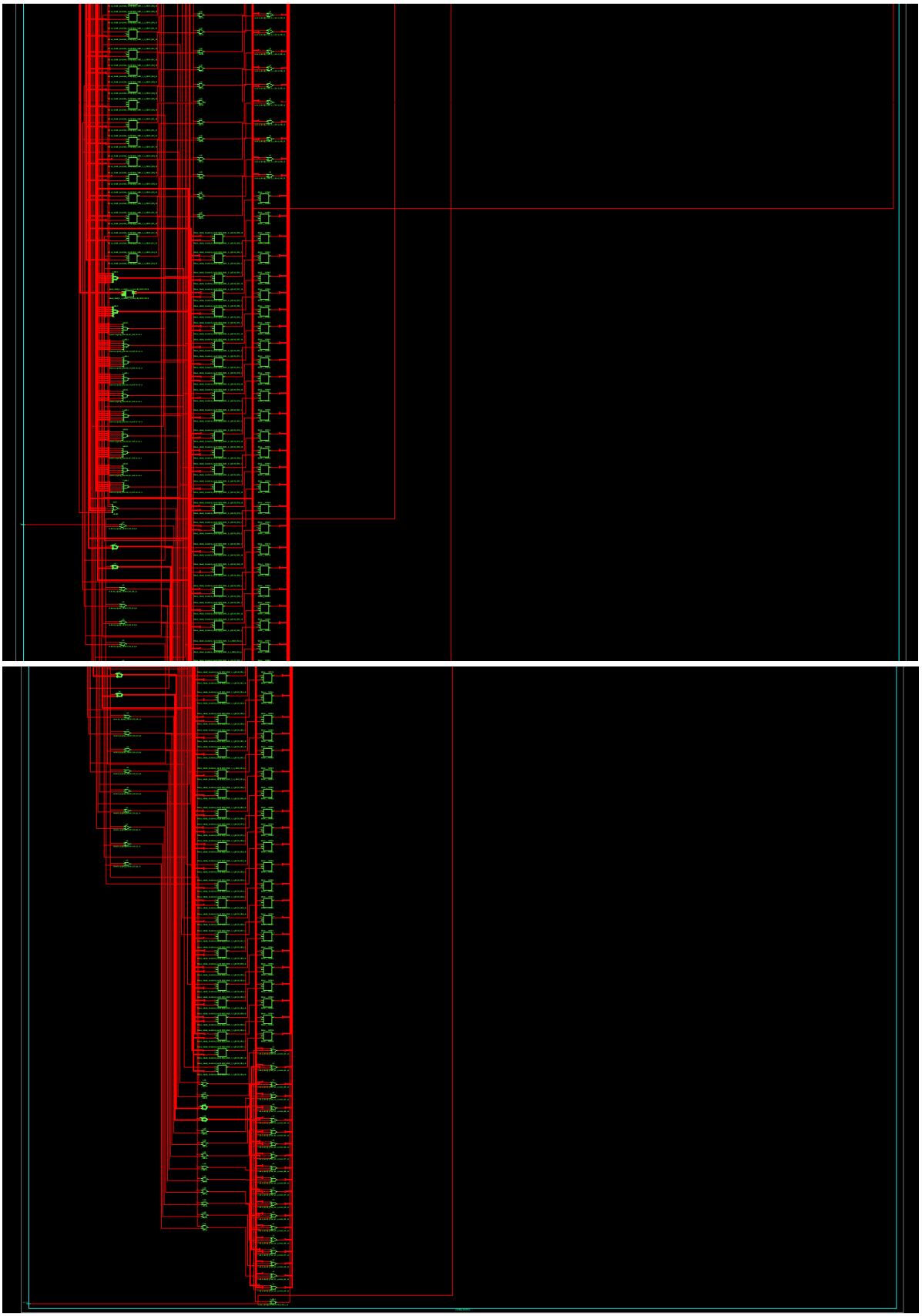
IV.**Schematics**

The I/O of the vending_machine module is shown below as generated from my Verilog code.



Below is the full RTL generated, which is large due to the complexity of this module, and we can see large multiplexers for the *selection_digits* register that controls the value outputted to COST, as well as many small multiplexers for other structures like *snack_counters*, *snack_counters_next*, AND gates for *current_state* as this controls the switch statements in each always block.





