C4TERO: Configurable Cascaded Carry Chains for High Reliability TERO PUFs on FPGAs

Fanny Spagnolo, *Senior Member, IEEE*, Massimo Vatalaro, *Member, IEEE*,
Stefania Perri, *Senior Member, IEEE*, Felice Crupi, *Senior Member, IEEE*,
and Pasquale Corsonello, *Senior Member, IEEE*

***Abstract*— In this paper we present a novel Transient Effect Ring Oscillator Physical Unclonable Function for FPGAs. It exploits in an original way the carry chain resources available in modern devices. The basic cell adopted in the proposed architecture can be runtime configured to implement different oscillation paths. This property enables the possibility to output more than one bit response per cell by choosing among the configurations those that exhibit the highest reliability. Such results are achieved by adopting a specific calibration process able to identify configurations of the cells showing the highest stability and the most uncorrelated responses. When implemented on several Series 7 Xilinx devices, no unstable bits were observed at 1 V and 25 °C. Under voltage variation in the manufacturer recommended ranges, a worst case bit error rate of 0.046% is achieved. The circuit designed as here described consists of 64 cells, produces 128 response bits and consumes just 535 look-up-tables and 256 carry chains.**

***Index Terms*— Physical unclonable function (PUF), field programmable gate array (FPGA), carry chain, configurable ring oscillators, high-reliability.**

I. Introduction

THE Internet of Things (IoT) revolution has taken hold in most of our daily activities, radically transforming several sectors like industry [1], healthcare [2], transportation and logistics [3]. The IoT network is made up of billions of remotely connected devices, which pose increasingly demanding technological challenges at both hardware and software design levels. In this context, security issues have received considerable attention because of the large amount of sensitive data exchanged between smart nodes mostly deployed in open environments [4]. As well-established solutions, the protection of network communication between devices can be achieved through encryption-based [5] and authentication-based [6] protocols that, in turn, rely on private keys.

Early smart devices used the secret key permanently stored in nonvolatile memories (NVMs) to grant straightforward access from on board encryption/authentication primitives. However, this design approach is susceptible to probing attacks and other side-channel attacks, thus showing a relatively high level of vulnerability [7]. Moreover, the usage of NVMs has a significant impact in terms of cost, size and energy on the final product, representing another important limitation for the realization of securely connected edge devices.

Thanks to its ability to get the secret key directly from the manufacturing process variations, without requiring any extra device, the Physical Unclonable Function (PUF) has emerged as a promising alternative approach. Basically, a PUF is a hardware primitive that produces at least one hard-to-clone output, named *response*, when it is stimulated by a given input, named *challenge*. Due to the randomness of the manufacturing process variations, a PUF is expected to generate unique and unpredictable responses, corresponding to something like a device fingerprint. To this purpose, it must be designed to assure that the challenge-response pair (CRP) just depends on physical properties of each silicon device that are related to the inherent random non systematic process variations, while rejecting the effect of all other variation contributions [8]. In addition to the unpredictability and uniqueness, a PUF is expected also to show a high level of stability: for a given challenge, it has to output the same response even under different operating conditions and over a relatively long period of time. To reduce its intrinsic instability, in most cases, post-processing techniques are adopted. One of the most common stabilization method is based on complex error correction codes (ECCs). While very effective, this strategy leads to significant hardware overheads. The temporal majority voting (TMV) is much simpler and mostly preferred, but its enhancement of stability level is relatively low, thus it is appropriate for PUF circuits characterized by a limited native instability. The dark bit masking technique, instead, causes significant loss of CRPs [9]. For these reasons, realizing PUFs with highly-stable raw response that require just lightweight post-processing is strongly desired.

Manuscript received 29 April 2024; revised 18 July 2024; accepted 31 July 2024. This work was supported in part by PON Ricerca e Innovazione Ministero dell'Università e della Ricerca under Grant 1062_R24_INNOVAZIONE; in part by the ICSC National Research Centre for High-Performance Computing, Big Data and Quantum Computing; and in part by the Project SERICS (PE00000014) under the MUR National Recovery and Resilience Plan funded by the European Union within the Next Generation EU Program. This article was recommended by Associate Editor M. Ye. *(Corresponding author: Pasquale Corsonello.)*

Fanny Spagnolo, Massimo Vatalaro, Felice Crupi, and Pasquale Corsonello are with the Department of Informatics, Modeling, Electronics and Systems Engineering, University of Calabria, 87036 Rende, Italy (e-mail: f.spagnolo@dimes.unical.it; massimo.vatalaro@unical.it; felice.crupi@unical.it; p.corsonello@unical.it).

Stefania Perri is with the Department of Mechanical, Energy and Management Engineering, University of Calabria, 87036 Rende, Italy (e-mail: s.perri@unical.it).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSI.2024.3438362.

Digital Object Identifier 10.1109/TCSI.2024.3438362

The realization platform in which the PUF architecture is integrated also significantly influences the design choices. Most architectures suitable for Application Specific Integrated Circuits (ASICs) cannot be easily (and sometimes even successfully) replicated on Field Programmable Gate Arrays (FPGAs). In fact, while in the first case the designer has total control over the positioning of the devices, their orientation and the routing of the interconnections on the silicon layout, thus obtaining logical paths with the desired behavior, the FPGA platforms pose some more intricate challenges. The latter mainly arise from the characteristics of the underlying fabric based on Programmable Interconnection Points (PIPs) and configurable tracks. Indeed, this makes difficult to obtain nets having the same nominal delay, as required by PUF circuits relying on the measure of delay differences [10], [11], [12]. In such cases, user's manual place and route (P&R) actions are mandatory to ensure identical routing of different PUF cells, at the cost of increased design efforts.

Nevertheless, the interest for IoT FPGA-based designs is growing up, thanks to their cost-effectiveness, acceleration capabilities and the wide range of covered applications [13]. For this reason, in the recent past, a plenty of works [14], [15], [16], [17], [18], [19], [20], [21], [22], [23], [24] focused on designing efficient PUF circuits suitable to be deployed on such realization platforms. While some proposals [15], [20], [22] aim at reducing the area/energy footprint to fit the requirements of miniaturized devices, some others [14], [19], [21] deal with the design of unified architectures able to operate as PUF or True Random Number Generator (TRNG) by differently configuring the same circuit. Finally, the papers [16], [17], [18], [23], [24] propose original designs to improve uniqueness and stability of PUF response, but at the cost of either area overhead or tedious manual P&R actions.

This paper originates from the observation that, despite the significant effort spent on this topic, to the date, the design space has been only partially explored, and novel solutions that fully exploit the unique hardware resources that modern FPGA devices are equipped with, are highly desirable. Here we present C4TERO: a configurable cascaded carry chain based Transient Effect Ring Oscillator (TERO) PUF cell. It exploits in an unconventional way the carry chain (CC) resources available onto modern FPGAs, usually used to accelerate arithmetic operations. It also takes advantage of the flexibility offered by the CC to runtime configure different oscillation paths for the same cell in order to dramatically reduce the need of additional post-processing for stability correction. This is achieved through the calibration process detailed below, which provides a soft configuration for each PUF cell, choosing the configuration with the highest stability and uncorrelated response from eight possible ones. As significant further advantage, a PUF circuit realized as here described does not require tedious and complex manual designer's interventions at the layout level to balance path delays. At the Golden Key (GK) conditions (i.e., at 1 V and 25 °C), the native results (i.e., without applying the calibration process and other post-processing techniques) extracted by the proposed C4TERO PUF show a Bit Error Rate (BER) of 3.07% along with a BER variation per 10 °C and 0.05 V of 0.335% and 3.15%, respectively. After implementing both the calibration process and a lightweight TMV on 11 readings (TMV11), no unstable bits were observed at GK conditions, while the BER variation per 10 °C and 0.05 V decrease down to 0.017% and 2.09%, respectively. Such results lead to a BER 4 times lower than the state-of-the-art PUFs.

The remainder of this paper is organized as follows. Section II overviews the FPGA-based PUF architectures available in literature. Section III describes the proposed architecture, including the design of the C4TERO cell and the soft-configuration process realized through the proposed calibration methodology. Section IV presents the results of the experimental analysis conducted on the C4TERO PUF with and without calibration, and provides a comprehensive comparison with prior works. Finally, Section V draws the conclusions.

II. Background and Related Works

A PUF circuit aims at producing unpredictable outputs by leveraging manufacturing variations that are intrinsic of the chip on which it is implemented. Typically, the behaviour of a PUF circuit is evaluated from the CRPs collected under several scenarios, including different chips and temperature/voltage operating conditions. The most common performance metrics are *uniqueness*, *stability*, *uniformity* and *randomness*. The uniqueness evaluates how much different is the PUF response produced by different chips that include the same PUF instance when stimulated by the same challenge. The stability determines how efficiently a PUF is able to produce the same response under different operating conditions over a period of time and for a given challenge. Finally, uniformity and randomness provide information on the correlation in the response of PUF instances accommodated within the same device.

Depending on the number of generable CRPs, PUFs can be categorized in *strong* and *weak* [25]. The former are mainly used for authentication purposes and they exhibit a number of CRPs which increases exponentially with the number of physically implemented cells. On the other hand, *weak* PUFs, which are the focus of this work, are typically employed for lightweight encryption, where the key generation on the fly makes it more resilient to memory read-out and data remanence attacks. Besides the target application, the evaluation of such two categories of PUF mainly differs for the performances to be taken into account. As an example, in *strong* PUFs, the large CRP space allows discarding unstable bits without compromising the length of the response, but, at the same time, it increases the possibility to generate correlated responses. On the contrary, the small CRP space of *weak* PUFs results in more stringent reliability requirements, while reducing the correlation issues [8].

