LEDPUF: Stability-Guaranteed Physical Unclonable Functions through Locally Enhanced Defectivity

Wei-Che Wang, Yair Yona, Suhas Diggavi and Puneet Gupta
Department of Electrical Engineering, University of California, Los Angeles
{weichewang, suhasdiggavi, yairyo99}@ucla.edu, puneet@ee.ucla.edu

***Abstract*—Stability has always been one of the major limitations that constraints Physical Unclonable Function (PUF) from being put in widespread practical use. In this paper, we propose a weak PUF and a strong PUF that are both completely stable with 0% intra-distance. These PUFs are called Locally Enhanced Defectivity Physical Unclonable Function (LEDPUF). A LEDPUF is a pure functional PUF which eliminates the instability of conventional parametric PUFs, therefore no helper data, fuzzy comparator, or any kinds of correction schemes are required. The source of randomness is extracted from Directed Self Assembly (DSA) process, and connections that are permanently closed or opened are formed randomly. The weak LEDPUF is constructed by forming arrays of DSA random connections, and the strong LEDPUF is implemented by using the weak LEDPUF as the key of a keyed-hash message authentication code (HMAC). Our simulation and statistical results show that the entropy of the weak LEDPUF bits is close to ideal, and the inter-distances of both weak and strong LEDPUFs are about 50%, which means that these LEDPUFs are not only stable but also unique.**

I. Introduction

A Physical Unclonable Function (PUF) is a small piece of circuitry such that its behavior, or Challenge Response Pair (CRP) [1], is uniquely defined and it is hard to be predicted and replicated because of the intrinsic random physical nature and the uncontrollability of process variations. As a security primitive, PUF can enable low overhead hardware identification, tracing, and authentication during the global manufacturing chain. The first PUF was introduced more than a decade ago [2]. Since then, many silicon PUF implementations have been proposed, such as Arbiter PUF [3], Ring Oscillator (RO) PUF [4], SRAM PUF [5], and many other variations. However, since the key commonality between all current silicon PUF implementations is their use of *parametric* manufacturing variations as the source of randomness, there exist several limitations that can cost expensive implementation overhead.

A. Limitations of Parametric PUFs

1) Random Local Variation Extraction: One of the major concerns of parametric PUFs is that local variation should be the *only* entropy source for these PUFs [6]. However, from our experiments on a large silicon data set [7], only 13% of total variation is random local variation, which means that most variation is coming from global or spatial variation. Any attempt to use global or spatial variation as the source of randomness can make them vulnerable to a class of *process side channel attacks*. For instance, two PUFs on the same (X,Y) location on different wafers are highly correlated (due to large wafer-level systematics present in most modern fabrication processes). As a result, a few sacrificial wafers can aid in developing a relatively straightforward side channel attack. We tested this side channel attack on silicon RO PUF measurements in 65nm technology across 300 wafers. Figure 1 shows that the inter-distance [6] on the same (X,Y) is much smaller than the inter-distance across all PUFs. Therefore, an adversary with possession of a reference PUF, which is fabricated at the same (X,Y) location as the target PUF, would have a higher probability of guessing the correct answer than random guessing. The radial nature of systematic across wafer variation [7] means that just a few reference PUFs drawn carefully may be sufficient for attackers instead of keeping full sacrificial wafers.

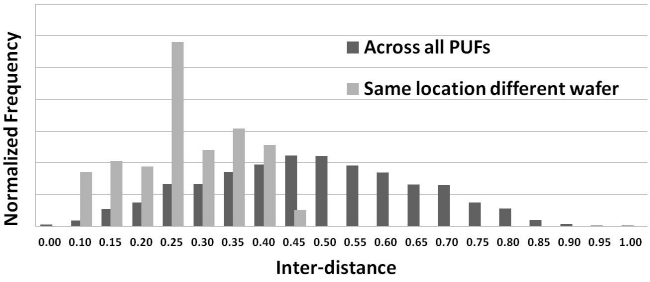

Fig. 1. The inter-distance of PUFs from same (X,Y) location on different wafers is much smaller than that of across all PUFs, which demonstrates a possible side channel attack.

