KU LEUVEN

ARENBERG DOCTORAL SCHOOL
Faculty of Engineering Science

Highly Reliable Physically Unclonable Functions

Design, characterization and security analysis

Kai-Hsin Chuang

Supervisor:
Prof. dr. ir. Ingrid Verbauwhede
Co-supervisor:
Prof. dr. ir. Guido Groeseneken

Dissertation presented in partial fulfillment of the requirements for the degree of Doctor of Engineering Science (PhD): Electrical Engineering

February 2020

Highly Reliable Physically Unclonable Functions

Design, characterization and security analysis

Kai-Hsin CHUANG

Examination committee:
Prof. dr. ir. Hugo Hens, chair
Prof. dr. ir. Ingrid Verbauwhede, supervisor
Prof. dr. ir. Guido Groeseneken, co-supervisor
Prof. dr. ir. Wim Dehaene
Prof. dr. ir. Nele Mentens
Dr. ir. Benedikt Gierlichs
Dr. ir. Robin Degraeve
(imec, Belgium)
Prof. dr. Said Hamdioui
(TU Delft, The Netherlands)

Dissertation presented in partial fulfillment of the requirements for the degree of Doctor of Engineering Science (PhD): Electrical Engineering

February 2020

Preface

It has been four years since I started my PhD in Leuven. I have never imagined that it would be so difficult to summarize this amazing period of my life. Many things can happen in four years, including moving into a new country, studying for PhD, and most importantly, meeting so many wonderful people. Without your company, I am pretty sure that I cannot get used to my life in Belgium and finish my PhD smoothly.

First of all, I would like to thank my promotor Prof. Ingrid Verbauwhede. This amazing journey has begun since I took the initiative to talk to Ingrid after her inspiring tutorial in Taiwan. Many thanks for your encouragement in the beginning and your guidance throughout my PhD study.

I would also like to thank all the jury members, your comments and questions are extremely helpful for polishing this thesis. Special thanks to Dr. Robin Degraeve for being my daily supervisor in imec, your supports, guidance and inspiring discussions are very much appreciated. Thanks to Prof. Guido Groeseneken for being my co-promotor, your advices during my study and on the thesis are invaluable. Thanks to Dr. Benedikt Gierlichs and Prof. Wim Dehaene for being assessors since the beginning of my PhD. Thanks to Prof. Nele Mentens and Prof. Said Hamdioui for being assessors in the jury. Thanks to Prof. Hugo Hens for chairing the two defenses.

Throughout my PhD study, I am very lucky to work with people in both COSIC and DRE. I would first like to thank Dimitri Linten for initiating this joint research project. Big thanks to Erik Bury, for being a great colleague and friend. I really appreciate your efforts on this project and all the useful skills I have learned from you. I also want to thank to Ben Kaczer, for your supports on this research and for always providing feedback on my papers.

I would like to thank all the COSICs, for making here a great place for doing research. Many thanks to Péla Noë, for your helps on solving all problems with great efficiency and kindness. Special thanks to the hardware group, for

all kinds of helps on my research and many insightful discussions: Adriaan Peetermans, Angshuman Karmakar, Antoon Purnal, Arthur Beckers, Danilo Šijačić, Furkan Turan, Jan-Pieter D'Anvers, Jeroen Delvaux, Jose Bermudo, Josep Balasch, Lennert Wouters, Milos Grujić, Pieter Maene, Sujoy Sinha Roy, Victor Arribas and Vladimir Rožić.

Thanks to many colleagues of imec, especially DREs, I really enjoyed working with you all: Alexander Grill, Barry O'Sullivan, Bart Vermeulen, Brecht Truijen, Geert Hellings, Jacopo Franco, João Bastos, Marko Simičić, Md Nur Kutubul Alam, Phillipe Roussel, Sofie Beyne, Stanislav Tyaginov, Sybren Santermans, Thomas Kallstenius, Yusuke Higashi and Zhicheng Wu.

My life in Leuven is joyful thanks to the great company of many friends. I would like to thank Jackson Wu for being a great friend throughout the entire four years. Many thanks to Wen-Chieh Chen, for being a great friend in so many perspectives. I also want to thank Cheng-Hsueh Tsai for always bringing up interesting discussions and frequently inviting me for travels. Thanks to Michiel Vandemaele, for being a thoughtful friend and for your helps on Dutch translations. Thanks to Shih-Hung Chen, for being a senior Taiwanese colleague who has helped me a lot. Thanks to Adrian Vaisman Chasin, Javier Diaz Fortuny, Simon Van Beek, Vamsi Putcha and Wei-Min Wu for being nice colleagues and friends. I would also like to thank Bohan Yang and Chaoyun Li, for your great friendship.

I would also like to thank the friends in the Mahjong group: Brent Hsu, Joy Liu, Kuang-Wei Cheng, Li-Hsin Li, Yi-Cheng Lai and Yun-Tzu Chang, for forming this group full of fun and warmth. The Taiwanese community in Leuven is like a big family, I want to thank you all for your friendship and the great moments we spent together: Ally Wang, Chia-Ling Chan, Cynthia Lin, Frank Kao, Fish Ling, Jason Dai, Jerry Lee, Joseph Huang, Judy Han, Kenny Wu, Kun Qian, Mark Hsu, Martin Fan, Michael Chen, Ping Huang, Shu-Chi Sheu, Tian-Li Wu, Ting-Wei Liao, Tsang-Hsuan Wang, William Lee, Wan-Ling Tsai, Wei-Ling Liao, Yu Fang and Yun-An Huang.

Finally, I want to express my greatest gratitude to my family. Thank you for being the strongest support during my study, I would not succeed on pursuing this degree without your encouragement.

Kent Chuang

Leuven, February 2020

Abstract

When moving into the upcoming Internet-of-Things (IoT) era, hardware security has become a very important research topic, due to the increasing demand of network-connected electronic products. New cryptographic algorithms and hardware implementations are proposed to make the IoT ecosystem more secure, but they cannot work without good randomness. As a root-of-trust that generates randomness, a physically unclonable function (PUF) is an essential building block for hardware security. A PUF provides each integrated circuit a unique fingerprint originating from the random variations of the electronic devices. The unique fingerprints are used for security applications such as entity authentication and cryptographic key generation. Ideally, the data generated by a PUF should be unpredictable, i.e., with full entropy, and stable over time and environmental changes. In reality, the generated data can be non-uniform or correlated, which results in entropy loss. Moreover, a PUF may not always exactly reproduce the same data, which results in instability. As a solution for these non-ideal phenomena, a new type of PUF implementations, the active PUF has been widely discussed in the literature. In the case of an active PUF, the PUF behavior is actively generated after chip fabrication. This thesis will discuss two active PUF implementations: a PUF using soft oxide breakdown (soft-BD PUF) and a reconfigurable type of PUF based on resistive-RAM (RRAM PUF).

The soft-BD PUF utilizes the gate oxide breakdown positions of CMOS transistors as the entropy source, since these positions are stable and unpredictable. This thesis covers the device/circuit-level design details and the characterization results from two test chips in 40nm CMOS. Excellent data stability and statistical properties have been demonstrated in this implementation. Secondly, we discuss the RRAM PUFs which are seemingly an ideal solution for reconfigurable PUFs. By the statistical analysis using an experiment-calibrated RRAM model, we have identified the physical property of RRAM that limits the reconfigurability. We have shown that true reconfigurability is unachievable in five listed RRAM PUF design examples.

In summary, we have demonstrated the advantages of active PUFs and also wrote down a cautionary note when using them in security applications.

Beknopte samenvatting

Met het betreden van het aankomende internet-der-dingentijdperk (Engels: Internet-of-Things (IoT) era), is hardwarebeveiliging een zeer belangrijk onderzoeksonderwerp geworden, omwille van de toenemende vraag naar elektronische apparaten verbonden over een netwerk. Nieuwe cryptografische algoritmen en hardware-implementaties worden voorgesteld om het IoT-ecosysteem veiliger te maken, maar deze kunnen niet werken zonder goede willekeur. Als een vertrouwensanker (Engels: root-of-trust) dat willekeur genereert, is een fysisch onkloonbare functie (Engels: physically unclonable function (PUF)) een essentieel bouwblok voor hardwarebeveiliging. Een PUF voorziet elke geïntegreerde schakeling van een unieke vingerafdruk die voortkomt uit de willekeurige variaties van de elektronische componenten. De unieke vingerafdrukken worden gebruikt voor beveiligingstoepassingen zoals entiteitsauthenticatie en generatie van cryptografische sleutels. Idealiter is de data die gegenereerd wordt door een PUF onvoorspelbaar, d.w.z. met volledige entropie en stabiel over tijd en bij veranderingen in de omgeving. In werkelijkheid kan de gegenereerde data niet-uniform of gecorreleerd zijn, wat zorgt voor entropieverlies. Bovendien is het mogelijk dat een PUF niet altijd exact dezelfde data reproduceert, wat resulteert in onstabiliteit. Een nieuw type van PUF-implementaties, nl. de actieve PUF, wordt uitgebreid besproken in de literatuur als een oplossing voor deze niet-ideale fenomenen. Bij een actieve PUF wordt het PUF-gedrag actief gegenereerd na chipfabricage. Dit proefschrift zal twee PUF's bespreken: een PUF gebaseerd op zwakke oxidedoorslag (Engels: soft oxide breakdown) (soft-BD-PUF) en een herconfigureerbaar type van PUF gebaseerd op resistieve RAM (Engels: resistive-RAM) (RRAM-PUF).

De soft-BD-PUF gebruikt de doorslagposities in het poortoxide (Engels: gate oxide) van CMOS-transistoren als de entropiebron, omdat deze posities stabiel en onvoorspelbaar zijn. Dit proefschrift behandelt de ontwerpdetails en karakterisatieresultaten op component- en circuitniveau voor twee testchips in 40nm-CMOS. Uitstekende datastabiliteit en statistische eigenschappen worden

aangetoond in deze implementatie. Ten tweede bespreken we RRAM-PUF's, die op het eerste gezicht een ideale oplossing lijken voor herconfigureerbare PUF's. Via een statistische analyse gebaseerd op een aan experimenten gekalibreerd RRAM-model, hebben we de fysische eigenschap van RRAM geïdentificeerd die de herconfigureerbaarheid beperkt. We hebben aangetoond dat werkelijke herconfigureerbaarheid niet kan behaald worden voor vijf genoemde RRAM-PUF-ontwerpvoorbeelden. Samengevat hebben we de voordelen van actieve PUF's aangetoond en ook een waarschuwing geformuleerd bij het gebruik ervan in beveiligingstoepassingen.

List of Abbreviations

AES Advanced Encryption Standard.

ASIC Application Specific Integrated Circuit.

BER Bit-Error Rate.

CMOS Complementary Metal-Oxide-Semiconductor.

CRP Challenge-Response Pair.

DFF D-type Flip-Flop.

EEPROM Electrically Erasable Programmable Read-Only Memory.

FPGA Field-Programmable Gate Array.

HCI Hot-Carrier Injection.

HRS High-Resistance State.

i.i.d. independent and identically distributed.

IC Integrated Circuit.

IoT Internet of Things.

LRS Low-Resistance State.

MOSFET Metal-Oxide-Semiconductor Field-Effect Transistor.

MRAM Magnetic Random Access Memory.

nBTI Negative Biased-Temperature Instability.

NVM Non-Volatile Memory.

OTP One Time Programmable Memory.

OxRAM Metal Oxide Resistive Random Access Memory.

pBTI Positive Biased-Temperature Instability.

PCM Phase-Change Memory.

PRNG Psuedo Random Number Generator.

PUF Physically Unclonable Function.

RRAM Resistive Random Access Memory.

SA Sense-Amplifier.

SHA Secure Hash Algorithm.

SRAM Static Random Access Memory.

TDDB Time-Dependent Dielectric Breakdown.

TMV Temporal Majority Voting.

TRNG True Random Number Generator.

XOR Exclusive-or.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The Internet of Things (IoT) is a major revolution for both commercial and industrial communication systems. In general, IoT is formed by a massive amount of connected devices deployed remotely throughout the environment. This revolution can make our daily life more convenient and the industry more efficient and eco-friendly. Unfortunately, due to the rapid growth of the number of IoT devices, many security challenges have been brought alongside all the benefits.

While improving our daily life, an enormous amount of data is gathered and being communicated between the IoT sensor nodes. The data in the IoT network can consist of sensitive information that would cause threats to system security and user privacy once leaked. Consequently, the communication channels between the IoT devices has to be secure, which is particularly challenging due to the amount of devices and the resource constraints for making a device. It is difficult to protect all the devices deployed in remote locations, while we cannot always afford strong cryptographic algorithms within the limited budget on both the manufacturing cost and battery energy.

1.1 Motivation

New security protocols and hardware primitives have to be developed in order to counter the increasing security and privacy threats in the IoT era. As a root of trust, true randomness (entropy) is required in most of the security protocols. A Physically Unclonable Function (PUF) generates *static* entropy, which is the

root of trust to create a unique feature for a device. The IoT devices need secret keys to secure their communications, and the PUF-based key generation scheme can be a more lightweight and secure solution to provide a unique key for each device.

Our goal in this thesis is to develop and characterize new types of PUFs, which will be a root of trust for IoT security as well as other security applications. As the purpose of a PUF is to generate static entropy, it has to be highly reliable. Our research focuses on the PUFs that are actively generated after a chip has been manufactured, instead of being generated during the chip manufacturing processes. This new type of *"active PUFs"* opens up a path towards zero instability and shows possibility to realize a reconfigurable PUF.

1.2 Thesis overview

This thesis is organized in six chapters. Following the current chapter which gives an overview of the thesis, chapter 2 provides the background and shows some recent developments on PUFs. Chapter 3 and chapter 4 discuss our works on a PUF based on soft oxide breakdown positions. Chapter 5 introduces the concept of reconfigurable RRAM PUF and shows that the reconfigurability is not ideal. Chapter 6 concludes this thesis.

- **Chapter 2.** This chapter gives a general introduction to the research field of PUFs. Starting from process variation, which is the origin of randomness, we will explain how PUFs are constructed in integrated circuits. Later on, the two main PUF-based applications, the key generation and entity authentication, will be introduced to show the use cases of PUFs. Several important PUF properties are listed to indicate which aspects need focus in PUF research. It is followed by a set of popular examples, which shows how the PUFs are designed and evaluated in practice. Finally, the importance of data stability will be reemphasized and the concept of active PUFs will be introduced in the last part of the chapter.

- **Chapter 3.** This chapter introduces the concept and device level characterization of the proposed PUF using soft oxide breakdown (soft-BD PUF). The chapter begins with the operating principle of the soft-BD PUF, using the binarized breakdown positions, followed by a detailed description of the test structure fabricated in a commercial 40nm CMOS technology. The experimental results on the forming process and the readout of the PUF values are introduced next, showing that this concept is feasible for realizing a PUF with near-ideal data stability. We will also

discuss the possible reliability concerns, while taking the observed non-ideal effects into account. Although this analysis does not provide solid proofs on good reliability, it does bring insights for further improvements. Finally, we will introduce an alternative implementation using the analog breakdown positions, which can generate multiple bits using one cell but it has much worse stability, comparing to the PUFs using the binarized breakdown positions.

The main contents of this chapter are derived from our papers: "*Physically Unclonable Function Using CMOS Breakdown Position*," presented at the *IEEE International Reliability Physics Symposium (IRPS)* 2017 [13] and "*A Multi-bit/cell PUF Using Analog Breakdown Positions in CMOS*," presented at the *IRPS* 2018 [14].

- **Chapter 4.** This chapter discusses a more complete implementation of the proposed soft-BD PUF. We first list the peripheral circuits for controlling the PUF circuit, showing the motivation of implementing this new design. The complete PUF implementation consists of a 32-by-32 PUF array, a serial control based on shift-registers, wordline/bitline drivers, sense-amplifiers and a scan-chain output interface. As one of the most important blocks, the sense-amplifiers are used to transform the current differences into digital bits, which has a full-custom design without the need of reference and analog bias. A set of Monte-carlo simulations is performed and it shows that the sense-amplifiers have excellent readout resolution but are sensitive to offsets. The experimental results later on show that this is a robust implementation, while the PUF is stable within a wide range of supply voltages and temperatures. The statistical analysis on the experimental results also show that the other PUF properties are well fulfilled. We have also tested the side-channel attack resilience of the PUF chips. Finally, we made a detailed comparison with prior PUF works, showing that the soft-BD PUF is competent.

  The main contents of this chapter are derived from our journal paper: "*A Physically Unclonable Function Using Soft Oxide Breakdown Featuring 0% Native BER and 51.8fJ/bit in 40nm CMOS,"* published in the *IEEE Journal of Solid-State Circuits (JSSC)* 2019 [16]. This journal paper is extended from the conference work [15], presented at the *IEEE Asian Solid-State Circuits Conference (A-SSCC)* 2018.

- **Chapter 5.** This chapter focuses on another type of active PUF, the RRAM PUFs, with an emphasis on analyzing the reconfigurability. The RRAM PUFs utilize the switching variability while programming the RRAM devices, which results in a random data pattern in each programming cycle, and the differences between cycles can be exploited as the source of reconfigurability. Five RRAM PUF implementations

are selected for detailed analysis, while each of these implementation is possible to be reconfigured. By studying the underlying physics of the programming procedure of the RRAM devices, we have identified that the differences between RRAM devices, the *RRAM-to-RRAM variations*, will limit the reconfigurability of the RRAM PUFs. By quantitative analysis based on the experiment-calibrated simulations, we have successfully shown how far the actual reconfigurability is apart from the ideal case. Finally, we have listed several possible solutions to use the RRAM PUFs without true reconfigurability.

The main contents of this chapter are derived from our paper: "*A Cautionary Note when Looking for a Truly Reconfigurable RRAM PUF*," published in the *IACR Transactions on Cryptographic Hardware and Embedded System (TCHES)* 2018 [17].

- **Chapter 6.** This chapter concludes this thesis, by summarizing the essential contributions of our work and providing future outlooks for further developments on the direction of active PUF.

Chapter 2

PUF concepts

PUFs in integrated circuits are essential hardware security primitives, which act as the root-of-trust for many security applications. Data stability is one of the most important PUF properties, and conventional solutions are not sufficient to push the data stability towards the ideal case. A new concept of active PUFs is therefore proposed to lead a path towards an ideal data stability. Moreover, it also brings up opportunities for reconfigurable PUFs.

2.1 PUFs in integrated circuits

The increasing demand of the *internet-of-things* (IoT) devices and applications brings big security challenges. As a root of trust for these IoT devices, PUFs embedded in the integrated circuits (ICs) are essential for securing the IoT ecosystem and many other communication networks. By harvesting the inherent process-induced random variations in CMOS technologies, PUF can generate unpredictable bits which enable various security applications, such as *key generation* [28, 61, 79] and *entity authentication* [29, 36].

A PUF implemented in ICs, or a so-called "PUF implementation" can be defined as follows:

Definition 2.1. *A PUF implementation is a circuit implementation consisting of a circuit design and a various amount of identically fabricated circuit instances based on this design, in which the circuit instances have all the required PUF properties. The subsequent circuit design is denoted as PUF design and the circuit instances are denoted as PUF instances.*

The PUF properties mentioned in this definition, are the circuit properties that a PUF instance must have, in order to be used in the security applications. More details about these properties will be discussed later in this chapter.

2.1.1 Process variation

Process variation in semiconductor manufacturing is an undesired phenomenon for most of the integrated circuits [50]. It might degrade the circuit functionality by introducing offsets, leakage current, glitches, ...etc. In the circuit designers' perspective, process variation is typically an effect that needs to be countered by utilizing various cancellation techniques, or by preserving a sufficient margin for any possible performance degradation. From the manufacturers' perspective, process variation is highly related to the yield of a technology platform, and manufacturers have to optimize the processing steps in order to reduce process variation so as to improve the yield. In summary, it requires massive amounts of efforts to counter the negative impacts of process variation in integrated circuits.

In contrast to other circuits, process variation is the foundation of conventional PUF implementations. It is exploited to generate the static entropy for PUFs [36, 58]. For a typical chip designed for commercial uses, millions of chips with the same design will be manufactured through identical processing steps. At a macro level, all the chips that have passed the yield tests are seemingly the same, but it is actually impossible to find two identical chips when taking the minor differences caused by process variations into account. A PUF circuit will transform these tiny differences in the devices and interconnects, into the unique fingerprint of a chip, typically in a digital format suitable for digital applications.

As the trend of technology scaling continues with Moore's law, the dimensions of the devices and interconnects in the newly developed technologies keep shrinking, which introduces relatively larger process variations, making the differences between circuits even more distinguishable. Moreover, as more processing steps and new materials are required for enabling these technologies, it also brings additional variability that can be potentially useful for PUFs. As a special circuit block with exceptional benefits from process variations, this increased variability has brought the PUF-related research further into the spotlight.

2.1.2 Weak and Strong PUFs

There are two main PUF categories depending on how the process variations are used. The main difference between these two designs is the number of outcomes that can be generated from a certain chip area. For weak PUFs, the number of outcomes scales polynomially (usually linearly) with the occupied area. On the other hand, the strong PUFs have outcomes that scale exponentially with the occupied area. These outcomes are typically in the form of binary bits.

The most common implementation of a weak PUF is a two-dimensional array of PUF cells, that are unit circuit blocks with identical design and physical layout. Each PUF cell can produce one bit or, occasionally, multiple bits. In this case, the number of produced bits from the entire array is linearly dependent to the number of cells, which is also linearly dependent on the area.

A typical design of a strong PUF consists of several cascading or parallel identical PUF circuit blocks. These circuit blocks are combined depending on the given input, or the so-called "challenge", and will produce one or multiple output bits based on this given challenge, that are known as the "response". A challenge in combination with the corresponding response is denoted as a challenge-response pair (CRP). For a strong PUF with N elements, the challenges typically also consist of N bits, and therefore there are 2^N possible challenges.

The terminology of "weak" and "strong" stands for the feasibility of using exhaustive search over all possible outcomes to build precise mathematical model. For example a precise model of a weak 1K-bit PUF can be constructed with minimum 1K queries, but for an ideal strong PUF with 64 stages, it requires 2^{64} queries to achieve the same precision. It shows that doing an exhaustive type of modeling for strong PUFs is almost impossible. If only considering exhaustive search type of attacks, one may suggest that a strong PUF is much more robust than a weak PUF. However, there are more efficient attacks [27,30] that can be exploited against strong PUF implementations, since the CRPs are highly correlated. As a result, the "strong" and "weak" PUFs are only used as the names to distinguish two different design concepts, and there is no implication on which one is better than the other.

2.1.3 Chapter organization

This chapter begins with listing the main applications that exploit PUFs, as discussed in section 2.2. Further on, the important requirements for PUFs will be introduced in section 2.3. Section 2.4 lists several well-known PUF implementations as examples to demonstrate the operating principle of PUFs.

Section 2.5 discusses the importance of data stability and shows three important stabilization techniques. The focus of this thesis, i.e. the new concept of *active PUFs*, will be introduced in section 2.6. Finally, section 2.7 will conclude this chapter.

2.2 PUF applications

Starting from the unique features of each chip inherently given by process variation, two main security applications can be build based on PUFs. The first application is the *key generation* [28, 61, 79], which extracts cryptographic keys from the random bits generated by the PUF elements embedded on-chip. The second application is the *entity authentication* [29, 36], which records the unique feature of each PUF chip and the records are later on used to examine whether a chip is authenticated or not. Among these two, there are still various kinds of PUF-based applications but they will not be thoroughly covered in this thesis.

2.2.1 Secret key generation and storage

For a device with cryptographic functions, it is necessary to have a secret key stored locally in order to perform encryption and decryption. The secrecy of the key is essential for the security of the system, according to the *Kerckhoff's principle*. In a typical approach, the key is stored in a non-volatile memory (NVM) such as flash memory, EEPROM, e-fuse, anti-fuse or battery-backed SRAM. As illustrated in Figure 2.1, a chip has an embedded secure NVM and certain cryptographic functions to secure the communications in the network. The secret key therefore needs to be generated using a true random number generator (TRNG) and a subsequent key generation function, which ensures the quality of the generated keys, and then transferred to the NVM for storage. Note that both the generation of secret key and the programming of the key need to be performed in a secure environment, either on-chip or off-chip, to preserve the integrity of the secret key. The programming interface of the NVM also needs to be disabled permanently, typically by blowing the fuse, to avoid possible attacks that could manipulate the key values.

The security of such system relies on that the NVM is not accessible by an adversary. This requirement is, however, difficult to fulfill, since one can reverse-engineer the chip and try to readout the content electrically or apply different attack techniques on the NVM [51, 78]. Moreover, the device needs to trust the party who generates the secret key, and it also needs to trust that the enrollment procedure is not tampered by an adversary. From a price perspective, using

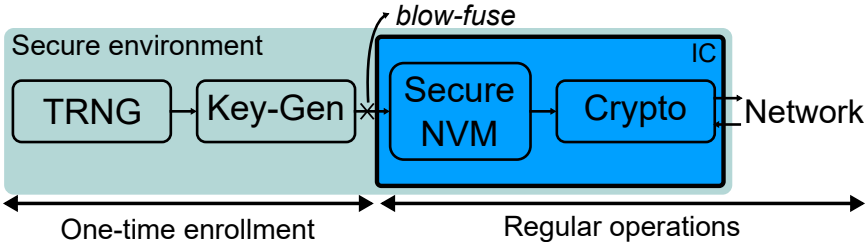


Figure 2.1: A simplified illustration for key-storage in a secure NVM. The key is generated by a TRNG and programmed into the NVM in a secure environment. The TRNG and key generation circuit in this example is on an off-chip device, but they can be also embedded on-chip in some other scenarios.

a NVM to store the key is also not cost-efficient, since it usually needs extra processing steps to make embedded NVM which increases the manufacturing cost for an IC, making it unsuitable for the lightweight security primitives.

2.2.2 PUF-based key generation

In order to improve the key generation and storage scenario, an alternative solution based on PUFs can be exploited. As illustrated in Figure 2.2, a PUF with an array that contains entropy from process variation is readout by an interface circuit as digital bits, and these bits are first sent to the helper-data algorithm [28] (usually off-chip) to generate the subsequent helper data for key generation in the field. The generated helper-data is then programmed into an embedded NVM block that can be readable but not writable (not fully secured), and the helper-data is later-on used to reconstruct the targeted secret key [61] for further cryptographic application, such as the advanced encryption standard (AES). Since cryptographic algorithms typically do not tolerate noise, the generated key must be completely stable.

The error-correction and entropy-extraction are the two main functions that need to be present in the PUF-based key generator. The error-correction is required because the PUF readouts are usually noisy, and hence the generated bits will contain errors that need correction. The purpose of the entropy-extraction is to counter the effect of non-uniformity of PUF bits, it will ensure the generated key can meet the specification of the target cryptographic application. More details about these non-ideal effects will be discussed in the next section.

The PUF-based key generation has several advantages compared to key storage in a secure NVM. First of all, the requirement of being physical secure is shifted

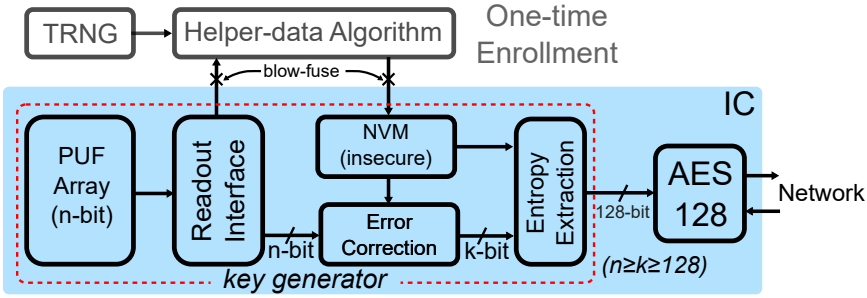


Figure 2.2: An example of a chip with embedded PUF-based AES key generator.

from NVM to the PUFs, which have better resilience against physical attacks since it is rather difficult to measure process variations. The second main advantage is that it is less harmful if the party who performs the enrollment is not trustworthy, since the entropy is not supplied by an external source. There are more features that motivate the study of PUFs, but they will not be thoroughly covered since these two advantages are the most representative ones in our opinion.

2.2.3 PUF-based entity authentication

As another popular application, the PUF-based entity authentication has also drawn a lot of attention. Based on the well-known challenge-response authentication method [72], a lightweight entity authentication scheme using the so-called "strong PUF" has been proposed [36]. As illustrated in Figure 2.3, the PUF-based entity authentication consists of two phases, the enrollment phase and the authentication phase.

In the enrollment phase, the server will generate a large number of random *challenges*, using a true/pseudo random number generator (TRNG/PRNG), for each device with the same type of embedded PUF block. The generated challenges will be sent to the devices and the server will then receive the *responses* corresponding to each challenge. These *challenge-response pairs* (CRPs) will be recorded as the references for authentication. Note that the enrollment phase has to happen in a secure environment.

