Dynamic control of entropy and power consumption in TRNGs for IoT applications

Honorio Martin Gonzalez[a)], **Enrique San Millan Heredia, and Luis Entrena Arrontes**

Department of Electronic Technology, Universidad Carlos III de Madrid, 28911 Leganés, Madrid, Spain

a) *hmartin@ing.uc3m.es*

Abstract: In this article we present a study about how to obtain a trade-off between two important metrics for IoT systems-Security vs Power Consumption. More specifically, we have studied how the min-entropy of two True-Random number generators can be adjusted dynamically in order to reduce the power consumption while guaranteeing the integrity of the system. To that end, we make use of some statistical tests that are typically used to measure the quality of the RNGs. Clock-gating and enable-gating are the selected techniques to reduce the power consumption.

Keywords: IoT, TRNGs, on-line test, power consumption, energy efficiency, hardware security, clock gating

Classification: Integrated circuits

References

[1] D. Jones: "True random number generators for truly secure systems," Synopsys white paper (2015).

[2] K. Wallace, *et al.*: "Toward sensor-based random number generation for mobile and iot devices," IEEE Internet Things J. **3** (2016) 1189 (DOI: 10.1109/JIOT.2016.2572638).

[3] T. T. Zhu, *et al.*: "Error-resilient integrated clock gate for clock-tree power optimization on a wide voltage iot processor," IEEE Trans. Very Large Scale Integr. (VLSI) Syst. **25** (2017) 1681 (DOI: 10.1109/TVLSI.2017.2652482).

[4] A. Rukhin, *et al.*: A statistical test suite for random and pseudorandom number generators for cryptographic applications. *Technical Report* (2001).

[5] W. Schindler and W. Killmann: "Evaluation criteria for true (physical) random number generators used in cryptographic applications," CHES'02 (2003) 431 (DOI: 10.1007/3-540-36400-5_31).

[6] E. Barker and J. Kelsey: *Recommendation for the Entropy Sources Used for Random Bit Generation* (2012).

[7] P. Haddad, *et al.*: "A physical approach for stochastic modeling of tero-based TRNG," CHES'15 (2015) 357.

[8] V. Fischer and D. Lubicz: "Embedded evaluation of randomness in oscillator based elementary TRNG," CHES'14 **8731** (2014) 527 (DOI: 10.1007/978-3-662-44709-3_29).

[9] B. Yang, *et al.*: "Total: Trng on-the-fly testing for attack detection using lightweight hardware," DATE'16 (2016) 127.

[10] P. Z. Wieczorek: "Lightweight trng based on multiphase timing of bistables,"

DOI: 10.1587/elex.14.20171157
Received November 21, 2017
Accepted December 4, 2017
Publicized December 21, 2017
Copyedited January 25, 2018

IEEE Trans. Circuits Syst. I, Reg. Papers **63** (2016) 1043 (DOI: 10.1109/TCSI.2016.2555248).

[11] H. Martin, *et al.*: "A new trng based on coherent sampling with self-timed rings," IEEE Trans. Ind. Informat. **12** (2016) 91 (DOI: 10.1109/TII.2015.2502183).

[12] K. Wold and C. H. Tan: "Analysis and enhancement of random number generator in FPGA based on oscillator rings," Int. J. Reconfig. Comput. **2009** (2009) 501672 (DOI: 10.1155/2009/501672).

[13] A. Cherkaoui, *et al.*: "A very high speed true random number generator with entropy assessment," CHES'13 **8086** (2013) 179 (DOI: 10.1007/978-3-642-40349-1_11).

[14] B. Yang, *et al.*: "On-the-fly tests for non-ideal true random number generators," ISCAS'15 (2015) 2017 (DOI: 10.1109/ISCAS.2015.7169072).

[15] I. Kuon and J. Rose: "Measuring the gap between fpgas and asics," IEEE Trans. Comput.-Aided Design Integr. Circuits Syst. **26** (2007) 203 (DOI: 10.1109/TCAD.2006.884574).

[16] J. Hussein, *et al.*: "Lowering power at 28 nm with xilinx 7 series devices," White Paper: 7 Series FPGAs (2015).

[17] S. Henzler: *Power Management of Digital Circuits in Deep Sub-Micron CMOS Technologies (Springer Series in Advanced Microelectronics)* (Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006).

