Analog Integrated Circuits and Signal Processing
https://doi.org/10.1007/s10470-019-01568-x

Design, FPGA implementation and statistical analysis of chaos-ring based dual entropy core true random number generator

İsmail Koyuncu[1] · Murat Tuna[2] · İhsan Pehlivan[3] · Can Bülent Fidan[4] · Murat Alçın[5]

Received: 3 August 2019 / Revised: 19 October 2019 / Accepted: 4 December 2019

Abstract
In this paper, a novel chaos-ring based dual entropy core TRNG architecture on FPGA with high operating frequency and high throughput has been performed and presented. The design of dual entropy core TRNG has been generated by uniting the chaotic system-based RNG and the RO-based RNG structures on FPGA. The chaotic oscillator structure as the basic entropy source has been implemented in VHDL using Euler numerical algorithm in 32-bit IQ-Math fixed point number standart on FPGA. The designed chaotic oscillator has been synthesized for the FPGA chip and the statistics related to chip resource consumption and clock frequencies of the units have been presented. The RO-based RNG structure has been designed as the second entropy source. Chaos-ring based dual entropy core novel TRNG unit have been created by combining of these two FPGA-based structures in the XOR function used at the post processing unit. The throughput of the designed dual entropy core TRNG unit ranges 464 Mbps. The output bit streams obtained from FPGA-based novel TRNG have been subjected to NIST 800-22 test suites.

Keywords Chaos · Chaotic systems · FPGA · Ring oscillator · TRNG · Statistical tests

1 Introduction

Nowadays, the rapid developments of technologies used in digital communications especially recent advances in the internet and the smart phones have gained much interest of many research groups worldwide [1]. These advances enable not only people making great changes in their life styles but also attackers trying to access the private and secret data. The various safety requirements for these systems must be considered. These requirements are data protection, user privacy, minimum resource demand and fast processing speed [2]. One of the solutions to protect the data is the cryptography [3, 4]. The cryptography has been used to transmit securely and fastly the confidential data, particularly in the last century [5]. In addition, cryptography is interested in encryption and decryption of the data [6]. As the data encryption is the process of converting the plain text to incomprehensible form (encrypted text) using a secret key, the data decryption is the exact opposite of the data encryption [7]. Since many computational science applications need huge-quantity of high-quality Random Numbers (RNs) [8] and the generation of these RNs is provided by Random Number Generator

✉ Murat Alçın
muratalcin@aku.edu.tr

İsmail Koyuncu
ismailkoyuncu@aku.edu.tr

Murat Tuna
murat.tuna@klu.edu.tr

İhsan Pehlivan
ipehlivan@sakarya.edu.tr

Can Bülent Fidan
cbfidan@karabuk.edu.tr

[1] Department of Electrical and Electronics Engineering, Technology Faculty, Afyon Kocatepe University, Afyon, Turkey

[2] Department of Electric, Technical Sciences Vocational School, Kirklareli University, Kirklareli, Turkey

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

[4] Department of Mechatronics Engineering, Engineering Faculty, Karabuk University, Karabuk, Turkey

[5] Department of Mechatronics Engineering, Technology Faculty, Afyon Kocatepe University, Afyon, Turkey

Published online: 14 December 2019

(RNG), the last advances in chaotic encryption [9] caused a strong connection between the chaos-based cryptography [10] and the chaos-based RNs [11–13]. The quality of random numbers used in cryptography is of great importance for the reliability and power of the system in which it is used. Therefore, the use of random numbers in cryptography, unlike other fields, must meet some strict requirements regarding the security of the system.

Chaotic systems exhibit complex and aperiodic behavior, sensitive dependence on initial conditions and emerge in deterministic nonlinear systems [14, 15]. Important studies have been performed for investigating and applying these concepts in scientific and industrial areas [16, 17]. Image encryption [18, 19], secure signal masking [20], chaos-based composing secure communication mechanisms [21], chaos control [22] and chaos-based synchronization [23, 24], noise generators [25] and RNGs [26] are among the important investigation and application areas of chaotic systems in electronic engineering [27, 28]. Heavy research efforts in the development of chaos-based True RNG (TRNG) structures have been consumed due to the sensitive dependence on initial conditions, exhibiting aperiodic behaviour, encrypting the data signal and having noise-like behaviour of chaotic oscillators [29, 30]. TRNGs have been used in many areas including cryptography, the applications used in Monte-Carlo method, numerical analysis applications, modeling and simulations of computer. Physical events that are not deterministic have been used as entropy source in TRNGs. Since this structure gives more reliable source, it is offen used in cryptography for initial vector, generation of private and secure key and in Pseudo RNGs (PRNGs) for the production of seed. Although TRNGs have disadvantages like low throughput, being expensive and hardware dependence, they have been preferred and utilized in applications like secure communication and cryptology that need high security.

TRNGs are divided into three main categories, namely Noise-based TRNGs, Free Ring Oscillator (RO)-based TRNGs and chaos-based TRNGs [31]. Various methods for many designs and applications of TRNG have been presented in literature. Avaroğlu et al. [12] have designed TRNG unit to use it as an additional input for the Hybrid PRNG. This design includes Sprott 94 G chaotic system coded in VHDL (**V**ery High-Speed Integrated Circuit **H**ardware **D**escription **L**anguage) with 32-bit IEEE 754-1985 floating point standard using RK5-Butcher numerical algorithm on FPGA. The designed system has passed the NIST test suites and has a maximum operating frequency of 339 MHz. Çiçek et al. [32] proposed the implementation of the TRNG structure using discrete time chaos-based new design method with Complementary Metal–Oxide–Semiconductor (CMOS) technology. Koyuncu et al. [33] proposed the design of high speed chaos-based TRNG on FPGA using the chaotic system that is developed by them. The design achieves operating frequency of 293 MHz and the throughput of 58.76 Mbps and passes full tests of the NIST 800-22 and FIPS 140-1. Tavas et al. [34] have accomplished the design of chaos-based TRNG with the production of Integrated Circuit (IC) using 0.35 μm CMOS process of new chaos generator that is compatible for integration and can be used in RNG circuit. The average throughput of the produced TRNG circuit has been given as 2 Mbps. Ergün et al. [35] proposed the design of TRNG structure with technology using non-autonomous chaotic system. The proposed design achieves operating frequency of 1.24 MHz and the throughput of 10 Mbps. Sunar et al. [36] have accomplished the design of RO-based new TRNG on FPGA in 2007 and stated that the produced bit streams are random. Schellekens et al. [37] have successfully designed the multiple RO-based TRNG on FPGA. This structure passed full tests of the NIST and features a 2.5 Mbps throughput. In recent years, the technical properties including implementation method, the applied test suite, operating frequency, throughput and the used method in TRNG structures that designed with different schemes have been given in detail in Table 1. It has been observed that the technical superiorities of TRNGs designed with FPGA-based chaotic oscillators on digital circuit over others have been arisen with respect to investigations and comparisons in Table 1. These designs significantly differ with respect to entropy sources and production methods. Each design has not only strong sides but also weak sides. While the operating frequencies of the analog CMOS-based designs vary from 1 MHz to 25 MHz, even if the throughputs achieve at rates of up to 40 Mbps, they remain low with respect to the TRNG systems generated with hardware. While the operating frequencies of TRNG designs realized with FPGA-based classical oscillators produces bit rate of up to 20–300 MHz, the throughputs decrease down to 1–40 Mbps levels due to post processing algorithm applied to weak outputs because of the used oscillators and methods. It has been observed that the TRNGs realized with FPGA-based chaotic systems provide both higher operating frequencies up to 400 MHz levels and throughputs. However, there has been a decrease on throughputs to 50–200 Mbps levels because of the post processing applied for the qualification on the international tests. Post-processing is used to improve the statistical distribution of the bit stream by sacrificing the throughput. After the post processing, it is possible to state that the bit stream received from the output of TRNG is random or not.

