ISSN: 2350-0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 3, Issue 2 , February 2016

Design and Analysis of a 128 Bit Linear Feedback Shift Register Using VHDL

B.Dharma Teja, V.Swetha, D.Lokesh,lokesh. K.V.R.L.Prasad

Sri Vaishnavi College of Engineering, India
Sri Sivani College of Engineering, India
Sri Vaishnavi College of Engineering, India
Sri Vaishnavi College of Engineering, India

ABSTRACT: This paper proposes a 128 Bit Linear Feedback Shift Register which generates pseudo-random test patterns as the input bit is a linear function of its previous state. The total number of random state generated on LFSR depends on the feedback polynomial. As it is simple counter so it can count maximum of 2n -1 by using maximum feedback polynomial. Here in this paper we implemented 128-bit LFSR on FPGA by using VHDL to study the performance and analysis the behaviour of randomness. The analysis is conceded out to find number of gates, memory and speed requirement in FPGA as the number of bits is increased. The design is simulated and synthesized in Xilinx 14.5 ISE.

KEYWORDS: LFSR, FPGA, VHDL.

I.INTRODUCTION

A.Overview

The main challenging areas in VLSI are performance, cost, testing, area, reliability and power. The demand for portable computing devices and communications system are increasing rapidly. These applications require low power dissipation for VLSI circuits. The power dissipation during the test mode is 200% more than in normal mode. Hence it is important aspect to optimize power during testing. Power optimization is one of the main challenges. Linear feedback shift registers have multiple uses in digital systems design. Here we have implemented a 128 bit length sequence on FPGA using VHDL with maximum length feedback polynomial to understand the memory utilization and speed requirement. Also, we have presented the comparison of performance analysis based on synthesis and simulation result as well identify the simulation problem for long bit LFSR. The target device we have used Xilinx Spartan 3E and performed simulation and synthesis using Xilinx ISE. The HDLs are VHDL and Verilog. We prefer VHDL for programming because it is widely used.

B.Application

For generating data encryption keys, random numbers are very much useful in the various applications such as communication channel, bank security, etc. it is used to design encoder and decoder for sending and receiving data in noisy communication channel. They have also been used aesthetically, for example in literature and music, and are of course ever popular for games and gambling. Applications of LFSRs also include generating pseudo-random numbers, pseudo-noise sequences, fast digital counters, and whitening sequences.

II. LINEAR FEEDBACK SHIFT REGISTER

Linear-feedback shift register (LFSR) is a shift register whose input bit is a linear function of its previous state. The most commonly used linear function of single bits is exclusive-or (XOR). Thus, an LFSR is most often a shift register whose input bit is driven by the XOR of some bits of the overall shift register value. An LFSR is a class of devices known as state machine. It is a shift register whose input bit is a linear function of its previous state. The only linear functions of single bits are XOR and XNOR. Thus it is a shift register whose input bit is driven by XOR or XNOR of some bits of overall shift register value.

A. Theory of Operation

Feedback around an LFSR's shift register comes from a selection of points (taps) in the register chain and constitutes XORing these taps to provide tap(s) back into the register. Register bits that do not need an input tap, operate as a standard shift register. It is this feedback that causes the register to loop through repetitive sequences of pseudo-random value. The choice of taps determines how many values there are in a given sequence before the sequence repeats. The implemented LFSR uses a one-to-many structure, rather than a many-to-one structure, since this structure always has the shortest clock-to-clock delay path.

Pseudo random number sequence generator is generated in VHDL according to the following circuit in Figure 1 based on the concept of shift register.

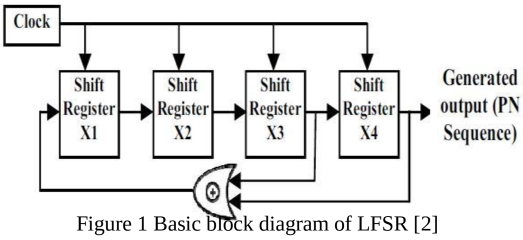


Figure 1 Basic block diagram of LFSR [2]

The bits in the LFSR state which influence the input are called taps. A maximum-length LFSR produces an m-sequence (i.e. it cycles through all possible 2n -1 state within the shift register except the state where all bits are zero), unless it contains all zeros, in which case it will never change. The sequence of numbers generated by this method is random. The period of the sequence is (2n - 1), where n is the number of shift registers used in the design.

