Design and Implementation of LFSR in Cryptography

Nagaraju. N[1] Uma Maheswari.S[2] Sruthi.S[3] Surya.V[4] Umasree.N[5]
[1]Assistant Professor [2,3,4,5]UG Scholars
[1,2,3,4,5]Department of Electronics and Telecommunication Engineering
[1,2,3,4,5]Adhiyamaan College of Engineering, Hosur

Abstract— Cryptography plays a predominant role in network security. LFSR (Linear Feedback Carry Shift Register) are widely used in order to generate pseudo random sequence, the linear function of its previous state is given as input bit. The total number of random states produced by LFSR depends on feedback polynomial. The output taken from few shift register with Boolean function, form the feedback path which could generate (2^n) - 1 random sequence. In this paper we have designed Galois LFSR and Fibonacci LFSR for generating pseudo random sequence using VHDL to analyze its performance and behavior of its randomness.

Keywords: LFSR, pseudo random sequence, VHDL

I. INTRODUCTION

Random numbers have broad application in many fields like simulation, cryptography, game playing, appraisement of multiple integrals, statistical sampling, particle transport calculations etc. LFSR is a good pseudo random number sequence generator for their use in digital broadcasting system and communication system. They are also used in stream cipher as their construction is simple in electromechanical,electronic circuits, long periods, and could produce a uniform distributed output streams. The output sequence obtained is based on specific mathematical algorithm and for large cycle periods the sequence are non-repetitive and exhibit randomness. The maximum length sequence of randomness is obtained depending on the tap position. The arrangements of taps for feedback in an LFSR are given by the expression of infinite field arithmetic as a polynomial mod2. This specifies that the coefficient of the polynomial should be 0's or 1's. This is referred as feedback polynomial or reciprocal characteristic.

The initial input of LFSR is seed [1]. If the number of taps is even then there must be no common divisor to all taps. The primitive polynomials and the output random sequence of Fibonacci LFSR and Galois LFSR configuration shows the performance characteristics due to structure adapted [2]. Stream ciphers do have less security weakness. Encryption is a process of transforming a part of information into unreadable form which is unpredictable. The input to this process is said to be a plain text (clear text) and the output thus obtained is cipher text (cryptogram). Inversion of this process is said to be decryption [3]. we have designed key generator using LFSR which is implemented in symmetric key cryptography using stream cipher structure.

II. PN SEQUENCE AND ITS PROPERTIES

The PN sequences of N length = (2^n) -1 are binary sequences in which a linear recurrence specified by the primitive polynomial of degree n is fulfilled that satisfies The three different properties that satisfies the Pseudo Noise (PN) sequence is a periodic binary sequence they are: Run, Balance, and Autocorrelation property [3], [4], [7].

By means of well-chosen primitive feedback polynomials pseudo random sequence is generated by using LFSR completed by a definite finite field.

These properties are listed below, they are,

1) Run property
2) Balanced property
3) Autocorrelation property.

1) Run Property: The run property is termed as a subsequence of identical symbols within the ML sequence. The length of the subsequence is the run-length. If the number of runs of zeros is equivalent to the number of runs of ones, the total number of runs = (N+1) / 2.
2) Balanced Property: In Maximum length (ML) sequence in every period, the total number of ones is one more than the numbers of zeros.
3) Autocorrelation Property: The autocorrelation function of a ML sequence is binary valued and periodic. And it is given by,

$$r(\mathbf{i})=(\mathbf{1}/\boldsymbol{N})*\Sigma(\boldsymbol{Cn})(\mathbf{Cn}-\mathbf{i}) \quad \text{----- (2.1)}$$

Where, N=length or period of PN sequence
i= lag of the autocorrelation sequence.

III. LINEAR FEEDBACK SHIFT REGISTERS

LFSR is a kind of shift register whose sequences will be ((2^n)-1) states, where n is the number of shift registers used for implementing the LFSR. For each clock pulse the contents of the registers are moved one bit towards right. The feedback from the LFSR registers are taped to the left most register of the LFSR through a XOR or XNOR gate [1], [2].

