University of Baghdad
College of Engineering
JOURNAL OF ENGINEERING

Journal of Engineering
journal homepage: www.joe.uobaghdad.edu.iq
Number 3 Volume 28 March 2022

Electrical, Electronics and communications, and Computer Engineering

Scheme for Generating True Random Numbers using Electromechanical Switches

Oday A. L. A. Ridha *
Assistant Professor
Dept. of Electronics and Communications,
College of Engineering
University of Baghdad
Baghdad / Iraq
Oday.ridha@coeng.uobaghdad.edu.iq

Ghassan Nihad Jawad
Lecturer
Dept. of Electronics and Communications,
College of Engineering
University of Baghdad
Baghdad / Iraq
Ghassan.jawad@coeng.uobaghdad.edu.iq

ABSTRACT

This paper proposes a novel method for generating True Random Numbers (TRNs) using electromechanical switches. The proposed generator is implemented using an FPGA board. The system utilizes the phenomenon of electromechanical switch bounce to produce a randomly fluctuated signal that is used to trigger a counter to generate a binary random number. Compared to other true random number generation methods, the proposed approach features a high degree of randomness using a simple circuit that can be easily built using off-the-shelf components. The proposed system is implemented using a commercial relay circuit connected to an FPGA board that is used to process and record the generated random sequences. Applying statistical testing on the experimentally generated sequences revealed a high degree of randomness, which proves its viability to modern applications, such as cryptography and communication system simulation and modeling.

Keywords: True Random Numbers, Encryption, Cryptography, Electro-Mechanical Switch Bounce, FPGA

طريقة لتوليد الأرقام العشوائية الحقيقية باستخدام المفاتيح الكهروميكانيكية

*** عدي عبد اللطيف عبد الرضا**
استاذ مساعد
قسم الهندسة الالكترونية والاتصالات
كلية الهندسة
جامعة بغداد

غسان نهاد جواد
مدرس
قسم الهندسة الالكترونية والاتصالات
كلية الهندسة
جامعة بغداد

الخلاصة

*Corresponding author
Peer review under the responsibility of University of Baghdad.
https://doi.org/10.31026/j.eng.2022.03.06

Article received: 15 / 1/ 2022
Article accepted: 20 / 2/ 2022
Article published:1/3/2022

يُقدم هذا البحث نظام جديد لتوليد الأرقام العشوائية الحقيقية والتي لها أهمية كبيرة في أنظمة الاتصالات والتشفير الحديثة. يعتمد النظام المقترح على ظاهرة الارتداد العشوائي للمفاتيح الكهروميكانيكية والتي تحدث عند فتح وغلق المفاتيح الكهربائية بكافة أنواعها، حيث يتم إدخال الإشارات العشوائية المتولدة من هذه الظاهرة الى عداد رقمي يقوم بتوليد الأرقام الثنائية كنتيجة للتغييرات العشوائية الداخلة اليه. تم تنفيذ النظام المقترح باستعمال مفتاح كهروميكانيكي يُفتح ويُغلق الياً عن طريق نبضات متعاقبة لتفعيل عملية الارتداد عند كل نبضة، ومن ثم ربط المفتاح بلوح مصفوفة منطقية قابلة للبرمجة (FPGA) لتقوم بتوليد سلاسل الأرقام العشوائية الحقيقية وتسجيلها. تبيّن من اختبار السلاسل المتولدة إحصائياً الدرجة العالية من العشوائية مما يعطي النظام المقترح الأفضلية لبساطته الشديدة مقارنة ببقية الأنظمة التي تقوم بتوليد سلاسل مشابهة لكن بدوائر أكثر كلفة وتعقيد، مما يجعل النظام المقترح في غاية الأهمية لأنظمة الاتصالات والتشفير المتقدمة والتي تعتمد بشكل كبير على سلاسل ذات عشوائية عالية لضمان سريّة المعلومات المتبادلة.

الكلمات الرئيسية: ارقام عشوائية حقيقية، تشفير، ارتداد المفاتيح الكهروميكانيكية، مصفوفة منطقية قابلة للبرمجة

1. INTRODUCTION

Recently, random number generation has gained significant importance in different fields of science, engineering, and economics: first and foremost, the growing need for the confidential exchange of information during the last decade have increased the demand for cryptography and data encryption systems, whose cornerstone is generating random number sequences **(Buchovecka, et al., 2016)**. Also, modern scientific and engineering fields demand extensive modeling and simulation to study different physical, chemical, and biological processes. Such operations require random numbers to mimic various conditions expected in real life **(Rossetti, 2015)**. Furthermore, other disciplines, such as economic studies, weather forecast, and online commerce, benefit from random sequences for various modeling purposes **(Crnjac and Kvesic, 2008), (Meyer, et al., 2008)** and **(Ahmad, et al., 2002)**. Due to the above, Random Number Generators (RNGs) have become indispensable parts of many present-day systems in different fields of science, engineering, and finance.

