**Task**

Build an electronic combination lock with reset button, two number buttons (0 and 1), and an unlock output. The combination should be “01011”.

**Code:**

|  |
| --- |
| module lock(inp, out, rst, clk);  reg [2:0] state;  reg [2:0] next\_state;  input inp,rst,clk;  output reg out;  parameter s0=3'b000, s1=3'b001,s2=3'b010,s3=3'b011, s4=3'b100,s5=3'b101;  always @(clk)  begin  state<=next\_state;  end  always @(inp,rst, state)  begin  if(rst)  begin  case(state)  s0:  if(inp==0)  begin  next\_state<=s0;  out<=0;  end  else  begin  next\_state<=s1;  out<=0;  end  s1:  if(inp==0)  begin  next\_state<=s2;  out<=0;  end  else  begin  next\_state<=s1;  out<=0;  end  s2:  if(inp==0)  begin  next\_state<=s0;  out<=0;  end  else  begin  next\_state<=s3;  out<=0;  end  s3:  if(inp==0)  begin  next\_state<=s1;  out<=0;  end  else  begin  next\_state<=s4;  out<=0;  end    s4:  if(inp==0)  begin  next\_state<=s5;  out<=0;  end  else  begin  next\_state<=s1;  out<=0;  end    s5:  if(inp==0)  begin  next\_state<=s0;  out<=1;  end  else  begin  next\_state<=s0;  out<=1;  end  default:  out=0;  endcase    end    else  begin  next\_state<=s0;  out<=0;  end  end  endmodule |

**Simulation Code:**

|  |
| --- |
| module lock\_tst;  reg inp,rst,clk;  wire out;  lock sequence(inp, out, rst, clk);  initial  begin  clk = 1;  rst=0;  #5 rst=1;  #5 inp=1;  #5 inp=0;  #5 inp=1;  #5 inp=1;  #5 inp=0;    #50 $finish;  end    always  begin  #5 clk = ~clk;  end    initial  begin  $display ("input | clock | Reset | Output");  $monitor ("%b %b %b %b ",inp,clk,rst,out);  end  endmodule |