Exploiting Lithography Limits for Hardware Security Applications

Raihan Sayeed Khan*, Nafisa Noor, Chenglu Jin, Sadid Muneer, Faruk Dirisaglik, Adam Cywar, Phuong Ha Nguyen, Marten van Dijk, Ali Gokirmak, *Senior Member, IEEE*, and Helena Silva, *Senior Member, IEEE*

***Abstract*— Hardware security primitives such as physical obfuscated keys (POKs) allow tamper-resistant storage of random keys based on manufacturing or physical variability. The output bits of existing POK designs need to be first corrected due to measurement noise using error correction methods and then de-correlated by privacy amplification processes. These additional requirements increase the hardware overhead and reduce the efficiency of the system. In this work, we propose an intrinsically reliable POK design capable of generating random bits by exploiting the limits of the lithographic process for a given technology. Our design does not require any error correction and requires only XOR circuits for privacy amplification which reduces the hardware overhead of the whole system.**

I. Introduction

Hardware security has become a growing concern as software security protocols can no longer provide sufficient protection, especially in the current age of 'internet of things', where attackers have increased opportunities to directly access the system and exploit hardware vulnerabilities to extract valuable information. In order to prevent such attacks, several hardware security primitives have emerged. Physical unclonable functions (PUFs), introduced in 2001 [1], have been considered as promising hardware security primitives for building secure systems, such as intellectual property (IP) protection, identification, authentication protocols, cryptographic keys etc. [2]–[6]. A random response (i.e. output) is generated by a PUF when queried with a specific challenge (i.e. input) based on intrinsic manufacturing or physical variations. All PUFs' behaviors are unique and *physically unclonable*. PUF designs can be partitioned into two different categories: *strong PUFs* and *weak PUFs* (also called *physical obfuscated keys* or *POKs*) [7]. The major difference between these two categories is that a strong PUF has an exponential number of challenge-response pairs (CRPs), whereas a weak PUF or POK has a single or limited number of CRPs.

PUFs utilize process variations in the manufacturing processes of the devices. Using optical lithography, devices and interconnects are printed layer by layer in specified patterns with a monochromatic light source focused through a complex system of lenses and masks. The entire IC fabrication process consists of hundreds of patterning, deposition, etching, doping, and polishing steps, all with inherent variabilities [8]–[11]. These include irregularities in etching, doping and polishing steps, light source intensity fluctuations, lens and mask defects and alignment variations etc. [8]. The variability in the final dimensions of a device are a combination of variations introduced by all these steps. For typical logic and memory applications, these variations must be within tolerances required for the desired yield and reliability. For hardware security applications, however, these variations may be utilized to achieve inherent physical randomness.

Typically, the smallest features on a chip are safely above the lithographic process limit to ensure a high yield. However, if devices with features around or below the limit are fabricated, one can get a random range of dimensions that, depending on the device geometry, can result in randomly connected or open devices. In this paper, we report a design that can produce intrinsically reliable random bits with minimal overhead circuitry, based on the connectivity of line cells in an array. Such an array can produce a sequence of unique, reliable, random bits that can be used to build a physical obfuscated key.

II. POK Basics and Related Work

A. POK Basics

Typically, the POK circuit consists of two parts: a source of randomness and a randomness extraction circuit. The randomness extraction circuit extracts the analog signal obtained from the source of randomness and converts it into a raw digital response to be used in further digital computation. For example, in a ring oscillator (RO) based POK [12], the source of randomness is the random delay of different chains of inverters and the randomness extraction circuit consists of RO pairs with counters and comparators. For each RO pair, the two frequencies (measured by a counter) are compared (by a comparator) to produce a one-bit raw output. The collection of the digital raw output bits from all RO pairs forms the response of the POK.

A POK has the following properties:

- *Uniqueness:* The POK responses generated from different devices for the same challenge must be different. For any pair of POK instances, the

*This research is supported by the multi-university research initiative (MURI) of Air Force Office of Scientific Research (AFOSR) under the grant FA9550-14-1-0351Z.

