Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2019) 27: 847 – 860
© TÜBİTAK
doi:10.3906/elk-1806-167

A new computer-controlled platform for ADC-based true random number generator and its applications

Selçuk COŞKUN[1], **İhsan PEHLİVAN**[2], **Akif AKGÜL**[2,*], **Bilal GÜREVİN**[3]

[1]Department of Electronics and Computer Education, Institute of Natural Sciences, Sakarya University, Sakarya, Turkey

[2]Department of Electrical and Electronics Engineering, Faculty of Technology, Sakarya University of Applied Sciences, Sakarya, Turkey

[3]Department of Mechatronics Engineering, Faculty of Technology, Sakarya University of Applied Sciences, Sakarya, Turkey

Received: 22.06.2018 • **Accepted/Published Online:** 29.11.2018 • **Final Version:** 22.03.2019

Abstract: The basis of encryption techniques is random number generators (RNGs). The application areas of cryptology are increasing in number due to continuously developing technology, so the need for RNGs is increasing rapidly, too. RNGs can be divided into two categories as pseudorandom number generator (PRNGs) and true random number generator (TRNGs). TRNGs are systems that use unpredictable and uncontrollable entropy sources and generate random numbers. During the design of TRNGs, while analog signals belonging to the used entropy sources are being converted to digital data, generally comparators, flip-flops, Schmitt triggers, and ADCs are used. In this study, a computer-controlled new and flexible platform to find the most appropriate system parameters in ADC-based TRNG designs is designed and realized. As a sample application with this new platform, six different TRNGs that use three different outputs of Zhongtang, which is a continuous time chaotic system, as an entropy source are designed. Random number series generated with the six designed TRNGs are put through the NIST800–22 test, which has the internationally highest standards, and they pass all tests. With the help of the new platform designed, ADC-based high-quality TRNGs can be developed fast and also without the need for expertise. The platform has been designed to decide which entropy source and parameter are better by comparing them before complex embedded TRNG designs. In addition, this platform can be used for educational purposes to explain how to work an ADC-based TRNG. That is why it can be utilized as an experiment set in engineering education, as well.

Key words: Chaos, chaotic system, Zhongtang chaotic system, chaotic circuit, true random number generator, statistical randomness test, NIST800-22 test suite

1. Introduction

With current technology at its peak today, communication security has turned out to be even more crucial, both in individual and international terms. Encryption techniques are used to maintain data security in correspondence and telecommunication technologies [1]. The core of encryption techniques is the random number generator (RNG). In accordance with the increase in the number of fields in which cryptology is used due to nonstop evolving technology, the demand for new random number generators increases at a similar pace [2–5].

[*] Correspondence: aakgul@sakarya.edu.tr

RNG outputs are systems that are created by using a virtual or physical source. Random number series consist of independent numbers with no correlation to each other [6–8]. Randomness of random numbers must be proven via statistical tests. RNGs can be divided into two main categories as pseudorandom number generators (PRNGs) and true random number generators (TRNGs) [8,9]. PRNGs are circuits created with a certain algorithm and they are composed of deterministic number series. They only demonstrate randomness during a period and these periods repeat themselves. PRNGs start to create random outputs with a specified seed value [8–11]. That is why the chosen seed value must also be random. When the used algorithm is known, random output values at any moment can be taken as references and in this way the following output values can be calculated. This situation limits the use of PRNGs [8–11]. TRNGs, unlike PRNGs, are systems that make use of unpredictable and uncontrollable entropy (noise) sources and generate random numbers [12–14]. Despite the fact that TRNGs also need hardware and the bit generation process is slow, they are still preferred more for platforms in which high reliability is required because they are unpredictable [15,16]. For the design of TRNGs, direct amplifying and sampling of noise sources, oscillator sampling methods, and chaotic systems are commonly used [17–19]. The randomness of TRNGs generally depends on the entropy source, sampling process, and postprocessing algorithms [17–19]. The entropy (noise) source is the physical randomness source for TRNGs. In recent years, sources like radio frequency (RF) with low amplitude, electronic sensors, and electronic component noises have been used even more and this is the case for chaotic systems as well [20–22].

Chaotic systems can, in short, be described as dynamic systems that are very sensitive to initial conditions and equation parameters [23–28]. Because of the sensitivity of chaotic systems to initial conditions and parameters, even tiny changes in these values can lead to a change in the system output. Due to this fact, even though chaotic structures are deterministic systems, only short-term system behavior can be predicted. In the following iterations, behaviors of chaotic systems become unpredictable [29,30]. Due to these behaviors of chaotic systems, they are more and more commonly employed in engineering and technological applications [31–33]. In parallel to this, the usage of chaotic oscillators as entropy source for TRNGs is becoming more and more widespread. The reason why chaotic systems are preferred in TRNG designs is that signal amplitudes are high enough and they are less influenced by environmental factors, compared to other noise sources [18,19,34].

During the design of TRNGs, while analog signals belonging to the used entropy sources are being converted to digital data (sampling), generally comparators, flip-flops, Schmitt triggers, and ADCs are used. Random bit series generated with TRNGs are put through postprocessing algorithms in order to increase their randomness and reliability [9]. An examination of studies already made reveals that some of the postprocessing algorithms commonly used are the XOR process [9] and Von Neumann [35] and H functions [36]. Examples for true random number generators with ADC are the embeddable ADC-based TRNG for cryptographic applications by Callegari et al. [13] and fast chaos-based TRNG for cryptographic applications by Pareschi et al. [37]. Throughout the design of TRNGs, hardware structures like postprocessing algorithms, ADC source bits, and sampling timing are shaped according to features of the entropy source. When an entropy source is altered, design parameters of the hardware also change. In order to design a good TRNG, the most suitable hardware parameters and postprocessing algorithms for entropy sources are required [16–18]. Prior to starting integrated electronic circuit realization, multiple trials on circuit realization must be done to find the most suitable design parameters. These trials require much effort and time.

Avaroglu et al. developed a chaos-based postprocessing technique as an alternative to postprocessing techniques in the literature [38]. Avaroglu et al. proposed a new hybrid system and subjected it to NIST 800-22 and FIPS statistical tests [39]. Tuncer et al. developed a TRNG-based ring oscillator that can be used in cryptographic applications. The randomness test results generated by the TRNG with nonperiodic sampling were shown to be used in cryptographic systems after passing the NIST 800.22 test [40]. Avaroglu presented a PRNG that generates bit sequences by sampling with two Arnold cats [41]. Özkaynak explained the requirements of a robust random generator and proposed a hybrid architecture [42]. Özkaynak and Yavuz performed safety analysis of a PRNG based on Chen's chaotic system [43]. Tuna et al. examined the performance differences between a traditional TRNG method using a chaotic system and FPGA platform-based chaotic system [44]. In terms of statistical test suites for randomness, several solutions as in [45–48] have been recently proposed to solve some of the pitfalls of NIST SP-800.22. However, in order to make our contribution more comparable with other solutions reported in the literature, we restrain ourselves to the use of tests based on NIST SP-800.22 only.

