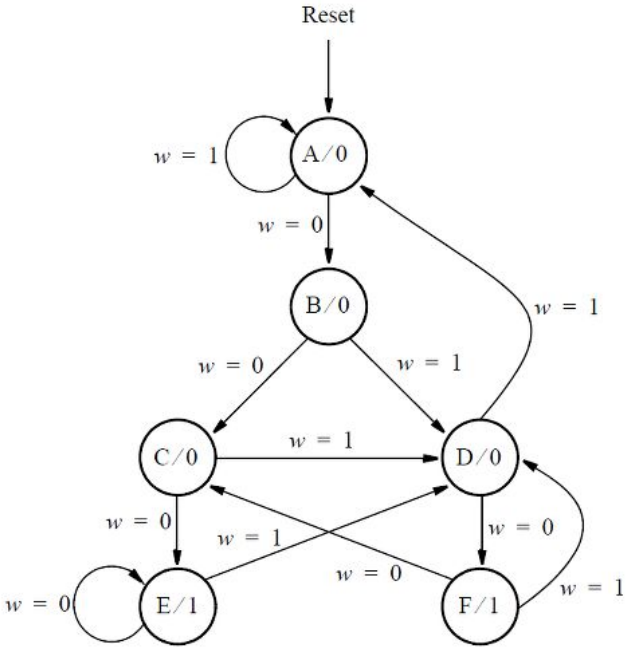
Exams/m2014 q6

Consider the state machine shown below, which has one input *w* and one output *z*.



Implement the state machine. (This part wasn't on the midterm, but coding up FSMs is good practice).

依照前兩題的脈絡 寫一個 one-hot finite state machine

```
module top_module (
    input clk,
    input reset,     // synchronous reset
    input w,
    output z);

    parameter [6:1] A=6'b000001;
    parameter [6:1] B=6'b000010;
    parameter [6:1] C=6'b000100;
    parameter [6:1] D=6'b001000;
    parameter [6:1] E=6'b010000;
    parameter [6:1] F=6'b100000;

    reg [6:1] curr_state;
    reg [6:1] next_state;

    always@(posedge clk)begin
        if(reset)begin
            curr_state <= A;
        end
        else begin
            curr_state <= next_state;
        end
    end
```

```
   always @(*)begin
      case(curr_state)
         A:next_state =  w? A:B;
         B:next_state =  w? D:C;
         C:next_state =  w? D:E;
         D:next_state =  w? A:F;
         E:next_state =  w? D:E;
         F:next_state =  w? D:C;
         default:next_state = A;
      endcase
   end

   assign z = curr_state[5] | curr_state[6];
              // E        |      F;

endmodule
```