Lab 3 Basic: Lights Out

Objective

1. Getting familiar with the LEDs, clock designs, and finite-state machines in Verilog.

Action Items

1 Lab3_basic.v (100%)

In this lab, you must design a game called “Lights Out.” This game must be controlled by a finite-state machine (FSM) which controls the LEDs based on the switches.

a. I/O list

I/O	Connected to	Definition
clk	W5	Clock signal with the frequency of 100MHz
rst	SW15	Asynchronous positive reset signal, resetting the FSM to the **INITIAL** state
restart	SW14	Restarting the game from the **FINISH** state to the **INITIAL** state
SW[9:0]	SW9~SW0	Controlling the LEDs
led[9:0]	LD9~LD0	Lighting up or turning off the LEDs based on the control of the switches

b. **Finite state machine (FSM)**

The FSM may consist of an **INITIAL** state to initialize the game, a **PLAYING** state that allows the user to turn on or turn off the light, and after finishing the game, the FSM will enter a **FINISH** state, and the LEDs will keep flashing until the state of FSM is switched. You can design your own variant instead. However, the clock rate of the FSM must be **100MHZ/2^{10}**. Note that in any state, when the rst == 1 (following the positive edge), the FSM will reset to the INITIAL state. When the rst goes back to 0, the FSM will operate normally.

✓ INITIAL state

- When entering this state, the FSM will switch to the **PLAYING** state at the next cycle if the restart is 0.
- In this state, the **LD9~LD0** should be as the pattern below. The LEDs in the **INITIAL** state:

LD9 ○○●○○●○○○○ LD0

(● : LED on, ○ : LED off)

✓ PLAYING state

- The **restart** is unfunctional in this state.
- In this state, when you toggle the switch, the adjacent LEDs will toggle between on and off.

  EX:

  At the beginning, the LEDs will light up as follows:

  LD9 ○○●○○●○○○○ LD0

  (● : LED on, ○ : LED off)

  If we toggle the SW5, the LEDs will light up as follows:

  LD9 ○○●●●○○○○○ LD0

  (● : LED on, ○ : LED off)

  If we toggle the SW6, the LEDs will light up as follows:

  LD9 ○○○○○○○○○○ LD0

  (● : LED on, ○ : LED off)
- If the LEDs are all **off**, the FSM will enter the **FINISH** state.
- For the demo, you **must** set the initial LED pattern to:

  LD9 ○○●○○●○○○○ LD0

  (● : LED on, ○ : LED off)

  Note: if you don't follow this rule, you won't the score of this part.

✓ FINISH state

- When entering this state, the **LD9~LD0** will keep flashing in the rate of **100MHZ/2^{27}** even if the switches are toggled.
- The switches for SW[9:0] are unfunctional in this state.
- The FSM can return to the **INITIAL** state by setting the **restart** to 1.

c. You have to use the following template for your design:

```
module lab3_basic (
    input wire clk,
    input wire rst,
    input wire restart,
    input wire [9:0] SW,
    output reg [9:0] led
     );
  /* Note that output ports can be either reg or wire.
  * It depends on how you design your module. */
  // add your design here
endmodule
```

Attention

✓ DO NOT copy-and-paste code segments from the PDF materials. It may also paste invisible non-ASCII characters, leading to hard-to-debug syntax errors.

- ✓ If you have two or more modules used for any specific lab, merge them into one Verilog file before the submission.
- ✓ You should submit one source files, **lab3.v**. Upload source file individually to its corresponding OJ problem. DO NOT hand in any compressed ZIP files, which will be considered an incorrect format.