Original scientific paper

Informacije MIDEM
Journal of Microelectronics,
Electronic Components and Materials
Vol. 44, No. 4 (2014), 303 – 311

A New Method for Hybrid Pseudo Random Number Generator

Erdinç Avaroğlu[1], Taner Tuncer[2], A. Bedri Özer[2], Mustafa Türk[3]

[1]Departmant of Information Technology, Inönü University, Malatya/Turkey
[2]Cumputer Engineering Department, Firat University,Elaziğ/Turkey
[3]Electrical and Electronics Engineering Department, Firat University,Elaziğ/Turkey

Abstract: Powerful cryptographic systems need qualified random numbers. Qualified random numbers need providing good statistical qualities, not predicting and not re-generating. The numbers generated by raw Pseudo Random Number Generators (PRNG) can be predicted when their seed value are detected or the functions used in the system are not complicated enough. Moreover, the stream generated repeats itself after its period is exhausted. Due to these shortcomings mentioned above, raw PRNGs are not suitable for the cryptographic applications. In order to eliminate these shortcomings, by adding an additional input to the raw PRNG system, a hybrid structure is suggested in this study. In the hybrid system, a chaotic attraction in order to generate pseudo random number and a TRNG system having 5 Ring Oscillator (RO) each of which includes 3 inverters as the additional input were used. The random numbers obtained from the suggested hybrid structure were exposed to the NIST 800.22 statistical tests and it is shown that hybrid system can be used in the cryptographic systems.

Keywords: Random Number Generator, Chaotic Attractor, Additional Input, Hybrid Pseudo Random Number Generator

Nova metoda za hibridne pseudo naključne generatorje števil

Izvleček: Dobra kriptografija potrebuje kvalitetna naključna števila. Kvalitetna naključna števila zahtevajo dobro statistično kvaliteto, ki ni predvidljiva in ponovljiva. Števila generirana s psevdonaključnim generatorjem (PRNG) so lahko predvidljiva, če je odkrito seme in če sistemske funkcije niso dovolj kompleksne. Poleg tega se po preteku periode generiranje števil ponovi, zaradi česar PRNG ni primeren za kriprografijo. Kot rešitev problemov predlagamo uvedbo hibridnega sistema, ki dodaja dodatno vhodno spremenljivko PRNG. V hibridnem sistemu je bil uporabljen TRNG sistem s petimi obročnimi oscilatorji s tremi razsmerniki. Generirana števila so bila podvržena NIST 800.22 testu, ki je pokazal uporabnost hibridnega sistema za kriptografske namene.

Ključne besede: generator naključnih števil, dadaten vhod, hibridni prevdo naključen generator števil

** Corresponding Author's e-mail: eavaroglu@gmail.com*

1 Introduction

The numbers obtained from the random number generators are needed in the statistical samplings, simulations, numerical analysis, entertainment and cryptography. In the various cryptographic applications, especially random numbers are obligatory because cryptography needs random numbers to generate and distribute the keys, to make the initial vector, to generate prime numbers and passwords and in the identity verification authentication protocols. The security of a cryptographic system depends on the true randomness of the numbers obtained. For that reason, the random numbers used in the cryptographic systems have to supply some basic requirements, which are good statistical qualities and not being predicted. As a result, qualified cryptography needs qualified random numbers [1].

In order to obtain random numbers, different random number generators were developed [2-4]. In general, these random number generators are classified as True Random Number Generators (TRNG) and pseudo random number generators (PRNG).

The True Random Number Generators generate random numbers by using the real physical processes which cannot be controlled and predicted as the noise

source. The randomness and the qualities of the random numbers generated by the TRNG depend on the randomness of the physical processes. If there are unpredictable physical processes, the numbers generated cannot be predicted and controlled as well. However, some bit stream generated may have statistical weaknesses. In order to eliminate these weaknesses, the bit stream is exposed to post processing. While post processing applications eliminate the weaknesses, it leads to a decrease in the bit rate. TRNG's disadvantage is that it is slow, costly and depends on the hardware. However, TRNGs were used in a lot of applications in the cryptology as they supplied the qualities of not being predicted, not being re-generated and good statistical qualities which are obligatory qualities for the cryptographic applications [2, 5, 6]. Because of these qualities, TRNGs are used in the hybrid PRNGs as the additional input.

Raw pseudo random number generators cannot generate numbers without having an initial (seed) value. The seed must be chosen randomly. When the specified seed value is used as an input in a certain algorithm, long random number streams are generated. Raw PRNG's advantage is that it is cheap, fast, easily realized and does not need hardware. However, the numbers generated by the raw PRNGs are easily predicted when their seed value is detected or the functions used in the system are not complicated enough. Also, the stream generated repeats itself after its period is exhausted. Due to these shortcoming mentioned above, raw PRNGs are not suitable for the cryptographic applications [2, 7, 8].

