A Ring Oscillator-Based Identification Mechanism Immune to Aging and External Working Conditions

Mario Barbareschi, Giorgio Di Natale, Lionel Torres, Antonino Mazzeo

A Ring Oscillator-Based Identification Mechanism Immune to Aging and External Working Conditions

Mario Barbareschi, Giorgio Di Natale, *Senior Member, IEEE*,
Lionel Torres, and Antonino Mazzeo

***Abstract*— Physically unclonable functions (PUFs) are one of the most important breakthrough for security of devices as they represent a low-cost means to provide authentication and secure storage. PUFs measure nano-scale mismatches that are inherently caused by the manufacturing process. However, the mechanisms exploited by PUF circuits depend on the working conditions, such as temperature, device aging, and current flow, making them unreliable and, hence preventing their wider employment. One of the most investigated PUF exploits pairs of ring oscillators (ROs): frequencies measured from each pair are compared for extracting one response bit. However, extracted bit-strings are not suitable for authentication purposes as they may change during time. In this paper, we propose a new identification mechanism, based on ROs, which is immune to aging and working conditions. Through a mathematical demonstration and an extensive experimental campaign, which involved real field programmable gate array devices, we demonstrate its ability to reliably accomplish identification of silicon devices.**

***Index Terms*— Physically unclonable function, hardware security, identification, ring oscillator, reliability.**

I. Introduction

NOWADAYS, identification and authentication are fundamental operations, which electronic devices have to implement and offer at hardware level. Common schemes involve asymmetric cryptographic keys, namely the identity of a device matches with its own private key. Even if cryptography guarantees a mathematic proof of security, it requires a secure storage to save and secretly keep the key. Unfortunately, memories are prone to several attacks, which succeeds in extracting sensitive data that they retain, jeopardizing secure schemes that are based on a secret.

Recently, some security mechanisms were modified by introducing silicon Physically Unclonable Functions (PUFs), since they intrinsically provide very attractive security properties. In fact, PUFs rely on nanoscale imperfections imprinted by the manufacturing process variability, which are random, uncontrollable and unpredictable, such that each manufactured silicon device turns out to be unique. Hence, two PUFs do not behave the same way and none can arbitrarily manufacture a device with a PUF that exhibits the same characteristics of another one. Therefore, PUFs are suitable to be the digital identity of devices on which they are embedded.

So far, a significant number of PUF architectures has been introduced, covering delay-based PUFs (Arbiter [1], Ring Oscillator [2], Anderson [3], CNN-based [4]), which use pairs of symmetric paths for a digital contest to establish the output, and memory-based PUFs (SRAM [5], Butterfly [6], STT-MRAM [7]), which exploit the random initial pattern of a memory cell when being powered up.

Among delay-based architectures, the ring oscillator (RO) PUFs have gained significant attention since the RO is a very common primitive available on every integrated circuit technology. in particular, ROPUFs compare frequencies measured from pairs of ROs to extract bit-strings.

Unluckily, the output of a PUF is not stable during time, since it is influenced by working conditions, such as voltage supply and temperature, or by the device aging process. Moreover, it may be not completely random when being compared among different devices. In order to overcome these issues and make PUFs eligible as secure primitive, the literature promoted some post-processing techniques, which are able to restore both the stability and randomness, but they require a significant effort in terms of design and resources overhead [8].

Bearing in mind previous considerations, in this paper we propose the Frequency Signature-based PUF (FSPUF), a basic mechanism for authentication and identification of digital devices based on ROs, which is immune to working conditions and aging. Contrary to a classic PUF architectures, our proposal does neither exploit differential measurements nor provide a string of bits, but introduces a method to obtain condition-independent signature and defines a proper function to compare two signatures. We formally provide a proof of its effectiveness and, through a large set of real experiments, performed over on 20 different Field Programmable Gate Array (FPGA) devices and under several different working conditions, namely varying the temperature, the voltage and

Manuscript received February 2, 2017; revised June 22, 2017; accepted July 2, 2017. Date of publication August 9, 2017; date of current version January 25, 2018. This paper was recommended by Associate Editor F. J. Kurdahi. *(Corresponding author: Mario Barbareschi.)*

M. Barbareschi and A. Mazzeo are with the Department of Electrical Engineering and Information Technologies, University of Naples Federico II, 80138 Napoli, Italy (e-mail: mario.barbareschi@unina.it; antonino.mazzeo@unina.it).

G. Di Natale and L. Torres are with the Laboratoire d’Informatique, de Robotique et de Microélectronique de Montpellier, University of Montpellier, 4090 Montpellier, France (e-mail: giorgio.di.natale@lirmm.fr; lionel.torres@lirmm.fr).

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

Digital Object Identifier 10.1109/TCSI.2017.2727546

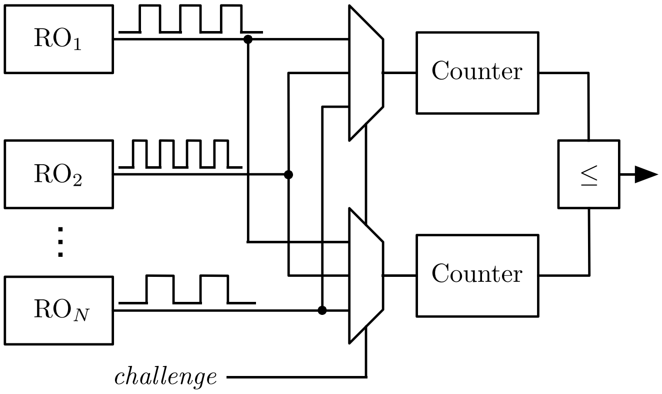


Fig. 1. Ring Oscillator based PUF architecture: a functional overview.

by using fresh and aged devices, we show that the FSPUF performs the identification better than other RO-based PUFs, such as the ROPUF and CROPUFs.

The remainder of the paper is structured as follows: Section II briefly collects related works; in Section III we detail a model of RO that we exploit to define, in Section IV, the Frequency Signature based PUF (FSPUF); Section V illustrates the FSPUF execution over Xilinx Spartan-6 FPGA devices, while Section VI concludes this paper.

II. Overview on the Ring Oscillator

The RO is a widespread adopted circuit and thanks to the ease of implementation in hardware description languages (HDLs), it can be used in any manufacturing technology. As for security aspects, ROs are mainly exploited to implement secure primitives, such as the True Random Number Generator (TRNG) and the PUF. The former is an unstable circuit that has to output a stream of random bits, conversely the latter is a primitive that is exploited to extract unique and stable bit-strings from hardware devices. Both rely on ROs, but differently, as the TRNG exploits the randomness of oscillations jitter [9], [10], while the PUF exploits the randomness of oscillation frequencies for different instances of ROs [2], [11].

Figure 1 illustrates a high level schematic of a ROPUF architecture. The mechanism exploited by the PUF is a differential frequency measurement operation. Providing a challenge, a pair of ROs is selected and two frequencies are measured by means of counters. Considering f_a and f_b the pair of measured frequencies, the PUF gives one response bit comparing them as follows:

$$r = \begin{cases} 1 & f_a \leq f_b \\ 0 & f_a > f_b \end{cases} \tag{1}$$

Each instance of RO is affected by variations and imperfections, randomly caused by the manufacturing process, such that it is characterized by a frequency that differs from one another, within the same chip and among different ones. Consequently, each pair of a single chip gives a response that depends on the random distribution of manufacturing variations, which are unique for each manufactured chip.

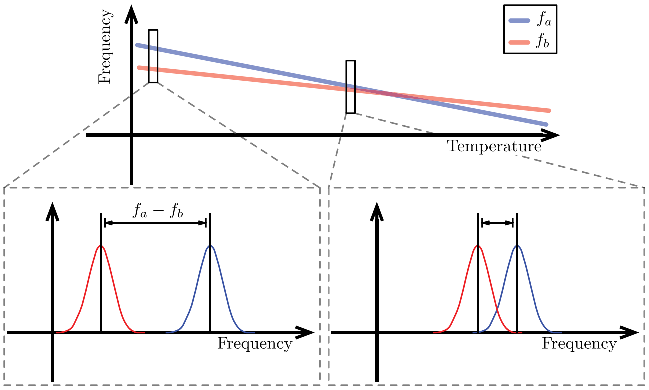


Fig. 2. Temperature, as well as other working parameters, are able to differently affect two ROs, causing unstable responses for the ROPUF.

As f_a and f_b require a measurement method in order to be characterized, their values are affected by the error measurement, such that successive measurements do not give the same value. The error is generated by the usage of a time source that works as time reference in which measure the frequency of oscillations of a RO. Such a time reference is surely not aligned in phase with RO oscillations and, for that reason, their phase offset is unknown, generating an error on measurements that turns out different each time.

Considering the Equation 1, if random distributions associated to the picked frequencies pair are characterized by two mean values that are very close to each other or, more generally, by two distributions that have a significant frequency range in common, the bit response is strongly affected by noise and that might change its value during time. Even if f_a and f_b turn out to be far enough, due to external conditions variations, such as temperature changes or supply voltage fluctuations, they may generate unstable response. For instance, with regards to the temperature variations, each RO frequency decreases with the temperature in a manner that depends on the physical characteristics of the circuit. Thus, even if the distance between two frequency values is great enough, by operating under a significant temperature increase, such frequency values might reverse their mutual order, as pictured in Figure 2.