2) Measurement Noise: Measurement noise could be another big issue for parametric PUFs and needs to be accounted for carefully. For instance, metastability of the arbiter circuit for Arbiter PUFs and accumulated jitter in RO PUFs can be sources of measurement noises. For weak PUF measurement, we evaluate the intra-distance [6] of SRAM PUFs using fifteen commercial 45nm SOI test chips, where each consists 176kB data memory. The power-up state is measured 10 times during an 8-hour period, and the mean of intra-distance distribution is 2.57%. Since the experiment is done in room temperature with exactly same settings, the difference is essentially contributed by the measurement noise.

3) Environmental fluctuations and wearout: Existing silicon PUFs are in nature susceptible to environmental fluctuations [8] and wearout [9]. To account for the instability issue, techniques such as error correction code (ECC), helper

This work was supported in part by NSF grants 1136174 and 1321120.

data or fuzzy comparator must be applied. A possible worse case scenario is that when the environmental factors change significantly but yet remain constant. For instance, the PUF is enrolled at 20°C and is verified at 80°C. In this case, a fuzzy extraction process may not be able to recover the initial PUF response, even for multiple samples of the PUF.

B. Techniques to Improve Parametric PUF Quality

A variety of techniques have been intensively studied over the years to extract random local variations or to make a PUF more stable and reliable. A Non-Volatile Memory (NVM) based PUF without helper data is presented in [10]. However, besides its hardware and calibration overhead, the results of uniqueness and entropy analysis are also missing. In [11], the local randomness is distilled by modeling and subtracting the systematic variation. A similar technique is to subtract the averaged frequency from multiple measurements to reveal the true local random variation [12]. However, the calculation and information storage requirement come with the cost of addition latency and hardware. Taking the majority vote [13] or finding stable responses [14] are possible techniques to eliminate the measurement noise, however, at the cost of large latency or reduced number of challenges. Other complex implementations have been proposed to mitigate stability issues that often induce lower hardware efficiency [6], additional circuit complexity [15], or making the PUF more susceptible to attacks [16]. Also, to protect PUFs from the worst case scenario as described creates a huge overhead as it requires to employ very strong ECC [17].

C. The LEDPUF

The issues of parametric PUFs, such as the described instability, wearout, measurement noise, limited local variation, and limited side channel attack resiliency, clearly motivate the need to design PUFs that do not rely on parametric performance variations as the entropy source. In this paper we propose a weak and a strong LEDPUF.

Rather than comparing parameter deviations, the response of an LEDPUF is stability-guaranteed because it depends on *random permanent connections* generated in Directed Self Assembly (DSA) process, which is highly compatible with existing CMOS technology and is expected to be used in manufacturing in the near future [18].

Compared to similar parametric PUFs such as hardware obfuscation [19] or digital PUFs [20], LEDPUF is completely stable and less susceptible to side channel attacks or model building attacks. The proposed LEDPUF is also a functional PUF where logic function itself is the signature and the strong LEDPUF can generate a variety of challenge-response pairs as needed. The Boolean nature of the response without any parametric dependence means that LEDPUF is not only immune to measurement noise and wearout, but also offers a greater level of reliability compared to existing PUFs as the output is resistant to changes in the environmental factors.

The contributions of this paper are:

- The first stability-guaranteed silicon PUF through locally enhanced defectivity is proposed.
- Detailed constructions of the weak LEDPUF using random DSA connections are presented. It is the first weak PUF with 0% intra-distance without using any stability enhancement techniques.
- The weak LEDPUF with 0% intra-distance enables the construction of the strong LEDPUF based on cryptographic hash functions.
- The simulation statistics and entropy calculation are presented. The results show that the proposed LEDPUFs can generate unique responses and their behaviors are hard to predict.

The rest of the paper is organized as follows: In Section II, we describe the randomness extraction from DSA and the DSA hard defective connection formation. In Section III, the structure of the stable signal unit (SSU) is first described, followed by the construction of the proposed weak and strong LEDPUFs from the SSU and the entropy analysis. In Section IV, the experimental results are presented, and finally, we conclude the paper in Section V.