The LFSR's initial bits is given as "seed". A seed value of all 1's cannot be initialized if the instance used is an XNOR gate as feedback. Similarly, A seed value of all 0's cannot be initialized. If the instance used is an XOR gate as feedback, since the LFSR would get locked-up in these states.

However, to be random and to take a very long cycle [2], [5], [6] an LFSR seems to have a means of well-chosen maximum feedback polynomial which will generate a pseudo random sequence. The LFSR block diagram is shown in Figure 1.

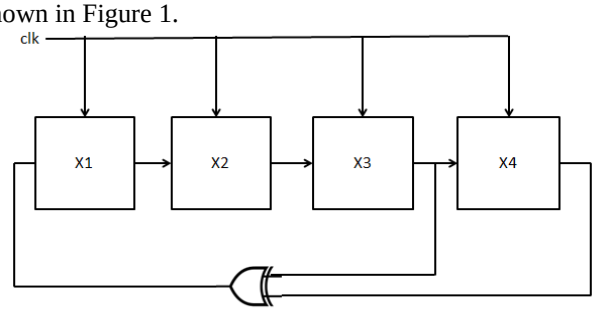

Fig. 1: LFSR Block Diagram

The conditions for choosing feedback polynomial for LFSR are as follows [1], [2]:

1) The first shift register input is represented as 1 in the polynomial and it never signifies a tap position of the gate used.
2) The tap bits are represented by the power of the terms in the maximum feedback polynomial. The first bit in the feedback polynomial and the last bit are permanently connected as a feedback tap respectively.
3) In the feedback polynomial if the number of taps are even , then the LFSR is said to be of maximum length.
4) Taps set should be chosen in such a way that they must relatively be a prime number.

IV. DESIGN OF 16 BIT FIBONACCI LFSR

The 16 bit Fibonacci LFSR whose maximum feedback polynomial is represented as X^16+ X^15+ X^14 + X^12 + 1 will produce (2^16)-1 = 65535 PNsequence [1], [2].

The block diagram of 8 bit Fibonacci LFSR is shown in Figure 2. The tap positions where the xor's to be placed is determined by the feedback polynomial used. And the corresponding bits are serially cascaded and fed back.

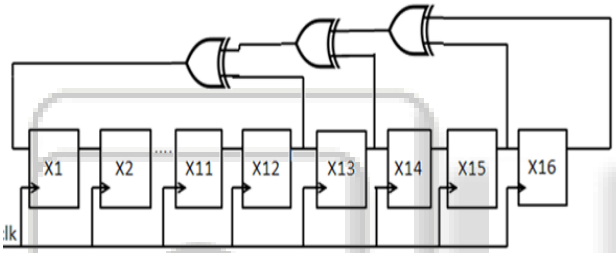

Fig. 2: Block Diagram of 16 bit Fibonacci LFSR

V. DESIGN OF 16 BIT GALOIS LFSR

Galois LFSR do not run serially thereby, it could not concatenate every tap to produce the new input (the XORing is done within the LFSR structure), therefore the propagation time is reduced to that of one XOR rather than a whole structure), thus the execution speed is increased and parallel computation is possible for each tap.The 8 bit Galois LFSR whose maximum feedback polynomial is represented as X^16+ X^15+ X^14 + X^12 + 1 will produce(2^16)-1 = 65535, PN sequence [1], [2]. The block diagram of 16 bit Galois LFSR is shown in Figure 3.