In this study, a novel computer-controlled platform in which circuit realizations for an ADC-based TRNG can be carried out fast and easily is designed and hardware parameters of the design can be controlled with software. The article consists of 3 main parts. First, the design of a new computer-controlled platform to generate ADC-based true random numbers is explained in four subparts: entropy source, microcontroller controlled data collection card, 0–5 V voltage level converter circuit for ADC, and computer interface program. In Section 3, a computer-controlled platform is used and a sample TRNG is designed. The generated random bit series are put through the NIST800–22 test, which has the internationally highest standards [49], and test results are given in a table. For the sample TRNG, Zhongtang, which is a continuous time chaotic system, is used as an entropy source. The Zhongtang chaotic system is modeled with electronic components and an electronic circuit is designed, and also time series and phase portrait analyses on the Orcad-PSpice program are conducted. Finally, the results of the study are evaluated.

2. A new computer-controlled platform for ADC-based TRNG designs

In the ADC-based TRNG design phase, a new computer-controlled platform is designed and realized in order to find the most suitable design parameters quickly and easily and to realize the TRNG. As can be seen from Figure 1, the designed platform is made up of 4 parts.

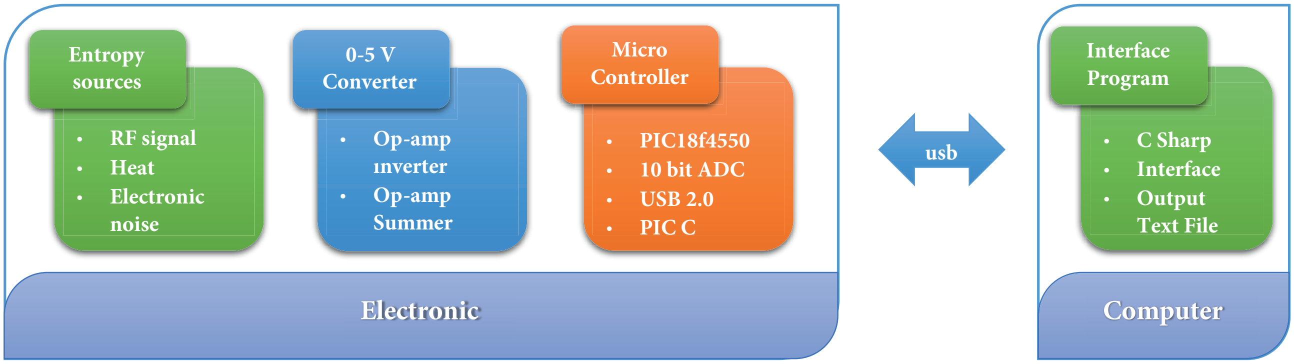


Figure 1. A new computer-controlled platform for designing a TRNG.

2.1. Entropy (noise) sources

In an electronic system, unwanted parasite signals with an unknown source that disrupt the ported signal are called noise. The entropy source is important for randomness of TRNGs. The entropy source is the physical randomness source for TRNG. The main noise sources used in TRNGs are the noise in RF signals; the noise in electronic sensors like heat, pressure, and dampness; and electrical noises from electronic circuit components [20–22].

2.2. Microcontroller-controlled data collection card

A microcontroller-controlled data collection card, whose circuit schema is given in Figure 2, is designed in order to convert analog signals obtained from the entropy source to digital data and to interpret these data and then to send them to the computer, in a certain format, via USB. A PIC18f4550 [50] microcontroller is preferred in the designed circuit as it has enough 10-bit internal ADC channels and USB 2.0 feature.

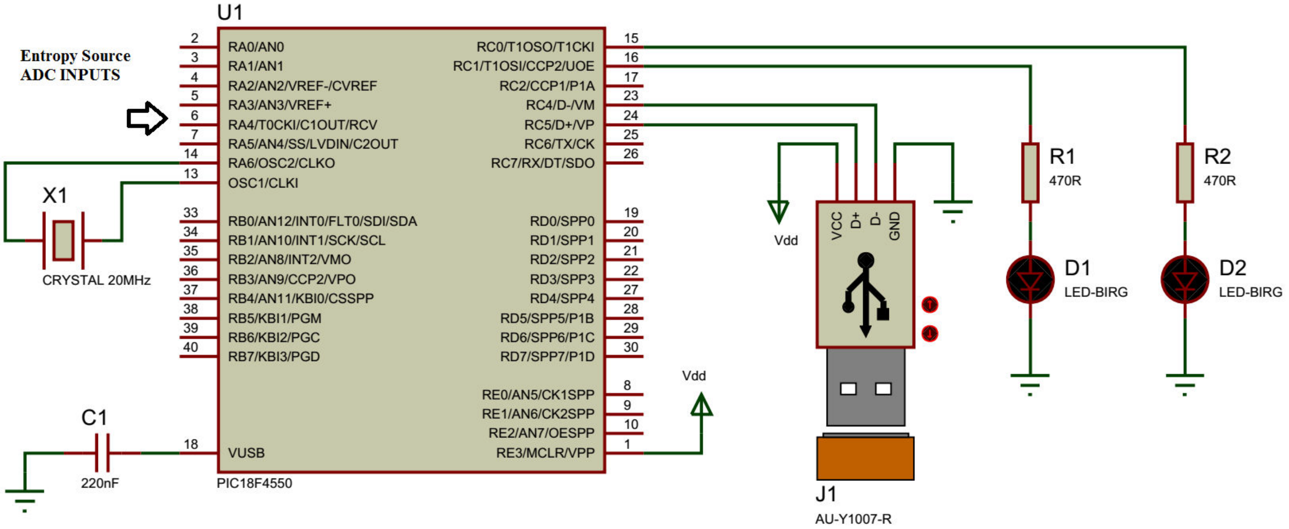


Figure 2. The circuit schema of microcontroller-controlled data collection card.

Using ADC, the data collection card realized here (Figure 3) converts three different signals from the entropy source to 10-bit digital data. The data collection card removes supply voltage from the USB and does not need an external supply voltage. One of the two LEDs on the data collection card shows that connection with the communication is on and the other LED shows there is data flow between the computer and data collection card.

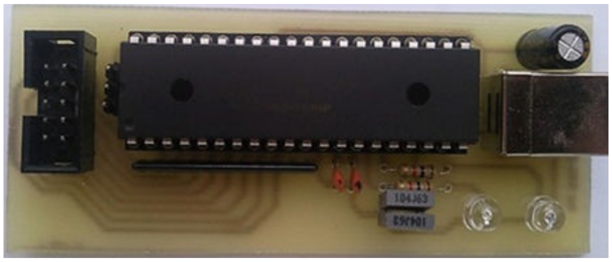


Figure 3. Microcontroller-controlled data collection card.

The software belonging to the PIC18f4550 microcontroller on the data collection card is written with the C programming language. The CCS PIC C COMPILER is used. A state diagram of the written software is given in Figure 4.

