IEEE Open Journal of the
Solid-State Circuits Society

Received 25 July 2022; revised 27 September 2022 and 17 October 2022; accepted 27 October 2022. Date of publication 18 November 2022; date of current version 7 April 2023.

Digital Object Identifier 10.1109/OJSSCS.2022.3223274

Aggressive Design Reuse for Ubiquitous Zero-Trust Edge Security—From Physical Design to Machine-Learning-Based Hardware Patching

MASSIMO ALIOTO (Fellow, IEEE)

(Invited Paper)

ECE Department, National University of Singapore, Singapore

CORRESPONDING AUTHOR: M. ALIOTO (e-mail: malioto@ieee.org)

This work was supported in part by the Singapore National Research Foundation and the Cyber Security Agency through the "SOCure" Project under Grant NRF2018NCR-NCR002-0001.

ABSTRACT This work presents an overview of challenges and solid pathways toward ubiquitous and sustainable hardware security in next-generation silicon chips at the edge of distributed and connected systems (e.g., IoT and AIoT). As the first challenge, the increasingly connected nature and the exponential proliferation of edge devices are unabatingly increasing the overall attack surface, making attacks easier and mandating ubiquitous security down to each edge node. At the same time, the necessity to incorporate zero-trust policies in large-scale distributed systems requires a complete set of security primitives for hardware-backed authentication, and a higher degree of physical context awareness (including primitives detecting the onset of physical attacks). Thus, making the inclusion of such security primitives economically sustainable even in low-end devices is a second key challenge. As third challenge, the ever-changing vulnerability landscape and the need for increased chip longevity in distributed systems require security assurance methods that are sustainable and adaptive across the entire chip lifecycle. In this work, design principles and promising directions to enable ubiquitous and sustainable security capabilities along with physical awareness are discussed. Such achievements require a fundamental rethinking of design methodologies to enable aggressive design and resource reuse (e.g., area, power, and design effort), along with low-cost on-chip sensorization and intelligence for physical attack detection. Such rethinking inevitably crosses over the traditional design abstractions, and requires innovation from the physical to the algorithmic level. At the physical and circuit levels, design and resource reuse is enabled by immersed-in-logic and in-memory security approaches. At the algorithm level, "hardware patching" is introduced and exemplified to show that runtime intelligence (machine learning) allows security capabilities to adapt and improve over time, as typical of security patching in software. Sensing techniques to detect attacks in situ from noninvasive to invasive are illustrated while still preserving fully automated design approaches. Overall, the above design principles are expected to push security capabilities in distributed systems to a new level, ultimately making the edge more intelligent and self-reliant, and security measures more distributed.

INDEX TERMS Hardware patching, hardware security, in-memory security, laser attacks, machine learning, physically unclonable functions (PUFs), sensors, side-channel attacks, true random number generators (TRNGs), ubiquitous and sustainable security.

I. INTRODUCTION

THE INCREASINGLY distributed and decentralized nature of the computing infrastructure is well known to lead to alarming trends in terms of yearly data breach count and cybercrime cost [1], [2], [3], [4], [5]. The global annual cost of cybercrime is growing by 15% per year from

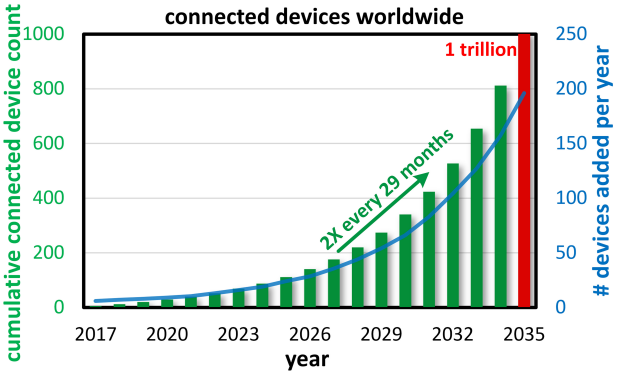


FIGURE 1. IoT devices added annually and cumulative count [8].

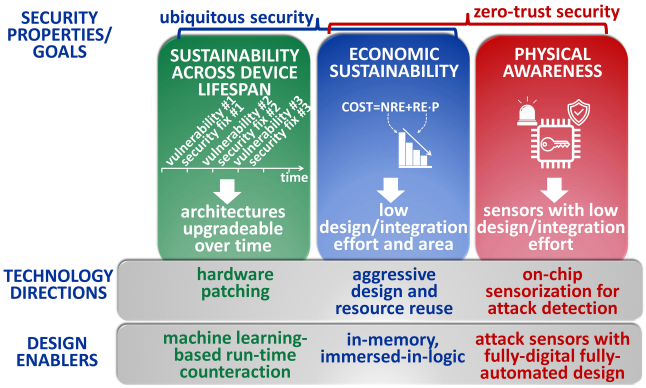


FIGURE 2. Graphical abstract: summary of security goals in edge devices, recent technology directions, and design enablers for scale-up to the trillions.

U.S. $3 trillion in 2015 to a projected U.S. $10.5 trillion by 2025 [6], which would correspond to the third largest economy worldwide if counted as GDP.

The above cybersecurity trends are mainly due to the increasing value of data, along with the increasing physical fragmentation and the proliferation of silicon systems, which, in turn, expose an unceasingly larger attack surface from cloud to edge. At the cloud level, the widespread adoption of hybrid multicloud architectures with ever-expanding infrastructure decreases threat visibility and leads to fragmented security solutions [7]. As focus of this work, the edge is exposed to similar considerations, as the number of connected devices added every year is exponentially increasing at a rate of 2× every 29 months, and is expected to reach the trillion scale in about a decade [8] (Fig. 1). Also, the security threats at the edge are becoming particularly concerning since edge devices are tightly resource constrained [9] and, hence, they cannot nearly afford the same level of security of silicon systems in the cloud [10]. The concern becomes more pronounced considering that the physical data generated by edge devices are increasingly used in actuation feedback schemes without a human-in-the-loop, which has major safety implications.

From the above considerations, security at the edge needs to be ubiquitous (i.e., assured in every node), since security in connected systems is only as good as the weakest link in the chain of trust. Ubiquitous security fundamentally conflicts with the constrained nature and low-cost requirement of edge nodes and, hence, requires innovative and inexpensive hardware security primitives and solutions to make security economically sustainable even in low-end devices (e.g., low area, design and system integration effort, and power).

Economic sustainability becomes particularly challenging in distributed systems, in which network perimeter defence (e.g., concentrating security measures on dedicated devices such as firewalls) is well known to be inadequate, since physical perimeters are vanishing [11]. Indeed, today's distributed systems mandate the adoption of zero-trust frameworks and, hence, the principles of: 1) hardware-backed device authentication with end-to-end session encryption and 2) active security assessment through physical context awareness [12], [13], [14]. From these principles, all edge nodes need to be equipped with a complete set of on-chip security primitives for: 1) secret key generation [e.g., physically unclonable function (PUF) and true random number generator (TRNG)] and encryption and 2) physical security primitives providing physical context awareness, i.e., the ability to be aware of the physical surroundings to detect the onset of physical attacks.

As a further challenge in distributed systems, the significantly prolonged longevity of edge devices (e.g., several years to decades) requires security solutions to be effective and sustainable over time. This would require the ability to incorporate security fixes in edge devices upon the discovery of new vulnerabilities over their lifecycle. Unfortunately, hardware patchability to adapt to an ever-changing vulnerability landscape is currently unavailable, and needs to be enabled as an expansion of traditional software patching.

In this work, promising directions to make edge security ubiquitous, sustainable (both economically and over time), and physically aware are discussed through case studies based on silicon demonstrations, as summarized in the graphical abstract in Fig. 2. Section II introduces key challenges posed by the scale-up to the trillions in the number of connected devices. Section III presents a broad overview of the technological trends in the generation of secret keys as root of trust, and foundation of any secure system. Sections IV and V present a range of low-cost and low-power solutions to embed security primitives for secret key generation into memory and logic fabrics ("immersed-in-logic"). Case studies of techniques enabling physical awareness under fully automated design are discussed in Section VI for noninvasive attacks, and in Section VII for invasive attacks. The enablement of hardware patching via on-chip machine learning modeling is discussed in Section VIII (not related to machine learning attacks). Conclusions are summarized in Section IX.

II. SECURITY AT THE EDGE TOWARD THE TRILLION SCALE

The physically scattered distribution of edge nodes and their exponential increase in number make zero-trust policies and the related security primitives supporting them necessary. In

zero-trust security frameworks, security primitives typically consist in a hardware root of trust (e.g., a PUF), renewable credential generation (e.g., true random number generation) and encryption/decryption capabilities [12]. Their addition to low-cost and low-power nodes must be made economically sustainable to enable ubiquitous security, as most edge nodes need to be very low cost (e.g., deep sub-$).

The cost of added security primitives is mainly determined by nonrecurring engineering (NRE) costs and the recurring expenses (REs). The NRE costs related to security primitives are directly impacted by the design effort from design entry to layout, the system integration effort to embed them into a system on chip, the related verification effort, and the ability to reuse/port the same design across systems and technology generations. The RE costs are mostly determined by the extra silicon area that is required to add the primitives. From a design methodology perspective, divide-and-conquer design approaches are generally adopted to facilitate design, partitioning a system into a collection of standalone blocks. However, they are inherently additive in terms of area, power, design effort and integration effort and, hence, preclude opportunities to create economies of scale across subsystems [3]. Also, conventional design partitioning conflicts with basic security requirements as it facilitates the identification of attack targets, and reduces the related effort by focusing on specific areas of the overall attack surface. Accordingly, aggressive design reuse and tight integration of security primitives with existing subsystems can be highly beneficial in terms of design and system integration costs, as well as in terms of security. As key examples of such general principles, Sections IV-V illustrate in-memory and immersed-in-logic embedment of security primitives in ubiquitously available subsystems (i.e., storage, processing). Immersion in memory and logic provides the further benefit of 1) leveraging the fully automated nature of their design as enabled by memory compilers and standard cell automated flows, 2) suppressing any system integration effort, 3) allowing technology and design portability as in Fig. 2.

The RE costs of security primitives can be directly reduced through resource reuse (e.g., circuit), so that their addition comes at very low area cost by mostly leveraging existing circuits as in Fig. 2 (e.g., a cryptographic core or an SRAM memory array). Once again, in-memory and immersed-in-logic primitives offer a higher level of security thanks to inherent physical-level obfuscation against physical attacks (e.g., microprobing, photon emission, laser voltage probing), as well as stricter data locality for architecture-level security [3]. At the same time, the additive power consumption of security primitives must be kept minimal to avoid increasing the system energy source size and cost.

From a physical awareness viewpoint, edge devices are particularly vulnerable since they are spatially scattered and, hence, more exposed to physical attacks, as opposed to the confinement of the cloud infrastructure. In addition, edge nodes are often placed in inaccessible locations, making their inspection unfeasible. As a consequence, physical awareness of edge nodes typically comes only from on-chip physical security sensors. Physical security primitives to detect the onset of physical attacks are also needed to support the necessary zero-trust policies, as discussed in the Introduction. As security primitives, such on-chip sensors need to be inexpensive from an NRE and RE perspective and, hence, need to be deployable through fully digital fully automated approaches as in Fig. 2.

Once edge devices are protected through inexpensive security primitives, security over their long lifespan is assured only until new vulnerabilities are discovered and exploited. In software systems, security fixes in the form of software patches are routinely released to withstand new threats, improving security over time. Unfortunately, hardware patches are largely unavailable, and even more so for physical attacks. The inability to cope with new threats fundamentally limits the usable lifespan of edge devices, determining premature obsolescence that can easily become shorter than the typical battery shelf life (e.g., two decades for micro-batteries [15]). To restore full lifespan potential, edge nodes need to acquire the ability to keep up with new threats and, hence, develop hardware patching capabilities to sustain the necessary level of security over time. At the logic level, hardware patchability is easily enabled via logic-level adaptation, whereas the real challenge lies in physical attack counteraction. In this respect, on-chip runtime machine learning comes to the rescue by offering the necessary intelligence to model new physical attacks and support reconfigurability (e.g., via weight update), based on on-chip features mimicking adversary's observations to extract information from physical (side) channels. In other words, machine learning for hardware patching is here referred to as a counteraction tool, rather than conventionally using it as a tool to attack secure systems.

