Emerging Physical Unclonable Functions with Nanotechnology

Yansong Gao, *Student Member, IEEE,* Damith C. Ranasinghe, *Member, IEEE,* Said F. Al-Sarawi, *Member, IEEE,* Omid Kavehei, *Member, IEEE* and Derek Abbott, *Fellow, IEEE*

(Invited Paper)

***Abstract*—Physical Unclonable Functions (PUFs) are increasingly used for authentication and identification applications as well as cryptographic key generation. An important feature of a PUF is the reliance on minute random variations in the fabricated hardware to derive a trusted random key. Currently, most PUF designs focus on exploiting process variations intrinsic to CMOS technology. In recent years, progress in emerging nanoelectronic devices has demonstrated an increase in variation as a consequence of scaling down to the nano region. To date, emerging PUFs with nanotechnology have not been fully established, but they are expected to emerge. Initial research in this area aims to provide security primitives for emerging integrated circuits with nanotechnology. In this paper, we review emerging nanotechnology-based PUFs.**

***Index Terms*—Physical Unclonable Functions, hardware security, nanoelectronic devices, nanotechnology, reconfigurable PUF, strong PUF.**

I. Introduction

THE earliest known lock, resembling the mechanical locks of this century, dates back to 4,000 years ago—a large Egyptian wooden lock found in the ruins of the Assyrian palace of Khorsabad near Niveveh [1]. Modern security systems still keep valuables under *lock and key* in order to ensure the safety and authenticity of goods, information or identities. Locks now, however, can refer to electronic security systems with digital keys that are coded in, for example, magnetic strips or silicon chips. Digital keys are traditionally stored in non-volatile memory (NVM) for cryptographic applications. However, it has been shown that digital keys in NVMs are vulnerable to invasive physical attacks. Complicated tamper sensing and tamper-proofing mechanisms have to be implemented in hardware to secure digital keys in NVM [2] with consequential increases in area and power overhead of the device—which also limits the use of these anti-tampering methods for resource-constrained devices such as smart cards.

The growing new area of PUFs—previously termed Physical One-Way Functions [3], [4], or Physical Random Functions [5]—is receiving increased attention because PUFs offer a simple alternative to generating unique volatile digital keys in a very small hardware device without the need for tamper-sensing mechanisms. Note that PUFs are easy to build but practically impossible to duplicate, because they rely on uncontrollable physical parameter variations that occur during hardware device manufacture. More importantly, secrecy of a PUF is derived from inherent complexity in a given physical system only when it is needed, and thus PUFs can thwart physical attacks [2].

In general, when a *challenge* (input) is presented to a PUF, a corresponding *response* (output) will be generated. This response is determined by a complex physical function that is unique to each device or PUF instance as shown in Fig. 1. Given the same challenge, different PUF instantiations built upon the same design will deliver a different response. The challenge and its corresponding response are commonly referred to as a Challenge Response Pair (CRP). A set of CRPs can be treated as a fingerprint of a PUF and therefore a PUF integrated device or object.

Conventional microelectronic circuit based PUFs such as ring oscillator PUFs [6], arbiter PUFs [7], and SRAM PUFs [8] exploit uncontrollable process variations in conventional Complementary-Metal-Oxide-Semiconductor (CMOS) fabrication technology. In fact, a silicon-based identification circuit first proposed in year 2000 can also be classified as a PUF since it also exploits the randomness within a hardware device to generate a secret key [9]. Note that PUFs based on conventional CMOS technology are well established and related industrial products are already on the market. However, technological developments are particularly important for building PUFs as they rely on process variations. Therefore, it is expected that the next generation of PUFs will be implemented using emerging nanoelectronic devices [10].

Nanotechnologies, such as the phase change memory (PCM) [11], spin-transfer torque magnetic random-access memory (STT-MRAM) [12], carbon-nanotube field-effect transistors (CNFETs) [13] and memristors[1] [14], have more severe levels of inherent randomness due to fabrication process variations (e.g. thickness, cross-sectional area or doping profile) as a consequence of scaling down to the nano re-

Y. Gao, S. F. Al-Sarawi, D. Abbott are with the School of Electrical and Electronic Engineering, The University of Adelaide, SA 5005, Australia. e-mail: {yansong.gao, said.alsarawi, derek.abbott}@adelaide.edu.au.

D. C. Ranasinghe is with the Auto-ID Labs, School of Computer Science, The University of Adelaide, SA 5005, Australia. e-mail: damith.ranasinghe@adelaide.edu.au.

O. Kavehei is with the Emerging Device Research and Architecture Design Group, School of Electrical and Computer Engineering, Royal Melbourne Institute of Technology, Victoria 3001, Australia. e-mail: omid.kavehei@rmit.edu.au.

Manuscript received xxx yyy, 2015; revised xxx yyy, 2015.

[1] In the literature, the terms *memristor* and *memristive device* are used interchangeably, and sometimes the term RRAM is used. There are different classifications of memristors, but this is out of the scope of this paper and we concentrate on security applications of the device, in particular, PUF designs. In this paper, the term memristor refers to a bipolar memristor.

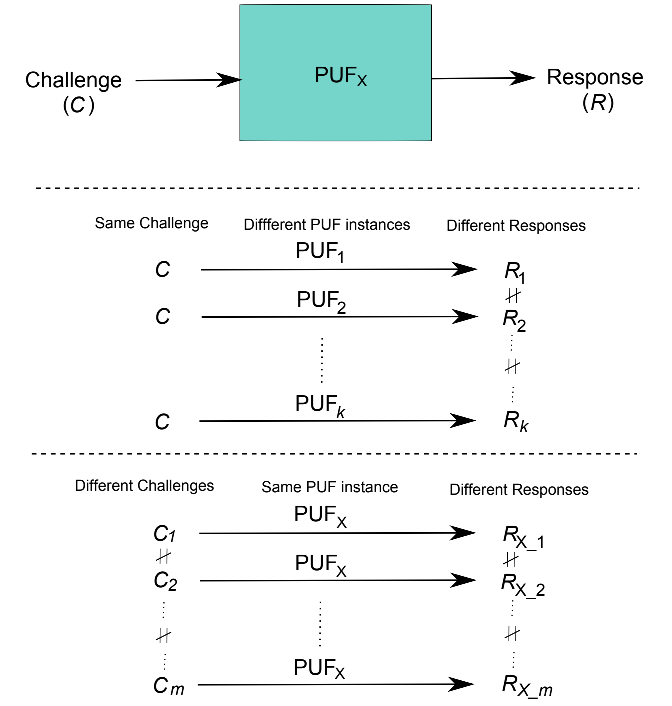


Fig. 1. A PUF can be simply treated as a black box. The response is determined by the complex physical function of a PUF that is unique to each device and the challenge applied to it.

gion. This inherent randomness provides new opportunities for building highly secure and reliable PUFs. Moreover, these nanodevices are relatively simple to fabricate and are usually compatible with CMOS fabrication processes. Together with simple nanoscale crossbar architectures that allow for ultra high-density information offer a potentially low-cost security primitive. Furthermore, special properties of these emerging nanoelectronic devices, such as cycle-to-cycle (C2C) variations during programming, offer further opportunities to construct novel PUF structures, e.g. reconfigurable PUF (see definition in Section II-C).

Already, there are a number of recent studies investigating emerging nanoelectronic devices for building PUF architectures and, consequently, they offer new solutions for PUF based security applications and opportunities for research and development in the emerging field of nanotechnology based PUFs [15]–[34]. This paper aims to give a timely review of emerging nanodevice based PUFs. In addition, we summarize the challenges and opportunities in this emerging research area, as a potential guide for hardware security designers and device engineers. The rest of this paper is organized as follows: Section 2 presents a primer on conventional PUFs. This is followed by a survey of recent nanotechnology-based PUFs in Section 3. Then a discussion of emerging PUFs architectures is given in Section 4, followed by a conclusion in Section 5.

II. A PUF Primer

A. Conventional PUFs

Pappu *et al.* introduced an optical PUF in 2001 [3], [4], also called a physical one-way function, where the response (speckle pattern) is dependent on the input laser location/polarization (i.e. challenge). The Optical PUF, however, requires relatively large and high-cost external measurement devices. Moreover, its reliability is highly dependent on very accurate calibration of the input location. Furthermore, it is difficult to integrate an Optical PUF into a resource-constrained hardware device such a contactless smart card.

Gassend *et al.* [5] then proposed a practical implementation of a microelectronic circuit based PUF—this was initially called a Physical Random Function, and later termed the Arbiter PUF (APUF). The APUF exploits manufacturing variability in gate and wire delays as the source of unclonable randomness. The response is generated based on the time delay difference between two signal propagation paths, consisting of serially connected individual stages, where the path through each stage is determined by a corresponding bit in a challenge (i.e input bit vector). This structure is simple and capable of generating an exponential number of CRPs.

However, an APUF is based on linear additive blocks and is demonstrated to be vulnerable to model building attacks [7], [35], [36] if an adversary is able to gain access to CRPs either by eavesdropping or through directly measuring the PUF to collect CRPs. To increase the complexity of such model building attacks, more variants of APUFs were proposed such as the XOR-APUF [6], [35] and the feed forward APUF [35], [37]. Another issue that results from the APUF architecture is the inconsistent responses to repeated application of certain challenges due to the arbitrator—commonly implemented using a latch to determine the winning signal path—entering into a metastable state, leading to poor reliability. Furthermore, it is difficult to implement APUFs on an FPGA based platform owing to strict symmetrical routing requirements on the two delay paths. To circumvent metastability issues, another time delay based PUF, Ring Oscillator PUF (ROPUF), is proposed in [5], [6] and further improved in [38]–[40]. An overview of different ROPUFs can be found in [41].

Besides the aforementioned delay-based PUFs, there are mismatch based silicon PUFs such as the SRAM PUF [8], [42], latch PUF [43], flip-flop PUF [44], [45], butterfly PUF [46], and analog PUFs based on silicon such as the current-based PUF [47] and nonlinear current mirror based PUF [48], which exploit nonlinear characteristics of current or voltage. Comprehensive reviews of conventional PUF architectures can be found in [49], [50]. Below we provide an overview of various classifications of PUFs.

B. Weak and Strong PUFs

In general, depending on the number of CRPs (or challenge space) that a PUF is capable of producing, PUFs can be classified into two general categories: weak and strong PUFs [50], [51].

Weak PUFs generate a limited number of CRPs, so CRPs can be fully read out within a very short time once an adversary has full physical access to it. A weak PUF can be defined, as follows:

- Impossible to be duplicated (cloned) physically.
- Number of CRPs is limited and linearly or polynomially dependent on the number of challenge bits.

Typical weak PUFs are the SRAM PUF, butterfly PUF and coating PUF [52].

Strong PUFs have been defined in [50], [51] based on an adversarial model where the adversary has access to the PUF and can apply an unlimited amount of chosen challenges to the PUF and is able to observe the raw responses of the PUF (ie. before any post processing). A strong PUF needs to prove its security under such a strong attack model. So we can define a strong PUF according to the following security properties:

- Impossible to be duplicated (cloned) physically.
- Supports a very large number of CRPs such that an adversary cannot mount a brute force attack within a realistic time, ideally demonstrated by exponential number of CRPs.
- Resilient to model building attacks by providing a polynomial number of chosen CRPs such that an adversary cannot predict the response of a PUF to a randomly selected unused challenge.

Typical strong PUF candidates include Optical-PUFs and XOR-APUF (e.g. 8 output XORs with 512 bit challenges) [50].

C. Reconfigurable PUFs

The concept of a reconfigurable PUF (rPUF) was first articulated in [35]. In an rPUF, the PUF itself has the ability to change its response to the same challenge. Instead of exhibiting static challenge-response behavior, the ability to update challenge-response behavior of a reconfigurable PUF is desirable for a number of practical applications such as the revocation or update of 'secrets' in PUF-based key generation and cryptographic primitives based on PUFs [53]. Since the definition expressed in [35], a stronger definition of a reconfigurable PUF has been proposed to ensure that the reconfiguration is difficult to reverse, even by an invasive attack measure [17]. Therefore the reconfiguration is not permitted to depend on a hidden device or parameter that can be influenced by an attacker.

Ideally, a reconfigurable PUF (rPUF) can be updated in such a way to alter the PUF into a new instance such that:

- The CRPs of an rPUF are unpredictable after reconfiguration even if the CRPs of an rPUF before reconfiguration are known.
- The security properties of the rPUF are preserved after reconfiguration.
- Reconfiguration is uncontrollable such that it does not rely on updating hidden devices or parameters.

D. Public PUF

The definition of a Public PUF (PPUF) is a multiple-input-multiple-output system that is much faster to execute on the physical device than it is to simulate by several orders of magnitude [54]. In particular, the secrets of PUF and PPUF are different. Secrets of a PUF rely on the unpredictability of its responses for a given challenge based on complex interactions with a physical function. The model of the PUF that mathematically impersonates the physical function of the PUF must be kept safe. In essence, a PUF can still be considered as a "storage" device to store secret bits using minuscule variations in the hardware device. In contrast, the PPUF hardware contains no secrets, since the PPUF model is known to every party including the verifier, prover and also the adversary. As long as the model storage is secure against tampering or rewriting, the authentication capability is solely derived from the computational time difference between the hardware based PPUF and its model, and the unclonability of the physical PPUF, while the authentication protocol is publicly known.

E. Performance Metrics

There are several metrics to evaluate PUF performance [55], [56]. Randomness (without bias), uniqueness and bit error rate (BER)—the complementary metric of reliability—are the three most used metrics among them.

Randomness tests for bias by evaluating the probability of a '1' or '0' bit in the response bits from one PUF instance. For the ideal case of no bias towards a '1' or '0' bit, there is a 50% probability for obtaining a '1' or '0'.

Uniqueness represents the capability of one PUF instance to distinguish itself from other PUF instances. It is measured by the inter-Hamming Distance (inter-HD) defined as the mean value of HD among different responses from different PUF instances when the same challenge is applied to all of these different PUFs. Ideally, uniqueness is expected to be 50%, which means that responses from two different PUF instances given the same challenge will have an average of half the bits different.