V. Design Summary

Included at the end of the report are:

- Implementation (Map) report
- Design Summary portion of synthesis report
- Design Summary

VI. Conclusion

Although the most complex and challenging lab, this project allowed me to learn the most in terms of seeing how Verilog may be used to create real-life systems that are common as the vending machine. Designing the FSM and translating it into code was challenging yet rewarding it felt like system design that is fundamental to engineering.

Implementation (Map) Report

```
Release 14.7 Map P.20131013 (lin64)
Xilinx Mapping Report File for Design 'vending_machine'

Design Information
------------------
Command Line   : map -intstyle ise -p xc6slx16-csg324-3 -w -logic_opt off -ol
high -t 1 -xt 0 -register_duplication off -r 4 -global_opt off -mt off -ir off
-pr off -lc off -power off -o vending_machine_map.ncd vending_machine.ngd
vending_machine.pcf
Target Device  : xc6slx16
Target Package : csg324
Target Speed   : -3
Mapper Version : spartan6 -- $Revision: 1.55 $
Mapped Date    : Mon Mar  1 06:00:06 2021

Design Summary
--------------
Number of errors:      0
Number of warnings:   24
Slice Logic Utilization:
  Number of Slice Registers:                   181 out of  18,224    1%
    Number used as Flip Flops:                  87
    Number used as Latches:                     94
    Number used as Latch-thrus:                  0
    Number used as AND/OR logics:                0
  Number of Slice LUTs:                        245 out of   9,112    2%
    Number used as logic:                      245 out of   9,112    2%
      Number using O6 output only:             188
      Number using O5 output only:               0
      Number using O5 and O6:                   57
      Number used as ROM:                        0
    Number used as Memory:                       0 out of   2,176    0%

Slice Logic Distribution:
  Number of occupied Slices:                    82 out of   2,278    3%
  Number of MUXCYs used:                         4 out of   4,556    1%
  Number of LUT Flip Flop pairs used:          271
    Number with an unused Flip Flop:           105 out of     271   38%
    Number with an unused LUT:                  26 out of     271    9%
    Number of fully used LUT-FF pairs:         140 out of     271   51%
    Number of unique control sets:              27
    Number of slice register sites lost
      to control set restrictions:             107 out of  18,224    1%

  A LUT Flip Flop pair for this architecture represents one LUT paired with
  one Flip Flop within a slice.  A control set is a unique combination of
  clock, reset, set, and enable signals for a registered element.
  The Slice Logic Distribution report is not meaningful if the design is
  over-mapped for a non-slice resource or if Placement fails.

IO Utilization:
  Number of bonded IOBs:                        17 out of     232    7%

Specific Feature Utilization:
  Number of RAMB16BWERs:                         0 out of      32    0%
  Number of RAMB8BWERs:                          0 out of      64    0%
  Number of BUFIO2/BUFIO2_2CLKs:                 0 out of      32    0%
  Number of BUFIO2FB/BUFIO2FB_2CLKs:             0 out of      32    0%
  Number of BUFG/BUFGMUXs:                       1 out of      16    6%
    Number used as BUFGs:                        1
    Number used as BUFGMUX:                      0
  Number of DCM/DCM_CLKGENs:                     0 out of       4    0%
  Number of ILOGIC2/ISERDES2s:                   0 out of     248    0%
  Number of IODELAY2/IODRP2/IODRP2_MCBs:         0 out of     248    0%
  Number of OLOGIC2/OSERDES2s:                   0 out of     248    0%
  Number of BSCANs:                              0 out of       4    0%
  Number of BUFHs:                               0 out of     128    0%
  Number of BUFPLLs:                             0 out of       8    0%
```

```
  Number of BUFPLL_MCBs:                            0 out of       4    0%
  Number of DSP48A1s:                               0 out of      32    0%
  Number of ICAPs:                                  0 out of       1    0%
  Number of MCBs:                                   0 out of       2    0%
  Number of PCILOGICSEs:                            0 out of       2    0%
  Number of PLL_ADVs:                               0 out of       2    0%
  Number of PMVs:                                   0 out of       1    0%
  Number of STARTUPs:                               0 out of       1    0%
  Number of SUSPEND_SYNCs:                          0 out of       1    0%

Average Fanout of Non-Clock Nets:                3.88

Peak Memory Usage:  766 MB
Total REAL time to MAP completion:  16 secs
Total CPU time to MAP completion:   14 secs

Table of Contents
-----------------
Section 1 - Errors
Section 2 - Warnings
Section 3 - Informational
Section 4 - Removed Logic Summary
Section 5 - Removed Logic
Section 6 - IOB Properties
Section 7 - RPMs
Section 8 - Guide Report
Section 9 - Area Group and Partition Summary
Section 10 - Timing Report
Section 11 - Configuration String Information
Section 12 - Control Set Information
Section 13 - Utilization by Hierarchy

Section 1 - Errors
------------------

Section 2 - Warnings
--------------------
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_PWR_74_o_Select_245_o is sourced by a combinatorial pin.
   This is not good design practice. Use the CE pin to control the loading of