[18] A. A. M. Bsoul, *et al.*: "An fpga architecture and cad flow supporting dynamically controlled power gating," IEEE Trans. Very Large Scale Integr. (VLSI) Syst. **24** (2016) 178 (DOI: 10.1109/TVLSI.2015.2393914).

[19] H. Martin, *et al.*: "Fault attacks on strngs: Impact of glitches, temperature, and underpowering on randomness," IEEE Trans. Inf. Forensics Security **10** (2015) 266 (DOI: 10.1109/TIFS.2014.2374072).

[20] A. Cherkaoui, *et al.*: "A self-timed ring based true random number generator," ASYNC'13 (2013) 99 (DOI: 10.1109/ASYNC.2013.15).

[21] P. Haddad, *et al.*: "On the assumption of mutual independence of jitter realizations in p-trng stochastic models," DATE'14 (2014) 1 (DOI: 10.7873/DATE.2014.052).

1 Introduction

An essential component of many security applications is randomness. Security and privacy rely on encryption, whose effectiveness depends on reliable True Random number generators (TRNGs). The necessity of random numbers in *Internet of Things* (IoT) devices [1, 2] stands out among the many applications that TRNGs have. Energy-efficiency optimization occupies an important position in the IoT application [3]. Devices designed for the IoT typically require a low-power consumption that not all the state-of-the-art TRNGs can fulfil.

The TRNGs that are usually integrated in IoT devices are generally used to generate session keys, nonces or padding plain-texts, which are critical parts in the security of the system. Hence the quality of the output should be tested during the operation to guarantee the necessary security level. The most extended way of testing the quality is checking the statistical distribution of the output using on-the-fly statistical test suites. There are well-known test suites such as NIST [4] or AIS31 [5], where not only the final output is checked, but also the raw TRNG

DOI: 10.1587/elex.14.20171157
Received November 21, 2017
Accepted December 4, 2017
Publicized December 21, 2017
Copyedited January 25, 2018

output (before post-processing) is evaluated. Following this trend, some on-the-fly tests that evaluate the entropy source have been presented in the literature [6]. Among the most interesting proposals, in [7, 8] the entropy rate is assessed by measuring some physical parameters of the entropy source. An interesting recent on-the-fly test based on the attack effects has been presented in [9]. The results from these tests are typically used to decide whether to stop the generation of random numbers when some degradation in the quality of the output is detected. In some cases, the TRNG is reset in order to try to solve the problems.

In this paper we present a new way of taking advantage of the test results to reduce power consumption. More specifically, we propose to use the results of on-line statistical tests in order to reach the minimum power consumption which is required to guarantee a certain level the security. Our method is not only capable of optimizing the power consumption for a given level of entropy, it can also increase the level of security when necessary. Our proposal allows to add as many oscillators as possible, and then control which ones are active and which ones are not. For this control we propose the well-known method of clock gating in order to select which oscillators will be contributing to the randomness of the output.

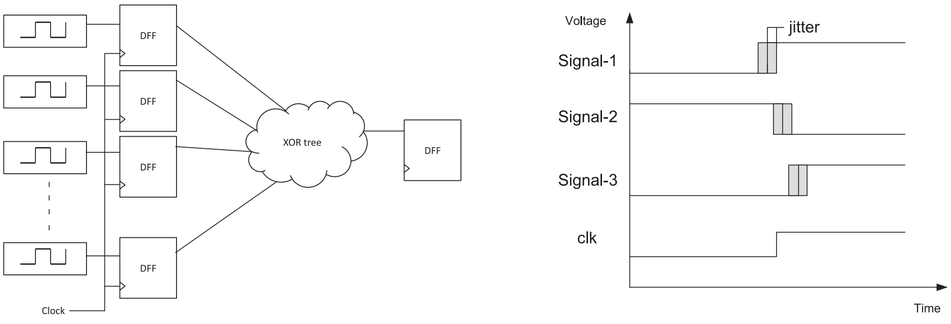
2 Background and motivation

In this section we present the different elements that will be involved in our experimental setup.

2.1 TRNGs

The increasing necessity of embedded security in a wide variety of applications has spawned a proliferation of TRNGs in the scientific literature [10, 11]. TRNGs that use sampling of jittery clocks as the entropy source stand out among these many proposals. The main advantages of these TRNGs are their flexibility and straightforward implementation.