A. Research Works on RO Characterization

In the literature, a significant amount of research papers addressed the characterization of ROs and the sensitiveness problem of ROs from external and uncontrolled working conditions. Sedcole *et al.* in [12] showed a characterization analysis for ROs implemented on 18 Altera Cyclone II FPGAs, demonstrating a systematic process variability. In [13], Maiti *et al.* characterized the ROs frequencies over a population of 193 Xilinx Spartan-3E S500 FPGAs, giving details about an implementation of ROPUF. Later, the authors of [14] gave a deeper analysis of frequency distribution used in [13]. Through the Anderson-Darling test, similarity analysis and principal component analysis, they demonstrated that ROs are eligible secure primitives, since they are well distributed among different devices and, hence, hard to predict. Authors of [15] proposed a technique to mitigate the temperature effect on the ROPUF responses stability. Merli *et al.* in [16] addressed the problem of the logic that surrounds RO, giving an exhaustive frequencies characterization on a Xilinx Spartan-3E devices, with the aim to define a useful technique to deal with PUF response instability. Amouri *et al.* analyzed the transistor aging effect on Xilinx Spartan 6 FPGA devices exploiting the oscillation frequencies from ROs [17]. In particular, they stressed devices and measured the impact of such a stress on ROs frequencies, demonstrating a degradation of 5.17% on read frequencies.

As for the ROPUF, since first introduction in [2], the ROPUF has been getting a huge interest by research community, mainly trying to improve responses quality [18]–[20] and enhance the responses stability [11], [21], [22]. More recently, Liu *et al.* in [23] demonstrated, over data collected by Maiti *et al.* in [11], that a symmetrical strategy leads to a better entropy on the bitstring extracted from a ROPUF. Agusting *et al.* in [24] proposed a fully-configurable RO for changing the oscillations duty cycle. They claimed that the duty cycle is a more robust value against temperature variations and aging effects and the devised RO configurations could be applied to enhance PUF quality. These proposals, contrary to FSPUF, which considers absolute values of frequency, take into account pairs of ROs and rely on differential schemes.

III. A Model for Ring Oscillator Frequency

Ideally, the ROPUF has to work with frequencies that have stable and deterministic values during time. But there are some unavoidable uncertainty sources, mainly caused by electric phenomena and by the measurement operation itself. The mere RO frequency value, of course, depends on imperfections of transistors and wires, caused by manufacturing process, and on working conditions, including the aging effect.

In order to try to characterize measured frequencies from ROs by means of a formal model, we assume to have N ROs implemented on a device that are identically manufactured, M measured samples for all the ROs, which are gathered by using the same measurement operation each time, and a group of D devices. Let n, m, d be the indexes indicating, respectively, a quantity measured from the n-th instance of a RO at m-th time on the d-th device. Therefore, a read frequency f is a quantity that can be characterized as follows:

$$\begin{aligned} f_{n,m,d} &= \widehat{f}_{n,m,d} + \varepsilon_{n,m,d} + t_{n,m,d}, \\ n &\in [0, N-1], m \in [0, M-1], d \in [0, D-1]. \end{aligned} \tag{2}$$

In the Equation 2, $\widehat{f}$ is the inherent frequency of the oscillations produced by the RO, ε is the error associated with the measure and t is a quantity that takes into account the uncontrolled operational conditions, such as the temperature.

The way in which we can characterize each member of the Equation 2 differs from one another. As for the $\widehat{f}$, frequencies associated to ROs are quantities that strictly depend only on the summation of delays generated by each component in their closed loop, such as transistors, wires and parasitic capacitance. Ideally, each manufactured RO should be characterized by a delay that is always the same, i.e. $\widehat{f} = \widehat{f}_{n,d}, \forall n, d$, but, conversely, components integrated in the loop suffer from variability when manufactured. Consequently, each RO is characterized by a delay that is distributed around an average value, which is the one established at design time. If there are not static process variations, which lead to imperfections that are not fully random among ROs, resulting delays depend on pure random physical characteristics and, hence, $\widehat{f}$ is a random variable. Since it is related to the process variability, the distribution of $\widehat{f}$ can be considered normal. The mean and variance of such distribution tightly depend on the RO design, for instance the number of stages, on the implementation, for instance the place and the route of its internal components, and on the productive technology.

Moreover, we can simplify $\widehat{f}_{n,m,d}$ omitting the index m, indicating that such a quantity does not vary during time.

The measurement operation introduces an error ε, which is random and cannot be directly controlled, but at least can be estimated. We can assume that it is an additive white gaussian noise (AWGN), hence its values are distributed all around the 0: formally $\varepsilon \sim \mathcal{N}\left(0, \sigma_{\varepsilon}^{2}\right)$. By keeping constant external conditions (i.e. at fixed temperature, fixed supplied voltage, and so on), such that $t_{n,m,d} = t_{n,d}, \forall m$, and considering that ε afflicts all the measures $f_{n,m,d}, \forall n, m, d$, its mean and standard deviation can be statistically estimated over M measurements as follows:

$$\begin{aligned} f_{n,d} &= \frac{1}{M}\sum_{m=0}^{M-1} f_{n,m,d} = \frac{1}{M}\sum_{m=0}^{M-1}\left(\widehat{f}_{n,d} + \varepsilon_{n,m,d} + t_{n,m,d}\right) \\ &= \widehat{f}_{n,d} + t_{n,d} + \frac{1}{M}\sum_{m=0}^{M-1}\varepsilon_{n,m,d} = \widehat{f}_{n,d} + t_{n,d} + \varepsilon_{n,d}. \\ \sigma_{\varepsilon_{n,d}} &= \sqrt{\frac{\sum\limits_{m=0}^{M-1}\left(f_{n,m,d} - f_{n,d}\right)^{2}}{M}} = \sqrt{\frac{\sum\limits_{m=0}^{M-1}\left(\varepsilon_{n,m,d} - \varepsilon_{n,d}\right)^{2}}{M}}. \end{aligned} \tag{3}$$

Then, σ_{ε} can be globally estimated by considering the average value of the variance associated to ROs and devices:

$$\sigma_{\varepsilon} = \frac{1}{D}\sum_{d=0}^{D-1}\left(\frac{1}{N}\sum_{n=0}^{N-1}\sigma_{\varepsilon_{n,d}}\right) = \frac{1}{D \cdot N}\sum_{d=0}^{D-1}\sum_{n=0}^{N-1}\sigma_{\varepsilon_{n,d}}.$$

The last variable in the Equation 2 (t) can be considered as a value that causes a shift for the frequency value. Indeed, external working conditions, such as temperature and voltage, just shift the measured value against the expected one. Of course, such shift quantity is strictly coupled with the frequency reference. For instance, if we could define a reference frequency measured under standard fixed conditions of temperature (e.g. 25°C), voltage (e.g. 1.2V), etc., we were able to properly define t as a function of that conditions. For the scope of this work, t does not need to be defined in someway, since, as shown in the next section, it is not taken into consideration by the proposed technique.

A. Frequencies as a Signature

This Subsection introduces the device signature, defined as collection of some measured frequencies. With this goal, let $S_{m,d}$ be the vector of frequencies associated to the d-th device; m indicates a single measurement among different trials, accomplished in order to have more than one signature, for each device, in different times. To better clarify, m can be considered as the discrete time variable, since $S_{m,d}$ and $S_{m',d}$ are collected from the same device, but in two different instants. Formally, $S_{m,d}$ is a N dimensional vector defined as follows:

$$\begin{aligned} S_{m,d} &= \left\{f_{n,m,d}\right\}, \quad \forall n \in [0,\ldots,N-1] \\ &= \left\{f_{0,m,d}, f_{1,m,d}, \ldots, f_{N-1,m,d}\right\} \in \mathbb{R}^N. \end{aligned}$$

Substituting each component $f_{n,m,d}$ with the Equation 2, we get a signature definition that is actually expressed as a sum of three vectors:

$$\begin{aligned} S_{m,d} = \left\{f_{n,m,d}\right\} = \left\{\widehat{f}_{n,m,d}\right\} \\ + \left\{\varepsilon_{n,m,d}\right\} + \left\{t_{n,m,d}\right\}, \quad \forall n \in [0, N-1]. \qquad (4) \end{aligned}$$

Let us now discuss about properties of each component of the vector. First, as anticipated before, the $\widehat{f}$ does not depend on m, because it is not the actual oscillation frequency, but it has to be considered together with the quantity t.

As for the measurement error, each n-th component of $S_{m,d}$ is affected by a value $\varepsilon_{n,m,d}$. We can assume that the distribution of ε does not depend on the specific device, trial or RO, since the error is given by frequency measurement process itself and all the considering ROs are designed and manufactured in the same manner. Consequently, at each measurement it can be considered as a value of an AWGN process. Therefore, ε exhibits the same behavior averaged over time ($\underset{m}{E}\left[\varepsilon_{n,m,d}|n,d\right]$), averaged over the realizations ($\underset{n}{E}\left[\varepsilon_{n,m,d}|m,d\right]$) and averaged over the devices ($\underset{d}{E}\left[\varepsilon_{n,m,d}|n,m\right]$). In fact, we are considering a unique measurement procedure for all ROs and for all sampled values, hence the AWGN process is not subject to variations. Of course, the measurement design has to be symmetric with respect to ROs and the same for all the devices.

$$\underset{m}{E}\left[\varepsilon_{n,m,d}|n,d\right] = \frac{1}{M}\sum_{m=0}^{M-1} \varepsilon_{n,m,d} \xrightarrow{M\to\infty} 0. \qquad (5a)$$

$$\underset{n}{E}\left[\varepsilon_{n,m,d}|m,d\right] = \frac{1}{N}\sum_{n=0}^{N-1} \varepsilon_{n,m,d} \xrightarrow{N\to\infty} 0. \qquad (5b)$$

$$\underset{d}{E}\left[\varepsilon_{n,m,d}|n,m\right] = \frac{1}{D}\sum_{d=0}^{D-1} \varepsilon_{n,m,d} \xrightarrow{D\to\infty} 0. \qquad (5c)$$

The Equations 5 are obliviously valid for a significant number of instances of ε values, consequently with finite frequency samples from ROs, we are only able to compute estimators that reach 0 when the number of samples approach the infinite.