   data into the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_PWR_70_o_Select_237_o is sourced by a combinatorial pin.
   This is not good design practice. Use the CE pin to control the loading of
   data into the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_PWR_66_o_Select_229_o is sourced by a combinatorial pin.
   This is not good design practice. Use the CE pin to control the loading of
   data into the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_PWR_62_o_Select_221_o is sourced by a combinatorial pin.
   This is not good design practice. Use the CE pin to control the loading of
   data into the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_PWR_58_o_Select_213_o is sourced by a combinatorial pin.
   This is not good design practice. Use the CE pin to control the loading of
   data into the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_GND_8_o_Mux_62_o is sourced by a combinatorial pin. This is
   not good design practice. Use the CE pin to control the loading of data into
   the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_PWR_54_o_Select_205_o is sourced by a combinatorial pin.
   This is not good design practice. Use the CE pin to control the loading of
   data into the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_PWR_50_o_Select_197_o is sourced by a combinatorial pin.
   This is not good design practice. Use the CE pin to control the loading of
   data into the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_PWR_46_o_Select_189_o is sourced by a combinatorial pin.
   This is not good design practice. Use the CE pin to control the loading of
```

```
   data into the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_PWR_42_o_Select_181_o is sourced by a combinatorial pin.
   This is not good design practice. Use the CE pin to control the loading of
   data into the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_PWR_38_o_Select_173_o is sourced by a combinatorial pin.
   This is not good design practice. Use the CE pin to control the loading of
   data into the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_PWR_34_o_Select_165_o is sourced by a combinatorial pin.
   This is not good design practice. Use the CE pin to control the loading of
   data into the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_PWR_30_o_Select_157_o is sourced by a combinatorial pin.
   This is not good design practice. Use the CE pin to control the loading of
   data into the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_PWR_26_o_Select_149_o is sourced by a combinatorial pin.
   This is not good design practice. Use the CE pin to control the loading of
   data into the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_PWR_22_o_Select_141_o is sourced by a combinatorial pin.
   This is not good design practice. Use the CE pin to control the loading of
   data into the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_PWR_18_o_Select_133_o is sourced by a combinatorial pin.
   This is not good design practice. Use the CE pin to control the loading of
   data into the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_GND_13_o_Mux_72_o is sourced by a combinatorial pin. This is
   not good design practice. Use the CE pin to control the loading of data into
   the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_PWR_86_o_Select_269_o is sourced by a combinatorial pin.
   This is not good design practice. Use the CE pin to control the loading of
   data into the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_PWR_94_o_Select_285_o is sourced by a combinatorial pin.
   This is not good design practice. Use the CE pin to control the loading of
   data into the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_GND_3_o_Mux_52_o is sourced by a combinatorial pin. This is
   not good design practice. Use the CE pin to control the loading of data into
   the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_PWR_78_o_Select_253_o is sourced by a combinatorial pin.
   This is not good design practice. Use the CE pin to control the loading of
   data into the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_PWR_90_o_Select_277_o is sourced by a combinatorial pin.
   This is not good design practice. Use the CE pin to control the loading of
   data into the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_PWR_82_o_Select_261_o is sourced by a combinatorial pin.
   This is not good design practice. Use the CE pin to control the loading of
   data into the flip-flop.
WARNING:PhysDesignRules:372 - Gated clock. Clock net
   current_state[3]_GND_7_o_Mux_60_o is sourced by a combinatorial pin. This is
   not good design practice. Use the CE pin to control the loading of data into
   the flip-flop.

Section 3 - Informational
-------------------------
INFO:MapLib:562 - No environment variables are currently set.
INFO:LIT:244 - All of the single ended outputs in this design are using slew
   rate limited output drivers. The delay on speed critical single ended outputs
   can be dramatically reduced by designating them as fast outputs.
INFO:Pack:1716 - Initializing temperature to 85.000 Celsius. (default - Range:
   0.000 to 85.000 Celsius)
INFO:Pack:1720 - Initializing voltage to 1.140 Volts. (default - Range: 1.140 to
   1.260 Volts)
```

INFO:Map:215 - The Interim Design Summary has been generated in the MAP Report
   (.mrp).
INFO:Pack:1650 - Map created a placed design.