Generally speaking, the most widely-used RNGs are based on algorithms that produce seemingly unrepeatable sequences of numbers after starting from certain initial conditions (usually called seeds). Such generators are considered *pseu*do-random number generators (PRNGs) because if two machines apply the same algorithm to produce a random number from the same seed, both of them will produce the same number **(Dorrendorf, et al., 2009)**. Despite their low cost and high speed, using such random number generators could significantly compromise the security of any cryptography system. Therefore, it is necessary to generate *truly* random sequences with extremely low repeatability.

True Random Number Generators (TRNGs) rely on physical phenomena that are genuinely random – such as the thermal noise in a semiconductor or the radioactive decay of an atomic nucleus – to produce the random sequences **(Yu, et al., 2019)** and **(Stipčević and Koç, 2014)**. It is worth noting that more straightforward approaches, such as tossing a coin, rolling a dice, or shuffling cards, have been used for hundreds of years to generate truly random numbers primitively. The randomness in such operations originates from the fact that the smallest

uncertainty in the initial conditions ensures equal probabilities of all the outcomes **(Diaconis et al., 2007)**. However, such methods are extremely slow and cannot be practically used to accommodate the requirements of modern technologies. Therefore, other sources of "physical entropy" have been incorporated in TRNGs. In **(Bonilla, et al., 2016)**, the chaotic oscillations semiconductor lasers were used to generate truly random bits at high speed. However, this method makes use of highly advanced sub-micron superlattice technology to achieve the necessary quantum fluctuation condition at room temperature. Another approach utilized the anisotropy in voltage-controlled magnetic tunnel junctions to make TRNGs. However, this method suffers from limited applicability since it is limited to certain conditions of an applied magnetic field **(Lee, et al., 2017)**. In **(Jiang, et al. 2017)**, the stochastic delay time of threshold switching in memristors was demonstrated to show true random behavior. However, in addition to its high cost and complicated design, the current memristor technology is prone to instability. In addition, it features a non-uniform range of resistance, high fault probability, and unreliability in physical implementation **(Hajto, et al., 2019)**. An alternative approach can be used to obtain true random sequences by using the function (TRUERAND) in the MATLAB software package **(Deoras, 2016)**. Using this function will instruct the compiler to contact an online resource that generates true random numbers based on the atmospheric noise. Clearly, this method is not practical for real-time and stand-alone applications that lack online connection.

This paper proposes a simple and affordable TRNG system by relying on the contact bounce phenomenon featured in electromechanical switches during its closing and opening operations. Firstly, the phenomenon mentioned above is studied and analyzed to examine its degree of randomness based on the bounce duration and time-dependent factors. Next, a simple off-the-shelf electromechanical switch is used alongside an FPGA board to produce the random number every time the switch is triggered.

2. BACKGROUND

Electrical contact can be simply defined as the interface between two conducting objects. Such interface is usually incorporated in a device with the ability to change its state, which might be a physically (or mechanically) operated switch, such as a push-button, or an electrically operated switch, such as a relay. **Fig. 1** shows a generic diagram of an electromechanical switch **(Vinod, et al., 2018)**.

Electrical contact bounce occurs for a brief time after closing the contact due to the rebound in the switch's moving parts following their impact. In electrical circuits, contact bounce usually results in fluctuations in the current supplied to the load for a certain duration of time (called the *bounce time*) before the contacts settle to their normal position.

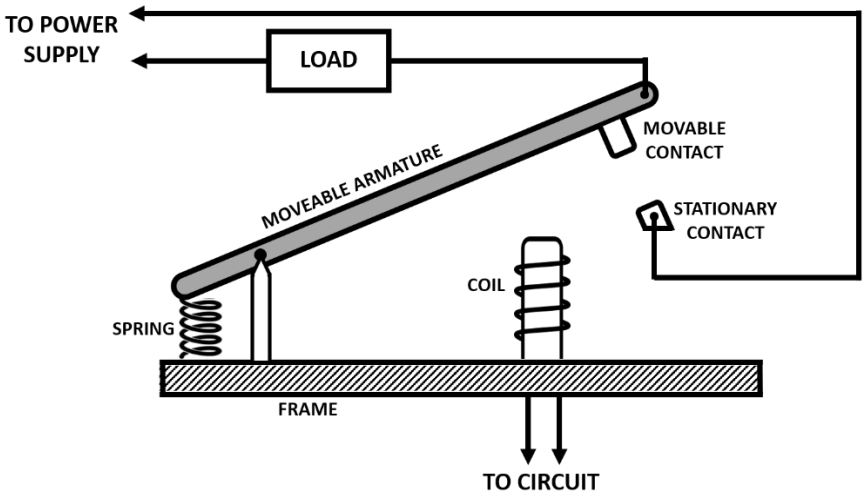

Figure 1. Generic diagram of a mechanical switch (adapted from **(Vinod, et al., 2018)**.