In the authentication phase, for a device claiming to be a registered one, the server will pick a challenge from the database and send it to the device that needs authentication. The device will generate and send back the response corresponding to the received challenge. This procedure can be repeated several times, until the server can decide the device is authenticated by receiving

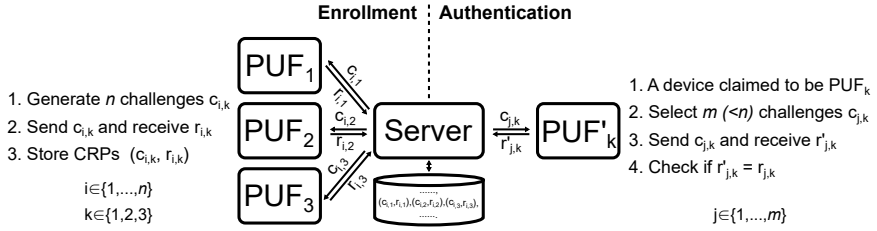


Figure 2.3: Illustration of a PUF-based entity authentication scheme [36].

enough correct responses, or is rejected by receiving one or multiple incorrect response(s). Since the communication between the server and the remote device can go through a public channel, the server has to discard the used CRPs after performing an authentication, in order to avoid replay attack.

As a requirement for this system, the PUF has to produce an enormous amount of CRPs at its disposal, and it brings up the need of a "strong" PUF, in which the number of CRPs is exponential in the size of the PUF circuit. Besides the requirement on the amount of CRPs, they also need to be uncorrelated in order to prevent the attacks based on CRP modeling, but this requirement is not yet fulfilled in any existing design and is still an open research question in this field. The lack of the ideal solution has brought up many challenges on designing a robust PUF-based entity authentication scheme.

2.2.4 Challenges for PUF-based entity authentication

The main challenge for the existing strong PUF design is the vulnerability to modeling attacks, in particular the ones based on machine learning (ML) techniques. As described in [42, 74], a simple arbiter PUF [36] without XOR can be accurately modeled with only a few thousand CRPs, which indeed shows serious vulnerability against modeling attacks.

One simple adjustment to improve modeling attack resilience [27, 30, 42] is XOR the response of multiple strong PUFs together [79], so-called XOR PUF. With k amount of arbiter PUFs in parallel, by applying an identical challenge c to all the arbiter PUFs and XORing the subsequent responses $r_1, r_2, ...r_k$ of the PUFs, the output of the XOR PUF becomes $r = r_1 \oplus r_2 \oplus ... \oplus r_k$. Since the XOR is a non-linear operation, it is more difficult to model comparing to the linear additive behavior of the arbiter PUFs. Consequently, by increasing the number of the PUFs in parallel, the effort of reaching certain modeling accuracy is increased, as shown in [42, 74]. Despite there are indeed positive effects brought by XORs, it is not possible to infinitely increase the amount of

XORed PUFs due to the noise amplification problem. For example, by XORing two independent digital signals both with error rates of 5%, the error rate of the XOR output will be 9.5%. If one keeps increasing the number of XORs, in the end the output will be too noise and this XOR PUF can no longer be used as a PUF.

As an attempt to increase the number of XORs for the XOR-PUF, it has been claimed in [95] that the noise amplification problem can be solved by limiting the CRP space, which only allows the noise-insensitive ones. Despite data stability is improved and it is possible to increase the number of XORs, the vulnerability against modeling attack still exists, since the number of available CRPs is also significantly reduced. As an example showing the vulnerability of this approach, a machine learning attack recently proposed in [27] has broken the "PUF-FSM" protocol [35], which is a PUF-based authentication protocol that using only the error-free CRPs.

Due to the vulnerability against modeling attacks of strong PUFs, PUF-based authentication protocols, such as the aforementioned PUF-FSM, need to be built, in order to secure the authentication process. Similar to all types of security protocols, there are many PUF-based protocols proposed, but according to the analysis shown in [27, 29], most of the protocols cannot fulfill the security requirements. As a result, there are still many critical challenges in the field of PUF-based entity authentication, and it certainly requires a major breakthrough of strong PUF designs.

2.3 PUF properties

A good PUF design needs to have certain properties in order to maintain secure and reliable cryptographic applications, including *uniqueness*, *reliability*, *unclonability* and *physical-attack immunity*. There are also more detailed lists of PUF properties and their definitions discussed in literature (e.g. in [58]), but they are not all relevant for a certain application, and therefore they will not be fully covered in this thesis. In addition, since this work focuses on studying weak PUFs, the following discussion will take the aspect of designing weak PUFs.

2.3.1 Preliminaries

For a better understanding of the PUF properties, a set of definitions needs to be first addressed.

Min-entropy

The *min-entropy* [71] of a random variable X is defined in Equation 2.1, where x is the outcome of X as a binary vector with length N and χ represents the set of all the possible outcomes. It is determined by the maximum probability of correctly guessing an outcome x the first time. By taking a logarithmic scale, the relation $0 \leq H_\infty \leq N$, which shows that the min-entropy is upper-bounded by the number of bits.

$$H_\infty = -\log_2 \left\{ \max_{x \in \chi} [\Pr(X = x)] \right\} \tag{2.1}$$

By definition, min-entropy is the worst-case estimation of predictability of a secret variable. As for a secret key with length of λ, it is required to have full entropy, i.e. $H_\infty = \lambda$, to fulfill the requirements for a cryptographic application. In the case of full entropy, the random variable X follows a uniform distribution.

Hamming weight

The Hamming weight is a measure for the number of ones in a binary vector. For a binary vector of length N, $x = x_1, x_2, ..., x_N$, where $x_i \in \{0, 1\}$, the Hamming weigth can be defined as Equation 2.2. It is also common to use the normalized Hamming weight, which is defined as $\mathrm{HW}_{\mathrm{norm}}(x) = \mathrm{HW}(x)/N$.

$$\mathrm{HW}(x) = \sum_{i=1}^{N} x_i \tag{2.2}$$

Hamming distance

The Hamming distance is a common measure for the difference between two binary vectors. For two binary vectors x and y with the same length of N, the hamming distance is defined as the number of bitwise differences between x and y. The hamming distance can also be expressed using bitwise XOR, as shown in Equation 2.3, and the normalized hamming distance is defined as $\mathrm{HD}_{\mathrm{norm}}(x, y) = \mathrm{HD}(x, y)/N$.

$$\mathrm{HD}(x, y) = \sum_{i=1}^{N} x_i \oplus y_i \tag{2.3}$$

2.3.2 Uniqueness

Uniqueness is the most fundamental property of a PUF design. For the PUFs implemented in ICs, the PUF circuit in each chip consists of self-derivative secret information, which acts as the chip-fingerprint. As a short and informal definition, we can say a PUF implementation and the subsequent chip-fingerprints are unique, if one cannot find any similarity between these chip-fingerprints.

As the uniqueness is usually seen as a concept rather than an actual measure, it is difficult to formally define the uniqueness of a PUF implementation. Alternatively, a more formal definition for the *ideal uniqueness* is stated below.

Definition 2.2. *For two arbitrary selected chips, c_i and c_j ($i \neq j$), which consist of identically manufactured PUF circuits with n-bit outputs, $o_i = o_{i,1}, o_{i,2}, ..., o_{i,n}$ and $o_j = o_{j,1}, o_{j,2}, ..., o_{j,n}$, where $o_{i,k} \in \{0,1\}$ and $o_{j,k} \in \{0,1\}$ $(1 \leq k \leq n)$. The ideal uniqueness is presented, if the probability of $o_{i,k} = o_{j,k}$ is equal to 0.5 for any combination of i, j and k.*

By fulfilling this criteria, for an adversary who tries to guess the output of an unknown PUF, the probability of having a correct guess will not increase with the number of PUFs which are known by the adversary.

Uniqueness evaluation

For evaluation purpose, the inter-chip (inter-PUF) hamming distance is often used as an index to check the uniqueness of a PUF implementation. Starting from the ideal case as defined in Definition 2.2, the inter-chip hamming distance of c_i and c_j, $\mathrm{HD}(o_i, o_j) = \sum_{k=1}^{n} o_{i,k} \oplus o_{j,k}$.

Given that $\Pr(o_{i,k} = o_{j,k}) = 0.5$ for any i, j and k, each $o_{i,k} \oplus o_{j,k}$ can be seen as a Bernoulli trial with probability of 0.5, and thus $\mathrm{HD}(o_i, o_j)$ is the sum of n Bernoulli trials, which can be therefore seen as a binomial random variable with parameters n and p=0.5. If we calculate the inter-chip hamming distances of PUFs with ideal uniqueness, the result will form a *binomial distribution.*

For a given set of PUFs that needs evaluation, the typical procedure is to compute the $\mathrm{HD}_{\mathrm{inter}}$ all possible combinations, and then check whether the result follows binomial distribution or not. If this distribution deviates from the binomial case, one can conclude that the uniqueness is not ideal. On the other hand, if the resulting $\mathrm{HD}_{\mathrm{inter}}$ shows a distribution close to the ideal case within a particular boundary, one can say that this PUF implementation has a good uniqueness. It is, however, insufficient to prove the ideal uniqueness

is achieved, since we will need an infinite amount of measures to prove that $\Pr(o_{i,k} = o_{j,k}) = 0.5$ is true. Moreover, to the best of our knowledge, there is no clear standard on how to choose the boundary for comparing the experimental $\mathrm{HD}_{\mathrm{inter}}$ and the ideal $\mathrm{HD}_{\mathrm{inter}}$.

2.3.3 Randomness

Randomness is a similar terminology frequently used together with uniqueness, to evaluate the performance of a PUF implementation. Similarly, the ideal randomness can be defined as follows.

Definition 2.3. *For a PUF circuit which generate n-bit outputs,* $o_i = o_{i,1}, o_{i,2}, ..., o_{i,n}$ *where* $o_{i,k} \in \{0, 1\}$ $(1 \leq k \leq n)$. *The ideal randomness is presented, if the min-entropy* $H_\infty(o_i)$ *is equal to n for any i.*

It should be noted that Definition 2.2 and 2.3 are actually equivalent, because we will find contradictions while assuming there is a case with ideal uniqueness but without ideal randomness.

Randomness within individual PUFs

Despite the definition for ideal randomness seems redundant, the term *randomness* is still widely used while evaluating PUFs. In most of the cases, randomness is evaluated on the outputs of PUFs individually, by examining the statistical properties of each PUF output without considering any possible information can be learned from the other PUFs.

The purpose of such evaluation is mainly to determine the vulnerability of the secret information stored within a PUF, in the condition where a part of the information is known by an adversary.

Definition 2.4. *Given a specific PUF instance which can generate k-bits outputs,* $o = \{o_1, o_2, ..., o_k\}$ *and the outputs are partitioned into two sequences with m-bit and n-bit* $(0 \leq m < k;\ m + n = k)$*:* $o_{public} = \{o_1, o_2, ..., o_m\}$ *and* $o_{private} = \{o_{m+1}, o_{m+2}, ..., o_{m+n}\}$. *We assume that an adversary has the goal of guessing* $o_{private}$*; and the initial guess, which is done before any PUF readout event, is denoted as* φ_0. *Next,* o_{public} *has been readout and revealed to this adversary, while* $o_{private}$ *remains in secret. The adversary now has a new guess based on the m-bit additional information, and the new guess is denoted as* φ_m. *The randomness within this PUF can be considered ideal, if* $\Pr(\varphi_0 = o_{private}) = \Pr(\varphi_m = o_{private})$ *is true for any* $0 \leq m < k$.

Despite showing a good randomness within a single PUF instance is far from sufficient to prove the PUF implementation is good to be used, it can help to understand the feasibility of a certain PUF implementation without massive production, since the uniqueness can only be well-evaluated with a large amount of instances, and it can be costly to manufacture these additional PUF instances.

For example, if one is designing a PUF using a field-programmable gate array (FPGA), there is typically only one instance as the prototype, making it infeasible to examine the uniqueness in the design phase. On the other hand, if the output of this prototype already shows poor randomness, the designer can quickly decide to change the design or platform, before performing trials on multiple FPGAs. The same advantage holds for ASIC designers, while they do not always have many chips in their disposal in the prototype design, evaluating randomness can help them to decide whether more chips should be fabricated to better test the uniqueness.

Randomness evaluation

Unlike evaluating uniqueness based on inter-chip hamming distance, the methods for evaluating randomness is less standardized. In practice, there are three popular approaches, including checking the hamming weight of PUF outputs, checking the bit-wise correlation of the output bits, and performing standard statistical tests (AIS31 [48], NIST800-22 [75]). These methods are usually used together in order to provide a better confidence on the randomness of a PUF implementation, and will also be used to evaluate our own design in the coming chapters.

The hamming weight of the PUF output can reveal that whether the output bits are biased, i.e., is more likely to be "1" or "0". By definition, the normalized hamming weight should be close to 0.5 if the bits are unbiased. On the other hand, if these bits are biased, the probability of correctly guessing the PUF output will be increased, which shows a sign for non-ideal randomness.

If there are correlations between PUF bits, by knowing a part of them, it would be more probable to correctly guess the rest. In this case, the equality $\Pr(\varphi_0 = o_{private}) = \Pr(\varphi_m = o_{private})$ is no longer true for any m, and hence showing that the randomness is not ideal. Similarly, the statistical tests also tell if there are patterns within the PUF output, which can be exploited to help guessing the unknown part.

As an important remark for these evaluation approaches, we should keep in mind that all of them are sanity checks, which can only tell whether the tested

PUF implementation has poor randomness or not, but is unable to "prove" that an ideal randomness is achieved.

2.3.4 Reliability

Reliability is essential for all types of circuits, in which PUF is definitely included. When discussing on how reliable a PUF is, there are mainly two aspects: (i) the data stability about the PUF behavior on a short time scale; and (ii) the long-term reliability focuses on the aging effects of the PUF circuits.

Data stability

As the PUF will be readout multiple times in the field, it is desired to have the same readout values every time. Data stability defines the ability of a PUF to produce the same output even if exhibiting transient fluctuations. There are several internal and external factors that can affect the data stability of a PUF. The internal ones can be device noise, clock jitter, cross-talks or glitches that might cause an error during data readout. On the other hand, the external factors like supply voltage, temperature, humidity or even radiation can as well affect the PUF readout, by changing the device characteristics or by injecting additional fluctuations into the circuit. Besides the indeterministic effects causing stability issues, these internal and external factors can possibly be controlled to intentionally inject faults to the PUFs as a type of attack.

A more formal definition can be described as follows:

Definition 2.5. *Given $o(t)$ as the output of a PUF measured at time t, and $o(t) = \mathrm{f}(\eta(t), t)$, where f is a deterministic function and $\eta(t)$ is a set of i time varying parameters, $\eta(t) = \{\eta_1(t), \eta_2(t), ..., \eta_i(t)\}$. In addition, a set Θ defines all the η variants that resulting in reasonable operating conditions (no extreme cases such as zero supply voltage), including any possible random fluctuations. The data stability of the PUF can be considered ideal, if $o(t)$ is a constant for any t such that $\eta(t) \in \Theta$.*

It should be noted that this definition is not limited to a short time scale. In practice, the data stability is evaluated within a time span $t_{\mathrm{begin}} < t < t_{\mathrm{end}}$. In such case, the data stability is considered ideal, *if $o(t)$ is a constant for any t such that $\eta(t) \in \Theta$ and $t_{\mathrm{begin}} < t < t_{\mathrm{end}}$.*

Data stability is typically measured by the bit-error rate (BER) of the PUF output, which can be defined using the following equation, in which n_{errors} is the number of error bits found in the measurement and n_{bits} is the total number

of measured output bits. *An error-bit is a bit which differs from its expected value.*

$$BER = \frac{n_{\text{errors}}}{n_{\text{bits}}} \times 100\% \tag{2.4}$$

Another frequently used term to evaluate the data stability of a PUF is the percentage of output bits which are not completely stable, which is usually denoted as *unstable-bits. A PUF output bit is considered unstable, if one can find at least two readout values of this PUF-bit that are different.* The index unstable-bit can be defined as the following equation, in which n_{unstable} is the total number of unstable PUF-bits found within a PUF, and n_{puf} is the total number of output bits of this PUF.

$$Unstable\ bit = \frac{n_{\text{unstable}}}{n_{\text{puf}}} \times 100\% \tag{2.5}$$

Long-term reliability

The long-term reliability mainly concerns the effects that are not recoverable by normal circuit operations within a short time period. These effects usually take a long time to accumulate, and are therefore referred as the aging effects. There are several known mechanisms in integrated circuits, include the Negative/Positive Biased-Temperature Instability (nBTI/pBTI) [39], hot-carrier injection (HCI) [80], time-dependent dielectric breakdown (TDDB) [26] and electromigration [8]. These effects usually cause degradation or even create permanent damages to the devices or interconnects, which can severely affect the circuit functionality.

The effects caused by these degradation mechanisms can also be considered as the *time-dependent variability*, in contrast to the aforementioned process variations that are considered as the *time-zero variability.* Since PUFs are first proposed utilizing the later term, by adding time-dependent variability on top, the original PUF behavior might be overwritten and there will be risks of losing the original secret key or the chip identity.

Following the definition for data stability, the ideal long-term reliability can be defined as follows:

Definition 2.6. *Given $o(t)$ as the output of a PUF measured at time t, and $o(t) = \mathrm{f}(\eta(t), t)$, where f is a deterministic function and $\eta(t)$ is a set of i time varying parameters, $\eta(t) = \{\eta_1(t), \eta_2(t), ..., \eta_i(t)\}$. In addition, a set Θ defines all the η variants that resulting in reasonable operating conditions (no extreme*

cases such as zero supply voltage), including any possible random fluctuations. The long-term reliability of the PUF can be considered ideal, if $o(t_1) = o(t_2)$ *for any combination of* t_1 *and* t_2 *such that* $\eta(t_1) = \eta(t_2) \in \Theta$.

The long-term reliability of a PUF is typically measured by its expected *lifetime*, t_{life}, such that Definition 2.6 for ideal reliability is true for any combination of t_1 and t_2 less than t_{life}. For a good PUF design, this lifetime needs to be longer than the system specification, e.g., 10 years.

2.3.5 Unclonability

As given the name of physically unclonable functions, a PUF, by definition, cannot be cloned physically. For the PUFs in ICs, it is technically impossible to make a physical clone, since the precision in the processing steps is not able to reproduce the exact process variations, otherwise they would not exist.

In addition, the mathematical unclonability is also widely discussed, i.e. it should be unfeasible to make a mathematical model for a certain PUF. By giving an adversary the ability of infinitely querying a PUF for its outcomes, mathematical unclonability will not hold for a weak PUF, since it can be accurately modeled by an exhaustive readout. For strong PUFs, mathematical unclonability is referring to the immunity against modeling attacks, which is not strong for most of the cases. As a result, most of the existing PUF designs are found not mathematically unclonable.

2.3.6 Physical attack immunity

As one of the concerns for secure NVMs is the physical attacks that may destroy the security claims, the physical attack immunity for PUFs is also an important property to be studied. These attacks may be reverse-engineering, laser injection attacks or any combination of various types of attacks. For example, if an adversary decapsulates the metal layers of a PUF chip and uses a microscope to observe the PUF cells, it should not be possible to find out the actual PUF data.

2.4 PUF implementations

As PUF has been in the spotlight of hardware security, a large number of PUF designs are introduced and new designs are continuously being proposed

from different research domains, including analog/digital ASICs, FPGAs, semiconductor devices and reliability physics. Due to the rapid growing of this research field, it is not possible to list all the state-of-the-art designs and it is also unnecessary to do so. Consequently, this section will provide a brief overview of more well-known PUF implementations, in order to help gaining better understanding via practical examples.

2.4.1 SRAM PUF

As one of the pioneer proposals, a SRAM PUF consists of a two-dimensional array of SRAM cells with a typically cell structure shown in Figure 2.4 (a). In an SRAM cell, since the two sub-inverters are connected in a loop, it results in a voltage transfer curve, as shown Figure 2.4(b). When acting as a memory cell, the data is stored based on the voltages of the internal nodes V_{L} and V_{R}, the stored data will be "1" when $V_{\mathrm{L}} = 0$ and $V_{\mathrm{R}} = V_{\mathrm{DD}}$ (state-1); conversely, the data will be "0" when $V_{\mathrm{L}} = V_{\mathrm{DD}}$ and $V_{\mathrm{R}} = 0$ (state-0). Both the read and write process are performed by adjusting the voltages of wordlines (WLs) and bitlines (BLs).

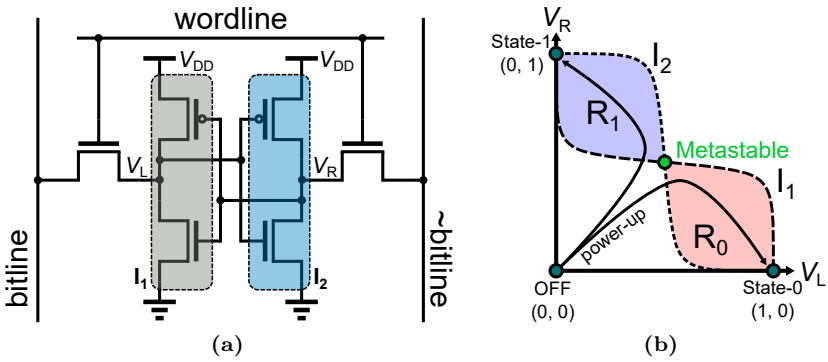
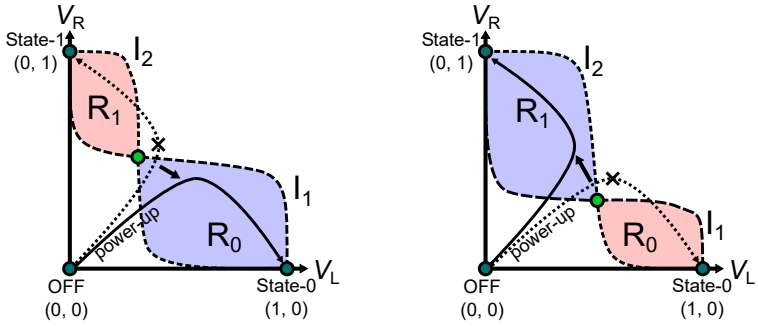


Figure 2.4: (a) Schematic of a conventional six-transistor (6T) SRAM cell and (b) The SRAM cell voltage transfer curve.

As illustrated in the voltage transfer curve, the state is *OFF* while the SRAM is not powered, i.e. both two nodes are at zero volt. Once the SRAM is connected to power, the voltage of the two internal nodes will simultaneously rise, and depending on the contribution of noise, the slope of the power-up curve can deviated for different cases, and will end up in either state-0 or state-1, depending on the trajectory first entering R_0 or R_1. The resulting state is called the *power-up state* of an SRAM.

In a typical SRAM cell, the dimension of the two inverters are designed to be identical, resulting in identical pull-up and pull-down strengths. The implemented SRAM cells are, however, not identical due to the mismatches, and hence the two inverters will have different strength, resulting in voltage transfer curves as illustrated in Figure 2.5. With the same initial trajectories as in Figure 2.4 (b), depending on how the transfer curves are skewed, the two cases can end up in the same final state. For the case in Figure 2.5(a), the NMOS of I_1 is stronger, which skews the transfer curve to the left and leads to the state-0 for both cases. On the other hand, in the case where the NMOS of I_2 is stronger, both cases will result in the state-1 instead. It should be noted here that we have only considered the process variation of a single transistor, while the actual case will be a mixed-up of variations in all the transistors. In summary, for any particular SRAM cell manufactured, there will be a unique voltage transfer curve depending on the mismatch between the two inverters. The skewed transfer curve will give a preference for the power-up state to be either state-0 or state-1.



(a) Power-up to state-0 **(b)** Power-up to state-1

Figure 2.5: Illustration of how the skew of transfer curve results in different power-up states.

Data stability of SRAM PUFs

As mentioned earlier, the output of PUFs can be unstable, and the SRAM PUF is not an exception. Considering an SRAM cell with negligibly small process variations, resulting in a voltage transfer characteristic of Figure 2.4 (b). In this case, there will be no preferred power-up state for this particular SRAM cell, hence, the generated PUF bits will be unstable with about 50% of "0"s and 50% of "1"s. Moreover, for the SRAM cells with the characteristics shown in Figure 2.5, the generated PUF bits will be more stable, but they can still

end-up in the non-preferred state if the power-up process is too much affected by noise. For different designs and technologies, the instability of SRAM PUF data in terms of bit-error-rate (BER) can vary from a few percent to more than 10%, as shown in prior work [18, 40, 41].

2.4.2 Other bi-stable PUFs

Following a similar concept as the SRAM PUF, there are several PUF implementations available based on circuit elements also with two stable states– the bi-stable circuits. These designs include the DFF PUFs, latch PUFs, sense-amplifier (SA) PUFs, and scan-chain PUFs [7, 59]. These designs are usually proposed while aiming for better controllability or possibly better data stability, since the SRAM cells are usually technology-optimized blocks provided by the foundries with less design freedom. These designs also utilize the mismatch of a crossed-coupled pair to result in different power-up states.

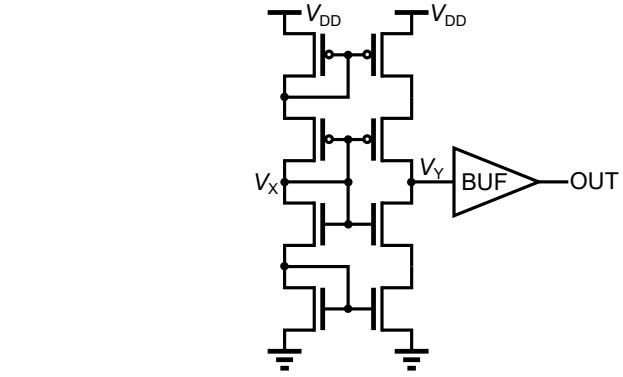
2.4.3 Mono-stable PUFs

As the data of the bi-stable PUFs will inevitably exhibit a period which is highly sensitive to noise, while the operating point is close to the metastable state, another type of PUF is proposed to avoid this issue and hence is expected to have better data stability. One approach is the inverter PUF proposed in [3] and refined design was proposed in [81]. The concept of this circuit is to amplify the relatively small voltage shift induced by process variations, without incorporating a positive feedback scheme.

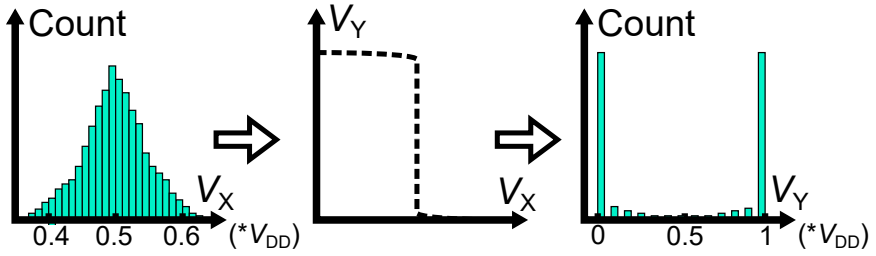
With the structure shown in Figure 2.6 (a), the input voltage to the inverter is generated using a replica biasing scheme. The bias voltage V_{X} is designed as $0.5V_{\mathrm{DD}}$, but the imbalance between the NMOS transistors and the PMOS transistors introduces voltage shifts as illustrated in Figure 2.6(b). By passing through an inverter with such high gain, i.e., has a very steep transition in the voltage transfer curve, the output V_{Y} will be close to a digital "1" or a digital "0". Consequently, the Gaussian-like input voltage distribution will be transformed into a binary-like output distribution.

The input-output relation is fully described by a single voltage transfer curve-there will be only one stable state for a particular PUF cell- and is hence categorized as a mono-stable PUF. Since there is no metastable state in such designs, the native stability is in general reported to be better than the bi-stable PUFs. However, given the fact that the slope of the voltage transfer curve cannot be infinitely steep, and the V_{X} will have values close to $0.5V_{\mathrm{DD}}$, there

will always be some PUF cells that are less stable. Moreover, a drawback for this type of PUFs is that the data can still change from cycle to cycle even if the power stays on, while the data in a bi-stable PUF will be stored until the power is turned-off.



(a) Schematic



(b) Operating concept

Figure 2.6: The schematic and operating concept of the inverter PUF.

2.4.4 Arbiter PUF

As an early proposal for PUF implementations [54], the arbiter PUF is a delay-based strong PUF, in which the number of CRPs is exponentially proportional to the number of delay stages. As shown in Figure 2.7, the main body of an arbiter PUF consists of cascading stages of configurable delay elements, which are usually formed by two multiplexers. The last stage of the circuit is an arbiter, which will determine whether the upper signal or the lower signal arrives first and output "0" or "1" accordingly.

Depending on the control bit provided to a certain delay element, the timing signals can either stay on the original paths or swap paths. Since all the

segments of the signal paths can be affected by the process variation, they will all provide different delays to the timing signals. By controlling the delay elements through providing challenges, different combinations of delays will be created. The timing signals will then be affected by the delays and results in different timing differences for all challenges. Finally, these timing differences will be translated into "0"s and "1"s as the responses to the challenges.