Section 4 - Removed Logic Summary
----
   1 block(s) optimized away

Section 5 - Removed Logic

Optimized Block(s):
TYPE   BLOCK
GND   XST_GND

Section 6 - IOB Properties

+--
--+
| IOB Name | Type | Direction | IO Standard | Diff
| Drive | Slew | Reg (s) | Resistor | IOB |
| | | | | Term
| Strength | Rate | | | Delay |
+--
--+
CARD_IN	IOB	INPUT	LVCMOS25
CLK	IOB	INPUT	LVCMOS25
COST<0>	IOB	OUTPUT	LVCMOS25
12	SLOW			
COST<1>	IOB	OUTPUT	LVCMOS25
12	SLOW			
COST<2>	IOB	OUTPUT	LVCMOS25
12	SLOW			
DOOR_OPEN	IOB	INPUT	LVCMOS25
FAILED_TRAN	IOB	OUTPUT	LVCMOS25
12	SLOW			
INVALID_SEL	IOB	OUTPUT	LVCMOS25
12	SLOW			
ITEM_CODE<0>	IOB	INPUT	LVCMOS25
ITEM_CODE<1>	IOB	INPUT	LVCMOS25
ITEM_CODE<2>	IOB	INPUT	LVCMOS25
ITEM_CODE<3>	IOB	INPUT	LVCMOS25
KEY_PRESS	IOB	INPUT	LVCMOS25
RELOAD	IOB	INPUT	LVCMOS25
RESET	IOB	INPUT	LVCMOS25
VALID_TRAN	IOB	INPUT	LVCMOS25
VEND	IOB	OUTPUT	LVCMOS25
12	SLOW			
+--
--+

Section 7 - RPMs

Section 8 - Guide Report

Guide not run on this design.

Section 9 - Area Group and Partition Summary

Partition Implementation Status

  No Partitions were found in this design.

Area Group Information

  No area groups were found in this design.

Section 10 - Timing Report

A logic-level (pre-route) timing report can be generated by using Xilinx static timing analysis tools, Timing Analyzer (GUI) or TRCE (command line), with the mapped NCD and PCF files. Please note that this timing report will be generated using estimated delay information. For accurate numbers, please generate a timing report with the post Place and Route NCD file.

For more information about the Timing Analyzer, consult the Xilinx Timing Analyzer Reference Manual; for more information about TRCE, consult the Xilinx Command Line Tools User Guide "TRACE" chapter.

Section 11 - Configuration String Details

Use the "-detail" map option to print out Configuration Strings

Section 12 - Control Set Information

Use the "-detail" map option to print out Control Set Information.

Section 13 - Utilization by Hierarchy

Use the "-detail" map option to print out the Utilization by Hierarchy section.

