

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/312017072>

A Pseudorandom Binary Generator Based on Chaotic Linear Feedback Shift Register

Article in *Iraqi Journal for Electrical And Electronic Engineering* · December 2016

DOI: 10.33762/eej.2016.118290

CITATION

1

READS

502

1 author:



Saad Muhi Falih

University of Basrah

9 PUBLICATIONS 15 CITATIONS

SEE PROFILE

A Pseudorandom Binary Generator Based on Chaotic Linear Feedback Shift Register

Saad Muhi Falih

Department of Computer Technical Engineering
Islamic University College
Al Najaf al Ashraf, Iraq
saadmuheyfalh@gmail.com

Abstract: This paper presents a simple method for the design of Chaotic Linear Feedback Shift Register (CLFSR) system. The proposed method is based on a combination of two known systems. The first is called Linear Feedback Shift Register (LFSR) system, and the other is called Chaotic Map system. The main principle of the proposed system is that, the output of the LFSR is modified by exclusive-or (XOR) it with the stream bit that is generated by using the chaotic map system to eliminate the linearity and the repeating in the output of the LFSR system. The proposed system is built under Matlab environment and the quality of sequence generation tested by using standard tests which shows that the proposed system is a good random number generator that overcome the linearity and repeating disadvantages.

Keywords: LFSR, Chaotic Map, PRNG, Chaotic Binary Sequence Generator.

I. INTRODUCTION

The information security is one of the famous concepts of the most modern communication systems and computer networks. Therefore, the studies of encryption systems are one of the most important fields of scientific research in the last two decades. One of the major important aims of these scientific researches is to design high quality PseudoRandom Number Generation (PRNG), which is a basic subject of any encryption system [1].

The linear feedback shift registers (LFSRs) gives an economic, fast, and efficient method for generating a wide variety of pseudorandom number sequences [2]. In spite of the advantages in hardware complexity, this architecture has significant drawback that each bit in a LFSR's sequence is linearly related to its initial state which causes a big possibility of hacking [3].

There are three proposed methods to eliminate this linearity in the literature. In the first one, the linearity is eliminated by using several LFSRs, and the key stream is generated by a suitable nonlinear Boolean function. The second method is based on using single LFSR only, and the key stream is generated from nonlinear Boolean

function of the different stages of LFSR. Finally, the last method based on the use of irregularly clocked of the LFSR to eliminate the linearity. The main idea behind a clock-controlled generator is to introduce nonlinearity into LFSR-based key stream generators by means of having the output of one LFSR control on the clocking of a second LFSR [4].

On the other hand, the Chaotic Binary Sequence Generator (CBSG) based on the chaotic map is another method to generate pseudorandom number signals. The random-like, high non-linearity and unpredictable dynamics of chaotic systems, their inherent determinism and simplicity of realization suggests their potential for exploitation as PRNGs [1]. However, the main disadvantage of the CBSG is that the initial condition can be estimated based on a binary sequence [5].

In this paper, a new method proposed to eliminate the linearity in LFSR as well as to hide the statistical characteristic pattern of the CBSG that is used to estimate the initial condition of the chaotic map (see [6] for more details). The new-proposed method is based on mixing the two previous methods. The output of the LFSR will be exclusive-or with the output of the CBSG. The result sequence is used as a key stream. The proposed method herein has all the advantages of

the LFSR and chaotic binary sequence, and it disposes all of their drawbacks.

The rest of the paper is organized as follows: Section 2 looks at the linear-feedback shift register; the chaotic map is discussed in section 3. Section 4 reports the proposed design procedure; the simulation results and discussions are reported in Section 5; finally, the conclusion is summarized in section 6.

II. THE LINEAR-FEEDBACK SHIFT REGISTER

The Linear-Feedback Shift Register (LFSR) is a shift register whose input bit is determined as a linear function of its previous state as shown in Fig.1, the exclusive-or (XOR) is used here as a linear feedback function for single bits [7]. The simplicity in the design and implementation, long period, and good statistical properties are the main advantages of this type of pseudorandom binary source [8].