Bit Error Rate is measured by intra-HD defined as the mean value of flipped bits among different responses when the same challenge is applied to the same PUF instance, under different ambient environments, eg. temperature, power supply fluctuations. Although it is not possible to obtain an ideal value of 0%, the BER should be as small as possible to facilitate the hardware design for error correction. Some literature on PUF designs focus on stabilizing the response with a wide range tolerance on temperature and aging influence [57], [58].

F. Applications

Weak PUFs are commonly used for cryptographic key generation. In such an application, error correction must be carried out with the use of *helper data* [6], [59], [60] considering that the raw responses of a PUF are impacted by ambient environmental parameters such as temperature, supply voltage, and aging. Interested readers can read [61] to obtain further details on helper data algorithms for PUF-based key generation. In addition, weak PUFs are used to bind software to hardware devices in order to secure booting [62] and remote computing platforms [63], [64].

Strong PUFs can also be used to generate cryptographic keys but they are preferred for use in lightweight authentication applications since a large number of CRPs is available, ensuring many rounds of authentication. This lightweight authentication protocol works as follows:

- Before the PUF is transfered to a prover, a CRP database of this PUF is established on the side of the verifier through measurements.
- Whenever a prover requests an authentication, the verifier sends a randomly selected challenge from the database to the prover and receives a response obtained by the prover who applies the challenge to the physical PUF in hand.
- The authentication is considered successful only if the response from the prover is similar/close to the response stored in the database.

In this authentication protocol, a CRP is never used more than once to avoid reply-attacks.

This simple authentication protocol protects against device substitution and counterfeits without using cryptographic operations and, thus, makes strong PUFs suitable for resource-constrained devices such as RFID devices [65]–[67]. However, it has been demonstrated that this protocol is susceptible to model building attacks [51]. To overcome such an attack and other possible attacks, enhanced authentication protocols have been proposed [68]–[71]. We refer readers to [72] for further details on different lightweight authentication protocols in the literature until 2014. Besides authentication applications, strong PUFs can be used for more complex cryptographic applications such as oblivious transfer (OT), bit commitment (BC), and key exchange (KE) [18], [36], [69], [73], [74].

rPUF can be used to update security tokens [53], such as electronic tickets, secure storage in untrusted memories [17], [75] or to prevent downgrading software versions by binding hardware to software [76].

PPUF can be used in advanced applications such as public-key cryptography, secure location authentication, k-anonymity security protocol, and trusted sensing and computing [50].

G. List of Review/Survey Papers

A summary of reviews and surveys are given in Table I. Note that in the last two papers, [77] and [78], in Table I survey different hardware security primitives based on emerging nanoelectronic devices. The PUF is only one of a number of security primitives surveyed in these two papers. In terms of PUFs based on emerging nanoelectronic devices, the work in [77] surveys four PUF structures based on memristors. In contrast, the scope of our article on PUFs is much broader in its coverage of nanotechnology based PUFs. In [78], Rajendran *et al.* choose the memristor as the candidate device for enabling security primitives based on nanodevices. In terms of a PUF based security primitive, the work in [78] focuses on a nano PPUF. We limit our survey to nanodevice based PUFs. Therefore, we give a broader and more detailed survey of those PUF structures—specifically, those based on PCM, STT-MRAM, memristor, Carbon-Nanotube Field-Effect Transistors (CNFET) and nano diode.

III. Emerging Nanoelectronics Based PUFs

Initial investigations of nanoelectronic-based PUFs are motivated by the desire to achieve more secure, robust and lightweight PUF designs whilst further combining a number of unique properties, such C2C (cycle-to-cycle) programming variations and multi-response capability per memory cell, of nanodevices to achieve desirable PUF characteristics such as reconfigurability and capability of multiple response per memory cell.

Specifically, we review novel PUF designs based on PCM, STT-MRAM and memristor nanodevices that also hold promise for future universal memory. These technologies provide opportunities for the design of novel PUF structures because of substantial process variations, small footprint, lower energy consumption, non-volatility, multi-level bits (MLB) capability per memory cell, programming sensitivity, and C2C programming variations. We also survey a PUF based on a nano diode where the readout speed is intentionally slowed down through the construction of a nanocrossbar to prevent the attacker reading all the information within the high information density nanocrossbar. Moreover, a PUF design based on a MOSFET (Metal–Oxide–Semiconductor Field-Effect Transistor) like device, CNFET, is also surveyed.

As we will see in the following subsections, each PUF structure is built on one or more aforementioned nanodevice properties to meet requirements of different application scenarios. A general classification is shown in Fig. 2. Note that several PUF designs, although demonstrated using a specific nanodevice, may be implemented using a different type of nanodevice; this will be pointed out when these designs are surveyed.

A. Super-High Information Content PUF (SHIC-PUF)

A special nanocrossbar array of nanoscale diodes as a large memory lattice capable of storing a significant number of random bits, called a SHIC-PUF, was proposed in [80]—see Fig. 9 for a brief description of a nanocrossbar. The SHIC-PUF cannot be modeled since values stored in the memory lattice are directly read out as independent responses and therefore cloning requires constructing a copy of its memory contents. A full readout of all the CRPs may require a long period, eg. around several months, due to the slow readout speed (100 bit/s) and the very large amount of information stored in the nanocrossbar. The application of SHIC-PUFs, however, is limited to situations where there is no restriction on readout speed and area requirements. Slowing the readout speed intentionally also increases the enrollment time when the verifier measures CRPs to create a secure database of CRPs for later authentication operations.

B. Carbon-Nanotube Field-effect Transistors (CNFET) based PUF

The CNFET is one of a promising new class of MOSFET-like transistors that avoids most of the fundamental limitations for traditional silicon MOSFETs [13]. The CNFET consists of carbon nanotubes (CNTs), which is a molecular electronic device attracting much attention as it can lead to further technology scaling [15], instead of bulk silicon. However, precise control of CNTs in a fabrication process is challenging. As a consequence, the CNFET performance shows large variations due to:

TABLE I
REVIEW AND SURVEY PAPERS OF DIFFERENT PUF RELATED RESEARCH AREAS

Reference	Covered PUF related research areas
[49], [50]	A tutorial and a review on PUFs respectively
[41]	A detailed survey on RO PUFs
[54]	A review on Public PUFs
[55], [56]	Detailed systematic definitions of PUF performance metrics
[61]	Overview and analysis of different helper data algorithms to enable stable response regeneration
[72]	Detailed descriptions and analysis of different lightweight authentication protocols
[77]*	Different memristor based security primitives including a number of PUFs
[78]*	Different security primitives implemented using nanotechnology where the focus on PUFs is limited to nano PPUFs.

*Note the PUF is only one of a number of security primitives surveyed in these two papers, while all other papers in this table are only related to PUF based security primitives. Specifically, in [77], it surveys four PUF structures based on memristors, while we provide a broader survey of structures found through literature searchers at the time of writing this paper. In [78], a memristor is used as the candidate device to enable security primitives. In [78], the discussion on PUFs is limited to nano PPUFs.

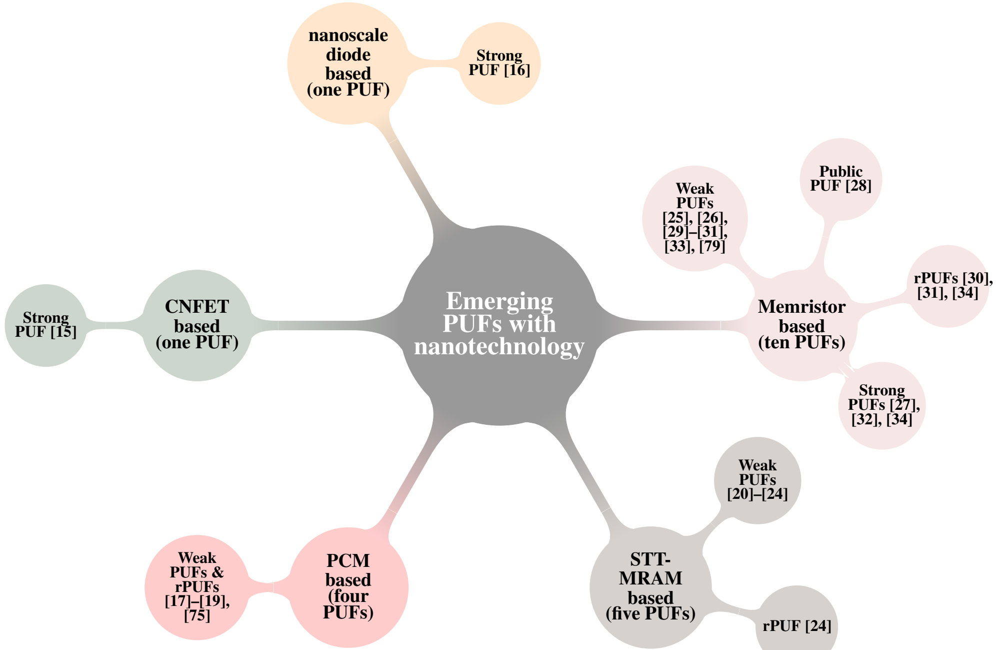

nanoscale diode based (one PUF)
Strong PUF [16]
Weak PUFs [25], [26], [29]–[31], [33], [79]
Public PUF [28]
Strong PUF [15]
CNFET based (one PUF)
Emerging PUFs with nanotechnology
Memristor based (ten PUFs)
rPUFs [30], [31], [34]
Strong PUFs [27], [32], [34]
Weak PUFs [20]–[24]
Weak PUFs & rPUFs [17]–[19], [75]
PCM based (four PUFs)
STT-MRAM based (five PUFs)
rPUF [24]

Fig. 2. Emerging PUFs with nanotechnology. A general classification according to the technology employed. Note that several PUF designs do not rely on a specific technology (medium) and these are clearly indicated in our article.

- chirality, which defines the type—metallic or semiconducting;
- diameter;
- growth density;
- alignment;
- doping concentration.

Particularly, the lack of chirality (and thus type) control is a major issue for CNT usage in digital circuits as this results in a large resistance change. In [15], Konigsmark *et al.* proposed a CNFET based PUF—CNPUF—turning the disadvantage of such variations into an advantage.

The CNPUF is shown in Fig. 3. It is, in essence, similar to an APUF. The CNPUF Parallel Elements (CNPUF-PEs), comprising two CNFETs that share the same gate voltage, are connected in series to build a chain. At the end of the chain, the current comparator is utilized to compare the top and bottom current to generate a 1-bit response.

The CNPUF-PE has two distinct and nearly independent states for a high gate voltage and a low gate voltage, respectively, due to the difference of a metallic-to-semiconducting ratio in the CNFET. The current characteristics for a low gate voltage are dominated by the metallic CNTs, while for a high gate voltage, the semiconducting CNTs dominate the current characteristics. The CNPUF-PE, in principle, acts as the multiplexer in an APUF. The difference between a CNPUF and APUF is that the generation of the response is based on comparison of summation of resistance between the top and bottom paths rather than the comparison of summation of time delays. In this paper, the authors demonstrate the high reliability performance of the CNPUF, up to 97%, without any post-processing. It appears that the simulation of the the current comparator was not taken into consideration, which may have led to a higher reliability performance during simulation.

The CNPUF design compares the summation of resistances while the APUF compares the summation of time delays. Therefore, like the APUF, the CNPUF model is built upon linear additive blocks. Consequently, we can expect the CNPUF to be broken by attacks, such as model building attacks, used for successfully breaking APUFs. However, the possible vulnerability of CNPUFs to known model building attacks is necessary for investigation.

The authors also proposed a security enhanced CNPUF—extended CNPUF (ex-CNPUF)—which XORs the challenge with the response from the other CNPUF. This design is similar to XORed-APUF. It is clear that the reliability is affected, which is also pointed out by the authors. In addition, the ex-CNPUF leads to a higher power and area overhead.

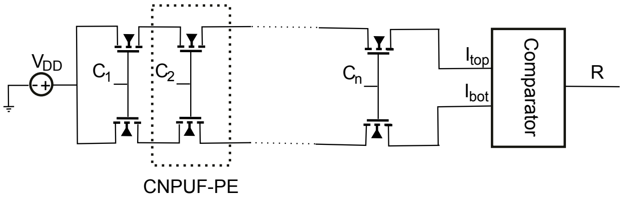


Fig. 3. CNFET based PUF design (CNPUF).

C. PCM based PUFs

Phase change memory (PCM) utilizes crystalline and amorphous nature of a phase change material. Crystalline and amorphous phases have a larger resistivity difference corresponding to low resistivity (*set* state) and high resistivity (*reset* state), respectively. A cross-sectional view of a conventional PCM cell is shown in Fig. 4. To reset the PCM cell into the amorphous phase, a high amplitude pulse with fast fall time is applied to the heater to, firstly, melt the active region and, subsequently, to quench it rapidly. Therefore, a region of amorphous, highly resistive material is left in the PCM cell. Conversely, a moderate amplitude pulse with a long period or long fall time is used to set the PCM cell into the crystalline phase, which anneals the active region, at a temperature between the crystallization temperature and the melting temperature for a time period sufficient for crystallization. The read operation is carried out by measuring the resistance of the PCM cell by passing a small current without disturbing its resistance [11]. In addition, the resistance of the PCM can be tuned to intermediate values between its high resistance and low resistance by applying a programming pulse with an intermediate amplitude and rise/fall time.

We review four PCM-based PUF structures that exploit abundant process variation, small footprint, and lower energy consumption of PCM technology together with programming sensitivity feature to formulate the basis for PCM-based rPUFs. Hence, all of the four PUFs discussed below are reconfigurable.

PCM-based PUF 1. Kursawe *et al.* [17] firstly proposed a concept that uses PCM to build up a rPUF. In this concept, as illustrated in Fig. 5, the PCM state is expected to be controlled well enough to reliably realize n logical states (resistance axis divided into n intervals), while measurements are precise enough to not only determine in which interval the resistance lies, but also where in that interval to obtain a multiple bit response. Each logical interval is subdivided into a number of more fine-grained intervals, eg. 'left' and 'right', as seen in Fig. 5. Since the programming process cannot be controlled very well in practice, a long-lived random state that can be reconfigured electronically can be derived from a PCM cell. This observation is used to create a reconfigurable PUF from embedded memory.