As previously stated, t defines, for each component, a frequency shift due to operational conditions. Among all the N components, t has not the same value (see Figure 2). Indeed, external conditions might affect differently the frequency because of manufacturing variations. Moreover, being implemented into integrated circuits, each RO could be affected by local effects that characterize its surrounding area. Consequently, even if under the same working conditions, t is a quantity that slightly differs along the N components of the signature $S_{m,d}$, being directly caused by the manufacturing process variability. Therefore, the assumption that all the t_n are equal for each component introduces a negligible approximation, w.r.t. the error value and the inherent frequency value. Hereafter $t_m = t_{n,m}, \forall n \in [0,\ldots,N-1]$.

Taking into account previous considerations, $S_{m,d}$ can be simplified in the following form:

$$S_{m,d} = \left\{\widehat{f}_{n,d}\right\} + \left\{\varepsilon_{n,m,d}\right\} + t_m.$$

Let us now consider the average value of the N components of a signature $S_{m,d}$, which has the following form:

$$\begin{aligned} &\underset{n}{E}\left[S_{m,d}|m,d\right] \\ &\quad = \underset{n}{E}\left[\widehat{f}_{n,d}|m,d\right] + \underset{n}{E}\left[\varepsilon_{n,m,d}|m,d\right] + \underset{n}{E}\left[t_m|m,d\right] \\ &\quad \approx \mu_{\widehat{f}_d} + t_m. \qquad (6) \end{aligned}$$

The Equation 6 contains two approximations. The first one is on the $\underset{n}{E}\left[\varepsilon_{n,m,d}|m,d\right]$: being an estimation on only N values, the evaluated error mean approximate the 0 (Equation 5b). The second approximation is on the mathematical expectation of $t_{n,m}$, because, as discussed before, it is different among the N components. Hence, $\underset{n}{E}\left[t_m|m,d\right] - t_m \approx 0$.

Interestingly, $\underset{n}{E}\left[S_{m,d}|m,d\right]$ contains the value t_m, which can be exploited to remove it from each component of $S_{m,d}$. Let $\widetilde{S}_{m,d}$ be the signature obtained by subtracting from each component of $S_{m,d}$ the average value $\underset{n}{E}\left[S_{m,d}|m,d\right]$:

$$\begin{aligned} \widetilde{S}_{m,d} &= S_{m,d} - \underset{n}{E}\left[S_{m,d}|m,s\right] \approx \left\{\widehat{f}_{n,d}\right\} - \mu_{\widehat{f}_d} + \left\{\varepsilon_{n,m,d}\right\} \\ &= \left\{\widetilde{f}_{n,d}\right\} + \left\{\varepsilon_{n,m,d}\right\}, \quad \forall n \in [0,\ldots,N-1]. \qquad (7) \end{aligned}$$

Consequently, $\widetilde{S}_{m,d}$ does not depend on the working conditions or other effects, and each frequency component has a new value, given by $\widehat{f}_{n,d} - \mu_{\widehat{f}_d} = \widetilde{f}_{n,d}$. In particular, $\widetilde{f}_{n,d}$ now is characterized by the gaussian distribution of $\widehat{f}_{n,d}$, but the mean value is close to 0 by definition, being $\mu_{\widehat{f}_d}$ an estimation of the average value. Such a distribution depends only on manufacturing variations and each value $\widetilde{f}_{n,d}$ is exclusively related on them. Furthermore, $\widetilde{f}_{n,d}$ depends on the involved ROs and, hence, on the number of ROs (N) in the signature, because picking different ROs changes the value $\mu_{\widehat{f}_d}$. Therefore, even if it is not possible to extract the pure $\widehat{f}_{n,d}$, at least $\widetilde{S}_{m,d}$ contains an expression of measured frequencies, which are not actually influenced by working conditions.

Figure 3 illustrates $S_{m,d}$ and $\widetilde{S}_{m,d}$ evaluated for 20 5-stages ROs, implemented on a Xilinx Spartan-6 device. Frequencies are measured by means of counters at 6 different temperatures (Figures 3a and 3b) and at 6 different voltage values (Figures 3c and 3d). $\widetilde{S}_{m,d}$, as observed from the Equation 7, is free of any shift caused by external conditions (in this case temperature and voltage), and only a small error can be appreciated over values, given by the introduced approximations. Moreover, all the components are distributed around 0, as a consequence of the subtraction of $\mu_{\widehat{f_d}}$.

B. Compute $\widetilde{S}_{m,d}$ in Hardware

Having the whole PUF mechanism implemented in hardware guarantees a stronger security than PUF designs that involve software modules, such as in the case of noisy responses that have to be recovered by post-processing techniques, as the fuzzy extractor schemes [25]. The software trustworthiness relies on the hardware, which has to guarantee protection against attacks. If the PUF is one of the mechanisms that cooperates to enable the system security, the software needs to be trusted in someway, with additional precautions, e.g. implementing a chain of trust. Since the main goal of this paper is to define an authentication and identification mechanism that would be available on every silicon technology, without implying any software module, $\widetilde{S}_{m,d}$ has to be available directly in hardware.

Once the frequencies are read from the ROs, to obtain $\widetilde{S}_{m,d}$ it is necessary to calculate the average value of $S_{m,d}$ and subtract it to each frequency component of the signature. As for the average value, it involves the addition of all the frequency and the division by N, that is a natural number. To avoid the adoption of a division algorithm, which requires a significant amount of resources to be accomplished in hardware, the PUF introduced in this paper can be designed with a signature $S_{m,d}$, which number of ROs is a power of 2: $N = 2^x$. Indeed, the evaluation of the mean from 2^x binary values requires to add them without considering the x least significant bits, since the division by a power of two, induced by the mean, is equivalent to a right shift. The possible loss of precision caused by the shifted-out bits needs to be evaluated against the frequency value. Anyway, with important frequency values the decimal truncation is not critical, since lost bits represent the decimal part of the mean value against really higher values.

Therefore, in order to obtain $\widetilde{S}_{m,d}$ in hardware, the only hardware operations required are the addition and the subtraction.

C. Characterization of $\widehat{f}$

The model introduced by the Equation 2 can be statistically characterized by means of measurements campaigns. As for the measurement error, we assumed ε as an AWGN process in every realization (along n, m and d). The standard deviation associated to the gaussian process depends on the specific circuit realized to extract the frequency values, while the mean value is 0. Consequently, from each $f_{n,d}$ we can estimate a value that is error-free by averaging the measured values on a significant number of trials (Equation 5a):

$$\begin{aligned} f_{n,d} &= \frac{1}{M}\sum_{m=0}^{M-1} f_{n,m,d} = \frac{1}{M}\sum_{m=0}^{M-1}\left(\widehat{f}_{n,d} + \varepsilon_{n,m,d} + t_m\right) \\ &= \widehat{f}_{n,d} + \mu_{t_d} + \frac{1}{M}\sum_{m=0}^{M-1}\varepsilon_{n,m,d} \approx \widehat{f}_{n,d} + \mu_{t_d}. \qquad (8)\end{aligned}$$

Contrary to the Equation 3, in the Equation 8 we do not need to assume that the external conditions do not change from a measurement to another one.

The random variable $\widehat{f}$ can be characterized in two dimensions. The first one is considering frequencies measured from ROs within the same device, while the second is considering frequency sampled from ROs belonging to different devices. Let f_d be the average value obtained from N ROs, that is the mean of the frequency for a device d:

$$f_d = \frac{1}{N}\sum_{n=0}^{N-1} f_{n,d} \approx \mu_{\widehat{f_d}} + \mu_t$$

The standard deviation of frequencies of devices around the mean value f_d, that is the can be statistically defined as follows:

$$\sigma_d = \sqrt{\frac{\sum_{n=0}^{N-1}\left(f_{n,d} - f_d\right)^2}{N}} = \sqrt{\frac{\sum_{n=0}^{N-1}\widetilde{f}^{2}_{n,d}}{N}}$$

This statistical estimator measures the uniqueness of signatures generated by the device d as it indicates how disperse are values from the average frequency. By taking into account its average value among different D devices, we are able to retrieve the σ_{intra} as:

$$\sigma_{\text{intra}} = \frac{1}{D}\sum_{d=0}^{D-1}\sigma_d$$

Conversely, to estimate the σ_{intra}, we need to define the f_n, that is the average value over D devices considering the n-th RO:

$$f_n = \frac{1}{D}\sum_{d=0}^{D-1} f_{n,d} \approx \mu_{\widehat{f_n}} + \mu_t$$

