

## 課題17-5

---

### コード

#### TopModule.v

```
module TopModule(
    //////////// CLOCK ////////////
    input                CLK1,
    input                CLK2,
    //////////// SEG7 ////////////
    output [7:0]         HEX0,
    output [7:0]         HEX1,
    output [7:0]         HEX2,
    output [7:0]         HEX3,
    output [7:0]         HEX4,
    output [7:0]         HEX5,
    //////////// Push Button ////////////
    input [1:0]          BTN,
    //////////// LED ////////////
    output [9:0]         LED,
    //////////// SW ////////////
    input [9:0]          SW

);
wire clk, res, wq;
wire [3:0] wq;
wire [7:0] hex;

m_rs_flipflop u1(BTN[0], BTN[1], clk, wq); //clock

assign res = SW[0];                //reset

m_counter(clk, res, wq);           //counter

m_seven_segment u2(wq, hex);

assign LED={6'h0,wq};
assign HEX0=hex;
assign HEX1=8'hff;
assign HEX2=8'hff;
assign HEX3=8'hff;
assign HEX4=8'hff;
assign HEX5=8'hff;

endmodule
```

#### Counter.v

```

module m_rs_flipflop(input set,input reset,output q,output nq);
    assign q=~(set & nq);
    assign nq=~(reset & q);
endmodule

module m_counter( ck, res, q );
    input  ck, res;
    output [3:0] q;
    reg    [3:0] q;

    always @( posedge ck or posedge res )
    begin
        if( res == 1'b1 )
            q <= 4'h0;
        else
            q <= q + 4'h1 ;
        end
    end
endmodule

```

## SevenSegment.v

```

module m_seven_segment(input [3:0] idat,output [7:0] odat);

    function [7:0] LedHex;
        input [3:0] num;
        begin
            case (num)
                4'h0:      LedHex = 8'b11000000;  // 0
                4'h1:      LedHex = 8'b11111001;  // 1
                4'h2:      LedHex = 8'b10100100;  // 2
                4'h3:      LedHex = 8'b10110000;  // 3
                4'h4:      LedHex = 8'b10011001;  // 4
                4'h5:      LedHex = 8'b10010010;  // 5
                4'h6:      LedHex = 8'b10000010;  // 6
                4'h7:      LedHex = 8'b11111000;  // 7
                4'h8:      LedHex = 8'b10000000;  // 8
                4'h9:      LedHex = 8'b10011000;  // 9
                4'ha:      LedHex = 8'b10001000;  // A
                4'hb:      LedHex = 8'b10000011;  // b
                4'hc:      LedHex = 8'b11000110;  // C
                4'hd:      LedHex = 8'b10100001;  // d
                4'he:      LedHex = 8'b10000110;  // E
                4'hf:      LedHex = 8'b10001110;  // F
                default:    LedHex = 8'b11111111;  // LED OFF
            endcase
        end
    endfunction

    assign odat = LedHex(idat);

```

```
endmodule
```

## 動作確認

- ボタンを押すとカウントが進み、HEX0に16進数が表示された
- SW0をあげるとリセットされた

## 解説

TopModuleモジュール内で、プッシュボタンでクロックを生成し、SW[0]でリセット信号を生成している。Counterモジュール内で、クロックの立ち上がりでカウントが進み、リセット信号が1のときにカウントが0になるように実装している。

SevenSegmentモジュールで、カウント値を7セグメント表示に変換し、HEX0に表示している。LEDの下位4ビットにカウント値を表示している。