II. DSA Randomness Extraction

Self-assembly is a mechanism that describes block copolymers (BCP) composed of immiscible blocks phase-separate into certain structures [21]. The guiding templates, which are used to *guide* the self-assembly process [22], can be lithographically-printed trenches (Graphoepitaxy) or chemically-treated surfaces (Chemoepitaxy). During the graphoepitaxy process for contact or via holes, the guiding templates are first lithographically printed, then the surface is spin-coated with the BCP solution. The phase separation occurs during the thermal annealing , and with a particular BCP and surface treatment of substrate [23], cylinders are formed of one block in a matrix of the other block [24].

In case of a diblock copolymer made of two blocks, say A and B: at equilibrium, the microphase separation is established by an energy balance between the stretching energy for the polymer chains and the energy of interactions at the interface between A and B microdomains [25]. Thermal equilibrium is achieved when the free energy is minimized, and the minimum energy state strongly depends on the level of confinement achieved by the layout of guiding templates. In other words, the size, shape, and critical dimension (CD) of the guiding template can greatly affect the DSA defect density [26–28].

For bigger-sized templates, it becomes energetically less expensive to induce a defect than to achieve a defect-free energy minimization [27, 29, 30]. Also, with less confinement forces from the guiding template due to its large size, random interactions from thermal fluctuation [31] or initial kinetics of collective density and state fluctuations [32] begin to dominate the assembly process. Therefore, final assembly results can be random by designing guiding templates that are large enough to cause random assembly errors even if there are no lithographic variations. Figure 2 shows three simulation results of the same large guiding template with an existing DSA simulator [33], where the model of the PS-b-PMMA copolymer has been validated in [34]. The three layers inside the polygonal guiding template are the top, middle, and bottom

layers of a via. If a cylindrical via hole is formed correctly, the three layers should be three overlapped concentric circles. However, for the large guiding template, random arrangement with different orientations begin to occur. In other words, the randomness of DSA is confined within predetermined local areas only by deliberately designing ”bad” guiding templates.

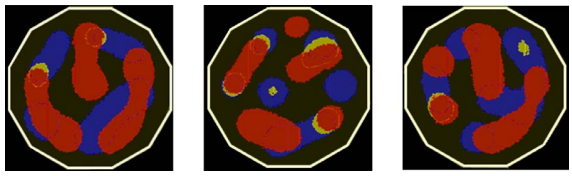

Fig. 2. Random via formations with a same large guiding template.

A. Hard Defective Connection Formation

We leverage the randomness extracted from DSA to form randomly assembled connections, and these connections are then used to fabricate LEDPUF. Though in conventional DSA, the goal of the guiding template design is to achieve high confinement and avoid regions of random phase transitions, we use the same theory but to enhance randomness in assembly. To construct a DSA random hard defective connection, we configure the size of the guiding template so that two vias are formed with a certain probability that they are connected permanently. A DSA hard defective connection is composed of the two vias along with the connection.

In our experiment, each simulation contains three guiding templates with a same shape, and two vias are formed in each of the guiding template. If the via pair in a same guiding template is merged, the DSA hard defective connection is in closed state; otherwise, the connection is in opened state. The states of the three connections in a simulation is independent with each other as expected in real DSA process [34]. In our statistical analysis, an open state is represented by a logic “1”, and a closed state is represented by a logic “0”. 500 simulation were performed in our experiments, so 1500 bits of raw data is obtained from the simulation. Based on our analysis, the empirical entropy of triples of bits is only 0.0063 bits smaller than the entropy of independent triple of bits. Examples of a simulation result in 2D and 3D views are shown in Figure 3 (a) and (b), respectively. In the 2D view, the rectangular shapes with rounding corners are the guiding templates, and the shapes inside of the guiding templates are the vias. If the via pair in a same guiding template is merged, the DSA hard defective connection is formed as shown in Figure 3 (c), and it is in permanent closed state; otherwise, the DSA hard defective connection is in permanent opened state as shown in Figure 3 (d). In Figure 3 (a) and (b), two hard defective connections are in opened state, and one connection is in closed state.