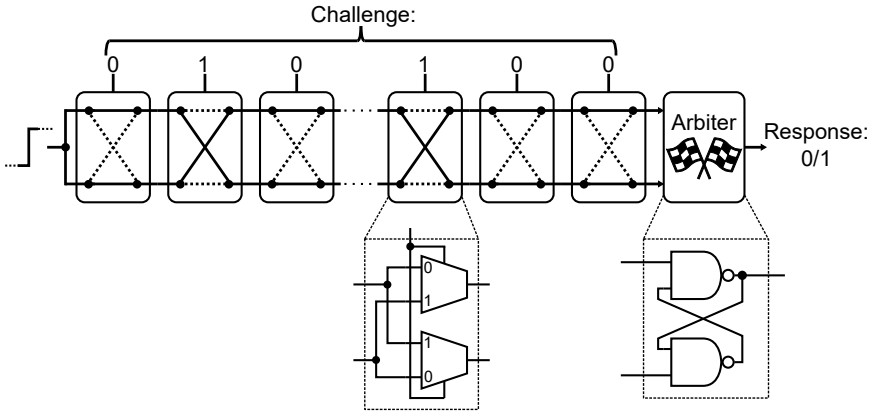


Figure 2.7: Schematic of a basic arbiter PUF.

For an arbiter PUF with N stages, there are 2^N possible N-bit challenges, and it therefore has 2^N CRPs, which meets the criteria for a strong PUF. As the concept is to have linear combinations of delays, there will be strong correlations between CRPs which makes modeling, e.g., by machine learning, fairly feasible for attacking the arbiter PUFs [27].

2.5 Data stability and stabilization techniques

As mentioned earlier, the key generation scheme described in Figure 2.2 typically requires error-correction to eliminate the bit-errors present in the generated PUF data. In order to reduce or even eliminate the need for error-correction logic and the helper data, it is an essential task to make the PUF data more stable. There are PUF designs aiming for a better native stability, and there are also techniques to improve the data stability without the need of a relatively heavy error-correction scheme. Here we will introduce three of the mainstream stabilization techniques: temporal majority voting (TMV), dark-bit masking and burn-in enhancement.

2.5.1 Temporal majority voting

The concept of temporal majority voting is to generate the output bit of a PUF cell multiple times, and choose the one that occurs most frequently as the final output. As illustrated in Figure 2.8, a PUF response including some unstable bits are evaluated N times and the bits which occurs more than $N/2$ times are chosen as the composition of the final output. In principle, the probability of having more incorrect readouts than the correct ones will be lower than the error probability, and this probability reduces with the number of readouts.

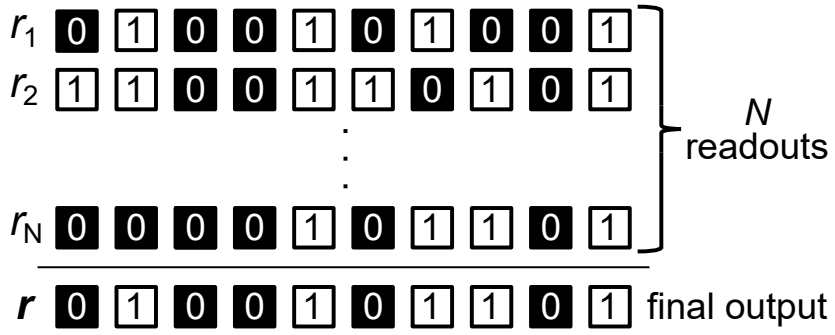
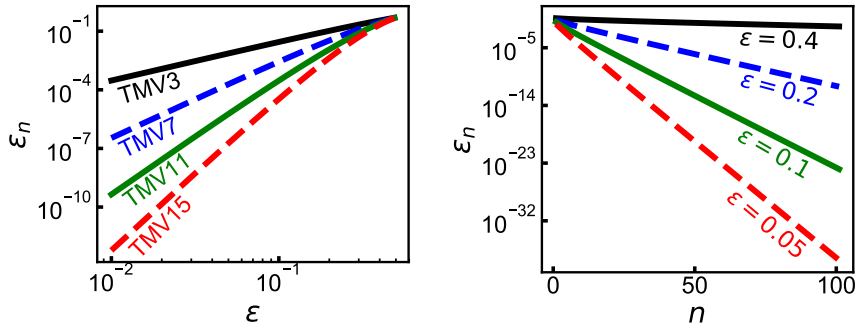


Figure 2.8: Illustration of the TMV algorithm, the final output r in this case is based on an example $N = 3$.

Quantitatively, assuming that a PUF cell has an error probability ϵ, which is the probability of producing an error in one readout. Each readout event can be defining a random variable X with outcome $x \in \{0, 1\}$, where 0 represents a correct readout and 1 represents an error, i.e. $\Pr(X = 1) = \epsilon$. For a TMV algorithm of $N = n$, the number of errors existing in n readouts can be defined as another random variable $Y = \sum_1^n X_i$, where X_i represents each individual readout event, and the events are independent and identically distributed (i.i.d.).

With these definitions, the error probability after being processed by TMV of $N = n$, denoted as ϵ_n, can be derived as $\epsilon_n = \Pr(Y > n/2)$. While Xs are Bernoulli trials with probability ϵ, Y is actually a binomial distribution with number n and probability ϵ. Deriving from the probability mass function of the binomial distribution, the resulting error probability can be computed as Equation 2.6. Where $\overline{n/2}$ represents the first integer which is greater than $n/2$, while n is usually an odd number in a majority voting scheme.

$$\epsilon_n = \sum_{k=\overline{n/2}}^{n} \binom{n}{k} \epsilon^k (1-\epsilon)^{n-k} \tag{2.6}$$



(a) ϵ v.s. ϵ_n **(b)** n v.s. ϵ_n

Figure 2.9: The simulated error probability ϵ_n using different number of TMV (n) and with different initial error probability ϵ.

By computing the resulting error probability under different conditions, as plotted in Figure 2.9, which shows how the final error probability scales with the number of TMV and the initial error probability. As shown in Figure 2.9 (a), all the curves of ϵ_n converge at $\epsilon = 0.5$ and $\epsilon_n = 0.5$, which shows that the TMV will not work with the highest error probability of 0.5. Figure 2.9 (b) shows that ϵ_n is reduced much more efficiently with lower initial error probability, indicating that TMV is not an ideal solution for the bits which have higher error probability since it will require too many readouts to generate one bit. It should be noted that the overall BER cannot be directly used to estimate the required n, since the overall error rate is composed of some bits with higher and lower error probability. For example, a PUF with overall BER=10% may have a bit-cell that has error probability higher than 0.4, and by using TMV11 may not sufficiently reduce the bit errors from this particular bit-cell.

2.5.2 Dark-bit masking

The concept of dark-bit masking is to first identify the bit-cells that are unstable, and these bits will be marked as the dark-bits. For the PUF readouts performed after the dark-bits are marked, the dark-bits will be excluded from the final output, as illustrated in Figure 2.10. By applying this scheme, the masked bit-cells will no longer contribute errors to the PUF data and, ideally, the resulting data will be 100% stable.

Despite masking can in theory eliminate all instability, its performance is still limited by practical constraints. First, it is not easy to identify all the unstable bits, in particular for the ones with relatively low error probabilities. For

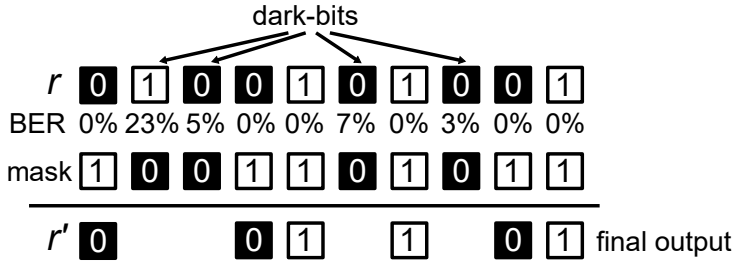


Figure 2.10: Illustration of the dark-bit masking algorithm.

example, for a bit cell with error probability of 1%, it can likely take more than 100 readouts to observe the first error, and yet we do not want to miss it on the list.

It has been proposed in [55] that using the remanance effect of SRAM circuits, the unstable bit-cells in a SRAM PUF can be accurately identified and resulting in completely stable PUF readouts. This is no doubt a right direction towards ideal data stability, but it still cannot surpass the next limitation.

As an auxiliary input for PUF readouts, the information about the location of these dark-bits needs to be provided during PUF readout and has to be stored on the chip. Since there is no better way than storing these information in an embedded NVM, this method is therefore commonly used in most of the PUF work. If we consider eliminating the needs for NVM as the major benefit from having fully stable PUFs, the dark-bit masking scheme which requires NVM will, therefore, not be the best choice as a stabilization technique.

It may look helpful by generating the dark-bit mask on-the-fly, i.e., first identify the unstable bits and then start reading each time, which is, however, infeasible in practice. The reason is not only that the identification time is too long, but also that always identifying exactly the same set of bit-cells is impossible. If the remaining bit-cells are not always the same, it will introduce another form of instability in the PUF data.

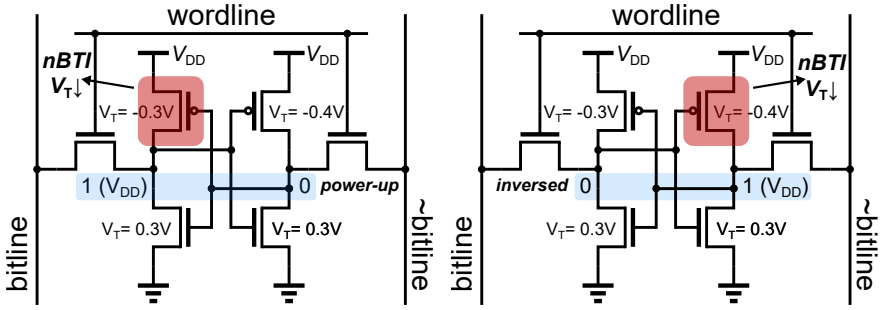
TMV and Dark-bit masking

The dark-bit masking works better for the bits that are more unstable, since they will be more easily identified. In the previous subsection, we also found that TMV works better for the bits that are less unstable. By these two facts, it is quite straightforward to combine these two methods, i.e., using the dark-bit masking to screen out the bit-cells with high error probability, and using TMV

to lower down the error probability of the rests. In the prior work aiming for better data stability, these two methods are indeed simultaneously used in most of the cases.

2.5.3 Burn-in enhancement

The third method for improving data stability is based on the long-term degradation mechanisms of devices, which may downgrade the circuit performance or even cause fatal failures. The concept of this method is to intentionally apply certain aging effects to the PUF circuit, which can be seen as providing additional *time-dependent variability* on top of the pre-existing time-zero variability induced by processing steps. Once the time-dependent variability is generated in the intended direction, it can make the mismatches within cells larger and can, hence, generate more stable PUF data.



(a) Stress condition – power-on **(b)** Stress condition – opposite

Figure 2.11: A simplified illustration of the nBTI stress conditions in an SRAM cell, when storing the power-up state or the opposite.

One example is to exploit the BTI degradation in the SRAM PUFs. As discussed in [60], if an SRAM PUF keeps storing the power-up state in the cells, the transistors will exhibit the stress condition as depicted in Figure 2.11 (a). In this case the PMOS with a higher threshold voltage (V_T), will see the nBTI stress and its V_T will decrease gradually while the same stress condition remains. The decreasing of V_T in this case is bad for the data stability of this PUF cell, since the mismatch between the two inverters is decreased, which makes it more sensitive to noise.

On the other hand, a solution to counter such effect was also proposed in [60]. The method is to inversely store the power-up states by re-programming the entire SRAM PUF array, and for the same example SRAM, the updated stress

condition is shown in Figure 2.11 (b). In this case, the PMOS with lower V_T is under nBTI stress, which will increase the mismatch between these two PMOS transistors and, hence, results in better data stability. It should be noted that it requires long time for the BTI stress to be effective under normal operating conditions, and thus, this burn-in enhancement method needs to be accelerated by applying higher voltage and temperature, which still takes several hours to days to achieve a sufficiently good result.

Besides this method being suitable for most of the SRAM PUFs, there are also burn-in enhancement methods proposed that are dedicated for certain PUF implementations. One method is proposed in [7], using a sense-amplifier (SA) PUF with additional burn-in function exploiting the HCI mechanism. Another example is the hybrid PUF proposed in [62, 76], that also uses BTI for burn-in enhancement.

The PUF implementations utilizing this technique have shown almost ideal data stability or at least significant improvements on their data stability. Besides the successful results, there are still some disadvantages of the burn-in method. The most obvious one is the time consumption, as exemplified in [60]. Moreover, both the BTI and HCI stresses are recoverable to an extent, making the retention of the burn-in effect another issue yet to be solved.

2.6 New concept– active PUFs

So far, the stabilization techniques introduced are still some distance away from being an ideal solution. In order to fundamentally solve the issues of instability, a new type of PUFs has been proposed. These PUFs utilize mainly the time-dependent variability as the entropy source rather than the conventional time-zero variability, i.e., the process variation. By using this concept, the PUF behavior is *actively* given by performing specific operations, and hence, they can be categorized as *active PUFs*. In contrast, the conventional PUFs, such as the SRAM PUFs, in which the PUF behavior is passively given by the processing steps (not the circuit itself), will be referred to as *passive PUFs*.

The definition for an active-PUF implementation is stated as follows:

Definition 2.7. *An active-PUF implementation is a circuit implementation consisting of a circuit design and a various amount of identically fabricated circuit instances, in which the circuit instances have the required PUF properties only after an "activation" step. The activation step should only be performed after the circuit has been fabricated.*

2.6.1 Overcome the limitation of process variations

The main reason for introducing active PUFs is to achieve good native data stability, which is not easy to obtain for passive PUFs. The origin of this difference is the yield optimization for all mature silicon technologies. Since the passive PUFs relies on the process variations, the PUFs would be more robust if the variations are more severe. This preference, however, contradicts the typical yield requirements, since variability is considered as the main obstacle for achieving a good yield.

The main advantage of the active PUFs is no doubt the good data stability, it can be achieved because the time-dependent variability can have much more significant impact than the process variations. Specifically, device aging can cause fatal damage to a circuit, but the process variation mostly can only degrade circuit performance. One reason of this difference is that a silicon technology is usually optimized by the foundry, in order to improve the yield, i.e., the remaining process variations usually will not cause fatal errors that reduce the yield.

To the best of our knowledge, there are now two major types of active PUFs in the field. The first one is based on the gate oxide breakdown mechanism of MOSFETs, as proposed in [13, 16, 56, 88]. This concept is also exploited to make the *antifuse* one time programmable memory (OTP), and these PUFs are therefore sometimes referred to as the antifuse-based PUFs. The second type of active PUF is based on the resistive-switching type of memory devices, such as metal oxide resistive-RAM (OxRAM) [57, 69, 89], phase change memory (PCM) [94] and magnetic-RAM (MRAM) [20, 92]. Since we will only study OxRAM in this thesis, it will be referred as RRAM from now on.

2.6.2 Oxide Breakdown PUFs

The gate oxide breakdown failure has been exploited to realize antifuse OTP memory [45]. Using oxide antifuses to store data has the advantages of excellent data retention and low cost, since oxide breakdown creates near-permanent damages in the transistors and it only requires normal CMOS transistors, i.e., it does not require additional processes for features like floating gates.

Considering these advantages, oxide breakdown is an ideal candidate for implementing active PUFs. The first known solution is proposed in 2010 [56], which has already demonstrated a native BER of 0% at that time. For this prior work, the PUF generating procedure is not yet optimized, since it in theory has a small amount of inevitable entropy loss.

Few more similar solutions are proposed recently, including our work [13,16] using soft oxide breakdown, antifuse-based PUF in [88] using hard oxide breakdown and a 2-bit/cell PUF using oxide breakdown mechanisms in FinFETs [31]. All these implementations are featuring nearly ideal data stability and have different advantages and disadvantages. In this thesis, we will discuss the concepts, implementation details and the experimental results of our proposed soft-BD PUF, in Chapter 3 and Chapter 4.

2.6.3 RRAM PUF

RRAM is a popular type of emerging memory technologies, which aims at replacing DRAM or storage type memory. A RRAM cell stores binary data in the form of resistive states, which are non-volatile and can be programmed multiple times. One of the issues that draws a lot of attention is the switching instability of RRAMs. As a requirement to be CMOS compatible, the programming voltage for RRAMs should be close to the operating voltage of core CMOS circuits. While using lower programming voltages, the RRAM cells cannot always switch to similar resistance under the same programming condition. The switching instability may result in a wide resistance distribution or even cause switching failure, i.e., a RRAM cell is not correctly set to the target resistive state.

While being a non-desired phenomenon for memory engineers, the switching instability can be well exploited to implement RRAM-based PUFs. The basic concept of this solution is to apply the same programming condition to all the RRAM cells, and the resulting resistance differences are used as the entropy source of PUFs. As an advantage of being multi-time programmable, the resistance of each RRAM cell can be randomly modified through re-programming, which gives RRAM PUF the feasibility of being reconfigurable. More details about how to reconfigure RRAM PUFs and the reason of non-ideal reconfigurability will be discussed in Chapter 5 of this thesis.

2.7 Conclusion

In this chapter, we have made an overview of the PUFs in ICs. The weak PUFs and strong PUFs can be utilized for cryptographic key generation and entity authentication, respectively. The PUFs need to be unique for every chip and the PUF data has to be stable to maintain good security functions. We have also shown that for conventional PUF implementations, it is difficult to eliminate instability without using complex error correction schemes. There are also several stabilization techniques that can be applied but each of them has

certain drawbacks. Consequently, it brings us to the main concept that will be discussed in this thesis, the *active PUFs*, that are designed to achieve near-ideal data stability. Our work on designing the soft-BD PUF and the analysis of reconfigurable RRAM PUF will be introduced in the following chapters of this thesis.

Chapter 3

Device level characterization of soft oxide breakdown PUF

Content Sources

The main material in this chapter was previously published in:

K.-H. Chuang, E. Bury, R. Degraeve, B. Kaczer, G. Groeseneken, I. Verbauwhede and D. Linten, "Physically unclonable function using cmos breakdown position," in *2017 IEEE International Reliability Physics Symposium (IRPS)*, April, 2017, pp. 4C–1.1–4C–1.7.

K.-H. Chuang, E. Bury, R. Degraeve, B. Kaczer, T. Kallstenius, G. Groeseneken, I. Verbauwhede and D. Linten, "A multi-bit/cell PUF using analog breakdown positions in CMOS," in *2018 IEEE International Reliability Physics Symposium (IRPS)*, March, 2018, pp. P-CR.2-1-P-CR.2-5.

Contribution: Main author.

The gate oxide soft breakdown mechanism has been chosen to implement the first active PUF design in this thesis. The proposed three-transistor unit PUF cell featuring a self-limiting breakdown mechanism ensures the existence of only one soft oxide breakdown spot within a cell. The binary current characteristic shown by the experiments implies a good data stability of the proposed PUF. Finally, by comparing with an alternative PUF design based on a two transistor

unit cell, which has a non-binary current profile, it is confirmed that generating binarized breakdown is a better strategy.

3.1 Introduction

As discussed in the previous chapter, the new design concept of active PUF opens a path towards the ideal data stability. In this chapter, we introduce a novel PUF circuit utilizing one well-studied reliability phenomenon, i.e. the gate oxide breakdown (BD) location, as the source of entropy. Oxide breakdown is a detrimental failure mode found in MOSFET devices, cannot be recovered by normal circuit operations. It is therefore non-desired in most cases. This irreversible property can, however, be exploited to significantly improve the data stability of PUFs. In this chapter, a PUF based on soft gate oxide breakdown will be introduced, with a focus on device level characterization of the so-called "soft-BD PUF" implementation.

3.1.1 Concept of gate oxide breakdown

Gate oxide breakdown is a time-dependent failure mode of the gate oxide of MOSFET devices. There are two main properties making it exceptionally suitable for designing a robust PUF solution. The first property is that the breakdown is irreversible by normal circuit operations, which implies a good stability in the first place. Secondly, the time dependence of the breakdown mechanism can be intensively accelerated by increasing the applied stress voltage, making the procedure less time consuming comparing to the burn-in methods discussed in the previous chapter.

The gate oxide in traditional silicon technology consists of an insulating thin layer made of SiO_2 or high-κ with interfacial SiO_2. During normal operations, the gate oxide allows leakage current of sub-nA level. When the device is heavily used, and the gate oxide is constantly stressed with high voltage, more oxide traps will be generated as illustrated in Figure 3.1. After a certain stress time, a part of the generated oxide traps and the pre-existing oxide traps may form a percolation path that assists the electron/holes to conduct through the oxide layer. Once a percolation path is formed, a significant current increase can be observed, which is identified as the soft-BD event. Along with the increased gate current, the growing power dissipation will heat up the soft-BD spot and further speed-up the trap generation process. The two effects create a positive feedback, causing the BD spot to grow, and eventually collapse to a catastrophic failure, which is identified as a hard-BD event.

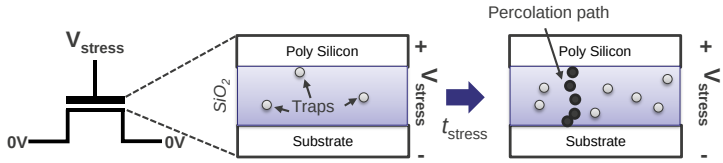


Figure 3.1: An illustration of a gate oxide breakdown event.

3.1.2 Randomness from BD positions

As illustrated in Figure 3.1, because the creation of oxide defects is uniform throughout the oxide, the location of this percolation path is unpredictable before the stress starts. If the stress condition is well controlled, such that only one percolation path will be formed, the location of the breakdown spot can be exploited as a static entropy source for PUF implementations. Note that the graph only shows one dimension cross-section view, but the actual BD position is in two dimensions. The determination procedure for the two dimensional BD position is discussed in [2], but will not be used in this work.

Analog BD position from a single transistor configuration

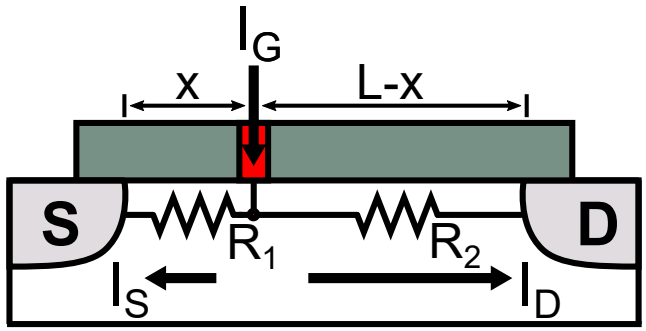


Figure 3.2: An illustration of the relation of the breakdown position and the current characteristic.

Considering the device cross-section as shown in Figure 3.2, in this lateral dimension, the BD position can happen at any place between source and drain. The series resistance in the conductive channel in inversion condition is related to the BD position x as $R_1 \propto x$ and $R_2 \propto (L - x)$. The current collected by the source and drain can be derived by the following equations.

$$I_{\mathrm{S}} = I_{\mathrm{G}} \cdot \frac{R_2}{R_1 + R_2} = I_{\mathrm{G}} \cdot \frac{L - x}{L} \tag{3.1}$$

$$I_{\mathrm{D}} = I_{\mathrm{G}} \cdot \frac{R_1}{R_1 + R_2} = I_{\mathrm{G}} \cdot \frac{x}{L} \tag{3.2}$$

By measuring the values of I_{S} and I_{D}, the breakdown position x can be computed based on the following equation, as derived from Equation 3.2.

$$x = L \cdot \frac{I_{\mathrm{D}}}{I_{\mathrm{G}}} = L \cdot \frac{I_{\mathrm{D}}}{I_{\mathrm{S}} + I_{\mathrm{D}}} \tag{3.3}$$

It is also common to normalize the BD position by the channel length L and resulting in the Equation 3.4, which describes the *relative* BD positions along the channel. The extracted position based on this equation will be an analog value ranging from 0 to 1, and is therefore given the name of "analog BD position".

$$\text{Extracted position} = \frac{I_{\mathrm{D}}}{I_{\mathrm{S}} + I_{\mathrm{D}}} \tag{3.4}$$

It should be noted that this analysis is a circuit level abstraction describing the current ratio technique for BD location determination, as discussed in [19]. There are more details about physical understanding and modeling in these references, which will not be covered in this thesis.

Binary BD position from the duo-transistor configuration

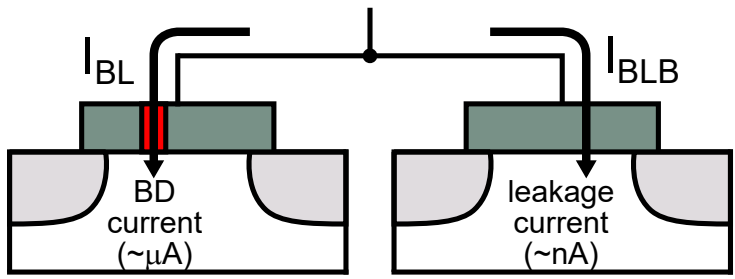


Figure 3.3: Illustration of the current profile for a pair of transistors with only one gate oxide breakdown spot.

In order to generate applicable data for PUF usage, the extracted BD position has to be converted into digital bits. Theoretically, different number of bits can be generated from one PUF cell depending on the number of quantization levels, e.g. three quantization levels may resulting in two bits. In reality, this conversion is not a trivial task for a circuit block implemented on-chip. Moreover, since there are no margins between quantization levels, the readout of BD position is sensitive to noise.

Consequently, using the BD position within one transistor is not a good choice when designing for better data stability. In order to overcome this drawback, a more robust solution is proposed by adding another transistor in parallel, providing a binary nature for the BD positions, as illustrated in Figure 3.3. In this duo-transistor concept, the BD spot will be located in one of the transistors, resulting in the so-called "binary BD position", *regardless* of the lateral BD position within that transistor. Unlike the current profile for the case of a single transistor, there will be a significant difference between the two current components, I_{BL} and I_{BLB}, and hence the duo-transistor concept will be less sensitive to noise.

In this chapter, we will be more focusing on the design and characterization of the PUF devices using the concept of binary BD. For completeness, the PUF circuit using the aforementioned analog BD positions will be also tested and compared, to proof that the binary BD position is a better choice.

3.1.3 Soft breakdown v.s. Hard breakdown

In the proposed PUF circuit, a soft breakdown event is more preferred than a hard breakdown event, resulting from the underlying difference in terms of security. In contrast to the targeted soft-BD, a hard-BD of a gate oxide corresponds to a more catastrophic failure of that device. It is the foundation of anti-fuse memory and the PUFs derived from such technology. As shown in [88], the anti-fuse based PUF has an excellent robustness against all kinds of variations, as it takes the advantage of the large resistance difference between programmed and unprogrammed devices.

The reason for using soft-BD- or to be more general, oxide breakdown with reduced hardness- is to prevent possible threats from invasive attacks, in particular, visualization attacks using different types of microscopes. It has been shown in [70] that a soft-BD spot is less visible than a hard-BD spot, since the damage created within the device is smaller. It should be noted that the results shown in this reference are obtained by transmission electron microscopy (TEM), which is not a feasible tool for attacking a PUF, due to the price of the tool and the preparation of the samples. It has also been shown in [12]

that the anti-fuse OTP memory is *not* visually attackable by the scanning electron microscopy (SEM). Despite there is no evidence showing that the data stored in an anti-fuse memory is vulnerable to such attacks, using soft-BD as an alternative is still attractive. Since soft-BD will be less observable, it should therefore forestall the rapid advancement of attack techniques.

3.1.4 Chapter organization

The chapter is organized as following. In Section 3.2, the design and basic properties of the 3T soft-BD PUF cell will be introduced. Section 3.3 will shown the experimental results of the 3T soft BD-PUF. Quantitative stability analysis based on the experimental results will be given in Section 3.4. In Section 3.5, the experimental results of the 2T soft-BD PUF with analog BD positions will be shown, and a comprehensive comparison with the binary BD position will be made. Finally, Section 3.6 will conclude this chapter.

3.2 3T soft-BD PUF cell

The unit cell of soft-BD PUF has been first introduced in [13]. The three transistor (3T) structure consists of two minimum-sized NMOS transistors and one PMOS transistor. The concept of this PUF design is to generate exactly *one* soft-BD spot in *one* of the two NMOS transistors in an indeterministic way, hence generating randomness. As illustrated in Figure 3.5, a "0" or a "1"-bit can be extracted based on the location of the soft-BD spot. This property is ensured by the *self-limiting* nature of this cell, and the readout of the PUF is based on the post-BD current characteristics.

3.2.1 Self-limiting BD generation process

The PUF cell is subject to a high voltage stress for generating a soft-BD spot, which is also referred to as the *forming* process. During forming, the on-state PMOS selector feeds the high voltage to the gate oxide of the two NMOS transitors, as shown in Figure 3.4. In the beginning, since the gate oxide has very high resistance, which is typically in the giga-Ω range, the stress voltage is entirely applied across the gate oxide of both NMOS transistors. As soon as a breakdown occurs in one of the two NMOS transistors, the current flow through the BD spot will induce a voltage drop (V_{DS}) on the PMOS selector, which reduces the stress voltage accross the gate oxides. Consequently, no further 2_{nd} BD spot can be generated.

