Verilog project6 b10601002 廖品捷

Github: https://github.com/JamesLiao714/DLD-verilog

(1) Write the behavioral description of a Moore FSM described by Figure 1.

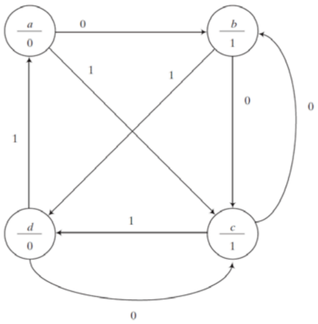
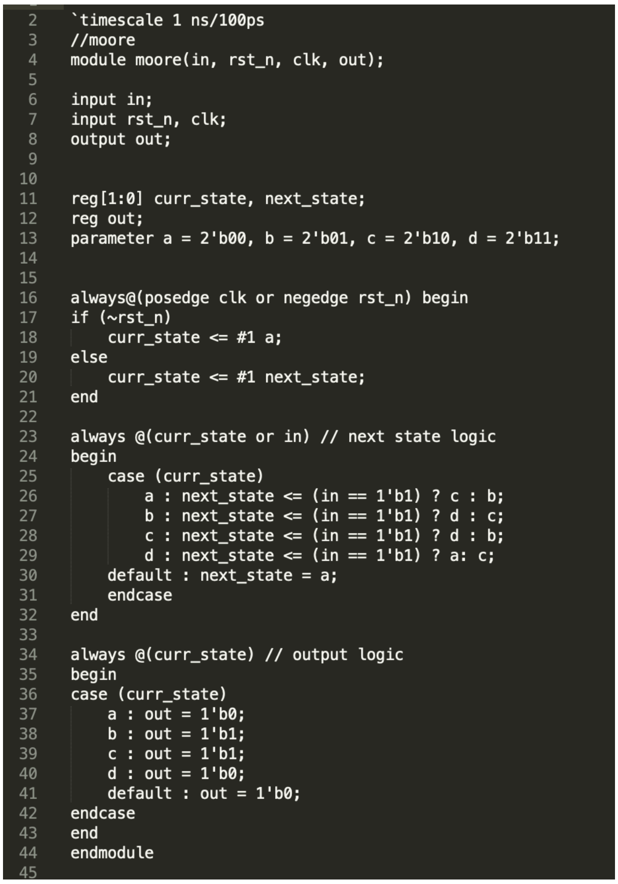


Figure 1. A Moore FSM.

我將a,b,c,d分別設為parameter 00,01,10,11 一開始當input為0時, state會跑到b,output為1…以此類推，此種machine的output不受input影響。

code:



```
`timescale 1 ns/100ps
//moore
module moore(in, rst_n, clk, out);

input in;
input rst_n, clk;
output out;


reg[1:0] curr_state, next_state;
reg out;
parameter a = 2'b00, b = 2'b01, c = 2'b10, d = 2'b11;


always@(posedge clk or negedge rst_n) begin
if (~rst_n)
    curr_state <= #1 a;
else
    curr_state <= #1 next_state;
end

always @(curr_state or in) // next state logic
begin
    case (curr_state)
        a : next_state <= (in == 1'b1) ? c : b;
        b : next_state <= (in == 1'b1) ? d : c;
        c : next_state <= (in == 1'b1) ? d : b;
        d : next_state <= (in == 1'b1) ? a: c;
    default : next_state = a;
    endcase
end

always @(curr_state) // output logic
begin
case (curr_state)
    a : out = 1'b0;
    b : out = 1'b1;
    c : out = 1'b1;
    d : out = 1'b0;
    default : out = 1'b0;
endcase
end
endmodule

