LAB Assignment #4 for ECE 337

Assigned: Wed., Nov. 16, 2011
Due: Wed., Nov. 30, 2011

Description: Write the VHDL code that implements a state machine that models a vending machine.

Your vending machine model will allow the user to insert 'coins' by associating presses of BTN1 for Nickels and presses of BTN2 for Dimes. For example, pressing BTN1 once and BTN2 once inserts 15 cents. You will use a state machine to keep track of how much money has been inserted using the following model:
State S0: Idle state (no money inserted). If a Nickel is inserted, goto state S1. If a Dime is inserted, goto state S2. Otherwise stay in state S0.
State S1: Nickel state. If a Nickle is inserted, goto state S2. If a Dime is inserted, goto state S3. Otherwise stay in state S1.
State S2: Dime state. If a Nickle is inserted, goto state S3. If a Dime is inserted, goto state S3.
State S3: Dispense state: Stay in S3 for one clock cycle and then return to S0.

BTN3 should be connected to RESET. While in any state, if BTN3 is pressed, perform an asynchronous reset on all registers (which will return the state machine to S0).

When state S3 is reached, you should turn on LED0. LED0 will remain on until a timer expires. The timer is to be implemented as a 26-bit counter (register) and will expire every time the count reaches 0. The behavior of the timer should also be made visible on LED6. Every time the timer fires (counter register becomes 0), you should toggle (invert) a 1-bit register to drives LED6. Therefore every couple seconds, LED6 should turn on and then a couple of seconds later, it should turn off again. This slow blinking behavior of LED6 should occur continuously after you program the FPGA.

Be sure to use the DEBOUNCE VHDL module from previous labs to debounce your buttons. Use the 'button_stable' outputs from the debouncer in your state machine.

I have supplied the UCF file that you should use. In addition to the above behavior, you should connect the 'button_level' outputs of the debouncer modules to LEDs. Connect BTN1 (Nickel) button_level to LED1 and BTN2 (Dime) to LED4. Be sure to change the signal names in my UCF file to match your signals.

Upto 20 extra points will be given for implementations that use the UART port and PUTTY to display state information, such as how much money has been inserted and when the user has inserted enough money to reach S3 (the dispense state).

Laboratory Requirements:

1) In class DEMO on the due date.
Grading:
100 pts will be given for a proper demonstration.