Many raw PRNGs are designed as the chaotic system [3, 4]. The most important feature of the chaotic systems is that they depend on the start (initial) condition. These systems show unpredictable behaviours and features which are not periodical [9, 10]. There are different studies performed by using chaotic signs. Some of these studies are double scroll chaotic structure [3, 11], performance scale for the random number generator based on discrete time chaos [4] and oscillator sampling method [5]. Though the discrete time chaotic systems are too simple models, they are insufficient in terms of complicatedness. That's why, more complicated chaotic systems must be used.

Different entropy sources such as jitter and metastable were used in TRNG. Especially the jitter in the ROs was preferred in many TRNG applications. TRNG in the study performed at [6] consist of 114 RO and each RO from 13 inverters. The random signals obtained by RO were combined XOR process. The signals obtained after XOR process was sampled by using D flip-flop. In this method, a random number that has a 2.5 Mbit/s data rate was generated. However, the power consumption is too high due to using too many logic components in the system. The design performed at [12, 6] was obtained by using 110 RO 3 inverters on Xilinx Virtex II Pro FPGA. The output rate obtained 2 Mbps. In [13], a new TRNG is suggested by adding a D flip-flop to the output of a RO in order to increase the randomness quality in the design performed at [6]. In the system, 25 ring oscillators and 3 inverters were used and the output bit streams were not exposed to a post processing. In this system, the output bit rate was obtained as 100 Mbit/s. 25 ROs and 3 inverters used in the study performed at [13] were decreased to 5 ROs and 3 inverters. As a result, the random numbers obtained were observed to pass the NIST statistical test.

In this study, in order to eliminate the shortcomings of the raw PRNGs such as being predicted, being regenerated and not providing good statistical qualities, additional input was supplied to the output function to provide complicatedness to the raw PRNG functions. The additional input increases the security because of bringing the quality of not being predicted and randomness qualities. Also, the PRNG structure used in the study is chaotic attractor with multiple modes. Random numbers were generated by sampling the state variable obtained from the 2+2, 2+4 and 5+4 chaotic attractor. By sampling the numbers obtained, raw random bit stream were obtained. The randomness of the raw bit stream obtained was tested by using the NIST test suit software performed at [14]. In order to eliminate the correlation in the raw random bit stream that have poor negative results, raw bit stream were exposed Von Neumann and XOR post processing. However, 2+4 and 5+4 chaotic attractor were detected not to pass the tests in the last process (post processing). For that reason, in order to increase the security of the system, the complicatedness of the output function and statistical quality, a TRNG generating true random number was added to the output function as the additional input.

The parts of this study were organized as follows. In part 2, pseudorandom number generators and additional input process was mentioned. In part 3, chaotic attractors used in the random number generator, adding an additional input to the raw PRNG system and the statistical test results of the random bit stream obtained by adding an additional input were given. In part 4, the fulfilment of the system by mentioning about the hybrid system proposed. In the last part, the results were assessed.

2 Pseudo random number generator

PRNGs have deterministic architecture and generated numbers by these generators show periodic characteristic [2, 7, 8]. Therefore, generated numbers by PRNGs are not true numbers. Pseudo random number can easily be generated by using a specific algorithm and seed. Although, obtaining a seed from random entropy source make difficult the estimation of the numbers generated by PRNG, it is possible to estimate the numbers.

In the Fig. 1, a general design of the raw PRNG is shown. $s_n \epsilon S$ is the internal state value of the PRNG. Here, the finite sets S and R are named as state space and output space. As seen in the Fig. 1, $\Psi:S\rightarrow R$ output function calculates the next random number r_n from the inner situation value s_n. Then, s_n value is changed with transition function $\phi_H(S_{n+1}=\phi(s_n))$ as s_{n+1}. The first state value s_1 is generated as $s_1=\phi(s_0)$ from the seed value used in the first entry s_0. The important part in the generating phase is that all the situation values $s_1,s_2,....$ and all the generated random numbers $r_1,r_2,\ldots$ depend on the seed value s_0. This poses a risk of exposure of the whole system if the s_0 value is known. That's why, in order to provide the quality of not being predicted, s_0 seed value has to be chosen randomly. Raw PRNG is defined with the stream of variables (S, R, Ψ, ϕ, p_s). Here, p_s is defined as the possibility distribution of the random seed. The generation of the seed is fulfilled outside of the PRNG system. In order to provide a quality of not being predicted, it is usually generated by a TRNG [2].

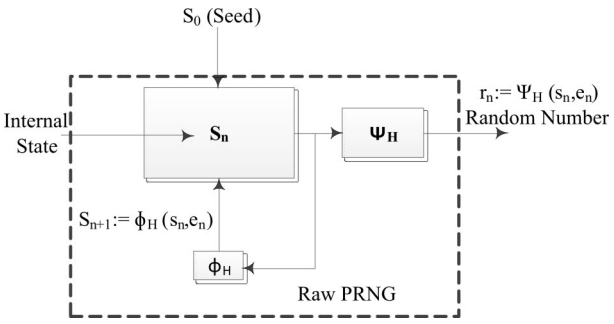


Figure 1: General design of the raw PRNG

The only disadvantage of the PRNG is that the random numbers are determined by the seed and the next random number depends only on the current internal state value. Their advantage is that it is cheaper compared with the other generators and does not need any hardware. In order to provide the feature of being unpredictable, the seed entropy must be large and the transition and exit functions must complicated enough.

