IUST
Iran University of Science and Technology

Iranian Journal of Electrical and Electronic Engineering

Journal Homepage: ijeee.iust.ac.ir

Research Paper

Low Power True Random Number Generator through Ring Oscillator for IoT and Smart Card Applications

G. Morankar*(C.A.)

Abstract: Tremendous developments in integrated circuit technology, wireless communication systems, and personal assistant devices have fuelled growth of Internet of Things (IoT) applications and smart cards. The security of these devices completely depends upon the generation of random and unpredictable digital data streams through random number generator. Low quality, low throughput, and high processing time are observed in software-based pseudo-random number generator due to interrelated data or programs and serial execution of codes respectively. In this paper, FPGA implementation of low power true random number generator through ring oscillator for IoT applications and smart cards is presented. Ring oscillators based on higher jitter and sampling techniques were exploited to present true random number generator. Further statistical parameters of the generated data streams are enhanced through feedback mechanism and post-processing technique. The presented true random number generator technique does not depend on the characteristics of a particular FPGA. The presented technique consumes low power, requires low hardware footprints and passes the entire National Institute of Standards & Technology (NIST) 800-22 statistical test suite. The presented low power and area true random number generator with enhanced security through post-processing unit may be applied for encryption/decryption of data in IoT and smart cards.

Keywords: Random Number Generator, Ring Oscillator, Jitter, Low Power, IoT, Smart Cards.

1 Introduction

INTERNET of Things and smart cards finds application in all the areas of consumer electronics, banking and non-banking financial transactions, health care, toll collections, wireless communication systems, etc. A tremendous development in integrated circuit technology, wireless communication systems and personal assistant devices has fuelled growth of IoT applications and smart cards. The complete success of IoT and smart cards entirely depends upon the low power and area requirements of the devices and systems without sacrificing on the security of the data or information. Also, extended use of smart cards such as memory cards (SD cards) is more vulnerable to attacks and has elevated security concerns to satisfy the requirements of security of information. The security of these devices completely depends upon the generation of random and unpredictable digital data streams through random number generator [1-3]. It is essential to employ the best quality random number generator that ensures the security of data in IoT and smart cards applications without sacrificing the intended properties such as low power and area [4-6].

Random number generator finds application in cryptography, communication, industrial instrumentation and measurements, Monte Carlo simulations, random padding, mobile and computer games, laboratory testing, to determine optimal input random variables for applying evolutionary algorithms for characteristics of the network [7], etc. It employs the random process through a source of unpredictability such as noise which is extracted, amplified and digitized. Low quality, low throughput and high

Iranian Journal of Electrical and Electronic Engineering, 2021.
Paper first received 27 June 2020, revised 29 October 2020, and accepted 06 November 2020.
* The author is with the Department of Electronics Engineering, Shri Ramdeobaba College of Engineering & Management, Nagpur, Maharashtra, India.
E-mails: morankarg@rknec.edu.
Corresponding Author: G. Morankar.
https://doi.org/10.22068/IJEEE.17.3.1914

processing time are observed in software based pseudo-random number generators (PRNGs) due to interrelated data or programs and serial execution of codes respectively [8, 9]. Field programmable gate array (FPGA) has appeared to be advantageous for the implementation of true random number generator (TRNGs) due to its inherent structure that supports parallelism and bitwise operations. Metastability and jitter offer the desired randomness in FPGA dues its resource-restricted environment that does not include analog blocks and digital to analog converters on board. Also, it is preferred to design TRNGs which does not depend on the technology employed for the manufacturing of FPGA and allows portability over all devices. Thus the design of TRNGs in FPGA is more challenging which needs to adhere to low power and area requirements without sacrificing on statistical quality of generated numbers.

In this paper, FPGA implementation of low power true random number generator through ring oscillator for IoT applications and smart cards is presented. Ring oscillators based on higher jitter and sampling techniques are exploited to present TRNGs. Further statistical parameters of the generated data stream are enhanced through the feedback mechanism and post-processing technique. The presented TRNG technique does not depend on the characteristics of a particular FPGA. The presented technique consumes low power, requires low hardware footprints and passes the entire NIST 800-22 statistical test suite. The presented low power and area TRNG with enhanced security through post-processing may be applied for encryption/decryption of data in smart cards and IoT. Mathematical model, modeling through VHDL, implementation on Virtex 5 XC5VLX20T & Spartan 3E Board FPGA devices, and experimental results are discussed.