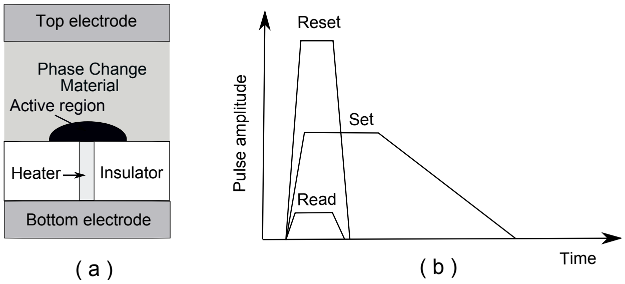


Fig. 4. (a) A cross-sectional view of a conventional PCM cell. (b) The programming pulse passing through the PCM will change the temperature in the active region using the heater based on the pulse amplitude and duration. The resistance is read out by passing a small amplitude and short duration pulse without disturbing the resistance of the PCM.

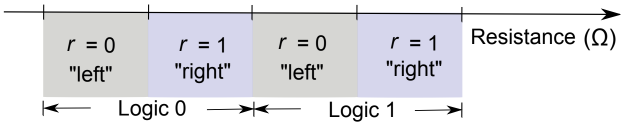


Fig. 5. Accurate control over resistance for 2 logical states and an accurate measurement gives one rPUF 2-bit response. After [17].

PCM-based PUF 2. Zhang *et al.* [18] following the concept in [17], proposed using a PCM based PUF to generate refreshable cryptographic keys (PCKGen), where the keys can be refreshed exploiting the reconfigurability of the CRPs of a PCM-based PUF through re-programming. Therefore, the kernel of the PCKGen is a PCM-based rPUF. In this work, unlike the concept in [17], a circuit solution is provided. A block diagram of the PCKGen is shown in Fig. 6, while the detailed circuit design can be found in [18]. The PCM-based rPUF is enabled by reprogramming PCM cells in an array. The response is based on the resistance comparison of two selected PCMs—addresses of these two selected PCM cells is the challenge. The reconfigurability is ensured by the amplitude and width variations of the programming pulse that refresh the resistance of all the PCMs in the array to different values. The variations of programming pulse width and amplitude depend on the intrinsic process variations of an auxiliary CMOS circuit—the imprecisely controlled current-pulse regulator (ICCR) block shown in Fig.6.

It is clear the response of this PCM-based rPUF is not reliable enough to satisfy cryptographic key generation, so further error correction assisted by helper data is required. Hashing of responses is also used to further improve the randomness of the responses. The power consumption of PCKGen—as a memory based PUF—is higher than typical memory based PUFs such as the butterfly PUF [46] and the SRAM PUF [42] mainly owing to the reprogramming operation and auxiliary circuits. But it has footprint advantage due to the high density of PCM arrays. The other benefit of PCM-based rPUF is that the key can be refreshed. Zhang *et al.* [75], following their previous work [18], carried out experimental evaluations of PCM based rPUF, on 180-nm PCM chips.

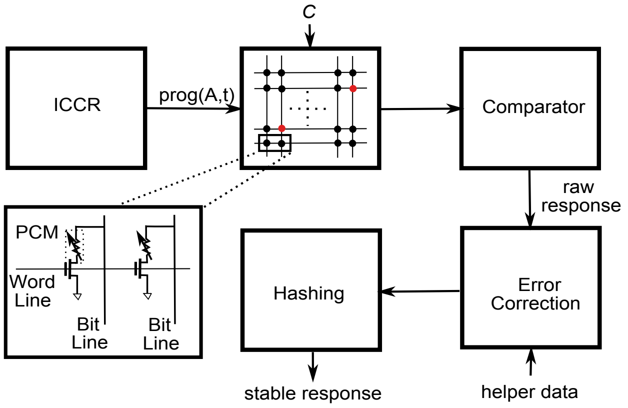


Fig. 6. Block diagram of PCKGen. The imprecisely controlled current-pulse regulator (ICCR) is responsible for generating imprecise duration and amplitude of the programming pulse to reconfigure the PCM array. Comparator compares resistance of two selected PCM cells (marked with red color)—where the address is determined by a challenge—in the PCM array to generate raw responses

PCM-based PUF 3. Further, Zhang *et al.* [19] proposed Memory-based Physical Unclonable Function (MemPUF) based on their previous work—PCM-based PUF 2—to overcome the susceptibility of secret keys to physical attacks [18], [75]. In this case, we may assume that an adversary has physical access to the MemPUF and is allowed to measure all CRPs given enough time, where a challenge is actually the address of one MemPUF cell. Hence, an adversary could impersonate the original MemPUF by the measured CRPs. To make these physical attacks ineffective, Zhang *et al.* proposed using periodic self updates of the PCM-based rPUF and subsequent enrollment to the verifier to update the CRP database of the verifier. As a result, the CRPs measured by an adversary for one updated rPUF instance cannot be used to predict the response to the same challenge after the rPUF is updated. This relies on the fact that the reconfigurability is irreversible, therefore, the responses to the same challenge generated at each update iteration are independent. This study carries out statistical analysis to demonstrate the effectiveness of their rPUF based on PCM against the measurement-prediction attack given an adversary with certain bounded attack capability. However, further research is needed to develop the means to secure the communication between the verifier and the prover to refresh the CRP database after each update.

D. STT-MRAM-Based PUFs

The STT-MRAM is made up of a CMOS transistor and a magnetic tunnel junction (MTJ) as shown in Fig. 7 [12]. By passing a current through the fixed layer, one can produce a spin-polarized current. If this spin-polarized current is directed into the free layer, angular momentum can be transferred to this layer, changing the orientation of its magnet. When both of these two layers are under parallel configuration (P), the MTJ has low resistance. Conversely, the MTJ shows high resistance under anti-parallel configuration (AP).

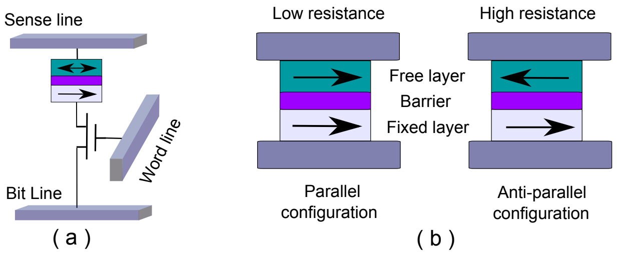


Fig. 7. (a) The STT-MRAM memory cell. (b) MTJ configuration.

The following five STT-MRAM-based PUFs in this subsection exploit prevalent process variations, non-volatility, and MLB capability of STT-MRAM.

STT-MRAM-based PUF 1. Marukame *et al.* proposed a PUF based on magnetic tunnel junctions (MTJs) and a method of extracting the PUF signature based on intrinsic properties of spin transfer switching (STS) [20]. They experimentally measured and characterized the STS characteristics, followed by a method to extract a PUF signature. They found that each measured MTJ exhibited different programming voltages for the AP to P state and for P to AP state transitions. The variability of the programming voltage between different MTJs results from the extrinsic properties such as junction resistance, and process damage.

Based on these variations, an operation sequence for extracting a PUF signature from a 4×4 array of MTJs was proposed as shown in Fig. 8. Firstly, a reset voltage is applied to reset all of the MTJs in the array to initial state P. Then a specific V_{PUF} voltage—a pre-calibrated voltage–is applied to the MTJs to induce switching with a 50% probability for the majority of the MTJs. The stochastic switching results in a random distribution of the P or AP state transitions across the array. The information—resistance—of every MTJ is read out and evaluated by the PUF signature extraction circuit. As a result, the MTJ cell is categorized into three patterns: i) much less than 50%—resistance close to P state—(white); ii) around 50%—in the middle of P and AP state (gray); and iii) much more than 50%—resistance close to AP state—(black). After the information is stored, the array is reset. To increase the reliability of the extraction, the above operation sequence is repeated multiple times to extract white and black bits and discard gray bits. The whole extraction operation is similar to majority voting to determine reliable bits. In the last step, the *black bits* will be chosen for the PUF signature.

Although this work investigates the possibility of building up a PUF based on STT-MRAM, further investigations are needed to: i) design the extractor; ii) propose an efficient approach to store the pattern information during each extraction process; iii) evaluate the relationship between PUF performance and period and amplitude of V_{PUF}, and the number of repeated extractions used to generate a signature.

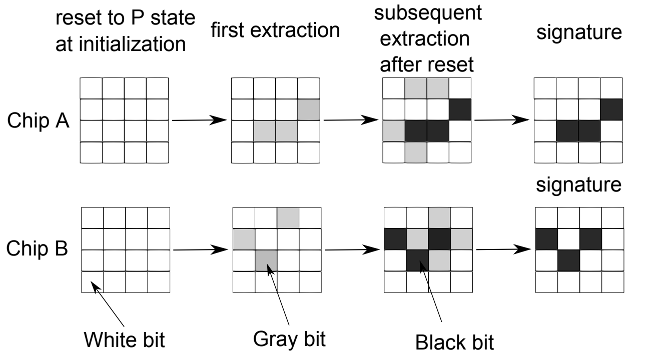


Fig. 8. Operation sequence for extracting a PUF signature from a 4×4 STT-MRAM array. The extracted data is categorized into three patterns: much less than 50% (white), around 50% (gray) and much more than 50% (black). After [20].

STT-MRAM-based PUF 2. Vatajelu *et al.* exploited variations in AP state to extract a PUF response by comparing current sensed from a selected MTJ in a STT-MRAM cell with a reference current value that is generated by averaging AP state resistance (high resistance) among a specific number of STT-MRAM cells. The STT-MRAM cells generating responses are named as *active cells*, conversely, STT-MRAM cells in charge of producing a reference current are named as *reference cells*. If the sensing value is higher than the reference current value, then logic '1' is given as a response corresponding to a challenge that is the address of this cell, otherwise, logic '0' is given. In contrast with the PUF design of Marukame *et al*, Vatajelu *et al.* provided the extraction circuit design and carried out performance evaluations on randomness, uniqueness and reliability [21].

STT-MRAM-based PUF 3. Zhang *et al.* proposed their STT-PUF as a low-cost cryptographic key generator for embedded computing platforms [22]. A response is generated based on the resistance comparisons between two MTJs in one selected 2T2MTJ (2-transistor-2-MTJ) cell, where both of these two MTJs are initially reset/set to AP/P state simultaneously. The first extraction of the response bit is carried out through comparing the resistance of these two MTJ cells. After that, an Automatic Write-Back (AWB) scheme [81] exploiting the non-volatility of STT-MRAM is employed to write the logic value from the first extraction. The persistent storage of the first extracted logic value demonstrates enhanced reliability. In particular, the two MTJs in this cell are automatically written back to the complementary states—one MTJ is set to P state and the other is set to the opposite AP state—according to the value of the first extracted response bit, that is logic '1' or '0'.

Clearly, the regeneration of a response after AWB scheme is robust because the response bit is now stored as the complementary MTJ states of one 2T2MTJ cell. The BER (Bit Error Rate) of STT-PUF is lower than 10^{-6}—the industrial standard of BER for a cryptographic key—and consequently reduces the chip area needed for STT-MRAM PUF based key generators by eliminating the need for ECCs (Error Correction Code) schemes in hardware. This work is based on simulations conducted using a 40 nm compact MTJ device model [82].

STT-MRAM-based PUF 4. The other study from Zhang *et al.* [23] investigated Buffer-Free Memory-based PUF (BF-MPUF). Specifically, this design exploits the non-volatility of emerging Non-Volatile Memory (eNVM). The BF-MPUF is able to generate PUF responses without disturbing the data stored in the memory. In conventional memory based PUFs such as SRAM PUF, the data preserved as charge will be overwritten if the same memory is used for extracting a PUF response. As a consequence, a buffer storage and a write back circuit are needed to store the current data that will be written back to the memory cell after generating the response. The BF-MPUF removes the need for such buffer storage and additional write back operation circuit by relying on the non-volatility of eNVM, and hence provide a lower power and area overhead compared with SRAM PUFs.

In BF-MPUF, the memory array is split into two parts, similar to the work by Vatajelu *et al.* [21]. One part generates a reference current, the other part produces responses according to the comparison between the current read out from the *active cells* and the reference current. The difference from [21] is that the *active cells* are checked first to determine its state (AP or P). If it is in AP state, then a reference current corresponding to AP state is generated from *reference cells*, otherwise, a reference current corresponding to P state is generated. Accordingly, both AP and P states are exploited to generate responses according to the current state of the eNVM without disturbing its state. A buffer and a write back circuit are not involved during the generation of responses in BF-MPUF. Such a BF-MPUF design is resistant to possible leakage or side-channel attacks. The authors also investigated a method to optimize the design parameters to maximally balance both the memory yield and PUF qualities, eg. uniqueness.

Overall, the BF-MPUF design methodology is demonstrated by employing STT-MRAM, nevertheless other eNVMs such as PCM and RRAM are also applicable for this BF-MPUF design methodology. The core concept employed by BF-MPUF is that a logic bit is stored as a resistance in these eNVMs instead of charge preserved in CMOS devices.

STT-MRAM-based PUF 5. Zhang *et al.* proposed a STT-MRAM PUF that brings together high reliability, reconfigurability and multiple bits per cell capability [24]. This PUF structure is also memory based. The memory cell consists of 2 transistors acting as selectors and 2 MTJs, similar to the memory cell used in STT-MRAM-based PUF 3 [22]. The high reliability is ensured by the Automatic Write Back scheme that is similar to that proposed in [22] (see the STT-MRAM-based PUF 3 in this section). The multiple bits per cell is achieved by exploiting *bit alteration phenomena* [24]. For example, comparison of resistances of STT-MRAM in the P or AP state may not be consistent. Specifically, the resistance of the first MTJ in a 2T2MTJ cell is higher than the second MTJ when both of them are set to the P state resulting in a response of logic '1'. However, the resistance of the first MTJ may be lower than the second MTJ when both of them are set to AP state giving rise to a response of logic '0'. Hence, altering the initial state to a different state allows the responses to be reconfigured.

