Digital System Design Capsule
Digital Design Semester Project

Objectives:

- Design and implement a Door Locker
- Design and implement a **Simple State Machine**
- Practice writing testbench for sequential circuits
- Use Structural VHDL style to connect multiple modules to implement a digital circuit.

Project Overview:

The purpose of the project is to implement a Door Locker on an FPGA board. Your student ID number will be a ***unique*** password to open the door. We would like to define the password ***P*** using your student ID number. Assume that your student ID number is in hexadecimal format and you convert it to the binary number.

For example, if your student ID number is 2011001234, you can convert it to the binary number as fallows;

$(1234)_{16} = (0001\ \mathbf{0010\ 0011\ 0100})_2$

In this case, the password of the door is **0010 0011 0100** and you are supposed to design a state machine to detect the correct password **0010 0011 0100.**

To be able to open the door, the user first should set the inputs to **0010 0011 0100** using 12 switches (SW0, SW1, -- SW11). Next, the user should push the button BTNR (which is an *Enable* input in the Figure) to store data in the register. Then, the user needs to push BTNL (which is a *Start* input in the Figure) to send data to the State Machine and initiate the password check process. The stored data is sent to the state machine in a serial manner, one bit of data is sent to the state machine in each clock.

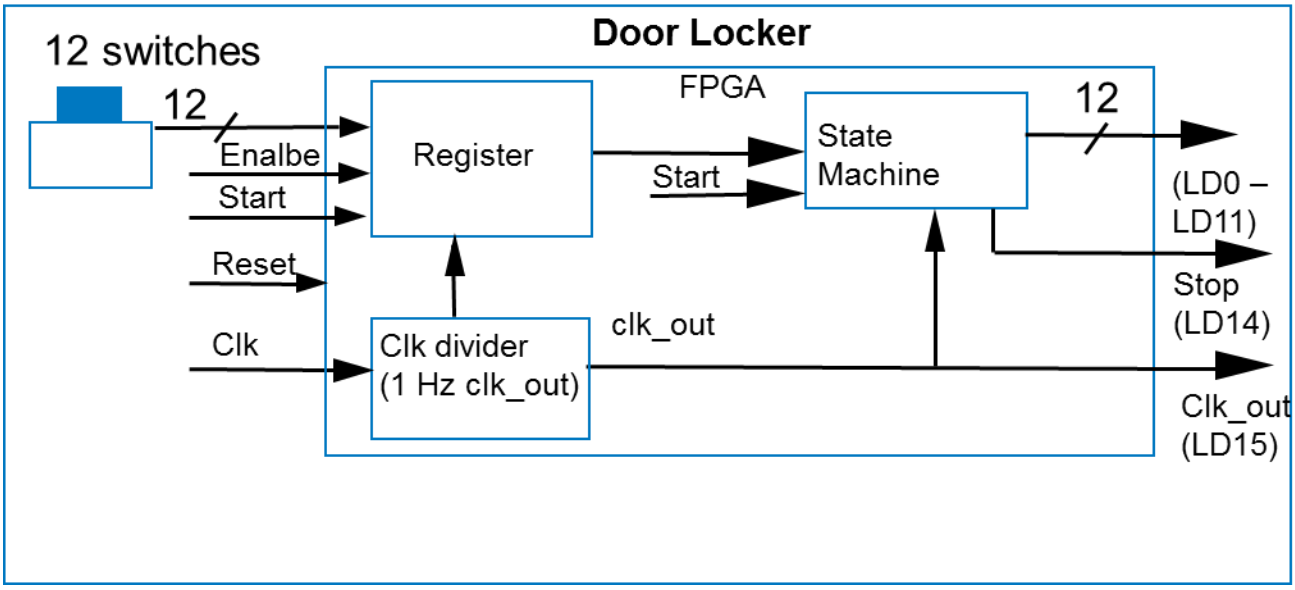

Figure 1. Door Locker Top-Level Design

12 LEDs (LD0, SW1, -- LD11) are utilized to show whether the entered password is correct or not. Starting from LSB, each bit of the entered data in register should be compared with your student ID. If the each bit of the entered data is correct their corresponding LED should be turned on. For example; if the first bit is correct LD0 should be turned on in the first clock, in the second clock LD1 should be turned on if the second bit is correct (LD0 and LD1 are on right now).

If the third bit is not correct, state machine should stay in the same state (second state in this example) and the wrong password indicator (LD14 which is the ***Stop*** output in the Figure) should be turned on. The user should push the Reset button to go to the initial state and to turn off all LEDs. If the entered password totally matches with your student ID the door should be opened (indicated with all 12 LEDs are turned on). We also need to implement a reset button in a way like that when the user pushes the reset button state machine goes to the initial state and resets register content.

Extra Credit:
Feel free to add extra specifications to your design, each specification will receive extra credits. For example; you can simplify the state machine by removing some of the states, and you can add 3 digit display using seven segment decoder that you designed earlier.

Action Items for project demo:

Step 1)
Write a **testbench** for your shift register. If your code is correct, you should get a parallel to serial shift register as we discussed in the lecture.

Step 2)
Get the following .vhd files and put them in the .srcs folder of this project

- ClkDiv.vhd file
- Register.vhd file and SM.vhd file

Step 3)
Create a new .vhd file to implement the top-level design. **Set this new .vhd file as Top module** (Right-click on the file name under Design Sources and choose "Set as Top")
Write ***Structural*** VHDL code to connect your components together as shown in Figure 1.

Step 4)
Assign the pins to your top-level design circuit (create .xdc file).

Step 5)
Generate Bitstream and download to the board.
Push the reset button to start over after your machine gets to the final.

Grading Criteria

Register design and its VHDL and Testbench	10 pts
State diagram of the State Machine and VHDL	10 pts
Testbench for State Machine and correct waveform	10 pts
Structural code for Top-level design and testbench with correct weveform	20 pts
Correct .xdc file for top-level circuit and bitstream file	10 pts
Whole design with working on the FPGA board	20 pts
Demo question and answer	20 pts

Code submission: failure in submission will result in a 40% penalty

1. Submit all the vhd source files (.vhd only) including testbenches on Canvas to the designated dropbox.
2. screenshots of your **post synthesis simulation** waveform for state machine.
3. screenshots of your **post synthesis simulation** waveform for top level design.