B.Types of LFSR

There are two conventional forms of LFSR designs:

a) **Standard LFSR:** Figure 2 shows an n-stage standard LFSR. It consists of n flip-flops and a number of XOR gates. Since XOR gates are placed on the external feedback path, the standard LFSR is also referred to as an external-XOR LFSR.

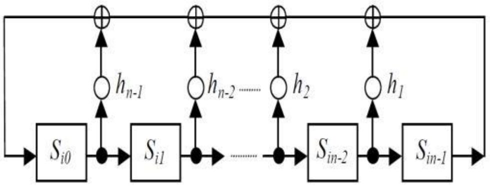


Figure 2 An *n*-stage (external-XOR) standard LFSR

b) Modular LFSR: Similarly, an *n*-stage modular LFSR with each XOR gate placed between two adjacent flip-flops, as shown in Figure 3, is referred to as an **internal-XOR LFSR.** This circuit runs faster than its corresponding standard LFSR, because each stage introduces at most one XOR-gate delay.

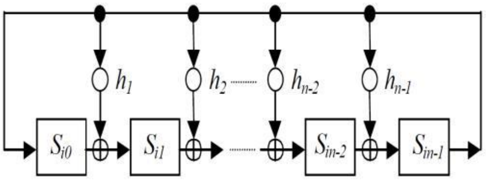


Figure 3 An *n*-stage (internal-XOR) Modular LFSR

Despite of different state trajectories, both structures are capable of generating an m-sequence for each stage output.

C.Types of LFSR depending on data length

a) 8-bit LFSR: 8-bit LFSR with maximum length feedback polynomial $X^8 + X^6 + X^5 + X^4 + 1$ generates $2^8 -1 = 255$ random outputs, which is verified from the simulation waveform. The circuit diagram for 8-bit LFSR with maximum length polynomial is shown in Figure 4.

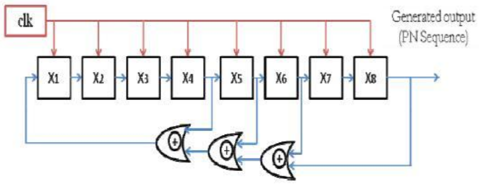


Figure 4 Circuit Diagram of 8- Bit LFSR with maximum length Feedback Polynomial $X^8 + X^6 + X^5 + X^4 + 1$

b) 16-Bit LFSR: 16-bit LFSR with maximum length feedback polynomial $X^{16} + X^{14} + X^{13}$ $X^{11} + 1$ generates $2^{16} -1 = 65535$ random outputs, which is verified from the simulation waveform.

The circuit diagram for 16-bit LFSR with maximum length polynomial is shown in Figure 5

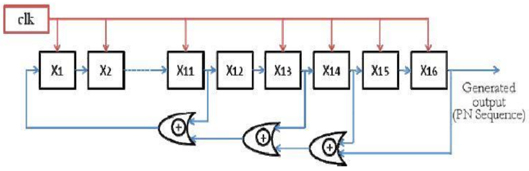


Figure 5 Circuit Diagram of 16- Bit LFSR with maximum length Feedback Polynomial $X^{16} + X^{14} + X^{13} + X^{11} + 1$

c) 32-bit LFSR: 32-bit LFSR with maximum length feedback polynomial $X^{32} + X^{22} + X^2 + X^1 + 1$ for which $2^{32} -1 =$ 429, 49, 67,295 random outputs. The circuit diagram for 32-bit LFSR with maximum length polynomial is shown in Figure 6.

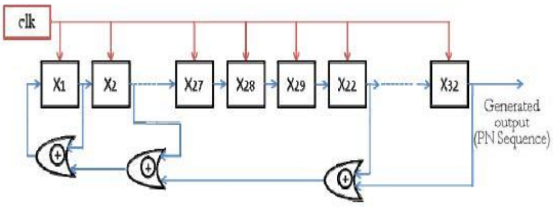


Figure 6 Circuit Diagram of 32- Bit LFSR for maximum length Feedback Polynomial $X^{32} + X^{22} + X^2 + X^1 + 1$

d) 128-bit LFSR: 128-bit LFSR with maximum length feedback polynomial $X^{128} + X^{127} + X^{126} + X^{121} + 1$ for which $2^{128} -1 =$ 429, 49, 67,295 random outputs. The circuit diagram for 128-bit LFSR with maximum length polynomial is shown in Figure 7.