The work in [24] has advantages in terms of chip area and energy consumption in comparison with conventional memory based PUFs. Nevertheless, the simulation results show that the portion of cells displaying the bit alteration phenomena out of all memory cells is only 11.4%. This means that 88.6% of the memory cells are not capable of generating different response bits when the two MTJs are set to P and AP states, respectively, due to strong self-correlation effects. For example, if the resistance of the first MTJ is larger than the resistance of the second MTJ in P state in the same 2T2MTJ cell due to the difference in their junction areas, the resistance of the first MTJ will still be larger than the resistance of the second MTJ in AP state. This is because the influence of junction areas on MTJ resistances in the P and AP states are similar.

E. Memristor-Based PUFs

We briefly introduce a nanocrossbar array and memristor by virtue of some memristor based PUF designs utilizing nanocrossbars. The nanocrossbar is in principle the simplest functional electrical circuit holding great promise in nanoelectronics due to its attractive, regular structure, relatively low cost and simple implementation. A nanocrossbar array (Fig. 9a) consists of parallel horizontal wires on the top and perpendicular vertical wires at the bottom. Nanocrossbars are usually constructed with a passive two-terminal resistive device, such as a memristor [14], [83], [84] shown in Fig. 9b, at the cross-point for data storage, computing, and neuromorphic applications. Together with nanocrossbar structures, properties of memristors such as non-volatility, switching behavior and nanoscale dimensions present new opportunities for realizing ultra high density memory arrays.

In [14], the simple memristor model (often referred to as the HP[2] model) treats a memristor as an ideal device, whose resistance is finely tuned according to the integral of amplitude and period of the pulse applied across it. This simple model is the basis for several memristor based PUF designs when simulations are carried out to evaluate their performance [25], [26], [32], [85]. Given the range of memristor-based PUFs exhibiting characteristics of various PUF types described in Section II, we categorize and describe them below as: i) weak PUFs; ii) public PUFs; iii) reconfigurable PUFs; and iv) strong PUFs.

1) **Weak PUFs:** *Weak-Write-Based PUFs.* Two memristor based PUFs [25], [26] employ a time and voltage constrained write mechanism (weak-write) to force each memristor to a theoretically undefined logic region (neither logic ‘1’ nor ‘0’). Subsequently, these memristors attain an unpredictable logic state (either logic ‘1’ or ‘0’) owing to process variations that influence memristance. Then the resistance can be read out by sensing the current passing through the memristor. This is achieved by applying a small read voltage across the memristor. Notably, the read signal—has a negative pulse followed by a positive pulse with equal magnitude and duration—must be carefully generated requiring highly accurate signal generators. Although [26] utilizes a nanocrossbar array to increase the information density, similar to a SRAM PUF, a memristor PUF [25], [26] is only capable of producing a limited number of CRPs,

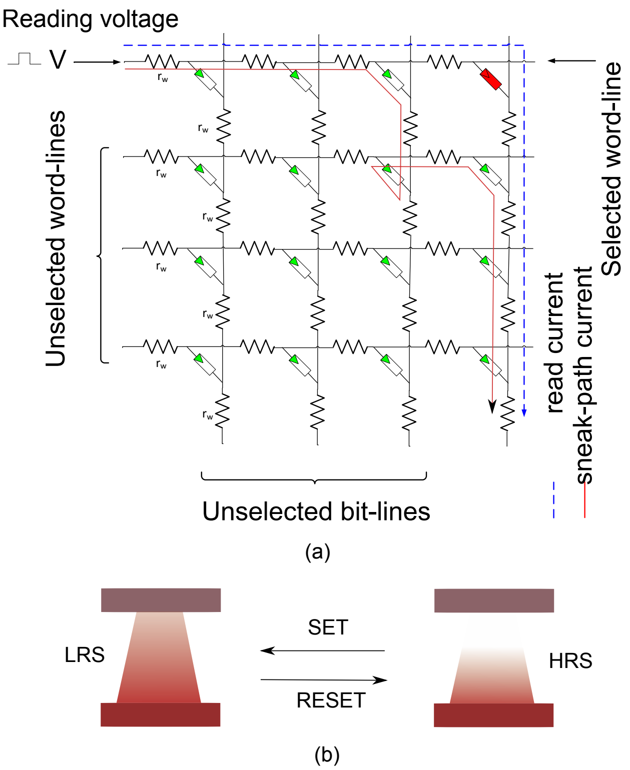


Fig. 9. **Nanocrossbar array and a memristor**. (a) Nanocrossbar array of memristors, where each memristor is located at the crosspoint of a top and a bottom electrode. When reading a targeted memristor (selected cell in the top-right corner), reading voltage is applied to the selected word line and the current passing through the selected bit line is sensed to determine the state of the memristor. For other unselected word and bit lines, they can be connected to ground or to a high impedance (floating). During reading it is important to note that there also exists many sneak path currents besides the desired read current. (b) Illustrates the operation principles of a memristor. There is a concentration gradient of filaments that can be moved back and forth using an applied electric field. The memristor switches from a HRS (High Resistance State) to a LRS (Low Resistance State) with a positive potential difference between the bottom electrode and top electrode corresponding to SET switching as one or more conductive filaments grows or forms, while it switches from LRS to HRS with a negative potential difference between the bottom electrode and top electrode corresponding to RESET switching as the filaments are disrupted. Once a memristor has been programmed, its memristance does not change even if its power supply is disconnected unless a voltage higher than a threshold voltage is applied across the device. Note that this is a very simple illustration of the memristor and does not cover all device-physics aspects of switching. After [34].

More significantly, the PUFs in [25] and [26] require a calibration procedure to determine the weak-write parameters (time and voltage) to force memristors into the undefined logic region. Furthermore, the work in [26] does not consider sneak path currents [86] within the nanocrossbar array that make the readout of the PUF responses (i.e resistance through sensing current) very difficult. Koeber *et al.* [26] only evaluated PUF performance in terms of uniqueness from simulations, while Rose *et al.* [25] performed evaluations on randomness, uniqueness and bit-biasing also through simulations. Reliability are not evaluated in both of these studies.

Mazady *et al.* [79] recently carried out a prototype demonstration of the weak-write time based PUF proposed by Rose *et*

[2] Hewlett-Packard labs or HP lab

al. [25]. They fabricated six memristors and implemented other necessary peripheral CMOS circuits with fabricated memristors on a breadboard to conduct the first memristor based PUF demo. The results show 50% randomness for responses generated from these six memristors individually—ideally, population of memristors used for evaluation is expected to be larger. However, detailed results of response consistency from the same memristor under multiple reprogramming cycles when taking C2C (cycle to cycle) variations into consideration and evaluation of reliability is needed in the future to better understand the performance of weak-write based PUFs.

Highly Reliable PUF. Che *et al.* [29] proposed a memristor based PUF that is capable of regenerating responses reliably without helper data or ECC (Error Correction Code). This work in essence is similar to the work of Zhang *et al.* [22], although a different methodology and device are used. The extraction of the PUF signature in this memristor-based PUF follows several steps:

a) All of the memristors in the nanocrossbar array are initially programmed into LRS (Low Resistance State). Where abundant resistance variations in LRS, see Fig. 10, are exploited as the source of uncontrollable variations of the PUF.
b) A voltage-to-digital converter (VDC) is employed to digitize the analog resistance of each cell to a value between 0 and 127.
c) The median of these digital values is obtained by counting the number of instances for each digitized value (from 0 to 127).
d) Each memristor cell in the nanocrossbar is programmed into LRS if the digital value of itself is lower than the median value, otherwise the memristor cell is programmed into HRS.

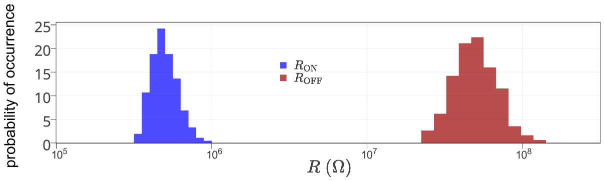


Fig. 10. Experimental resistance variation distribution extracted from a 40 × 40 nanocrossbar array with 1600 memristors obtained from experimental data in [83].

To regenerate PUF responses, the state of memristors are read out for given challenges that are actually selected addresses. Since the ratio of HRS to LRS is large enough to distinguish these two states, see Fig. 10, even taking the right tail of the LRS resistance distribution and the left tail of the HRS resistance distribution into consideration, we can see a large gap between HRS and LRS. Therefore, the reliable response regeneration is ensured.

In summary, the high reliability as illustrated in [29] and [22], firstly, exploits the resistance variations as shown in Fig 10, in either LRS or HRS to extract a one-time response. Notably, this response extraction is only carried out once. Secondly, memristors (STT-MRAMs) are programmed to the LRS or HRS according to the extracted one-time response value. For example, the memristor is programmed to be HRS if the one-time extracted response is '1', or vice versa [29]. This step exploits the large HRS/LRS ratio—large resistance gap between HRS and LRS shown in Fig 10—to ensure that the readout of the write-back response extracted first time is always robust. However, writing the extracted PUF response to a memristor (STT-MRAM) is similar to storing secret keys in NVM, which can make the secret keys vulnerable to physical attacks, although the one-time extracted responses—keys—are inherently unique from one device to another.

2) **Public PUF:** A novel nano public PUF (PPUF) based on a memristor nanocrossbar was first proposed in [28] built upon their previous work [87] and further improved in [88]. The PPUF is suitable for authentication and public-key communication. They exploited several unique characteristics of memristors such as:

a) Bidirectionality to allow the input signal be applied at any end of the circuit.
b) Nonlinearity of the memrsitor device model.
c) The fabrication process variations inherent in memristors.

Note the nano PPUF designed in [87], [88] is demonstrated by using memristors [28]. However, the nano PPUF can also be implemented through other types of nanoelectronic devices such as PCM or STT-MRAM, if these devices are properly designed.

The time-bounded authentication protocol of nano PPUF can be described as follows—also illustrated in Fig. 11:

a) Whenever verifier (Alice) wants to authenticate the prover (Bob), the verifier (Alice) obtains the model of the nano PPUF from a trusted third party.
b) The verifier (Alice) randomly picks up a challenge ($V_{\rm in1}, V_{\rm in2}$, etc.) and sends it to the prover (Bob).
c) The prover (Bob) measures the responses ($V_{\rm out1}, V_{\rm out2}$, etc.) for the given challenge, and sends the responses ($V_{\rm out1}, V_{\rm out2}$, etc.) to the verifier (Alice).
d) The verifier (Alice) selects a subsection of the nano PPUF, for example the polyomino in the bottom-right corner of the nanocrossbar shown in step ④ in Fig. 11. Then the verifier (Alice) requests boundary voltages of the selected subsection (eg., $V_{\rm C}, V_{\rm D}, V_{\rm E}, V_{\rm F}$ shown as red points) from the prover (Bob).
e) The prover (Bob) measures the boundary voltage and sends them to the verifier.
f) The verifier (Alice) simulates the selected subsection for the given challenge based on boundary voltages (eg., $V_{\rm C}, V_{\rm D}, V_{\rm E}, V_{\rm F}$) and determines the corresponding responses, say $V_{\rm out1}$ and $V_{\rm out2}$. If the simulated responses match the responses from the prover (Bob), then the verifier authenticates the prover.

During the authentication, the bounded time for allowing the prover to measure the responses and boundary voltages should be less than T_0, which makes stimulation of the corresponding responses and boundary voltages by the adversary impossible. In particular, the boundary voltages help to significantly reduce the computing time for the verifier (Alice) to simulate the response. However, an adversary cannot provide correct boundary voltages within the bounded time T_0 by using the

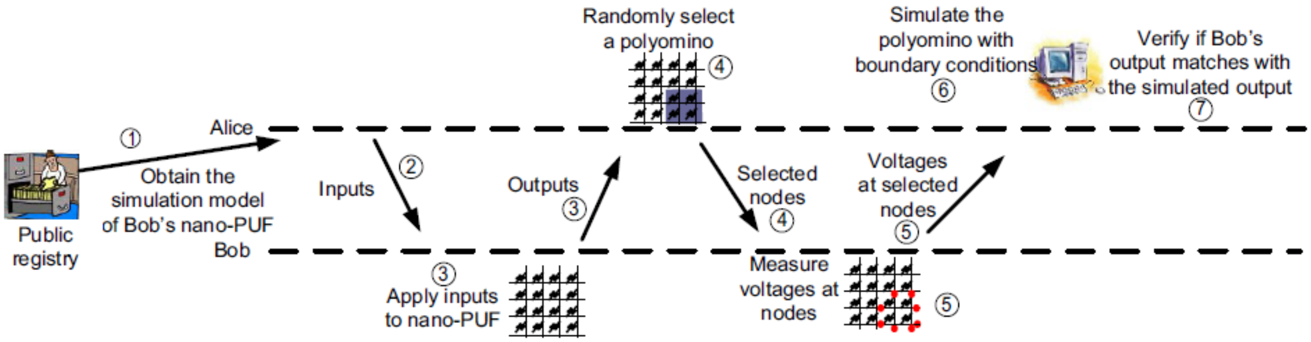

Randomly select a polyomino
④
Simulate the polyomino with boundary conditions
⑥
Verify if Bob's output matches with the simulated output
⑦
①
Alice
Obtain the simulation model of Bob's nano-PUF
Public registry
Inputs
②
Outputs
③
Selected nodes
④
Voltages at selected nodes
⑤
Bob
③
Apply inputs to nano-PUF
Measure voltages at nodes
⑤

Fig. 11. Time-bounded authentication protocol using nano PPUF. After [28].

model of the PPUF. So the time T_0 must be pre-defined. Notably, in the last step, the simulation time of the selected subsection costs much less time than the time to simulate the entire nano PPUF.

A nano PPUF always needs very accurate measurements of its physical model parameters to create a model of the nano PPUF, which is inconvenient and expensive. In addition, the high reliability requirement of PPUF designs may pose a significant challenge for their realization in practice.