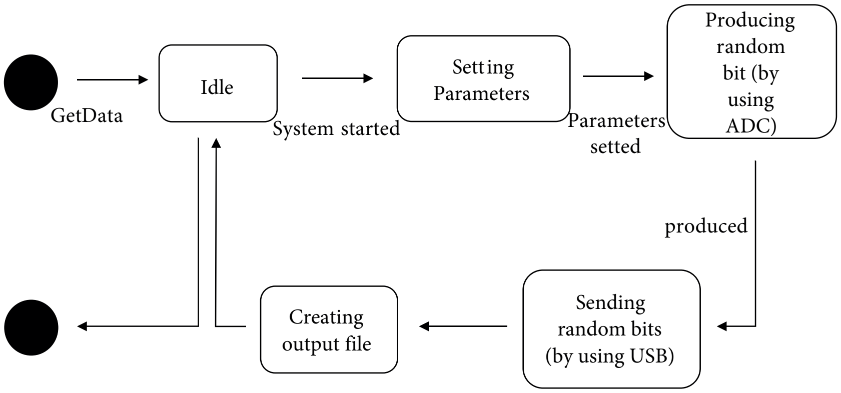


Figure 4. PIC18f4550 state diagram of the microcontroller's software.

As can be observed from the state diagram in Figure 4, when the microcontroller-controlled data collection card is connected to the computer via USB, the data LED on the card vibrates to show that the connection is achieved. Following this, the arrival of a data pack that contains parameters to be used during random bits generation and the "Start" command is expected from the computer program. These parameters are ADC sample bit, sample time, input of entropy, postprocessing algorithm, and number of random bits. When the data pack is received, parameter adjustments are made and random bit generation at a desired number is initiated. The ongoing process of random bit generation can be observed via data LED on the electronic card. The random bits generated are immediately transferred to the computer program in 500-bit packs.

2.3. 0–5 V Voltage level converter circuit for ADC

The signals that belong to chaotic oscillators used as entropy sources for random number generation are not suitable for sampling directly with microcontroller ADC channels because of their high level of peak-to-peak amplitude and negative voltage values. This is why voltage levels of this kind of entropy sources need to be adjusted to fit the 0–5 V measurement range required for ADC.

The converter circuit initially minimizes the signal on its input to 5 V level peak to peak. Then it collects the signal with an offset voltage at the smallest negative value of the minimized signal. In this way, the signal reaches the 0–5 V amplitude level required for the ADC sampling process. To offer a better understanding of how the converter circuit works, a sample chaotic signal applied to the circuit input is given in Figure 5a, the chaotic signal's minimized peak-to-peak to 5 V level is given in Figure 5b, and the chaotic signal conditioned to 0–5 V amplitude level is given in Figure 5c.

During signal reduction and 0–5 V converter processes, in this study, buffer circuits with op-amps are used for converter impedance among circuits. The voltage level converter circuit is designed as three channeled so as to enable signal outputs of different entropy sources to be used at the same time. In the voltage level converter circuit, signals are reduced with RV1 trimpot and offset voltages are adjusted with RV2 trimpot.

Prior to random bit generation, in this study, a program is written with C Sharp programming language in order to make adjustments of the microcontroller on the data collection card and also to convert random bits to text files in a certain format. The computer program and the microcontroller communicate through USB.

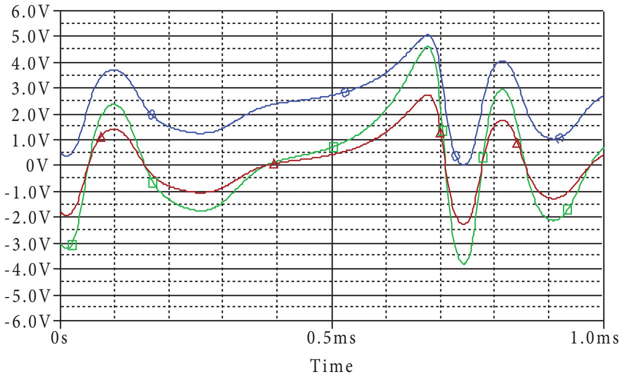


Figure 5. Example chaotic signal (green), reduced example chaotic signal to 5 V voltage level peak to peak (red), adjusted chaotic signal to 0–5 V voltage level (blue).

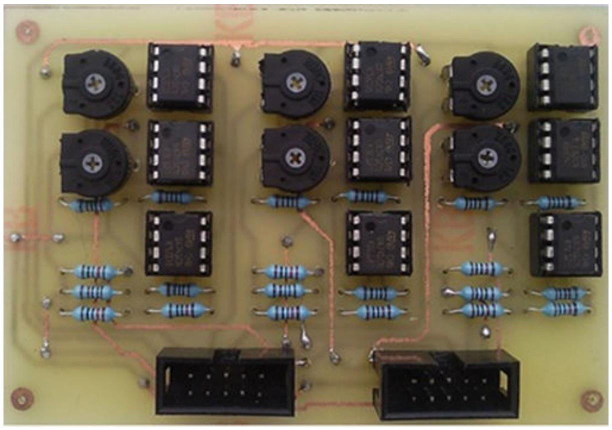
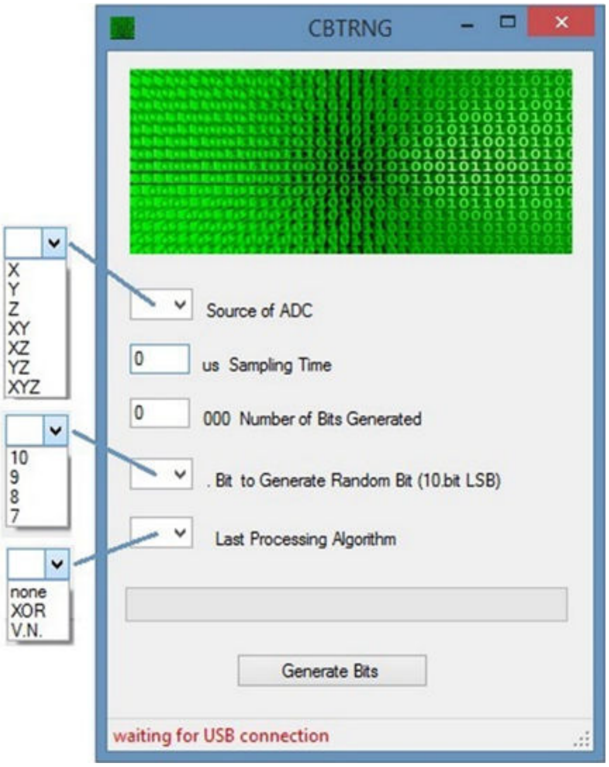


Figure 6. 0–5 V Voltage level converter circuit for ADC.

Figure 7 exhibits the interface of the written computer program.


