Lab 10: Vending Machine

Due: May 14, 2015

Objective

- To be familiar with modeling finite state machines with Verilog.

Action Items

1. Design a controller for a soft drink vending machine. The controller has the following input ports:

   ```
   input clk;
   input rst_n;
   input cancel;
   input tea;
   input coke;
   input sprite;
   input money_5;
   input money_10;
   input money_50;
   ```

   and the following output ports:

   ```
   output drop_tea;
   output drop_coke;
   output drop_sprite;
   output [3:0] DIGIT;
   output [7:0] DISPLAY;
   ```

   - When pressing the rst_n button, the two leftmost 7-segment displays will show "00", and the two rightmost 7-segment displays will show nothing. The three LEDs respectively corresponding to drop_tea, drop_coke and drop_sprite will not light up.
   - Every time after a buyer deposits a coin (NT$ 5, 10, or 50), the two leftmost 7-segment displays will immediatley show the total amount of money that has been deposited so far. If a buyer wants

to deposit NT$ 5, he/she presses the money_5 button; if a buyer wants to deposit NT$ 10, he/she presses the money_10 button; if a buyer wants to deposit NT$ 50, he/she presses the money_50 button.

- A buyer may deposit any number of coins. If the buyer deposits more than NT$ 50, treat it as NT$ 50.
- Each bottle of tea costs NT$ 15, each bottle of coke costs NT$ 20 and each can of sprite costs NT$ 25. If the deposited money is not enough, then pushing the tea, coke or sprite button does not effect, and the LED corresponding to the drop_tea, drop_coke or drop_sprite will not light up.
- When the money deposited so far is enough for buying a drink, the LED corresponding to drop_tea, drop_coke or drop_sprite will light up. After the buyer selects tea, coke or sprite by pushing the corresponding button, the two lefttmost 7-segment displays will show the change to be returned to the buyer, where the change will be decrmented by NT$ 5 (to simulate droppong a coin of NT$ 5) at each clock cycle (**with the frequency of clk/(2^25)**) until 00 is reached. After the change becomes 00, the buyer can deposit coins again. The buyer cannot buy another drink while the machine is returning the change.
- If the buyer decides not to buy any dirnk (no matter how much money has been deposited), he/she can press the cancel button for getting the money back (but at most NT$ 50 will be returned if more than NT$ 50 have been deposited). The money will be returned in the same way as returning the change.

More details about the I/O signals of the design are given below.

- **clk**: clock signal (which is connected to the FPGA pin **R10).**
- **rst_n:** asynchronous negative trigger reset which is generated from an onepulse circuit whose input comes from the output of a debouncer; the data input of the debouncer is the pushbutton **S1** connected to the FPGA pin **N3**. When rst_n is enabled, the numbers being shown on the two leftmost 7-segment displays are set to 0.
- **cancel**: the cancel button which is generated from an onepulse

circuit whose input comes from the output of a debouncer; the data input of the debouncer is the pushbutton **S8** connected to the FPGA pin **T2**.

- **tea**: the tea button which is generated from an onepulse circuit whose input comes from the output of a debouncer; the data input of the debouncer is the pushbutton **S2** connected to the FPGA pin **P4**.
- **coke**: the coke button which is generated from an onepulse circuit whose input comes from the output of a debouncer; the data input of the debouncer is the pushbutton **S3** connected to the FPGA pin **P3**.
- **sprite**: the sprite button which is generated from an onepulse circuit whose input comes from the output of a debouncer; the data input of the debouncer is the pushbutton **S4** connected to the FPGA pin **L6**.
- **money_5**: the money_5 button which is generated from a onepulse circuit whose input comes from the output of a debouncer; the data input of the debouncer is the pushbutton **S5** connected to the FPGA pin **M5**.
- **money_10**: the money_10 button which is generated from a onepulse circuit whose input comes from the output of a debouncer; the data input of the debouncer is the pushbutton **S6** connected to the FPGA pin **U2**.
- **money_50**: the money_50 button which is generated from a onepulse circuit whose input comes from the output of a debouncer; the data input of the debouncer is the pushbutton **S7** connected to the FPGA pin **U1**.
- **DISPLAY**: signal to show the money that has been deposited so far, or the money to be returned.
- **DIGIT**: signal to enable one 7-segment display.
- **drop_tea:** the drop_tea signal which is the LED **D1** connected to the FPGA pin **K4**.
- **drop_coke:** the drop_coke signal which is the LED **D2** connected to the FPGA pin **K3**.
- **drop_sprite:** the drop_sprite signal which is the LED **D3** connected to the FPGA pin **L5**.

2. Before you add the debouncing and onepusle circuits to your design, write a testbench, including at least the following input stimuli, to verify your controller.
    a. Deposits NT$ 15 (NT$ 5 + NT$ 10) and selects tea.
    b. Deposits NT$ 20 (2 * NT$ 5 + NT$ 10) and selects coke.
    c. Deposits NT$ 60 (NT$ 10 + NT$ 50) and selects sprite.
    d. Deposits NT$ 100 (2 * NT$ 5 + 4 * NT$ 10 + NT$ 50) and selects cancel.