A Unified Multibit PUF and TRNG based on Ring Oscillators for Secure IoT Devices

Iluminada Baturone, Roberto Román, and Ángel Corbacho

***Abstract*— Physically Unclonable Functions (PUFs) and True Random Number Generators (TRNGs) are cryptographic primitives very well suited for secure IoT devices. This paper proposes a circuit, named multibit-RO-PUF-TRNG, which offers the advantages of unifying PUF and TRNG in the same design. It is based on counting the oscillations of pairs of ring oscillators (ROs), one of them acting as reference. Once the counter of the reference oscillator reaches a fixed value, the count value of the other RO is employed to provide the TRNG and the multibit PUF response. A mathematical model is presented that supports not only the circuit foundations but also a novel and simple calibration procedure that allows optimizing the selection of the design parameters. Experimental results are illustrated with large datasets from two families of FPGAs with different process nodes (90 nm and 28 nm). These results confirm that the proposed calibration provides TRNG and PUF responses with high quality. The raw TRNG bits do not need post-processing and the PUF bits (even 6 bits per RO) show very small aliasing. In the application context of obfuscating and reconstructing secrets generated by the TRNG, the multibit PUF response, together with the proposal of using error-correcting codes and RO selection adapted to each bit, provide savings of at least 79.38% of the ROs compared to using a unibit PUF without RO selection. The proposal has been implemented as an APB peripheral of a VexRiscv RV32I core to illustrate its use in a secure FPGA-based IoT device.**

***Index Terms*— Hardware Security, Physical Unclonable Functions, Ring Oscillators, True Random Number Generators.**

I. Introduction

Security of IoT devices is crucial for the IoT ecosystem. If attackers succeed in compromising IoT devices severe damages can be provoked. Physically Unclonable Functions (PUFs) are lightweight cryptographic primitives very well suited to authenticate IoT devices that operate under power and area constraints [1]. An electronic PUF inside the device hardware generates an output bit sequence as response to an input (so-called challenge). The response of a PUF to a given challenge should show uniqueness (varies from device to device), reliability (remains stable over device lifetime), and unpredictability (due to exploiting the random variations introduced during the semiconductor fabrication process) [2]. Hence, a PUF response can be used as device identifier (ID).

Several lightweight authentication protocols have been reported that employ PUF IDs [3]-[7]. PUFs allow preserving the secrecy of IDs by generating them whenever needed. This is cheaper than storing them in secure non-volatile memories. Moreover, PUFs provide device counterfeiting and tampering because fake and manipulated PUFs are not able to generate the genuine IDs. These features of PUFs have been exploited to ensure the trustworthiness of the hardware of IoT devices and also to guarantee the security of their software [7].

A True Random Number Generator (TRNG) is another cryptographic primitive required by cryptographic protocols to generate secret keys, nonces, initialization vectors, etc. TRNGs exploit the random variations introduced by physical noise sources to generate an unpredictable output bit sequence that changes every time it is generated (note that the PUF response should not change at every measurement).

Among the electronic PUFs studied in greater depth, those that predominate in IoT devices are delay-based PUFs and memory-based PUFs. The first ones, such as arbiters and PUFs based on ring oscillators (RO PUFs), are more popular in IoT devices with FPGAs [5]. The second ones, such as PUFs based on static random access memories (SRAM PUFs), are more suitable for IoT devices with microcontrollers whose embedded memories are not initialized [6]-[7]. Among the digital TRNGs, those widely employed are based on metastable circuits and oscillators. The latter provide higher entropy with simpler designs, although at a slower speed (which can be enough for many IoT applications). Like RO PUFs, TRNGs based on ROs are more popular in devices with FPGAs [8]-[9].

Approaches that unify PUF and TRNG in the same design offer the advantages of being simpler than separate designs. Unified solutions based on memory cells (comprising cross-coupled inverters like SRAMs or other gates) have been already reported in [10]-[13]. In [10]-[12], the memory cells whose start-up values are stable are used for PUF and those unstable are used for TRNG responses. In [13], the random behavior of

Manuscript received 21 May 2022; revised 25 July 2022; accepted 10 November 2022. Date of current version 21 November 2022. This research was conducted thanks to Grant PDC2021-121589-I00 funded by MCIN/AEI/10.13039/501100011033 and the "European Union NextGenerationEU/PRTR", and Grant PID2020-119397RB-I00 funded by MCIN/AEI/ 10.13039/501100011033. The work of Roberto Román was supported by VI Plan Propio de Investigación y Transferencia through the University of Seville. *(Corresponding author: Iluminada Baturone).*

Iluminada Baturone, Roberto Román, and Ángel Corbacho are with the Instituto de Microelectrónica de Sevilla, (IMSE-CNM), Universidad de Sevilla, CSIC, Sevilla, 41092, SPAIN (e-mail: {lumi, roman}@imse-cnm.csic.es, and angel.corbachomendez@gmail.com).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org

the SRAM bitline discharge rate is used as common principle, harnessing noise for TRNG responses and chip-specific local variations for multibit PUF responses. Solutions based on ROs have also been reported in [14]-[16]. The solution in [14] is not a truly unified design but rather a merge between two different RO structures, one for PUF and the other for TRNG. The solutions in [15] and [16] are unified designs based on RO pairs whose oscillations are counted simultaneously by two counters, until one of them overflows. A weakness of the work in [15] is that the authors explain empirically instead of mathematically the reasons why several bits are suitable for generating true random sequences and others for generating PUF sequences and do not explain how to select the value that fixes overflow. The PUF proposed in [16] is not multibit. The multibit PUF and TRNG proposed in [17]-[18] is based on comparing the number of oscillations of pairs of TERO (Transient Effect Ring Oscillator) cells, which are metastable structures composed of two cross-coupled branches that enter a transient oscillating state before reaching a stable state. The problem of TERO cells is that their design in FPGAs is very challenging since the two branches should be identical and symmetrical. In addition, the time during which oscillations are counted has a great impact on the PUF performance and a methodology to determine it is not obvious.

Moreover, a problem of many RO-based TRNGs and PUFs is that active attacks that inject electromagnetic signals can synchronize temporarily the ROs, which is known as locking phenomenon. The consequence is a statistically manipulated output in TRNGs and a denial-of-service (DoS) in PUFs [19]-[20]. One major issue for PUFs is that efficient methods based on side-channel passive attacks have been reported to clone RO-PUFs and TERO-PUFs, breaking their security [20].

This paper presents a multibit PUF and TRNG circuit based on ring oscillators (multibit-RO-PUF-TRNG) that offers the following advantages over existing works:

- A unified structure that employs a very simple circuitry to act as a multibit PUF and TRNG, without requiring a challenging design.
- A multibit PUF with high uniqueness and reliability, which can be increased with an effective RO selection procedure.
- A mathematical model that stablishes the foundations of how to select the parameters to optimize the throughput and randomness of the PUF and TRNG responses as well as the uniqueness and reliability of the multibit PUF response.
- A simple calibration procedure that is carried out prior to the IoT device deployment and that uses simple Health Tests that evaluate continuously the behavior of the TRNG response.
- A multibit PUF and TRNG that is not sensitive in first approximation to locking phenomenon and, moreover, that is resistant to reported side-channel passive attacks, particularly to the easiest attacks based on electromagnetism emanation.

The application of the proposal to generate, obfuscate, and reconstruct secret cryptographic keys for cryptographic protocols is detailed. The performance of the proposal is illustrated with two large datasets of RO frequencies from 90- and 28-nm FPGAs provided in [21] and [22], respectively. These two datasets were available online, provided by their authors. Hence, our results can be reproduced.

The paper is organized as follows. Section II describes briefly the interest of using PUFs and TRNGs to manage secret keys and summarizes the related work. Section III presents the multibit-RO-PUF-TRNG circuit, its mathematical model, and its design and calibration procedure, in particular to manage secret keys. Section IV validates the proposal with results obtained from two large experimental datasets of FPGAs. The advantages of using the proposal for secure IoT devices instead of other circuits from the literature are discussed in Section V. Finally, conclusions are given in Section VI.

II. Preliminaries and Related Work

A. PUFs and TRNGs for Managing Secret Keys

Both PUFs and TRNGs are required for the secure management of secret keys. In the one side, the needed secret keys should be generated inside the hardware of the IoT device by the TRNG. It is not convenient to provide secret keys from outside the device. Of course, the TRNG can be also employed to generate nonces and initialization vectors needed by other cryptographic operations. In the other side, PUF responses are employed to obfuscate and later reconstruct the secret keys. Let us denote R_l^k the PUF response to a given challenge of a PUF instance l at the k-th measurement. A well-known way to obfuscate a secret with a PUF response is the Code-Offset Construction [23]. In an enrollment phase, a secret S obtained from a TRNG is encoded into a sequence P with an error-correcting code ($P = encode(S)$). The sequence P is XORed with a binary PUF response R_l^0 to generate helper data, PD, which can be public ($PD = R_l^0 \oplus P$). The PUF response R_l^0, should not have any correlation or bias so that no information about the secret S could be leaked from the helper data. During verification, a new PUF response is taken, R_l^k, which is XORed with the helper data to obtain a noisy version of P, $P' = R_l^k \oplus PD = R_l^k \oplus R_l^0 \oplus P$. Since some bit flipping can appear in the PUF response, $R_l^k \oplus R_l^0$ is not zero but small and the error-correcting code is employed to recover S from P', $S = decode(P')$ [2], [24].

