ISSN: 2395-1680 (ONLINE)
DOI: 10.21917/ijme.2021.0195

ELECTRONIC CIRCUIT REALISATION OF A CHAOTIC PSEUDO RANDOM BIT GENERATOR

H. Soumya Babu[1] and K. Gopakumar[2]

[1] *Department of Electronics and Communication Engineering, LBS Centre for Science and Technology, India*
[2]*Department of Electronics and Communication Engineering, TKM College of Engineering, India*

Abstract

Most of the properties of chaotic systems like sensitivity to initial conditions, deterministic dynamics, mixing property, structural complexity can be considered as analogous to diffusion and confusion with small changes in plain text or the secret key, deterministic pseudo randomness and complex properties of cryptographic systems. The interesting relationship between cryptography and chaos leads to new, highly secure cryptographic techniques. The development of chaotic stream ciphers in cryptography requires the need for the generation of pseudo random bits and hence the need for pseudo random bit generators (PRBG). In this paper, circuit realisation of a pseudo random bit generator is presented, which is based on two chaotic maps, namely the logistic maps, running in parallel and starting from two random independent initial conditions. The circuit is being implemented and simulated for different initial conditions using Multisim software. The results obtained from simulation are further tested for randomness using the NIST suite tests and the detailed results of the statistical testing are also presented in this paper.

Keywords:

Chaos, Logistic Map, PRBG, Multisim, NIST Suite

1. INTRODUCTION

The advances in the field of telecommunications like mobile networks and internet technologies have largely extended the range of information transmission, although it adds new challenges for protecting the data from unauthorized users. This in turn leads to a strong demand for new and secure cryptographic techniques [1] [2]. The interesting relationship between cryptography and chaos hence leads to new secure cryptographic techniques. The main properties of chaotic systems are sensitivity to initial conditions, deterministic dynamics, mixing property, and structural complexity. Most of these properties can be considered analogous to diffusion and confusion with small changes in plain text or the secret key, deterministic pseudo randomness, or diffusion with a very small change within a single block of plaintext. The use of discrete chaotic dynamical systems in the generation of pseudo random bits is been widely studied in the development of chaotic stream ciphers.

The present manuscript deals with the circuit realization of a cryptographically secure random bit sequence generators that further can be used for the construction of fool-proof stream ciphers.[13] The paper also deals with the statistical testing of the randomness of the such generated sequences [13] [15]. The realized circuit consists of two chaotic maps, namely, the logistic maps, running in parallel starting from two independent initial conditions. Since two chaotic maps are used, it increases the complexity of the random number generation, hence making it difficult for an intruder to extract information of the chaotic system.

Some of the efforts undertaken in this area are summarized briefly. Oishi and Inoue [3] put up an idea of designing a pseudo-random number generator in 1982, using a chaotic first order linear difference equation using which, uniform random number generators could be constructed with an arbitrary Kolmogorov entropy. A pseudo random generator was designed by using a second-order digital filter and realized on a digital hardware by Lin and Chua [4]. Gonzalez and Pino [5] presented generalized the logistic map and designed a truly unpredictable random function used for the generation of random numbers. The application of a chaotic piecewise linear1-D map as a random number generator was analysed by Kocarev and Stojanovski [6]. Later, with a theoretical analysis, Li et al. [7] suggested that piecewise linear chaotic maps have perfect cryptographic properties and that bit streams generated through a single chaotic system can be potentially insecure. So, they proposed a new pseudo random bit generator based on a couple of piecewise linear chaotic maps. A pseudo random generator based on the chaotic logistic map was later proposed by Patidar et al. and its testing analysis was also using DIEHARD and NIST suites [8]. A pseudo random bit generator was proposed by Pareek et al. [9], based on two chaotic logistic maps iterated independently starting from independent initial conditions. M. Francois et al. [10], proposed another pseudo random bit generator using two chaotic maps to generate multiple key sequences, with the ability to produce a very large number of pseudo-random sequences. Hamdi et al. [11] proposed a very efficient pseudo random number generator based on chaotic maps and S-box tables.

