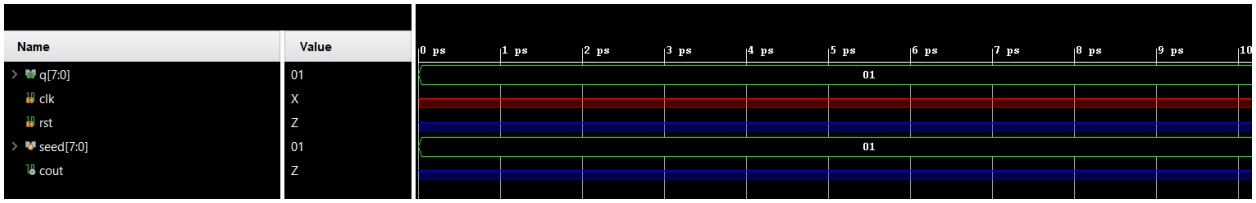
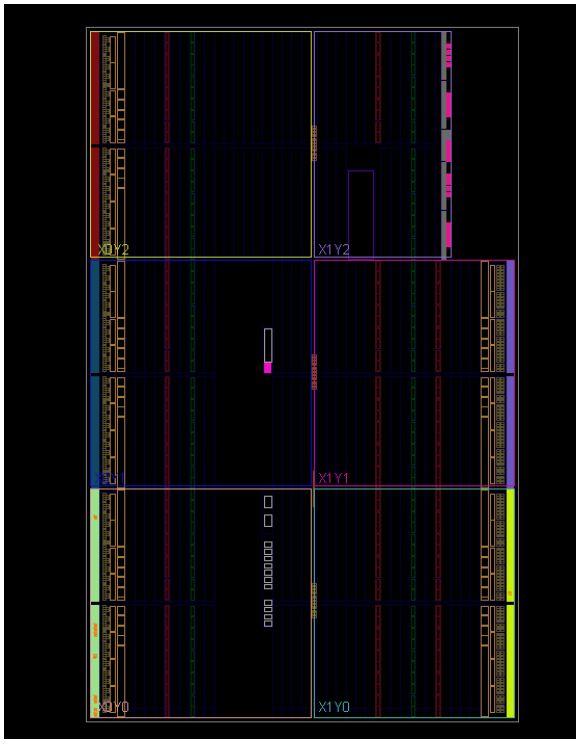
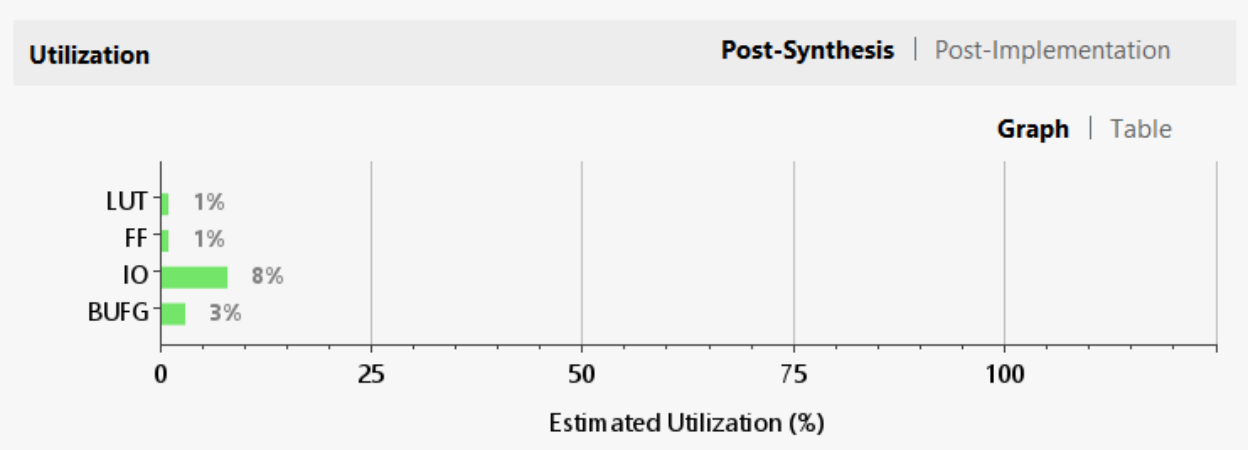
Vincent Han

Lab 4 Notebook

1. In this lab we are making a random number generator that displays onto a 7 segment display. We first created a clock divider, then a LFSR module. We then used both into a singular module called lab4_top which allows us to connect it to the seven segment display.
2. Without the seven segment, I wasn't able to test it since the only outputs it gave me were 0s.
3. I needed to add the .v file that was provided to use in the lab4_top module, in order to be able to display anything to the display.
4. 

5. 

6. 

7. **Design Timing Summary**

Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	inf	Worst Hold Slack (WHS):	inf	Worst Pulse Width Slack (WPWS):	NA
Total Negative Slack (TNS):	0.000 ns	Total Hold Slack (THS):	0.000 ns	Total Pulse Width Negative Slack (TPWS):	NA
Number of Failing Endpoints:	0	Number of Failing Endpoints:	0	Number of Failing Endpoints:	NA
Total Number of Endpoints:	16	Total Number of Endpoints:	16	Total Number of Endpoints:	NA

There are no user specified timing constraints.

