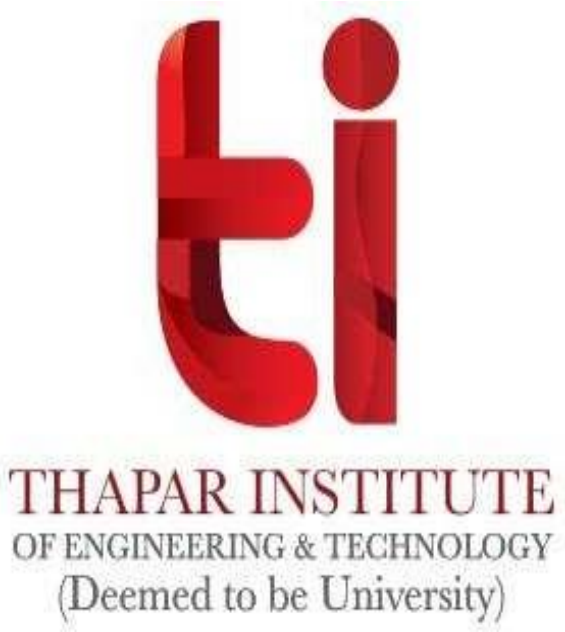
VLSI TESTING AND VERIFICATION

PVL-208

Micro Project Report

GROUP 1



Master of Technology

In

VLSI DESIGN

Submitted By:

Aman Gupta (602062001)

Anmol Agarwal (602062003)

Ashutosh Jaiswal (602062004)

Asjad Fahmi (602062005)

Baby Shabanaz (602062006)

Under the guidance of:

Dr. Gaganpreet Kaur

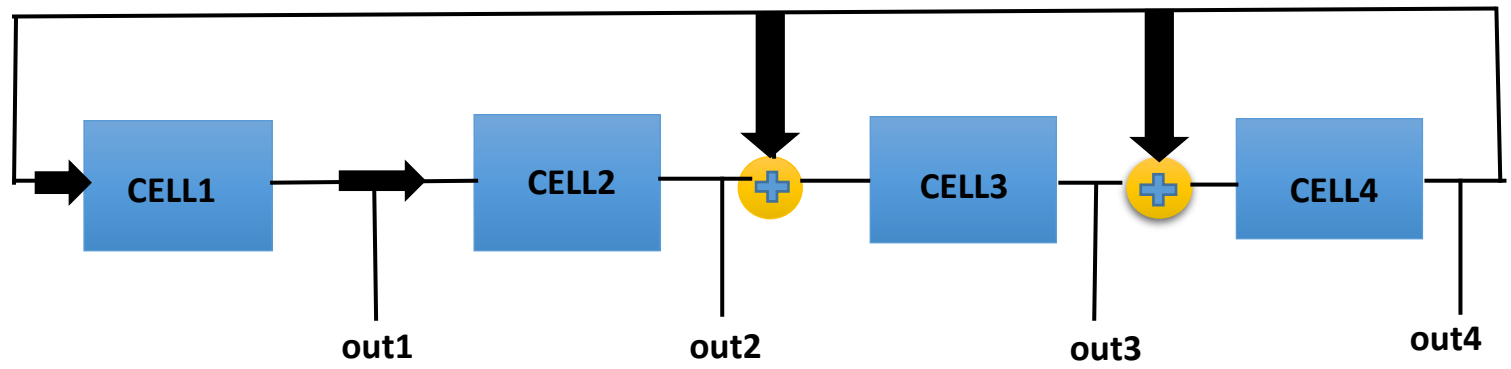
Assistant Professor

Dept. of ECED

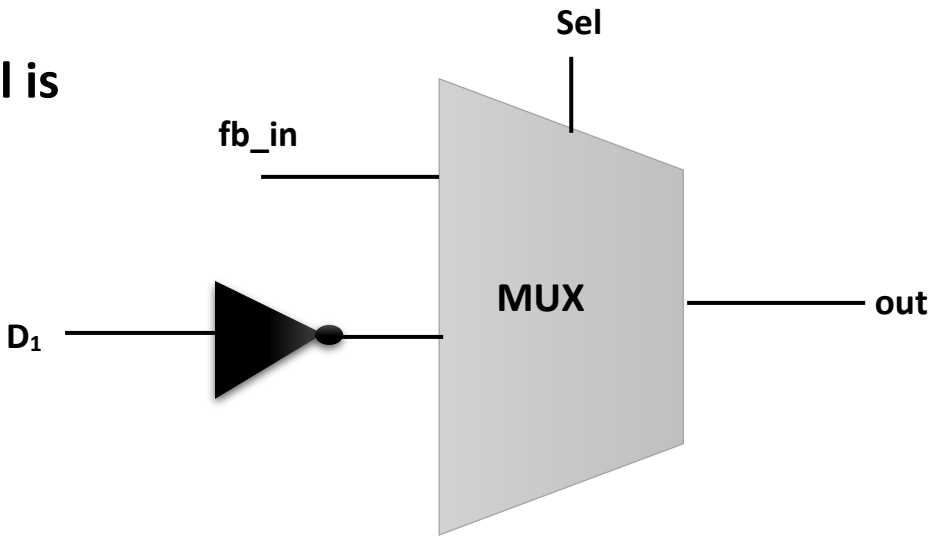
TABLE OF CONTENTS

Name of the Chapters **Page No.**

Problem Statement: Implement using VHDL / VERILOG



Each Cell is



D_1 is given initial non zero seed value while 'sel' signal for each cell is given randomly.

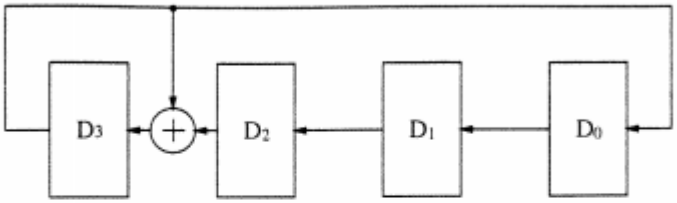
Introduction: Linear Feedback Shift Register

Use of linear feedback shift registers (LFSR) is being studied extensively by engineers, designers and researchers working in testing, design for testability and built-in self-environments. LFSRs are rather attractive structures for use in these environments for some of the following reasons.

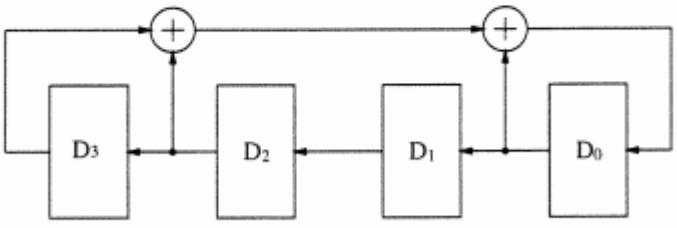
- LFSRs have a simple and fairly structure.
- Their shift property is easily integratable in the scan design environment.
- They are capable of generating exhaustive and/or random vectors.
- Their error detection and error correction properties make them a prime candidate for signature analysis applications.

Typically, the component used to construct them are D-flip flop and ex-or gates. Despite their simple appearance, LFSRs are based on rather complex mathematical theory and have a several interesting applications, particularly in the area of digital system testing and fault tolerant computing. Example of some of the applications are the random testing of logic circuits, fault signature analysis and error detecting/correcting codes.

There are following two type of LFSR structure is generally used.



(a): Internal EOR type



(b): External EOR type

Figure 1: Example LFSRs

Note that both these structure use D-FF and logic elements (EOR GATES) to realize LFSRs.

The basic difference in these two structures being the circuit of being the figure of figure 1(a) uses elements interspersed between the flip flops whereas the circuit of figure 1(b) has no linear element appearing between the flipflop instead the linear element appears only in one feedback path. It is for this reason the realization of figure 1(a) is called internal EOR LFSR and the realization of figure 1(b) is called as external EOR LFSR. An equivalence exists between the two structures in the sense that knowing the properties of the first structure one can deduce the properties of second structure.

APPLICATION OF LFSR

LFSR is used for different practical purposes. Some of the application of LFSR is given below.

1. LFSR as a test pattern generator.

Testing speed is important not only because testing is necessary step in a manufacturing l ine, but because it is often desirable to apply tests at operational speed. The problem of testing at operational speeds can be partially overcome by putting a high-speed memory in tester so that test pattern can be applied for short, high speed bursts. Nevertheless, these bursts are relatively short since block of patterns must still be retrieved from a secondary store.

An alternative to algorithmically generated tests which overcome these advantages is the use of pseudo-random or pseudo-exhaustive tests generated by an LFSR.