R. S. Khan, N. Noor, C. Jin, S. Muneer, P. H. Nguyen, M. Dijk, A. Gokirmak, and H. Silva are with Department of Electrical and Computer Engineering at the University of Connecticut, Storrs, CT 06269 USA (e-mail: {raihan.khan, nafisa.noor, chenglu.jin, sadid.muneer, phuong_ha.nguyen, marten.van_dijk, ali.gokirmak, helena.silva}@uconn.edu). F. Dirisaglik and A. Cywar were with Department of Electrical and Computer Engineering at University of Connecticut, Storrs, CT 06269 USA. F. Dirisaglik is now with Department of Electrical and Electronics Engineering, Eskisehir Osmangazi University, Eskisehir, 26480, Turkey (e-mail: farukdirisaglik@gmail.com). A. Cywar is now with Analog Devices, Boston, MA, USA (email: adam.cywar@gmail.com).
*Corresponding author: *Raihan Sayeed Khan (raihan.khan@uconn.edu)*.

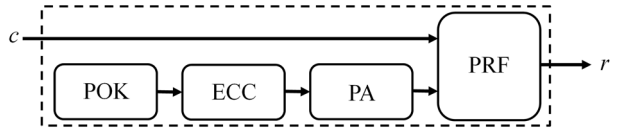

Fig. 1: A Digital PUF based on a POK architecture. The POK response is corrected by ECC and de-correlated by PA process. The processed POK response is then used as the key for the PRF which produces the final PUF response.

probability that the same challenge will lead to the same response should be close to $\frac{1}{2^n}$, where n is the size of the POK response.

- *Robustness/Reliability:* The POK response from a certain device must be the same when interrogated by the same challenge every time.
- *Unpredictability:* The POK response must not be predictable (without access to the POK) even if the challenge is known, i.e., the response should be randomly generated.
- *Tamper-resistance*: If an adversary gets physical access to the device and tries to perform reverse engineering, the source of randomness of the POK, and consequently the raw response must change. Therefore, if the raw response of a POK is used as a secret key in a system, the adversary cannot physically attack the device for retrieving the secret key.

The properties above are necessary for a POK response to be used as a key in a secure system. In particular, a POK's response bits must be reliable, independent and identically distributed (i.e. not correlated). These required properties may not be satisfied immediately for basic POK designs (such as the RO POK). To overcome the reliability and unpredictability problem, an error correcting code (ECC) [12], [13] with privacy amplification (PA) [14] is used to make a POK's (or more generally a PUF's) output robust and unpredictable. For example, in a digital PUF concept [Fig. 1], a keyed pseudo random function (PRF) is used as a PUF, where challenges are inputs to the PUF, the responses are the outputs of the PRF, and the key of the PRF is a POK response to a fixed challenge. The POK response is processed by ECC and PA block before going to the PRF.

In ECC, an extra circuit implements the error correction algorithm. Several constructions for error correction have been proposed such as fuzzy extractors (FEs) [14], computational fuzzy extractors (CFEs) [15], and learning parity with noise (LPN) [16]. In order to implement error correction algorithms, in addition to the basic POK circuit, several additional digital circuits need to be implemented and these account for a large hardware overhead on the final POK architecture [17]. Furthermore, since many digital processes are involved in producing a secret key, the efficiency of the system is significantly reduced in terms of execution time and power consumption [18], [19]. The difficulties with ECCs can be avoided if the basic POK output bits are intrinsically reliable (the ECC block in Fig. 1 will not be required).

For an intrinsically reliable POK, if the output bits are correlated, they cannot be directly used as a secret key. The output first needs to de-correlated by a privacy amplification process. However, if the to-be-decorrelated bits are reliable, a simple trick used in the construction of true random number generators can be applied: an XOR operation on several correlated bits can produce a response bit. This results in close-to-independent and identically distributed response bits without expensive ECC and PA circuitry.

