LRR-DPUF: Learning Resilient and Reliable Digital Physical Unclonable Function *

Jin Miao‡, Meng Li♯, Subhendu Roy‡, and Bei Yu†

‡Cadence Design Systems, San Jose, CA, USA
♯ECE Department, University of Texas at Austin, Austin, TX, USA
†CSE Department, The Chinese University of Hong Kong, NT, Hong Kong
{jmiao,subhroy}@cadence.com, meng_li@utexas.edu, byu@cse.cuhk.edu.hk

ABSTRACT

Conventional silicon physical unclonable function (PUF) extracts fingerprints from transistor's analog attributes, which are vulnerable to environmental and operational variations. Recently, digitalized PUF prototypes have emerged to overcome the vulnerability issues, however, the existing prototypes are either hybrid of analog-digital PUFs which are still under the shadow of vulnerability, or impractical for real-world implementation. To address the above limitations, we propose a learning resilient and reliable digital PUF (LRR-DPUF). The fingerprints are extracted from VLSI interconnect geometrical randomness induced by lithography variations. Crucially, we use strongly skewed latches to ensure the immunity against environmental and operational variations. Further, a cross-coupled, highly non-linear logic network is proposed to effectively spread and augment even subtle interconnect randomness, as well as to achieve strong resilience to machine learning attacks.

We demonstrate that a 64-bit LRR-DPUF exhibits close to ideal statistical performances, including 0 intra Hamming Distance. We also mathematically prove that each output of the LRR-DPUF follows uniform distribution. Various state-of-the-art machine learning models show almost no better than random prediction accuracies when applied to LRR-DPUF.

1. INTRODUCTION

The pervasive embedded computing devices require reliable and low cost authentication mechanisms to ensure safety of sensitive information and life-critical actions. An increasing demand on high performance security solutions can be foreseen in the era of Internet of Things (IoT). Silicon Physical Unclonable Function (PUF) is an innovative low cost hardware security primitive [1–5] that derives authentication fingerprints from integrated circuits manufacturing process variations.

PUFs are classified into strong PUF and weak PUF, depending on the number of unique challenge-response pairs (CRPs) the PUF can produce. A strong PUF has sufficiently large CRP space, hence it is impossible to enumerate or predict any CRP within a limited time-frame. By contrast, a weak PUF is often used for secure key generation due to the limited CRP space. We discuss strong PUF in this paper.

A number of PUF architectures have been proposed in literature, and today the most popular PUFs are often modeled with memory [6–8], delay [9–12], etc. All those PUF architectures utilize **transistors**' intrinsic randomness under process variations, which are by nature analog attributes. Just as any analog circuit systems, analog PUFs are commonly vulnerable to environmental and operational variations. Therefore, analog PUF architectures are often equipped with a fuzzy extractor [13] or an error correction system [14]. Those auxiliary circuits often require hardware cost and power consumptions several order magnitude higher than the PUF circuit itself. In addition, some analog PUF models, e.g., delay based PUF, have been demonstrated insecure from side channel and machine learning attacks [15, 16].

There is a strong demand to derive digital PUFs in order to tackle the reliability limitations in analog PUFs as well as to provide low latency, high throughput PUF solutions. In the recent years, there have been some literatures proposing prototype digital PUFs [17, 18]. Xu et al. [17] proposed an FPGA based digital PUF that combines FPGA fabric with a standard analog PUF. The analog PUF is used at initialization stage and can be discarded afterwards. However, this is not a completely digitalized PUF, hence it faces the same limitations as analog PUF does, particularly in the initialization stage. Later in [18], Xu et al. proposed the first conceptual level fully digitalized PUF by using defective digital IC chips. It is based on the observation that a small circuit fault can drastically impact the overall functionality of a digital logic, hence such circuit faults, which were modeled as stuck-at-faults and bridge-faults, can be used as the fingerprint for digital PUF. However, those fault models are too simplified to catch the physical impacts induced by the fault. Depending on specific contexts, some circuit faults may lead to serious physical and logical chain-effect, and ultimately break down the entire circuit system. This contradicts to the original purpose of using digital PUFs. For example, the Wired-AND model in CMOS circuits may result in a direct current path from supply voltage to ground, and put the CMOS gate to an uncertain operating region. This may cause large power waste and even unstable outputs in CMOS gates. Therefore, it is necessary to reconsider the feasibility by **direct** use of a defective IC chip as digital PUF. Nevertheless, the concept of utilizing faulty circuits is still valuable and inspires our digital PUF development.

In addition, one critical question is yet to be addressed– what would be the optimal logical circuit to use for the maximal security? In [18], an array multiplier was taken to demonstrate the defective IC PUF concept, however, neither linearity anal-

*This work is supported in part by CUHK Direct Grant for Research.

ICCAD '16, November 07-10, 2016, Austin, TX, USA

 ISBN 978-1-4503-4466-1/16/11…$15.00

DOI: http://dx.doi.org/10.1145/2966986.2967051

ysis nor learning based reverse engineering was conducted. In fact, unlike analog PUFs where the non-linearity is derived by transistor natures, many digital logic circuits intrinsically can be **linearly-separable** [19], and relying on arbitrary logic circuits may lead to insecure PUFs vulnerable to even linear model machine learning attacks. Therefore, a highly non-linear logic architecture is desired to realize a secure digital PUF.

In this paper, we propose a novel learning resilient and reliable digital PUF (LRR-DPUF) that resolves the reliability drawbacks in analog PUFs as well as the practicality and security issues in existing digital PUFs. Our LRR-DPUF takes advantages of **Boolean** type interconnect randomness induced by lithography variations, and crucially, the digitalization is realized by using strongly skewed latch to ensure the Boolean status for all internal signals. The interconnect randomness is ultimately spread and cross-coupled by a novel highly non-linear logic network architecture. The proposed LRR-DPUF shows close to ideal statistical performance and strong resilience to machine learning attacks.

Our major contributions are summarized as follows:

- Quantitatively justify the feasibility of utilizing the interconnect randomness induced by lithography variations;
- Propose to use strongly skewed latches to ensure a complete and reliable digital PUF architecture;
- Propose a novel highly non-linear logic network architecture that can effectively spread and augment any interconnect randomness, as well as achieve strong resilience to machine learning attacks.