In this paper, in the first step, suggested chaotic oscillator has been modeled in VHDL using Euler numerical algorithm in 32-bit IQ-Math Fixed Point Number (FPN) Standart on FPGA. In the second step, the RO-based RNG design has been performed using VHDL on FPGA. The

Table 1 The technical properties of TRNG structures that designed with different schemes in recent years

Study in literature	Used method	Design	Statistical tests	Operating frequency (MHz)	Throughput (Mbps)
Avaroğlu [11, 12]	Chaotic oscillator	FPGA	NIST 800-22	339	–
Çiçek et al. [32]	Logistic chaotic oscillator	CMOS	NIST 800-22	–	–
Koyuncu et al. [33]	Chaotic oscillator	FPGA	NIST 800-22	293	58.76
Tavas et al. [34]	Oscillator sampling	CMOS	FIPS 140-1		2
Ergün et al. [35]	Oscillator sampling	CMOS	NIST 800-22	1.24	10
Sunar et al. [36]	Ring oscillator	FPGA	NIST 800-22	–	2.5
Schellekens et al. [37]	Multiple ring oscillator	FPGA	NIST 800-22	40	2.5
Ning et al. [38]	Ring oscillator	CMOS	DieHard	20	10–20
Park et al. [39]	Boolean chaotic oscillator	CMOS	NIST 800-22	300	–
Çiçek et al. [40]	Bernoulli chaotic oscillator	FPGA	NIST 800-22	2	1.5
Wieczorek et al. [41]	Bistable Flip/Flop	FPGA	NIST 800-22	50	5
Wold et al. [42]	Ring oscillator	FPGA	NIST 800-22	–	100
Tuncer et al. [43]	Ring oscillator	FPGA	NIST 800-22	450	25
Wang, et al. [44]	Chaotic oscillator	FPGA	NIST 800-22	–	240
Fatemi, et al. [45]	Chaotic oscillator	FPGA	NIST 800-22	–	–
Proposed method	Chaos-ring based dual entropy core novel TRNG	FPGA	NIST 800-22	464	464

statistics of FPGA chip resource consumption with the parameters related to units' clock frequencies of the design of the FPGA-based RO-chaotic oscillator has been investigated. In the third step, a novel chaos-ring based dual entropy core TRNG design has been implemented by uniting the chaotic-based RNG and the RO-based RNG in the XOR function used at the post processing unit on FPGA. The developed model has been coded using VHDL. The statistics related to chip resource consumption and clock frequencies of the designed TRNG unit has been investigated. In the last step, the randomness tests of the random bit stream obtained from chaos-ring based TRNG unit has been realized. To perform this task, NIST 800-22 test suite has been used. 1 Mbit bit streams obtained from TRNG unit have been subjected to NIST 800-22 test suites and random bits generated by suggested design successfully passed all of the tests. Besides, the suggested design has better performance in terms of throughput than the similar TRNG structures in the literature. The successful results have shown that the proposed chaos-ring based dual entropy core TRNG can be used in the cryptographic and secure communication systems that need high operating frequency and throughput.

2 Method

2.1 The mathematical model of chaotic P3DS

Pehlivan et al. have presented a new 3D continuous time autonomous chaotic oscillator in their study [46]. The mathematical model of the chaotic P3DS proposed for chaos-based engineering applications has been given in Eq. 1. Here x, y, z are the state variables of chaotic systems and $a = 0.5$, $b = 1.0$ are the system parameters. Also the initial conditions of chaotic system have been defined as $x_0 = 0$, $y_0 = 0$, $z_0 = 0$.

$$\begin{aligned} \dot{x} &= y - x - a \cdot z \\ \dot{y} &= x \cdot z - x \\ \dot{z} &= -x \cdot y - y + b \end{aligned} \tag{1}$$

The chaotic P3DS oscillator has 8 terms which of 2 are nonlinear terms with second degree (xz and xy) and 2 parameters. The chaotic time series obtained from different numerical algorithms, 2D and 3D phase portraits and Lyapunov exponents obtained from Lyapunov Exponent Toolbox (LET) have been given in Fig. 1. Since the sign of Lyapunov exponents (λ_1, λ_2, λ_3) are (+, 0, −), the system exhibits chaotic behavior.

2.2 Discrete model of chaotic system

In this part, chaotic system which has not been performed any FPGA-based study, has been modeled with VHDL

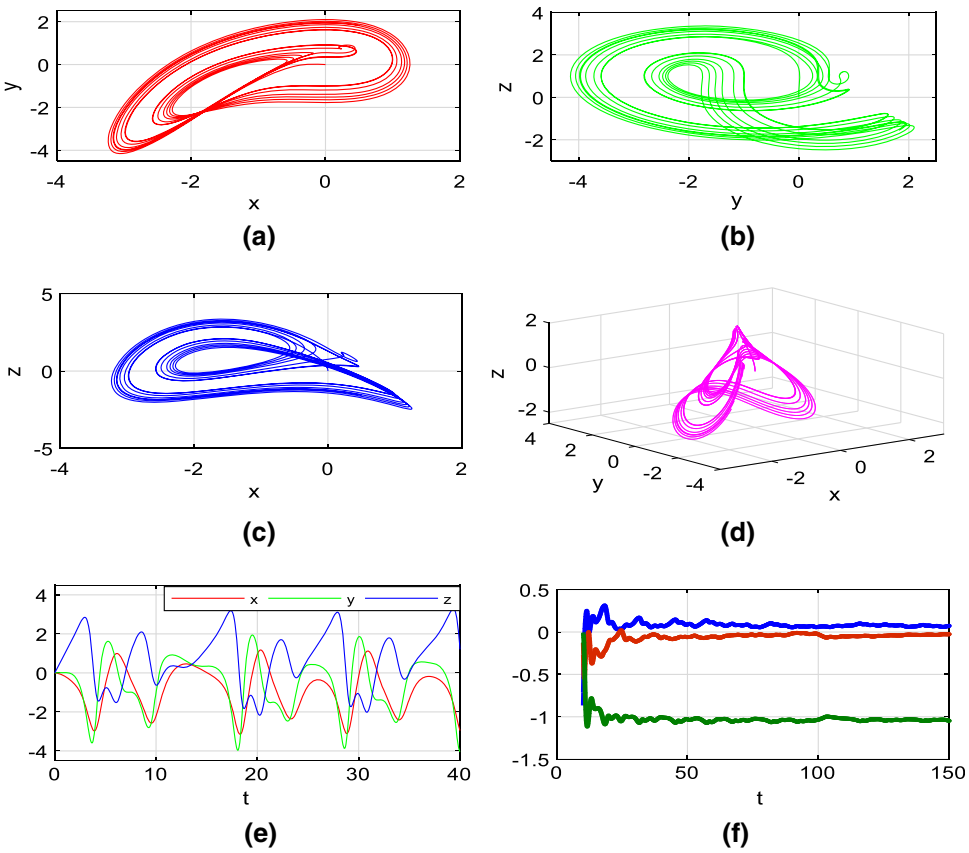

Fig. 1 **a** x–y, **b** y–z, **c** x–z 2D phase portraits, **d** x–y-z 3D phase portraits, **e** x–y–z chaotic time series obtained from different numerical algorithms, **f** Lyapunov exponents obtained from LET of the chaotic system