III. LEDPUF Construction

A. Weak LEDPUF Construction

The proposed weak LEDPUF is composed of arrays of SSUs. Each SSU is constructed from a DSA defective connection, which can be considered as random switches with

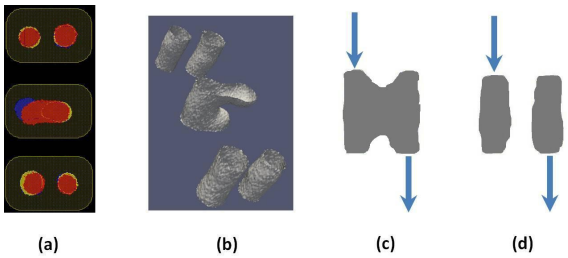

Fig. 3. (a) 2D view of 3 DSA hard defective connections on three pairs of vias. (b) 3D view of 3 DSA hard defective connections on three pairs of vias. The connections on the top and bottom are in permanent opened state; the middle one is in permanent closed state. (c) Vias are partialy merged, so the DSA hard defective connection is in closed state. (d) Vias are not merged, so the DSA hard defective connection is in opened state.

permanent states that determine the unique and stable function of the circuit. Figure 4 (a) shows the implementation of a SSU. Two ends of the DSA connection are connected to VDD and GND through opposite switches. Figure 4 (b) shows the abstraction of a SSU. In standby mode or before the evaluation, the evaluation signal EVA is low and the output is zero. During evaluation mode, EVA becomes high, and the output is either one or zero depending on the permanent state of the DSA connection. If the DSA connection is closed, the output is one; otherwise, the output is zero.

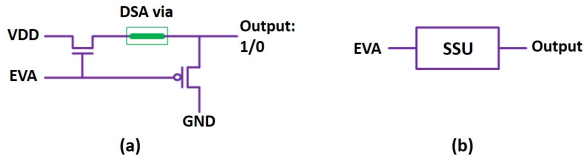

Fig. 4. (a) Stable signal unit implementation. When EVA is high, the output is either one or zero permanently depending on the state of the DSA via. (b) Abstraction of a SSU.

The proposed weak LEDPUF is constructed by arranging the SSUs in forms of arrays. Figure 5 illustrates a weak LEDPUF with n rows and m columns, where the number of SSUs is nm, and the number of CRPs is n. Since only one of the rows is being evaluated at a time, a one-hot decoder is used so that only one bit of the EVA vector is logic 1. The challenge fed into the decoder is a $log(n)$-bit input, and the response is a m-bit output.

Compared with existing weak SRAM PUFs, the weak LEDPUF has several evident advantages:

- It is completely stable, so it has no area or latency overhead. To generate a bit response, the weak LEDPUF requires only one SSU and a transistor, or 3 transistors equivalently, as for a standard SRAM cell, 6 transistors are required. Once the state of the DSA via is determined, the output is fixed permanently, so no additional ECC, fuzzy extraction, or helper data is needed. As stated in [35], for a SRAM PUF to generate a 128-bit response, more than 4k SRAM cells are needed under a condition with 15% bit error probability. Therefore, the total number of transistors needed for SRAM PUF to generate

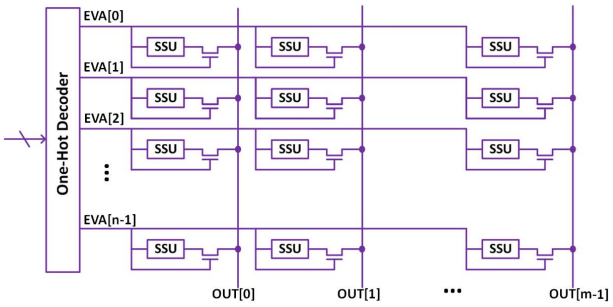

Fig. 5. A weak LEDPUF with n challenges and m-bit response. Only one bit of the EVA vector is logic 1 at a time.

a 128-bit response would be 24k, where for the weak LEDPUF, only 384 transistors are needed, which is more than 600x less than a SRAM PUF, thus the area is also much smaller even assuming that the hardware cost of the peripheral circuits are similar.

- In addition to model building attacks [36], the weak LEDPUF is also more resistant to existing attacks to SRAM PUFs, such as laser stimulation [37] or Photonic Emission Analysis (PEA) [38]. The laser stimulation attack focuses on retrieving the on/off state of transistors, but for weak LEDPUF, the states of the transistors, which depend on the EVA signal, do not reveal secret information. The PEA attack does not work effectively because for each SSU, the source voltage (VDD) of the NMOS is always higher than the drain voltage, and the PMOS at the output will not stay in saturation region since the output will be pulled down even if the DSA connection is formed.

When using a weak PUF for a CRP authentication scheme, it is meaningful to consider the chance of guessing the response. The min-entropy [39]

$$H_{\min}(X) = -\log_2\left(\max_i p_i\right) \tag{1}$$

is a means of quantifying the chance of guessing the response in a single round. It corresponds to the exponent of the probability of the most probable response, assuming that each element is identically and independently distributed. Based on our own experimental results for the formation of connections we evaluate the probability mass function of a bit generated by a LEDPUF

$$p_X(1) = 0.4626 \quad p_X(0) = 0.5374. \tag{2}$$

The min-entropy of the empirical probability mass function is

$$H_{\min}(X) = 0.8962 \tag{3}$$

whereas the maximal min-entropy which is achieved by a fair coin toss, equals to 1. Hence, when the response consists of m bits, the probability of guessing the response of the LEDPUF is equal to $2^{-0.8962m}$. Essentially, it means that when the response of a LEDPUF is $1.11m$ bits long, the probability of guessing the response is equal to 2^{-m} which is the probability of guessing the result of m independent fair coin tosses.

Another possible attack is a dictionary attack in which the attacker guesses the most probable responses in an ascending order. The number of attempts it takes to find the response is coined *guesswork* [40] which we denote by G. For a stream of m bits which are drawn i.i.d the expected guesswork is lower bounded by [41]

$$E\{G\} \geq \frac{1}{4}2^{mH_{Sh}(x)} \tag{4}$$

where $H_{Sh}(x) = \sum_i -P_i \log_2(p_i)$ is the Shannon entropy. Assigning (2) to the Shannon entropy we get that

$$E\{G\} \geq \frac{1}{4}2^{0.996m}. \tag{5}$$

Further, the exponential growth rate of the expected guesswork (as a function of m) scales according to the Renyi entropy [40] with parameter $\alpha = \frac{1}{2}$

$$\lim_{m\to\infty}\frac{1}{m}\log_2 E\{G\} = H_{1/2}(X) = 2\cdot\log_2\left(\sum_i p_i^{1/2}\right). \tag{6}$$

Assigning the empirical probability (2) to (6), gives a growth rate that equals to 0.998, whereas the maximal growth rate which is achieved by a fair coin toss, is again equal to 1. Therefore, when the response is $1.002m$ bits long, the guesswork scaling behavior is equal to the scaling behavior of m independent fair coin tosses.

Interestingly, the loss induced by the fact that the bits are not drawn uniformly, is much higher when considering only a single guess. On the other hand, when considering multiple guesses, the loss decreases significantly. It is also worth mentioning that even though the loss for a single guess is not negligible, for large m the chance of guessing the response is still very low.

Based on the empirical results (2) we deduce that the bits are drawn according to a biased probability function. However, the probability mass function can always be adjusted by changing slightly the size of the guiding template. Another possibility to balance out the probability, is by using a randomness extractor [42], which outputs a shorter stream of bits that correspond to independent fair coin tosses.

B. Strong LEDPUF Construction

One of the shortcomings of using memory-based PUFs for CRPs, is the scaling of the hardware size as a function of the number of CRPs [43]. In general, each channel response pair requires a different set of circuits, and as a result the hardware size is proportional to the number of CRPs. On the other hand for strong PUFs the hardware size scales logarithmically as a function of the number of CRPs.

In order to create a strong LEDPUF we consider a keyed hash function along with a weak LEDPUF. The weak LEDPUF response is used as a key for the keyed hash function. The challenges serve as the input to the hash function, whereas the response is the output of the keyed hash function. Figure 6 presents a strong LEFPUF based on a keyed hash function and on a weak LEDPUF.

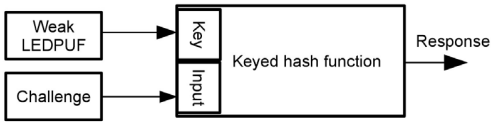

Fig. 6. A strong LEDPUF based on a keyed hash function (HMAC or NMAC) and a weak LEDPUF.

It is important that the keyed hash function uses the key in such a way that does not enable the attacker to predict responses to unobserved challenges based on the observed ones. Therefore, concatenating the key directly to the challenge, which is vulnerable to extension or collision attacks, is not a good realization of the strong LEDPUF.

We create strong LEDPUF by using a weak LEDPUF as a key for a keyed-hash message authentication code (HMAC) [44]. Any cryptographic hash function, such as SHA-1 or SHA-2 can be used for HMAC. It is worth mentioning that in [45] the authors also propose the use of PUFs with an HMAC in a somewhat similar manner; however, they do not take into consideration the overhead incurred by the instability of parametric PUFs.

To give a rough estimation of the hardware implementation cost of the strong LEDPUF, for a HMAC-SHA1, the implementation requires about 30k gates [46], and just the ECC part, BCH for example, of a parametric PUF would require same order of gates [17].

The level of security of a strong LEDPUF depends on the underlying hash function and the quality of the weak LEDPUF that serves as a key, whereas weak LEDPUFs rely solely on the randomness in the manufacturing process.

IV. Experimental Results

A. Uniqueness Evaluation

For the weak LEDPUF, the uniqueness is evaluated by calculating the fractional inter-distance [6] of 1000 weak LEDPUFs, each produces 512 bits of response. The distribution is with mean=0.503 and standard deviation=0.02 as shown in the second row of Table I. Since the variance value is proportional to the inverse of the length of the response, as the length of the response increases the variance value goes to zero while the mean value goes to 0.505.

For the strong LEDPUF, the structure used in our experiment is based on the NMAC structure, and the results are obtained from simulations. Each strong LEDPUF consists of a weak LEDPUF that provides 2x256 bits for the two initial vectors (IV) of the nested hash, and each response is a 256-bit stream because SHA-256 is used in the construction. The same challenge is given to 1000 strong LEDPUFs, and the inter-distance of the responses is a distribution with mean=0.500 and standard deviation=0.03 as shown in the third row of Table I.

B. Stability-Guaranteed Weak LEDPUF Requirement

To construct a strong LEDPUF, only the weak LEDPUF can be used because of its 0% intra-distance. If other existing weak PUFs with even small intra-distance are used, the intra-distance of the strong LEDPUF would be increased dramatically due to the avalanche effect. In other words, even a single bit flip of the weak PUF can completely change the response of the strong LEDPUF. Figure 7 (a) shows that the intra-distance of the strong LEDPUF jumps from 0% to 50% as the number of bit flips increases from zero to one.

Figure 7 (b) shows how the intra-distance of the strong LEDPUF rises as the intra-distance of the weak PUF increases in logarithmic scale. Since 2x256 bits of the IVs are from the weak PUF, for a weak PUF with 0.1% intra-distance, the probability that it generates a same 512-bit response twice is about 60%, which translates to a roughly 20% intra-distance of the strong LEDPUF. Therefore, only the weak LEDPUF with a guaranteed 0% intra-distance can be used for the IV generation.

TABLE I
Fractional Inter-distance of the LEDPUF

	Response Bits	Mean	Standard Deviation
Weak LEDPUF	512	0.503	0.02
Strong LEDPUF	256	0.500	0.03

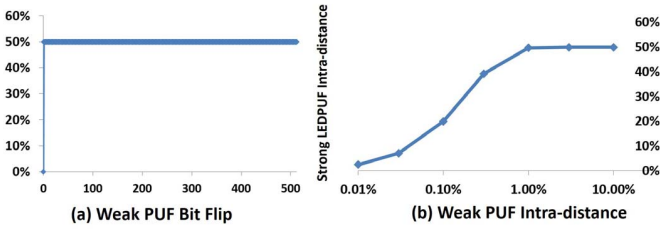

Fig. 7. (a) A single bit flip from the weak PUF can induce a completely different response of the strong LEDPUF due to the avalanche effect of the hash function. (b) Intra-distance of the strong LEDPUF rises dramatically if other weak PUFs with small intra-distances are used in the strong LEDPUF construction.