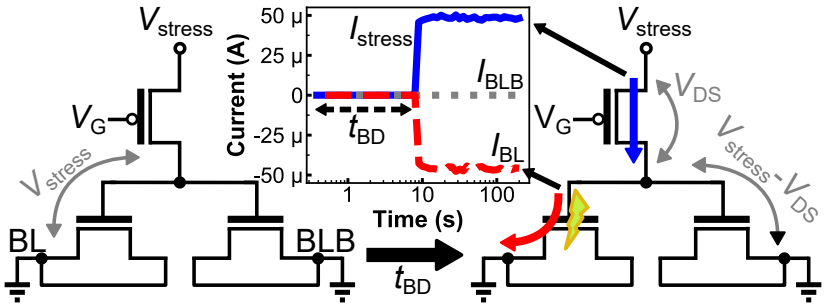


Figure 3.4: Illustration of the bias condition for the forming process and the subsequent transient behavior. The self-limiting property is enabled by the PMOS selector and is activated after a BD occurs, as illustrated in the right-hand sided schematic. The transient current waveform is shown in the inset plot. Note that the current value for BLs is negative to indicated that the current is flowing into the measurement equipment. For this particular experiment, V_{stress} is 3.8V, which results in a long breakdown time of $\sim 10s$ for a clearer observation purpose.

Note that the current level during breakdown is essential for the "hardness" of the BD spot, and it can be controlled by the saturation current of the PMOS selector, which is designed to be around 50µA in this case. By both the voltage-limiting and current-limiting breakdown mechanism, it is assured that there will be only one "soft"-BD spot within each PUF cell.

3.2.2 Test chip implementation

The test structure for the first experiments of BD-PUF was designed as a simple array with shared word-lines and bit-lines. There is no additional control circuit included, in order to ensure the purity of the measurement data, i.e., the current characteristics are solely determined by the PUF cell. As shown in Figure 3.5, the PUF array consists of 5 columns and 12 rows, so in total 60 PUF cells. Each word-line and bit-line is directly connected to a probing pad, allowing direct voltage control and current sensing on these lines. By tuning the voltages applied to the PUF array, all the PUF cells can be measured individually, and can also be stressed simultaneously or separately.

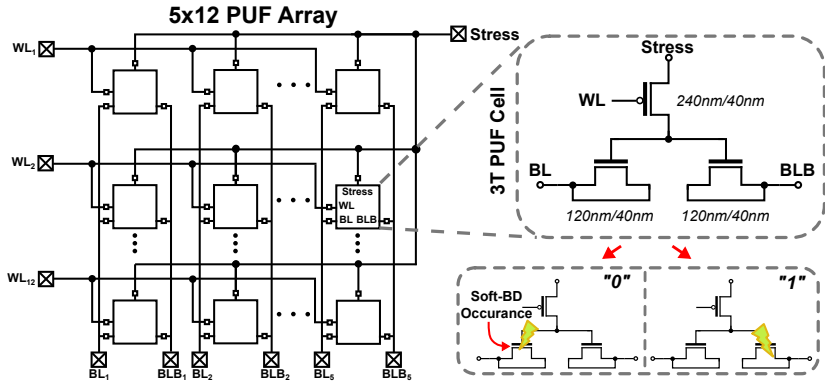


Figure 3.5: The 5x12 PUF array fabricated in a 40nm CMOS process, the dimensions of each transistors are listed. Note that 120nm and 40nm are the minimum width and length for this technology.

3.3 Experimental results

The test chips were fabricated in a commercial 40nm CMOS technology. Extensive experiments have been done using these test chips to demonstrate the quality of the soft-BD PUF.

3.3.1 Forming analysis

As all the PUF cells must be subjected to the *"Forming"* process before having the actual PUF properties, it is important to first study the feasibility of the forming process. In particular there are two main requirements to be fulfilled. First, the forming process needs to be reliable, i.e., only the targeted PUF cells should be affected by the voltage stress, the rest of the circuit must stay intact. Second, the forming time, i.e. the time to generate BD spots on all of the PUF cells, needs to be sufficiently low, in order to make the cost for the additional testing or setup time acceptable.

Following the well-known methods for analyzing TDDB [43, 86, 87], the breakdown event under different stress voltages has been measured and the statistics of the time to breakdown (t_{BD}) are extracted for analysis. Since t_{BD} is known to follow the Weibull distribution, the measured t_{BD} of 300 PUF devices under 5 different voltages are plotted in a Weibull plot, as shown in Figure 3.6. In this figure, the cumulative distribution function (CDF) of t_{BD} is

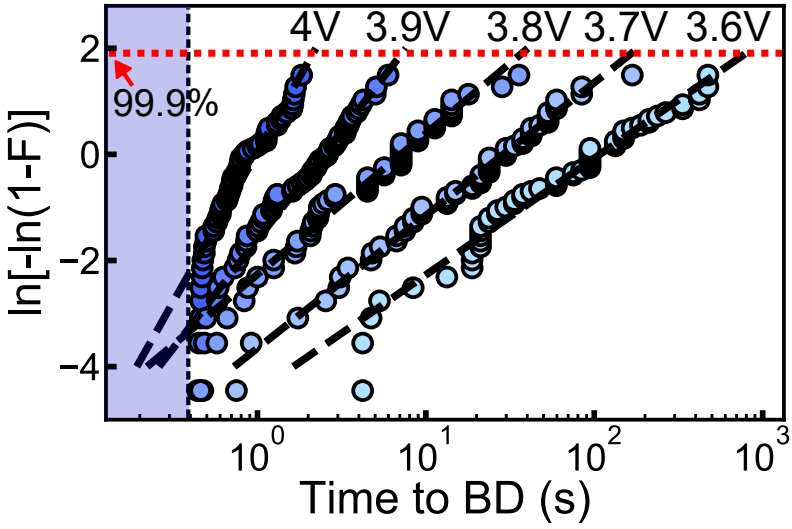


Figure 3.6: Time-to-breakdown distribution for single BD-PUF cell under different stress voltages in Weibull scale. The shaded region shows the region where the t_{BD} is not observable due to equipment limitation. The slope is varying for different stress conditions. The dark dashed lines show the maximum-likelihood fitting of Weibull distributions.

transformed into Weibull scale, by changing the values of the y-axis from F to $\mathrm{Ln}\left[-\mathrm{Ln}(1-F)\right]$, where F is the CDF of t_{BD}.

It can be clearly observed that the Weibull slope (β) [87] is varying with stress voltage, this phenomenon typically can be observed in a gate stack with multiple layers [68]. Since there is no public information about the gate oxide process in this commercial technology, the actual reason of observing the varying slopes cannot be studied in further details.

The main purpose of these experiments is to predict the required stress voltage to generate a BD spot in every PUF cell. As shown by the cumulative distribution function, it can be found that a certain percentage (y-axis) of devices will have a BD below a certain time (x-axis). As limited by the sample size, the required time for breaking most of the devices, e.g. 99.9% of devices, cannot be directly seen from the measurement result. An extrapolation is needed to find these numbers, i.e. the t_{BD} corresponding to the Weibit of 1.93 (99.9%). It can be seen that the left tail of these distributions shows a larger discrepancy from the fitting curves, but it does not affect the overall analysis since this side is out of our interests when aiming for a prediction at 99.9%.

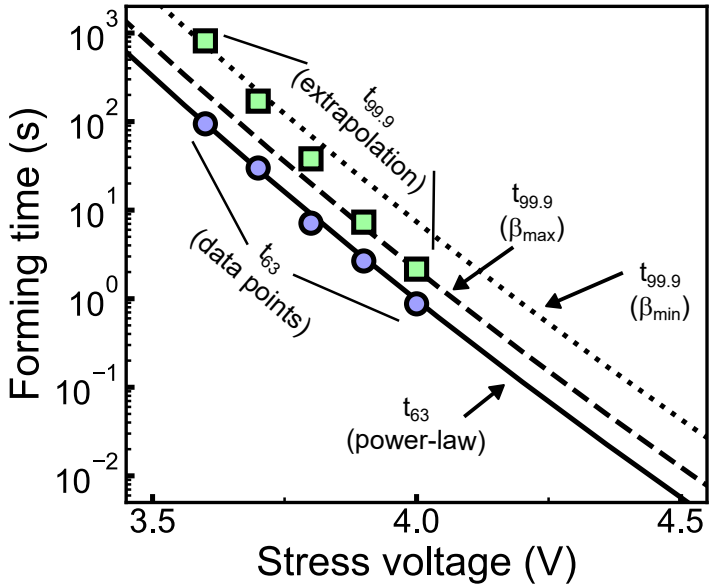


Figure 3.7: Approximated array forming time at CDF of 99.9%.

The voltage dependence of the t_{BD} is also an important factor for the forming process, which determines the voltage level that needs to be supplied in order to gain enough acceleration. For the conventional TDDB analysis, it can be done trivially by extrapolating all the distribution curves with the same β, and then check the voltage dependence of the extrapolated values. This method is, however, infeasible for the case of varying β. In order to find the correct parameters, the standard method is to fit with the bimodal distribution, but it requires more data points, which is not possible for the limited amount of devices.

As shown in Figure 3.7, the green square shows the 99.9% extrapolation of the t_{BD} distributions (t_{99}) under different stress voltages, which clearly shows a voltage dependence different from the 63% (zero weibit) t_{BD}, namely t_{63}. With the varying β, the power-law fitting cannot be applied to the resulting t_{99} values.

In order to have good insights on the relation between stress voltage and forming time, an estimation of the boundary conditions was done by making certain assumptions. We first assume that the voltage dependence of t_{63} follows power-law [86], and it can be therefore fitted as shown in this figure. Using this fitting curve, the t_{99} extrapolation based on the β of 4V (β_{max}) and 3.6V (β_{min}) are

also derived and plotted. Since β shows an increasing trend with voltage, for the extrapolation at higher voltages, the curve using $\beta_{\max}$ would provide adequate results, and conversely for the lower voltages, the curve using $\beta_{\min}$ should be used.

Based on the $\beta_{\max}$ curve, the forming time for an entire PUF array is expected to be around 10ms when using a stress voltage of 4.5V, which is not a typical voltage being used in the core circuits in this technology, but it is within the range of battery voltages and can also be provided by a conventional embedded high voltage generator. As a result, it is indeed feasible to perform this forming process on the PUF chips.

3.3.2 BD position extraction

As the BD position inside the channel in the source-drain direction can be determined using the current ratio technique, the same method can be applied to the 3T BD-PUF cell and it is expected to show a binary-shaped distribution function. It should be noted that when using on-chip readout circuits such as sense-amplifiers to identify the BD position of the 3T BD-PUF cell, the concept is to compare the absolute current difference, instead of using the current ratio technique. The reason of using current ratio throughout this chapter is to clearly show the binary behavior and also to help comparing with the 2T BD-PUF cell with analog BD characteristic.

For the case of a 3T BD-PUF cell, the current ratio follows the same formula but *does not represent the actual BD position* in the circuit. This extracted BD-position, as defined in Equation 3.5, shows how the BD position at the left and the right transistor can be distinguished.

$$\text{Extracted position (quantized)} = \frac{I_{\text{BLB}}}{I_{\text{BL}} + I_{\text{BLB}}} \tag{3.5}$$

The resulting positions are shown by the cumulative distribution functions (CDFs), as shown in Figure 3.8. A clear binary characteristic can be observed at all three different operating voltages, indicating that only one BD spot is occurring in one PUF cell. The PUF cells with the extracted position below 0.5 will generate a “0”-bit and the cells with positions greater than 0.5 will generate an “1”-bit. By observing the distributions, it can be seen that there is likely an equal probability of 0.5 for “0” and “1” bits, since the two populations are almost the same.

Between the current ratio distributions at different V_{DD}, there is a clear voltage dependence, in particular between V_{DD}=0.9V and V_{DD}=1.2V. The separation

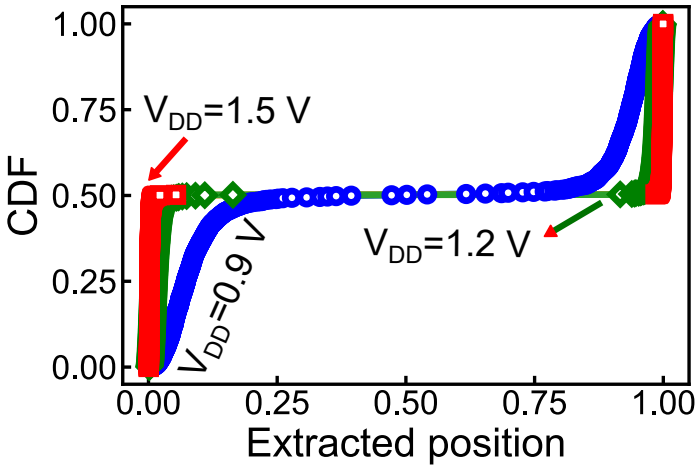


Figure 3.8: CDF of the extracted positions under three different VDD values using Equation 3.5. The result is close to binary for 1.2V and above, but the separation is less clear for V_{DD}=0.9V.

between the "0" and the "1" populations are less clear at V_{DD}=0.9V, showing that it will be more challenging to distinguish the two cases, and will potentially cause stability issues when the readout procedure is performed on-chip. This effect must be taken into account while trying to operate the soft-BD PUF at lower supply voltages.

To further investigate the origination of the decreased readout window at lower V_{DD}, the current characteristics are directly examined and discussed in the next subsection.

3.3.3 Post-BD current characteristic

To begin with the analysis, the current flowing through the gate oxide with and without breakdown are divided into two groups based on the current level measured at V_{DD}=1.5V, which has the most binarized current ratio distribution among all measurement conditions. The current flow through the broken oxide is denoted as the "BD" current (I_{BD}) and the current flow through the unbroken oxide is denoted as the "leakage" current (I_{leak}).

As shown in Figure 3.9, the breakdown current has a broad distribution and an exponential voltage dependence. The large current variation is due to the shape and size variation of the breakdown spots, and the exponential voltage

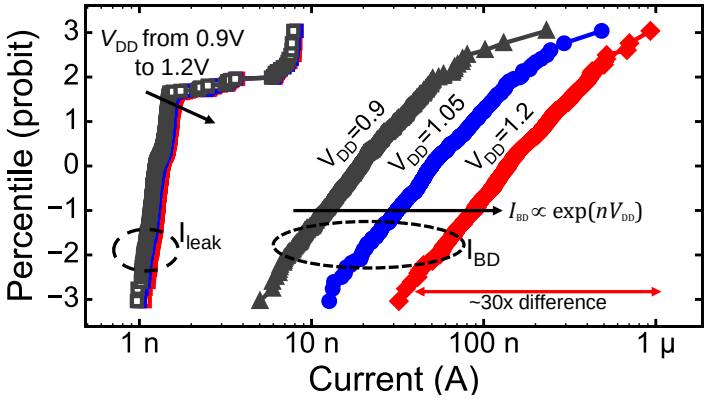


Figure 3.9: The experimental statistics of the two current components from the soft-BD PUF cells. The BD current (I_{BD}), which flows through the soft-BD spot is widely distributed and has an exponential voltage dependence, as described in the equation (inset), in which n is a scaling parameter. The leakage current (I_{leak}), which flows through the unbroken gate oxides, is also widely distributed but has no strong voltage dependence.

dependence originates from the conducting mechanism of soft oxide breakdown, which can be described as a quantum point contact (QPC) [65]. The leakage current results from trap-assisted tunneling (TAT) [25] and shows less voltage dependence.

By taking a closer look at the distributions of the leakage current, as shown in Figure 3.10, there are three different modes found in the distribution. These modes may represent the number of defects which are effective for the TAT conduction mechanism. According to [25], these oxide defects can be induced by the voltage stress in the forming process, suggesting that the gate oxides without soft-BD are also damaged, and hence these transistors can no longer be seen as fresh devices.

3.3.4 Thermal stability

For a thorough understanding of the data stability of PUFs, there are several aspects that need to be considered, in particular the stability with respect to time (readout cycle), voltage, and temperature. The time aspect can only be verified with a valid readout circuit, which will be introduced in the next chapter. The impact of voltage variation has been demonstrated in the previous two

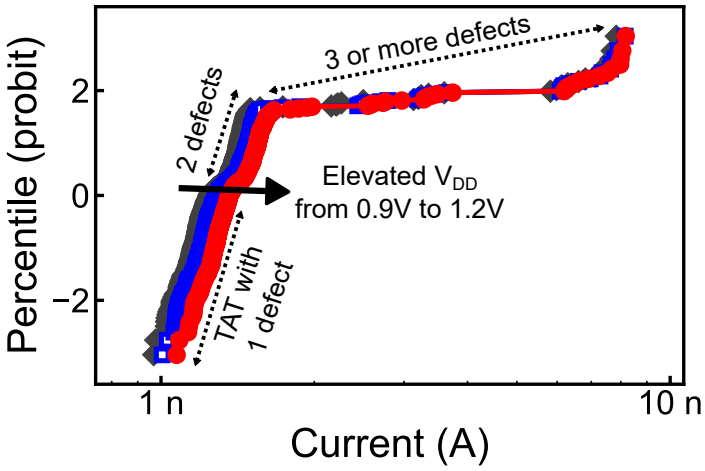
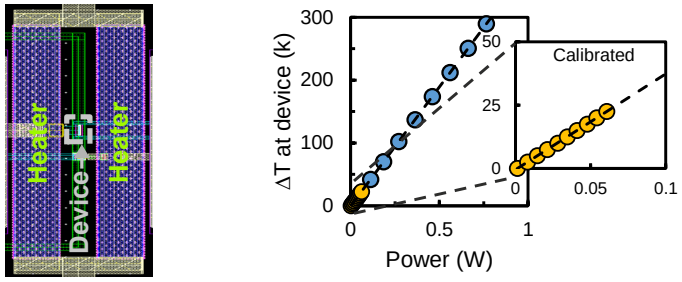


Figure 3.10: The distribution of the leakage current with three different V_{DD} values, from left to right are 0.9V, 1.05V and 1.2V.



(a) **(b)**

Figure 3.11: (a) Illustration of resistive poly-heaters and (b) temperature raised by Joule-heating of the standalone BD-PUF cells as the power dissipation of the heater increases. The inset figure shows the power-temperature dependency calibrated using external heating. The temperature range for calibration is relatively small, since the chips are attached to a tape which cannot be heated-up more than 70° C.

subsections. The temperature stability is still missing and will be introduced in this subsection.

In order to heat the devices up to a relatively high temperature, a part of the test chip contains standalone PUF cells surrounded by dedicated poly-resistor

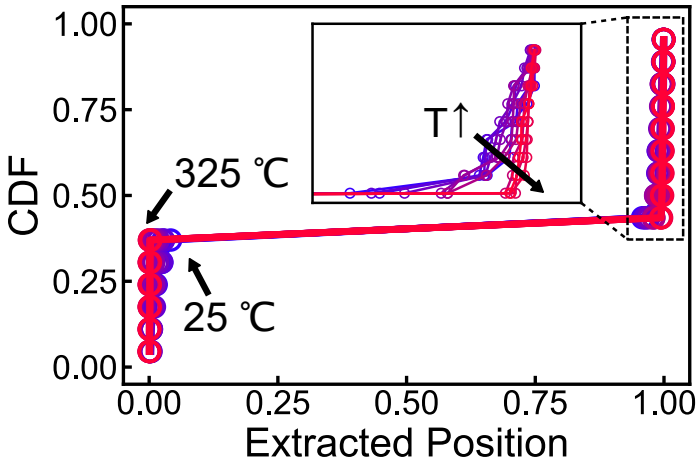


Figure 3.12: CDF of standalone devices with BD heated to different temperatures measured under 0.9V V_{DD}. Heating does not have negative impact on readout even for temperatures way above normal operation requirements, i.e. high temperature does not introduce physical changes to the BD path.

based heaters as shown in Figure 3.11(a). The heater is designed to dissipate large power for heating up the device almost instantaneously to several hundred degrees above room temperature [34]. The actual device temperature can be sensed by a dedicated diode placed next to the targeted device. The temperature is measured based on the ratio of diode voltage while injecting two constant currents, this ratio is linearly proportional to device temperature. Finally, the diode output voltages are calibrated with an external temperature control as shown in Figure 3.11(b). The standalone PUF cell can be heated 300K above room temperature before reaching the breakdown voltage of poly-resistors.

The extracted breakdown position of 15 standalone PUF cells under different temperatures are shown in Figure 3.12. The distribution of extracted position under 0.9V V_{DD} has no significant change from room temperature up to more than 300°C, implying that the outcome of a BD-PUF cell is not sensitive to temperature. Moreover, the characteristic of a PUF cell stays intact after being heated up to such high temperatures, which also implies that the physical structure of the BD path is not changed (e.g. anneal) by this amount of heat.

Besides the measurement on standalone devices, a similar set of experiments has been performed on PUF arrays using an external heating equipment. In this case the temperature cannot be set above 70°C due to setup limitation. As shown in Figure 3.13, the scattering plot of the extracted BD positions at room

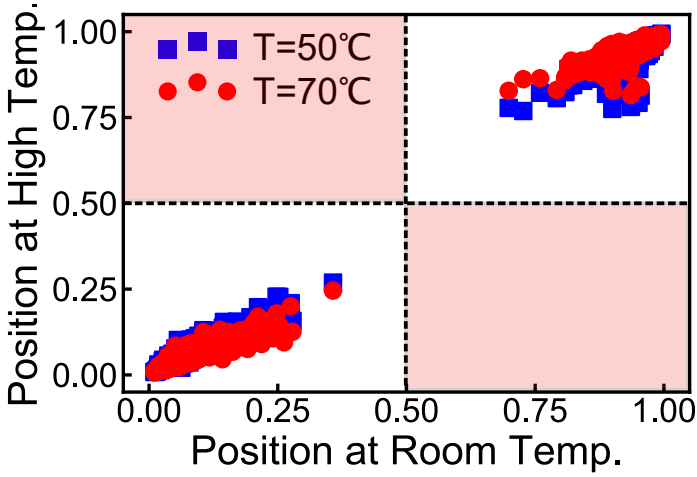


Figure 3.13: The relation between extracted position of the test arrays under room temperature and 50°C and 70°C at V_{DD}=0.9V. No device is observed within gray-colored region which may cause different readout at elevated temperatures, which proves that BD-PUF is stable over temperature.

temperature versus the extracted BD positions at 50°C and 70°C reconfirms the BD-PUF is insensitive to temperature. There is also no PUF cell that causes bit-flips during readout, since there is no data point locate in the colored region. Although we observe some fluctuation under different temperatures, it does not imply instability since the measurement environment changes when repeating for different temperatures, e.g. contacts between probes and pads are different. Both results from full-array and standalone PUF cells prove that the readout of BD-PUF is stable at high temperature.

3.4 Stability concerns

According to the experimental results and analysis shown above, two main concerns related to stability can be found. The first one is about instability during data readout and the second one is about having another BD spot in the same PUF cell.

3.4.1 Readout instability

As observed in the post-breakdown current characteristic (Figure 3.9), there is a small population of devices that have higher leakage current conducted by more defects. These increased leakage currents are the cause of the reduced readout window at lower V_{DD}, as shown in Figure 3.8.

Consequently, these devices with such high leakage current are expected to produce less-stable PUF bits at lower voltage, due to the small difference between the BD current and the leakage current. Note that even an overlap of I_{BD} and I_{leak} can be found in Figure 3.9, it does not imply that there will certainly be devices with the overlapped I_{BD} and I_{leak}. Since these two current components within one device are not strictly dependent, i.e. a higher leakage current does not imply a lower BD current and vice versa. Quantitatively, there is only a weak positive correlation of $\rho = 0.05$ between these two current components, and a positive correlation further implies a lower probability of the co-existence of high I_{leak} and low I_{BD} within a PUF cell. In this measurement set, the expected probability of finding a device with both current locating in the overlapped region for V_{DD}=0.9V is lower than 0.1%. With this low probability, an excellent overall data stability for a soft-BD PUF array is well expected, and will be confirmed in the next chapter.

3.4.2 Probability of having a 2$_{\mathrm{nd}}$ breakdown

The other concern is related to the concept of having only one BD spot. By the inspection of the leakage current, a certain device population is found to be more damaged, i.e. a percolation path is close to being formed. Although it has not been observed in the experiments, the second breakdown event of a PUF cell may happen in two time frames. The first one is during the forming process, it is possible that the two gate oxides have nearly identical t_{BD}, i.e., the difference is smaller than the response time of the compliance transistor, resulting in a second breakdown happening before the stress voltage is automatically reduced. The probability of this event, given the probability distribution function of t_{BD} and the response time of the compliance transistor, can be derived from the following equation:

$$\Pr(2_{\mathrm{nd}}\ \mathrm{BD}) = \int_{t=0}^{t=\infty} f_{BD}(t) \int_{u=t-\tau}^{u=t+\tau} f_{BD}(u) du dt \tag{3.6}$$

in which f_{BD} is the probability distribution function of the t_{BD} and τ is the response time of the compliance transistor. The equation can also be rewritten

into the format of cumulative distribution function F_{BD}, which can be directly obtained by fitting the measured distributions.

$$\Pr(2_{\mathrm{nd}}\ \mathrm{BD}) = \int_{t=0}^{t=\infty} f_{\mathrm{BD}}(t)\left[F_{\mathrm{BD}}(t+\tau) - F_{\mathrm{BD}}(t-\tau)\right] dt \tag{3.7}$$

Typically, the response time τ of a transistor is in the scale of nanosecond in the sub-micron CMOS technologies, which is much smaller than the t_{BD}. Under this condition, the following approximation can be made:

$$F_{\mathrm{BD}}(t+\tau) - F_{\mathrm{BD}}(t-\tau) \approx 2\tau f_{\mathrm{BD}}(t) \tag{3.8}$$

and the approximation of the second breakdown probability can be derived as Equation 3.9.

$$\Pr(2_{\mathrm{nd}}\ \mathrm{BD}) \approx 2\tau \int_{t=0}^{t=\infty} \left[f_{\mathrm{BD}}(t)\right]^2 dt \tag{3.9}$$

It shows that this probability has a linear dependence with respect to τ for a certain t_{BD} distribution. As a result, the compliance transistor needs to switch sufficiently fast, in order to reduce τ and the resulting probability of having the second BD.

Moreover, while the forming time is accelerated with higher stress voltage, the t_{BD} distribution becomes narrower and makes the integral term in this equation to become larger. Consequently, there is a higher risk of having a second breakdown spot within a PUF cell when aiming for a shorter forming time. The Equation 3.9 has to be considered when deciding the stress condition for the forming step, in order to reduce the probability of having a 2_{nd} breakdown.

Take the Weibull distribution fitted with the measurement data of forming with 4V as an example (see Figure 3.6), the probability of having two BDs in one device with $\tau = 1\mu\mathrm{s}$, which is far beyond the worst case, is around 1.04×10^{-6}.

Probability of having a 2_{nd} breakdown during readout

Another possible cause of the second breakdown spot is a breakdown event that happens at nominal operating condition. For a mature technology, the estimated time for a gate oxide breakdown to happen under nominal condition has to be long, e.g. longer than 10 years. If the gate oxide of the unbroken transistor is not damaged, this concern can be considered negligible, since it

is no different than other transistors outside the PUF circuit. However, as we have observed a certain amount of these transistors are more damaged, this could result in a shorter t_{BD} even under nominal condition.

Assuming a pair of transistors with time-to-breakdown equal to t and u under a certain stress voltage (e.g. $V_{\mathrm{stress}} = 4\mathrm{V}$), a 2_{nd} breakdown can occur during readout if $\tau < |t - u| \leq \tau + \nu$, where τ is the response time of the compliance transistor, and ν is the stress time that is equivalent to a 10-year operation time under a particular $\mathrm{V_{DD}}$. The parameter ν can be extrapolated using the power-law fitting as shown in Figure 3.7. For a $\mathrm{V_{DD}}$ equal to 0.9V and a stress voltage equal to 4V, it can be computed that $\nu = 7.5 \times 10^{-22}(\mathrm{s})$. Deriving from Equation 3.6-3.9, the probability of having a 2_{nd} breakdown during readout can be computed using the following equation, given that both τ and μ are much lower than the actual time to breakdown.

$$\Pr(2_{\mathrm{nd}}\ \mathrm{BD}, \mathrm{readout}) \approx 2\nu \int_{t=0}^{t=\infty} \left[f_{\mathrm{BD}}(t)\right]^2 dt \tag{3.10}$$

Based on this calculation, we can first conclude that having a 2_{nd} breakdown when operating at $\mathrm{V_{DD}}$=0.9V has a negligible probability below 1×10^{-20}. It should be noted that this calculation is valid if the PUF readout is continuously happening in this 10-year, and this is very unlikely to happen for a PUF circuit, i.e., the probability will be even lower when considering the actual workload.

According to power-law extrapolation, we can computed that ν will be around 1μs when $\mathrm{V_{DD}}$=2V. In this case, the probability of having a 2_{nd} breakdown during readout is about 1×10^{-6}. If choosing this probability as the standard, it implies that we can safely operate the PUF for more than ten years if $\mathrm{V_{DD}}$ is below 2V.

The two analysis in this subsection have shown that having a 2_{nd} breakdown during forming and readouts are not big concerns in theory. Although not done in this study, a more extensive set of experiments that including a long-term burn-in measurement will be very useful to provide solid proofs to further reduce these concerns.

3.5 Multi-bit/cell PUFs using analog BD positions

Following the initial idea of using the randomness of BD positions, the use of BD positions within the channel of a single MOSFET is still within our interests and will be introduced in this section.