B. *Lithography Based Hardware Security Primitives*

Process variations can be 'systematic', such as mask defects or deviations from layout design, which are reflected similarly on all dies, or 'random', such as local fluctuations in exposure, etch, deposition, or polishing steps, which result in device-to-device variations within each die or between dies.

Several resolution enhancement techniques, such as optical proximity correction (OPC), alternating phase shift mask, and immersion lithography, which are used to overcome sub-wavelength lithographic challenges to print 10 nm features using a much longer wavelength UV light source (193nm) [8], [9], can be intentionally avoided to engineer highly sensitive lithographic variations based PUFs (litho-PUFs) [8], [11]. Exploiting linewidth, length, and height variations in lithography simulation of litho-PUFs with forbidden pitch, Sreedhar et al. demonstrated output voltage fluctuations as a function of focus variation in the simulated devices [11]. Kumar et al. addressed fluctuations in the light intensity and durations and focus variations due to wafer tilt and resist thickness variations and showed improved inter-die and inter-wafer uniqueness with lithography simulation in the forbidden pitch zone [8]. To improve the performance of litho-PUFs the systematic variations should be suppressed and the random variations should be preserved or enhanced [8]. Forte et al. delineated OPC optimized for litho-PUF which enhances the random variations while reducing the systemic variations [20], [21]. Wang et al. proposed a stability guaranteed PUF based on random assembly errors which result in random permanent connections during a directed self-assembly (DSA) process [22].

All the reported works are based on results from simulations which deal with the assigned tolerance limits of a combination of certain parameters and may not reflect the actual variability expected from the physical variations arising from different lithography steps accurately. The results presented here show device level implementation of exploitation of lithography processes, a significant contribution to the lithography-based security primitive field.

III. Lithography Limit Based Physical Obfuscated Key

For a given lithography process, the yield can be varied between 0% and 100% for devices with sizes below the resolution limit and 100% for devices with sizes well above the resolution limit. We have exploited this yield variability below lithography-limit as randomness source for connectivity of line cells in a dense array of two contact devices. In this work, $Ge_2Sb_2Te_5$ (GST) line cells are used to

demonstrate the proof of concept, but the same idea can be applied to other materials (e.g. silicon, metal) to produce a reliable POK.

A. POK Fabrication and Operation

The bottom TiN contacted 50 nm thick GST line cells used in the experiments were fabricated on 700 nm silicon dioxide (SiO_2) thermally grown on silicon wafers and capped with silicon nitride (Si_3N_4) [Fig. 2] [23]. We have performed experiments on line cells from 10 dies from a 200 mm wafer to demonstrate the proof of concept. Devices (two contact line cells) with varying widths and lengths were fabricated using 90 nm technology. Design widths start from 40 nm, well below the expected capability of the lithography system, and increase with 2 nm increments. It is observed that at smaller device lengths the wider contact regions merge, hence GST structure is continuous from one contact region to the other, even for 40 nm design widths, resulting in line cells being always connected. On the other hand, the longer design lengths do not survive the pattern transfer process. Between these two extremes, there are regions where the connectivity of the line cells is random [Fig. 3]. Cells fabricated within this range can be used in arrays to produce secret keys unique to the chip.

In order to access a particular line cell in an array, we can integrate a FET with each line cell [Fig. 4(a)]. When the Read signal is ON, if the line cell is connected and the nMOSFET is appropriately sized, the output will be high (1). If the cell is broken, the output will be low (0). A number of such devices can be put in an array to create a POK where an *m* bit challenge sent to the row decoder. The decoder decides the row of line cells to produce the n bit response from 2^m rows. Every bit is generated from the connectivity of each line cell of the selected row [Fig. 4(b)].

As demonstrated in [19], a reliable SRAM POK can be made by applying a response reinforcement technique. However, to produce one raw bit secret, 6 transistors are needed, i.e., one SRAM cell. Our approach can produce one raw bit secret at the cost of 1 transistor and one line cell (very small compared to the transistor). This reduces the area overhead by almost 5 times.