Assuming that the probability distribution that models the occurrence of exactly e bit errors in the n bits provided by a PUF to obfuscate a secret bit is a binomial distribution, and knowing that the bit error probability in the PUF response is p_e, the failure probability, P_{fail}, in reconstructing a bit of the secret key when using an error-correcting code with n-bit codewords and capacity to correct up to E errors is given by:

$$P_{fail}(e > E) = 1 - \sum_{b=0}^{E} \binom{n}{b} p_e^b \cdot (1 - p_e)^{n-b} \quad (1)$$

Fixing P_{fail} to 10^{-6} (a typical value found in many works [24]), E to $n/2 - 1$ (using a simple repetition error-correcting code, suitable for resource-constrained IoT devices), and knowing p_e (from the PUF reliability), the value of n can be obtained. If the secret S has s bits, the length N of the PUF response to employ should meet that $\left\lfloor \frac{N}{n} \right\rfloor \geq s$, where $\lfloor x \rfloor$ rounds

x to the nearest integer smaller than or equal to x.

B. RO PUFs

ROs usually consist of an odd number of inverters connected in cascade and forming a closed-loop, which makes them oscillate freely when enabled. The first PUF based on ROs (RO-PUF) was proposed in [25]. The RO-PUF is composed of pairs of identically laid-out ROs whose oscillations are counted by counters. Only one bit is generated from the comparison of the counter results of a pair, because the bit codifies which RO in the pair is faster. Later, several works focused on improving the uniqueness, reliability, and efficiency of RO-PUFs [26]-[34].

In order to improve uniqueness, the work in [26] proposes to avoid systematic variations of the fabrication process by placing the ROs as close as possible to each other and picking physically adjacent pairs of ROs to obtain output bits. However, this solution is not enough because the internal routing and components of the ROs and the rest of the circuitry in the FPGA cause that many pairs generate the same output bits in all the devices, as shown in [27] and [28]. The solution proposed in [28] is to subtract the device-population-wide mean frequency of each RO in the pair from the sampled frequency before generating a PUF bit. In [16], a PUF bit response is obtained by comparing the binary counter result of a RO pair with a reference binary word. The work in [29] clusters the ROs into groups and normalizes the frequency of each RO by the frequency of a reference RO or by the mean frequency of all the ROs in the same group.

In order to improve reliability, the work in [25] proposes to select the pairs whose ROs have maximum differences in frequencies. In an initialization or registration step, the bit vector indicating these selections is saved in a binary mask to generate later a reliable output with the same pairs. The authors also state that other masking schemes based on RO's frequencies are also possible, as studied in [30] and [26]. The problem of these selection processes is the reduction of PUF uniqueness and entropy, as depicted in [27] and [31], because the frequencies of ROs show systematic bias, as commented above. It is better a selection process based on PUF output bits (already improved for uniqueness). The classification method presented in [10] is based on evaluating the PUF output bits several times in a registration phase, saving a binary mask to identify the challenges to generate the stable output bits. Recently, a PUF architecture that combines a process mismatch amplifier in an oscillator collapse topology has been presented to improve reliability and provide a fast response [32]. This proposal is adequate for full-custom designed circuits but not for FPGAs.

In order to improve efficiency, several works focused on generating more than only a single bit with a pair of ROs [33]. The work in [33] uses the basic RO-PUF construction proposed in [25], but instead of taking into account only the sign bit of the comparison (i.e. which RO has greater frequency), they define four intervals with equal probability and obtain 2-bit responses from each RO pair. As commented above, the problem is that RO frequencies show systematic bias. On the other hand, since a multibit response reduces PUF reliability, the proposal in [33] is to activate the neighboring ROs that are not being compared to introduce a background noise that affects the frequency of the RO pairs under comparison. However, this increases the power consumption. The multibit PUF proposed in [34] and used in [15] employs pairs of ROs without no specific constraints on their placement. The oscillations of the two ROs are counted simultaneously with two counters. As soon as one of the counters overflows, the resulting value in the other counter is selected. Since this value is binary, a statistical analysis of all the bits is carried out to select for the multibit PUF response the bits showing high stability in multiple measurements and high entropy in multiple devices. In [35], this solution is compared with two others that use a stable reference oscillator to fix the oscillation time of the pair of ROs, concluding that the three presented solutions offer similar performance. The work in [29] proposes a multibit RO-PUF in which the normalized frequencies of the ROs are sorted and then grouped in such a way that the amount of ROs assigned to each group are almost the same. The works in [29] and [34]-[35] do not present any RO selection procedure to increase the reliability of the multibit PUF response.

A drawback of many RO PUFs is that no mathematical model is provided to support the design but the validation is done experimentally only [15], [22], [28], [29], [33]. Recently, a major problem reported for RO PUFs, including TERO PUFs, is their vulnerability to electromagnetic analysis, which makes them clonable [20].

C. RO TRNGs

RO TRNGs in FPGAs mostly use the jitter as physical noise source. The jitter is the frequency/phase instability of the oscillating signals generated by the ROs. In order to validate the jitter as noise source, several works focused on modelling and observing it. In [36], the jitter is modelled as the delay variations accumulated during one half oscillation period in the gates of the RO, which are caused by deterministic and non-deterministic sources. They consider the local non-deterministic source as the Gaussian source that should be exploited by a RO TRNG. They validate experimentally its jitter model by using a counter that counts the rising edges of the RO output signal during a fixed measurement time controlled by an external quartz reference clock, thus measuring the accumulated jitter.

The RO-TRNG proposed in [15] uses two ROs whose oscillations are counted simultaneously by two counters during a fixed measurement time controlled by the overflow of the counter of the fastest RO. The LSBs of the counter of the slowest RO are used to generate random sequences after being post-processed by Von Neumann or XOR correction. The authors do not propose any calibration technique to select the size of the counters and do not explain mathematically how many LSBs are adequate to generate true random sequences.

The authors in [37] present a mathematical model that describes how the bits of the binary-coded count value of RO oscillations during a fixed measurement time are related to the deterministic and non-deterministic jitter. Since the LSBs are related to the Gaussian non-deterministic jitter, they should be

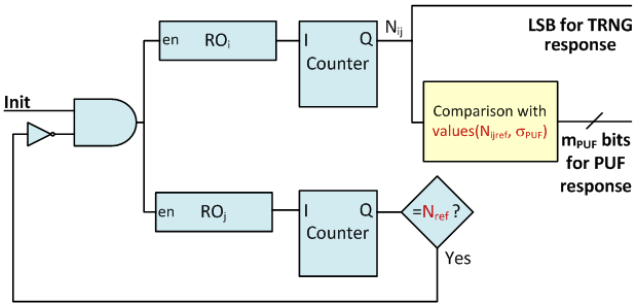


Fig. 1. Block diagram of the multibit-RO-PUF-TRNG core proposed.

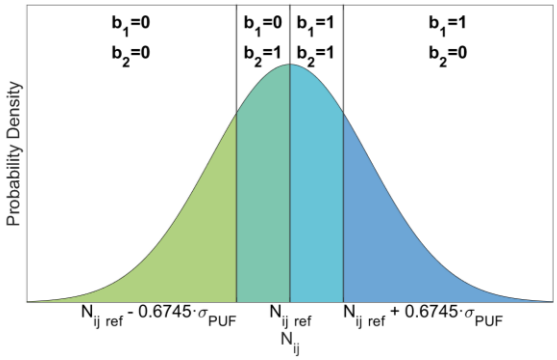


Fig. 2. Gaussian distribution of the counter values in RO pairs due to manufacturing noise, with 4 areas of the same probability depicted.

the output of the TRNG. The authors propose an auto-calibrated TRNG based on measuring the standard deviations of the count values to fix the measurement time and obtain the highest TRNG speed. Although an external quartz reference clock, as proposed in [36], can fix the measurement time, the authors in [37] conclude that using two theoretically identical ROs as oscillators is better for calibration.

As depicted in [36], deterministic jitter accumulates faster than Gaussian jitter. Hence, there are TRNGs faster than those presented in [36]-[37], but electromagnetic attacks have been reported on some of them [19]. Recently, a high-throughput TRNG has been proposed that is based on two symmetrically designed slow ROs (which employ current starved inverters biased in the weak inversion region) and an ultra-high-speed counter (which employs dynamic ratioed logic instead of static CMOS logic) [38]. This proposal is adequate for full-custom designed circuits but not for FPGAs.