```

(2) Write the behavioral description of a Mealy FSM described by Figure 2

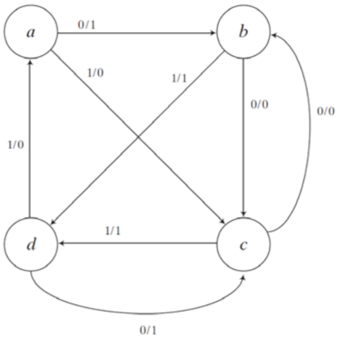
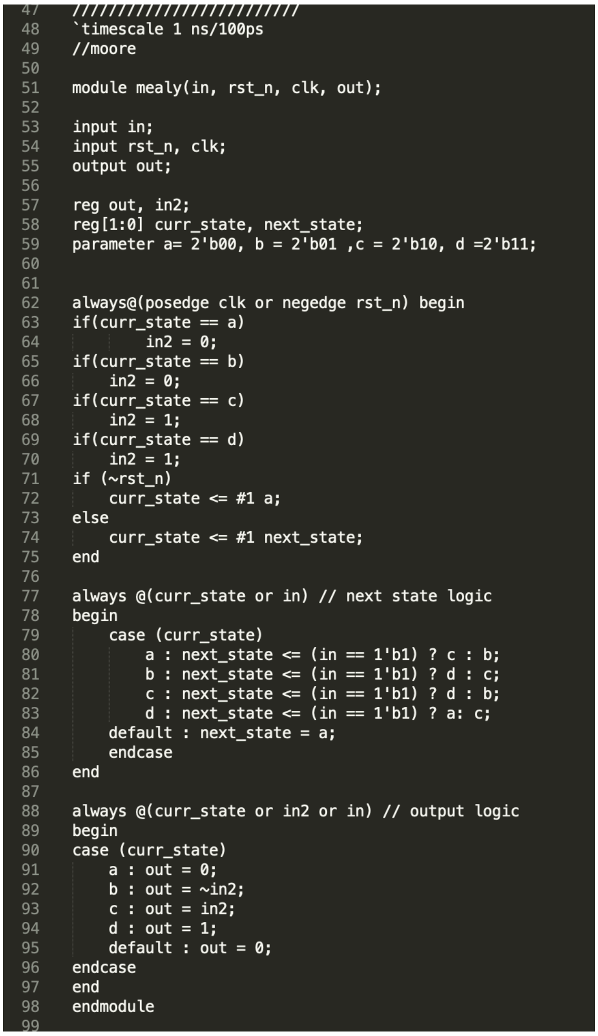


Figure 2. A Mealy FSM.

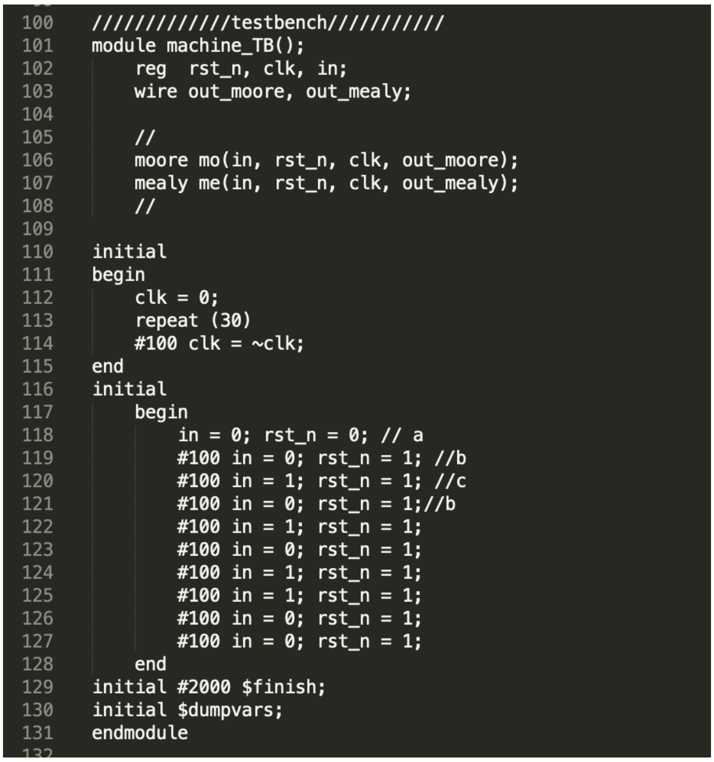
我將a,b,c,d分別設為parameter 00,01,10,11 一開始當input為1時, state會跑到c, output為0…以此類推，此種machine的output受input影響。

