DLD Project 2017

Combination Lock
10% of final grade
Due: 11:59pm (Monday, Dec 11)

Submission Procedure: through Sakai Assignments and submit under 'Project'. Projects submitted late will be accepted with **20%** penalty (upto 2 days). No project will be accepted after **13 Dec.**

Questions will be answered in class from **12 noon until 1:20pm**. No answers will be given out at other times.

I will not respond to emailed questions. I will only give the answers during class sessions. No questions will be answered answered after **1.20pm Friday, December 8**. So finish your project early. Don't expect any help unless you come to class and ask your questions for all to hear.

Task:

Design an electronic combination lock state machine that will only open with your RUID. First, convert your RUID to binary because you will need the least 12 significant bits to open the lock. **Use a decimal-to-binary converter that can be found on Google**.

Your design will require a 16-state state machine in which 8 of the states will be good and 8 will be not used. List the 16 states in binary order starting with 0000 and ending with 1111.

Use the least four significant bits of your binary RUID as your starting point. Your next 8 bits will be your X input to the machine. Starting with the 5th bit (b4), advance through the machine until you reach the 12th bit as follows: if your bit is a “0", then advance to the next binary state: but if your bit is a "1", then advance by skipping the next state and go to the state after that.

For example, if you are in state 1001 and your X = 0, then go to 1010; but if X = 1, then go to 1011. Use this procedure for all 8 bits from bit 5 to 12. In your advancing when you reach 1111, wrap around to 0000 and then continue. **Remember, 5th bit is b4 and the 12th bit is b11.**

There is one input, X, and two outputs: R (lights a red LED) and G (lights a green LED). The 12th bit will return you to the start state whether it is correct or not.

Each correct input will advance to the next expected state, any wrong input bit will return to the start state and output R will be =1. When successfully starting at the right place and receiving bits 5 through 12 correctly, the machine will return to the start state and output G = 1, which will unlock the lock.

Deliverables:

1) Cover Page with name, date, RUID, and in binary.
2) State diagram with all 16 states including the start state, the 8 states that you are using and the 8 unused states.
3) State table showing the 8 states you are using and X=0, X=1.
4) Excitation table for J-K flip-flop.
5) One 8 state transition table for J and K with X=0.
6) One 8 state transition table for J and K with X=1.
7) One 8 state transition table for the outputs R and G with X=0.
8) One 8 state transition table for the outputs R and G with X=1.
9) One each 5 variable K-map for outputs R and G.
10) Eight, 5 variable K-maps for q3, q2,q1,q0 and X **or** you can do:
Sixteen, 4 variable K-maps. Eight for J-K, x=0 and Eight for J-K X=1.
11) Reduced equations for J3, j2, J1, j0 and K3, K2, K1, K0.
12) One page discussion of your design, explain problems you had with this projects and how you resolved them. Also, how the project can be made better.

** Minus 1 point for each missing item!