III. The Proposal of Multibit-RO-PUF-TRNG

A. The Core of the Circuit

The core of the proposed multibit-RO-PUF-TRNG circuit is composed of two oscillators, RO_i and RO_j, and two counters, as shown in Fig. 1. When the core is activated (the signal $Init$ is '1'), both oscillators oscillate simultaneously and stop when the counter of RO_j reaches a value N_{ref} that is conveniently set in a calibration step. From the counter of RO_i, the count value N_{ij} is obtained, which is assigned to the pair ij. The least significant bit (LSB) in N_{ij} is taken for the TRNG response since it measures the random physical internal noise.

Since the random noise in the manufacturing process is assumed to follow a Gaussian distribution (as supported by the model explained in the following subsection), let us call $N_{ij_{ref}}$ the average of the count value N_{ij} in a group of L PUF instances and σ_{PUF} the standard deviation, which are conveniently set in the calibration step. From the comparisons of the count value N_{ij} with the values that divide the probability distribution function of the manufacturing noise into $2^{m_{PUF}}$ areas with the same probability (which take into account $N_{ij_{ref}}$ and σ_{PUF}), m_{PUF} bits are taken for the PUF response. For example, if $2^2 = 4$ areas with the same probability are considered, as illustrated in Fig. 2, the following comparisons illustrate how to obtain 2 bits (b_1 and b_2) for the PUF response

$$b_1 = \begin{cases} 1 \text{ if } N_{ij} > N_{ij_{ref}}, b_2 = \begin{cases} 0 \text{ if } N_{ij} > N_{ij_{ref}} + 0.6745 \cdot \sigma_{PUF} \\ 1 \text{ if } N_{ij} \leq N_{ij_{ref}} + 0.6745 \cdot \sigma_{PUF} \end{cases} \\ 0 \text{ if } N_{ij} \leq N_{ij_{ref}}, b_2 = \begin{cases} 1 \text{ if } N_{ij} > N_{ij_{ref}} - 0.6745 \cdot \sigma_{PUF} \\ 0 \text{ if } N_{ij} \leq N_{ij_{ref}} - 0.6745 \cdot \sigma_{PUF} \end{cases} \end{cases}$$

The expression above and Fig. 2 illustrate the use of a 2-bit Gray code. A Gray code is recommended instead of other codes to improve the reliability of the bits in the PUF response.

Using N pairs of oscillators, a TRNG sequence of N bits and a PUF sequence of $N \cdot m_{PUF}$ bits are generated. A simple way to obtain N pairs of oscillators generating independent bits is to use $N + 1$ ROs. It is better to activate the pairs of oscillators in series to reduce the power consumption and the resources employed (the pairs can be multiplexed to use only two counters).

B. Mathematical Model Supporting the Proposal

If RO_j has an oscillation frequency f_j ($f_j = 1/T_j$), the oscillation time interval of the pair ij between initialization and stop signal is

$$t_{ref} = \frac{N_{ref}}{f_j} = N_{ref} \cdot T_j \quad (2)$$

If the oscillation frequency of RO_i is f_i ($f_i = 1/T_i$), its counter reaches the following value at stop signal

$$N_{ij} = \lfloor t_{ref} \cdot f_i \rfloor = \lfloor N_{ref} \cdot T_j \cdot f_i \rfloor \quad (3)$$

The period of an oscillator can be modelled as

$$T_i = T_{oi} + \Delta T^{gD} + \Delta T^{lD} + \Delta T^{gN} + \Delta T^{lN} = T_{oi} + \Delta T_i \quad (4)$$

where T_{oi} is the oscillator nominal period . ΔT^{gD} is the jitter produced deterministically by global causes (e.g. temperature, power supply voltage, and placement and routing of the oscillator). ΔT^{lD} is the jitter produced deterministically by local causes (e.g. local temperature and power supply voltage variations, and local variations of the manufacturing process). ΔT^{gN} is the jitter produced by global non-deterministic

variations (e.g. power supply noise and electromagnetic interference signals), and ΔT^{lN} is caused by local non-deterministic variations due to semiconductor noise.

Hence, the frequency of an oscillator can be modelled as

$$f_i \cong \frac{1}{T_{0\mathrm{i}}} - \frac{\Delta T_{\mathrm{i}}}{T_{0\mathrm{i}}^2} = f_{0\mathrm{i}} \cdot \left(1 - \frac{\Delta T_i}{T_{0i}}\right) = f_{0\mathrm{i}} \cdot (1 - \delta_{\mathrm{i}}) \quad (5)$$

Substituting T_j by (4) and f_i by (5) in (3), it follows that

$$N_{ij} = \left\lfloor N_{ref} \cdot T_{oj} \cdot (1 + \frac{\Delta T_j}{T_{oj}}) \cdot f_{0\mathrm{i}} \cdot (1 - \frac{\Delta T_i}{T_{0i}}) \right\rfloor \cong$$

$$\left\lfloor N_{ref} \cdot \frac{f_{0\mathrm{i}}}{f_{oj}} \cdot (1 + \frac{\Delta T_j}{T_{oj}} - \frac{\Delta T_i}{T_{0\mathrm{i}}}) \right\rfloor = \left\lfloor N_{ref} \cdot \frac{f_{0\mathrm{i}}}{f_{oj}} \cdot (1 + \delta_{\mathrm{j}} - \delta_{\mathrm{i}}) \right\rfloor \quad (6)$$

If RO_j and RO_i have the same configuration ($f_{oj} = f_{oi} = f_o$), N_{ij} will be

$$N_{ij} = \left\lfloor N_{ref} \cdot (1 + \frac{\Delta T_j^{gD} - \Delta T_i^{gD}}{T_o} + \frac{\Delta T_j^{lD} - \Delta T_i^{lD}}{T_o} + \frac{\Delta T_j^{gN} - \Delta T_i^{gN}}{T_o} + \frac{\Delta T_j^{lN} - \Delta T_i^{lN}}{T_o}) \right\rfloor = \left\lfloor N_{ref} \cdot (1 + \delta_{ij}^{gD} + \delta_{ij}^{lD} + \delta_{ij}^{gN} + \delta_{ij}^{lN}) \right\rfloor \quad (7)$$

Non-deterministic variations affecting globally to the ROs, like those provoked by the injection of electromagnetic signals, are avoided with the difference $\Delta T_j^{gN} - \Delta T_i^{gN}$, thus reducing locking phenomenon.

If the pair of ROs is implemented in a PUF instance l and a measurement k is taken, the value of N_{ij}, now renamed as N_{ijkl}, can be expressed as follows

$$N_{ijkl} = \left\lfloor N_{ref} \cdot (1 + \delta_{ij}^{gD} + \delta_{ijl}^{lD} + \delta_{ijk}^{N}) \right\rfloor \quad (8)$$

The relative and differential variations due to global and deterministic causes, δ_{ij}^{gD}, are considered independent of the PUF instance and the measurement, those due to local and deterministic causes, δ_{ijl}^{lD}, are considered independent of the measurement, and those due to non-deterministic causes, δ_{ijk}^{N}, are considered independent of the PUF instance. Considering that $\sum_{k=1}^{K} \delta_{ijk}^{N}$ is zero (the noise has no bias), the average of the count values after several measurements is

$$\overline{N_{\imath jl}} = \frac{1}{K}\sum_{k=1}^{K} N_{ijkl} = \left\lfloor N_{ref} \cdot (1 + \delta_{ij}^{gD} + \delta_{ijl}^{lD}) \right\rfloor \quad (9)$$

If the average $\sum_{l=1}^{L} \delta_{ijl}^{lD}/L$ is named $\overline{\delta_{ij}^{lD}}$ (local and deterministic differential variations may have bias), the average of the count values in the L instances is

$$N_{ij_{ref}} = \overline{N_{\imath j}} = \frac{1}{L}\sum_{l=1}^{L} \overline{N_{\imath jl}} = \left\lfloor N_{ref} \cdot (1 + \delta_{ij}^{gD} + \overline{\delta_{ij}^{lD}}) \right\rfloor \quad (10)$$

Assuming that $\delta_{ijl}^{lD} = \overline{\delta_{ij}^{lD}} + \frac{\Delta T_{ijl}^{lD}}{T_0}$ and substituting $N_{ij_{ref}}$ in (9)

$$\overline{N_{\imath jl}} = \left\lfloor N_{ij_{ref}} + N_{ref} \cdot \frac{\Delta T_{ijl}^{lD}}{T_0} \right\rfloor \quad (11)$$

The term $N_{ij_{ref}}$ includes the influence of all the variations produced deterministically by global causes as well as the possible constant bias of the local deterministic variations. The term $N_{ref} \cdot \frac{\Delta T_{ijl}^{lD}}{T_0}$ includes the influence of the local deterministic variations without biasing, which is the term that should be measured by the PUF. Equation (11) confirms (as shown in Fig. 2) that the count values $\overline{N_{\imath jl}}$ deviate from the average $N_{ij_{ref}}$ with a standard deviation that we call σ_{PUF}. In addition, it explains why our proposed multibit PUF provides equi-probable bits by dividing the probability distribution function of the manufacturing noise into areas with the same probability.