3) Reconfigurable PUFs: **Reconfigurable PUFs:** Gao *et al.* [34] and Chen *et al.* [30], [31] noticed that memristors have unique cycle to cycle (C2C) variations introduced by each programming operation. This phenomenon is caused by the random change of locations of some filaments during disruption and formation [31]. Hence, the resistance observed in HRS state or LRS state varies from cycle to cycle as illustrated in Fig. 12. This unique variation is desirable for creating a reconfigurable PUF, whose CRPs are refreshable by simply reprogramming the memristor from HRS to LRS or from LRS to HRS. The update of rPUF based on C2C variations is cost-effective, requires no additional circuitry and is endowed with an unlimited reconfiguration space—the number of times that rPUF can be transformed into a new PUF instance—as opposed to other rPUFs. More significantly, the reconfiguration is nearly impossible to reverse due to the random behavior of the disruptions and formations of the filaments.

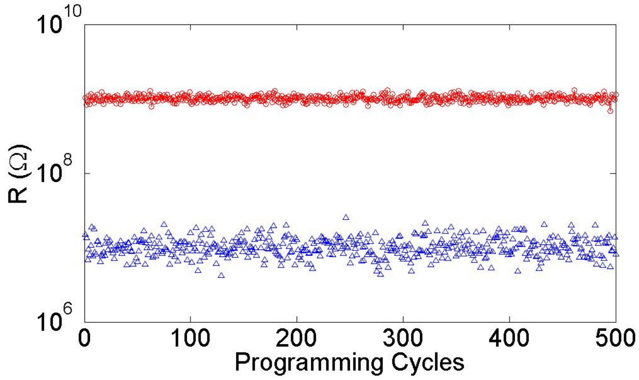


Fig. 12. Cycle to cycle (C2C) variations. R_{OFF}/HRS and R_{ON}/LRS variation of an individual memristor for 500 cycles, experimental data is adopted from [89].

4) Strong PUFs: **Strong PUFs:** The aforementioned memrirstor based PUF designs only produces a limited number of CRPs, except the nano PPUF. Gao *et al.* [34]—based on their previous work in [33] to design mrPUF, Rose *et al.* [27], and Mathew *et al.* [32] proposed three different memristor based strong PUF designs that are capable of generating a very large number of CRPs. Analyses on resistance to model building attacks are provided in [32], [34].

XBARPUF. The XBARPUF was proposed by Rose *et al.* [27] to circumvent weaknesses—limited number of CRPs and calibration procedure—of their previous design in [25] (see *Weak-Write-Based PUFs*) based on a similar write-time mechanism. In this work [27], a nanocrossbar is utilized to improve the information density. A simplified architecture without the circuits for generating control signals, eg. R/W and *RST* are illustrated in Fig. 13. The switch box behaves like a single stage in an APUF. Whenever a challenge bit, eg. C_0, is applied to a switch box, two memristors—eg. $\mathrm{M}_{0,0}$ and $\mathrm{M}_{1,0}$—located in two rows, first and second rows, are selected and a write voltage is applied through R/W. One memristor, eg. $\mathrm{M}_{0,0}$, is actively written to either HRS or LRS, while the other one, $\mathrm{M}_{1,0}$, is inactive (retains its resistance). To generate a response bit, for example R_0, during write, currents from both columns—eg. first column and second column—are compared. Depending on the fastest column to reach a threshold current level, the arbiter produces either '1' or '0'. The currents are determined by the resistances of memristors connected to that column.

The XBARPUF eliminates the calibration procedure to find optimized default weak-write parameters by resorting to the relative write-times of pairs of memristor circuits to generate the response. Also, it can generate M-bit responses simultaneously by exploiting the ultra high density of nanocrossbars. Although this PUF is able to generate 2^N CRPs, its model is similar to the model of the APUF. Therefore, it appears to be vulnerable to modeling attacks. The reliability of XBARPUF is not evaluated as the simple model (McDonald model [90]) that is used in simulations to verify their design lacks temperature dependent parameters.

memristor PUF. In the study by Mathew *el al.* [32], the PUF structure proposed is illustrated in Fig. 14 and is similar to a typical APUF. However, the model of this PUF is different from the model of an APUF. The operation of the PUF follows:

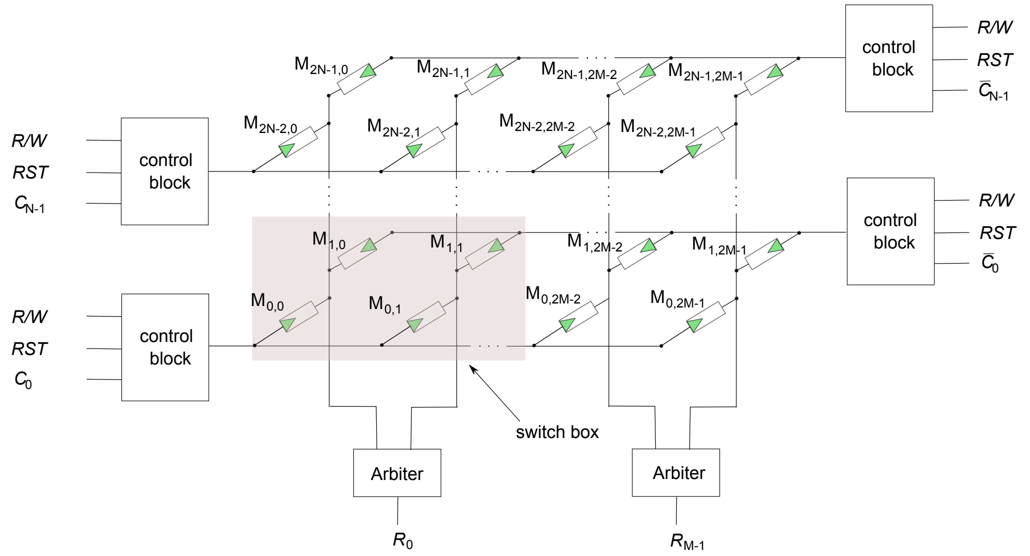


Fig. 13. Simplified architecture of the XBARPUF for N configuration and M-bit response [27].

a) *Reset*: This operation resets all the serially connected memristors to random states. The control voltages are V_{RST} and $\overline{V_{\mathrm{RST}}}$.
b) *Challenge application*: The challenge enable voltage, V_{CE}, enables the challenges when the V_{pulse} is applied. Duration of V_{CE} is shorter than V_{pulse}. When V_{CE} is enabled, the challenge applied determines how far the V_{pulse} can propagate. Specifically, V_{pulse} can only reach the first memristor segment with an applied challenge bit of ‘1’. Furthermore, V_{pulse} changes the resistance of memristors along the propagation paths, which ultimately determines the time delay of these two paths. The authors refer to this approach as *challenge-dependent stage delays*.
c) *Signal propagation*: Once the V_{CE} is disabled, the V_{pulse} propagates to the end of the delay chain. Disabling V_{CE} turns all transistors along the two paths into cut-off state. Hence these transistors exhibit drain capacitances. Then the time delay can be modeled through the delay of RC networks [32].
d) *Response generation*: The wining pulse from the race between two paths determines the response as done in the APUF.

The stage-delay parameters of this PUF are challenge-dependent, which is different from a traditional APUF. From the pespective of machine learning, the nonlinearity resulting from challenge-dependent properties makes common machine learning techniques—eg. support vector machine and logistic regression—hard to apply. Mathew *el al.* not only evaluate the performance of this proposed memristor-based PUFs, but also provide results from machine learning based modeling attacks using known attack methodologies against PUFs. Their works show good performance, especially, with regard to reliability and resistance to modeling attacks.

However, the PUF response generation takes several steps that seem to complicate its operation. The precise control of voltage and their sequence may cause difficulties in practical applications. The response generation is based on time delay difference between two paths, which is similar to the response generation of the APUF. Therefore, when the delay difference are too small to be resolved by the arbiter, we expect the arbiter to transition into metastable state as demonstrated in APUFs [91]. Furthermore, the temperature dependent coefficient of the employed memristor model (HP model) is not described. Hence, the superior reliability obtained based on simulation is speculative. Moreover, the ideal memristor model used to validate the results does not take C2C variations into account, which may further aggravate the reliability of the PUF.

The maximum number of CRPs used for training a model in [32] is 3500, which appear to be inadequate based on published modeling attack tests on PUFs [51]. Therefore, more work is needed to confirm the illustrated modeling attack resistance. We note that symmetrical challenge vectors should be avoid for this PUF structure as they can lead to deterministic responses for other challenges. For example, if $\mathbf{C_1} = \{01......10\}$ gives response of ‘1’, a different challenge $\mathbf{C_2} = \{01\mathrm{xxxxxx}10\}$ will always produce response of ‘1’ irrespective of the challenge bit pattern in xxxxxx.

mrSPUF. This PUF structure (mrSPUF) proposed by Gao *et al.* [34] is illustrated in Fig. 15. The mrSPUF design circumvents the limited number of CRPs generated in their previous work for mrPUF [33]. The resistance variation is more prevalent in the LRS state than in the HRS state due to the thickness of memristors demonstrated in [92]. Furthermore, in [93] it is demonstrated that resistance is resilient to temperature and telegraph noise in the LRS state more than in the HRS state. Therefore, all memristors are programmed into LRS state initially. The mrSPUF architecture combines nanocrossbar and current mirror controlled ring oscillators (CM-ROs) to realize not only a strong PUF but also a reconfigurable PUF, here the variations exploited is sourced not only from manufacturing variations but also C2C variations introduced from reprogramming operations. The response generation is based on comparison of a pair of frequencies of two CM-ROs, where the frequency of each

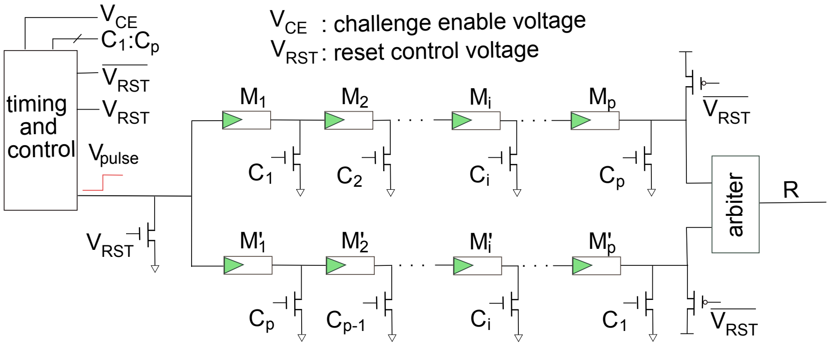


Fig. 14. Architecture of the proposed memristor-based PUF circuit [32].

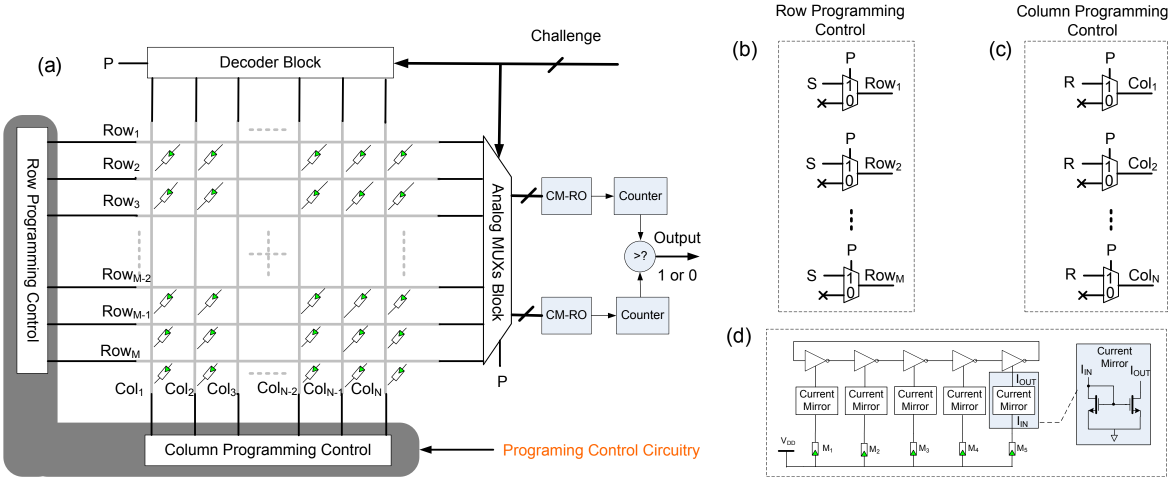


Fig. 15. mrSPUF architecture. (a) Simplified mrSPUF architecture. All the memristors are in the ON state. The shaded programming control circuit comprises the row programming control circuitry in (b) and column programming control circuitry in (c), which is employed to program memristors in the nanocrossbar array before it acts as a PUF and facilitates reconfigurability of mrSPUF by subsequent reprogramming to refresh CRPs of mrSPUF to transform it into a new PUF instance. In contrast to the programing control circuit, the top decoder block and left analog multiplexers block, CM-ROs and counters enables the stimulation by an challenge and the extraction of a corresponding response. A challenge encoded as a vector of binary values (bits) is used to provide the address bits for both the analog multiplexers block and the decoder block. (d) CM-RO. Each current mirror starves only an inverter in the RO structure, where the bias memristor for each current mirror, M_i, is selected from the nanocrossbar array. Although variation in the oscillation frequency of each CM-RO is slightly influenced by the threshold voltage variation in the CMOS transistor composing the starved inverter and current mirror structures, the overall variation in the oscillation frequency is primarily determined by the variations in memresistance of M_i if the supply voltage, $\mathrm{V_{DD}}$, is kept constant.

ring oscillator is configured by i ($i = 5$ in this example implementation case) memristors selected by a challenge. The detailed configuration is shown in Fig. 15(d), where each memristor starves one inverter in a CM-RO through a current mirror based on the current passing through it. This means the delay of the given inverter is predominantly determined by the resistance of the memristor that configures this inverter.

In general, mrSPUF has two CM-ROs, each CM-RO has i inverters, therefore $2 \times i$ memristors are randomly selected. So the total CRPs ($\mathrm{NT_{CRP}}$) in this configuration can be estimated as:

$$\mathrm{NT_{CRP}} = \frac{N \times \binom{M}{i} \times \binom{M-i}{i}}{2}, \tag{1}$$

where N is the number of columns, M is the number of rows. Therefore, the number of inverters in CM-RO or the nanocrossbar array size or all of the design parameters can be altered according to the desired number of CRPs.