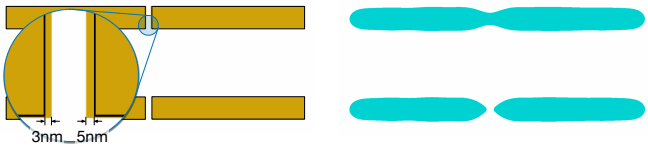
The rest of the paper is organized as follows: in Section 2, we discuss the source of Boolean type randomness during lithography process. In Section 3, we propose our solution to make the interconnect randomness compatible to CMOS technology. In Section 4, we propose our LRR-DPUF architecture. In Section 5, we analyze the properties of the LRR-DPUF. In Section 6, we show statistical evaluations and learning resilience. In Section 7, we further discuss additional issues on LRR-DPUF, followed by the conclusion in Section 8.

2. BOOLEAN RANDOMNESS BY LITHOGRAPHY

Identifying a feasible Boolean randomness source is half the battle to make a digital PUF. Conventional analog PUFs rely on transistor's intrinsic randomness, including delay, current, resistance, capacitance, etc. Xu et al. [18] first proposed to take advantage of the randomness from VLSI interconnect, namely the metal wires. Such interconnect randomness, by itself, is a Boolean type variable, i.e., in either connected or disconnected status[1]. The feasibility of utilizing the interconnect randomness, however, was not justified. It was also unclear whether or how such randomness can be controlled during design and manufacturing stages. In this section, we will analyze how process variations affect the VLSI interconnect, and will quantitatively demonstrate the feasibility of utilizing such randomness.

As VLSI technology node scales down to nanometer regime, one of the major interconnect geometrical variations comes from lithography. The lithography variations can be categorized into "systematic" and "local" variations[2]. The systematic variation, including dose (light density) and focus variations in lithography system, refers to a systematic offset applied to a group of adjacent layout patterns, and is often considered as inter die

[1]We will discuss further on high-resistance case in Section 7.
[2]Local variations are also referred as random variations.



(a) (b)

Figure 1: Interconnect under lithography variations. (a) Mask stripe-pairs with split distance of $20nm$ (top) and $28nm$ (bottom). The top mask is zoomed in. (b) Lithography simulation outputs (shapes on wafer).

variation. The local variation, by contrast, including mask errors and line edge roughness (LER), refers to localized or intra die randomness for each individual layout pattern. Considering typical PUF circuits to be small in size, the local variation is more of importance to generate unique fingerprints and should dominate the systematic variations.

There have been literatures utilizing lithography variations for PUF designs [20–22]. Kumar et al. [20] proposed to utilize local pitch variations for PUF design. Rather than Boolean type randomness, the lithography variations were used to generate transistor analog fingerprints, leading to an analog PUF. Both [21] and [22] claimed to use focus and dose variations as a ubiquitous approach applied to general PUF designs. As aforementioned, dose and focus are systematic variations mostly impacting inter-die variations, the overall performance improvements can be limited.

In this paper, we utilize the interconnect randomness from the very local **mask variation** when producing a photo mask by an electron beam lithography system (the conventional mask manufacturing tool). Mask error enhancement factor (MEEF) [23] is used to quantify how mask variations will be reflected in the final wafer. In advanced technology nodes, even with gridded design rules (GDR) and resolution enhancement techniques (RET), the MEEF value on line-end can be up to $10\times$ [24], which means a $5nm$ variation in mask line-end pattern would cause $50nm$ change in the final wafer pattern. In practice, *an electron beam system can easily lead to mask variations well exceed this 5nm variation threshold* [25].

In this paper, the interconnect randomness is realized by intentionally positioning two interconnect layout line-ends close to each other, and due to mask variations, the generated masks will have mismatches. Such mismatch is further magnified by MEEF factor on the wafer, and ultimately leads to uncertain connectivity status. Figure 1(a) shows two mask stripe-pairs split by a small distance. The bottom stripe-pair is with the original split distance of $28nm$, while the top stripe-pair is with distance of $20nm$ due to mask variations. Here we assume $3nm$ and $5nm$ mismatches for the two line-ends, respectively. We use an industry lithography simulator [26] to simulate these two mask stripe-pairs, and Figure 1(b) shows the final output images. The $8nm$ difference of the split distance in the two stripe-pairs is now converted to two Boolean connectivity statuses: the top stripe-pair is merged and connected, while the bottom remains disconnected.

We now justify how the overall connectivity statistics are impacted by the layout split distance as well as local and systematic lithography variations. Before that, we introduce the concept of **connectivity rate** as the number of eventually connected stripe-pairs over the number of total stripe-pairs. Note that the split distance can be controlled by circuit engineers when designing VLSI layout, and the local variation, specifically the mask error, can be modeled by centered Gaussian distribu-

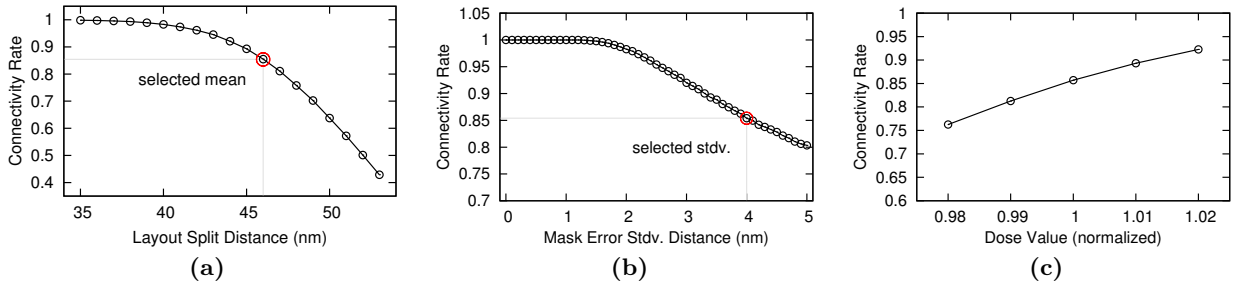


Figure 2: Interconnect connectivity rate under lithography variations: (a) Layout split distance under mask error stdv. of $4nm$; (b) Mask error stdv. under split distance of $46nm$; (c) Dose values.

tion[3] [23, 27], where the standard deviation (stdv) is depending on the accuracy and settings of the electron beam system. Both factors can be configured in today's VLSI manufacturing setup. By contrast, the dose variation is a systematic offset applied to specific wafer zones which should be minimized. Such offsets could shift the connectivity status towards a single direction on certain wafer zones, hence degrade the level of interconnect randomness and further the PUF performance. Therefore, the central task in the rest of this section is to justify the feasibility of minimizing the impact from systematic variations by carefully configuring the split distance and mask error.

