國立高雄科技大學

電子工程系(第一校區)

硬體描述語言

Lab. 8: Finite State Machine

指導教授：陳銘志

班　　級：電子三甲

學生姓名：蕭詠釗

學　　號：C111112132

Lab. 8: Finite State Machine

1. Please use Verilog HDL to implement the Mealy Finite State Machine in Figure 1 .

- Clock period is 20ns.
- All states must be passed, the order is S0→S1→S2→S3→S4→S0→S1.
- The unstable state of the output signal cannot exceed half of the period.
- The signals to be included in $monitor are the following three, in, State and out.
- Total simulation time is 150ns

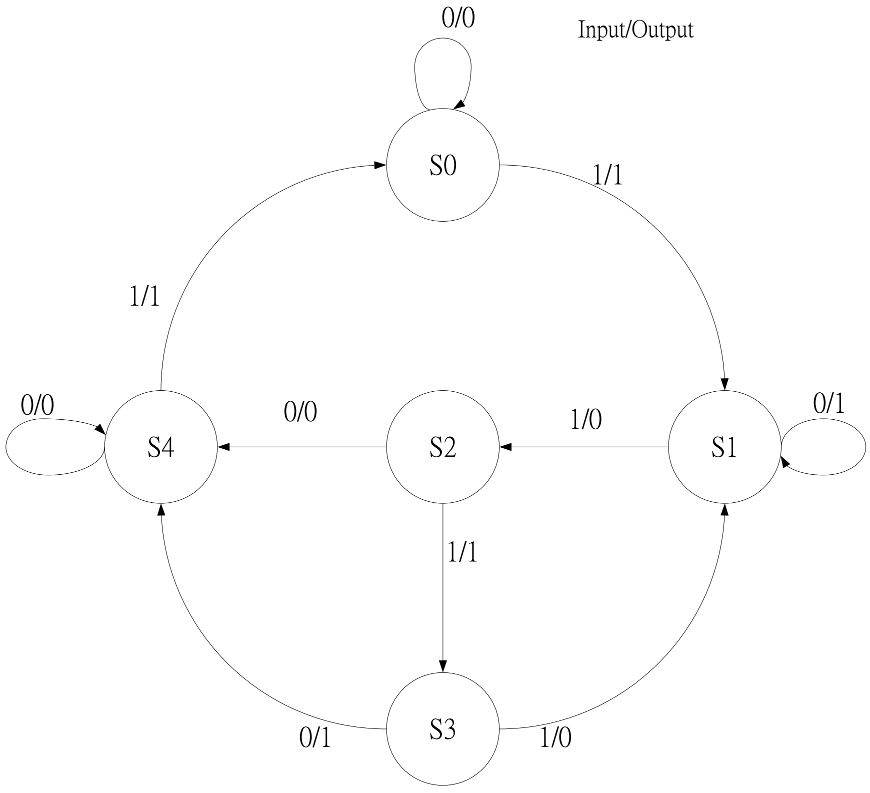
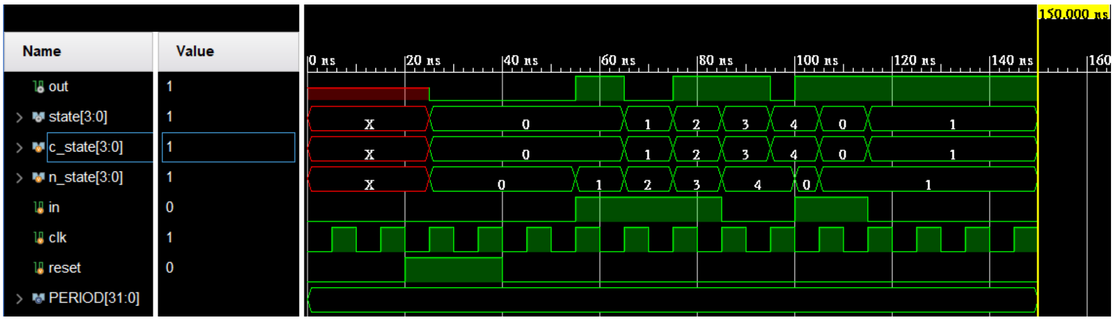


Figure 1 Mealy FSM

作業目標

- 透過調整測試程式使狀態機正常的改變狀態。
  - 狀態機運行流程 ：S0→S1→S2→S3→S4→S0→S1.
  - 狀態改變的間隔時間 ：10 ns
  - Total Simulation Time : 150ns

作業範例



Name	Value
out	1
state[3:0]	1
c_state[3:0]	1
n_state[3:0]	1
in	0
clk	1
reset	0
PERIOD[31:0]	

片段測試程式

```
initial begin
        #20 reset = 1;
        #20 reset = 0;
        #15 in = 1 ;
        #30 in = 0 ;
        #15 in = 1 ;
        #15 in = 0 ;
    end

    initial begin
        #150 $finish;
    end
```

作業作答格式

檢附項目 ： 主(電路)程式、測試程式、RTL schematic、Technology Schematic、Behavioral Simulation Waveform(含 $monitor)， (每一項 18 分)。