CBTRNG
X
Y
Z
XY
XZ
YZ
XYZ
Source of ADC
0
us Sampling Time
0
000 Number of Bits Generated
10
9
8
7
. Bit to Generate Random Bit (10.bit LSB)
none
XOR
V.N.
Last Processing Algorithm
Generate Bits

waiting for USB connection

Figure 7. Interface of computer program prepared in C Sharp program.

When the computer program turns on, primarily, it is shown in the window under the interface whether the communication between the electronic card is on. “Source of ADC” of the interface determines which of 3 different ADC channels named X, Y, and Z will be utilized. The choice may occur in 7 different ways like X, Y, Z, XY, XZ, YZ, and XYZ. In XY, XZ, YZ, and XYZ choices, signals are subjected to XOR process. ADC sampling time is adjusted to 1 to 1000 ms with “Sampling Time”. With “Number of Bits Generated”, the number of bits to be generated is made between 1000 and 1,000,000 in 1000 bit resolution. With “Use xth Bit to Generate Random Bit”, it is determined which bit of the signal converted to 10-bit digital data will be used for random bit generation. As a source, the 10th, 9th, 8th, and 7th bits can be chosen, with the 10th bit being the LSB. With the “Postprocessing Algorithm”, it is determined which postprocessing algorithm the generated bits will go through. As postprocessing algorithms, the XOR process [9], Von Neumann [35] algorithms, or no options can be chosen. The program use case is given in Figure 8.

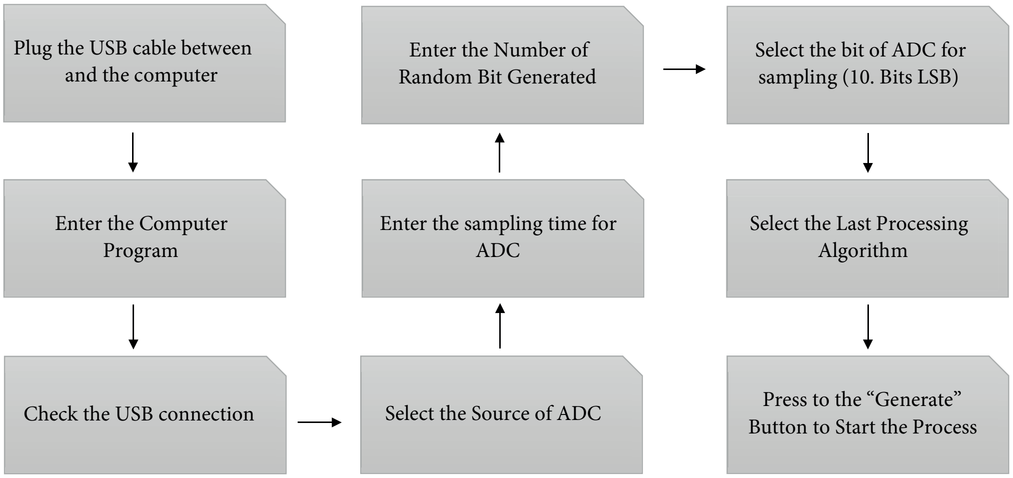

Plug the USB cable between and the computer
Enter the Computer Program
Check the USB connection
Select the Source of ADC
Enter the sampling time for ADC
Enter the Number of Random Bit Generated
Select the bit of ADC for sampling (10. Bits LSB)
Select the Last Processing Algorithm
Press to the “Generate” Button to Start the Process

Figure 8. Use case of computer program.

3. Chaos-based TRNG design

In recent years, the use of chaotic oscillators as entropy sources for TRNGs has become even more common. The reason why chaotic oscillators are preferred in TRNG designs is that signal amplitudes are high enough and they are less influenced by environmental factors compared to other noise sources [13,37]. There is always demand for novel chaotic systems with dynamic and complicated structures. As examples of this kind of continuous time chaotic system, Pehlivan’s four-scroll system [51], another system with golden proportion equilibrium point [52], and a new system of Zhongtang with a dynamic and very complicated structure [53] can be given. For the ADC-based random number generation realized in this study, the Zhongtang chaotic system is employed as an entropy source.

3.1. Zhongtang chaotic system

The differential equation set that belongs to the Zhongtang chaotic system is shown in Eq. (1):

$$\begin{aligned} \dot{x} &= a(y-x), \\ \dot{y} &= b(x+y)-xz^2, \\ \dot{z} &= -ex-cz+x^2. \end{aligned} \tag{1}$$

Typical parameter values for the Zhongtang system are a = 40, b = 10, c = 15, and e = 20 and initial condition values are x(0) = 1 V, y(0) = 0 V, and z(0) = 1 V [53]. Since dynamic limits of the system exceed power source voltage limits of op-amps on the circuit, x, y, and z variables need to be scaled. The scaled Zhongtang chaotic system is shown in Eq. (2):

$$\begin{aligned}\dot{x} &= 80y - 40x,\\ \dot{y} &= 5x + 10y - 2xz^2,\\ \dot{z} &= -10x - 15z + x^2.\end{aligned} \tag{2}$$

With new variables assumed as x, y/2, and z/2, the Zhongtang equations whose circuits will be set will be as exhibited in Eq. (3):

$$\begin{aligned}\underline{\mathrm{x}} &= \tfrac{1}{\mathrm{R_1C_1}}\mathrm{y} - \tfrac{1}{\mathrm{R_2C_1}}\mathrm{x},\\ \underline{\mathrm{y}} &= \tfrac{1}{\mathrm{R_4C_2}}\mathrm{x} + \tfrac{1}{\mathrm{R_3C_2}}\mathrm{y} - \tfrac{1}{\mathrm{R_5C_2}}\mathrm{xz^2},\\ \underline{\mathrm{z}} &= -\tfrac{1}{\mathrm{R_7C_3}}\mathrm{x} - \tfrac{1}{\mathrm{R_6C_3}}\mathrm{z} + \tfrac{1}{\mathrm{R_8C_3}}\mathrm{x^2z}.\end{aligned} \tag{3}$$

Initial conditions of the scaled Zhongtang system are x(0) = 1 V, y(0) =0 V, and z(0) = 1 V. When the Zhongtang chaotic circuits are modeled with electronic components, the differential equation is found as in Eq. (4):

$$\begin{aligned}\underline{\mathrm{x}} &= \tfrac{1}{\mathrm{R_1C_1}}\mathrm{y} - \tfrac{1}{\mathrm{R_2C_1}}\mathrm{x},\\ \underline{\mathrm{y}} &= \tfrac{1}{\mathrm{R_4C_2}}\mathrm{x} + \tfrac{1}{\mathrm{R_3C_2}}\mathrm{y} - \tfrac{1}{\mathrm{R_5C_2}}\mathrm{xz^2},\\ \underline{\mathrm{z}} &= -\tfrac{1}{\mathrm{R_7C_3}}\mathrm{x} - \tfrac{1}{\mathrm{R_6C_3}}\mathrm{z} + \tfrac{1}{\mathrm{R_8C_3}}\mathrm{x^2z}.\end{aligned} \tag{4}$$

