[H. Idriss, P. Rojas, S. Alahmadi, T. Idriss, A. Carlson and M. Bayoumi. (May 27, 2022 to Jun. 01, 2022). Shadow PUFs: Generating Temporal PUFs with Properties Isomorphic to Delay-Based APUFs, 2022 IEEE International Symposium on Circuits and Systems (ISCAS), Austin, TX, USA, 2022, pp. 175-179, doi: https://doi.org/10.1109/ISCAS48785.2022.9937489 Reproduced for educational purposes only. Fair Use relied upon.]

Shadow PUFs: Generating Temporal PUFs with Properties Isomorphic to Delay-Based APUFs

Haytham Idriss*, Pablo Rojas*, Sara Alahmadi*,Tarek Idriss†, Albert Carlson‡, Magdy Bayoumi*
*Department of Electrical and Computer Engineering, University of Louisiana at Lafayette, USA
†Department of Computer Science, Western Washington University, USA; ‡Camqed Labs, USA
Email: {haytham.idriss1, pablo.rojas1, sara.alahmadi1, magdy.bayoumi}@louisiana.edu*;
idrisst@wwu.ed†; ltzap1@protonmail.com‡

***Abstract*—Physical Unclonable Functions (PUFs) are popular hardware security primitives that offer lightweight authentication for constrained devices. However, lightweight PUF-based authentication often limits the number of authentications provided or even throttle the device to prevent adversaries from collecting enough information that could compromise the device's security. This work introduces a Shadow PUF design, a controlled Strong PUF, to secure challenge-response exchanges against attackers and allow for an unlimited generation of unique responses. The proposed Shadow PUF design ensures security by periodically reconfiguring its behavior. The reconfiguration bits are generated by a static PUF primitive and are never exposed, while all authentication exchanges are done using the reconfigurable Shadow PUF. An ASIC Synthesis of the Shadow PUF demonstrates its small implementation area requirements and low power consumption. The security of the proposed PUF design against modeling attacks has also been analyzed.**

***Index Terms*—Hardware Security, Physically Unclonable Functions, Lightweight Authentication, Constrained Devices.**

I. Introduction

Advancements in miniaturization technologies have led to an exponential growth in the number of resource-constrained devices connected to the internet - resulting in the internet of things (IoT). The number of IoT devices connected to the internet is expected to exceed 75 billion by 2025 [1]. This increasing number of resource-constrained devices poses severe problems for IoT networks. Due to their limitations in utilizing conventional security techniques, IoT devices can become weak points in deployed networks [2] as reliable and lightweight encryption solutions are yet to be established for IoT devices [3]. Physically unclonable functions (PUFs) are hardware cryptographic primitives that have been a subject of research in lightweight security, where they are utilized for authentication and key generation protocols [4]–[7]. Due to their small implementation area and high resistance to probing attacks, PUF-based protocols are suitable for securing IoT devices that often lack reliable authentication schemes [2].

While ongoing research efforts continue to tackle the security of PUF-based authentication and key exchange protocols, an essential aspect of PUFs that has not been sufficiently investigated and addressed is their staticity. While devices utilizing traditional security primitives can be configured with new secret keys, a PUF's defining characteristics are embedded in the hardware and cannot be changed. Exposing information about the PUF circuit can allow an adversary to model the PUF through various machine learning (ML) algorithms, such as Logistic Regression (LR), Evolution Strategies (ES), and Deep Neural Networks (DNN). Hence, once a PUF circuit exposes enough information, its security is permanently compromised. This shortcoming has been the subject of several research efforts that aim to offer dynamic PUF constructs [8], [9]. The proposed controlled PUF-based design presents a novel approach for constructing temporary "Shadow" PUFs with properties isomorphic to the delay-based Arbiter PUF (APUF) in the sense that it mirrors the APUF's challenge-response behavior and statistical properties (i.e., randomness, uniqueness). The controlled element of this design (i.e., the Shadow PUF) is a logic implementation of the mathematical model of the Strong APUF, which makes the control logic a clone or a "Shadow" of the actual APUF on the device. The Shadow PUF, which is reconfigured periodically, would be used for authentication and secret message exchange; While the actual PUF circuitry would only be used to reconfigure the Shadow PUF module periodically.

