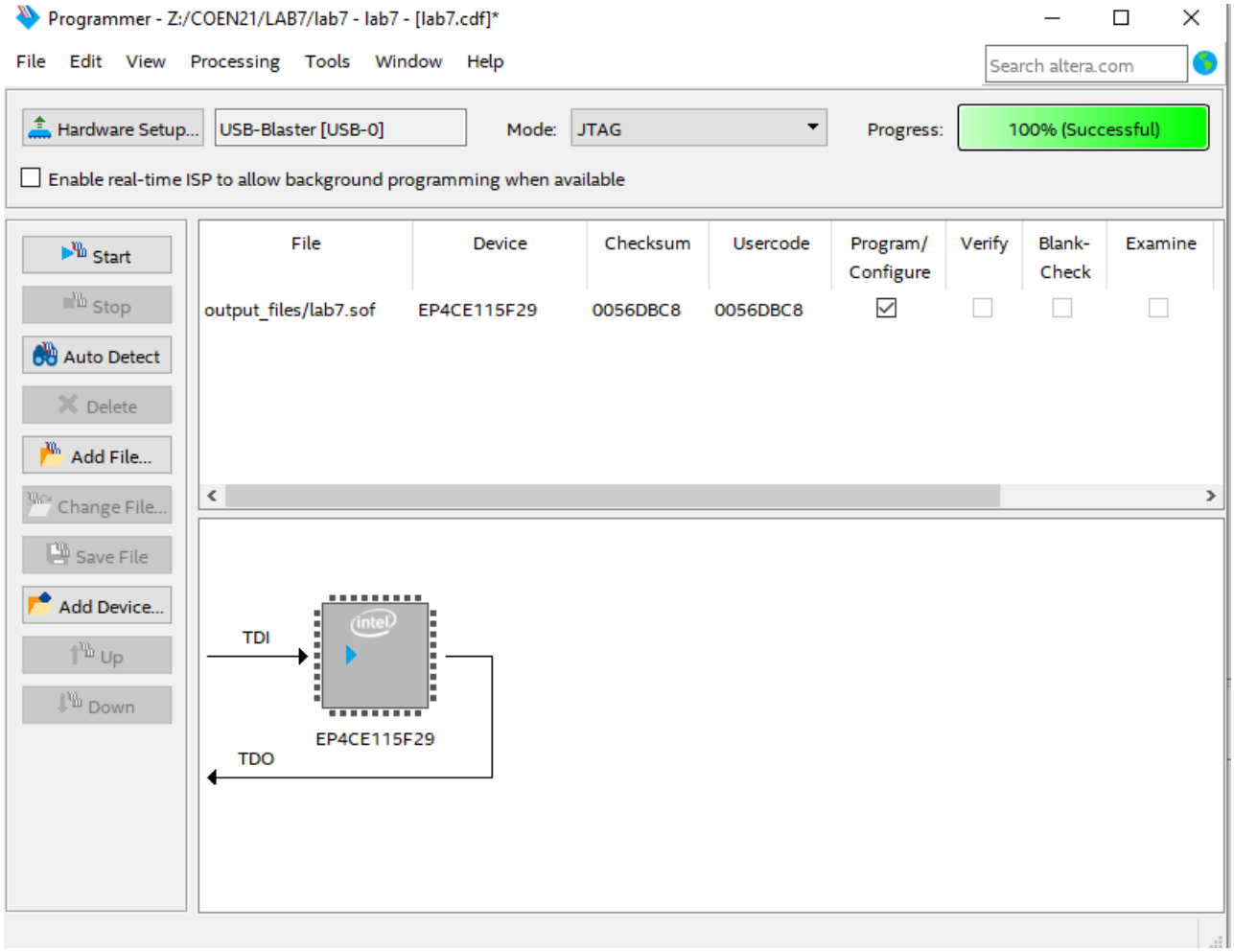
Lab Report7

Yen-Jung(Tim) Lu, Xingting Peng

1. Include an introduction, circuit diagrams and Verilog code from the prelab and lab.

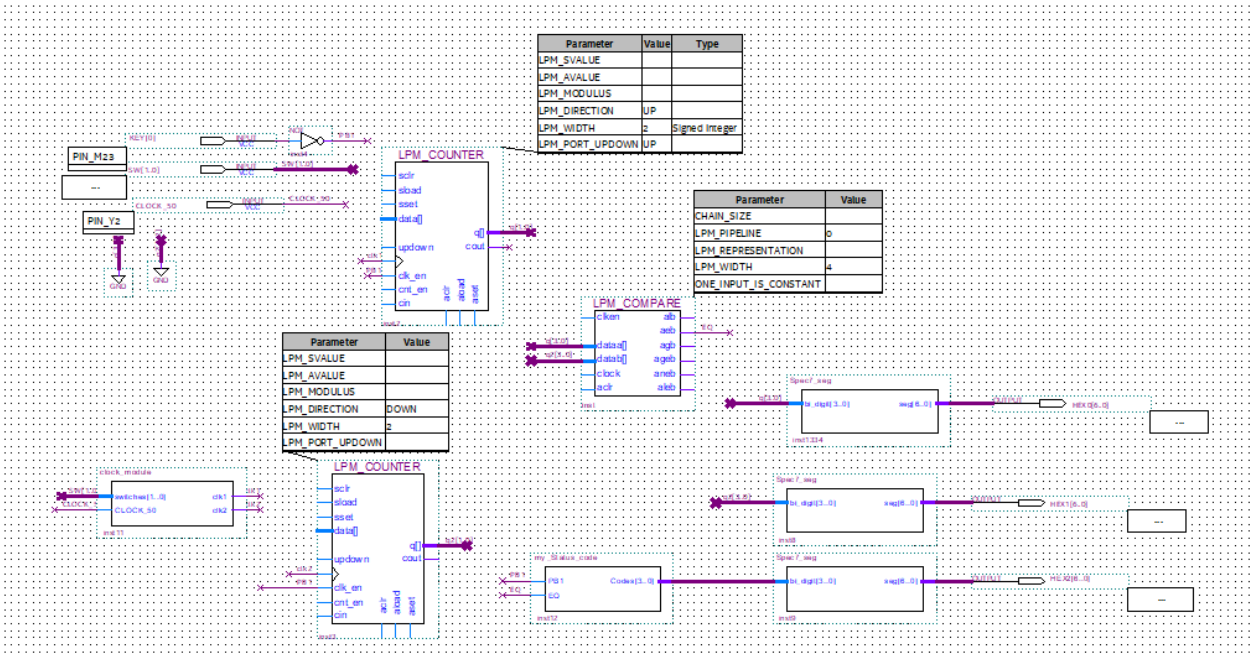
Introduction:

In this lab, students will learn how to use a comparator and counter in a circuit. Students would utilize comparators and counters to create a slot machine which would show win or lose as the user pressed and released the button. Students would also need to write the verilog code for *spec_7seg* and the *my_status _code* to display the status on the circuit board.



```
module my_Status_code(PB1, EQ, Codes);
input PB1, EQ;
output reg [3:0]Codes;
wire [1:0]S;
assign S[0]=EQ;
assign S[1]=PB1;
always @ (*)
    case(S)
    0: Codes = 4'b1100;
    1: Codes = 4'b1101;
    2: Codes = 4'b1110;
    3: Codes = 4'b1111;

    endcase
endmodule
```

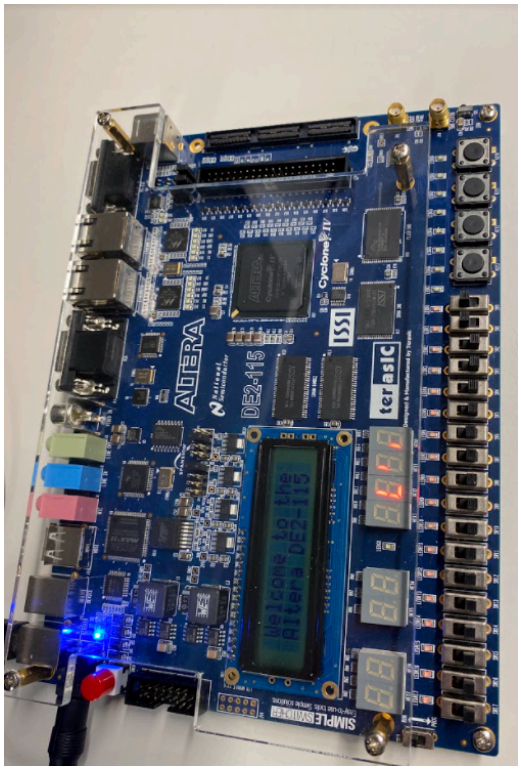
```
module Spec7_seg(bi_digit,seg);
input [3:0] bi_digit;
output [6:0] seg;
reg [6:0] seg;
// seg = {g,f,e,d,c,b,a};
always @ (bi_digit)
case (bi_digit)
4'h0: seg = ~7'b0000010;
4'h1: seg = ~7'b0000100;                    // ---a----
4'h2: seg = ~7'b0010000;                    // |      |
4'h3: seg = ~7'b0100000;                    // f      b
4'h4: seg = ~7'b0000000;                    // |      |
4'h5: seg = ~7'b0000000;                    // ---g----