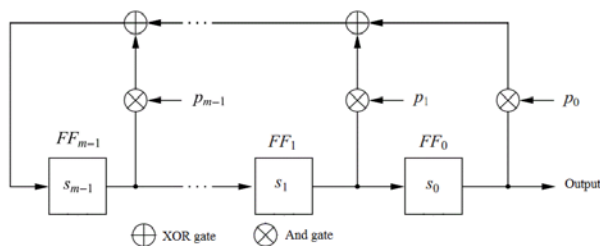


Fig. 1 m-bit LFSR with feedback coefficients p_i and initial values s_{m-1}, \dots, s_0 [9].

The LFSR is used in many applications, such as white noise generation, error detection and correction codes, hiding algorithms, compression algorithms, communication systems, and cryptography systems [10].

The operation of the LFSR is completely deterministic; therefore, its output stream is completely depending on the LFSR initial state that is called the seed [7].

However, because the fact that any register has a finite number of possible states, therefore, the LFSR should be in the end repeats its output cycle. The maximum length of this repeated cycle is equal to $(2^L - 1)$ where L is the length of the LFSR, and this occurs when the feedback function is primitive polynomial [3 and 11]. In general, determining the primitive polynomials for L -bit LFSR is not a simple task. However, Ahmad and Elabdalla study the maximum cycle

of LFSR in [12] and several special cases is reported in [13] that will be used in this work.

III. CHAOTIC MAP

The chaotic map is a simple nonlinear model, but it has a complicated dynamic behavior. The chaotic sequence produced by the chaotic map is extremely sensitive to the change of its initial value. Any chaotic map can be defined as [14]:

$$x_n = f(x_{n-1}) \quad , \quad n = 1, 2, \dots \quad (1)$$

Where x_n is the value of variable x in step number n , and for simplicity purpose it is called the state. The function $f(x_{n-1})$ is mapped the state x_{n-1} to the next state x_n .

In this work, two chaotic maps are chosen to be discussed, the first one is called logistic map, [15] and the other is called quadratic map as described in equations (2) and (3), respectively:

$$x_n = r \cdot x_{n-1} (1 - x_{n-1}) \quad (2)$$

$$x_n = 1 - r(x_{n-1})^2 \quad (3)$$

Where r is a bifurcation parameter lies in the interval $[0, 4]$ and $x_n \in [0, 1]$ for logistic map, on the other hand, $r \in [-0.25, 2]$ and $x_n \in [-1, 1]$ for quadratic map.

The behavior of these systems shows a great dependency on the value of the bifurcation parameter (r). However, this dependency can be revealed by studying the bifurcation diagram, which is a graphical depiction of all values of x visited by iterates the solution of the chaotic map equation with that bifurcation parameter (r) [16].

Figures (1) and (2) show the bifurcation diagram of the two chaotic maps. As it is clear from these figures that the chaotic behavior occurs when the bifurcation parameter (r) lies in the range $[3.68, 4]$ for the logistic map, and lies in the range $[1.43, 2]$ for the quadratic map.

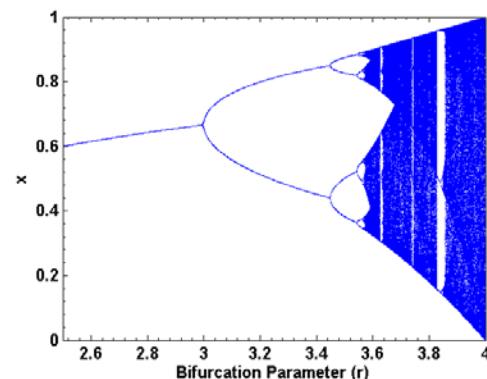


Fig. 2 Bifurcation diagram of the logistic chaotic map.