The rest of the paper is organized as follows. Sec. II provides a background on physically unclonable functions and the Strong APUF. Sec. III introduces the proposed Shadow PUF architecture and presents experimental results on its statistical characteristics. Sec. IV analyzes the security of the Shadow PUF and the frequency by which it needs to be reconfigured. Sec. V summarizes the results and concludes the paper.

II. Background

A. Physically Unclonable Functions

A PUF is a physically disordered system that reacts to an external challenge $\vec{C}$ with a unique response R [10]. Due to the minuscule variations in the manufacturing process, different PUF circuits sharing the same layout would have unique responses. The 'unclonability' of PUF circuits makes them suitable for security applications as the response to a challenge cannot be predicted even when an adversary knows the circuit layout.

PUFs can be categorized as either Weak PUFs or Strong PUFs according to their challenge-response space [11]. The challenge-response space is the number of CRPs attainable from a PUF. It determines whether a PUF is classified as a Weak PUF or a Strong PUF. Weak PUFs feature a small

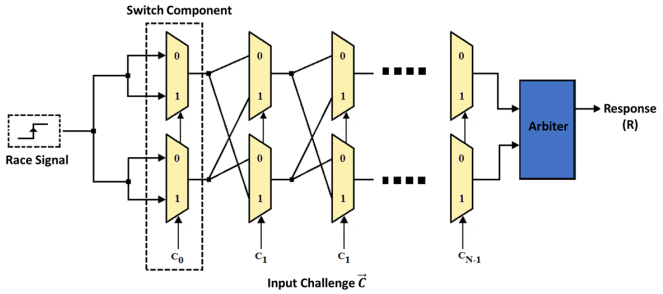

Figure 1. Delay Arbiter PUF Architecture.

challenge-response space and are usually structured as an array of unstable elements such as the SRAM PUF (S-PUF) [12] and the Ring Oscillator (RO-PUF) [13]. In such designs, the number of CRPs scales linearly with the size of the circuit. Strong PUFs such as the Arbiter PUF (APUF) [14], on the other hand, offer an ample challenge-response space that grows exponentially with its implementation size.

B. Arbiter PUF

The APUF is one of the earliest and most examined silicon-based PUFs. Fig. 1 describes the APUF's architecture and its components. An APUF consists of two symmetrically interconnected chains of switch components and an arbiter, usually a D flip-flop or an SR-Latch. The arbiter compares the delay of two identical paths to generate either a '0' or a '1' bit. The delay difference at the arbiter Δ can be expressed as a function of the differential delay vector $\vec{\omega}$, and $\vec{\Phi}$ the feature vector that is a function of the input challenge bits c_i [15]:

$$\Delta = \vec{\omega}^T \vec{\Phi} \text{ where, } \Phi_i = \prod_{i=1}^{k} (1 - 2c_i) \tag{1}$$

The APUF was found to be vulnerable to modeling attacks [15]–[18] where an adversary can collect challenge-response pairs (CRPs) and build a soft model of the PUF circuit. The scaling of PUF circuits through XOR-ing (XOR PUF) [13], Feed-Forward PUFs (FF-PUF) [19] and Lightweight Secure PUFs (LS-PUF) [20], are popular design approaches used to increase the non-linearity of the APUF's responses, and increases its resiliency against ML attacks. However, utilizing XOR-ing and Feed-Forward loops is often bound by the design's increased area overhead and low reliability [21].

