NATIONAL UNIVERSITY HO CHI MINH CITY

UNIVERSITY OF TECHNOLOGY – FACULTY OFCOMPUTER SCIENCE AND ENGINEERING

REPORT LAB 2
LOGIC DESIGN WITH HDL
ADVISOR: NGUYỄN THẾ BÌNH
TEAM 8

Name:	Students's ID
Phạm Anh Tài	2350023
Phạm Ngọc Huy	2352404
Trần Đình Duy Khương	2352638
Lục Tấn Phúc	2352935

I. Exercise:

2.1 Exercise 1 Design a circuit that has one 4-bit input and 4-bit output with functions as follow (all outputs are active HIGH):

• Output 0: active when there are even number of bit 1 in the input.
• Output 1: active when there are only 1 bit 1 in the input.
• Output 2: active when there is no bit 1 in the input.
• Output 3: active when all bit in the input are 1. Write Verilog HDL RTL code and test bench for the design. Test the circuit on FPGA using buttons and LEDs. Hint: Students should use reduction operators.

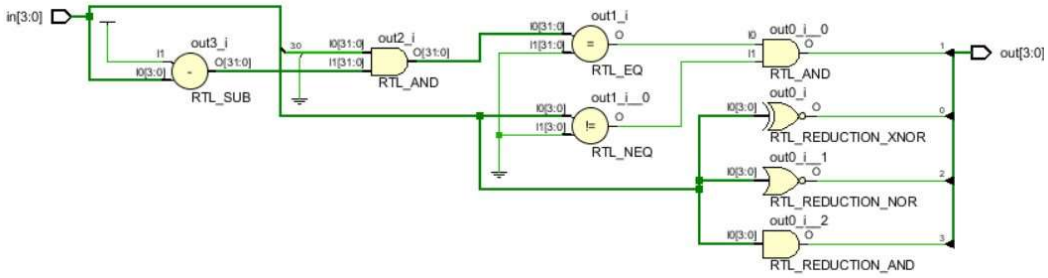
-Module code:

```
module lab2_ex1(input [3:0] in, output[3:0] out );

assign out[0] = ~^in;
assign out[1] = ((in[3:0] & (in[3:0] - 1)) == 0) && (in[3:0] != 0);
assign out[2] = ~|in;
assign out[3] = &in;

endmodule
```

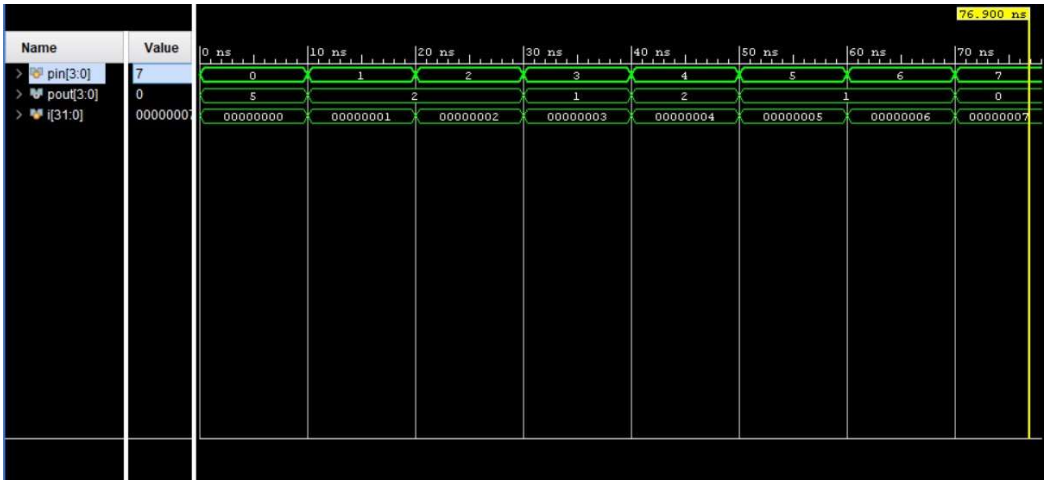
-Testbench code:

```
module tb_lab2_ex1;
reg [3:0] pin;
wire [3:0] pout;
integer i;
lab2_ex1 test(.in(pin), .out(pout));

initial
begin
    for(i=0;i<15;i=i+1)
        begin
            pin=i;
            #10;
        end
        $finish;
end

endmodule
```

-RTL analysis:



-Simulation:



2.2 Exercise 2 Design a circuit that generate a 1 Hz output signal using behavioral model. This signal connect to LED on Arty-Z7 FPGA Board to make it blink (turn on for 0.5s - turn off for 0.5s). Know that the input clock frequency is 125 MHz. Write test benches to simulate the circuits. Test the circuits on FPGA board using LEDs or RGB LED.

