```
module HW2Q2(restart,pause, goto_third, out1, out2, odd, even, terminal) ;


input logic restart, pause, goto_third;
output logic [2:0] out1, out2;
output logic odd, even, terminal;


enum {first, second, third, fourth, fifth} state;


always_comb begin

state<= first;
terminal<= 0;
out1 <= 3'd3;
out2 <= 3'd2;
even <= 0;
odd <= 1;



case(state)

first: begin

    if ( restart==1 | pause ==1)begin
    state<= first;
    terminal<= 0;
    out1 <= 3'd3;
    out2 <= 3'd2;
    even <= 0;
    odd <= 1;

    end

    else begin
    state<= second;
    terminal<=0;
    out1 <= 3'd5;
    out2 <= 3'd4;
    even <= 1;
    odd <= 0;

    end

end

second: begin


    if ( restart == 1) begin

    state<= first;
    terminal<= 0;
    out1 <= 3'd3;
    out2 <= 3'd2;
    even <= 0;
    odd <= 1;

    end


    else if ( restart == 0 && pause == 1) begin

    state<= second;
    terminal<=0;
    out1 <= 3'd5;
    out2 <= 3'd4;
```

```
        even <= 1;
        odd <= 0;

        end



        else if ( restart == 0 && pause == 0) begin

        state<= third;
        terminal <=0;
        out1 <= 3'd2;
        out2 <= 3'd7;
        even<=0;
        odd<= 1;
        end


    end

    third: begin

        if ( restart == 1) begin

        state<= first;
        terminal<= 0;
        out1 <= 3'd3;
        out2 <= 3'd2;
        even <= 0;
        odd <= 1;

        end



        else if ( restart == 0 && pause == 1) begin

        state<= third;
        terminal<=0;
        out1 <= 3'd2;
        out2 <= 3'd7;
        even <= 0;
        odd <= 1;

        end



        else if ( restart == 0 && pause == 0) begin

        state<= fourth;
        terminal <=0;
        out1 <= 3'd6;
        out2 <= 3'd3;
        even<=1;
        odd<= 0;
        end


    end

    fourth: begin

        if ( restart == 1) begin

        state<= first;
        terminal<= 0;
        out1 <= 3'd3;
        out2 <= 3'd2;
        even <= 0;
        odd <= 1;

        end

        else if ( restart == 0 && pause == 1) begin

        state<= fourth;
        terminal <=0;
        out1 <= 3'd6;
        out2 <= 3'd3;
        even<=1;
        odd<= 0;
        end

        else if ( restart == 0 && pause == 0) begin

        state<= fifth;
        terminal <=1;
        out1 <= 3'd5;
        out2 <= 3'd2;
        even<=0;
        odd<= 1;
        end



    end

    fifth: begin

        if ( goto_third == 1) begin

        state<= third;
        terminal<=0;
        out1 <= 3'd2;
        out2 <= 3'd7;
        even <= 0;
        odd <= 1;

        end


        else if ( restart == 1 && goto_third == 0) begin

        state<= first;
        terminal<= 0;
        out1 <= 3'd3;
        out2 <= 3'd2;
        even <= 0;
        odd <= 1;

        end

        else if ( restart == 0 && pause == 1 && goto_third == 0) begin

        state<= fifth;
        terminal <=1;
        out1 <= 3'd5;
        out2 <= 3'd2;
        even<=0;
        odd<= 1;
        end



    end
```

```
default: state<= first;

endcase

//end



end


endmodule
```