In this type of TRNGs, the outputs of several high frequency oscillators (mainly generated by ROs, STRs and PLLs) are sampled by a clock and then collected through an XOR-tree to obtain a single bit at the output. A general scheme of this kind of generator is depicted in Fig. 1a. The principle of these generators is presented in Fig. 1b. If at least one of the high frequency signals is sampled on the jitter zone, the output will be random. It seems intuitive that higher



(a) (b)

Fig. 1. General architecture of a TRNG based on CS and Jitter sampling principle.

© IEICE 2018
DOI: 10.1587/elex.14.20171157
Received November 21, 2017
Accepted December 4, 2017
Publicized December 21, 2017
Copyedited January 25, 2018

frequency signals mean that it is more likely to sample one of them in the jitter zone. In this paper, we have selected two representative TRNGs that use this principle. The Wold et al. proposal [12] uses ROs as high frequency clocks. On the other hand, the Cherkaoui TRNG [13] generates the high frequency signals using an STR. The main difference between ROs and STRs is that STRs generate multiphase signals that can be finely controlled. We refer the reader to the original works for more details about stochastic models, implementations and statistical results of these generators.

2.2 On-line tests

Due to the key role of TRNGs in cryptography, it is necessary to guarantee their correct operation, and for that they are usually subjected to tests. First testing proposals only checked the statistical properties of the final output [4]. Lately, some proposals testing the entropy source [6, 8] or on-the-fly tests for attack detection [9] have been presented in the literature.

The quality of the generator output is closely related to the quality of the source of randomness and to the randomness extraction method employed. With this in mind, NIST recommended some design principles and requirements for the entropy sources used by TRNGs [6]. Among these recommendations, NIST has proposed 5 statistical tests intended for estimating the *min-entropy* of non identically distributed (non-IID) number generators. These kind of generators are those that may have dependencies in time and/or state that may result in an overestimation of entropy if typical test suites are used.

We have selected three of these tests due to their generality and straightforward implementation: The frequency test, the collision test and the partial collection test. It is convenient to execute several tests in order to not overestimate the minimum entropy. As in [14], we have made some assumptions in order to simplify the implementation of these tests. The TRNG will generate one bit each clock cycle, which means that the size of the output space is two (0 or 1).

2.3 Power saving techniques

One of the main drawbacks of FPGAs compared to ASICs is power consumption. According to some authors, FPGA implementations consume 12% more power on average [15]. In new technologies, FPGAs built on 28-nm technology, the dynamic power consumption is becoming more and more important [16]. Factors that contribute to the dynamic power are: capacitance, switching activity, voltage and clock frequency. Keeping these factors in mind, many researchers have proposed power saving techniques -some of them previously applied in ASICs-. Among these techniques the more significative ones are the clock gating technique and the reduction of switching activity [17]. Special attention deserve the techniques that dynamically controll power gating at run-time [18].

Clock gating is a power optimization technique that involves turning off the clock of blocks which do not contribute to the functionality of the system. This measure aims to avoid switching activity and dynamic power consumption within these blocks. Clock gating has been shown to be effective even if the inputs of the unused logic do not change. The main disadvantage of clock gating is related to the

DOI: 10.1587/elex.14.20171157
Received November 21, 2017
Accepted December 4, 2017
Publicized December 21, 2017
Copyedited January 25, 2018

voltage regulators speed. Most popular FPGA vendors have included in their tools different methods in order to make the use of this technique easier.

The reduction of switching activity is closely related to the clock gating technique and the input patterns applied to the circuit. In the case of circuits using self-oscillating logic that do not depend on a clock signal (e.g. ROs or STRs), the dynamic consumption due to switching activity becomes critical. A considerable amount of power can be saved if some of these self-oscillating elements can be switched off from time to time.

3 Experimental framework

In this section we present the implementations of the the two TRNGs and the online tests.

3.1 TRNGs implementations

We have implemented the two selected TRNGs [12] and [13] both in a Spartan-3E and in an Artix-7. We have selected these FPGAs in order to analyze the impact of the manufacturing technology (90 nm vs 28 nm) in the dynamic power consumption of the TRNG.