We first evaluate the mask variations under various split distances ranging from $35nm$ to $53nm$ with $1nm$ step. For each split distance, we further sweep the mask error stdv ranging from $0nm$ to $5nm$, leading to various mask stripe-pair sets, each with size of 10K. These variously configured 10K stripe-pair sets are later fed into the lithography simulator [26] to get the ultimate stripe shapes on wafer. We then measure the connectivity rate of each 10K stripe set. For example, Figure 2(a) shows when the mask error stdv. is $4nm$, by changing the layout split distance, connectivity rate can vary from about 0.4 to 0.99. And Figure 2(b) shows when the layout split distance is $46nm$, connectivity rate changes from 1.0 to 0.8 when the mask error varies from $0nm$ to $5nm$.

Further, we evaluate the impact from dose variation that would cause potential systematic offsets to the split distance [4]. In Figure 2(c), we sweep the normalized dose value from 0.98 to 1.02 (the maximum available range) with split distance of $46nm$ and mask error stdv. of $4nm$, where the $46nm$ and $4nm$ are the selected configurations that can minimize the dose impact. Clearly, the connectivity rate retains in a high value between 0.75 and 0.93. We show in later section that, higher connectivity rate (but less than 1) generally leads to better security performance in our proposed digital PUF architecture. Therefore, by carefully configuring the layout split distance and electron beam system accuracy, it is feasible to minimize the impact from the systematic offset like dose.

Overall, we have justified the feasibility of using interconnect geometrical variation for PUF design. However, such interconnect randomness cannot be directly used in the VLSI circuit systems, especially for CMOS circuits, due to serious physical incompatibilities. In next section, we will propose our solution to make such randomness compatible to digital VLSI systems.

3. MAKING IT CMOS COMPATIBLE

[3]We show in Section 5 that, the performance of LRR-DPUF relies on the cumulative connectivity rate regardless of any specific distribution pattern.

[4]In this paper, we ignore the systematic variations from focus, as it can be dealt with in a similar manner as dose [22].

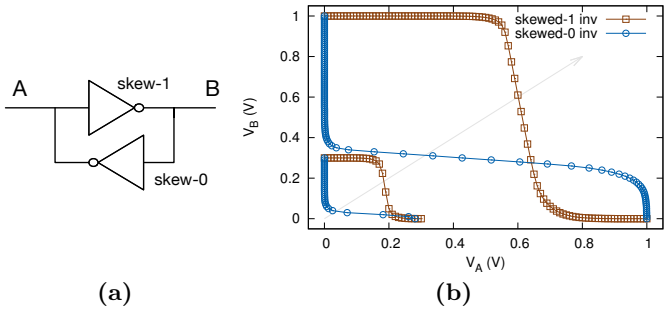


Figure 3: Handling dangled poly-gate by strongly skewed latch. (a) Inverter pair skewed latch structure; (b) The VTC relation of a strongly skewed latch.

In CMOS circuits, any **unexpected** open-circuit and short-circuit may lead to serious circuit failures. Such failures are not only in logical perspective, but can also adversely affect the physical reliability and power efficiency. Particularly, for the case of open-circuit failure, the transistor poly-gate is dangled with floating voltage level. This makes the transistor vulnerable to environmental noise which may even change the transistor operating region. For the case of short-circuit failure, where two CMOS gates' drains are connected, there will be a good chance to create a direct current path from supply voltage to ground, resulting huge power waste as well as unknown transistor operating region. Apparently, both scenarios are opposite to the original wish of digital PUF. In this section, we propose our solution to completely eliminate the aforementioned issues and make the interconnect randomness compatible to CMOS circuits.

The goal is to identify a pure logical structure so that it can work with both open-circuit failures as well as normally connected circuits[5]. A CMOS latch, by connecting two inverters head-to-tail, can be a good candidate. A latch is supposed to either remain a pre-existing state until a new input is applied, or preset an initial state and later never change it. The first feature ensures the compatibility to normal circuit operations, and the latter one makes it possible to work with open-circuit failures. For open-circuit failures, the input to latch is dangled, and the initial state will be automatically set during the power up state [6]. However, for a regular symmetric latch, the initial power up state may be affected by static noises, hence can be inconsistent from time to time. To completely eliminate such uncertainties induced by static noise, in Figure 3(a), we propose to use a strongly skewed-1 inverter that head-to-tail connects

[5]The short-circuit failure case will be constructively avoided in our LRR-DPUF architecture.

to a strongly skewed-0 inverter. The skewed-0 inverter can further strengthen the skewed-1 inverter, hence together form a strongly skewed-1 latch. Note that, without any loss of generality, we only discuss skewed-1 latch in this paper. Here, the skewed-1 inverter is realized by specifying i) PMOS width several times wider than its NMOS counterpart, and ii) lower voltage threshold (VT) for PMOS and higher VT for NMOS. The skewed-0 inverter can be derived by the opposite configuration. In Figure 3(a), if pin A gets disconnected, minor static noise cannot change the power up state of this skewed latch, and the latch will favor more to stay at logic 1. Note that after power up phase, the latch will remain in the logic value until supply voltage is removed. In addition, the skewed-1 inverter has to be designed with larger size than the skewed-0 inverter for the case that when the latch is normally connected to the network without open-circuit failure, the skewed-1 inverter should dominate the skewed-0 inverter. In that case, the skewed latch is reduced to a regular inverter.

In Figure 3(b) we show the HSPICE simulations with PTM-45nm model for voltage-transfer curves (VTC). The skewed-1 inverter has 10× wider width on PMOS than its NMOS counterpart, and the skewed-0 inverter has 4× wider width on NMOS than its PMOS. Besides, the PMOS in skewed-1 inverter uses low VT PMOS and high VT NMOS transistors, and vice versa for skewed-0 inverter. It can be observed that in the power up voltage region, i.e., less than 0.3V, the noise margin is maximized to favor logic 1, hence practically guarantees a deterministic power up state. When voltage increases to 1.0V, the latch will remain in logic 1 unless pin-A is applied by a new input.

Note that in memory based PUF literatures, latch skewness was also discussed and used as the power-up fingerprint [6, 7]. The major difference against memory PUF is that, in memory circuits, all latches are designed to be symmetric, and the skewness comes from the **intrinsic** process variation and is used as the **source** of fingerprint. However, such intrinsic variation does not guarantee all latches to be skewed in memory PUFs, which is the very root cause of the reliability drawbacks for such type of PUFs. By contrast, we intentionally skew **all** latches in order to completely eliminate any possible environmental vulnerability as well as to make the interconnect randomness compatible to CMOS systems. We will further demonstrate by HSPICE in Section 6 that, such strongly skewed latches retain consistent power-up state across very wide temperature and voltage ranges. With above preparation, in next section, we introduce our learning resilient digital PUF architecture.