3.5.1 Concept of analog-BD

As illustrated in Figure 3.2 and Equation 3.4, the extracted BD positions of a single MOSFET device will have analog values between 0 and 1, and can be quantized into multiple bits to exploit more entropy. Ideally, this technique is better than the binary variant, since it uses smaller two-transistor (2T) cells and more bits can be generated for a PUF cell, which shows a better area efficiency and more randomness. Moreover, if considering visual inspection on the BD spot, it would be even more difficult to identify the analog position of a BD spot within these nano-scale MOSFET devices.

3.5.2 Test structure

The test structure is designed and fabricated in the same commercial 40nm CMOS process, containing PUF arrays with gate lengths of 40, 100, 200 and 400nm respectively, as shown in Figure 3.14. This new version of test chip consists of two 32x32 PUF arrays and is evenly divided into 512 PUF cells of each gate length. Due to the increased amount of cells compared to the first test chip, all the word and bit-lines are no longer directly accessible, and hence peripheral circuits as shown in the schematic are needed. By providing a sequence of control signals to the periphery, we are able to measure analog current characteristics of every single cell, before, during or after the forming step, with less measurement precision compared to the array structure in Figure 3.5.

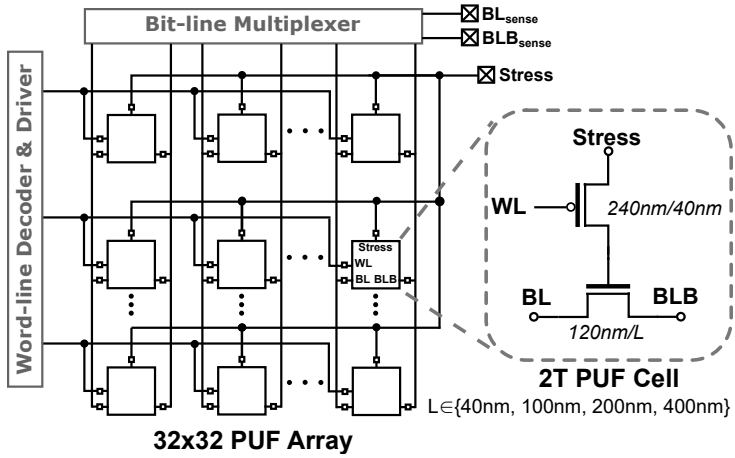
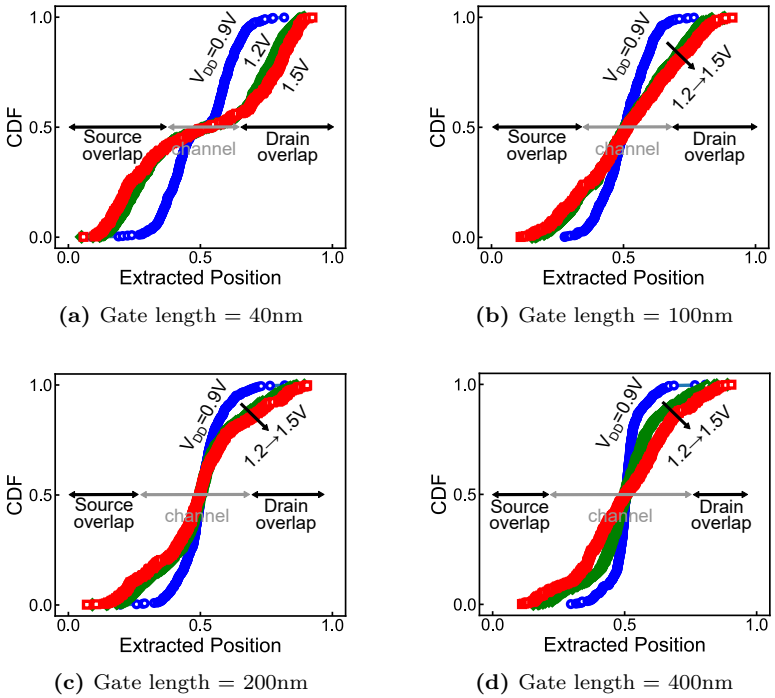


Figure 3.14: The 2T unit cell and the 32x32 array of the analog-BD PUF.



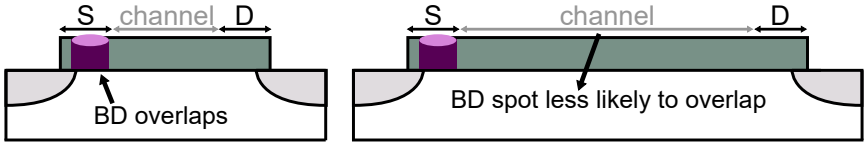
(a) Gate length = 40nm

(b) Gate length = 100nm

(c) Gate length = 200nm

(d) Gate length = 400nm

Figure 3.15: CDFs of the extracted position under three V_{DD}.



(a) **(b)**

Figure 3.16: The effect of the source/drain overlaps of breakdown spots. With shorter gate length, the breakdown spots are more likely to overlap with S/D, resulting in a more binarized distribution.

3.5.3 Analog BD positions

The extracted BD positions of the analog-BD PUFs are measured at different V_{DD}, as shown in Figure 3.15, using the current ratio technique (Equation 3.4). A first observation is that the resulting BD positions are not uniformly distributed

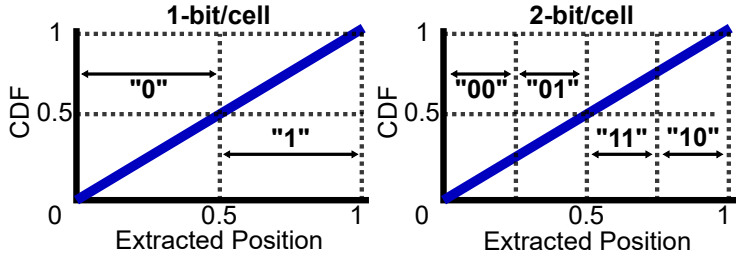


Figure 3.17: Illustration of digitizing the analog BD position into bits. Ideally, more bits can be generated by increasing the quantization levels, which makes the difference between the 1-bit/cell case and the 2-bit/cell case. For the 2-bit/cell case, the bits are encoded into Gray codes [37].

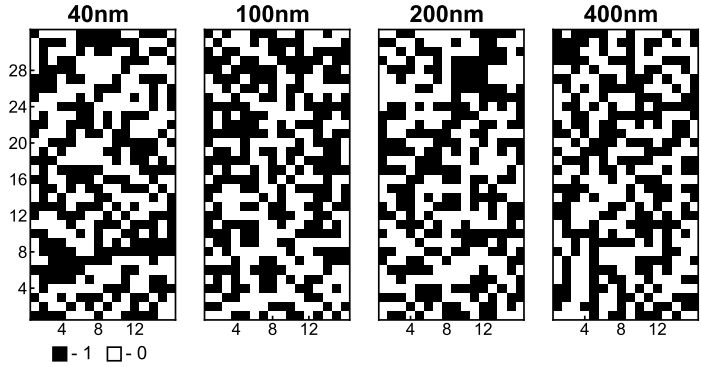
from 0 to 1, and the shape of distribution is different for each gate length variation.

In these distributions, there are in general three segments showing different slopes, in particular the slope of the middle segment is different from the other two. This effect can be attributed to the overlap of BD spots with the source (S) or drain (D) region, and the main evidence is that the population of the middle segment is increasing with gate length, as it reflected in the illustration of Figure 3.16.

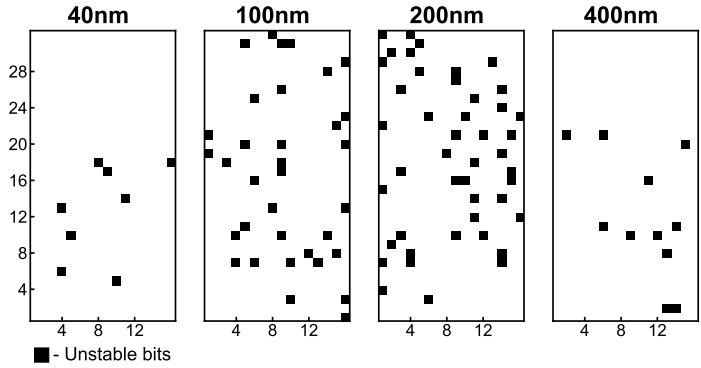
For the BD spots overlapping with S/D, the difference between I_{D} and I_{S} will be increased since the current flow to the overlapped terminal sees lower resistance and hence has larger magnitude. Consequently, the resulting current ratios of these overlapping spots are not representative for the actual BD positions.

Compared to the binary BD-PUF, the voltage dependence of the analog BD-PUF is less obvious especially for V_{DD} above 1.2V, since the exponential voltage dependence of the tunneling current will equally affect both the source and drain current. However, it still shows a narrowing distribution at lower voltage (V_{DD}=0.9V). This effect can be explained by taking the offset current into account, as illustrated in Equation 3.11. The offset current can be viewed as the summation of all the possible leakage currents from the inactive PUF cells, which has less voltage dependence comparing to the BD current. According to the updated equation, the resulting current ratio moves towards 0.5 since both I_{BL} and I_{BLB} decreases more rapidly. This narrowed distribution also discourage the use of this technique at lower V_{DD}.

$$CR = \frac{I_{\mathrm{BL}} + I_{\mathrm{offset}}}{I_{\mathrm{BL}} + I_{\mathrm{offset}} + I_{\mathrm{BLB}} + I_{\mathrm{offset}}} \tag{3.11}$$



(a) Binarized PUF data

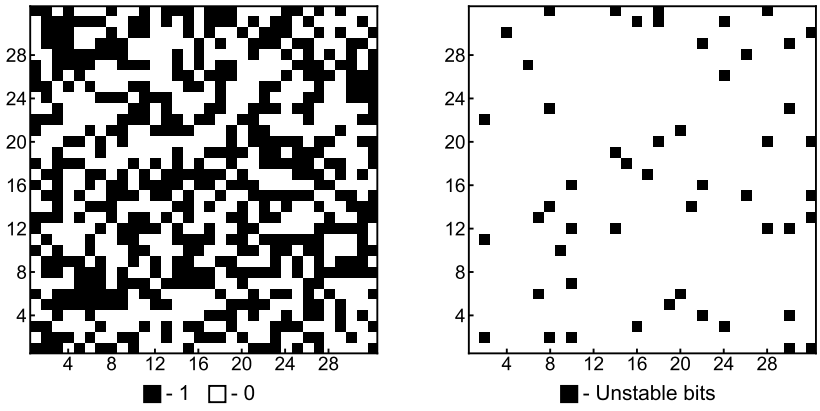


(b) Unstable bits

Figure 3.18: (a) The binarized data for the four 512-bit array with different gate length, showing no pattern. (b) The unstable PUF cells that have bit-flips during 5 measurements. The percentage of unstable bit from left to right are 1.56%, 6.64%, 8.78% and 2.14%.

3.5.4 Data stability

In order to compare the data stability with the binary BD PUF, the resulting BD positions are digitized into bits, for both the single-bit/cell or multi-bit/cell cases, as the examples shown in Figure 3.17.



(a) 2bit/cell PUF data **(b)** Unstable bits

Figure 3.19: (a)The 2b/cell encoded data from the 512-bit PUF array with gate length of 40nm and (b) the location of unstable bits. The percentage of unstable bits in this case is 4.98% which is about three times larger than the 1b/cell case (see Figure 3.18).

Single-bit/cell

The resulting PUF data is shown in Figure 3.18 (a). For the four different cases, there is no pattern observed and the ratio of "0" and "1" bits are almost equal, showing a good randomness for the PUF data. Regarding the stability, since there is no dedicated readout circuit available for this PUF implementation, the stability is roughly estimated by repeating the DC measurement multiple times. Even though the actual BD position will not be shifted over time, the measured current values may change due to noise and environmental changes. By measuring at V_{DD}=1.5V for five times, the PUF cells which shows at least one bit flip ("0" to "1" or "1" to "0") is indictated in Figure 3.18 (b). A significant amount of unstable bits are observed with only five measurements, indicating a poor data stability of the analog-BD PUF.

In the case of generating a single bit, the array with the minimum gate length has the best stability among all. This is the consequence of the higher probability to have breakdown spots overlapping with S/D as observed in Figure 3.15, i.e. more binarized. The stability is degraded as the gate length increases to 100nm, because the breakdown spot is less likely to overlap the terminals and the current ratio becomes more uniformly distributed. This trend, however, does not continue since the breakdown spot is less likely to be located right at the middle as the gate length is further increased. Consequently, the PUF array

of 400nm gate length has the second-best stability.

Multi-bit/cell

As one of the advantage expected for the analog-BD PUF, the breakdown position of the analog BD-PUF can be digitized into more bits to harvest more entropy. Taking the PUF with the gate length of 40nm as an example, the breakdown position is divided into four regions, by the decision thresholds of 0.25, 0.5 and 0.75, and these positions are encoded into 2-bit Gray codes. The resulting data with the doubled number of bits (from 512-bits to 1024-bits) is shown in Figure 3.19 (a). Note that due to the non-uniform distribution shown in Figure 3.15, the actual entropy of the generated data is not doubled.

As the number of the decision thresholds increases, more devices may result in a current ratio close to these thresholds, and will generate bits which are less stable. By observing the unstable bits shown in Figure 3.19 (b), a clear stability degradation can be found. The percentage of the unstable bits in the 1024-bit case is about three times larger than the 512-bit case. This result clearly shows that the number of unstable bits is closely related to the number of decision thresholds, which makes the idea of generating more bits from one PUF cell less reliable.

3.5.5 Comparison with binary BD PUF

It has been proven that the analog BD PUFs are feasible of generating more than one bit out of a single PUF cell, and hence more entropy can be extracted. Apart from this advantage, the analog BD position can only be determined from current readouts with high accuracy, which is done by external equipment in this work. This readout circuit is expected to be a bottleneck for this design when considering to integrate it on-chip, due to resource constraints. In contrast, the readout circuit for the binary BD PUF is rather simple, since it only needs to compare the current magnitudes and hence is easier to be implemented on-chip.

Moreover, even using an external measurement equipment with high precision, the resulting current ratio can still fluctuate slightly, causing instability when the BD spot is close to the middle. For the binary-BD PUF, since the current ratios are well separated, there will be no instability found in such experiment. While generating two bits out of one PUF cell introduces more instability, it shows that generating multiple bits is not only limited by the precision of readout circuit but also limited by the degrading data stability.

Consequently, the analog BD PUF is so far not considered as a better choice comparing to the binary BD PUF. It would need a significant improvement on the bit stabilization techniques to be able to use it in real applications.

3.6 Conclusion

In this chapter, we have introduced the concept of using the positions of gate oxide breakdown location as entropy source. The proposed PUF structure featuring the self-limiting breakdown generation property has been manufactured and proven as a viable solution for *active* PUF implementations. It has been demonstrated that the initialization step, the forming process, is feasible to generate the required soft-BD spots in the PUF cells reliably within a sufficiently short period.

The resulting current characteristic has been measured and the binary-like distribution indicates that the data readout of the soft-BD PUF has good stability. The experiments at high temperature have shown that the soft-BD PUF is insensitive to temperature variation, which is another important reliability aspect for a PUF. The quantitative analysis for assessing the stability concerns are performed and the result shows that both the probabilities of having similar $I_{\mathrm{BD}}/I_{\mathrm{leak}}$ in oppositely formed PUF cells and having a BD spot in both transistors are very low.

Finally, by comparing with an alternative PUF design based on analog BD positions, we have proven that using a binarized strategy is indeed better than using an analog one. Consequently, the binary soft BD-PUF is further integrated in a more complete PUF circuit structure to perform more detailed experiments and analysis. This is the topic of the next chapter.

Chapter 4

Design, characterization and statistical analysis of soft oxide breakdown PUF

Content Source

The main material in this chapter was previously published in:

K.-H. Chuang, E. Bury, R. Degraeve, B. Kaczer, D. Linten, and I. Verbauwhede, "A Physically Unclonable Function Using Soft Oxide Breakdown Featuring 0% Native BER and 51.8fJ/bit in 40nm CMOS," *IEEE Journal of Solid-State Circuits*, vol. 54, no. 10, 2019.

Contribution: Main author.

The PUF based on soft oxide breakdown is designed to achieve excellent data stability. With all the essential peripheral circuits, the complete soft-BD PUF test chip is designed and fabricated in 40nm CMOS process. These experimental results have successfully demonstrated the excellent data stability and show that the other statistical properties are close to the ideal case.

4.1 Introduction

It has been demonstrated in the previous chapter that the 3T soft-BD PUF cell has a binary current characteristic, providing stable PUF outcomes. The device level characterization can indeed help to gain fundamental understanding of the circuit behavior, that is, however, insufficient to show the feasibility for system integration. In order to demonstrate the good PUF properties within a complete circuit structure, the PUF array described in the previous chapter is widely expanded and integrated with custom designed readout interface and other control circuits.

4.1.1 Overview of circuit blocks for soft-BD PUF

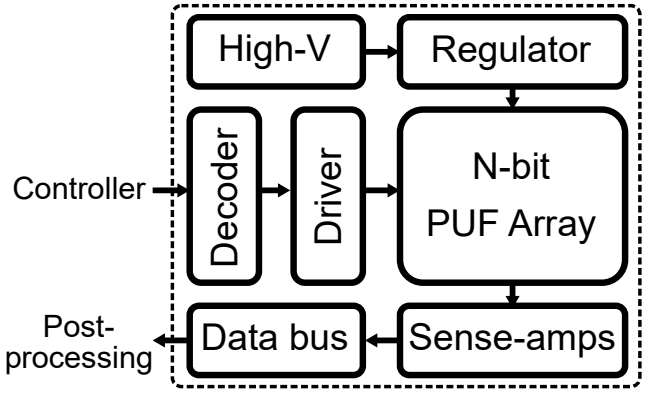


Figure 4.1: A simplified block diagram showing most of the essential circuit blocks providing a complete functionality. The dashed box indicates the border between the PUF module and other circuits.

In order to transform the entropy from the PUF cells into usable digital bits, several peripheral blocks are required. A simplified block diagram showing the essential circuit blocks enabling the PUF operation is shown in Figure 4.1, and the purpose of each block will be briefly explained in the following paragraphs.

Unlike the SRAM PUFs, the BD-PUF cell does not transform the transistor level variations into digital domain by itself, i.e. there is no “1” or “0” bit defined by voltage level within a PUF cell. Consequently, a readout interface with the capability of distinguishing the analog difference is required to transform the breakdown positions into digital bits. In this design, a custom designed differential sense-amplifier is exploited for this purpose, and the design details will be discussed later.

A weak PUF array has a similar functionality as memory arrays, and hence it also requires an address decoder and an output interface. For this design, the address decoder only needs to control the access of word-lines (WLs) and the parallel output is scanned into serial bits by a DFF scan chain, as shown in Figure 4.2. Due to the high voltage forming process, a specific wordline driver is required to provide high voltage control signals for the wordline selectors.

In order to perform the forming process, a PUF array must be supplied with a high voltage source. For this 40nm CMOS technology, a forming voltage greater than 4V is needed to achieve a sufficiently low forming time, as discused in the previous chapter. While the supply voltage for I/O in this technology is 2.5V, it is infeasible to reuse this voltage for the forming process. Consequently, the PUF chip would require an additional pin for an external high voltage supply or an embedded high voltage generator, such as a boost converter or a charge-pump.

It should be noted that there is no obvious winner between an extra pin or an extra circuit, since both the number of bonding pads and the chip area can be the main constraint for a chip design, depending on the design criteria and use cases. For our prototype design, there is more design freedom than real products, and we have chosen to use an extra pin for external voltage supply, which provides a more flexible voltage range for experimental purposes. On the other hand, if the targeted application needs to provide program capability in different environments, e.g. on a contactless card, an on-chip voltage generator will be a better choice. This functionality is planned to be included and tested in the follow-up designs.

4.1.2 Chapter organization

In this chapter, we will first demonstrate how the complete soft-BD PUF test circuit was implemented, including the custom-designed reference-free differential sense amplifier. The data stability will be tested by thorough experiments, with focus on voltage and temperature variations. Most of the important statistical properties of the generated PUF data will be discussed, showing the randomness, uniqueness and spatial correlation of the PUF circuit. A first-order analysis on the side-channel attack immunity of the PUF circuit will also be discussed. Finally, the summary of this PUF work and a detailed comparison with prior work will be given.

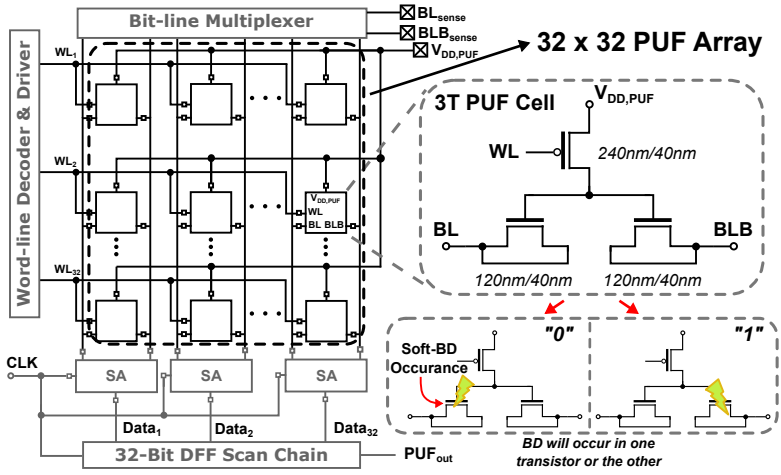


Figure 4.2: The schematic of the PUF array including the periphery circuits and the 3T PUF cell. The stress voltage is applied through the $V_{\mathrm{DD,\,PUF}}$ pin. © 2019 IEEE.

4.2 Circuit implementation

The soft-BD PUF has been first designed as a 5-by-12 array without any periphery circuit, as shown in [13] and the previous chapter. It allowed a more precise characterization of the circuit behavior before, during and after oxide breakdown, which provided the initial insight for designing a complete PUF circuit. The PUF circuit in this chapter is designed and fabricated in a 40nm CMOS process, and consists of a 32-by-32 PUF array and periphery circuits including the sense-amplifier based readout interface.

4.2.1 1024-bit PUF array

As shown in Figure 4.2, the PUF array has a typical structure with 32 shared wordlines (WLs) and 32 shared bitlines (BLs). Using this structure, the dimension of the PUF array is rather flexible, there are only concerns regarding the chip area and the summed-up leakage current in the shared bitlines. Since a 32-by-32 array provides sufficient statistical data for experimental purposes, the array is not further up-sized.

4.2.2 Peripheral circuit blocks

The control logic of our design follows a simple serial-in parallel-out (SIPO) principle, utilizing a chain of shift-registers to input the control sequences. This type of control logic is typically slower but has more flexibility, which makes it more suitable for this prototype.

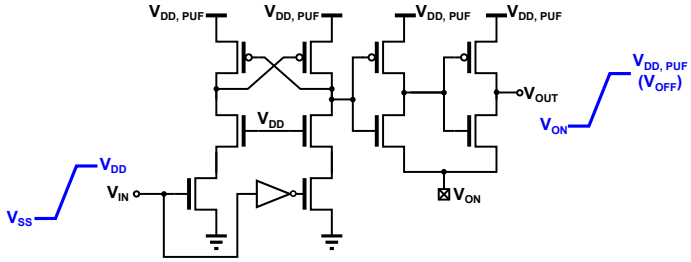


Figure 4.3: The schematic of a wordline driver, the input logic level is shifted to the level defined by $V_{\mathrm{DD,PUF}}$ and V_{ON}. Note that this circuit functions as a normal buffer when $V_{\mathrm{DD,PUF}} = V_{\mathrm{DD}}$ and $V_{\mathrm{ON}} = V_{\mathrm{SS}}$.

The wordline driver is modified from a conventional level shifter, as shown in Figure 4.3. The logic levels for the core circuit are typically 0V for "0" and V_{DD} (around 1V in this 40nm CMOS technology) for "1". The first cross-coupled stage can shift the logic level "1" from V_{DD} to a higher V_{OFF} voltage. This shifted signal will then drive the output to generate logic levels from V_{ON} to V_{OFF}, which can control the on/off of the wordline selectors during forming process. In the readout phase, the V_{ON} voltage will be set to zero and the V_{OFF} voltage will be set as the V_{DD} of the PUF circuit ($V_{\mathrm{DD,PUF}}$).

The bitline multiplexer is not a required circuit block for PUF operation, but is added for experimental purpose. The bitline multiplexer can help to observe the actual breakdown current – the current flowing through individual BLs can be directly multiplexed to sensing pads for external DC measurements. The current characteristic is similar to that reported in [13], with some inevitable interference from other cells due to the periphery. The PUF data extracted from DC measurements exactly matches the PUF data generated by the SAs, which reconfirms the operating principle of the soft-BD PUF.

Another peripheral block is the DFF scan chain to transmit the PUF data out of the chip. As a common circuit element for testing, the scan chain can collect the data in each individual data register and transform it to a serial bitstream. For the 32 PUF columns, 32 scan DFF is required and only one output pin is needed for data transmitting.

4.3 Sense-amplifier readout scheme

As an interface between the analog and digital domain, the sense-amplifier in our PUF design has a decisive impact on the data stability. The resolution of a sense-amplifier determines the resilience against the internal and external variations, which is particularly important for the soft-BD PUF with exponential voltage-current dependence. Moreover, the operating speed of the sense-amplifier also closely impacts the throughput and latency of the PUF circuit.

4.3.1 Challenges

There are many existing SA designs for memory readout, including the conventional voltage mode SA for SRAM [85] and the current mode SA for the emerging resistive type memory devices [66]. As the PUF bits are determined by the current differences, the voltage mode sensing techniques are not suitable for this application, and a current mode sensing method is thereby chosen.

The main challenge for designing a good sense-amplifier for the soft-BD PUF, is the exponential voltage dependence of the breakdown current. As shown in the previous chapter, the breakdown current at the nominal V_{DD}=0.9V can be lower than 5nA, making the current sensing procedure rather challenging. Typically, a current mode SA is targeting input resistances of kΩ range, as the

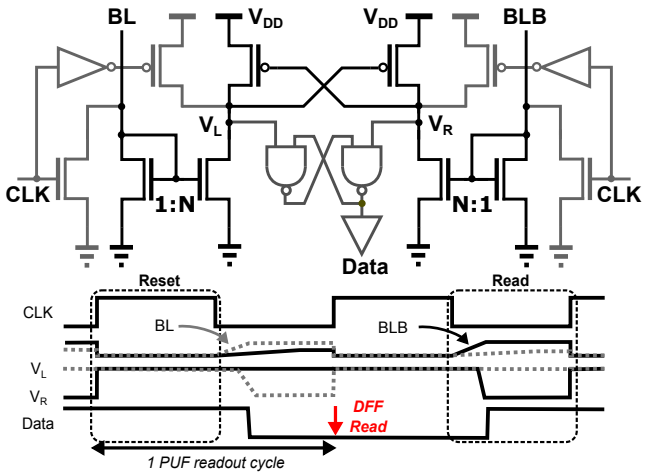


Figure 4.4: The schematic of the proposed reference-free sense-amplifier and the corresponding timing diagram. © 2019 IEEE.

examples shown in [66, 88]. Moreover, once the SA input has a DC value, it might reduce the voltage drop across the gate oxide and hence largely reduce the current. In order to maintain enough current difference during readout, the SA is required to have a low input DC value during the entire readout procedure.

Note that, the single-ended design such as the SA in [88] is not suitable for soft-BD PUF, since it is difficult to define an optimal reference current or voltage for the current profile in Figure 3.9. Besides this limitation, using a differential readout scheme has the advantage of providing better immunity against side-channel attacks.

4.3.2 Topology selection and SA operation

The schematic of the proposed SA is shown in Figure 4.4. The main structure of this differential SA consists of a current mirror input stage followed by a cross-coupled pair active loading. The input current difference from a PUF cell is amplified by N (N=20 in this design) times by the current mirror, and the amplified difference will trigger the positive feedback loop of the active loading, which will then generate the rail-to-rail output signal.

The operation is enabled by the peripheral elements, including NMOS/PMOS switches and an SR-latch to hold the generated data. As illustrated by the timing diagram, a readout cycle starts with resetting the SA when *CLK* is set to 1. In the reset phase, the BL and BLB nodes will be set to the default value of 0 and the V_{L} and V_{R} nodes will be set to the default value of V_{DD}. The reset procedure is imposed to eliminate the memory effect from the previous readout.

The readout procedure starts when *CLK* is set to 0. The soft-BD current from either *BL* or *BLB* will be amplified to cause a fast discharging on V_{L} or V_{R}. The second stage will be latched once sensing a certain difference between V_{L} and V_{R}. The state of the final SR-latch (*Data*) will be updated accordingly, and be ready for D flip-flop registering at the next clock rising edge.

In order to reduce the interference between PUF cells during readout, every BL pair is connected to an individual SA without multiplexing. The area penalty is small because of the compact layout of the SA, as shown in Figure 4.5 (a). Furthermore, having multiple SAs allows parallel operation, the throughput is thereby increased by 32 times comparing to using single SA.