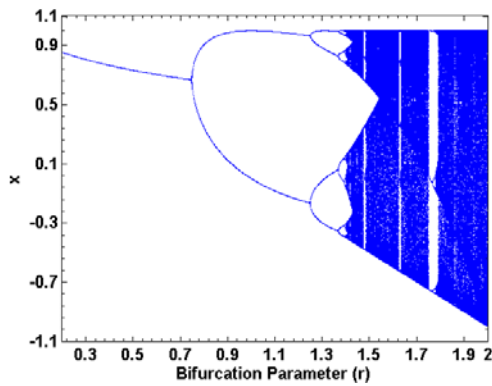


Fig. 3 Bifurcation diagram of the quadratic chaotic map.

IV. PROPOSED CLFSR BINARY SOURCE

The proposed CLFSR pseudorandom binary source is shown in Fig.4. The figure shows the building block diagram for 7-bit LFSR companied with the output of the chaotic binary source.

As shown in Fig. 4, the output of the LFSR is exclusive-or with the output of the CBSG and the result sequence is used as the key stream. This proposed method has all the advantages of LFSR and chaotic binary sequence and it disposes their drawbacks as previously stated.

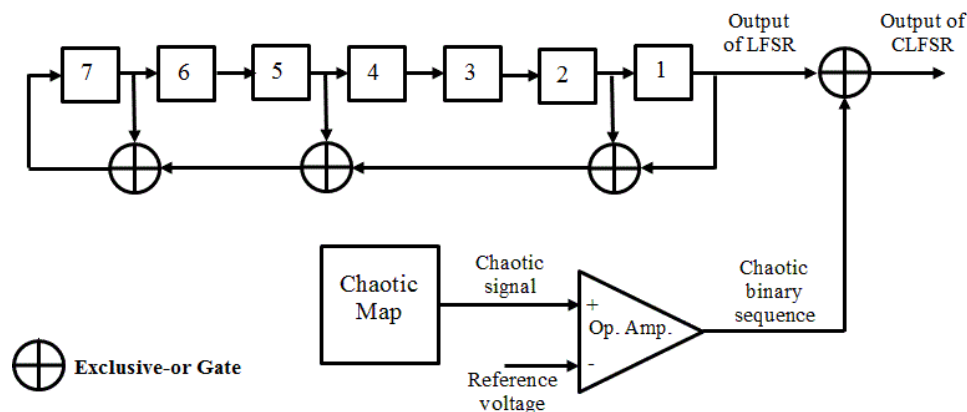


Fig. 4 The proposed chaotic linear-feedback shift registers.

V. SIMULATION RESULTS AND DISCUSSIONS

The described system in section V was constructed under Matlab environment. Also, the system was built based on two types of chaotic map; the first was logistic map, and the other was quadratic map. The parameters that were used in the simulation are reported in Table I.

Table I
Simulation parameters

Parameter name	Value
r for logistic map	4
r for quadratic map	2
Length of LFSR (L)	16

The American National Institute of Standards and Technology (NIST) propose the Federal Information Processing Standard (FIPS) 140-2 tests, which consist of four tests: monobit, poker, runs, and long runs tests. These four tests are used

to verify the randomness of pseudorandom bit sequences by analyzing the distribution of a set of data to see if it is random [17]. Each one of these tests needs a bits stream of 20,000 bits from the sequence under test. Any non-random result in one of these four tests means that the test sequence is not random. However, these tests can be described as following [17]:

1) *The monobit test*: The test sequence is random if the number of one in bits stream generated by tested system lies in the range [9654-10346].

2) *The poker test*:

- The 20,000 bits stream generated by tested system is divided into 5,000 contiguous 4 bit segments.
- The decimal values of each 4 bit segments are determined and stored. However, there are 16 possible value

of each 4 bit segment (*i.e.* : $i \in [0 - 15]$)

- The occurrences of each of the 16 possible 4 bit segment values are counted and stored in $g(i)$, where $0 < i < 15$, as the number of each 4 bit value.
- Finally, the test result is evaluated by the following equation:

$$I = \left(\left(\frac{16}{5000} \right) \left(\sum_{i=0}^{15} g(i)^2 \right) \right) - 5000 \quad (4)$$

If $1.03 < I < 57.4$ then the sequence is random.