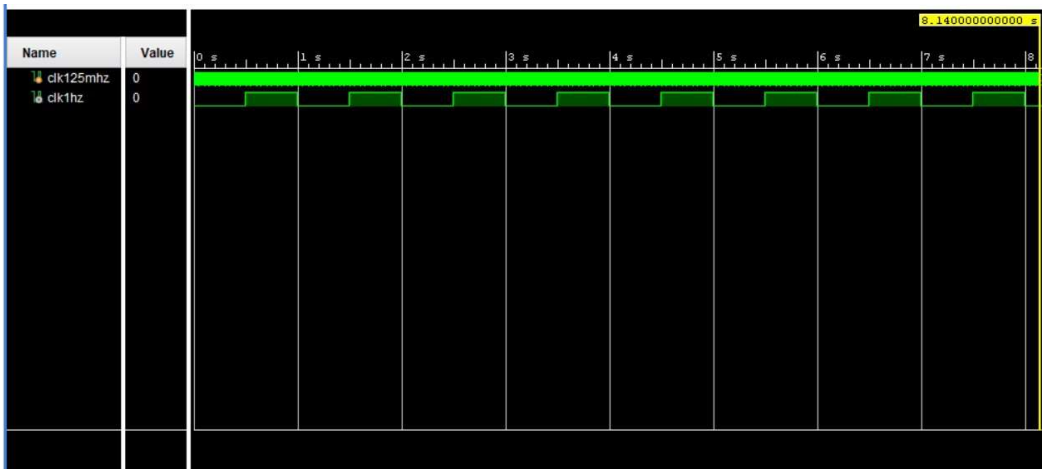
-Module code:

```
module lab2_ex2(
  input clk125mhz,
  output clk1hz);

reg [25:0] counter=0;

always @(posedge clk125mhz )
        if(counter == 1_250_000 -1)
        counter <= 0;
        else counter <= counter +1;

assign clk1hz = counter >=625_000;

endmodule
```

-Testbench code:

```
module tb_lab2_ex2;

reg clk125mhz;
wire clk1hz;

lab2_ex2 dut ( .clk125mhz(clk125mhz), .clk1hz(clk1hz));

initial
begin
clk125mhz=0;
    forever #400 clk125mhz=~clk125mhz;
end

endmodule
```

-Simulation:



2.3 Exercise 3 Design a 7-segment LED decoder which will accept a 4-bit input and generate a 7-bit output. • Interface: module bin2led7(enable, bin in, led out); • The enable signal controls LEDs. If enable = 0, LEDs are turned off. • The bin in is a 4-bit binary input signal. • The led out is a 7-bit output signal for 7-segment LED display.

