Casino-type Game

PURPOSE - In this lab you will design a casino-type game using the random number generator from the previous lab.

1. Writing the Verilog description

You will have to design a simple game using the random number generator from the previous lab. This game will function as follows: The random number generator will start generating numbers. When the Roll button is pressed you will compare these two hex digits and declare a win if they are equal, or a lose otherwise.

In order to reuse the most code possible you will have the two random hex numbers displayed in real time on the seven segment display (on either the left two or the right two digits, just like in lab 4), and then display the win "UI" or lose "LO" notification on the remaining two digits. **Read the following example to understand the proper operation of the game:**

A sample play of the game is as follows:
1) The player starts to generate the random numbers on two of the digits just as in lab 4.

2) The player then **taps** the Roll button. If the current two hex numbers are equal then "UI" is displayed, if not, then "LO" is displayed (on the other remaining two digits). The "UI" or "LO" is still displayed until the next tap of the Roll button. The two hex numbers are still changing throughout this entire process.

3) The player can then tap the Roll button again and the (new) current two hex numbers are compared again to see if the other two digits should display "UI" or "LO".

For simplicity "LO" can be displayed before the first tap of the roll button.

Of course, the player can always see the value of the hex digits, and thus tap the Roll button at the right time in order to win or lose the game, but this is done for debugging purposes.

2. Verification

Synthesize and implement the game to verify the correctness. You will have to write an .xdc file to correctly map the pins (you only need to add two more lines for the Roll button.) Download the bit-stream file to the board and try different “feedback(fbk)” combinations. Show the correct functioning to your TA for full credit.

SUMMARY -- In this lab you designed and verified a simple game.

Important Note: With the simplifications mentioned above, you should be able to implement this game without the use of a state machine, although you are free to use one if you please.