In the equations found, capacitor values depend on the timing scale value of the circuit. According to Cuomo and Oppenheim's study [54], the timing scale is 2505. In this study, as well, the timing scale is assumed as β = 2505. Coefficients given in Eq. (5) are equalized and resistance values are calculated. The AD633 multiplier divides multiplication results by 10. Therefore, resistance values found must be divided by 10 in order to reincrease the gain 10 times.

$$\begin{aligned}&\mathrm{R_1} = \tfrac{1}{2505.10^{-9}.80} = 5\mathrm{k} &&\mathrm{R_2} = \tfrac{1}{2505.10^{-9}.40} = 10\mathrm{k}\\ &\mathrm{R_3} = \tfrac{1}{2505.10^{-9}.10} = 40\mathrm{k} &&\mathrm{R_4} = \tfrac{1}{2505.10^{-9}.5} = 80\mathrm{k}\\ &\mathrm{R_5} = \tfrac{1}{2505.10^{-9}.2.10.10} = 2\mathrm{k} &&\mathrm{R_6} = \tfrac{1}{2505.10^{-9}.15} = 26,666\mathrm{k}\\ &\mathrm{R_7} = \tfrac{1}{2505.10^{-9}.10} = 40\mathrm{k} &&\mathrm{R_8} = \tfrac{1}{2505.10^{-9}.10.10} = 4\mathrm{k}\end{aligned} \tag{5}$$

For a = 40, b = 10, c = 15, and e = 20 parameter values, condenser values in the equations found are chosen as 1 nF and resistance values are calculated as R1 = 5 K, R2 = 10 K, R3 = R7 = 40 K, R4 = 80 K, R5 = 2 K, R6 = 26,666 K, R8 = 4 K, and R11 = R12 = R13 = R14 = 100 K. The electronic circuit schema of the scaled Zhongtang system designed with the found resistance and capacity values is exhibited in Figure 9. The electronic circuit realized here is made up of fundamental electronic components such as resistance, op-amp, multiplier, and condenser. For electronic circuit realization, TL081 is used as an op-amp and an AD633 (Analog Devices) is used as a multiplier.

X, Y, and Z time series outputs found as a result of simulation of the modeled scaled Zhongtang system electronic circuit in the Orcad-PSpice program are shown in Figure 10 and X-Y, X-Z, and Y-Z phase portrait outputs are shown in Figure 11.

For electronic circuit realization of the Zhongtang system, a microcontroller-controlled chaotic circuit experiment set, which is designed by Coşkun et al. and can perform quick circuit modeling, is used [55]. Phase portrait oscilloscope outputs of the set real circuit are exhibited in Figure 12.

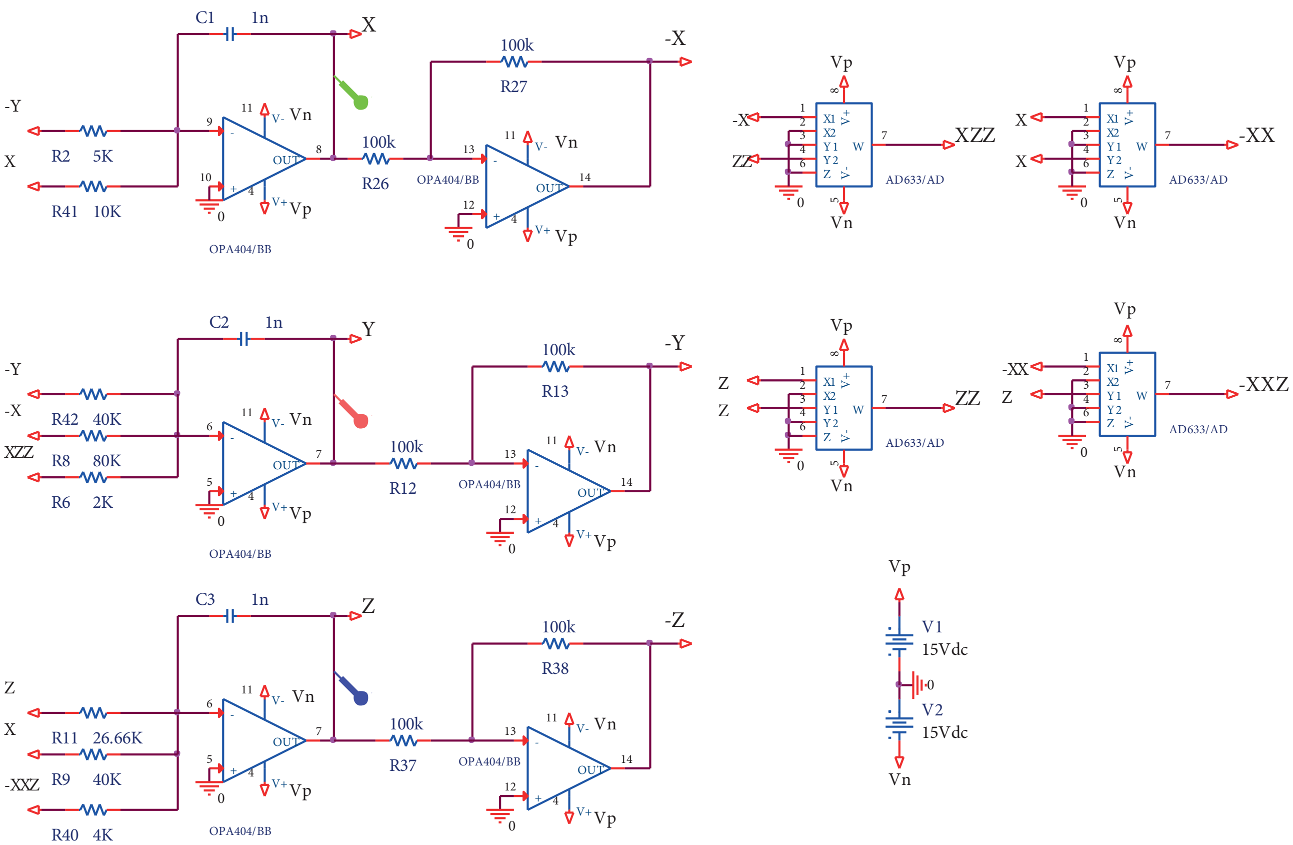


Figure 9. The electronic circuit schema of the scaled Zhongtang system.

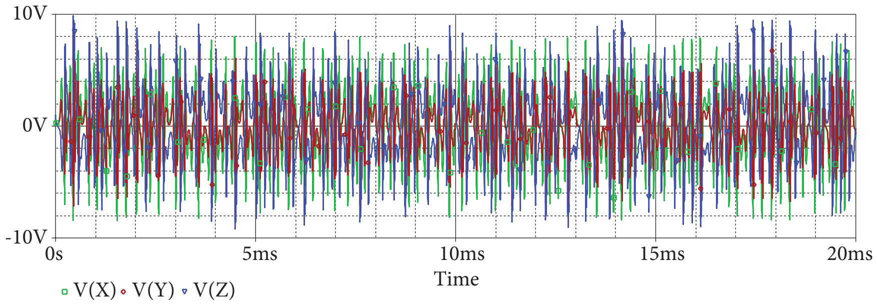


Figure 10. X, Y, and Z time series Orcad-PSpice outputs of the Zhongtang system.