V. Conclusion

In this paper we propose the first stability-guaranteed PUF that requires no stability enhancement techniques, where the source of randomness is extracted from locally enhanced DSA process. Detailed constructions of two LEDPUFs: the weak LEDPUF and the strong LEDPUF, are presented. Inter-distance measurements on the LEDPUFs show that both weak and strong LEDPUFs are ideally unique. The area and latency of the weak LEDPUF is much smaller than existing weak PUFs because no error correcting schemes are needed. The strong LEDPUF provides large CRP space because of its cryptographic hash based structure. The weak LEDPUF used in the strong LEDPUF construction cannot be replaced by existing weak PUFs because an absolute 0% intra-distance is required for the weak PUF to avoid the avalanche effect of the strong LEDPUF. Furthermore, we quantify the level of security provided by weak LEDPUF by calculating the expected guesswork resulting from weak LEDPUFs empirical probability function; the loss compared to a fair coin toss is negligible.

Our ongoing work includes exploring other sources of LEDPUF that are more resilient to invasive attacks, such as Scan Electron Microscopy (SEM) or Transmission Electron Microscopy (TEM).

ACKNOWLEDGEMENT

The authors would like to thank Dr. Andres Torres for his support of the DSA simulation and valuable inputs.

REFERENCES

[1] Roel Maes and Ingrid Verbauwhede. Physically Unclonable Functions: A Study on the State of the Art and Future Research Directions. In *Towards Hardware-Intrinsic Security*, pages 3–37. Springer Berlin Heidelberg, 2010.

[2] Blaise Gassend et al. Silicon physical random functions. In *Proc. CCSC*, 2002.

[3] G.E. Suh and S. Devadas. Physical Unclonable Functions for Device Authentication and Secret Key Generation. In *Proc. DAC*, 2007.

[4] Chi-En Yin, Gang Qu, and Qiang Zhou. Design and implementation of a group-based RO PUF. In *Proc. DATE*, 2013.

[5] 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*, 2009.

[6] A. Maiti and P. Schaumont. Improving the quality of a Physical Unclonable Function using configurable Ring Oscillators. In *International Conference on FPL*, 2009.

[7] Lerong Cheng et al. Physically Justifiable Die-Level Modeling of Spatial Variation in View of Systematic Across Wafer Variability. *IEEE TCAD*, 2011.

[8] Lang Lin, S. Srivathsa, D.K. Krishnappa, P. Shabadi, and W. Burleson. Design and Validation of Arbiter-Based PUFs for Sub-45-nm Low-Power Security Applications. *IEEE TIFS*, 2012.

[9] D. Ganta and L. Nazhandali. Study of IC Aging on Ring Oscillator Physical Unclonable Functions. In *IEEE ISQED*, 2014.

[10] Wenjie Che, Jim Plusquellic, and Swarup Bhunia. A Non-Volatile Memory Based Physically Unclonable Function without Helper Data. In *Proc. ICCAD*, 2014.

[11] Chi-En Yin and Gang Qu. Improving PUF security with regression-based distiller. In *Proc. DAC*, 2013.

[12] Linus Feiten, Tobias Martin, Matthias Sauer, and Bernd Becker. Improving RO-PUF Quality on FPGAs by Incorporating Design-Dependent Frequency Biases. In *IEEE ETS*, 2015.

[13] M. Majzoobi, F. Koushanfar, and S. Devadas. FPGA PUF using programmable delay lines. In *IEEE International Workshop on WIFS*, 2010.

[14] Teng Xu and M. Potkonjak. Robust and flexible FPGA-based digital PUF. In *International Conference on FPL*, 2014.

[15] M.T. Rahman, D. Forte, J. Fahrny, and M. Tehranipoor. ARO-PUF: An aging-resistant ring oscillator PUF design. In *Proc. DATE*, 2014.

[16] Jeroen Delvaux and Ingrid Verbauwhede. Key-recovery attacks on various RO PUF constructions via helper data manipulation. In *Proc. DATE*, 2014.

[17] Xinmiao Zhang. VLSI Architectures for Modern Error-Correcting Codes. 2015.

[18] International Technology Roadmap for Semiconductors. http://public.itrs.net/.

[19] James B. Wendt and Miodrag Potkonjak. Hardware obfuscation using PUF-based logic. In *Proc. ICCAD*, 2014.

[20] Teng Xu, James B. Wendt, and Miodrag Potkonjak. Secure Remote Sensing and Communication Using Digital PUFs. In *Proc. ANCS*, 2014.

