

```

`timescale 1ns / 1ps

module intro(
    input [3:0] button,
    output [7:0] led,
    input sys_clk,
    input sys_clkp
);

    reg [23:0] clkdiv;
    reg [7:0] counter;
    reg slow_clk;

    // This section defines the main system clock from two
    // differential clock signals: sys_clk and sys_clkp
    // Clk is a high speed clock signal running at ~200MHz
    wire clk;
    IBUFGDS osc_clk(
        .O(clk),
        .I(sys_clkp),
        .IB(sys_clk)
    );

    initial begin
        clkdiv = 0;
        counter = 8'h00;
    end

    assign led = ~counter;

    // This code creates a slow clock from the high speed Clk signal
    // You will use the slow clock to run your finite state machine
    // The slow clock is derived from the fast 20 MHz clock by dividing it 10,000,000 times
    // Hence, the slow clock will run at 2 Hz
    always @(posedge clk) begin
        clkdiv <= clkdiv + 1'b1;
        if (clkdiv == 10000000) begin
            slow_clk <= ~slow_clk;
            clkdiv <= 0;
        end
    end

/*
//The main code will run from the slow clock. The rest of the code will be in this section.
//The counter will increment when button 0 is pressed and on the rising edge of the slow clk
//The counter will decrement when button 0 is pressed and on the rising edge of the slow clk
always @(posedge slow_clk) begin
    if (button [0] == 1'b0) begin

```

```

        counter <= counter + 1'b1;
    end
    else if (button [1] == 1'b0) begin
        counter <= counter - 1'b1;
    end
end
*/
// code to turn on all LEDs when button[0] is pressed
always @(posedge slow_clk) begin
    if ((button [0] == 1'b0) && (button[1]==1'b0 || button[2]==1'b0 || button[3]==1'b0)) begin
        counter <= counter + 1'b0;
    end
    else if ((button [1] == 1'b0) && (button[0]==1'b0 || button[2]==1'b0 || button[3]==1'b0))
begin
        counter <= counter + 1'b0;
    end
    else if ((button [2] == 1'b0) && (button[1]==1'b0 || button[0]==1'b0 || button[3]==1'b0))
begin
        counter <= counter + 1'b0;
    end
    else if ((button [3] == 1'b0) && (button[1]==1'b0 || button[2]==1'b0 || button[0]==1'b0))
begin
        counter <= counter + 1'b0;
    end
    else if (button [0] == 1'b0) begin
        counter <= 8'hFF;
    end
    else if (button [1] == 1'b0) begin
        counter <= 8'h00;
    end
    else if (button [2] == 1'b0) begin

        if (counter > 8'hFD) begin
            counter <= 8'h00;
        end
        else
            counter <= counter + 2'b10;
    end
    else if (button [3] == 1'b0) begin
        counter <= counter - 2'b10;
        if (counter < 8'h02) begin
            counter <= 8'h00;
        end
        else
            counter <= counter - 2'b10;
    end
end

```

end

endmodule