The above security goals for scale-up to the trillions at the edge are discussed in the following sections along with the promising technology directions and the key design enablers in Fig. 2.

III. TRENDS IN SECRET KEY GENERATION (ROOT OF TRUST)

The chain of trust in secure systems has its own foundation in the root of trust, which is a secret shared between parties exchanging data securely [1]. To confine their generation and utilization within the chip environment for physical security reasons, digital keys are generated on chip and on the fly by binarizing device physical deviations either in space (across the chip) or time (over the chip lifespan). Their on-the-fly generation avoids the many well-known physical vulnerabilities of on-chip memories (or other storage methods such as fuses) and the predictability of deterministic generation, eliminating any exposure of the keys to physical attacks when the device (and physical protections) is off [1].

From an application viewpoint, secret key generation is routinely split into two main functions with opposite focus

TABLE 1. Secret key generation and properties of related security primitives.

key extraction process	static	dynamic
extraction from device deviations across	space	time
physical phenomenon	mismatch and its derivations	noise and its derivations
security primitives	Physically Unclonable Functions (**PUFs**)	True Random Number Generators (**TRNGs**)
circuit goal	amplify mismatch, suppress noise	amplify noise, suppress mismatch
post-processing for enhancement in time	temporal stability enhancement for repeatability	entropy enhancement for bit decorrelation
examples of security functions	device authentication, remote attestation, chip ID, secure exchange of private keys without public-key cryptography	fresh session keys, nonces and initialization vectors for encryption / decryption, bit padding

on space and time physical deviations [10], as shown in Table 1.

1) *SPACE:* Chip-specific mismatch patterns generate unpredictable and perfectly repeatable keys serving as silicon fingerprint (e.g., for identification and authentication of individual physical chips, among the others); this function is routinely executed by PUFs. PUFs simply respond to an external digital word (challenge) with a unique digital response. Although ideally repeatable, the response bits inevitably suffer from occasional instability, which is routinely mitigated through digital postprocessing [2] (e.g., error correction codes), and error correction codes for residual instability suppression (ECC).
2) *TIME:* Time-varying bit sequences are generated to provide fresh session keys, nonces and initialization vectors for encryption/decryption, and bit padding to stretch the word length to the encryption block size; this is generally achieved through TRNGs [1].

A summary of the state of the art in PUFs and TRNGs is kept up to date in the HWsecdb public database [16], as companion of this article. Trends in PUFs and TRNGs are discussed in the following sections. The utilization of the resulting root of trust via digital cryptographic accelerators is not discussed in the following, as their implementation and integration challenges are no different from any other digital block (e.g., power–performance–area tradeoff in standard cell-based designs). As a more distinctive requirement of secure systems, physical security challenges and solutions common to digital blocks are discussed in Sections VI–VIII.

A. TRENDS AND ADVANCES IN PHYSICALLY UNCLONABLE FUNCTIONS

In general, PUFs are based on circuit principles ranging from analog-, to delay-, memory-, metastability-based to monostable PUFs, hybrid architectures combining multiple principles, and others [16]. To date, most of the innovation effort has been focused on weak PUFs, whose limited responses are generally used as cryptographic keys within the chip (or the server) to avoid the direct exposure to the off-chip environment. Strong PUFs have an exponential number of challenge–response pairs at the cost of much higher energy consumption, and their use is much less frequent (also due to its lack of usability as cryptographic key). The limited sample of available architectures does not reveal any clear technological trend [16]. Hence, the following analysis focused on weak PUFs.

Fig. 3(a) shows that the area/bit improvements in raw PUFs (i.e., before postprocessing) are more pronounced in volatile memory-based [17], [18], [19], [20], [21], [22], [23] and analog [24], [25], [26], [27] architectures. The figure clearly shows that such improvements are mostly taking place somewhat faster than allowed by technology scaling only. Monostable PUFs essentially scale at the same pace enabled by technology scaling [28], [29], [30], [31], [32], [33], [34], [35], being mostly digital circuits. Metastability-based PUFs tend to be on the higher side of the silicon area range [36], [37], [38], [39], [40]. Delay-based PUFs are not taking full advantage of technology scaling due to their slower scaling, due to the cost of peripheral circuits to preserve margin against PVT variations [41], [42], [43], [44].

From Fig. 3(b) [16], the native stability of raw PUFs is not improving over time, as revealed by the generally increasing trend of the bit error rate (BER). In general, the BER needs to be sufficiently low to make the mean time between failures (MTBF) long enough to make the effect of PUF instability irrelevant. In edge nodes with typical MTBF targets between months and a decade, the percentage key error rate (KER) must lie in the 10^{-6}–10^{-3} range [33]. In turn, the KER is approximately equal to the product of the PUF wordlength and the BER, when no ECC is employed and the BER is reasonably low. As a consequence, edge nodes routinely target a percentage BER in the ∼0.00001%–0.001% (∼0.000001%–0.0001%) range for short (long) 32-bit (256-bit) PUF words [see gray area in Fig. 3(b)]. In most PUF demonstrations, the BER is well above this allowed range, thus requiring the addition of the invariably dominant area and energy cost of ECC (e.g., 2–3 orders of magnitude higher than the raw PUF and postprocessing). In a few silicon demonstrations, the native BER has reached sufficiently low levels that allow the suppression of ECC in embedded DRAM PUFs [19], oxide rupture-based PUFs [45], and RRAM arrays in view of their pronounced process variations [46].

As in Fig. 3(c), innovation in postprocessing techniques is substantially improving the stability in most PUF architectures, especially in nonvolatile memory-based [46], [47], [48], [49], [50] and analog PUFs [24], [25], [26], [27]. Volatile memory-based PUFs are also substantially improving [17], [18], [19], [20], [21], [22], [23], and allow ECC-less operation through methods, such as VSS-bias generator for

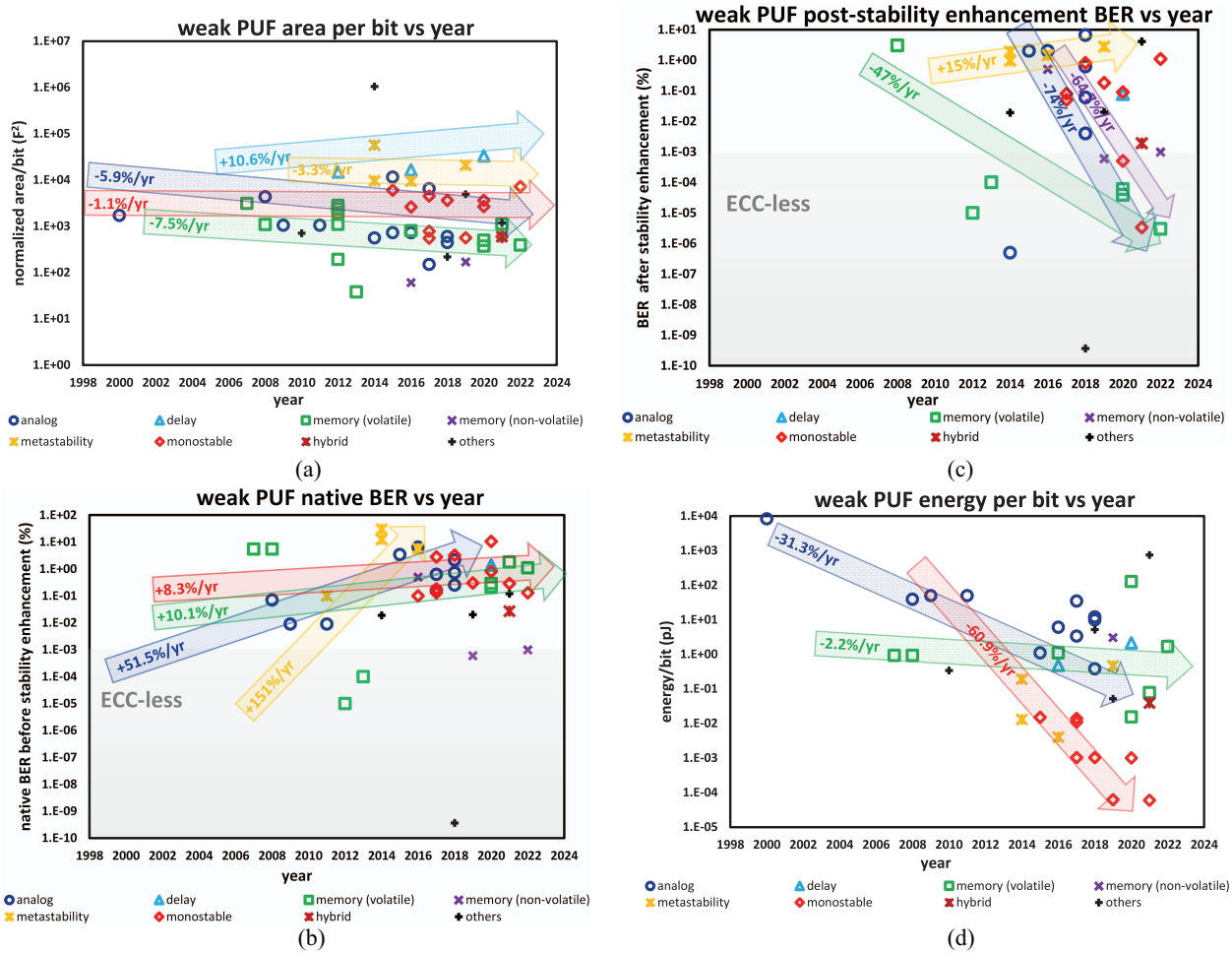


FIGURE 3. PUF trends in terms of (a) silicon area, (b) native bit stability (BER), (c) post-stability enhancement BER, and (d) energy/bit.

dark bit detection [20] and hot carrier injection-induced stability reinforcement [21]. Monostable PUF stability is becoming divergent with a rapidly improving trend for some of recent demonstrations equipped with machine learning for runtime PVT margin handling [33] and self-healing [34], and a moderately increasing BER for some others [28], [29], [30], [31], [32], [35]. Explicit runtime design margin handling in [33] and [34] also enables ECC-less operation, as opposed to other monostable PUF types. On the other hand, metastability-based PUFs are on the higher side of the post-stability enhancement range, and are experiencing a degradation over time due to the substantial increase in their native BER [16].

The energy efficiency of PUFs keeps improving relentlessly and faster than allowed by pure technology scaling for digital PUF classes, such as volatile memory-based [17], [18], [19], [20], [21], [22], [23] and monostable [28], [29], [30], [31], [32], [33], [34], [35], thanks to the adoption of energy-aware circuit principles. The energy of analog PUFs is also significantly decreasing, although it remains on the higher side of the energy range [24], [25], [26], [27]. The energy/bit in best-in-class PUFs is achieved by monostable PUFs and is in the fJ/bit range and below. Further reductions are unnecessary since the overall power to generate static entropy would be dominated by postprocessing (and/or ECC) building blocks anyway.

In summary, ubiquitous adoption of PUFs is well supported by monostable PUFs in view of their competitive energy efficiency and stability, with area improvements being mainly set by technology scaling. Better area efficiency for low cost is expectedly enabled by volatile and non-volatile memory PUFs, although at higher energy by orders of magnitude.