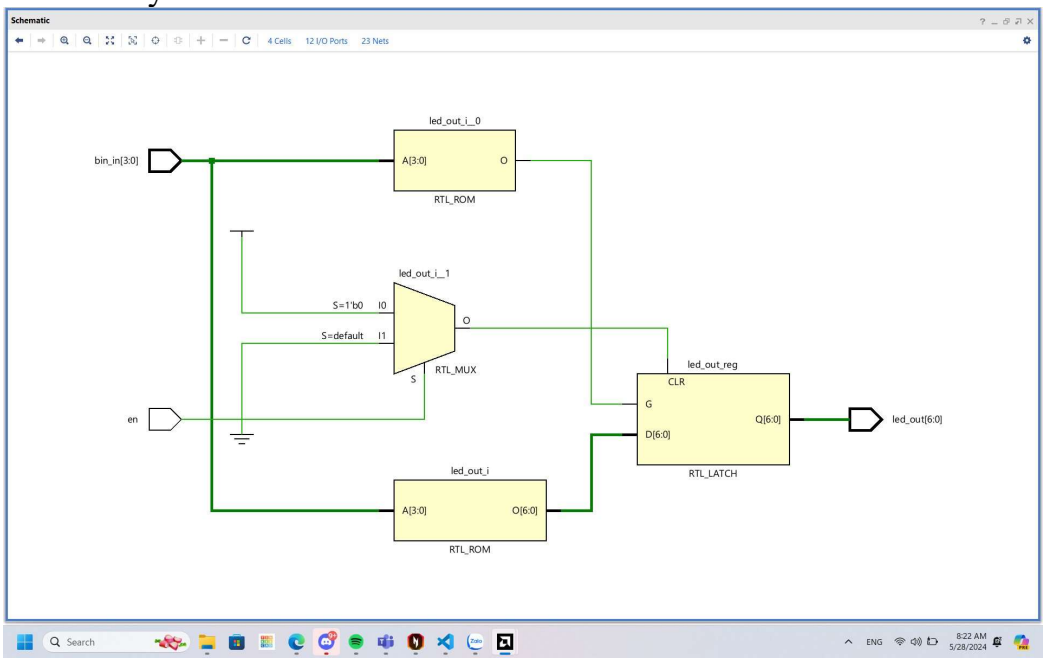
-Module code:

```
22
23  module lab2_ex3 (en, bin_in, led_out);
24
25  input [3:0] bin_in;
26  input en;
27  output reg [6:0] led_out;
28  always @*
29  begin
30  if(en==1'b0) led_out <= 7'd0;
31  else
32      case(bin_in)
33          4'd0: led_out <= 7'd0;
34          4'd1: led_out <= 7'b0110000;
35          4'd2: led_out <= 7'b1101101;
36          4'd3: led_out <= 7'b1111001;
37          4'd4: led_out <= 7'b0110011;
38          4'd5: led_out <= 7'b1011011;
39          4'd6: led_out <= 7'b1011111;
40          4'd7: led_out <= 7'b1110000;
41          4'd8: led_out <= 7'b1111111;
42          4'd9: led_out <= 7'b1111011;
43      endcase
44  end
45
46  endmodule
47
```

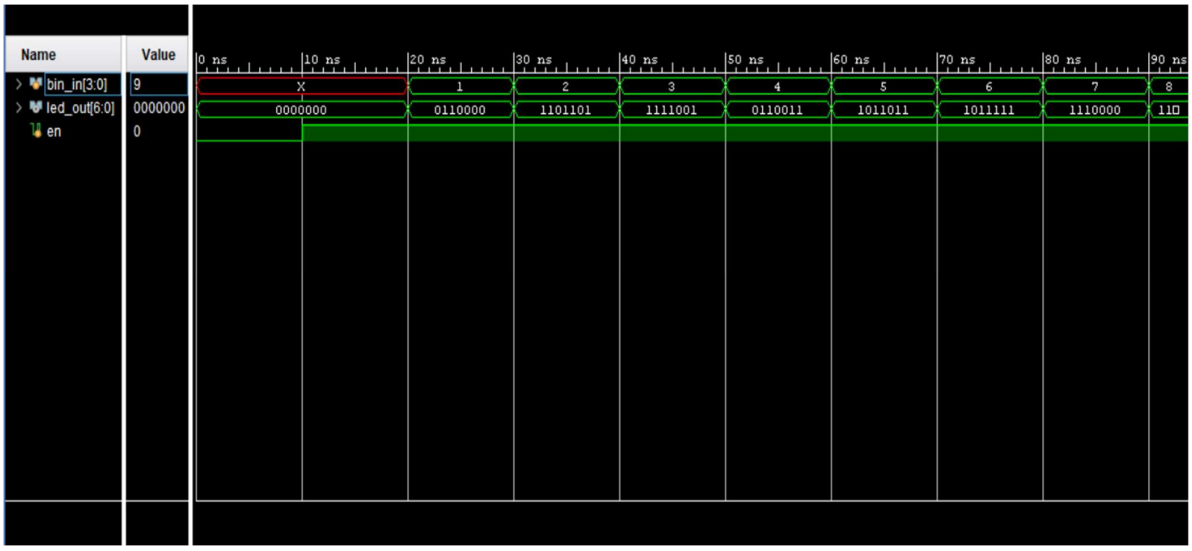
-Testbench code:

```
22
23  module tb_lab2_ex3;
24  reg [3:0] bin_in;
25  wire [6:0] led_out;
26  reg en;
27
28  lab2_ex3 uut (.bin_in(bin_in), .en(en), .led_out(led_out));
29
30  initial begin
31  en = 1'b0;
32  #10 en = 1'b1;
33  #10 bin_in <= 4'd1;
34  #10 bin_in <= 4'd2;
35  #10 bin_in <= 4'd3;
36  #10 bin_in <= 4'd4;
37  #10 bin_in <= 4'd5;
38  #10 bin_in <= 4'd6;
39  #10 bin_in <= 4'd7;
40  #10 bin_in <= 4'd8;
41  #10 bin_in <= 4'd9;
42
43  #10 en = 4'd0;
44  #10 bin_in <= 4'd1;
45  #20;
46  $finish;
47  end
48  endmodule
49
```