4.3.3 Circuit Analysis

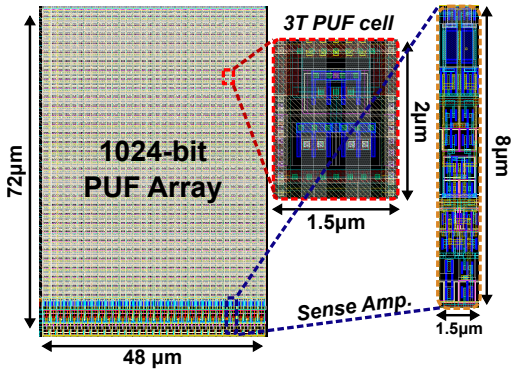


Figure 4.5: Layout of the 1024-bit PUF array and the sub-cells.

In order to verify the functionality of the proposed SA, the readout resolution, i.e. the minimum sensable current difference, is the most important index to be checked. As the breakdown current reduces significantly with decreased V_{DD}, the readout operation will eventually fail when the current difference is below the SA resolution. Consequently, the minimum applicable V_{DD} will be determined by the resolution of the SA. In other words, a SA with higher resolution may result in a wider operating range.

The SA resolution can be estimated by performing a set of Monte-Carlo simulations using the Spectre™ circuit simulator and the standard device models provided by the foundry. Since there is no standard model for post-breakdown current characteristics, the PUF cells used in our simulations are manually modeled by adding the leakage resistors and the soft-BD path, as shown in Figure 4.6. The soft-BD path is described in the Verilog-A langauge, having a current-voltage dependence characterized by the following equation:

$$I_{\mathrm{BD}} = I_{\mathrm{ref}} \times \exp\left[c\left(V_{\mathrm{BD}} - V_{\mathrm{ref}}\right)\right] \tag{4.1}$$

where the I_{ref} is the current value when $V_{\mathrm{BD}} = V_{\mathrm{ref}} = 0.9V$ and $c \approx 6.8$ is the constant for exponential scaling, which is derived from the current characteristic shown in Figure 3.9. The leakage resistance $R_{\mathrm{leak}} \approx 120\mathrm{M}\Omega$ describes the worst case leakage current, while neglecting the relatively weak voltage dependence. In this simplified circuit model, there are two identical resistors connected to the BL and BLB terminals, which have the purpose of avoiding bit-flips caused by the PUF cell, in particular at lower V_{DD}. As a result, only the bit-flips caused by the SAs will be observed in the simulation, and hence helps to characterize the performance of the SAs.

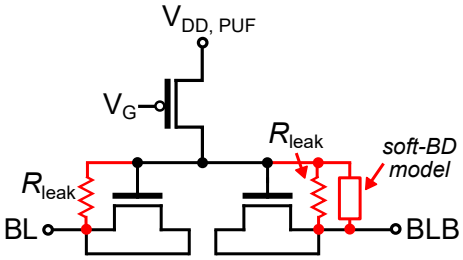


Figure 4.6: Circuit model for the soft-BD PUF cell used in circuit simulation. This model is for a PUF cell with "1"-bit, the soft-BD path will be moved to the left (BL) side in the case of "0"-bit. © 2019 IEEE.

By adjusting the model parameter, the simulations can show the failure rate introduced by the SAs under different input current differences between I_{BD} and I_{leak}. For this simulation set, the current difference at V_{DD}=0.9V is selected as the variable for parametric sweep, which is equivalent to the I_{ref} described in equation 4.1. As shown in Figure 4.7 (a), the SA has a resolution below 2nA, while simulating with process corners and transient noise. As an analog circuit element, the SA is also expected to be vulnerable to the input offset introduced by circuit mismatches. When including local mismatch in the Monte-Carlo simulation, the readout resolution is severely degraded to 8nA, showing that mismatch is a serious concern for the SAs.

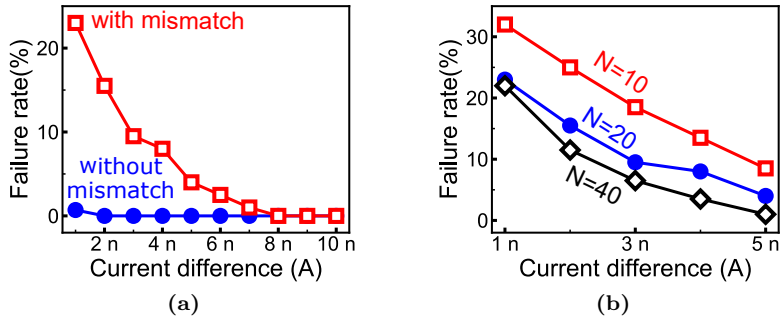


Figure 4.7: (a) Failure rate in 1000 Monte-Carlo simulations with and without local mismatch v.s. input current difference at V_{DD}=0.9V and (b) failure rate with mismatch using different current mirror ratio (N). © 2019 IEEE.

As a common strategy to cope with the negative impact of mismatch, increasing the size of transistors is usually beneficial. For this particular design, the current mirror ratio (N) was found to be the key parameter. As shown in Figure 4.7 (b),

a larger N indeed results in a better data stability. The improvement, however, comes at the cost of chip area. As described in Figure 4.8 and equation Equation 4.2, once the W_1 and L_2 are set to the minimum value, either M_1 or M_2 has to be enlarged to get an increased N. As a result, the trade-off between data stability and chip area needs to be considered in the design. The N=20 for this design is not the optimum value with respect to stability, it was chosen to achieve a reasonably good stability without too much area overhead.

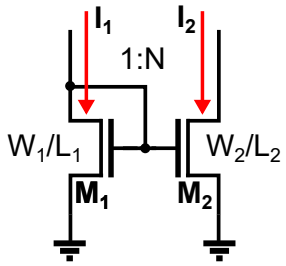


Figure 4.8: Illustration of a current mirror, with a certain combination of the width/length of the transistors M_1 and M_2, the ratio of input current I_1 and output current I_2 can be determined.

$$N = \frac{W_2/L_2}{W_1/L_1} = \frac{W_2 L_1}{W_1 L_2} \tag{4.2}$$

As a result, there are two main directions to improve the data stability of soft-BD PUF. The first one is to increase the current difference introduced by the PUF cells by forming the PUFs with higher current. This method forces the input current of SAs to be higher than the minimum resolution, but with the price of a higher risk on visibility and higher power consumption. The next method is to increase the size of the SAs to lower the effect of mismatch, but it increases the area consumption. Following this direction, we can consider to increase both the size of SAs and the number of PUF cells sharing a SA, which maintains the overall area occupied by the SAs. This method will be infeasible when the interference from the neighboring PUF cells starts dominating. With a certain area constraint, it is possible to find an optimum combination.

4.4 Experimental setup and results

The fabricated PUF test chips are packaged and mounted on a PCB. A wire-bonded chip is shown in Figure 4.9. An FPGA is being used to generate the control signals and receive the digital output signals from the PUF chips. In

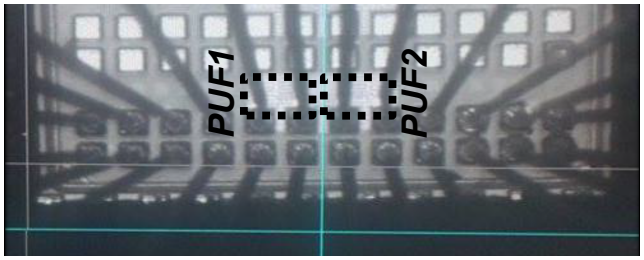
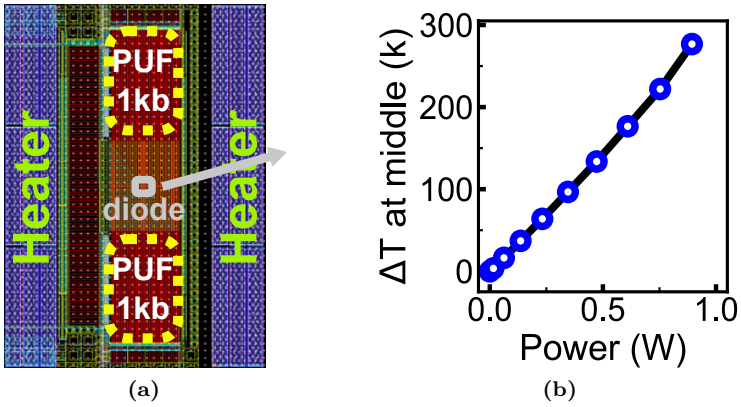


Figure 4.9: Photo of a wire-bonded PUF chip.

order to characterize the effect of temperature variations, the PUF chips are placed in a temperature chamber for both cooling and heating. In addition, an on-chip resistive heater is integrated in the design, which acts as an alternative heat source, in particular for high temperatures exceeding the specification of the connection cables (70° C for this setup). Using an on-chip heater also has the advantage of fast temperature tuning within few seconds, while the temperature chamber requires several minutes to adjust the target temperature. The structure of the PUF circuit with on-chip heaters is shown in Figure 4.10 (a), and the temperature at the middle of the array can be characterized using the method proposed in [34], as shown in Figure 4.10 (b). The temperatures extracted by the diode sensor were calibrated by external heating in a different setup, up to 200°C.



(a) (b)

Figure 4.10: (a) Layout of PUF arrays with surrounding heaters and (b) temperature raised by raise of the array v.s. the power dissipated by the heaters.

4.4.1 Stability against voltage variation

As the voltage dependence is a major concern for the soft-BD PUF, we first examine one PUF array for the stability at different values of V_{DD} under room temperature, as shown in Figure 4.11. The quantity, *"Unstable bits"*, represents the percentage of PUF cells that introduce at least one error in all readouts, which also indicates the percentage of "dark bits" needed to be treated by the dark-bit masking scheme for stability improvement. The *"BER"* is the percentage of error bits produced by all PUF cells throughout all measurement cycles, which is used to determine the specification of the error correction scheme to be used in the helper data algorithm. A clearer definition for each term can be referred to section 2.3.4.

For stability evaluation, there is another important quantity, "flipped bits". It is the ratio of PUF bits which generate different values under a certain operating condition, compared to the *golden* PUF data under nominal condition. In order to maintain a consistent secret key under various operating conditions, these flipped bits have to be taken into account for the masking or the error correction schemes. This term can also be merged in to the unstable bits and the BER, by always taking the golden PUF data as the reference when determining bit errors.

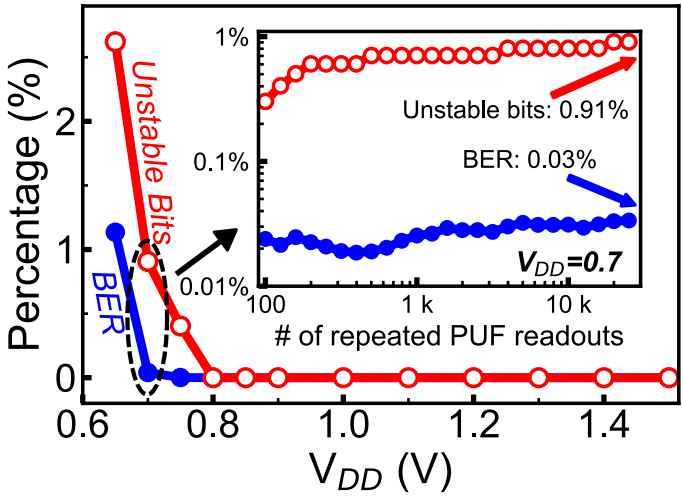


Figure 4.11: The percentage of unstable bits and BER v.s. V_{DD} and v.s. repeating readout cycles at V_{DD}=0.7V (inset). All data are at room temperature.

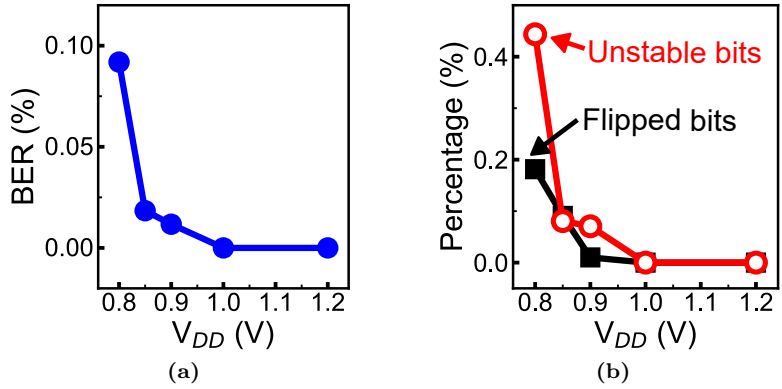


Figure 4.12: (a) The BER of 10k PUF readout cycles from 10 PUF arrays (10k PUF cells) at different V_{DD} and (b) the percentage of unstalbe bits and flipped bits of the same measurement set. © 2019 IEEE.

This PUF chip is well functioning down to V_{DD}=0.7V, while the BER is below 0.1% at 0.7V and is 0% (no bit error observed in all readouts) for $V_{\mathrm{DD}}\geq$0.8V with more than 20k readout. There are no flipped bits from 0.8V to 1.5V comparing to the golden data obtained at 1.2V. The BER of 0% shows that the soft-BD PUF has approached to the ideal stability. The error-free operating region covers very well the ±10% of the nominal V_{DD} of 0.9V, which satisfies the minimum requirement for a commercial IC. For these particular PUF cells, there is no need for helper data if operated under nominal condition. Figure 4.11 (inset) shows the cumulative number of unstable bits and BER at 0.7V for up to 20k PUF readouts.

With the sample size of only one PUF array, some unstable cells might be overlooked due to the low probability. For the current distribution at V_{DD}=0.9 shown in Figure 3.9, around 3% of the $\mathrm{I_{leak}}$ population might overlap with $\mathrm{I_{BD}}$. Assuming the BD current and leakage current are uncorrelated, the probability of having an overlapped current characteristic at V_{DD}=0.9V will be less than 0.1%. This estimation is also valid when considering the actual weak positive correlation between the two current components, since a higher leakage current is more likely to be accompanied by a higher BD current, making it less probable to have an unstable PUF cell.

With this low probability, it is therefore reasonable that no unstable bit has been observed at V_{DD}=0.9V in a sample size of only 1k. To obtain a wider sample size, more PUF chips have been measured extensively, i.e. with more than 10k measurement cycles and covering the nominal operating range of ±10% V_{DD}.

Figure 4.12 (a) shows the BER of 10 PUF arrays (10k PUF cells), the BER remains 0 for V_{DD}>1V, but is no longer zero at lower voltages. Figure 4.12 (b) shows the percentages of unstable bits and flipped bits with respect to the golden data obtained at V_{DD}=1.2V. Both curves follow the same trend as the BER, showing that more bits tend to become unstable or flipped to a different value at lower operating voltage. It also confirms that there are no flipped bits when BER is zero.

These results also reconfirm that the soft-BD PUF is not an ideal solution when considering ultra low voltage operations, due to the limitation of the soft-BD current characteristics. In case there is a need for expanding the operating range, it has to come with certain trade-offs. One straightforward method is to increase the breakdown strength, which will increase the window between BD current and leakage current. This method, however, contradicts the purpose of using soft-BD, since it might result in a more visible BD spot. Another solution is to increase the number of PUF cells and store the helper data in these additional PUF cells. It can be done by adjusting the bias condition during the forming process, which will then generate breakdown on the selected transistors. The helper data stored in these cells can be used to recover the correct PUF outputs, particularly when operating at lower voltages. In this case, although the embedded NVM is not required, it still has the cost of increased PUF size and the additional post-processing circuits.

Bit-error-rate estimation

Through these experiments, we have recorded several BER values that are equal to zero, i.e., no error is observed in an experiment. It is obvious that zero BER is the ideal result that one can find in an experiment. Beyond that, it is also important to understand the implications of finding no error in an experiment. In particular, our goal is to estimate the actual BER of a PUF chip, while there is no error found through one million readouts.

Let a readout of a PUF bit be a random variable X, where x is the outcome of X and $x \in \{0, 1\}$. For a correct readout, $X = 0$; and for an incorrect readout, $X = 1$. Given a BER ϵ sampled from a random variable E with an uniform distribution ranging from 0 to 0.5. The probability density function of E can be described as: $\mathrm{f}_E(\epsilon) = 2$ for $0 \leq \epsilon \leq 0.5$ and $\mathrm{f}_E(\epsilon) = 0$ elsewhere. In this case, the probability of having an incorrect readout: $\Pr(X = 1) = \epsilon$. For an experiment on a n-bit PUF array with k readouts, assuming all the PUF cells have an identical BER of ϵ, this experiment is equivalent of having $m = n \times k$ samples from a random variable X with a particular ϵ. In the m readouts, the

total number of errors is another random variable $\tilde{X} = \sum_{i=1}^{m} X_i$, where X_i represents the i_{th} sample of X.

Given that no error is found in the experiment, we want to derive the conditional probability of the real BER (E) to be smaller than a particular value ϵ', as written in Equation 4.3.

$$\Pr(E < \epsilon' | \tilde{X} = 0) = \frac{\Pr\left[(E < \epsilon') \cap (\tilde{X} = 0)\right]}{\Pr(\tilde{X} = 0)} \tag{4.3}$$

For a given ϵ, the probability of $\tilde{X} = 0$ is derived as Equation 4.4. Based on this equation, the two probabilities $\Pr(\tilde{X} = 0 | E < \epsilon')$ and $\Pr(\tilde{X} = 0)$ are derived as Equation 4.5 and Equation 4.6, respectively.

$$\Pr(\tilde{X} = 0 | E = \epsilon) = (1 - \epsilon)^m \tag{4.4}$$

$$\Pr\left[(E < \epsilon') \cap (\tilde{X} = 0)\right] = \int_{\epsilon=-\infty}^{\epsilon=\epsilon'} (1 - \epsilon)^m \mathrm{f}_E(\epsilon) d\epsilon = 2 \times \frac{1 - (1 - \epsilon)^{m+1}}{m + 1} \tag{4.5}$$

$$\Pr(\tilde{X} = 0) = \int_{\epsilon=-\infty}^{\epsilon=\infty} (1 - \epsilon)^m \mathrm{f}_E(\epsilon) d\epsilon = 2 \times \frac{1 - 0.5^{m+1}}{m + 1} \tag{4.6}$$

Consequently, Equation 4.3 can be rewritten as Equation 4.7. Given the experimental results in Figure 4.12, where $m = 1 \times 10^7$, the probability of having a BER below 1×10^{-7} is around 0.63 and the probability of having a BER below 1×10^{-9} is around 0.01. In summary, if we want to proof that the BER is below the common target of 1×10^{-9} with a good confidence, a much larger sample size is needed.

$$\Pr(E < \epsilon' | \tilde{X} = 0) = \frac{1 - (1 - \epsilon')^{m+1}}{1 - 0.5^{m+1}} \tag{4.7}$$

4.4.2 Stability against temperature variation

It is also important for a PUF circuit to remain functional under different temperatures, as it might be used in IoT devices deployed in different types of uncontrolled conditions. Another reason is that temperature sensitivity of a circuit can be exploited to perform certain fault injection attack schemes [44].

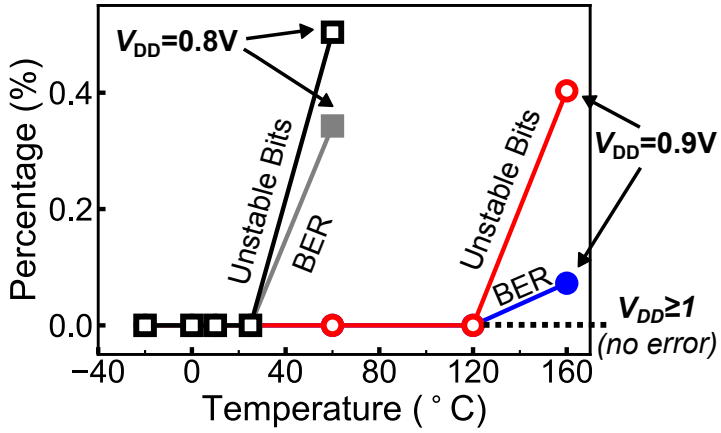


Figure 4.13: The percentage of unstable bits and BER at different temperature, operating at V_{DD}=0.8V and 0.9V. © 2019 IEEE.

In order to understand the impact of temperature variation, the data stability at reduced and elevated temperatures has been tested, using both a temperature chamber and the on-chip heater.

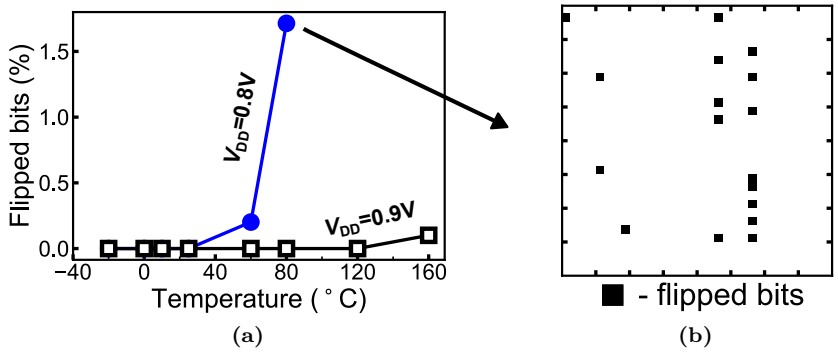


Figure 4.14: (a) The percentage of flipped bits in different temperatures and (b) the positions of flipped bits within an array at 0.8V and 80 °C. © 2019 IEEE.

As shown in Figure 4.13, *Unstable bits* and *BER* of a PUF array remains 0 at room temperature and below for all the values of V_{DD}. When operating at V_{DD}=0.9V, the 0% BER holds until 120°C. Once the V_{DD} is lowered to 0.8V, some of the PUF bits already become unstable at 60°C. It can be clearly observed that the stability degrades at elevated temperatures, but this dependence cannot

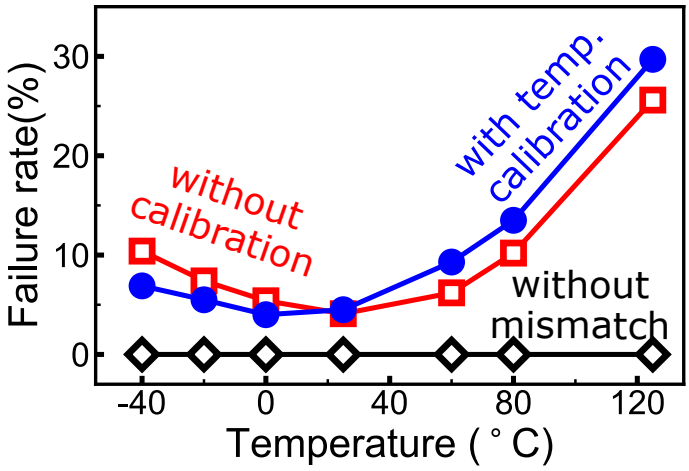


Figure 4.15: Monte-Carlo simulation of SA under different temperature. The PUF cell used in this simulation set has a current difference of 5nA at 0.9V. The results without considering local mismatch shows no failure in all temperature. The simulation trend move closer to actual measurement once temperature dependence is applied to the PUF cell. © 2019 IEEE.

be explained by the measurement results shown in the previous chapter, since the current characteristic of PUF cells is insensitive to temperature.

To further investigate the cause of temperature dependence, the percentage of flipping bits and their physical locations are plotted in Figure 4.14. It can be clearly seen that most of the flipping bits are located in the same columns, in which these devices are sharing same SAs. This observation thereby implies that for these columns, the input offsets of the SA Are larger than others and this is dominating the readout of the corresponding PUF cells.

In order to verify this hypothesis, a new set of Monte-Carlo simulations is performed, with the results shown in Figure 4.15. For the case without device mismatch, the failure rate remains zero for all simulated temperatures, showing that the readout procedure is temperature insensitive if no offset is present. When taking device mismatch into consideration, the temperature dependence can be observed, showing different trends for temperatures above and below the room temperature.

When operating above room temperature, the failure rate increases with temperature, showing the same trend as the experiments. This result suggests that the SA offsets caused by mismatch are more effective and result in higher

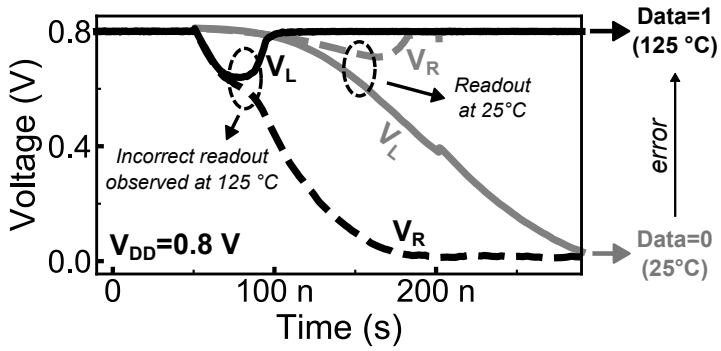


Figure 4.16: An example of a SA simulation showing an error during one read cycle at 125°C. Simulated transient voltage at SA nodes V_{L} and V_{R} (see Figure 4.4) at V_{DD}=0.8V and 25°C/125°C are shown. In these simulations, the input current difference from the PUF cell is set as a constant, in order to clearly show the temperature dependence of the SA. © 2019 IEEE.

instability at elevated temperatures. It can be reconfirmed by observing the transient behavior of a SA, as shown in the simulation result of Figure 4.16. When reading out the same PUF cell without temperature dependence, the difference between two internal nodes, V_{L} and V_{R}, is smaller at 125°C than at 25°C, making the data readout more sensitive to both noise and offsets.

On the other hand, the worse stability for temperatures below room temperature is not observed in measurement. One possbility is that there is no observation due to sample size, but this is not likely the case since a clear effect caused by offset can be observed in Figure 4.13. Another hypothesis is that the leakage current might become lower as temperature decreases. As shown in Figure 4.15, when the R_{leak} (see Figure 4.6) is calibrated based on the simulation temperatures (the blue curve), the failure rates below room temperature are decreased, in closer agreement to the measurement results.

4.4.3 Speed and energy efficiency

The maximum measured throughput is 40Mb/s, for all tested conditions. Note that the design has not been optimized for speed, as speed is not a critical specification for a PUF. Except in the PUFs with TMV function, the read operation happens at most once per key generation procedure, which is less frequent than in other circuits, and thus, the operating speed is less important for PUFs. The average energy consumption for one PUF bit readout is 51.8fJ at V_{DD}=0.9V, when achieving the maximum throughput.

4.5 Statistical properties

The quality of PUF data is typically examined by their statistical properties, and the most important figure-of-merit are randomness, uniqueness and spatial correlation.

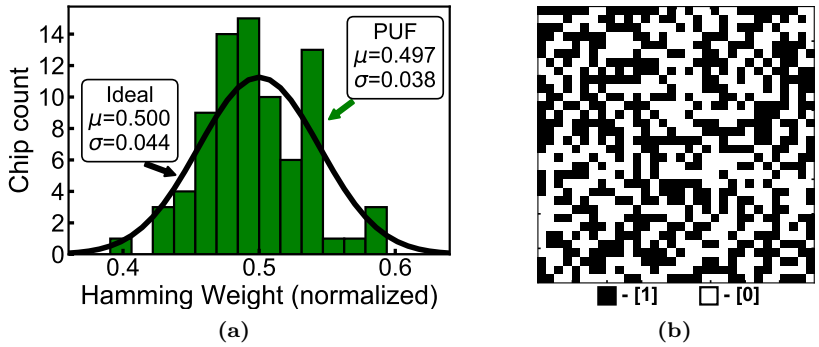


Figure 4.17: (a) Normalized hamming weight distribution of the 128-bit PUF words generated from 20 PUF arrays and (b) an example PUF data. © 2019 IEEE.

4.5.1 Randomness

The randomness of a PUF can be understood as the difficulty to predict the PUF data. As discussed in subsection 2.3.3, for a PUF with the ideal randomness, an adversary has the minimum probability to successfully predict the PUF data no matter how many PUF bits are revealed. For randomness, the hamming weight is one of the most used indices. Figure 4.17 (a) shows the normalized *hamming weight*, i.e. percentage of "1"-bits within a bit sequence, distribution of the 128-bit words measured from 20 PUF arrays. With the assumption of equal probability of 0.5 for the ideal case, the resulting statistics will follow a binomial distribution with n=128 and p=0.5, as plotted in the black line. It should be noted that there is some discrepancy between this ideal distribution and a normal distribution, especially for a small number of bit cells. Since the experimental results show no obvious discrepancy from the ideal distribution, it can be considered that the randomness of the proposed PUF is close to the ideal case.

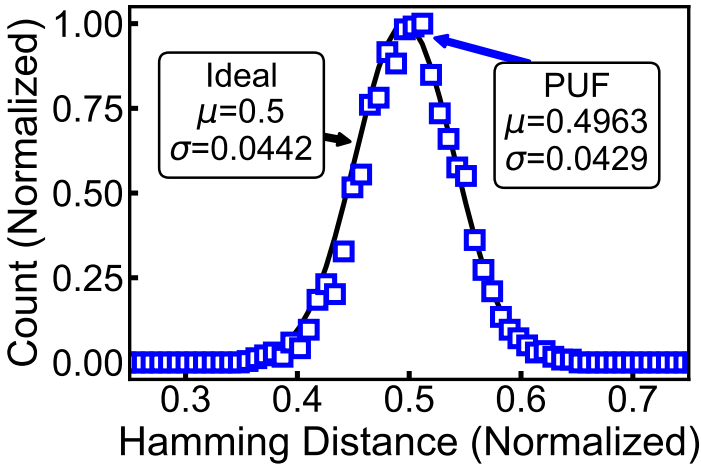


Figure 4.18: The inter hamming distance resulting from 160x 128-bit words (20 PUF chips), showing an uniqueness nearly indistinguishable from the ideal PUF. © 2019 IEEE.