B. TRENDS AND ADVANCES IN TRUE RANDOM NUMBER GENERATORS

From [16], the throughput of TRNGs is consistently increasing in most TRNG architectures, supporting the exchange of larger volumes of data with fresher keys as summarized in Fig. 4(a). High-speed TRNGs reach a throughput in the multi-Gb/s range, whereas it is in the 1–100-Mb/s range for most of the others. Metastability-based TRNGs are on the higher end of the throughput range [40], [51], [52], [53], [54], [55], [56], although their advantage is becoming less pronounced over time due to the heavier effect of

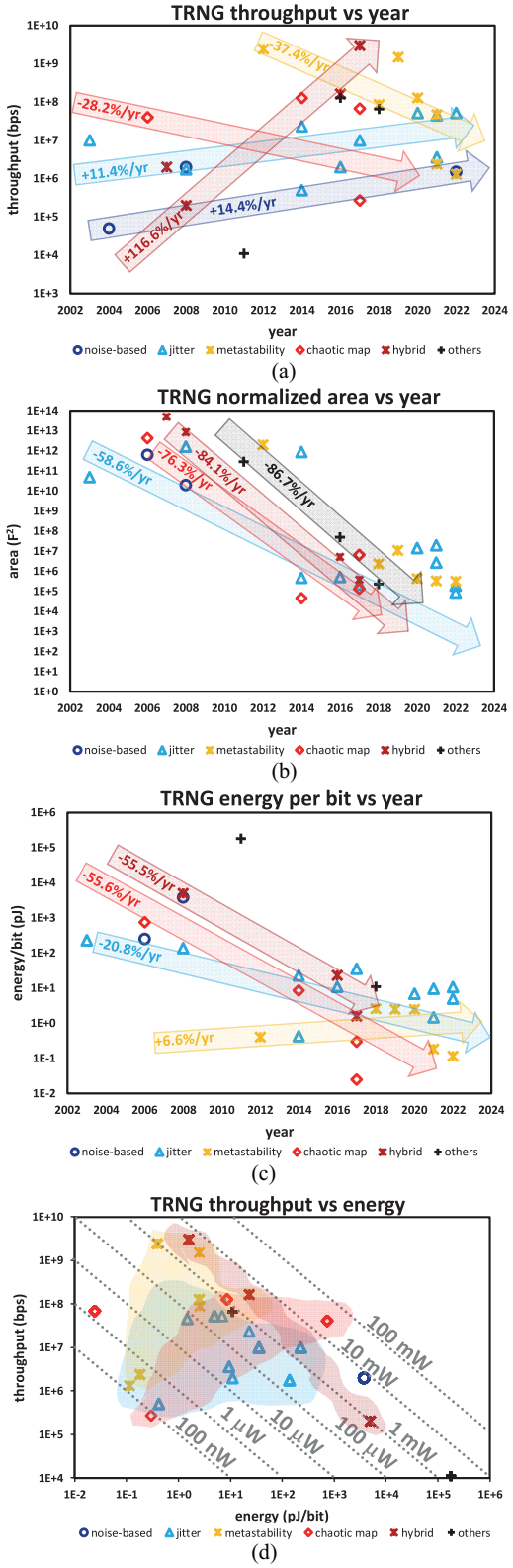


FIGURE 4. **TRNG trends in terms of (a) throughput, (b) silicon area, (c) energy/bit, and (d) throughput–energy tradeoff and resulting power consumption.**

mismatch and, hence, the number of raw bits to be postprocessed to extract each output bit. Jitter- [22], [57], [58], [59], [60], [61], [62], [63], [64], [65], [66] and chaotic map-based TRNGs [67], [68], [69], [70] generally have lower throughput than metastability-based chaotic maps. The benefits brought by different randomness sources in hybrid TRNGs is making them particularly well suited for mid-to-high throughputs [71], [72], [73], [74], and in particular [73], [74].

As shown in Fig. 4(b), the area efficiency of TRNGs is consistently improving beyond allowed by technology scaling for most architectures. Chaotic maps [67], [68], [69], [70], hybrid [71], [72], [73], [74] and other TRNG classes [75], [76], [77] are exhibiting the most pronounced improvements, thanks to the recent adoption of more or mostly digital approaches. Jitter-based TRNGs still exhibit quite pronounced improvements, thanks to the recent adoption of PVT-resilient schemes and noise enhancements [22], [57], [58], [59], [60], [61], [62], [63], [64], [65], [66]. The best-in-class area efficiency is achieved by specific demonstrations belonging to the jitter-based and chaotic map TRNGs.

From Fig. 4(c), the energy per bit keeps decreasing significantly, especially for chaotic map [67], [68], [69], [70], jitter-based [22], [57], [58], [59], [60], [61], [62], [63], [64], [65], [66] and expectedly hybrid architectures [71], [72], [73], [74]. These classes are benefitting from energy downscaling well beyond allowed by technology scaling. As an exception, metastability-based TRNGs are experiencing a moderate increase in their consumption [40], [51], [52], [53], [54], [55], [56], again due to the progressively heavier effect of mismatch and, hence, amount of postprocessing to suppress it. However, their energy efficiency is still competitive and near to best-in-class. Noise-based TRNGs lie in the higher end of the area and energy range [46], [78], [79], [80], [81]. Best-in-class energy efficiency is achieved by specific demonstrations belonging to the jitter-based [63], chaotic map [69], [70] and metastability-based TRNGs [55], [56].

Fig. 4(d) shows that the throughput–energy tradeoff is most favorable in jitter-based TRNGs, leading to a full-throughput power consumption in the μW to mW range. Some specific instances of chaotic maps [69] and metastability-based TRNGs [56] are also equivalent to the best jitter-based ones in terms of throughput–energy tradeoff and, hence, power efficiency. The same figure shows that the adoption of multiple entropy sources in hybrid TRNGs inevitably pushes their power on the higher end of its overall range.

In summary, both jitter- and metastability-based TRNGs exhibit the lowest area and the lowest energy/bit along with some specific instances of chaotic maps, and are hence the most suited for the enablement of ubiquitous security.

IV. IN-MEMORY AND OTHER UNIFIED SECURITY PRIMITIVES

Unified security primitives combine multiple functions into the same circuitry to enable aggressive design reuse and tight integration with existing subsystems, reduce area via circuit sharing, and eliminate any integration effort over multiple

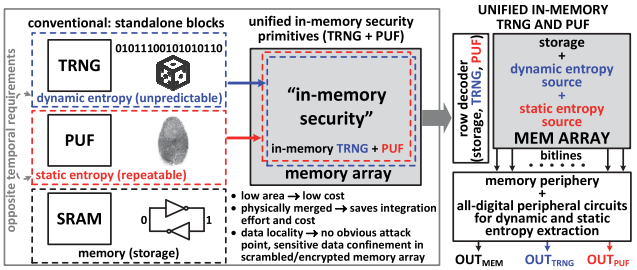


FIGURE 5. Principles and advantages in in-memory unified security primitives.

systems once the primitive is designed. A higher degree of design reuse is achieved when unifying security primitives with blocks that are widespread across systems on chip. This suggests that unified primitives should be embedded in building blocks that are inherently ubiquitous such as on-chip memories (Section IV-A), although they can also be incorporated into several other blocks (Section IV-B).

A. IN-MEMORY PRIMITIVES

The combination of a PUF embedded in an SRAM array is just an SRAM PUF [17], [18], [21], which should allow conventional memory access anyway and is hence not really a unified primitive. As an interesting class of unified primitives, on-chip SRAM memories embedding both a PUF and a TRNG have been recently demonstrated [22], as in Fig. 5.

The unified in-memory PUF+TRNG SRAM architecture in [22] is a complete and low-cost root of trust in terms of area, design, and integration effort. The randomness generated by the SRAM array is harvested through an enhanced column periphery while sharing the conventional SRAM circuitry, limiting the area overhead to 12% for the smallest memory comprising one subarray. The overhead is further reduced when amortized across typical multiple subarrays. The row and column pitch-matched physical design of the periphery retains the traditional capability of using compiler-based automated SRAM design for immediate reuse across systems. SRAM bank read/write can be temporally interspersed with TRNG and PUF operation without intermediate data flushing for seamless system integration, as opposed to prior SRAM PUFs [17], [18], [21]. In [22], in-memory TRNG and PUF operation is based on the same principle of digitizing the bitline discharge time, as determined by different currents with very high (leakage) or very low (bitcell read current) noise component, compared to mismatch. In TRNG mode, randomness is generated by digitizing the discharge time of the bitline capacitance discharge due to the leakage current provided by all bitcells within the same column, when all wordlines are disabled. The accumulated noise translates into time jitter that is a white stochastic process with a negligible effect of mismatch, thanks to the noise participation of all bitcells in a column and the mismatch averaging throughout the several bitcells. In the PUF mode, the selected bitcell read current is digitized to generate multiple bits per bitcell, as opposed to single-bit extraction from the power-up bitcell state of conventional SRAM PUFs [17], [18], [21]. The extraction of two bits per bitcell increases the capacity of the PUF and, hence, allows to reduce its area for a given PUF word count requirement. PUF bits are harvested from adjacent bitcell pairs of a selected and a half-selected bitcell under common column multiplexing. This allows full reuse of the half-selected bitcells in terms of both area and energy for lower cost and power.

Although less widespread than SRAMs, other on-chip volatile memory arrays are encountered in connected systems, and are hence candidates for low-cost low-power unified primitives. Within the eDRAM memory class, in-memory PUFs, such as [19] and [23] are not really unified primitives for the same reasons mentioned for SRAM PUFs. Off-chip DRAM-based primitives have also been explored [54], although the immediate physical accessibility of the root of trust through their I/Os makes them unappealing from a security perspective (unless area/energy penalty is paid for additional off-chip memory encryption).

Unified in-memory primitives have also been recently embedded in nonvolatile memory arrays. Security primitives can be embedded in eFlash memories, as in the case of the unified PUF+TRNG eFlash array in [49]. An in-memory TRNG was also shown in a voltage-controlled MRAM array [82], which eliminates the need for calibration in TRNG mode. A unified PUF+TRNG has been also demonstrated in an RRAM array [46], which enables design reuse in an inherently high-density fabric.

In summary, SRAMs are more widespread than other on-chip memories, and are hence the best candidate for ubiquitous and inexpensive security primitive integration. Additional opportunities are available in DRAM and nonvolatile memories. In addition, the ability to reuse conventionally unused bitcells and extract multiple bits/bitcell is crucial to make in-memory primitives inexpensive and, hence, ubiquitous down to low-end devices, and is expectedly raising significant interest in the field (see next section).

B. OTHER UNIFIED PRIMITIVES WITH SINGLE- AND MULTIBIT/BITCELL

Although outside memory arrays, various unified security primitives incorporated in other building blocks have been demonstrated for ubiquitous security.

Starting from the class of multibit/bitcell-unified PUFs, the SRAM macro in [83] unifies a cache, a PUF, and a temperature sensor by connecting bitcells to form pairs of voltage reference generators, and digitizing the mismatch-induced voltage difference. The formation of voltage reference generators through the combination of different bitcells enables the generation of multiple bits/bitcell. As another example of multibit/bitcell PUF, [84] generates 2 bits/cell in an array of MOS transistors (not including the periphery) where the analog position of soft breakdown spots is exploited as a randomness source. The TRNG in [58] generates multiple bits per cycle in a two-phase oscillator-based architecture by extracting random bits from both oscillator phases.

As another example of aggressive design reuse, a PUF+TRNG primitive has been unified by reusing the unstable bits of a metastability-based PUF as TRNG source [40]. PUFs and TRNGs have been unified into on-chip communication fabrics, as demonstrated in the 3-D threshold switching crossbar in [81]. Also, TRNGs have been unified in analog-to-digital converters based on SAR [69] and Delta-Sigma architectures [60].

V. IMMERSED-IN-LOGIC SECURITY PRIMITIVES

Aggressive design reuse can also be achieved by embedding security primitives into the processing subsystem, which is routinely designed with standard cells as in Fig. 6. From this figure, the root of trust is just part of the same design of the digital logic that uses it. This drastically simplifies design and integration, while enforcing data locality and physical obfuscation since there is no obvious point of attack, as opposed to standalone root of trusts.

A. IMMERSED-IN-LOGIC PUFS

The main challenge in immersed-in-logic PUFs is in the requirement to: 1) have the randomness source implemented in the form of standard cells and 2) keep the PUF unaffected by the unpredictable placement (i.e., the surrounding layout environment) and routing (i.e., load and noise coupled with the cells used in the root of trust).

Among the existing PUF classes in Section III-A, the most suited class that is compatible with the standard cell design discipline and can potentially fulfill the above requirements is represented by monostable PUFs. Indeed, analog and volatile/nonvolatile memory PUFs are incompatible with standard cell flows. On the other hand, delay and metastability PUFs are heavily impacted by placement and routing, as their operating principle is dictated by signal timing. Instead, monostable PUFs are based on the generation of dc voltages, and their binarization through a logic network restoring full-swing signaling. In monostable PUFs, voltage generation can be easily embedded into a special standard cell [28], [30], [33] or created by connecting existing standard cells [32], [85], and are hence naturally immersed in logic as a result.

The above two subclasses of immersed-in-logic monostable PUFs share the same ability to be robust against the unpredictability of automated routing. Indeed, they are both based on dc voltage generation, which makes its distribution unaffected by the layout parasitics and, hence, timing during its transient. At the same time, the effect of coupling noise is negligible in view of its transient nature [28]. On the other hand, the two classes differ in terms of their robustness against the layout environment surrounding the cells generating randomness. Indeed, the adoption of special standard cells allows to include layout structures that maintains the same layout environment around the circuitry that is most sensitive to layout-dependent variations [30]. The resulting insensitivity to placement has been extensively evidenced across placement patterns in prior art [30].

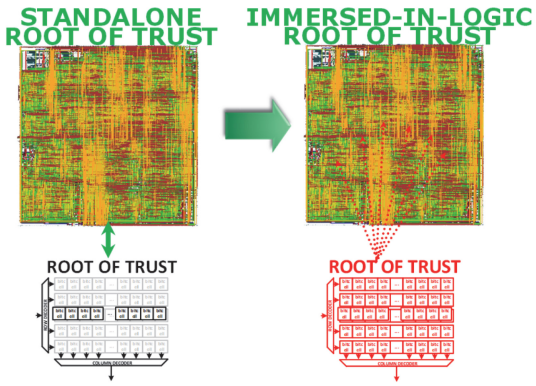


FIGURE 6. Immersed-in-logic root of trust enables automated standard cell design along with seamless integration, data locality, and obfuscation.

Immersed-in-logic monostable PUFs using solely existing standard cells have the advantage of being implementable in any technology, suppressing the extra effort of laying out and characterizing the special PUF cell. At the same time, their inevitably high sensitivity to the layout environment and placement mandates a very regular array organization of cells generating randomness (see the implementation in [32]). In turn, this prohibits full interspersion of PUF and the surrounding logic, makes the PUF design less automated, and makes points of attack within standard cell logic more obvious (although less obvious than a standalone PUF).

B. IMMERSED-IN-LOGIC AND UNIFIED TRNGS

Various immersed-in-logic TRNGs have been demonstrated in prior art. As TRNG class that is most interesting for ubiquitous security, unified immersed-in-logic TRNGs have been recently introduced. In particular, Taneja and Alioto [53] introduced a novel class of architectures that unify the root of trust generation (TRNG) and private-key cryptographic accelerators by reusing the latter for both tasks. This architecture is based on the implementation of a cryptographic accelerator with pulsed latch clocking. Narrow clock pulse width makes pulsed latch operate like conventional flip-flops, as necessary to correctly execute encryption. When the clock pulse is overstretched, randomness is generated by inducing latch metastability due to the violation of hold time constraints. Further pulse width overstretching activates combinational loops since pulsed latches are kept transparent for a long time, therefore triggering jittered oscillations. In other words, clock pulse overstretching injects randomness as in metastability- and jitter-based TRNGs (see Section III-A). The cryptographic datapath amplifies the changes of individual bits, and makes the output entropy of cryptographic grade. As further benefits, the unification of TRNG and cryptographic accelerator preserves tight data locality and physical obfuscation of key generation within the logic making use of it.

Regarding nonunified TRNGs, various other fully synthesizable architectures that can be implemented automatically

with standard cell flows[1] have been demonstrated in recent years [62], [64], [73]. The TRNG in [64] is based on 3-edge ring oscillators with odd number of stages and three 120°-shifted output phases. The race of the three accumulated jitters at the three edges ultimately leads to state collapse, shifting from 3× to 1× oscillation frequency with a random time (i.e., the TRNG output). The effect of process, voltage, temperature variations and load differences due to automated placement&routing is canceled by construction since the three edges pass through the same delay stages. Although this method to counteract the effect of automated placement&routing is effective, it is specific to the TRNG architecture in [64] and, hence, not generally applicable.

As a more general approach to suppress the effect of timing mismatch induced by uneven placement and routing, automatic tuning loops have been explored to restore the targeted 0/1 bias (i.e., nearly 50%) [62], [73]. Automatic tuning loop allows to cancel the effect of process variations, including mismatch, slow voltage and temperature variations, as well as the uneven load or layout environment seen by the elementary cells generating randomness. In particular, [73] is based on an architecture that is similar to [64], where the output randomness is generated by the random time-to-collapse in a ring of delay stages (although with an even number of stages). Similarly, Mathew et al. [73] combined multiple independent sources of randomness to extract a cryptographic-grade bitstream via an entropy extractor, while suppressing the effect of process, voltage, temperature and layout-dependent variations with a runtime tuning loop. As an alternative to automatic tuning loops, Pamula et al. [52] combined multiple sources of randomness through Markov chain-based correction to de-correlate and de-bias the output bitstream.

VI. ON-CHIP SENSORIZATION FOR NONINVASIVE PHYSICAL ATTACK DETECTION: POWER ANALYSIS CASE STUDY

Ubiquitous security against physical attacks is also required in connected devices. Indeed, until not so long ago, the main objective of hardware security was to simply counteract cryptanalytic attacks, based on secure bitstream analysis as main channel of observation. Their counteraction required a solid on-chip root of trust with cryptographic-grade randomness, and a trusted cryptographic algorithm. Since physical attacks have become very effective and widely accessible, they now represent a very appealing opportunity for adversaries. Hence, their counteraction needs to be embedded ubiquitously even in low-end devices.

Among physical attacks [2], noninvasive attacks are particularly simple since they do not require any de-packaging or die processing, and can hence be carried out unnoticeably during in-field system operation. In particular, side-channel attacks based on the observation of physical channels are now very inexpensive and require minimal equipment. As highly representative class of attacks, power analysis and EM attacks correlate the data-dependent consumption patterns to the data being processed, including the secret key being used during the encryption/decryption. Nowadays, low-cost (e.g., U.S. $50 [86]) open-source boards to execute power analysis and EM attacks are widely available to black- and white-hat hackers to break or assess the security of silicon systems, mandating ubiquitous protection against such attacks.

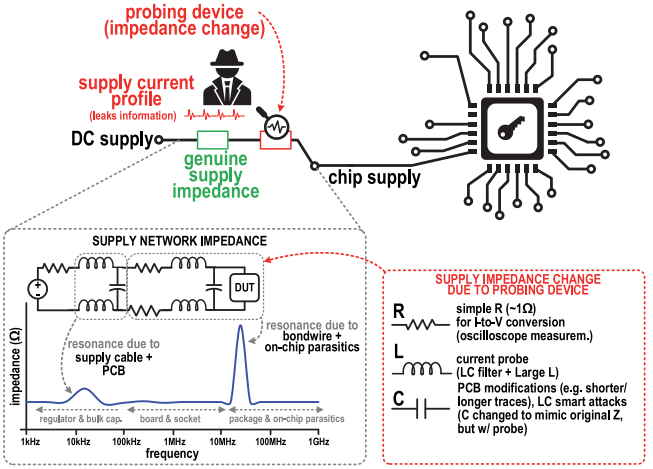


FIGURE 7. Power analysis attack detection corresponds to the detection of changes in the impedance seen from the supply pad of the chip under attack.

The rising need for ubiquitous protection against side-side-channel attacks has led to advances in power analysis attack detection only in the last few years, whereas counteraction has been widely explored in the last two decades (see Section VIII). Since the analysis of the power consumption during an attack requires the insertion of a probing device, attack detection essentially consists in revealing changes in the off-chip supply impedance seen from the chip [87], [88], [89]. In other words, the detection of power analysis attacks requires the inclusion of physical context awareness around the supply pad. This is equivalent to developing an intelligent impedance sensor suitable for the supply pad, and a simple binary classifier that associates the sensed impedance with a threat flag (or threat level, under multiple classes), as illustrated in Fig. 7.

Among the techniques for supply impedance monitoring summarized in Table 2, Kim et al. [87] introduced a digital low drop-out regulator (DLDO) embedding some attack detection capability by monitoring the resistive component of the supply resistance. In other words, the DLDO includes a resistor meter and allows to check if a resistor has been inserted at any point of time, as the simplest form of probing device. The effectiveness (and power) of the embedded side-channel counteraction is raised only when needed (i.e., when an attack is detected), to minimize the overall impact on the average consumption. This solution has no awareness of the supply environment above dc, and cannot detect the insertion

1. This category includes all TRNG architectures that are based on logic cells and can hence be potentially designed automatically, although some of the demonstrations were chosen to be designed with custom cells and might also be considered as a class on its own (e.g., TRNGs based on 3-edge ring oscillator, time-to-collapse principle).

TABLE 2. Summary of state-of-the-art techniques to detect impedance changes (useful for power analysis attack detection).

	VLSI'22 [90]	VLSI'21 [87]	JSSC'18 [88]	JSSC'17 [89]
process [nm]	28	65	180	28
type of monitor	supply impedance	supply resistance	chip-package-board silicon fingerprint	power delivery monitor (supply noise & impedance)
quantification of resistance/impedance deviation	YES	YES	NO	YES
normalized area [a] [$\mu m^2/F^2$]	$6.5 \cdot 10^6$	$6.6 \cdot 10^6$	$0.097 \cdot 10^6$	$138.4 \cdot 10^6$
under-pad placement [b]	YES	NO	YES	NO
monitoring at arbitrary frequency	YES	NO	NO	YES
supports run-time monitoring	YES	YES	YES	NO
bandwidth [Hz]	2 GHz	N/A	N/A	0.8 GHz
average active power [mW]	2.9	0.00065 (quiescent current)	7.92	0.42
calibration-less	YES	NO	YES	-

(a) area normalized to F^2 (F = minimum feature size of the process)
(b) placement under pair of supply wirebond pads (60 µm x 60 µm)

of current probes, and PCB- and package-level tampering. Alternatively, changes in the imaginary parts of the supply impedance are captured by the interactive PUF in [88]. In this case, the inductive component changes due to modifications in the physical surroundings is evaluated through a chaotic oscillator coupled with an inductor. Although this solution interestingly gains physical context awareness in the imaginary part of the impedance, it is not suitable for supply monitoring due to its oscillatory nature. Also, it cannot capture the resistive part or quantify the impedance deviation, as necessary for reliable discrimination and balance of false positives and false negatives. Both the real and imaginary part of the supply impedance can be monitored by the on-chip digital sampling oscilloscope in [89], which accurately models both supply noise and impedance at chip bring-up thanks to dedicated software support (not easily available in low-cost devices or affordable during workload execution). This capability requires in-field recalibrations against temperature fluctuations and an area of ∼40 pads, which make it suitable only for higher end silicon systems.

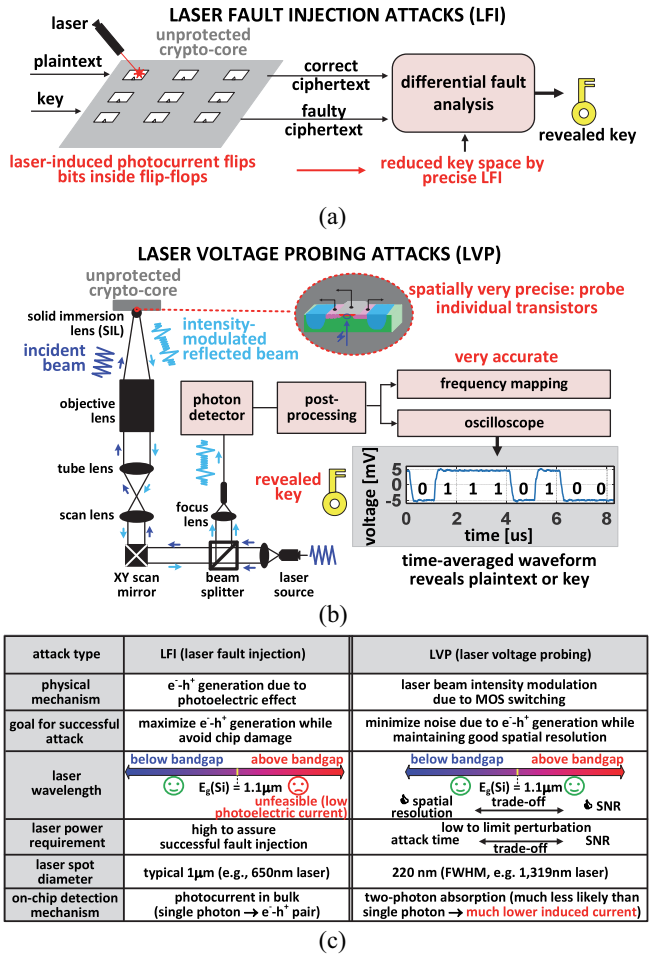
A low-cost solution for runtime supply impedance monitoring of its both real and imaginary part has been recently introduced in the form of a broadband runtime monitor [90]. The bandwidth from dc to 2 GHz covers the impedance resonance peaks at the PCB, packaging and bonding level, allowing end-to-end supply physical context awareness. The fully digital architecture based on ring oscillator voltage sensing and variable-frequency current excitation makes the impedance monitor technology and design portable [90]. The effect of global process variations, moderately fast voltage fluctuations, and temperature variations is suppressed through ratiometric acquisitions with and without current perturbation, when exciting the supply impedance for its quantification. As shown in Table 2, the solution in [90] eliminates the restriction to resistance [87] and inductance monitoring [88], and expands the bandwidth by 2.5× over [89]. Compared to solutions suitable for supply monitoring, the normalized area of [90] is close to [87], and 21 × smaller than [89]. In particular, its area fits a pair of supply pads and, hence, allows its integration underneath the supply pads, introducing near-zero area overhead for ubiquitous adoption.