using Euler numerical algorithm in 32-bit IQ-Math FPN standart on FPGA. The sensitivity analyses of the used algorithm have been carried out by investigating the performance and chip statistics obtained from the implementation of the design on FPGA. First of all, the whole processes have been done with discrete model of Euler numerical algorithm that will be used for the chaotic system. The initial conditions have been taken as $x(k)$= 0.0, $y(k)$= 0.0 and $z(k)$= 0.0. The discrete mathematical model of chaotic P3DS using Euler algorithm can be shown as:

$$\begin{aligned} x(k+1) &= x(k) + (y(k) - x(k) - a \cdot z(k)) \cdot \Delta h \\ y(k+1) &= y(k) + (x(k) \cdot z(k) - x(k)) \cdot \Delta h \\ z(k+1) &= z(k) + (-x(k) \cdot y(k) - y(k) + b) \cdot \Delta h \end{aligned} \tag{2}$$

$x(k+1)$, $y(k+1)$ and $z(k+1)$ values, which are the values of the chaotic system when the step size, Δh increases, have been calculated by replacing these coefficients in the Euler algorithm. The system outputs for each iteration, namely, $x(k+1)$, $y(k+1)$ and $z(k+1)$ values have been used as not only the outputs of the system but also the initial conditions of the algorithm for the next iteration.

2.3 The implementation of chaotic system on FPGA

In this part, the reference chaotic system has been implemented on FPGA and has been coded in VHDL. The units like *subtractor, multiplier* and *adder* which have been used in the production of the designs of units and which are compatible with FPN standard, have been developed using IP-CORE Generator that generated by Xilinx ISE Design Tools. The top level block diagram of the FPGA-based chaotic oscillator unit has been illustrated in Fig. 2.

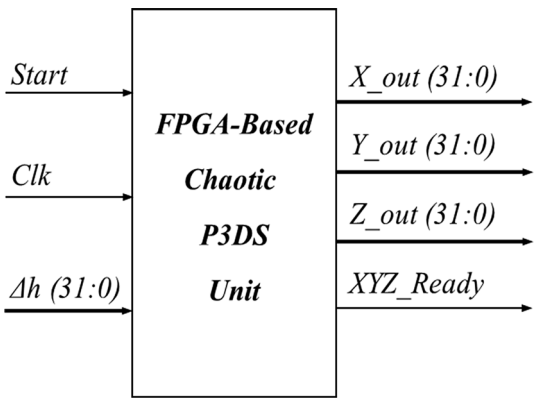

Fig. 2 The top level block diagram of FPGA-based chaotic P3DS unit

One bit signals, namely, *Start* and *Clk* have been presented at the inputs of the unit. These signals have an obligation for timing of whole units and synchronizing these units and their connected system. Δh, which specifies the precision of the algorithm, indicates the parameter of the step size. This signal has been implemented from outside to give flexibility for the design. The initial values, which are essential for the system startup, are embedded into the design for the distension of FPGA chip resource utilization. But if it is required to use these signals, it is convenient to implement the system to set the initial values defining 3 signals each having 32-bit by using small modifications. In the implemented FPGA-based chaotic units, there are three 32-bit output signals (*X_out, Y_out* and *Z_out*) each have convenience with fixed point number standard and one bit *XYZ_Ready* signal to indicate if the output signals are ready.

2.4 The test results of FPGA-based chaotic oscillator

Euler-based chaotic oscillator unit has been synthesized for the *Xilinx Virtex-6* (*xc6vlx75t*) FPGA chip. The area utilization report of the FPGA and the statistical parameters related to clock frequencies of the units have been investigated. The data processing time of unit designed on Euler-based structure has been obtained using Xilinx ISE Design Tools 14.2. Here, the output signals namely, *X_out*, *Y_out* and *Z_out* obtained from the implementation of chaotic system on FPGA have been presented in 32-bit IQ-Math FPN format using ISE Design Tools. The results obtained from Xilinx ISE Simulator have been illustrated in Fig. 3.

At the end of the Place&Route process which is performed after synthesizing of chaotic oscillator unit, *Xilinx Virtex-6* (*xc6vlx75t*) FPGA chip statistics have been obtained and given in Table 2. As can be seen from the chip statistics, maximum operating frequency of the chaotic oscillator is 464.688 MHz. Besides, the binary values in 32-bit IQ-Math format related to time series of *X_out*, *Y_out* and *Z_out* signals obtained from the chaotic oscillator on FPGA have been saved into a file during simulation test stage. After the conversion of the saved values to real number system, the time series and phase portraits of *X_out*, *Y_out* and *Z_out* signals have been obtained using the first 3×5000 data set produced by chaotic oscillator. For example, time series of Euler-based chaotic oscillator implemented on Matlab and implemented on FPGA have been presented and compared in Fig. 4(a and b), respectively. It is observed that there is a good convergence between two figures.

Table 2 Xilinx Virtex-6 chip statistics of chaotic system designed on FPGA

FPGA-based design	Euler-based chaotic system Used/utilization %
Number of Slice Regs.	1196/0
Number of LUTs	1070/0
Number of Bonded IOBs	99/13
Max. Clock Frequency (MHz)	464.688

2.5 FPGA-based RO design and test results

RO structure has been used as a randomness source by most of IC applications and FPGA-based TRNGs. ROs are independent oscillators that include odd number invertor. Each invertor in the ring produces rising edge and falling edge of clock signal generated in two half period, respectively. The delay instability of inverters connected to inside of the ring seems as frequency/phase instability (jitter) of produced clock. Jitter is discarded using flip/flop or latch in the sampling unit. Because the jitter that produced inside of the signal has been used directly as a randomness source. In this section the RO design using VHDL on FPGA has been implemented.