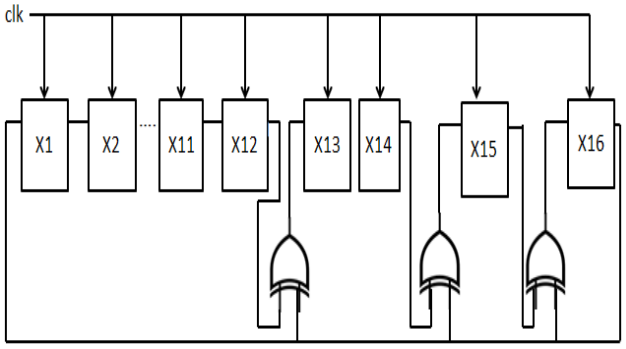

Fig. 3: Block Diagram of 16 bit Galois LFSR

VI. DESIGN OF STREAM CIPHER STRUCTURE

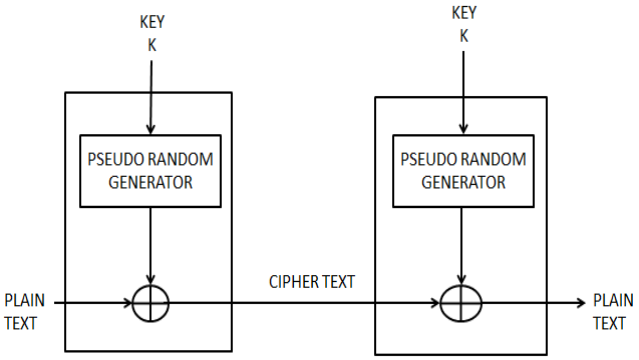

Fig. 4: Stream Cipher Structure

Symmetric key cryptography is grouped into stream cipher based on number of bits encrypted or decrypted. Encrypting a bit of plain text at a time is said to be stream cipher. The original input is called plain text. The encrypted text is called cipher or intermediate text. After decryption the original input is retrieved.

A. Block Diagram of Key Stream Cipher Structure:

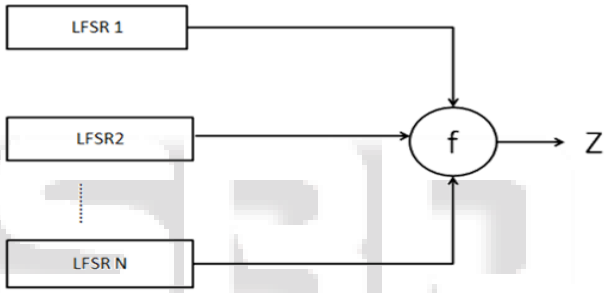

Fig. 5: Block Diagram of Key Stream Cipher Structure

Where, f denotes the Boolean function; Z denotes final key obtained from stream cipher.

VII. SIMULATION RESULTS

SIMULATION OUTPUT FOR 8-BIT FIBONACCI LFSR

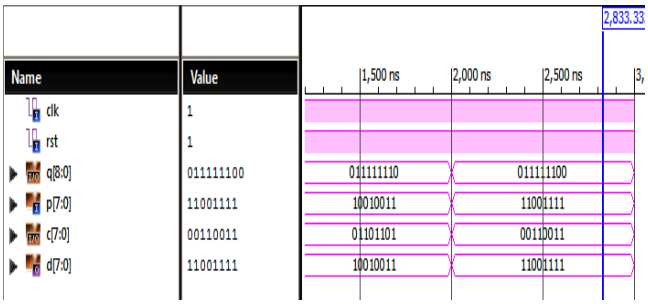

Fig. 6: Simulation output for 8-bit Fibonacci LFSR

Where, q-key stream; p-plain text; c-cipher text; d-decrypted text.

A. Simulation Output For 8-Bit Galois LFSR:

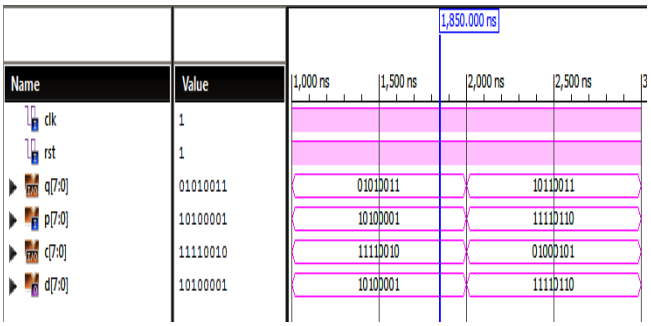

Fig. 7: Simulation output for 8-bit Galois LFSR

B. Analysis Of 8-Bit Fibonacci LFSR And Galois LFSR:

Components	Cryptography Of 8-Bit Fibonacci LFSR	Cryptography Of 8-Bit Galois LFSR
Number Of Slice Flip Flops	8	5
Number Of Slices Occupied	15	10
Number Of 4 Input Luts	9	8
Delay	9.033ns	9.033ns

Table 1:Analysis Of 8-Bit Fibonacci LFSR And Galois LFSR

Hence, from the above results we could analyze that design of Galois LFSR is efficient in terms of area parameters to that of Fibonacci LFSR.