B. POK Characteristics

1) Uniqueness

In Fig. 5, the cells with the dimensions presented by yellow boxes display 40%-60% probability of having electrical connectivity measured in 10 dies. This means that, the line cells in this region, almost have an equal probability

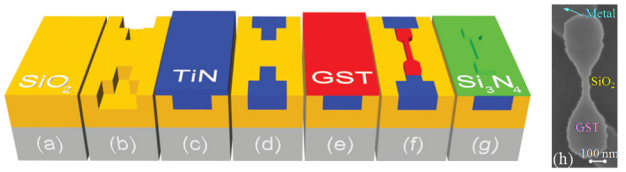

Fig. 2: Illustration of GST line cell fabrication steps (a) 700 nm SiO_2 thermal growth, (b) 250 nm deep trench formation using photolithography and reactive ion etching (RIE), (c) 300 nm TiN fill using chemical vapor deposition (CVD), (d) chemical mechanical planarization (CMP), (e) 50 nm GST film deposition by sputtering, (f) patterning of GST film using photolithography and RIE, (g) Si_3N_4 cap layer deposition, (h) SEM image of a fabricated line cell. The steps are compatible with CMOS process through back-end-of-line integration.

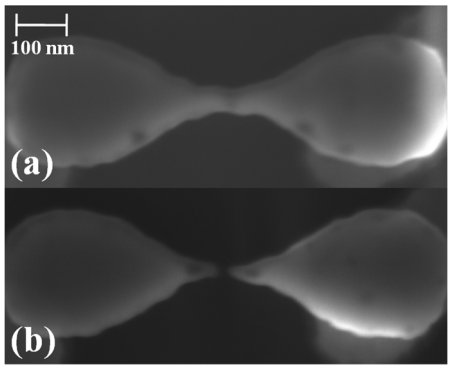

Fig. 3: SEM image of (a) connected cell and (b) broken cell.

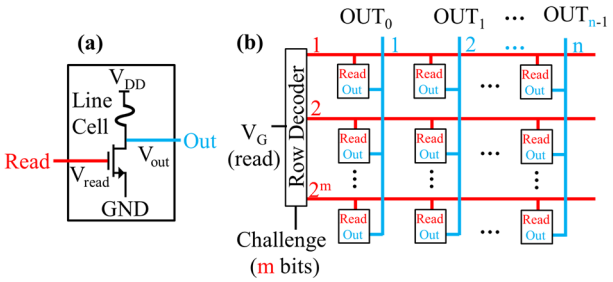

Fig. 4: Schematic of (a) one bit generation and (b) the architecture of the POK. The m bit challenge decides the row of line cells responsible for producing the n bit output. The nFETs connected to the cell ensure proper selection of line cells.

of being connected or broken. A bias (yield lower or higher than 50%), if exists, can be reduced by adding a XOR circuit as discussed in section II.A to produce independent and identically distributed bits.

54 pt
0.75 in
19.1 mm

37 pt
0.514 in
13.1 mm

2) Reliability

The proposed POK is based on the connectivity of the line cell. The connectivity of a line cell can only be changed at melting temperature (615°C for GST [24]) which requires much higher current than read current. Fig. 6 shows how the resistances of a broken/open and a connected/closed line cell change as temperature is varied from 295 K to 675 K. There are ~5 orders of difference in resistance between the two cells and no overlapping of resistance with temperature. This

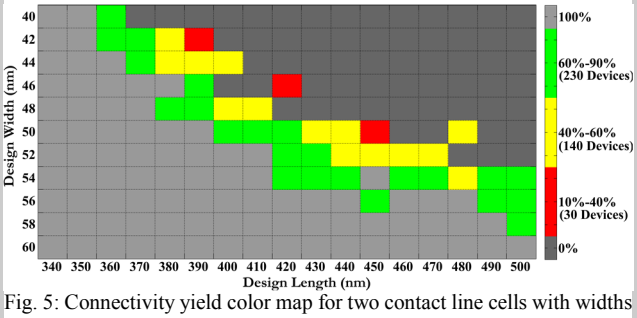

Fig. 5: Connectivity yield color map for two contact line cells with widths below the lithography limit (90 nm technology) for 10 dies (~1870 devices). Each block represents a group of line cells with similar design length and width. The width of the cells varies from group to group by 2 nm (y axis) and length varies by 10 nm (x axis). The colors represent percentage of line cells that are functional in a group consisting of 10 cells of same design dimensions (e.g. yellow corresponds to size blocks where 4 to 6 out of the 10 cells are connected). The 40-60% region of cell dimensions can be used to generate random, unique security keys. A yield map needs to be obtained for a given fabrication technology process which will vary depending on fabrication parameters.

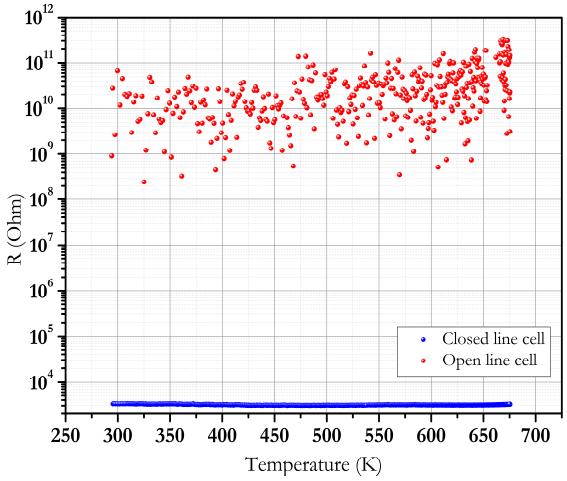

Fig. 6: Resistance of open/broken (red) and closed/connected (blue) line cell with temperature.

eliminates the need of error correction circuitry as the output of the POK is intrinsically reliable.

3) Unpredictability and Tamper-resistance

The cell to cell variations at lithography limit are highly dependent on tools and process parameters; hence, the proposed device is immune to cloning. Also, a soft material like GST (compared to silicon, SiO_2, and Si_3N_4) tends to erode very quickly under focused ion beam. This increases the time, effort and tool complexity required to tamper with the IC. With today's technology, even the manufacturer cannot choose the key which makes the design more secure.

IV. Conclusion

In this paper, we describe a method of generating a random bit as connectivity of line cells by exploiting the limits of lithography process. Since the lithography process is heavily dependent on process parameters, the POK is immune to cloning. The permanent nature of line cell connection frees the proposed design from additional reliability enhancement techniques to produce a reliable output. A simple privacy amplification process consisting only XOR circuit to produce truly uncorrelated output bits makes the design a lightweight one. The permanent nature of connection and finite number of line cells limits the number of challenge response pairs; making the system weak PUF or POK.

Acknowledgment

The line cells were fabricated at IBM Watson Research Center through a Joint Study Agreement. The authors would like to especially thank Dr. Chung Lam and Dr. Yu Zhu for their support in device fabrication. The facilities at UConn-Thermo Fisher Scientific Center at the University of Connecticut are used for the SEM images.

References

[1] R. Pappu, B. Recht, J. Taylor, and N. Gershenfeld, "Physical one-way functions," *Science (80-.).*, vol. 297, no. 5589, pp. 2026–2030, Sep. 2002.

[2] F. Armknecht, R. Maes, A.-R. Sadeghi, B. Sunar, and P. Tuyls, "Memory Leakage-Resilient Encryption Based on Physically Unclonable Functions," Springer, Berlin, Heidelberg, 2010, pp. 135–164.

[3] C. Brzuska, M. Fischlin, H. Schröder, and S. Katzenbeisser, "Physically uncloneable functions in the universal composition framework," in *Annual Cryptology Conference*, 2011, pp. 51–70.

