

第二次实验报告

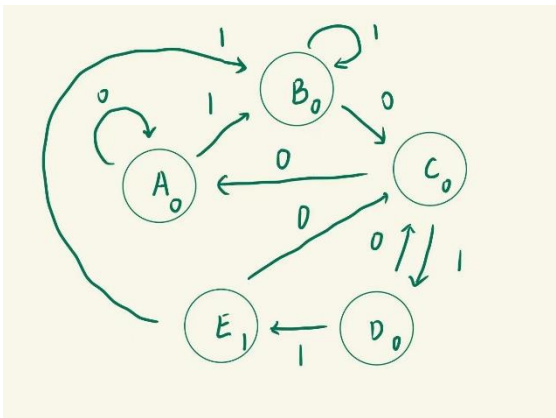
陈刚老师班级 22336313 郑鸿鑫

一 实验内容

本实验实现了一个有限状态机来进行序列检测，用该状态机来检测序列“1011”，串行输入的测试信号为“1110110110111011”。输出信号为 valid，检测到有效序列时，输出为 1，否则为 0，考虑序列叠加情况。比如“1011011”，则有两个“1011”，这需要输出两个 valid 有效信号。主要有一个模块 FSM1011 构成以实现功能。

二 实验过程

先根据要求画出有限状态机的转换图，如图所示：



图中 ABCDE 为状态机的 5 个状态，其中只有在状态 E 时 valid 为 1 其余状态下都为 0，图中箭头上的数字表示对应的输入，表示当前状态下输入为 0 的下一个状态是什么或者输入为 1 时下一个状态是什么。

再根据有限状态机的状态转换图编写 FSM1011 的模块代码如下：(代码的功能解释已经在注释中给出)

```

module FSM1011(
    input CLK,//用于传入时钟信号
    input [31:0]Sq,//传入输入的测试序列
    output reg valid//输出是否检测到“1011”
);
localparam A = 0,B = 1,C = 2,D = 4,E = 5;//定义状态常量
reg[4:0] index = 31;
reg in;//用于接收输入序列的单个字符
reg [2:0] state,nxtState;//用于表示当前状态和下一状态
initial state = A;//初始化为 A
always @(posedge CLK)begin//每次时钟上升沿到来时，in 接收一个字符
    if(index!=0)begin
        in = Sq[index];
        index = index-1;
    end

    case(state)
        A: if(in) begin nxtState = B;valid = 0; end
           else begin nxtState = A;valid = 0; end
        B: if(in) begin nxtState = B;valid = 0; end
           else begin nxtState = C;valid = 0; end
        C: if(in) begin nxtState = D;valid = 0; end
           else begin nxtState = A;valid = 0; end
        D: if(in) begin nxtState = E;valid = 1; end
           else begin nxtState = C;valid = 0; end
        E: if(in) begin nxtState = B;valid = 0; end
           else begin nxtState = C;valid = 0; end
        default:begin
            valid = 1'bX;
            nxtState = 3'bX;
        end
    endcase//判断状态的下一状态
    state = nxtState;
end
endmodule

```

然后编写完整的测试仿真文件 Testbench 代码（代码功能的解释在注释中给出）

```

module sim;
    reg CLK;//用于传入时钟信号
    reg [31:0]Sq;//传入输入的测试序列
    wire valid;//输出是否检测到“1011”
    FSM1011 uu(//实例化一个 FSM1011 模块用于检测序列
        .CLK(CLK),
        .Sq(Sq),
        .valid(valid)
    );
endmodule

```

```

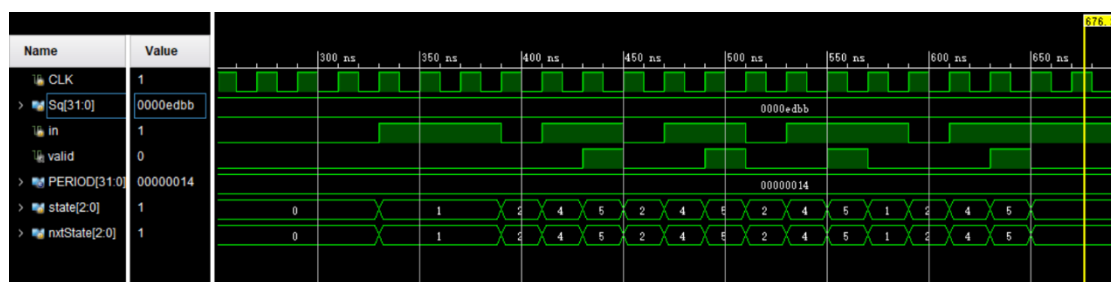
);
initial begin
    Sq = 32'b1110110110111011; //序列 Sq 赋初值为待测试的序列
    CLK = 0;
end
parameter PERIOD = 20;
always begin
    CLK=0;
    #(PERIOD/2);
    CLK=1;
    #(PERIOD/2);
end
endmodule

```

最后进行仿真运行，得到实验结果。

三 实验结果分析

仿真运行后，得到结果如下图：



分析：图中可以看出，in 每次在时钟上升沿到来的时候会接收一个 Sq 的字符并且当连续接收到“1011”以后 valid 将会变为高电平，而且从图中可以看出，即使当序列有重叠部分，也可以正常识别，如图中 valid 第一次变为 1 和第二次变为 1 之间的这段子序列为“11011011”可以使 valid 两次变为 1，则说明功能正常实现。

四 实验总结

实验结果符合预期，更换检测的序列，功能依旧正常实现。实验中遇到的问题：如果在一个 if 语句中写的语句超过 1 条，则需要用 begin 和 end 包括进去，否则会报错。对于一个 reg 变量赋值时，

注意他的数值所需要的位数，否则可能导致出错。还应注意区别阻塞赋值和非阻塞赋值。