明德扬科技教育有限公司

利用 random 产生随机数练习

官　网：www.mdy-edu.com

淘　宝：mdy-edu.taobao.com

QQ　群：97925396

QQ 咨询：158063679

目录

明德扬

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。

testbench4 模块

```verilog
`timescale 1 ns/1 ns

module testbench4();

//时钟和复位
reg clk   ;
reg rst_n;

//uut 的输入信号
reg               din_vld   ;
reg[7:0]          din       ;
reg               din_sop   ;
reg               din_eop   ;
reg               din_err   ;
reg[7:0]          len       ;

    //uut 的输出信号
    wire              dout_vld   ;
    wire[7:0]         dout       ;
    wire              dout_sop   ;
    wire              dout_eop   ;
    wire              dout_err   ;

        //时钟周期，单位为 ns，可在此修改时钟周期。
        parameter CYCLE    = 20;
```

//复位时间，此时表示复位 3 个时钟周期的时间。
parameter RST_TIME = 3 ;

integer   i   ;

//待测试的模块例化
mess_detection u_mess_detection(
    .clk   (clk   ),
    .rst_n   (rst_n   ),
    .din   (din   ),
    .din_sop   (din_sop),
    .din_eop   (din_eop),
    .din_vld   (din_vld),
    .din_err   (din_err),
    .dout   (dout   ),
    .dout_sop   (dout_sop),
    .dout_eop   (dout_eop),
    .dout_vld   (dout_vld),
    .dout_err   (dout_err)
);

    //生成本地时钟 50M
    initial begin
        clk = 0;
        forever
        #(CYCLE/2)
        clk=~clk;
    end

    //产生复位信号
    initial begin
        rst_n = 1;
        #2;
        rst_n = 0;
        #(CYCLE*RST_TIME);
        rst_n = 1;
    end

    //
    initial begin
        #1;
        din_vld = 0;
        din   = 0;
        din_sop = 0;

```
            din_eop = 0;
            din_err = 0;

            #(5*CYCLE);                          //package_a
            for(j=0;j<=5;j=j+1)begin
                len = $random;
                len = len%99 + 2;
                for(i=0;i<=len;i=i+1)begin
                    din_vld = (i==len)?0:1;
                    din     = $random;
                    din_sop = (i==0 && j!=2)?1:0;
                    din_eop = ((i==len-1) && j!=4)?1:0;
                    #(CYCLE);
                end
                #(3*CYCLE);
            end

    /*          #(3*CYCLE);                      //package_b
            len = $random;
            len = len%99 + 2;
            for(i=0;i<=len;i=i+1)begin
                din_vld = (i==len)?0:1;
                din     = $random;
                din_sop = (i==0)?1:0;
                din_eop = (i==len-1)?1:0;
                #(CYCLE);
            end

            #(3*CYCLE);                          //package_c
            len = $random;
            len = len%99 + 2;
            for(i=0;i<=len;i=i+1)begin
                din_vld = (i==len)?0:1;
                din     = $random;
                din_sop = 0;
                din_eop = (i==len-1)?1:0;
                #(CYCLE);
            end

            #(3*CYCLE);                          //package_d
            len = $random;
            len = len%99 + 2;
            for(i=0;i<=len;i=i+1)begin
                din_vld = (i==len)?0:1;
```

```
            din     = $random;
            din_sop = (i==0)?1:0;
            din_eop = (i==len-1)?1:0;
            #(CYCLE);
        end

        #(3*CYCLE);                        //package_e
        len = $random;
        len = len%99 + 2;
        for(i=0;i<=len;i=i+1)begin
            din_vld = (i==len)?0:1;
            din     = $random;
            din_sop = (i==0)?1:0;
            din_eop = 0;
            #(CYCLE);
        end

        #(3*CYCLE);                        //package_f
        len = $random;
        len = len%99 + 2;
        for(i=0;i<=len;i=i+1)begin
            din_vld = (i==len)?0:1;
            din     = $random;
            din_sop = (i==0)?1:0;
            din_eop = (i==len-1)?1:0;
            #(CYCLE);
        end

        #(3*CYCLE);                        //package_f1
        len = $random;
        len = len%99 + 2;
        for(i=0;i<=len;i=i+1)begin
            din_vld = (i==len)?0:1;
            din     = (i==0)?0:(din+1);
            din_sop = (i==0)?1:0;
            din_eop = (i==len-1)?1:0;
            #(CYCLE);
        end

*/
    end

    endmodule
```