The scaled Zhongtang chaotic system circuit is used as a source for the TRNG in this study. X, Y, and Z signal outputs of the Zhongtang chaotic system located on the chaotic circuit experiment set are conditioned to 0–5 V level for ADC measurement. After that, necessary connections are made and the system becomes ready for random bit generation experiments. For each time in the experiments, 1,000,000 random bits are generated and NIST800-22 tests are run on the random bits generated.

As a result of these experiments, 1,000,000 true random bits are generated 10 times with 1 ms sampling time for X, Y, Z, XY, YZ, and XZ outputs of the Zhongtang chaotic oscillator, with 10th bit (LSB) ADC sampling bit and Von Neumann postprocessing algorithm parameters, and these random bits generated here are put through the NIST800-22 test. NIST800-22 test results of the last generated random bits are given in

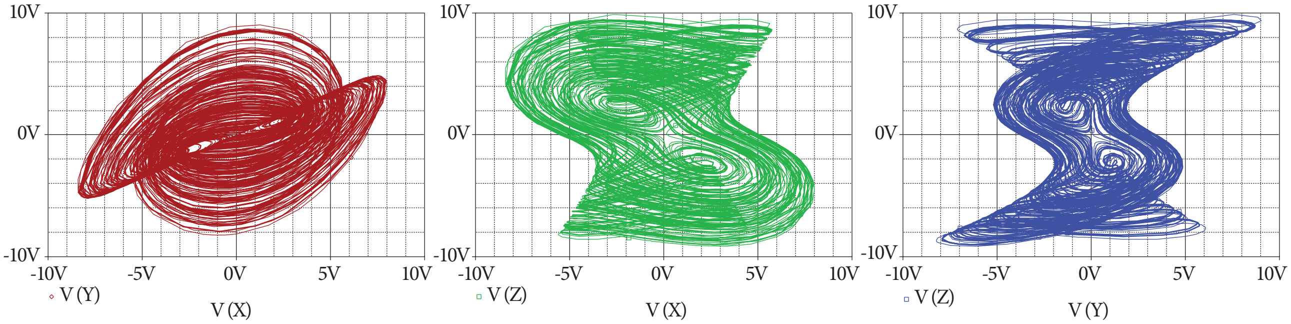


Figure 11. X-Y, X-Z, and Y-Z phase portrait Orcad-PSpice outputs of the Zhongtang system.

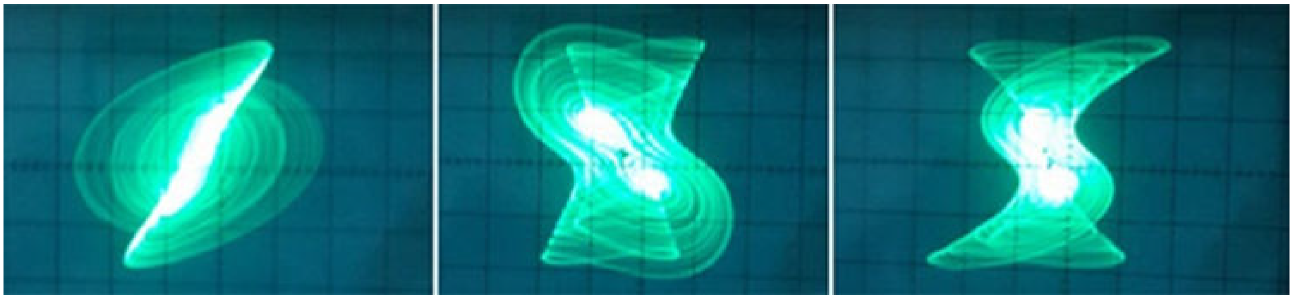


Figure 12. Phase portrait oscilloscope outputs of the Zhongtang system.

Table 1 and Table 2. As can be gathered from Table 1 and Table 2, when all outputs of the Zhongtang chaotic oscillator are used as a source for the TRNG, 10 different random number series can pass all NIST800-22 tests. As the Von Neumann postprocessing algorithm is used, random bit generation speed is not fixed. Average random bit generation speed is 50 kb/s.

Figure 13 exhibits the realization of the TRNG with computer-controlled platform, which uses X and Y outputs of the scaled Zhongtang chaotic system circuit as an entropy source.

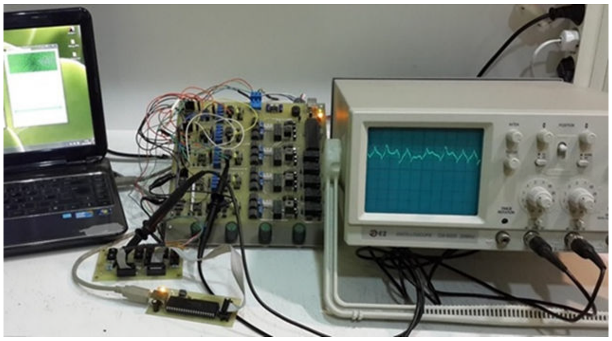


Figure 13. Realization of TRNG with computer-controlled platform.

4. Conclusion

In this article, a novel computer and microcontroller-controlled platform that can be used in ADC-based TRNG designs is developed with the Zhongtang chaotic system. The unique properties of this platform are that it can use, along with a chaotic system, a variety of sources like RF, electronic circuit component noise, and solar

Table 1. NIST800-22 test results of TRNG based on Zhongtang system X, Y, and Z outputs.

Zhongtang system	X		Y		Z	
Statistical tests	P-value	Result	P-value	Result	P-value	Result
Frequency (monobit) test	0.876	Succeed	0.0652	Succeed	0.1325	Succeed
Block-frequency test	0.2479	Succeed	0.116	Succeed	0.821	Succeed
Cumulative-sums test	0.7673	Succeed	0.0823	Succeed	0.1918	Succeed
Runs test	0.261	Succeed	0.2242	Succeed	0.8292	Succeed
Longest-run test	0.0877	Succeed	0.5386	Succeed	0.8752	Succeed
Binary matrix rank test	0.6091	Succeed	0.7622	Succeed	0.2606	Succeed
Discrete Fourier transform test	0.5088	Succeed	0.3084	Succeed	0.0126	Succeed
Nonoverlapping templates test	0.9846	Succeed	0.6136	Succeed	0.8037	Succeed
Overlapping templates test	0.2601	Succeed	0.7477	Succeed	0.8837	Succeed
Maurer's universal statistical test	0.6839	Succeed	0.0565	Succeed	0.5993	Succeed
Approximate entropy test	0.3774	Succeed	0.4431	Succeed	0.3108	Succeed
Random-excursions test	0.7527	Succeed	0.7212	Succeed	0.8133	Succeed
Random-excursions variant test	0.9239	Succeed	0.8595	Succeed	0.9418	Succeed
Serial test-1	0.9641	Succeed	0.4583	Succeed	0.0309	Succeed
Serial test-2	0.9564	Succeed	0.2149	Succeed	0.1156	Succeed
Linear-complexity test	0.8377	Succeed	0.1921	Succeed	0.8175	Succeed
Bit generation speed/success rate	46 kb/s	10/10	50 kb/s	10/10	52 kb/s	10/10

Table 2. NIST800-22 test results of TRNG based on Zhongtang system X-Y, X-Z, and Y-Z outputs.