4.5.2 Uniqueness

A PUF is considered unique if it cannot be better predicted when given the data from other identically manufactured PUFs, as previously discussed in subsection 2.3.2. It can be checked by the hamming distance between PUFs (HD_{inter}), which is the total number of bits with different values in two bit sequences. The resulting hamming distance from the 128-bit words of 20 PUF chips are plotted in Figure 4.18, which follows closely the ideal binomial distribution with n=128 and p=0.5. This result confirms that the uniqueness of the proposed PUF is also close to the ideal case.

4.5.3 Spatial correlation

The auto-correlation function (ACF) of PUF data, from the example shown in Figure 4.17 (b), is also computed and plotted in Figure 4.19. All the data sequences have passed the requirement of the auto-correlation test specified in AIS31-T5 [48], that checks the bitwise correlation. As a result, we conclude that the PUF bits are uncorrelated, and hence no spatial correlation within PUF chips exists.

As a conventional figure-of-merit for spatial correlation, the 95% confidence

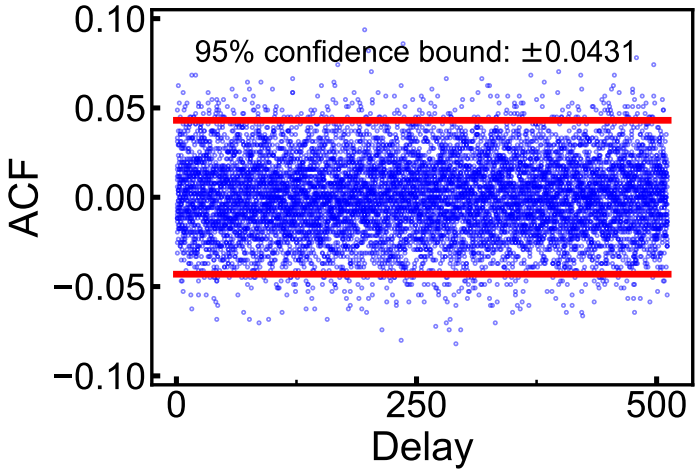


Figure 4.19: The auto-correlation function (ACF) of the PUF arrays with the indication of 95% confidence bound, which shows no observable spatial correlation. © 2019 IEEE.

bound of the ACF is 0.0431, which is close to the ideal value of 0.0433. For truly random bitstreams with a length n, the 95% confidence bound of this ideal case can be computed using equation 4.8, which is derived based on the properties of a binomial distribution.

$$95\%\ confidence\ bound = \pm\sqrt{\frac{0.5}{n}} * 1.96 \tag{4.8}$$

4.5.4 NIST statistical tests

As widely used in a lot of prior work, the measured PUF data is also tested by the NIST 800-22 suite [75], as shown in Table 4.1. All the relevant tests in NIST 800-22 are passed, indicating that the statistical properties of the proposed PUF are indistinguishable from the true random sequences. It should be noted that these tests only confirm the statistical properties of the input bit sequences are indistinguishable to the ideal random sequences, i.e. passing these tests does not imply that the PUF data is truly random. The purpose of running this type of test can be seen as a sanity check, showing that the statistics of PUF data does not have any obvious issue.

Table 4.1: NIST 800-22 Statistical test results (© 2019 IEEE)

Test	Length	Pass % (20 runs)	Avg. p-value	Pass?
Frequency	1024	100%	0.61077	YES
Block frequency	1024	100%	0.65781	YES
Runs	1024	100%	0.47959	YES
Longest runs	1024	100%	0.54749	YES
DFT	1024	100%	0.49642	YES
Nonoverlapping template (m=7)	1024	100%	0.43347	YES
Serial (m=5)	1024	100%	0.42506	YES
Approx. entropy (m=4)	1024	100%	0.53746	YES
Cumulative sum	1024	100%	0.54989	YES

Regarding randomness evaluation for PUFs, the more up-to-date NIST 800-90B test standard [83] is usually inapplicable, because it requires the tested bit sequences to have at least a length of one million. In this test standard, there are a list of min-entropy estimation methods, which can be exploited to provide a coarse estimation for PUF entropy. We concatenate the data sequence from 20 PUFs into a longer sequence of 20480-bits. The min-entropy estimation of this bit sequence is listed in Table 4.2. Considering the worst min-entropy of 0.679/bit, we can estimate that for a PUF array of 1024 bits, the min-entropy is roughly 695 bits, which is sufficient for generating two 256-bit secret keys with full-entropy.

Table 4.2: Min-entropy estimation using NIST 800-90B

Test	Min-entropy
Most Common Value	0.9641
Collision	0.7847
Markov	0.9911
Compression	0.6790
t-Tuple	0.8909
Longest Repeated Substring	0.9612
Multi MCW Prediction	0.9998
Lag Prediction	0.9789
Multi MMC Prediction	0.9845
LZ78Y Prediction	0.9687

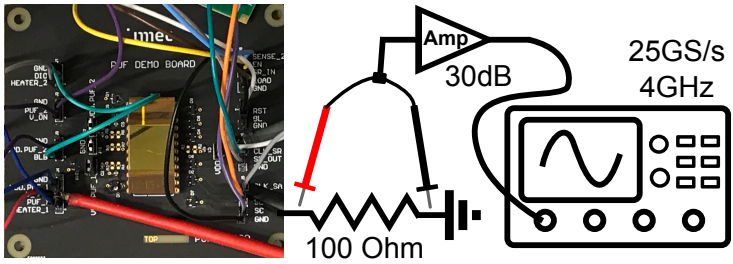


Figure 4.20: The experimental setup for side-channel evaluation, which can sense the power consumption during PUF operation. © 2019 IEEE.

4.6 Side-channel evaluation

Side-channel attacks and fault attacks have been considered as possible threats for PUFs [30, 63, 91]. Despite most of the prior work in this field is mainly about attacking the delay-based strong PUFs, the memory-like weak PUFs (e.g. SRAM PUF) can be also vulnerable to this type of attacking schemes, as discussed in [91]. Consequently, the resilience against side-channel attacks of the proposed PUF is also in our interest, and we have performed the following experiments and analysis to gain a better understanding on the possible threats introduced by the side-channel leakage.

4.6.1 Evaluation method and platform

By surveying multiple side-channel attack schemes, the simple power analysis (SPA) described in [49] has been chosen for performing actual experiments. Apart from the widely used differential power analysis [49] and the correlation power analysis [9], SPA does not require the information of the input data, which does not exist during the data readout operation of the soft-BD PUF.

The concept of this attack is to gather the traces of the transient power consumption during PUF readout, and to extract relevant leakage information from these traces. The goal is to use the information to predict the actual PUF data. Once the prediction is closer to the actual PUF data than random guessing, the attack can be considered as effective and the PUF circuit would have certain vulnerability against this attack scheme.

A simplified illustration of the experimental setup is shown in Figure 4.20. The current flowing out of the common ground from the PCB is converted to a

voltage signal by a 100Ω resistor, and the probed signal is amplified by a 30dB voltage amplifier before sending it to the oscilloscope (Tektronix DPO 70404C).

4.6.2 Analysis of power traces

In order to do statistical analysis, the PUF readout is performed 5,000 times for each PUF row, and the corresponding power traces are recorded. Figure 4.21(a) shows the averaged power traces of the 32 rows of a PUF array, each of them is averaged from 5,000 traces. This figure is zoomed into the time frame where the switching events of the sense-amplifiers are happening. During these switching events, currents are dynamically drawn from the supply, introducing significant current fluctuations which are more observable than the static current. Consequently, it is more probable to find relevant information by focusing on this time frame.

Parallel PUF readout

During each PUF readout event, a word-line is enabled, and the data of the corresponding 32 PUF cells are generated by the SAs in the same clock cycle. Because PUF readout is performed in parallel, the power traces are more likely to reveal the information that is affected by all PUF cells within the same row, and the most straightforward one is the *hamming weight* of each 32-bit word. As a result, our first attempt is to extract the information about HW using these power traces.

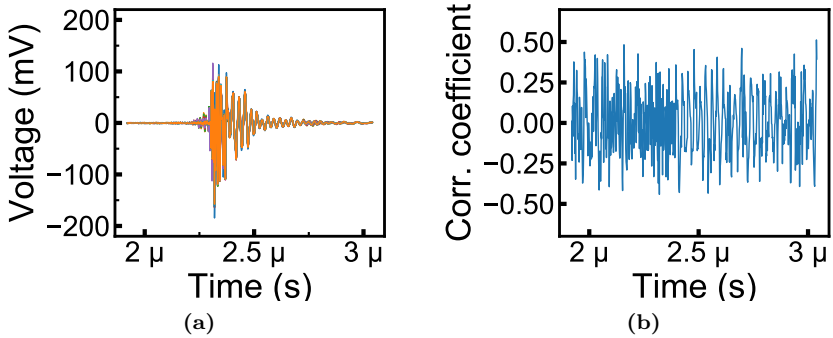


Figure 4.21: (a) Averaged power traces for the readout of 32 PUF rows and (b) correlation coefficients computed at every time sample.

For every time sample in the power traces, we compute the correlation coefficient of the power consumption and the hamming weight, as shown in Figure 4.21 (b). Assuming the power consumption of reading a "1"-bit is higher than reading a "0"-bit, the total power consumption of reading a 32-bit PUF row will be larger for a higher hamming weight. If this assumption is true, we should find a positive correlation between these two sets. Conversely, if the power consumption of reading a "0" is higher than reading a "1", a negative correlation will be found. By computing the correlation coefficients, we can verify if one of the assumptions is valid and to know if one can gather relevant information based on the targeted side-channel leakage.

Most of the data points in Figure 4.21 (b) have small absolute values, showing that it is difficult to extract relevant information using these data points. There are, however, few data points that have correlation coefficient with absolute values around 0.5. One could suspect that these time samples are leaking more side-channel information, since a correlation coefficient of 0.5 usually indicates a strong correlation. This result is, however, more likely because of the small sample size of 32. The critical value of the correlation coefficient for a 99.9% confidence interval is 0.554 for a sample size of 32. The computed correlation coefficients at all data points are within this boundary, and hence there is no strong indication that the power consumption and hamming weight are correlated.

The result of this first experiment shows that based on the power traces while reading out the entire PUF row, it is difficult to determine whether this particular PUF row has a lower or a higher hamming weight. It is possible that the difference of reading a "1" and a "0" is not easily distinguishable, but it is also possible that the parallel readout is too noisy. Consequently, this brings us to perform the next experiment, as discussed in the following subsection.

Individual PUF readout

Despite this PUF circuit is designed to perform only parallel PUF readout, it is still interesting to check the side-channel resilience if an adversary has the ability to violate this operating principle. Consequently, in this experiment setup, only one SA will be activated during a PUF readout cycle, which is done by connecting all the other BLs and BLBs to ground. In this case, the unselected SAs will not generate data corresponding to the PUF cells, and hence are expected to have less interference to the power consumption.

In this scenario, it is more likely to extract individual PUF bits, rather than only the hamming weight of 32-bit words. Take the first PUF row as an example, the 32-bit data of this row is set as the reference data. By selecting only one

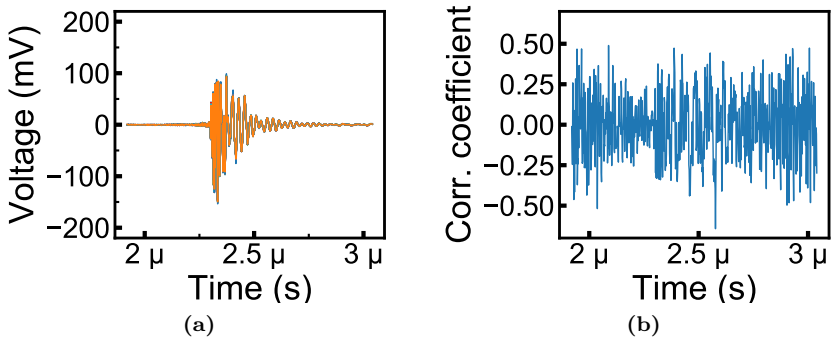


Figure 4.22: (a) Averaged power traces for the readout of 32 PUF columns and (b) correlation coefficients computed at every time sample.

SA at a time, the corresponding power traces of the 32 column readouts from the first row are recorded for 5,000 times. The averaged power traces are shown in Figure 4.22 (a).

Following the same approach, the correlation coefficient of the power consumption and the reference data are computed for every time sample, as shown in Figure 4.22 (b). In general, no strong correlation is observed in the result. There is one particular time point which exceeds the confidence bound of 0.554, indicating there may be a stronger correlation at that time. By performing the same experiment on different PUF rows (not shown), the correlation coefficients are also mostly within the confidence interval. There are also some peak values exceeding 0.554 in these experiments, but they are not located at the same time point for different PUF rows. Based on these observations, we can say that this second approach is indeed more powerful than the first one, since stronger correlations are found. It is, however, still difficult to attack these PUF chips, because there are only few time samples that might leak side-channel information, and the location varies for different PUF rows. More investigation is needed to proof if this method is effective, but will not be covered in this thesis.

4.6.3 Discussion

The side-channel analysis of this work does not show any clear vulnerability against SPA. It is most likely a benefit from the differential readout scheme, which makes the current consumption insensitive to the readout value. To show the difference between single-end and differential readout, a conceptual

simulation is performed. In this simulation setup, the same SA is used in a single-ended configuration. Instead of connecting BL and BLB to the corresponding SA inputs, only BLB is connected to the SA and the BL is left floating. The other terminal of the SA is then biased by a fixed reference, and hence the SA will sense the current difference between BLB and the reference to produce output bits.

As shown in Figure 4.23 (a), the current peak when reading out a “0” locates apart from the current peak when reading out a “1” for the single-ended configuration. The differences between the “1”-curves are because of the non-identical current characteristics of the simulated PUF cells.

For differential readout, the timing of these peak values are less distinguishable for traces of reading “0”s and “1”s, as shown in Figure 4.23 (b). It should be noted that the differences in peak values are found to be simulation artifacts, and is therefore not considered as a viable side-channel leakage. These results show that for the single-ended readout scheme, there is a higher chance to find out the PUF bits by observing the current waveform during readout. Therefore, in terms of side-channel attack immunity, a differential readout scheme is indeed a better choice.

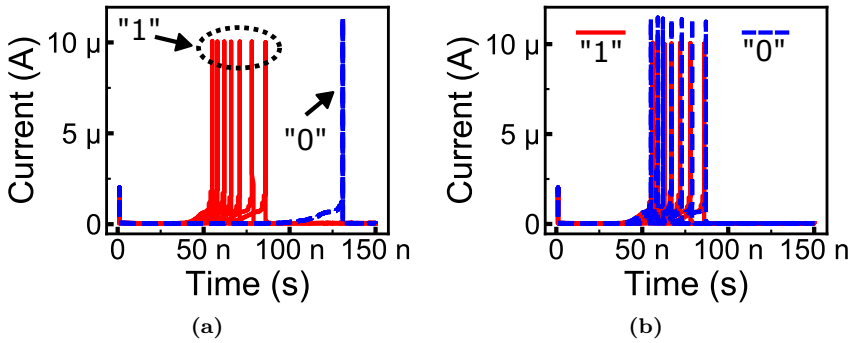


Figure 4.23: (a) Simulated transient current during single ended PUF readout and (b) during differential PUF readout. © 2019 IEEE.

Moreover, the parallel PUF readout also makes the detection of individual PUF bits more difficult, adding extra complexity to performing power analysis. In order to gain better side-channel attack immunity, having multiple SAs operating in parallel is a better choice, comparing to the use of a single SA with multiplexers. While amortizing the SAs has the advantage of less area consumption, and it can also exchange the area reduction for the offset reduction,

the side-channel attack immunity also has to be taken into account when making this trade-off.

There may be doubts about not applying side-channel analysis to the DFF scan chain, which may reveal information as well. We did not perform experiments into this direction, since the PUF data has already entered the domain of standard logic circuits, after being readout by the SAs. For standard logic circuits, the protection techniques against the side-channel attacks belong to other research fields, including the *secure logic design* [82] and *masking* [4]. These topics are outside the scope for this thesis, and they will, therefore, not be experimented and discussed.

It should also be noted that these experiments do not provide solid proofs on the side-channel attack immunity, since it is possible that the right method is not chosen. In the field of side-channel analysis, there are different approaches for analyzing the power traces, and there are also other attacking schemes, such as the electromagnetic (EM) side-channel attack [6, 46]. We did not make a complete survey of attacking schemes since it is not exactly within the scope of this thesis. Nevertheless, the evaluation results did show that the proposed soft-BD PUF is not an easy subject to be attacked.

4.7 Prior art comparison

The proposed PUF is compared with several prior work, in particular the ones with good data stability, as listed in Table 4.3. For this work, there are several advantages that have been mentioned earlier and will be summarized in the following comparisons. There is also no downside among all the listed performance indices. Although the PUF unit cell area of this work is the largest among all listed active PUF implementations, it is not an actual drawback since the area was not a primary concern when designing this prototype. The cell size of 1.5µm by 2µm was chosen to make the custom placement and routing more elegant. Consequently, this unit cell area can still be optimized. As an example of optimization, the soft-BD PUF cell can easily be fitted into a block of 1µm by 1µm, which is only one-third of the original area.

For a general comparison on the unit cell area, it can be seen that the unit cells of the active PUFs ([56, 69, 88, 89] and this work) typically have smaller sizes. It is not only because they benefit from the technology optimization for the memory-like configurations, but also because in these cases the required readout circuits are not taken into account. Therefore, depending on the design of readout circuits and the overall structure, e.g. how many PUF cells can share a readout circuit, the effective area will be increased by a certain amount. In

order to make fair comparisons, the contribution of the readout circuits has to be considered, but the exact amounts are not always given in the literature.

Comparing to the other two work based on oxide breakdown, the soft-BD PUF has a narrower operating range with 0% BER than [88], but the energy consumption is much lower. It should be noted that this is not exactly a fair comparison, since this reference work consists of more peripheral blocks and a larger PUF array of 64k bits. Anyhow, this difference in energy efficiency is well expected, since the current level of a hard-BD spot is significantly higher than a soft-BD spot, especially at low V_{DD}. Consequently, it is confirmed that using soft-BD instead of hard-BD can reduce the robustness against some extreme operating conditions. It however keeps several advantages, including good energy efficiency and stronger immunity against visual attacks. Moreover, we have confirmed that the degrading stability at higher temperature mainly originates from the custom designed sense-amplifier, i.e. not the PUF cell itself. In other words, there is still room for improvement for the proposed soft-BD PUF.

The SA-PUF with hot-carrier injection based stabilization technique [7] shows BER=0% as well, but it necessitates a relatively long burn-in period (25s) to achieve this target. The uniqueness in this case is worse than others, as the average $\mathrm{HD}_{\mathrm{inter}}$ differs from 0.5. The worse uniqueness is most likely due to the mismatches caused by the additional circuitry for HCI burn-in procedure. From the visibility perspective, the transistors with increased HCI degradation have more charged oxide traps, which is not observable by TEM and thus it has the same visual attack immunity as the PUFs using process variations.

The RRAM based PUF in [69, 89] also shows good stability and other PUF properties in a wide operating range, but it necessitates additional process steps to integrate RRAM devices in a CMOS process. Although the energy consumption is not provided in this reference, it is not expected to be low, since the energy consumption in another RRAM PUF work [57] is around 9pJ/bit. With the disadvantage of extra process and power consumption, RRAM PUF is not an ideal solution when targeting a lightweight IoT application. Note that an RRAM PUF has the potential of being reconfigurable, although the reconfigurability is still far from ideal, it would be more suitable for high-end applications. For the visibility, since the RRAMs are all formed with conductive filaments, the resistance change is more like the soft-BD behavior and thus RRAM PUFs has similar visual attack immunity as the soft-BD PUF.

For the SRAM-like hybrid PUF [76], it utilizes burn-in, TMV and dark-bit masking to reduce the instability while keeping an excellent energy efficiency, but this is insufficient to achieve a BER of 0%. The error correcting logic and NVM is therefore not removable for this case. The same argument can be

applied to the mono-stable based design [81], even though the native stability is improved with various techniques, the helper data is still a hard requirement for such case. In terms of energy consumption, the SRAM-like PUFs can possibly win over the Anti-fuse based PUFs even including error correction, but the NVM for helper data storage is still a hard constraint. This type of PUFs using process variations as the source of entropy, has the strongest visual attack immunity, since process variations such as doping concentration are much more difficult to observe. It also should be noted that the visual attack using emission microscopy can observe the PUF data [67], but it is not in our consideration since this technique requires the PUF circuit to be correctly operated, which belongs to a different attacking scenario.

Besides all the advantages of the soft-BD PUF, it should be noted that the additional *forming* procedure to generate the oxide breakdown spots will occupy more resources. It requires either an additional pin or an on-chip high-voltage generator and more testing time, and hence increase the overall cost. The trade-off between this cost and the advantage of reducing embedded NVM needs to be considered when choosing a proper PUF design.

4.8 Conclusion

A PUF utilizing the underlying randomness of the soft oxide breakdown mechanism in MOSFET devices has been demonstrated. The test chip fabricated in a 40nm CMOS process consists of PUF arrays and all necessary periphery circuits. The proposed reference-free sense amplifier is capable of distinguishing the current differences in the nA range, resulting in an excellent bit stability over a wide range of voltage and temperature variations. The statistics of the generated data also show that soft-BD PUF satisfies the fundamental requirements on randomness and uniqueness. Taking the advantage of soft-BD, the PUF circuit consumes less energy per bit and is less visible compared to the PUFs using hard-BD.

The experiment and simulation have also shown several insights for designing a better soft-BD PUF circuit in future work. In summary, it always comes with certain costs when aiming for a better robustness, i.e., 0% BER for a wide operating range. As listed below, there are some important design trade-offs that have to be considered:

- **Robustness v.s. Visibility**: The most straightforward way to improve data stability, especially at lower V_{DD}, is to increase breakdown hardness. This method, however, will increase the possible threats on visibility.

- **Robustness v.s. Energy efficiency**: The increase of breakdown hardness, also implies a higher current consumption during PUF readout, and hence reduces the energy efficiency.
- **Robustness v.s. Chip area**: In order to reduce the negative impact of the SA offsets, the size of SA has to be increased, which has a price of chip area increment.
- **Robustness v.s. SCA immunity**: By amortizing the SAs, the offset can be reduced while keeping the same chip area. In this case, the complexity of performing side-channel attacks is, however, also reduced.

Despite the fact that there is still room for further improvements, it has been proven that the soft breakdown mechanism is suitable for high quality PUF implementations, in particular for the on-chip stable key generation without error correction. For the applications with higher concerns on invasive and side-channel attacks, the proposed soft-BD PUF is even a more viable candidate which provides extra protection.

Table 4.3: Comparison Summary with Prior PUF Work

	CHES' 13 [7]	VLSI' 10 [56]	ISSCC' 18 [88]	VLSI' 16 [89]	ISSCC'19 [69]	JSSC' 17 [76]	JSSC' 18 [81]	**This work**
Design	SA	Anti-fuse	Anti-fuse	RRAM	RRAM	Hybrid cell	Monostable	**Soft-BD**
Technology	65nm	65nm	55nm	40nm	130nm	14nm	40nm	**40nm**
Cell area (μm^2) [1]	7.87	1.13	0.35	0.1	0.27	15	5.83	3
Readout circuit req.	no	yes	yes	yes	yes	no	no	yes
Stabilizing method	hot-carrier injection	native	native	native	native	delay-hardened	native	**native**
BER (%)	0	0	0	0.49	0	1.46	3.2	**0.02 / 0**
Number of cells [1]	1,600	1,792	4,032k	N/A	8k	1024	3k	**$10k^2$ / 1k**
V_{DD} (V)	0.8–1.2	1.0–1.2	0.81–1.32	1.0–1.2	1.4-2.2	0.55–0.75	0.8–1.0	**0.9–1.5**
Temp. (°C)	-20–85	0–85	-40–150	-40–125	25-125	25–110	-40–125	**-20–120**
HD_{inter}	0.468	0.501	0.50	0.4985	0.4999	0.486	0.491	**0.496**
Energy/bit (fJ) [3]	N/A	340	5200	N/A	N/A	4	56.5	**51.8**
Visual attack immunity [4]	Strong	Weak	Weak	Normal	Normal	Strong	Strong	**Normal**

[1] The bit-cell areas are normalized to a 40nm technology utilzing the typical technology scaling factor.
[2] Number of PUF cells used to evaluate BER and unstable bits, which may not be the same as the number of all measured PUF cells.
[3] 10k PUF cells are measured without temperature variation.
[4] The energy consumption for reading one PUF bit, which includes the contribution from all peripheral blocks during readout. It is difficult to perform a fair comparison since each work have different number of PUF cells and periphery circuits.
[5] Theoretical analysis based on the assumption that TEM can be applied for such attacks, which is not yet a relevant threat model. Using TEM, the damage from hard-BD is more observable than the damage from soft-BD, and the classical process variation is even more difficult to observe.

Chapter 5

Security analysis on reconfigurable RRAM PUFs

Content Source

The main material in this chapter was previously published in:

K.-H. Chuang, R. Degraeve, A. Fantini, G. Groeseneken, D. Linten, and I. Verbauwhede, "A cautionary note when looking for a truly reconfigurable resistive RAM PUF," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2018, no. 1, pp. 98-117, 2018.

Contribution: Main author.

Because of the stochastic switching nature, the resistive-RAM (RRAM) has been considered as a promising candidate when looking for reconfigurable PUF solutions. It is, however, shown by our quantitative analysis that the studied RRAM PUF implementations do not achieve true reconfigurability.

5.1 Introduction

Using PUFs to generate device unique secret keys has plenty of advantages comparing to the traditional method, i.e., storing secret keys in an embedded

NVM. Besides these advantages, one limitation for this scheme is that the PUF data cannot be updated. When considering this requirement, the concept of *reconfigurable PUF* is introduced. Reconfigurable PUFs have the ability to change the configuration of its entropy source, and hence reproduce a new set of PUF data [52]. The reconfigurability is useful in case the original key has been revealed and needs to be discarded for security reasons. It can also be used in the case when one needs to revoke or update the ownership of a PUF-based token [47, 73, 90].

From another perspective, one may suggest to move back to the conventional method– storing keys in the NVMs, when updates are required. This approach, however, relies on another party to provide the secret key, which may still introduce trust issues. Even though an on-chip TRNG can be as well included, it can be a hardware overhead that is rarely used, since a secret key will not be updated frequently. On-chip TRNGs also need to be protected to ensure the integrity of this scheme. On the other hand, a reconfigurable PUF has the advantage of generating secret keys on-chip without additional hardware, and hence it will be the only hardware block that requires verification and protection for security.

5.1.1 Physically reconfigurable PUF

The reconfigurable PUFs can be categorized into two groups depending on the reconfiguration methods. One is called the *logically reconfigurable* PUF [47, 90] and the other is called the *physically reconfigurable* PUF [11, 52, 73, 94]. The logical reconfiguration is done by changing the control states of logic circuits, which will update the PUF output or challenge-response pairs accordingly. This type of PUF reconfiguration requires additional circuits and algorithms, and the actual entropy source will not be updated under this operation. There are algorithms to ensure that the unused configurations cannot be predicted, and also to prevent reuse of the revealed configurations. However, an invasive attack scheme may be able to bypass this hardware and directly hack into the entropy source.

On the other hand, the entropy source of a physically reconfigurable PUF has different physical structures between configurations. As long as the structural change is independent from the previously formed structures, the renewed secrecy will be unpredictable. In this ideal case, even if an adversary knows the exact configuration of the entropy source, it does not have the ability to revert back to the previous configuration or to predict the next one. Consequently, the physically reconfigurable PUFs are a solution that can provide updated secrecy with the same backward and forward security. In the rest of the chapter, we

will only discuss the physically reconfigurable PUFs, and they will be referred to as reconfigurable PUFs for simplicity.

5.1.2 RRAM-based reconfigurable PUFs

Recently, there are several emerging memory technologies, e.g. RRAM, MRAM and PCM, appearing in the research field and the actual markets. The purpose of these emerging technologies is to replace the traditional memory such as SRAM and FLASH. Since the manufacturing process is not yet mature and these emerging memory elements are usually used in more scaled CMOS technologies, e.g. 28nm, 16nm and beyond, there is more variability in these devices. The feasibility and reliability studies of these memory elements have revealed the possibility of them being used as PUFs exploiting the underlying variability [32].

For the case of reconfigurable PUFs, there are also candidates in the family of emerging memories. One implementation is based on phase change memory (PCM) [93, 94] and another on resistive random access memory (RRAM) [11]. Especially RRAM has drawn a lot of attention in recent years, and PUFs implemented by RRAM are widely discussed. One of the important facts about RRAM operation is that in each programming cycle, the physical structure within the RRAM element is profoundly changed, making it a viable candidate for implementation as a reconfigurable PUF.