Substituting (10) in (8), we have

$$N_{ijkl} = \left\lfloor N_{ij_{ref}} + N_{ref} \cdot \frac{\Delta T_{ijl}^{lD}}{T_0} + N_{ref} \cdot \delta_{ijk}^{N} \right\rfloor = \left\lfloor N_{ij_{ref}} + N_{ref} \cdot \frac{\Delta T_{ijl}^{lD}}{T_0} + N_{ref} \cdot \frac{\Delta T_{ijk}^{N}}{T_0} \right\rfloor \quad (12)$$

The term $N_{ref} \cdot \frac{\Delta T_{ijk}^{N}}{T_0}$, which depends on the measurement k, includes the influence of the local Gaussian non-deterministic variations, which are the variations that should be exploited by a TRNG.

Considering in a first-order estimation that all the inverting delay stages of a RO introduce a nominal propagation delay of τ, as in the simplified models in [36] and [39], the nominal period of a RO with M delay stages can be approximated as

$$T_o = 2 \cdot M \cdot \tau \quad (13)$$

Assuming that the local deterministic variations without biasing in the M stages are uncorrelated (because they are supposed to be independent), the variations ΔT_{ijl}^{lD} are normally distributed with mean zero and standard deviation $\sqrt{M} \cdot \sigma^{lD}$, where σ^{lD} is the standard deviation considering only one stage. Hence, using (13), the term $N_{ref} \cdot \frac{\Delta T_{ijl}^{lD}}{T_0}$ to be measured for the PUF response is proportional to $\frac{N_{ref}}{\sqrt{M}} \cdot \frac{\sigma^{lD}}{2 \cdot \tau}$. Assuming that the inverting stages in the ROs have a given nominal propagation delay τ and are implemented in a manufacturing process with a given σ^{lD} (due to, mainly, random dopant fluctuations, line-edge roughness, and oxide thickness variations [39]), ROs with a small number of stages, M, are desirable for the PUF response.

Similarly, the variations ΔT_{ijk}^{N} are normally distributed with mean zero and standard deviation $\sqrt{M} \cdot \sigma^{N}$, where σ^{N} is the standard deviation considering only one stage and due to, mainly, white semiconductor noise. Hence, using (13), the term $N_{ref} \cdot \frac{\Delta T_{ijk}^{N}}{T_0}$ to be measured for the TRNG response is

proportional to $\frac{N_{ref}}{\sqrt{M}} \cdot \frac{\sigma^N}{2 \cdot \tau}$. The work in [40] provides a compact expression for uncertainty in the propagation delay of an inverting stage, which can be considered generic enough for CMOS designs. This expression shows that $(\sigma^N)^2$ depends on the manufacturing technology and is directly proportional to the temperature and nominal propagation delay and inversely proportional to the power supply voltage V_{DD} and the average pullup and pulldown current I at the inverting stage. Assuming that the RO inverting stages are designed with a given nominal propagation delay τ and work with a given V_{DD} and I in a manufacturing process with given noise coefficients, it is also desirable to have a small number of stages in the ROs for the TRNG response.

Since local non-deterministic variations are usually one order of magnitude inferior to local deterministic ones [15], [21], [22], N_{ref} should be large enough to make the term $N_{ref} \cdot \frac{\Delta T_{ijk}^N}{T_0}$ influential. In any case, N_{ref} should be large enough to reduce the discretization noise introduced by the rounding operator $\lfloor x \rfloor$. According to the model above, if there are no variations, all the count values N_{ijkl} and the references $N_{ij_{ref}}$ coincide with N_{ref}.

C. Design and Calibration Procedure

An issue to consider in the design of the multibit-RO-PUF-TRNG proposed are the recommendations in [41], which state that the raw data (before any possible conditioning) provided by a TRNG should be tested to detect quickly and with a high probability any failure in the noise source. Two of these tests, named Health Tests, are recommended: the Repetition Count test and the Adaptive Proportion test. The first one quickly detects catastrophic failures that cause the noise source to provide the same single output value for a long time. The second one detects when some value begins to occur much more frequently than expected. Following these recommendations, it is assumed that these two approved Health Tests evaluate continuously the TRNG response.

A first parameter to select in the design is N_{ref}. In the one side, the value of N_{ref} should be large enough to measure true local non-deterministic variations and to reduce the discretization noise. In the other side, the throughput of the circuit decreases as N_{ref} increases (since t_{ref} is proportional to N_{ref}, as shown in (2)). In order to optimize the selection of N_{ref}, we propose a calibration procedure that starts with a large value of N_{ref}, named as N_{top}. The two Health Tests are applied to the B bits of the count value N_{ijkl} of a pair ij. Each test is applied several times, each one taking into account 1,024 successive samples since the entropy source is binary. The percentage of tests passed by each bit is evaluated from the least (LSB) to the most significant bit (MSB) in the N_{ijkl}, until the first bit that fails the tests is found. Let us assume that bit is the q^{th} LSB. If the percentage of passed tests is not complete but relatively high (e.g. equal or greater than 95%), the value of N_{ref} is selected as $\left\lfloor {N_{top}}/{2^{q-2}} \right\rfloor$. If the percentage of passed tests is not complete and relatively low (e.g. smaller than 95%), the value of N_{ref} is selected as $\left\lfloor {N_{top}}/{2^{q-3}} \right\rfloor$. This calibration selects a resolution for N_{ref} that ensures the LSB passes the statistical tests (and probably the second LSB too). The idea is that it is much faster to obtain several random bits in successive measurements than several bits in one measurement. This calibration procedure is simpler than that proposed in [37]. Of course, the values of N_{ref} can be adjusted finer around these coarser values but even with these values, the results are quite interesting, as shown in the Section IV.

According to (12), the influence of the local non-deterministic variations together with the rounding operator can cause some bits of the PUF response flip from one measurement to another. This happens to pairs whose count values N_{ij} are close to the values that divide the probability distribution function of the manufacturing noise. Following the classification method presented in [10], our proposal is that the manufacturer of the IoT devices should evaluate the multiple bits provided by each RO pair at several measurements to select which pairs are adequate to generate them because they never showed bit flipping. Then, the proportion of pairs selected to generate each bit is calculated. For example, a proportion p_i of pairs is selected for the bit, b_i. Then, using the selected pairs for each bit, the error probabilities associated to the generation of each bit (p_e in (1)) are calculated. Finally, using (1), the lengths of the repetition codes for each bit are computed (repetition codes are used for simplicity). For example, the length of the repetition code is n_1 for the first bit, b_1, provided by the pairs of ROs for the PUF response; the length is n_2 for the second bit, b_2, and so on. Note that we treat each bit independently.

Let us assume that the multibit-RO-PUF-TRNG is an Intellectual Property (IP) module whose number $(N + 1)$ of ROs can be selected by the manufacturer. For the adequate selection of N, the manufacturer has to consider the number of bits, s, of the secrets to generate and obfuscate in the application context (as explained in the end of Subsection II.A for the unibit case without classification). Since the manufacturer knows the proportions p_i and the lengths n_i of the repetition codes for each bit, due to the evaluation carried out as described above, the number N of RO pairs should meet that:

$$\left\lfloor \frac{N \cdot p_1}{n_1} \right\rfloor + \cdots + \left\lfloor \frac{N \cdot p_{mPUF}}{n_{mPUF}} \right\rfloor \geq s \tag{14}$$

Hence, in order to minimize the use of FPGA resources and the power consumption, the best value for N is the minimum that verifies Equation (14).

Once the values of N and N_{ref} are selected, the manufacturer generates the bit stream that contains the multibit-RO-PUF-TRNG (and the rest of the system to implement in the FPGA) and downloads it in the IoT devices. Then, the manufacturer calculates the values $N_{ij_{ref}}$ as the average of the count values $\overline{N_{ijl}}$ in a group of L PUF instances (according to Equation (10)) and also calculates their standard deviation σ_{PUF}. The values with which to compare the N_{ij} (as shown in Fig. 1 and 2) are obtained from $N_{ij_{ref}}$ and σ_{PUF}. This solution to improve PUF

TABLE I
PERCENTAGES OF CONTINUOUS HEALTH TESTS PASSED BY EACH BIT IN THE N_{ijkl}

Bit position	1st (MSB)	…	6th	7th	8th	9th	10th	11th	12th	13th	14th	15th	…	18th	19th	…	22nd (LSB)
Dataset90nm	0	…	0	4.08	14.29	26,53	46,94	75,51	100	100	100	100	…	100	-		-
Dataset28nm	0	…	0	0	0	0	0	5.16	7.10	29.68	98.71	100	…	100	100	…	100

uniqueness is simpler than that in [28] and [29]. Finally, the binary masks to identify which RO pairs are adequate to generate the m_{PUF} bits of each PUF are obtained with the classification procedure applied to each PUF instance. Once the calibration step finishes, the IoT devices contain the instances of the multibit-RO-PUF-TRNG implemented with the number $(N+1)$ of ROs, the parameters N_{ref}, the values depending on $N_{ij_{ref}}$ and σ_{PUF}, and the binary masks.