-RTL analysis:



-Simulation:



2.4 Exercise 4 Design a circuit to control the RGB LEDs on Arty-Z7 board as follow: • Switch 0: select display mode - 1 LED or 2 LEDs. • Switch 1:
– to select left or right LED in 1-LED display mode.
– to select color code as table below.
Write Verilog HDL RTL code and test bench for the design. Test the circuit on FPGA using switches, buttons and RGB LEDs on the Arty-Z7 board.

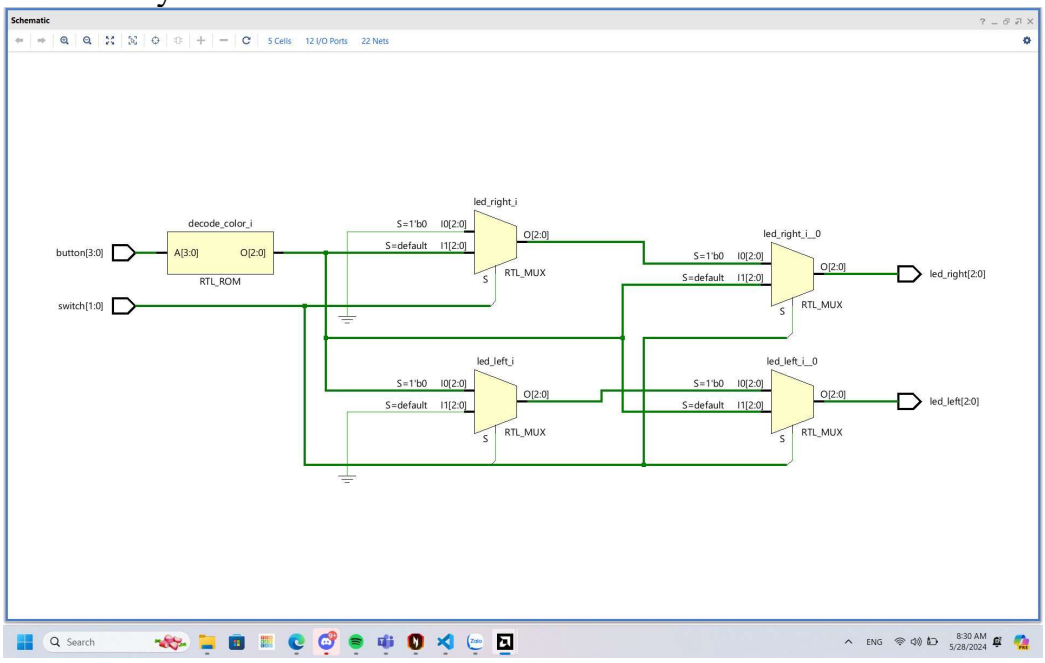
-Module code:

```
`timescale 1ns / 1ps
module rgb_led(bcd,mode,select,led);
 input [3:0] bcd;
 input mode; // sw0
 input select; // sw1
 output reg [5:0] led;

always @(bcd)
    begin
        if(mode == 1'b0)
            begin
            if(select == 1'b0)
              begin
              case (bcd) //case statement
                           0 : led = 6'b000000;
                           1 : led = 6'b000001;
                           2 : led = 6'b000010;
                           4 : led = 6'b000100;
                           8 : led = 6'b000111;
                           default : led = 6'b000000;
              endcase
    end
              else
               begin
               case (bcd) //case statement
                           0 : led = 6'b000000;
                           1 : led = 6'b001000;
                           2 : led = 6'b010000;
                           4 : led = 6'b100000;
                           8 : led = 6'b111000;
                           default : led = 6'b000000;
               endcase
              end
            end
         else
           begin
  if(select == 1'b0)
if(select == 1'b0)
              begin
               case (bcd) //case statement
                          0 : led = 6'b000000;
                          1 : led = 6'b001001;
                          2 : led = 6'b010010;
                          4 : led = 6'b100100;
                          8 : led = 6'b111111;
default : led = 6'b000000;
               endcase
             end
            else
              begin
               case (bcd) //case statement
                         0 : led = 6'b000000;
                         1 : led = 6'b011011;
                         2 : led = 6'b110110;
                         4 : led = 6'b101101;
                         8 : led = 6'b111111;
                         default : led = 6'b011011;
                endcase
             end
         end
    end
endmodule
```

