```
`timescale 1ns / 1ps
//////////////////////////////////////////////////////////////////////////////////
// Company:
// Engineer:
//
// Create Date: 05/09/2021 11:51:40 PM
// Design Name:
// Module Name: lab5_top
// Project Name:
// Target Devices:
// Tool Versions:
// Description:
//
// Dependencies:
//
// Revision:
// Revision 0.01 - File Created
// Additional Comments:
//
//////////////////////////////////////////////////////////////////////////////////


module lab5_top(
    input btnU,
    input [15:0] sw,
    input clkin,
    input btnR,
    output [15:0] led,
    output [3:0] an,
    output dp,
    output [6:0] seg
    );
    wire Up, swA, swB, DTC, IncA, IncB, Load, Run;
    wire digsel, qsec, clk;
    wire Score;
    wire [15:0] Q;
    wire [7:0] Din;
    wire [3:0] sel;
    wire [3:0] n;
    //wire UpA, UpB;
    lab5_clks slowit (.clkin(clkin), .greset(btnR), .clk(clk), .digsel(digsel),
.qsec(qsec));


    FDRE #(.INIT(1'b0) ) syncUp (.C(clk), .CE(1'b1), .D(btnU), .Q(Up));
    FDRE #(.INIT(1'b0) ) syncA (.C(clk), .CE(1'b1), .D(sw[0]), .Q(swA));
```

```
    FDRE #(.INIT(1'b0) ) syncB (.C(clk), .CE(1'b1), .D(sw[15]), .Q(swB));

    lab5_state_machine lab5_fsm (.btnU(Up), .TimeUp(DTC), .Anow(swA), .Bnow(swB),
.clk(clk), .LoadTime(Load), .RunTime(Run), .IncA(IncA), .IncB(IncB),
.ShowScore(Score));

    LFSR lfsr (.clk(clk), .Q(Din));
    Time_Counter time_counter (.clk(clk), .Dw(Run & qsec), .LD(Load), .D(Din),
.DTC(DTC), .Q(Q[11:4]));


    //FDRE #(.INIT(1'b0) ) syncIncA (.C(clk), .CE(1'b1), .D(IncA), .Q(UpA));
    //FDRE #(.INIT(1'b0) ) syncIncB (.C(clk), .CE(1'b1), .D(IncB), .Q(UpB));

    counter4L countA (.clk(clk), .LD(1'b0), .Up(IncA), .Dw(1'b0), .Q(Q[3:0]));
    counter4L countB (.clk(clk), .LD(1'b0), .Up(IncB), .Dw(1'b0), .Q(Q[15:12]));

    ring_counter RingCounter (.clk(clk), .adv(digsel), .sel(sel));
    Selector selector (.sel(sel), .N(Q), .H(n));
    hex7seg seven_segment (.n(n), .e(1'b1), .seg(seg));
    //FDRE #(.INIT(1'b0) ) ff_instance_1095 (.C(clk), .R(reset), .CE(enable),
.D(inD), .Q(outQ));


    //Leds and Switches
    assign an[0] = ~(Score & sel[0]);
    assign an[1] = ~(sw[3] & sel[1]);
    assign an[2] = ~(sw[3] & sel[2]);
    assign an[3] = ~(Score & sel[3]);
    assign led[0] = swA;
    assign led[15] = swB;
    assign led[7] = DTC;
    assign led[8] = DTC;
    assign led[6:1] = 1'b0;
    assign led[14:9] = 1'b0;
    assign dp = 1'b1;
endmodule
```