As any other integrated circuit, FPGAs are affected by stochastic variations related to both imperfections during the fabrication process and the discrete nature of materials at the nanometer scale [26]. However, the pre-structured architecture of FPGA devices, as well as the pre-determined operating mode of their internal resources, makes some traditional approaches adopted in the ASIC world simply inaccessible [7].

TABLE I
OVERVIEW OF STATE-OF-THE-ART *weak* PUF ARCHITECTURES

	Strength	Weakness
Anderson [18]	Low area	Design fine-tuning based on the target FPGA
SR Latch [14], [15], [20]	Low area	Reduced stability, manual P&R
RO [10], [16], [19] [21], [22], [24] [27]–[29]	Easy to implement	High sensitivity to environmental conditions and external attacks

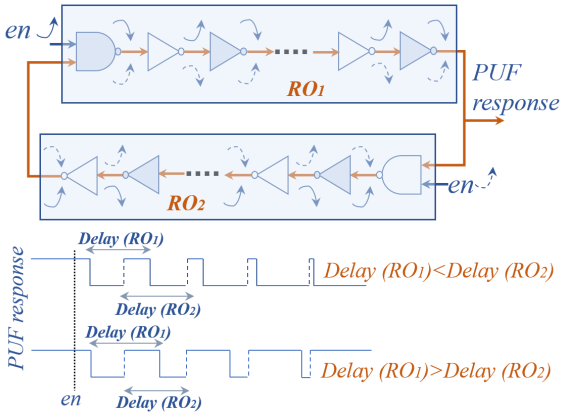


Fig. 1. The TERO cell design [12] and its operating principle.

Table I summarizes the main strength and weakness of most relevant state-of-the-art FPGA-based *weak* PUF architectures. The Anderson PUF [18] and those relying on SR-latches [14], [15], [20] both consume limited amount of chip resources. The former exploits CC as a delay element to propagate possible glitches towards the preset input of a flip-flop, whose output is used as PUF response; it however requires proper design fine-tuning in order to minimize the response bias. SR-latches based PUFs exploit the difference of the path delay between two symmetrical branches to produce a stable response from a metastable cell; they however show a somewhat reduced stability and require manual P&R.

Ring oscillator (RO) PUFs [10], [16], [19], [22], [24], [27], [28], [29] are very popular since they have good uniqueness and reliability and can be easily implemented on FPGAs. This architecture consists of a certain number of ROs having the same odd number of inverting stages. Due to the process variations, a slight difference in the oscillation frequencies is expected. Therefore, the PUF response is derived by measuring such a difference for selected RO pairs through two digital counters and a comparator circuit. Despite of its simplicity, the RO PUF suffers for some security issues [10], and, as shown in [16], the total manufacturing process variation is mitigated by an average effect among the inverting stages that, in turn, makes the PUF response unreliable in case of temperature or voltage migration.

The TERO PUF [12], [17], [23] addresses these drawbacks by extracting entropy from two identical cross-coupled ROs that collapse towards a logic state after a certain number of oscillation cycles, as illustrated in Fig. 1. Opposite to the classical RO, the TERO circuit requires two initialization stages, which are implemented by as many NAND gates receiving the *en* signal, and has two functioning modes: a transient oscillatory state followed by a stable steady state. When the *en* signal is driven high, two events are triggered and propagated across the whole ring structure, giving life to the transient oscillatory regime that results in a periodic signal with variable duty cycle. The latter corresponds to the timing distance between the two events as they propagate across the ring. Then, after a number of transient oscillations depending on the delay difference between the two ROs, the TERO circuit collapses in one of the two possible stable states, as shown in Fig. 1. Therefore, the PUF response is determined by the fastest among the two branches that, if correctly designed, are influenced only by the manufacturing process variations.

While relatively simple from the theoretical point of view, the practical implementation of TERO cells in FPGAs most often requires specific constraints and tedious manual routing operations at the layout level to ensure that the delay due to logic and interconnection contributions is exactly the same in the two branches [17].

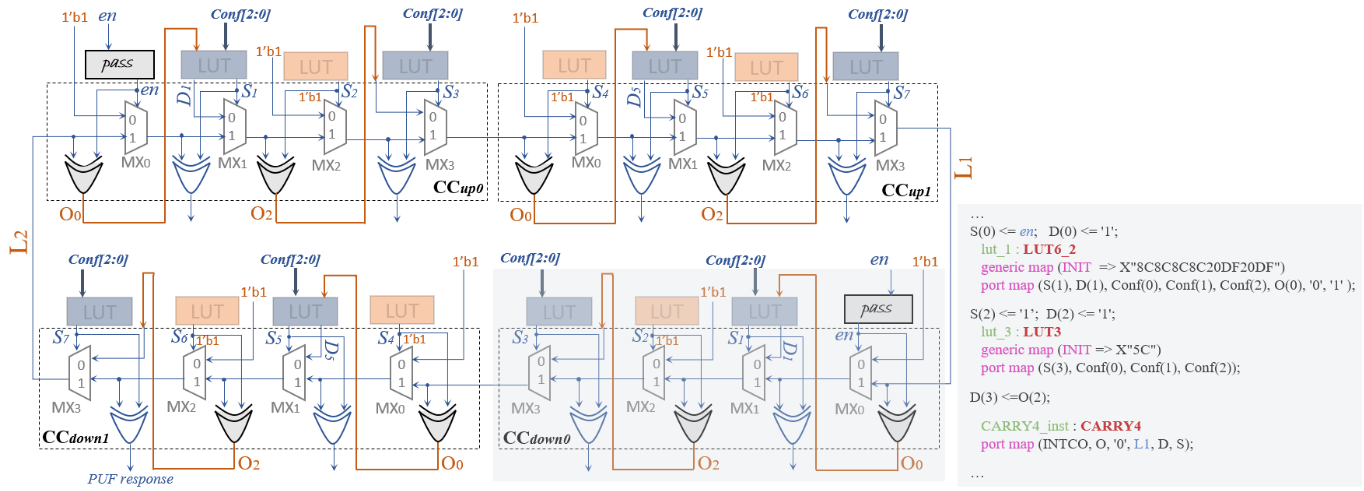
Recently, in order to improve the response stability, a per-device configuration method applied to Anderson PUF cells has been presented in [18] that allows reducing the average BER under temperature variations. This technique relies on tuning each cell in the array by changing the length of a CC-based delay line and the positions of LUTs acting as shift registers. As a drawback, each trial requires the circuit to be re-synthesized and re-implemented in order to produce a new FPGA bitstream. For these reasons, a runtime configuration of the cells, especially useful to mitigate aging effects of the device, is not allowed.

III. The Proposed PUF Architecture

A. The C4TERO Cell

The C4TERO cell proposed here exploits in an unconventional way the CC resources. As it is well known, CCs are usually used to implement basic arithmetic or logical functions and rely on dedicated routing traces, both internally and towards the outermost interface of the slice. The preliminary electrical and thermal characterization provided in [30] demonstrated that such a property can be exploited for the realization of multi-stage ROs. Here, for the first time, we disclose a novel strategy to realize TERO-cell exploiting CC-based oscillators and then its integration in a novel highly configurable TERO PUF architecture.

Fig. 2 illustrates the design of the proposed C4TERO cell. It includes two branches named *up* and *down*, each composed by two cascaded CCs, which end with the crossed loops L_1 and L_2, respectively. In our design, both XOR gates available within the CCs and external LUTs are conveniently utilized to implement the inverting stages. Each CC is settled so that even position multiplexers propagate the signal coming from the previous multiplexer by permanent setting of the selection bit. Whereas, the functioning of odd position multiplexers depends on the 3-bit word *Conf*. The latter enables the soft-configuration of LUTs highlighted in blue, thus allowing to dynamically change the path on which the oscillation is propagated. Moreover, the LUT aligned to the multiplexer



```
...
S(0) <= en;   D(0) <= '1';
  lut_1 : LUT6_2
  generic map (INIT  => X"8C8C8C8C20DF20DF")
  port map (S(1), D(1), Conf(0), Conf(1), Conf(2), O(0), '0', '1' );

S(2) <= '1';  D(2) <= '1';
  lut_3 : LUT3
  generic map (INIT => X"5C")
  port map (S(3), Conf(0), Conf(1), Conf(2));

D(3) <=O(2);

  CARRY4_inst : CARRY4
  port map (INTCO, O, '0', L1, D, S);

...
```

Fig. 2. The proposed C4TERO design. Each RO is designed as two CCs cascaded connected. On each branch, even positions can be used to transfer the XOR gate outputs to the subsequent stage; odd ones exploit the corresponding LUTs to soft-configure the oscillation path by runtime computing the S and D signals through the 3-bit word *Conf*. In the gray box, a sketch of the VHDL description for the CC_{down0} slice is reported.

TABLE II

Possible Configurations for the C4TERO Cell Depending on the *Conf* Word

Conf[2:0]	*up*						*down*					
	S_1	S_3	S_5	S_7	D_1	D_5	S_1	S_3	S_5	S_7	D_1	D_5
000	0	0	1	1	not	pass	0	0	1	1	not	pass
001	0	0	1	1	not	pass	1	1	0	0	pass	not
010	1	1	0	0	pass	not	0	0	1	1	not	pass
011	1	1	0	0	pass	not	1	1	0	0	pass	not
100	0	1	0	1	not	pass	0	1	0	1	not	pass
101	0	0	1	0	pass	not	0	0	1	0	pass	not
110	0	1	1	0	not	pass	0	1	1	0	not	pass
111	1	0	0	1	pass	not	1	0	0	1	pass	not

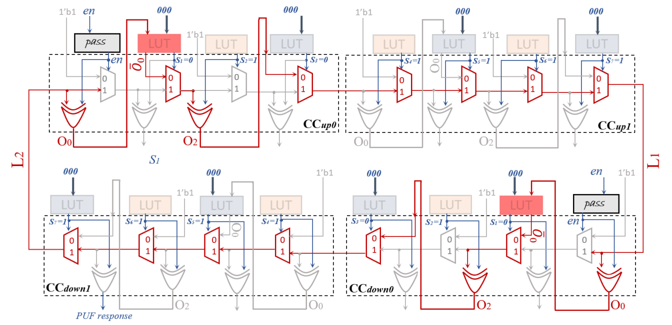


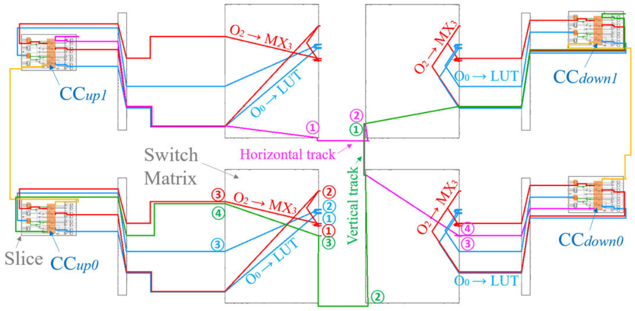
Fig. 3. Oscillation path (red color) enabled in the C4TERO cell when *Conf*=000.

MX_1 (MX_5) in each CC exploits the *Conf* word also to establish if the signal O_0 received by the previous XOR gate has to be inverted or not, thus producing the D_1 (D_5) output.

Table II summarizes the eight possible configurations that can be assumed by the proposed C4TERO cell. Basically, depending on *Conf*, the selector signals S relative to the multiplexers at the odd positions are properly set to configure the oscillation path, while the D_1 and D_5 LUT outputs are computed to be either O_0 (*pass*) or its negated version (*not*). It can be easily verified that, for all configurations, each branch of the proposed C4TERO design actually includes 3 inverting stages. Just as an example, let us focus on the case *Conf*=000, which enables the oscillation path highlighted in red in Fig. 3. According to Table II, this configuration makes symmetric the *up* and *down* branches: after the low-to-high transition of the *en* signal, the loop is closed and the first XOR gate of each branch initiates the oscillation phase. Then, the O_0 signal produced by the CC_{up0}/CC_{down0} block is inverted by the LUT of the subsequent stage, that also sets to zero the selector of the corresponding multiplexer. Therefore, the third inversion is performed by the XOR gate outputting O_2. Finally, the oscillation is transferred to the CC_{up1}/CC_{down1} block, whose multiplexer selectors S_4-S_7 are all set to enable the propagation along the remaining path. The PUF response is retrieved using the last XOR gate of the CC_{down1} block. It is worth noting that such a gate is never used in the oscillation loop, thus the capacitive load due to the readout circuit does not interfere in any way with the TERO loops.

As above discussed, ensuring that the two branches of each C4TERO cell have identical nominal delays is crucial for the correct functioning of the PUF. Fig. 4a illustrates the layout obtained for a C4TERO cell when implemented on a xc7a100tcsg324-1 device, by using the Vivado 2023.2 Development Tool. There, CCs forming the TERO-cell branches and all interconnection segments are explicitly indicated using labels adopted in Fig. 2. To obtain such a layout, a macro is defined for the TERO cell, in order to establish the relative position of all its logic components. Multiple instances of this cell can be easily placed in the desired chip area by constraining just the CC up0 acting as their anchor point. Finally, the routing order described in the caption of Fig. 4 and implemented by automated TCL commands, allows all nets to be routed through perfectly balanced and deterministic paths.

In Fig. 4a, nets highlighted in blue represent the interconnections $O_0 \rightarrow LUT$, whereas those in red transport the signal O_2 to the multiplexer stage MX_3. Since both such nets enter and leave the same slice, they pass only through the corresponding adjacent switch matrix and follow a path that



(a)

O0 → LUT (814 ps)

①	FAN_ALT_5
②	FAN_ALT_7
③	IMUX_L26

O2 → MX3 (582 ps)

①	BYP_ALT6
②	BYP_ALT7
③	BYP_L7

L1 (638 ps)

①	SL1BEG1
②	WL1BEG0
③	BYP_ALT0
④	BYP_L0

L2 (639 ps)

①	EL1BEG0
②	SL1BEG0
③	BYP_ALT0
④	BYP0

(b)

Fig. 4. Post-implementation result obtained for a TERO cell by using the automatic P&R and the routing order (1) $O_0 \rightarrow LUT$ nets; (2) $O_2 \rightarrow MX_3$ nets; (3) L_1 and L_2 nets; (4) *en* net; (5) remaining nets through the command **route_design -preserve**. (a) the layout view; (b) detail of the PIPs and delays for each programmable interconnection.

is pre-determined for most of its length. As a consequence, the only actual PIPs for these nets are three, as detailed in Fig. 4b.

Conversely, the loops L_1 and L_2, highlighted in green and purple respectively in Fig. 4a, connect the CCs in a crossed manner. However, also in this case most of the routing tracks are pre-determined, which is the result of leaving/entering the crossed slices using dedicated pins, i.e. the MX_3 output and the MX_0 input. Therefore, if the suggested routing order is followed, also the vertical/horizontal interconnection segments between the adjacent switch matrices are routed through predictable and perfectly balanced paths without any manual P&R actions.

Post-implementation net delays, estimated through static timing analysis, for the slow corner, are reported in Fig. 4b. They confirm that the proposed C4TERO design is perfectly balanced across the two branches. The same occurs when multiple C4TERO instances are used. The designer can easily verify that all C4TERO cells in the PUF array exhibit the same nominal behaviour by the static timing analysis reports.

In order to extract the stochastic model of the architecture showed in Fig. 2, the delay contributions of each path (i.e., *up* and *down*) must be distinguished into: (1) logic delay referred to the sum of contributions due to logic gates; (2) intra-net delay referred to the sum of contributions associated to the connections within each path; and (3) inter-net delay referred to the contributions associated to the cross-coupled connections L_1 and L_2. Let us suppose that each delay contribution follows a normal distribution and hence it is characterized by a mean value (i.e., μ_{logic}, μ_{intra}, μ_{inter}) and by a standard deviation (i.e., σ_{logic}, σ_{intra}, σ_{inter}). In such a case, each contribution is uncorrelated from the others, and it is possible to express the difference between the delays of the two paths (i.e., ΔT = T_{up}-T_{down}) by a normal distribution, with the mean and standard deviation values defined in (1), (2).

$$\mu_{\Delta T} = \mu_{up} - \mu_{down} = (\mu_{logic,up} - \mu_{logic,down}) + (\mu_{intra,up} - \mu_{intra,down}) + (\mu_{inter,up} - \mu_{inter,down}) \tag{1}$$

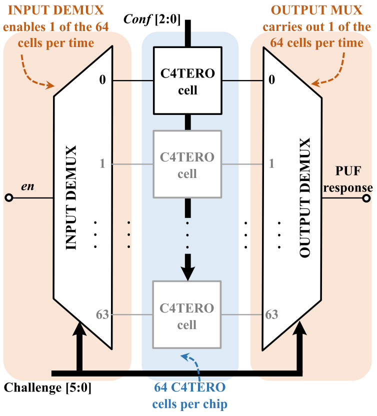


Fig. 5. Array design of the proposed PUF architecture.

$$\sigma_{\Delta T} \simeq \sqrt{\sigma^2_{up} + \sigma^2_{down}} = \sqrt{(\sigma^2_{logic,up} + \sigma^2_{logic,down}) + } \\ \sqrt{+(\sigma^2_{intra,up} + \sigma^2_{intra,down}) + (\sigma^2_{inter,up} + \sigma^2_{inter,down})} \tag{2}$$

Thereby, the output and expected PUF response can be expressed as given in (3), (4). The latter highlight that the PUF output follows a Bernoulli distribution with mean value depending on the expected sign of ΔT. Under the hypothesis of perfectly matched delay between the *up* and *down* paths, $E[PUFResponse]$=0.5, resulting in the highest possible value for an entropy source.

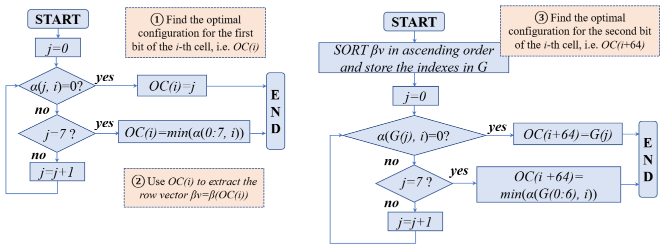
$$PUFResponse = \frac{1 + sign(\Delta T)}{2} \tag{3}$$

$$\begin{aligned} E[PUFResponse] &= E[\frac{1 + sign(\Delta T)}{2}] = \\ &= \frac{1}{2} + \frac{1}{2}E[sign(\Delta T)] \\ &= \frac{1}{2} + \frac{1}{2}erf(\frac{\mu_{\Delta T}}{\sqrt{2}\sigma_{\Delta T}}) \end{aligned} \tag{4}$$

B. Array Design and Calibration Methodology

The proposed PUF architecture is composed by 64 C4TERO cell instances and it is organized as shown in Fig. 5. Basically, the generic i-th cell is enabled by a de-multiplexer when the input challenge i is requested. At the same time, a 64-to-1 multiplexer outputs the PUF response from the corresponding C4TERO instance. All cells in the array also receive the *Conf* word as an input. Such 3-bit information is stored in a LUT-based small memory so that every time a challenge is received, the related chosen configuration is retrieved and sent on the *Conf* bus.

As shown in Section III-A, the soft-configuration enabled by the C4TERO cell allows selecting the oscillation path among eight different combinations. Such a widened design space can be exploited to identify the path leading to the highest possible stability; moreover, since different configurations may exhibit good stability, the proposed design can be also made able to produce more than one bit response per cell. To this aim, the calibration method described in the following was conceived. It relies on the α and β factors defined below,


START
j=0
① Find the optimal configuration for the first bit of the i-th cell, i.e. OC(i)
α(j, i)=0?
yes
OC(i)=j
no
j=7 ?
yes
OC(i)=min(α(0:7, i))
no
j=j+1
② Use OC(i) to extract the row vector βv=β(OC(i))
END
START
③ Find the optimal configuration for the second bit of the i-th cell, i.e. OC(i+64)
SORT βv in ascending order and store the indexes in G
j=0
α(G(j), i)=0?
yes
OC(i+64)=G(j)
no
j=7 ?
yes
OC(i +64)= min(α(G(0:6), i))
no
j=j+1
END

(a) (b)

Fig. 6. The proposed calibration methodology to produce 2 output bits per C4TERO cell with the lowest possible instability and correlation: (a) process for the first bit; (b) process for the second bit.

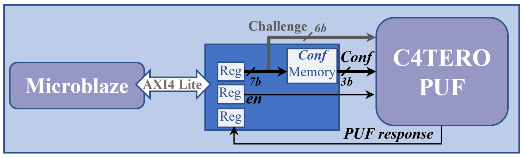


Fig. 7. Block scheme of the C4TERO PUF measurement testbed.

and allows two configurations with the highest stability and uncorrelated responses to be found for each cell in the array. The instability level representing the fraction of bit flippings over a certain number of consecutive evaluations of each cell, in the following, is indicated as the α factor. Therefore, the α factor ranges from 0, in the case of total stability, to 0.5, in the case of maximum instability. It is extracted for all the 8 possible configurations in Table II and by using two different waiting times before activating the cell readout process. In particular, by exploiting the property of TERO cells for which the lowest the time-to-collapse the higher the stability, a shorter waiting time is used to detect marginally unstable cells, whose response could vary as a consequence of environmental conditions changes. At the end, two 8×64 matrices are produced, one for each adopted waiting time during the readout. Finally, the generic (j, i) element of α factor matrix is obtained by considering the worst-case stability scenario between the two matrices at the homologous locations.

The β factor, instead, indirectly measures the similarity between each one of the 8 possible configurations of the C4TERO cell by considering the percentage of shared circuit elements involved in the oscillation paths. Such similarity causes a certain correlation level in the response produced by different configurations. Therefore, to make the array of 64 C4TERO cells able to output 128 response bits, two configurations with optimum (α, β) setting has to be identified. Since such a parameter is related to how the selectors of the multiplexers MXs are configured, the generic (t, v) element of β factor matrix is extracted as the XNOR Popcount operation calculated between the *up* and *bottom* selectors of the configurations $Conf{=}t$ and $Conf{=}v$, normalized to the number of compared selectors (i.e. 14).

The flow charts in Fig. 6 summarize the steps performed to calibrate the proposed C4TERO PUF. This process identifies two optimal configurations (*OCs*) for the generic i-th cell in the array. In the following, we refer to those configurations as $OC(i)$ and $OC(i{+}64)$. First of all, as shown in Fig. 6a, the instability of all configurations is evaluated scanning the α matrix: the first configuration that exhibits zero instability, for the generic *i-th* position, is assigned to $OC(i)$; if no stable configurations are found at the end of this evaluation, the one showing the minimum α value is used. The second optimal configuration that can be used to produce a further response bit from the same physical cell, that is $OC(i{+}64)$, is chosen following the steps illustrated in Fig. 6b. First, the βv row vector, corresponding to the $OC(i)$-th row of the β matrix, is extracted to analyze the correlation between the already assigned $OC(i)$ configuration and the remaining ones. Based on the sorted version of such a vector, which reports the possible configurations with an ascending order according to their correlation, the α matrix is scanned to assign to $OC(i{+}64)$ the configuration having the minimum instability and correlation with $OC(i)$ (Fig. 6b).

It is worth noting that, even though based on TERO cells, the proposed C4TERO PUF has some characteristics that should make it less susceptible to side-channel attacks. The main critical components from this point of view [31] are indeed either avoided or minimized in our architecture by: (1) avoiding the usage of counters, in contrast to the original TERO PUF architecture [17]; (2) limiting the number of oscillating cells to just one at a time; (3) selecting, for each cell, the configuration with the lowest time-to-collapse, thus reducing the oscillation time.

IV. Measurement Results

This Section presents a comprehensive hardware characterization of the proposed C4TERO PUF. For the purpose of a general analysis, the 64 cells array has been included within the on-chip system depicted in Fig. 7. The optimal 128×3-bits soft-configuration identified by the proposed calibration process is stored within the *Conf* memory, which is realized through just 6 LUTs operating as distributed RAM blocks. It is worth noting that this solution allows facilitating the PUF re-calibration in case of effects due to chip aging, with respect to the usage of permanent data to represent the configuration selection. The soft-core Microblaze processor is used to orchestrate the test measurement procedure through AXI4-Lite transactions. First, it sends a 7-bit word to the corresponding configuration register. Such an information, which accommodates the challenge value on the 6 less significant bits, is used as address to read the 128-bit *Conf* memory. Once the correct soft-configuration has been transferred to the array, the Microblaze generates the *en* signal needed to activate the requested C4TERO cell. Finally, the PUF response is stored within a register to be accessed by the Microblaze.

As shown in Fig. 8, the test setup also consists of a mixed signal oscilloscope Tektronix series 6 MSO64 (2.5 GHz), used to electrically monitoring the PUF response and to evaluate the time-to-collapse of each cell, and a Precision DC Supply Keithley 2280S-60-3 that power supplies the devices under test while also precisely monitoring the drawn current.

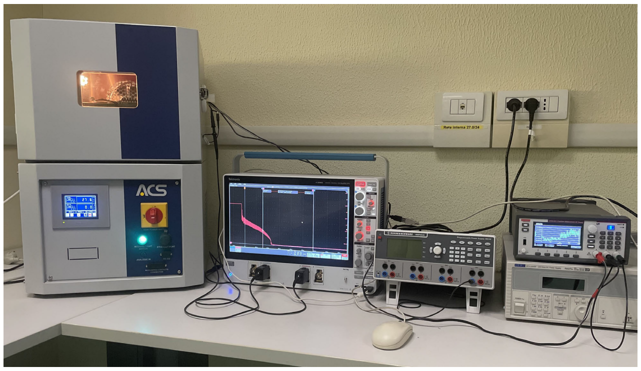


Fig. 8. The adopted experimental measurement setup.

A. Measurements at GK Conditions and Under Temperature Variations

The thermal analysis was conducted using the ACS DY16-T climatic chamber, varying the temperature from 5°C to 75°C. The acquisition of the PUF outputs was always performed after the thermal transient was concluded. To this purpose, the die temperature was monitored through the internal sensor and the Precision Measurement DC Supply that also allowed verifying that the standard deviation of the absorbed current is below 10nA.

Fig. 9 shows the measurement results of the 64 PUF instances at GK conditions (V_{DD} = 1.0 V and T = 25 °C) across ten Digilent Nexys4 DDR board equipped with Artix-7 xc7a100tcsg324-1 commercial devices. The number of devices was limited due to the limited availability. Fig. 9a provides the speckle diagram of the eight possible configurations (*000* bottom, *111* top) of each chip. Detailed results like the proportion of 0's and 1's and the presence of repetitive patterns in the array will be demonstrated in the following through proper uniformity and randomness evaluations. Anyway, by averaging the speckle diagram over the chip dimension we can state that 62/64 cells have 5 or more configurations whose proportion of 1's is in the range [0.3-0.7]; the number of configurations verifying this condition decreases to 4 and 3 for the remaining two cells, which, however, is enough to conclude the absence of heavily biased positions. The α factor was extracted by performing 500 consecutive evaluations of each cell in the array, marking as unstable each position that flips at least once under the different readout operations due to the on-chip noise [32]. In particular, a waiting time of 6 ms was used as nominal time for the readout process, while a waiting time of 0.1 ms was adopted for detecting the marginally unstable cells. Such a process produces the α factor reported in Fig. 9b. The latter highlights that, for each chip and cell within the array, the proposed calibration method is able to identify at least one stable configuration. Fig. 9c reports the β factor, as defined in Section III-B. From this plot, it can be observed that the configuration *011* is the least correlated to the configuration *000*. Therefore, they can be used to generate the native 128-bit response, which corresponds to the output obtained from the C4TERO PUF without applying the calibration procedure discussed in Section III-B.

Fig. 10a shows the unstable bit ratio (UBR) as the percentage of unstable bits averaged over ten chips under temperature variations (5)-75 °C) before and after implementing stability enhancement techniques. In particular, red columns refer to native data, while blue and green columns refer to data achieved after the calibration process, without and with applying the TMV11 operation, respectively. From such results it can be observed that the percentage of native unstable bits is 18.91% at GK conditions (1.0 V and 25 °C). It increases up to just 19.06% and 21.02% at 5 °C and 75 °C, respectively. This demonstrates the low sensitivity of the proposed solution to the temperature variations. After implementing the calibration process at GK conditions, this percentage is reduced of 242× at 25°C and of 27.2× and 5.7× at 5°C and 75 °C, respectively. When considering both the calibration process and the TMV11, no unstable bits were observed in the range of 5-50 °C (i.e., the observed UBR is lower than $7.81\times10^{-2}\%$ which is the minimum observable UBR value for the adopted statistical set). On the other hand, at 75 °C, we experienced the presence of one unstable bit that leads the UBR to rise up to $7.81\times10^{-2}\%$. Fig. 10b illustrates the BER histogram representing the average of the simultaneous instability exhibited by the PUF output word [8]. The proposed PUF architecture shows a native BER of 3.07% at 25°C which increases up to 3.74% and 4.43% at 5 °C and 75 °C, respectively. After the calibration process, the BER is reduced to 2.81×10^{-4} at 25°C, thus leading to a 10,925.3× improvement; moreover, the observed BER at 5°C and 75 °C is 5.19×10^{-3} and 1.34×10^{-1}, corresponding to an improving of 720.6× and 33.1×, respectively, over the native solution. This proves the effectiveness of the proposed calibration process in detecting and re-configuring the potential unstable cells. Finally, the condition of no unstable bits, observed in the range (5)-50 °C) when both calibration process and TMV11 are applied, results in a BER lower than $1.56\times10^{-4}\%$. The only one unstable bit observed at 75 °C leads to a BER increase up to $4.13\times10^{-2}\%$.

B. Uniqueness, Reliability and Randomness Analysis

For such an analysis, we tested the PUF by 10,000 challenges and evaluated N_{bit}=32 output bits extracted from the 128-bit array. The resulting inter-device and intra-PUF Hamming Distance (HD) as defined in (5) and (6) have been then calculated, considering N_{chips} = 10.

$$HD_{inter} = \frac{2}{N_{chip}(N_{chip}-1)} \sum_{i=0}^{N_{chip}-1} \sum_{j=i+1}^{N_{chip}} \frac{HD(R_i, R_j)}{N_{bit}} \tag{5}$$

$$HD_{intra} = \frac{1}{N_{cond}N_{chip}} \sum_{i=1}^{N_{cond}} \sum_{j=1}^{N_{chip}} \frac{HD(R_j^{GK}, R_j^i)}{N_{bit}} \tag{6}$$

The inter-device distance (5), also referred to as inter-PUF when only one PUF instance is accommodated within a chip, measures the variation between the R_i and R_j N_{bit}-sized native responses obtained from the chips i and j (with i ≠ j) under the same challenge at GK conditions (1.0 V and 25 °C),

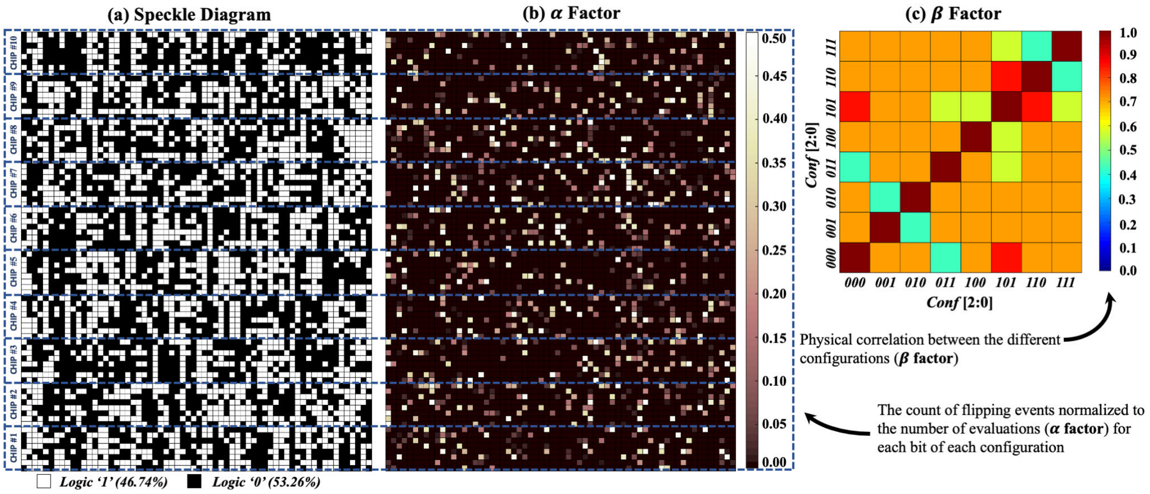


Fig. 9. Measurements of the eight possible configurations of the 64 PUF instances across ten xc7a100tcsg324-1 FPGA chips at GK conditions (V_{DD} = 1.0 V and T = 25 °C). (a) Speckle diagram shows a global average proportion of 46.74% 1's; (b) the α factor (i.e. the count of flipping events normalized to the number of evaluations); (c) the β factor (i.e. the physical correlation between all the possible configurations).

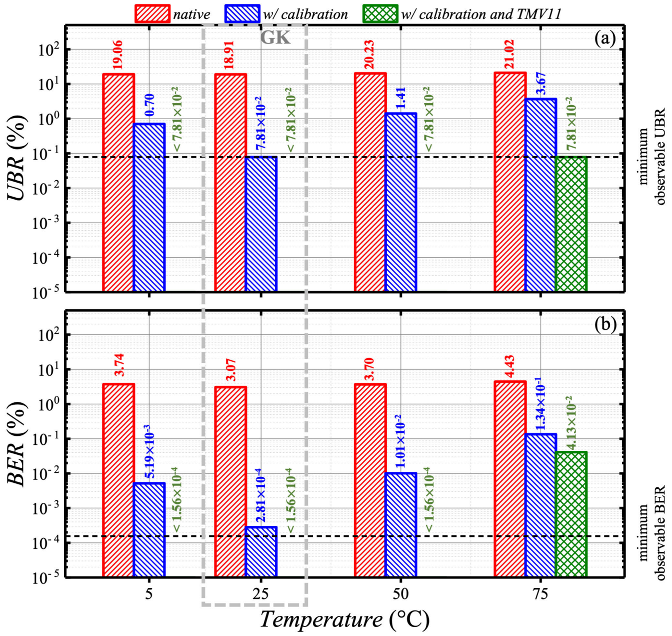


Fig. 10. (a) Percentage of the unstable bit ratio; (b) BER averaged over ten xc7a100tcsg324-1 FPGA chips for the PUF array under temperature variations (5)-75 °C). The representations <7.81×10^{-2}% and <1.56×10^{-4}% indicate the minimum observable UBR and BER values for the adopted statistical set.

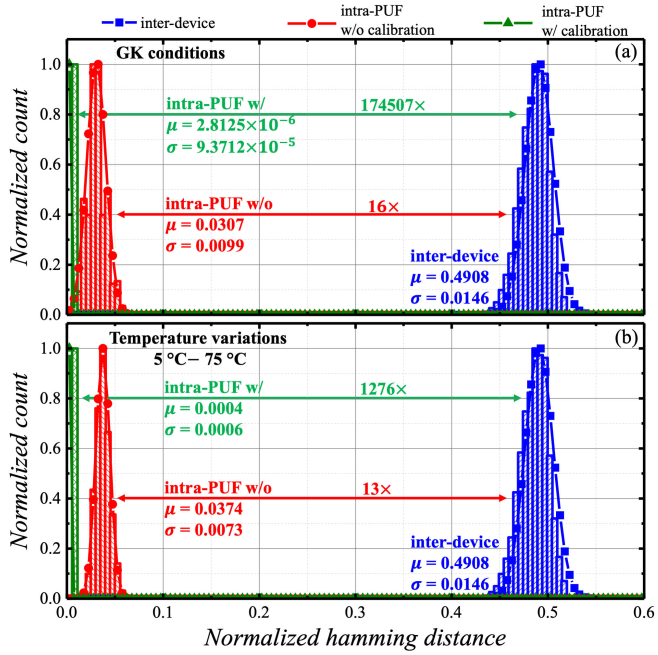


Fig. 11. Normalized inter-device and intra-PUF Hamming Distance (HD), averaged over 10 dice at (a) GK conditions (1.0 V and 25 °C) and (b) under temperature variations.

and allows calculating the *uniqueness* [7]. On the other hand, the intra-PUF HD (6) considers the N_{bit}-sized PUF response R_j produced by the chip j and evaluates its variation due to the on chip noise (at GK conditions) and variations in the environmental conditions (when moving from the GK case, i.e. R_j^{GK}, to other VT conditions, i.e. R_j^i with $1 < i < N_{cond}$). Such an information can be exploited to compute the *reliability* [7].

Fig. 11 plots the distribution of the inter-device and intra-PUF HD obtained by 10 k CRPs, where each bit of the 32 response bits was randomly extracted from the 500 evaluations of each cell. In particular, Fig. 11a provides statistical information on the uniqueness and the reliability at the GK conditions (considering only the effect of the on-chip noise), showing a mean value of the inter-device HD for the native response of 0.4908, which is very close to the ideal value of 0.5. Such a result was confirmed also by the evaluation of the post-calibration response. Even more interesting, the mean value of the intra-PUF HD at GK conditions before and after the calibration process is 0.0307 and 2.8125×10^{-6}, respectively. This leads to a corresponding identifiability (i.e., the ratio between inter- and intra-PUF HD) of 16× and 174507×. In Fig. 11b, the effects of both on-chip noise and different temperature conditions are taken into account for the intra-PUF HD. There, it can be noted that the mean value of the intra-PUF HD before and after the calibration process

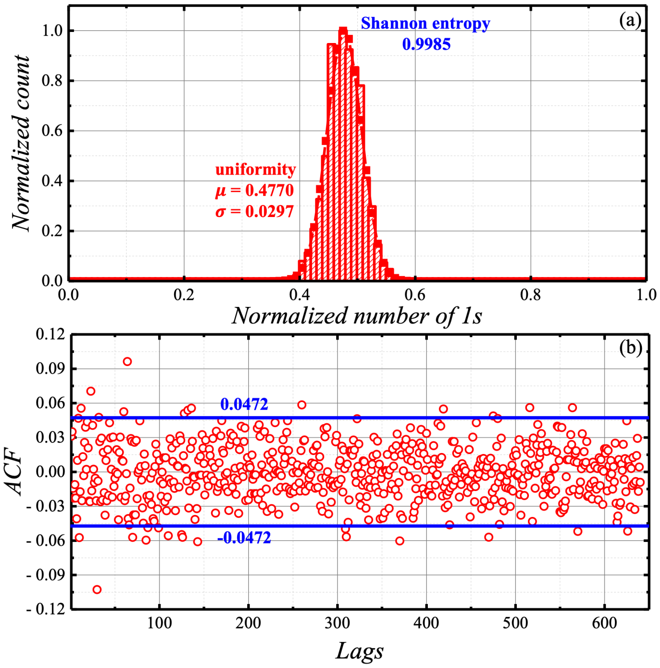


Fig. 12. (a) Normalized number of bit '1' at GK conditions, and (b) spatial auto-correlation function (ACF).

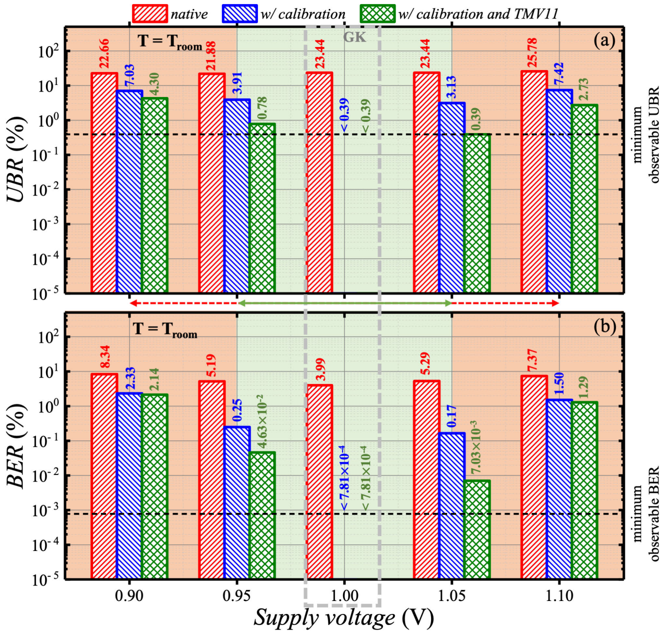


Fig. 13. UBR (a) and BER (b) averaged over two xc7z045ffg900-2 FPGA chips for the PUF array under voltage variations (0.90-1.10 V).

TABLE III
NIST Test Results (Average Over 10 Chips)

	Stream length (n)	Average of p-value	Pass Rate [%]
Frequency	128	0.4765	100
Block frequency	128	0.4765	100
Runs	128	0.6897	100
Longest runs of ones	128	0.5725	100
FFT	128	0.5281	100
Non-overlapping template	128 (m=4)	0.5159	100
Serial	128 (m=4)	0.5431	100
Cumulative sum	128	0.4771	100

is, respectively, of 0.0374 and 0.0004, thus leading to an identifiability of, 13× and 1276×, respectively.

Fig. 12 illustrates the *uniformity* and the auto-correlation function (ACF) exhibited by the C4TERO PUF before the calibration. From Fig.12a, it can be observed that the number of 1s normalized to the PUF word length is equal to 0.4770, thus leading to a Shannon Entropy of 0.9985, which is very close to the ideal value of 1 [8]. Furthermore, the plot in Fig. 12b shows that the spatial ACF, evaluated on ten chips, at 95% confidence bounds is 0.0472, thus proving a low spatial correlation between neighboring cells. To complete the characterization, we also analysed results achieved by the proposed PUF after the calibration. In such a case, the mean value of the *uniformity* and the ACF range are 0.4829 and ±0.048, respectively, which confirms that the proposed calibration method has mainly impact on the PUF reliability performance.

The *randomness* achieved by the proposed architecture has been assessed by performing statistical NIST test [34]. Table III provides such results and demonstrates that all evaluated chips pass all implemented NIST tests, which indicates a good randomness. Furthermore, we adopted the approach demonstrated in [33] to assess the predictability of the proposed PUF through joint entropy measurements over several chips ($H_{p,c}$) and several challenges ($H_{p,d}$), achieving 90.75% and 99.20%, respectively. Such results are perfectly in line with the desired behavior. Finally, the resilience of the proposed PUF to machine learning attacks has been also evaluated by means of a trained Binary GLM Logistic Regression classifier. Results show that the prediction accuracy is limited to 52%, that is very close to the 50% target value.

C. Measurements Under Voltage Variations

To characterize the C4TERO PUF under voltage variations, a range of 0.90-1.1 V supply voltage, corresponding to the nominal V_{DD} ± 10%, was considered. It is worth noting that this represents a quite wide voltage range, since the recommended voltage range for the device under test is 0.95-1.05 V. Fig. 13 provides UBR and BER measured during these tests performed on two AMD ZC702 boards equipped with xc7z045ffg900-2 devices at room temperature. In particular, red bars refer to native data, while blue and green bars refer to data achieved after the calibration process, without and with the implementation of TMV11 process, respectively. Fig. 13a shows that the percentage of native unstable bits is 23.44% at GK conditions, and it becomes 22.66% and 25.78% at 0.90 V and 1.10 V, respectively. After implementing the calibration process, no unstable bits were observed at nominal voltage. Furthermore, by using the calibration process and the TMV11, the UBR is reduced to 4.30% and 2.73% at 0.90 V and 1.10 V, respectively, with up to 5.27× and 9.44× improvements.

A similar behavior has been obtained for the BER and it is reported in Fig. 13b. The adopted calibration process reduces the BER by up to 3.58× (3.90×) at 0.90 V and 4.91× (5.71×) at 1.10 V, with respect to the native results, without (with) the TMV11. This outcome highlights a higher

TABLE IV

COMPARISON AGAINST PRIOR WORKS BASED ON WEAK PUFs

	Proposed	[14]	[15]	[16]	[17]	[18]	[19]	[20]	[21]	[27]
Architecture	C4TERO	DD	XOR	FRO	TERO	ANDER.	RO	NAND	RO	RO
Platform	Artix-7	Artix-7	Artix-7	Virtex-7	Spartan-6	Virtex-7	Spartan-6	Artix-7	Spartan-3E	Artix-7
Slice/bit	2.0	0.5	0.5	3.0	4.0	2.0	21.0	0.5	N/A	0.417
Uniqueness [%]	49.08	49.48	49.47	48.74	48.46	46.25	50.21	49.50	50.17[2]	48.91
Reliability [%]	>99.99	98.33	98.94	98.91	97.37	97.53	96.98	98.62	98.70[1,3]	99.39
BER [%]	$<1.56\times10^{-4}$	1.67	1.06	1.09	2.63	2.47	3.02	1.38	1.30[1,3]	0.61
Voltage range [V]	0.9-1.1	0.9-1.1	0.9-1.1	N/A	1.1-1.3	N/A	N/A	1.1-1.3	0.96-1.44	0.95-1.05
BER var. per 0.05V [%]	2.09	5.24[1]	3.88[1]	N/A	20.25[1]	N/A	4.87[1]	0.125[1,3]	N/A	N/A
Temp. range [°C]	5-75	0-80	N/A	N/A	-15-65	10-50	N/A	5-80	25-65	0-85
BER var. per 10°C [%]	0.017	0.29[1]	N/A	N/A	7.00[1]	3.38	N/A	1.90[1]	0.07[1,3]	0.16
Worst BER V/T [%]	2.14 (0.046) @0.9V (0.95V)	9.23 @0.9V	6.25 @1.1V	3.4 @N/A	14.91[1] @-15°C	11.24 @50°C	N/A	6.7 @0.81V	1.63[1,3] @0.96V	2.9 @85°C, 1.05V
Stabilization technique	Calibration + TMV11	-	-	-	-	Per-device configuration	-	-	Calibration	TMV

[1] extracted from graphs;
[2] extracted considering 511 RO pairs;
[3] extracted considering 63 RO pairs;

sensitivity to the voltage variations, compared to that observed under temperature variations. However, the effectiveness of the proposed stabilization process is demonstrated also outside the manufacturer recommended voltage range. Referring to the latter, the proposed solution achieves a BER of 0.25% (4.63×10^{-2}%) and 0.17% (7.03×10^{-3}%) at 0.95 V and 1.05 V, respectively, without (with) the TMV11.