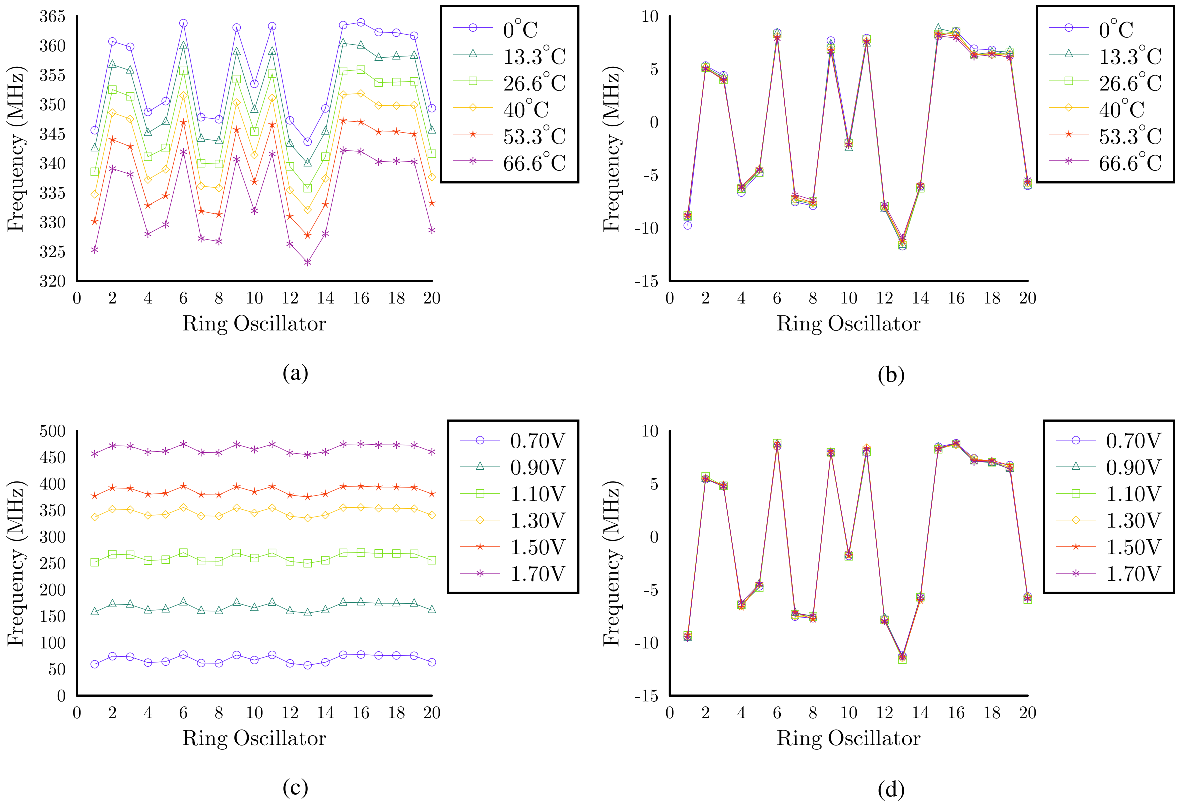
Then, the dispersion of frequencies among different devices around the value f_n can be evaluated as follows:

$$\sigma_n = \sqrt{\frac{\sum_{d=0}^{D-1}\left(f_{n,d} - f_n\right)^2}{D}}$$

The value σ_n, contrary to σ_d, is an estimator of the signatures uniqueness generated by different devices, and, evaluating the average value for all N ROs, we are able to define the σ_{inter} as follows:

$$\sigma_{\text{inter}} = \frac{1}{N}\sum_{n=0}^{N-1}\sigma_d$$

As for the FSPUF, both σ_{intra} and σ_{inter} are relevant, since the first is directly related to the entropy that can be extracted on the same device by involving different ROs, while the second gives a quantitative value of the signature uniqueness when the frequency extracted from ROs are used to identify the devices.



(a) (b)

(c) (d)

Fig. 3. Comparison between $S_{m,d}$ and $\widetilde{S}_{m,d}$. (a) Linear plot of $S_{m,d}$ with 20 ROs frequencies sampled at different temperatures. (b) Linear plot of $\widetilde{S}_{m,d}$ with 20 ROs frequencies sampled at different temperatures. (c) Linear plot of $S_{m,d}$ with 20 ROs frequencies sampled at different voltages. (d) Linear plot of $\widetilde{S}_{m,d}$ with 20 ROs frequencies sampled at different voltages.

IV. Frequencies Signature Based PUF

This Section introduces a new technique to recognized a device by exploiting a signature obtained by ROs. Contrary to available ROPUF architectures, which compares two ROs to extract a 1-bit response, this technique considers directly the frequency values to discriminate devices. For this reason, the devised PUF is named Frequencies Signature-based PUF (FSPUF). The FSPUF does not provide a stable sequence of bits, but rather it introduces a mechanism to distinguish two signatures generated from an array of ROs from two devices. Such a mechanism does not work in challenge-response manner, since the FSPUF gives a single response value exploited as a unique ID. Most important, by transforming read frequencies previously introduced in the Subsection III-A, the FSPUF is inherently immune to external conditions. The FSPUF effectiveness is here mathematically demonstrated and empirically evaluated trough experimental results in the next Section.

A. FSPUF: Signatures Comparison

The aim of the FSPUF is to provide an effective method to authenticate devices by means of measured frequencies from ROs. Given a test procedure ξ and two signatures $S_{m,d}$, $S_{m',d'}$, the authentication mechanism has to work as follows:

$$\xi\left(\widetilde{S}_{m,d}, \widetilde{S}_{m',d'}\right) = \begin{cases} 1 & d = d', \quad \forall m, m' \\ 0 & d \neq d', \quad \forall m, m' \end{cases} \tag{9}$$

The ξ function discriminates the case in which d and d' represents the same device or not, giving in output 1 whenever the authentication succeeds. Implicitly, the ξ has to recognize the signature of the devices even though they are generated in different instants, hence under different working conditions. Furthermore, ξ has to be suitable for a hardware implementation. As stated before, this enables to embed not only the ROs within a device, but also the comparison mechanism requiring a minimal overhead in area occupancy.

There are many available metric functions, which are able to compare two vectors to measure how similar are them. For instance, the euclidean distance or the angles between two vectors could be candidates of ξ function, but they involve too much complex arithmetic operations that needs for floating point units. Further, their outputs are not binary, as represented by the Equation 9, and need for further evaluations to decide if two signatures are generated from the same devices. Indeed, once computed, the euclidean distance and the angle formed by vectors give a number that has to be compared with a threshold. Whenever the distance between two signature is under the threshold or whenever the angle falls within a given range, they are enough similar and, hence, are classified as provided by the same device. The threshold definition is not a trivial task and even not so effective, since the choice of such a value affects all FSPUF instances effectiveness. To better clarify, let $\left\|W_{d,d'}\right\|$ be the euclidean norm (L^2 norm) of the vector obtained by subtraction of the homologous components of $\widetilde{S}_{m,d}$ and $\widetilde{S}_{m',d'}$. Substituting each signature with their definition given by Equation 7, we get:

$$\begin{aligned} \|W_{d,d'}\| &= \|W_{d',d}\| = \|\widetilde{S}_{m,d} - \widetilde{S}_{m',d'}\| \\ &= \left\|\left\{\widetilde{f}_{n,d} - \widetilde{f}_{n,d'}\right\} + \left\{\varepsilon_{n,m,d} - \varepsilon_{n,m',d'}\right\}\right\| \\ &= \left\|\Delta\widetilde{f}_{n,d,d'} + \Delta\varepsilon_{n,m,m',d,d'}\right\| \\ &= \begin{cases} \sqrt{\sum\limits_{n=0}^{N-1} \Delta\varepsilon^2_{n,m,m',d,d'}} & d = d' \\ \sqrt{\sum\limits_{n=0}^{N-1} \left(\Delta\widetilde{f}_{n,d,d'} + \Delta\varepsilon_{n,m,m',d,d'}\right)^2} & d \neq d' \end{cases} \end{aligned}$$

The corresponding ξ function obeys to this inequality:

$$\xi\left(\widetilde{S}_{m,d}, \widetilde{S}_{m',d'}\right) = \|W_{d,d'}\| \leq \tau. \tag{10}$$

The goal of ξ is to discriminate if $W_{d,d'}$ contains only measurement error ($d = d'$) or not ($d \neq d'$), hence τ has to be defined such that it is greater than all the $W_{d,d'}$ with $d = d'$ and less than $W_{d,d'}$ with $d \neq d'$. It is easy to figure out that this bi-partition of $W_{d,d'}$ values cannot be easily accomplished, implying a high number of false positives ($\widetilde{S}_{m,d}$ is recognized as a signature generated by another d' device) with a more permissive threshold or, conversely, high number of false negatives ($\widetilde{S}_{m,d}$ is not recognized as a signature provided by the device d itself) with a less permissive threshold. Moreover, to design conservatively, avoiding a high False Rejection Rate (FRR), τ should be fixed taking into account the worst case value, that is the case in which measurement errors on each component are maximum:

$$\tau = \sqrt{\sum_{n=0}^{N-1} \max_{m,d,m',d'} \left(\Delta\varepsilon^2_{n,m,m',d,d'}\right)}.$$

Such defined threshold lead to have a high False Acceptance Rate (FAR), since it properly classifies not only legit signatures generated by same devices, but also signatures generated by different devices, which distance is under the threshold due a partial proximity, i.e. some of the homologous measured frequencies are close each other. On the other hand, the FAR can decrease with a higher number of instantiated ROs (N) such that the probability to generate false positives becomes smaller. The same discussion is still valid for other tests that exploit some kind of distances, but are not reported here for sake of brevity.