code:



```
//////////////////////////
`timescale 1 ns/100ps
//moore

module mealy(in, rst_n, clk, out);

input in;
input rst_n, clk;
output out;

reg out, in2;
reg[1:0] curr_state, next_state;
parameter a= 2'b00, b = 2'b01 ,c = 2'b10, d =2'b11;


always@(posedge clk or negedge rst_n) begin
if(curr_state == a)
        in2 = 0;
if(curr_state == b)
    in2 = 0;
if(curr_state == c)
    in2 = 1;
if(curr_state == d)
    in2 = 1;
if (~rst_n)
    curr_state <= #1 a;
else
    curr_state <= #1 next_state;
end

always @(curr_state or in) // next state logic
begin
    case (curr_state)
        a : next_state <= (in == 1'b1) ? c : b;
        b : next_state <= (in == 1'b1) ? d : c;
        c : next_state <= (in == 1'b1) ? d : b;
        d : next_state <= (in == 1'b1) ? a: c;
    default : next_state = a;
    endcase
end

always @(curr_state or in2 or in) // output logic
begin
case (curr_state)
    a : out = 0;
    b : out = ~in2;
    c : out = in2;
    d : out = 1;
    default : out = 0;
endcase
end
endmodule
```

test bench:



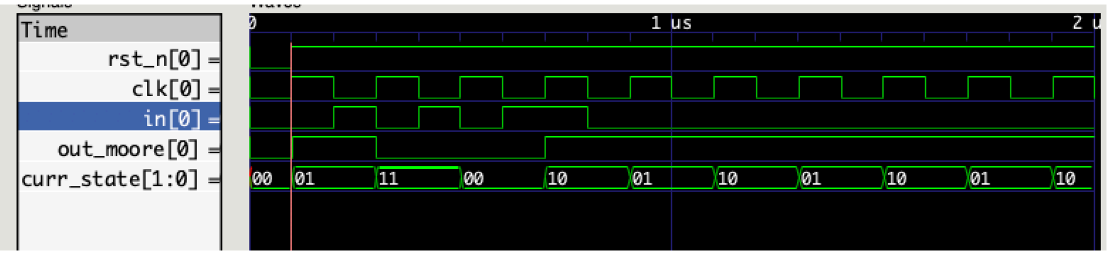
```
/////////////testbench///////////
module machine_TB();
    reg  rst_n, clk, in;
    wire out_moore, out_mealy;

    //
    moore mo(in, rst_n, clk, out_moore);
    mealy me(in, rst_n, clk, out_mealy);
    //

initial
begin
    clk = 0;
    repeat (30)
    #100 clk = ~clk;
end
initial
    begin
        in = 0; rst_n = 0; // a
        #100 in = 0; rst_n = 1; //b
        #100 in = 1; rst_n = 1; //c
        #100 in = 0; rst_n = 1;//b
        #100 in = 1; rst_n = 1;
        #100 in = 0; rst_n = 1;
        #100 in = 1; rst_n = 1;
        #100 in = 1; rst_n = 1;
        #100 in = 0; rst_n = 1;
        #100 in = 0; rst_n = 1;
    end
initial #2000 $finish;
initial $dumpvars;
endmodule
```

waveform:

實作模擬如下圖

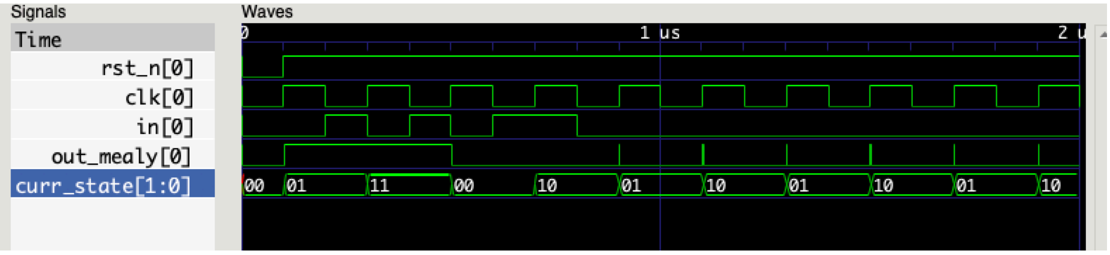
1. Moore FSM

   此 positive edge trigger 一開始透過 rst_n 讓 state initiate 為 a(00)，我們可以透過 curr_state 來查看 state 的跑動和 input 的關聯。



2. Mealy FSM

   此 positive edge trigger 一開始透過 rst_n 讓 state initiate 為 a(00)，我們可以透過 curr_state 來查看 state 的跑動,output 和 input 的關聯。實作結果中，也可以看到 glitch 的發生，因為我有用一個 in2 register 來計算 ouput 的值，而造成一些 delay。


Signals
Waves
Time
rst_n[0]
clk[0]
in[0]
out_mealy[0]
curr_state[1:0]
0
1 us
2 u
00
01
11
00
10
01
10
01
10
01
10