Design Summary

```
================================================================================
*                            Design Summary                                    *
================================================================================

Top Level Output File Name         : vending_machine.ngc

Primitive and Black Box Usage:
------------------------------
# BELS                             : 323
#      GND                         : 1
#      INV                         : 2
#      LUT2                        : 88
#      LUT3                        : 6
#      LUT4                        : 8
#      LUT5                        : 67
#      LUT6                        : 131
#      MUXCY                       : 1
#      MUXF7                       : 17
#      XORCY                       : 2
# FlipFlops/Latches                : 181
#      FD                          : 80
#      FDR                         : 4
#      FDRE                        : 3
#      LD                          : 94
# Clock Buffers                    : 1
#      BUFGP                       : 1
# IO Buffers                       : 16
#      IBUF                        : 10
#      OBUF                        : 6

Device utilization summary:
---------------------------

Selected Device : 6slx16csg324-3


Slice Logic Utilization:
 Number of Slice Registers:             181  out of  18224     0%
 Number of Slice LUTs:                  302  out of   9112     3%
    Number used as Logic:               302  out of   9112     3%

Slice Logic Distribution:
 Number of LUT Flip Flop pairs used:    311
   Number with an unused Flip Flop:     130  out of    311    41%
   Number with an unused LUT:             9  out of    311     2%
   Number of fully used LUT-FF pairs:   172  out of    311    55%
   Number of unique control sets:        27

IO Utilization:
 Number of IOs:                          17
 Number of bonded IOBs:                  17  out of    232     7%

Specific Feature Utilization:
 Number of BUFG/BUFGCTRLs:                1  out of     16     6%

---------------------------
Partition Resource Summary:
---------------------------

  No Partitions were found in this design.

---------------------------


================================================================================
Timing Report

NOTE: THESE TIMING NUMBERS ARE ONLY A SYNTHESIS ESTIMATE.
```

```
      FOR ACCURATE TIMING INFORMATION PLEASE REFER TO THE TRACE REPORT
      GENERATED AFTER PLACE-and-ROUTE.

Clock Information:
------------------
--------------------------------------------------------------------------------
+-----------------------------------+-------+
Clock Signal                                                                | Clock
buffer(FF name)             | Load  |
--------------------------------------------------------------------------------
+-----------------------------------+-------+
current_state[3]_PWR_18_o_Select_133_o(current_state[3]_PWR_18_o_Select_133_o2:0)| NONE(*)
(snack_counters_next<19>_3)| 4     |
current_state[3]_PWR_22_o_Select_141_o(current_state[3]_PWR_22_o_Select_141_o1:0)| NONE(*)
(snack_counters_next<18>_3)| 4     |
current_state[3]_PWR_26_o_Select_149_o(current_state[3]_PWR_26_o_Select_149_o1:0)| NONE(*)
(snack_counters_next<17>_3)| 4     |
current_state[3]_PWR_30_o_Select_157_o(current_state[3]_PWR_30_o_Select_157_o1:0)| NONE(*)
(snack_counters_next<16>_3)| 4     |
current_state[3]_PWR_34_o_Select_165_o(current_state[3]_PWR_34_o_Select_165_o1:0)| NONE(*)
(snack_counters_next<15>_1)| 4     |
current_state[3]_PWR_38_o_Select_173_o(current_state[3]_PWR_38_o_Select_173_o1:0)| NONE(*)
(snack_counters_next<14>_3)| 4     |
current_state[3]_PWR_42_o_Select_181_o(current_state[3]_PWR_42_o_Select_181_o1:0)| NONE(*)
(snack_counters_next<13>_3)| 4     |
current_state[3]_PWR_46_o_Select_189_o(current_state[3]_PWR_46_o_Select_189_o1:0)| NONE(*)
(snack_counters_next<12>_3)| 4     |
current_state[3]_PWR_50_o_Select_197_o(current_state[3]_PWR_50_o_Select_197_o1:0)| NONE(*)
(snack_counters_next<11>_3)| 4     |
current_state[3]_PWR_54_o_Select_205_o(current_state[3]_PWR_54_o_Select_205_o1:0)| NONE(*)
(snack_counters_next<10>_3)| 4     |
current_state[3]_PWR_58_o_Select_213_o(current_state[3]_PWR_58_o_Select_213_o1:0)| NONE(*)
(snack_counters_next<9>_3) | 4     |
current_state[3]_PWR_62_o_Select_221_o(current_state[3]_PWR_62_o_Select_221_o1:0)| NONE(*)
(snack_counters_next<8>_3) | 4     |
current_state[3]_PWR_66_o_Select_229_o(current_state[3]_PWR_66_o_Select_229_o1:0)| NONE(*)
(snack_counters_next<7>_3) | 4     |
current_state[3]_PWR_70_o_Select_237_o(current_state[3]_PWR_70_o_Select_237_o1:0)| NONE(*)
(snack_counters_next<6>_1) | 4     |