2 Related Work

In [1] a true random number generator (TRNG) using modern FPGAs through utilizing dynamic partial reconfiguration (DPR) capabilities for varying the digital clock manager (DCM) modeling parameters was proposed. DPR facilities are available with Xilinx clock management tiles (CMTs) that contain a dynamic reconfiguration port (DRP) which allows DPR to be accomplished. DPR can be performed by much simpler means in Xilinx but not in an Altera FPGAs that clearly advocates importability. Threats that may originate due to the availability of DPR at IoT nodes were analyzed and a possible solution technique based on Physically Unclonable Function (PUF) circuits was proposed in [9]. In [10] numerous methods to implement PRNGs in FPGAs using oscillator rings based on inverters were analyzed and its sampling frequency was determined. It consists of an oscillator ring with 3 NOT gates which achieved a sampling frequency of 300 MHz, such 16 oscillator rings were realized. Thus true randomness was observed in the circuit using 48 inverters, 17 D flip flops (DFFs), 31 XORs, and about 100 routing resources. Numerous researchers have proposed LUT and shift registers based TRNG with the help of first in first out shift register (FIFO SR), parallel in parallel out shift register (PIPO SR) quadratic residue block and XOR gates connection block [11-13]. It was demonstrated that LUT-FIFO TRNG uses RAM blocks available in FPGA to generate high quality random bit streams. Also, LUT-SR TRNGs do not generate good quality random bitstreams as compared to LUT-FIFO TRNGs [3]. Chaotic non-linear systems exhibit random-like behaviors that can be exploited for the generation of random numbers but most of these are found to be insecure [14-16]. Two issues in chaotic TRNGs were addressed in [17] first is the effects of finite precision in all processors that causes a chaotic system to degenerate into a periodic function and second one is the lack of sufficient evaluation criteria. However, these generators are slower and complex in structure. A random generator based on the logistic was proposed in [18], so as to improve statistical properties through dynamically changes in its chaotic parameter. It clearly demonstrates that better randomness than other generally implemented PRNGs / TRNGs such as a 32-order linear feedback shift register (LFSR) or a FIFO. However, it is observed that the complexity involved is much more which results in lower throughput and higher power dissipation [19, 20]. Therefore chaos-based TRNGs / PRNGs may not be applied in limited power applications such as IoT and smart cards. It is very much essential to propose low power and reliable TRNGs/PRNGs that satisfies all security concerns and protects information/data.

3 Proposed True Random Number Generator

TRNG circuit using beat frequency detection (BFD) was demonstrated in [21] that consists of high and low-frequency oscillators X and Y respectively, D flip flop and counter depicted in Fig. 1. Beat frequency interval between two oscillators high-frequency X and low-frequency Y, the output of D flip flop is logic ‘1’ for random intervals. Therefore the counter generates random bitstreams due to random beat frequency internals depicted in Fig. 1. However, the BFD circuit,

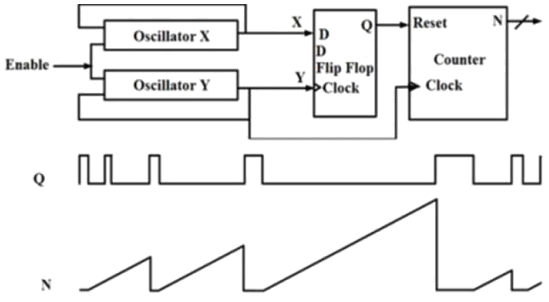


Fig. 1 TRNG BFD architecture [21].

if implemented using FPGA may result in variable count due to placement and routing of oscillators with an equal number of inverters. Randomness completely depends upon the difference in the frequency of oscillators X & Y and resultant jitter. Ring oscillators based on inverters are most suitable for low power and area applications and are robust to 1/f noise [4].

3.1 Architecture for Proposed TRNG

The proposed architecture consists of two ring oscillators X and Y with center frequencies selected arbitrarily, XOR gates, D flip flop, synchronous counter, and post-processing unit depicted in Fig. 2.

