Digital Systems Design with FPGAs: Lab-2

Sukhwinder Singh
Department of Electrical and Computer Engineering
University of Washington
Seattle, Washington
singh27@uw.edu

***Abstract*— Lab 2 Applying Sequential Logic and LFSR in FPGA**

I. Introduction

The Learning objective of the lab is divided into three main parts

1. FSM and Sequential Logic functions
2. Communicating Sequential Logic
3. LFSR- Linear Feedback Shift Register

These three parts are divided mainly into 3 sub projects one is on Hazard lights and the remaining two on Tug of war player vs player and Player vs bot.

II. Ease of Use

All of the Three projects consist of Screenshots of the wave formation in simulation and the video recording of them running on the Board DE1_SoE. The details are as below.

1.2 SEA-TAC Landing Hazard Light

In this Project, we are provided with two combinational inputs and they light have to perform different patterns as per the input.

On a Circuit board, we only have 4 patterns of light and their state Diagram is as below.

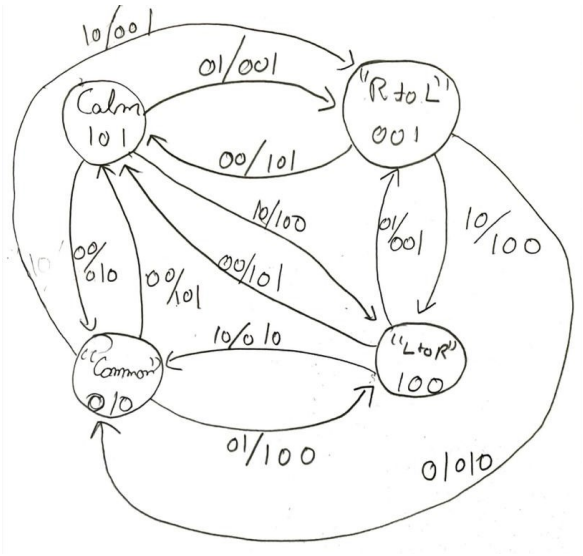

Fig 1: State Diagram of Hazard Light

The first combination provided is the Calm where the lights have a neutral Pattern, This is the ideal condition where no input is provided and the switch 1 and 0 are at 0 level. However, when we change the switches to 0 and 1 the light pattern will indicate the flow of lights from Right and left and similarly when we switch the light to 1 and 0, the pattern will switch from left to right.

The waveform of the Hazard Light Inputs and Output are Simulated on Model Sim as below.

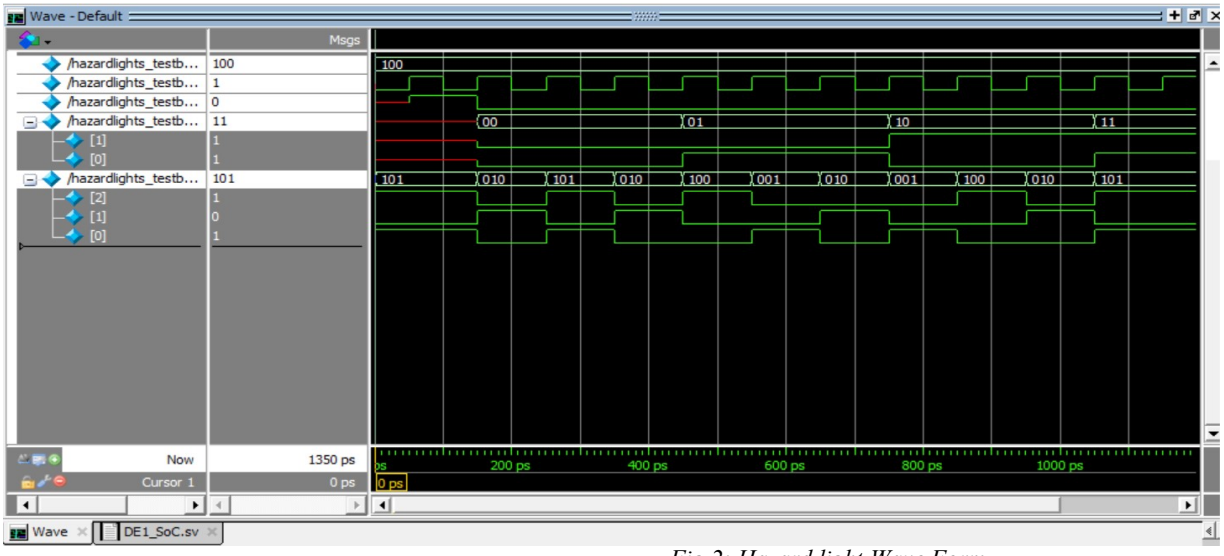

Fig 2: Hazard light Wave Form

As a part of Deliverables, the Resource Utilization by Entity for the Hazard Light project is configured below.