III. Shadow PUF

A. Architecture

A Shadow PUF can be a representation of any Strong PUF that can be mathematically modeled. However, depending on the mathematical model of the Strong PUF, its implementation can be computationally heavy and not appropriate as a lightweight method to secure devices. In this work, the proposed Shadow PUF implements a reconfigurable, digitized version of the Arbiter PUF. The mathematical model of the APUF allows for a lightweight digital implementation using fixed-point arithmetic. As shown in Fig. 2, the Generator PUF (i.e., the static PUF) stores the response R_G of the corresponding challenge $\vec{C_G}$ in an addressable volatile memory block. The memory elements represent the differential delay values of $\vec{\omega}$ from equation (1) of the APUF. Each delay element is represented using 4 bits, requiring a total of $4 * 64 = 256$ bits (i.e., 256 memory cells) to represent all 64 delay values of a 64-bits APUF. The "control logic" in Fig. 2 evaluates the responses of the Shadow PUF based on the APUF's linear mathematical model. The arithmetic operation is implemented through a 10-bit, fixed-point, signed (2's complement), serial adder/subtractor circuit that is fed the values of $\vec{\omega}$ and $\vec{\Phi}$ one stage at a time. The vector $\vec{\Phi}$ is generated serially from the input challenge $\vec{C_S}$ according to equation (1). The final response R_S would be the sign bit of the accumulated output.

The weight values of the Shadow PUF are generated using the Generator PUF, after which they are used for the generation of future responses. However, the weights are periodically replaced. A counter is used to track the number of evaluated Shadow PUF responses and signal a request to reconfigure the Shadow PUF's weights. By limiting the number of CRPs queried per configuration, we prevent the Shadow PUF from exposing sufficient information that can be used to perform a successful ML-based modeling attack. The details of such attacks are discussed in further detail in section IV, where the security analysis of the Shadow PUF is presented.

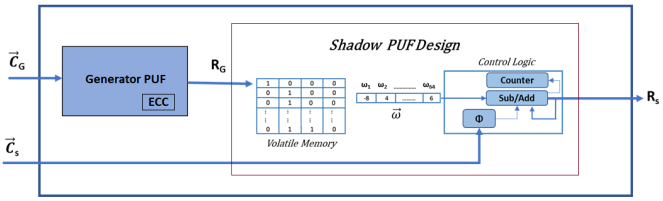

Figure 2. Proposed Shadow PUF Architecture.

B. Challenge-Response Exchange Scenario

The proposed challenge-response exchange demonstrates how the Shadow PUF's reconfigurability can be used to protect it against machine learning attacks. In this setup, the device would have an arbitrary Generator PUF, and the server would have access to a soft model of said PUF. The authentication steps for a simplified challenge-response exchange are shown in Fig. 3. The server first sends a set of "generation" challenges C_G, to the PUF device. These challenges are then fed through the Generator PUF, where the evaluated responses are used to represent the weights of the Shadow PUF. By using the same C_G, the server creates an identical Shadow PUF model based on its stored Generator PUF model. The device then sends a set of randomly generated challenges to the server C_A to query the server for the corresponding responses $R_{A'}$. Finally, the device can deem the server to be a legitimate party if the server's responses $R_{A'}$ are identical to its own R_A. A counter is used on the device to keep track of the

number of evaluated Shadow PUF responses and ensure not enough CRPs are exchanged to warrant a successful ML-based modeling attack. When weights need replacing, the device would inform the other party, and a new set of challenges C_G must be exchanged.

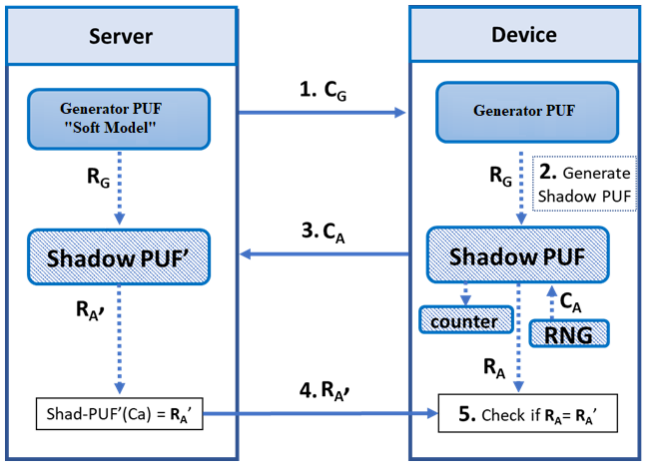

Figure 3. An illustrative diagram of a challenge-response exchange utilizing a Shadow PUF