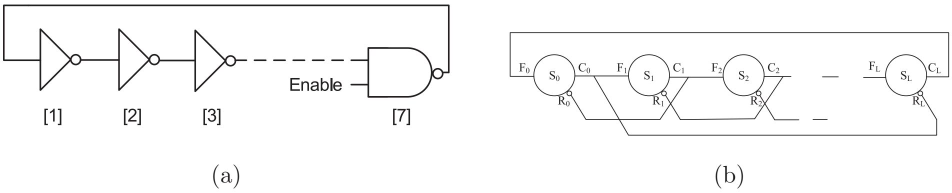
3.1.1 Wold et al. implementation

Following the scheme presented in Fig. 1a, we have implemented 64 ROs of 7 inverters each one. In order to switch off and on each RO, we have replaced in each RO an inverter by a NAND gate as depicted in Fig. 2a. In order to see the relation between the number of ROs and the *min-entropy* for the different tests, we will activate the ROs in groups of 8. The XOR-tree uses a ripple structure in order to avoid the effects of power and clock glitches as those presented in [19]. Finally, the sampling frequency has been set to 50 MHz.

3.1.2 Cherkaoui et al. implementation

As stated in the previous section, the Cherkaoui proposal makes use of an STR. The STR consists of L stages each composed of a Muller gate and an inverter. The general architecture of an STR is shown in Fig. 2b. The STR must be carefully configured in order to obtain a steady regime that assures the evenly space of events through the ring. This configuration is mainly related to the tokens and bubbles distribution through the ring [20]. We have implemented an STR that consists of 511 stages that have been modified in order to allow switching off and on the different stages. This modification involves the addition of an enable on each stage and a block that resets the STR state in order to set a optimum number of tokens and bubbles in each configuration. This modification allows us to activate different configurations (63,127,255 and 511 stages). As recommended in [13], we have created a hard macro in order to avoid bottleneck effects in the ring. As in the previous TRNG, we have used a ripple structure for the XOR-tree and a sampling frequency of 50 MHz.

DOI: 10.1587/elex.14.20171157
Received November 21, 2017
Accepted December 4, 2017
Publicized December 21, 2017
Copyedited January 25, 2018



(a) (b)

Fig. 2. Modified Ring Oscillator and Self-Timed Ring structure.

3.2 Tests implementation

We have implemented the three aforementioned tests (Frequency Test, Collision Test and Partial Collection Test) following the scheme presented in [14]. The dataset length will be 2^{15}. We have precomputed some cut-off values that will determine the different entropy levels in order to obtain a lightweight implementation.

The different blocks used in the implementation of these tests are shown in Fig. 3. The main counter controls the timing for the comparison with the cut-off values. The frequency test is implemented using an up/down counter in order to obtain a compact implementation. The partial collection test has been implemented using a 2-bit shift register and an XOR gate that generates the enable of a simple counter. Finally, a simple FSM that detects 3-bit collisions generates the enable of another counter.

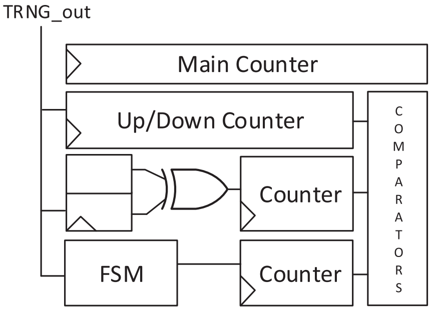


Fig. 3. Entropy test architecture.

3.3 Control system implementation and applications

The control block will control the enable/disable inputs of the modified ROs or STRs in each case. In addition to this, the control block will also manage the clock enable of the different configurations (clock gating). With this Clock enable, we can reduce the sampling activity in ROs/stages and also disable the clock of the registers of the ripple XOR tree if needed.

It is important to note that our implementation does not alter the pre-existing logic or clock placement, nor does create new clocks. As a result, the optimization does not affect timing [16] and therefore the TRNGs stochastic models can be used without any change.