IV. EXPERIMENTAL RESULTS

Two open-sourced datasets were used for the evaluation of the proposed multibit-RO-PUF-TRNG, the dataset of Maiti et al. [21] and that of Hesselbarth et al. [22]. The first one contains data from 193 Xilinx SPARTAN3E S500 FPGAs of 90 nm, each one having 512 five-stage ROs placed in a 16x32 array in the middle of the FPGA. They provide 100 measurements for each RO. Herein, this dataset is referred to as *Dataset90nm*. On the other hand, the dataset of Hesselbarth et al. uses 217 Xilinx Artix-7 XC7A35T FPGAs of 28 nm. This dataset, herein referred to as *Dataset28nm*, is divided into 4 groups. As an example, we chose for our experiments the group called "left-lower" that comprises 1600 three-stage ROs that occupy one slice, using the two slice variants on the Artix-7. They provide also 100 measurements per RO. Among the data provided for 15 different evaluation times, we made the experiments with the longest one of 10.0 ms to reduce the quantization noise. In order to obtain the PUF and TRNG responses, we used whenever needed the conversion from frequencies to counter values (Equation (3)) and to time of measurements (Equation (2)).

A. Results on Calibration and TRNG Responses

The mean of the count values found in the datasets was 256866 in decimal (which in binary requires 18 bits) for the *Dataset90nm* and 2270084 (which in binary requires 22 bits) for the *Dataset28nm*. As explained in the calibration procedure, these are the values selected for N_{top}. Then, the two approved Health Tests in [41] (the Repetition Count and the Adaptive Proportion Tests) were applied to each bit of the count value N_{ijkl} of the proposed multibit-RO-PUF-TRNG, using $N_{ref} = N_{top}$ and bitstreams with 1,024 bits. The number of tests was limited by the quantity of data available per FPGA. The percentages of tests passed by each bit are shown in Table I. Looking from the LSB to the MSB, the first bit that fails the tests in the *Dataset90nm* is the 11th MSB (equivalently the 8th LSB). Since the percentage of passed tests by this bit is relatively low, the value of N_{ref} is selected as $\left\lfloor {N_{top}}/{2^{8-3}} \right\rfloor$, that is, N_{ref} = 8027 = $\left\lfloor \frac{256866}{2^5} \right\rfloor$, which reduces the resolution of the counters from 18 to 13 bits and increases the throughput of the multibit-RO-PU-TRNG in 2^5. Looking from the LSB to the MSB, the first bit that fails the tests in the *Dataset28nm* is the 14th MSB (equivalently the 9th LSB). Since the percentage of passed tests by this bit is relatively high, the value of N_{ref} is selected as $\left\lfloor {N_{top}}/{2^{9-2}} \right\rfloor$, that is, N_{ref} = 17735= $\left\lfloor \frac{2270084}{2^7} \right\rfloor$, which reduces the resolution of the counters from 22 to 15 bits and increases the throughput in 2^7.

While the Health Tests are carried out during the operation of the multibit-RO-PUF-TRNG (i.e.on-line), the tests widely used off-line to verify the randomness of a TRNG are taken from the NIST test suite [42]. In particular, the NIST Frequency (Monobit), Block Frequency (with M=20), Cumulative Sums (in forward and backward modes), Runs, Longest Run of ones in a block, and Approximate Entropy statistical tests were applied as well as the minimum entropy was calculated to the bits of the count values N_{ijkl}, using the values commented above for N_{ref}. The minimum entropy of a bit *b* in *n* sequences made of *q* times that bit is calculated as

$$\overline{\mathrm{H}}_{min}^{TRNG}[b] = -\frac{1}{n}\sum_{i=1}^{n} log_2(p_{max}[b]) \cdot 100 \qquad (15)$$

where $p_{max}[b]$ is the maximum probability of the bit *b* taking logic value '0' or '1' in the *q* times.

In order to apply the typical significance level of α=0.01 in these tests, 100 sequences generated by the proposed multibit-

TABLE II
NIST TEST RESULTS AND HMIN FOR DATASET90NM AFTER CALIBRATION

	3rd LSB		2nd LSB		LSB	
	P-val	Prop.	P-val	Prop.	P-val	Prop.
Frequency	0.00	0.50	0.80	0.99	0.29	1.00
BlockFrequency	0.00	0.00	0.68	1.00	0.08	1.00
CumSum(f)	0.00	0.35	0.08	0.99	0.07	0.99
CumSum(b)	0.00	0.32	0.01	0.99	0.24	1.00
Runs	0.00	0.05	0.53	1.00	0.24	0.96
LongestRun	0.00	0.32	0.21	0.99	0.49	1.00
ApprEntropy	0.00	0.01	0.29	0.98	0.02	0.99
$\overline{\mathrm{H}}_{min}^{TRNG}$	83.05		94.60		95.04	

TABLE III
NIST TEST RESULTS AND HMIN FOR DATASET28NM AFTER CALIBRATION

	3rd LSB		2nd LSB		LSB	
	P-val	Prop.	P-val	Prop.	P-val	Prop.
Frequency	0.04	0.95	0.30	1.00	0.23	1.00
BlockFrequency	0.00	0.79	0.76	0.98	0.05	0.99
CumSum(f)	0.00	0.96	0.76	1.00	0.23	1.00
CumSum(b)	0.00	0.95	0.03	1.00	0.55	0.99
Runs	0.00	0.56	0.38	0.97	0.53	0.99
LongestRun	0.00	0.90	0.06	1.00	0.68	0.99
ApprEntropy	0.00	0.77	0.40	1.00	0.46	1.00
$\overline{\mathrm{H}}_{min}^{TRNG}$	93.51		95.44		94.81	

RO-PUF-TRNG in each FPGA were analyzed in both datasets (n=100 in (15)). Each sequence has 500 measurements of these bits (q=500 to evaluate p_{max} in (15)), obtained from the 100 measurements of 5 RO pairs (the data available in *Dataset90nm* limit this number of bits). The results are shown in Table II and Table III for one of the FPGAs in both datasets and the three least significant bits of the count values N_{ijkl}. The statistical tests passed are shown filled in grey. As can be seen, the LSB (and even the 2nd LSB) passes the entire tests, providing a high value of minimum entropy. This confirms that the values of N_{ref} selected by the proposed calibration are adequate to ensure the TRNG performance with a high throughput.

B. Results on PUF Reponses

The metrics most usually employed to evaluate PUF performance are based on the average Hamming distances ($\overline{HD}$) evaluated on PUF responses of different PUF instances ($\overline{HD}_{inter}^{PUF}$) or different measurements of the same PUF response ($\overline{HD}_{intra}^{PUF}$). Considering that each of the L devices provides a PUF response with $N \cdot m_{PUF}$ bits, we have evaluated the performance of each of the m_{PUF} bits separately, as follows:

$$\overline{HD}_{inter}^{PUF}[b] = \frac{2}{L\cdot(L-1)} \sum_{l=1}^{L-1} \sum_{m=l+1}^{L} \frac{HD(R_l[b], R_m[b])}{N} \cdot 100 \qquad (16)$$

$$\overline{HD}_{intra}^{PUF}[b] = \frac{1}{L\cdot K} \sum_{l=1}^{L} \sum_{k=1}^{K} \frac{HD(\overline{R_l}[b], R_l^k[b])}{N} \cdot 100 \qquad (17)$$

with b ranging from b_1 to b_{mPUF}.

The first one measures PUF uniqueness/randomness and the second, its reliability. Their ideal values are, respectively, 50% and 0%. In order to measure more in depth PUF uniqueness/randomness, we also measure the bit aliasing:

$$BA^{PUF}[b] = \frac{1}{L} \sum_{l=1}^{L} R_l[b] \cdot 100 \qquad (18)$$

Its ideal value is 50%. We have measured $UB_{BA}^{PUF}[b]$ as the percentage of the N RO pairs that provide the bit b with a $BA^{PUF}[b] \in [45\%, 55\%]$. The ideal value of $UB_{BA}^{PUF}[b]$ is 100%, which means that the bit b provided by our proposed PUF takes approximately the same times a value '0' or '1' in the L devices. In addition, the NIST Frequency (Monobit), Block Frequency (with M=20), Cumulative Sums (in forward and backward modes), Runs, Longest Run of ones in a block, and Approximate Entropy statistical tests were applied as well as the minimum entropy was calculated to also evaluate the uniqueness/randomness of the PUF responses, as was done with the TRNG responses. The minimum entropy of each bit b provided by N RO pairs in L devices is calculated as

$$\overline{\mathrm{H}}_{min}^{PUF}[b] = -\frac{1}{N} \sum_{ij=1}^{N} log_2\left(p_{ij,max}[b]\right) \cdot 100 \qquad (19)$$

where $p_{ij,max}[b]$ is the maximum probability of the bit b provided by the RO pair ij taking logic value '0' or '1' in the L devices.