The center frequencies of the oscillator X and Y can be selected so that they are not integer multiple of the clock frequency.

$$F_{clock} \neq m.F_{ocsx} \quad (1)$$

$$F_{clock} \neq n.F_{ocsy} \quad (2)$$

$$F_{oscx} \gg F_{ocsy} \quad (3)$$

where F_{clock}, F_{ocsx}, and F_{ocsy} are frequencies of the input clock, oscillator X and oscillator Y signals respectively and integer m, n = 0, 1, 2, 3, 4, …

The schematic exploited to implement oscillator X is depicted in Fig. 3. It consists of series and parallel connections of programmable inverters through EXOR gate. The output of the oscillator can be periodically changed through a control pin named alt. The simulated output of the oscillator X using Xilinx ISE 14.2 is illustrated in Fig. 4. It clearly shows the change in the output of the oscillator X by changing the logic at alt pin. It also demonstrates that after placement and routing of the oscillator X in FPGA, the output is random. It is applied as a seed value to the D flip flop and counter to generate a random sequence of better statistical quality.

The use of series and parallel connections of programmable inverters through EXOR gate as compared to directly connected serial inverters results in the following merits. Firstly, it consists of several loops which result in wired OR logic that brings into metastability. Secondly, placement and routing of these EXOR gates every time with few differences may result in changes in oscillation. Finally, larger jitter may be achieved through placement and routing which enhances randomness. Asynchronous modulo 8 counter using T flip flop was implemented to reduce the clock frequency at the output of oscillator Y. D flip flop resets the 32-bit synchronous counter randomly whenever the output of oscillator X is logic 1 and at positive edge of the clock signal obtained from asynchronous counter alias oscillator Y. Asynchronous counter was selected over the synchronous counter since it generates glitches that add to the randomness by resetting D flip flop and counter. Post-processing unit is a post-digital processor that improves the statistical properties of the random number and enhances randomness. The architecture of the post-processing unit is depicted in Fig. 5, which consists of 32-bit linear feedback shift registers and four nonlinear functions. It is used to enhance unpredictability and reduces the correlation coefficient

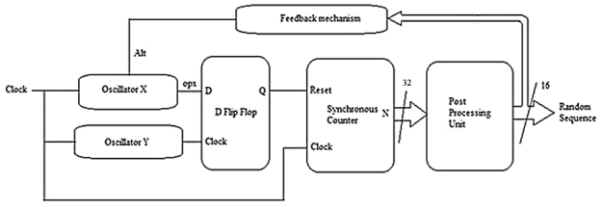


Fig. 2 Proposed PRNG architecture.

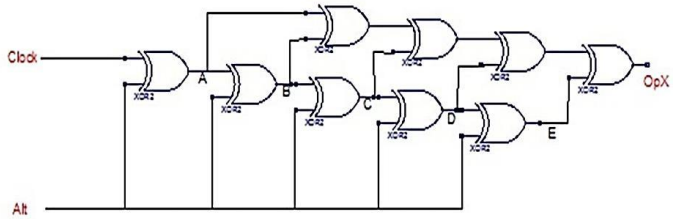


Fig. 3 Schematic of oscillator X through programmable EXOR gate.

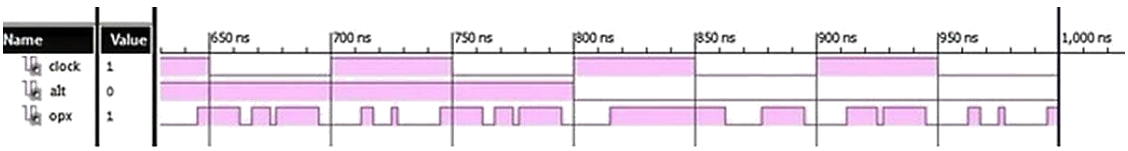
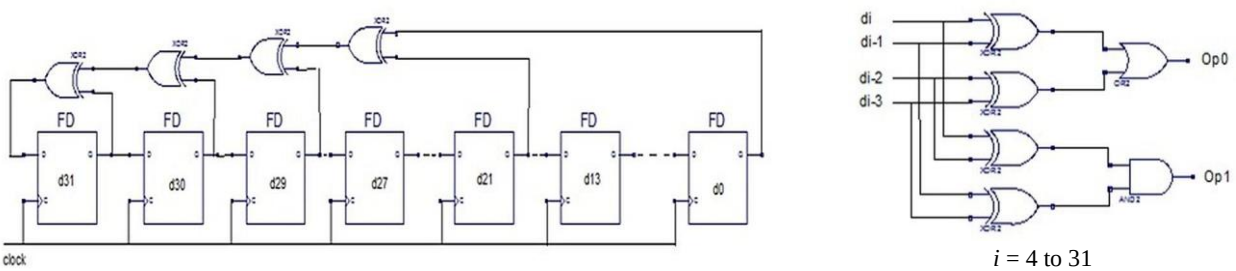