Analysis & Synthesis Resource Utilization by Entity

<<Filter>>

	Compilation Hierarchy Node	Combinational ALUTs	Dedicated Logic Registers	Block Memory Bits	DSP Blocks	Pins	Virtual Pins	Full Hierarchy Name	Entity Name
1	|DE1_SoC	1 (0)	0 (0)	0	0	31	0	|DE1_SoC	DE1_SoC
1	|hazardlights:ONE|	1 (1)	0 (0)	0	0	0	0	|DE1_SoC|hazardlights:ONE	hazardlights

Fig 3: Resource Utilization by Entity

To Reset the Signal at any time, Key[0] can be pressed to reset signal in that particular state.

In addition to this, the working video of runway lights circuit working in DE1_SoC board. Showing the behavior of all three wind conditions is available in the Media folder of folder Lab2- 1.2.

2. Communicating Sequential Logic

2.1 Tug of War

In This Lab, we are assigned to do the Tug of War between two players. The player one play from the Key[0] and the Player 2 will play from the Key[4].

On the Initial setup, the LEDR5, will glow up which show the center light of the game. As the Player's keep pressing their buttons respectively the light will go from LEDR5 to 6 to 7 to 8 to 9 for player 2 and if he presses again the HEX0 will display the winner's number as 2. Similarly, in case of Player 1, his victory light Patten will be from 5 to 4 to 3 to 2 to 1 and then for an additional press the HEX0 will win the game.

On an Additional note, SW[9] is used to reset the state at any time.

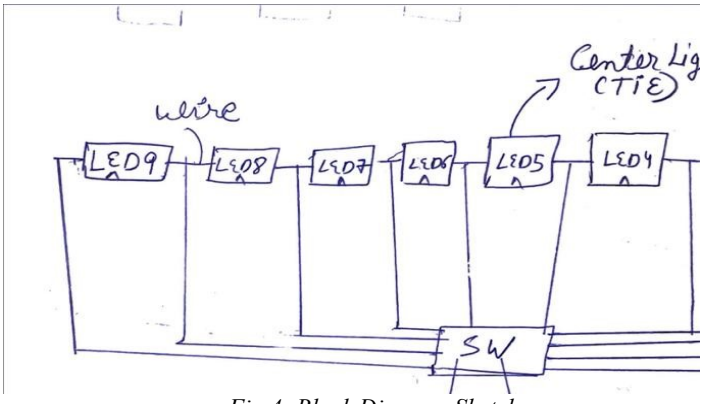

Fig 4: Block Diagram Sketch

From the Circuit design of View, it is configured into 4 sub modules, Center Light, Normal Light, Button and Hex display. The Basic Block Diagram Sketch of the Board is as below. It shows the Center light, Normal light and the wire we have used. It also marks how a Metastability state is implemented.

The second State diagram Is for Normal state Diagram and to show how it will turn ON and OFF. It marks the number of combinations required to switch around the states

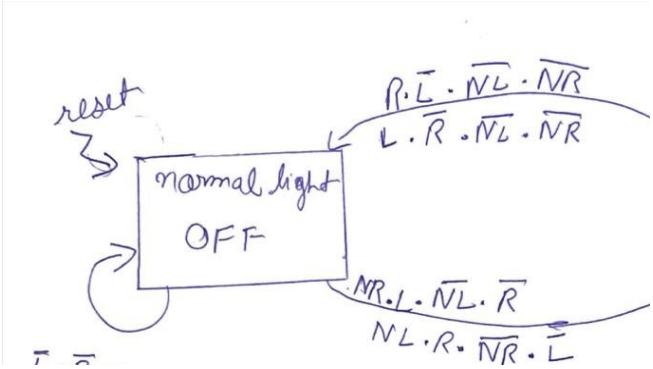

Fig 5: State Diagram for Normal Light

Similarly, the third state Diagram is the Combination for the Center Light. Which is marked as below.