In order to eliminate these mentioned shortcomings and to make its functions complicated, an additional input must be provided from $e_n \in E$ which is a finite set as seen in the Fig. 2, which is different from raw PRNG. The additional input increases the security due to its bringing the features of being unpredictable and randomness.

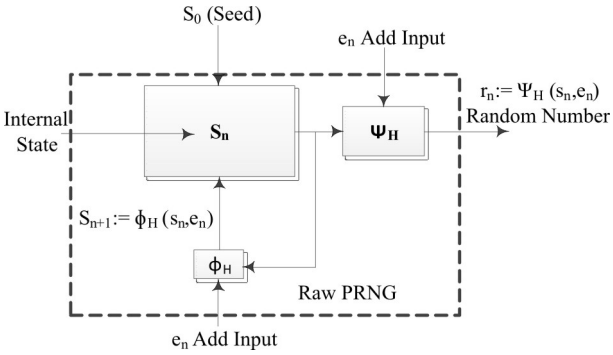


Figure 2: Hybrid design of the raw PRNG

3 Chaotic attractors

Recently, chaos has been started to use a lot in the random number generators [10]. It has been proved that chaotic number stream is easy and fast to generate and store. Only a few functions (chaotic map) and parameters (initial condition) are enough to generate long number streams. Also, too many different number streams can be easily generated by simply changing the condition of start. Thanks to these advantages, chaos has been started to use as the random number generator [10].

In this study, Chua circuit was used whose equations were given in (1), (2), (3) and (4) in order to obtain the chaotic attractor [15].

$$\begin{aligned} \dot{x} &= y + f_1(y) \\ \dot{y} &= z \\ \dot{z} &= -a \cdot x - a \cdot y - a \cdot z + f_2(x) \end{aligned} \tag{1}$$

$$f_1(y) = \sum_{i=1}^{M_1} g_{\frac{(-2i+1)}{2}}(y) + \sum_{i=1}^{N_1} g_{\frac{(2i+1)}{2}}(y) \tag{2}$$

$$f_2(x) = \sum_{i=1}^{M_2} g_{\frac{(-2i+1)}{2}}(x) + \sum_{i=1}^{N_1} g_{\frac{(2i+1)}{2}}(x) \tag{3}$$

$$g_\theta(\alpha) = \begin{cases} 1 & \alpha \geq \theta, \theta > 0 \\ 0 & \alpha < \theta, \theta > 0 \\ 0 & \alpha \geq \theta, \theta < 0 \\ -1 & \alpha < \theta, \theta < 0 \end{cases} \quad (4)$$

In the equation 2 and 3, M_1, N_1, M_2, N_2, i, j, $\in \Re^+$. The effect of $f_1(y)$ on the system is ignored and when a value is taken as 0.4, the system shows a double scroll attractor). $f_1(y)$ and $f_2(x)$ represent the split linear elements with many breakpoints and have the same characteristics. Y state variable affects $f_1(y)$ and x state variable affects $f_2(x)$. These elements show a dynamic structure depending on the M and N parameters. The number of the breakpoints and places are determined with the M and N values.

In the system, chaotic attractors will form along the linear of x and y=-ax+b. Chaotic scrolls under the effect of the $f_2(x)$ form along the axis of x. Chaotic scrolls under the effect of the $f_1(y)$ form along the linear of y=-ax+b. As chaotic attractors form in both directions, this behaviour type is named as the attractor of n+n. The first n value shows the axis of x and the other n value shows the total scrolls formed along the linear of y=-ax+b.

The behaviour of the system was obtained by solving the differential equation given in the equation of (1). Runge-Kutta method with four steps was used by using Matlab program. The start values of the state variables were taken as (x_0, y_0, z_0)=(0.1,0.1,-0.1) and a value was taken as 0.4. In the study performed in the [15], the experimental fulfilment of the attractors and their oscilloscope outputs were given.

The sampling process from X state variable obtained from the 2+2, 2+4 and 5+4 chaotic attractor given in the Fig. 3 was made as shown in Fig. 4. A raw bit stream was obtained by applying the rule shown in the equation 5 with the obtained numbers.

