

明德扬科技教育有限公司

周期跳转状态机练习

官 网: www.mdy-edu.com

淘 宝: mdy-edu.taobao.com

QQ 群: 97925396

QQ 咨询: 158063679

目录

state2 模块	3
-----------------	---

明德扬点拔FPGA课程

明德扬科技公司主要是以 FPGA 为核心,专业从事 FPGA 配套视频开发板教程、FPGA 培训班或其他培训、研发 FPGA 技术开发、承接 FPGA 项目开发。欢迎咨询加入明德扬 FPGA 和 ASIC 交流群 97925396。

明德扬以 PDF 格式提供源代码,是为了鼓励大家多思考,不要拿来就用,否则是学不好 FPGA 的。

本代码对应的设计思路,请参考明德扬视频课程。

state2 模块

```
/*  
*****  
*****
```

本代码由明德扬工作室精心设计和制作。

我们希望通过规范、严谨的代码,使同学们接触到纯正的集成电路/FPGA 代码。

明德扬工作室长期招募成员,可免费培训,欢迎集成电路/FPGA 爱好者加入。

学习交流群: 97925396

```
*****  
*****/
```

```
/******www.mdy-edu.com 明德扬科教 注释开始*****
```

模块功能说明: 状态机有三个状态, IDLE, S1 和 S2。在 IDLE 状态, 如果 en=1, 则跳到 S1 状态, 否则保持不变; 在 S1 状态时, 如果 en=1, 则跳到 S2 状态, 否则保持不变; 在 S2 状态, 如果 en=1, 则跳回到 IDLE 状态, 否则保持不变。

接口定义:

clk : 时钟信号, 频率为 50MHz

rst_n : 复位信号, 低电平有效

en : 状态转换信号, 高电平有效

```
*****www.mdy-edu.com 明德扬科教 注释结束*****/
```

```
module state2(clk,rst_n,en,state_c);

/*****www.mdy-edu.com 明德扬科教 注释开始*****/
参数定义，明德扬规范要求，verilog 内的用到的数字，都使用参数表示。
参数信号全部大写
*****/

parameter STATE_WID = 2;
parameter IDLE = 2'b00;
parameter S1 = 2'b01;
parameter S2 = 2'b10;

input clk;
input rst_n;
input en;
output [STATE_WID-1:0] state_c;

reg [STATE_WID-1:0] state_c;

reg [STATE_WID-1:0] state_n;

/*****www.mdy-edu.com 明德扬科教 注释开始*****/
思路：定义两个寄存器来表示当前状态和下一状态，然后根据 en 的值
来改变寄存器的值。
*****/

/*****www.mdy-edu.com 明德扬科教 注释开始*****/
复位后，当前状态（state_c）为状态 0；
否则，每一个 clk 上升沿来临都使当前状态=下一状态。
*****/

always @(posedge clk or negedge rst_n)begin
    if(rst_n==1'b0)begin
        state_c<=IDLE;
    end
    else begin
        state_c<=state_n;
    end
end

/*****www.mdy-edu.com 明德扬科教 注释开始*****/
当 en=1 时，下一状态才转换（转换规则见模块功能说明）；
否则，下一状态维持当前状态。
*****/
```

```
always  @(*)begin
    case(state_c)
        IDLE:begin
            if(en==1'b1)begin
                state_n=S1;
            end
            else begin
                state_n=IDLE;
            end
        end
        S1:begin
            if(count==4)begin
                state_n=S2;
            end
            else begin
                state_n=S1;
            end
        end
        S2:begin
            if(count==6)begin
                state_n=IDLE;
            end
            else begin
                state_n=S2;
            end
        end
        default:begin
            state_n=state_c;
        end
    endcase
end

always  @(posedge clk or negedge rst_n)begin
    if(rst_n==1'b0)begin
        count <= 0;
    end
    else if(state_c==S1)begin
        if(count==4)
            count <= 0;
        else
            count <= count + 1;
    end
    else if(state_c==S2)begin
```

```
        if(count==6)
            count <= 0;
        else
            count <= count + 1;
        end
    else begin
        count <= 0;
    end
end
end

endmodule
```