The average frequency observed by using five inverters configured by five memristors is in the vicinity of 25 MHz. Therefore, an advantage of using the current starved ring oscillator is that the frequency is sufficiently low that an accurate counter is not required to measure the frequency of CM-RO. The advantages of using CM-ROs in mrSPUF are summarized below:

a) The current starved ring oscillator slows down the frequency of the RO, as a result, facilitates the design of a simple counter, because measuring high frequency requires a more accurate counter, higher power consumption and takes up more silicon area accordingly.
b) The CM-RO is used to translate the analog variations in memresistance into digital values. The use of CM-ROs to generate responses avoids potential metastability issues occurred in APUFs.
c) Different memristors are selected to configure each inverter in this RO loop to significantly increase the number of CRPs.
d) Two CM-ROs are used to construct a differential circuit. This design will mitigate some fluctuations caused by noise, such as thermal noise, to improve the reliability of the mrSPUF, since the response is generated from the

comparison of a pair of frequencies of two CM-ROs.

To enable the mrSPUF to act as a rPUF, additional circuitry is not necessary. Reconfiguration can be achieved in two simple sequential operations on memristors: RESET, and then SET. Specifically, memristors are firstly switched from the LRS state to the HRS state by applying $R = |V_{\mathrm{RESET}}|$, a voltage that is higher than the absolute value of the negative threshold voltage, to the selected column (bit line) and applying $S = 0$ to the selected row (word line), see Fig. 15 (b) and (c). Then they are switched back from the HRS state to the LRS state by applying $S = V_{\mathrm{SET}}$ to the selected row and $R = 0$ to the selected column using the programming control circuitry shown in Fig. 15.

IV. Discussion

Considering different architectures, application scenarios and security features of the surveyed PUFs, a direct comparison would be difficult, instead we provide typical performance metrics for comparison as shown in Table II. These metrics include randomness, uniqueness and BER—this is the complementary metric to reliability. In addition, we provide an order of magnitude description of the number of CRPs generated by a PUF in relationship to its basic building blocks—detailed definitions are in the table notes^2 of Table II. Moreover, we classify the surveyed PUFs into different PUF classes including weak PUF, strong PUF, rPUF, and PPUF. Further we discuss the opportunities and challenges facing these emerging PUFs with nanotechnology.

A. Opportunities

Based on the surveyed PUF architectures, we summarize the opportunities for PUF designs based on emerging nanoelectronic devices.

a) **Small Footprint.** As the fabrication process scales down to nano dimensions, the uncontrollable process variations become more prevalent, which is desirable for PUF designs. The footprint of these nanodevices is smaller, leading to higher density information that can be integrated in a smaller integrated circuit. For example, SHIC PUF takes advantage of such a high density property offered by nanocrossbar arrays. The small footprint is desirable for securing resource constrained computing platforms such as RFID tags when taking lower area overhead into consideration.

b) **Low Cost Fabrication.** The fabrication of a majority of these nanoscale devices is compatible with current CMOS fabrication processes. In addition, the fabrication complexity is low. Furthermore, their fabrication cost is expected to be lower in the long term when these devices become mainstream.

c) **C2C Variations.** The unique C2C variations observed in the case of memristors can be exploited to construct rPUFs without any extra circuitry overhead offering refreshable CRPs that still retain a PUF instance's original security properties. In addition, the reconfiguration operation is impossible to be reversed by all parties, which is one of the key requirements in the design of a secure rPUF. As for rPUFs built upon the C2C variations shown in [34] and [31], future works need to investigate the dominance of C2C variations with respect to process variations to better understand the security properties of reconfigurable PUFs.

d) **Programming Sensitivity.** In contrast to memristor-based rPUFs relying on C2C programming variations, the MLB capability and the programming sensitivity create opportunities to build rPUFs, eg. [19], [22], [75], based on PCM and STT-MRAM.

e) **Nonvolatility.** Nonvolatility can be employed to reliably regenerate PUF responses such as in [22], [24], [29]. In addition, it can double memory as a key generator without additional buffer storage and other affiliated circuitry that is more cost-effective in comparison with conventional memory based PUFs such as SRAM PUF [23].

f) **Bidirectionality.** The nanodevices aforementioned in this paper are all two terminal devices. Hence they are bidirectional unlike PMOS/NMOS devices where input and output cannot be reversed and therefore facilitates the design of novel nano PPUFs [28], [87], [88].

g) **Formation Process.** The formation process of memristors is a one-time irreversible process. Therefore, the formation process [94] before they can be programmed normally between HRS and LRS can eliminate security issues caused by untrusted manufacturers: a manufacturer that measure CRPs without authority. If the formation process is only authorized to be carried out by a trusted party, then only a trusted party is able to safely collect CRPs. If the manufacturer attempts to collect CRPs, it will be discovered by the trusted party. Hence, formation is a benefit to PUF security in terms of tamper detection.

h) **Security.** Further, the PUF structures based on emerging nanoelectronic devices may circumvent some security issues challenging current silicon based PUFs. For example, the SRAM PUF has been broken in a non-invasive way through a fault injection attack based on remanence decay in volatile memory [95], and semi-invasive and fully invasive attack methods [96], [97]. Furthermore, it has been demonstrated that APUF and its variants can be physically characterized by means of photonic emission analysis in CMOS [98].

B. Challenges

Since these nanoelectronic devices are emerging devices, technologies themselves have challenging issues to overcome, and so do PUF designs based on these devices.

a) **Experimental Validation.** A majority of the proposed PUF structures are not experimentally validated. There are studies that show limited experimental results [16], [34], [75]. However, the experimental results only evaluate part of the PUF architecture. For example, the nanodevice characteristics such as resistance variation are based on experiments, whereas an efficient peripheral circuit to extract secret information are only proposed but not experimentally implemented to validate the entire PUF design.

TABLE II
PERFORMANCE COMPARISON OF SURVEYED EMERGING PUFS WITH NANOTECHNOLOGY

		Randomness	Uniqueness	BER[1]	Number of CRPs[2]	Weak PUF	Strong PUF	rPUF	PPUF
	Ideal PUF	50%	50%	0%	Exponential		✓	✓	✓
	SHIC PUF [16]	-	-	-	Linear		✓		
	CNPUF [15]	-	49.67%	3.5% (−20°C–80°C)	exponential		✓		
PCM based PUFs	PCM-based rPUF [17]	-	-	-	Linear	✓		✓	
	PCM-based RPUF [75][3]	-	≈ 50%	9.7% (25°C–85°C)	Linear	✓		✓	
	PCKGen [18]	-	≈ 50%	11.26% (7°C–87°C) V(±8.3%)	Linear	✓		✓	
	MemPUF [19][4]	-	-	15%	Linear	✓		✓	
STT-MRAM based PUFs	[20]	-	-	-	Linear	✓			
	[21][5]	49.62%	49.99%	13.5% (0°C–100°C) V(±10%)	Linear	✓			
	STT-PUF[6] [22]	-	50.01%	6.6×10^{-6} (−23°C–127°C) V(±10%)	Linear	✓			
	BF-MPUF [23]	49.90%	49.86%	6.60×10^{-6} (after error correction)	Linear	✓			
	[24][7]	-	49.9%	4.60×10^{-6} (−40°C–85°C) V(±10%)	Linear	✓		✓	
Memristor based PUFs	[25]	49.99%	≈ 50%	-	Linear	✓			
	[26]	≈ 50%	-	-	Linear	✓			
	[27]	50.15%	49.96%	-	Exponential		✓		
	Nano-PPUF [28]	-	49%	49%	Exponential		✓		✓
	[29]	-	-	-	Linear	✓			
	[30][8]	40%–60%	47%	-	Linear	✓		✓	
	[31]	≈ 50%	≈ 50%	10% (52°C–127°C) V(±10%)	Linear	✓		✓	
	[32]	53.8%	50.03%	1.5%[9]	Exponential		✓		
	mrPUF [33]	50.17%	49.96%	4.4% (−20°C–80°C) V(±20%)	Polynomial (2 orders)	✓			
	mrSPUF [34]	50.07%	50.76%	7.5% (−20°C-85°C) V(±20%)	Polynomial (10 orders)[10]		✓	✓	

[1] BER is bit error rate under different ambient environments. Here, we refer to worst-case BER under maximum temperature or power supply voltage deviation from the nominal value.

[2] Linear implies that the number of CRPs is linear with the number of basic components/cells that generate one bit response such as in a SRAM PUF. Polynomial means that PUFs generate a polynomial number of CRPs with regard to basic components such as in a conventional RO PUF. Exponential means that the PUF is able to produce an exponential number of CRPs such as in an APUF. Notably, as for SHIC PUF, even though the CRP number is linear, full characterization of its CRPs is difficult due to the slow readout speed and large information contained.

[3] There are two reconfigurable PUF structures proposed in this work, the performance is from one of two proposed PUF structures that shows better overall performance.

[4] This work based on [75] is used to avoid information leakage from MemPUFs (Memory based PUFs), so the evaluation of PUF performance on different metrics is not the focus of this work.

[5] In this work, 64 STT MRAM cells are *active cells* that are responsible for producing responses, while the other 64 STT MRAM cells are *reference cells* that are responsible for generating reference currents.

[6] The evaluation of randomness is omitted, but this paper shows that the entropy per response/cell is 0.958. The demonstrated BER is obtained by employing AWB (Automatic Write Back).

[7] All MTJs are initially set to the AP state. A low BER is obtained by employing AWB, while the BER is as high as 21% before employing AWB.

[8] The randomness of responses has different biases according to its location within an array.

[9] The descriptions of environmental conditions to evaluate the reliability are inconsistent in this work. Furthermore, the temperature dependent device model of a memristor is not described in this work.

[10] This order can be even higher by changing the configuration of the mrSPUF.

b) **PUF Performance Evaluation.** Statistical analysis on trade-offs among uniqueness, randomness, BER, resilience to model building attacks and side-channel attacks, and overhead costs are expected to be taken into consideration for a more comprehensive evaluation of PUF performance. Performance evaluations of some memristor-based PUF designs are based on simple device behavioral models [86] instead of *physical compact models* [99] without carefully taking other inherent characteristics of nanodevices into consideration. For example, C2C variations, which can potentially cause reliability issues when regenerating PUF responses are not investigated in [25], [26], [32]. These studies assume a memristor to return to its original resistance value when the same programming parameters, eg. duration or amplitude of programming pulse, are employed to reprogram it.

c) **Standard Models.** There are no industry standard models for these emerging nanodevices. Therefore, results obtained through simulations may be speculative without experimental implementation and validation. So collaboration between hardware security designers and device engineers are essential to advance these PUF designs and finally push forward these designs from laboratory work to practical applications.

d) **Reliability.** For memristor-based PPUFs, reliability issue is crucial because the response produced from a physical PPUF device and that from a simulator may be vastly different due to the poor reliability of the PPUF device. Unlike conventional PUF types, normal reliability enhancement techniques such as ECC are not directly applicable to PPUF (or may require an extremely high redundancy for an acceptable error correction capability) owing to the complexity of PPUF design. Similarly, specific problems may also exist for other nano technology based PUFs where repeated programing of nanodevices may deteriorate the reliability of PUFs, especially those requiring RESET operations (eg. [20], [32]), due to the specific writing endurance capability of nanodevices—number of switching cycles nanodevices can reversibly and reliably perform. In addition, the lack of temperature dependent memristor device models, as highlighted in [27], makes some reliability results speculative—eg. [32], where the temperature dependent coefficient is not described.

V. Conclusion

In this paper, we survey recent emerging PUF designs with nanotechonolgy. These PUF designs provide a number of unique properties such as abundance of process variations, bidirectionality, C2C variations and formation process to realize different PUF designs and architectures. We highlighted a number of opportunities offered by these nanodevices that can potentially be exploited by PUF designers. In addition, we have discussed major limitations of currently proposed PUF designs that are based on emerging nanoelelctronics. These limitations are expected to be addressed through the combined effort between PUF designers and device engineers. Continued progress in emerging PUF designs with nanotechnology will secure future memory and circuit applications with low energy and area overhead, while taking advantage of unique device level properties offered by nanodevices.

Acknowledgment

This research was supported by a grant from the Australian Research Council (DP140103448). DA acknowledges funding from the Australian Research Council Future Fellowship scheme (FT100100585). The authors also appreciate sponsorship from the China Scholarship Council (201306070017).

References

[1] J. Bonomi and P. E. Botta, *Nineveh and its Palaces. The discoveries of Botta and Layard applied to the elucidation of Holy Writ.* Illustrated London Library, 1852.

[2] O. Kömmerling and M. G. Kuhn, "Design principles for tamper-resistant smartcard processors," in *USENIX Workshop on Smartcard Technology*, vol. 12, 1999, pp. 9–20.

[3] P. S. Ravikanth, "Physical one-way functions," Ph.D. dissertation, Massachusetts Institute of Technology, 2001.

[4] R. Pappu, B. Recht, J. Taylor, and N. Gershenfeld, "Physical one-way functions," *Science*, vol. 297, no. 5589, pp. 2026–2030, 2002.

[5] B. Gassend, D. Clarke, M. Van Dijk, and S. Devadas, "Silicon physical random functions," in *Proceedings of the 9th ACM Conference on Computer and Communications Security*, 2002, pp. 148–160.

[6] G. E. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," in *Proceedings of the 44th Annual Design Automation Conference*, 2007, pp. 9–14.

[7] D. Lim, J. W. Lee, B. Gassend, G. E. Suh, M. Van Dijk, and S. Devadas, "Extracting secret keys from integrated circuits," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 13, no. 10, pp. 1200–1205, 2005.

[8] D. E. Holcomb, W. P. Burleson, and K. Fu, "Initial SRAM state as a fingerprint and source of true random numbers for RFID tags," in *Proc. Conference on RFID Security*, Malaga, Spain, art. no. 1.2, 2007.

[9] K. Lofstrom, W. R. Daasch, and D. Taylor, "IC identification circuit using device mismatch," in *Proc. IEEE International Solid-State Circuits Conference*, 2000, pp. 372–373.