--第一題作答區--

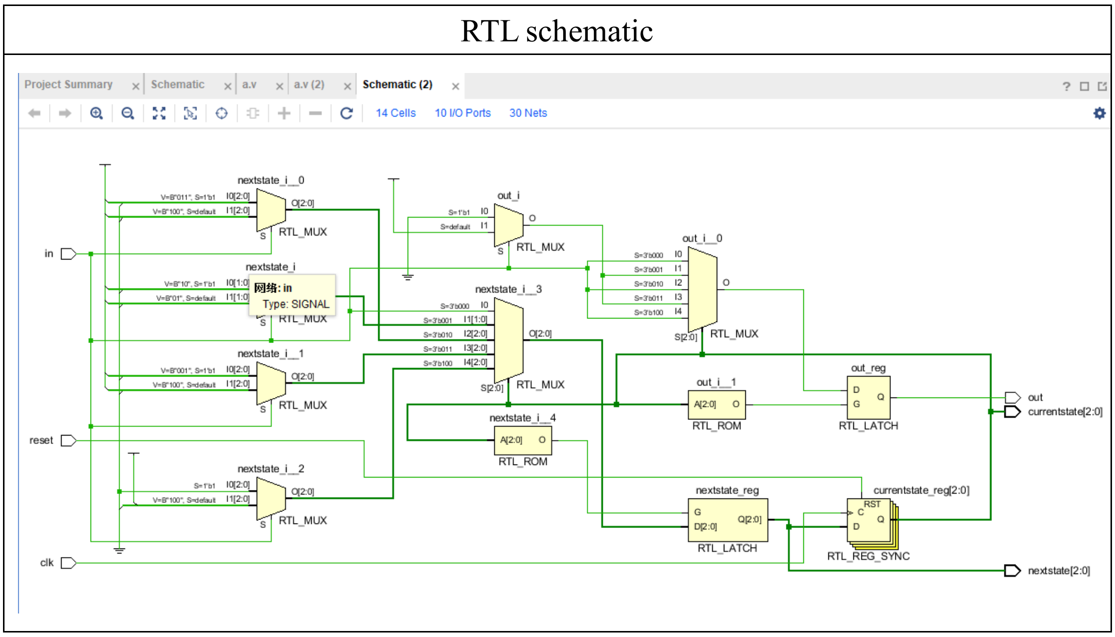
主(電路)程式

```
module count (
    input clk,
    input reset,
    input in,
    output reg out,
    output reg [2:0] currentstate,
    output reg [2:0] nextstate
);
    parameter [2:0] s0 = 3'd0,
                    s1 = 3'd1,
                    s2 = 3'd2,
                    s3 = 3'd3,
                    s4 = 3'd4;
    always @(posedge clk) begin
        if (reset)
            currentstate <= s0;
        else
            currentstate <= nextstate;
    end
    always @(currentstate or in) begin
        case(currentstate)
            s0: begin
                if (in) begin
                    nextstate <= s1;
                    out <= 1'd1;
                end else begin
                    nextstate <= s0;
                    out <= 1'd0;
                end
            end

            s1: begin
                if (in) begin
                    nextstate <= s2;
                    out <= 1'd0;
                end else begin
                    nextstate <= s1;
                    out <= 1'd1;
                end
            end

            s2: begin
                if (in) begin
                    nextstate <= s3;
                    out <= 1'd1;
                end else begin
                    nextstate <= s4;
                    out <= 1'd0;
                end
            end

            s3: begin
                if (in) begin
                    nextstate <= s1;
                    out <= 1'd0;
                end else begin
                    nextstate <= s4;
                    out <= 1'd1;
                end
            end

            s4: begin
                if (in) begin
                    nextstate <= s0;
                    out <= 1'd1;
                end else begin
                    nextstate <= s4;
                    out <= 1'd0;
                end
            end

            default: begin
                nextstate <= nextstate;
                out <= out;
            end
        endcase
    end
endmodule
```