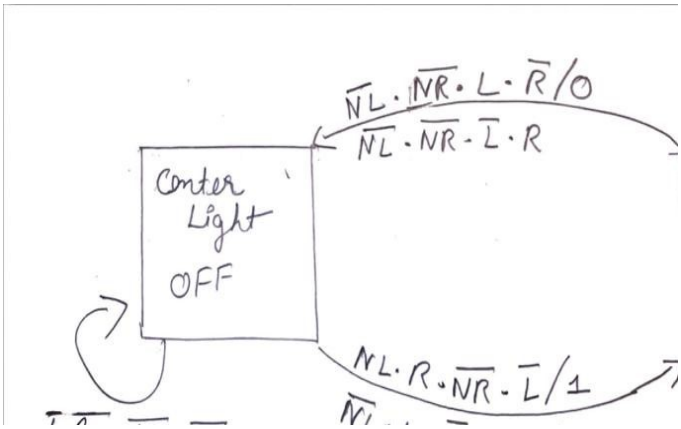

Fig 6: State Diagram for Center Light

Below is the Screenshot of the Center Light simulation run on MultiSim. It shows the pattern of Light of go high.

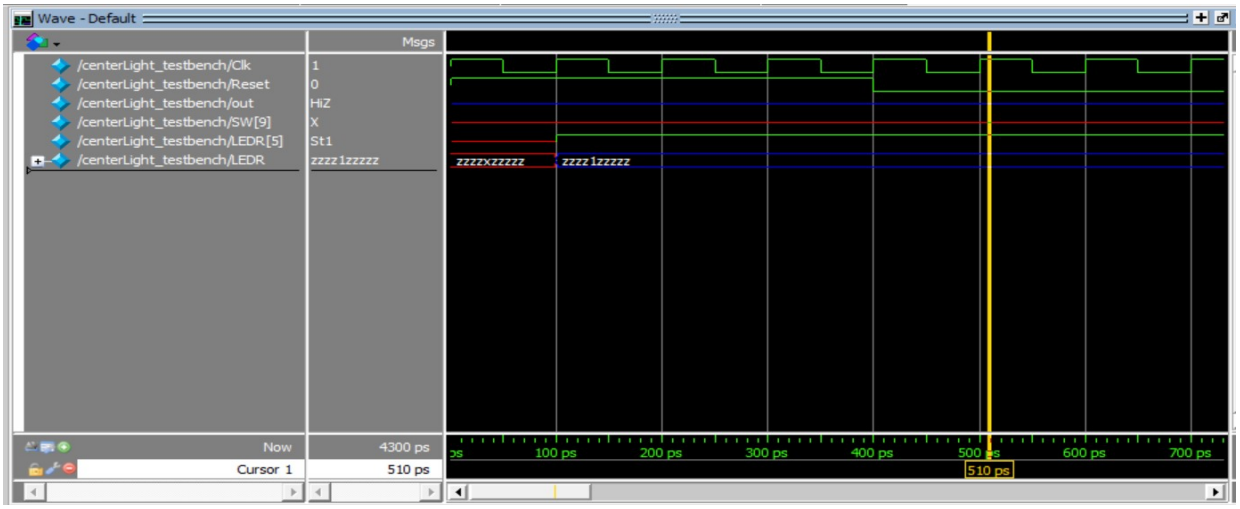

Fig7. Test Bench for Center Light

Below is the Screenshot of the Normal Light simulation run on MultiSim.

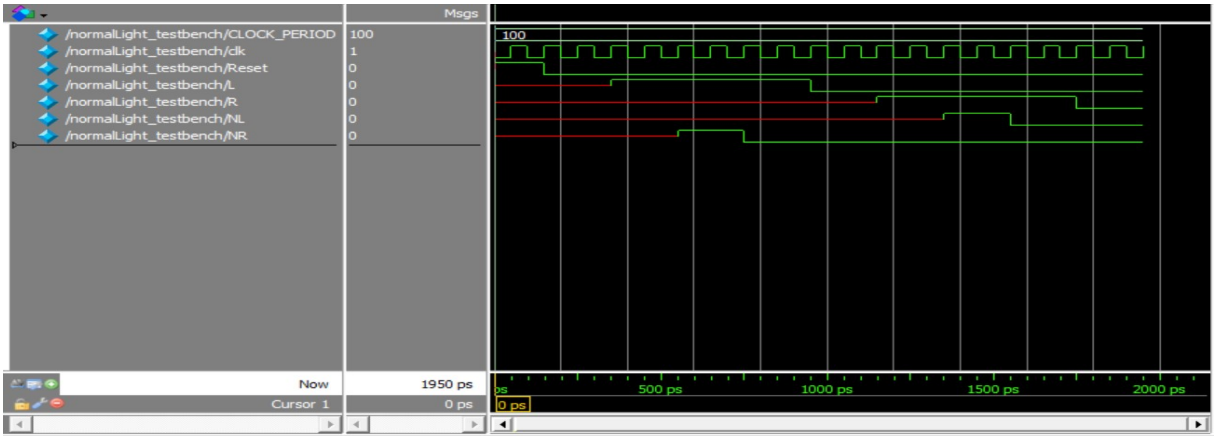

Fig8. Test Bench for Normal Light

The Top Level Entity screenshot on DE1_SoC is attached below. The below shot shows eight normal lights from LEDR9 to LEDR6 and then from LEDR4 to LEDR1 in reference to Center light LEDR5.

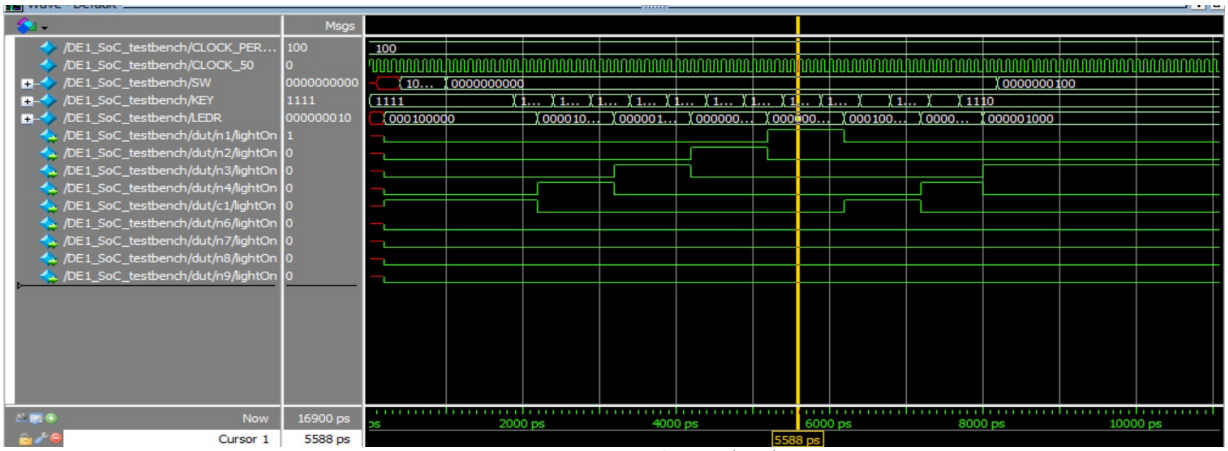

Fig9. Top-level

As a part of Deliverables, the Resource Utilization by Entity for the Tug of War project is configured as:

Analysis & Synthesis Resource Utilization by Entity

<<Filter>>

	Compilation Hierarchy Node	Combinational ALUTs	Dedicated Logic Registers	Block Memory Bits	DSP Blocks	Pins	Virtual Pins	Full Hierarchy Name	Entity Name
1	|DE1_SoC	18 (0)	16 (0)	0	0	67	0	|DE1_SoC	DE1_SoC
1	|button:player1|	3 (3)	2 (2)	0	0	0	0	|DE1_SoC|button:player1	button
2	|button:player2|	3 (3)	2 (2)	0	0	0	0	|DE1_SoC|button:player2	button
3	|centerLight:c1|	1 (1)	1 (1)	0	0	0	0	|DE1_SoC|centerLight:c1	centerLight
4	|hexdisplay:disp|	3 (3)	3 (3)	0	0	0	0	|DE1_SoC|hexdisplay:disp	hexdisplay
5	|normalLight:n1|	1 (1)	1 (1)	0	0	0	0	|DE1_SoC|normalLight:n1	normalLight
6	|normalLight:n2|	1 (1)	1 (1)	0	0	0	0	|DE1_SoC|normalLight:n2	normalLight
7	|normalLight:n3|	1 (1)	1 (1)	0	0	0	0	|DE1_SoC|normalLight:n3	normalLight
8	|normalLight:n4|	1 (1)	1 (1)	0	0	0	0	|DE1_SoC|normalLight:n4	normalLight
9	|normalLight:n6|	1 (1)	1 (1)	0	0	0	0	|DE1_SoC|normalLight:n6	normalLight
10	|normalLight:n7|	1 (1)	1 (1)	0	0	0	0	|DE1_SoC|normalLight:n7	normalLight
11	|normalLight:n8|	1 (1)	1 (1)	0	0	0	0	|DE1_SoC|normalLight:n8	normalLight
12	|normalLight:n9|	1 (1)	1 (1)	0	0	0	0	|DE1_SoC|normalLight:n9	normalLight