D. Comparison With Prior Works

Results obtained by comparing the proposed architecture to some of the most relevant works based on weak PUF solutions are provided in Table IV. There, the FPGA platform used in each work, the number of slices consumed for each bit cell of the PUF array and the adopted stabilization technique, if any, are also indicated. It is worth noting that, differently from techniques used in [18] and [21], the calibration approach used here relies on the soft-configuration of the cell, thus allowing a much higher flexibility to be reached. At a glance, among the compared architectures, the proposed solution exhibits the lowest BER and the highest *reliability* at the GK conditions. Such a result is achieved without compromising the area efficiency of the C4TERO cell that just requires 2 slices per response bit. The adopted stabilization technique also effectively improves the reliability under VT variations. In such a case, the BER variation per 10 °C exhibited by the proposed PUF is 9.4 and 4 times lower than the best competitors [21] and [27], respectively.

Further experiments have been performed on the proposed PUF under simultaneous voltage and temperature variations. We found that, referring to the voltage and temperature recommended ranges, the worst case scenario occurs at 1.05 V and 75 °C. In this case, we obtained a BER (UBR) of 6.81% (27.35%), 1.1% (8.21%) and 0.7% (2.35%) for the native response, with calibration and with calibration and TMV11, respectively.

Table V reports the area occupancy of the proposed PUF array, considering the architecture shown in Fig. 5, the LUT-based memory storing the configuration bits and the overhead due to the TMV11 module. In this case, the power consumption is about 9 mW. The amount of consumed LUTs, FFs and CCs are compared with those required by [18] and [21] that, at the parity of number of bits generated, i.e. 128, are the only competitor works that furnish such information for the whole PUF architecture. The PUF design [18] (w/o ECC) occupies an amount of LUTs comparable to the C4TERO design, while using 765 and 254 more FFs and CCs. However, in order to achieve *reliability* and BER comparable with those obtained in this work, it requires an ECC unit costing of additional 1689 LUTs and 1924 FFs. On the other hand, the adopted post-processing TMV11 module requires just 6 LUTs and 10 FFs. Finally, with respect to [21], the proposed PUF utilizes 87.6% less LUTs, thus confirming its superiority to achieve the highest stability with the lowest area footprint.

TABLE V

RESOURCE REQUIREMENTS FOR THE PUF ARRAY AND COMPARISON WITH PRIOR WORKS

	LUTs	FFs	CCs
[18] (w/o ECC)	510	765	510
[18] (w ECC)	2199	2689	>510
[21]	4325	122	N/A
Proposed (w/o TMV11)	535	0	256
Proposed (w TMV11)	541	10	256

V. CONCLUSION

In this paper, we have presented a novel strategy to design PUF architectures suitable for implementation on modern FPGA devices. The main contributions of this work include:

- the design of a novel TERO cell that exploits the CC primitives in order to implement the cross-coupled ring oscillator branches. In contrast to state-of-the-art, the proposed implementation does not require manual P&R designer's interventions to balance path delays.
- the realization of a PUF architecture with runtime configuration capabilities, which enable to dynamically change the path on which the oscillation is propagated and

to configure the same cell as different circuits. This allows dramatically reducing the complexity of additional post-processing for stability correction while increasing the number of response bits per cell.

- a specific calibration methodology that provides a soft configuration for each cell to achieve the highest stability and uncorrelated response from eight possible ones. At the GK conditions, the post-calibration response extracted by the proposed C4TERO PUF reduces the BER by 4 orders of magnitude with respect to the native response.

Results obtained on several Xilinx Series 7 devices demonstrate that the C4TERO + TMV11 implementation produces a 128-bit response with no unstable bits at the GK conditions; moreover, its worst V/T BER is up to 7 times lower than state-of-the-art competitors. Such a gain is achieved with a significant reduction of the area footprint for the whole PUF architecture.

References

[1] B. Babayigit and M. Abubaker, “Industrial Internet of Things: A review of improvements over traditional SCADA systems for industrial automation,” *IEEE Syst. J.*, vol. 18, no. 1, pp. 120–133, May 2023, doi: 10.1109/JSYST.2023.3270620.

[2] A. Rejeb et al., “The Internet of Things (IoT) in healthcare: Taking stock and moving forward,” *Internet Things*, vol. 22, Jul. 2023, Art. no. 100721.

[3] N. Sharma and R. D. Garg, “Real-time IoT-based connected vehicle infrastructure for intelligent transportation safety,” *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 8, pp. 8339–8347, Jun. 2023.

[4] M. Mahamat, G. Jaber, and A. Bouabdallah, “Achieving efficient energy-aware security in IoT networks: A survey of recent solutions and research challenges,” *Wireless Netw.*, vol. 29, no. 2, pp. 787–808, Feb. 2023.

[5] R. Praveen and P. Pabitha, “Improved Gentry–Halevi’s fully homomorphic encryption-based lightweight privacy preserving scheme for securing medical Internet of Things,” *Trans. Emerg. Telecommun. Technol.*, vol. 34, no. 4, pp. 1–22, Apr. 2023.

[6] J. Plusquellic, E. E. Tsiropoulou, and C. Minwalla, “Privacy-preserving authentication protocols for IoT devices using the SiRF PUF,” *IEEE Trans. Emerg. Topics Comput.*, vol. 11, no. 4, pp. 918–933, Oct. 2023.

[7] N. N. Anandakumar, M. S. Hashmi, and M. Tehranipoor, “FPGA-based physical unclonable functions: A comprehensive overview of theory and architectures,” *Integration*, vol. 81, pp. 175–194, Nov. 2021.

[8] M. Alioto, “Trends in hardware security: From basics to ASICs,” *IEEE Solid State Circuits Mag.*, vol. 11, no. 3, pp. 56–74, Summer. 2019.

[9] S. Taneja and M. Alioto, “PUF-based key generation with design margin reduction via in-situ and PVT sensor fusion,” in *Proc. IEEE 45th Eur. Solid State Circuits Conf. (ESSCIRC)*, Cracow, Poland, Sep. 2019, pp. 61–64.

[10] U. Mureddu, B. Colombier, N. Bochard, L. Bossuet, and V. Fischer, “Transient effect ring oscillators leak too,” in *Proc. IEEE Comput. Soc. Annu. Symp. VLSI (ISVLSI)*, Miami, FL, USA, Jul. 2019, pp. 37–42.

[11] L. Tebelmann, J.-L. Danger, and M. Pehl, “Interleaved challenge loop PUF: A highly side-channel protected oscillator-based PUF,” *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 69, no. 12, pp. 5121–5134, Dec. 2022.

[12] L. Bossuet, X. T. Ngo, Z. Cherif, and V. Fischer, “A PUF based on a transient effect ring oscillator and insensitive to locking phenomenon,” *IEEE Trans. Emerg. Topics Comput.*, vol. 2, no. 1, pp. 30–36, Mar. 2014.

[13] A. Magyari and Y. Chen, “Review of state-of-the-art FPGA applications in IoT networks,” *Sensors*, vol. 22, no. 19, p. 7496, Oct. 2022.

[14] R. D. Sala and G. Scotti, “Exploiting the DD-cell as an ultra-compact entropy source for an FPGA-based re-configurable PUF-TRNG architecture,” *IEEE Access*, vol. 11, pp. 86178–86195, 2023.

[15] R. Della Sala, D. Bellizia, and G. Scotti, “A lightweight FPGA compatible weak-PUF primitive based on XOR gates,” *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 69, no. 6, pp. 2972–2976, Jun. 2022.

[16] Z. Huang, J. Bian, Y. Lin, H. Liang, and T. Ni, “Design guidelines and feedback structure of ring oscillator PUF for performance improvement,” *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 43, no. 1, pp. 71–84, Jan. 2024.

[17] C. Marchand, L. Bossuet, U. Mureddu, N. Bochard, A. Cherkaoui, and V. Fischer, “Implementation and characterization of a physical unclonable function for IoT: A case study with the TERO-PUF,” *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 37, no. 1, pp. 97–109, Jan. 2018.

[18] M. A. Usmani, S. Keshavarz, E. Matthews, L. Shannon, R. Tessier, and D. E. Holcomb, “Efficient PUF-based key generation in FPGAs using per-device configuration,” *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 27, no. 2, pp. 364–375, Feb. 2019.