current_state[3]_PWR_74_o_Select_245_o(current_state[3]_PWR_74_o_Select_245_o1:0)| NONE(*)
(snack_counters_next<5>_3) | 4     |
current_state[3]_PWR_78_o_Select_253_o(current_state[3]_PWR_78_o_Select_253_o1:0)| NONE(*)
(snack_counters_next<4>_3) | 4     |
current_state[3]_PWR_82_o_Select_261_o(current_state[3]_PWR_82_o_Select_261_o1:0)| NONE(*)
(snack_counters_next<3>_3) | 4     |
current_state[3]_PWR_86_o_Select_269_o(current_state[3]_PWR_86_o_Select_269_o1:0)| NONE(*)
(snack_counters_next<2>_3) | 4     |
current_state[3]_PWR_90_o_Select_277_o(current_state[3]_PWR_90_o_Select_277_o1:0)| NONE(*)
(snack_counters_next<1>_3) | 4     |
current_state[3]_PWR_94_o_Select_285_o(current_state[3]_PWR_94_o_Select_285_o1:0)| NONE(*)
(snack_counters_next<0>_3) | 4     |
current_state[3]_GND_13_o_Mux_72_o(Mmux_current_state[3]_GND_13_o_Mux_72_o11:0)  | NONE(*)
(digits_tens_3)             | 4     |
current_state[3]_GND_8_o_Mux_62_o(Mmux_current_state[3]_GND_8_o_Mux_62_o1:0)     | NONE(*)
(selection_digits_4)        | 5     |
current_state[3]_GND_3_o_Mux_52_o(Mmux_current_state[3]_GND_3_o_Mux_52_o1:0)     | NONE(*)
(next_state_3)              | 4     |
CLK                                                                         | BUFGP
| 87    |
current_state[3]_GND_7_o_Mux_60_o(Mmux_current_state[3]_GND_7_o_Mux_60_o12:0)    | NONE(*)(hold)
| 1     |
--------------------------------------------------------------------------------
+-----------------------------------+-------+
(*) These 24 clock signal(s) are generated by combinatorial logic,
and XST is not able to identify which are the primary clock signals.
Please use the CLOCK_SIGNAL constraint to specify the clock signal(s) generated by combinatorial
logic.
INFO:Xst:2169 - HDL ADVISOR - Some clock signals were not automatically buffered by XST with
BUFG/BUFR resources. Please use the buffer_type constraint in order to insert these buffers to
the clock signals to help prevent skew problems.

Asynchronous Control Signals Information:
```

```
----------------------------------------
No asynchronous control signals found in this design

Timing Summary:
---------------
Speed Grade: -3

   Minimum period: 6.032ns (Maximum Frequency: 165.782MHz)
   Minimum input arrival time before clock: 8.259ns
   Maximum output required time after clock: 7.620ns
   Maximum combinational path delay: No path found

Timing Details:
---------------
All values displayed in nanoseconds (ns)

=========================================================================
Timing constraint: Default period analysis for Clock 'current_state[3]_PWR_18_o_Select_133_o'
  Clock period: 5.659ns (frequency: 176.724MHz)
  Total number of paths / destination ports: 10 / 4
-------------------------------------------------------------------------
Delay:               5.659ns (Levels of Logic = 5)
  Source:            snack_counters_next<19>_0 (LATCH)
  Destination:       snack_counters_next<19>_3 (LATCH)
  Source Clock:      current_state[3]_PWR_18_o_Select_133_o falling
  Destination Clock: current_state[3]_PWR_18_o_Select_133_o falling

  Data Path: snack_counters_next<19>_0 to snack_counters_next<19>_3
                                Gate     Net
    Cell:in->out      fanout   Delay   Delay  Logical Name (Net Name)
    ----------------------------------------  ------------
     LD:G->Q               1   0.498   0.580  snack_counters_next<19>_0
(snack_counters_next<19>_0)
     LUT2:I1->O            2   0.205   0.845  Mmux_snack_counters_next[19]
[3]_GND_1_o_mux_9_OUT741 (snack_counters_next[19][3]_GND_1_o_mux_9_OUT<76>)
     LUT6:I3->O            1   0.205   0.000  Mmux_selection_digits[4]_X_1_o_wide_mux_97_OUT_3
(Mmux_selection_digits[4]_X_1_o_wide_mux_97_OUT_3)
     MUXF7:I1->O          62   0.140   1.855  Mmux_selection_digits[4]_X_1_o_wide_mux_97_OUT_2_f7
(Msub_GND_1_o_GND_1_o_sub_99_OUT<3:0>_cy<0>)
     LUT4:I1->O           11   0.205   0.883  Msub_GND_1_o_GND_1_o_sub_99_OUT<3:0>_xor<3>11
(GND_1_o_GND_1_o_sub_99_OUT<3>)
     LUT5:I4->O            1   0.205   0.000  current_state[3]_PWR_1_o_Select_132_o1
(current_state[3]_PWR_1_o_Select_132_o)
     LD:D                      0.037          snack_counters_next<19>_3
    ----------------------------------------
    Total                      5.659ns (1.495ns logic, 4.164ns route)
                                       (26.4% logic, 73.6% route)

=========================================================================


Total REAL time to Xst completion: 17.00 secs
Total CPU time to Xst completion: 14.59 secs

-->


Total memory usage is 485284 kilobytes

Number of errors   :    0 (   0 filtered)
Number of warnings :   96 (   0 filtered)
Number of infos    :    1 (   0 filtered)
```