4. THE LRR-DPUF ARCHITECTURE

To this point, we have converted the PUF design to be a pure logic design problem. Our focus now is to identify a non-linear logic network that can maximize and spread any subtle interconnect randomness, and ultimately realize a highly secure digital PUF. We propose to derive a logic network constructed by **regularly repeated** nodes, and we call each node a **unit cell**. In the following, we will first discuss the design of unit cells, and then propose the overall logic network topology. In Section 5 , we will further analyze in details the properties of the proposed LRR-DPUF.

4.1 Unit Cell

In cryptography, Exclusive-OR (XOR) logic is the most popular function due to the simplicity in realization and perfect security nature. Due to the linearly non-separable attribute, an XOR logic outstands other logics like AND, OR, etc., offering intrinsic resilience to many learning based attacks. As shown in Figure 4, to separate the output of a 2-input XOR logic, at least 2 lines are required, and 3 planes are required for a 3-input XOR logic. Another intriguing feature of XOR logic is the uniform output distribution. In Lemma 1, we show that for a 2-input XOR, as long as **one** of the inputs is uniformly distributed, the XOR output is ensured uniformly distributed. This property will be further discussed in Section 5 once the entire LRR-DPUF topology is introduced. Therefore, XOR logic can be a perfect candidate for our unit cell design.

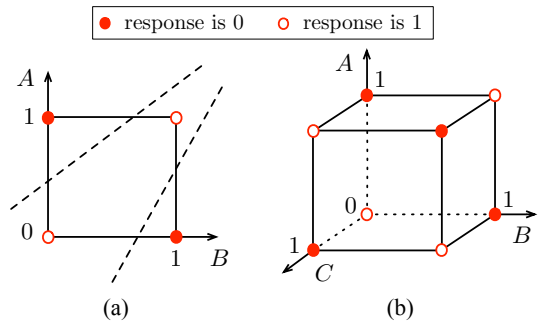


Figure 4: Linear non-separable nature for XOR logic. (a) The 2-input XOR logic $Y = A \oplus B$ requires at least 2 lines to separate the 1 and 0 dots; (b) The 3-input XOR logic $Y = A \oplus B \oplus C$ requires at least 3 planes for separation.

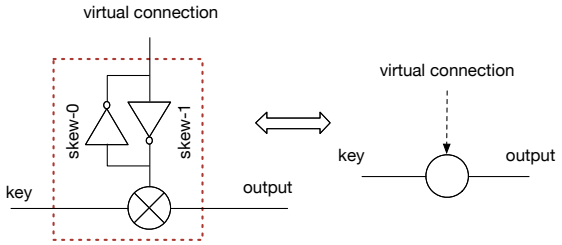


Figure 5: Unit cell: left figure shows the complete logic structure; right figure is the simplified symbol.

Lemma 1. $\Pr[y = 1] = \Pr[y = 0] = 0.5$ *holds, as long as* $\Pr[a = 1] = \Pr[a = 0] = 0.5, \ \forall b \in B$,
where a and b are the two inputs of a 2-input XOR gate, and y is the output. The symbol $\Pr[y = 1]$ *refers to the probability of output y being logic 1, and vice versa.*

PROOF. For a 2-input XOR gate, the probability of output being logic 1 can be written as $\Pr[y = 1] = \Pr[a = 1] \times \Pr[b = 0] + \Pr[a = 0] \times \Pr[b = 1]$. Consider $\Pr[a = 1] = \Pr[a = 0] = 0.5$, we have $\Pr[y = 1] = 0.5 \times \Pr[b = 0] + 0.5 \times \Pr[b = 1] = 0.5 \times (\Pr[b = 0] + \Pr[b = 1]) = 0.5$. Hence $\Pr[y = 0] = 1 - \Pr[y = 1] = 0.5$. □

Figure 5 shows the unit cell structure. It is a 2-input 1-output logic block, constructed by a 2-input XOR gate with one of its inputs connected to the strongly skewed-1 latch. The **key** pin is the actual information bit that passes through this XOR gate. The **virtual connection** pin is the source of the randomness. It may or may not connect to the logic network depending on the interconnect randomness status. If this virtual connection pin is connected to a stable logic value, the **output** of the entire unit cell has logic expression of $key \otimes \overline{virtual\ connection}$. If it is dangled, as discussed in Section 3, since the skewed-1 latch stays in logic 1 state after power up, the output of the unit cell equals to $\overline{key}$. In general, the unit cell can be viewed as a random "bit-flip" block, where a 1-bit information (key pin) may or may not get inverted depending on the interconnect randomness. Apparently, in any case, the unit cell output is

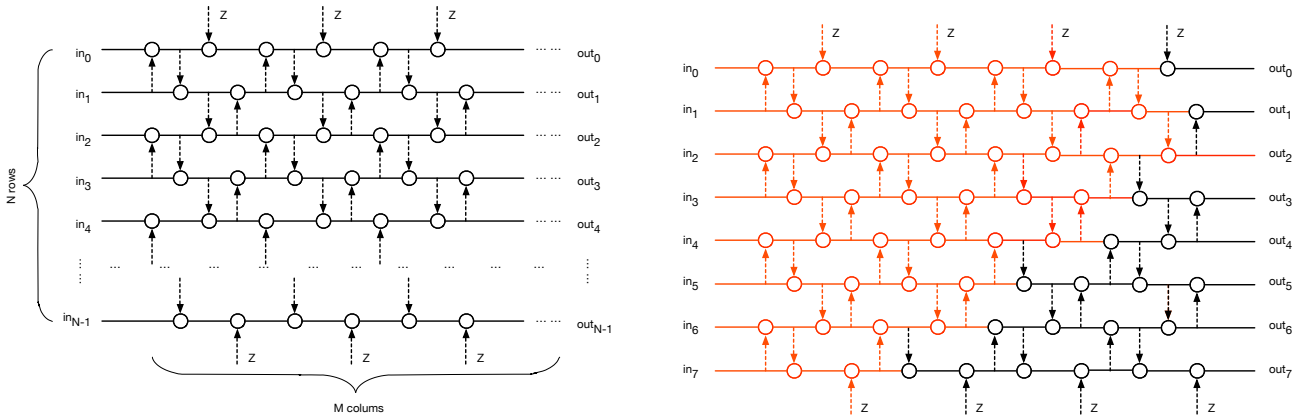


Figure 6: The LRR-DPUF architecture. (a) The general architecture with N-row by M-column. Some boundary virtual connection pins are marked by "Z" indicating dangling status; (b) An 8-row by 8-column LRR-DPUF. Logic cone of out_2 is highlighted in red color.

a legal and stable logic value. For simplicity, we use a bubble symbol to represent the unit cell in Figure 5, and the dashed arrow for the virtual connection pin.