The aim of this work is to take advantage of the dynamic control of entropy in order to save power or increase security (at the cost of higher power consumption). For this purpose we propose two operation modes that can be used for different applications:

DOI: 10.1587/elex.14.20171157
Received November 21, 2017
Accepted December 4, 2017
Publicized December 21, 2017
Copyedited January 25, 2018

Selectable security level: The first operation mode involves the definition of different security levels that can be controlled. This mode is intended for those TRNG applications that can make use of different entropy levels. For example, a TRNG that generates nonces and keys does not need the same level of security for both, some low/medium level entropy could be used for the first, but high level would be advised for the second. Note that the higher the security level, the higher the power consumption is going to be. We have selected for our experimental results three entropy levels - low, medium and high-. We have precomputed different cut-off values for the three levels and tests. The low-entropy boundaries correspond to a *min-entropy* range between 0,80 and 0,88. The medium-entropy level corresponds to the *min-entropy* range between 0,88 to 0,92. Finally, the high entropy level corresponds to *min-entropy* values higher than 0,92. Depending on the selected entropy level, a control block will select the number of active ROs/Stages, that are contributing to the final entropy, until the desired level is reached.

Minimum power consumption: The second operation mode consists on reducing the power consumption to the minimum while guaranteeing a *min-entropy* level during all the generation. The entropy level can be selected. In this case, the control block will enable/disable the ROs/stages in order to guarantee a selected *min-entropy* (for instance 0,92 for high level security) in all the tests. In the case that all the ROs/Stages are be activated but the *min-entropy* cannot be reached, an alarm will be set.

4 Experimental results

In order to verify the effectiveness of the proposed architectures, we have carried out different entropy and power measurements.

4.1 Entropy level

The level of security of the TRNG is measured in terms of the entropy level of the output of the TRNG. The ***min-entropy*** value is obtained from the results of the statistical tests, as a value between 0 and 1. In order to obtain an accurate *min-entropy* value, 1000000 of traces of 2^{15} bits for each configuration and FPGA were acquired. Raw data were gathered using a FIFO memory and a RS232 protocol at 115.2 Kb/s. The computation of the *min-entropy* has been carried out using an implementation of the aforementioned tests in Matlab.

In the following subsections we show the results of the three considered tests for each of the selected TRNGs.

4.1.1 Wold et al.

The relation between the *min-entropy* value and the number of active ROs for the different tests is presented in Fig. 4. The min-entropy has been obtained as the average of the aforementioned tests.

As expected, the entropy increases with the number of active ROs. Regarding the fabrication technology impact, for our two different FPGAs it is appreciable that the Spartan-3 obtains better results in terms of *min-entropy*. As the experiments

DOI: 10.1587/elex.14.20171157
Received November 21, 2017
Accepted December 4, 2017
Publicized December 21, 2017
Copyedited January 25, 2018

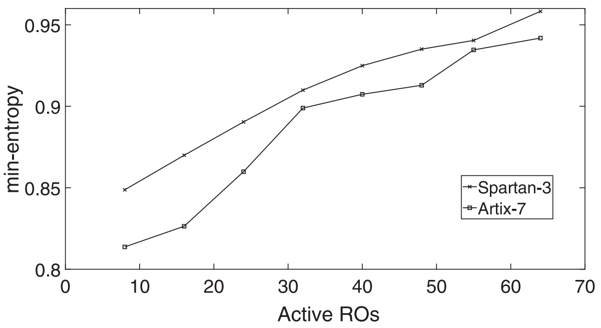


Fig. 4. *min-entropy* average for the entropy tests (Wold et al. proposal).

have been carried out trying to keep the same experimental environment (temperature, power supply, and sampling frequencies), this difference could be explained because of the increasing importance of flicker noise in new technologies. As stated in [21] the presence of the flicker noise, which typically has a Gaussian distribution, makes realizations of the jitter mutually dependent.

4.1.2 Cherkaoui et al.

In Fig. 5 is presented the relation between *min-entropy* and the number of active stages. As in the previous section, the min-entropy depicted is the average of the different tests.

With the results from the implementation of Cherkaoui et al. TRNG, we can observe once again that the *min-entropy* is directly related to the number of active stages. In this case, little difference can be appreciated between the two different FPGA technologies. This can be attributed to the robustness of the Cherkaoui et al. proposal.

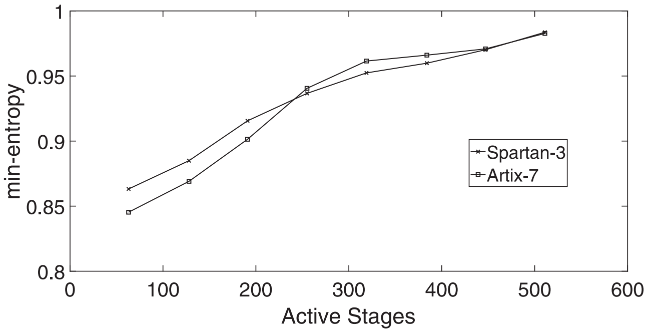


Fig. 5. *min-entropy* average for the entropy tests (Cherkaoui et al. proposal).

