Laboratory Exercise 3

Due February 12, 2016.

Part I.
Using Aldec as the front end for development, create a model for a modulo-13 up-down counter (SW[1]) with an asynchronous reset (key[1]), a synchronous load (key[2]) from switches SW[2..5], an enable (SW[1]). Your counter must also have a terminal count output that is 1 when the counter is both enabled and on its terminal count. Display the count on the red LEDs ledr[0..3] (ledr[3] is msb) and the terminal count on ledr[4]. The clock is to be input from key[0]. Your state machine model should be similar to the behavioral model for state machines presented in Chapter 2 of Roth & John.

Debounce the clock as demonstrated in class and have any debounce circuitry separate (either as a component or process) from the counter. As a reference, review Figure 4-22 from Roth & John. Using the **`altpll`** MegaFunction, initially create a clock with a period of 100ns to be used for the double sampling circuit. Next create a process that double samples the signal from key[0]. The output from the double sampling circuit will be the clock for the previously described state machine. Experiment with different clock periods from 4ns to 1000ns and note whether the circuit operates correctly.

Include in your write-up, any design you had to perform and timing diagram for important steps.

Part II.
The result of Laboratory Exercise 2 is the start of a basic calculator. Merge Laboratory 2 with this lab and make the following modifications:

- Use switches SW[2..5] to specify the operation
- Add logical shift left, logical shift right, arithmetic shift left, arithmetic shift right, rotate left, rotate right, Boolean Not, and negate and assign them inputs $SW_{5..2}$=0111, 1000,..., 1110 respectively.
- Use SW[6] to specify whether the modulo counter output (SW[6]=0) or the switches (SW[6]=1) are used to specify the operation.
- Operand 1 should be input from switches SW[10 to 17] where SW17 is the MSB, and should be displayed on the HEX6 and HEX7 displays. Sign extend operation 1 to 16 bits.
- Add a 16 bit register that is loaded with the result of any operation. Its purpose is similar the value displayed in a calcular and serves as the source for one binary operation as well as the destination. In addition, it serves as the source and destination for unary operations. This value should be displayed on HEX0, HEX1, HEX2, and HEX3.
- Light ledg(8) if overflow occurs on any operation.

As a part of your write-up, include any design that was necessary and important timing diagrams. In addition, give your testing strategy for verifying that all functional requirements are met. Try and be as specific as possible.