B. FSPUF: The Score Test

The threshold selection, illustrated above, is a critical operation, since it affects the FSPUF instances in terms of FAR and FRR at the same time. In particular, this happens because the threshold is applied to a single value given by a metric (e.g. norm or angle between vectors). So, splitting the signatures in L sub-parts enables to compare them with more than one ξ function, and practically with different thresholds for each homologous signature part. Assuming that each comparison gives 0 or 1, as illustrated before in the Equation 10, all the tests establish a score in the range $[0, L]$. A score of L implies a high similarity between $\widetilde{S}_{m,d}$ and $\widetilde{S}_{m',d'}$, and viceversa a score of 0 implies a high dissimilarity. Contrary to the case of only one comparison test, the choice of the threshold τ with L tests is less critical because comparisons results are masked by the score. Picking different thresholds affects a subset of L tests at time; indeed, scores are lower with less permissive thresholds, or higher with more permissive values. The minimum score required to recognize two signatures as generated from the same device can be tuned together with the τ in order to obtain good FAR and FRR values. Hence, with L comparisons between sub-parts of $\widetilde{S}_{m,d}$ and $\widetilde{S}_{m',d'}$, the two signatures can be considered as provided by the same device if at least $l \in [0, L]$ of them succeed. Considering the limit case $L = N$ and a unique threshold for each test, comparisons take into account all homologous components of each vector with the same test.

Formally, let χ be the binary function that compares two real numbers and outputs 0 or 1:

$$\chi : \mathbb{R} \times \mathbb{R} \rightarrow \{0, 1\} \,|\, \chi(a, b) = \begin{cases} 1 & \text{if } |a| \leq b \\ 0 & \text{if } |a| > b \end{cases}$$

Then, ξ can be defined as follows:

$$\xi\left(\widetilde{S}_{m,d}, \widetilde{S}_{m',d'}\right) = \chi\left(l, \sum_{n=0}^{N-1} \chi\left(\Delta\widetilde{f}_{n,d,d'} + \Delta\varepsilon_{n,m,m',d,d'}, \tau\right)\right). \tag{11}$$

The inner χ function takes into consideration just two homologous components of the signatures, giving 1 in case of the distance is under the threshold. Then, this test is iteratively applied to all the components and the outcomes are accumulated. This intermediate result is actually compared to the l value by means of the χ function.

In other words, the Equation 11 defines a ξ that exploits the score obtained by comparing each component of the signature $\widetilde{S}_{m',d'}$ against $\widetilde{S}_{m,d}$ (or conversely $\widetilde{S}_{m,d}$ against $\widetilde{S}_{m',d'}$) with a threshold. Hence, if at least l tests succeed, the signatures are classified as provided by the same device. Furthermore, such ξ function requires only additions and comparisons, suitable to be implemented in hardware without adopting any software module or expensive arithmetic units.

C. Statistical Model of the Score Test

The effectiveness of the score test, given in the Equation 11, relies on the choice of an appropriate threshold τ and minimum score l. Such parameters can be deduced from a statistical model that characterizes the test introduced by ξ over a set of signatures in terms of FAR and FRR.

Fundamentally, the function ξ can me modeled as a Bernoulli process, in which each test performed by $\chi\left(\Delta\widetilde{f}_{n,d,d'} + \Delta\varepsilon_{n,m,m',d,d'}, \tau\right)$ has a value of either 0 or 1 and, for each n, the probability that $\chi = 1$ is p. Each comparison is memoryless, so all evaluations of χ are independent one from each other. This assumption relies on the independence of each measured frequency, hence on the FSPUF design that has to properly retrieve frequencies from ROs. In other words, given that the probability p is known, $n - 1$ χ tests do not provide any additional information about the n-th χ.

As for p, the probability that $\chi = 1$ can be defined as follows:

$$
\begin{aligned}
p &= P\left(\chi\left(\Delta\widetilde{f}_{n,d,d'} + \Delta\varepsilon_{n,m,m',d,d'}, \tau\right) = 1\right)\\
&= P\left(\left|\Delta\widetilde{f}_{n,d,d'} + \Delta\varepsilon_{n,m,m',d,d'}\right| \le \tau\right)\\
&= \begin{cases} P\left(\left|\Delta\widetilde{f}_{n,d,d'} + \Delta\varepsilon_{n,m,m',d,d'}\right| \le \tau\right) = p_{dd'} & d \neq d' \\ P\left(\left|\Delta\varepsilon_{n,m,m',d,d'}\right| \le \tau\right) = p_{dd} & d = d' \end{cases}
\end{aligned}
\tag{12}
$$

The Equations 12 reports two possible cases for p. As indeed, whenever we take into consideration homologous components extracted from the same device, the χ function compares the measurement error contributions to the defined threshold. Conversely, whenever we considers homologous components of two signatures generated by two different devices, the χ function compares the contribution of the inherent frequency values and the measurement error.

As for the FAR and the FRR, they can be estimated by exploiting the Equations 11 and 12. The FAR can be calculated as the probability to have at least l χ tests on N trials that succeed given that $d \neq d'$, hence:

$$
\text{FAR} = \sum_{i=l}^{N} \binom{N}{i} p_{dd'}^{i} \left(1 - p_{dd'}\right)^{(N-i)} \tag{13}
$$

Conversely, the FRR can be expressed by considering the complementary probability to have at least l χ tests on N trials that succeed, given that $d = d'$:

$$
\text{FRR} = 1 - \sum_{i=l}^{N} \binom{N}{i} p_{dd}^{i} \left(1 - p_{dd}\right)^{(N-i)} \tag{14}
$$

It is clear that the involved probabilities p_{dd} and $p_{dd'}$ are strictly coupled with the technological target, the RO design and the architecture used to measure frequencies. Indeed, to be calculated, the Equation 12 requires the statistical distribution of $\widetilde{f}$ and ε. Such distributions can be estimated over either measurement campaigns or by simulating ROs realized on the target manufacturing process.

V. Experimental Validation

In order to prove the effectiveness in a real case scenario, this Section illustrates the set-up for the statistical model, that is the statistical characterization of $\widetilde{f}$, and a configuration for parameters of the FSPUF, so choosing l and τ. To this aim, we adopted the Xilinx Spartan-6 technology, in particular XC6LX16 devices.

A. Model Characterization

We estimated the frequency characterization involving 10 Xilinx XC6LX16 devices, and from each one we extracted frequencies from 938 ROs, sampled 25 times. We measured each frequency sample implementing all designs with only one RO at time allocated in different places. The adopted RO was configured with 5 inverting stages (4 inverters and 1 nand control gate) and hard-placed trough relationally placed macros (RPMs), as well as other involved hardware entities, such as counters needed for extracting frequency value. The clock counter was fixed to 20 bits, while the frequency counter had a length of 24 bits. The system clock reference was set to 100 MHz.

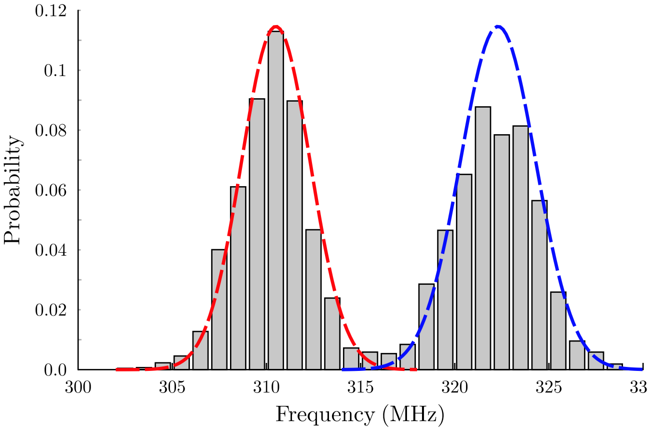


Fig. 4. Distribution of frequencies measured over 10 devices and fitted with bimodal normal distribution.

In Figure 4 we report the distribution of frequencies extracted from all ROs by averaging their values on the 25 samples. As one can notice, the distribution is a bimodal gaussian and higher frequencies are associated to ROs that were placed in configurable logic blocks (CLBs) belonging to even columns, which are equipped with a different basic elements with respect to other CLBs [26].

For the sake of ease, hereafter we consider only ROs placed exclusively on odd or even CLBs columns, such that we can characterized frequencies with a single normal distribution. Moreover, gaussian distributions exhibit a variance that can be assumed to be the same. The random variable ε, being AWGN process, is characterized by $\mathcal{N}\left(0, \sigma_{\varepsilon}^{2}\right)$, while the random variable $\widetilde{f}$, by definition, has an average value equal to 0 and a standard deviation σ_{intra} or σ_{inter}, which are described in Subsection III-C. Hence, the Equation 12 can be specialized as the following:

$$
p = \begin{cases} \displaystyle\int_{-\tau}^{\tau} \frac{1}{\sqrt{2\pi\left(2\sigma_{\text{inter}}^{2} + 2\sigma_{\varepsilon}^{2}\right)}} e^{-\frac{\tau^{2}}{2\left(2\sigma_{\text{inter}}^{2} + 2\sigma_{\varepsilon}^{2}\right)}} & d \neq d' \\ \displaystyle\int_{-\tau}^{\tau} \frac{1}{\sqrt{2\pi\left(2\sigma_{\varepsilon}^{2}\right)}} e^{-\frac{\tau^{2}}{2\left(2\sigma_{\varepsilon}^{2}\right)}} & d = d' \end{cases} \tag{15}
$$

In particular, frequency variables $\widehat{f}$ picked from different devices are distributed as $\mathcal{N}\left(\mu, \sigma_{\text{inter}}^{2}\right)$, hence when $d \neq d'$ the variance associated with $\left(\widetilde{f}_{n,d} + \varepsilon_{n,m,d}\right) - \left(\widetilde{f}_{n,d'} + \varepsilon_{n,m',d'}\right)$ is $\left(2\sigma_{\text{inter}}^{2} + 2\sigma_{\varepsilon}^{2}\right)$.

