



明德扬  
科技·教育

# 点拨 · FPGA之 状态机练习2设计思路

点透学习误区 拨出设计精髓

主讲：潘文明

# 明德扬科教



**QQ群: 97925396**

**官 网: <http://www.mdy-edu.com>**

**淘 宝: <http://mdy-edu.taobao.com>**

# 课程大纲

1. 功能要求
2. 设计思路(状态机设计)
3. 代码设计

# 一、功能要求

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，保持5个时钟周期后，跳到S2状态；在S2状态，保持7个时钟周期后，跳回到IDLE状态。

信号名↵	I/O↵	位宽↵	说明↵
clk↵	I↵	1↵	系统工作时钟 50M↵
rst_n↵	I↵	1↵	系统复位信号，低电平有效↵
en↵	I↵	1↵	状态转换信号，高电平有效↵
state_c↵	O↵	2↵	当前状态机的值↵

## 二、设计思路

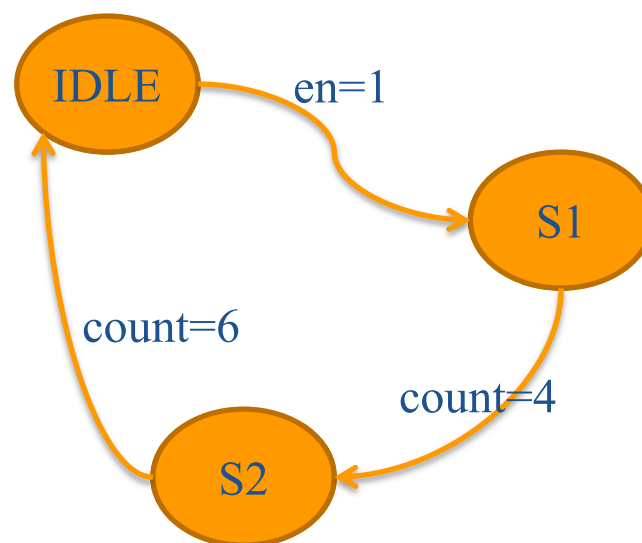
1. 一个一个信号设计，逐个击破
2. 状态机的转移条件，要精确到1个时钟周期

## 三、设计思路—state

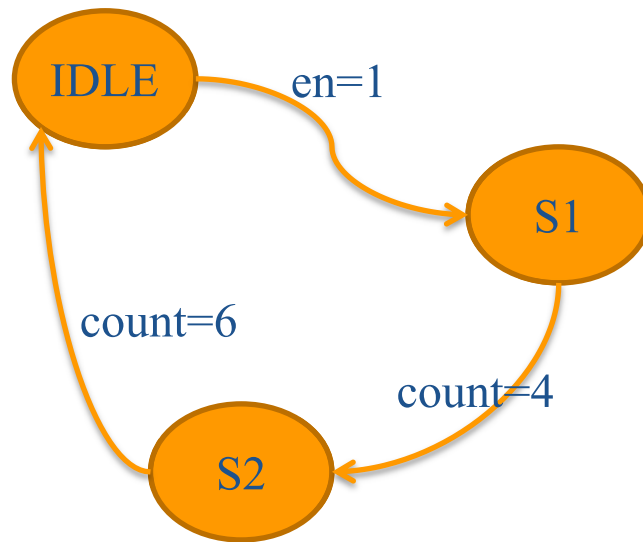
1. 当前状态机的值
2. 状态机有三个状态，IDLE，S1和S2。

### 三、状态机设计

1. 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，保持5个时钟周期后，跳到S2状态；在S2状态，保持7个时钟周期后，跳回到IDLE状态。



# 三、状态机设计



```

always @(*)begin
  case(state_c)
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
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
  
```



## 三、信号设计--count

1. 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，保持5个时钟周期后，跳到S2状态；在S2状态，保持7个时钟周期后，跳回到IDLE状态。

初值：0；加1条件：S1或S2状态；结束条件：S1时为4，S2时为6

```
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
```

## 四、总结

1. 转移条件要精确到1个时钟
2. 条件不够时，要构造信号。这个过程，就相当于设计过程。

# 明德扬科教



**QQ群: 97925396**

**官 网: <http://www.mdy-edu.com>**

**淘 宝: <http://mdy-edu.taobao.com>**



# Thank You !