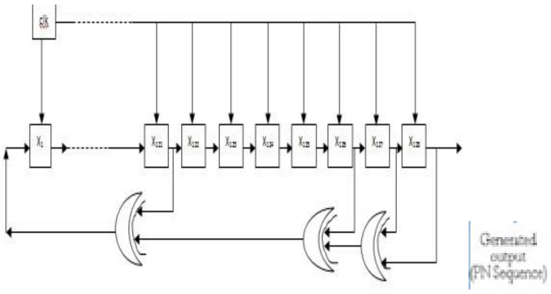
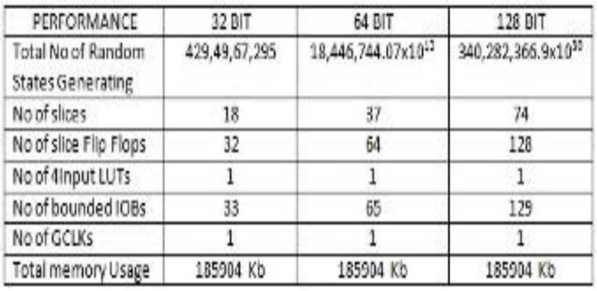


Figure 7 Circuit Diagram of 128- Bit LFSR for maximum length Feedback Polynomial $X^{128} + X^{127} + X^{126} + X^{121} + 1$

D.Comparison between 32,64,128 bit LFSR

The synthesis and simulation for 32,64,128 bit LFSR by using maximum length feedback polynomial are given in Table 1. From the table we can find the total memory usage and simulation time of different length LFSR.



PERFORMANCE	32 BIT	64 BIT	128 BIT
Total No of Random States Generating	429,49,67,295	$18,446,744.07\times10^{12}$	$340,282,366.9\times10^{30}$
No of slices	18	37	74
No of slice Flip Flops	32	64	128
No of 4input LUTs	1	1	1
No of bounded IOBs	33	65	129
No of GCLKs	1	1	1
Total memory Usage	185904 Kb	185904 Kb	185904 Kb

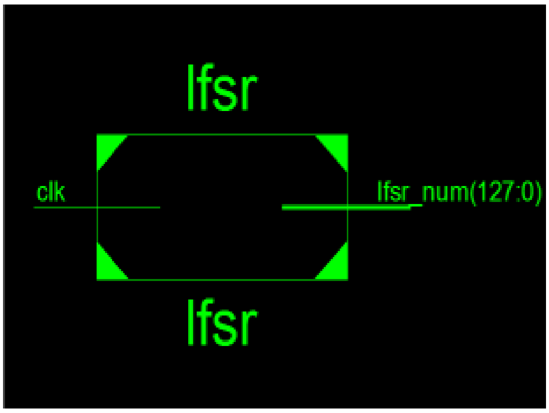
Table 1. Comparison between 32,64 and 128 bit LFSR depending on the performance

The memory utilization is found to be same for all three LFSR.

Optimum Tap Points

The choice of which taps to use determines how many values are included in a sequence of pseudo-random values before the sequence is repeated. Certain tap settings yield the maximal length sequences of (2^N-1).

III. SIMULATION RESULTS



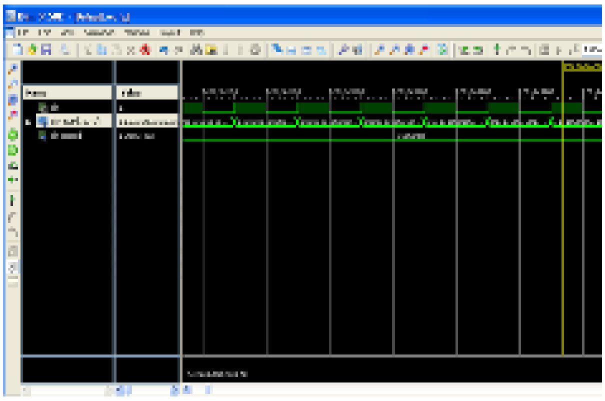
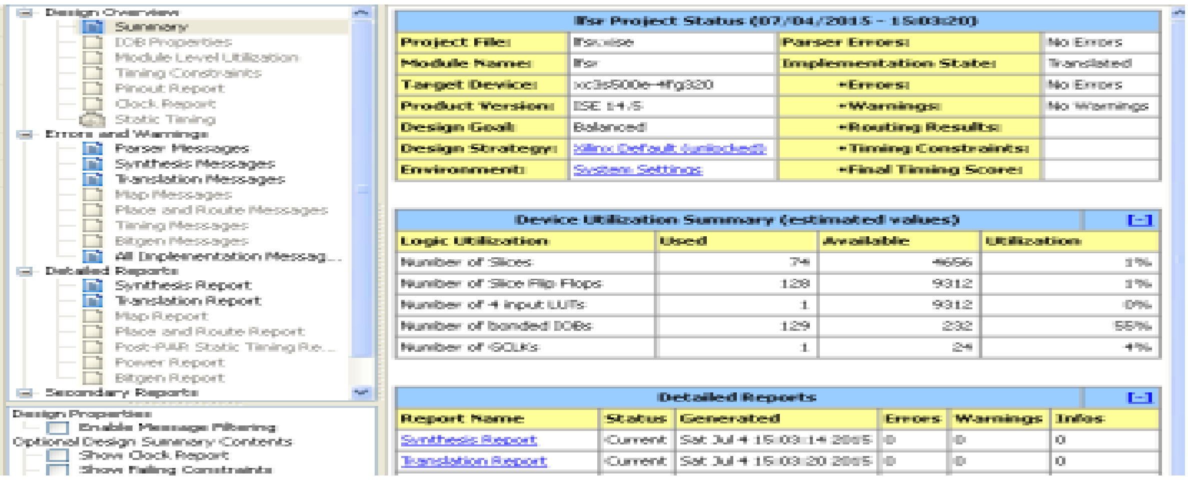