Each probability can be easily evaluated by means of the primitive erf (x):

$$
\begin{aligned}
p_{\mathcal{N}(0,\sigma^{2})}(\tau) = \frac{1}{2}\left(1 + \text{erf}\left(\frac{\tau}{\sqrt{2\sigma^{2}}}\right)\right)& \\
- \frac{1}{2}\left(1 + \text{erf}\left(\frac{-\tau}{\sqrt{2\sigma^{2}}}\right)\right)&
\end{aligned}
\tag{16}
$$

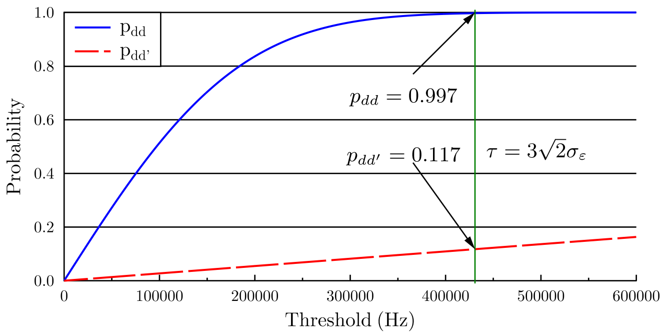


Fig. 5. p_{dd} and $p_{dd'}$ evaluated by varying the threshold τ.

Since the probability p has 2 different forms, which depend on the fact that d and d' could be the same device or not, let $p_{dd'}$ be the value given by Equation 15 in the first case and p_{dd} the value in the second case. Consequently they can be evaluated as:

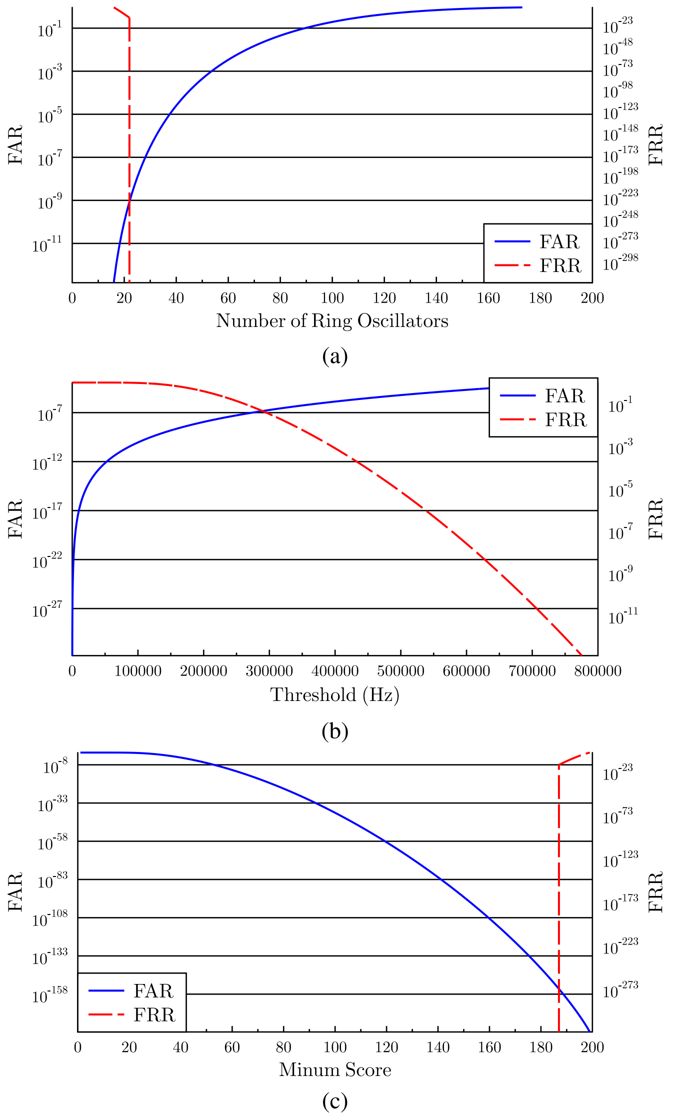
$$p_{dd} = p_{\mathcal{N}(0,2\sigma_\varepsilon^2)}(\tau) \tag{17a}$$

$$p_{dd'} = p_{\mathcal{N}(0,2(\sigma_{\text{inter}}^2+\sigma_\varepsilon^2))}(\tau) \tag{17b}$$

Figure 5 reports graphs for both the probability p_{dd} and $p_{dd'}$ for 5 stages ROs implemented on Spartan-6 technology, varying the variable τ, which is the threshold for the comparison introduced by the single test defined in Equation 11. The p_{dd} curve increases faster than the $p_{dd'}$, since the variance associated with the gaussian distribution for $p_{dd'}$ is greater than the one of p_{dd}, as it contains only the variance associated to the error introduced by measuring operations, therefore this result can be generalized for any implementation and technology.

As we are looking for having p_{dd} close to 1, hence passing the test when two frequencies from the same device are compared, and, at the same time, $p_{dd'}$ close to 0, hence not passing the test when two frequencies come from two different devices, τ should be picked such that $p_{dd} \approx 1$ keeping $p_{dd'}$ still at a very low value near 0. Applying the 3 sigma rule over the p_{dd}, we get a threshold equals to ~430 kHz and a $p_{dd'}$ equals to 0.117. It follows that the effectiveness of FSPUF is directly related to the fact that $\sigma_{\text{inter}} > \sigma_\varepsilon$. The greater is the distance between these two variances, the better FSPUF is.

It is worth remarking that, even if not explicitly reported, the FAR and the FRR depend not only on l and N, but also on τ, σ_ε, σ_{inter}. In particular, the last two parameters are given by the specific implementation of ROs and depend also on the technological target. For the Xilinx Spartan-6, we refer to the values reported in [27]. The Figure 6 illustrates semilogarithmic charts for FAR and FRR varying N, τ and l. In particular, the Figure 6a shows the two rates varying N, with $l = N - 1$ and $\tau = 3\sqrt{2}\sigma_\varepsilon$. The picked value of the threshold guarantees that 99.7% of the components of $W_{d,d'}$ belongs to the interval $[-\tau, \tau]$ when $d = d'$, i.e. when $W_{d,d'}$ contains only the measurement error. The probability to reject a legit signature grows with the number of involved ROs, since it is less probable to have a positive response with a greater number of χ tests, and the probability to accept a non-legit signature decreases, since it is unlikely that $N - 1$ χ tests give a positive response. It is worth noting that with more that 22 ROs, the FRR assumes a null probability. Figure 6b shows FAR and FRR varying on the threshold value, keeping $N = 9$ and $l = 8$. As one can notice, the FAR decreases with more permissive threshold values, as it is unlikely that each χ test gives a positive response with tight ranges, and the FRR increases since the probability to have positive responses from χ tests grows with bigger threshold values. The last one, Figure 6c, shows the impact of the minimum score value l on the rates keeping constant $\tau = 3\sqrt{2}\sigma_\varepsilon$ and $N = 25$. While the FAR decreases on the number of the minimum score, since the probability to succeed a at least l χ tests becomes lower with bigger l, the FRR turns out to be 0 up to $l = 187$.



(a)

(b)

(c)

Fig. 6. FAR and FRR evaluated trough the Equations 14 and 13 varying the number of ROs, the threshold and the minimum score value. (a) FAR and FRR evaluation varying with the number of involved ROs N. (b) FAR and FRR evaluation varying with the threshold τ. (c) FAR and FRR evaluation varying with the minimum score value l.

B. Statistical Model Parameters

Let us resort the statistical model for the FAR and FRR values, previously given in the subsection. It takes into account 5 parameters:

1) σ_ε, the standard deviation associated with the measurement error;

2) σ_{intra}, the standard deviation associated with the frequency distribution;
3) τ, the threshold for the χ tests;
4) N, the number of involved ROs;
5) l, the minimum number of χ tests required by the function ξ to give a positive result;

As for the first two parameters, they depend only on the technological target in which the FSPUF has to work and on the design of both the measurement architecture and ROs. Indeed, σ_{ε} and σ_{inter}, discussed in Section III, are different when the number of stages in the loop changes, as also reported in [27]. Hence, the only way to manipulate them is working with the number of stages, routing strategies, etc. Obviously, the measurement error can be further reduced considering wider counters and slower system clocks, that is longer measurement time.

As for the FPGA technology, the estimation of such quantities is not trivial, since the only way to obtain them is just to run experiments on real devices and retrieve as many frequencies as possible, such that these parameters can be estimated with enough accuracy. It is hard to exactly know how many experiments are enough to obtain a confident approximation for both the parameters. For the ASIC technology, the evaluation of such two parameters can be accomplished by means of simulation campaigns (e.g. exploiting the Monte Carlo method), hence there is no need to have available manufactured ICs.