This phenomenon usually adversely affects the fidelity of any electrical circuit since it causes many problems with high current circuits and systems and results in significant load damage in many cases. Here, however, the same phenomenon is being analyzed and evaluated to generate true random sequences.

As clarified before, generating true random numbers by utilizing physical phenomena can be achieved when the phenomena are considered stochastic. Such a process usually features the property of its output being highly affected by the slightest change in one of its initial conditions. For example, in a simple coin toss experiment, it has been shown that there are numerous effects on the outcome of every time the coin is tossed **(Diaconis et al., 2007)**, which makes the process very close to being truly random.

Here, contact bounce fluctuations will be analyzed to understand the origin of its behavior and assess its degree of randomness. The changing in the state of any switch can be categorized into two operations: opening the contact (or the *break* operation) and closing the contact (or the *make* operation). The main effect of the former operation is the *arcing*, where the current travels between the two contacts during their separation. This effect is minimized for small values of currents, as in the considered case. Therefore it will not be addressed further in this discussion. However, the latter operation can be split into two episodes: the impact of the contact and the contact bounce. Generally speaking, when an external force is applied to the contact area in a switch, the kinetic energy is absorbed in the interface or the structure of the switch. When the external force is released, the contacts are given back some energy, resulting in their separation. Because of the spring mechanism acting on the contacts, the impact is repeated until all the kinetic energy of the first impact is dissipated. This process constitutes the contact bounce, where each bounce is considered as a very fast break and make operation.

In addition to the mechanical consideration, the electrical contact also affects the bounce process in terms of duration, frequency, and number of bounces by altering the events during contact bounce. **Fig. 2** illustrates the behavior of an electrical contact voltage during the contact "make" operation measured for a commercial relay switch. It can be seen that the voltage fluctuates for a period of time (bounce time) of around 1 ms before settling to its final value.

It has been found that, in most circuits, the current is delayed by the effect of circuit inductance. Hence, the impact time (the time at which a bounce occur), and the number of bounce, increase with the current. However, the bounce time is independent of the current passing through the

contact. Instead, bounce time decreases as the static contact force increases. Moreover, the static contact force depends on the wear and tear in the spring contacts, which is not stable **(Slade, 2017)**. From above, it can be deduced that the initial conditions of a mechanical switch have considerable effects on its bounce behavior. However, to rely on the contact bounce to produce True Random Numbers, it is necessary to investigate the parameters that affect the different aspects of this phenomenon, especially the impact time, the number of bounces, and the frequency of the bounce.

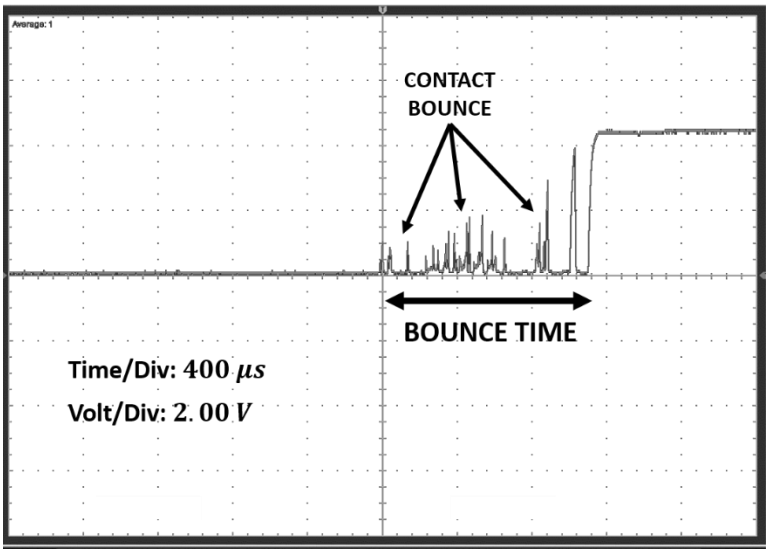

Figure 2. Measured switch voltage during the "make" operation showing the contact bounce.

Starting with the impact time, it was reported in **(Ksiazkiewicz, et al., 2019)** that the speed of the first impact has considerable effects on the subsequent events, which have a major influence on the production of the random numbers, as will be illustrated in the next section. However, this factor is affected by two factors: mechanical and electrical. Mechanical effects include contact material, surface degradation, gravitational forces, dust contamination, wear, sliding, and fretting **(Barkan, 1967)** and **(Ren, et al., 2016)**. On the other hand, electrical effects include dimensions of the current paths, resistivity of the contact point, relay electromagnetic coil, and the self-inductance of the connecting wires. Moreover, the values of some of these factors are highly dependent on other physical entities. For example, the ambient temperature and humidity affect both mechanical and electrical quantities of the electrical contact **(Ren, et al., 2016)**, **(Ksiazkiewicz, et al., 2019)**, **(Zhang, et al., 2018)** and **(Zhang, et al., 2019)**.
Based on the above, it can be concluded that there are numerous factors affecting the velocity of the first impact and, consequently, the impact times, bounce frequency, and bounce number. These complex interactions will render the process of generating the contact bounce a suitable candidate for generating truly random numbers.