[10] M. Rostami, J. B. Wendt, M. Potkonjak, and F. Koushanfar, "Quo vadis, PUF?: Trends and challenges of emerging physical-disorder based security," in *Proc. Conference on Design, Automation & Test in Europe*. European Design and Automation Association, 2014, p. 352.

[11] H.-S. P. Wong, S. Raoux, S. Kim, J. Liang, J. P. Reifenberg, B. Rajendran, M. Asheghi, K. E. Goodson *et al.*, "Phase change memory," *Proceedings of the IEEE*, vol. 98, no. 12, pp. 2201–2227, 2010.

[12] Y. Huai, "Spin-transfer torque MRAM (STT-MRAM): Challenges and prospects," *AAPPS Bulletin*, vol. 18, no. 6, pp. 33–40, 2008.

[13] J. Deng and H. P. Wong, "A compact SPICE model for carbon-nanotube field-effect transistors including nonidealities and its applicationpart i: Model of the intrinsic channel region," *IEEE Transactions on Electron Devices*, vol. 54, no. 12, pp. 3186–3194, 2007.

[14] D. B. Strukov, G. S. Snider, D. R. Stewart, and R. S. Williams, "The missing memristor found," *Nature*, vol. 453, no. 7191, pp. 80–83, 2008.

[15] S. Konigsmark, L. K. Hwang, D. Chen, and M. D. Wong, "CNPUF: A carbon nanotube-based physically unclonable function for secure low-energy hardware design," in *Proc. IEEE/ACM Asia and South Pacific Des. Autom. Conf.*, 2014, pp. 73–78.

[16] U. Ruhrmair, C. Jaeger, M. Bator, M. Stutzmann, P. Lugli, and G. Csaba, "Applications of high-capacity crossbar memories in cryptography," *IEEE Transactions on Nanotechnology*, vol. 10, no. 3, pp. 489–498, 2011.

[17] K. Kursawe, A. Sadeghi, D. Schellekens, B. Skoric, and P. Tuyls, "Reconfigurable physical unclonable functions-enabling technology for tamper-resistant storage," in *IEEE International Workshop on Hardware-Oriented Security and Trust*, 2009, pp. 22–29.

[18] L. Zhang, Z. H. Kong, and C.-H. Chang, "PCKGen: a phase change memory based cryptographic key generator," in *IEEE International Symposium on Circuits and Systems* (ISCAS), 2013, pp. 1444–1447.

[19] L. Zhang, C.-H. Chang, A. Cabrini, G. Torelli, and Z. H. Kong, "Leakage-resilient memory-based physical unclonable function using phase change material," in *Proc. IEEE Int. Carnahan Conference on Security Technology* (ICCST), 2014, DOI:10.1109/CCST.2014.6987047.

[20] T. Marukame, T. Tanamoto, and Y. Mitani, "Extracting physically unclonable function from spin transfer switching characteristics in magnetic tunnel junctions," *IEEE Transactions on Magnetics*, vol. 50, no. 11, art. no. 3402004, 2014.

[21] E. I. Vatajelu, G. Di Natale, M. Indaco, and P. Prinetto, "STT MRAM-Based PUFs," in *Proc. Design, Automation & Test in Europe Conference & Exhibition*. EDA Consortium, 2015, pp. 872–875.

[22] L. Zhang, X. Fong, C.-H. Chang, Z. H. Kong, and K. Roy, "Highly reliable memory-based physical unclonable function using Spin-Transfer Torque MRAM," in *IEEE International Symposium on Circuits and Systems* (ISCAS), 2014, pp. 2169–2172.

[23] L. Zhang, X. Fong, C.-H. Chang, Z. Kong, and K. Roy, "Optimizating emerging non-volatile memories for dual-mode applications: Data storage and key generator," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2015, DOI:10.1109/TCAD.2015.2427251.

[24] L. Zhang, X. Fong, C.-H. Chang, Z. H. Kong, and K. Roy, "Highly reliable spin-transfer torque magnetic RAM based physical unclonable function with multi-response-bits per cell," *IEEE Transactions on Information Forensics and Security*, pp. 1630–1642, 2015, DOI:10.1109/TIFS.2015.2421481.

[25] G. S. Rose, N. McDonald, L.-K. Yan, and B. Wysocki, "A write-time based memristive PUF for hardware security applications," in *IEEE/ACM International Conference on Computer-Aided Design* (ICCAD), 2013, pp. 830–833.

[26] P. Koeberl, Ü. Kocabaş, and A.-R. Sadeghi, "Memristor PUFs: a new generation of memory-based physically unclonable functions," in *Proceedings of the Conference on Design, Automation and Test in Europe*. EDA Consortium, 2013, pp. 428–431.

[27] G. S. Rose and C. A. Meade, "Performance analysis of a memristive crossbar PUF design," in *Proc. 2015 52nd ACM/EDAC/IEEE Design Automation Conference* (DAC), 2015, DOI: 10.1145/2744769.2744892.

[28] J. Rajendran, G. S. Rose, R. Karri, and M. Potkonjak, "Nano-PPUF: A memristor-based security primitive," in *IEEE Computer Society Annual Symposium on VLSI* (ISVLSI), 2012, pp. 84–87.

[29] W. Che, J. Plusquellic, and S. Bhunia, "A non-volatile memory based physically unclonable function without helper data," in *Proc. IEEE/ACM Int. Conf. on Computer-Aided Design* (ICCAD), 2014, pp. 148–153.

[30] A. Chen, "Reconfigurable physical unclonable function based on probabilistic switching of RRAM," *Electronics Letters*, vol. 51, no. 8, pp. 615–617, 2015.

[31] A. Chen, "Utilizing the variability of resistive random access memory to implement reconfigurable physical unclonable functions," *IEEE Electron Device Letters*, vol. 36, no. 2, pp. 138–140, 2015.

[32] J. Mathew, R. S. Chakraborty, D. P. Sahoo, Y. Yang, and D. K. Pradhan, "A novel memristor-based hardware security primitive," *ACM Transactions on Embedded Computing Systems* (TECS), vol. 14, no. 3, p. 60, 2015.

[33] Y. Gao, D. C. Ranasinghe, S. F. Al-Sarawi, O. Kavehei, and D. Abbott, "mrPUF: A novel memristive device based physical unclonable function," in *13th International Conference on Applied Cryptography and Network Security* (ACNS), 2015, DOI: 10.1007/978-3-319-28166-7 29.

[34] Y. Gao, D. C. Ranasinghe, S. F. Al-Sarawi, O. Kavehei, and D. Abbott, "Memristive crypto primitive for building highly secure physical unclonable functions," *Scientific Reports*, 2015, DOI:10.1038/srep12785.

[35] D. Lim, "Extracting secret keys from integrated circuits," Ph.D. dissertation, Massachusetts Institute of Technology, 2004.

[36] U. Ruhrmair and M. Van Dijk, "PUFs in security protocols: Attack models and security evaluations," in *IEEE Symposium on Security and Privacy* (SP), 2013, pp. 286–300.

[37] B. Gassend, D. Lim, D. Clarke, M. Van Dijk, and S. Devadas, "Identification and authentication of integrated circuits," *Concurrency and Computation: Practice and Experience*, vol. 16, no. 11, pp. 1077–1098, 2004.

[38] A. Maiti and P. Schaumont, "Improving the quality of a physical unclonable function using configurable ring oscillators," in *Proc. IEEE Int. Conference on Field Programmable Logic and Applications*, 2009, pp. 703–707.

[39] A. Maiti, I. Kim, and P. Schaumont, "A robust physical unclonable function with enhanced challenge-response set," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 1, pp. 333–345, 2012.

[40] M. Gao, K. Lai, and G. Qu, "A highly flexible ring oscillator PUF," in *Proc. 51st ACM/EDAC/IEEE Design Automation Conference* (DAC), 2014, DOI:10.1145/2593069.2593072.

[41] J.-L. Zhang, G. Qu, Y.-Q. Lv, and Q. Zhou, "A survey on silicon PUFs and recent advances in ring oscillator PUFs," *Journal of Computer Science and Technology*, vol. 29, no. 4, pp. 664–678, 2014.

[42] D. E. Holcomb, W. P. Burleson, and K. Fu, "Power-up SRAM state as an identifying fingerprint and source of true random numbers," *IEEE Transactions on Computers*, vol. 58, no. 9, pp. 1198–1210, 2009.

[43] Y. Su, J. Holleman, and B. P. Otis, "A digital 1.6 pJ/bit chip identification circuit using process variations," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 1, pp. 69–77, 2008.

[44] R. Maes, P. Tuyls, and I. Verbauwhede, "Intrinsic PUFs from flip-flops on reconfigurable devices," in *3rd Benelux Workshop on Information and System Security* (WISSec), Eindhoven, The Netherlands, 2008, https://www.researchgate.net/publication/228615879.

[45] V. van der Leest, G.-J. Schrijen, H. Handschuh, and P. Tuyls, "Hardware intrinsic security from D flip-flops," in *Proc. 5th ACM Workshop on Scalable Trusted Computing*, 2010, pp. 53–62.

[46] S. S. Kumar, J. Guajardo, R. Maes, G.-J. Schrijen, and P. Tuyls, "The butterfly PUF protecting IP on every FPGA," in *IEEE International Workshop on Hardware-Oriented Security and Trust* (HOST), 2008, pp. 67–70.

[47] M. Majzoobi, G. Ghiaasi, F. Koushanfar, and S. R. Nassif, "Ultra-low power current-based PUF," in *Proc. IEEE Int. Sym on Circuits and Systems* (ISCAS), 2011, pp. 2071–2074.

[48] R. Kumar and W. Burleson, "On design of a highly secure PUF based on non-linear current mirrors," in *Proc. IEEE Int. Sym on Hardware-Oriented Security and Trust* (HOST), 2014, pp. 38–43.

[49] M. Roel, "Physically unclonable functions: Constructions, properties and applications," Ph.D. dissertation, University of KU Leuven, 2012.

[50] C. Herder, M.-D. Yu, F. Koushanfar, and S. Devadas, "Physical unclonable functions and applications: A tutorial," *Proceedings of IEEE*, vol. 102, no. 8, pp. 1126–1141, 2014.

[51] U. Rührmair, F. Sehnke, J. Sölter, G. Dror, S. Devadas, and J. Schmidhuber, "Modeling attacks on physical unclonable functions," in *Proc. 17th ACM Conference on Computer and Communications Security*, 2010, pp. 237–249.

[52] P. Tuyls, G.-J. Schrijen, B. Škorić, J. Van Geloven, N. Verhaegh, and R. Wolters, "Read-proof hardware from protective coatings," in *Cryptographic Hardware and Embedded Systems* (CHES). Springer, 2006, pp. 369–383.

[53] S. Katzenbeisser, Ü. Kocabaş, V. Van Der Leest, A.-R. Sadeghi, G.-J. Schrijen, and C. Wachsmann, "Recyclable PUFs: Logically reconfigurable PUFs," *Journal of Cryptographic Engineering*, vol. 1, no. 3, pp. 177–186, 2011.

[54] M. Potkonjak and V. Goudar, "Public physical unclonable functions," *Proceedings of the IEEE*, vol. 102, no. 8, pp. 1142–1156, 2014.

[55] Y. Hori, T. Yoshida, T. Katashita, and A. Satoh, "Quantitative and statistical performance evaluation of arbiter physical unclonable functions on FPGAs," in *Proc. IEEE Int. Conf. Reconfiguratble Computing and FPGAs* (ReConFig), 2010, pp. 298–303.

[56] A. Maiti, V. Gunreddy, and P. Schaumont, "A systematic method to evaluate and compare the performance of physical unclonable functions," in *Embedded Systems Design with FPGAs*. Springer, 2013, pp. 245–267.

[57] T. Rahman, D. Forte, J. Fahrny, and M. Tehranipoor, "ARO-PUF: An aging-resistant ring oscillator PUF design," in *Proc. Conf. on Design, Automation & Test in Europe*. European Design and Automation Association, 2014, p. 69.

[58] Y. Cao, L. Zhang, C.-H. Chang, and S. Chen, "A low-power hybrid RO PUF with improved thermal stability for lightweight applications," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 7, pp. 1143–1147, 2015.

[59] R. Maes, A. Van Herrewege, and I. Verbauwhede, "PUFKY: A fully functional PUF-based cryptographic key generator," in *Cryptographic Hardware and Embedded Systems* (CHES). Springer, 2012, pp. 302–319.

[60] S. Devadas and M. Yu, "Secure and robust error correction for physical unclonable functions," *IEEE Design and Test*, 2013, DOI:10.1109/MDT.2009.163.

[61] J. Delvaux, D. Gu, D. Schellekens, and I. Verbauwhede, "Helper data algorithms for PUF-based key generation: Overview and analysis," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 6, pp. 889–902, 2015.

[62] A. Van Herrewege, A. Schaller, S. Katzenbeisser, and I. Verbauwhede, "DEMO: Inherent PUFs and secure PRNGs on commercial off-the-shelf

microcontrollers," in *Proc. ACM SIGSAC Conference on Computer and Communications Security*, 2013, pp. 1333–1336.

[63] S. Schulz, A.-R. Sadeghi, and C. Wachsmann, "Short paper: lightweight remote attestation using physical functions," in *Proc. Fourth ACM Conference on Wireless Network Security*, 2011, pp. 109–114.

[64] J. Kong, F. Koushanfar, P. K. Pendyala, A.-R. Sadeghi, and C. Wachsmann, "PUFatt: Embedded platform attestation based on novel processor-based PUFs," in *51st ACM/EDAC/IEEE Design Automation Conference (*DAC), San Francisco, CA, 2014, DOI: 10.1145/2593069.2593192.

[65] D. C. Ranasinghe, D. Engels, and P. Cole, "Security and privacy: Modest proposals for low-cost RFID systems," in *Auto-ID Labs Research Workshop,* Zurich, Switzerland, 2004.

[66] P. H. Cole and D. C. Ranasinghe, *Networked RFID Systems and Lightweight Cryptography*. Springer-Verlag, 2008.