C. Simulation Output for Stream Cipher Using 8-Bit Fibonacci LFSR:

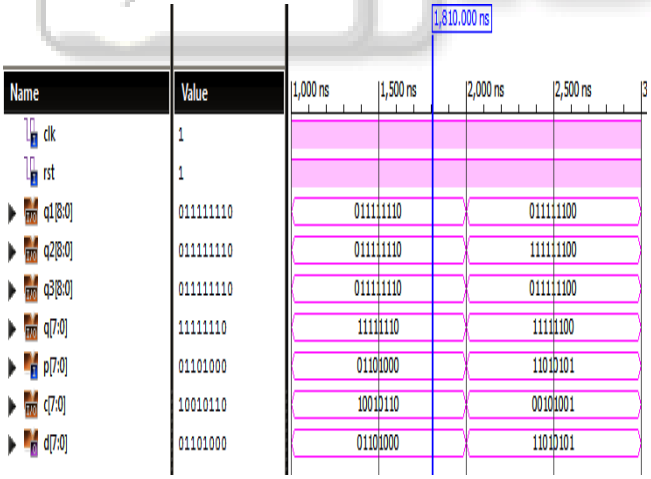

Fig. 8: Simulation output for stream cipher using 8-bit Fibonacci LFSR

D. Simulation Output for Stream Cipher Using 8-Bit Galois LFSR:

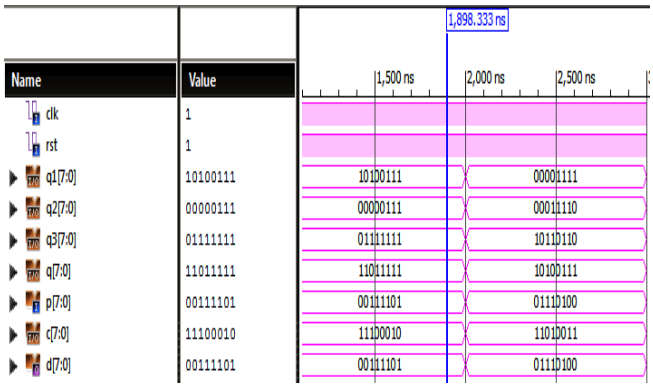

Fig. 9: Simulation output for stream cipher using 8-bit Galois LFSR

E. Analysis of Stream Cipher Using 8-Bit Fibonacci LFSR And Galois LFSR:

Components	Stream Cipher Using 8-Bit Fibonacci LFSR	Stream Cipher Using 8-Bit Galois LFSR
Number Of Slice Flip Flops	24	10
Number Of Slices Occupied	32	17
Number Of Slice Luts	19	12
Delay	7.824ns	7.824ns

Table 2: Analysis of Stream Cipher using 8-Bit Fibonacci LFSR and Galois LFSR.

Hence, from the above results we could analyse that design of Galois LFSR is efficient in terms of area parameters to that of Fibonacci LFSR.