$$S(x) = \begin{cases} 0 & x < 0.5 \\ 1 & x >= 0.5 \end{cases} \quad (5)$$

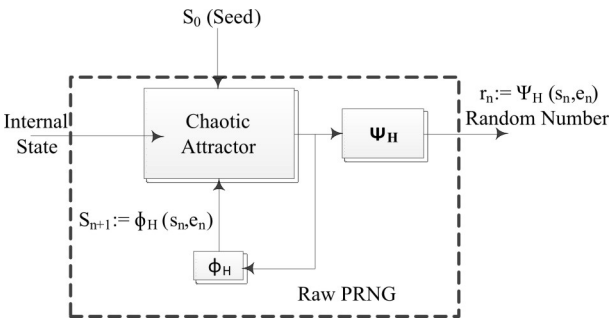


Figure 3: Raw PRNG based chaotic attractor

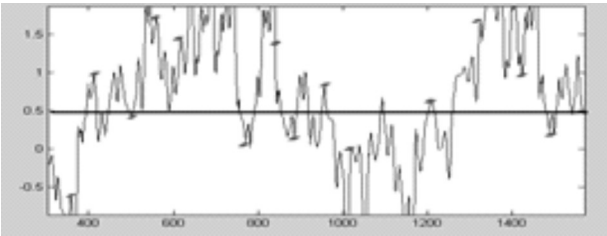
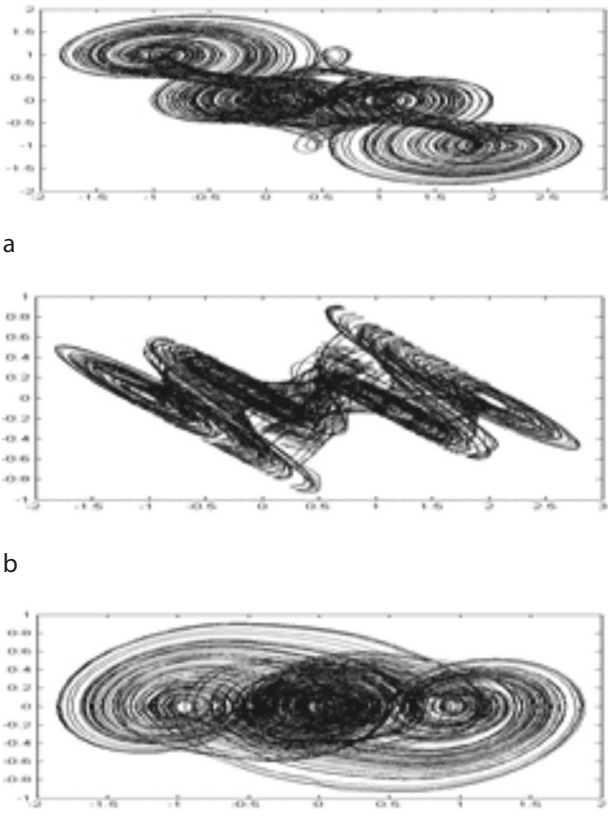


Figure 4: The simulation of a certain area of 2+2 attractor

3.1 Random Number Generation by Obtaining 2+2 Attractor and the Statistical Results

In order to obtain 2+2, M_1=1, N_1=2, M_2=0 and N_2=1 was taken and the changes of x-y, x-z and y-z obtained were shown in the Fig. 5 [15]. In the direction of PRNG design shown in the of Fig. 4, 108418 raw bit stream was obtained by taking a value in every 300 steps as from the first 500 numbers and without exposing to any post processing[16]. The statistical test results of the raw bit stream obtained were shown in Table 1.



a

b

c

Figure 5: 2+2 attractor illustration for M_1=1, N_1=2, M_2=0 and N_2=1 , (a) x-y change (b) x-z change (c)y-z change

Table 1: NIST test results for 2+2 attractor

Test Name	P value	Result
Frequency (Monobit) Test	0.151	Passed
Frequency Test within a Block	0.484	Passed
Runs Test	0.028	Passed
Test for the Longest Run of Ones in a Block	0.069	Passed
Binary Matrix Rank Test	0.675	Passed
Discrete Fourier Transform Test	0.442	Passed
Non-overlapping Template Matching Test	0.086	Passed
Overlapping Template Matching Test	0.778	Passed
Maurer's Universal Statistical Test	0.050	Passed
Linear Complexity Test	0.580	Passed
Serial Test	0.246	Passed
	0.594	
Approximate Entropy Test	0.106	Passed
Cumulative Sums Test	0.085	Passed

3.2 Random Number Generator By Obtaining 2+4 Attractor And The Statistical Results

In order to obtain 2+4 attractor, M_1=2, N_1=2, M_2=0 and N_2=1 and x-y, x-z and y-z obtained were shown in the Fig. 6 [15]. From the X state variable obtained from 2+4 attractor, a value was taken in every 1000 steps from the first 500 numbers and 122807 raw bit stream was obtained. The statistical test results of this obtained raw bit stream and the bit stream obtained after the post processing were shown in the Table 2.

3.3 Random Number Generator By Obtaining 5+4 Attractor And The Statistical Results

In order to obtain 5+4 attractor, M_1=2, N_1=2, M_2=2 and N_2=2 and x-y, x-z and y-z obtained were shown in the Fig. 7 [15]. From the X state variable obtained from 5+4 attractor, a value was taken in every 300 steps from the first 500 numbers and 99062 raw bit stream was obtained. The statistical test results of this obtained raw bit stream and the bit stream obtained after the post processing were shown in the Table 3.