2. LFSR for response compaction.

- It uses cyclic redundancy check code (CRCC) generator (LFSR) for response compacter
- In this method, data bits from circuit Pos to be compacted as a decreasing order coefficient polynomial.
- CRCC divides the PO polynomial by its characteristic polynomial that leaves remainder of division in LFSR. LFSR must be initialized to seed value (usually 0) before testing.
- After testing, signature in LFSR is compared to known good machine signature

For an output sequence of length N, there is a total of 2N-1 faulty sequence. Let the input sequence is represented as P(x) as $P(x)=Q(X)G(x)+R(x)$. G(x) is the characteristic polynomial; Q(x) is the quotient; and R(x) is the remainder or signature. For those aliasing faulty sequence, the remainder R(x) will be the same as the fault-free one.

Since, P(x) is of order N and G(x) is of order r, hence Q(x) has an order of N-r. Hence, there are 2N-r possible Q(x) or P(x). One of them is fault-free.

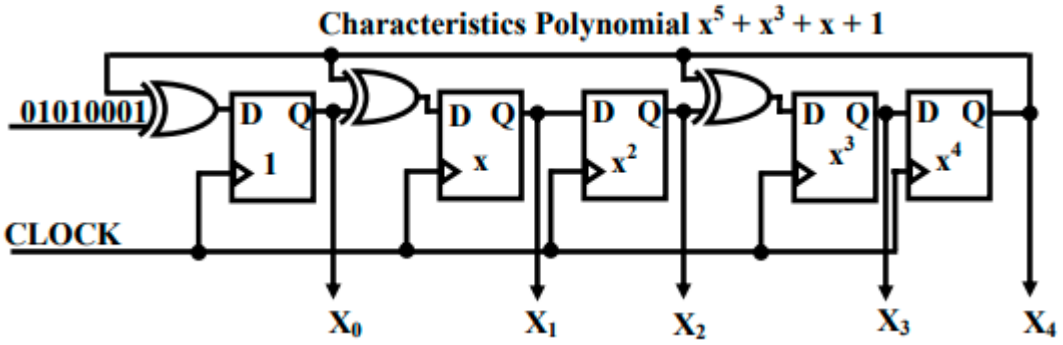


Fig: LFSR as a response compactor.

3. LFSR as pseudo random generator.

A string of 0's and 1's is called a pseudo-random binary sequence when the bits appear to be random in the local sense, but they are in some way repeatable. The linear feedback shift register (LFSR) pattern generator is most commonly used for pseudo-random pattern generation. In general, this requires more patterns than deterministic ATPG, but less than the exhaustive test. In contrast with other methods, pseudo-random pattern BIST may require a long test time and necessitate evaluation of fault coverage by fault simulation. This pattern type, however, has the potential for lower hardware and performance overheads and less design effort than the preceding methods. In pseudorandom test patterns, each bit has an approximately equal probability of being a 0 or a 1. The number of patterns applied is typically of the order of 103 to 107 and is related to the circuit's testability and the fault coverage required. Linear feedback shift register reseeding [5] is an example of a BIST technique that is based on controlling the LFSR state. LFSR reseeding may be static, that is LFSR stops generating patterns while loading seeds, or dynamic, that is, test generation and seed loading can proceed simultaneously. The length of the seed can be either equal to the size of the LFSR (full reseeding) or less than the LFSR (partial reseeding). In [5], a dynamic reseeding technique that allows partial reseeding is proposed to encode test vectors. A set of linear equations is solved to obtain the seeds, and test vectors are ordered to facilitate the solution of this set of linear equations.

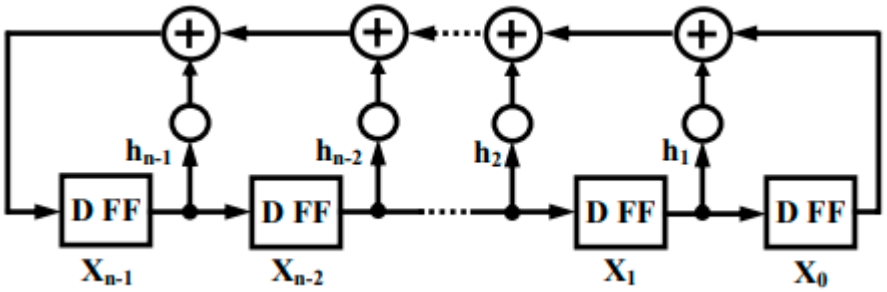


Fig: LFSR

Project Working Algorithm:

Circuit is similar to 4-bit internal EOR LFSR. Where each cell is 2:1 multiplexer.