4.2 The LRR-DPUF Logic Network

Recall that in Figure 4, higher dimension of XOR inputs require more number of hyper-planes for separation hence indicating higher level of non-linearity. This hints a cross-coupled XOR logic network. We therefore propose our LRR-DPUF architecture in Figure 6(a). It is an XOR based logic network with *N-input* and *N-output*. There are N *rows* and M *columns*, where one *column* refers to one unit cell per *row*. Note that the dashed arrow refers to virtual connection pin of the unit cell, indicating possible disconnection from the network. The unit cell's output pin is sequentially cascaded to next unit cell's key pin, and the virtual connection pin is **possibly** connected to its neighbor row. For 0^{th} and $(N-1)^{th}$ rows, i.e., the boundary rows, some unit cells' virtual connection pins are always dangled, and they are marked by letter "Z". The two boundary rows will have undistinguishable impact on the overall PUF performance due to the highly coupled XOR network dependencies.

We write down the logical expression for non-boundary nodes in recursion manner. Boundary node expressions can be easily derived by substituting with *input* or Z pins. Here the $k_{i,j}$ refers to *i-row j-column* output, and v refers to the virtual connection status.

$$k_{i,j} = \begin{cases} k_{i,j-1} \oplus (v \cdot k_{i+1,j-1} + \overline{v}), & i \text{ even, } j \text{ even;} \\ k_{i,j-1} \oplus (v \cdot k_{i-1,j-1} + \overline{v}), & i \text{ even, } j \text{ odd;} \\ k_{i,j-1} \oplus (v \cdot k_{i-1,j} + \overline{v}), & i \text{ odd, } j \text{ even;} \\ k_{i,j-1} \oplus (v \cdot k_{i+1,j} + \overline{v}), & i \text{ odd, } j \text{ odd.} \end{cases} \quad (1)$$

For better understanding of the LRR-DPUF network, we can view each row as a magic "signal tunnel". When the 1-bit input information is passing through this tunnel, it may or may not be flipped due to the uncertain status of virtual connections pins. Crucially, since each pair of neighbor rows have bi-directed virtual connections in between, each virtual connection also relies on its precedent as well as upper and lower neighbors' virtual connection statuses, resulting in a highly non-linear dependency graph. As long as the number of column M is no less than the number of row N, i.e., $M \geqslant N$, the logic cone of each out_i will have the potential to cover all the inputs in_j. We show an 8-*row* by 8-*column* LRR-DPUF example in Figure 6(b). The logic cone of out_2 is highlighted in red color which covers all inputs. The same coverage is true for every output. In next section, we will discuss in detail the LRR-DPUF properties.

5. PROPERTY ANALYSIS

In this section, we reveal important properties of the LRR-DPUF. First of all, it is intriguing to figure out how does the probability of the overall interconnect status impact the LRR-DPUF performance. In line with the definition used in Section 2, we define "connectivity rate" as the number of the non-dangled virtual connection pins over the total number of virtual connection pins in an LRR-DPUF architecture. For the two corner cases, i.e., the connectivity rate is 0 or 1, this LRR-DPUF logic network is reduced to a deterministic Boolean function, hence no longer a PUF. We therefore need to carefully control the connectivity rate. As we have discussed in Section 2, the connectivity rate can be controlled by circuit designers and/or lithography system accuracy.

Consider an N-row by N-column LRR-DPUF, the total number of the virtual connections, is N^2. From PUF design perspective, on one hand, the connectivity rate should not be too high in order to generate a rich space of unique PUFs. Suppose m out of the N^2 virtual connections are connected, i.e., the rate is $\frac{m}{N^2}$, there will be m-combinations of N^2, i.e., $\frac{N^2!}{m!(N^2-m)!}$ unique PUFs. We know the maximum number of unique combinations occurs when connectivity rate is 0.5. For a 64-row by 64-column LRR-DPUF, this ends up with $1.8e+1696$ unique PUF chips. However, on the other hand, intuitively, higher connectivity rate means more complex dependencies in the logic network, hence leads to stronger resilience to learning based attacks.

Property 1. *The LRR-DPUF logic network is non-linear, and higher connectivity rate leads to stronger non-linearity.*

As shown in Figure 4, since an XOR gate is linearly non-separable, the cascaded XOR is also linearly non-separable. High connectivity rate means more unit cells are cross-coupled, hence a higher level of non-linearity for the dependency graph. The maximum non-linearity happens when connectivity rate is 1, whereas, such a circuit is no longer a PUF. We will show in Section 6 that the LRR-DPUF shows strong resilience to non-linear machine learning attacks.

Theorem 1: *Equation* $\Pr[out_j = 1] = \Pr[out_j = 0] = 0.5$ *holds as long as* $\Pr[in_j = 1] = \Pr[in_j = 0] = 0.5,\ \forall j \in N$, *where* N *refers to the number of rows in LRR-DPUF.*

PROOF. Since $\Pr[in_j = 1] = \Pr[in_j = 0] = 0.5$, with Lemma 1, the *output* of the first unit cell U_{j0} in row j has $\Pr[U_{j0} = 1] = \Pr[U_{j0} = 0] = 0.5$, regardless of the virtual connection status on the node U_{j0}. Notion U_{jk} refers to the k^{th} unit cell in row j. By repeatedly applying Lemma 1 to U_{jk} $\forall k \in M$, we have $\Pr[U_{jk} = 1] = \Pr[U_{jk} = 0] = 0.5$. Hence, $\Pr[out_j = 1] = \Pr[out_j = 0] = 0.5$. Here M refers to the number of columns. □

Theorem 1 ensures that, when input follows uniform distribution, output in the same row retains the uniform distribution nature, i.e., equal chance to output 1 and 0. This theorem holds regardless of the virtual connection status.

Property 2. *There will be a sufficiently large space of unique LRR-DPUFs even if the connectivity rate is high.*

Consider a 64-row by 64-column LRR-DPUF, if 10 virtual connections get disconnected, there will be 3.6e+29 unique PUFs, and the connectivity rate is $\frac{4086}{4096} = 99.76\%$. When it increases to 20, the unique PUF space size goes up to 6.9e+53 and the connectivity rate is still as high as $\frac{4076}{4096} = 99.51\%$. Therefore, high connectivity rate does not adverse the uniqueness of the LRR-DPUF and is more preferred for better learning resilience.

