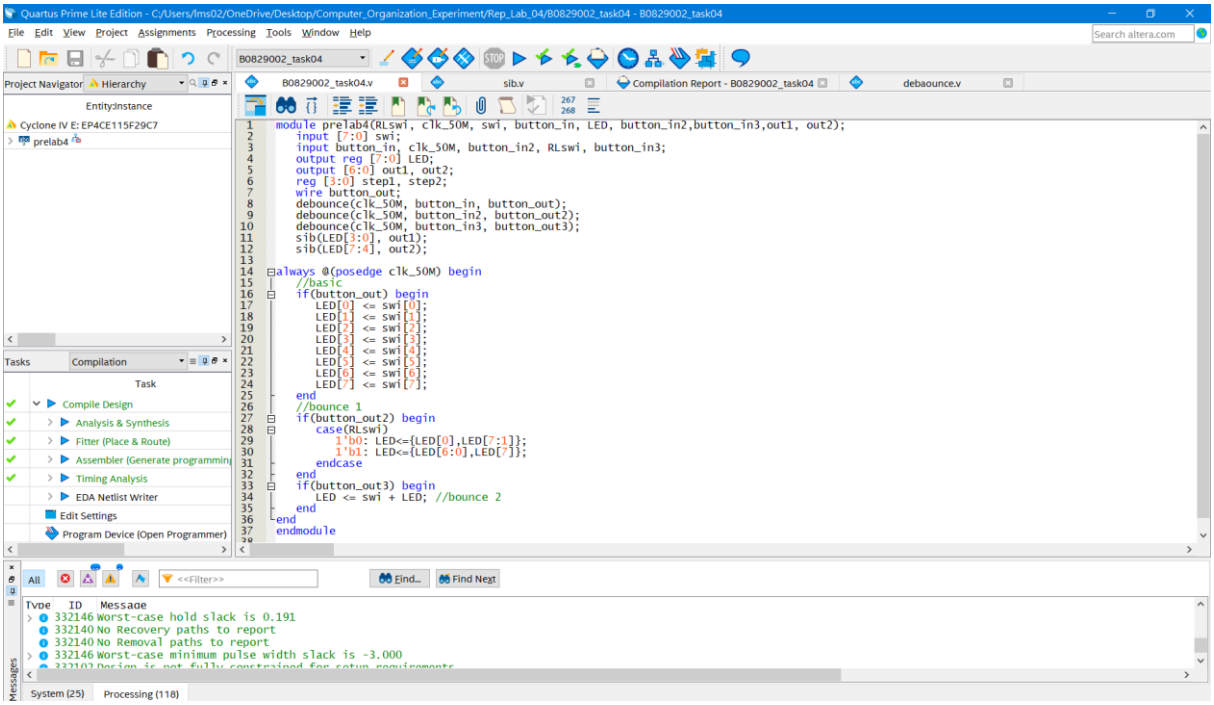
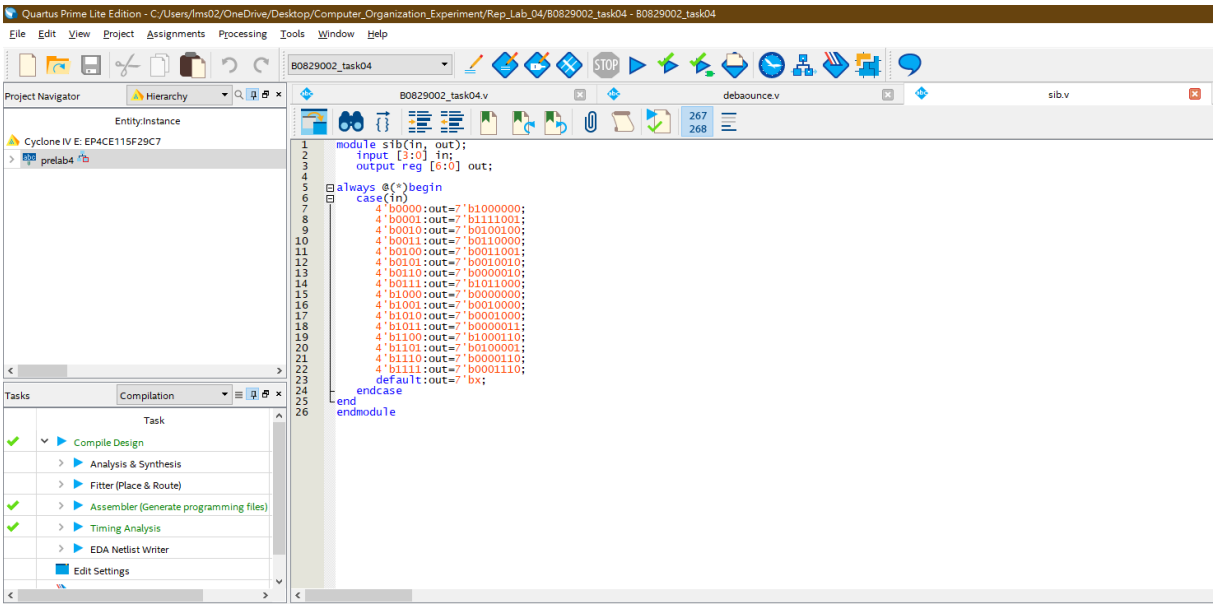
Lab4-Registers and shift Registers

