



<https://edaplayground.com/x/AQkV>

testbench.sv

```
// Vitaliy Prymak
module test;
reg mclk, reset,x; //inputs
// Outputs
wire clk,y;
wire [1:0]state;
// Instantiate the Unit Under Test (UUT)
sequence_detector uut (mclk, clk, reset, x,
y, state);
initial begin
    // Initialize Inputs
    mclk =0;
    forever #5 mclk = ~mclk;
    #200 $finish;
end
initial begin
    // Wait 100 ns for global reset to
    finish
    $dumpfile("dump.vcd");$dumpvars(1,test);
$monitor("state=%b,x=%b,y=%b",state,x,y);
    x = 0; reset =1;
    #13 reset = 0;
    #10 x = 0;
    #10 x = 0;
end
endmodule
```

design.sv

```
// Vitaliy Prymak
SV/Verilog Design
module sequence_detector(mclk, clk, reset, x,
y, state);
input mclk, reset, x;
output clk, y;
output reg [1:0] state;
// reg we use if something will change
assign clk = mclk; //uncomment this line
for simulation
    //clkDiv cd(mclk, clk); uncomment for
Basys2
always @(posedge clk, posedge reset)
// posedge means go up 1
if (reset)
state = 2'b00;
else
case(state)
    2'b00: if(x) state = 2'b00; else state =
2'b01;
    2'b01: if(x) state = 2'b00; else state =
2'b10;
    2'b10: if(x) state = 2'b11; else state =
2'b10;
    2'b11: if(x) state = 2'b01; else state =
2'b00;
endcase
endmodule
```

