

# Controle de Periféricos - Projeto 6 - sled

## Controle de Periféricos - Projeto 6 - sled

Projeto em funcionamento

## Código do projeto

### Código do módulo sled

```
module sled( input wire clock, output wire [7:0] segs, output wire [3:0] digs );
    // Variavel com os dados a serem exibidos no display de 7 segmentos
    reg [3:0] disp_dat;
    reg [7:0] segs_dsp;
    // Contador:
    reg [36:0] count;

    // Ligar os 4 digitos do display
    assign digs = 4'b0000;

    // --> Bloco always executado nas bordas de subida do clock:
    always @ (posedge clock )
    begin
        // Incrementar 'count'
        count = count + 1'b1;
    end

    // --> Bloco always executado quando o 25o bit de 'count' for 1
    always @ ( count[24] )
    begin
        disp_dat = { count[28:25] };
    end

    // --> Bloco 'always' executado a cada mudanca em 'disp_dat'
    always @ ( disp_dat )
    begin
        case (disp_dat)
            4'h0 : segs_dsp = 8'b11000000;
            4'h1 : segs_dsp = 8'b11111001;
            4'h2 : segs_dsp = 8'b10100100;
            4'h3 : segs_dsp = 8'b10110000;
            4'h4 : segs_dsp = 8'b10011001;
            4'h5 : segs_dsp = 8'b10010010;
            4'h6 : segs_dsp = 8'b10000010;
            4'h7 : segs_dsp = 8'b11111000;
            4'h8 : segs_dsp = 8'b10000000;
            4'h9 : segs_dsp = 8'b10010000;
```

```

        4'ha : segs_dsp = 8'b10001000;
        4'hb : segs_dsp = 8'b10000011;
        4'hc : segs_dsp = 8'b11000110;
        4'hd : segs_dsp = 8'b10100001;
        4'he : segs_dsp = 8'b10000110;
        4'hf : segs_dsp = 8'b10001110;

    endcase

end

// --> Atribuicao aos segmentos do display
assign segs = segs_dsp;
endmodule

```

## Código do módulo principal

```

module demo_ep4ce6(input wire FPGA_CLK, output wire [3:0] DIG, output wire [7:0] SEG );
    sled obj_7seg_dsp( .clock(FPGA_CLK), .segs(SEG), .digs(DIG) );
endmodule

```