In this Task, we are going to make a register, shifter and adder module, furthermore, we also need to make the output signal into the 7-segment display decoder.

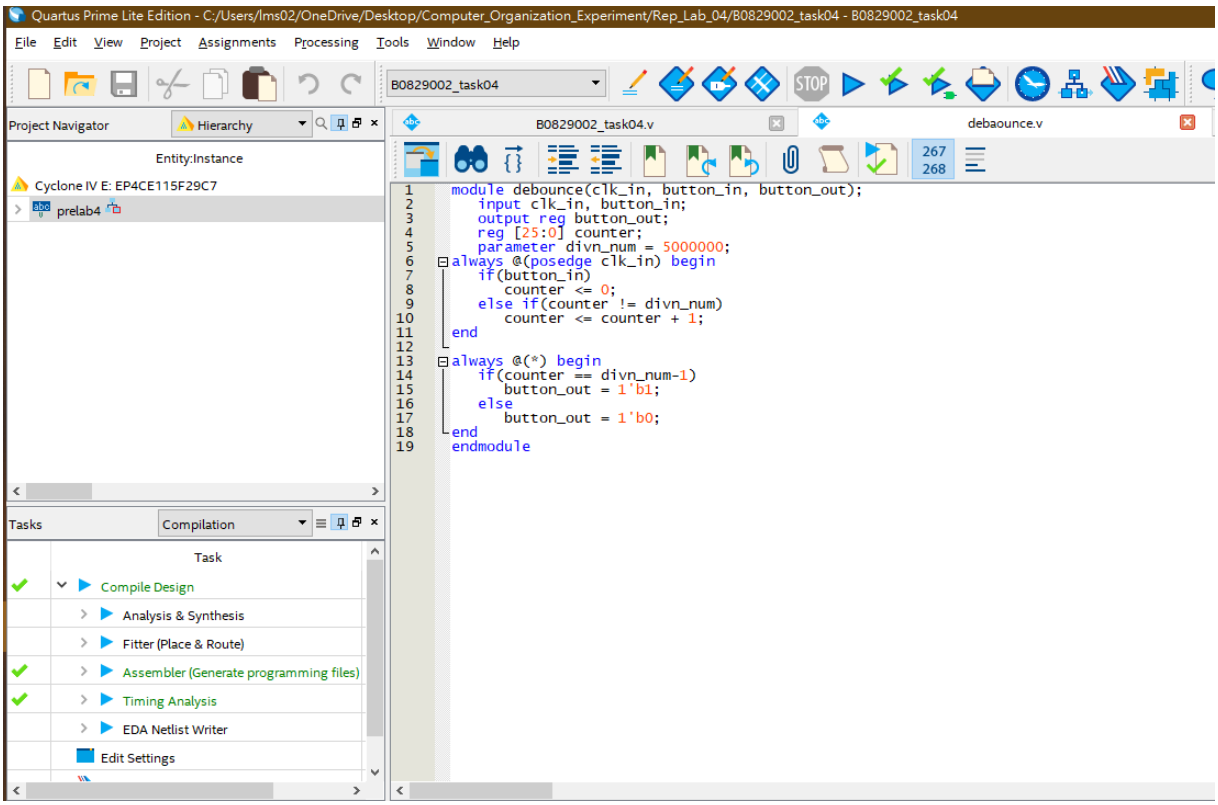
Code:



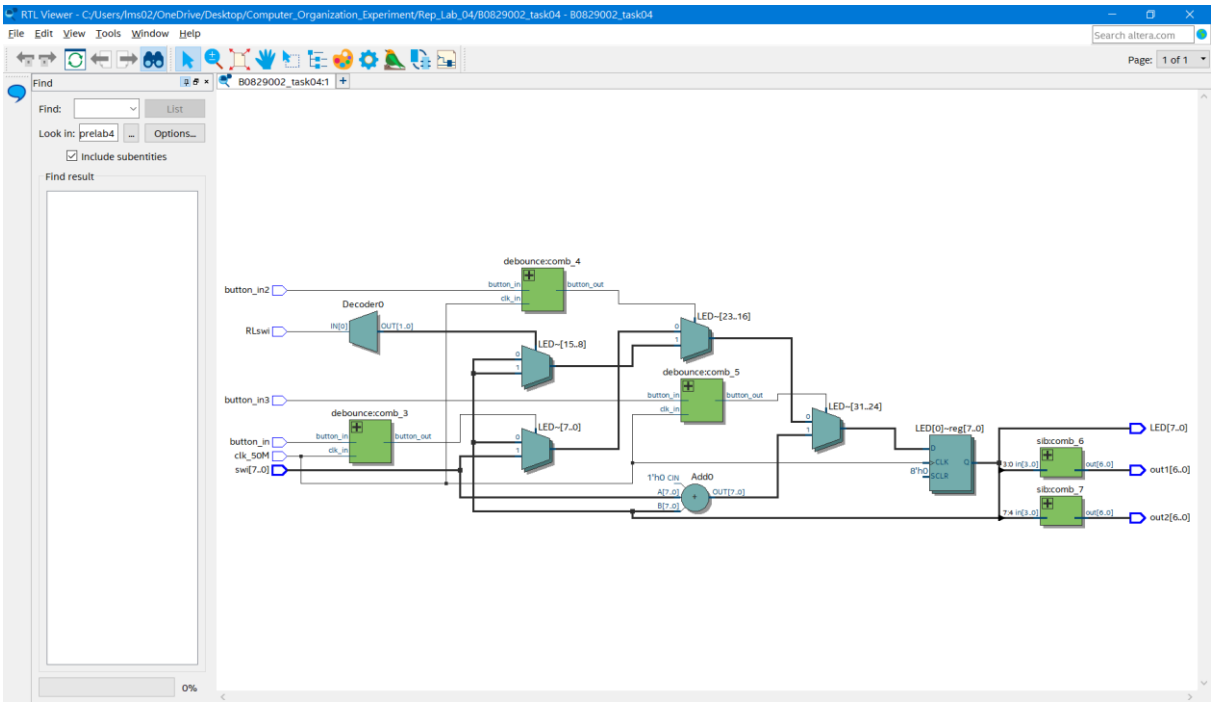

```
module prelab4(RLswi, clk_50M, swi, button_in, LED, button_in2,button_in3,out1, out2);
    input [7:0] swi;
    input button_in, clk_50M, button_in2, RLswi, button_in3;
    output reg [7:0] LED;
    output [6:0] out1, out2;
    reg [3:0] step1, step2;
    wire button_out;
    debounce(clk_50M, button_in, button_out);
    debounce(clk_50M, button_in2, button_out2);
    debounce(clk_50M, button_in3, button_out3);
    sib(LED[3:0], out1);
    sib(LED[7:4], out2);

always @(posedge clk_50M) begin
    //basic
    if(button_out) begin
        LED[0] <= swi[0];
        LED[1] <= swi[1];
        LED[2] <= swi[2];
        LED[3] <= swi[3];
        LED[4] <= swi[4];
        LED[5] <= swi[5];
        LED[6] <= swi[6];
        LED[7] <= swi[7];
    end
    //bounce 1
    if(button_out2) begin
        case(RLswi)
            1'b0: LED<={LED[0],LED[7:1]};
            1'b1: LED<={LED[6:0],LED[7]};
        endcase
    end
    if(button_out3) begin
        LED <= swi + LED; //bounce 2
    end
end
endmodule
```