This paper is organized as follows. Section 2 is a brief introduction about the chaotic logistic map dynamics. Section 3 describes the Proposed Pseudorandom bit Generator (PRBG) (PRBG). Section 4 presents the details of the implementation of the proposed technique. Section 5 presents the performance of the proposed technique and the result analysis. NIST suite of statistical test for randomness is shown in Section 6. Finally, the conclusions are presented in Section 7.

2. CHAOTIC LOGISTIC MAP DYNAMICS

The Logistic map represents a simple mathematical model used for the population growth of a species, exhibiting complex, dynamic behaviour [12]. The present manuscript deals with the circuit realization of a Pseudo Random Bit Generator using such a chaotic iterated map. The Logistic Map forms the fundamental building block of the proposed Pseudo Random Bit Generator. The simplicity of the map makes it a useful bed for new advanced researches in chaos and cryptography.

The logistic map is a dynamical system described by the state Eq.(1).

$$X_{n+1} = f(X_n) = aX_n(1-X_n) \quad (1)$$

where, the state variable $X_n \in [0,1]$ and the control parameter 'a', also called the bifurcating parameter, such that 'a' ϵ [0,4].

The logistic map exhibits extremely complicated dynamics [14] for different values of 'a'. The logistic system always converges to the steady state X_n= 0 for $0 \le a \le 1$ and converges to a different steady state or fixed point for $1 \le a \le 3$. The system begins to oscillate between two states, instead of settling to a fixed point, for a = 3. With the further increase in the value of the bifurcating parameter, in the interval $3 \le a \le 3.57$, a phenomenon referred to as period doubling occurs, where the system undergoes a series of bifurcations that in turn leads to an oscillatory behaviour between four, eight, sixteen states and so on. The interesting part of this map is that, for the interval $3.58 \le a \le 4$, the variable X_n fills out continuous intervals, instead of taking on values from a finite set. Chaos refers to such a non-convergent and non-periodic behaviour of the system and the system exhibiting this behaviour is said to be in the chaotic regime. Since the physical measurement of the initial state is always associated with some amount of error, an interesting characteristic of chaotic dynamics, is; 'extreme sensitivity to initial conditions' ensures that the chaotic orbits cannot be predicted in the long run [8].

3. PROPOSED PSEUDO-RANDOM BIT GENERATOR (PRBG)

A random bit generator or RBG is an algorithm or device that produces a sequence of statistically independent binary digits which in turn requires a naturally occurring source of randomness. It is a difficult task to design a hardware device or develop a software programme to exploit the natural source of randomness to produce bits free from correlation. In such situations, a Pseudo Random Bit Generator (PRBG) replaces a RBG. A PRBG is a deterministic algorithm and uses the so-called 'seed' as the input and produces a binary pseudo random sequence as the output. The seed is a random binary sequence of length k and the output is a random sequence of length $l >> k$, that appears to be random. The main idea behind it is to expand a small truly random sequence of length k into a sequence of greater length l, in a way that, it is not possible to efficiently distinguish between a truly random sequence of length l and output sequence of PRBG [2] [8].

In this paper, an electronic circuit realisation of a PRBG is realized and simulated using Multisim and the degree of randomness is analysed using NIST suite. The realised PRBG is based on two 1-D iterated maps, the logistic maps, starting from random independent initial conditions $X_0, Y_0 \in (0, 1)$ and $X_0 \neq Y_0$. The two logistic maps are defined by

$$X_{n+1} = a_1 X_n (1 - X_n) \tag{2}$$

$$Y_{n+1} = a_2 Y_n (1 - Y_n) \tag{3}$$

By comparing the outputs of both the logistic maps in the following way, the pseudo random bit sequence is generated

$$\begin{aligned} g(X_{n+1}, Y_{n+1}) \quad &= 1 \; if \; X_{n+1} > Y_{n+1} \\ &= 0 \; if \; X_{n+1} < Y_{n+1} \end{aligned} \tag{4}$$

