

题目：序列检测器

在SystemVerilog中，你可以使用线性反馈移位寄存器（LFSR）来实现一个随机数生成器。以下是一个简单的LFSR实现，它使用了always块来在时钟边沿更新寄存器的值，并使用了异或运算来生成新的位。

```
module LFSR_random_generator(  
    input wire clk,  
    input wire reset,  
    output wire [7:0] random_out  
);  
  
    // LFSR的反馈多项式为  $x^8 + x^6 + x^5 + x^4 + 1$   
    reg [7:0] lfsr;  
    always_ff @(posedge clk or posedge reset) begin  
        if (reset) begin  
            // 当复位信号为高时，将LFSR重置  
            lfsr <= 8'b1;  
        end else begin  
            // 计算新的LFSR值  
            lfsr <= {lfsr[6:0], lfsr[7] ^ lfsr[5] ^ lfsr[4] ^ lfsr[3]};  
        end  
    end  
  
    // 将LFSR的值输出作为随机数  
    assign random_out = lfsr;  
  
endmodule
```

要求一：动态随机序列显示

- 将LFSR生成的8位随机序列从高位到低位依次对应到8个LED灯。
- LED显示需以2秒为周期动态更新。
- 添加复位按键，复位按下后所有LED熄灭，抬起后恢复动态更新。

要求二：1的密度检测

- 添加1个带译码器数码管，若当前显示的8位序列中1的数量 ≥ 4 ，数码管显示1；否则显示0；
- 复位按下后数码管显示0，所有LED熄灭，抬起后恢复电路功能。

推荐检测逻辑用组合电路实现

要求三：停止反查

在要求二的基础上实现如下功能，保存上一个“1”的数量 ≥ 4 的数列，增加一个拨码开关和按键，当拨码开关为低电平时电路工作模式同功能二。当拨码开关为高电平时暂停更新8位随机序列，在暂停状态下可以通过按下并松开按键来切换查看当前的随机序列和保存的上一个 ≥ 4 的随机序列。