Zhongtang system	$X \bigoplus Y$		$X \bigoplus Z$		$Y \bigoplus Z$	
Statistical tests	P-value	Result	P-value	Result	P-value	Result
Frequency (monobit) test	0.1169	Succeed	0.6759	Succeed	0.3821	Succeed
Block-frequency test	0.8434	Succeed	0.5102	Succeed	0.311	Succeed
Cumulative-sums test	0.2296	Succeed	0.545	Succeed	0.4079	Succeed
Runs test	0.2315	Succeed	0.7458	Succeed	0.8613	Succeed
Longest-run test	0.3974	Succeed	0.1382	Succeed	0.6539	Succeed
Binary matrix rank test	0.8824	Succeed	0.3395	Succeed	0.5513	Succeed
Discrete Fourier transform test	0.3588	Succeed	0.4408	Succeed	0.8185	Succeed
Nonoverlapping templates test	0.4852	Succeed	0.453	Succeed	0.0224	Succeed
Overlapping templates test	0.7502	Succeed	0.352	Succeed	0.4898	Succeed
Maurer's universal statistical test	0.2907	Succeed	0.0697	Succeed	0.0463	Succeed
Approximate entropy test	0.3548	Succeed	0.2841	Succeed	0.824	Succeed
Random-excursions test	0.8534	Succeed	0.1569	Succeed	0.7559	Succeed
Random-excursions variant test	0.9565	Succeed	0.8589	Succeed	0.8231	Succeed
Serial test-1	0.5019	Succeed	0.3252	Succeed	0.9941	Succeed
Serial test-2	0.3929	Succeed	0.1739	Succeed	0.9405	Succeed
Linear-complexity test	0.3965	Succeed	0.7479	Succeed	0.8335	Succeed
Bit generation speed/success rate	50 kb/s	10/10	51 kb/s	10/10	54 kb/s	10/10

radiations as entropy source; it can mix and use different entropy sources; it can choose different postprocessing algorithms; and it can help create random number series of desired length. TRNG realization is performed with 6 different entropy sources from the Zhongtang chaotic system. In order to increase TRNG reliability, 60 one million bit number series, 10 times from each of 6 different TRNGs, are generated. With the help of the data collection card, 1,000,000 bit number series from the TRNG are transferred to the computer program via USB and saved in a file. Sixty number series that are saved on files are put through NIST800-22 randomness tests, which have the internationally highest standards, and the whole random bit series proves successful in the tests. The Zhongtang chaotic system-based TRNG developed in this study can be safely used in areas such as cryptology applications, communication, defense industry, medicine, and industrial systems where random number series are required. The designed platform makes ADC-based TRNG design easier and also reduces realization time. Thus, it can be utilized as an experiment set in engineering education, as well.

Acknowledgment

We would like to express our appreciation to the anonymous reviewers for their insightful comments, which have greatly improved the quality of the paper.

References

[1] Menezes AJ, Paul C, Van O, Scott AV. Handbook of Applied Cryptography. Boca Raton, FL, USA: CRC Press, 1996.

[2] Koç CK. Cryptographic Engineering. Boston, MA, USA: Springer, 2009.

[3] Zhao L, Liao X, Xiao D, Xiang T, Zhou Q, Duan S. TRNG from mobile telephone photo based on chaotic cryptography. Chaos, Solitons & Fractals 2009; 42: 1692-1699.

[4] Cavusoglu U, Akgul A, Kacar S, Pehlivan I, Zengin A. A novel chaos-based encryption algorithm over TCP data packet for secure communication. Security and Communicatıon Networks 2016; 9: 1285-1296.

[5] Hellman ME. An overview of public key cryptography. IEEE Communications 2002; 16: 42-49.

[6] Ergün S, Özoğuz S. TRNGs based on a non-autonomous chaotic oscillator. International Journal of Electronics and Communications 2007; 61: 235-242.

[7] Angulo JAA, Kussenar E, Barthelemy H, Duval B. A new oscillator-based RNG. IEEE Faible Tension Faible Consommation 2012; 1-4.

[8] Kocarev L, Jakimoski G. Pseudorandom bits generated by chaotic maps. IEEE Transactions on Circuits and Systems I 2003; 50: 123-126.

[9] Avaroğlu E, Türk M. Son işlemin gerçek rasgele sayı üreteçleri üzerindeki etkisinin incelenmesi. In: 6th International Information Security and Cryptology Conference, ISCTURKEY 2013. pp. 290-294 (in Turkish).

[10] Merah L, Ali A, Said N, Mamat M. A pseudo random number generator based on the chaotic system of Chua's circuit, and its real time FPGA implementation. Applied Mathematical Sciences 2013; 7: 2719-2734.

[11] Zeng K, Yang C, Wei D, Rao TRN. Pseudorandom bit generators in stream-cipher cryptography. Computer 1991; 24: 8-17.

[12] Akram R, Konstantinos M, Keith M. Pseudorandom number generation in smart cards: an implementation, performance and randomness analysis. In: 2012 5th International Conference on NTMS, 2012. pp. 1-7.

[13] Callegari S, Rovatti R, Setti G. Embeddable ADC-based true random number generator for cryptographic applications exploiting nonlinear signal processing and chaos. IEEE Transactions on Signal Processing 2005; 53: 793-805.

[14] Bucci M, Germani L, Luzzi R, Tommasino P, Trifiletti A, Varanonuovo M. A high-speed IC random-number source for smart card microcontrollers. IEEE Transactions on Circuits and Systems I 2003; 50: 1373-1380.

[15] Sobotka J, Zeman V. Design of the true random numbers generator. Elektrorevue 2011; 2: 1-6.

[16] Murphy JP. Field-programmable true random number generator. Electronics Letters 2012; 48: 565-566.

[17] Wang L, Meilin W, Kui D, Xuecheng Z. Scalable truly random number generator. In: Proceedings of the World Congress on Engineering 2015. p. 1.

[18] Yalçın M, Suyken J, Vandewalle J. True random bit generation from a double scroll attractor. IEEE Transactions on Circuits and Systems 2004; 51: 1395-1404.

[19] Yıldırım S, Bazlaccı C. A true random number generator and test platform built in FPGA. In: International Information Security and Cryptology Conference, ISCTURKEY 2012, pp. 262-267.

