Code:

`timescale 1ns / 1ps

module bit32\_rand\_gen(

input clk,

input reset,

output [31:0] rand\_out

);

wire feedback=random[31]^random[21]^random[1]^random[0];

reg [31:0]random,rand\_next,random\_done;

reg [5:0]count,count\_next;

always@(posedge clk or posedge reset)

begin

if(reset)

begin

random<=32'hf;

count<=0;

end

else

begin

random<=rand\_next;

count<=count\_next;

end

end

always@(\*)

begin

rand\_next=random;

count\_next =count;

rand\_next={random[30:0],feedback};

count\_next=count+1;

$monitor("%b",count);

if(count == 32)

begin

count=0;

random\_done=random;

end

end

assign rand\_out=random\_done;

endmodule

Test\_bench:

`timescale 1ns / 1ps

module bit32\_lfsr\_tb;

reg clk;

reg reset;

wire [31:0]rand\_out;

bit32\_rand\_gen uut(

.clk(clk),

.reset(reset),

.rand\_out(rand\_out));

initial begin

clk=0;

forever

#50 clk=~clk;

end

initial

begin

reset=0;

#100

reset=1;

#200

reset=0;

end

initial

begin

$monitor("%b,%b",clk,rand\_out);

end

endmodule

Result:

