IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 27, 2021, accepted September 19, 2021, date of publication October 4, 2021, date of current version October 12, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3117240

Enhancing the Performance of Lightweight Configurable PUF for Robust IoT Hardware-Assisted Security

FATHI AMSAAD[1], (Senior Member, IEEE), AHMED OUN[2], (Graduate Student Member, IEEE), MOHAMMED Y. NIAMAT[2], (Life Member, IEEE), ABDUL RAZAQUE[3], SELCUK KOSE[4], (Senior Member, IEEE), MOHAMED MAHMOUD[5], (Senior Member, IEEE), WALEED ALASMARY[6], (Senior Member, IEEE), AND FAWAZ ALSOLAMI[7], (Member, IEEE)

[1]School of Information Security, Eastern Michigan University (EMU), Ypsilanti, MI 48197, USA
[2]Department of Electrical Engineering and Computer Science, The University of Toledo, Toledo, OH 43606, USA
[3]Department of Electrical Engineering and Computer Science, International IT University, 050000 Almaty, Kazakhstan
[4]Department of Electrical and Computer Engineering, University of Rochester, Rochester, NY 14627, USA
[5]Department of Electrical and Computer Engineering, Tennessee Technological University (TTU), Cookeville, TN 38505, USA
[6]Department of Computer Engineering, Umm Al-Qura University, Makkah 21421, Saudi Arabia
[7]Department of Computer Science, King Abdulaziz University, Jeddah 21589, Saudi Arabia

Corresponding author: Fathi Amsaad (fathi.amsaad@emich.edu)

The Deanship of Scientific Research (DSR) at King Abdulaziz University, Jeddah, funded this project under grant no. (KEP-18-611-42).

ABSTRACT Lightweight physical unclonable functions (LPUFs) exploit manufacturing process variations of semiconductor integrated circuits (ICs) to protect IoT-based electronic and smart devices from new cyberattacks. This paper proposes two novel security techniques to enhance the robustness of LPUFs using configurable-based ring oscillator PUFs (CF-ROPUFs). These techniques are the intra-die frequency aware (IFA) approach to improve PUF reliability and the logarithmic gamma function (Ln_γ) technique to enhance PUF randomness. The lightweight CF-ROPUF design is realized on hardware, and data samples are collected under varying temperatures and supply voltages over a population of 30 Spartan-3E FPGAs. Experimental results of the IFA technique in terms of average Hamming Weight (HM) demonstrate that the percentage of the reliable RO sample frequencies PUF output is 98.5%. For the analysis, PUF reliability is evaluated in terms of accuracy, repeatability, and reproducibility, the international organization for Standardization (ISO) standards. The results indicate that the RO samples are accurately measured from the CF-ROPUFs mapped in all the chips. After using the proposed 1-out-of-r coding algorithm, the results demonstrate high average repeatability of 98.2% and a magnified average reproducibility of 99.63%. It is also shown that our CF-ROPUF design is immune from accelerated aging impacts reliability issues. Statistical results show that Ln(Ln_γ) enhances the normality and mitigate the negative impacts of the systematic process variations on RO sample frequencies. Randomness results show that CF-ROPUF binary response bits can successfully pass the 15 NIST test suites for true randomness with an enhanced percentage, 93.3%, with the application of the 1-out-of-r coding.

INDEX TERMS Lightweight hardware-assisted security, trusted Internet of Thing (IoT) consumer electronic devices, configurable ROPUF, PUF reliability, PUF aging, ISO standards.

I. INTRODUCTION

The internet of things (IoT) devices have emerged for many applications such as edge computing, intelligent and connected cities, implantable and medical devices, smart power grid, and intelligent autonomous transportation/aerospace systems [1]–[5]. These applications are increasingly integrated into insecure physical environments and need to be protected from the new cyber and physical system attacks. The life-cycle of IoT-based electronic devices is

The associate editor coordinating the review of this manuscript and approving it for publication was Tyson Brooks.

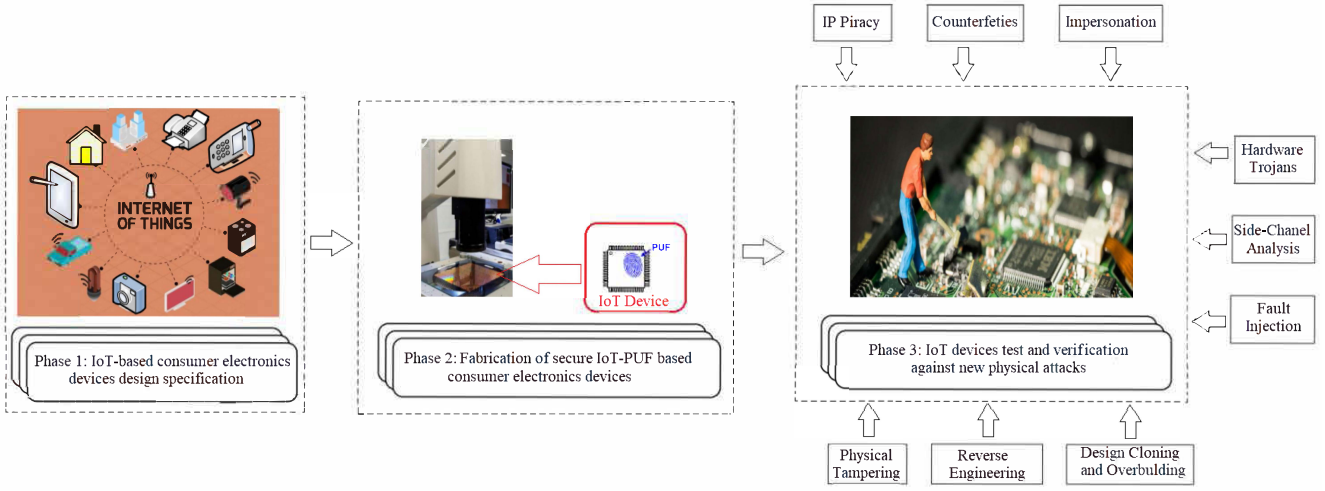


FIGURE 1. Life-cycle of PUF-based IoT-based and smart consumer electronic devices.

illustrated in Fig. 1. The figure shows design specifications, fabrication, and test and deployment of IoT-based consumer electronic devices. As shown in the figure, the life cycle involves multiple parties and facilities, and thus, diverse security threats affect these devices. Further, the major security threats associated with IoT-based and intelligent electronic devices are also shown in Fig. 1. These emerged attacks have seriously threatened the security, operability, and reliability of IoT-based consumer electronic devices and their wide applications [2], [3]. For instance, the increasing outsourcing demands of IC fabrication led to the emergence of hardware/IP-based design flow imposed by an untrusted foundry. As a consequence, cyber physical system attack to steal a design IP, IP piracy, may be launched by an adversary during the device manufacturing process at an untrusted house or foundry. Also, an experienced adversary may mount a hardware Trojan, perform side-channel attacks, physical tampering, or reverse engineering to clone the design or cause malfunction during operation. These attacks may lead to disastrous consequences in critical operations or leaking secret information from an IoT-based and intelligent electronic device.

Physical unclonable functions (PUFs) are new low-cost hardware security primitive with a comparatively simple design ideal for smart and energy-constrained IoT devices. Recent research has proposed to use PUF for low-cost IoT authentications and encryption schemes [4]–[6]. However, the secret keys generated by the proposed PUF may suffer from reliability issues that negatively impact the security and trust in IoT applications. For that, the IoT and smart devices will need to rely on finding robust security solutions (reliable and cost-efficient) for the IoT nodes to securely communicate over the internet and mitigate a wide range of new security attacks. For that, the reliability shortcoming of lightweight PUF needs to be addressed for trusted and enhanced security of resources-constrained IoT devices. Motivated by this, in this paper, we propose robust, secure and trusted, lightweight Configurable ROPUFs (CF-ROPUFs) that utilize dedicated FPGA resources more efficiently for area-critical designs [1], [7].

The proposed CF-ROPUF enhances PUF entropy, allowing a small ROPUF design to generate a large number of challenge-response pairs for secure and trusted IoT applications that require robust and lightweight secret and cryptographic key generation [1], [8], [9], [15], [16]. For the application of the technique, the CF-ROPUF design is implemented on 30 Xilinx Spartan-3E FPGA chips. Our lightweight PUF design is a low-cost and efficient hardware security design intended to generate low-cost secret keys for IoT security, including IoT encryption and robust authentication. A large set of RO frequency measurements are collected under varying temperature and supply voltage values for the reliability analysis. The CF-ROPUF reliability is evaluated based on the standards of both International Organization for Standardization (ISO) [1], [10]–[13] and the National Institute of Standards and Technology [1], [14], [17].

The main contributions in this paper can be summarized as follows:

- A novel approach, intra-die frequency aware (IFA), is proposed to ensure a reliable lightweight CF-ROPUF operation for secure and trusted IoT applications.
- A new security technique based on logarithmic gamma fornication, Ln_γ, is proposed to enhance the normality and mitigate the impact of the systematic variations to enhance ROPUF randomness.
- Inspired by 1-out-of-K [1], [19], the 1-out-of-r coding algorithm is proposed to enhance PUF reliability at varying temperatures and supply voltages based on ISO standards, as well as PUF randomness based on the 15 NIST test suites for true randomness.

The rest of this paper is organized as follows. The related works are provided in Section II. The proposed techniques are

detailed in Section III. The experimental setup and detailed implementation are explained in Section IV. The analysis and discussion of the experimental results are presented in Section V. Finally, conclusions are drawn in Section VI.

II. RELATED WORK

A. SILICONE PUFs

Silicon physical unclonable functions (sPUFs) are one-way physical functions that exploit manufacturing process variation parameters of semiconductor integrated circuits (ICs) to relate an input challenge (c_i) to an output response (r_i) for device authentication and secret key generation [1], [19]–[23]. Different types of sPUFs designs have been proposed as lightweight hardware-based security solutions for IoT applications, detection of physical attacks, i.e. physical tampering and reverse engineering, and hardware Trojan attacks [1]–[6], [15], [24]–[29]. Further, sPUFs have recently emerged as promising hardware-based security primitive to protect consumer electronic devices against invasive and non-invasive attacks. Such attacks include counterfeit parts in electronics consumer manufacturing, physical tampering and reverse-engineering attacks used by an adversary for design cloning and/or overbidding, detection of IC devices fault-injection attacks (attacks on SRAM and EEPROM based devices), detection of malicious circuitry like hardware Trojans [1]–[6], [15], [24]–[29].

The concept of using silicon SRAM PUF as one of the state of the art topics in hardware security primitives is interesting for many reasons. First, SRAM PUFs are simple to realize on hardware and useful for resource-constrained applications, including IoT devices [1], [30]–[33]. SRAM PUF do not need hardware overhead as they use SRAM memory cells that can be integrated in the IoT memory design. The manufacturing imperfections of the CMOS transistors impose random process variations delay or random mismatch of SRAM memory cells. As the SRAM is powered up, process variations influence the power-up state of the associated CMOS transistors. When powered-up, SRAM PUF stores unique binary secret keys influenced by the manufacturing process variation mismatches associated with their CMOS transistors. Some SRAM cells can have a reliable power-up state set to a '1' or '0' state. The research shows that some SRAM cells can be impacted by transistor noise (noisy SRAM PUF) with a non-reliable power-up state. In this case the SRAM cells are considered neutral with unreliable power-up state, that are not suitable for reliable secret key generation [30]. SRAM-PUF based authentication scheme is proposed for resource-constrained IoT Devices [31]. The proposed scheme applies re-ordered SRAM memory addresses (challenges) to extract the corresponding responses from the SRAM cells' startup binary output ('0' or '1'). The results illustrate that the scheme can be used to uniquely authenticate resources-constrained IoT devices with a low computation overhead and small memory capacity [31]. The reliability factor determines how often a PUF can generate the same response to a given challenge. Therefore, to obtain the same response at a specific operating condition, PUFs should only utilize SRAM cells with strongly reliable power-up states to a '0' or '1'. Also, a reliable SRAM PUF needs to generate the same response to a given challenge at all operating conditions, e.g. temperatures and voltage variations.

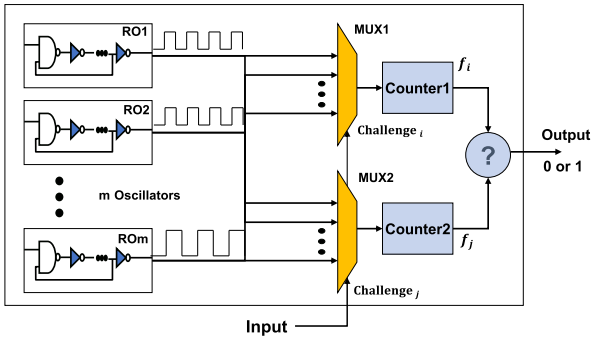


FIGURE 2. Working principle of a ring oscillator PUF circuitry.

B. RING OSCILLATOR PUFs

Ring oscillator (RO) PUF implementations are widely used in many secure applications due to the simple design requirement and high-performance [1], [19], [23]. The working principle of a ROPUF is illustrated in Fig. 2 [1], [19]. Owing to its high performance and simple design requirements, ROPUF can be easily realized on both field programmable gate arrays (FPGAs) and application specific integrated circuits (ASICs) to protect these devices from such type of attacks [1], [7]–[9], [19]–[23], [29], [34], [36], [38]. A ROPUF structure is implemented on silicon chips, i.e. FPGA or ASICs, but the behavior of its challenge-response pairs (CRPs) are difficult to be cloned. Mathematically speaking, a ROPUF is a one-way probabilistic function that relies on random process variations in mapping m challenge bits to n response bits. The response bit r_i is determined by the manufacturing variations during fabrication that lead to slight changes in the frequency of each RO. Based on the input challenges (challenge$_i$ and challenge$_j$), two ROs are selected by the two multiplexers (Mux1 and Mux2) to compare their relative frequencies (f_i and f_j) by counting the number of clock cycles for a specific duration. A one-bit PUF response r_i is generated based on the frequency comparison.

$$r_i = \begin{cases} 1, & if\ f_i \geq f_j \\ 0, & otherwise \end{cases} \tag{1}$$

There are several algorithms proposed to evaluate the security of the ROPUF response [14], [17], [19], [23], [38]–[41]. For example, fFor n ROs, the chain-like neighbor coding algorithm is used to form $(n-1)$ unique RO frequency pairs as

$$F_i = \{(f_0, f_1), (f_1, f_2), \ldots, (f_{n-1}, f_n)\} \tag{2}$$

$$r_i = \{r_0, r_1, \ldots, r_{n-1}\} \tag{3}$$

A total of $r = (n-1)$ response bits, one response bit '0' or '1' for each pair, are obtained using the chain-like

neighbor coding algorithm. The chain-like neighbor coding algorithm has a correlated frequency pair issue where one frequency is used in two pairs. This issue causes a correlation in the generated PUF-based secret key, making it vulnerable to potential cyberattacks aiming to predict the PUF output. To remove such correlation in the generated PUF response bits, an improved version of the chain-like neighbor coding algorithm, known as the decoupled chain-like neighbor coding algorithm, is proposed [1], [8]. This algorithm breaks the RO frequency chain by allowing each RO frequency to appear one time in one frequency pair to mitigate the dependency in RO frequency pairs. This will degrade the number of generated frequency pairs as well as the generated PUF response bits by half, or $r = (\frac{n-1}{2})$ as

$$r_i = \{r_0, r_1, \ldots, r_{(\frac{n-1}{2})}\} \tag{4}$$

Configurable ROPUFs have been initially proposed to improve PUF reliability and overcome area inefficiency of the 1-out-of-k scheme [1], [7], [19]. An improved version of a configurable PUF that generates larger secret keys within the same area has been proposed in [1], [8], [9]. The 1-out-of-k masking coding scheme with $k = 8$ is proposed to improve the reliability of the simple and configurable ROPUF design [1], [7]–[9], [19]. In this scheme, n RO frequency pairs are generated from a total of n ROs implemented on a silicon chip. The 1-out-of-k scheme divides the implemented n ROs into m groups, each with a size of $k = 8$ ROs. Two groups (i, j) are used to obtain k frequency pairs. For the obtained k frequency pairs, only the pair (f_i, f_j) with the maximum frequency difference is selected, where f_i and f_j represent two frequencies generated by group i and group j, respectively. The disadvantage of this technique lies in the chip area inefficiency since k times area is used when implemented on real hardware. In addition to the area overhead, this scheme requires the generation of response bits that are k times longer than the desired secret key length, adding a considerable computational overhead to PUF output generation time.

C. MODELING ATTACKS AGAINST SILICON PUFs

Modeling attacks are an example for the emerging cyberattacks that aim to replicate the behavior of PUF's challenge and response for cloning the secret keys generated by a PUF design [42]. According to recent research, modeling attacks against lightweight PUF-based schemes for IoT authentication and security are classified into three main categories: (a) ML software-based attacks, (b) side-channel hardware-based attacks at the hardware level, and (c) hybrid ML and side-channel attacks [42]. Even though ML attacks are considered one of the most successful software-based attacks to clone the behavior of PUF design, the efficiency and predictability of these ML algorithms decrease with the increase of the complexity of the PUF design, i.e., strong PUF with a large number of nonlinear logical component used to build the PUF structure [43], [44]. Therefore, the time needed to model or clone the PUF behavior will not be feasible.

Side-channel attacks are hardware-based attacks that exploit different side-channel parameters like current leakage, voltage variations, and power and time consumption, electromagnetic fields to launch an attack against semiconductor integrated circuit (IC) devices. Typical side channel hardware attacks include power analysis side channel [45], time consumption side channel [46], electromagnetic side challenge [47], differential fault analysis side channel and photonic emission side channel analysis [48], [49]. Even though side-channel analysis attacks take advantage of the side challenge parameters to model a robust PUF design, sometimes it is difficult for the attacks to obtain a high accuracy PUF attack model [50]. To enhance the modeling time of ML attacks and improve the accuracy of side-channel hardware attacks, recent research proposes to use hybrid (software-based/hardware-based) cyberattacks that apply the side channel parameters as inputs to ML algorithms for enhancing both the PUF modeling time and accuracy [51].

Recent research proposes a novel voltage over-scaling (VOS) as a lightweight PUF-based authentication technique for resource-constrained IoT applications [15], [54]. The proposed approach employs adders to extract the manufacturing process variation to create a two-factor authentication protocol. The paper also presents new machine learning (ML) attacks to hack the proposed authentication protocol. The proposed approach obfuscates the PUF challenge for the VOS-based authentication technique to resist ML attacks. This paper is the extension of two interesting conference papers proposed by the author [53]. The paper deliver an excellent obfuscation ability for strong PUFs. Experimental results demonstrate that different ML algorithms can successfully clone the VOS-based authentication with up to 99.65% accuracy. The results also show that the prediction accuracy is less than 51.2% after deploying the proposed ML resilient technique. The overall conclusion is that the VOLtA is vulnerable to ML modeling attacks.

The same article also evaluates the reliability of voltage over-scaling-based lightweight authentication (VOLtA), a two-factor authentication protocol based on lightweight PUF design [53]. The results show an intra-Hamming distance (intra HD) value of approximately 0.47%, when the temperature decreases from $25°C$, the room temperature, to $23°C$. This value (0.47%) indicates that the reliability of the response bits is decreased from 100%, the optimal value, to about 99.53%. The results also show that the intra HD value is approximately 0.62% when the temperature increases from $25°C$ to $27°C$. This value (0.62%) indicates that the PUF response bits' reliability also decreases from 100% to 99.38%. By computing the average value between 99.53% and 99.38%, we can see that the average reliability is about 99.46%, which is an excellent value. As shown in Table 4, our proposed Cf-ROPUF has an average reliability value of 99.63%, which is slightly better than the average reliability of the PUF used in VoLtA [52], [53].

III. PROPOSED SECURITY TECHNIQUES

This section discusses the two primary proposed security techniques to enhance the robustness of lightweight CF-ROPUF, the Intra-die frequency aware (IFA) reliability technique and the logarithmic gamma (Ln_γ) randomness technique.

A. INTRA-DIE FREQUENCY AWARE (IFA) RELIABILITY TECHNIQUE

Integrating reliability in IoT lightweight PUF designs plays a critical role in ensuring robust trusted and secure operation for IoT-based and smart consumer electronic devices. The intra-die frequency aware (IFA) reliability approach basically states a PUF design exhibits reliability if and only if its output (RO frequencies) can be regenerated when applying the same input challenge for n times. A flowchart explaining the calculations of the Intra-die frequency aware (IFA) value is depicted in Fig. 3. As seen in the figure, after parameter initialization, the frequency value, $f_{i,j,k}$, of the (k^{th}) RO that is mapped on the (j^{th}) FPGA region of the (i^{th}) FPGA is calculated as follows:

$$f_{i,j,k} = \frac{(CC_{i,j,k} \times ref_{clock})}{CC_{ref}} \tag{5}$$

Equation 1 is essential to estimate the value of frequency generated by each ring oscillator. The $CC_{i,j,k}$ represents the number of clock cycles of the activate ROs. To avoid the negative impact of self-heating on the frequency of the active RO, the on-chip heating imposed by neighbor ROs, a deactivation period of 0.1 ms is allowed before a new RO is activated. Also, each RO is only activated for a short period of 0.1 *ms*. The ref_{clock} is the default generated clock by Spartan-3E FPGA family (reference clock of 50 MHz), CC_{ref} is the number of clock cycles used by the reference clock during the activation period of the RO (0.1 *ms*) which equal to 10,000 clock cycles of the 50 MHz FPGA clock (reference clock). For all the ROs mapped on the FPGA chip, each RO has a fixed average delay (d_{RO}). Each RO also has a process variation (d_{PV}) component that is supposed to be constant (neglecting the impact of aging and other temporal on the RO process variation). Note that the average RO delay and process variation delay are both fixed and unique to each RO. Each Ro also has another delay component (d_{NOISE}) due to the local noise factor. This is a dynamic delay and can change over time.

Assuming that $f_{i,j,k}$ and $f'_{i,j,k}$ are two RO frequencies generated by the same RO, the deviation (i.e., frequency difference) $D = f_{i,j,k} - f'_{i,j,k}$ should be theoretically zero when obtaining the frequency again from the same RO. However, in a real hardware implementation, due to RO local loop noise variations (d_{NOISE}), these frequencies ($f_{i,j,k}$ and $f'_{i,j,k}$) differ slightly, and thus, the D value is equal to or less than a finite value x such as

$$|D = f_{i,j,k} - f'_{i,j,k}| \leq x \tag{6}$$

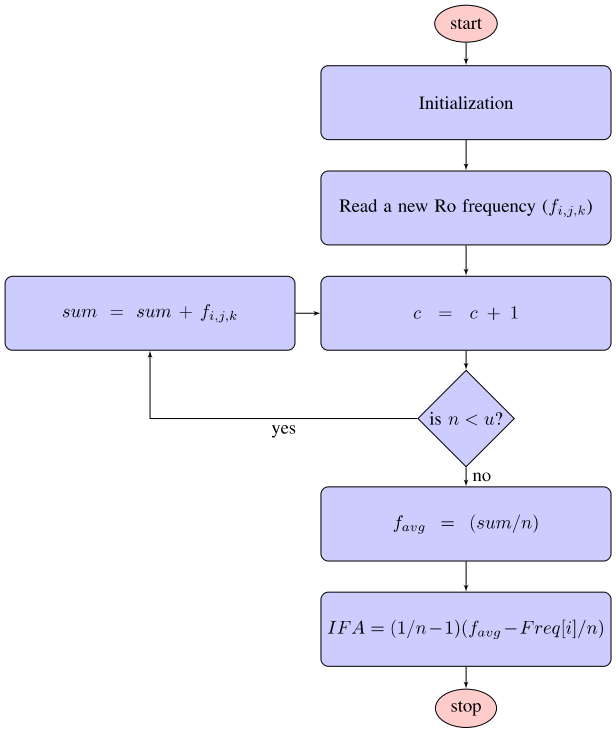


FIGURE 3. The IFA reliability technique flowchart.

The optimum value of x is the smallest value that can be tolerated without degrading ROPUF repeatability in terms of the ability of each RO to regenerate the same sample of RO frequencies for n times (the closer repeatability value to 100%). After empirically trying many values (ranging from 0 to 2 MHZ), we found that the bigger the value of x, the closer the repeatability to 100%, which is the desired reliability. Decreasing the value of x results in degraded reliability value, i.e., the repeatability starts to degrade from 100%. After extensive experiments with different empirical values, we found that intra-die frequency ($IFA_{i,j,k}$ in MHZ) is the smallest possible numeric value that achieves the highest possible repeatability. The IFA value is calculated based on the average RO frequency F_{avg} of the n generate frequencies as follows:

$$F_{avg} = \frac{1}{n}\sum_{i=1}^{n} f_{i,j,k} \tag{7}$$

$$IFA = \frac{1}{n-1}\sum_{i=1}^{n}(F_{avg} - f_{i,j,k}) \tag{8}$$

Based on the calculated IFA value, the below steps are then followed:

1) The same input challenge (C) is applied for n times to the same RO to generate n RO sample frequencies. The IFA values (n values) of the generated RO frequencies are then calculated using the above equations of IFA.

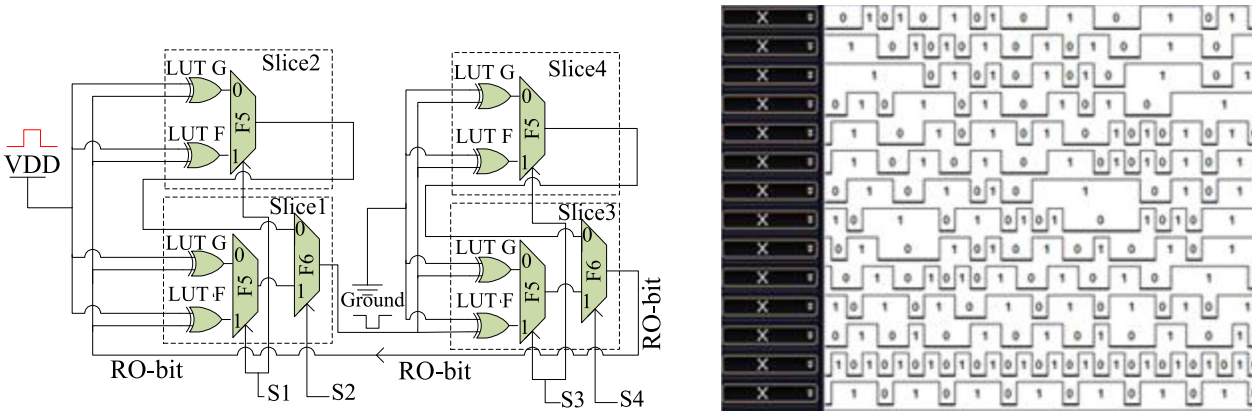


FIGURE 4. 4(a) Implementation of CF-ROPUF on single CLB; 4(b) Measurement of RO frequencies in a single CLB.

2) IFA PUF response bits (r) are generated as

$$r = \begin{cases} 1, & if\ (f_{i,j,k} - f'_{i,j,k}) \leq IFA \\ 0, & if\ (f_{i,j,k} - f'_{i,j,k}) > IFA \end{cases} \tag{9}$$

3) The length of the generated response bits ($_m$) can be calculated using the following equation:

$$m = C_r^n = \frac{n!}{r!(n-r)!}. \tag{10}$$

4) The percentage of the reliable frequencies generated by one RO, where HW is the Hamming Weight of r, is calculated as

$$HW = \sum_{i=1}^{m} \frac{r_i}{m} \times 100\% \tag{11}$$

5) The average Hamming weight is then used to calculate the percentage of the reliable frequencies for the individual ROs (k ROs) mapped in one FPGA area as follows:

$$Avg\_HW = \sum_{j=1}^{t} \frac{HW_j}{k} \times 100\% \tag{12}$$

The IFA threshold value is calculated offline based on the average value of 10 RO frequencies, after the frequencies are collected using the logic analyzer, as shown in equations 3 and 4. The generated frequency values are stored in excel sheets (.CVS) by the logic analyzer. We then use the Math lab for implementing the IFA algorithm and calculating the results. The RO frequencies are first collected using the analyzed and manipulated offline. For that, the computational overhead latency is irrelevant because this has been done offline. Rather, we aim to focus on the reliability and performance aspect of the proposed design. As a future research, we will consider hardware implementation, vulnerability to cyberattacks, on-chip key storage and online calculation of the latency.

B. THE LOGARITHMIC GAMMA (Ln_γ) RANDOMNESS TECHNIQUE

Data normality is an important measure for data homogeneity, uniformity, and randomness [18]. According to the central limit and normal Gaussian theorems, a normal distribution with sufficiently large data samples also represents random data samples [17], [18]. The Ln_γ technique is proposed to improve PUF normality and mitigate the systematic process variation effects on RO frequencies. This enhances the randomness of the generated PUF binary response bits. Data samples are distributed normally by implementing log and/or square root transformation [17], [18]. Mathematically, gamma function of a positive integer n represents the factorial function of $n-1$ as follows:

$$Ln_\gamma(n) = (n-1)! \tag{13}$$

The main steps of the proposed Ln_γ technique are as follows:

1) For each FPGA chip, one average frequency is calculated using $FPGA_{(avg)}$ average values for the r reliable RO frequency mapped on p FPGA regions (six regions) as follows:

$$FPGA_{(avg)} = \frac{1}{p} \sum_{i=1}^{p} \sum_{j=1}^{r} F_{avg(j)} \tag{14}$$

2) As a result, the mean frequency value for all FPGAs is calculated as follows:

$$mean = \frac{1}{n} \sum_{i=1}^{n} (FPGA_{(avg)}) \tag{15}$$

3) This results in total averages of 640 RO sample frequencies generated by all the FPGAs using all six regions (Total average values).
Since data measurements are collected under varying operating conditions, a total of 640 frequencies are similarly generated from the FPGA for the other conditions including the temperatures and supply voltages.

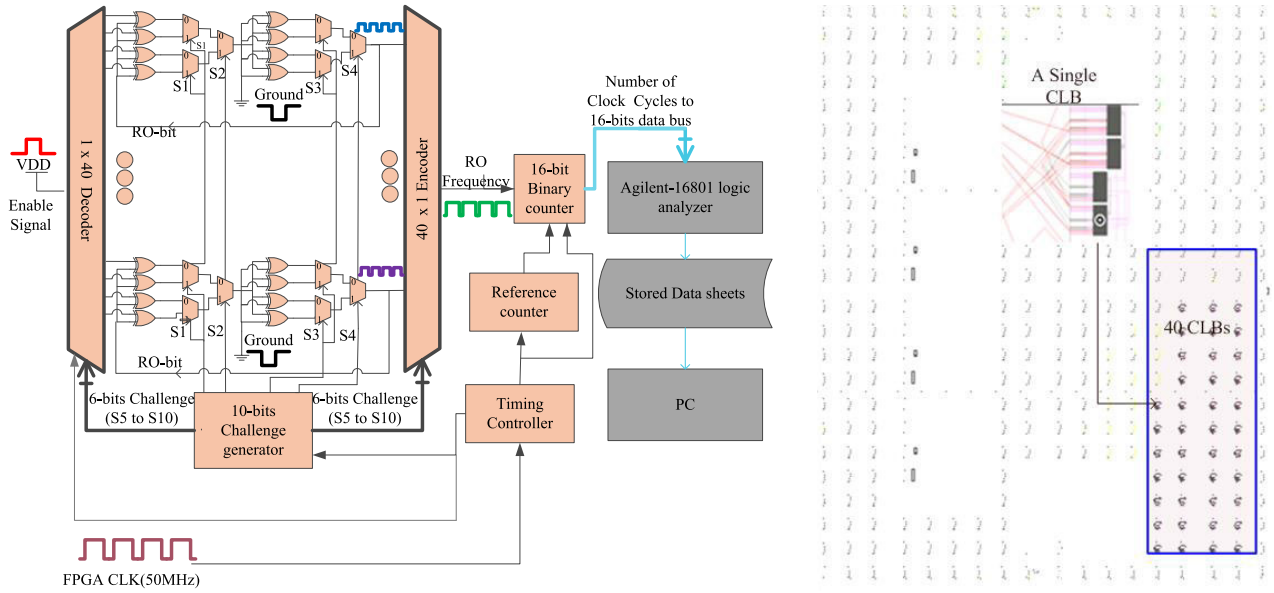


FIGURE 5. 5(a) Proposed CF-ROPUF design for an FPGA region; 5(b) Mapping CF-ROPUF design on an FPGA region.

4) The random deviations values (D_i), that represent the deviations of the $FPGA_{(avg)}$ frequencies from their mean values at a certain operating condition, is used to randomize the normalized frequencies as follows:

$$D_i = mean - FPGA_{(avg)} \tag{16}$$

5) The final average RO frequencies are normalized as

$$NZ = \sqrt{Ln\_\gamma\_(mean)} \times D\_i) \tag{17}$$

IV. EXPERIMENTAL SETUP AND IMPLEMENTATION

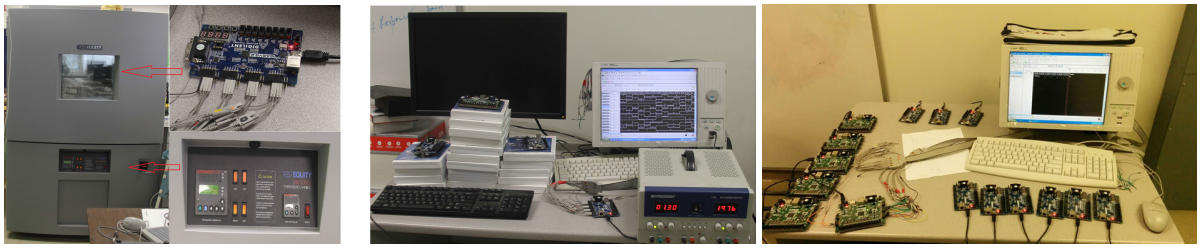
Spartan-3 100E FPGA contains 120 CLBs distributed among 22 rows and 16 columns. The CF-ROPUF design is implemented inside a single CLB of Spartan-3E FPGA using a hard-macro design, as shown in Fig. 4(a). An instance of 16 sample frequencies measured from one CLB with the help of the Agilent Logic Analyzer is shown in Fig. 4(b). These frequencies are calculated from 16 ROs where each RO is selected using dedicated multiplexers (F5 and F6) and activated for a short period (*i.e.,* 0.1ms each). For the implementation of the CF-ROPUF on the FPGAs, each FPGA is divided into six equal regions (three top FPGA regions and three bottom FPGA regions) with 40 CLBs each. A gate-level design of the PUF for an FPGA region (40 CLBs) is shown in Fig. 5(a). An instance of this PUF design mapped at the bottom right 40 CLBs as demonstrated in Fig. 5(b).

Data samples (RO frequencies) are collected from all FPGA regions at five different temperatures, 0°C, 25°C, 50°C, 75°C and room temperature (RT), which is around 20°C. The environmental temperature is managed using a control panel embedded in the chamber, as shown in Fig. 6(a). For accurate measurements, multiple FPGAs are physically placed inside the chamber. The temperature environmental is adjusted according to the desired value, and FPGAs are connected to the logic analyzer through a 16-bit data bus for PUF data measurements. Similarly, RO frequencies are also measured at five different supply voltage levels (VCCINT), 1.0V, 1.1V, 1.2V (nominal), 1.3V, and 1.4V for the FPGAs with the help of a DC power supply and logic analyzer, as shown in Fig. 6(b). To study the impact of aging on CF-ROPUF, RO sample frequencies are measured daily over 30 days, where each RO is activated for 0.1 ms every 10 ms, as shown in Fig. 6(c).

The PUF response bits are generated once every day and compared to the previous responses to compute the percentage of the bit flips that may occur due to accelerated aging, negatively affecting the PUF reliability. The proposed PUF is an improvement of weak ROPUF that enlarges the secret keys by 16-times as compared to a simple weak ROPUF. Our CF-ROPUF design efficiently utilizes the same area (one CLB) of Spartan 3E FPGA to generate a 16-time larger response (more robust response) bit as compared to simple ROPUF and 1-out-of 8 techniques. For the sake of further comparison, the neighbor coding algorithm is used to generate PUF response (256-bits) from our CF-ROPUF and two state-of-the-art. For that, we can see that our design uses the 40 CLBs while the other two dings will need to occupy 256 CLBs of the same FPGA family, Spartan 3E FPGA.

V. EXPERIMENTAL RESULTS ANALYSIS AND DISCUSSIONS

This section discusses the obtained experimental results of PUF performance and quality metrics in terms of hardware overhead, uniqueness, reliability, and randomness using the



(a) (b) (c)

FIGURE 6. Experimental setup for the collection of RO sample frequencies from the tested FPGAs: 6(a) Under varying environmental temperature; 6(b) Under varying supply voltage; 6(c) The impact of accelerated aging on c-ROUF.

above-proposed security techniques and based on the ISO and NIST standards definitions.

A. HARDWARE OVERHEAD

Table 1 shows the hardware overhead comparison between the proposed design and the competing designs. As seen in the table, the XCOR PUF is implemented on Xilinx Spartan-6 FPGAs [52]. The design requires 8 LUTs (7 XOR gates and one AND gate) and is mapped inside a single CLB of each Xilinx Spartan FPGA.

A low cost configurable Ring Oscillator (CRO) PUF design is evaluated on six Xilinx Spartan-6 FPGAs. The design requires 4 CLBs where each configurable logic block (CLB) can implement two delay units [53]. Hard macros are used to ensure that all CROs are identically routed. The design map eight different configurations using 4 CLBs [53]. The design is implemented using 10 Xilinx Virtex II XC2VP30 boards.

A 3 × 3 (9 Tristate inverter) matrix is implemented on each FPGA [54]. A total of 5 CLBs are needed to implement the matrix, where each CLB only implements two tristate inverters. The hard macro procedure is used to map the matrix on each FPGA identically. In this design, two delay paths of an identically mapped tristate inverter matrix are needed for generating one response bit with a hardware overhead of 10 CLBs. The crossover RO consists of m inverters with a particular frequency. The frequency of the ROs is generated using Xilinx Spartan3 FPGA boards, with 46 rows × 34 columns and 1,164 CLBs. The cross ROPUF consists of m levels of inverters; the outputs of an inverter level are fed as the inputs to the next inverter level after passing through an interstage crossing logic. The interstage crossing logic determines the routing path of step signals inputted without any additional logical operation. There are m-1 interstage crossings to change the configuration of the delay loop with selection inputs. Each interstate is implemented using 4 LUTs (3 inputs LUTs). For that, a simple implementation of (4 × 5) crossover RO PUF structure, which requires 16 LUTs for implementing the interesting logic and 25 LUTs needed to implement the 25 inverters. The proposed lightweight CF-ROPUF is an area efficient design mapped inside a single CLB, as shown in Table 1.

As seen in the table, our CF-ROPUF and XOR PUF consumes less hardware resources on FPGA, including one CLB that has 8 LUTs and 8 flip flops (F.F.), as compared with other PUF designs, that require four or five CLBs with larger number of LUTs and F.F. when mapped on reconfigurable logic (FPGAs devices). For each FPGA chip, an instance of the design is mapped into an FPGA area (40 CLBs). In our design, CLB logic is efficiently utilized to map 16 ROs with an area overhead of eight LUTs and six dedicated multipliers, as seen in Fig. 4(a). The main logic of the design only consumes 320 LUTs (8 LUTs * 40 CLBs) and 240 dedicated multiplexers (6 multiplexers * 40 CLBs). Also, for the sake of more area overhead comparison, the neighbor coding algorithm is used to generate 256-bit PUF response from our CF-ROPUF and the state of the art [16], [19], [23].

For that, our design only uses the 40 CLBs while the other designs need to occupy 256 CLBs of the same FPGA family, Spartan 3E FPGA. This clearly shows that compared to the existing PUF techniques, our design is more area efficient that uses less on-chips design area for the generation of a large number of highly reliable secret keys. Further, the proposed cROPUF design efficiently utilizes the same area (one CLB) of Spartan 3E FPGA to generate a 16-time larger response bit as compared to simple ROPUF and 1-out-of-8 techniques, mentioned in these papers [16], [19], [23].

B. PUF UNIQUENESS RESULTS

PUF uniqueness is one of the essential performance metrics of a secure PUF design. An average PUF uniqueness value determines how the digital signatures generated after implementing the same PUF design on different silicon devices can differ. The uniqueness of our PUF design is calculated using inter-die Hamming Distance (HD) for Spartan 3E FPGA chips using the following equation [23]:

$$U = \frac{2}{m(m-1)} \sum_{u=1}^{m-1} \sum_{v=u+1}^{m} \frac{HD(R_u, R_v)}{n} \tag{18}$$

where u and v are two chips, Ru and Rv are response vectors of u and v chips, n is the number of PUF instances, and m is the number of FPGAs. Ideally, bit responses should only depend on the random process variation independent of RO locations. According to earlier research [19]–[23], statistically average Hamming distance value is theoretically expected to be 0.5.

TABLE 1. Hardware overhead comparison.

PUF Design	Device Type	Hardware Overhead		
		Number of CLBs/ Slices	Number of LUTs	Number of F.F.
XORPUF [57]	Xilinx Spartan-6 FPGAs	One Slice	8	16
Lowcost PUF [58]	Xilinx Spartan-6 FPGAs	Four CLBs	32	64
TCRO SRAM PUF [59]	Xilinx Virtex II FPGAs	Five CLBs	45	45
Cross PUF [60]	Xilinx Spartan3 FPGAs	Five CLBs	45	45
Our CF ROPUF	Xilinx Spartan3 FPGAs	One CLB	8	8

TABLE 2. Uniqueness comparison between our cROPUF and earlier sPUF designs.

PUF design	Simple ROPUF [19]	XOR PUF [57]	Low cost PUF [58]	Ultra PUF [59]	Cross. PUF [60]	SRAM PUF [33]	Our PUF
Uniq. (50%)	46.15%	48.76%	49.97%	48.3%	49%	52.02%	49.90%

An average uniqueness value close to 0.5 indicates that any response generated by implementing PUF on a certain deceive is genuinely random and uniquely independent of any other response generated by implementing the same PUF on a different chip. Therefore, PUF responses can be seen as random sets with a 50% probability of having 0 and a 50% likelihood of having 1 for each response. In this case, the average HD value of the responses is expected to be 50%.

For n generated PUF responses (secret keys), the average PUF uniqueness can be found using the average value of the calculated Hamming distances between PUF responses. The Hamming distance between two PUF responses generated from two different devices is produced by comparing response bits of a PUF instance with the corresponding response bits of other PUF instances. The distribution of Hamming Distance values produced is then constricted using the obtained HDs, known as the inter-die Hamming distance distribution. To generate a PUF response with 256 bits, neighbor coding selection algorithm is used. The following comparison equation is used to generate each response:

$$r_i = \begin{cases} 1, & if\ f_i \geq f_j \\ 0, & Otherwise \end{cases} \tag{19}$$

To estimate the average uniqueness of our PUF design, a total of 30 PUF secret keys (each key has a length of 256-bits) are generated after mapping the PUF instance on 30 different FPGA devices. For n generated PUF responses (each response has 256-bits secret key), the average PUF uniqueness is calculated using the average value of Hamming distances between PUF response bits. The Hamming distances between each two PUF responses r_i and r_j, generated from two different FPGA chips ($chip_i$ and $chip_j$) are obtained. For that, a an XOR binary comparison that compare each bit in r_i PUF response with the corresponding bit in the r_i PUF response is needed. The average value of Hamming distances is then calculated to determine the overall average uniqueness of the proposed PUF design. As shown in Fig. 7, the distribution of the calculated Hamming distance values

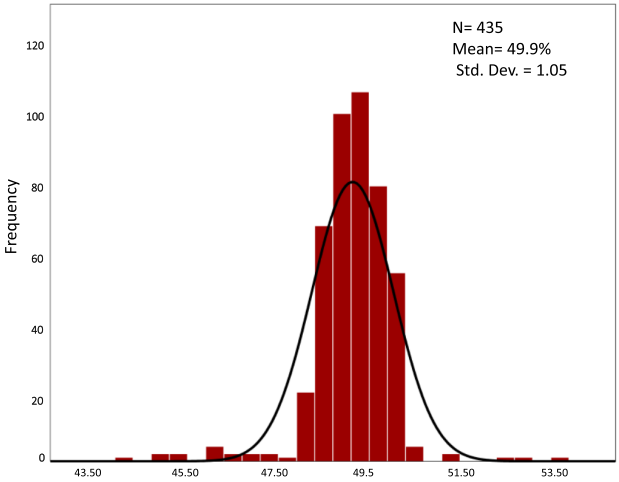


FIGURE 7. Average CF-ROPUF Uniqueness test results.

is then constricted using the obtained HDs, which is known as the inter-die Hamming distance distribution. The average value of our CF-ROPUF uniqueness is calculated using inter-die Hamming Distance (HD) 30 Spartan 3E FPGA. For the average uniqueness shown in Fig. 7, there are a number of ($N = 435$) binary comparison between $n = 30$ unique PUF responses. The value of N can be also calculated based on the following equations:

$$N = \frac{n!}{2!\,(n-2)!} \tag{20}$$

Table 2 compares the uniqueness of our lightweight, configurable PUF design with the competing PUF techniques. The table shows an improved average uniqueness of our PUF of 49.9% with a standard deviation of 1.05%. As compared to earlier PUF techniques, which is very close to the optimal value (50%). This indicates that the proposed lightweight is promising to authenticate consumer electronic and smart devices, including IoT devices. This shows that the proposed LFSR PUF is a robust hardware-based design that can be

utilized to generate unique secret keys for cryptographic applications.

C. PUF RELIABILITY RESULTS

1) IFA RELIABILITY TECHNIQUE

Inspired by the 1-out-of-k scheme, the 1-out-of-r PUF coding algorithm is proposed to quantify the IFA RO sample frequencies to generate reliable PUF response bits. After the reliable RO sample frequencies are obtained using the IFA technique, the 1-out-of-r coding algorithm uses these r RO frequencies to generate reliable PUF-based secret keys for IoT-based applications. For that, a 1-out-of-r coding algorithm obtains $(r-1)$ reliable RO frequency pairs. The CF-ROPUF maps the r ROs in a single CLB, hence for any two CLBs, only the two RO frequencies with the maximum frequency difference are selected.

The following equation represents the frequency pairs obtained from n CLBs using the 1-out-of-r as:

$$F_i = \{(f_{[CLB_0]}, f_{[CLB_1]}), (f_{([CLB_1]}, f_{[CLB_2]}) \ldots, (f_{[CLB_{n-1}]}, f_{[CLB_{(n)}]})]\} \quad (21)$$

The CF-ROPUF design is mapped on an FPGA region with $r = 40$ CLBs, as shown in Fig 5(b). There are $r-1 = 39$ RO frequency pairs in total. Since each pair is used to generate one PUF response bit (PUF-based secret key), the length of the PUF-based key will also be $(r-1 = 39)$ for each FPGA region. The average frequency ranges between 268 MHz and 282 MHz, with an average of 274.78 MHz and a standard deviation of 3.062 MHZ. As mentioned earlier, HW is used to select the RO sample frequencies that are reliably regenerated. The average HWs range from 97.3% to 99.2% with an average of 98.5% and a relatively low standard deviation of 0.71% for all 30 FPGA regions, as shown in Fig. 8.

This indicates that by using the IFA technique to improve reliability, on average, 98.5% of the generated RO sample frequencies can be used to obtain a reliable (accurate, repeatable, and reproducible) PUF response bit.

2) PUF ACCURACY

Based on ISO definition, data measurements that exhibit trueness and precision are considered *accurate*. Test measurements that exhibit trueness may not show good precision and vice versa. The measure of trueness is usually expressed in terms of bias between a set of actual or expected measurements and their mean (i.e., reference) value.

The RO sample frequencies are measured and compared with their mean values (reference values) to estimate trueness. Alternatively, precision is expressed in terms of average diverseness (standard deviations) of data measurements. For an estimation of the expected values, RO sample frequencies are directly-measured from the hard-macro designs (expected values). These measurements are collected from 30 CLBs mapped on the 30 FPGAs, where one CLB is randomly selected from each FPGA. For the selected CLB, 16 RO frequencies are measured directly from the hard macro design

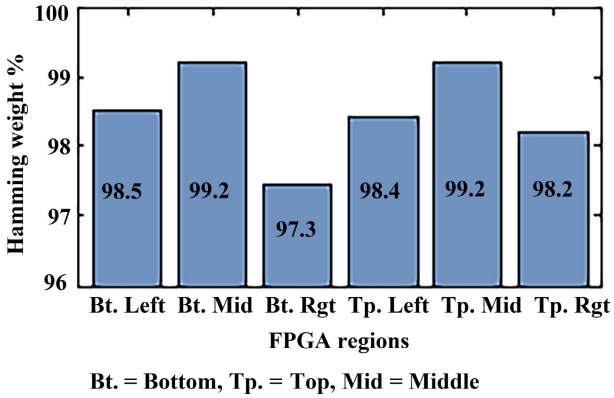


FIGURE 8. Distribution of HW percentage for reliable RO frequencies after applying IFA technique.

with the help of the logic analyzer. A total of 16 RO sample frequencies are measured per FPGA chip from one CLB. The 16 collected frequencies for each FPGA chip are denoted as

$$F_{(expected)} = \{f_0, f_1, \ldots, f_{15}\} \quad (22)$$

Each of the measured values is then compared with calculated mean values (reference values). For the calculations of the mean values, the CF-ROPUF structures that incorporate the ROs, Muxes, De-Muxes, counters, challenge generator, etc. are physically implemented on all FPGAs. After this, the average values of the 16 RO sample frequencies collected from each chip under nominal operating conditions are calculated. Average trueness values are estimated for each chip in terms of the absolute variability (difference) between the actual values (expected values) of the individual RO frequencies and their reference (mean) values as

$$Trueness = |F_{reference} - F_{expected}| \quad (23)$$

Theoretically, the variability between the reference and expected values is zero, which indicates ideal trueness (100%) of the generated RO frequencies. However, as shown in Fig. 9(a), the trueness values of sample RO frequencies that are collected from 30 CLBs range from 0.49 MHz to 0.09 MHz with an average of 0.27 MHz and a standard deviation value of 0.1 MHz. These low average absolute trueness and standard deviation values of RO sample frequencies indicate that they are close to their mean values, and thereby they are truly measured.

Precision is an essential factor to determine the accuracy of results since it indicates the closeness of the agreement between independent test results obtained under the same conditions. To calculate the precision value, the total average value (T_{avg}) of the mean values ($F_{reference}$) for each FPGA is calculated as follows:

$$T_{avg} = \frac{1}{m} \sum_{i=1}^{m} F_{reference} \quad (24)$$

where m represents the number of FPGAs. The average precision of RO frequencies is calculated as

$$Precision_i = \sqrt{\frac{\sum_{i=1}^{m}(T_{avg} - F_{reference})^2}{(m-1)}} \quad (25)$$

The average RO frequencies collected from the top-right areas of 30 FPGAs (640 frequencies) are shown in Fig. 9(b). The average precision for these RO frequencies is determined to be 3.78 MHz. The collected frequencies range from 258.4 MHz to 276.7 MHz, with an average of 267.5 MHz. The range (Max_{value} - Min_{value}) is 18.3 MHz, and data are precisely measured without any noticeable outlier due to a potentially faulty FPGA chip or aging effects. It can be concluded that the collected measurements are relatively close to each other and, therefore, exhibit good precision.

A measurement exhibits repeatability only if the probability of regenerating the same measurement results under fixed operating conditions is high, according to the ISO definition. These operating conditions may include test equipment, operators, and/or environmental conditions. The repeatability is not applicable if the experiment is performed by another person, even using the same lab equipment under the same environmental conditions. The repeatability of the CF-ROPUF is evaluated from 15 randomly selected FPGA regions from the 30 FPGA chips. Each one of these regions contains $n = 40$. Reliable RO frequencies are determined from these CLB after applying the IFA reliability technique. The 1-out-of-r coding is used to generate PUF response bits for each group. The generation of the response bits is repeated $n = 10$ times using ten different RO sample frequencies.