[20] Petrie CS, Connelly JA. A noise-based IC RNG for applications in cryptography. IEEE Transactions on Circuits and Systems I 2000; 47: 615-621.

[21] Holman WT, Connelly JA, Dowlatabadi AB. An integrated analog/digital random noise source. IEEE Transactions on Circuits and Systems I 1997; 44: 521-528.

[22] Zhun H, Hongyi C. A truly random number generator based on thermal noise. In: IEEE Proceedings of 4th International Conference on ASIC, 2001. pp. 862-864.

[23] Li C, Pehlivan I, Sproot JC, Akgul A. A novel four-wing strange attractor born in bistablity. IEICE Electronics Express 2015; 12: 1-12.

[24] Wei Z, Pehlivan I. Chaos, coexisting attractors, and circuit design of the generalized Sprott C system with only two stable equilibria. Optoelectronics and Advanced Materials–Rapid Communications 2012; 6: 742–745.

[25] Li C, Pehlivan I, Sproot JC. Amplitude-phase control of a novel chaotic attractor. Turkish Journal of Electrical Engineering & Computer Sciences 2016; 1: 1-11.

[26] Akgul A, Calgan H, Koyuncu I, Pehlivan I, Istanbullu A. Chaos-based engineering applications with a 3D chaotic system without equilibrium points. Nonlinear Dynamics 2016; 84: 481-495.

[27] Akgul A, Shafqat H, Pehlivan I. A new three-dimensional chaotic system, its dynamical analysis and electronic circuit applications. International Journal for Light and Electron Optics 2016; 127: 7062–7071.

[28] Akgul A, Pehlivan I. A new three-dimensional chaotic system without equilibrium points, its dynamical analyses and electronic circuit application. Technical Gazette 2016; 23: 209-214.

[29] Addison PS. Fractals and Chaos. An Illustrated Course. London, UK: IOP Publishing Limited, 1997.

[30] Hilborn RC. Chaos and Nonlinear Dynamics. An Introduction for Scientists and Engineers. Oxford, UK: Oxford University Press, 1994.

[31] Koyuncu I, Ozcerit AT, Pehlivan I. An analog circuit design and FPGA-based implementation of the Burke-Shaw chaotic system. Optoelectronics and Advanced Materials–Rapid Communications 2013; 7: 635–638.

[32] Akgul A, Moroz I, Vaidyanathan S, Pehlivan I. A new four-scroll chaotic attractor and its engineering applications. OPTIK 2016; 13: 5491-5499.

[33] Pehlivan I. Yeni kaotik sistemler: elektronik devre gerçeklemeleri, senkronizasyon ve güvenli haberleşme uygulamaları. PhD, Sakarya University, Sakarya, Turkey, 2007 (in Turkish).

[34] Stojanovski T, Kocarev L. Chaos-based random number generators-part I: analysis. IEEE Transactions on Circuits and Systems 2001; 48: 281-288.

[35] Von Neumann J. Various techniques used in connection with random digits. Applied Math Series, Notes by G. E. Forsythe, in National Bureau of Standards 1951; 12: 6-38.

[36] Dichtl M. Bad and good ways of post-processing biased physical random numbers. Fast Software Encryption Lecture Notes in Computer Science 2007; 4593: 137-152.

[37] Pareshi F, Setti G, Rovatti R. A fast chaos-based true random number generator for cryptographic applications. In: ESSCIRC 2006, Proceedings of the 32nd European Solid-State Circuits Conference, 2006. pp. 130-133.

[38] Avaroğlu E, Tuncer T, Özer AB, Ergen B, Türk M. A novel chaos-based post-processing for TRNG. Nonlinear Dynamics 2015; 81: 189-199.

[39] Avaroğlu E, Koyuncu İ, Özer AB, Türk M. Hybrid pseudo-random number generator for cryptographic systems. Nonlinear Dynamics 2015; 82: 239-248.

[40] Tuncer T, Avaroglu E, Türk M, Ozer AB. Implementation of non-periodic sampling true random number generator on FPGA. Informacije MIDEM 2015; 44: 296-302.

[41] Avaroğlu E. Pseudorandom number generator based on Arnold cat map and statistical analysis. Turkish Journal of Electrical Engineering & Computer Sciences 2017; 25: 633-643.

[42] Özkaynak F. Cryptographically secure random number generator with chaotic additional input. Nonlinear Dynamics 2014; 78: 2015-2020.

[43] Özkaynak F, Yavuz S. Security problems for a pseudorandom sequence generator based on the Chen chaotic system. Computer Physics Communications 2013; 184: 2178-2181.

[44] Tuna M, Fidan CB. A Study on the importance of chaotic oscillators based on FPGA for true random number generating (TRNG) and chaotic systems. Journal of the Faculty of Engineering and Architecture of Gazi University 2018; 33: 469-486.

[45] L'Ecuyer P, Simard R. TestU01: AC library for empirical testing of random number generators. ACM Transactions on Mathematical Software 2007; 33: 22.

[46] Renyi A. On the theory of order statistics. Acta Mathematica Academiae Scientiarum Hungaricae 1953; 4: 191.

[47] Pareschi F, Rovatti R, Setti G. Second-level NIST randomness tests for improving test reliability. In: IEEE International Symposium on Circuits and Systems, New Orleans, LA, USA, 2007. pp. 1437-1440.

[48] Pareschi F, Rovatti R, Setti G. On statistical tests for randomness included in the NIST SP800-22 test suite and based on the binomial distribution. IEEE Transactions on Information Forensics and Security 2012; 7: 491-505.

[49] Rukhin A, Soto J, Nechvatal J, Smid M, Barker EA. Statistical test suite for random and pseudo RNGs for cryptographic applications. National Institute of Standards and Technology, Booz-Allen and Hamilton Inc., McLean, VA, USA 2001.

[50] Sarkar D, Chowdhury A. Low cost and efficient ECG measurement system using PIC18F4550 microcontroller. In: Electronic Design, Computer Networks and Automated Verification (IEEE EDCAV), 2015. pp. 6-11.

[51] Pehlivan I. Four-scroll stellate new chaotic system. Optoelectronics and Advanced Materials-Rapid Communications 2011; 5: 1003-1006.

[52] Pehlivan I, Uyaroğlu Y. A new 3D chaotic system with golden proportion equilibria: analysis and electronic circuit realization. Computers & Electrical Engineering 2012; 38: 1777-1784.

[53] Zhongtang W, Wang M, Jianxiu J, Jiuchap F. A novel strange attractor and its dynamic analysis. Journal of Multimedia 2014; 9: 408-415.

[54] Cuomo KM, Oppenheim AV. Circuit implementation of synchronized chaos with applications to communication. Physical Review Letters 1997; 71: 65-68.

[55] Coşkun S, Tuncel S, Pehlivan I, Akgül A. Microcontroller-controlled electronic circuit for fast modelling of chaotic circuits. Electronics World 2015; 121: 24-25.