Design of 12-hour Clock in Verilog

Ujwal Vikas Uttarwar (21307R021)

Department of Electrical Engineering

Indian Institute of Technology Bombay

Problem Statement

Design a set of counters suitable for a 12-hour clock system with an AM/PM indicator. These counters will be driven by a high-frequency clock signal (clk) and will be incremented by one every time a pulse is received on the 'ena' input, indicating a one-second interval. A 'reset' input will be used to set the clock to 12:00 AM. The 'pm' signal will be 0 for AM and 1 for PM. The counters include 'hh'

for hours, 'mm' for minutes, and 'ss' for seconds, each represented by two Binary-Coded Decimal (BCD) digits. It's important to note that the reset operation takes precedence over the enable operation and can occur even when the enable signal is not active.

To provide clarity on the operation, consider the following timing diagram which illustrates the transition from 11:59:59 AM to 12:00:00 PM, as well as the behavior of synchronous reset and enable signals.

Verilog Code –

```
module top_module(
    input clk,
    input reset,
    input ena,
    output pm,
    output [7:0] hh,
    output [7:0] mm,
    output [7:0] ss);

    wire ena1,ena2,ena3;
    BCD_SS_MM Sec(clk,reset,ena,ena1,ss);
    BCD_SS_MM Min(clk,reset,ena1,ena2,mm);
    BCD_HH Hr(clk,reset,ena2,ena3,hh);
    AM_PM AP(clk,reset,ena3,pm);

endmodule
```

```
module BCD_SS_MM(
    input clk,
    input reset,
    input ena_in,
    output ena_out,
    output [7:0] q);

    assign ena_out = ena_in&q[3:0]==4'd9&q[7:4]==4'd5;

    always @ (posedge clk)
        begin
            if(reset)
                begin
                                        q<=8'd0;
                end
            else if(ena_in&~(q[3:0]==4'd9))
                                begin
                    q[3:0]<=q[3:0]+4'd1;
                end
            else if(ena_in&q[3:0]==4'd9&~(q[7:4]==4'd5))
                begin
                    q[3:0]<=4'd0;
                    q[7:4]<=q[7:4]+4'd1;
                end
            else if(ena_in&q[3:0]==4'd9&q[7:4]==4'd5)
```

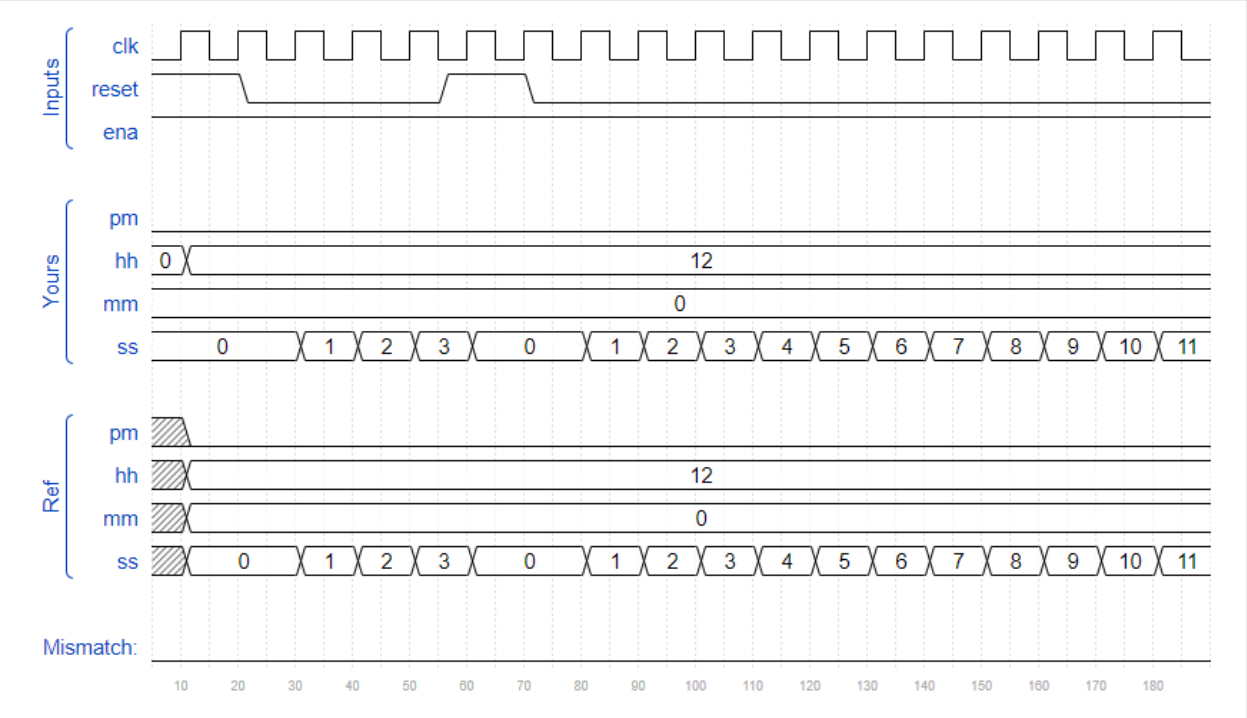
```
            begin
                q<=8'd0;
            end
        end
endmodule


module BCD_HH(
    input clk,
    input reset,
    input ena_in,
    output ena_out,
    output [7:0] q);

    assign ena_out = ena_in&q[3:0]==4'd1&q[7:4]==4'd1;

    always @ (posedge clk)
        begin
            if(reset)
                begin
                                    q<=8'b00010010;
                end
            else if(ena_in&~(q[3:0]==4'd9)&q[7:4]==4'd0)
                                begin
                q[3:0]<=q[3:0]+4'd1;
                end
            else if(ena_in&q[3:0]==4'd9&q[7:4]==4'd0)
```

```
            begin
                q[3:0]<=4'd0;
                q[7:4]<=q[7:4]+4'd1;
            end
        else if(ena_in&~(q[3:0]==4'd2)&(q[7:4]==4'd1))
            begin
                q[3:0]<=q[3:0]+4'd1;
            end
        else if(ena_in&q[3:0]==4'd2&q[7:4]==4'd1)
            begin
                q<=8'b00000001;
            end
    end
endmodule

module AM_PM(
    input clk,
    input reset,
    input ena_in,
    output pm);

    always @ (posedge clk)
        begin
            if(reset)
                begin
                    pm<=1'b0;
```