- The input is selected on the basis of select line.
- If select line is '1' external input is selected.
- If select line is '0' feedback input is selected.

Verilog Code:

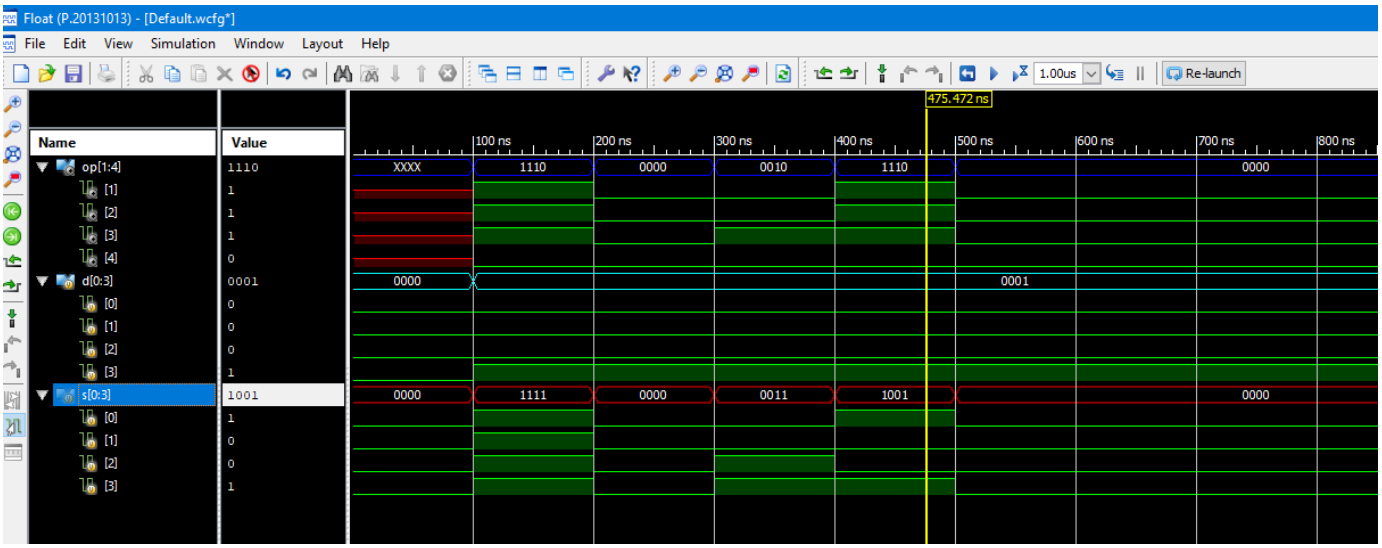
```
module lfsr(d,s,op);
output [1:4]op ;
input [0:3] d;
input[0:3] s;

wire x,y;
assign op[2]=op[4]^x;
assign op[3]=op[4]^y;

m21 cell_1(d[0],op[4],s[0],op[1]);
m21 cell_2(d[1],op[1],s[1],x);
m21 cell_3(d[2],op[2],s[2],y);
m21 cell_4(d[3],op[3],s[3],op[4]);
endmodule

module  m21(d0, d1, s, op);
output op;
input d0, d1, s;
wire d;
assign d = ~d0;   //EXTERNAL INPUT
assign op =(s)?d:d1;
endmodule
```

Test Bench:

```
module tb_lfsr_project_grp1;

// Inputs
reg [0:3] d;
reg [0:3] s;

// Outputs
wire [1:4] op;
```

```
// Instantiate the Unit Under Test (UUT)
lfsr uut (
.d(d),
.s(s),
.op(op)
);

initial begin
// Initialize Inputs
d = 0;
s = 0;

#100; d=4'b0001; s=4'b1111;
#100; d=4'b0001; s=4'b0000;
#100; d=4'b0001; s=4'b0011;
#100; d=4'b0001; s=4'b1001;
#100; d=4'b0001; s=4'b0000;

end

endmodule
```

Simulation Result:



Conclusion: We have successfully implemented the circuit using Verilog HDL

- According to the select line either the external input or the feedback input is applied.
- Functionality of the circuit is similar to the internal EOR type LFSR.