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

[5] R. Maes and I. Verbauwhede, "Physically Unclonable Functions: A Study on the State of the Art and Future Research Directions," in *Towards Hardware-Intrinsic Security*, A.-R. Sadeghi and D. Naccache, Eds. Berlin Heidelberg: Springer, 2010, pp. 3–37.

[6] M.-D. (Mandel) Yu, D. M'Raihi, R. Sowell, and S. Devadas, "Lightweight and Secure PUF Key Storage Using Limits of Machine Learning," in *CHES*, 2011, pp. 358–373.

[7] B. Gassend, "Physical random functions," Massachusetts Institute of Technology, 2003.

[8] R. Kumar, S. N. Dhanuskodi, and S. Kundu, "On Manufacturing Aware Physical Design to Improve the Uniqueness of Silicon-Based Physically Unclonable Functions," in *VLSI*, 2014, pp. 381–386.

[9] S. Kundu, A. Sreedhar, and A. Sanyal, "Forbidden pitches in sub-wavelength lithography and their implications on design," in *Journal of Computer-Aided Materials Design*, 2007, vol. 14, no. 1, pp. 79–89.

[10] C. Mack, *Fundamental principles of optical lithography: the science of microfabrication*. 2008.

[11] A. Sreedhar and S. Kundu, "Physically unclonable functions for embeded security based on lithographic variation," in *DATE*, 2011, pp. 1–6.

[12] G. E. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," in *DAC*, 2007, pp. 9–14.

[13] B. Gassend, D. Clarke, M. van Dijk, and S. Devadas, "Controlled physical random functions," in *18th Annual Computer Security Applications Conference, 2002. Proceedings.*, pp. 149–160.

[14] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," in *EUROCRYPT*, 2004, pp. 523–540.

[15] B. Fuller, X. Meng, and L. Reyzin, "Computational Fuzzy Extractors," in *ASIACRYPT*, 2013, pp. 174–193.

[16] C. Herder, L. Ren, M. van Dijk, M.-D. Yu, and S. Devadas, "Trapdoor computational fuzzy extractors and stateless cryptographically-secure physical unclonable functions," *IEEE Trans. Dependable Secur. Comput.*, vol. 14, no. 1, pp. 65–82, Jan. 2017.

[17] C. Jin *et al.*, "FPGA Implementation of a Cryptographically-Secure PUF Based on Learning Parity with Noise," *Cryptography*, 2017.

[18] M. Bhargava, C. Cakir, and K. Mai, "Reliability enhancement of bi-stable PUFs in 65nm bulk CMOS," in *HOST*, 2012.

[19] M. Bhargava and K. Mai, "A High Reliability PUF Using Hot Carrier Injection Based Response Reinforcement," in *CHES*, G. Bertoni and J.-S. Coron, Eds. 2013, pp. 90–106.

[20] D. Forte and A. Srivastava, "On improving the uniqueness of silicon-based physically unclonable functions via optical proximity correction," *Des. Autom. Conf.*, pp. 96–105, 2012.

[21] D. Forte and A. Srivastava, "Manipulating manufacturing variations for better silicon-based physically unclonable functions," *2012 IEEE Comput. Soc. Annu.*, 2012.

[22] W.-C. Wang, Y. Yona, S. Diggavi, and P. Gupta, "LEDPUF: Stability-guaranteed physical unclonable functions through locally enhanced defectivity," in *2016 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, 2016, pp. 25–30.

[23] F. Dirisaglik *et al.*, "High speed, high temperature electrical characterization of phase change materials: metastable phases, crystallization dynamics, and resistance drift.," *Nanoscale*, vol. 7, no. 40, pp. 16625–30, 2015.

[24] H. P. Wong *et al.*, "Phase change memory," *Proc. IEEE*, vol. 98, no. 12, pp. 2201–2227, Oct. 2010.