Talal Jawaid

11/4/2018

CSC 137

Project #2

```
Sequence to be detected: 1010
Input given: 0101010101101011011101111100010101
                  0   CLK: 0  Y: x  RST: 0   Digit: 0
                  5   CLK: 1  Y: 0  RST: 0   Digit: 1
                 10   CLK: 0  Y: 0  RST: 0   Digit: 0
                 15   CLK: 1  Y: 0  RST: 0   Digit: 1
                 20   CLK: 0  Y: 0  RST: 0   Digit: 0
                 25   CLK: 1  Y: 0  RST: 0   Digit: 1
                 30   CLK: 0  Y: 0  RST: 0   Digit: 0
                 35   CLK: 1  Y: 0  RST: 0   Digit: 1
                 40   CLK: 0  Y: 0  RST: 0   Digit: 0
                 45   CLK: 1  Y: 0  RST: 0   Digit: 1
                 50   CLK: 0  Y: 0  RST: 0   Digit: 1
                 55   CLK: 1  Y: 0  RST: 0   Digit: 0
                 60   CLK: 0  Y: 0  RST: 0   Digit: 1
                 65   CLK: 1  Y: 0  RST: 0   Digit: 0
                 70   CLK: 0  Y: 0  RST: 0   Digit: 1
                 75   CLK: 1  Y: 0  RST: 0   Digit: 1
                 80   CLK: 0  Y: 0  RST: 0   Digit: 0
                 85   CLK: 1  Y: 0  RST: 0   Digit: 1
                 90   CLK: 0  Y: 0  RST: 0   Digit: 1
                 95   CLK: 1  Y: 0  RST: 0   Digit: 0
                100   CLK: 0  Y: 0  RST: 0   Digit: 1
```

```
GNU nano 2.0.9                          File: mooreFINAL.v

`timescale 1ns/1ps
module mooredetector(digit, rst, clk, y);

input digit;
input clk;
input rst;

output reg y;

reg [2:0] cs, ns;

parameter S0 = 3'b000, S1 = 3'b001, S2 = 3'b010, S3 = 3'b100, S4 = 3'b101;

always @(cs or digit)
  begin
  case (cs)
    S0: if (digit == 1'b1)
            begin
                ns = S1;
                y=1'b0;
            end
        else ns = cs;
    S1: if (digit == 1'b0)
            begin
                ns = S2;
                y=1'b0;
            end
        else
            begin
                ns = cs;
                y=1'b0;
            end
    S2: if (digit == 1'b1)
            begin
                ns = S3;
                y=1'b0;
            end
            else
            begin
                ns = S0;
                y=1'b0;
            end
    S3: if (digit == 1'b0)
            begin
                ns = S4;
                y=1'b0;
            end
        else
            begin
                ns = S1;
                y=1'b0;
            end
    S4: if (digit == 1'b0)
            begin
                ns = S0;
                y=1'b1;
            end
            else
            begin
                ns = S3;
                y=1'b1;
            end
    default: ns = S0;
  endcase
end
always@(posedge clk)
begin
        if (rst)
        cs <= S0;
        else
        cs <= ns;
end
endmodule

module mooredetector_tb;
reg digit,clk,rst;
wire y;

mooredetector uut(digit, rst, clk, y);

^G Get Help     ^O WriteOut     ^R Read File    ^Y Prev Page    ^K Cut Text     ^C Cur Pos
^X Exit         ^J Justify      ^W Where Is     ^V Next Page    ^U UnCut Text   ^T To Spell
```

```verilog
                    y=1'b0;
            end
    S3: if (digit == 1'b0)
            begin
                ns = S4;
                y=1'b0;
            end
        else
            begin
                ns = S1;
                y=1'b0;
            end
    S4: if (digit == 1'b0)
            begin
                ns = S0;
                y=1'b1;
            end
        else
            begin
                ns = S3;
                y=1'b1;
            end
    default: ns = S0;
  endcase
end
always@(posedge clk)
begin
        if (rst)
            cs <= S0;
        else
            cs <= ns;
end
endmodule

module mooredetector_tb;
reg digit,clk,rst;
wire y;

mooredetector uut(digit, rst, clk, y);


initial begin
rst=0;
clk=0;
digit=0;

$display("Sequence to be detected: 1010");
$display("Input given: 0101010101101011011011100010101");
$monitor($time, , ,"CLK: %b",clk,,"Y: %b",y,,"RST: %b ",rst,,"Digit: %b",digit);

#5 digit=1;
#5 digit=0;
#5 digit=1;
#5 digit=0;
#5 digit=1;
#5 digit=0;
#5 digit=1;
#5 digit=0;
#5 digit=1;
#5 digit=1;
#5 digit=0;
#5 digit=1;
#5 digit=0;
#5 digit=1;
#5 digit=0;
#5 digit=1;
#5 digit=0;
#5 digit=1;
#5 digit=0;
#5 digit=1;
#5 digit=1;
#5 digit=1;
#5 digit=0;
#5 digit=0;
#5 digit=0;
#5 digit=1;
#5 digit=0;
#5 digit=1;
#5 digit=0;
```

```verilog
initial begin
rst=0;
clk=0;
digit=0;

$display("Sequence to be detected: 1010");
$display("Input given: 0101010101101011011011100010101");
$monitor($time, , ,"CLK: %b",clk,,"Y: %b",y,,"RST: %b ",rst,,"Digit: %b",digit);

#5 digit=1;
#5 digit=0;
#5 digit=1;
#5 digit=0;
#5 digit=1;
#5 digit=0;
#5 digit=1;
#5 digit=0;
#5 digit=1;
#5 digit=0;
#5 digit=1;
#5 digit=0;
#5 digit=1;
#5 digit=1;
#5 digit=0;
#5 digit=1;
#5 digit=0;
#5 digit=1;
#5 digit=1;
#5 digit=1;
#5 digit=0;
#5 digit=0;
#5 digit=0;
#5 digit=1;
#5 digit=0;
#5 digit=1;
#5 digit=0;
#5 digit=1;


end

always
#5 clk=~clk;
initial
#100 $finish;


endmodule
```