4'h6: seg = ~7'b0000000;                    // |      |
4'h7: seg = ~7'b0000000;                    // e      c
4'h8: seg = ~7'b0000000;                    // |      |
4'h9: seg = ~7'b0000000;                    // ---d----
4'ha: seg = ~7'b0000000;
4'hb: seg = ~7'b0000000;
4'hc: seg = ~7'b0111000;
4'hd: seg = ~7'b0111110;
4'he: seg = ~7'b0000110;
4'hf: seg = ~7'b0000110;

endcase

endmodule
```



2. How often were you able to win using each mode? Include a photo of your board for both a win and a loss.

We were able to win seven times with the slower speed and win three times with the faster speed.

	slow	fast
1	lose	lose
2	win	win
3	lose	lose
4	win	lose
5	lose	lose
6	lose	lose
7	win	win
8	win	lose
9	lose	lose
10	lose	win



3. From the prelab, which form of the Verilog module my_status_code was the easiest or most intuitive for you? Explain briefly.

In the prelab, it is easier to write the verilog module my_status_code with the if-else statements, because we can easily compose the code by setting up the statement of EQ.

4. If you want to use 8 symbols instead of only 4 to make it harder to win the game, list all the things you would need to change.

- Verilog module of *spec_7seg*
- Verilog module of *my_status code*
- Numbers of counters utilized in the schematic
- Numbers of comparator used in the schematic