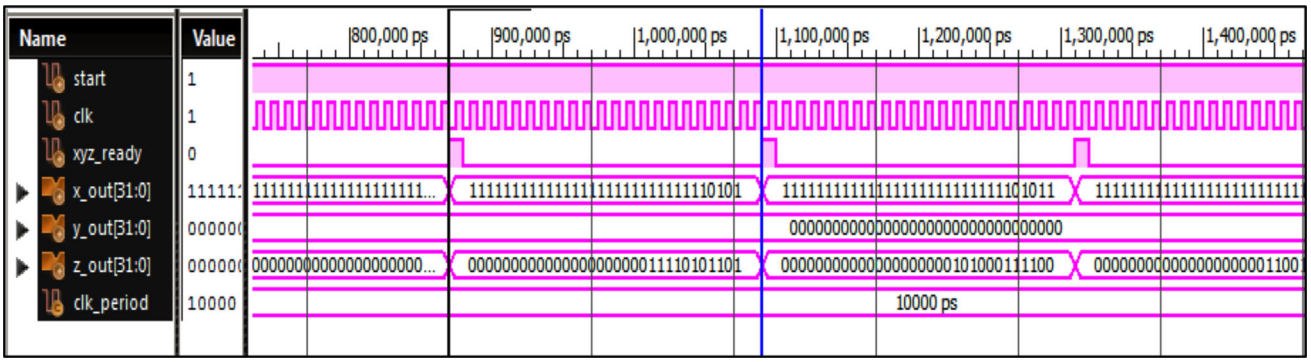

Fig. 3 Xilinx ISE Simulator results of Euler-based chaotic oscillator unit

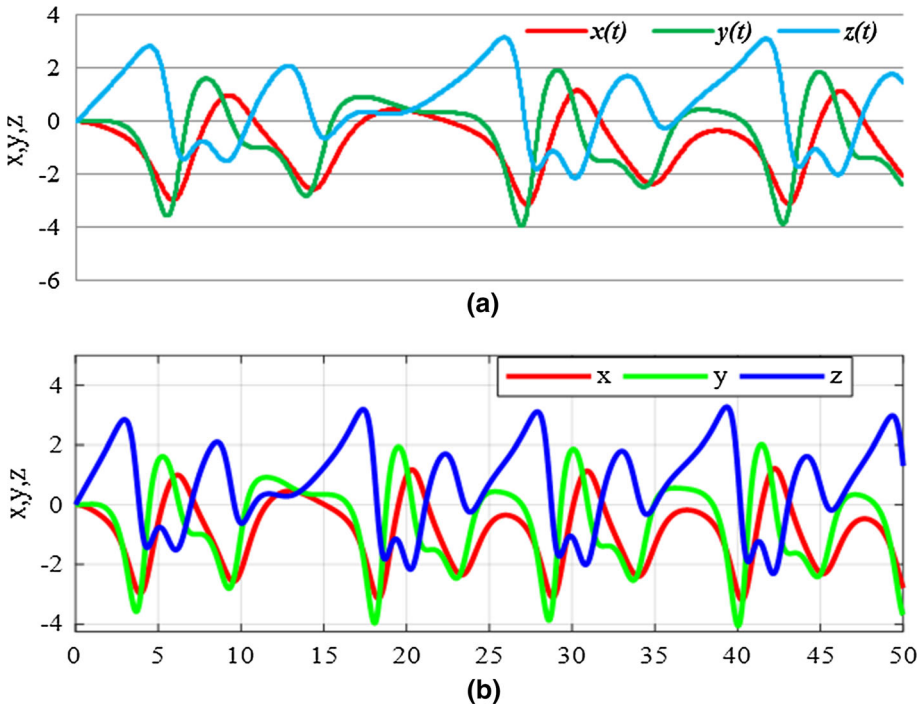

Fig. 4 The comparison of x(t), y(t), and z(t) time series of Euler-based chaotic systems **a** on Matlab, **b** on FPGA

FPGA-based RO unit has been synthesized for *Xilinx Virtex-6* (*xc6vlx75t*-3ff784) FPGA chip. The delay of the system is 0.635 ns and the operating frequency of the system is 1.5 GHz. The data processing time of the designed units have been obtained using Xilinx ISE Design Tools 14.2 simulation program. The obtained results of RO unit designed on FPGA from Xilinx ISE Simulator have been demonstrated in Fig. 5.

2.6 The design of proposed TRNG on FPGA

In this part, the design of dual entropy core new TRNG has been implemented by uniting the designs of chaotic-based RNG and RO-based RNG structures on FPGA in the post processing unit. The performances and FPGA chip statistics of chaotic-ring based new TRNG unit have been investigated with respect to performed designs. Figure 6 gives the proposed block diagram of discrete time chaotic-ring based dual entropy core new TRNG designed on FPGA. IQ-Math FPN method, developed by Texas Instruments [47], has been used as the quantification method in the design of chaos-based TRNG. FPN-based TRNG structure enables taking the 21 bits from fractional part of the 32-bits in fixed point standard each produced by chaotic oscillator unit.

Unlike the standard MUX structures, the *MUX* structure presented in this study has designed in 3X1 format. The Select input (*Sel*) is a 2-bit signal ranging from *00* to *10*. This input signal has been connected to *Counter* unit (C). The *Counter* unit counts as *00, 01, 10* and then returns to the initial value. In this way, when the output of the *Counter* takes "*00*" value, the related 1-bit value received from *X_RN* line has been transmitted to the output of the *MUX* unit. As the output of the *Counter* takes "*01*" value, the related 1-bit value received from *Y-RN* line has been sent to the output of the *MUX* unit. Finally, when the output

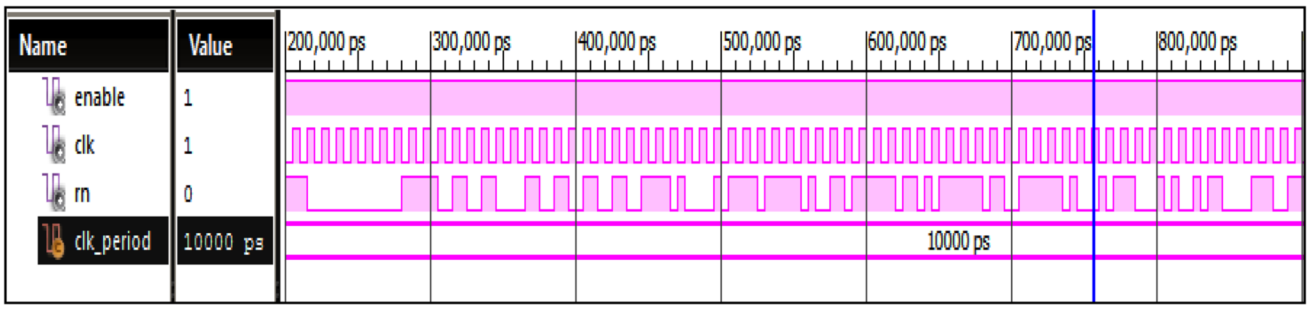

Fig. 5 The obtained results of RO unit designed on FPGA from Xilinx ISE simulator