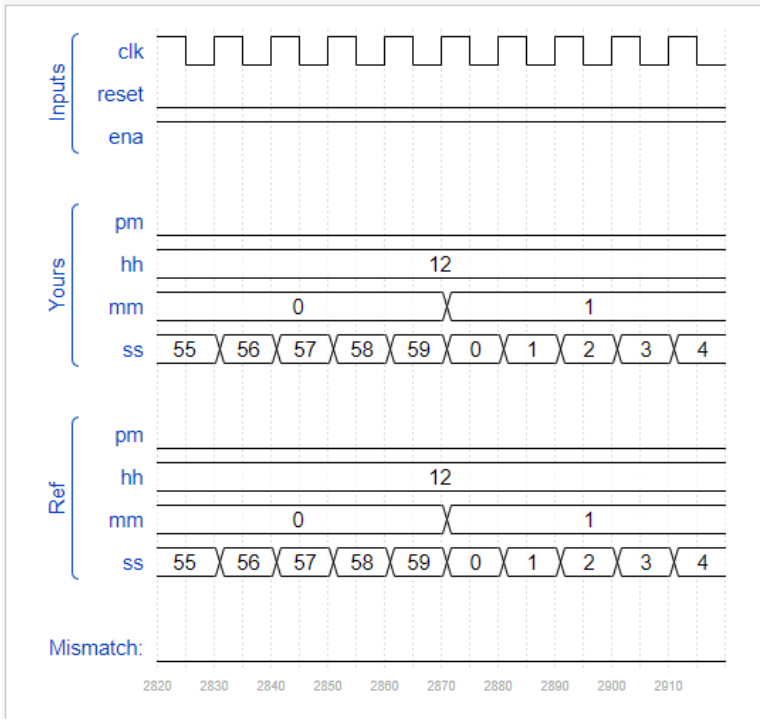
```
            end
        else if(ena_in)
            begin
                pm<=~pm;
            end
    end
endmodule
```