The set of initial conditions ($X_0, Y_0 \in (0, 1)$ and $X_0 \neq Y_0$) serves as the seed for the PRBG. Due to the deterministic nature of the logistic map, if the same seed is supplied to the PRBG, it will produce the same bit sequence. The value of the control parameter 'a', which is selected in the range $3.58 \le a \le 4$, to ensure the system to be in the chaotic regime. But it would be appropriate to choose the value of the control parameter to be close to 4.0, so as to make available a large interval for the seed values X_0 and Y_0, thus increasing the key space of the stream cipher.

The pseudo random bit generator is based on two logistic maps, starting from two random independent initial conditions. A comparator is used to compare the same seed is supplied to the PRBG, it will produce the same bit sequence. The output of the comparator obtained will be a pseudo random sequence of 0's and 1's. The initial conditions mentioned for the two logistic maps define the analog outputs of the logistic maps, since the map is purely deterministic. Since the chaotic maps are highly sensitive to initial conditions, the outputs become highly divergent and unpredictive, which in turn results in an efficient pseudo random bit generator. Since the physical measurement of the initial state will be always associated with some amount of error, it becomes impossible to predict the chaotic orbits in the long run, when the circuit is realised in hardware. Also, since there is bound to be at least a minute natural difference in the initial state, it is enough to make the orbits divergent and uncorrelated so as to result in different orbits even for two identical systems [9].

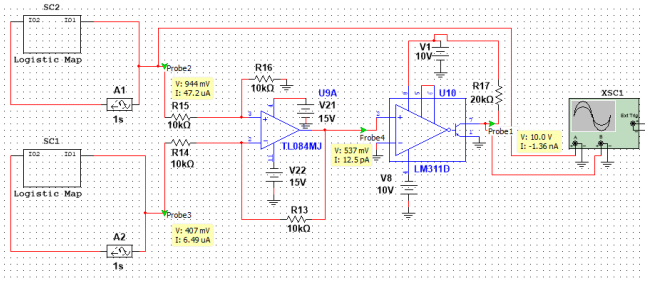

Fig.1. Schematic of the electronic circuit realisation of the proposed PRBG in Multisim

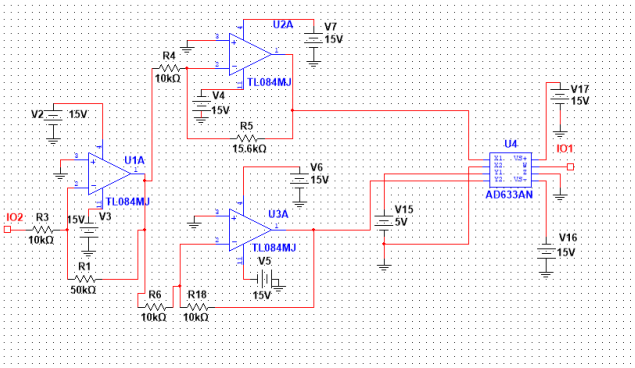

Fig.2. Schematic of the Logistic Map used as the Subcircuit (SC) in the Fig.1

4. CIRCUIT IMPLEMENTATION OF PRBG

The proposed PRBG is realised with electronic components and simulated using Multisim. The circuit consists of two logistic maps in the chaotic regime, starting from two independent initial conditions. The chaotic outputs of the two logistic maps are

given to a subtractor circuit, which inturn is fed to a comparator circuit. The output of the comparator will be either high or low depending upon the output of the subtractor circuit, representing a pseudo random bit sequence that changes chaotically with a slight change in the initial conditions given to the two chaotic maps.