Regarding EM attacks, passive techniques based on routing discipline have been recently introduced to vastly increase the necessary attack effort [91]. Indeed, lower metal routing and upper metal shielding drastically reduce the radiation of the EM fields correlated to the secret being processed during encryption. This translates into at least five to six orders of magnitude heavier attack effort [91], [92], as evidence of the effectiveness of physical-level counteraction. As an alternative approach to be used when routing discipline is not acceptable, active techniques for EM probe detection have also been demonstrated [93]. In detail, the EM probe detection circuit in [94] is based on a fully digital sensor circuit with reference-free dual-coil sensing, as well as a ring-oscillator-based sensor calibration with percentage point-range area overhead and intermittent operation for lower power. The sensor is suited for short range detection and, hence, when the probe is in proximity of the chip under attack.

A related sensor class for detecting EM fault injection attacks has also been widely investigated [95], [96], [97], [98], aiming to detect external field radiation with intense fields. Compared to the above EM attack sensors, these sensors have a much more relaxed sensitivity requirement and are hence well established in terms of technological development. Accordingly, the challenge in the foreseeable future is to enable increasingly reliable and higher sensitivity detection of EM attacks, rather than EM fault injection.

VII. ON-CHIP SENSORIZATION FOR INVASIVE PHYSICAL ATTACK DETECTION: LASER VOLTAGE PROBING CASE STUDY

Compared to the side-channel attacks discussed in Section VI, invasive attacks lie at the other end of the level of sophistication and effectiveness in physical threats. As class of particularly insidious threats, laser beams are very effective in terms of both laser fault injection (LFI) [99] and laser voltage probing (LVP) capabilities [100], [101], as illustrated in Fig. 8(a)–(c). LFI attacks aim to induce faults into logic through exposure of individual cells to a high-power laser beam (e.g., bit flip in registers). LFI attacks allow a fine and strict control down to individual bits, compared to other fault injection attacks leveraging clock, voltage, or timing. In contrast, LVP attacks aim to read individual bits by shining a highly focused laser beam that does not upset the state of the cell(s) being attacked. Hence, LVP attacks take place at much lower power levels and above-bandgap wavelengths as opposed to LFI attacks, as necessary to prevent photon absorption in the circuitry under attack. The data-dependent optical signal reflected by individual transistors



attack type	LFI (laser fault injection)	LVP (laser voltage probing)
physical mechanism	e^--h^+ generation due to photoelectric effect	laser beam intensity modulation due to MOS switching
goal for successful attack	maximize e^--h^+ generation while avoid chip damage	minimize noise due to e^--h^+ generation while maintaining good spatial resolution
laser wavelength	below bandgap / above bandgap; E_g(Si) ≈ 1.1µm; unfeasible (low photoelectric current)	below bandgap / above bandgap; E_g(Si) ≈ 1.1µm; spatial resolution trade-off SNR
laser power requirement	high to assure successful fault injection	low to limit perturbation attack time trade-off SNR
laser spot diameter	typical 1µm (e.g., 650nm laser)	220 nm (FWHM, e.g. 1,319nm laser)
on-chip detection mechanism	photocurrent in bulk (single photon → e^--h^+ pair)	two-photon absorption (much less likely than single photon → much lower induced current)

(c)

FIGURE 8. Pictorial description of (a) laser fault injection attacks (LFI), (b) laser voltage probing attacks (LVP), along with (c) their comparison.

is then read out as in Fig. 8(b). Significant digital postprocessing is required to uncover the signal from the underlying noise (e.g., incorrelated reflections). In either case, laser-based attacks are now quite inexpensive due to the steadily lowering cost of equipment and the relatively short time required for the attack.[2]

The low cost and the high effectiveness of laser-based attacks make them a much more concerning threat than just a few years ago. Accordingly, the availability of intelligent on-chip sensors for laser-based attacks is becoming a requirement in a wider range of silicon systems, and is expected to continue to expand down to low-end devices. As pointed out above, on-chip sensors to detect laser fault injection have a rather relaxed sensitivity requirement, compared to those for laser voltage probing. This makes LVP attack detection much harder than LFI attacks, and currently represents the main challenge in the area of laser attacks.

2. For example, the most expensive piece of equipment for LVP attacks (more costly than LFI) is the SEM microscope, whose cost has dropped to less than $20 000 for used equipment. Attack execution in reliability and failure analysis labs typically takes hours or few tens of hours at a typical hourly cost of $1000.

On-chip detection of LFI attacks is now well established [102], [103], and exploits mechanisms at the logical or the physical level. In particular, the approach in [102] detects LFI attacks by duplicating portions of the encryption data path and checking data consistency. Although this comes at a nearly doubled area cost, duplication of logic also enforces data independence in the power patterns, thus improving the resistance against side-channel attacks. On the other hand, [103] is based on distributed bulk built-in current sensors (BBICS) with sparse placement for moderate area overhead (28%). Sparse placement with full area coverage is enabled by the nature of the laser-silicon interaction at wavelengths below the bandgap (1.1 μm), which are invariably used in LFI attacks to significantly perturb circuit operation and induce bitflips in flip-flops. Indeed, a laser beam at sub-bandgap wavelengths determines abundant generation of electron–hole pairs, which disperse in the common substrate of the cryptographic core. Hence, they are readily caught with BBICS sensors detecting anomalous bulk currents, and can be placed near body/well taps (i.e., at a pitch of few tens of μm).

Regarding LVP attacks, their main challenge lies in the fact that circuit operation needs to be largely unperturbed through the adoption of much lower laser power, above-bandgap wavelengths, and precise laser spot (down to single transistor) [100], [101]. In LVP attacks, the laser beam hitting the die backside is scattered by the transistors within the targeted standard cell depending on the digital cell output, which is hence highly correlated with the reflected light. Time averaging across several acquisitions recovers the necessary signal-to-noise ratio for successful retrieval of the bit under attack. Since LVP attacks can use above-bandgap wavelengths, the generation of electron–hole pairs is typically reduced by at least four orders of magnitude compared to subbandgap wavelengths in LFI attacks in Fig. 8(c). This drastically tightens the sensor sensitivity requirement, making BBICS unusable and preventing any significant diffusion of electron–hole pairs. Accordingly, full area coverage requires truly in-situ sensors that are densely distributed within a laser spot from every single transistor. Also, like LFI detection, sensors for LVP attacks need to be embedded within the logic being protected and, hence, must be compatible with standard cell-based design flows.

Various techniques to detect LVP attacks have been investigated to date [104], [105], [106], [107], [108]. Some of them require major manufacturing process modifications [104], [105], and are hence unsuitable for most mass-produced silicon products. Simple ring oscillators have been exploited as standard cell-based LVP sensors to monitor their surroundings, although their short sensing range and significant power and area penalty make full area coverage unfeasible [106], [107].

LVP attack detection with always-on full area coverage under standard cell-based sensors has been recently enabled in [108], as illustrated in Fig. 9(a) and (b). This scheme is based on the creation of a standard cell library where each

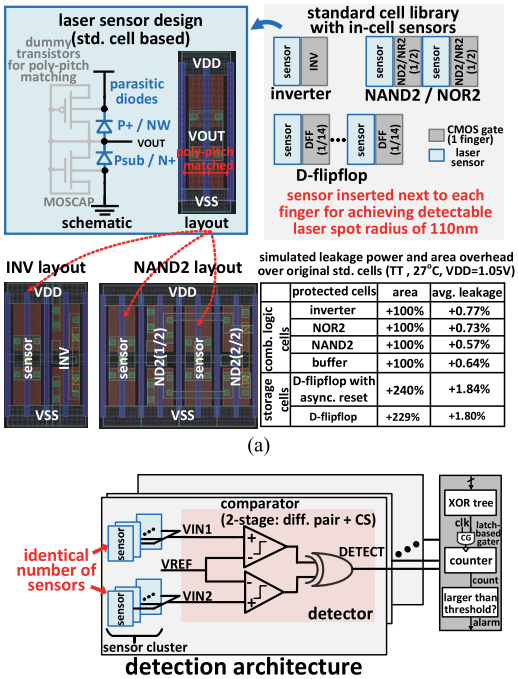


FIGURE 9. Always-on standard cell-based LVP sensors with full area coverage in [108]: (a) layout architecture of LVP-aware standard cell library with interspersed sensors and (b) aggregation of sensor signals across neighboring cells.

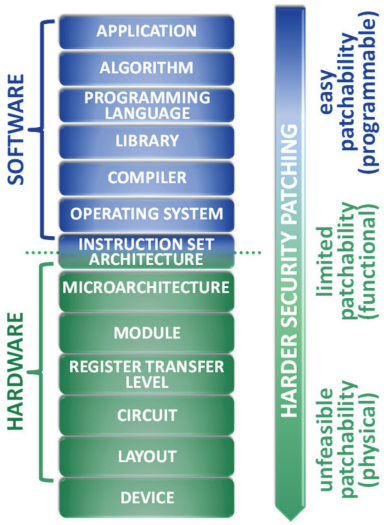


FIGURE 10. Security vulnerability patching over the system lifespan is easy at software levels of abstraction, hard toward the physical level, and limited at intermediate levels (e.g., via reconfigurable logic).

cell contains photosensors (i.e., pn junctions) at every other transistor finger to sense the laser beam at any location. The sensor embedment is fully compatible with restricted design rules, and is hence scalable to advanced technologies. Multiple sensor outputs (e.g., 100) are then aggregated as in Fig. 9(b) through a detector cell, and further aggregated through a simple logic tree to generate a single LVP attack detection flag. The approximately doubled area is reduced when full coverage is necessary only in portions of the digital subsystem (e.g., crypto-core, rather than the entire system). The effect of process, voltage, and temperature variations is rejected by the symmetric and differential nature of sensor aggregation. The sensitivity of the sensors is sufficient to capture any laser beam with intensity leading to a successful attack.

In summary, laser-based attacks and, most importantly, LVP attacks have become inexpensive and are extremely effective in revealing on-chip secrets. Hence, they require detectors that are always-on and provide full area coverage, while being compatible with standard cell flows. The main challenge in the coming years lies in the development of new LVP attack sensing techniques that reduce the area overhead ($> 2\times$ currently) to make it affordable in low-end devices.

VIII. HARDWARE PATCHING ENABLEMENT VIA ON-CHIP MACHINE LEARNING: CASE STUDY

Traditionally, system security upgradeability fundamentally depends on the level of abstraction that the underlying reconfiguration targets, as summarized in Fig. 10. At the highest level, software update routinely allows to fix vulnerabilities discovered over the system lifespan. Also, limited functional upgradeability can often be allowed via reconfigurable logic or firmware update. On the other hand, physical reconfiguration is generally unfeasible, due to the lack of runtime flexibility of the circuit implementation. From an attack viewpoint, this translates into today's massively deployed software patching capabilities, as well as nonexistent hardware and physical patching. Equivalently, vulnerabilities involving software levels can be fixed over time, improving the level of system security over time. In contrast, physical (e.g., side-channel) vulnerabilities discovered simply accumulate over time, limiting system usability over time and, hence, its lifespan under practical security requirements.

