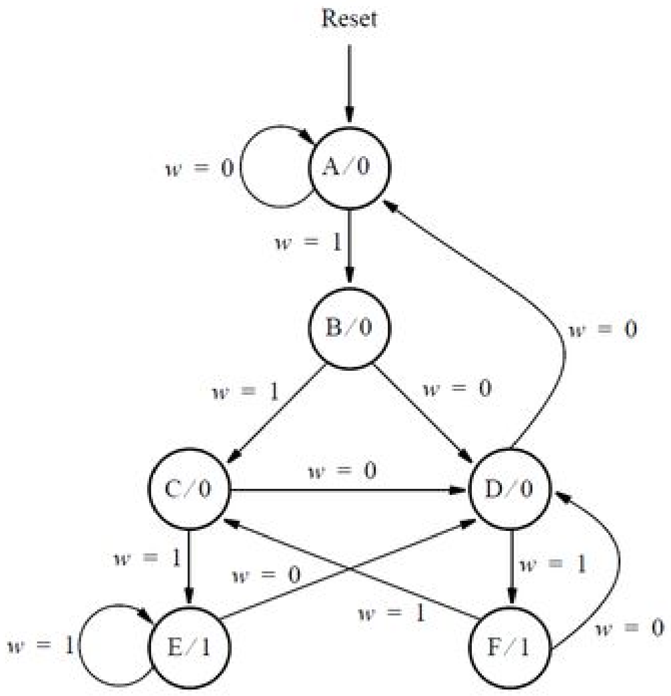
Exams/2012 q2fsm

Consider the state diagram shown below.



Write complete Verilog code that represents this FSM. Use separate **always** blocks for the state table and the state flip-flops, as done in lectures. Describe the FSM output, which is called *z*, using either continuous assignment statement(s) or an **always** block (at your discretion). Assign any state codes that you wish to use.

跟上一題類似 只是上一題在找連續0, 這一頭找的是連續1, 改一下next_state的case跳動就可以了。

```
module top_module (
    input clk,
    input reset,   // Synchronous active-high 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? B:A;
            B:next_state =  w? C:D;
            C:next_state =  w? E:D;
            D:next_state =  w? F:A;
            E:next_state =  w? E:D;
            F:next_state =  w? C:D;
            default:next_state = A;
        endcase
    end

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

endmodule
```