[19] Y. Wang et al., “A reconfigurable PUF structure with dual working modes based on entropy separation model,” *Microelectron. J.*, vol. 124, Jun. 2022, Art. no. 105445.

[20] R. Della Sala and G. Scotti, “A novel FPGA implementation of the NAND-PUF with minimal resource usage and high reliability,” *Cryptography*, vol. 7, no. 2, p. 18, Apr. 2023.

[21] I. Baturone, R. Román, and Á. Corbacho, “A unified multibit PUF and TRNG based on ring oscillators for secure IoT devices,” *IEEE Internet Things J.*, vol. 10, no. 7, pp. 6182–6192, Apr. 2023.

[22] C. Gu, C. H. Chang, W. Liu, N. Hanley, J. Miskelly, and M. O’Neill, “A large scale comprehensive evaluation of single-slice ring oscillator and PicoPUF bit cells on 28nm Xilinx FPGAs,” in *Proc. 3rd ACM Workshop Attacks Solutions Hardw. Secur. Workshop*, Nov. 2019, pp. 101–106.

[23] M. Varchola, M. Drutarovsky, and V. Fischer, “New universal element with integrated PUF and TRNG capability,” in *Proc. Int. Conf. Reconfigurable Comput. FPGAs (ReConFig)*, Dec. 2013, pp. 1–6.

[24] Z. Wei, Y. Cui, Y. Chen, C. Wang, C. Gu, and W. Liu, “Transformer PUF: A highly flexible configurable RO PUF based on FPGA,” in *Proc. IEEE Workshop Signal Process. Syst. (SiPS)*, Coimbra, Portugal, Oct. 2020, pp. 1–6.

[25] C. Herder, M.-D. Yu, F. Koushanfar, and S. Devadas, “Physical unclonable functions and applications: A tutorial,” *Proc. IEEE*, vol. 102, no. 8, pp. 1126–1141, Aug. 2014.

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

[27] N. N. Anandakumar, M. S. Hashmi, and S. K. Sanadhya, “Design and analysis of FPGA-based PUFs with enhanced performance for hardware-oriented security,” *ACM J. Emerg. Technol. Comput. Syst.*, vol. 18, no. 4, pp. 1–26, Oct. 2022.

[28] Y. Cui et al., “An efficient ring oscillator PUF using programmable delay units on FPGA,” *ACM Trans. Design Autom. Electron. Syst.*, vol. 29, no. 1, pp. 1–20, Nov. 2023.

[29] W. Yan, C. Jin, F. Tehranipoor, and J. A. Chandy, “Phase calibrated ring oscillator PUF design and implementation on FPGAs,” in *Proc. 27th Int. Conf. Field Program. Log. Appl. (FPL)*, Ghent, Belgium, Sep. 2017, pp. 1–8.

[30] F. Spagnolo, S. Perri, M. Vatalaro, F. Frustaci, F. Crupi, and P. Corsonello, “Exploring the usage of fast carry chains to implement multistage ring oscillators on FPGAs: Design and characterization,” *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, early access, May 7, 2024, doi: 10.1109/TVLSI.2024.3395302.

[31] L. Tebelmann, M. Pehl, and V. Immler, “Side-channel analysis of the TERO PUF,” in *Constructive Side-Channel Analysis and Secure Design* (Lecture Notes in Computer Science), vol. 11421. Cham, Switzerland: Springer, Mar. 2019, pp. 43–60, doi: 10.1007/978-3-030-16350-1_4.

[32] M. Vatalaro, R. De Rose, M. Lanuzza, and F. Crupi, “Static CMOS physically unclonable function based on 4T voltage divider with 0.6%–1.5% bit instability at 0.4–1.8 V operation in 180 nm,” *IEEE J. Solid-State Circuits*, vol. 57, no. 8, pp. 2509–2520, Aug. 2022.

[33] M. Pehl, A. R. Punnakkal, M. Hiller, and H. Graeb, “Advanced performance metrics for physical unclonable functions,” in *Proc. Int. Symp. Integr. Circuits (ISIC)*, Singapore, Dec. 2014, pp. 136–139.

[34] A. Rukhin et al., “A statistical test suite for random and pseudorandom number generators for cryptographic applications,” Nat. Inst. Standards Technol., Gaithersburg, MD, USA, Tech. Rep. 800-22 Rev 1a, Sep. 2010, p. 131.

[35] A. B. Alvarez, W. Zhao, and M. Alioto, “Static physically unclonable functions for secure chip identification with 1.95.8% native bit instability at 0.6 V and 15 fJ/bit in 65 nm,” in *IEEE J. Solid-State Circuits*, vol. 51, no. 3, pp. 763–775, Mar. 2016.

Fanny Spagnolo (Senior Member, IEEE) was born in Belvedere Marittimo, Cosenza, Italy, in April 1991. She received the master's degree in electronics engineering and the Ph.D. degree in information and communication technologies from the University of Calabria, Rende, Italy, in 2016 and 2019, respectively. She is currently an Assistant Professor with the Department of Informatics, Modeling, Electronics and System Engineering (DIMES), University of Calabria. Her research interests include VLSI architectures for image processing, high-performance reconfigurable circuits, embedded systems design, emerging technologies, and approximate computing techniques for low-power deep neural networks. She has co-authored more than 30 articles in these fields. She serves as a peer reviewer for several VLSI journals. She is an Associate Editor of *Integration, the VLSI Journal*.

Massimo Vatalaro (Member, IEEE) received the Ph.D. degree from the University of Calabria, Rende, Italy, in 2023. He is currently a Post-Doctoral Researcher with the Department of Computer Engineering, Modeling, Electronics and Systems Engineering (DIMES), University of Calabria. His research interests include circuit design in CMOS and emerging technologies and hardware-level security.

Stefania Perri (Senior Member, IEEE) was born in Cosenza, Italy, in April 1971. She received the master's degree in computer science engineering from the University of Calabria, Rende, Italy, in 1996, and the Ph.D. degree in electronics engineering from the University Mediterranea of Reggio Calabria, Reggio Calabria, Italy, in 2000. In 1996, she joined the Department of Electronics, Computer Sciences and Systems, University of Calabria, as an Associate Researcher. In 2002, she was appointed as an Assistant Professor of electronics with the Department of Electronics, Computer Science and Systems, University of Calabria. In Summer 2004, she was a Visiting Researcher with the Department of Electrical and Computer Engineering, University of Rochester, Rochester, NY, USA, where in 2005, she was appointed as an Adjunct Assistant Professor for four years. In 2010, she was appointed as an Associate Professor of electronics with the Department of Electronics, Computer Sciences and Systems, University of Calabria, where she joined the Department of Mechanical, Energy and Management Engineering, in 2017. Her current research interests include quantum-dot cellular automata (QCA)-based circuits, high-performance embedded systems, low-power design, VLSI circuits for image processing and multimedia, reconfigurable computing, and VLSI design. She is the co-author of more than 140 technical articles and holds two patents in these fields. She is a HiPEAC Member and serves on technical committees of several VLSI conferences and as a peer reviewer for several VLSI journals. She is an Associate Editor of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: EXPRESS BRIEFS, *Journal of Low Power Electronics and Applications*, and *Sensors*.

Felice Crupi (Senior Member, IEEE) is currently a Full Professor of electronics with the University of Calabria, Rende, Italy. His research interests include electronic device reliability, the design of ultralow-power analog circuits, and early assessment of emerging technologies for logic and memory applications. He was the Technical Program Committee Member of the International Electron Devices Meeting and the International Reliability Physics Symposium.

Pasquale Corsonello (Senior Member, IEEE) was born in Cosenza, Italy, in May 1964. He received the master's degree in electronics engineering from the University of Naples Federico II, Naples, Italy, in 1988. He joined the Institute of Research on Parallel Computers, National Council of Research of Italy, Naples, where he was working on the design and modeling of electronic transducers for high-precision measurement, receiving a post-graduate two-year grant. In 1992, he joined the Department of Electronics, Computer Science and Systems, University of Calabria, Rende, Italy, as a Research Associate. In 1997, he was appointed as an Assistant Professor of electronics with the Department of Electronics Engineering and Applied Mathematics, University of Reggio Calabria, Reggio Calabria, Italy, where he also served as the Director for the Microelectronics Laboratory. In 2001, he was appointed as an Associate Professor of electronics and the Chair of the Ph.D. Program in Electronics Engineering, University of Reggio Calabria. In Summer 2004, he was a Visiting Researcher with the Department of Electrical and Computer Engineering, University of Rochester, Rochester, NY, USA. In 2005, he was appointed as an Adjunct Associate Professor with the Department of Electrical and Computer Engineering. He is currently a Full Professor of electronics with the Department of Informatics, Modeling, Electronics and System Engineering (DIMES), University of Calabria. His research interests include embedded systems design, low-power design, VLSI architecture for image processing, and quantum-dot cellular automata (QCA)-based circuits. He has co-authored over 180 technical articles and holds two patents in these fields. He serves on technical committees of several VLSI conferences and as a peer reviewer for several VLSI journals. He served as the Editor-in-Chief for *Journal of Low Power Electronics and Applications* and the Associate Editor-in-Chief for IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS. Currently, he is a Senior Area Associate Editor of IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: EXPRESS BRIEFS and a member of the Steering Committee of IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS.

Open Access funding provided by 'Università della Calabria' within the CRUI CARE Agreement