8. Design goals were not met, I wasn't able to process the implementation without coming across errors.
9. I was unable to add in the seven segment display properly as well as the test bench file.
10. I would need to develop a better testbench file in order to run the implementation.

Source Code:

LFSR.v

```
module lfsr (output reg [7:0]q, input [7:0]seed, input rst, input clock);

    wire din;

    assign din = q[1]^q[2]^q[3]^q[7];

    initial
    begin
        q <= 8'b00000001;
    end

    always @(posedge clock)
    begin
        if(rst == 1)
        begin
            q <= seed;
        end
        else if (rst == 0)
        begin
```

```
            q[7:1] <= q[6:0];
            q[0] <= din;
        end
    end
endmodule
```

LAB4_TOP.V

```
module lab4_top(output [7:0] q,
input clk,
input rst
);
    reg [7:0] seed = 8'b00000001;
    wire cout;
    clock_divider CDIV (clk,cout);
    lfsr LFSR (q,seed,cout,rst);
endmodule
```

LFSR_TB.V

```
module lfsr_tb;
  reg [7:0] seed;
  reg clock, rst;
  wire [7:0] q;

  lfsr dut (
    .q(q),
    .seed(seed),
    .clock(clock),
    .rst(rst)
  );

  always #5 clock = ~clock;

  initial begin
    rst = 1;
    #10 rst = 0;
  end

  initial begin
    seed = 8'hFF;
    #20;

    repeat (256) begin
      #10;
      $display("q = %h", q);
    end;

    // End simulation
    $finish;
  end
Endmodule
```

Lab4_top.xdc

```
# contraints file (.xdc file) for lab 4
# Clock signal
set_property PACKAGE_PIN W5 [get_ports clk]
    set_property IOSTANDARD LVCMOS33 [get_ports clk]

#create_clock -add -name sys_clk_pin -period 10.00 -waveform {0 5} [get_port sclk]

# leds
set_property PACKAGE_PIN U16 [get_ports {q[0]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {q[0]}]
set_property PACKAGE_PIN E19 [get_ports {q[1]}]
```

```
    set_property IOSTANDARD LVCMOS33 [get_ports {q[1]}]
set_property PACKAGE_PIN U19 [get_ports {q[2]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {q[2]}]
set_property PACKAGE_PIN V19 [get_ports {q[3]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {q[3]}]
set_property PACKAGE_PIN W18 [get_ports {q[4]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {q[4]}]
set_property PACKAGE_PIN U15 [get_ports {q[5]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {q[5]}]
set_property PACKAGE_PIN U14 [get_ports {q[6]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {q[6]}]
set_property PACKAGE_PIN V14 [get_ports {q[7]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {q[7]}]

# Display
set_property PACKAGE_PIN W7 [get_ports {disp[0]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {disp[0]}]
set_property PACKAGE_PIN W6 [get_ports {disp[1]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {disp[1]}]
set_property PACKAGE_PIN U8 [get_ports {disp[2]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {disp[2]}]
set_property PACKAGE_PIN V8 [get_ports {disp[3]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {disp[3]}]
set_property PACKAGE_PIN U5 [get_ports {disp[4]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {disp[4]}]
set_property PACKAGE_PIN V5 [get_ports {disp[5]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {disp[5]}]
set_property PACKAGE_PIN U7 [get_ports {disp[6]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {disp[6]}]

set_property PACKAGE_PIN U2 [get_ports {anode[0]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {anode[0]}]
set_property PACKAGE_PIN U4 [get_ports {anode[1]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {anode[1]}]
set_property PACKAGE_PIN V4 [get_ports {anode[2]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {anode[2]}]
set_property PACKAGE_PIN W4 [get_ports {anode[3]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {anode[3]}]

# Switches
set_property PACKAGE_PIN V17 [get_ports {seed[0]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {seed[0]}]
set_property PACKAGE_PIN V16 [get_ports {seed[1]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {seed[1]}]
set_property PACKAGE_PIN W16 [get_ports {seed[2]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {seed[2]}]
set_property PACKAGE_PIN W17 [get_ports {seed[3]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {seed[3]}]
set_property PACKAGE_PIN W15 [get_ports {seed[4]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {seed[4]}]
set_property PACKAGE_PIN V15 [get_ports {seed[5]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {seed[5]}]
set_property PACKAGE_PIN W14 [get_ports {seed[6]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {seed[6]}]
set_property PACKAGE_PIN W13 [get_ports {seed[7]}]
    set_property IOSTANDARD LVCMOS33 [get_ports {seed[7]}]

#Buttons
set_property PACKAGE_PIN U18 [get_ports rst]
    set_property IOSTANDARD LVCMOS33 [get_ports rst]
set_property PACKAGE_PIN W19 [get_ports roll]
    set_property IOSTANDARD LVCMOS33 [get_ports roll]
```