Fig. 4 Output of oscillator X that can be changed using alt.



i = 4 to 31

Fig. 5 Architecture of post processing unit.

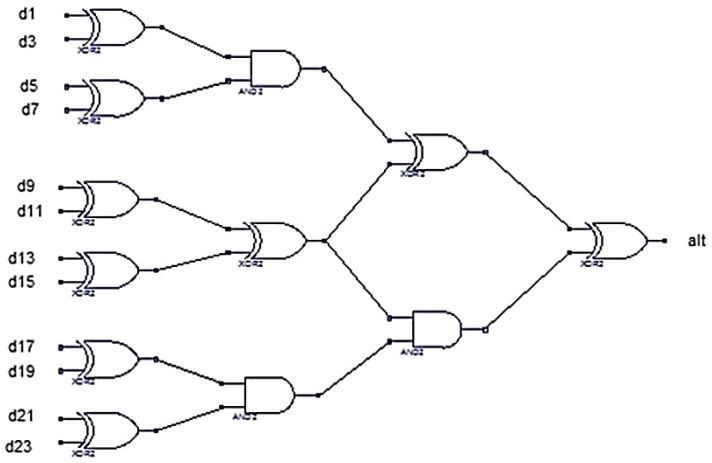


Fig. 6 Feedback mechanism.

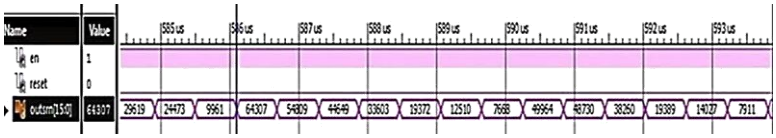


Fig. 7 Simulation result of proposed TRNG.

Fig. 8 Test setup.

Table 1 Resource utilization summary.

Logic utilization		Used	
		Virtex 5	Spartan 3E board
Number of slice registers		20	20
Number of slice LUTs		29	29
Number of fully used LUT-FF pairs		20	20
Number of bonded IOBs		18	18
Power dissipation at 50 MHz [mW]	Static	112.5	108.25
	Dynamic	43.75	42.55
	Total	156.25	150.8

of the generated output random numbers. Furthermore, the feedback signal from the generated random number is obtained to add metastability in oscillator X. Feedback mechanism function is depicted in Fig. 6. Entire architecture was directed towards obtaining low power dissipation for IoT and smart card applications without sacrificing on security and statistical properties of the sequence. The functionality of the design was demonstrated and verified using FPGA. Further, it forms the basis of ASIC implementation which can be undertaken in the future with adjoining circuitry.

4 Results and Discussion

The functional or logical verification of the proposed TRNG was simulated using Xilinx ISE 14.2 software platform using VHDL and implemented on Spartan 3E Board XC3S500E-4FG320 and Virtex 5 XC5VLX20T FPGA devices. The random number generated after simulation of the proposed TRNG architecture is demonstrated in Fig. 7 and test setup is shown in Fig. 8. It was observed that hardware resources required for Virtex 5 and Spartan 3E board were almost the same. Thus the total hardware requirement in terms of FPGA resources was comparatively less and bears low hardware footprints. Moreover variations in placement and routing of oscillator X and feedback mechanism generates random bit streams in FPGA that can be most useful in IoT and smart cards devices. Table 1 depicts the resource utilized by the FPGA devices and its corresponding power dissipation. The 16-bit random numbers generated with mean and entropy of 32355 and 15.1 respectively. The NIST 800-22 [22] statistical performance of the TRNG from the implemented circuit is illustrated in Table 2. An experimental result demonstrates randomness properties of the TRNG circuit low hardware resources and power dissipation. The comparison of proposed TRNG with recently implemented TRNG using coherent sampling with self-timed rings [2], beat frequency oscillators [23], and modified ring oscillators [24] is illustrated in Table 3. Hardware complexities are categories as low complexities: easily implemented, medium

Table 2 NIST 800-22 statistical parameters summary.