RESULTS

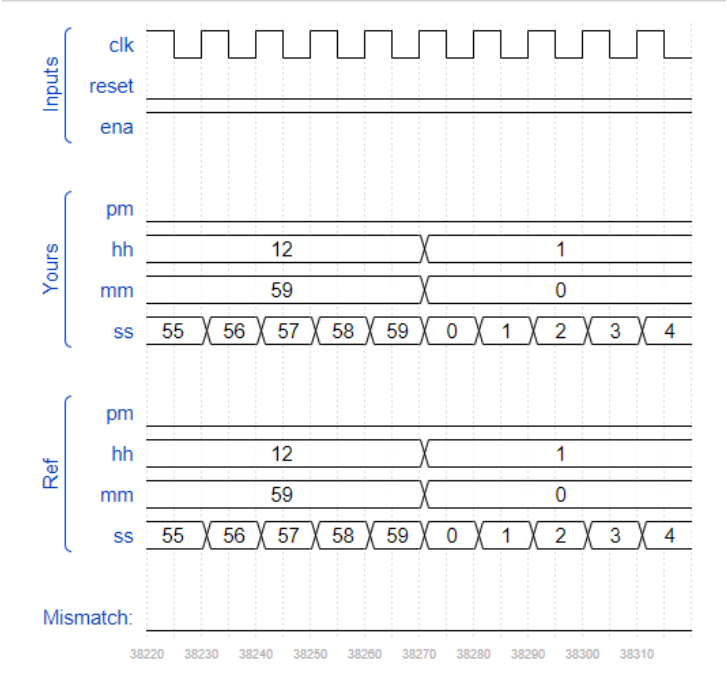
Reset and Count to 10



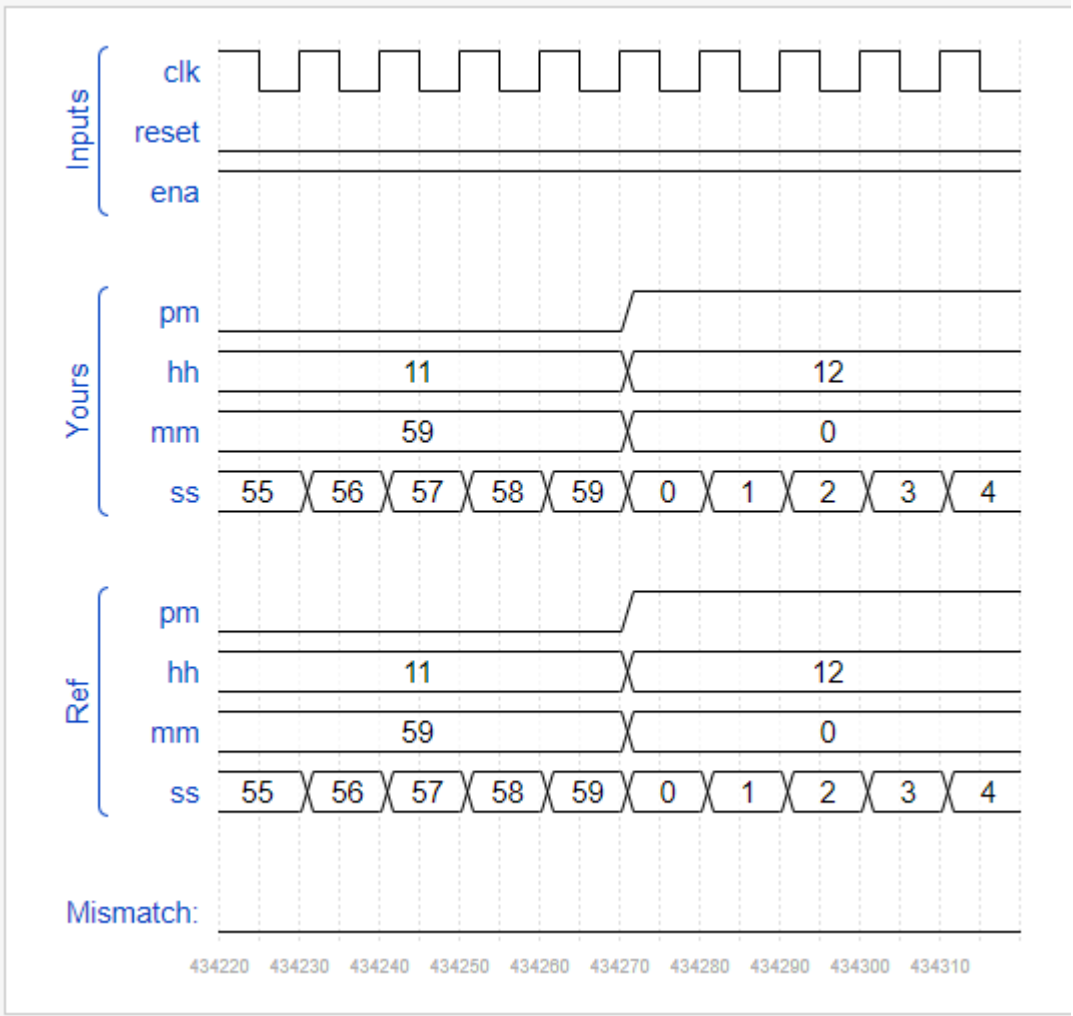
Minute roll-over



Hour roll-over



PM roll-over


Inputs
clk
reset
ena
Yours
pm
hh
mm
ss
11
12
59
0
55
56
57
58
59
0
1
2
3
4
Ref
pm
hh
mm
ss
11
12
59
0
55
56
57
58
59
0
1
2
3
4
Mismatch:
434220
434230
434240
434250
434260
434270
434280
434290
434300
434310