Using the values of N_{ref}, N_{ijref}, and σ_{PUF} as resulting from the calibration, Table IV shows the metrics above and the proportion of passing sequences for the NIST tests evaluated on 6 bits provided by our proposed PUF (m_{PUF}= 6). The results shown correspond to the *Dataset90nm*, so that L is 193 and N is 511. The randomness was evaluated in the number of devices since we want to evaluate the randomness of the manufacturing variability, so that 511 sequences with 193 bits were evaluated for b_1 to b_6. All the NIST statistical tests were passed for the typical significance level of α=0.01. Concerning reliability, it is worse as the bit distinguishes finer areas dividing the probability distribution function of the manufacturing noise, as expected. Note that b_1 distinguishes between 2 areas, b_2 distinguishes between 2 areas within each of the 2 areas already distinguished by b_1, and so on.

C. Robustness against Changes in Operating Conditions

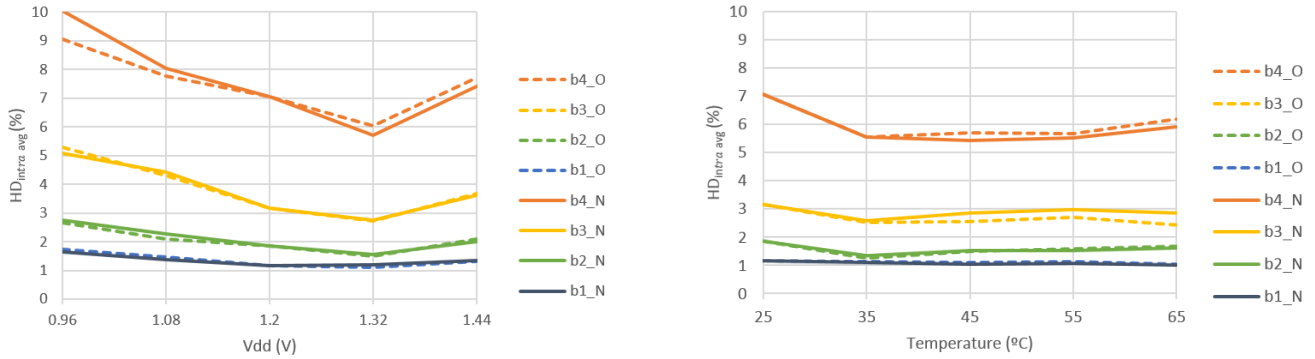
The results shown above correspond to parameters N_{ref}, $N_{ij_{ref}}$, and σ_{PUF} calibrated at nominal operating conditions (1.2V and 25ºC). The results obtained by using those values under other operating conditions were also analyzed. For this purpose, we used the *Dataset90nm*, which offers data for additional FPGAs under different temperatures and power supply voltages, and the *Dataset28nm*, which offers data for additional FPGAs under different temperatures. We obtained no significant variations in the randomness, uniqueness, and reliability results. As example, Fig. 3 illustrates the $\overline{HD}_{intra}^{PUF}$ (defined in Equation (17)) obtained for the bits 1 to 4 provided by our proposal when using N=63 pairs of ROs, K=100 measurements per pair, L=40 PUF instances, and $\overline{R}_l$ the reference response of each instance evaluated as the mean of the K measurements in the corresponding operating condition, with the *Dataset90nm*. It can be seen how the results obtained with the parameters calibrated at each operating condition (b1_O to b4_O) differ only slightly from the results obtained with the calibration at only nominal conditions (b1_N to b4_N), considering variations of the power supply (Fig. 3(a)) and of the temperature (Fig. 3(b)). Hence, for simplicity, the parameters can be calibrated using only nominal conditions.

D. Error-Correcting Codes and RO Selection

The RO selection described in Subsection III.C was applied to the 1,599 RO pairs of each FPGA provided in the *Dataset28nm*. A number of 1, 20 and 40 measurements were considered to evaluate bit flipping in 6 bits of the PUF response. Table V shows the proportions p_i of pairs selected to generate each bit. It can be seen that the proportions decrease as more bits are generated and more measurements are taken. Using the

TABLE IV
PERFORMANCE OF THE PROPOSED MULTIBIT-RO-PUF-TRNG ACTING AS 6-BIT PUF USING DATASET90NM AFTER CALIBRATION

	b_1	b_2	b_3	b_4	b_5	b_6
$\overline{HD}_{intra}^{PUF}$	0.94	1.49	2.69	5.28	10.53	20.86
$\overline{HD}_{inter}^{PUF}$	50.17	50.09	50.04	49.99	50.00	49.98
UB_{BA}^{PUF}	99.02	91.98	88.26	83.17	85.13	83.17
$\overline{\mathrm{H}}_{min}^{PUF}$	95.28	93.37	92.81	91.71	92.23	91.83
Frequency Prop.	1.00	1.00	1.00	0.99	0.99	0.98
BlockFreq. Prop.	1.00	1.00	0.99	0.99	0.99	0.98
CumSum(f) Prop.	1.00	1.00	1.00	0.99	0.99	0.98
CumSum(b) Prop.	1.00	1.00	1.00	0.99	0.99	0.98
Runs Prop.	0.99	0.99	1.00	0.99	1.00	0.99
LongestRun Prop.	0.99	1.00	0.99	0.99	0.99	0.99
ApprEntrop. Prop.	0.99	0.98	0.98	0.98	0.98	0.99



(a) (b)

Fig. 3. $\overline{HD}_{intra}^{PUF}$ using *Dataset90nm* of 4 bits/RO provided by the multibit-RO-PUF-TRNG when (a) the power supply varies and (b) the temperature varies. The results b1_N to b4_N correspond to parameters calibrated at nominal conditions and used at other operating conditions. The results b1_O to b4_O correspond to parameters calibrated and used at each operating condition.

selected pairs, $\overline{HD}_{inter}^{PUF}$ is maintained around the 50% while the $\overline{HD}_{intra}^{PUF}$ is improved significantly (no selection corresponds to only 1 measurement). Estimating the error probabilities of each bit (p_e in Equation (1)) as the $\overline{HD}_{intra}^{PUF}$, and using Equation (1), the lengths of the repetition codes, n_i, for each bit were computed (each bit is analyzed independently). If the ROs are selected as proposed, the bit sizes of the codewords needed for the obfuscation of one secret bit decrease compared to the situation of no RO selection. This reduces in turn the number $(N + 1)$ of ROs needed to obfuscate a secret key of a fixed size, as depicted in Equation (14). Using the best value of N in Equation (14), Fig. 4 compares the number of ROs needed for the obfuscation of a secret of 128, 192 and 256 bits, using our construction as unibit PUF without selection and using it as a multibit PUF with 6 bits per RO pair after a selection process with 20 and 40 measurements. The combination of the multibit PUF along with the selection of the RO pairs result in savings of at least 79.38% of ROs. Since the RO pairs are usually activated sequentially, this further improves greatly the throughput of the proposed multibit-RO-PUF-TRNG.

V. The Multibit-RO-PUF-TRNG in Secure IoT Devices

The system programmed in the FPGA of an IoT device usually contains a processor or microcontroller and peripherals. As example, we have implemented into a Xilinx Artix-7 XC7A35T FPGA a Murax SoC that includes a VexRiscv RV32I core (CPU), a JTAG debugger, 32 kB of on-chip RAM, an APB bus for peripherals [43], and one timer, one UART, and the proposed multibit-RO-PUF-TRNG as peripherals. The latter contains 168 three-stage ROs, adequate to obfuscate a secret key of 128 bits, as shown in Figure 4, organized into 167 pairs, whose outputs are multiplexed to two counters. The counter acting as the reference stops when the bit 15 takes the value '1', since N_{ref} equal to 16,384 was proven adequate to measure Gaussian noise in a technology of 28 nm. The output, N_{ij}, of the other counter goes to the CPU through the APB bus. Since the proposed multibit-RO-PUF-TRNG does not require a challenging design of the ROs, we did not fix their placement and routing. The occupation of the multibit-RO-PUF-TRNG together with the interface with the APB bus is 4,325 slice LUTs (20.79 % of the FPGA LUTs) and 122 slice registers (0.29 % of the FPGA registers). The evaluation time per RO was measured with the timer of the Murax as 41.44 µs in average, which means an oscillation frequency of 395.33 MHz in average. The power consumption estimated by Vivado 2018.1 for the peripheral is 2 mW (each RO pair is activated sequentially). Since each RO is able to provide 6 bits for the PUF response, the throughput is 24.1 Kbps for the TRNG (5.30 ms to generate a true random seed of 128 bits) and 144.8 Kbps for the PUF response (6.92 ms to obfuscate or reconstruct a 128-bit secret). Our PUF throughput is higher than the 14.1 Kbps reported in [22] and 0.8 Kbps in [21]. The occupation of the