Parameters at 50 MHz	Values
Max count	64307
Min count	403
FFT	0.365
Normalized Frequency	0.91
Mean	32355
Entropy	15.1
Overlapping template	0.21
Non-overlapping template	0.79
Random excursions	0.928

Table 3 TRNG comparison.

Parameters	[2]	[23]	[24]	Our work
LUTs	32	50	160	29
Registers	48	33	19	20
Approximate entropy	0.998	0.931	0.9825	0.973
Complexities	Medium	Low	Low	Low
Portability	Yes	Yes	Yes	Yes
Tunability	No	Yes	No	Yes
Throughput [Mb/s]	4	5- 25	4	5 – 25

complexities: uses PLL, DCM, analog blocks and high complexities: manual placement and routing [1].

5 Conclusion

FPGA implementation of low power true random number generator through ring oscillator for IoT applications and smart cards is presented. Its implementation using Spartan 3E Board and Virtex 5 FPGA devices is demonstrated for verification and authenticity of the circuit. Series and parallel combination of programmable inverters through EXOR gate is proposed that adds to metastability, jitter and randomness. Feedback mechanism from the generated random number enhances metastability in oscillator X and minor changes in placement and routing further aids randomness among various devices. The presented technique consumes low power, requires low hardware footprints and passes the entire NIST 800-22 statistical test suite. Further, it forms the basis of ASIC implementation which can be undertaken in the future with adjoining circuitry.

References

[1] A. P. Johnson, R. S. Chakraborty, and D. Mukhopadyay, "An improved DCM-based tunable true random number generator for Xilinx FPGA," *IEEE Transactions on Circuits and Systems—II: Express Briefs*, Vol. 64, No. 4, pp. 452–456, 2017.

[2] H. Martin, P. Peris-Lopez, J. E. Tapiador, and E. San Millan "A new TRNG based on coherent sampling with self-timed rings," *IEEE Transactions on Industrial Informatics*, Vol. 12, No. 1, pp. 91–100, 2016.

[3] R. Justina, B. K. Mathew, and S. Abe, "FPGA implementation of high quality random number generator using LUT based shift registers," *Procedia Technology*, Vol. 24, pp. 1155–1162, 2016.

[4] D. Liu, Z. Liu, L. Li, and X. Zou, "A low-cost low-power ring oscillator-based truly random number generator for encryption on smart cards," *IEEE Transactions on Circuits and Systems—II: Express Briefs*, Vol. 63, No. 6, pp. 608–612, 2016.

[5] S. V. Suresh and W. P. Burleson, "Entropy and energy bounds for metastability based TRNG with lightweight post-processing," *IEEE Transaction on Circuits and Systems I*, Vol. 62, No. 7, pp. 1785–1793, 2015.

[6] G. K. Balachandran and R. E. Barnett, "A 440-nA true random number generator for passive RFID tags," *IEEE Transaction on Circuits Systems I*, Vol. 55, No. 11, pp. 3723–3732, 2008.

[7] F. Farabi F, M. R. Mosavi, and S. Karami, "Optimal choice of random variables in D-ITG traffic generating tool using evolutionary algorithms," *Iranian Journal of Electrical and Electronics Engineering*, Vol. 11, No. 2, pp. 101–108, 2015.

[8] S. Saab, J. Hobeika, and I. Ouaiss, "A novel pseudorandom noise and band jammer generator using a composite sinusoidal function," *IEEE Transaction on Signal Processing*, Vol. 58, No. 2, pp. 535–543, 2010.