- 3) *The runs test*: The run can be defined as the repeated of the same bit in contiguous bits. The tested system can pass this test if the number of runs of length 1, 2, 3, 4, 5, and longer than 5 lies in specified limits described in Table II.

Table II

The runs test interval required under FIPS 140-2[17].

Length of run	Requiren Interval
1	2315-2685
2	1114-1386
3	527-723
4	240-384
5	103-209
Longer than 5	103-209

- 4) *The long run test*: The tested sequence is random if there are no run of length equal to or greater than 34 bits.

The results of all tests are reported in Table III. It can be noticed that the chaotic map systems and the CLFSR systems passes these tests and the sequences produced from them are random sequences. However, as shown from the previous tests any one of the chaotic map can be used to eliminate the linearity and the repeated in the output of LFSR to get a good pseudorandom generator. On the other hand the LFSR system cannot pass these tests. The shadow cells in the test table refer to the reject test results. Therefore, the sequence generated by it cannot be classified as a random sequence.

The simple visual test is another test done by the graphically representation of the data. In this

test, the $(320 \times 320 \times 8 = 819,200)$ bits from the test sequences are divided to the $(320 \times 320 = 102,400)$ consecutive 8 bits. Each 8 bits is read as a single unsigned integer and the resulted value (0-255) is plotted as a pixel brightness (0 being black) on a grayscale image from 320×320 images. The patterns on the image will be shown if there is a presence for the periodic or non-randomness in the sequence.

The simple visual test is excellent in detecting large-scale periodic or patterns in the data generated by a system. Table IV shows the image built from the sequences generated from studied systems. The results clearly show that the CLFSR is a random sequence and the pattern shown in LFSR system is referred to the periodic natural in the sequence, which is not appeared in the proposed system.

VI. CONCLUSION

In this paper, a pseudorandom number generator is proposed based on CLFSR system. The proposed system was built by using Matlab program and the quality of its output sequence was tested by comparing its performance with the performances of the LFSR and the chaotic map systems.

Two types of tests are used to verify the performance of the system, the first one, named FIPS 140-2, which consist from four tests used to verify the randomness of bit sequences. The other one is the visual test, which is used to detect the large-scale periodic or patterns in the bit sequences generated by the system. The results clearly show that the performance of the proposed system is better than that of LFSR in eliminating the linearity and the repeating in the output stream bit. However, the simulation results show that any one of the chaotic map can be used here.

REFERENCES

- [1] J. M. Bahi et. al., "Evaluating Quality of Chaotic Pseudo-Random Generators: Application to Information Hiding," International Journal on Advances in Security, Vol. 4, No. 1 , 2011, pp.118-130.
- [2] R. Z. Khalaf and A. A. Abdullah, "Generate Quantum Key by Using Quantum Shift Register," International Journal of Computer