Figure 7 RTL Schematic of the System Design

Figure 8 Simulation Results of Test Generator



lfsr Project Status (07/04/2015 - 15:03:20)			
Project File:	lfsr.xise	Parser Errors:	No Errors
Module Name:	lfsr	Implementation State:	Translated
Target Device:	xc3s500e-4fg320	•Errors:	No Errors
Product Version:	ISE 14.5	•Warnings:	No Warnings
Design Goal:	Balanced	•Routing Results:	
Design Strategy:	Xilinx Default (unlocked)	•Timing Constraints:	
Environment:	System Settings	•Final Timing Score:	

Device Utilization Summary (estimated values) [-]

Logic Utilization	Used	Available	Utilization
Number of Slices	74	4656	1%
Number of Slice Flip Flops	128	9312	1%
Number of 4 input LUTs	1	9312	0%
Number of bonded IOBs	129	232	55%
Number of GCLKs	1	24	4%

Detailed Reports [-]

Report Name	Status	Generated	Errors	Warnings	Infos
Synthesis Report	Current	Sat Jul 4 15:03:14 2015	0	0	0
Translation Report	Current	Sat Jul 4 15:03:20 2015	0	0	0

Figure 9 Synthesis Results

Device Utilization

Number of Slices	:	74 out of 4656
Number of Slice Flip Flops	:	128 out of 9312
Number of 4 input LUTs	:	1 out of 9312
Number of bounded IOBs	:	129 out of 232
Number of GCLKs	:	1 out of 24

IV. CONCLUSION

Design of a random testing circuit based on LFSR for the external memory interface is discussed in this paper. The random test patterns can improve testing efficiency, and reduce the artificial dependence in testing process in any circuit. Definitely 128 bit LFSR with maximum length feedback polynomial will generate large sequence which is more secure than other but because of simulation difficulties modification in long bit LFSR is needed. In the practical use 8-bit and 16-bit LFSR is sufficient for different cryptographic applications. Here we designed a 128 bit linear feedback shift register.

REFRENCES

[1] Janick Bergeron. *Writing testbenches functional verification of HDL Models(2nd Edition).*Springer- Verlag-2003.

[2] Amit Kumar Panda et.al, "*FPGA Implementation of 8, 16 and 32 Bit LFSR with Maximum Length Feedback Polynomial using VHDL*," International Conference on Communication Systems and Network Technologies, 2012.

[3] Madhusudan Dey, Abhishek Singh, "*Design and IP core based implementation of a programmable 8-bits random sequence generator,* "In Proceedings of the International Symposium on Nuclear Physics, 2009, pp.678-679.

[4] Mohammed Gazi.J et.al, "*Design of Random Testing Circuit Based on LFSR for the External Memory Interface*," International Journal of Application or Innovation in Engineering & Management (IJAIEM), vol. 2, issue 3, , pp.145-150, March 20

[5] P.Bhanuchander et.al, "*BIST architecture Implementation Based on Advanced LFSR for testing EMIFs for SRAM*," International Journal of Industrial Electrical, Electronics, Control and Robotics (IISRC), vol. 3, issue 5, pp. 1-8 August 2013.

[6] Panda Amit K, Rajput P, Shukla B, "*Design of Multi Bit LFSR PNRG and Performance comparison on FPGA using VHDL*",International Journal of Advances in Engineering & Technology(IJAET), vol.3, issue 1, pp. 566-571, March 2013.