[9] A. P. Johnson, R. S. Chakraborty, and D. Mukhopadhyay, "A PUF-enabled secure architecture for FPGA-based IoT applications," *IEEE Transaction on Multi-Scale Computing System*, Vol. 1, No. 2, pp. 110–112, 2015.

[10] X. Xu and Y. Wang, "High speed true random number generator based on FPGA," in *International Conference on Information Systems Engineering (ICISE)*, pp. 18-21, 2016.

[11] D. B. Thomas and W. Luk, "The LUT-SR family of uniform random number generators for FPGA architectures," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Vol. 21, No. 4, pp. 761–770, 2013.

[12] D. B. Thomas and W. Luk, "High quality uniform random number generation using LUT optimized state-transition matrices," *Journal of VLSI Signal Processing*, Vol. 47, No. 1, pp. 77–92, 2007.

[13] A. Marghescu, P. Svasta, and E. Simion, "High speed and secure variable probability pseudo/true random number generator using FPGA," in *IEEE 21st International Symposium on Design and Technology in Electronic Packaging*, pp. 323–328, 2015.

[14] L. G. de la Fraga, E. Torres-Pérez, E. Tlelo-Cuautle, and C. Mancillas-López, "Hardware implementation of pseudo-random number generators based on chaotic maps," *Nonlinear Dynamics*, Vol. 90, pp. 1661–1670, 2017.

[15] L. M. Barakat, A. S. Mansingka, A. G. Radwan, "Hardware stream cipher with controllable chaos generator for colour image encryption," *IET Image Processing*, Vol. 8, No. 1, pp. 33–43, 2014.

[16] F. L. Wang, "A new pseudo-random number generator and application to digital secure communication scheme based on compound symbolic chaos," *Acta Physica Sinica*, Vol. 60, No. 11, p. 110517, 2011.

[17] Y. Liu and X. Tong, "Hyperchaotic system-based pseudorandom number generator," *IET Information Security*, Vol. 10, No. 6, pp. 433–441, 2016.

[18] M. Garcia-Bosque, A. Pérez-Resa, and C. Sánchez-Azqueta, "Chaos-based bitwise dynamical pseudorandom number generator on FPGA," *IEEE Transactions on Instrumentation and Measurement*, Vol. 68, No. 1, pp. 291–293, 2019.

[19] U. Guler and S. Ergun, "A high speed, fully digital IC random number generator," *AEU International Journal of Electronics Communication*, Vol. 66, No. 2, pp. 143–149, 2012.

[20] M. Garcia-Bosque, A. Pérez-Resa, C. Sánchez-Azqueta, C. Aldea, and S. Celma, "A new technique for improving the security of chaos based cryptosystems," in *IEEE International Symposium on Circuits and Systems (ISCAS)*, Florence, Itlay, pp. 1–5, 2018.

[21] Q. Tang, B. Kim, Y. Lao, K. K. Parhi, and C. H. Kim, "True random number generator circuits based on single- and multi-phase beat frequency detection," in *Proceedings of IEEE Custom Integrated Circuits Conference*, pp. 1–4, 2014.

[22] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, E. Barker, S. Leigh, M. Levenson, M. Vangel, D. Banks, A. Heckert, J. Dray, and S. Vo, "A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications," *National Institute of Standards & Technology (NIST)*, Apr. 2010, [Online]. Available: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-22r1a.pdf

[23] G. Morankar, "True random number generator through beat frequency oscillators in FPGA," *Helix Journal*, Vol. 8, No. 4, pp. 3442–3447, 2018.

[24] M. A. Şarkışla and S. Ergün, "An area efficient true random number generator based on modified ring oscillators," in *IEEE Asia Pacific Conference on Circuits and Systems (APCCAS)*, Chengdu, pp. 274–278, 2018.

G. Morankar received her Ph.D. in Electronics Engineering from RTMN University, Nagpur, India, and Post-Graduation M.E. in Electronics Engineering from SRTM University Nanded, India. In 2010 she joined as Assistant Professor in the Department of Electronics Engineering Shri Ramdeobaba College of Engineering & Management, Nagpur, India. Her research interest includes FPGA implementation of signal and image processing algorithms, deep learning algorithms and techniques. She has published many papers in national/international conferences and journals of repute and has two patents to her credit.