3. PROPOSED SYSTEM

The proposed TRNG system consists of two main parts: a source of randomness, which is the electromechanical switch, and an up- (or down-) counter. The clock input of the counter is fed from the output of the electromagnetic switch, where the clock is connected to the ground when

the switch is ON, or left open when the switch is OFF, as shown in **Fig. 3**. Here, the counter is implemented using four D-type flip-flops to produce a 4-bit random number **(Mano, 2017)**; however, it is possible to use any type of flip-flops to achieve the same function.

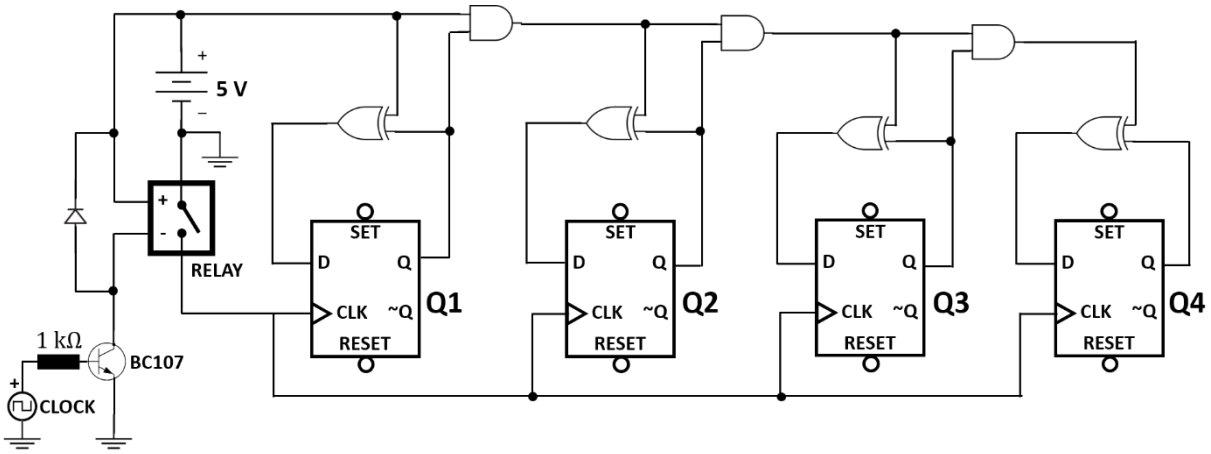

Figure 3. Diagram of a 4-bit true random number generator circuit.

By considering logic gates implemented using the Transistor-Transistor-Logic (TTL) technology, connecting the clock input to the ground is equivalent to feeding all the flip-flops of the counter with a logic zero. On the other hand, leaving the same input open is equivalent to providing them with a logical one.

The necessary pulses required to operate the counter are obtained when the electromagnetic switch is toggled between the ON and OFF states. By taking into account the contact bounce phenomenon that takes place every time the switch is turned ON, each state toggle of the switch generates a number of pulses that result in a fast and random alternating of the clock input. As discussed earlier, the complex interactions that affect the number and duration of contact bounce render this process truly random to a high degree. Therefore, the number generated by the counter after each switch toggle can be considered as a True Random Number (TRN).

As shown in **Fig. 3**, to automate the generation of the random contact bounce pulses, the switching operation is performed using a relay that is driven by an external pulse generator and provided with the necessary current for its operation using a transistor. Moreover, to avoid a surge in the voltage due to the abrupt reduction in current when the relay is open, a diode (known as a *freewheeling diode*) is connected across the relay coil.

It is worth noting that the system shown in **Fig. 3** is not the only possible arrangement that can be used to obtain true random numbers using the contact bounce phenomenon. A different arrangement can be used to perform the same task by incorporating different types of switches, counters, or logic circuitry. However, the circuit shown in **Fig. 3** represents the simplest approach to demonstrating the proposed system without unnecessarily increasing its complexity. Any future development in the system can simply be made by upgrading each part separately.

The proposed TRNG system was implemented using an FPGA kit, as shown in **Fig. 4**. In this circuit, the built-in logic gates of the FPGA kit (DE2-115) were used to build the counter and the logic circuitry required to generate the random number.