F. Simulation Output for Stream Cipher Using 16-Bit Fibonacci LFSR:

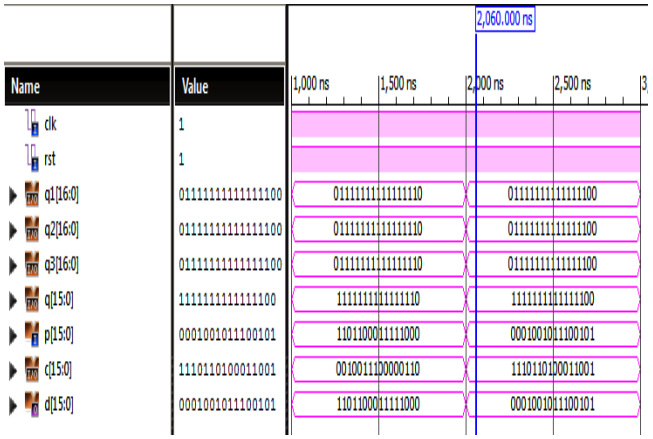

Fig. 10: Simulation Result for Stream Cipher using 16-bit Fibonacci LFSR

G. Simulation Output for Stream Cipher Using 16-Bit Galois LFSR:

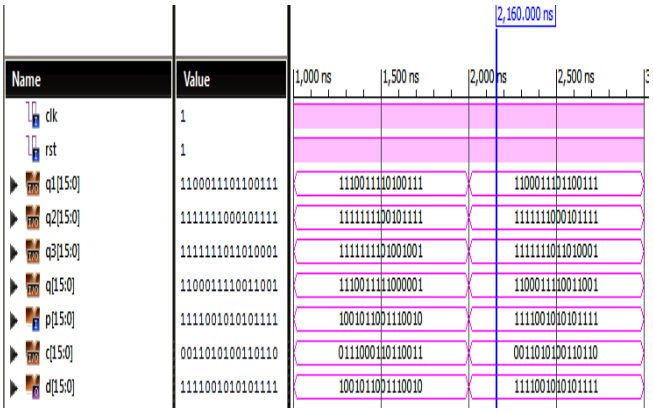

Fig. 11: Simulation result of stream cipher using 16-bit Galois LFSR

H. Analysis of Stream Cipher Using 16-Bit Fibonacci LFSR And Galois LFSR:

Components	Stream Cipher Using 16-Bit Fibonacci LFSR	Stream Cipher Using 16-Bit Galois LFSR
Number Of Slice Flip Flops	48	18
Number Of Slices Occupied	67	34
Number Of Slice Luts	35	23
Delay	3.774ns	3.774ns

Table 3: Analysis of Stream Cipher Using 16-Bit Fibonacci LFSR and Galois LFSR.

Hence, from the above results we could analyze that design of Galois LFSR is efficient by means of area to that of Fibonacci LFSR.

VIII. Conclusion

The role of cryptography has increased in data privacy and authentication in critical applications for providing security in today's world. An efficient Cryptographic system is designed using Fibonacci LFSR and Galois LFSR. In which Galois LFSR seems to be more efficient when compared to Fibonacci LFSR by means of area and power consumption. Hence, this system could be used in the real time application for providing security. This cryptographic system could be made even more efficient by adopting different cryptographic algorithms for meeting high security needs. complexity can be increased by using additional techniques such as by using one of the keys as a public key and the other can be used as private key.

References

[1] Amit Kumar Panda, Praveena Rajput, Bhawna Shukla, "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.

[2] Purushottam Y. Chawke, R.V.Kshirsagar, "Design of 8 and16 bit LFSR with maximum length Feedback polynomial using Verilog HDL", 13th IRF International Conference, ISBN: 978-93-84209-37-7, 20th July-2014, Pune, India.

[3] R.N. Mutagi, "Pseudo noise sequences for engineers", Electronics & Communication Engineering Journal April 1996.

[4] Simon Haykin, Digital communication, JOHN WILEY & SONS INC, second edition, 2009.

[5] Afaq Ahmad, Sayyid Samir Al-Busaidi and MufeedJuma Al-Musharafi, "Properties of PN Sequences Generated by LFSR – a Generalized Study and Simulation Modelling", Indian Journal of Science and Technology, October 2013. Kawal .K, Saluja, "linear feedback shift register theory and application", department of electrical and computer engineering, University of Wisconsin-Madison, revised October 1988 updated 1991.

[6] Musher Ahmad and Omar Farooq, "Chaos Based PN Sequence Generator for Cryptographic Applications", International Conference on Multimedia, Signal Processing and Communication Technologies, 978-1-4577-110-7/ 2011.