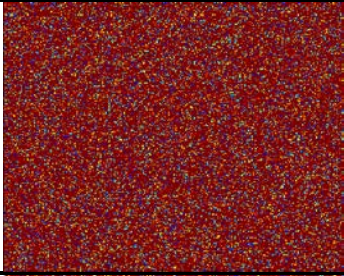
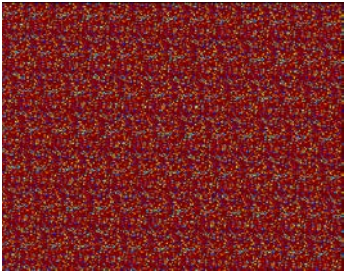
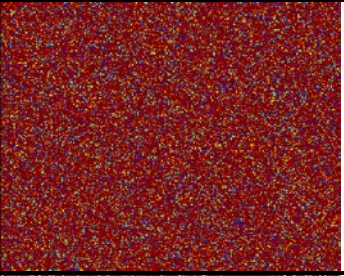
- Networks and Communications Security, Vol. 3, No. 6, June 2015, pp.248-252.
- [3] P. P. Deepthi and P. S. Sathidevi, "Hardware Stream Cipher Based on LFSR and Modular Division Circuit," International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering, Vol. 2, No. 10, 2008, pp. 2251-2259.
- [4] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, "Handbook of Applied Cryptography", CRC Press, Boca Raton, Florida, USA, 1997.
- [5] A. Vlad et. al. "Generating Chaotic Secure Sequences Using Tent Map and a Running-Key Approach," Volume 14, Special Issue 2013, pp. 295–302.
- [6] D. Arroyo, G. Alvarez, J. M. Amigo and S. Li, "Cryptanalysis of a family of self-synchronizing chaotic stream ciphers," Communications in Nonlinear Science and Numerical Simulation, Vol. 16, No.2, 2011, pp. 805–813.
- [7] M. Babaei and M. Ramyar, "Improved Performance of LFSR's System with Discrete Chaotic Iterations," World Applied Sciences Journal, Vol. 13, No. 7, 2011, pp. 1720-1725.
- [8] F. Masoodi, S. Alam, and M. U. Bokhari, "An Analysis of Linear Feedback Shift Registers in Stream Ciphers," International Journal of Computer Applications, Vol. 46, No.17, May 2012, pp.46-49.
- [9] C. Paar, J. Pelzl, 86 B. Preneel, "Understanding Cryptography: A Textbook for Students and Practitioners," Springer, New York, 2010.
- [10] S. Golomb, "Shift Register Sequence," Aegean Park Press, Laguna Hills, CA, 1982.
- [11] A. Ahmad, N. K. Nanda, K. Garg, "Are Primitive Polynomials Always Best in Signature Analysis?," IEEE Design & Test, Vol. 7, Issue. 4, July 1990, pp. 36-38.
- [12] A. Ahmad, A. M. Elabdalla, "An efficient method to determine linear feedback connections in shift registers that generate maximal length pseudo-random up and down binary sequences," Computers & Electrical Engineering, Vol. 23, Issue. 1, Jan. 1997, pp. 33-39.
- [13] P. Alfke, Application Note: "Efficient Shift Registers, LFSR Counters, and Long Pseudo-Random Sequence Generators," Technical report, Xilinx Inc., San Jose, CA, App. note XApp052, 1996.
- [14] S. Azou, G. Burel, and C. Pistre, "A Chaotic Direct-Sequence Spread-Spectrum System for Underwater Communication," IEEE-Oceans'2002, Biloxi, Mississippi, October 29-31, 2002.
- [15] S. E. Borujeni and M. S. Ehsani, "Modified Logistic Maps for Cryptographic Application," Applied Mathematics, Vol. 6, 2015, pp. 773-782.
- [16] S. S. Pratt, "Bifurcations Are Not Always Exclusive," International Journal of Complexity and Education, Vol. 5, No. 1, 2008, pp. 125-128.
- [17] L. Min, T. Chen, and H. Zang, "Analysis of FIPS 140-2 Test and Chaos-Based Pseudorandom Number Generator," Proce. of the 5th Chaotic Modeling and Simulation Intern. Conference, Athens Greece, June 2012, pp. 345-352.

Table III
Randomness Test of the Different Systems Studies

System Under Test	Monobit Test	Poker Test	Runs Test						Long Runs Test
			L=1	L=2	L=3	L=4	L=5	L=6	
Acceptance* Range	9,725-10,275	2.16-46.17	2,315-2,685	1,114-1,386	527-723	240-384	103-209	103-209	0
Logistic map ⁽¹⁾ system	10074	10.992	2490	1223	629	328	171	158	0
Quadratic map ⁽²⁾ system	10168	12.9632	2453	1268	585	314	171	150	0
LFSR system	9999	-0.9312	2499	1245	629	313	156	160	0
CLFSR based on (1)	9995	15.824	2630	1205	611	310	163	153	0
CLFSR based on (2)	10029	12.0672	2420	1276	654	294	179	162	0

* The test results must be located in the range of acceptance to ensure that the sequence under test is random.

Table IV
The Simple Visual Analysis of the Different Systems Studies

	Chaotic Map system	LFSR system	CLFSR system
Logistic map			
Quadratic map	