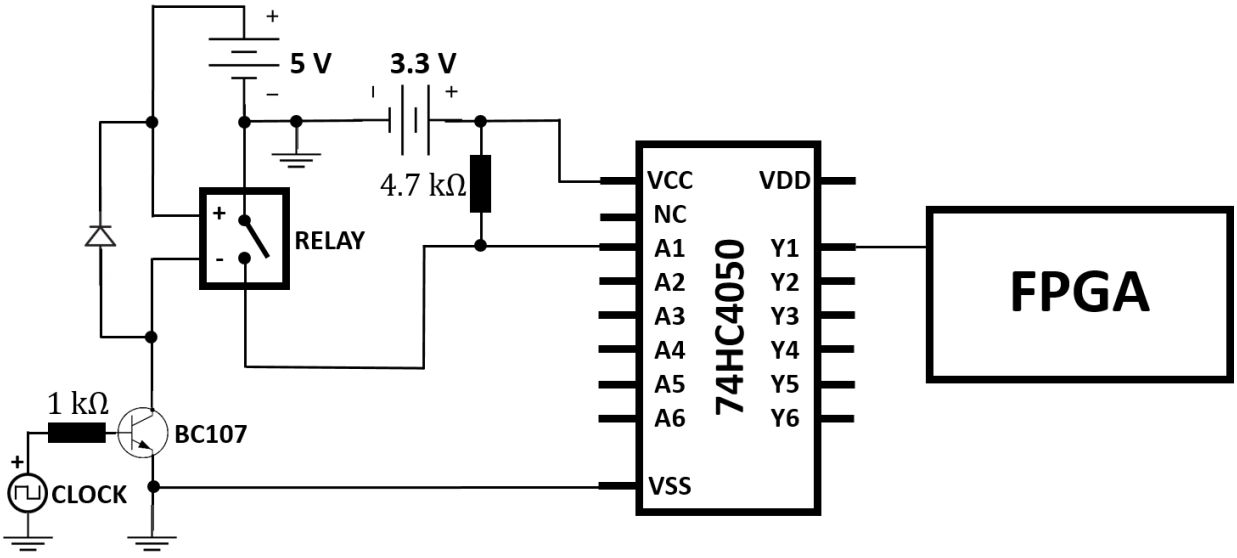

Figure 4. Circuit diagram of the TRNG generator implemented using FPGA.

Since the built-in buffers in the FPGA kit are designed to accommodate quick changes from zero to one, they have the tendency to eliminate the multiple triggers from the input pins during the rise time. For most circuits, this increases the tolerance of the kit to external noise. However, the idea behind the TRNG is to utilize these random triggers to generate the random numbers. Therefore, an external buffer was used to emphasize the random effects and take the full benefit of contact bounce actions during the switching of the relay. The general-purpose integrated circuit 74HC4050 shown in **Fig. 4** was used as a driver that works as a random contact bounce noise magnifier. Two voltage sources were used to accommodate the requirements for the relay (5 V) and the FPGA board (3.3 V).

Once the circuit started to operate, the input pulse was connected to a waveform generator with a frequency of 1 kHz. The input pulse stream toggles the relay between the ON and OFF states at the same frequency. For each relay toggle, the buffer circuit conveys the random contact bounce pulses as the input clock for the FPGA board. As a result, an internally-built 4-bit counter generates a random number based on the random clock changes.

4. RESULTS AND DISCUSSION

Fig. 5 shows a photograph of the proposed TRNG system implemented using the FPGA board (DE2-115). The system was used to produce a sequence of 4-bit random numbers, as illustrated in Table 1, which lists the first 15 numbers generated by the system.

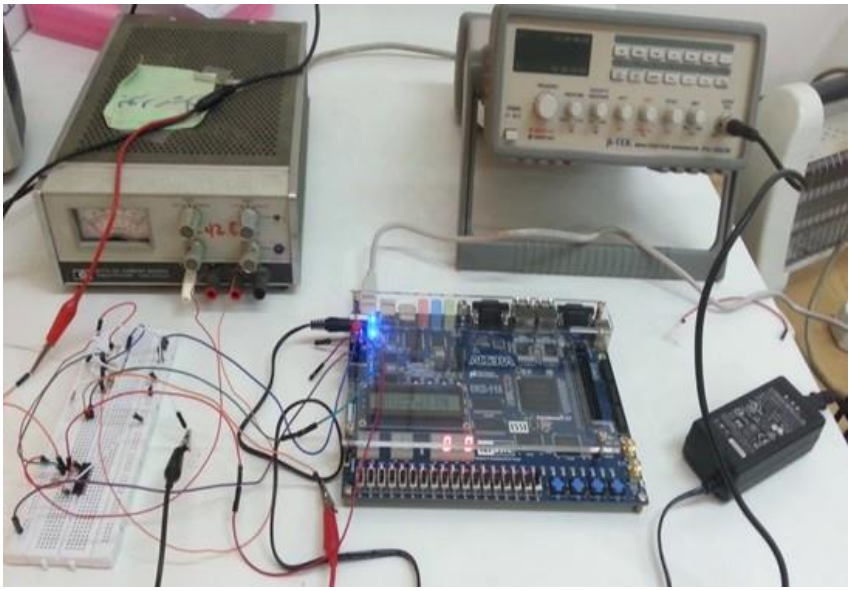

Figure 5. Photograph of the implemented true random number generator circuit.