Consequently, the remaining parameters can be tuned in order to obtain the wanted FAR and FRR. As indeed, the statistical model provided by Equations 14 and 13 can be used only to retrieve an estimation of the rates knowing the 5 previous parameters, but it is not possible to query which parameters can satisfy given values of FAR and FRR. Nonetheless, we know trends of both rates varying one parameter at time, as reported by the Figure 6.

Remarks About t_m: The introduced score test relies on the signature $\widetilde{S}_{m,d}$ reported in the Equation 7. In particular, $\widetilde{S}_{m,d}$ is defined through an approximation because of the average value of the measurement error. But another source of uncertainty is given by the assumption that $t_{n,m} = t_m, \forall n$. Indeed, even if this approximation introduces a small error, the score test illustrated in this paper is able to inherently handle it. Assuming t_n distributed as a gaussian (as demonstrated in [27]), the value $t_{n,m} - \underset{n}{E}\left[t_{n,m}|m\right]$ involved in the computation of $\widetilde{S}_{m,d}$ is characterized by a standard variation that can be estimated trough previous experimental campaign and, for the sake of simplicity, it can be included within the σ_{ε}. Even though, since the contribution of the σ_t is extremely smaller than σ_{ε} (10^1 Hz against 10^5 Hz), the FAR and FRR modestly benefit from such threshold adjustment.

C. Experimental Result

Exploiting the frequencies value collected during the experimental campaigns illustrated in [27], the score test can be evaluated on a significant amount of real data; in particular three data sets can be analyzed:

- normal working conditions: 10 devices, 938 ROs per device, 25 measurements for each RO;
- temperature variations: 1 device, 938 ROs, 9 measurements for each RO, 6 temperature values;
- aging: 1 device, 938 ROs, 20 measurements for each RO, two characterizations (fresh and aged).

TABLE I

FSPUF Evaluated in Terms of FAR and FRR Trough Experimental Campaigns Conducted on the Xilinx Spartan-6 FPGA Family. The Value Are Obtained by Tuning the Minimum Value of the Score

Number	Uncontrolled Conditions		Temperature Variations		Aging	
of ROs	FAR	FRR	FAR	FRR	FAR	FRR
2	0	9.239e-03	3.217e-08	8.152e-02	2.011e-09	7.731e-02
3	2.001e-09	3.703e-02	1.044e-08	1.703e-02	1.233e-08	3.099e-02
4	0	1.268e-03	7.150e-09	3.484e-02	6.255e-10	5.211e-02
5	0	0	0	1.103e-02	0	2.893e-03
6	0	0	0	0	4.924e-08	3.782e-03
7	0	0	5.267e-07	1.137e-04	0	0
8	1.739e-11	6.641e-13	0	2.259e-06	0	0
9	0	0	0	0	2.152e-10	0
10	0	0	0	0	0	2.439e-09
11	0	0	0	2.747e-13	0	0

The empirical results in terms of FAR and FRR are reported in Table I. They are calculated by compounding frequencies extracted from the same device, defining signatures with a variable length. This approach leads to a conservative situation since, as demonstrated in [27], the intra variance is less than inter variance, hence the FSPUF will perform better in a real scenario in which signature are compared among different devices. For each row of the Table I, we chose a l value in order to get FAR and FRR as low as possible. Indeed, l is the minimum score value that establishes if two signatures are produced from the same device or not, as defined by the ξ function in the Equation 11. Such a fine tuning, together with the fact that we are changing at the same time n and l, yields to a Table without a clear pattern. Even though, generally speaking, the non-zero values decrease on the number of n. Worst results are related to the adoption of ROs less than six in number. Indeed, by considering more than six ROs, the FAR and FRR values becomes greater than $\sim 10^{-4}$.

In the case of uncontrolled working conditions, that is without keeping the temperature stable and without considering the aging effect, even with only 2 ROs the FAR is 0 and FRR is equal to 9.239e-03. This is an important result because a classical ROPUF is able to discriminate only 2 device by employing with only 2 ROs. Contrary to this case, the FSPUF is actually discriminating $\frac{10\times938}{2} = 4690$ different devices.

As for the temperature case, the best conditions are with 9 and 10 ROs, while for the aging the best conditions are with 7 and 8 ROs.

D. Implementation of the FSPUF

In order to prove the effectiveness of the FSPUF in providing an authentication mechanism, we designed a hardware component that: (i) extracts frequencies from ROs; (ii) computes the signature (as described by Equation 7) by subtracting the average frequency value to each component; (iii) compares the obtained signature with a reference by exploiting the score test (as illustrated in Equation 11). As for frequency extraction, we implemented the same technique described in [27], that is a single counter design, and a multi-counter design. They differs each other in the area overhead, since the

TABLE II

OVERHEAD OF SOME FSPUF CONFIGURATIONS IMPLEMENTED ON A XILINX SPARTAN-6 FPGA DEVICE

ROs	Single Counter			Multiple Counters			FRR
	Area		Time (ms)	Area		Time (ms)	
2	265 LUTs	179 FFs	12.49	347 LUTs	230 FFs	6.24	$\leq 10^{-238}$
4	301 LUTs	275 FFs	25.16	418 LUTs	362 FFs	6.24	$\leq 10^{-142}$
8	361 LUTs	467 FFs	50.57	554 LUTs	623 FFs	6.25	$\leq 10^{-90}$
16	401 LUTs	685 FFs	100.99	734 LUTs	974 FFs	6.25	1.6210×10^{-13}
32	650 LUTs	1085 FFs	194.53	1297 LUTs	1463 FFs	6.25	1.210×10^{12}

TABLE III

COMPARISON OF ROPUF, CROPUF AND FSPUF BY CONSIDERING A MULTIPLE-COUNTER ARCHITECTURE ON A XILINX SPARTAN-6 FPGA DEVICE

RO	ROPUF				CROPUF				FSPUF			
	LUTs	FFs	FRR	Devices	LUTs	FFs	FRR	Devices	LUTs	FFs	FRR	Devices
2	269	70	0.7	2	279	78	0.2	2	347	230	$\leq 10^{-238}$	4096
4	295	120	0.67	4	304	137	0.14	4	418	362	$\leq 10^{-142}$	$\geq 1.6 \times 10^{7}$
8	349	221	0.86	16	353	252	0.13	16	554	623	$\leq 10^{-90}$	$\geq 2.8 \times 10^{14}$
16	402	415	0.87	256	452	466	0.14	256	734	1463	1.6×10^{-13}	$\geq 7.9 \times 10^{28}$
32	608	809	0.89	65536	640	937	0.19	65536	1297	1463	1.210×10^{12}	$\geq 6.3 \times 10^{57}$
384	5033	9256	0.9	$\geq 6.3 \times 10^{57}$	5089	10802	0.21	$\geq 6.3 \times 10^{57}$	*384 is not a power of 2*			

former requires less resources, but also in time required to measure the frequencies, as the latter does the task in parallel along the ROs.

As for the second step, we implemented only an adder/subtractor since, by employing 2^x ROs, the average operation corresponds to a shifted sum, discarding the decimal bits. As for the last one, it requires a subtractor and a comparator. We sum-up in the Table II the area and time overhead of some configuration on a Xilinx Spartan-6 FPGA. We summarize the differences between the single counter and parallel counter implementation of the FSPUF in Table II and estimated FRR (Equation 14), varying on the number of RO. The time reported in the table refers to the computation of the signature from ROs and the comparison with the score test. Moreover, the time required by the single counter architecture varies linearly on the number of ROs, since they are sequentially measured, while the time required by the multiple counters is almost constant w.r.t. the number of ROs. Of course, the area overhead required by the multiple counters architecture of the FSPUF is not negligible. It is worth noting that the time required to measure a single frequency value from a RO is the major contribution on the time, since the FSPUF operations require a negligible amount of clock cycles. For instance, considering the case of 2 ROs, for both architectures, the time required to compute the signature and to compare it with another one is almost 1%.

In Table III we give a comprehensive comparison among such architectures. It is worth noting that the conventional ROPUF architecture with 32 ROs guarantees the generation of, at most, 16 bit signatures by comparing RO pairs, that is 65.536 different signatures, while the FSPUF is characterized by a higher discriminant power as it takes the whole frequency value for each RO for comparing two signatures. Indeed, the number of actual bits involved into the frequency comparison is given by the chosen threshold. In particular, for the real case illustrated in Section V-A, the threshold is equal to 470 KHz, meant that the measurement error is spread on the ∼18/24 LSB. The remaining 6/24 bits are effectively involved in the frequencies comparison, hence the total number of actual bits equals 192, meant that the FSPUF is potentiality able to discriminate $2^{192} > 10^{57}$ different devices. The last row reports the configuration of both ROPUF and CROPUF for 384 ROs, which guarantees to extract 192 bits and to discriminate more than 10^{57}. The same result is reached by the FSPUF considering only 32 ROs, requiring about 25% less LUTs and 86% less registers compared to the CROPUF. Moreover, experimental data for the FSPUF with 384 ROs are missing due to the fact that 384 is not a power of 2.

W.r.t. a ROPUF, the area overhead required by the FSPUF is represented by the adder/subtractor circuit and the finite state machine that controls the data flow. Nevertheless, conventional ROPUF requires post-processing technique to restore response stability and randomness among different devices, adding a significant area and time overhead [8], even though a Controlled ROPUF is able to significantly reduce the bit-error probability with a negligible hardware overhead [11].

