# 时序逻辑

always@(posedge clk or negedge rst\_n) begin

if( rst\_n == 1'b0 )

<= ;

else begin

<= ;

end

end

# 组合逻辑

always@(\*) begin

if( rst\_n == 1'b0 )

= ;

else begin

= ;

end

end

# 状态机

//第一段:同步时序always模块，格式化描述次态寄存器迁移到现态寄存器(不需更改)

always@(posedge clk or negedge rst\_n)begin

if(!rst\_n)begin

state\_c<= IDLE;

end

else begin

state\_c<= state\_n;

end

end

//第二段:组合逻辑always模块，描述状态转移条件判断

always@(\*)begin

case(state\_c)

IDLE: begin

if(idl2sl\_start)begin

state\_n=S1;

end

else begin

state\_n = state\_c;

end

end

S1:begin

if(s12s2\_start)begin

state\_n = S2;

end

else begin

state\_n = state\_c;

end

end

S2:begin

if(s22s3 start)begin

state\_n = S3;

end

else begin

state\_n= state\_c;

end

end

default:begin

state\_n = IDLE;

end

endcase

end

//第三段:设计转移条件

assign id12s1\_start = state\_c==IDLE && ;

assign sl2s2\_start = state\_c==Sl && ;

assign s22s3\_start = state\_c==S2 && ;

//第四段:同步时序always模块，格式化描述寄存器输出(可有多个输出)

always@(posedge clk or negedge rst\_n)begin

if(!rst\_n)begin

outl <= 1’b0;//初始化

end

else if(state\_c==Sl)begin

outl <= 1’bl;

end

else begin

out1 <= 1’b0;

end

end