3) PUF REPEATABILITY

The CF-ROPUF repeatability is evaluated in terms of intra-die Hamming distance (HD) for the response bits generated by each group individually using

$$Repeatability = \frac{1}{n}\sum_{i=1}^{n}\frac{HD(r_l, r_l')}{n} \times 100\% \quad (26)$$

where r_l and r_l' are two response bits, from n responses, regenerated using the same input challenge under nominal operating conditions. The HD for ideal PUF repeatability is 100%. HD percentage of the average repeatability of the CF-ROPUF responses at nominal operating conditions is shown in Fig. 10. For the 15 FPGA regions, the average HD values range from 97.1% to 99.8% with an average of 98.2% and 0.98 standard deviations at the nominal operating conditions. These results demonstrate that the CF-ROPUF exhibits high repeatability at nominal operating conditions.

4) PUF REPRODUCIBILITY

ISO defines reproducibility as the probability of regenerating the same measurement result from an independent test under varying operating conditions. PUF reproducibility is determined in terms of the 'reproducibility' of the generated response bits under varying temperatures and supply voltages

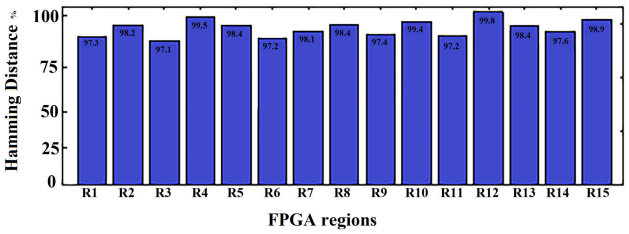


FIGURE 9. Percentage of average repeatability of CF-ROPUF under nominal operating conditions.

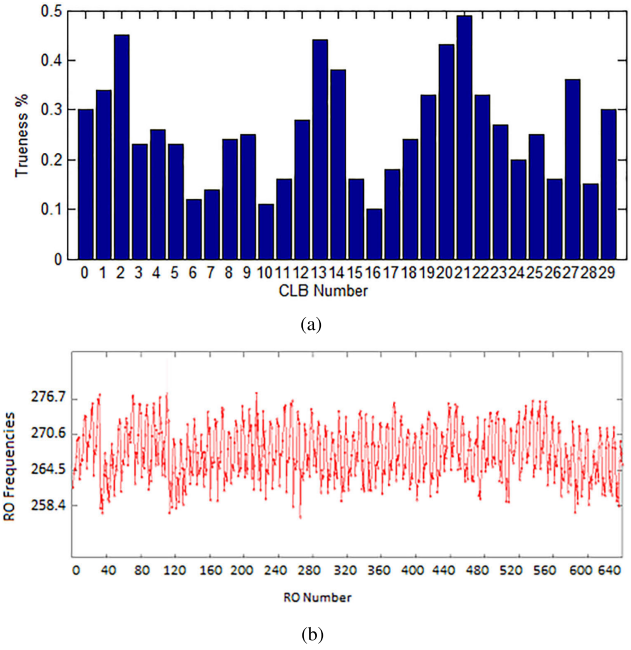


FIGURE 10. Estimation of PUF accuracy: 10(a) Average trueness of 30 CLBs; 10(b) Average precision of 640 RO frequencies.

in this work. The average values of the regional frequencies change between 263.2 MHz and 281.2 MHz with an average of 274.46 MHz and a standard deviation of 5.6 MHz under temperature variations, as listed in Table 3. Alternatively, the average value of the regional frequencies varies between 227.2 MHz and 300.4 MHz with an average of 263.1 MHz and a standard deviation of 27.05 MHz under a supply voltage varying between 1.0 V to 1.3 V.

Environmental variations cause the range of measurement results to fluctuate, leading to reliability issues. A higher temperature decreases the average regional frequency while a higher voltage increases average regional frequency. The average PUF reproducibility is evaluated at five temperatures and four voltage levels. Accordingly, PUF response bits generated at nominal operating conditions (r_c) are compared to the PUF response bits generated at varying operating conditions (r_c') using Hamming distances (HDs) as

$$Reproducibility = \frac{1}{m}\sum_{j=1}^{m}\frac{HD(r_c, r_c')}{m} \times 100\% \quad (27)$$

TABLE 3. The environmental variation effects on regional average RO frequencies.

Temperatures	FPGA Regions						Voltages	FPGA Regions					
	Bottom			TOP				Bottom			TOP		
	Left	Middle	Right	Left	Middle	Right		Left	Middle	Right	Left	Middle	Right
85^oC	-	-	-	-	-	-	1.5V	-	-	-	-	-	-
75^oC	263.2	264.3	263.6	263.8	264.6	264.6	1.4V	-	-	-	-	-	-
50^oC	269	270.5	267.8	269.2	270.8	266.6	1.3V	298.6	300.2	297.6	299.3	300.4	299.46
25^oC	273.9	275.6	272.9	274.7	275.8	274.7	1.2V	274.1	275.9	273.6	274.2	276.4	274.12
Room Temp	274.1	275.9	273.6	274.2	276.4	274.1	1.1V	249	250.1	249.3	249.6	250.3	250.2
0^oC	279.6	281.2	278.6	279.8	281	279.7	1.0V	228.4	229.9	227.2	228.6	230.2	228.1

'-' depicts reading cannot be taken at these temperatures or supply voltages.

where r_c and r_c' are two response bits, from n responses, regenerated using the same input challenge under nominal operating conditions. The reproducibility of CF-ROPUF responses generated at five temperatures and four voltage levels for 30 different regions of the tested chips are shown in Figs: 11 (a) and 10(b). RO sample frequencies of selected 30 FPGA regions (one region for each chip) are paired and quantified to apply the 1-out-of-r coding algorithm described earlier. The length of the quantified response bits is 39 $bits \times 30\ chips = 1170\ bits$. Fig. 11(a) shows the reproducibility value, under different temperature variations, is ranging from 99.5% to 99.9% with an average of 99.73% and a standard deviation of 0.17%.

Similarly, Fig. 11(b) shows that the average reproducibility value, under different voltage variations, is ranging from 99.3% to 99.8% with an average of 99.53% and a standard deviation of 0.25%. Bit-flip is the probability that a response bit flips from '0' to '1' or '1' to '0' due to different external factors including environmental variations. From Figs. 12(a) and (b), it is indicated that the average percentage of unstable bits due to temperature variations is only 0.27%, which is smaller than the percentage of bit-flips (0.47%) due to voltage variations. This leads to the observation that the CF-ROPUF reproducibility is slightly more sensitive to voltage fluctuations than temperature variations.

As seen in Table 4, the overall average PUF reliability concerning both temperature and voltage variations is found to be 99.63%. The PUF reproducibility also has a relatively low calculated standard deviation of 0.14% and an overall bit-flip percentage of 0.37%. Therefore, these experimental results demonstrate a good overall reproducibility of CF-ROPUF sample frequencies under varying temperatures and supply voltages. The reliability of our configurable PUF with different types of silicon PUFs that are suitable for FPGA [16], [55], [56] are compared in Table 4. Our CF-ROPUF shows a slightly better average reliability as compared to the other designs. This shows that the CF-ROPUF is a strong candidate for lightweight IoT devices due to the area-efficient design that can generate a larger number of reliable secret keys at varying operating conditions as compared to conventional ROPUFs.

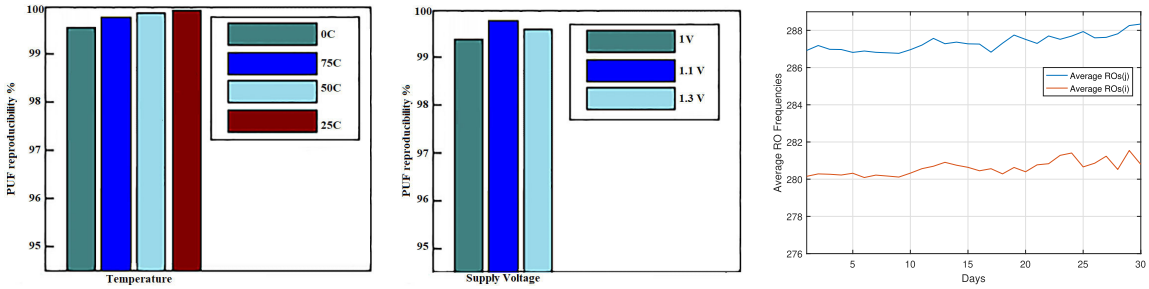
5) AGING

As soon as the voltage exceeds 1.4V and/or temperature exceeds $85°C$, RO frequencies are not captured by the Logic Analyzer, as indicated in Table 3. This is an example of a temporal functional failure. Due to the temperature variations, the average regional frequency generated by the ROs exceeds 300 MHz, which is the maximum frequency limit the Spartan 3E FPGA operates on. Additionally, the used FPGAs do not operate properly when the temperature is higher than $85°C$. The failure causes FPGA to work unreliably, and therefore, data samples cannot be completely measured from the individual ROs. While the effects of temporal variations (change in the ambient temperature and supply voltage) can be reversible, the effects caused by accelerated aging are irreversible. Accordingly, a simple experiment to study the potential impact of accelerated aging on PUF reliability is performed.

A total of ten ROs (five frequency pairs) are mapped on ten adjacent CLBs. Using the reliable IFA technique, the ability of these ROs to reliably regenerate the same response for n times is evaluated. For enhanced reliability, the ROs with the maximum frequency difference is selected by applying 1-out-of-r coding. The impact of aging on the average value of the RO frequency pairs mapped on nine Spartan 3E FPGAs over 30 days is shown in Fig. 11(c). The ROs have minimal frequency fluctuations within 1 MHZ after 30 days of aging. Since there is no overlap between the generated frequencies, the results indicate no bit flip events in the response bits and no reliability issues due to the accelerated aging. The overall observation is that the reliability of FPGA chips can be affected temporally by the change of the operating conditions (temperature and voltages), as depicted in Table 3. Ignoring the impact of varying operating conditions, we conclude that CF-ROPUF design is functioning reliably after applying the proposed IFA technique and 1-out-of-r coding.

D. RANDOMNESS

As previously mentioned, the central limit theorem (CLT) states that a normal distribution should exhibit true randomness for its uniformly distributed data samples [17], [18]. So as a first step, we define normality, study, and enhance



(a) (b) (c)

FIGURE 11. Average CF-ROPUF reproducibility (a) Under varying temperatures; (b) Under varying supply voltages; (c) Aging impact on the average value of five RO frequency pairs mapped on 9 Spartan-3E FPGAs.

TABLE 4. Reliability comparison between our c-ROPUF and earlier sPUF designs.

PUF Design	APUF [16]	ROPUF [55]	LFSR ROPUF [56]	Crossover ROPUF [60]	SRAM PUF [33]	our c-ROPUF
PUF Reliability (100%)	99.20%	99.14%	96.30%	98.7%	97.73%	99.63%

the normality of RO data sample frequencies. After improving the normality, the true randomness of the normalized generated PUF response bits is tested using the 15 NIST randomness standard tests.

1) NORMALITY

The normal distribution, also called the Gaussian distribution, shows a symmetric probability distribution around a mean value that appears as a bell curve shape. Average regional RO frequencies are calculated, the average RO frequency for each FPGA region, at varying temperatures 't' or voltages 'v'. The IBM-SSPS software is used to examine the normality of the implemented generated RO sample frequencies.

Figs. 12(a)-(e) show the constructed data histograms that represents the distribution of average RO frequencies for 30 chips before and after the application of Ln_γ normalization technique at $0°C$, $25°C$, $RT°C$, $50°C$ and $75°C$. From the figures, one cannot conclude whether or not the data samples are well represented by normal distributions. Thus, data normality is evaluated in terms of mean, median, standard deviation, variance, skewness, kurtosis, and the standard error of the mean under varying temperatures, as tabulated in Table 5. In statistics, when the mean and median are close to each other, it is expected that the data sample exhibits good normality. As seen in Table 5, the mean and median values are very close to each of the applied temperatures. This is true for the measured average RO frequencies before applying Ln_γ, as well as the normalized RO frequencies using Ln_γ. It is expected (but not confirmed) that both data samples exhibit normality (before and after the application of Ln_γ. Standard deviations and variances measure the diverseness and variability of the data samples, respectively.

From the same table (Table 5), it is noticed that with the application of the probability Ln_γ technique, data samples exhibit higher diverseness and variability. Average diverseness and variability are significant measures for the PUF reliability [2], [5], [26]. Thus, it is expected that data samples are more reliable under varying conditions with the implementation of Ln_γ on real hardware. The standard error of the mean (SEM) is calculated by dividing the standard deviation (S.D.) of the test results by the square root of the data sample size (n) as follows:

$$SEM = \frac{S.D.}{\sqrt{n}} \tag{28}$$

The SEM provides an estimate of the confidence interval for the mean of the population. Similar to standard deviations, SEM values are calculated using IBM-SPSS and multiplied by 1.96 to obtain a rough estimate of 95% from the population means to settle in the normal distribution as assumed by the software. The standard errors are inversely proportional to the sample size. Since the sample size is fixed (640), the standard errors are proportionally affected by standard deviation values. This means large data samples with small standard deviations result in small values of the standard errors. The smaller the standard error, the more representative the mean of the data. In cases of large standard errors, the data samples can experience significant irregularities and less likely to be normally distributed.