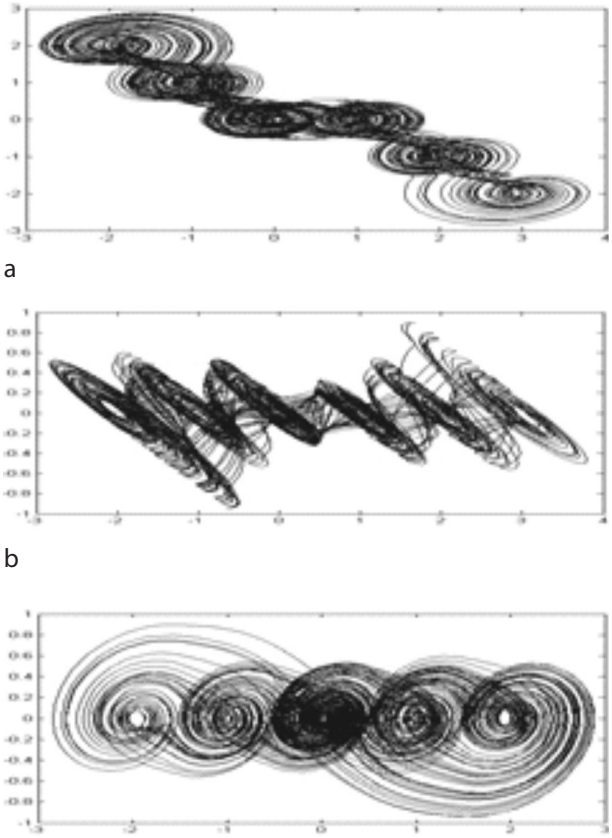
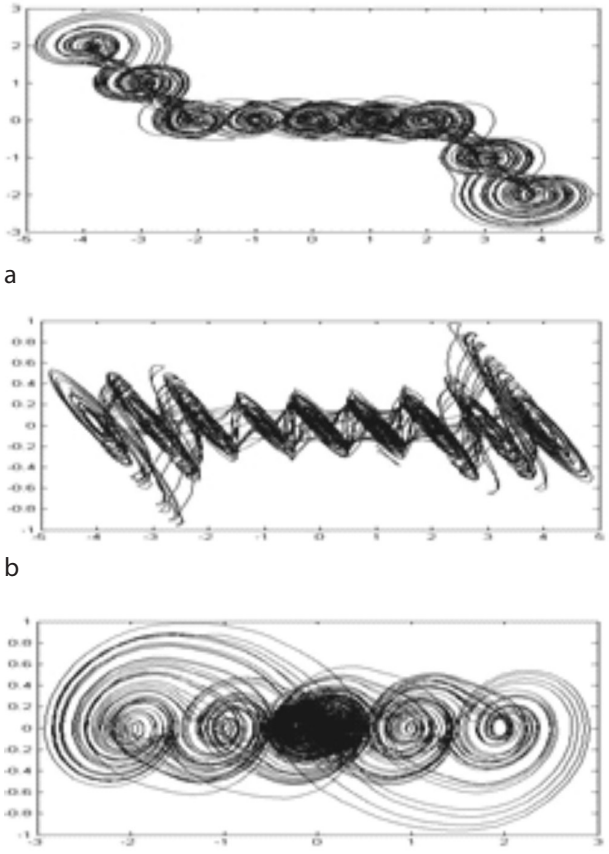


Figure 6: 2+4 attractor illustration for M_1=2, N_1=2, M_2=0 and N_2=1 , (a) x-y change (b) x-z change (c)y-z change

Table 2: NIST test results for 2+4 attractor

The name of the Test	P value 122807 bit	Van Neumann 24061 bit	Xor 61403 bit	Result
Frequency (Monobit) Test	0.786	0.259	-	Unpassed
Frequency Test within a Block	-	0.935	-	Unpassed
Runs Test	-	-	-	Unpassed
Test for the Longest Run of Ones in a Block	-	-	-	Unpassed
Binary Matrix Rank Test	0.343	0.809	0.491	Passed
Discrete Fourier Transform Test	-	0.516	-	Unpassed
Non-overlapping Template Matching Test	-	0.012	-	Unpassed
Overlapping Template Matching Test	-	0.138	-	Unpassed
Maurer's Universal Statistical Test	-	0.078	-	Unpassed
Linear Complexity Test	0.754	0.100	0.859	Passed
Serial Test	-	-	-	Unpassed
	-	0.496	-	
Approximate Entropy Test	-	-	-	Unpassed
Cumulative Sums Test	0.126	0.231	-	Unpassed



a

b

c

Figure 7: 5+4 attractor illustration for M_1=2, N_1=2, M_2=2 and N_2=2 , (a) x-y change (b) x-z change (c)y-z change

4 The proposed hybrid PRNG

In the system of raw PRNG performed by using chaotic attractors, the sampling process used in obtaining random numbers do not always produce good results.