vending_machine Project Status			
Project File:	lab3.xise	**Parser Errors:**	No Errors
Module Name:	vending_machine	**Implementation State:**	Placed and Routed
Target Device:	xc6slx16-3csg324	• **Errors:**	No Errors
Product Version:	ISE 14.7	• **Warnings:**	120 Warnings (24 new)
Design Goal:	Balanced	• **Routing Results:**	All Signals Completely Routed
Design Strategy:	Xilinx Default (unlocked)	• **Timing Constraints:**	All Constraints Met
Environment:	System Settings	• **Final Timing Score:**	0 (Timing Report)

Device Utilization Summary [-]				
Slice Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Registers	181	18,224	1%	
Number used as Flip Flops	87			
Number used as Latches	94			
Number used as Latch-thrus	0			
Number used as AND/OR logics	0			
Number of Slice LUTs	245	9,112	2%	
Number used as logic	245	9,112	2%	
Number using O6 output only	188			
Number using O5 output only	0			
Number using O5 and O6	57			
Number used as ROM	0			
Number used as Memory	0	2,176	0%	
Number of occupied Slices	82	2,278	3%	
Number of MUXCYs used	4	4,556	1%	
Number of LUT Flip Flop pairs used	271			
Number with an unused Flip Flop	105	271	38%	
Number with an unused LUT	26	271	9%	
Number of fully used LUT-FF pairs	140	271	51%	
Number of unique control sets	27			
Number of slice register sites lost to control set restrictions	107	18,224	1%	
Number of bonded IOBs	17	232	7%	
Number of RAMB16BWERs	0	32	0%	
Number of RAMB8BWERs	0	64	0%	
Number of BUFIO2/BUFIO2_2CLKs	0	32	0%	
Number of BUFIO2FB/BUFIO2FB_2CLKs	0	32	0%	
Number of BUFG/BUFGMUXs	1	16	6%	

Number used as BUFGs	1			
Number used as BUFGMUX	0			
Number of DCM/DCM_CLKGENs	0	4	0%	
Number of ILOGIC2/ISERDES2s	0	248	0%	
Number of IODELAY2/IODRP2/IODRP2_MCBs	0	248	0%	
Number of OLOGIC2/OSERDES2s	0	248	0%	
Number of BSCANs	0	4	0%	
Number of BUFHs	0	128	0%	
Number of BUFPLLs	0	8	0%	
Number of BUFPLL_MCBs	0	4	0%	
Number of DSP48A1s	0	32	0%	
Number of ICAPs	0	1	0%	
Number of MCBs	0	2	0%	
Number of PCILOGICSEs	0	2	0%	
Number of PLL_ADVs	0	2	0%	
Number of PMVs	0	1	0%	
Number of STARTUPs	0	1	0%	
Number of SUSPEND_SYNCs	0	1	0%	
Average Fanout of Non-Clock Nets	3.88			

Performance Summary			[-]
Final Timing Score:	0 (Setup: 0, Hold: 0)	**Pinout Data:**	Pinout Report
Routing Results:	All Signals Completely Routed	**Clock Data:**	Clock Report
Timing Constraints:	All Constraints Met		

Detailed Reports					[-]
Report Name	**Status**	**Generated**	**Errors**	**Warnings**	**Infos**
Synthesis Report	Current	Mon Mar 1 05:42:30 2021	0	96 Warnings (0 new)	1 Info (0 new)
Translation Report	Current	Mon Mar 1 06:00:05 2021	0	0	0
Map Report	Current	Mon Mar 1 06:00:29 2021	0	24 Warnings (24 new)	6 Infos (1 new)
Place and Route Report	Current	Mon Mar 1 06:00:42 2021	0	0	3 Infos (3 new)
Power Report					
Post-PAR Static Timing Report	Current	Mon Mar 1 06:00:49 2021	0	0	4 Infos (4 new)
Bitgen Report					

Secondary Reports			[-]
Report Name	**Status**	**Generated**	
ISIM Simulator Log	Out of Date	Mon Mar 1 05:42:03 2021	

Date Generated: 03/01/2021 - 06:01:13