After converting each binary random number to the decimal form, the first 250 generated random numbers were recorded and plotted in **Fig. 6**. The randomness of the generated sequence is clear from the fluctuations of the generated numbers between the 16 possible values.

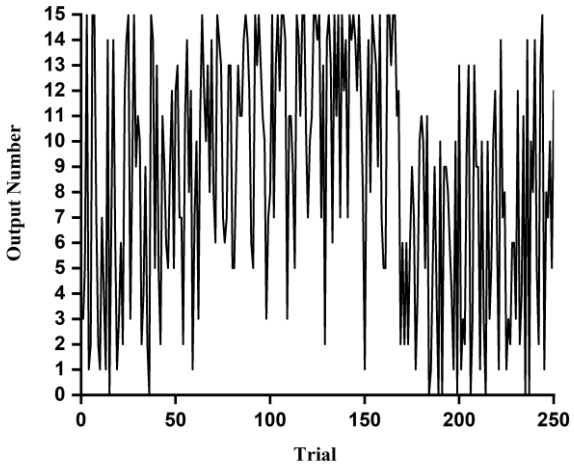

Figure 6. Output from the 4-bit TRNG for the first 250 trials.

Table 1. A sample of the binary output of the proposed TRNG for the first fifteen trials.

Output / trial	Q_0	Q_1	Q_2	Q_3
1	0	0	0	0
2	0	1	1	1
3	1	0	0	0
4	1	1	0	0
5	1	1	0	1
6	0	0	1	1
7	0	1	0	0
8	1	1	0	0
9	1	1	0	1
10	0	1	0	0
11	0	1	0	1
12	1	0	1	1
13	1	1	0	0
14	0	0	0	0
15	0	0	0	1

To analyze the probability of occurrence for each random number within the generated sequence, the occurrence of each possible random number within the generated sequence was found, and a histogram plot was produced, as shown in **Fig. 7**. Given the number of generated numbers, the ideal average occurrence for each number should be $\frac{250}{16} = 15.625$ times. For the generated sequence, the average occurrence was found to be 15.06, which is close to the ideal case. It is worth noticing that the obtained average occurrence of each number gets closer to the ideal value as the length of the random sequence increases.

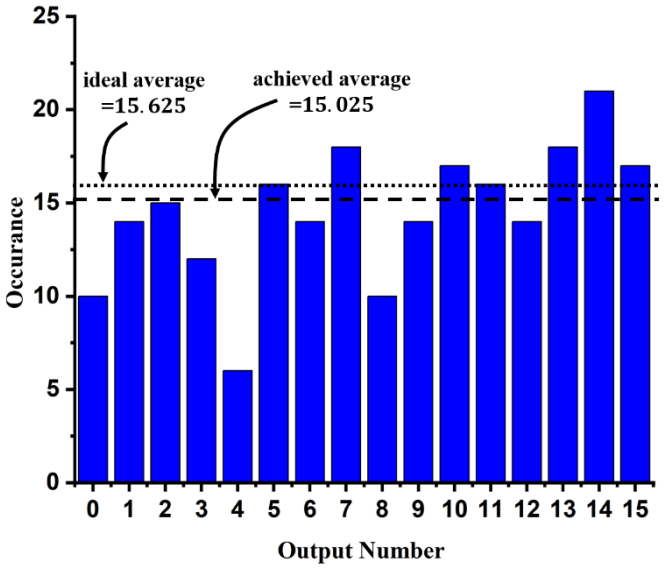

Figure 7. Distribution of the random numbers generated by the proposed system for the first 250 trials.

Next, a statistical analysis was performed on the generated sequence to test the randomness of the system and to compare the proposed system to another available random generator. This analysis is done by taking the auto-correlation function for the generated sequence. This function, when applied to a discrete-time process $X[n]$, is defined as the expected value of the product $X[n_1]\ X[n_2]$ as follows **(Miller and Childers, 2012)**:

$$R_{xx}[n_1, n_2] = E\big[X[n_1]\ X[n_2]\big] = \iint_{-\infty}^{\infty} x_1 x_2 f_{X_1, X_2}(x_1, x_2; t_1, t_2)\ dx_1\ dx_2 \qquad (1)$$

This function is intended to measure the degree of correlation between the samples of a random process as a function of the duration between the time instants at which the two samples occur, which is referred to as the *lag* of the operation. Therefore, the function should maximize at the origin and decay rapidly as the lag increases.

