ECE 530 Activity 03 – Soda Machine Design

Richard Groves

October 14, 2023

1 Overview

The objective of this activity was to design soda-dispensing machine using finite state machines (FSM). The design was synthesized, simulated, and the resource utilization was examined.

2 Design Procedure

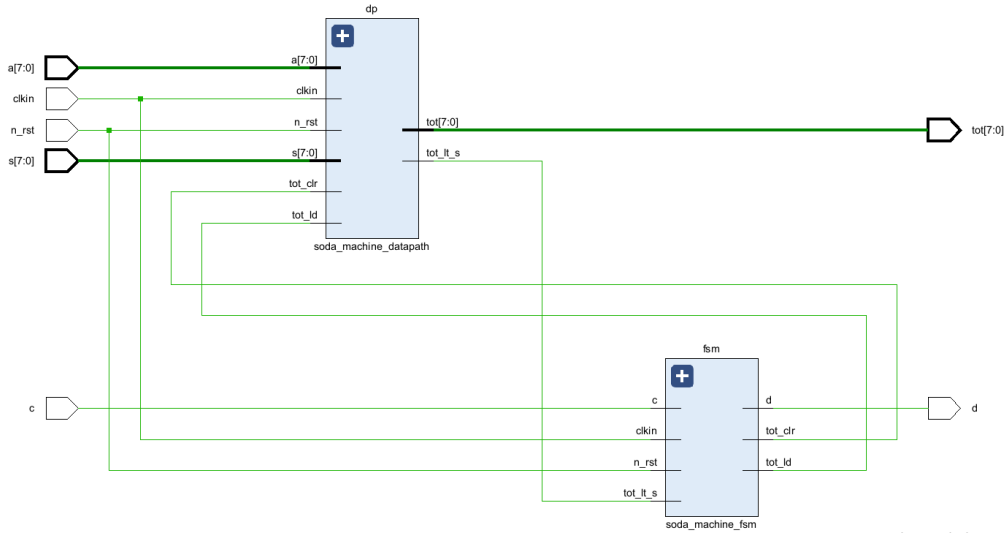

Figure 1: Soda Machine Top Module Schematic

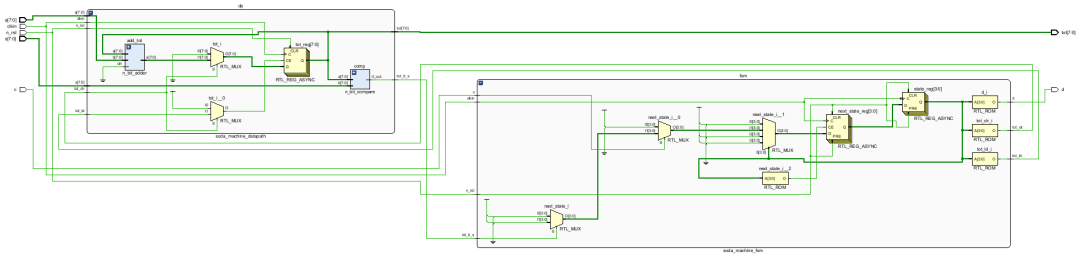

Figure 2: Soda Machine Detailed Schematics

2.1 Soda Machine FSM Module

This module encompassed the FSM design and was divided into three procedural blocks: fsm_register, next_state_logic, and output_logic. There were four states:

- Init: The initialization state. This state initializes the soda machine.
- Wait: The wait state. In this state, the machine is waiting for conditions to either go to "Add" or "Disp".

- Add: The coin adding state. This state is used when a coin is added to the soda machine.
- Disp: The soda dispensing state. This state occurs when the total coins deposited is greater than the cost of the soda.

2.2 Soda Machine Data Path Module

This module incorporated the supporting components required for the FSM operation as outlined in the block diagram. This module was responsible for keeping track of the amount of money deposited, and comparing the total to the cost of the soda.

2.3 Soda Machine Top Module

The top module, 'soda_machine_top', comprised instantiations for the other two modules and established the connection between them.

2.4 Simulation

A simple test bench was constructed to validate the design functionality. Testing involved:

- Setting the cost to $1.50.
- Performing a reset.
- Adding 4 quarters, 5 dimes and 3 nickels.
- Performing a reset.
- Setting the cost to $2.00.
- Adding coins to total $2.50.

