VERILOG SOURCE CODE:

```
module moorefsm(a,clk,z);
    input a;
    input clk;
    output z;
reg z;
parameter st0=0,st1=1,st2=2,st3=3;
reg [0:1] moore_state;
initial
begin
moore_state=st0;
end
always@(posedge(clk))
case(moore_state)
st0: begin
z=1;
if(a)
moore_state=st2;
end
st1:begin
z=0;
if(a)
moore_state=st3;
end
st2:begin
z=0;
if(a)
moore_state=st3;
else
moore_state=st1;
end
st3:begin
z=1;
if(a)
moore_state=st0;
end
endcase
endmodule
```

TESTBENCH PROGRAM:

```
ENTITY MOORETS IS
END MOORETS;
 ARCHITECTURE behavior OF MOORETS IS
     COMPONENT moorefsm
    PORT(a : IN  std_logic;
         clk : IN  std_logic;
         z : OUT  std_logic  );
    END COMPONENT;
          signal a : std_logic := '0';
   signal clk : std_logic := '0';
   signal z : std_logic;
   constant clk_period : time := 1us;
 BEGIN
   uut: moorefsm PORT MAP (a => a, clk => clk, z => z  );
   clk_process :process
```

```
   begin
  clk <= ’0’;
  wait for clk_period/2;
  clk <= ’1’;
  wait for clk_period/2;
   end process;
    stim_proc: process
   begin
      wait for 100ms;
      wait for clk_period*10;
      wait;
   end process;
END;
```