Fig10: Resource Utilization by Entity

In addition to this, the working video of runway lights circuit working in DE1_SoC board. Showing the behavior of all three wind conditions is available in the Media folder of folder Lab2- 2.1.

3.1 LFSRs

LFSR's well know for Linear feedback Shift Registers. Are used here to generate the random sequence.

Though we have used 10 big LFSR for the project but as a part of Deliverables, the truth table of the 4-bit LFSR is as below.

Clock Pulse	Q_1	Q_2
1	0	0
2	1	0
3	0	1
4	0	0
5	1	0
6	1	1
7	0	1
8	1	0
9	0	1
10	1	0
11	1	1

Table 1: 4 bit LFSR

In the Table we have used the XOR gate in combination to 4 D FLIP FLOPS, the input of the D1 is the XOR of and D3 and D4. Since D-FF's output is equal to the next's input, so it does act as a Shift Register as well.

This project consists of 3 Parts which are further divided into to 3-bit Up Counter, 10-bit LFSR's and 10 Bit Comparator. The Recorded waveform from their test benches are as below.

Below attached is the 3-bit Counter's test bench run on MultiSim.

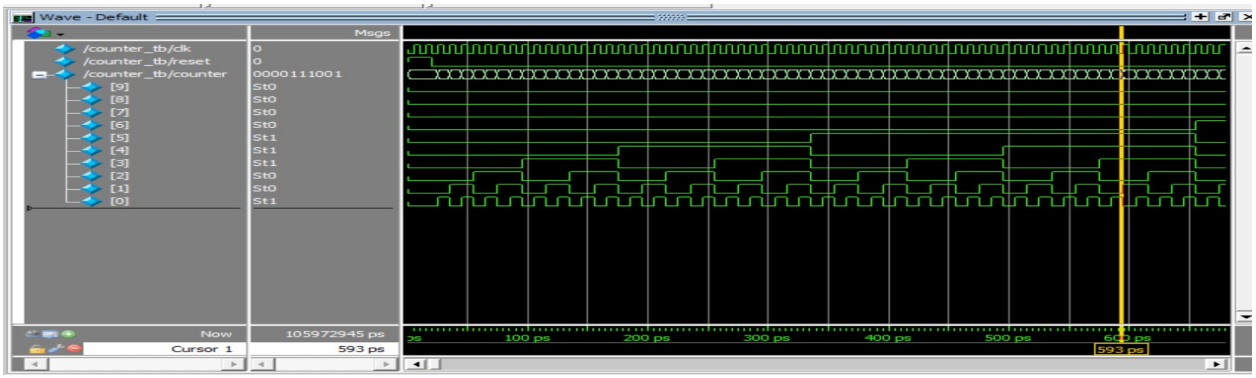

Fig 11: Counter

10-Bit Comparator used in the programming and its test bench on Multisim is as below.