4.2 Power consumption

In this subsection we present the power consumption for the two selected TRNGs in the two different selected FPGA technologies, the Spartan-3 90 nm and The Artix-7 28 nm FPGAs. It is important to note that the following results include the implementation of the complete system (TRNG + Tests + Control).

The power consumption results presented for the Spartan-3 have been obtained by removing the V_{CCINT} jumper and measuring directly with a high quality

DOI: 10.1587/elex.14.20171157
Received November 21, 2017
Accepted December 4, 2017
Publicized December 21, 2017
Copyedited January 25, 2018

ammeter. We are aware of the fact that it is impossible to measure any system without effecting the system which is attempted to measure. But in this case, as combinational loops are synthesized in both TRNGs, the power estimator tools (e.g. Xpower Analyzer) are not an option.

The power consumption results for the Artix-7 have been obtained using the IP XADC. This FPGA includes circuitry for monitoring the voltage of the 5 Volt supply as well as the current consumed from this supply. We have configured Channels 1 and 9 of the XADC as unipolar inputs and then performed a simultaneous conversion of the two channels to receive digital values that can be used to compute the instantaneous power consumption. In the presented results, the power consumption used by this module has been discounted.

In order to avoid placement effects, we have tried four different locations for the TRNGs in the FPGAs. The results presented hereafter are the average results in all these different locations.

4.2.1 Wold et al.

The power consumption of Wold proposal for both FPGAs is shown in Table I. The dynamic power consumption for this TRNG is quite important because of the contribution of ROs. As expected, the power consumption increases with the number of ROs. In terms of power saving, we have obtained a difference between the minimum and maximum configurations (8 and 64 ROs) of 77% in terms of dynamic power consumption for the Spartan-3 and 64% for the Artix-7.

Table I. Power consumption for different configurations Wold et al.

Active Stages	Static	8	16	24	32	40	48	56	64
Spartan-3E (mW)	27,3 (58%)	31,8	33,7	35,9	38,1	41,0	43,5	45,3	47,0
Artix-7 (mW)	17,1 (52%)	22,5	24,1	25,4	26,9	28,0	29,7	31,0	32,4

4.2.2 Cherkaoui et al.

The same analysis has been done for the Cherkaoui et al. proposal. The power consumption of this implementation is shown in Table II for both FPGAs and several active stages configurations. The static power consumption for this TRNG is 69% for the Spartan-3 and 59% for the Artix-7. In terms of power saving, we have obtained a difference between the minimum and maximum configurations (63-511 Stages) of 69% in terms of dynamic power consumption for the Spartan-3 and 58% for the Artix-7.

Table II. Power consumption for different configurations Cherkaoui et al.

Active Stages	Static	63	128	191	255	319	384	447	511
Spartan-3E (mW)	49,4 (69%)	56,0	58,0	59,8	61,0	62,7	64,5	67,3	71,2
Artix-7 (mW)	33,7 (59%)	36,7	38,6	41,6	45,0	47,6	50,2	53,2	57,0

© IEICE 2018
DOI: 10.1587/elex.14.20171157
Received November 21, 2017
Accepted December 4, 2017
Publicized December 21, 2017
Copyedited January 25, 2018

4.3 Applications: selectable security level and minimum power consumption

Once we have analized the *min-entropy* and power consumption depending on the number of ROs/stages of the implementation, we can use this information for different applications, using the operation modes applications proposed in Section 3.3. To that end, we have obtained the average min-entropy for the different tests.

4.3.1 Wold et al.

After combining the information from the experimental results in the previous section, a relation between the power consumption and the security level -*min-entropy*- can be obtained. This relation is shown in Fig. 6 for the Wold proposal, for the two selected FPGAs. The three possible selectable security levels of the proposed first operation mode -low, medium, high-entropy- are shown in the graph with limits set in 0.8–0.88 (low), 0.88–0.92 (medium) and 0.92 (high) *min-entropy*.