C. Shadow PUF Physical Characteristics

A defining feature of a PUF circuit is the unpredictability of its responses, which is reflected by its randomness and uniqueness attributes. As we are using the APUF in our case-study to generate the weights of the Shadow PUF, we measure both of the PUFs' randomness and uniqueness to show that there is no decline in the attributes of generated Shadow PUFs. In this analysis, both the Shadow PUF and the APUF are 64-bit PUFs. The weights of the Shadow PUF ($\vec{\omega}$) are each represented by 4 bits. Using more than 4 bits would increase the design's area overhead with little benefit to its randomness and uniqueness. The APUF's software model used in our analysis is based on 65 nm predictive CMOS technology [22]. The evaluation of each metric is accomplished using randomly-generated 100 PUF instances and 10k CRPs.

1) Randomness: A PUF should have high randomness to ensure no bias in its responses. Let P_1 be the bias of responses towards the value' 1'. Then the randomness (H) of a PUF device can be defined as follows: [20], [23]:

$$H = min(P_1, 1 - P_1),\ where\ 0 \leq H \leq 0.5 \qquad (2)$$

Fig. 4 shows the randomness based on equation (2) for both PUFs. The APUF has a median and mean randomness H of 0.488 and 0.4855. Similarly, the Shadow PUF's randomness median and mean is 0.489 and 0.4875. Both PUFs shows high randomness, close to an ideal value of 0.5.

2) Uniqueness: The uniqueness of a PUF is measured by calculating the Inter-Chip Hamming Distance (Inter-Chip HD). Let D be the normalized Inter-Chip HD between two PUF instances. Uniqueness (U) is defined as follows [20], [23]:

$$U = min(D, 1 - D),\ where\ 0 \leq U \leq 0.5 \qquad (3)$$

In our design, more comparisons between the APUFs and the Shadow PUFs are necessary to ensure that there exists no correlation between the two. For this purpose, we introduce two additional uniqueness metrics, Inter-Uniqueness and Intra-Uniqueness, to evaluate the uniqueness of a Shadow PUF. The Inter-Uniqueness is measured by calculating the Hamming Distance between the responses of Shadow PUFs that are generated by distinct APUFs, using an identical set of challenges. While, Intra-Uniqueness is the Hamming Distance between responses of Shadow PUFs generated by the same APUF, using distinct sets of randomly generated responses.

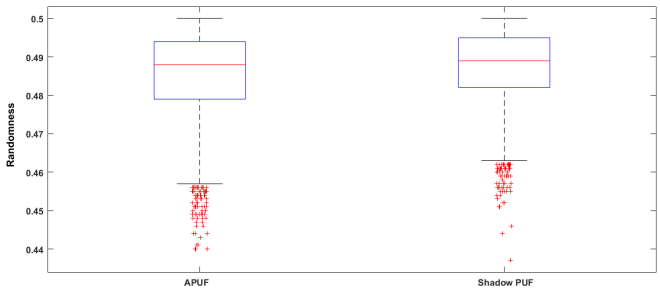

Figure 4. The Randomness of the Shadow PUF compared to that of the APUF.

The uniqueness of the APUF and the Shadow PUF is shown in Fig. 5. The lowest reported uniqueness is the Shadow PUF's inter-uniqueness with median and mean U of 0.467 and 0.46, with a negligible difference to other measured uniqueness metrics. The results have shown that the Shadow PUF has high uniqueness and it exhibited no loss in uniqueness compared to its Generator PUF, the APUF.

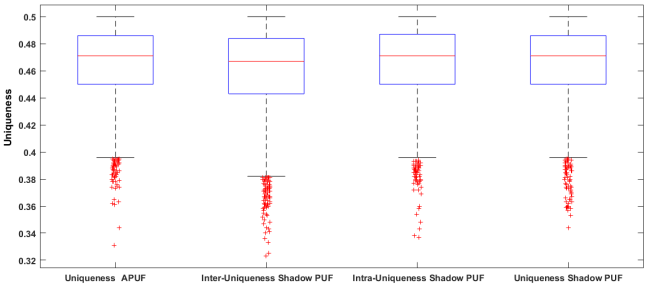

Figure 5. The uniqueness of the Shadow PUF compared to that of the APUF.

3) ASIC Design and Synthesis: A Verilog behavioral-level design of the proposed Shadow PUF was synthesized in $45nm$ [24] CMOS technology using Synopsys Design Vision. The 64-bit Shadow PUF utilizes a 10-bit counter for TTL and a 10-bit adder/subtractor. The weight values of the Shadow PUFs are stored on a synchronous RAM (256-bits), which is inferred by the synthesizer as Latches. The Shadow PUF, running at 1000 MHz, has an area overhead of 1583 gate equivalent (GE), and total power of 0.572 mW. This represents the additional area overhead imposed by utilizing the

proposed Shadow PUF design. The total area overhead would be dependent on the choice of Generator PUF. For instance, a 64-bits APUF has an area overhead of 356 GE, aside from the required error correction circuitry.

IV. Shadow PUF Security Analysis

There have been many studies on the modeling resistance of the APUF [15]–[18], where Deep Neural Networks (DNN) and Covariance Matrix Adaptation - Evolution Strategy (CMA-ES) have shown to be most successful. A single APUF instance can be modeled with 95% accuracy using either a DNN or a CMA-ES approach, as they both share the same mathematical model. Consequently, the same would hold true for the Shadow PUF. To ensure its security against modeling attacks, the Shadow PUF must reconfigure its weights periodically to avoid exposing compromising information about its weight values.

A. Modeling Attack on the Shadow PUF

In our analysis, we employ CMA-ES to model the PUF instances as it makes it easier to infer the numerical weight values of the PUF model, this is especially true if we are to utilize XOR-ing or Feed-Forward loops. We recall that the weights of the Shadow PUF are nothing but the responses generated by the APUF. Hence, by modeling the Shadow PUF, we are able to predict the values of the APUF's obfuscated responses.

Fig. 6 shows the Shadow PUF's trained model accuracy when using an increasing number of CRPs collected by the attacker. The figure also shows the prediction accuracy of each of the Shadow PUF's weights as the model converges to an accurate one. Looking at the results, we can observe that a certain subgroup of the Shadow PUF's weights can be predicted with higher accuracy than others when presented with a limited CRP set. The prediction accuracy of the first weight value is expected to surpass the overall model's accuracy, where the first weight in our design represents the delay difference of the last switch component in a traditional arbiter PUF. This aligns with the fact that a bit-flip at that stage has a higher chance of affecting the response bit-value at the output. Therefore, in order to guarantee security we must consider the worst-case scenario where the first weight is predicted earlier than other weights.

B. Modeling the Generator PUF and Time-To-Live Analysis

A reasonable assumption would be that the APUF's responses are never exposed to the attacker and only temporarily used to generate a Shadow PUF on the device. Side-channel and probing attacks that aim to extract internal information from the circuit are considered beyond the scope of this work. Therefore, one goal is to define the maximum number of responses that can be generated by the Shadow PUF without revealing sufficient information for the attacker to accurately model the Shadow PUF's weights.

The allowed number of evaluations of the Shadow PUF is referred to as Time-to-Live (TTL). TTL represent the minimum number of observed CRPs required to accurately predict the Shadow PUF's weight values. From Fig. 6, we observe that limiting the Shadow PUF to 100 CRPs would prevent an attacker from accurately predicting any of the Shadow PUF's weights with an accuracy higher than 60%. By limiting the TTL to 100 CRPs, we guarantee that the mean prediction accuracy of all weight representations is not consequential. If demanded by the application, a higher TTL would allow for less frequent reconfiguration of the Shadow PUF. The TTL can be easily increased through XOR-ing multiple responses or incorporating feed-forward loops in our arithmetic operations. For example, if we are to use a 2-XOR Shadow PUF design, that would increase the Shadow PUF's TTL to 2000 CRPs. Additionally, the Shadow PUF design can be used as a reconfigurable PUF instance in other proposed PUF-based protocols [4]–[7] to further increase the resiliency against ML attacks and allow for a longer TTL.

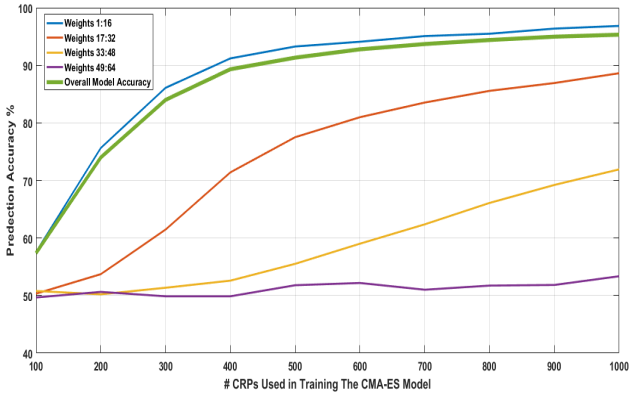

Figure 6. The accuracy of the Shadow PUF's model and its weight values vs. the number of observed CRPs

V. Summary and Conclusion

This work introduces a controlled Shadow PUF design for securing PUF primitives against ML-based modeling attacks. The Shadow PUF's weights are configured using a Strong PUF that can deterministically generate values uniquely associated with the device. Analysis of the Shadow PUF showed that its randomness and uniqueness metrics are near ideal and on par with the APUF circuit. Shadow PUF instances are reconfigured periodically to ensure resilience against machine learning attacks. A time-to-live metric for reconfiguring the weights has also been investigated based on experimental CMA-ES modeling attacks. This TTL metric indicates how often the weights need to be reconfigured to avoid exposing the on-device PUF primitive. The Shadow PUF can be easily scaled through suggested obfuscation techniques, such as XOR-ing and FF-loops, without fear of accumulating errors at the output. This is because any error correction that might be needed is done prior to the Shadow PUF generation. The Shadow PUF was synthesized using $45nm$ ASIC technology to showcase its small area and low power requirements, making it suitable for constrained devices.

REFERENCES

[1] L. Horwitz, "Internet of things (iot) - the future of iot miniguide: The burgeoning iot market continues," Mar 2021.

[2] P. Williams, P. Rojas, and M. Bayoumi, "Security taxonomy in iot – a survey," in *2019 IEEE 62nd International Midwest Symposium on Circuits and Systems (MWSCAS)*, pp. 560–565, 2019.

[3] K. Khalil, K. Elgazzar, A. Abdelgawad, and M. Bayoumi, "A security approach for coap-based internet of things resource discovery," in *2020 IEEE 6th World Forum on Internet of Things (WF-IoT)*, pp. 1–6, IEEE, 2020.

[4] M. Majzoobi, M. Rostami, F. Koushanfar, D. S. Wallach, and S. Devadas, "Slender puf protocol: A lightweight, robust, and secure authentication by substring matching," in *2012 IEEE Symposium on Security and Privacy Workshops*, pp. 33–44, 2012.

[5] T. A. Idriss, H. A. Idriss, and M. A. Bayoumi, "A lightweight puf-based authentication protocol using secret pattern recognition for constrained iot devices," *IEEE Access*, 2021.

[6] J. Zhang and C. Shen, "Set-based obfuscation for strong pufs against machine learning attacks," *IEEE transactions on circuits and systems I: regular papers*, vol. 68, no. 1, pp. 288–300, 2020.

[7] H. Xu, X. Chen, F. Zhu, and P. Li, "A novel security authentication protocol based on physical unclonable function for rfid healthcare systems," *Wireless Communications and Mobile Computing*, vol. 2021, 2021.

[8] E. Dubrova, O. Näslund, B. Degen, A. Gawell, and Y. Yu, "Crc-puf: A machine learning attack resistant lightweight puf construction," in *2019 IEEE European Symposium on Security and Privacy Workshops (EuroS PW)*, pp. 264–271, 2019.

[9] S. T. C. Konigsmark, D. Chen, and M. D. F. Wong, "Polypuf: Physically secure self-divergence," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, no. 7, pp. 1053–1066, 2016.

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

[11] U. Rührmair, F. Sehnke, J. Sölter, G. Dror, S. Devadas, and J. Schmidhuber, "Modeling attacks on physical unclonable functions," in *Proceedings of the 17th ACM conference on Computer and communications security*, pp. 237–249, 2010.

[12] J. Guajardo, S. S. Kumar, G.-J. Schrijen, and P. Tuyls, "Fpga intrinsic pufs and their use for ip protection," in *International workshop on cryptographic hardware and embedded systems*, pp. 63–80, Springer, 2007.

[13] G. E. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," in *2007 44th ACM/IEEE Design Automation Conference*, pp. 9–14, IEEE, 2007.

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

[15] U. Rührmair, J. Sölter, F. Sehnke, X. Xu, A. Mahmoud, V. Stoyanova, G. Dror, J. Schmidhuber, W. Burleson, and S. Devadas, "Puf modeling attacks on simulated and silicon data," *IEEE transactions on information forensics and security*, vol. 8, no. 11, pp. 1876–1891, 2013.

[16] F. Ganji, S. Tajik, and J.-P. Seifert, "Why attackers win: on the learnability of xor arbiter pufs," in *International Conference on Trust and Trustworthy Computing*, pp. 22–39, Springer, 2015.

[17] J. Tobisch and G. T. Becker, "On the scaling of machine learning attacks on pufs with application to noise bifurcation," in *International Workshop on Radio Frequency Identification: Security and Privacy Issues*, pp. 17–31, Springer, 2015.

[18] P. Santikellur, A. Bhattacharyay, and R. S. Chakraborty, "Deep learning based model building attacks on arbiter puf compositions.," *IACR Cryptol. ePrint Arch.*, vol. 2019, p. 566, 2019.

[19] J. W. Lee, D. Lim, B. Gassend, G. E. Suh, M. Van Dijk, and S. Devadas, "A technique to build a secret key in integrated circuits for identification and authentication applications," in *2004 Symposium on VLSI Circuits. Digest of Technical Papers (IEEE Cat. No. 04CH37525)*, pp. 176–179, IEEE, 2004.

[20] M. Majzoobi, F. Koushanfar, and M. Potkonjak, "Lightweight secure pufs," in *2008 IEEE/ACM International Conference on Computer-Aided Design*, pp. 670–673, IEEE, 2008.

[21] P. Rojas, H. Idriss, and M. Bayoumi, "Comparative analysis on the scaling properties of arbiter-based pufs," in *2020 IEEE 6th World Forum on Internet of Things (WF-IoT)*, pp. 1–6, IEEE, 2020.

[22] P. Sedcole and P. Y. Cheung, "Within-die delay variability in 90nm fpgas and beyond," in *2006 IEEE International Conference on Field Programmable Technology*, pp. 97–104, IEEE, 2006.

[23] Y. Hori, T. Yoshida, T. Katashita, and A. Satoh, "Quantitative and statistical performance evaluation of arbiter physical unclonable functions on fpgas," in *2010 International Conference on Reconfigurable Computing and FPGAs*, pp. 298–303, IEEE, 2010.

[24] J. E. Stine, I. Castellanos, M. Wood, J. Henson, F. Love, W. R. Davis, P. D. Franzon, M. Bucher, S. Basavarajaiah, J. Oh, *et al.*, "Freepdk: An open-source variation-aware design kit," in *2007 IEEE international conference on Microelectronic Systems Education (MSE'07)*, pp. 173–174, IEEE, 2007.