[67] D. C. Ranasinghe and P. H. Cole, "Confronting security and privacy threats in modern RFID systems," in *Proceedings of the IEEE 40th Asilomar Conference on Signals, Systems and Computers*, 2004, pp. 2058–2064.

[68] A. Van Herrewege, S. Katzenbeisser, R. Maes, R. Peeters, A.-R. Sadeghi, I. Verbauwhede, and C. Wachsmann, "Reverse fuzzy extractors: Enabling lightweight mutual authentication for PUF-enabled RFIDs," in *Financial Cryptography and Data Security*. Springer, 2012, pp. 374–389.

[69] M. Rostami, M. Majzoobi, F. Koushanfar, D. S. Wallach, and S. Devadas, "Robust and reverse-engineering resilient PUF authentication and key-exchange by substring matching," *IEEE Transactions on Emerging Topics in Computing*, vol. 2, no. 1, pp. 37–49, 2014.

[70] M.-D. Yu, D. M'Raihi, I. Verbauwhede, and S. Devadas, "A noise bifurcation architecture for linear additive physical functions," in *2014 IEEE International Symposium on Hardware-Oriented Security and Trust* (HOST), 2014, pp. 124–129.

[71] Y. Gao, D. C. Ranasinghe, G. Li, S. F. Al-Sarawi, O. Kavehei, and D. Abbott, "A challenge obfuscation method for thwarting model building attacks on PUFs," Cryptology ePrint Archive, 2015, https://eprint.iacr.org/2015/471.pdf.

[72] J. Delvaux, D. Gu, D. Schellekens, and I. Verbauwhede, "Secure lightweight entity authentication with strong PUFs: Mission impossible?" in *Cryptographic Hardware and Embedded Systems (*CHES). Springer, 2014, pp. 451–475.

[73] M. van Dijk and U. Rührmair, "Physical unclonable functions in cryptographic protocols: Security proofs and impossibility results." *IACR Cryptology ePrint Archive*, vol. 2012, p. 228, 2012.

[74] Y. Gao, "Secure key exchange protocol based on virtual proof of reality," Cryptology ePrint Archive, 2015, https://eprint.iacr.org/2015/524.pdf.

[75] L. Zhang, Z. H. Kong, C.-H. Chang, A. Cabrini, and G. Torelli, "Exploiting process variations and programming sensitivity of phase change memory for reconfigurable physical unclonable functions," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 6, pp. 921–932, 2014.

[76] I. Eichhorn, P. Koeberl, and V. van der Leest, "Logically reconfigurable PUFs: Memory-based secure key storage," in *Proc. 6th ACM Workshop on Scalable Trusted Computing*, 2011, pp. 59–64.

[77] M. Arafin, C. Dunbar, G. Qu, N. McDonald, and L. Yan, "A survey on memristor modeling and security applications," in *Proc. IEEE Int. Symp. on Quality Electronic Design* (ISQED), 2015, pp. 440–447.

[78] J. Rajendran, R. Karri, J. B. Wendt, M. Potkonjak, N. McDonald, G. S. Rose, and B. Wysocki, "Nano meets security: Exploring nanoelectronic devices for security applications," *Proceedings of IEEE*, vol. 103, no. 5, pp. 829–849, 2015.

[79] A. Mazady, M. T. Rahman, D. Forte, and M. Anwar, "Memristor PUF—a security primitive: Theory and experiment," *IEEE Journal ON Emerging and Selected Topics in Circuits and Systems*, vol. 5, no. 2, pp. 222–229, 2015.

[80] C. Jaeger, M. Algasinger, U. Rührmair, G. Csaba, and M. Stutzmann, "Random pn-junctions for physical cryptography," *Applied Physics Letters*, vol. 96, no. 17, art. no. 172103, 2010.

[81] Y. Morita, H. Fujiwara, H. Noguchi, Y. Iguchi, K. Nii, H. Kawaguchi, and M. Yoshimoto, "An area-conscious low-voltage-oriented 8T-SRAM design under DVS environment," in *Proc. 2007 IEEE Symposium on VLSI Circuits*, 2007, pp. 256–257.

[82] X. Fong, S. H. Choday, and K. Roy, "Bit-cell level optimization for non-volatile memories using magnetic tunnel junctions and spin-transfer torque switching," *IEEE Transactions on Nanotechnology*, vol. 11, no. 1, pp. 172–181, 2012.

[83] K.-H. Kim, S. Gaba, D. Wheeler, J. M. Cruz-Albrecht, T. Hussain, N. Srinivasa, and W. Lu, "A functional hybrid memristor crossbar-array/CMOS system for data storage and neuromorphic applications," *Nano Letters*, vol. 12, no. 1, pp. 389–395, 2011.

[84] K. Eshraghian, O. Kavehei, K.-R. Cho, J. M. Chappell, A. Iqbal, S. F. Al-Sarawi, and D. Abbott, "Memristive device fundamentals and modeling: applications to circuits and systems simulation," *Proceedings of the IEEE*, vol. 100, no. 6, pp. 1991–2007, 2012.

[85] G. S. Rose, N. McDonald, L.-K. Yan, B. Wysocki, and K. Xu, "Foundations of memristor based PUF architectures," in *IEEE/ACM International Symposium on Nanoscale Architectures* (NANOARCH), 2013, pp. 52–57.

[86] Y. Gao, O. Kavehei, S. F. Al-Sarawi, D. C. Ranasinghe, and D. Abbott, "Read operation performance of large selectorless cross-point array with self-rectifying memristive device," *Integration, the VLSI journal*, 2016.

[87] J. B. Wendt and M. Potkonjak, "Nanotechnology-based trusted remote sensing," in *Proc. 2011 IEEE Sensors*, 2011, pp. 1213–1216.

[88] J. B. Wendt and M. Potkonjak, "The bidirectional polyomino partitioned PPUF as a hardware security primitive," in *IEEE Global Conference on Signal and Information Processing* (GlobalSIP), 2013, DOI:10.1109/GlobalSIP.2013.6736864.

[89] S. Wu, L. Ren, J. Qing, F. Yu, K. Yang, M. Yang, Y. Wang, M. Meng, W. Zhou, X. Zhou *et al.*, "Bipolar resistance switching in transparent ITO/LaAlO3/SrTiO3 memristors," *ACS Applied Materials & Interfaces*, vol. 6, no. 11, pp. 8575–8579, 2014.

[90] N. R. McDonald, "Al/CuxO/Cu memristive devices: Fabrication, characterization, and modeling," DTIC Document, Tech. Rep., 2012.

[91] D. C. Ranasinghe, D. Lim, S. Devadas, D. Abbott, and P. Cole, "Random numbers from metastability and thermal noise," *Electronics Letters*, vol. 41, no. 16, pp. 891–893, 2005.

[92] J. Rajendran, R. Karri, and G. S. Rose, "Improving tolerance to variations in memristor-based applications using parallel memristors," *IEEE Transactions on Computers,*, vol. 64, no. 3, pp. 733–746, 2015.

[93] S. Choi, Y. Yang, and W. Lu, "Random telegraph noise and resistance switching analysis of oxide based resistive memory," *Nanoscale*, vol. 6, no. 1, pp. 400–404, 2014.

[94] J. P. Strachan, D. B. Strukov, J. Borghetti, J. J. Yang, G. Medeiros-Ribeiro, and R. S. Williams, "The switching location of a bipolar memristor: chemical, thermal and structural mapping," *Nanotechnology*, vol. 22, no. 25, art. no. 254015, 2011.

[95] Y. Oren, A.-R. Sadeghi, and C. Wachsmann, "On the effectiveness of the remanence decay side-channel to clone memory-based PUFs," in *Cryptographic Hardware and Embedded Systems* (CHES). Springer, 2013, pp. 107–125.

[96] C. Helfmeier, C. Boit, D. Nedospasov, and J.-P. Seifert, "Cloning physically unclonable functions," in *Proc. IEEE Int. Symposium on Hardware-Oriented Security and Trust* (HOST), 2013, DOI:10.1109/HST.2013.6581556.

[97] D. Nedospasov, J.-P. Seifert, C. Helfmeier, and C. Boit, "Invasive PUF analysis," in *2013 Workshop on Fault Diagnosis and Tolerance in Cryptography* (FDTC). IEEE, 2013, pp. 30–38.

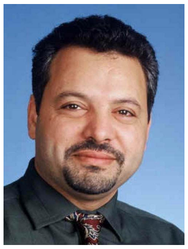
[98] S. Tajik, E. Dietz, S. Frohmann, J.-P. Seifert, D. Nedospasov, C. Helfmeier, C. Boit, and H. Dittrich, "Physical characterization of arbiter PUFs," in *Cryptographic Hardware and Embedded Systems* (CHES). Springer, 2014, pp. 493–509.

[99] S. Kim, S. Choi, and W. Lu, "Comprehensive physical model of dynamic resistive switching in an oxide memristor," *ACS Nano*, vol. 8, no. 3, pp. 2369–2376, 2014.

Yansong Gao (S'15) received his B.Sc degree in Electronic Information Science and Technology from Henan University of Science and Technology, Luoyang, China in 2008 and M.Sc in Signal Information Processing from University of Electronic Science and Technology of China, ChengDu, China in 2013. He is currently working toward the Ph.D degree at the School of Electrical and Electronic Engineering in the University of Adelaide.

His current research interests are hardware security circuit design and its applications, and memristive device characterization.

Damith C. Ranasinghe received his Ph.D. degree in electrical and electronic engineering from the University of Adelaide, Australia, in 2007. In 2005–2006, he was a Visiting Scholar at Massachusetts Institute of Technology (MIT) and a postdoctoral research fellow at the University of Cambridge from 2007–2009. He joined the University of Adelaide in 2010 and is currently a tenured Senior Lecturer at the School of Computer Science where he leads the research group at the Adelaide Auto-ID Lab. His research interests include pervasive and ubiquitous computing, wearable computing, human activity recognition, gerontechnology, lightweight cryptography for resource constrained devices, and physical cryptography.



Said F. Al-Sarawi (S'92–M'96) received the general certificate in marine radio communication and the B.Eng. degree (first class honors) in marine electronics and communication from the Arab Academy for Science and Technology (AAST), Alexandria, Egypt, in 1987 and 1990, respectively, and the Ph.D. degree in mixed analog and digital circuit design techniques for smart wireless systems with special commendation in electrical and electronic engineering from The University of Adelaide, Adelaide, S.A., Australia, in 2003. He also received the Graduate Certificate in education (higher education), in 2006, from the same university.

Currently, he is the Director of the Centre for Biomedical Engineering and a founding member of Education Research Group of Adelaide (ERGA) in the University of Adelaide. His research interests include design techniques for mixed signal systems in complementary metal–oxide–semiconductor (CMOS) and optoelectronic technologies for high-performance radio transceivers, low-power and low-voltage radio-frequency identification (RFID) systems, data converters, mixed signal design, and microelectromechanical systems (MEMS) for biomedical applications. His current educational research is focused on innovative teaching techniques for engineering education, research skill development, and factors affecting students evaluations of courses in different disciplines.

Dr. Al-Sarawi was awarded The University of Adelaide Alumni Postgraduate Medal (formerly Culross Prize) for outstanding academic merit at the postgraduate level. While pursuing his Ph.D., he won the Commonwealth Postgraduate Research Award (Industry).

Omid Kavehei (S'05–M'12) is a Lecturer at the School of Electrical and Computer Engineering, Royal Melbourne Institute of Technology (RMIT), Melbourne, Australia. He received his M.Eng. from Shahid Beheshti University, Tehran, Iran and his Ph.D. in Electrical and electronic Engineering from The University of Adelaide, South Australia in 2012. He was a Research Fellow at the Centre for Neural Engineering, The University of Melbourne from 2011 to 2014. Dr. Kavehei was the 2013 recepient of The University of Adelaide's Postgraduate University Alumni Medal, a Doctoral Research Medal in 2012, and the South Australian Young Nanotechnology Ambassador award in 2011. His research interests include emerging computational paradigms with CMOS and beyond CMOS technologies, microelectronics, neural engineering and physical cryptography.

Derek Abbott (M'85–SM'99–F'05) was born in South Kensington, London, U.K., in 1960. He received the B.Sc. (honors) degree in physics from Loughborough University, Loughborough, Leicestershire, U.K., in 1982 and the Ph.D. degree in electrical and electronic engineering from The University of Adelaide, Adelaide, S.A. Australia, in 1995, under K. Eshraghian and B. R. Davis.

From 1978 to 1986, he was a Research Engineer at the GEC Hirst Research Centre, London, U.K. From 1986 to 1987, he was a VLSI Design Engineer at Austek Microsystems, Australia. Since 1987, he has been with The University of Adelaide, where he is presently a full Professor with the School of Electrical and Electronic Engineering. He coedited *Quantum Aspects of Life* (London, U.K.: Imperial College Press, 2008), coauthored *Stochastic Resonance* (Cambridge, U.K.: Cambridge University Press, 2012), and coauthored *Terahertz Imaging for Biomedical Applications* (New York, NY, USA: Springer-Verlag, 2012). He holds over 800 publications/patents and has been an invited speaker at over 100 institutions. His interests are in the area of multidisciplinary physics and electronic engineering applied to complex systems. His research programs span a number of areas of stochastics, game theory, photonics, biomedical engineering, and computational neuroscience.

Prof Abbott is a Fellow of the Institute of Physics (IOP) and a Fellow of the IEEE.. He has won a number of awards including the South Australian Tall Poppy Award for Science (2004), the Premier's SA Great Award in Science and Technology for outstanding contributions to South Australia (2004), an Australian Research Council (ARC) Future Fellowship (2012), and The David Dewhurst Medal (2015). He has served as an Editor and/or Guest Editor for a number of journals including the IEEE Journal of Solid-State Circuits, *Journal of Optics B*, the *Microelectronics Journal*, *Chaos*, *Smart Structures and Materials*, *Fluctuation and Noise Letters*, Proceedings of the IEEE, IEEE Photonics Journal, and *PLOSONE*. He is currently on the editorial boards of Nature's *Scientific Reports*, IEEE Access, *Royal Society Online Science* and *Frontiers in Physics*.