From Table 5, one can notice that the application of Ln_γ leads to increments in the calculated SEM value. This is expected due to the high standard deviations. Skewness and kurtosis are among the important parameters to confirm the normality of histograms of data samples. In theory, the skewness of a random data sample ($n = f_0, f_1, \ldots, f_n$) that is represented in a histogram distribution is referred to as the

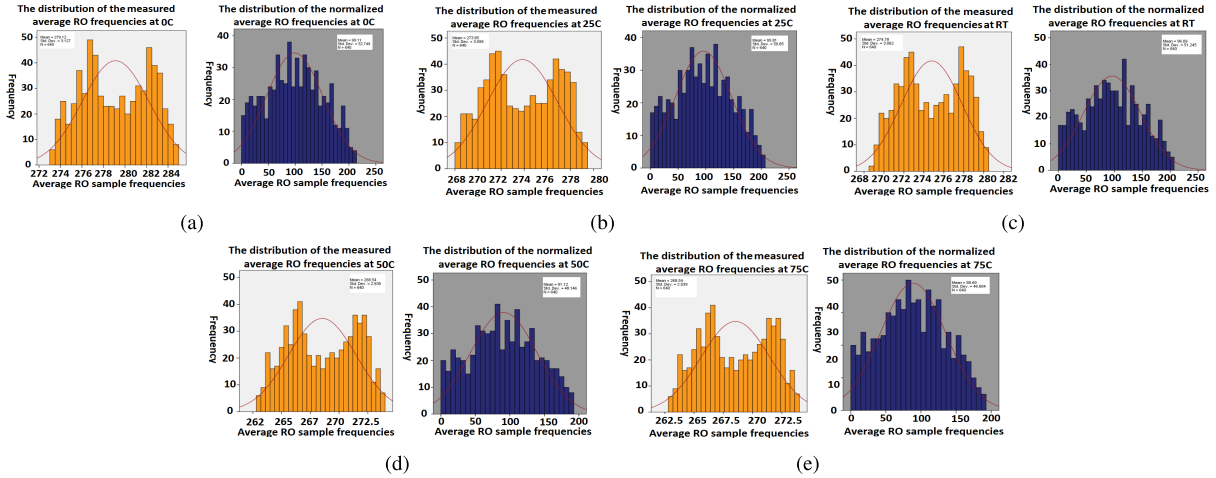


FIGURE 12. Distribution of average RO frequencies for 30 chips before and after the application of logarithmic gamma function normalization technique: (a) 0°C, (b) 25°C (c) RT°C, (d) 50°C and (e) 75°C.

TABLE 5. The estimation of data normality with different parameters.

Temp.	RO frequencies	Descriptive Statistics									
		Range	Min	Max	Mean	Mean	Std. Deviation	Variance	Skewness	Kurtosis	Std. Error of the mean
0C	MD	11.76	273.16	284.92	279.12	279.03	3.13	9.778	-0.014	-1.24	0.1236
	NZ	210	1	211	99.11	99	52.75	2782.47	0.043	-0.886	2.085
25C	MD	11.47	268.02	279.5	273.85	273.79	3.056	9.341	-0.014	-1.243	0.1208
	NZ	204	0	204	95.81	96	50.85	2585.77	0.041	-0.875	2.01
RT	MD	11.54	268.95	280.5	274.79	274.74	3.062	9.377	-0.015	-1.236	0.1211
	NZ	206	0	206	96.09	95.5	51.25	2626.06	0.048	-0.868	2.026
50C	MD	10.99	262.98	273.97	268.54	268.46	2.94	8.636	-0.014	-1.248	0.1162
	NZ	192	0	193	91.12	91	48.15	2318.07	0.04	-0.862	1.903
75C	MD	10.87	258.8	269.66	264.24	264.18	2.89	8.33	-0.012	-1.252	0.1141
	NZ	191	0	191	88.69	88.5	46.68	2179.35	0.039	-0.854	1.845
MD: Measured average RO frequencies; NZ:Normalized average RO frequencies.											

Pearson's moment coefficient of skewness [18], [61], [62]. Skewness is a quantified probability measurement that determines the asymmetry of the histogram distribution about its mean value. The average skewness of this data can be zero (normally distributed data), positive (skewed to the right), or negative value (skewed to the left) that is determined using the following equation [61]:

$$Avg_{(skewness)} = \sum_{i=1}^{n} E(\frac{f_i - \mu}{\delta})^3 \tag{29}$$

where f_i is an average RO sample frequency, μ is the mean value of the population, δ is the standard deviation of the sample population, and E is the expectation operator. Similarly, kurtosis, a Greek term which literally means curved, is a quantified measurement to determine the sharpness of the histogram distribution about its mean value [18], [61], [62]. The average kurtosis of n random variables can be zero (normally distributed data), positive or negative value defined by the following formula [61]:

$$Avg_{(kurtosis)} = \frac{f_i^4}{\mu^4} = \frac{E(f_i - \mu)^4}{(E(f_i - \mu)^2)^2} \tag{30}$$

The μ represents the mean value of the population, μ^4 is known as the fourth central moment, and δ is the standard deviation of x_i. For normally distributed data, both the skewness and kurtosis values should be theoretically zero. However, having a negative average kurtosis ($avg_{skewness} \leq 0$) means that distribution will become a smoothened peak with more readings near the tail. On the other hand, a distribution that has more readings near the center (sharper peak than normal) will have a positive kurtosis ($kurtosis \geq 0$). It can be concluded from the table that Ln_γ shows no improvement in the skewness of data samples and lead to more reading near the right tails, i.e., data histogram is more skewness to the right. However, the application of Ln_γ noticeably improves the kurtosis values, as it is also noticed from Figs. 12(a)-(e). From the results, it is not clear how the average frequencies vary according to the normal distribution. Therefore, further tests to confirm data normality are needed.

To further assess the normality of the RO sample frequencies represented by data histograms, Q-Q plots are constructed using IBM-SPSS statistical software. Since the main objective is to compare the obtained data histogram distribution to the normal distribution that only varies in location and scale, the location and scale parameters are estimated

TABLE 6. Estimated QQ plot parameters for normality.

Temp.	0C		25C		RT		50C		75C	
RO Frequency	MD	NZ	MD	NZ	MD	NZ	MD	NZ	MD	NZ
Location	279.12	99.16	273.85	95.81	274.78	96.09	268.54	91.12	264.23	88.69
Scale	3.13	52.75	3.06	50.85	3.06	51.25	2.94	48.15	2.89	46.68
MD: Measured average RO frequencies; NZ: Normalized average RO frequencies.										

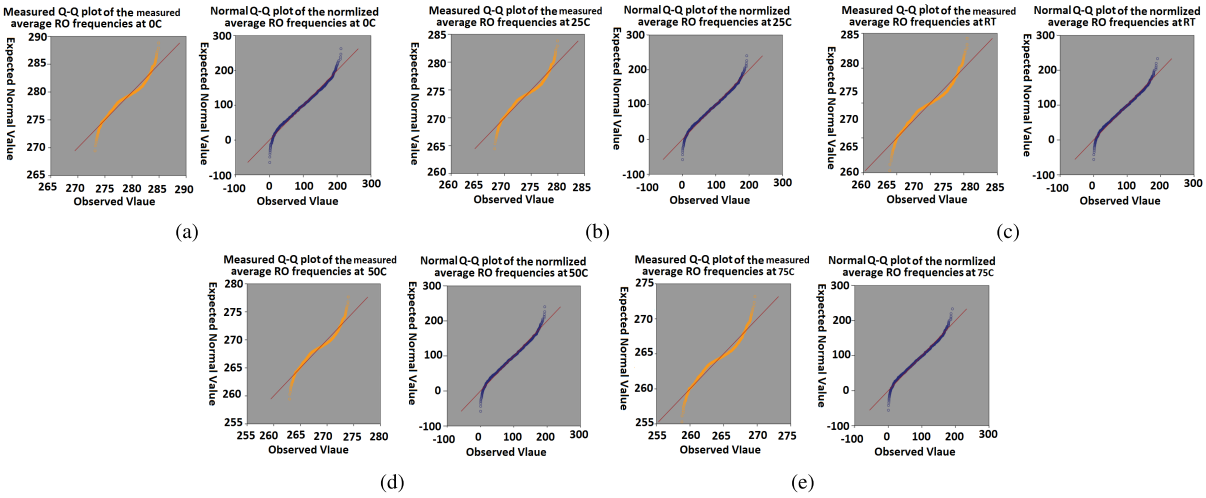


FIGURE 13. Normal Q-Q plots for the average RO frequencies before and after the application of Ln_γ normalization technique: (a) 0°C, (b) 25°C (c) RT°C, and (d) 50°C and (e) 75°C.

using the Q-Q plot shown in Table 6. A point of linear patterns indicates that data samples reasonably describe the normal distribution. The location and scale parameters are estimated for data samples, such as the intercept and slope of the linear pattern shown in Figs. 13(a)-(e). A Q-Q plot is a typical graphical illustration for determining the exact normality of data histograms. It compares the empirical cumulative distribution probabilities of a data set (RO sample frequencies) with their theoretical cumulative probabilities (normal distribution) [18]. The correlation between the expected (normal) and observed probabilities (cumulative) can be written as follows:

$$Expected(f_i) = P(f < i) \tag{31}$$

$$Observed(i) = \frac{(i - \frac{1}{2})}{n} \tag{32}$$

The location parameter shown in Table 6 represents the mean values of the distortion that are tabulated in Table 5. However, scales are the Y slope of the plots. As shown in Figs. 13(a)-(e) of Q-Q plots, with the application of Ln_γ, Ro sample frequencies collected under different temperatures follow almost the same trend, which typically lies on the straight (normality line). From the same figures, it is noticed that data samples collected before the application of Ln_γ exhibit irregularity and do not fall on the normality line. This confirms that the normalized RO sample frequencies (NZ) are better represented by using the normal distribution as compared to the measured frequencies (MD) before the application of gamma, and thus, the normality is improved.

2) SYSTEMATIC VARIATIONS MITIGATION

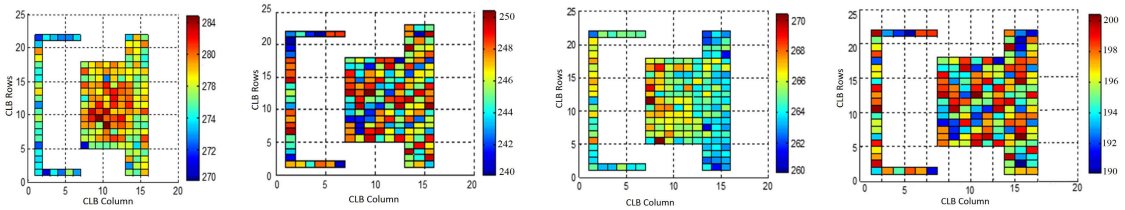
Spatial systematic correlations negatively affect the randomness of the generated responses with the pairing strategy of the original chain-like coding. Several security techniques, including regression-based distiller and a pseudo-random (PRN) technique, and logarithmic and absolute diverseness technique (LDT), have been proposed to mitigate systematic process variations and improve PUF randomness. For example, the regression-based distiller technique has been applied with multiple coding algorithms, including 1-out-of-8 coding, chain-like neighbor coding, decoupling neighbor coding, S-sequence, and T-sequence. Additionally, a pseudo-random technique (PRN) was introduced to decrease the impact of systematic process variation on ROPUFs. The LDT technique (a novel security technique based on base-10 logarithm function and the square root of RO deviations from the global mean) has been proposed in our earlier work to mitigate the effect of systematic spatial variation and also to improve the randomness of the response bits generated using a unique reconfigurable ROPUF design.

Figs. 14(a) and 14(b) show the distribution of average RO sample frequencies of 240 CLBs after and before the application of the proposed Ln_γ normalization techniques explained earlier. As shown in the figures, the application of Ln_γ with the 1-out-of-r coding on the average RO sample frequency of two FPGA chips at both 0°C and 75°C temperatures has efficiently randomized the frequency distributions across all CLBs and removed clustered frequency regions due to the negative impact of systematic process variation [9].

TABLE 7. Analyses of $P-values$ and *Proportion* probabilities for the 15 NIST test suites for randomness.

Statistical test	Chain like neighbor		Decoupled chain like neighbor		1-out-of-r	
	$P-value$	$Proportional$	$P-value$	$Proportional$	$P-value$	$Proportional$
Frequency	0.574077	30/30	0.303409	28/30	0.153436	30/30
Block frequency	0.000053	29/30	0.000000	12/30	0.109640	29/30
Cumulative sums (m-2)	0.004740	30/30	0.00399	30/30	0.00952	30/30
Cumulative sums (m-3)	0.000488	29/30	0.000499	28/30	0.00314	29/30
Runs	0.000035	29/30	0.000017	30/30	0.003650	30/30
Longest run	0.000000*	22/30*	0.000007*	0/30*	0.740921	29/30
Approximate entropy	0.000001*	0/30*	0.096812	29/30	0.000312	22/30*
Serial (forward)	0.379030	29/30	0.000305	28/30	0.023324	30/30
Serial (backward)	0.000025	1/30*	0.02728	30/30	0.000289	30/30
Binary matrix rank	0.000014	30/30	0.07931	29/30	0.842342	29/30
Discrete Fourier transform	0.430245	30/30	0.000000	0/30	0.000017	29/30
Non-overlapping template	0.000000*	0/30*	0.000015	30/30	0.000002*	2/30*
Overlapping Template	0.330245	30/30	0.045682	29/30	0.012589	30/30
Universal statistical	0.000000*	29/30	0.001508	30/30	0.002978	30/30
Linear complexity	0.000576	3/30*	0.000428	11/30*	0.000380	30/30

[*] indicates a randomness failure in one of the applied NIST tests.