The Fig.1 shows the overall schematic of the electronic circuit realization of the PRBG, simulated for different values of initial conditions for the two logistic maps, using Multisim. The two sub-circuits 'SC1' and 'SC2' shown in the schematic, are identical circuits, that represents the two logistic maps in the chaotic regime, with the bifurcating parameter selected appropriately as '*a*'= 3.9 and fed with two different initial conditions. The value of the bifurcating parameter is so selected to make the logistic maps in the chaotic regime. The circuit is realized using the active and passive components comprising of operational amplifiers, delay elements, resistors, power supply, and a comparator circuit realized using IC LM311.

The Fig.2 shows a schematic of the logistic maps realized using the active and passive components identified as subcircuits SC1 and SC2 in Fig.1, but starting from two independent initial conditions [14]. The basic components of the subcircuits to construct a logistic map are the operational amplifier TL084MJ, the multiplier AD633AN, resistors R_{13}, R_{14}, R_{15}, R_{16} and power supply.

From the Fig.2, the output of the op-amp U2A is given by

$$X_1 = -\frac{R_5}{R_4}V_a \tag{5}$$

where

$$V_a = -\frac{R_1}{R_3}V_{in} \tag{6}$$

Similarly, the output of the op –amp U3A is given by

$$Y_2 = +\frac{R_1}{R_3}\frac{R_{18}}{R_6}V_{in} \tag{7}$$

The output of the multiplier AD633 W, with four inputs X_1, X_2, Y_1 and Y_2 is given by

$$W = \frac{(X_1 - X_2)(Y_1 - Y_2)}{10} \tag{8}$$

where $X_2 = 0$ and $Y_1 = -5V$.

Table 1. Simulated outputs of the two logistic maps for two sets of initial conditions

Time	Outputs of two Logistic Maps for different initial conditions				Time	Outputs of two Logistic Maps for different initial conditions			
	LM1	LM2	LM1	LM2		LM1	LM2	LM1	LM2
	IC_1=0.01	IC_2=0.61	IC_1=0.01	IC_2=0.62		IC_1=0.01	IC_2=0.61	IC_1=0.01	IC_2=0.62
0.000000	0.010000	0.610000	0.010000	0.620000	1.794385	0.178817	0.242877	0.178817	0.272967
0.002000	0.046951	0.936724	0.046951	0.927774	1.994385	0.403378	0.959419	0.403378	0.952138
0.002029	0.046065	0.935903	0.046065	0.926953	2.158631	0.580185	0.724735	0.580185	0.781576
0.002086	0.046057	0.935895	0.046057	0.926945	2.292105	0.580209	0.724735	0.580209	0.781576
0.002202	0.046065	0.935903	0.046065	0.926953	2.417919	0.580197	0.724723	0.580197	0.781565
0.002343	0.046057	0.935895	0.046057	0.926945	2.539625	0.580220	0.724728	0.580220	0.781573
0.002625	0.046065	0.935903	0.046065	0.926953	2.739625	0.580200	0.724723	0.580200	0.921150
0.003189	0.046057	0.935895	0.046057	0.926945	2.900038	0.822607	0.925634	0.822607	0.490960
0.004318	0.046065	0.935903	0.046065	0.926953	3.010856	0.958488	0.242933	0.958488	0.674266
0.006576	0.046057	0.935895	0.046057	0.926945	3.087127	0.982857	0.566812	0.982857	0.674286
0.011091	0.046065	0.935903	0.046065	0.926953	3.157975	0.958389	0.784726	0.958389	0.674272
0.020122	0.046057	0.935895	0.046057	0.926945	3.227696	0.957951	0.786361	0.957951	0.292132
0.038183	0.046065	0.935903	0.046065	0.926953	3.319028	0.957937	0.786374	0.957937	0.962227
0.074305	0.046057	0.935895	0.046057	0.926945	3.407211	0.957950	0.786382	0.957950	0.972811
0.146550	0.046065	0.935903	0.046065	0.926953	3.583577	0.957932	0.786384	0.957932	0.882743
0.291039	0.046057	0.935895	0.046057	0.926945	3.783577	0.899331	0.678209	0.899331	0.864776
0.491039	0.046065	0.935903	0.046065	0.926953	3.960589	0.369571	0.972188	0.369571	0.864759
0.691039	0.046057	0.935895	0.046057	0.926945	4.088995	0.077192	0.962484	0.077192	0.955500
0.891039	0.046065	0.935903	0.046065	0.926953	4.235374	0.166072	0.663680	0.166072	0.982269
1.091039	0.178809	0.242885	0.178809	0.272975	4.367671	0.166094	0.663637	0.166094	0.119073
1.249799	0.178820	0.242880	0.178820	0.272971	4.502426	0.166105	0.663635	0.166105	0.326991
1.398783	0.178818	0.242877	0.178818	0.272967	4.693367	0.276909	0.782375	0.276909	0.439546
1.594385	0.178826	0.242875	0.178826	0.272966	4.893367	0.963484	0.476719	0.963484	0.464759
1.794385	0.178817	0.242877	0.178817	0.272967	5.093367	0.294014	0.181659	0.294014	0.413961

