Lab5

Submitted By:
Snehal Gupta
2016201

//Question 4

//top.v

```
`timescale 1ns / 1ps
module top(
input clk,
input clr,
input [1:0] btn,
output [3:0] anode,
output [7:0] cathode);
wire [7:0] binary_count;
wire [3:0] hundreds,tens,ones;
wire inp;
assign inp = btn[0] || btn[1];
clk_div CD1 (.mclk(clk),.clr(clr),.clk190(clk_190));
clock_pulse CP(.inp(inp),.cclk(clk_190),.clr(clr),.outp(out_pulse));
sipo S1(.clk(out_pulse),.rst(clr),.a(btn[1]),.q(binary_count));
binary2bcd f3(binary_count,hundreds,tens,ones);
disp_sevenseg f4(clk,ones,tens,hundreds,anode,cathode);
endmodule
```

//clk_div.v

```
`timescale 1ns / 1ps
module clk_div(
input wire mclk,
input wire clr,
output wire clk190);
reg [18:0] q;
```

```
always @(posedge mclk)
begin
if(clr == 1)
q <= 0;
else
q <= q+1;
end
assign clk190 = q[18];
endmodule
```

//clock_pulse.v

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 26.09.2017 18:04:34
// Design Name:
// Module Name: clock_pulse
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////
```

```
module clock_pulse(
input wire inp,
input wire cclk,
input wire clr,
output wire outp);
reg delay1;
reg delay2;
reg delay3;
always@(posedge cclk or posedge clr)
begin
if(clr == 1)
begin
delay1 <= 1'b0;
delay2 <= 1'b0;
delay3 <= 1'b0;
end
else
begin
delay1 <= inp;
delay2 <= delay1;
delay3 <= delay2;
end
end
assign outp = delay1 & delay2 & delay3;
endmodule
```

//sipo.v

```
`timescale 1ns / 1ps
module sipo(a,clk,rst,q);
input clk,rst;
input a;
output [7:0]q;
wire [7:0]q;
```

```
reg [7:0]temp;
always@(posedge clk,posedge rst)
begin
if(rst==1'b1)
temp<=8'b0000;
else
begin
temp <= temp<<1'b1;
temp[0]<=a;
end
end
assign q=temp;
endmodule
```

//binary2bcd.v

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 23.09.2017 15:15:42
// Design Name:
// Module Name: binary2bcd
// 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;

    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: 23.09.2017 02:17:04
// Design Name:
// Module Name: disp_sevenseg
// 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
);
reg [18:0] 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[18:17])
   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 U18 [get_ports {clr}]
set_property IOSTANDARD LVCMOS33 [get_ports {clr}]
set_property PACKAGE_PIN T18 [get_ports {btn[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {btn[0]}]
set_property PACKAGE_PIN U17 [get_ports {btn[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {btn[1]}]
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]}]
```

//Question 5

//top.v

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
```

```
// Create Date: 26.09.2017 23:13:19
// Design Name:
// Module Name: top
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////

module top(
input clock_in,
input enable,
input [3:0] addrA,
input [3:0] addrB,
output [3:0] anode,
output [7:0] cathode
    );
    wire [3:0] A,B;
    wire [7:0] sum;
    blk_mem_gen_0
romA(.clka(clock_in),.ena(enable),.addra(addrA),.douta(A));
    blk_mem_gen_0
romB(.clka(clock_in),.ena(enable),.addra(addrB),.douta(B));
    bcd_adder f2(A,B,sum);
    disp_sevenseg f1(clock_in,A,sum[3:0],sum[7:4],B,anode,cathode);
```

```
endmodule
```

//bcd_adder.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 bcd_adder(
    input [3:0] A,
    input [3:0] B,
    output [7:0] Sum
    );
    reg [7:0] reg_sum;
```

```
    always@(A,B)
    begin
    reg_sum = A+B;
    end

binary2bcd(.number(reg_sum),.hundreds(4'b0),.tens(Sum[7:4]),.ones(Sum[
3:0]));

endmodule
```

//binary2bcd.v

```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 23.09.2017 15:15:42
// Design Name:
// Module Name: binary2bcd
// 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;

    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: 23.09.2017 02:17:04
// Design Name:
// Module Name: disp_sevenseg
// 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,
    input [3:0] thousands,
    output [3:0] anode,
    output [7:0] cathode
```

```
);
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 = thousands;
     an_temp = 4'b0111;
    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
```

//brom.coe

```
memory_initialization_radix=2;
memory_initialization_vector=
0000,
0001,
0010,
0011,
0100,
0101,
0110,
0111,
1000,
```

1001;

//const.xdc

```
set_property PACKAGE_PIN W5 [get_ports clock_in]
set_property IOSTANDARD LVCMOS33 [get_ports clock_in]

set_property PACKAGE_PIN V2 [get_ports enable]
set_property IOSTANDARD LVCMOS33 [get_ports enable]


set_property PACKAGE_PIN W17 [get_ports addrA[3]]
set_property IOSTANDARD LVCMOS33 [get_ports addrA[3]]
set_property PACKAGE_PIN W16 [get_ports addrA[2]]
set_property IOSTANDARD LVCMOS33 [get_ports addrA[2]]
set_property PACKAGE_PIN V16 [get_ports addrA[1]]
set_property IOSTANDARD LVCMOS33 [get_ports addrA[1]]
set_property PACKAGE_PIN V17 [get_ports addrA[0]]
set_property IOSTANDARD LVCMOS33 [get_ports addrA[0]]

set_property PACKAGE_PIN W13 [get_ports addrB[3]]
set_property IOSTANDARD LVCMOS33 [get_ports addrB[3]]
set_property PACKAGE_PIN W14 [get_ports addrB[2]]
set_property IOSTANDARD LVCMOS33 [get_ports addrB[2]]
set_property PACKAGE_PIN V15 [get_ports addrB[1]]
set_property IOSTANDARD LVCMOS33 [get_ports addrB[1]]
set_property PACKAGE_PIN W15 [get_ports addrB[0]]
set_property IOSTANDARD LVCMOS33 [get_ports addrB[0]]


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