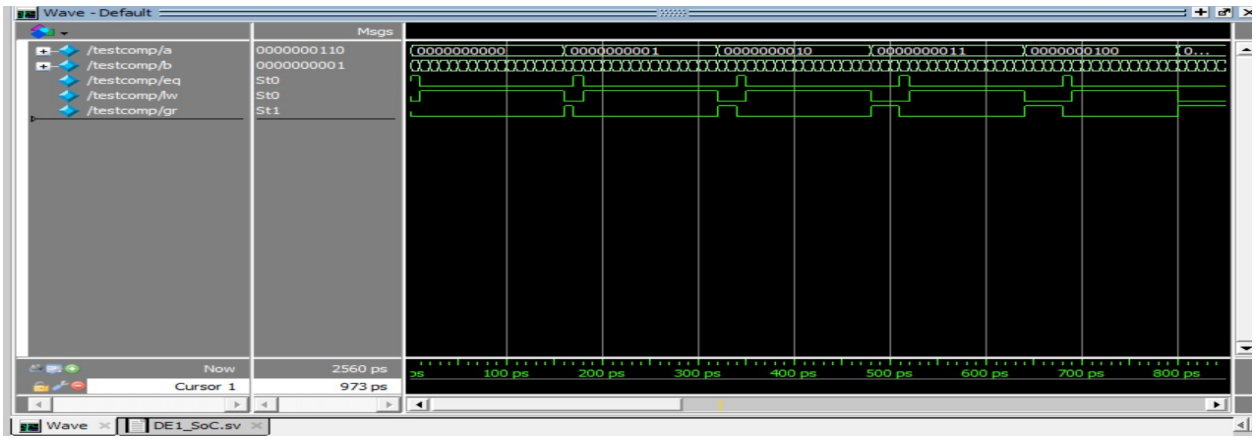

Fig 12: Comparator

Similarly, LFSR's Test Bench is as below

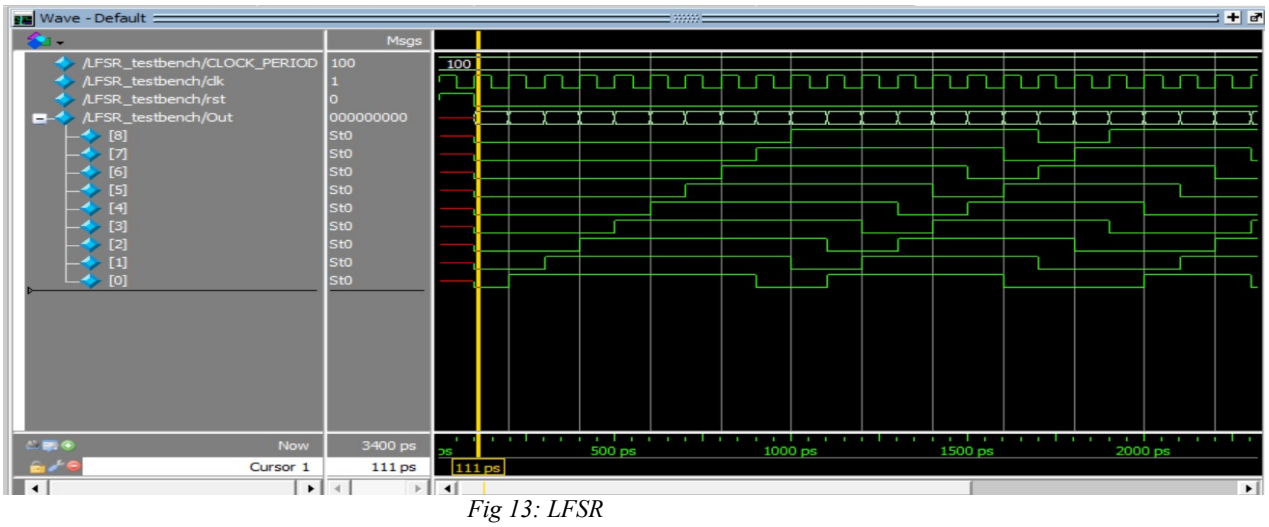

Fig 13: LFSR

This is the Top-Level Entity of the program and its test bench on MultiSim.

