Lab4

Submitted By:
Snehal Gupta
2016201

//top.v

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 26.09.2017 17:01:34
// Design Name:
// Module Name: bcd_adder
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////


module top(
    input clk,
    output [3:0] anode,
    output [7:0] cathode
    );
```

```
    wire [7:0] binary_count;
    wire [3:0] hundreds,tens,ones;
    wire clock_fd,clk_in;
    divideby10 f0(clk,clk_in);//onehz clock
    counter f2(clk_in,binary_count);//counter
    binary2bcd f3(binary_count,hundreds,tens,ones);//binary2bcd
    disp_sevenseg f4(clk,ones,tens,hundreds,anode,cathode);//seven
endmodule
```

//divideby10.v

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 26.09.2017 17:01:34
// Design Name:
// Module Name: bcd_adder
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////


module divideby10(
```

```
    input clkin,
    output reg clkout
    );
reg [26:0] counter;
    initial begin
        counter = 0;
        clkout = 0;
    end

    always @(posedge clkin) begin //onehz frequency
        if (counter == 50000000) begin
            counter <= 0;//onehalf
            clkout <= ~clkout;
        end else begin
            counter <= counter +1;
        end
    end
endmodule
```

//counter.v

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 26.09.2017 17:01:34
// Design Name:
// Module Name: bcd_adder
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
```

```
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////


module counter    (
  input clk,
  output reg[7:0] out
  );
always @(posedge clk)
begin
out <= out+1;//increment counter
end
endmodule
```

//binary2bcd.v

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 26.09.2017 17:01:34
// Design Name:
// Module Name: bcd_adder
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
```

```
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////


module binary2bcd(
    number,hundreds,tens,ones
    );
    input[7:0] number;
    output reg [3:0]hundreds;
    output reg [3:0]tens;
    output reg [3:0]ones;
    // double dabble algorithm
    reg[19:0] shift;
    integer  i;
    always @(number)
    begin
        shift[19:8] =0;
        shift[7:0] = number;
        for (i=0;i<8;i=i+1)
        begin
            if(shift[11:8] >= 5)
                shift[11:8] = shift[11:8] + 3;
            if(shift[15:12] >= 5)
                shift[15:12] = shift[15:12] + 3;
            if(shift[19:16] >= 5)
                shift[19:16] = shift[19:16] + 3;
            shift = shift <<1;
         end
```

```
        hundreds = shift[19:16];
        tens     = shift[15:12];
        ones     = shift[11:8];
    end



endmodule
```

//disp_sevenseg.v

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 26.09.2017 17:01:34
// Design Name:
// Module Name: bcd_adder
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////
```

```
module disp_sevenseg(
    input clock_in,
    input [3:0] ones,
    input [3:0] tens,
    input [3:0] hundreds,
    output [3:0] anode,
    output [7:0] cathode
    );
    // template
    reg [17:0] count;
    always@(posedge clock_in)
    begin
    count <= count+1;
    end
    reg [6:0] sseg_temp;
    reg [6:0]sseg;
    reg [3:0] an_temp = 4'b1110;
    always @ (*)
     begin
      case(count[17:16])
       2'b00 :
        begin
         sseg = ones;
         an_temp = 4'b1110;
        end
       2'b01:
        begin
         sseg = tens;
         an_temp = 4'b1101;
        end
       2'b10:
        begin
         sseg = hundreds;
```

```
        an_temp = 4'b1011;
       end
      2'b11:
       begin
        sseg = hundreds;
        an_temp = 4'b1011;
       end
     endcase
    end
   assign anode = an_temp;
   always @ (*)
    begin
     case(sseg)
      4'd0 : sseg_temp = 7'b0000001;
      4'd1 : sseg_temp = 7'b1001111;
      4'd2 : sseg_temp = 7'b0010010;
      4'd3 : sseg_temp = 7'b0000110;
      4'd4 : sseg_temp = 7'b1001100;
      4'd5 : sseg_temp = 7'b0100100;
      4'd6 : sseg_temp = 7'b0100000;
      4'd7 : sseg_temp = 7'b0001111;
      4'd8 : sseg_temp = 7'b0000000;
      4'd9 : sseg_temp = 7'b0000100;
      default : sseg_temp = 7'b1111110;
     endcase
    end
    assign cathode = {sseg_temp,1'b1};
endmodule
```

//const.xdc

```
set_property PACKAGE_PIN W5 [get_ports {clk}]
set_property IOSTANDARD LVCMOS33 [get_ports {clk}]
```

```
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]}]

set_property PACKAGE_PIN W7 [get_ports {cathode[7]}]
set_property IOSTANDARD LVCMOS33 [get_ports {cathode[7]}]
set_property PACKAGE_PIN W6 [get_ports {cathode[6]}]
set_property IOSTANDARD LVCMOS33 [get_ports {cathode[6]}]
set_property PACKAGE_PIN U8 [get_ports {cathode[5]}]
set_property IOSTANDARD LVCMOS33 [get_ports {cathode[5]}]
set_property PACKAGE_PIN V8 [get_ports {cathode[4]}]
set_property IOSTANDARD LVCMOS33 [get_ports {cathode[4]}]
set_property PACKAGE_PIN U5 [get_ports {cathode[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {cathode[3]}]
set_property PACKAGE_PIN V5 [get_ports {cathode[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {cathode[2]}]
set_property PACKAGE_PIN U7 [get_ports {cathode[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {cathode[1]}]
set_property PACKAGE_PIN V7 [get_ports {cathode[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {cathode[0]}]
```