From the above considerations, sustainable security assurance over time would require the enablement of hardware patchability. The latter is expected to become a main goal and driver of innovation in hardware security in the coming years, although it is largely unavailable in present systems on chip. To exemplify the key principles that hardware patching needs to follow, a first demonstration of physical patching [92] is here discussed in the context of side-channel analysis attacks (see considerations in Section II).

In prior art, side-channel attack counteraction techniques can be classified as in Fig. 11. Design-specific solutions need to be redesigned to protect any new design, as typical of early counteraction techniques [109], [110], [111] (e.g., dual-rail precharge logic). More recent solutions are now design-reusable in that the same counteraction design IP can be just reutilized to protect different designs (e.g., voltage regulation with embedded counteraction), although it cannot be upgraded over time [91], [112], [113], [114], [115], [116], [117].

As recently proposed class of counteraction techniques, design-adaptive solutions enable post-silicon upgrade and

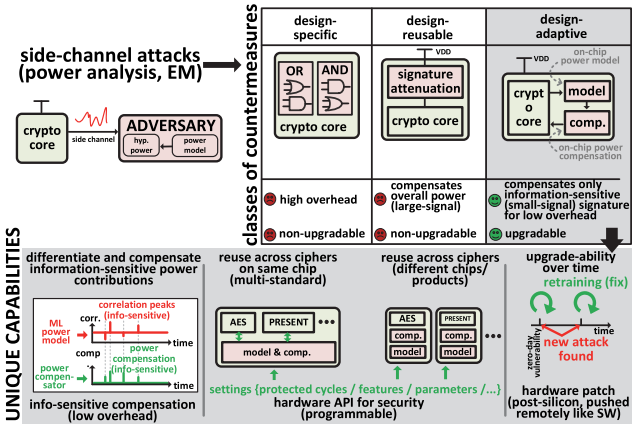


FIGURE 11. Hardware patching for side-channel security vulnerability fix requires adaptation over time (design adaptive), beyond traditional and nonupgradeable design-specific and design-reusable countermeasures [92].

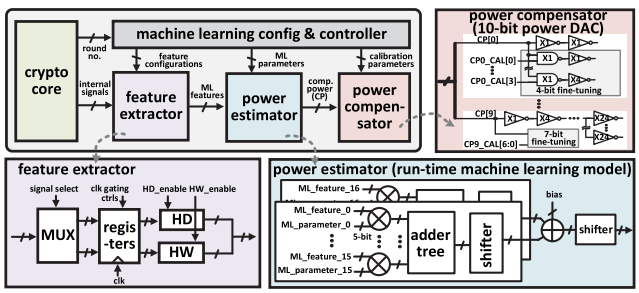


FIGURE 12. General architecture of side-channel counteraction with hardware patching, as enabled by a machine-learning-based power estimator [92].

hardware patching against side-channel attacks [92]. Design-adaptive techniques allow reuse across single/multiciphers, standard updates, and targeted compensation of information-sensitive power contributions to minimize the power overhead. Their general architecture in Fig. 12 is centered around a lightweight runtime on-chip machine learning power model [92] (e.g., linear regression, being power linear with activity). The latter is driven by a feature extractor and drives a power compensator (i.e., a power DAC). Post-silicon improvements are supported by weight updates, which are trained to counteract a set of attacks and known vulnerabilities, and are then massively distributed throughout the connected devices. The architecture in Fig. 12 is fully digital and implemented with standard cell-based flows for ubiquitous adoption. The counteraction technique in [92] was shown to increase the attack effort to best-in-class levels (>1.2 billion power traces necessary to retrieve the secret key) across different ciphers and different microarchitectural designs of the same algorithm. The counteraction of new vulnerabilities was demonstrated for a newly found attack to the PRESENT lightweight cryptographic algorithm, showing that weight update simultaneously protects the system from previous and newly discovered side-channel attacks.

In summary, on-chip machine learning algorithms can serve as runtime threat models that can be reused across designs to reduce design costs [92]. The amenability for standard cell design enables low-effort implementation, seamless integration with the logic to be protected, as well as design portability. At the same time, machine learning modeling allows to compensate only the truly information-sensitive power contributions, pushing conventional large-signal to small-signal (low-power) compensation. More importantly, it uniquely enables hardware patching for sustainable security assurance and lifespan extension, and even more so when system replacement is unaffordable.

IX. CONCLUSION

In this work, an overview of trends, challenges and promising directions to support ubiquitous and sustainable hardware security in connected systems has been presented. At the root of trust level, ubiquitous and economically sustainable security is pursued through aggressive design and resource reuse (e.g., in-memory and immersed-in-logic primitives). At the on-chip sensing level, physical context awareness is mandated by zero-trust frameworks, and requires low-cost intelligent sensors implementable with standard cell methodologies. Sustainable security over time requires design reuse over time and, hence, hardware patching capabilities, as enabled by machine learning-based architectures.

In perspective, such challenges will become even more interesting in the context of heterogeneous integration, where solutions are naturally broken down into multiple silicon dice.

ACKNOWLEDGMENT

The author would like to thank TSMC for chip fabrication support.

REFERENCES

[1] M. Alioto, "Trends in hardware security: From basics to ASICs," *IEEE Solid-State Circuits Mag.*, vol. 11, no. 3, pp. 56–74, Aug. 2019.
[2] M. Alioto, "F6: Computer systems under attack—Paying the performance price for protection," in *Proc. ISSCC*, San Francisco, CA, USA, Feb. 2022, pp. 543–546.
[3] S. Taneja and M. Alioto, *In-Memory and Immersed-In-Logic Primitives for Ubiquitous Hardware Security*. Cham, Switzerland: Springer, 2023.
[4] C. Brooks. "Alarming cyber statistics for mid-year 2022 that you need to know." Forbes. Jun. 2022. [Online]. Available: https://www.forbes.com/sites/chuckbrooks/2022/06/03/alarming-cyber-statistics-for-mid-year-2022-that-you-need-to-know/?sh=71777c747864
[5] "Cost of a data breach report 2021." IBM Security Report. 2021. [Online]. Available: https://www.ibm.com/security/data-breach
[6] S. Morgan. "Cybercrime To cost the world $10.5 trillion annually by 2025." Special Report: Cyberwarfare In The C-Suite. Nov. 2020. [Online]. Available: https://cybersecurityventures.com/cybercrime-damages-6-trillion-by-2021
[7] "Securing your journey to hybrid multicloud—Protecting workloads to enable business innovation and growth." IBM Digital Assets. 2019. [Online]. Available: https://www.ibm.com/security/digital-assets/hybrid-multicloud-ebook
[8] P. Sparks, "The route to a trillion devices—The Outlook for IoT investment to 2035," ARM Ltd., Cambridge, U.K., White Paper, Jun. 2017. [Online]. Available: https://community.arm.com/cfs-file/__key/telligent-evolution-components-attachments/01-1996-00-00-00-01-30-09/Arm-_2D00_-The-route-to-a-trillion-devices-_2D00_-June-2017.pdf
[9] M. Alioto, Ed., *Enabling the Internet of Things—From Integrated Circuits to Integrated Systems*. Cham, Switzerland: Springer, 2017.
[10] M. Alioto, "Hardware security—From basics to ASICs," in *Proc. ISSCC*, San Francisco, CA, USA, Feb. 2019.

[11] A. Weinert et al., "Traditional perimeter-based network defense is obsolete—Transform to a zero trust model," Microsoft, Redmond, WA, USA, White Paper, Oct. 2019. [Online]. Available: https://www.microsoft.com/security/blog/2019/10/23/perimeter-based-network-defense-transform-zero-trust-model

[12] "Zero trust—Cybersecurity for the Internet of Things," Microsoft, Redmond, WA, USA, White Paper, 2021. [Online]. Available: https://azure.microsoft.com/en-us/resources/zero-trust-cybersecurity-for-the-internet-of-things

[13] "Zero trust security solutions." IBM Webpage. 2022. [Online]. Available: https://www.ibm.com/security/zero-trust

[14] A. Borkar. "When implementing zero trust, context is everything." CISO. May 2020. [Online]. Available: https://securityintelligence.com/posts/when-implementing-zero-trust-context-is-everything

[15] M. Alioto, "From less batteries to battery-less integrated systems through ultra-wide power-performance adaptation down to pWs," *IEEE Design Test*, vol. 38, no. 5, pp. 90–133, Oct. 2021.

[16] M. Alioto. "HW security primitives database." 2020. [Online]. Available: http://www.green-ic.org/hwsecdb

[17] Y. Su, J. Holleman, and B. Otis, "A 1.6pJ/bit 96% stable chip-ID generating circuit using process variations," in *ISSCC Dig. Tech. Papers*, 2007, pp. 406–408.

[18] S. Chellappa, A. Dey, and L. T. Clark, "Improved circuits for microchip identification using SRAM mismatch," in *Proc. CICC*, 2011, pp. 3–6.

[19] D. Fainstein, S. Rosenblatt, A. Cestero, N. Robson, T. Kirihata, and S. S. Iyer, "Dynamic intrinsic chip ID using 32nm high-K/metal gate SOI embedded DRAM," in *Proc. IEEE Symp. VLSI Circuits*, vol. 128, 2012, pp. 146–147.

[20] K. Liu, Y. Min, X. Yang, H. Sun, and H. Shinohara, "A 373-F^2 0.21%-native-BER EE SRAM physically unclonable function with 2-D power-gated bit cells and V_{SS} bias-based dark-bit detection," *IEEE J. Solid-State Circuits*, vol. 55, no. 6, pp. 1719–1732, Jun. 2020.

[21] K. Liu, X. Chen, H. Pu, and H. Shinohara, "A 0.5-V hybrid SRAM physically unclonable function using hot carrier injection burn-in for stability reinforcement," *IEEE J. Solid-State Circuits*, vol. 56, no. 7, pp. 2193–2204, Jul. 2021.

[22] S. Taneja, V. K. Rajanna, and M. Alioto, "36.1 unified in-memory dynamic TRNG and multi-bit static PUF entropy generation for ubiquitous hardware security," in *ISSCC Dig. Tech. Papers*, Feb. 2021, pp. 498–500.

[23] J. Song et al., "A 3T eDRAM in-memory physically unclonable function with spatial majority voting stabilization," *IEEE Solid-State Circuits Lett.*, vol. 5, pp. 58–61, 2022.

[24] K. Lofstrom, W. R. Daasch, and D. Taylor, "IC identification circuit using device mismatch," in *ISSCC Dig. Tech. Papers*, 2000, pp. 372–373.

[25] S. Stanzione, D. Puntin, and G. Iannaccone, "CMOS silicon physical unclonable functions based on intrinsic process variability," *IEEE J. Solid-State Circuits*, vol. 46, no. 6, pp. 1456–1463, Jun. 2011.

[26] J. Li and M. Seok, "A 3.07μm^2/bitcell physically unclonable function with 3.5% and 1% bit-instability across 0 to 80°C and 0.6 to 1.2V in a 65nm CMOS," in *Proc. IEEE Symp. VLSI Circuits*, 2015, pp. 250–251.

[27] J. Lee, D. Lee, Y. Lee, and Y. Lee, "A 445F^2 leakage-based physically unclonable function with lossless stabilization through remapping for IoT security," in *ISSCC Dig. Tech. Papers*, 2018, pp. 132–134.

[28] A. Alvarez, W. Zhao, and M. Alioto, "15 fJ/bit static physically unclonable functions for secure chip identification with < 2% native bit instability and 140x inter/intra PUF hamming distance separation in 65nm," in *ISSCC Dig. Tech. Papers*, 2015, pp. 256–258.

[29] K. Yang, Q. Dong, D. Blaauw, and D. Sylvester, "A 553F^2 2-transistor amplifier-based physically unclonable function (PUF) with 1.67% native instability," in *ISSCC Dig. Tech. Papers*, 2017, pp. 160–162.

[30] S. Taneja, A. B. Alvarez, and M. Alioto, "Fully synthesizable PUF featuring hysteresis and temperature compensation for 3.2% native BER and 1.02 fJ/b in 40 nm," *IEEE J. Solid-State Circuits*, vol. 53, no. 10, pp. 2828–2839, Oct. 2018.