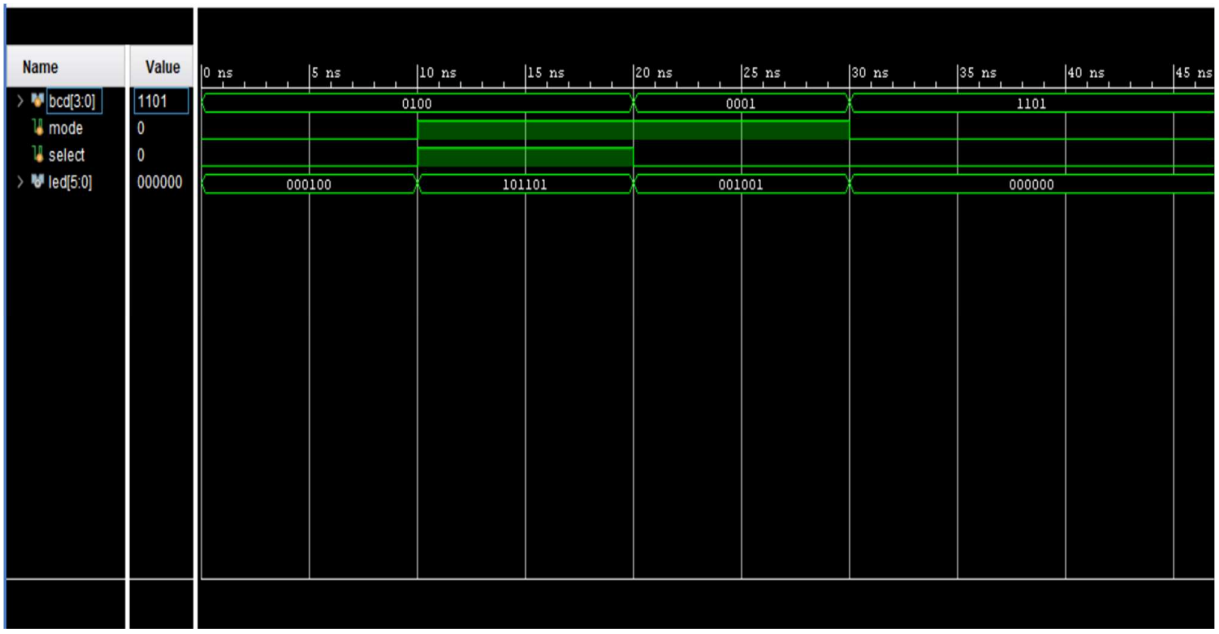
-Testbench code:

```
////////////////////////////////////////////////////////////////////////////////
module rgb_led_tb;
    reg [3:0] bcd;
    reg mode,select;
    wire [5:0] led;

   rgb_led UUT(bcd,mode,select,led);
    initial $monitor("%time: led=%b",$time,led);
    initial begin
    bcd=4'b0100; mode=1'b0; select=1'b0;
    #10
    bcd=4'b0100; mode=1'b1; select=1'b1;
    #10
    bcd=4'b0001; mode=1'b1; select=1'b0;
    #10
    bcd=4'b1101; mode=1'b0; select=1'b0;
    end
endmodule
```

- RTL analysis:



-Simulation:


Name
Value
bcd[3:0]
1101
mode
0
select
0
led[5:0]
000000
0 ns
5 ns
10 ns
15 ns
20 ns
25 ns
30 ns
35 ns
40 ns
45 ns
0100
0001
1101
000100
101101
001001
000000