For this first operation mode **-selectable security-** we can observe that for the Artix-7 implementation we have a configuration distribution of 3-3-2 corresponding respectively to the levels low-medium-high entropy. On the other hand, the Spartan3 implementation presents a configuration distribution of 2-2-4. As there are several points in the three considered regions for each implementation, the final working point can be chosen in each region depending on the needs of the application, giving priority to security or power consumption.

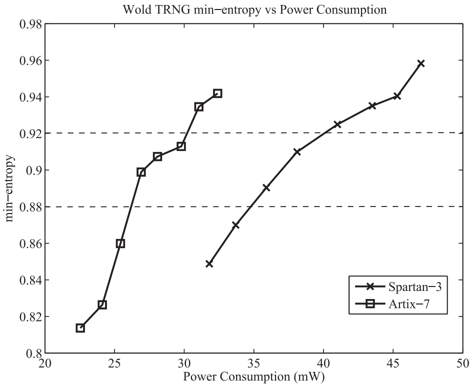


Fig. 6. *min-entropy* vs power consumption (Wold et al. proposal).

On the other hand, we have also tested the second operation mode **-minimum power consumption-**. For our experimental results we have considered an example where the operation mode is intended for an application that needs a high level of security, for instance an algorithm where the TRNG is used to generate only keys. So we have selected a value of 0.92 for *min-entropy*. In order to check if a *min-entropy* of 0.92 is guaranteed, we have injected faults (modifying the design) in some XOR-tree registers in order to switch off the entropy contribution of some active ROs. Once the faults were injected, we checked that more ROs were switched on in order to maintain the *min-entropy* level of 0.92.

© IEICE 2018
DOI: 10.1587/elex.14.20171157
Received November 21, 2017
Accepted December 4, 2017
Publicized December 21, 2017
Copyedited January 25, 2018

4.3.2 Cherkaoui et al.

We have carried out the same measurements for the Cherkaoui et al. proposal. The *min-entropy* average vs power consumption for both FPGAs is shown in Fig. 7. For the first operation mode (3 entropy levels), we can observe that for the Artix-7 implementation we have configuration distribution of 2-1-5 corresponding respectively to the levels low-medium-high entropy. On the other hand, the Spartan-3 implementation presents a configuration distribution of 1-2-5. As before, we have checked the two operation modes obtaining the expected results for both of them.

All in all, we can conclude that Wold et al. proposal offers a better trade-off between power consumption and min-entropy for both FPGAs. However, Cherkaoui et al. TRNG will be selected for applications that requires a high level of min-entropy (key generation).

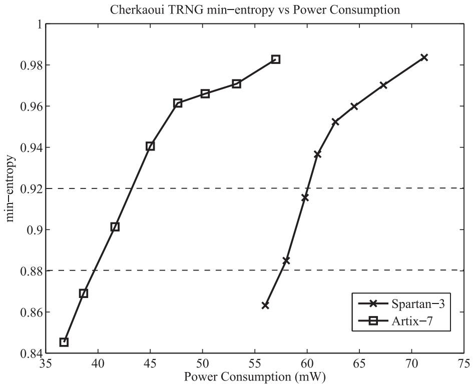


Fig. 7. *min-entropy* vs power consumption (Cherkaoui et al. proposal).

5 Conclusions

In this work we have addressed some of the challenges that the IoT poses, specially security and power consumption. In this paper, two well-known True RNGs have been studied, obtaining for them the entropy level (closely related to the number of active ROs/stages) and the power consumption. The entropy level has been obtained by considering three well-known tests -the Frequency Test, the Collision Test and the Partial Collection Test-. With the obtained results we have established the relation between the *min-entropy* value and the power consumption.

In addition, two novel applications have been presented in order to provide more flexibility than the state-of-the-art implementations with a fixed number of stages, where power consumption is not optimized, and the only option in case the entropy level is below a threshold is to just stop or reset the system. With our proposal, the number of stages can be increased or decreased dynamically -using the technique of clock gating- to increase or decrease the entropy level while adjusting the power consumption.

Acknowledgements

This work was partially supported by the Spanish Ministry of Economy and Competitiveness under contract ESP2015-68245-C4-1-P.

© IEICE 2018
DOI: 10.1587/elex.14.20171157
Received November 21, 2017
Accepted December 4, 2017
Publicized December 21, 2017
Copyedited January 25, 2018