TABLE V
Multibit PUF Response After RO Selection Using Dataset28nm

	No. measur.	b_1	b_2	b_3	b_4	b_5	b_6
p_i	1	1.00	1.00	1.00	1.00	1.00	1.00
	20	0.98	0.96	0.93	0.86	0.72	0.45
	40	0.97	0.95	0.91	0.82	0.65	0.33
$\overline{HD}_{inter}^{PUF}$	1	50.15	49.86	49.93	49.95	49.97	49.98
	20	50.01	49.98	50.00	49.97	49.90	49.53
	40	50.01	49.99	49.99	49.98	49.86	49.44
$\overline{HD}_{intra}^{PUF}$	1	0.70	1.13	2.10	4.05	8.07	16.10
	20	0.21	0.33	0.63	1.35	3.21	8.30
	40	0.11	0.17	0.33	0.71	1.80	4.67
n_i	1	7	7	9	13	19	37
	20	5	5	7	9	11	19
	40	5	5	5	7	9	13

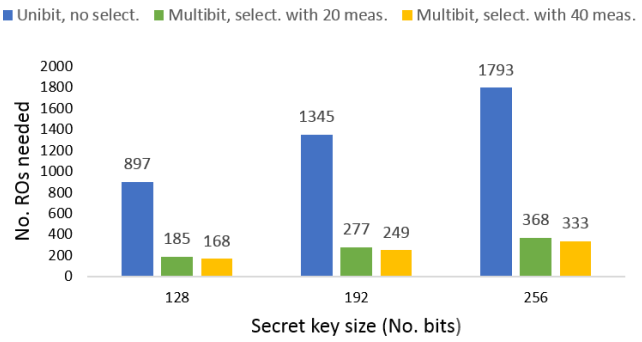


Fig. 4. Number of ROs needed to obfuscate a secret key with 128, 192 and 256 bits, using *Dataset28nm*.

ROs can be reduced considerably at expense of a careful place and route. As reported in [22], each RO can occupy one slice, thus providing higher oscillation frequency. Using fewer stages in the ROs also increases frequency. Further throughput improvement can be achieved if several RO pairs are activated in parallel, at expense of higher power consumption.

As explained in the mathematical model in Section III.b, it is desirable to minimize the number of stages in the ROs for the PUF and TRNG responses. The practical limit to this theoretical result is the appearance of glitches, particularly in the asynchronous counters, since fewer stages increases the oscillation frequency. As a good tradeoff, the ROs were designed with three stages in this peripheral.

Once the values depending on $N_{ij_{ref}}$ and σ_{PUF}, and the binary masks to select the ROs are calculated at calibration stage, the functions to obtain the PUF and TRNG responses together with the Health Tests are programmed by the manufacturer in the internal memory of the CPU, which cannot be read externally. This code is very simple to be executed during the 41.44 μs between the evaluation of RO pairs (the CPU runs at 100 MHz). Since the manufacturer is assumed to be honest, the calibration is assumed to be done correctly.

The proposed multibit-RO-PUF-TRNG is resistant to the side channel passive attacks reported in literature because the knowledge of the average oscillation frequency of each RO is not enough to clone the PUF. The attacker would have to be successful in discovering the $N_{ij_{ref}}$, σ_{PUF}, and the binary masks associated to the ROs, and it is much more difficult to attack the internal memory of the CPU than the external non-volatile memory, which stores the application code of the device.

Advantages of the proposed multibit-RO-PUF-TRNG compared with other RO solutions proposed in the literature for FPGAs are summarized in Table VI. The symbol '-' means that feature has not been addressed specifically in that proposal.

VI. Conclusions

The proposed multibit-RO-PUF-TRNG is a unified design that exploits the same hardware (RO pairs and two counters, one of them as reference) to generate a TRNG response and a multibit PUF response, without the need of a challenging design. The foundations of the circuit and its calibration is supported by a mathematical model, and confirmed with large datasets of experimental results of 28- and 90-nm FPGAs. The proposal has been implemented as an APB peripheral of a VexRiscv RV32I core for an FPGA-based IoT device. The count values of the non-reference counter are processed by the functions programmed in the internal memory of the core, which cannot be accessed externally. Hence, the proposal is more difficult to attack with electromagnetic signals than other RO PUFs and TRNGs reported in literature. The Health Tests programmed in the core to validate the TRNG responses, are also used in the calibration stage to select the reference count value, achieving a good trade-off between throughput and PUF-TRNG metrics (randomness, uniqueness and reliability). The rest of calibration parameters are set after a simple procedure carried out under nominal operating conditions and prior to the device deployment. The proposed selection of the ROs and the use of error-correcting codes adapted to the reliability of the multiple bits generated for the PUF response, allows optimizing the number of ROs required to obfuscate and reconstruct secrets with simple repetition error-correcting codes.

References

[1] Y. Yang et al., "A survey on security and privacy issues in internet-of-things," *IEEE Internet of Things Journal*, vol. 4, no. 5, pp. 1250–1258, 2017.

[2] Y. Gao, S. F. Al-Sarawi and D. Abbott, "Physical unclonable functions", *Nature Electronics*, vol. 3, pp. 81-91, February 2020.

[3] J. Delvaux, R. Peeters, D. Gu, and I. Verbauwhede, "A survey on lightweight entity authentication with strong PUFs," *ACM Comput. Surv.* vol. 48, no. 2, pp. 1-42, Nov. 2015.

[4] W. Che, M. Martin, G. Pocklassery, V. K. Kajuluri, F. Saqib, and J. Plusquellic, "A privacy-preserving, mutual PUF-based authentication protocol," *Cryptography*, vol. 1, no. 1, p. 3, 2017.

[5] M. Ebrahimabadi, M. Younis, and N. Karimi, "A PUF-based modeling-attack resilient authentication protocol for IoT devices", *IEEE Internet of Things Journal*, vol. 9, no. 5, pp. 3684-3703, March 2022.

[6] M. A. Prada-Delgado, I. Baturone, G. Dittmann, J. Jelitto, A. Kind, "PUF-derived IoT identities in a zero-knowledge protocol for blockchain". *Internet of Things*, Elsevier, 2020, vol. 9, 100057.

[7] J. Arcenegui, R. Arjona, R. Román, I. Baturone, "Secure Combination of IoT and Blockchain by Physically Binding IoT Devices to Smart Non-Fungible Tokens Using PUFs". *Sensors*. 21, 9, 3119, 2021.

[8] D. Schellekens, B. Preneel, and I. Verbauwhede, "FPGA vendor agnostic true random number generator," in Proc. FPL 2006, pp. 139-144, 2006.

[9] O. Petura, U. Mureddu, N. Bochard, V. Fischer, L. Bossuet, "A survey of AIS-20/31 compliant TRNG cores suitable for FPGA devices", in Proc. FPL 2016.

[10] I. Baturone, M.A Prada-Delgado, S. Eiroa, "Improved generation of identifiers, secret keys, and random numbers from SRAMs". *IEEE Transactions on Information Forensics and Security*, vol. 10, no 12, pp. 2653-2668. 2015.

[11] J.L. Danger, et al., "Analysis of mixed PUF-TRNG circuit based on SR-latches in FD-SOI technology", in Proc. 21st Euromicro Conference on Digital System Design, DSD 2018, pp. 508-515.

[12] S. K. Satpathy et al., "An All-Digital Unified Physically Unclonable Function and True Random Number Generator Featuring Self-Calibrating Hierarchical Von Neumann Extraction in 14-nm Tri-gate CMOS," *IEEE J. Solid-State Circuits*, vol. 54, no. 4, pp. 1074-1085, April 2019.

[13] S. Taneja, V. K. Rajanna and M. Alioto, "In-Memory Unified TRNG and Multi-Bit PUF for Ubiquitous Hardware Security," *IEEE J. Solid-State Circuits*, vol. 57, no. 1, pp. 153-166, Jan. 2022.

[14] A. Maiti y R. Nagesh, "Physical unclonable function and true random number generator: a compact and scalable implementation", in Proc. on Great Lake Symp. on VLSI (GLSVLSI), 2009, pp. 425–428.

[15] S. Buchovecka, R. Lorencz, F. Kodytek, J. Bucek, "True random number generator based on ring oscillator PUF circuit", *Microprocessors and Microsystems*, 53, pp. 33–41, 2017.

[16] C. Martínez-Gómez, I. Baturone, "Calibration of ring oscillator PUF and TRNG", in Proc. European Conference on Circuit Theory and Design (ECCTD), 2020.

[17] L. Bossuet, X. T. Ngo, Z. Cherif, V. Fischer, "A PUF based on a transient effect ring oscillator and insensitive to locking phenomenon", *IEEE Trans. on Emerging Topics in Computing*, vol. 2, no. 1, pp. 30-36, March 2014.