Property 3. *Increasing the number of columns strengthens the resilience to machine learning attack.*

In Figure 6(a), increasing the column number of unit cells creates more interleaving connections between the neighbor rows, hence higher level of XOR logic dependency can be foreseen for each output. Furthermore, wider columns means more inter-dependent paths exist in the LRR-DPUF network, hence stronger resilience to learning based attack. Related discussion will be verified in Section 6.

Property 4. *Any subtle change on the virtual connections will be reflected to multiple outputs.*

Unlike other logics, like AND, OR, etc., for XOR logic, any input change will be reflected on the output. In addition, due to the cross-coupled and recursive dependencies in LRR-DPUF network (see Equation (1)), such changes will be propagated to multiple outputs. Even slight difference between two LRR-DPUFs can lead to significantly different CRPs characterizations, realizing high uniqueness of fingerprints. This will be verified by the inter Hamming Distance and "avalanche" effect in Section 6.

6. LRR-DPUF EVALUATIONS

In this section, we evaluate the performance of the LRR-DPUF, including statistical metrics and resilience to adverse attacks. We evaluate two LRR-DPUF configurations: 8-row by 8-column (8×8) and 64-row by 64-column (64×64), where we use the 8×8 LRR-DPUF in order to illustrate a full statistical image over the entire 256 CRPs. The 64×64 LRR-DPUF, by contrast, represents a practical PUF implementation, but due to the huge CRP space, we only evaluate a CRP subset.

6.1 Statistical Evaluation

There are four commonly used metrics for evaluating the statistical performance of a PUF [28]: inter Hamming Distance (inter HD), intra Hamming Distance (intra HD), Uniformity, and Bit-aliasing. Inter HD represents the ability of a PUF to uniquely distinguish two chips under the same challenge. Intra HD captures how reliable of a particular PUF under operational and environmental variations. Uniformity checks how uniform the ratio of 1s and 0s is in the response bits of a PUF. And the Bit-aliasing captures whether any response bit is biased and showing nearly identical result across different chips.

Table 1: Statistical evaluation on 8 × 8 LRR-DPUF with 256 CRPs

Type (Ideal Value)	conn. rate = 0.2		conn. rate = 0.9	
	Mean	Stdv.	Mean	Stdv.
Inter HD (0.5)	0.4188	0.0302	0.4943	0.0061
Intra HD (0.0)	0	0	0	0
Bit Alias (0.5)	0.5000	0.2067	0.5000	0.0730
Uniformity (0.5)	0.5000	0.1768	0.5000	0.1678

Table 2: Statistical evaluation on 64 × 64 LRR-DPUF with 100K CRPs

Type (Ideal Value)	conn. rate = 0.2		conn. rate = 0.9	
	Mean	Stdv.	Mean	Stdv.
Inter HD (0.5)	0.4999	0.0009	0.5000	0.0009
Intra HD (0.0)	0	0	0	0
Bit Alias (0.5)	0.5000	0.0504	0.5000	0.0499
Uniformity (0.5)	0.5000	0.0625	0.5000	0.0624

We first simulate the simple 8×8 LRR-DPUF with HSPICE using 45nm-PTM SPICE model. The skewed latch is designed to have 10× wider PMOS width for skewed-1 inverter, and 4× wider NMOS width for skewed-0 inverter. Besides, the PMOS in skewed-1 inverter uses low VT PMOS and high VT NMOS transistors, and vice versa for skewed-0 inverter. We specify the connectivity rate of 0.2 and 0.9 two cases, in order to reveal the connectivity impact on the LRR-DPUF performance. When the virtual connection pin of the latch is disconnected, we set high impedance to its input. For intra HD simulation, we sweep temperatures from $-20°C$ to $100°C$ and voltages from 0.7V to 1.2V. All 256 unique CRPs are used for the simulation. We show results in Table 1. Clearly, the intra HD is 0 across wide environmental and operational conditions, due to the nature of digitalized system as well as the use of strongly skewed latch. Higher connectivity rate shows better statistical performance which agrees the earlier discussions in Section 5.

We further examine the performance of 64×64 case. However, it is impractical to simulate a 64×64 LRR-DPUF by HSPICE due to the unacceptable simulation runtime. We therefore developed a behavioral emulator. Note that the only difference between the HSPICE simulation and the behavioral emulation is the emulator can not catch the intra HD metric, whereas all the rest statistical metrics can be fully emulated due to the digital nature. Consider the intra HD for 8×8 LRR-DPUF shows 0 by SPICE simulation, which is regardless of the network size,

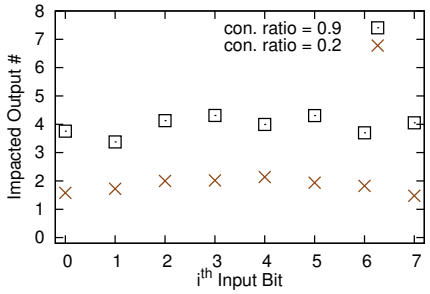


Figure 7: Avalanche effect of 8×8 LRR-DPUF over each input.

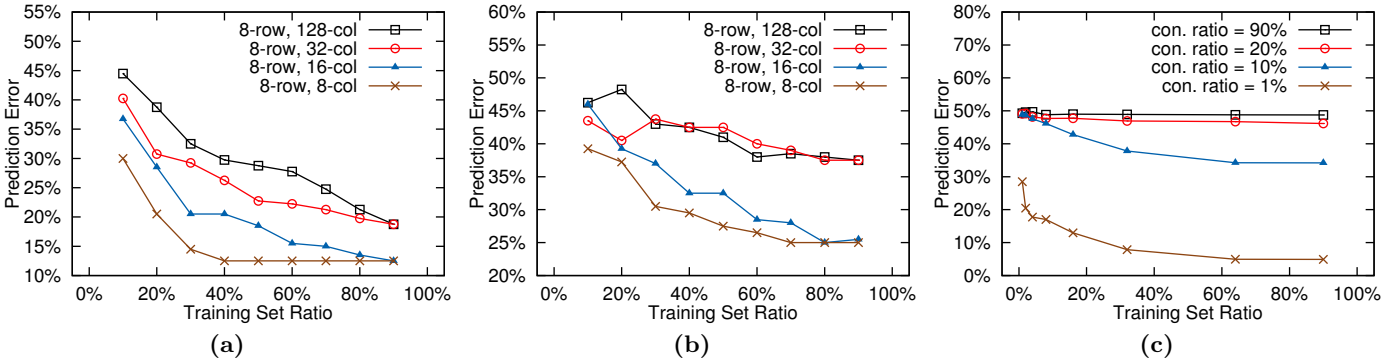


Figure 8: SVM attack for LRR-DPUFs over different configurations. (a) 8-row LRR-DPUFs with connectivity rate of 0.2 over different column sizes and training sizes; (b) 8-row LRR-DPUFs with connectivity rate of 0.9 over different column sizes and training sizes; (c) 64×64 LRR-DPUF over different connectivity rate and training size.