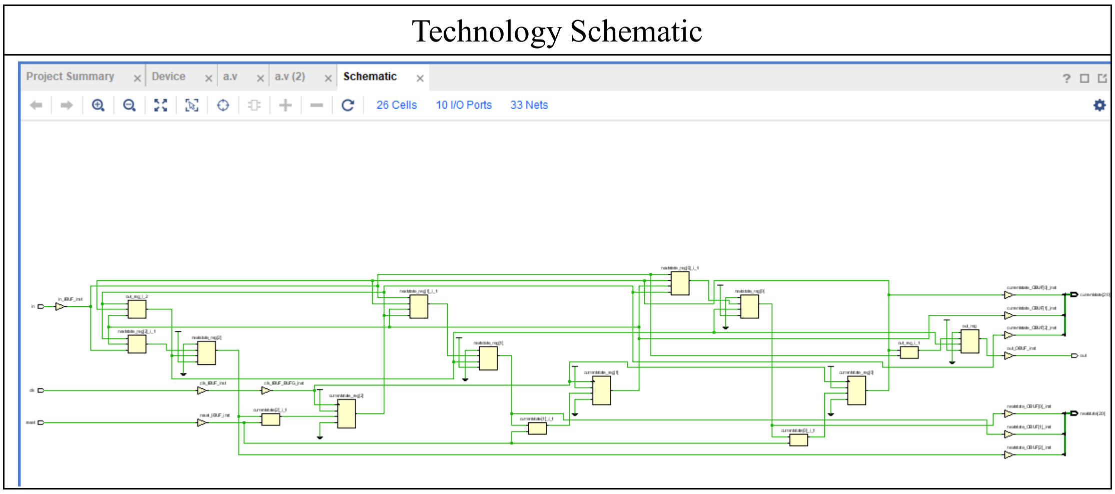
測試程式(testbench)

```
module stimulus;
    reg in, clk, reset;
    wire out;
    wire [2:0] currentstate, nextstate;
    count UUT (
        .clk(clk),
        .reset(reset),
        .in(in),
        .out(out),
        .currentstate(currentstate),
        .nextstate(nextstate)
    );
    initial begin
        clk = 1'b0;
        forever #5 clk = ~clk;
    end
    initial begin
        $monitor($time, " \t in = %b currentstate = %b nextstate = %b out = %b",
                  in, currentstate, nextstate, out);
    end
    initial begin
        in = 1'b0; reset = 1'b0;
        #20 reset = 1'b1;
        #20 reset = 1'b0;
        #15 in = 1'b1;
        #30 in = 1'b0;
        #15 in = 1'b1;
        #15 in = 1'b0;
    end
    initial begin
        #150 $finish;
    end
endmodule
```

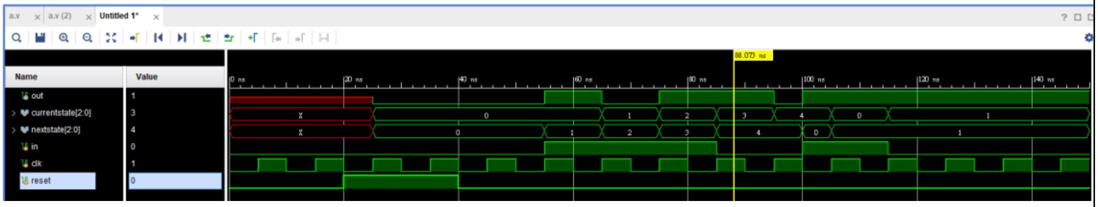
RTL schematic

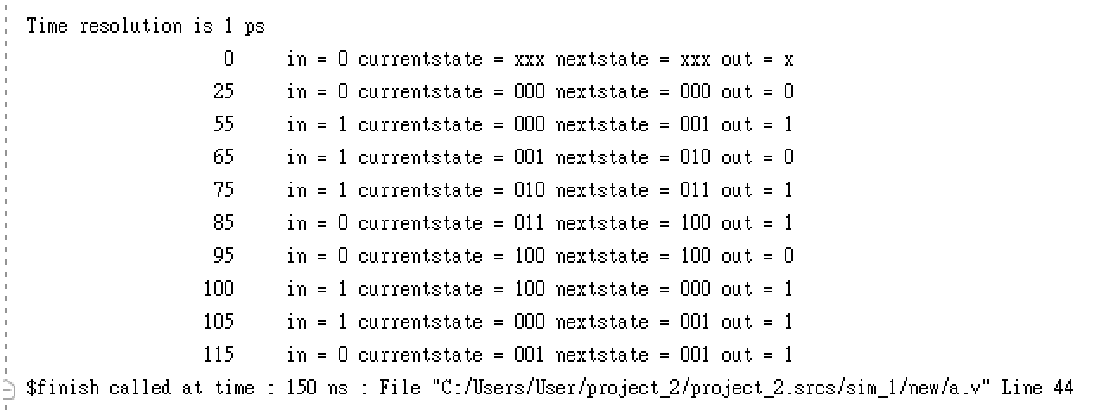


Technology Schematic



Behavioral Simulation Waveform(含 $monitor)





```
Time resolution is 1 ps
                   0  in = 0 currentstate = xxx nextstate = xxx out = x
                  25  in = 0 currentstate = 000 nextstate = 000 out = 0
                  55  in = 1 currentstate = 000 nextstate = 001 out = 1
                  65  in = 1 currentstate = 001 nextstate = 010 out = 0
                  75  in = 1 currentstate = 010 nextstate = 011 out = 1
                  85  in = 0 currentstate = 011 nextstate = 100 out = 1
                  95  in = 0 currentstate = 100 nextstate = 100 out = 0
                 100  in = 1 currentstate = 100 nextstate = 000 out = 1
                 105  in = 1 currentstate = 000 nextstate = 001 out = 1
                 115  in = 0 currentstate = 001 nextstate = 001 out = 1
$finish called at time : 150 ns : File "C:/Users/User/project_2/project_2.srcs/sim_1/new/a.v" Line 44
```

---作答區結束---