Because, the correct determination of the threshold value shown in the equation 2 is very significant and this affects the quality of the entropy. For instance; 2+2 attractor passes the statistical results without exposing a post processing according to the results in the Table 1; whereas, it is observed that, with and without last processes (post processing), the results of the raw bit stream obtained from 2+4 and 5+4 attractors failed to pass the statistical tests, as seen in the of Table 2 and 3. Therefore, in order to increase the safety and the randomness of the 2+4 and 5+4 chaotic attractors, TRNG system was given as additional input as shown in the Fig. 8. In the system of TRNG, there are 5 RO, each of which contains 3 inverters. Fig. 9 shows the TRNG system used in the hybrid system.

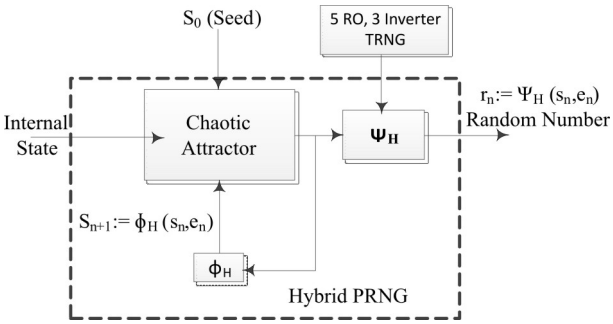


Figure 8: The suggested hybrid system

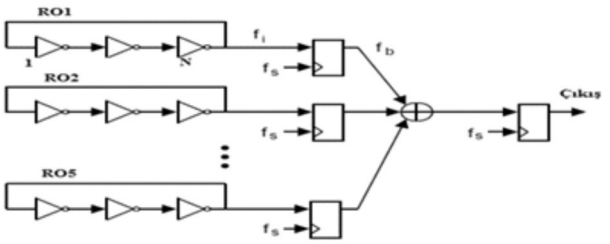


Figure 9: The design of 3 inverter 5RO TRNG

Table 3: NIST test results for 5+4 attractor

The name of the Test	P value 99062 bit	Van Neumann 13505 bit	XOR 49530 bit	Result
Frequency (Monobit) Test	-	0.636	-	Unpassed
Frequency Test within a Block	-	0.973	-	Unpassed
Runs Test	-	-	-	Unpassed
Test for the Longest Run of Ones in a Block	-	-	-	Unpassed
Binary Matrix Rank Test	0.194	0.683	0.414	Passed
Discrete Fourier Transform Test	-	0.022	-	Unpassed
Non-overlapping Template Matching Test	-	-	-	Unpassed
Overlapping Template Matching Test	-	0.142	-	Unpassed
Maurer's Universal Statistical Test	-	0.286	-	Unpassed
Linear Complexity Test	0.529	0.967	0.347	Passed
Serial Test	-	-	-	Unpassed
	-	0.013	-	
Approximate Entropy Test	-	-	-	Unpassed
Cumulative Sums Test	-	0.743	-	Unpassed

Fig. 10 shows RO in the TRNG system performed in FPGA. Each RO used in the system was performed by using data flow and schematic design methods with VHDL language [14].

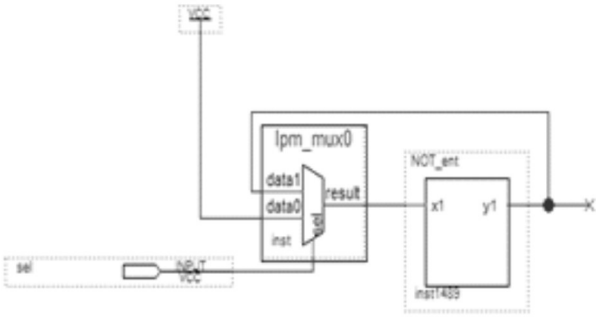
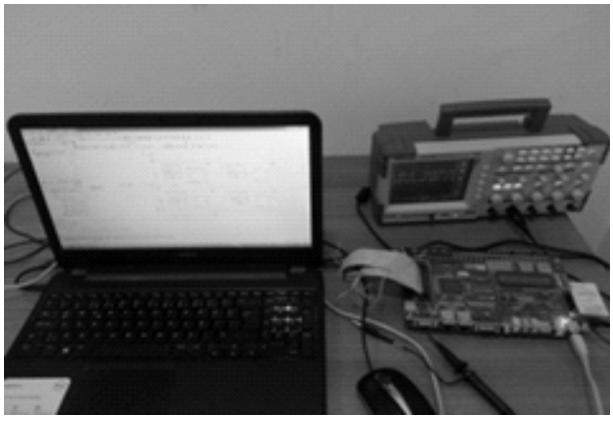
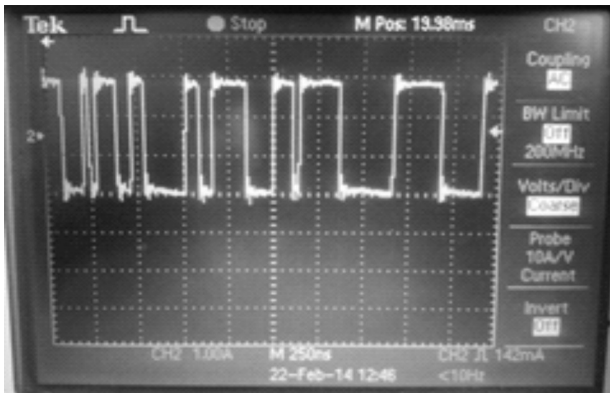