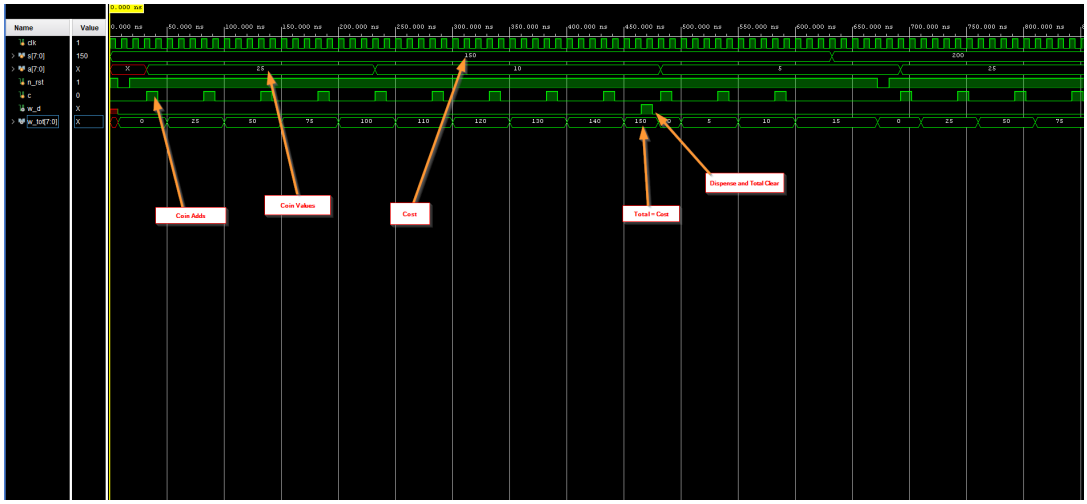

Figure 3: First Simulation Test

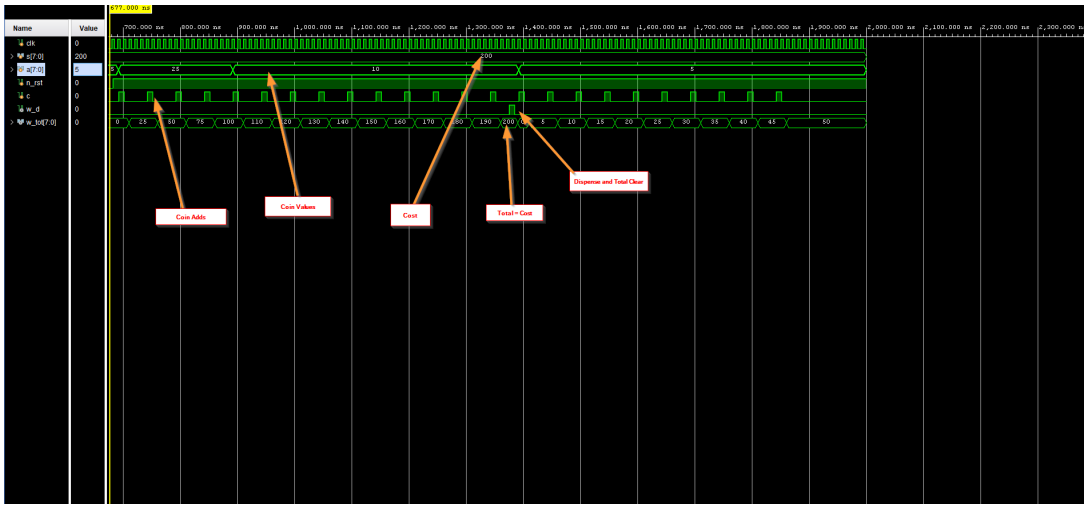

Figure 4: Second Simulation Test

2.5 Wrapper for Soda Machine

The wrapper integrated the soda machine module with components that interfaced the switches, pushbuttons, OLED display, edge detectors, and frequency divider. It also included instantiations for a module to convert the binary cost and total values to BCD. The BCD value was used for displaying on the OLED. This top module contained a state machine to control the OLED.

2.6 Utilization

The design was synthesized, and the resource utilization was examined to ensure the absence of latches.

1. Slice Logic

Site Type	Used	Fixed	Prohibited	Available	Util%
Slice LUTs*	733	0	0	17600	4.16
LUT as Logic	733	0	0	17600	4.16
LUT as Memory	0	0	0	6000	0.00
Slice Registers	330	0	0	35200	0.94
Register as Flip Flop	330	0	0	35200	0.94
Register as Latch	0	0	0	35200	0.00
F7 Muxes	32	0	0	8800	0.36
F8 Muxes	1	0	0	4400	0.02

No Latches

Figure 5: Utilization Table