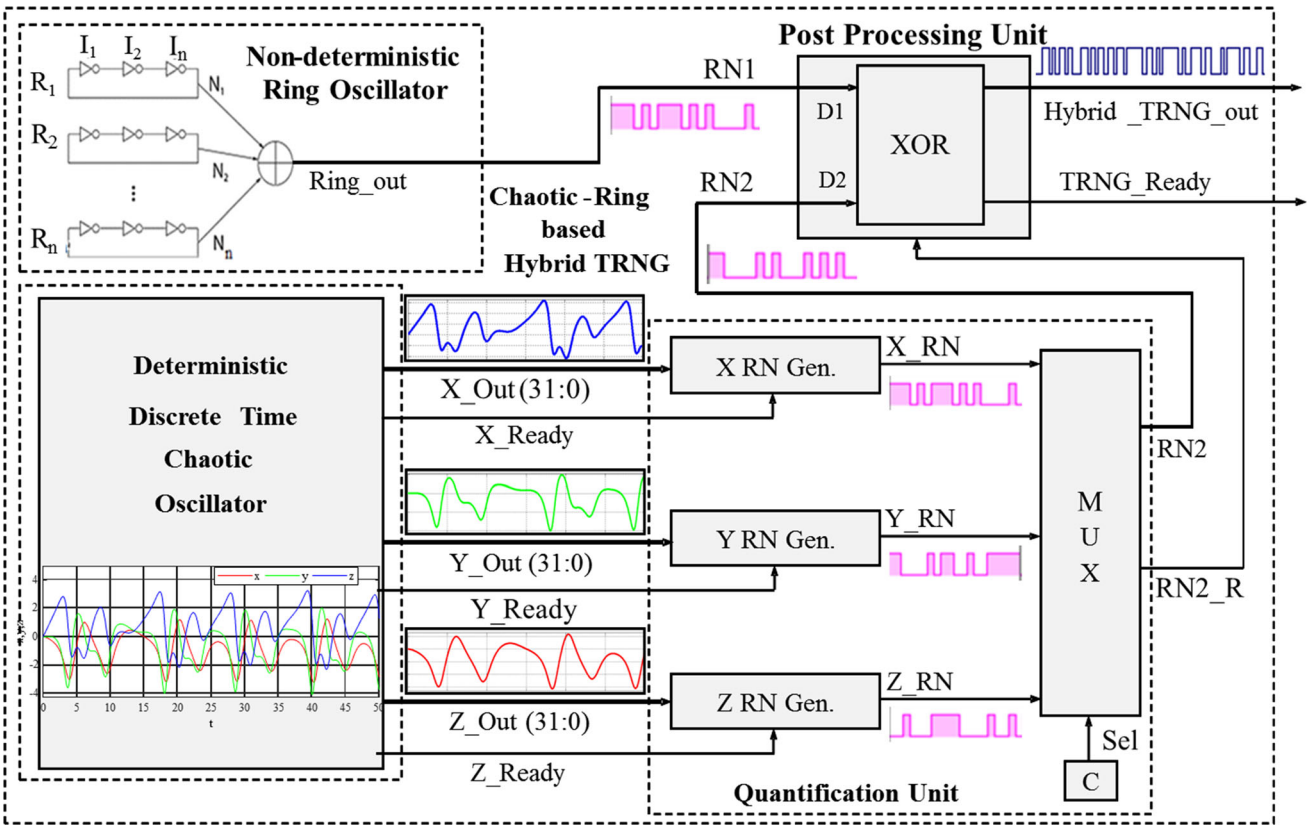

Fig. 6 The proposed architecture design of chaos-ring based dual entropy core TRNG

of the *Counter* takes "*10*" value, the related 1-bit value received from *Z-RN* line has been transmitted to the output of the *MUX* unit. This process repeats itself periodically.

Post processing unit has 2 inputs with one bit receiving from ring oscillator and chaos-based TRNG. These 2 input signals are the inputs of the designed XOR gate. XOR unit has 2 outputs as *TRNG_Ready* and *Hybrid_TRNG_out*. When the *XOR unit* produces a result, *TRNG_Ready* sends '*1*' value to the output. One of the post processing methods used in TRNG structures for passsing the statistical tests in literature is XOR operation. As the output signals of RNG having only one entropy source are applied to XOR operation, it reduces the throughput by half. Since the post processing unit presented in this study has a dual entropy structure formed as ring oscillator and chaos-based TRNG, there is no decrease in throughput.

3 Results

3.1 The proposed TRNG on FPGA

In this part, chaotic-ring based dual entropy core new TRNG unit using Euler numerical algorithm has been synthesized for the Xilinx Virtex-6 (xc6vlx75t-3ff784) FPGA chip and then the statistics related to chip resource consumption and clock frequencies of the unit have been analyzed. The data processing time of the proposed chaotic-ring based dual entropy core novel TRNG unit has been obtained using Xilinx ISE Design Tools 14.2 simulation program. The obtained results of the proposed chaotic-ring based dual entropy core TRNG unit from Xilinx ISE simulator have been shown in Fig. 7. The proposed TRNG unit has been synthesized and after the Place&Route process, the chip statistics have been given for the FPGA chip in Table 3. As can be seen from the results, there is no decrease for the data rate of the output of the post processing algorithm in the dual entropy core new TRNG units.

3.2 The randomness test results of FPGA-based dual entropy core new TRNG

In order to use the developed RNGs in the cryptographic applications, the randomness and statistical properties of the RNGs should be investigated and tested. Even though the outputs of the RNGs cannot be proven as random mathematically, it is possible to verify that the outputs of the RNGs have been random or not by using valid statistical tests. These tests declare whether the output of any

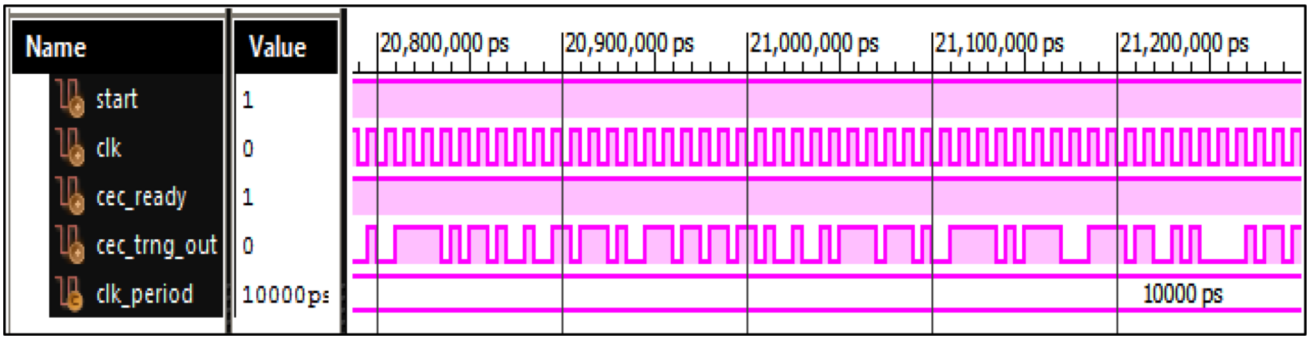

Fig. 7 Xilinx ISE simulator result of the designed dual entropy core new TRNG unit

Table 3 The chip statistics of the proposed chaos-ring based dual entropy core TRNG unit on FPGA

Xilinx Virtex–6 FPGA chip statistics	Used/utilization %
Number of Slice Registers	1318/1
Number of Slice LUTs	1355/3
Numbe of fully used LUT-FF pairs	1050/54
Number of bonded IOBs	4/1
Operating Frequency(MHz)	464.688
The data rate of TRNG (Mbps)	464.688

RNG has met the conditions, that a true random array can supply, or not. Besides, comments related to quality of the RNG can be made according to test results. To state a bit stream as random, it should pass all of the applied statistical tests. Even if one test fails, bit stream cannot be accepted as random. In this part, international statistical test namely, NIST 800-22 has been carried out to use the proposed chaotic-ring based dual entropy core new TRNG unit in the cryptographic applications safely.

The randomness of the generated numbers produced by TRNGs that used in cryptographic applications affects directly the security of these applications. In this respect, TRNGs have a critical role for cryptographic applications. Various studies for FPGA-based TRNG structures have been performed in the literature. The structures including classical oscillators, ring oscillators and Flip-Flops have generally been utilized in these studies. Exhibiting aperiodic behavior [48], sensitive dependence on initial conditions [49] and having sensitivity on the alterations of the system parameters [50, 51], are among the fundamental and important properties of chaotic systems. Although chaotic dynamics of such systems are identified in deterministic terms, having high sensitivity to the alterations in initial conditions, when united with their exponentially divergent aperiodic feature driven by the positive Lyapunov exponents, makes them suitable alternative for TRNG applications. In addition, the modifications of the initial conditions, system parameters and step size (Δh) make it impossible to replicate the chaotic dynamics exactly, as a consequence supplying the expected security and unpredictability for TRNG [32]. Because of the properties mentioned above, chaotic oscillator-based TRNG applications have intensively been carried out. Although RO-based TRNG designs provide high throughput, they cannot often pass statistical randomness tests including NIST Tests [11, 40, 52–54]. For this reason, post processing has been applied to these structures. As a result, each post processing operation reduces the bit generation rate of not only RO-based TRNG designs but also chaos-based TRNG designs. In this presented study, the throughput achieves the value of the maximum operating frequency of the design by combining RO-based TRNG structure and chaos-based TRNG structure in the post processing unit. That means, non-deterministic TRNG structure has been obtained having high data rates without a decrease in bit generation rate and any possibility to having access to elements of these sequences.