Figure 10: Ring Oscillator

In the Fig. 10, the inverter exit is always at the level of logic 0. In order for RO to obtain a random sign, RO exit (output) is obtained after giving an excitation signal (sel=1) from the physical environment. Random number generation is completed after sampling the random change obtained. Fig. 11 shows the performance of TRNG, used in the hybrid system, in FPGA. In order for the random numbers generated by the TRNG system to perform the statistical tests, the numbers were recorded to a memory unit. Fig. 12.a shows the appearance of the experimental set and Fig. 12.b shows the numbers generated by TRNG in the real time.



a



b

Figure 12: (a) The appearance of the experimental set (b)The bits for 5 RO and 3 inverters generated in the real time

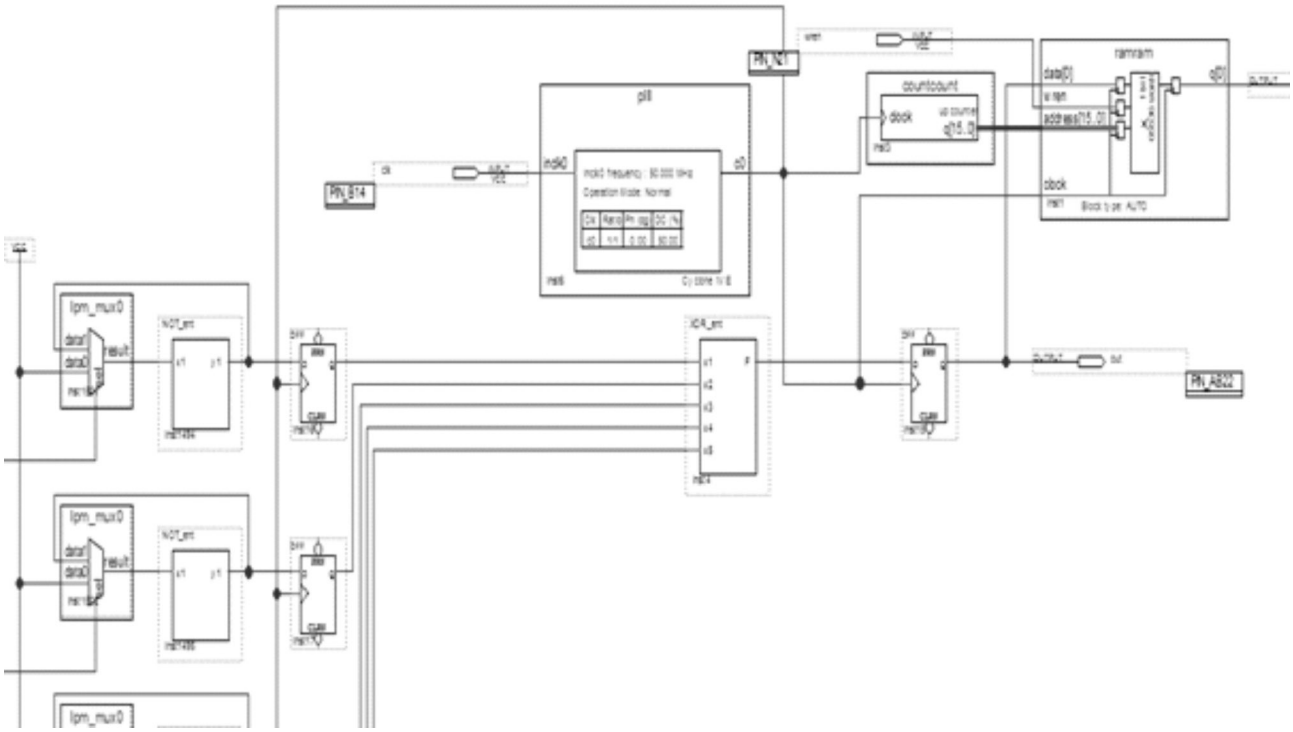


Figure 11: The performance of 5 RO and 3 Inverter TRNG FPGA

As seen in the Fig. 8, an output bit stream was formed by exposing the bit stream obtained from the chaotic attractor and TRNG design to XOR process in the exit function. As a result, the bit stream obtained was exposed to the NIST statistical test whose software was done in [16], without performing a post processing. The results of the suggested hybrid PRNG were given Table 4 and 5. It was observed that the bit stream obtained passed all the statistical tests without performing a post processing.