(a) (b)

FIGURE 14. Mitigation of systematic process variations (a) Chip 1 at 0°C; (b) Chip 2 at 75°C.

3) NIST RANDOMNESS TEST

The NIST statistical test suite for random and pseudorandom number generators and cryptographic applications including the approximate entropy test, binary matrix rank test, discrete Fourier transform, non-overlapping template, overlapping template, universal statistical, and linear-complexity require a large random sequence with over 1000 binary response bits. Thus, they are not suitable for designing true random numbers using simple ROPUF designs [17]. To test the PUF randomness, different coding algorithms, chain-like neighbor, decoupled chain-like neighbor, and 1-out-of-r that exploits the RO frequencies, normalized using logarithmic gamma Ln_γ technique, are used to generate the PUF-based binary response bits.

The randomness of the generated response bits are justified using the 15 NIST test suites for randomness over a population of 30 chips (30 random sequencers). For an applied sequence, a NIST test calculates an empirical *Proportional* probability [17]. The probability $P-value$ of $P-values$ in n sub-intervals (S_i), $i \leftarrow 1,\ldots,n$, is computed using the incomplete gamma function ($igamc$) based on (χ^2) distribution and the goodness-of-fit distribution F_i as follow [17]:

$$\chi^2 = \frac{\sum_{i=1}^{n}(F_i - \frac{S_i}{n})^2}{(\frac{S_i}{n})} \tag{33}$$

$$P-value = igamc\,(\frac{n-1}{2}, \frac{\chi^2}{2}) \tag{34}$$

According to NIST, for a particular NIST test, if 4% or more of all tested sequences are significant (more than 4% of the tested sequences have a P-value $\leq 1\%$), the 'Proportional' test fails; otherwise, it passes. For the Proportional event, since there are 30 sequences that are generated by the CF-ROPUF scheme, at least 29 sequences (96%) should pass with a significance level of $\alpha = 1\%$ [17].

Besides, the NIST frequency test examines all sequences to determine their uniformity to test whether or not the distribution of 1's to 0's are uniformly distributed in all of the tested sequences. Tested sequences are considered uniformly distributed only when critical $P-value$ of the calculated $P-values$ is greater than or equal to 0.0001 ($P-value$ of $P-values$) $\leftarrow$ 0.0001 [17].

Based on the chain-like coding explained earlier, the number of generated random binary responses out of any FPGA region is $16 \times (n-1) = 624$ bits. The total generated binary sequence from each chip (6 regions) will be $624 \times 6 = 3744$ bits. This represents a relatively large number of binary responses that satisfy all 15 NIST test requirements where at least 1000 responses are needed per each sequence. The obtained randomness results from the NIST statistical test for CF-ROPUF after applying the Ln_γ function technique with chain-like neighbor coding are given in Table 7. From the table, it can be seen that out of the 15 NIST tests, the chain-like coding completely fails to pass the Longest Run, Non-overlapping and Approximate Entropy. It also partially fails the Serial (back-word), Universal Statistic, and Linear Complexity Tests. We believe that these expected failures are due to the high intrinsic dependencies by the systematic spatial correlations caused by the layout

dependency [7]–[9], [36]. Although the chain-like coding passes (partially or fully) at least 12 NIST tests out of 15 trials with a percentage of 80% for the $P-value$ test and 73.3% for the *Proportional* analysis, the results are better than the existing techniques, including the regression-based distiller and PRN techniques where chain-like-coding completely fails the entire NIST test suites [17]. We strongly believe that the failure rate can be significantly reduced with the application of different normality techniques and coding algorithms to nullify the negative impact of systematic process variations completely. Considering decoupled neighbor coding as a solution to this problem helps to improve passing NIST randomness tests. The generated response using decoupled neighbor coding is $16 \times 20 = 320$ bits and the total binary sequence generated from FPGA (6 regions) is $624 \times 6 = 1920$ bits.

As shown in Table 7, decoupled neighbor coding only fails to pass the Longest Run thoroughly. It fails to partially pass the Linear complexity with a 13.3% improvement rate over the original chain like coding. Future investigation using different random responses generated with varying security techniques is needed to improve the randomness of the generated PUF responses. For the generation of random binary sequences with significant binary response bits, the 1-out-of-r coding algorithm uses the five least important bits of each input challenge of the fastest RO in each CLB. Therefore, test sequences that are 5 times larger are generated. For example, assume that the input challenge 000000101 picks the fastest RO frequency RO_5 out of 16 frequencies that are mapped on the first CLB (CLB_0). In this case, the 5 least significant bits of this challenge (00101) are used in the generation process of PUF binary sequences from each FPGA. Thus, the binary bits generated out of each FPGA region are maximized to 40 $CLBs \times 5 = 200$ bits. Hence, each FPGA produces a random sequence with a length of $6 \times 200 = 1200$ binary bit responses.

From Table 7, it is observed that 1-out-of-r coding performs well after applying (Ln_γ) transformation technique with only one '$P-value$' failure and two marginal *Proportional* failures. The 1-out-of-r coding algorithm completely fails to pass the non-overlapping and partially fails the approximate entropy test. We believe that these failures are because the selection process of the random bits (five-bits challenge) may lead to similar patterns and increase the chance of intrinsic correlation probability in the generated responses. It is expected that an excellent shuffling technique can help to overcome such an inherent correlation behind these selections.

The 1-out-of-r coding passes 13 out of 15 NIST tests with a passing percentage of 93.3% for $P-value$ test and 86.7% for the *Proportional* test. The passing rates are significantly better than another coding such as chain-like neighbor coding and decoupled chain-like neighbor algorithms that are proposed in [8], [9].

VI. CONCLUSION

In this paper, two new security techniques, intra-die frequency aware (IFA) and (Ln_γ), are proposed to enhance the robustness of IoT-based lightweight PUF using configurable ring oscillator PUFs (CF-ROPUFs) design. The CF-ROPUF is realized on real hardware and RO sample frequencies that are collected at varying temperatures and supply voltage from a population of 30 Spartan-3E FPGAs. Experimental results show an average of a 98.5% Hamming Weight (HM) value after applying the IFA technique, which indicates a highly reliable PUF output. Additionally, the CF-ROPUF reliability is defined and evaluated in terms of accuracy, repeatability, and reproducibility, which are the International Organization for Standardization (ISO). The results show that the collected measurements exhibit high accuracy in terms of average trueness and precision of the generated sample RO frequencies. With the application of the IFA and 1-out-of-r coding algorithm, it is also shown that CF-ROPUF has notable average repeatability of 98.2%, and an enhanced average reproducibility of 99.63% as compared to traditional ROPUF design. From the aging result, it is concluded that after applying the proposed IFA technique and the 1-out-of-r coding, the CF-ROPUF design is more immune against accelerated aging impacts with no bit flip that can lead to reliability issues. Further, our results show that the proposed Ln_γ enhances the normality and mitigates the negative impacts of the systematic process variations on RO sample frequencies. Consequently, PUF randomness results show that generated IoT PUF-based binary response bits can successfully pass the 15 NIST test suites for true randomness with an enhanced percentage, 93.3%, with the application of the 1-out-of-r coding.

ACKNOWLEDGMENT

The statements made herein are solely the responsibility of the authors. The authors, therefore, acknowledge with thanks to DSR for technical and financial support.

REFERENCES

[1] F. Amsaad, A. Razaque, M. Baza, S. Kose, S. Bhatia, and G. Srivastava, "An efficient and reliable lightweight PUF for IoT-based applications," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Jun. 2021, pp. 1–6.
[2] A. Sengupta and S. Kundu, "Guest editorial securing IoT hardware: Threat models and reliable, low-power design solutions," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 25, no. 12, pp. 3265–3267, Dec. 2017.
[3] C. Dong, G. He, X. Liu, Y. Yang, and W. Guo, "A multi-layer hardware Trojan protection framework for IoT chips," *IEEE Access*, vol. 7, pp. 23628–23639, 2019.
[4] P. Gope and B. Sikdar, "Lightweight and privacy-preserving two-factor authentication scheme for IoT devices," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 580–589, Feb. 2019.
[5] V. P. Yanambaka, S. P. Mohanty, E. Kougianos, and D. Puthal, "PMsec: Physical unclonable function-based robust and lightweight authentication in the internet of medical things," *IEEE Trans. Consum. Electron.*, vol. 65, no. 3, pp. 388–397, Aug. 2019.
[6] B. Zhao, P. Zhao, and P. Fan, "EPUF: A lightweight double identity verification in IoT," *Tsinghua Sci. Technol.*, vol. 25, no. 5, pp. 625–635, Oct. 2020.