The structure designed for testing the randomness of the generated data set obtained from FPGA-based dual entropy core TRNG unit is given in Fig. 8. A testbench file has been created after the implementation of dual entropy core TRNG unit in Xilinx ISE Design Tools program on Virtex-6 (XC6VLX75T) FPGA chip. The data set of 1Mbit needed for the NIST-800-22 Test Suite has been obtained from FPGA-based dual entropy core TRNG unit by running the test bench file and has been saved into a txt file. Then, the obtained data set has been tested on PC using NIST-800-22 Test Suite.

Fig. 8 The designed structure for testing the randomness of generated data set obtained from TRNG unit

NIST Test Suite is the other statistical test that has international validity including 16 test. As Random Excursions and Random Excursions Variant tests need 1 Mbit data, in general 1 Mbit data have been saved into a file in computer environment and then the bit file has been subjected to 16 statistical tests in NIST Test Suite. Table 4 illustrates the NIST 800-22 test results of the proposed chaos-ring based dual entropy core new TRNG unit using Euler numerical algorithm on FPGA. As can be seen from the test results in Table 4, since $P\ value \geq 0{,}01$, the subjected bit streams have been accepted as random. In testing phase, as Random-Excursions Variant Test has 18 sub-tests for x variant that has $-9 \leq x \leq -1$ and $9 \leq x \leq 1$ conditions, 18 test results has been obtained from the proposed TRNG unit. In order to reduce the complexity in Table 4, only the result for $x = -9$ has been presented. For the rest of the situations of x variant, the generated bits have been met the test criterias. The proposed chaotic-ring based dual entropy core new TRNG unit on FPGA successfully passed all of the tests.

4 Conclusion

In this paper, the chaotic-ring based dual entropy core discrete time novel TRNG structure has been implemented on FPGA. In the first step, the proposed 3-D chaotic system has been modeled using Euler numerical integration method and then chaos analyses have been carried out by investigating the dynamic behaviours of the system. After that, the chaotic system has been modeled using VHDL with respect to 32-bit IQ-Math FPN standard on FPGA. In the modeling phase, Euler numerical algorithm has been used. The designed unit has been synthesized for the Xilinx Virtex-6 (xc6vlx75t-3ff784) FPGA chip using Xilinx ISE Design Tools 14.2 simulation program. After the Place&Route phase, maximum operating frequency of the chaotic oscillator reaches 464 MHz with respect to obtained results. In the second step, the RO-based RNG design has been performed using VHDL on FPGA. In general, the proposed chaotic-ring based dual entropy core TRNG unit has been created by uniting FNP Euler

Table 4 The NIST 800-22 Test results of the proposed TRNG unit on FPGA

NIST 800-22 statistical tests		Proposed chaotic-ring based TRNG	
		P value	Result
1.	Frequency (Monobit) Test	0.72184	Successful
2.	Block-Frequency Test	0.06380	Successful
3.	Runs Test	0.30368	Successful
4.	Longest-Run Test	0.19640	Successful
5.	Binary Matrix Rank Test	0.99834	Successful
6.	Discrete Fourier Transform Test	0.12786	Successful
7.	Non-Overlapping Templates Test	0.69314	Successful
8.	Overlapping Templates Test	0.90598	Successful
9.	Maurer's Universal Statistical Test	0.02262	Successful
10.	Linear-Complexity Test	0.01101	Successful
11.	Serial Test-1, Serial Test-2	0.05060	Successful
		0.87105	Successful
12.	Lempel–Ziv Test	0.94369	Successful
13.	Approximate Entropy Test	0.15224	Successful
14.	Cumulative-Sums Test	0.56254	Successful
15.	Random-Excursions Test		
	(for $x = -4$)		
	$x = -4$	0.49514	Successful
	$x = -3$	0.54785	Successful
	$x = -2$	0.15987	Successful
	$x = -1$	0.07354	Successful
	$x = 1$	0.35479	Successful
	$x = 2$	0.74585	Successful
	$x = 3$	0.35478	Successful
	$x = 4$	0.86541	Successful
16.	Random-Excursions Variant Test		
	$x = -9$	0.35789	Successful

numerical algoritm-based chaotic oscillator and RO unit in the XOR function used at the post processing unit on FPGA. The data rate of the proposed novel TRNG unit ranges 464 Mbps. In the last step, the generated bit streams obtained from FPGA-based proposed dual entropy core new TRNG has been subjected to NIST 800-22 test suites and all of them passed. As a result, in comparison with TRNGs based on the other techniques and studies, 464 Mbps throughput, which is the highest data rate to date with fulfilled test results, has been achieved without any postprocessing.