Fig. 8-a shows the normalized autocorrelation function calculated for a random sequence generated by the proposed system. As can be seen, the resulting function maximizes at $lag = 0$ and drops to smaller values (less than 11 % of its peak value) as the lag increases, which illustrates the high degree of randomness of the generated sequence. The bound for this process is 0.1096.

To compare the random process with another process, the same function was calculated for the uniformly-distributed random number generator used in MATLAB, which is a fundamental building block in many built-in functions in this software package and is extensively used in most simulation and modeling of communication systems. This process artificially generates a uniformly distributed random number within the interval [0,1]. According to the software documentation, the generation of these numbers is based on a pseudo-random process that is deterministic and repeatable **(Mathworks, 2021)**. Fig. 8-b shows the normalized autocorrelation function of a MATLAB-generated random sequence of the same length as in the one generated by the proposed algorithm. It is clear that the result is very similar to the one obtained from the generated proposed algorithm. However, the system proposed in this paper is much simpler than building the software package's process. In addition, the proposed system provides a true random

number generation that is non-repeatable, which the software cannot provide unless it is connected online to a true random integer number generator, that is generated based on atmospheric noise **(Mathworks, 2021)**.

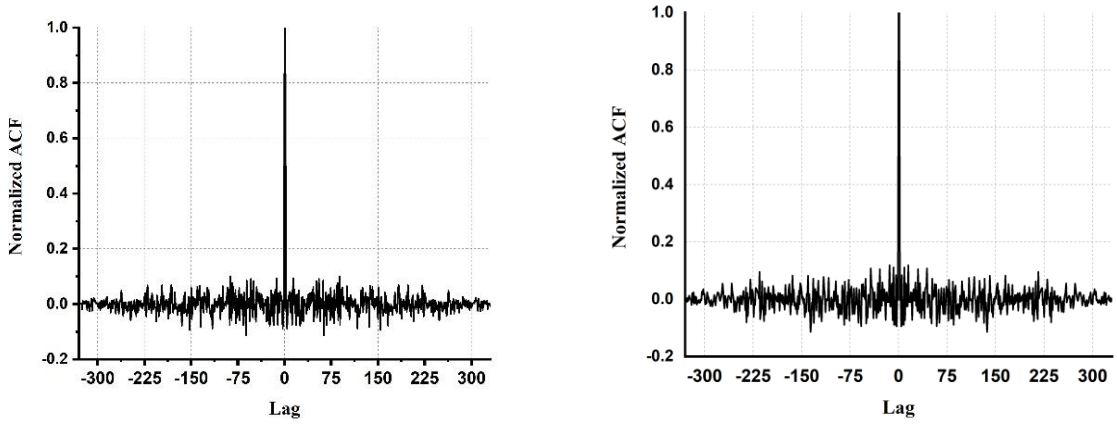

Figure 8: Resulted autocorrelation function (ACF) for a 300 number-long sequence resulted from (a) the proposed system and (b) a MATLAB-based random number generator.

Random number generation speed is essential for numerous applications, such as cryptography and high-speed communication systems. This issue can be addressed by modifying the proposed system to increase the number of bits generated per second. For example, three 4-bit counter circuits are used parallel with the same pulse generator connected to them to generate twelve bits. Moreover, both randomness and speed can be improved by using any pseudo-random number generator circuits (such as a linear feedback shift register or a chaotic generator) and utilizing the electrical connector bounce phenomenon to provide the seed for the sequences **(Miller and Childers, 2012)**.

5. CONCLUSIONS

The generation of True Random Numbers (TRNs) is a vital process in numerous fields in science and engineering, such as cryptography, system modeling, and weather forecast. This paper aims to utilize the contact bounce phenomenon that exists in electromechanical switches to construct a simple and affordable TRN generator. The proposed system has been designed such that an automatically-triggered relay is connected to the clock input of a counter, which produces the random number based on the fluctuations in the input signals during the contact bounce period. Next, the proposed TRN generator has been implemented to produce a sequence of 4-bit random numbers. Statistical investigation of the produced sequences revealed a high degree of randomness with lower complexity than other TRN generators, such as chaotic oscillations, semiconductor lasers, and magnetic tunnel junctions. Also, applying the autocorrelation function to the experimentally produced sequence has shown very similar behavior to the one produced by Matlab

without the complications associated with random number production in the software package and other TRNG systems.

ACKNOWLEDGEMENT
The authors would like to thank Mr. Omar Ghanim Murad and Mr. Mustafa Mahdi Mustafa for their assistance in performing the practical tests of the system.

NOMENCLATURE

n = time step, dimensionless
R_{xx} = autocorrelation function, dimensionless
$X[n]$ = discrete-time process, dimensionless

ABBREVIATIONS

FPGA: Field Programmable Logic Array
PRNG: Pseudo-random Number Generator
RNG: Random Number Generator
TRNG: True Random Number Generator
TTL: Transistor Transistor Logic

REFERENCES

Ahmad, Afaq, Mufeed Juma Al-Mushrafi, and Samir Al-Busaidi, 2002. Design and study of a strong crypto-system model for e-commerce, In Proceedings of the 15th international conference on Computer communication, pp. 619-630. 2002.