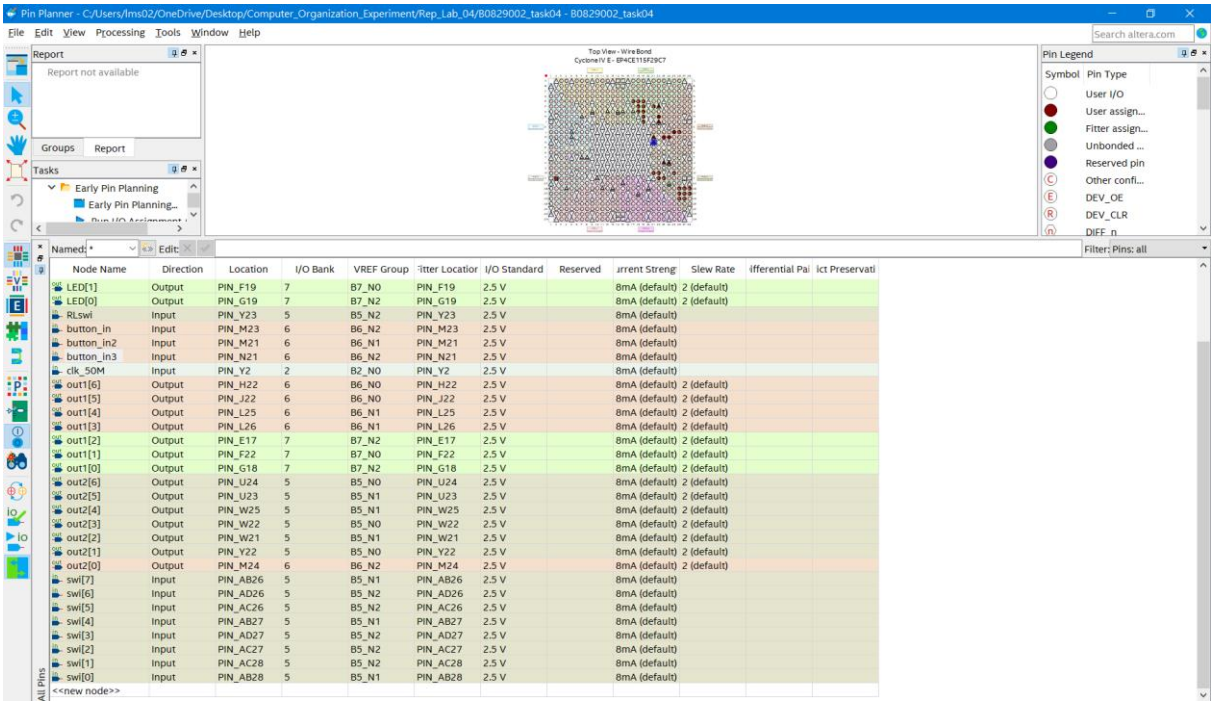

```
module sib(in, out);
    input [3:0] in;
    output reg [6:0] out;

always @(*)begin
    case(in)
        4'b0000:out=7'b1000000;
        4'b0001:out=7'b1111001;
        4'b0010:out=7'b0100100;
        4'b0011:out=7'b0110000;
        4'b0100:out=7'b0011001;
        4'b0101:out=7'b0010010;
        4'b0110:out=7'b0000010;
        4'b0111:out=7'b1011000;
        4'b1000:out=7'b0000000;
        4'b1001:out=7'b0010000;
        4'b1010:out=7'b0001000;
        4'b1011:out=7'b0000011;
        4'b1100:out=7'b1000110;
        4'b1101:out=7'b0100001;
        4'b1110:out=7'b0000110;
        4'b1111:out=7'b0001110;
        default:out=7'bx;
    endcase
end
endmodule
```



RTL:



Pin planer:



Discussion and difficulties:

In my opinion, the most difficult part is to know how to use and code the v file for 7-segment display decoder because when I coding this file, I had some problems from the input to this file and output from this file. In the beginning, I have no idea why I use reg to input to this v file from main.v and output from this v file and to main.v will have some problems to the undeclare variables. Thus, I solve a lot of time to check in which need reg and in which does not need. Another difficulty I thought was the making a adder.

Because of storage of the result signal, we also need to use register to store the value from the addition operation. Here we can use the reg from the last problem to solve this problem easily, just do some easy addition and assign the value into the reg from 7-segment display decoder file. After that we can get an addition module.

About the shifter is using the feature of register, input the value to next flipflop and the last one rotate to the beginning of flipflop.