Compliance with ethical standards

Conflict of interest The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

1. Valtierra, J. L., Tlelo-Cuautle, E., & Rodríguez-Vázquez, Á. (2017). A switched-capacitor skew-tent map implementation for random number generation. *International Journal of Circuit Theory and Applications, 45*(2), 305–315. https://doi.org/10.1002/cta.2305.
2. Palacios-Luengas, L., Pichardo-Méndez, J. L., Díaz-Méndez, J. A., Rodríguez-Santos, F., & Vázquez-Medina, R. (2019). PRNG based on skew tent map. *Arabian Journal for Science and Engineering, 44,* 3817–3830. https://doi.org/10.1007/s13369-018-3688-y.
3. Jakimoski, G., & Kocarev, L. (2001). Chaos and cryptography: Block encryption ciphers based on chaotic maps. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, 48*(2), 163–169. https://doi.org/10.1109/81.904880.
4. Akkaya, S., Pehlivan, İ., Akgül, A., & Varan, M. (2018). The design and application of bank authenticator device with a novel chaos based random number generator. *Journal of the Faculty of Engineering and Architecture of Gazi University, 33*(3), 1171–1182. https://doi.org/10.17341/gazimmfd.416418.
5. Akhshani, A., Akhavan, A., Mobaraki, A., Lim, S.-C., & Hassan, Z. (2014). Pseudo random number generator based on quantum chaotic map. *Communications in Nonlinear Science and Numerical Simulation, 19*(1), 101–111. https://doi.org/10.1016/J.CNSNS.2013.06.017.
6. Özkaynak, F. (2014). Cryptographically secure random number generator with chaotic additional input. *Nonlinear Dynamics, 78*(3), 2015–2020. https://doi.org/10.1007/s11071-014-1591-y.
7. Banerjee, S., & Kurths, J. (2014). Chaos and cryptography: A new dimension in secure communications. *The European Physical Journal Special Topics, 223*(8), 1441–1445. https://doi.org/10.1140/epjst/e2014-02208-9.
8. Lee, J., Bi, Y., Peterson, G. D., Hinde, R. J., & Harrison, R. J. (2009). HASPRNG: Hardware accelerated scalable parallel random number generators. *Computer Physics Communications, 180*(12), 2574–2581. https://doi.org/10.1016/J.CPC.2009.07.002.
9. Mazrooei-Sebdani, R., & Dehghan, M. (2008). A non-trivial relation between some many-dimensional chaotic discrete dynamical systems and some one-dimensional chaotic discrete dynamical systems. *Computer Physics Communications, 179*(9), 628–633. https://doi.org/10.1016/J.CPC.2008.05.010.
10. Karakaya, B., Çelik, V., & Gülten, A. (2017). Chaotic cellular neural network-based true random number generator. *International Journal of Circuit Theory and Applications, 45*(11), 1885–1897. https://doi.org/10.1002/cta.2374.
11. Avaroğlu, E., Tuncer, T., Özer, A. B., Ergen, B., & Türk, M. (2015). A novel chaos-based post-processing for TRNG. *Nonlinear Dynamics, 81*(1–2), 189–199. https://doi.org/10.1007/s11071-015-1981-9.
12. Avaroğlu, E., Koyuncu, İ., Özer, A. B., & Türk, M. (2015). Hybrid pseudo-random number generator for cryptographic systems. *Nonlinear Dynamics, 82*(1–2), 239–248. https://doi.org/10.1007/s11071-015-2152-8.
13. Tuna, M., Karthikeyan, A., Rajagopal, K., Alçın, M., & Koyuncu, İ. (2019). Hyperjerk multiscroll oscillators with megastability: Analysis, FPGA implementation and a novel ANN-ring-based true random number generator. *AEU - International Journal of Electronics and Communications*. https://doi.org/10.1016/j.aeue.2019.152941.
14. Tuna, M., & Fidan, C. B. (2016). Electronic circuit design, implementation and FPGA-based realization of a new 3D chaotic system with single equilibrium point. *Optik - International Journal for Light and Electron Optics, 127*(24), 11786–11799. https://doi.org/10.1016/j.ijleo.2016.09.087.
15. Pourmahmood, M., Hasan, A., Aghababa, P., Aghababa, M. P., & Aghababa, H. P. (2013). A novel finite-time sliding mode controller for synchronization of chaotic systems with input nonlinearity. *Arabian Journal for Science and Engineering, 38,* 3221–3232. https://doi.org/10.1007/s13369-012-0459-z.
16. Sundarapandian, V., & Pehlivan, I. (2012). Analysis, control, synchronization, and circuit design of a novel chaotic system. *Mathematical and Computer Modelling, 55*(7–8), 1904–1915. https://doi.org/10.1016/j.mcm.2011.11.048.
17. Tuna, M., & Fidan, C. B. (2018). 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, 33*(2), 469–486. https://doi.org/10.17341/GUMMFD.71479.
18. Khanzadi, H., Eshghi, M., Khanzadi, H., & Borujeni, S. E. (2014). Image encryption using random bit sequence based on chaotic maps. *Arabian Journal for Science and Engineering, 39,* 1039–1047. https://doi.org/10.1007/s13369-013-0713-z.
19. Kaur, M., & Kumar, V. (2018). Adaptive differential evolution-based lorenz chaotic system for image encryption. *Arabian Journal for Science and Engineering, 43,* 8127–8144. https://doi.org/10.1007/s13369-018-3355-3.
20. Çiçek, S., Ferikoğlu, A., & Pehlivan, İ. (2016). A new 3D chaotic system: Dynamical analysis, electronic circuit design, active control synchronization and chaotic masking communication application. *Optik - International Journal for Light and Electron Optics, 127*(8), 4024–4030. https://doi.org/10.1016/j.ijleo.2016.01.069.
21. Çavuşoğlu, Ü., Akgül, A., Kaçar, S., Pehlivan, I., & Zengin, A. (2016). A novel chaos-based encryption algorithm over TCP data packet for secure communication. *Security and Communication Networks, 9*(11), 1285–1296. https://doi.org/10.1002/sec.1414.
22. Effati, S., Saberi Nik, H., & Jajarmi, A. (2013). Hyperchaos control of the hyperchaotic Chen system by optimal control design. *Nonlinear Dynamics, 73*(1–2), 499–508. https://doi.org/10.1007/s11071-013-0804-0.
23. Yang, C.-C., & Yang, C.-C. (2014). Adaptive single input control for synchronization of a 4D Lorenz-Stenflo chaotic system. *Arabian Journal for Science and Engineering, 39,* 2413–2426. https://doi.org/10.1007/s13369-013-0768-x.
24. Rajagopal, K., Tuna, M., Karthikeyan, A., Koyuncu, İ., Duraisamy, P., & Akgul, A. (2019). Dynamical analysis, sliding mode synchronization of a fractional-order memristor Hopfield