Barkan, Paul, 1967. "A study of the contact bounce phenomenon." IEEE Transactions on Power Apparatus and Systems 2, 231-240.

Bonilla, Luis L., Mariano Alvaro, and Manuel Carretero 2016. Chaos-based true random number generators, Journal of Mathematics in Industry 7, no. 1: 1-17.

Buchovecka, S., Lorencz, R., Kodýtek, F. and Bucek, J., 2016. True random number generator based on ROPUF circuit. In 2016 Euromicro Conference on Digital System Design (DSD) (pp. 519-523). IEEE.

Milic, Dominika Crnjac, and Ljiljanka Kvesic, 2008. Role of Random Numbers in Simulations of Economic Processes, Interdisciplinary Management Research 4: 562-570.

Deoras, Ameya 2016. True Random Number Generator, MathWorks File Exchange, [online]. Available at: <https://www.mathworks.com/matlabcentral/fileexchange/21353-true-random-integer-generator>. Accessed on 10/01/2022.

Diaconis et al., 2007. Diaconis, Persi, Susan Holmes, and Richard Montgomery. Dynamical bias in the coin toss, SIAM Review 49, no. 2: 211-235.

Dorrendorf, Leo, Zvi Gutterman, and Benny Pinkas, 2009. Cryptanalysis of the random number generator of the windows operating system, ACM Transactions on Information and System Security (TISSEC) 13.1: 1-32.

Hajtó, Dániel, Ádám Rák, and György Cserey 2019. Robust memristor networks for neuromorphic computation applications, Materials 12, no. 21: 3573.

Jiang, Hao, Daniel Belkin, Sergey E. Savel'ev, Siyan Lin, Zhongrui Wang, Yunning Li, Saumil Joshi, 2017. A novel true random number generator based on a stochastic diffusive memristor, Nature communications 8, no. 1: 1-9.

Ksiazkiewicz, Andrzej, Grzegorz Dombek, Karol Nowak, and Jerzy Janiszewski., 2019. Electrodynamic Contact Bounce Induced by Fault Current in Low-Voltage Relays, Energies 12, no. 20: 3926.

Lee, Hochul, Farbod Ebrahimi, Pedram Khalili Amiri, and Kang L. Wang, 2017]. Design of high-throughput and low-power true random number generator utilizing perpendicularly magnetized voltage-controlled magnetic tunnel junction, AIP Advances 7, no. 5: 055934.

Mano, M. Morris, 2017]. Digital logic and computer design. Pearson Education India.

Mathworks, 2021. MathWorks Support Documentation. Controlling Random Number Generation, Matlab 2021 documentation. [online]. Available at: https://www.mathworks.com/help/matlab/math/controlling-random-number-generation.html. Accessed on 10/01/2022.

Meyer, Charles R., Chris S. Renschler, and Roel C. Vining, 2008. Implementing quality control on a random number stream to improve a stochastic weather generator, Hydrological Processes: An International Journal 22, no. 8: 1069-1079.

Miller, Scott, and Donald Childers, 2012. Probability and random processes: With applications to signal processing and communications. Academic Press.

Ren, Wanbin, Yuan He, Jianbing Jin, and Sida Man, 2016. Investigations of the contact bounce behaviors and relative dynamic welding phenomena for electromechanical relay, Review of Scientific Instruments 87, no. 6: 065111.

Rossetti, Manuel D, 2015. Simulation modeling and Arena. John Wiley and Sons.

Slade, Paul G., 2017. *Electrical contacts: principles and applications*. CRC Press.

Stipčević, Mario, and Çetin Kaya Koç, 2014. True random number generators, Open Problems in Mathematics and Computational Science. Springer, Cham, 275-315.

Vinod, M., S. R. Devadasan, D. Rajanayagam, D. T. Sunil, and V. M. M. Thilak., 2018. Theoretical and industrial studies on the electromechanical relay, International Journal of Services and Operations Management 29, no. 3: 312-331.

Yu, F., Li, L., Tang, Q., Cai, S., Song, Y., and Xu, Q., 2019. A survey on true random number generators based on chaos. Discrete Dynamics in Nature and Society.

Zhang, Xu, Zhe Zheng, Wanbin Ren, and Zhefeng Zhou, 2018. An experimental investigation of dynamic welding mechanism of contacts used in low current switching devices, In 2018 IEEE Holm Conference on Electrical Contacts, pp. 488-494. IEEE.

Zhang, Xu, Wanbin Ren, Zhe Zheng, and Shujuan Wang, 2019. "Effect of electrical load on contact welding failure of silver tin oxide material used in DC electromechanical relays, IEEE Access 7: 133079-133089.