By suitably assigning the resistors as R_3, R_4, R_6 and R_{18} equal to 10KΩ and R_1 = 50KΩ, the output of the multiplier will be given by

$$W = \frac{V_{in} R_5 (1 - V_{in})}{4k} \tag{9}$$

By comparing the Eq.(9) with Eq.(1), it can be seen that the bifurcating parameter 'a' will be given by

$$a = \frac{R_5}{4k}.$$

So, in the Fig.2, the system is set to be in the chaotic regime by selecting R_5 to be equal to 15.6 kΩ, thus, setting the control parameter to 3.9.

5. CIRCUIT PERFORMANCE AND SIMULATION RESULTS

The proposed circuit for the PRBG is designed and simulated in Multisim software. The simulations are carried out for different sets of very close but independent initial conditions. The Table.1 summarizes the outputs of the two logistic maps in the chaotic regime, for the two such sets of initial conditions. In the first case, the initial conditions specified for the first and second logistic maps are IC_1 = 0.01, IC_2 = 0.61 whereas, for the second case, it is IC_1 = 0.01 and IC_2 = 0.62. The initial condition specified for the first logistic map were the same in both cases but it is slightly changed for the second logistic map. It is clearly evident from Table.1 that, even though, the outputs of the maps remain same for the initial few runs, it changes drastically after that, even for a very small change in the initial condition.

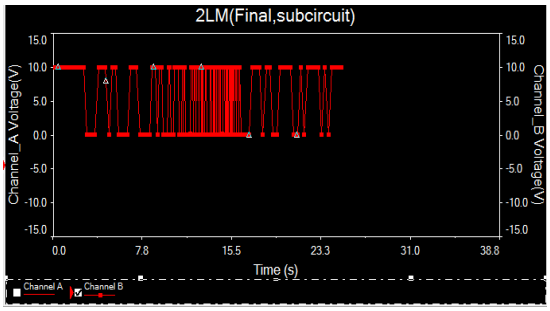

Fig.3. Comparator Output Voltage

Fig.3 shows the screenshot of the output of the comparator, realized using LM311 for the first set of initial conditions to the logistic maps ie, IC_1 = 0.01 and IC_2 = 0.61, simulated in Multisim. The output of the comparator is either high or low voltage, equivalent to a stream of 1's and 0's, which are random in nature and corresponds to the random bits generated.