[31] D. Li and K. Yang, "A 562F^2 physically unclonable function with a zero-overhead stabilization scheme," in *ISSCC Dig. Tech. Papers*, San Francisco, CA, USA, 2019, pp. 400–401.

[32] Y. Choi et al., "Physically unclonable function in 28nm fdsoi technology achieving high reliability for aec-q 100 grade 1 and iso 26262 asil-b," in *ISSCC Dig. Tech. Papers*, San Francisco, CA, USA, 2020, pp. 426–428.

[33] S. Taneja and M. Alioto, "PUF architecture with run-time adaptation for resilient and energy-efficient key generation via sensor fusion," *IEEE J. Solid-State Circuits*, vol. 56, no. 7, pp. 2182–2192, Jul. 2021.

[34] Y. He, D. Li, Z. Yu, and K. Yang, "36.5 An automatic self-checking and healing physically unclonable function (PUF) with <3 × 10-8 bit error rate," in *ISSCC Dig. Tech. Papers*, San Francisco, CA, USA, 2021, pp. 506–508.

[35] 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.

[36] H. Fujiwara, M. Yabuuchi, H. Nakano, H. Kawai, K. Nii, and K. Arimoto, "A chip-ID generating circuit for dependable LSI using random address errors on embedded SRAM and on-chip memory BIST," in *IEEE Symp. VLSI Circuits Dig. Tech.*, vol. 5, 2011, pp. 76–77.

[37] S. K. Mathew et al., "16.2 A 0.19pJ/b PVT-variation-tolerant hybrid physically unclonable function circuit for 100% stable secure key generation in 22nm CMOS," in *ISSCC Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2014, pp. 278–280.

[38] S. Satpathy et al., "13fJ/bit probing-resilient 250K PUF array with soft darkbit masking for 1.94% bit-error in 22nm tri-gate CMOS," in *Proc. ESSCIRC*, 2014, pp. 239–242.

[39] S. Mathew et al., "A 4fJ/bit delay-hardened physically unclonable function circuit with selective bit destabilization in 14nm tri-gate CMOS," in *Proc. IEEE Symp. VLSI Circuits*, 2016, pp. 248–249.

[40] S. K. Satpathy et al., "An all-digital unified physically unclonable function and true random number generator featuring self-calibrating hierarchical von neumann extraction in 14-nm tri-gate CMOS," *IEEE J. Solid-State Circuits*, vol. 54, no. 4, pp. 1074–1085, Apr. 2019.

[41] 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 *IEEE Symp. VLSI Circuits Dig. Tech.*, 2004, pp. 176–179.

[42] K. Yang, Q. Dong, D. Blaauw, and D. Sylvester, "A physically unclonable function with BER < 10^{-8} for robust chip authentication using oscillator collapse in 40nm CMOS," in *ISSCC Dig. Tech. Papers*, 2015, pp. 254–256.

[43] Y. Hori, T. Katashita, and Y. Ogasahara, "A 65-nm SOTB implementation of a physically unclonable function and its performance improvement by body bias control," in *Proc. IEEE SOI-3D-Subthreshold Microelectron. Unified Conf. (S3S)*, Mar. 2018, pp. 1–3.

[44] Z.-Y. Liang, H.-H. Wei, and T.-T. Liu, "A wide-range variation-resilient physically unclonable function in 28 nm," *IEEE J. Solid-State Circuits*, vol. 55, no. 3, pp. 817–825, Mar. 2020.

[45] M.-Y. Wu et al., "A PUF scheme using competing oxide rupture with bit error rate approaching zero," in *ISSCC Dig. Tech. Papers*, San Francisco, CA, USA, 2018, pp. 130–132.

[46] B. Gao, B. Lin, X. Li, J. Tang, H. Qian, and H. Wu, "A unified PUF and TRNG design based on 40-nm RRAM with high entropy and robustness for IoT security," *IEEE Trans. Electron Devices*, vol. 69, no. 2, pp. 536–542, Feb. 2022.

[47] Y. Yoshimoto, Y. Katoh, S. Ogasahara, Z. Wei, and K. Kouno, "A ReRAM-based physically unclonable function with bit error rate < 0.5% after 10 years at 125°C for 40nm embedded application," in *Proc. Symp. VLSI Technol.*, 2016, pp. 256–257.

[48] Y. Pang et al., "A reconfigurable RRAM PUF utilizing post-process randomness source with <6×10-6 N-BER," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, San Francisco, CA, USA, 2019, pp. 402–403.

[49] 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.

[50] Y. Wang et al., "A homogeneous, reconfigurable, and efficient implementation of PUF in 3-D selector-free RRAM," *IEEE Trans. Electron Devices*, vol. 68, no. 5, pp. 2577–2581, May 2021.

[51] S. K. Mathew et al., "2.4 Gbps, 7 mW all-digital PVT-variation tolerant true random number generator for 45 nm CMOS high-performance microprocessors," *IEEE J. Solid-State Circuits*, vol. 47, no. 11, pp. 2807–2821, Nov. 2012.

[52] V. R. Pamula, X. Sun, S. Kim, F. ur Rahman, B. Zhang, V. S. Sathe, "An all-digital true-random-number generator with integrated de-correlation and bias correction at 3.2-to-86 MB/s, 2.58 pJ/bit in 65-nm CMOS," in *Proc. IEEE Symp. VLSI Circuits*, Honolulu, HI, USA, Ju. 2018, pp. 1–2.

[53] S. Taneja and M. Alioto, "Fully synthesizable unified true random number generator and cryptographic core," *IEEE J. Solid-State Circuits*, vol. 56, no. 10, pp. 3049–3061, Oct. 2021.

[54] A. Olgun et al., "QUAC-TRNG: High-throughput true random number generation using quadruple row activation in commodity DRAM chips," in *Proc. ISCA*, Jun. 2021, pp. 944–957.

[55] R. Zhang et al., "A 0.186-pJ per bit latch-based true random number generator with mismatch compensation and random noise enhancement," in *Proc. IEEE Symp. VLSI Circuits*, Honolulu, HI, USA, Jun. 2021, pp. 1–2.

[56] X. Wang, R. Zhang, Y. Wang, K. Liu, X. Wang, and H. Shinohara, "A 0.116pJ/bit latch-based true random number generator with static inverter selection and noise enhancement," in *Proc. VLSI-DAT*, Apr. 2022, pp. 1–4.

[57] J. Park, B. Kim, and J.-Y. Sim, "A PVT-tolerant oscillation-collapse-based true random number generator with an odd number of inverter stages," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 69, no. 10, pp. 4058–4062, Oct. 2022.

[58] Y. Cao, X. Zhao, W. Zheng, Y. Zheng, and C.-H. Chang, "A new energy-efficient and high throughput two-phase multi-bit per cycle ring oscillator-based true random number generator," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 69, no. 1, pp. 272–283, Jan. 2022.

[59] X. Cheng et al., "A feedback architecture of high speed true random number generator based on ring oscillator," in *Proc. IEEE ASSCC*, Nov. 2021, pp. 1–3.

[60] S. T. Chandrasekaran, V. E. G. Karnam, and A. Sanyal, "0.36-mW, 52-Mb/s true random number generator based on a stochastic Delta–Sigma modulator," *IEEE Solid-State Circuits Lett.*, vol. 3, pp. 190–193, 2020.

[61] E. Kim, M. Lee, and J. Kim, "8.2 8Mb/s 28Mb/mJ robust true-random-number generator in 65nm CMOS based on differential ring oscillator with feedback resistors," in *ISSCC Dig. Tech. Papers*, Feb. 2017, pp. 144–145.

[62] K. Yang, D. Blaauw, and D. Sylvester, "An all-digital edge racing true random number generator robust against PVT variations," *IEEE J. Solid-State Circuits*, vol. 51, no. 4, pp. 1022–1031, Apr. 2016.

[63] T.-K. Kuan, Y.-H. Chiang, and S.-I. Liu, "A 0.43pJ/bit true random number generator," in *Proc. ASSCC*, Kaohsiung City, Taiwan, 2014, pp. 33–36.

[64] K. Yang, D. Fick, M. B. Henry, Y. Lee, D. Blaauw, and D. Sylvester, "16.3 A 23Mb/s 23pJ/b fully synthesized true-random number generator in 28nm and 65nm CMOS," in *ISSCC Dig. Tech. Papers*, Feb. 2014, pp. 280–281.

[65] M. Bucci and R. Luzzi, "Fully digital random bit generators for cryptographic applications," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 55, no. 3, pp. 861–875, Apr. 2008.

[66] M. Bucci, L. Germani, R. Luzzi, A. Trifiletti, and M. Varanonuovo, "A high-speed oscillator-based truly random number source for cryptographic applications on a smart card IC," *IEEE Trans. Comput.*, vol. 52, no. 4, pp. 403–409, Apr. 2003.

[67] F. Pareschi, G. Setti, and R. Rovatti, "A fast chaos-based true random number generator for cryptographic applications," in *Proc. ESSCIRC*, Montreux, Switzerland, Sep. 2006, pp. 130–133.

[68] S. N. Dhanuskodi, A. Vijayakumar, and S. Kundu, "A chaotic ring oscillator based random number generator," in *Proc. HOST*, 2014, pp. 160–165.

[69] M. Kim, U. Ha, K. J. Lee, Y. Lee, and H.-J. Yoo, "A 82-nW chaotic map true random number generator based on a sub-ranging SAR ADC," *IEEE J. Solid-State Circuits*, vol. 52, no. 7, pp. 1953–1965, Jul. 2017.

[70] A. T. Do and X. Liu, "25 fJ/bit, 5Mb/s, 0.3 V true random number generator with capacitively-coupled chaos system and dual-edge sampling scheme," in *Proc. ASSCC*, Seoul, South Korea, 2017, pp. 61–64.

[71] V. von Kaenel and T. Takayanagi, "Dual true random number generators for cryptographic applications embedded on a 200 million device dual CPU SoC," in *Proc. CICC*, 2007, pp. 269–272.

[72] C. Tokunaga, D. Blaauw, and T. Mudge, "True random number generator with a metastability-based quality control," *IEEE J. Solid-State Circuits*, vol. 43, no. 1, pp. 78–85, Jan. 2008.

[73] S. K. Mathew et al., "μRNG: A 300–950 mV, 323 Gb/s/W all-digital full-entropy true random number generator in 14 nm FinFET CMOS," *IEEE J. Solid-State Circuits*, vol. 51, no. 7, pp. 1695–1704, Jul. 2016.

[74] S.-G. Bae, Y. Kim, Y. Park, and C. Kim, "3-Gb/s high-speed true random number generator using common-mode operating comparator and sampling uncertainty of D flip-flop," *IEEE J. Solid-state Circuits*, vol. 52, no. 2, pp. 605–610, Feb. 2017.

[75] K. Yang et al., "A 28nm integrated true random number generator harvesting entropy from MRAM," in *Proc. IEEE Symp. VLSI Circuits*, Honolulu, HI, USA, 2018, pp. 171–172.

[76] N. Massari et al., "A 16×16 pixels SPAD-based 128-Mb/s quantum random number generator with −74dB light rejection ratio and −6.7ppm/C bias sensitivity on temperature," in *ISSCC Dig. Tech. Papers*, 2016, pp. 292–293.

[77] N. Liu, N. Pinckney, S. Hanson, D. Sylvester, and D. Blaauw, "A true random number generator using time-dependent dielectric breakdown," in *Symp. VLSI Circuits Dig. Tech.*, 2011, pp. 216–217.

[78] S. Yasuda, H. Satake, T. Tanamoto, R. Ohba, K. Uchida, and S. Fujita, "Physical random number generator based on MOS structure after soft breakdown," *IEEE J. Solid-State Circuits*, vol. 39, no. 8, pp. 1375–1377, Aug. 2004.

[79] R. Brederlow, R. Prakash, C. Paulus, and R. Thewes, "A low-power true random number generator using random telegraph noise of single oxide-traps," in *IEEE ISSCC Dig. Tech. Papers*, Feb. 2006, pp. 1666–1675.