[18] C. Marchand, et al., "Implementation and characterization of a physical unclonable function for IoT: a case study with the TERO-PUF", in IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, vol. 37, no. 1, pp. 97-109, Jan. 2018.

[19] P. Bayon, L. Bossuet, A. Aubert, V. Fischer, "Fault model of electromagnetic attacks targeting ring oscillator-based true random number generators", *Journ. Cryptogr. Eng.*, vol. 6, pp. 61-74, 2016.

[20] U. Mureddu, B. Colombier, N. Bochard, L. Bossuet, V. Fischer, "Transient effect ring oscillators leak too", in Proc. IEEE Computer Society Annual Symposium on VLSI (ISVLSI), 2019, pp. 37-42.

[21] A. Maiti, J. Casarona, L. McHale, P. Schaumont, "A large characterization of RO-PUF", in Proc. HOST 2010, pp. 66-71.

[22] R. Hesselbarth, F. Wilde, C. Gu and N. Hanley, "Large scale RO PUF analysis over slice type, evaluation time and temperature on 28nm Xilinx FPGAs", in Proc. HOST 2018, pp. 126-133.

[23] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith, "Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data," *SIAM Journal on Computing*, vol. 38, no. 1, pp. 97–139, Mar. 2008.

TABLE VI
COMPARISON OF THE PROPOSED MULTIBIT-RO-PUF-TRNG WITH OTHER PROPOSALS BASED ON ROS FOR FPGAS

	[21]	[22]	[28]	[33]	[29]	[18], [44]	[15], [34]	This work
Unified PUF/TRNG structure	no	no	no	no	no	yes	yes	yes
Multibit PUF response (bits provided)	no	no	no	yes (2 bits)	yes (3 bits)	yes (3 bits)	yes (4 bits)	yes (6 bits)
Mathematical model supporting source entropy	yes	no	no	no	no	yes	no	yes
PUF response without bit aliasing	no	no	yes	no	yes	yes	yes	yes
PUF response with throughput improvement	no	yes	no	no	no	yes	no	yes
Not challenging place&route of ROs	no	no	no	no	no	no	yes	yes
Robust against reported electromagnetic attacks	no	no	-	-	-	no	-	yes
Built-in calibration and test	no	no	no	no	no	no	no	yes

[24] C. Bösch, J. Guajardo, A.-R. Sadeghi, J. Shokrollahi, and P. Tuyls, ``Efficient Helper Data Key Extractor on FPGAs," in Cryptograph. Hardw. Embedded System. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 181_197.

[25] G.E. Suh and S. Devadas,"Physical unclonable functions for device authentication and secret key generation", in Proc. Design Automation Conference, DAC 2007.

[26] A. Maiti, and P. Schaumont, "Improved ring oscillator PUF: an FPGA-friendly secure primitive", *Journ. of Cryptology*, vol. 4 (2), pp. 375-397, April 2011.

[27] S. Eiroa and I. Baturone, "An analysis of ring oscillator PUF behavior on FPGA", in 2011 International Conference on Field-Programmable Technology, New Delhi, India, pp. 1–4, 2011.

[28] L. Feiten, J. Oesterle, T. Martin, M. Sauer, and B. Becker, "Systemic frequency biases in ring oscillator PUFs on FPGAs", *IEEE Trans. on Multi-scale Computing Systems*, 2 (3), pp. 174–185, 2016.

[29] H. Mandry, A. Herkle, S. Müelich, J. Becker, R. F. H. Fischer, and M. Ortmanns, "Normalization and multi-valued symbol extraction from RO-PUFs for enhanced uniform probability distributions", *IEEE Trans. on Circuits and Systems-II: Express Briefs*, vol. 67 (12), pp. 3372-3376, 2020.

[30] C.-E. D. Yin and Q. Gang, "LISA: Maximizing RO PUF's secret extraction", in Proc. IEEE Int. Symp. on Hardware-Oriented Security and Trust (HOST), 2010, pp. 100–105.

[31] W. Liu, Y. Yu, C. Wang, Y. Cui, and M. O'Neill, "RO PUF design in FPGAs with new comparison strategies," in Proc. IEEE Int. Symp. Circuits Syst. (ISCAS), pp. 77–80, 2015.

[32] J. Park, B. Kim and J. -Y. Sim, "A BER-Suppressed PUF With an Amplification of Process Mismatch Effect in an Oscillator Collapse Topology," *IEEE J. Solid-State Circuits*, vol. 57, no. 7, pp. 2208-2219, July 2022.

[33] H. Martin, P. Peris-Lopez, G.D. Natale, M. Taouil, S. Hamdioui, "Enhancing PUF Based Challenge-Response Sets by Exploiting Various Background Noise Configurations," *Electronics*, vol. 8 (145), 2019.

[34] F. Kodýtek, R. Lórencz, and J. Bucek, "Improved ring oscillator PUF on FPGA and its properties", *Microprocessors and Microsystems*, 47, pp. 56–63, 2016.

[35] F. Kodytek, R. Lorencz, and J. Bucek, "Comparison of three counter value based ROPUFs on FPGA", in Proc. 23rd Euromicro Conference on Digital System Design, DSD 2020, pp. 205-212.

[36] B. Valtchanov, A. Aubert, F. Bernard, V. Fischer, "Modeling and observing the jitter in ring oscillators implemented in FPGAs", in Proc. 2008 11th IEEE Workshop on Design and Diagnostics of Electronic Circuits and Systems.

[37] M. A. Prada-Delgado, C. Martínez-Gómez, I. Baturone, "Auto-calibrated ring oscillator TRNG based on jitter accumulation", in Proc. IEEE Int. Symp. Circuits Syst. (ISCAS), 2020.

[38] Y. Cao, X. Zhao, W. Zheng, Y. Zheng and C. -H. Chang, "A New Energy-Efficient and High Throughput Two-Phase Multi-Bit per Cycle Ring Oscillator-Based True Random Number Generator," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 69, no. 1, pp. 272-283, Jan. 2022.

[39] P. Sedcole and P. Y. K. Cheung, "Within-die delay variability in 90nm FPGAs and beyond," in Proc. 2006 IEEE Int. Conf. on Field Programmable Technology, 2006, pp. 97-104.

[40] A. A. Abidi, "Phase Noise and Jitter in CMOS Ring Oscillators," *IEEE Journal of Solid-State Circuits*, vol. 41, no. 8, pp. 1803-1816, Aug. 2006.

[41] M.S. Turan, E. Barker, J. Kelsey, K.A. McKay, M.L. Baish, M. Boyle, "Recommendation for the entropy sources used for random bit generation," NIST Special Publication 800-90B, 2018.

[42] A. Rukhin et al., "A statistical test suite for random and pseudorandom number generators for cryptographic applications," NIST Special Publication 800-22-1a, 2010.

[43] Murax SoC available at https://github.com/SpinalHDL/VexRiscv#murax-soc

[44] A. Cherkaoui, L. Bossuet, C. Marchand, "Design, Evaluation, and Optimization of Physical Unclonable Functions Based on Transient Effect Ring Oscillators", in *IEEE Trans. on Information Forensics and Security*, vol. 11, no. 6, pp. 1291-1305, June 2016.

Iluminada Baturone received the 5-year degree (Hons.) in physics and the Ph.D. degree (Hons.) in physics-electronics from the University of Seville, Seville, Spain, in 1991 and 1996, respectively.

She is Full Professor at the University of Seville with the Department of Electronics and Electromagnetism and Senior Researcher at the Microelectronics Institute of Seville (IMSE-CNM), University of Seville, CSIC. She has coauthored 2 books and more than 150 scientific articles. She has filed 4 patents and has participated in more than 40 Spanish and European research and industrial projects, leading 13 of them. Her current research interests include hardware security, post-quantum cryptography, biometrics, blockchain technologies, and neuro-fuzzy systems.

Roberto Román received the bachelor's degree in Electronic Engineering of Telecommunications from the University of Barcelona in 2018 and the Master of Science in Microelectronics from the University of Seville in 2020.

He is currently pursuing a Ph.D. in Microelectronics at the Microelectronics Institute of Seville (IMSE-CNM), University of Seville, CSIC. In 2019 he won a JAEIntro grant of CSIC and now his research is supported by the VI Plan Propio de Investigación y Transferencia of the University of Seville. His current research interests are post-quantum cryptography, blockchain technologies, self-sovereign identities, and biometrics.

Ángel Corbacho received the bachelor's degree in Electronic and Automation Engineering from University of Extremadura, Badajoz, Spain, in 2019, and the master's degree in Electronic, Automation and Robotic Engineering from University of Seville, Seville, Spain, in 2020.

From 2020 to 2021, he was with the Microelectronics Institute of Seville (IMSE-CNM), University of Seville, CSIC, as a Researcher.