

## 実習 4-2 進数を 10 進数に表示するプログラム (1 桁)

```
module bin2dec (CLOCK_50, KEY, SW, HEX0);
```

```
    input CLOCK_50; // ボードクロック 50MHz
```

```
    input [3:0] KEY; // rstn (リセット) 用プッシュスイッチ
```

```
    input [17:0] SW; // 2 進数入力用スライドスイッチ
```

```
    output [6:0] HEX0; // 出力用 7 セグ LED
```

//以降はコメントです。コメントはわかりやすくするためのものなので、無理に書く必要はありません。

**ポート宣言:** 入力と出力は必ず基板に対応した表示にする必要があります。

input [17:0] SW;は input [3:0] SW; という形式でも問題はありません。入力信号は 4 ビット (1001 まで) しか使用しないため、4 ビットの指定で十分です。

```
    seg7_decimal_1digit u0(.clk(CLOCK_50), .rstn(KEY[0]), .in(SW[3:0]), .out0(HEX0));
```

```
endmodule
```

seg7\_decimal\_1digit モジュールを呼びだします。CLOCK\_50 は seg7\_decimal\_1digit 内の clk に対応しています。

```
module seg7_decimal_1digit (clk, rstn, in, out0);
```

```
    input clk, rstn;
```

```
    input [3:0] in; // 10 進数に変換したい 2 進数
```

```
    output [6:0] out0; // 各 7 セグ LED が表示すべき 10 進数
```

in には、スイッチで入力された 2 進数が入ります。

reg はレジスタを表し、記憶素子を意味します。詳細は「Verilog-HDL の基本文法」を参照してください

```
    reg [3:0] tmp_reg; //変換用計算用レジスタ
```

```
    reg [3:0] digit0; // 各 7 セグ LED が表示すべき 10 進数
```

```
    seg7_decoder d0(.in(digit0), .out(out0));
```

```
    always @(posedge clk) begin
```

```
        if (!rstn) begin
```

```
            tmp_reg <= in;
```

```
            digit0 <= 4'd0;
```

```
        end else begin
```

if (rstn)だと、rstn=1

if (!rstn)は論理否定で rstn=0 のときに動作し、

このとき tmp\_reg に in の値を、digit0 に 0 を代入します。

if (rstn == 1'b0)や if (~rstn)という記述でも同様の動作になります。

```

        digit0 <= tmp_reg;

    end

end

endmodule

```

リセットボタンが押されていない場合、digit0 には tmp\_reg の値、つまりスイッチで入力した値が代入されます。

```

module seg7_decorder (in, out);

```

in には digit0 が代入されます。つまり、現在はスイッチで入力された 2 進数が in に入っています。

```

    input [3:0] in;

```

```

    output [6:0] out;

```

out には、7 セグメントディスプレイのどの部分を点灯させるかを指示する信号が入ります。0 が点灯する部分を示し、1 が消灯する部分を示します。

```

    assign out = (in == 4'd0)? 7'b1000000 :
                  (in == 4'd1)? 7'b1111001 :
                  (in == 4'd2)? 7'b0100100 :
                  (in == 4'd3)? 7'b0110000 :
                  (in == 4'd4)? 7'b0011001 :
                  (in == 4'd5)? 7'b0010010 :
                  (in == 4'd6)? 7'b0000010 :
                  (in == 4'd7)? 7'b1111000 :
                  (in == 4'd8)? 7'b0000000 :
                  (in == 4'd9)? 7'b0010000 : 7'b1111111;

```

```

endmodule

```

## 7-segment Displays