[80] M. Matsumoto, S. Yasuda, R. Ohba, K. Ikegami, T. Tanamoto, and S. Fujita, "1200μm^2 physical random-number generators based on SiN MOSFET for secure smart-card application," in *ISSCC Dig. Tech. Papers*, Feb. 2008, pp. 414–415.

[81] Q. Ding et al., "Unified 0.75pJ/bit TRNG and attack resilient 2F2/Bit PUF for robust hardware security solutions with 4-layer stacking 3D NbOx threshold switching array," in *Proc. IEDM*, San Francisco, CA, USA, Dec. 2021, pp. 1–4.

[82] J. Yang et al., "A calibration-free in-memory true random number generator using voltage-controlled MRAM," in *Proc. ESSCIRC*, Sep. 2021, pp. 115–118.

[83] J. Li, T. Yang, M. Yang, P. R. Kinget, and M. Seok, "An area-efficient microprocessor-based soc with an instruction-cache transformable to an ambient temperature sensor and a physically unclonable function," *IEEE J. Solid-State Circuits*, vol. 53, no. 3, pp. 728–737, Mar. 2018.

[84] K.-H. Chuang et al., "A multi-bit/cell PUF using analog breakdown positions in CMOS," in *Proc. IEEE Int. Rel. Phys. Symp.*, Mar. 2018, pp. 1–5.

[85] B. Karpinskyy, Y. Lee, Y. Choi, Y. Kim, M. Noh, and S. Lee, "8.7 Physically unclonable function for secure key generation with a key error rate of 2E-38 in 45nm smart-card chips," in *ISSCC Dig. Tech. Papers*, 2016, pp. 158–160.

[86] "Hacking at home for $0-$250." NewAR Technology Inc. Feb. 2021. [Online]. Available: https://www.newae.com/post/hacking-at-home

[87] S. J. Kim, D. Kim, A. Sharma, and M. Seok, "EQZ-LDO: A near-zero EDP overhead, >10M-attack-resilient, secure digital LDO featuring attack-detection and detection-driven protection for a correlation-power-analysis-resilient IoT device," in *Proc. IEEE Symp. VLSI Circuits*, Jun. 2021, pp. 1–2.

[88] N. Miura, M. Takahashi, K. Nagatomo, and M. Nagata, "Chip-package-board interactive PUF utilizing coupled chaos oscillators with inductor," *IEEE J. Solid-State Circuits*, vol. 53, no. 10, pp. 2889–2897, Oct. 2018.

[89] P. N. Whatmough, S. Das, Z. Hadjilambrou, and D. M. Bull, "Power integrity analysis of a 28 nm dual-core arm cortex-A57 cluster using an all-digital power delivery monitor," *IEEE J. Solid-State Circuits*, vol. 52, no. 6, pp. 1643–1654, Jun. 2017.

[90] V. K. Rajanna, H. S. Raghav, T. Wang, and M. Alioto, "Fully-digital broadband calibration-less impedance monitor for probe insertion detection against power analysis attacks," in *Proc. IEEE Symp. VLSI Circuits*, Honolulu, HI, USA, Jun. 2022, pp. 140–141.

[91] D. Das et al., "27.3 EM and power SCA-resilient AES-256 in 65nm CMOS through >350× current-domain signature attenuation," in *ISSCC Dig. Tech. Papers*, Feb. 2020, pp. 424–425.
[92] Q. Fang, L. Lin, Y. Z. Wong, H. Zhang, and M. Alioto, "Side-channel attack counteraction via machine learning-targeted power compensation for post-silicon HW security patching," in *IEEE ISSCC Dig. Tech. Papers*, Feb. 2022, pp. 516–517.
[93] M. Nagata, T. Miki, and N. Miura, "Physical attack protection techniques for IC chip level hardware security," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 30, no. 1, pp. 5–14, Jan. 2022.
[94] N. Miura et al., "A local EM-analysis attack resistant cryptographic engine with fully-digital oscillator-based tamper-access sensor," in *IEEE Symp. VLSI Circuits Dig. Tech.*, Jun. 2014, pp. 172–173.
[95] L. Zussa et al., "Efficiency of a glitch detector against electromagnetic fault injection," in *Proc. DATE*, 2014, pp. 1–6.
[96] D. El-Baze, J.-B. Rigaud, and P. Maurine, "An embedded digital sensor against EM and BB fault injection," in *Proc. Workshop Fault Diagnosis Tolerance Cryptogr. (FDTC)*, Aug. 2016, pp. 78–86.
[97] N. Miura et al., "PLL to the rescue: A novel EM fault countermeasure," in *Proc. DAC*, Jun. 2016, pp. 1–6.
[98] J. Breier, S. Bhasin, and W. He, "An electromagnetic fault injection sensor using hogge phase-detector," in *Proc. ISQED*, Mar. 2017, pp. 307–312.
[99] S. P. Skorobogatov and R. J. Anderson, "Optical fault induction attacks," in *Proc. CHES*, Aug. 2002, pp. 2–12.
[100] C. Boit, C. Helfmeier, and U. Kerst, "Security risks posed by modern IC debug and diagnosis tools," in *Proc. Workshop Fault Diagnos. Tolerance Cryptogr. (FDTC)*, Aug. 2013, pp. 3–11.
[101] U. Kindereit, G. Woods, J. Tian, U. Kerst, R. Leihkauf, and C. Boit, "Quantitative investigation of laser beam modulation in electrically active devices as used in laser voltage probing," *IEEE Trans. Device Mater. Rel.*, vol. 7, no. 1, pp. 19–30, Mar. 2007.
[102] M. Doulcier-Verdier, J.-M. Dutertre, J. Fournier, J.-B. Rigaud, B. Robisson, and A. Tria, "A side-channel and fault-attack resistant AES circuit working on duplicated complemented values," in *IEEE ISSCC Dig. Tech. Papers*, Feb. 2011, pp. 274–275.
[103] K. Matsuda et al., "A 286 F^2/cell distributed bulk-current sensor and secure flush code eraser against laser fault injection attack on cryptographic processor," *IEEE J. Solid-State Circuits*, vol. 53, no. 11, pp. 3174–3182, Nov. 2018.
[104] H. Shen, N. Asadizanjani, M. Tehranipoor, and D. Forte, "Nanopyramid: An optical scrambler against backside probing attacks," in *Proc. 44th Int. Symp. Test. Failure Anal. (ISTFA)*, 2018, p. 280.
[105] C. Boit et al., "From IC debug to hardware security risk: The power of backside access and optical interaction," in *Proc. Phys. Failure Anal. Integr. Circuits (IPFA)*, 2016, pp. 365–369.
[106] S. Tajik, J. Fietkau, H. Lohrke, J.-P. Seifert, and C. Boit, "PUFMon: Security monitoring of FPGAs using physically unclonable functions," in *Proc. Int. Symp. On-Line Test. Robust Syst. Des. (IOLTS)*, 2017, pp. 186–191.
[107] Y. Gao, H. Ma, D. Abbott, and S. F. Al-Sarawi, "PUF Sensor: Exploiting PUF unreliability for secure wireless sensing," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 64, no. 9, pp. 2532–2543, Sep. 2017.
[108] H. Zhang, L. Lin, Q. Fang, and M. Alioto, "On-chip laser voltage probing attack detection with 100% area coverage at above/below the bandgap wavelength and fully-automated design," in *Proc. IEEE Symp. VLSI Circuits*, Honolulu, HI, USA, Jun. 2022, pp. 144–145.
[109] T. Popp, M. Kirschbaum, T. Zefferer, and S. Mangard, "Evaluation of the masked logic style MDPL on a prototype chip," in *Proc. CHES*, 2007, pp. 81–94.
[110] D. D. Hwang et al., "AES-based security coprocessor IC in 0.18-μm CMOS with resistance to differential power analysis side-channel attacks," *IEEE J. Solid-State Circuits*, vol. 41, no. 4, pp. 781–792, Apr. 2006.
[111] W. Shan, S. Zhang, J. Xu, M. Lu, L. Shi, and J. Yang, "Machine learning assisted side-channel-attack countermeasure and its application on a 28-nm AES circuit," *IEEE J. Solid-State Circuits*, vol. 55, no. 3, pp. 794–804, Mar. 2020.
[112] C. Tokunaga and D. Blaauw, "Secure AES engine with a local switched-capacitor current equalizer," in *ISSCC Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2009, pp. 64–65.
[113] M. Kar, A. Singh, S. Mathew, A. Rajan, V. De, and S. Mukhopadhyay, "8.1 improved power-side-channel-attack resistance of an AES-128 core via a security-aware integrated buck voltage regulator," in *ISSCC Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2017, pp. 142–144.
[114] A. Singh, M. Kar, S. Mathew, A. Rajan, V. De, and S. Mukhopadhyay, "25.3 A 128b AES engine with higher resistance to power and electromagnetic side-channel attacks enabled by a security-aware integrated all-digital low-dropout regulator," in *ISSCC Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2019, pp. 404–406.
[115] R. Kumar et al., "A SCA-resistant AES engine in 14nm CMOS with time/frequency-domain leakage suppression using non-linear digital LDO cascaded with arithmetic countermeasures," in *Proc. IEEE Symp. VLSI Circuits*, Honolulu, HI, USA, Jun. 2020, pp. 1–2.
[116] Y. He and K. Yang, "25.3 A 65nm edge-chasing quantizer-based digital LDO featuring 4.58ps-FoM and side-channel-attack resistance," in *ISSCC Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2020, pp. 384–385.
[117] A. Ghosh, D. Das, J. Danial, V. De, S. Ghosh, and S. Sen, "36.2 An EM/power SCA-resilient AES-256 with synthesizable signature attenuation using digital-friendly current source and RO-bleed-based integrated local feedback and global switched-mode control," in *ISSCC Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2021, pp. 500–501.

MASSIMO ALIOTO (Fellow, IEEE) received the M.Sc. degree in electronics engineering and the Ph.D. degree in electrical engineering from the University of Catania, Catania, Italy, in 1997 and 2001, respectively.

He is a Professor with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore, where he leads the Green IC Group, the Integrated Circuits and Embedded Systems area, and the FD-FAbrICS industry-sponsored lab. Previously, he held positions with the University of Siena, Siena, Italy; Circuit Research Lab, Intel Labs, Hillsboro, OR, USA, in 2013; University of Michigan at Ann Arbor, Ann Arbor, MI, USA, from 2011 to 2012; Berkeley Wireless Research Center, University of California at Berkeley, Berkeley, CA, USA, from 2009 to 2011; and EPFL, Lausanne, Switzerland, in 2007. He has authored or coauthored 350 publications on journals and conference proceedings. He has coauthored four books, such as the popular *Enabling the Internet of Things—From Circuits to Systems* (Springer, 2017) with other two books being published. His primary research interests include self-powered wireless-integrated systems, green silicon systems, widely energy-scalable integrated systems, data-driven systems for machine intelligence, hardware security, and emerging technologies.

Prof. Alioto has been the Editor-in-Chief of the IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS since 2019, and was the Deputy Editor-in-Chief of the IEEE JOURNAL ON EMERGING AND SELECTED TOPICS IN CIRCUITS AND SYSTEMS. He is/was the Distinguished Lecturer of the IEEE Solid-State Circuits Society from 2020 to 2021 and the Circuits and Systems Society from 2022 to 2023 and from 2009 to 2010. For the latter, he was also a member of the Board of Governors from 2015 to 2020, and the Chair of the "VLSI Systems and Applications" Technical Committee from 2010 to 2012. In the last five years, he has given more than 50 invited talks in top conferences, universities, and leading semiconductor companies. He served as a Guest Editor of several IEEE journal special issues, such as the IEEE JOURNAL OF SOLID-STATE CIRCUITS, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS PART—I: REGULAR PAPERS, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS PART—II: EXPRESS BRIEFS, and IEEE JOURNAL ON EMERGING AND SELECTED TOPICS IN CIRCUITS AND SYSTEMS. He also serves or has served as an Associate Editor of a number of IEEE and ACM journals. He is/was the Technical Program Chair of ISCAS, APCCAS, SOCC, ICECS, NEWCAS, VARI, ICM, and PRIME and the Track Chair in a number of conferences, such as ICCD, ISCAS, ICECS, VLSI-SoC, APCCAS, and ICM. He is currently also in the IEEE "Digital Architectures and Systems" ISSCC Subcommittee, and the IEEE ASSCC Technical Program Committee.