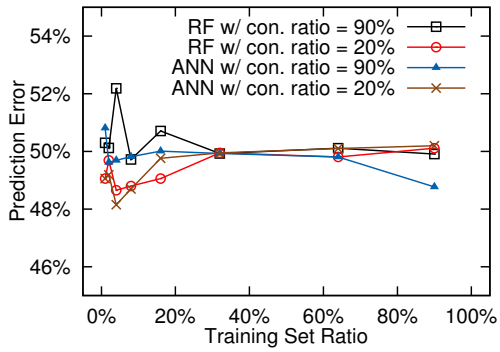


Figure 9: Additional learning model attacks on 64×64 LLR-DPUF, including a) Artificial neural network (ANN) with 10 hidden layers using Sigmoid function, and b) Random Forest (RF) with 15 trees in the forest.

the intra HD can be safely extrapolated to be 0 for 64×64 case as well. The results are shown in Table 2. 100K randomly selected unique CRPs are tested. For 64×64 LRR-DPUF, even small connectivity rate produces close to ideal inter HD as well as Bit-Alias and Uniformity, showing a outstanding statistical performance.

In addition, we examine the "avalanche effect" of LRR-DPUF, i.e., the Property 4. The avalanche effect checks that how many outputs will be affected (flipped) by changing an individual input bit. This property needs to be examined over the exhaustive CRP space, hence the 8×8 LRR-DPUF is used. We simulate 1000 unique 8×8 LRR-DPUF chips over the complete 256 CRPs. For each input bit flip, we exhaustively collect the number of flipped outputs over all the 2^7 cases. The average number of output flips for each input is plotted in Figure 7. For connectivity rate of 0.9, about 4 outputs, i.e., half of the 8 outputs, are flipped due to this single input change. Hence the adversary prediction via one bit change at a time is no better than a simple random guess. In other words, the LRR-DPUF provides the theoretically best potential of anti-prediction. In addition, when connectivity rate is 0.2, the number of impacted outputs is reduced to 2. This further supports the conclusion that a higher connectivity rate leads to better avalanche effect.

6.2 Adversary Attacks

In the end, we evaluate the resilience to various learning based reverse engineering attacks. We also verify that the connectivity rate and the LRR-DPUF column size will affect the resilience, i.e., the Properties 1 and 3. We attack the LRR-DPUF by support vector machine (SVM), where the nonlinear radial bias function (RBF) kernel is used. Again, we first check the 8-row LRR-DPUF chip family with different column sizes of 8, 16, 32, 128. The 256 CRPs are divided into training and testing sets. Considering that the training set size may affect the overall prediction accuracy, we sweep the training size from 10% to 90% of the 256 CRPs, and test with the rest CRPs for each case. The ideal prediction error should be exactly 0.5, meaning the machine learning prediction is no better than just random guess. Prediction error of 0 means the PUF can be completely predicted. For each LRR-DPUF, we apply the SVM attack onto **each** output bit at a time, and we report only the best prediction error among all output bits. It can be seen in Figure 8(a) and Figure 8(b) that, large size of columns helps to increase the prediction error. Besides, LRR-DPUFs with connectivity rate of 0.9 in Figure 8(b) generally show stronger resilience than that of 0.2 in Figure 8(a). Combining these two factors together, in Figure 8(b), we can see the attacks on the LRR-DPUF of 8-row 128-column show constantly bad predictions (about 0.4 prediction error) even when training set size is 90%.

Further, we apply SVM attack onto 64-row by 64-column LRR-DPUF in Figure 8(c). We randomly sampled 100K CRPs and divide them into various sizes of training and testing sets as well. For connectivity rate of 0.9 and 0.2, the LRR-DPUF constantly shows close to ideal resilience to the attack. Even when connectivity rate is reduced to 0.1, the prediction error is still above 0.35, and until the connectivity rate drops to 0.01, we start to observe low prediction errors.

Ultimately, some additional state-of-the-art learning models, including artificial neural network (ANN) and random forest (RF), are also applied to attack the 64×64 LRR-DPUF in Figure 9, where the ANN model is configured with 10 hidden layers and Sigmoid functions, and the RF model consists of 15 random trees. The prediction accuracy for both models, however, is constantly around 0.5 across wide range of connectivity rate and training set size. Overall, the proposed LRR-DPUF exhibits extraordinary resilience to learning based reverse engineering attacks.

7. FURTHER DISCUSSIONS

In this section, we further discuss some manufacturability issues for the proposed LRR-DPUF. Recall in Section 2, the VLSI interconnect variation is a Boolean variable with status of either connected or disconnected. When the metal stripe-pair is connected through tiny area, however, there will be high resistance (high-R) in between the two wire ends. Although

this still counts a connected status, the connectivity may be changed after certain time period which is called interconnect aging. To resolve this aging problem, we may need to pre-shrink all such high-R wires by applying high supply voltage and temperature before the PUF is characterized based on Black's equation [29, 30]:

$$MTF = \frac{A}{j^n} exp(\frac{E_a}{kT}), \quad (2)$$

where MTF refers to mean time to failure, i.e., the time needed to shrink the wire ends to be disconnected. j refers to current density, and T is the temperature. For typical metal wires, n equals to 2. All the others are irrelevant or constant factors.

Depending on the metal materials, the MTF for high-R region (i.e., metal wires with marginal connections) can take from hours to days [31]. Overall, by taking advantages of EM effects, such high-R wires will quickly shrink and transit into a stable disconnected status. Afterwards, the CRP can be characterized. Due to the limited space, we do not perform detailed simulation on this aspect which can be further validated by silicon results in future work.

8. CONCLUSION

In this paper, we proposed a novel learning resilient and reliable digital PUF. The randomness of the LRR-DPUF comes from the lithography process variations and is reflected in the form of interconnect randomness. We use strongly skewed latch to make the interconnect randomness compatible with digital CMOS circuit system ensuring 0 intra HD. A novel highly nonlinear logic architecture is developed to effectively spread and augment any interconnect randomness throughout the logic network. The LRR-DPUF has been demonstrated with outstanding statistical performance as well as strong resilience to various state-of-the-art machine learning attacks. We hope the proposed techniques can open a door for further exploring digital PUF designs and stimulate related researches in the future.