neural network with parameter uncertainties and its non-fractional-order FPGA implementation. *The European Physical Journal Special Topics, 228*(10), 2065–2080. https://doi.org/10.1140/epjst/e2019-900005-8.
25. Çiçek, S., Uyaroğlu, Y., & Pehlivan, İ. (2013). Simulation and circuit implementation of Sprott Case H chaotic system and its synchronization application for secure communication systems. *Journal of Circuits, Systems and Computers, 22*(04), 1350022. https://doi.org/10.1142/S0218126613500229.
26. Rajagopal, K., Akgul, A., Jafari, S., & Aricioglu, B. (2018). A chaotic memcapacitor oscillator with two unstable equilibriums and its fractional form with engineering applications. *Nonlinear Dynamics, 91*(2), 957–974. https://doi.org/10.1007/s11071-017-3921-3.
27. Akgul, A., Calgan, H., Koyuncu, I., Pehlivan, I., & Istanbullu, A. (2015). Chaos-based engineering applications with a 3D chaotic system without equilibrium points. *Nonlinear Dynamics, 84*(2), 481–495. https://doi.org/10.1007/s11071-015-2501-7.
28. Rajagopal, K., Akgul, A., Jafari, S., Karthikeyan, A., & Koyuncu, I. (2017). Chaotic chameleon: Dynamic analyses, circuit implementation, FPGA design and fractional-order form with basic analyses. *Chaos, Solitons & Fractals, 103,* 476–487. https://doi.org/10.1016/J.CHAOS.2017.07.007.
29. Koyuncu, İ., Şahin, İ., Gloster, C., & Sarıtekin, N. K. (2017). A neuron library for rapid realization of artificial neural networks on FPGA: A case study of Rössler chaotic system. *Journal of Circuits, Systems and Computers, 26*(01), 1750015. https://doi.org/10.1142/S0218126617500153.
30. Tuna, M., Alçın, M., Koyuncu, İ., Fidan, C. B., & Pehlivan, İ. (2019). High speed FPGA-based chaotic oscillator design. *Microprocessors and Microsystems, 66,* 72–80. https://doi.org/10.1016/J.MICPRO.2019.02.012.
31. Kaya, T. (2019). A true random number generator based on a Chua and RO-PUF: Design, implementation and statistical analysis. *Analog Integrated Circuits and Signal Processing*. https://doi.org/10.1007/s10470-019-01474-2.
32. Cicek, I., Pusane, A. E., & Dundar, G. (2014). A novel design method for discrete time chaos based true random number generators. *Integration, the VLSI Journal, 47*(1), 38–47. https://doi.org/10.1016/j.vlsi.2013.06.003.
33. Koyuncu, İ., & Turan Özcerit, A. (2017). The design and realization of a new high speed FPGA-based chaotic true random number generator. *Computers & Electrical Engineering, 58,* 203–214. https://doi.org/10.1016/J.COMPELECENG.2016.07.005.
34. Tavas, V., Demirkol, A. S., Ozoguz, S., Kılınç, S., Toker, A., & Zeki, A. (2010). An IC random number generator based on Chaos. In *International conference on applied electronics* (*AE*) (pp. 1–4). Pilsen.
35. Ergün, S., & Özoguz, S. (2007). Truly random number generators based on a non-autonomous chaotic oscillator. *AEU - International Journal of Electronics and Communications, 61*(4), 235–242. https://doi.org/10.1016/j.aeue.2006.05.006.
36. Sunar, B., Martin, W., & Stinson, D. (2007). A provably secure true random number generator with built-in tolerance to active attacks. *IEEE Transactions on Computers, 56*(1), 109–119. https://doi.org/10.1109/TC.2007.250627.
37. Schellekens, D., Preneel, B., & Verbauwhede, I. (2006). FPGA vendor agnostic true random number generator. In *2006 International conference on field programmable logic and applications* (pp. 1–6). Madrid. https://doi.org/10.1109/fpl.2006.311206.
38. Ning, L., Ding, J., Chuang, B., & Xuecheng, Z. (2015). Design and validation of high speed true random number generators based on prime-length ring oscillators. *The Journal of China Universities of Posts and Telecommunications, 22*(4), 1–6. https://doi.org/10.1016/S1005-8885(15)60661-6.
39. Park, M., Rodgers, J. C., & Lathrop, D. P. (2015). True random number generation using CMOS Boolean chaotic oscillator. *Microelectronics Journal, 46*(12), 1364–1370. https://doi.org/10.1016/j.mejo.2015.09.015.
40. Cicek, I., Pusane, A. E., & Dundar, G. (2014). A new dual entropy core true random number generator. *Analog Integrated Circuits and Signal Processing, 81*(1), 61–70. https://doi.org/10.1007/s10470-014-0324-y.
41. Wieczorek, P. Z., & Golofit, K. (2014). Dual-metastability time-competitive true random number generator. *IEEE Transactions on Circuits and Systems I: Regular Papers, 61*(1), 134–145. https://doi.org/10.1109/TCSI.2013.2265952.
42. Wold, K., & Tan, C. H. (2009). Analysis and enhancement of random number generator in FPGA based on oscillator rings. *International Journal of Reconfigurable Computing, 2009,* 1–8. https://doi.org/10.1155/2009/501672.
43. Tuncer, T., Avaroglu, E., Türk, M., & Ozer, A. B. (2015). Implementation of non-periodic sampling true random number generator on FPGA. *Informacije MIDEM, 44*(4), 296–302.
44. Wang, Y., & Li, S. (2016). A high-speed digital true random number generator based on cross ring oscillator. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, E99.A*(4), 806–818. https://doi.org/10.1587/transfun.e99.a.806.
45. Fatemi-Behbahani, E., Ansari-Asl, K., & Farshidi, E. (2016). A new approach to analysis and design of chaos-based random number generators using algorithmic converter. *Circuits, Systems, and Signal Processing, 35*(11), 3830–3846. https://doi.org/10.1007/s00034-016-0248-0.
46. Pehlivan, İ., & Uyaroğlu, Y. (2012). A new 3D chaotic system with golden proportion equilibria: Analysis and electronic circuit realization. *Computers & Electrical Engineering, 38*(6), 1777–1784. https://doi.org/10.1016/j.compeleceng.2012.08.007.
47. Vipin Chandra, S. (2014). A Survey on CORDIC Algorithm Implementations Using Different Number Format. *International Journal of Innovative Research in Science, Engineering and Technology, 3*(6), 13452–13458.
48. Strogatz, S. (2001). *Nonlinear dynamics and chaos: With applications to physics, biology, chemistry, and engineering (studies in nonlinearity)*. Cambridge: Westview Press.
49. Ott, E. (2002). *Chaos in dynamical systems*. Cambridge: Cambridge University Press.
50. Koyuncu, I., Ozcerit, A. T., & Pehlivan, I. (2014). Implementation of FPGA-based real time novel chaotic oscillator. *Nonlinear Dynamics, 77*(1–2), 49–59. https://doi.org/10.1007/s11071-014-1272-x.
51. Alligood, K. T., Sauer, T., & Yorke, J. A. (1996). *Chaos: An introduction to dynamical systems*. New York: Springer.
52. Garipcan, A. M., & Erdem, E. (2019). Implementation and performance analysis of true random number generator on FPGA environment by using non-periodic chaotic signals obtained from chaotic maps. *Arabian Journal for Science and Engineering, 44*(11), 9427–9441. https://doi.org/10.1007/s13369-019-04027-x.
53. Çiçek, I., & Dündar, G. (2011). A hardware efficient chaotic ring oscillator based true random number generator. In *2011 18th IEEE international conference on electronics, circuits, and systems, ICECS 2011* (pp. 430–433). https://doi.org/10.1109/icecs.2011.6122305.
54. Jiteurtragool, N., & Masayoshi, T. (2017). Hybrid random number generator based on chaotic oscillator. In *2016 Management and innovation technology international conference, MITiCON 2016* (pp. MIT133–MIT136). Institute of Electrical and Electronics Engineers Inc. https://doi.org/10.1109/miticon.2016.8025231.

İsmail Koyuncu has an M.S. from Abant Izzet Baysal University, Bolu-Turkey. He completed his doctoral research in the Department of Electrical and Electronics Engineering at Sakarya University, Sakarya-Turkey in 2014. Since 2017, he is an Associate Professor in the Department of Mechatronics Engineering at Afyon Kocatepe University, in Afyon-Turkey. His main research interests are FPGA-based digital system design, chaos, TRNG and reconfigurable computing. He is also interested in FPGA-based artificial neural networks and computer graphics.

Murat Tuna M.S. and graduated from Kocaeli University in 2004, in 2008 respectively and completed his Ph.D in the Department of Electrical and Electronics Engineering at Karabuk University, Karabuk-Turkey in 2017. He is currently working as Assistant Professor at Kirklareli University in Turkey since 2009. His research topics include chaos, TRNG, FPGA-based digital system design and reconfigurable computing. He is also interested in mathematical model and control of nonlinear systems.

Ihsan Pehlivan received the B.S. from Istanbul Technical University in 1997, M.S. and Ph.D. degrees in 2001 and 2007 respectively in Electrical-Electronic Engineering from Sakarya University, Sakarya-Turkey. He is an Professor in the Department of Electrical and Electronics Engineering at Sakarya Applied Sciences University, in Sakarya-Turkey. He is the author of a book, more than 30 articles. His research interests include chaos, electric circuits and signals-system.

Can Bülent Fidan He received the B.Sc. and M.Sc. Degree in Electronics and Communications Engineering in 1988 and 1991, respectively, and the Ph.D. degree in Electrical Engineering in 2001, all from the Yildiz Technical University, Istanbul, Turkey. He is currently Assistant Professor of Mechatronics Engineering at the Karabuk University, Karabuk, Turkey. His main research interest is signal processing and intelligent control applications, neural networks.

Murat Alçın was born in Esence Village, Eskisehir, in 1982. He received the B.S., M.S. and Ph.D. degrees in Electronic-Computer Teaching from the University of Marmara, Turkey, in 2006 and in 2009, and department of Electrical and Electronics Engineering at Sakarya University, in Sakarya-Turkey, 2017, respectively. From 2006 to 2008, he was a teacher in Simav Anatolian Vocational High School in Kutahya. Since 2018, he is an Assistant Professor in the Department of Mechatronics Engineering at Afyon Kocatepe University, in Afyon-Turkey. His research interests include Neural Networks, Chaotic Systems and FPGA-based digital system design.