Table 4: Test result of hybrid PRNG for 2+4 attractor

The name of the Test	P value	Result
Frequency (Monobit) Test	0.506	Passed
Frequency Test within a Block	0.923	Passed
Runs Test	0.638	Passed
Test for the Longest Run of Ones in a Block	0.393	Passed
Binary Matrix Rank Test	0.833	Passed
Discrete Fourier Transform Test	0.029	Passed
Non-overlapping Template Matching Test	0.070	Passed
Overlapping Template Matching Test	0.997	Passed
Maurer's Universal Statistical Test	0.120	Passed
Linear Complexity Test	0.024	Passed
Serial Test	0.641	Passed
	0.210	
Approximate Entropy Test	0.868	Passed
Cumulative Sums Test	0.709	Passed

Table 5: Test result of hybrid PRNG for 5+4 attractor

The name of the Test	P value	Result
Frequency (Monobit) Test	0.656	Passed
Frequency Test within a Block	0.941	Passed
Runs Test	0.714	Passed
Test for the Longest Run of Ones in a Block	0.980	Passed
Binary Matrix Rank Test	0.404	Passed
Discrete Fourier Transform Test	0.016	Passed
Non-overlapping Template Matching Test	0.020	Passed
Overlapping Template Matching Test	0.269	Passed
Maurer's Universal Statistical Test	0.661	Passed
Linear Complexity Test	0.099	Passed
Serial Test	0.458	Passed
	0.584	
Approximate Entropy Test	0.243	Passed
Cumulative Sums Test	0.810	Passed

5 Results

In this study, an additional input was added to the system in order to eliminate the shortcomings of the pseudorandom number generators and increase the complicatedness of the functions used. Random bit stream obtained from the 2+2 chaotic attractor passed the statistical test results without performing a post processing. However, random bit stream obtained from 2+4 and 5+4 chaotic attractors failed to pass the statistical tests with and without post processing. In order to eliminate this disadvantage, TRNG was used as the additional input based on RO. Thus, the randomness and the safety of the system were increased and the random bit stream generated by the hybrid system passed the statistical tests. This result indicates that the hybrid system can be used in the fields of cryptography.

6 References

1. Wold, K., "Security Properties of a Class of True Random Number Generators in Programmable Logic", Thesis submitted to Gjøvik University College for the degree of Doctor of Philosophy in Information Security, 2011.
2. Koç Ç., "Cryptographic Engineering", Springer-Verlag, 2009.
3. Ergün, S., Özoğuz, S., «A chaos-modulated dual oscillator-based truly random number generator.», In Proceedings, International Symposium on Circuits and Systems, 2482–2485, 2007
4. Beirami A., Nejati H., Massoud Y., "A performance metric for discrete-time chaos-based truly random number generators", 51st Midwest Symposium on Circuits and Systems, p:133-136, 2008.
5. Yalcın M. E., Suykens J. A. K., Vandewalle, J., «True random bit generation from a double scroll attractor.», IEEE Trans. Circuits and Systems-I, 51(7):1395–1404, 2004.
6. Sunar B., Martin W.J., Stinson, D.R.,»A Provably Secure True Random Generator with Built-In Tolerance to Active Attacks», in IEEE Transaction On Computers, vol. 56, No.1, January 2007.
7. Akram, R.N., "Pseudorandom Number Generation in Smart Cards: An Implementation, Performance and Randomness Analysis, New Technologies", Mobility and Security (NTMS), 2012 5th International Conference on, 1-7, 7-10 May 2012.
8. Sobotka, J. and Zeman, V., "Design of the true random numbers generator", Elektrorevue, 2(3):1-6, September 2011.
9. Strogatz, S., «Nonlinear Dyanamics and Chaos», Westview Press, Cambridge, 2001.
10. Tuncer, T., Celik, V., «Hybrid PRNG based on Logis-

tic Map,» 21st Signal Processing and Communications Applications Conference (SIU), pp.1,4, 2013.
11. Ergün S., Ozoguz S., "Truly Random Number Generators Based On a Double-Scroll Attractor", 49th IEEE International Midwest Symposium on Circuits and Systems, p:322-326, 2006.
12. Schellekens D., Preneel B., Verbauwhede I., «FPGA Vendor Agnostic True Random Number Generator», In Proceedings of the 16th International Conference on Field-Programmable Logic and Applications (FPL'06) (2006), IEEE, pp. 1–6, 2006.
13. Wold K., Tan C.H.,, «Analysis and Enhancement of Random Number Generator in FPGA Based on Oscillator Rings», In Proceedings of the 4th IEEE International Conference on Reconfigurable Computing and FPGAs (ReConFig'08) (2008), pp. 385–390, 2008.
14. Avaroğlu E., "Hardware Based Realization Of Random Number Generator", Phd Thesis, Electrical and Electronics Engineering Fırat, University, 2014
15. Türk, M., Ata, F., «The multi-mode chaotic behaviours: N+N and 2D N-scroll chaotic attractors», COMPEL: The International Journal for Computation and Mathematics in Electrical and Electronic Engineering, 25(4):929-939, 2006.
16. Avaroglu, E., Türk M., «Random number generation using multi-mode chaotic attractor», Signal Processing and Communications Applications Conference (SIU), 2013 21st ,1-4, 2013.

Arrived: 04. 06 .2014
Accepted: 18. 08. 2014