VI. CONCLUSION

One of the most critical factor of embedded system design is the security, since attacks are getting more and more sophisticated and effective. In particular, physical attacks are able to extract sensitive information from a device, including the secret key, whereby cryptography schemes guarantee the system protection. In such a context, the scientific literature introduced PUFs, which represent an advanced solution for guaranteeing security, since they provide unclonability, uniqueness and tamper-evident properties with a sufficiently low overhead. Unfortunately, PUFs suffer from a variability that makes them unreliable and, hence, require costly post-processing techniques that actually prevent their wider adoption in security schemes. Our original contribution was the investigation of a technique to obtain a reliable authentication and identification mechanism based on PUFs without incurring in expensive hardware overhead.

In this paper, we introduced the FSPUF, a novel approach to have available a reliable authentication mechanism by exploiting a RO array. FSPUF, contrary to conventional PUF circuits that uses ROs in a differential way, makes use of the absolute values of ROs frequencies to extract a unique signature, that can be used for identification purposes. We mathematically demonstrated its effectiveness, even in case of different working conditions that alter the ROs frequencies, such as taking into account the temperature variations, and the aging effects. Then, we proved, by means of real implementations on FPGA devices, namely Xilinx Spartan-6, that the overhead required by the FSPUF is very lower than the one introduced by ROPUF and CROPUF, leading to a much greater reliability.

REFERENCES

[1] D. Lim, J. W. Lee, B. Gassend, G. E. Suh, M. van Dijk, and S. Devadas, "Extracting secret keys from integrated circuits," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 13, no. 10, pp. 1200–1205, Oct. 2005.

[2] G. E. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," in *Proc. 44th ACM/IEEE DAC*, Jun. 2007, pp. 9–14.

[3] J. H. Anderson, "A PUF design for secure FPGA-based embedded systems," in *Proc. 15th Asia South Pacific Design Autom. Conf.*, Jan. 2010, pp. 1–6.

[4] T. Addabbo, A. Fort, M. Di Marco, L. Pancioni, and V. Vignoli, "Physically unclonable functions derived from cellular neural networks," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 60, no. 12, pp. 3205–3214, Dec. 2013.

[5] P. A. Layman, S. Chaudhry, J. G. Norman, and J. R. Thomson, "Electronic fingerprinting of semiconductor integrated circuits," U.S. Patent 6 738 294, May 18, 2004.

[6] S. S. Kumar, J. Guajardo, R. Maes, G.-J. Schrijen, and P. Tuyls, "Extended abstract: The butterfly PUF protecting IP on every FPGA," in *Proc. IEEE Int. Workshop Hardw.-Oriented Secur. Trust*, Jun. 2008, pp. 67–70.

[7] E. I. Vatajelu, G. D. Natale, M. Barbareschi, L. Torres, M. Indaco, and P. Prinetto, "STT-MRAM-based PUF architecture exploiting magnetic tunnel junction fabrication-induced variability," *J. Emerg. Technol. Comput. Syst.*, vol. 13, no. 1, pp. 5:1–5:21, May 2016. [Online]. Available: http://doi.acm.org/10.1145/2790302

[8] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," *SIAM J. Comput.*, vol. 38, no. 1, pp. 97–139, 2008.

[9] P. Z. Wieczorek and K. Golofit, "Dual-metastability time-competitive true random number generator," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 61, no. 1, pp. 134–145, Jan. 2014.

[10] Ü. Güler and G. Dündar, "Modeling CMOS ring oscillator performance as a randomness source," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 61, no. 3, pp. 712–724, Mar. 2014.

[11] A. Maiti and P. Schaumont, "Improved ring oscillator PUF: An FPGA-friendly secure primitive," *J. Cryptol.*, vol. 24, no. 2, pp. 375–397, 2011.

[12] P. Sedcole and P. Y. K. Cheung, "Within-die delay variability in 90nm FPGAs and beyond," in *Proc. IEEE Int. Conf. Field Program. Technol.*, Dec. 2006, pp. 97–104.

[13] A. Maiti, J. Casarona, L. Mchale, and P. Schaumont, "A large scale characterization of RO-PUF," in *Proc. IEEE Int. Workshop Hardw.-Oriented Secur. Trust*, Jun. 2010, pp. 94–99.

[14] F. Wilde, M. Hiller, and M. Pehl, "Statistic-based security analysis of ring oscillator PUFs," in *Proc. 14th Int. Symp. Integr. Circuits (ISIC)*, Dec. 2014, pp. 148–151.

[15] C.-E. Yin and G. Qu, "Temperature-aware cooperative ring oscillator PUF," in *Proc. IEEE Int. Workshop Hardw.-Oriented Secur. Trust (HOST)*, Jul. 2009, pp. 36–42.

[16] D. Merli, F. Stumpf, and C. Eckert, "Improving the quality of ring oscillator PUFs on FPGAs," in *Proc. 5th Workshop Embedded Syst. Secur.*, 2010, p. 9.

[17] A. Amouri, F. Bruguier, S. Kiamehr, P. Benoit, L. Torres, and M. Tahoori, "Aging effects in FPGAs: An experimental analysis," in *Proc. 24th Int. Conf. Field Program. Logic Appl. (FPL)*, Sep. 2014, pp. 1–4.

[18] A. Maiti and P. Schaumont, "Improving the quality of a physical unclonable function using configurable ring oscillators," in *Proc. Int. Conf. Field Program. Logic Appl.*, Aug. 2009, pp. 703–707.

[19] S. Devadas and M.-D. Yu, "Recombination of physical unclonable functions," in *Proc. 35th Annu. GOMACTech Conf.*, Reno, NV, USA, Mar. 2010.

[20] G. Kömürcü, A. E. Pusane, and G. Dündar, "A ring oscillator based PUF implementation on FPGA," *IU-J. Elect. Electron. Eng.*, vol. 13, no. 2, pp. 1647–1652, 2013.

[21] C.-E. D. Yin and G. Qu, "LISA: Maximizing RO PUF's secret extraction," in *Proc. IEEE Int. Symp. Hardw.-Oriented Secur. Trust (HOST)*, Jun. 2010, pp. 100–105.

[22] G. Kmre, A. E. Pusane, and G. Dndar, "Dynamic Programming based grouping method for RO-PUFs," in *Proc. 9th Conf. Ph.D. Res. Microelectron. Electron. (PRIME)*, Jun. 2013, pp. 329–332.

[23] 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)*, May 2015, pp. 77–80.

[24] J. Agustin and M. Lopez-Vallejo, "An in-depth analysis of ring oscillators: Exploiting their configurable duty-cycle," *IEEE Trans. Circuits Syst. I: Reg. Papers*, vol. 62, no. 10, pp. 2485–2494, Oct. 2015.

[25] J.-P. Linnartz and P. Tuyls, "New shielding functions to enhance privacy and prevent misuse of biometric templates," in *Audio-and Video-Based Biometric Person Authentication*. Berlin, Germany: Springer, 2003, pp. 393–402.

[26] Xilinx. *Spartan-6 FPGA Configurable Logic Block*, accessed on Jan. 2015. [Online]. Available: http://www.xilinx.com/support/documentation/user_guides/ug384.pdf

[27] M. Barbareschi, G. D. Natale, F. Bruguier, P. Benoit, and L. Torres, "Ring oscillators analysis for security purposes in Spartan-6 FPGAs," *Microprocess. Microsyst.*, vol. 47, pp. 3–10, Nov. 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0141933116300709

Mario Barbareschi received the master's *(cum laude)* degree in computer engineering and the Ph.D. degree in computer and automation engineering from the University of Naples Federico II, in 2012 and 2015, respectively, where he is currently a Post-Doctoral Fellow. His research interests include hardware security and trust, cyber physical security, approximate computing and embedded systems design on the FPGA technology.

Giorgio Di Natale (SM'17) received the Ph.D. degree in computer engineering from the Politecnico di Torino, Italy, in 2003, and the Habilitation Diriger les Recherches from the University of Montpellier II, France, in 2014. He is currently a Researcher with the National Research Center of France, LIRMM Laboratory, Montpellier. He has authored over 100 publications spanning diverse disciplines, including VLSI testing, memory testing, Fault tolerance, reliability, and hardware security and trust. He is the Action Chair of the COST Action IC1204 (TRUDEVICE) on Trustworthy Manufacturing and Utilization of Secure Devices. He is the Chair of the European Group of the TTTC, Golden Core Member of the Computer Society.

Lionel Torres received the master's and the Ph.D. degrees from the University of Montpellier 2 in 1993 and 1996. From 1996 to 1997, he was with ATMEL as IP Core Methodology Research and Development Engineer. From 1997 to 2000, he was Assistant Professor with the University of Montpellier 2 and LIRMM Laboratory. Since 2004, he is currently a Full Professor and was the Head with the Microelectronic Department, LIRMM from 2007 to 2010. He is currently the Deputy Head with the University of Montpellier in charge of research and industrial relationship. He has co-authored over 30 journal papers, 150 conference publications, and eight patents. His research interests and skills concern reconfigurable computing and system level architecture, with a specific focus in the security, reliability, and nano-design. He leads several European, National, and Industrial Projects in this field. He is involved in different major conferences, such as DATE, VLSI, FPL, ISVLSI, and DAC.

Antonino Mazzeo is currently a Full Professor with the University of Naples Federico II, Italy, where he is a teacher in computer architecture. He led major Research Programs in conjunction with international Universities, Research Agencies (CNR, ASI, and EC), and the technology leading industries in Italy and abroad. He has wide experience in the field of complex systems modeling, embedded systems, general and special purpose parallel architectures, and security and performance evaluation.