[21] Bing Xu et al. Self-assembly of liquid crystal block copolymer peg-b-smectic polymer in pure state and in dilute aqueous solution. *Faraday discussions*, 2009.

[22] Nathan Jarnagin. *High X Block Copolymers For Sub 20 Nm Pitch Patterning: Synthesis, Solvent Annealing, Directed Self Assembly, And Selective Block Removal*. PhD thesis, Geogia Institute of Technology, 2013.

[23] Myungwoong Kim et al. Interplay of surface chemical composition and film thickness on graphoepitaxial assembly of asymmetric block copolymers. *Soft Matter*, 2013.

[24] Yasmine Badr, Andres J Torres, and Puneet Gupta. Mask assignment and synthesis of DSA-MP hybrid lithography for sub-7nm contacts/vias. In *Proc. DAC*, 2015.

[25] Ho-Cheol Kim, Sang-Min Park, and William D Hinsberg. Block copolymer based nanostructures: materials, processes, and applications to electronics. *Chemical reviews*, 2009.

[26] Hari Pathangi et al. Defect mitigation and root cause studies in IMEC's 14 nm half-pitch chemo-epitaxy DSA flow. *Proc. SPIE*, 2015.

[27] Deepak Sundrani, SB Darling, and SJ Sibener. Hierarchical assembly and compliance of aligned nanoscale polymer cylinders in confinement. *Langmuir*, 2004.

[28] Germain Fenger et al. Compact model experimental validation for grapho-epitaxy hole processes and its impact in mask making tolerances, 2014.

[29] Charles T. Black. Polymer self-assembly as a novel extension to optical lithography. *ACS Nano*, 2007.

[30] Huiman Kang et al. Degree of perfection and pattern uniformity in the directed assembly of cylinder-forming block copolymer on chemically patterned surfaces. *Macromolecules*, 2011.

[31] Kenji Yoshimotoa and Takashi Taniguchi. Large-Scale Dynamics of Directed Self-Assembly Defects on Chemically Pre-Patterned Surface. *Proc. SPIE*, 2013.

[32] Marcus Mller et al. Kinetics of directed self-assembly of block copolymers on chemically patterned substrates. *Journal of Physics*, 2015.

[33] Brandon L. Peters et al. Graphoepitaxial Assembly of Cylinder-Forming Block Compolymers in Cylindrical Holes. In *Journal of Polymer Science*, 2014.

[34] F. Detcheverry et al. Monte Carlo simulations of a coarse grain model for block copolymers and nanocomposites. In *Macromolecules*, 2008.

[35] J. Guajardo, G.-J. Schrijen S. S. Kumar, and P. Tuyls. FPGA Intrinsic PUFs and Their Use for IP Protections. In *CHES*, 2007.

[36] C. Herder et al. Physical Unclonable Functions and Applications: A Tutorial. *Proc. of the IEEE*, 2014.

[37] D. Nedospasov, J.-P. Seifert, C. Helfmeier, and C. Boit. Invasive PUF Analysis. 2013.

[38] C. Helfmeier et al. Cloning Physically Unclonable Functions. In *IEEE International Symposium on HOST*, 2013.

[39] Y. Dodis, A. Reyzin, and A. Smith. Fuzzy extractor, A brief survey of results from 2004 to 2006. In *Security with Noisy Data*. Springer Berlin Heidelberg, 2007.

[40] E. Arikan. An inequality on guessing and its application to sequential decoding. *IEEE Tran. on Inf. Th.*, 42, 1996.

[41] J.L. Massey. Guessing and entropy. In *ISIT 1994*, 1994.

[42] J.V. Neumann. Various techniques used in connection with random digits. *Applied Math Series*, 1951.

[43] R. Maes and I. Verbauwhede. Physically Unclonable Functions: a Study on the State of the Art and Future Research Directions. *Towards Hardware-Intrinsic Security*, 2010.

[44] M. Bellare, R. Canetti, and H. Krawczyk. Keyed Hash Functions and Message Authentication. In *Crypto*, 1996.

[45] S.W. Jung and S. Jung. HRP: A HMAC-based RFID mutual authentication protocol using PUF. In *ICOIN 2013*.

[46] Mao-Yin Wang et al. An HMAC processor with integrated SHA-1 and MD5 algorithms. In *Proc. ASP-DAC*, 2004.