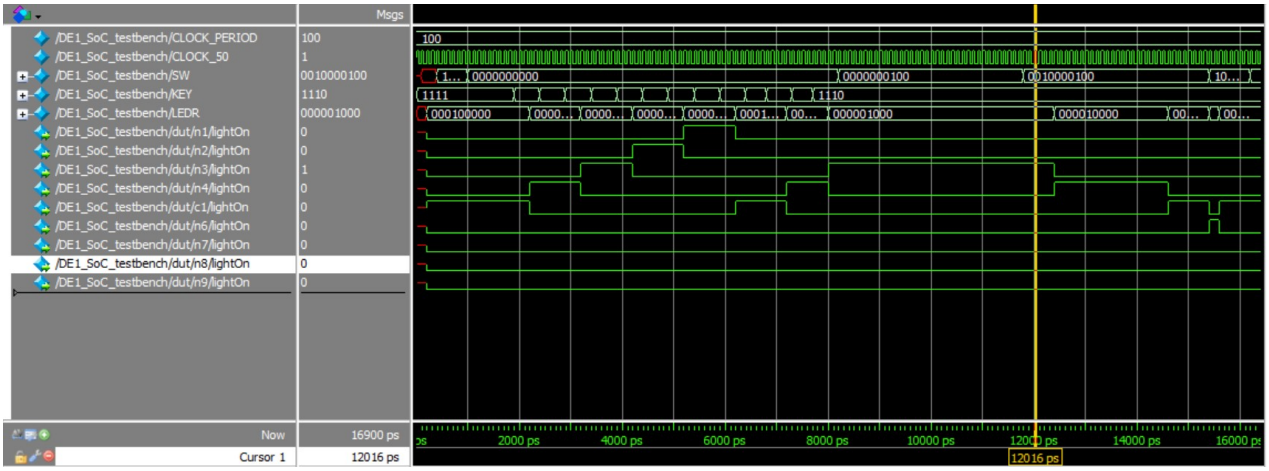

Fig 14: Top Level Entity

As of part of Deliverables, the screenshot of the Resource Utilization by Entity is as below:

Analysis & Synthesis Resource Utilization by Entity

<<Filter>>

	Compilation Hierarchy Node	Combinational ALUTs	Dedicated Logic Registers	Block Memory Bits	DSP Blocks	Pins	Virtual Pins	Full Hierarchy Name	Entity Name
1	|DE1_SoC	48 (0)	35 (0)	0	0	39	0	|DE1_SoC	DE1_SoC
1	|LFSR:RNG|	1 (1)	9 (9)	0	0	0	0	|DE1_SoC|LFSR:RNG	LFSR
2	|button:computer|	3 (3)	2 (2)	0	0	0	0	|DE1_SoC|button:computer	button
3	|button:humanPlayer|	3 (3)	2 (2)	0	0	0	0	|DE1_SoC|button:humanPlayer	button
4	|centerLight:c1|	3 (3)	1 (1)	0	0	0	0	|DE1_SoC|centerLight:c1	centerLight
5	|clock_divider:cdiv|	1 (1)	1 (1)	0	0	0	0	|DE1_SoC|clock_divider:cdiv	clock_divider
6	> |counter:leftCount|	4 (2)	6 (6)	0	0	0	0	|DE1_SoC|counter:leftCount	counter
7	> |counter:rightCount|	4 (2)	6 (6)	0	0	0	0	|DE1_SoC|counter:rightCount	counter
8	|hexdisplay:disp|	14 (14)	0 (0)	0	0	0	0	|DE1_SoC|hexdisplay:disp	hexdisplay
9	|normalLight:n1|	1 (1)	1 (1)	0	0	0	0	|DE1_SoC|normalLight:n1	normalLight
10	|normalLight:n2|	1 (1)	1 (1)	0	0	0	0	|DE1_SoC|normalLight:n2	normalLight
11	|normalLight:n3|	1 (1)	1 (1)	0	0	0	0	|DE1_SoC|normalLight:n3	normalLight
12	|normalLight:n4|	1 (1)	1 (1)	0	0	0	0	|DE1_SoC|normalLight:n4	normalLight
13	|normalLight:n6|	1 (1)	1 (1)	0	0	0	0	|DE1_SoC|normalLight:n6	normalLight
14	|normalLight:n7|	1 (1)	1 (1)	0	0	0	0	|DE1_SoC|normalLight:n7	normalLight
15	|normalLight:n8|	1 (1)	1 (1)	0	0	0	0	|DE1_SoC|normalLight:n8	normalLight
16	|normalLight:n9|	1 (1)	1 (1)	0	0	0	0	|DE1_SoC|normalLight:n9	normalLight
17	> |tenBitAdder:together|	7 (0)	0 (0)	0	0	0	0	|DE1_SoC|tenBitAdder:together	tenBitAdder

Figure 15: Resource Utilization by Entity

In addition to this, the working video of runway lights circuit working in DE1_SoC board. Showing the behavior of all three wind conditions is available in the Media folder of folder Lab2- 3.1.