[7] A. Maiti and P. Schaumont, "Improving the quality of a physical unclonable function using configurable ring oscillators," in *Proc. Int. Conf. Field Program. Log. Appl.*, Prague, Czech Republic, Aug. 2009, pp. 703–707.
[8] X. Xin, J. Kaps, and K. Gaj, "A configurable ring-oscillator-based PUF for Xilinx FPGAs," in *Proc. 14th Euromicro Conf. Digit. Syst. Design*, Oulu, Finland, 2011, pp. 651–657.
[9] F. Amsaad, T. Hoque, and M. Niamat, "Analyzing the performance of a configurable ROPUF design controlled by programmable XOR gates," in *Proc. IEEE 58th Int. Midwest Symp. Circuits Syst. (MWSCAS)*, Fort Collins, CO, USA, Aug. 2015, pp. 1–4.
[10] *Technical Corrigendum, Accuracy (Trueness and Precision) of Measurement Methods and Results—Part 1: General Principles and Definitions*, Standard ISO 5725-1:1994, International Organization for Standardization, Geneva, Switzerland, 1998.
[11] *Statistical Interpretation of Data Part 6: Determination of Statistical Tolerance Intervals*, Standard ISO 16269-6:2014, 2014.
[12] (2017). *TestXpo, Reliable Test Results, International Forum for Materials Testing*. [Online]. Available: https://www.testxpo.com
[13] *Information Technology—Vocabulary—Part 37: Biometrics*, document ISO/IEC 2382-37:2017(en). [Online]. Available: https://www.iso.org/obp/ui/#iso:std:66693:en
[14] *Statistical Interpretation of Data Intervals*, Standard ISO 16269-6:2014, 2014. [Online]. Available: https://www.iso.org/obp/ui/#iso:std:iso:16269:-6:ed-2:v1:en
[15] J. Zhang and G. Qu, "Physical unclonable function-based key sharing via machine learning for IoT security," *IEEE Trans. Ind. Electron.*, vol. 67, no. 8, pp. 7025–7033, Aug. 2020.
[16] A. Maiti, V. Gunreddy, and P. Schaumont, "A systematic method to evaluate and compare the performance of physical unclonable functions," in *Embedded Systems Design with FPGAs*, P. Athanas, D. Pnevmatikatos, and N. Sklavos, Eds. New York, NY, USA: Springer, 2013.
[17] L. E. Bassham, A. L. Rukhin, J. Soto, J. R. Nechvatal, M. E. Smid, S. D. Leigh, M. Levenson, M. Vangel, N. A. Heckert, and D. L. Banks, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," NIST, Gaithersburg, MD, USA, Tech. Rep. SP 800-22 Rev 1a, Apr. 2010.
[18] V. Krishnan, "Statistics for the behavioral sciences," in *Cengage Learning*, 9th ed. Boston, MA, USA, Jan. 2012.
[19] G. E. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," in *Proc. 44th ACM/IEEE Design Autom. Conf.*, San Diego, CA, USA, Jun. 2007, pp. 9–14.
[20] J. W. Lee, D. Lim, B. Gassend, G. E. Suh, M. van Dijk, and S. Devadas, "A technique to build a secret key in integrated circuits for identification and authentication applications," in *Symp. VLSI Circuits. Dig. Tech. Papers*, Honolulu, HI, USA, 2004, pp. 176–179.
[21] M. J. Azhar, F. Amsaad, and S. Kose, "Duty-cycle-based controlled physical unclonable function," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 26, no. 9, pp. 1647–1658, Sep. 2018.
[22] N. Pundir, F. Amsaad, M. Choudhury, and M. Niamat, "Novel technique to improve strength of weak arbiter PUF," in *Proc. IEEE 60th Int. Midwest Symp. Circuits Syst. (MWSCAS)*, Boston, MA, USA, Aug. 2017, pp. 1532–1535.
[23] A. Maiti, J. Casarona, L. McHale, and P. Schaumont, "A large scale characterization of RO-PUF," in *Proc. IEEE Int. Symp. Hardw.-Oriented Secur. Trust (HOST)*, Anaheim, CA, USA, Jun. 2010, pp. 94–99.
[24] B. Chatterjee, D. Das, S. Maity, and S. Sen, "RF-PUF: Enhancing IoT security through authentication of wireless nodes using *in-situ* machine learning," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 388–398, Feb. 2019.
[25] Z. Guan, H. Liu, and Y. Qin, "Physical unclonable functions for IoT device authentication," *J. Commun. Inf. Netw.*, vol. 4, no. 4, pp. 44–54, Dec. 2019.
[26] S. Larimian, M. R. Mahmoodi, and D. B. Strukov, "Lightweight integrated design of PUF and TRNG security primitives based on eFlash memory in 55-nm CMOS," *IEEE Trans. Electron Devices*, vol. 67, no. 4, pp. 1586–1592, Apr. 2020.
[27] B. Chen and F. M. J. Willems, "Secret key generation over biased physical unclonable functions with polar codes," *IEEE Internet Things J.*, vol. 6, no. 1, pp. 435–445, Feb. 2019.
[28] F. Amsaad, A. Sherif, A. Dawoud, M. Niamat, and S. Kose, "A novel FPGA-based LFSR PUF design for IoT and smart applications," in *Proc. IEEE Nat. Aerosp. Electron. Conf. (NAECON)*, Dayton, OH, USA, Jul. 2018, pp. 99–104.
[29] A. P. D. Nath, F. Amsaad, M. Choudhury, and M. Niamat, "Hardware-based novel authentication scheme for advanced metering infrastructure," in *Proc. IEEE Nat. Aerosp. Electron. Conf. (NAECON) Ohio Innov. Summit (OIS)*, Dayton, OH, USA, Jul. 2016, pp. 364–371.
[30] D. E. Holcomb, W. P. Burleson, and K. Fu, "Power-up SRAM state as an identifying fingerprint and source of true random numbers," *IEEE Trans. Comput.*, vol. 58, no. 9, pp. 1198–1210, Sep. 2009.
[31] F. Farha, H. Ning, K. Ali, L. Chen, and C. Nugent, "SRAM-PUF-based entities authentication scheme for resource-constrained IoT devices," *IEEE Internet Things J.*, vol. 8, no. 7, pp. 5904–5913, Apr. 2021.
[32] Y. Shifman, A. Miller, O. Keren, Y. Weizman, and J. Shor, "An SRAM-based PUF with a capacitive digital preselection for a 1E-9 key error probability," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 67, no. 12, pp. 4855–4868, Dec. 2020.
[33] M.-K. Oh, S. Lee, Y. Kang, and D. Choi, "Wireless transceiver aided runtime secret key extraction for IoT device security," *IEEE Trans. Consum. Electron.*, vol. 66, no. 1, pp. 11–21, Feb. 2020.
[34] F. Amsaad, C. R. Chaudhuri, and M. Niamat, "Reliable and reproducible PUF based cryptographic keys under varying environmental conditions," in *Proc. IEEE Nat. Aerosp. Electron. Conf. (NAECON) Ohio Innov. Summit (OIS)*, Dayton, OH, USA, Jul. 2016, pp. 468–473.
[35] N. Pundir, F. Amsaad, M. Choudhury, and M. Niamat, "Novel technique to improve strength of weak arbiter PUF," in *Proc. IEEE 60th Int. Midwest Symp. Circuits Syst. (MWSCAS)*, Boston, MA, USA, Aug. 2017.
[36] C. R. Chaudhuri, F. Amsaad, and M. Niamat, "Impact of temporal variations on the performance and reliability of configurable ring oscillator PUF," in *Proc. IEEE Nat. Aerosp. Electron. Conf. (NAECON) Ohio Innov. Summit (OIS)*, Dayton, OH, USA, Jul. 2016, pp. 458–463.
[37] F. Amsaad, A. Prasad, C. Roychaudhuri, and M. Niamat, "A novel security technique to generate truly random and highly reliable reconfigurable ROPUF-based cryptographic keys," in *Proc. IEEE Int. Symp. Hardw. Oriented Secur. Trust (HOST)*, McLean, VA, USA, May 2016, pp. 185–190.
[38] F. Amsaad, M. Niamat, A. Dawoud, and S. Kose, "Reliable delay based algorithm to boost PUF security against modeling attacks," *Information*, vol. 9, no. 9, pp. 1–15, Sep. 2018.
[39] M.-D. Yu, M. Hiller, and S. Devadas, "Maximum-likelihood decoding of device-specific multi-bit symbols for reliable key generation," in *Proc. IEEE Int. Symp. Hardw. Oriented Secur. Trust (HOST)*, Washington, DC, USA, May 2015, pp. 38–43.
[40] M.-D. Yu and S. Devadas, "Secure and robust error correction for physical unclonable functions," *IEEE Des. Test Comput.*, vol. 27, no. 1, pp. 48–65, Jan./Feb. 2010.
[41] C.-E. Yin and G. Qu, "Temperature-aware cooperative ring oscillator PUF," in *Proc. IEEE Int. Workshop Hardw.-Oriented Secur. Trust*, San Francisco, CA, USA, Jul. 2009, pp. 36–42.
[42] J. Shi, Y. Lu, and J. Zhang, "Approximation attacks on strong PUFs," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 39, no. 10, pp. 2138–2151, Oct. 2020.
[43] U. R Ührmair, F. Sehnke, J. S Ölter, G. Dror, S. Devadas, and J. Ü. Schmidhuber, "Modeling attacks on physical unclonable functions," in *Proc. 17th ACM Conf. Comput. Commun. Secur.*, 2010, pp. 237–249.
[44] U. Rührmair, J. Sölter, F. Sehnke, X. Xu, A. Mahmoud, V. Stoyanova, G. Dror, J. Schmidhuber, W. Burleson, and S. Devadas, "PUF modeling attacks on simulated and silicon data," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 11, pp. 1876–1891, Nov. 2013.
[45] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Proc. Annu. Int. Cryptol. Conf.* Santa Barbara, CA, USA: Springer, 1999, pp. 388–397.
[46] P. C. Kocher, "Timing attacks on implementations of Diffie–Hellman, RSA, DSS, and other systems," in *Proc. Annu. Int. Cryptol. Conf.*, 1996, pp. 104–113.
[47] D. Merli, D. Schuster, F. Stumpf, and G. Sigl, "Semi-invasive EM attack on FPGA RO PUFs and countermeasures," in *Proc. Workshop Embedded Syst. Secur.*, 2011, pp. 1–9.
[48] J. Delvaux and I. Verbauwhede, "Side channel modeling attacks on 65 nm arbiter PUFs exploiting CMOS device noise," in *Proc. IEEE Int. Symp. Hardw.-Oriented Secur. Trust (HOST)*, Jun. 2013, pp. 137–142.
[49] S. Tajik, E. Dietz, S. Frohmann, H. Dittrich, D. Nedospasov, C. Helfmeier, J.-P. Seifert, C. Boit, and H.-W. Hubers, "Photonic side-channel analysis of arbiter PUFs," *J. Cryptol.*, vol. 30, no. 2, pp. 550–571, 2017.
[50] A. Mahmoud, U. Rhrmair, M. Majzoobi, and F. Koushanfar, "Combined modeling and side channel attacks on strong PUFs," *IACR Cryptol. ePrint Arch.*, vol. 63, no. 2, Oct. 2013.
[51] Y. Cao, W. Zheng, X. Zhao, and C.-H. Chang, "An energy-efficient current-starved inverter based strong physical unclonable function with enhanced temperature stability," *IEEE Access*, vol. 7, pp. 105287–105297, 2019.
[52] J. Zhang, C. Shen, H. Su, M. T. Arafin, and G. Qu, "Voltage over-scaling-based lightweight authentication for IoT security," *IEEE Trans. Comput.*, early access, Jan. 6, 2021, doi: 10.1109/TC.2021.3049543.

[53] M. T. Arafin, M. Gao, and G. Qu, "VOLtA: Voltage over-scaling based lightweight authentication for IoT applications," in *Proc. Asia South Pacific Design Autom. Conf.*, Jan. 2017, pp. 336–341.
[54] H. Su and J. Zhang, "Machine learning attacks on voltage over-scaling-based lightweight authentication," in *Proc. Asian Hardw. Oriented Secur. Trust Symp.*, Dec. 2018, pp. 50–55.
[55] R. Kumar, H. K. Chandrikakutty, and S. Kundu, "On improving reliability of delay based physically unclonable functions under temperature variations," in *Proc. IEEE Int. Symp. Hardw.-Oriented Secur. Trust*, San Diego, CA, USA, Jun. 2011, pp. 142–147.
[56] B. Srinivasu, P. Vikramkumar, A. Chattopadhyay, and K.-Y. Lam, "CoLPUF: A novel configurable LFSR-based PUF," in *Proc. IEEE Asia Pacific Conf. Circuits Syst. (APCCAS)*, Chengdu, China, Oct. 2018, pp. 358–361.
[57] L. Zhang, C. Wang, W. Liu, M. O'Neill, and F. Lombardi, "XOR gate based low-cost configurable RO PUF," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Baltimore, MD, USA, May 2017, pp. 1–4.
[58] Y. Cui, C. Wang, W. Liu, Y. Yu, M. O'Neill, and F. Lombardi, "Low-cost configurable ring oscillator PUF with improved uniqueness," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Montreal, QC, Canada, May 2016, pp. 558–561.
[59] Y. Cui, C. Gu, C. Wang, M. O'Neill, and W. Liu, "Ultra-lightweight and reconfigurable tristate inverter based physical unclonable function design," *IEEE Access*, vol. 6, pp. 28478–28487, 2018.
[60] Z. Pang, J. Zhang, Q. Zhou, S. Gong, X. Qian, and B. Tang, "Crossover ring oscillator PUF," in *Proc. 18th Int. Symp. Quality Electron. Design (ISQED)*, Santa Clara, CA, USA, Mar. 2017, pp. 237–243.
[61] *Measures of Skewness and Kurtosis*. Accessed: May 2020. [Online]. Available: http://www.itl.nist.gov/div898/handbook/eda/section3/eda35b.htm
[62] T. Tae-Hwan and W. Halbert, "On more robust estimation of skewness and kurtosis: Simulation and application to the SP500 index," *Finance Res. Lett.*, vol. 1, no. 1, pp. 56–73, Mar. 2003.

FATHI AMSAAD (Senior Member, IEEE) received the Ph.D. degree in engineering from The University of Toledo (UT), Toledo, OH, USA, in 2017. He is currently an Assistant Professor with the School of Information Security and Applied Computing (SISAC), Eastern Michigan University (EMU). His research interests include hardware-oriented security and trust for the IoT and smart systems.

AHMED OUN (Graduate Student Member, IEEE) received the M.S. degree in electrical engineering from the University of Bridgeport, Bridgeport, CT, USA, in December 2012. He is currently pursuing the Ph.D. degree with the Electrical Engineering and Computer Science Department, The University of Toledo, Toledo, OH, USA. He also worked as a Project Manager with General Electric International Inc. (GEII) before deciding to pursue his Ph.D. degree. He is also working with the Hardware Oriented Security Laboratory, The University of Toledo. His research interests include hardware-oriented security and trust, testing of digital VLSI, field programmable gate arrays, machine learning algorithms, swarm intelligence optimization techniques, neural networks, and their applications.

MOHAMMED Y. NIAMAT (Life Member, IEEE) received the bachelor's degree in electrical engineering from Aligarh Muslim University, Aligarh, India, the master's degree in electrical engineering from the University of Saskatchewan, Saskatoon, SK, Canada, and the Ph.D. degree from The University of Toledo, Toledo, OH, USA, in 1989. From 1996 to 1997, he was a Visiting Associate Professor with the Center for Reliable Computing, Stanford University. He is currently the Group Leader of the High-Performance Computing Research Group, Electrical Engineering and Computer Science Department, The University of Toledo. He has supervised more than 50 graduate students, including Ahmed Oun.

ABDUL RAZAQUE received the Ph.D. degree in computer science and engineering from the University of Bridgeport, USA, in 2014. He is currently an Associate Professor with the Department of Computer Engineering, International Information Technology University, Almaty, Kazakhstan. His research interests include the wireless sensor networks, cybersecurity, cloud computing security, design and development of mobile learning environments, and ambient intelligence.

SELCUK KOSE (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the University of Rochester, Rochester, NY, USA, in 2012. He is currently an Associate Professor of electrical engineering with the University of Rochester. His research interests include low-power VLSI design and hardware-oriented security.

MOHAMED MAHMOUD (Senior Member, IEEE) received the Ph.D. degree from the University of Waterloo, in April 2011. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, Tennessee Technological University, USA. His research interests include security and privacy preserving schemes for smart grid communication networks, and mobile ad hoc and sensor networks.

WALEED ALASMARY (Senior Member, IEEE) received the Ph.D. degree in electrical and computer engineering from the University of Toronto, Toronto, ON, Canada, in 2015. He is currently an Assistant Professor of computer engineering with Umm Al-Qura University, Saudi Arabia. His current research interests include mobile computing, ubiquitous sensing, intelligent transportation systems, privacy, and anonymity.

FAWAZ ALSOLAMI (Member, IEEE) received the M.A.Sc. degree in electrical and computer engineering from the University of Waterloo, Canada, in 2008, and the Ph.D. degree in computer science from KAUST University, Thuwal, Saudi Arabia, in 2016. He joined the Department of Computer Science, King Abdulaziz University, as an Assistant Professor of computer science. Since 2018, he has been the Chairperson of the Computer Science Department, King Abdulaziz University. He also published many articles and one monograph. His research interests include artificial intelligence, machine learning, data mining, and combinatorial optimization.