6. NIST SUITE OF STATISTICAL TESTS FOR RANDOMNESS

The outputs of pseudo random bit generators are used in many of the cryptographic applications, such as the generation of a key material. The generators used for such applications should meet stronger requirements than that needed for other applications. The randomness of the generated sequence can be checked by carrying out a variety of statistical tests such as NIST Test Suite, Donald Knuth's statistical tests, The DIEHARD suite of statistical tests [8] [13]. These randomness tests may be useful as a first step to determine if or not a generator is suitable for a particular cryptographic application, although these tests cannot absolutely certify a generator as appropriate for usage in a particular application. The tests basically focus on different types of non-randomness that can exist in a binary sequence. To check the randomness of the bit sequences at the outputs of the proposed pseudo random bit generator, few statistical tests of the NIST suite were carried out on the generated bits [13][15]. They are:

- **Frequency or monobit test**: The focus of the test is the proportion of zeroes and ones for the entire sequence. The purpose of the test is to determine whether the number of ones and zeroes in a sequence are approximately the same as would be expected for a truly random sequence [8][13].
- **Frequency test within a block**: The focus of the test is the proportion of one's within M-bit blocks. The purpose of this test is to determine whether the frequency of ones in an M-bit block is approximately $M/2$, as would be expected under the assumption of randomness [13].
- **Runs test:** The focus of this test is the total number of runs in the sequence, where a run is an uninterrupted sequence of identical bits. The purpose of the test is to determine whether the number of runs of ones and zeros of various lengths is as expected for a random sequence or particularly, it determines whether the oscillation between such zeros and ones is too fast or too slow [8] [13].
- **Test for the longest run of ones in a block**: The focus of the test is the longest run of the ones within M-bit blocks. The purpose of this test is to determine whether the length of the longest run of ones within the tested sequence is consistent with the length of the longest runs of the ones that would be expected in a random sequence [8] [13].
- **One Sample Kolmogorov Smirnov Test:** K-S test is a non-parametric test of the equality of one-dimensional, continuous probability distributions that can be used to compare a sample with a reference probability distribution. The test is based on the empirical distribution function (ECDF) [13].
- **Chi-Squared Test:** It is a statistical hypothesis test, wherein the sampling distribution of the test statistic is a chi-square distribution when the null hypothesis is true.[13]

The Table.2 summarizes the results of the various parametric and non-parametric tests conducted on the output sequences obtained using two nearly close initial conditions for the two logistic maps. The statistical results are obtained for PRBG, for four random sequences of 100 bits each, obtained with logistic maps of control parameter 3.9 and following initial conditions:

- IC_1 = 0.01, IC_2 = 0.61
- IC_1 = 0.01, IC_2 = 0.62
- IC_1 = 0.02, IC_2 = 0.08
- IC_1 = 0.03, IC_2 = 0.09.

Table 2. Statistical Test Results

Statistical Tests	**P –values**				**Result**
Parametric Tests	**Sequence 1**	**Sequence 2**	**Sequence 3**	**Sequence 4**	
Frequency (monobit) test	0.6485	1	0.5485	0.617	Random
Runs Test	0.111	0.2316	0.512	0.3639	Random
Test for longest run of ones in a block	0.1436	0.9703	0.813	0.0872	Random
Binary matrix rank Test	0.165	0.745	0.267	0.1278	Random
Cumulative sums test	0.234	0.435	0.1954	0.2567	Random
L-Z compression test	0.178	0.367	0.635	0.3623	Random
Non-Parametric Tests					
Frequency test within a block	0.765	0.9821	0.5643	0.989	Random
	h – values				**Result**
Kolmogorov Smirnov (KS) Test	1	1	1	1	Random
Chi-Squared Test	1	1	1	1	Random

- Number of binary sequences tested: 4
- Length of each binary sequence: 100 bits
- Significance level (α) = 0.01
- Null Hypothesis (H_0): The binary sequence is random
- If P-value $\geq \alpha$ (0.01), then the null hypothesis (H_0) is accepted
- If P-value < α (0.01), then the null hypothesis (H_0) is rejected
- For KS and chi-squared tests, h=1 implies a random sequence
- h=1, indicates the test accepts the null hypothesis at a specified significance level
- Sequence 1: Initial Conditions IC_1 = 0.01 and IC_2 = 0.61
- Sequence 2: Initial Conditions IC_1 = 0.01 and IC_2 = 0.62
- Sequence 3: Initial Conditions IC_1 = 0.02 and IC_2 = 0.08
- Sequence 4: Initial Conditions IC_1 = 0.03 and IC_2 = 0.09