Acknowledgment

The authors would like to thank Jian Kuang from CUHK for providing guidance on lithography simulation.

9. REFERENCES

[1] Ingrid Verbauwhede and Roel Maes. Physically unclonable functions: manufacturing variability as an unclonable device identifier. In *Proc. GLSVLSI*, pages 455–460, 2011.

[2] Charles Herder, Meng-Day Yu, Farinaz Koushanfar, and Srinivas Devadas. Physical unclonable functions and applications: A tutorial. *Proceedings of the IEEE*, 102(8):1126–1141, 2014.

[3] Masoud Rostami, James B Wendt, Miodrag Potkonjak, and Farinaz Koushanfar. Quo vadis, PUF?: Trends and challenges of emerging physical-disorder based security. In *Proc. DATE*, pages 352:1–352:6, 2014.

[4] Mehrdad Majzoobi, Farinaz Koushanfar, and Miodrag Potkonjak. Testing techniques for hardware security. In *Proc. ITC*, pages 1–10, 2008.

[5] Meng Li, Jin Miao, Kai Zhong, and David Z. Pan. Practical public puf enabled by solving max-flow problem on chip. In *Proceedings of the 53rd Annual Design Automation Conference*, DAC '16, pages 164:1–164:6. ACM, 2016.

[6] Daniel E. Holcomb, Wayne P. Burleson, and Kevin Fu. Power-up SRAM state as an identifying fingerprint and source of true random numbers. *IEEE Transactions on Computers*, 58(9):1198–1210, 2009.

[7] Mafalda Cortez, Apurva Dargar, Said Hamdioui, and Geert-Jan Schrijen. Modeling SRAM start-up behavior for physical unclonable functions. In *Proc. DFT*, pages 1–6, 2012.

[8] Yu Zheng, Maryam S. Hashemian, and Swarup Bhunia. RESP: A robust physical unclonable function retrofitted into embedded SRAM array. In *Proc. DAC*, pages 60:1–60:9, 2013.

[9] Blaise Gassend, Dwaine Clarke, Marten Van Dijk, and Srinivas Devadas. Silicon physical random functions. In *Proc. CCS*, pages 148–160, 2002.

[10] G. Edward Suh and Srinivas Devadas. Physical unclonable functions for device authentication and secret key generation. In *Proc. DAC*, pages 9–14, 2007.

[11] Sandeep S. Kumar, Jorge Guajardo, Roel Maes, Geert-Jan Schrijen, and Pim Tuyls. The butterfly PUF protecting IP on every FPGA. In *Proc. HOST*, pages 67–70, 2008.

[12] Gang Qu and Chi-En Yin. Temperature-aware cooperative ring oscillator PUF. In *Proc. HOST*, pages 36–42, 2009.

[13] Yevgeniy Dodis, Leonid Reyzin, and Adam Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In *Proc. EUROCRYPT*, pages 523–540, 2004.

[14] Roel Maes, Anthony Van Herrewege, and Ingrid Verbauwhede. PUFKY: A fully functional PUF-based cryptographic key generator. In *Proc. CHES*, pages 302–319. Springer, 2012.

[15] Ulrich Ruhrmair, Jan Solter, Frank Sehnke, Xiaolin Xu, Ali Mahmoud, Vera Stoyanova, Gideon Dror, Jurgen Schmidhuber, Wayne Burleson, and Srinivas Devadas. PUF modeling attacks on simulated and silicon data. *IEEE Transactions on Information Forensics and Security*, 8(11):1876–1891, 2013.

[16] Xiaolin Xu and Wayne Burleson. Hybrid side-channel / machine-learning attacks on PUFs: A new threat? In *Proc. DATE*, pages 349:1–349:6, 2014.

[17] Teng Xu and Miodrag Potkonjak. Robust and flexible FPGA-based digital PUF. In *Proc. FPL*, pages 1–6, 2014.

[18] Teng Xu and Miodrag Potkonjak. Digital PUF using intentional faults. In *Proc. ISQED*, pages 448–451, 2015.

[19] Tibor Hegedűs and Nimrod Megiddo. On the geometric separability of boolean functions. *Discrete Applied Mathematics*, 66(3):205–218, 1996.

[20] Raghavan Kumar and Wayne Burleson. Litho-aware and low power design of a secure current-based physically unclonable function. In *Proc. ISLPED*, pages 402–407, 2013.

[21] Aswin Sreedhar and Sandip Kundu. Physically unclonable functions for embeded security based on lithographic variation. In *Proc. DATE*, pages 1–6, 2011.

[22] Domenic Forte and Ankur Srivastava. On improving the uniqueness of silicon-based physically unclonable functions via optical proximity correction. In *Proc. DAC*, pages 96–105, 2012.

[23] Alfred K. Wong, Richard A. Ferguson, and Scott M. Mansfield. The mask error factor in optical lithography. *IEEE TSM*, 13(2):235–242, 2000.

[24] V. Axelrad, K. Mikami, M. Smayling, K. Tsujita, and H. Yaegashi. Characterization of 1D layout technology at advanced nodes and low k1. In *Proc. SPIE*, volume 905213–905213, 2014.

[25] Wen-Hao Cheng and Jeff Farnsworth. Fundamental limit of ebeam lithography. In *Proc. SPIE*, volume 6607, 2007.

[26] Shayak Banerjee, Zhuo Li, and Sani R. Nassif. ICCAD-2013 CAD contest in mask optimization and benchmark suite. In *Proc. ICCAD*, pages 271–274, 2013.

[27] Yuri Granik and Nicolas B. Cobb. MEEF as a matrix. In *Photomask*, pages 980–991, 2002.

[28] Abhranil Maiti, Vikash Gunreddy, and Patrick Schaumont. A systematic method to evaluate and compare the performance of physical unclonable functions. In *Embedded Systems Design with FPGAs*, pages 245–267. Springer, 2013.

[29] R.L. De Orio, Hajdin Ceric, and Siegfried Selberherr. Physically based models of electromigration: from Black's equation to modern TCAD models. *Microelectronics Reliability*, 50(6):775–789, 2010.

[30] Kaustav Banerjee and Amit Mehrotra. Coupled analysis of electromigration reliability and performance in ULSI signal nets. In *Proc. ICCAD*, pages 158–164, 2001.

[31] Boon-Khin Liew, Peng Fang, Nathan W. Cheung, and Chenming Hu. Circuit reliability simulator for interconnect, via, and contact electromigration. *IEEE TED*, 39(11):2472–2479, 1992.