7. CONCLUSION

In this paper, an electronic circuit realization of a pseudo random bit generator is presented and simulated using MULTISIM. The PRBG is based on two chaotic maps, logistic maps, starting from two random independent initial conditions. Simulations were carried out for different sets of initial conditions and the outputs are presented. The output sequences obtained from the logistic maps are also tested for randomness using various statistical tests under the NIST suite. The results of the tests were found to be highly encouraging and seem to have perfect cryptographic properties, hence can be used in the design of new stream ciphers. The results obtained also proved the property of 'sensitivity to initial conditions 'of the realized PRBG circuit.

REFERENCES

[1] B. Schneier, "Applied Cryptography-Protocols, Algorithms and Source Code in C", John Wiley and Sons, 1996.

[2] A.J. Menezes, P.C.V. Oorschot and S. Vanstone, "*A Handbook of Applied Cryptography*", CRC Press, 1997.

[3] S. Oishi and H. Inoue, "Pseudo-Random Number Generators and Chaos", *Transactions of the Institute of Electronics and Communication Engineers of Japan E*, Vol. 65, pp. 534-541, 1982.

[4] T. Lin and L.O. Chua, "New Class of Pseudo-Random Number Generator based on Chaos in Digital Filters", *International Journal of Circuit Theory and Applications*, Vol. 21, pp. 473-480, 2017.

[5] J.A. Gonzalez and R. Pino, "Random Number Generator based on Unpredictable Chaotic Functions", *Computer Physics Communications*, Vol. 120, pp. 109-114.1999.

[6] T. Stojanovski and L. Kocarev, "Chaos-Based Random Number Generators - Part I Analysis", *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, Vol. 48, pp.281-288, 2001.

[7] S. Li, X. Mou and Y. Cai, "Pseudo-Random Bit Generator based on Couple Chaotic Systems and its Application in Stream-Ciphers Cryptography", *Proceedings of International Conference on Computer Science*, pp. 316-329, 2001.

[8] Patidar Vinod and Sud K.K. Anovel, "Pseudorandom Bit Generator based on Chaotic Standard Map and its Testing", *Electronic Journal of Theoretical Physics*, Vol. 4, No. 2, pp. 327-344, 2009.

[9] N.K. Pareek, Patidar Vinod and K.K. Sud, "A Pseudo Random Generator based on Chaotic Logistic Map and its Statistical Testing", *Informatica*, Vol. 33, pp. 441-452, 2009.

[10] M. Francois, T. Grosges, D. Barchiesi and R. Erra, "A New Pseudo Random Number Generator Based on Two Chaotic Maps", *Informatica*, Vol. 24, No. 2, pp. 181-197, 2013.

[11] M. Hamdi, R. Rhouma and S. Belghith, "A Very Efficient Pseudo-Random Number Generator Based on Chaotic Maps and S-Box Tables", *International Journal of Computer, Electrical, Automation, Control and Information Engineering*, Vol. 9, No. 2, pp. 1-13, 2015.

[12] Kathleen T. Alligoodet al., "*Chaos, An Introduction to Dynamical Systems*", Springer, 2008.

[13] H. Soumya Babu and K. Gopakumar, "Chaos: A Pseudo Random Bit Generator using Iterated Maps", *Proceedings of International conference on Signal and Speech Processing*, pp. 1-9, 2017.

[14] M. Suneel, "Electronic Circuit Realization of the Logistic Map", *Sadhana*, Vol. 31, No. 1, pp. 1-14, 2006.

[15] A.L. Runkin and L.E. Bassham, "Statistical Test Suite for Random and Pseudo Random Number Generators for Cryptographic Applications", *Proceedings of International Conference on Cybersecurity and Statistical Analysis*, pp. 800-822, 2001.