5.1.3 Operating principle of reconfigurable RRAM PUF

In order to understand the potential of reconfigurability in the RRAM PUFs, we first show the construction flow of the conventional PUF comparing to the RRAM PUF. The conventional PUFs such as the SRAM PUF harvest and amplify the intrinsic process variation on the metal-oxide-silicon (MOS) transistors as illustrated in Figure 5.1(a). The PUF behavior originates from the transistor-to-transistor variation, resulting in the cell-to-cell variation (randomness) and the chip-to-chip variation (uniqueness). The result is then readout as the PUF response in digital format, and the non-ideal read-to-read variation is introduced at this stage. Regarding the physical reconfigurability, the transistor-to-transistor variation, which determines the PUF response, will not change except for long-term aging. Therefore the PUF array cannot be reconfigured per request.

The RRAM-based PUFs rely not only on the variation induced by the fabrication and forming process, but also on the variability within the RRAM cells during the programming phase as shown in Figure 5.1(b). If we do not consider

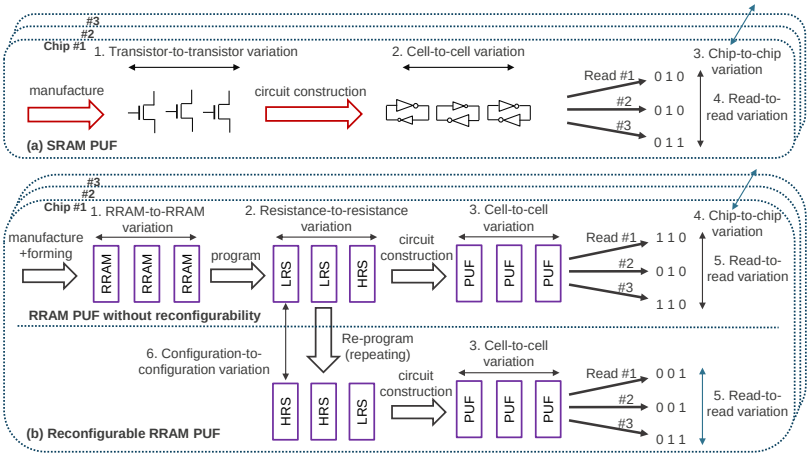


Figure 5.1: The operation flow and different types of variation on (a) conventional SRAM PUF and (b) reconfigurable RRAM PUF

reconfigurability, the resulting resistance-to-resistance variation is similar to the transistor-to-transistor variation of the SRAM arrays. With any particular circuit configuration, the variation of this resistance can be transformed into PUF responses. Several research papers have shown that the RRAM-based PUF can provide enough randomness and uniqueness [57, 89]. The read-to-read variation is also small due to the nonvolatile property, resulting in a good stability.

The reconfigurability of the RRAM PUFs is enabled if the RRAM devices can be reprogrammed into new resistive states, either the low-resistance state (LRS) or the high-resistance state (HRS), as shown in Figure 5.1(b). The switching variability of the RRAM devices introduces the configuration-to-configuration variation, making the new configurations and the resulting PUF responses different to the original ones. Since the PUF properties of the non-reconfigurable part have been well-studied, we will focus on the uniqueness between different configurations from the same RRAM array.

These definitions will be further used in the remaining contexts and are summarized as follows:

1. **RRAM-to-RRAM variation**: The structural differences after the RRAM array is fabricated and formed, which will not change after normal operations including the programming and reading as a memory device.

2. **Resistance-to-Resistance variation**: The variation between the resistive states of the devices in an RRAM array after a programming step.

3. **Cell-to-Cell variation**: The variation between PUF bits generated by the unit cells constructed by the programmed RRAMs. It represents the *randomness* of the PUF and the ideal case is that every cell has a 0.5 probability to differentiate from one another.

4. **Chip-to-Chip variation**: The variation between the PUF bits at the same location from different PUF chips. It represents the *uniqueness* of the PUF and is normally assessed by the *inter-chip hamming distance*, as discussed in subsection 2.3.2.

5. **Read-to-Read variation**: The variation between different PUF readouts, which represents the *stability* of the PUF. In the ideal case, the hamming distance between readouts should be zero. The SRAM PUFs have higher read-to-read variations comparing to the NVM-based PUF.

6. **Configuration-to-Configuration variation**: The variation between the old and new configurations after the RRAM array is re-programmed, which determines the uniqueness between the reconfigured key materials. It can also be assessed by the *inter-configuration* hamming distance, and the ideal distribution has a normalized mean equal to 0.5. The focus of this chapter is to show that the RRAM PUFs cannot approach the ideal case. This is shown by experiments and explained with physical modeling.

5.1.4 Contribution

- First, we will summarize several PUF implementations using filamentary oxygen vacancy-based RRAM from literature. We unify them using a RRAM model to show the PUF behavior and the possible true reconfigurability.

- Second, we describe the two types of variability existing in this RRAM, showing that the configuration-to-configuration variation enables the reconfigurability and the RRAM-to-RRAM variation is limiting the reconfigurability.

- Finally, using the physics-based hourglass model, quantitative analysis on how reconfigurability is degraded with the RRAM-to-RRAM variation is demonstrated.

5.1.5 Chapter organization

This chapter is organized as follows. Section 2 introduces the oxygen vacancy-based RRAM and shows how it can be modeled. Section 3 describes five RRAM PUF implementations that are possibly reconfigurable. Section 4 discusses the RRAM-to-RRAM variability and shows the impact on the reconfigurability. Section 5 provides quantitative analysis of the uniqueness between the reconfiguration cycles. Section 6 shows the loss of min-entropy caused by non-ideal reconfigurability. Section 7 summarizes the uniqueness degradation and compares different PUF implementations. Section 8 discusses three possible methods to correctly use the non-ideal reconfigurability. Section 9 concludes this chapter.

5.2 Concept and modeling of the RRAM

The oxygen vacancy-based resistive random access memory has demonstrated robust scaling ability down to 10nm and promising performance and reliability [1]. The concept of RRAM operation relies on the voltage-controlled resistance modulation of a conductive filament that is formed in the dielectric material of a metal-insulator-metal (MIM) stack. In the dielectric stack shown in Figure 5.2, the oxygen vacancies or the charged oxygen ions are identified as the mobile defects, forming the conductive filament [32]. In this chapter, we will focus on oxygen vacancy-based RRAM since we have sufficient understanding and have the ability to model the stochastic behavior within the filament.

5.2.1 The hourglass model for RRAM switching

The RRAM model used for device and circuit level analysis was published in [21,23,24]. The physics-based hourglass model describes the set/reset transient and captures all the main operation features of oxygen-based RRAM devices, including the statistical and stochastic behavior. As summarized in [24], the hourglass model has five basic ingredients:

- An electron conduction model for describing the current voltage characteristics, based on the quantum point contact model [64, 84]
- A structural model describing the shape of the filament
- A kinetic model describing the vacancy movement inside the filament

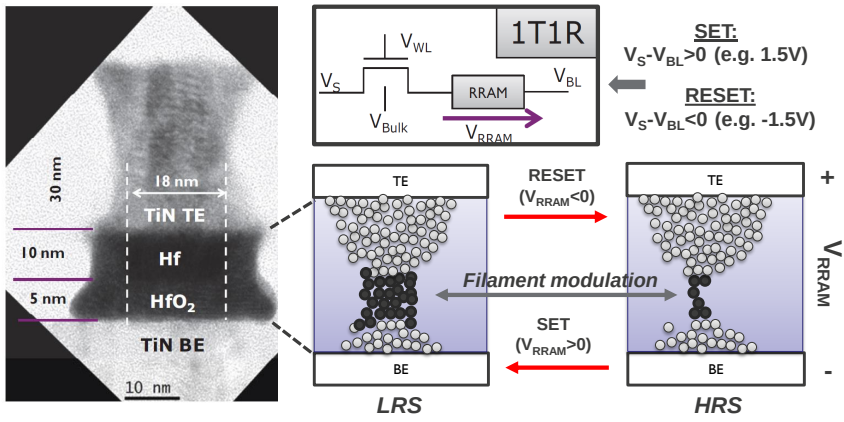


Figure 5.2: The cross section of an oxygen-based RRAM and the concept of filament modulation resulting from set/reset operations. The RRAM devices using in the experiments are in a one-transistor one-resistor (1T1R) configuration. The black dots in the right-bottom figure show the mobile defects located in the current-limiting filament constriction. The white dots show the mobile defects forming the top and bottom conductive part in the filament.

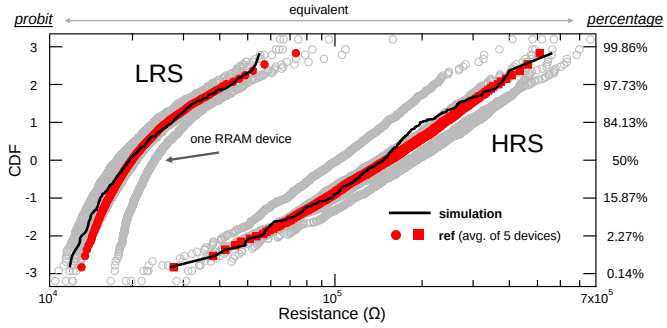


Figure 5.3: The resistance distributions of LRS and HRS over one thousand set/reset cycles of a RRAM, showing a good match between simulation and the referenced measurement data [22]. The vertical axis is shown in probit scale corresponding to the cumulative percentage of a standard Gaussian distribution.

- A thermal model describing the heat generation and its catalyzing effect on switching
- A stochastic model describing the statistical variations in the switching behavior

5.2.2 The RRAM device for experiment and modeling

The RRAM technology used to construct and calibrate the hourglass model is shown in Figure 5.2. The N-channel driving MOSFET has a channel length of 0.13 µm and was fabricated in a 65nm technology, allowing compatible operating voltages for both forming and set/reset operation. The resistive switching stack consists of 65nm physical vapor deposition (PVD) TiN, 5nm atomic layer deposition (ALD) HfO_2, 10nm PVD Hf, and 30nm PVD TiN. The cross-bar RRAM elements are in a one-transistor one-resistor (1T1R) configuration, which is demonstrated to have excellent performance down to 10x10nm [38].

5.2.3 RRAM switching variability

In general, the data stored in a RRAM is distinguished by the conductivity of the filament. There are two basic states: the *low resistance state* (LRS) and the *high resistance state* (HRS) shown in Figure 5.2. The SET operation switches the RRAM element from the HRS to the LRS, the RESET operation switches it from the LRS to the HRS. The resistances of both states are known to be statistically distributed variables [32]. This switching variability is the main feature that caught the attention of PUF-related researchers. The underlying stochastic process can be described by the hourglass model, resulting in an accurate fit to the actual data as shown in Figure 5.3. Note that the distributions of RRAM in this chapter are plotted using the *probit* scale, which is commonly used in the RRAM research papers to plot the *lognormal*-like distributions, so as our reference materials. It has the advantages of showing all data points and emphasizing the tails of the distribution.

A RRAM device has two sources of variability, the first one is the resistance variation between each set/reset cycle, resulting in the configuration-to-configuration variation mentioned in Figure 5.1(b). This variation originates from two sources: (i) the varying number of particles in the filament constriction and (ii) the shape of the filament.

There are also pre-existing variations after the forming process of the RRAM devices, which is denoted as the RRAM-to-RRAM variation. This variation stays in the device no matter how many times the device is reprogrammed. In state of the resistances, the first mechanism is more outperformed than the second one, and therefore the impact of the RRAM-to-RRAM variation is often overlooked when considering the RRAM-based reconfigurable PUF implementations.

5.3 RRAM PUF implementations

In earlier work, several RRAM PUF implementations have been reported [10,11, 57,89]. Even though only a part of these work have claimed reconfigurability, the potential is present in most implementations. Since these implementations are utilizing physical filament modulation as the source of entropy, the underlying stochastic switching behavior can be well exploited. In this section, we will give an overview of these RRAM PUF candidates and study their possible reconfigurability. We will reproduce results of these methods based on the simulation using the hourglass model described in the previous section. The measurement results are only collected from our own technology to support the hourglass simulation, since the real circuits and measurement data of prior work are not accessible.

5.3.1 Resistance variation-based PUF

As proposed in [89], a straightforward RRAM PUF implementation is to directly use the resistance variation. Since the filament of different RRAM cells have different properties such as shape, it results in different resistance following a certain distribution. This method is to define a threshold halfway the resistance distribution, be it either the LRS or the HRS.

Take the case of HRS as an example, the algorithm is illustrated in Figure 5.4. The median resistance R_M is defined as the threshold. Due to the stochastic nature, a RRAM cell can have an arbitrary resistance located in this distribution, which has equal probability of being less or greater than R_M. By defining these two conditions as “1” and “0” respectively, each RRAM cell can provide a PUF bit with equal probability of 50%, depending on which quantization region the resistance does fall in.

This implementation typically results in a good data stability, since the HRS distribution of an RRAM is sufficiently wide. The results in [89] have indeed shown very low native BER, even though there is no readout window between the regions of “1” and “0”.

The reconfiguration of this RRAM PUF can be done by applying one more SET/RESET cycle. The new resistance at the HRS will be remapped onto the same distribution with a different value, and for the same RRAM cell, it may result in a different PUF bit if it moves to another quantization region.

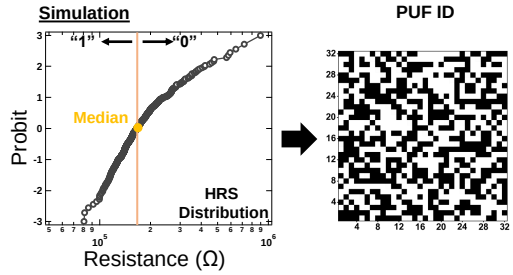


Figure 5.4: The PUF implementation method using the HRS variation. The threshold value is defined by the median resistance (150 kΩ). The right hand figure show schematically an example of binarized PUF data resulting from the resistance comparison.

5.3.2 Split resistance variation-based PUF

The temperature and voltage dependence of RRAM resistance may degrade data stability of RRAM PUF if the resistance is directly compared as in [89]. This reference work proposed a threshold tracking method to find the optimal threshold for different operating conditions. This method requires, however, an additional circuit block to perform the realtime tracking, and the tracking procedure also enlarges the latency of PUF readout.

In order to further enhance data stability, a simpler and more robust modification can be done by the instantaneous resistance comparison and reprogramming procedure [10,57]. As illustrated in Figure 5.5, the cells having a lower resistance, corresponding to "1", receive a SET pulse, thereby making a clear distinction between the LRS ("1"s) and the HRS ("0"s). Since the two states are further split apart, it is referred as the *split* procedure. In this way the RRAM PUF can achieve better data stability while using exactly the same entropy source. In reference [57], the authors do not attempt to reconfigure the PUF, although an identical procedure as discussed in subsection 5.3.1 can be considered.

5.3.3 RRAM PUF based on SET failure

Besides the variability on resistance, the set/reset transient is a stochastic process which has a variable success probability depending on the conditions applied to the RRAM. Specifically, the set failure is examined in [24, 33], showing a clear voltage dependence. A higher SET voltage results in less failure probability and vice versa. This dependency can be well reproduced by the hourglass model,

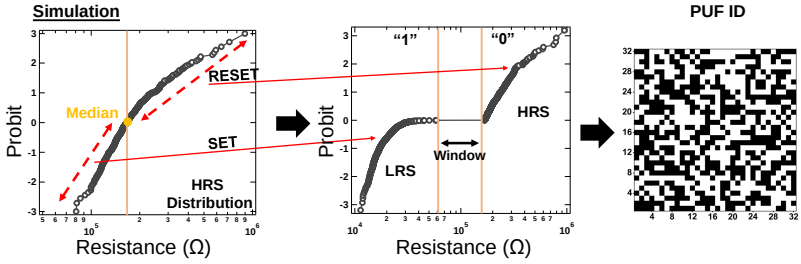


Figure 5.5: The implementation method using the HRS variation with the split procedure. The threshold for the split procedure is also defined as the median resistance (150 kΩ).

as shown in Figure 5.6. Due to this dependence, one particular voltage can result in equal SET success and SET failure probability, and it is defined as the *half-SET* voltage.

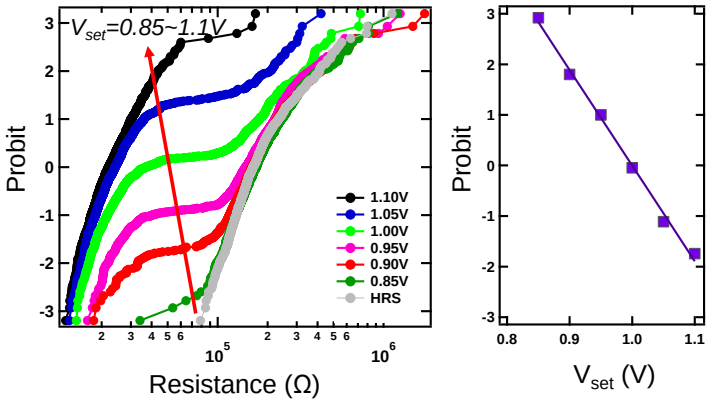


Figure 5.6: A simulation on the resistance distribution after SET operation for different V_{set}. The right figure shows the set failure probability extracted from the left curves.

By applying a half-SET pulse to a RRAM cell at HRS, the resistance has 50% probability of switching to the LRS, resulting in a "1"-bit, and it has the other 50% probability to stay at the HRS, resulting in a "0"-bit. The complete algorithm for this implementation is described in Algorithm 1. Note that before applying a half-SET pulse, each RRAM cell has to be reset to the HRS.

To further improve the data stability, an additional *reinforcement* step can be applied to all RRAM elements. Similar to the split step, a SET pulse with full

Algorithm 1 Algorithm for the RRAM PUF implementation using half-SET

```
R: k-bit RRAM array
R_th: Resistance threshold for digitization
for each cycle ∈ [1, ∞] do
    for each r ∈ R do
        r = Reset(r);
        r = Half_Set(r);
        if r < R_th then
            r = Set(r);
        else
            r = Reset(r);
        end if
    end for
end for
```

strength will be applied to the devices at the LRS and a full strength RESET pulse will be applied to those at the HRS. As shown in Figure 5.7, a readout window will be created after the reinforcement step, which improves the data stability.

A RRAM PUF using half-SET can be reconfigured by repeating the same programming procedure. Since a half-SET pulse does not provides deterministic results to the same RRAM cell, a PUF cell might be successfully switched in one cycle, but the SET procedure fails the next time. Consequently, the new PUF bit is possibly different compared to the previous one, i.e, can be reconfigured.

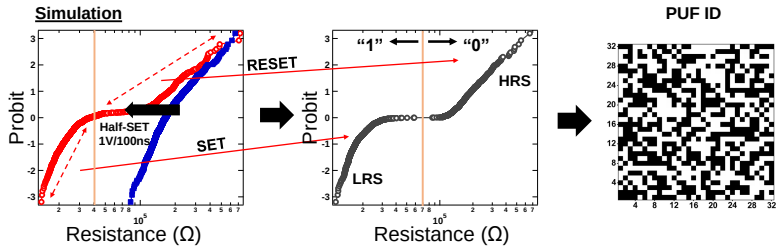


Figure 5.7: The PUF implementation method relies on (i) first applying a SET pulse aiming at half of the population in LRS, (ii) reading and subsequent reinforcement of LRS and HRS aiming at eliminating the unstable bits. The threshold to determine high/low resistance for the reinforcement is 40 kΩ.

5.3.4 RRAM PUF based on multiple SET

Following the previously discussed SET failure phenomenon, there is another possible implementation which was first proposed for an antifuse-based PUF [56]. Similar to the half-SET method, the first step is to apply a SET pulse with reduced voltage or pulse-width (low-SET), resulting in a SET probability sufficiently lower than 50% (e.g. 40% is not a good choice), as shown in Figure 5.8. As illustrated in Algorithm 2, this algorithm then checks the percentage of cells remain at LRS. If this population is larger than 50%, the low-SET pulse is repeatedly applied to the RRAM cells. This loop will continue until the population of LRS is more than the population of HRS, and the resulting resistance distribution of RRAM cells will be close to 50/50 HRS and LRS. The low-SET process is also followed by the same reinforcement step as applied in subsection 5.3.3, to further separate the high and low resistance states.

Algorithm 2 Algorithm for the RRAM PUF implementation using multi-SET

```
1:  R: k-bit RRAM array
2:  R_th: Resistance threshold for digitization
3:  R': RRAMs ∈ R with resistance > R_th
4:  for each cycle ∈ [1, ∞] do
5:      for each r ∈ R do
6:          r = Reset(r);
7:          r = Low_Set(r);
8:      end for
9:      while (No. of R') > k/2 do
10:         for each r ∈ R' do
11:             r = Low_Set(r);
12:         end for
13:     end while
14:     for each r ∈ R do
15:         if r < R_th then
16:             r = Set(r);
17:         else
18:             r = Reset(r);
19:         end if
20:     end for
21: end for
```

The method proposed in this section can be summarized as an *active control* algorithm compared to the method in subsection 5.3.3. It has the advantage of being less sensitive to environmental changes, since the number of required

pulses will be automatically adjusted. However, due to the boundary conditions for terminating low-SET pulses, the resulting PUF data is also subjected to an inevitable bias. The reconfiguration procedure of this implementation is the same as in subsection 5.3.3.

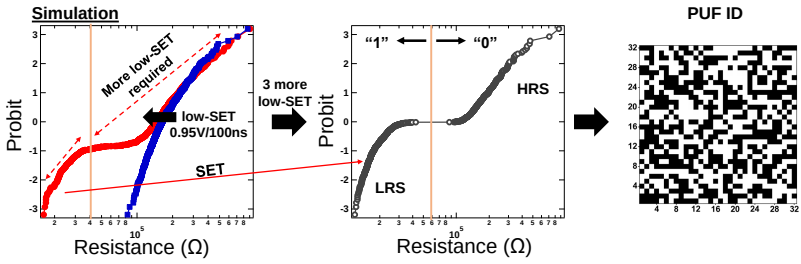


Figure 5.8: The PUF implementation method using multiple SET: (i) first applying a SET pulse aiming at under 50% of the population in LRS (ii) repeating the SET pulses until the population in LRS is sufficiently close to 50%. The threshold to determine SET failure is 40 kΩ.

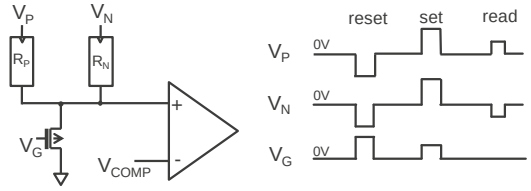


Figure 5.9: Schematic and timing diagram of the RRAM with parallel SET. The readout can be done by comparing the voltage at middle node to a reference voltage.

5.3.5 Coupled RRAM PUF with parallel SET

The last implementation discussed in this chapter is proposed in [5], originally to implement a TRNG, which will continuously generate random bits instead of storing a fixed value. This implementation requires an one-transistor/two-resistor structure as shown in Figure 5.9. Note that the topology must allow for the application of different voltages across two RRAM cells.

As illustrated in Figure 5.9, the two RRAM cells are both initially reset to the HRS. A SET pulse with nominal operation voltage is then applied to both RRAMs in parallel. The voltage will drop equally over the two RRAMs, and

thus, the two RRAMs are under the same SET condition. Since the SET transient is a stochastic process, one of the RRAMs will be switched to the LRS before the other one. As soon as one RRAM is switched, the voltage across both RRAMs will significantly drop due to the voltage divider between the RRAMs and the controlling transistor. The SET process will be nearly terminated since the remaining voltage is insufficient to switch the other RRAM cell to the LRS within the period of the SET pulse.

After the parallel SET procedure, one RRAM cell will be SET to the LRS and the other one will remain at the HRS. As proposed in [5], the output can be read by applying a readout voltage across V_P and V_N. The voltage comparator will give an output of "1" or "0" bit if R_P or R_N is set to LRS respectively. The strong advantage of this implementation is that it requires only one SET step and there is no need to track V_{set} or the median resistance. The 50/50 probability is naturally given if the two RRAM sharing an identical dynamic behavior.

Similar to the PUF based TRNG in [5], this RRAM PUF can also be reconfigured by resetting the two RRAMs to the HRS and start over the parallel SET procedure again. Once the two RRAM cells are both RESET back to HRS, it is uncertain about which RRAM cell will be switched first, and thus, the resulting new PUF bit can be different from the original one.

5.4 RRAM to RRAM variation and its impact on reconfigurability

All the implementations discussed in the previous section rely on the stochastic switching behavior of RRAM, which is assumed to provide completely independent resistance values for every new set/reset operations. In general, the reconfiguration process is done by resetting all cells back to the same resistance state, and apply the PUF generating algorithm again. It should be noted that the resistance distributions shown in the previous section solely include the configuration-to-configuration variation of the RRAM cells. It means that all the RRAM cells are assumed to follow the same resistance distribution when being configured. If this assumption is valid, all of these implementations can be reconfigurable since each element has the equal probability to change state in every new set/reset operations.

Unfortunately, the pre-existing RRAM-to-RRAM variation has an impact on the reconfiguration process. Since the RRAM cells are not identical, the resistance distribution and the sensitivity to SET/RESET conditions will be

different, which introduce device-dependent biases to the RRAM PUFs after reconfiguration. In this section we will discuss how the RRAM-to-RRAM variation affects the reconfigurability of RRAM PUFs.

5.4.1 RRAM to RRAM variation on an array

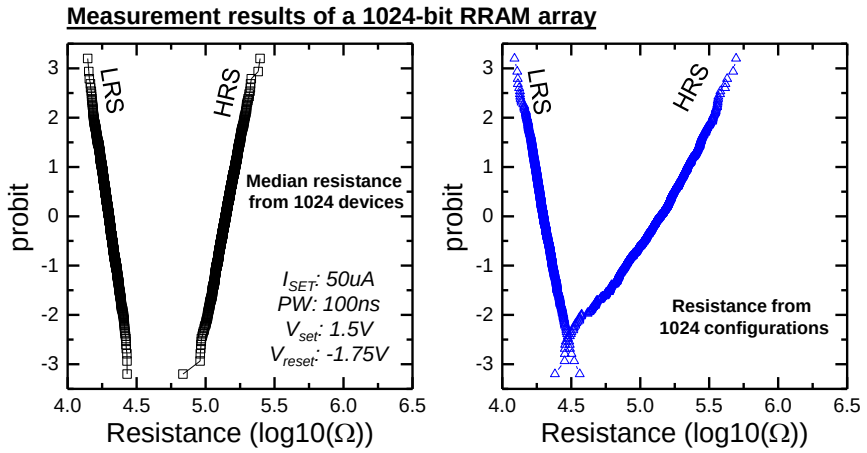


Figure 5.10: The median resistance distribution from 1024 RRAM devices and the resistance distribution from 1024 randomly selected configurations.

The RRAM-to-RRAM variation is investigated using a 1M-bit RRAM array embedded in a CMOS chip. The resistance statistics are extracted by measuring the resistance of 1024 RRAM devices, each for 1024 set/reset configurations. As an important index for the RRAM-to-RRAM variation, the median value of the individual resistance distributions of each RRAM cells are shown in Figure 5.10. The median values represent the fixed differences between RRAM cells, which will not be affected by the stochastic switching behavior. The median resistance in LRS and HRS can both be described by a lognormal distribution, and shows the maximum dispersion within 1024 RRAM cells is about half a decade. When also taking the configuration-to-configuration variation into account, the so-called *"mixed"* resistance distribution becomes significantly wider, in particular for the HRS distribution. Due to this difference, the effect of RRAM-to-RRAM variation is easily overlooked and was not well considered in many prior work.

By observing the median resistance distribution, a reconfiguration problem is immediately identified for the implementations in subsection 5.3.1 and subsection 5.3.2 that directly use the resistance distribution. The threshold in

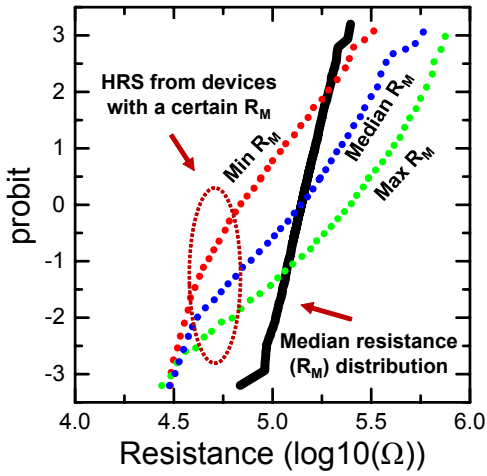


Figure 5.11: The HRS distributions from the devices with the min/max and median of the median resistance. With a fixed threshold, the probabilities to produce "1" and "0" show large deviations between each device, which disprove the reconfigurability of this algorithm.

these two methods is by definition the median value of the mixed resistance distribution. Since this threshold is applied to the entire RRAM array, and the medians of HRS distributions corresponding to each individual RRAM devices are different, the probability of getting a "0" or a "1" is not identical for all devices. As shown in Figure 5.11, the RRAM device with the highest median resistance has only about 13% probability to produce "1"s, in contrast, the RRAM device with the lowest median resistance has about 88% probability to produce "1"s. Consequently, the configuration-to-configuration variation has less impact on the devices with a relatively large or small median resistance, causing a degenerated reconfigurability. A more quantitative analysis will be presented in section 5.5, illustrating the severity in this bias.

5.4.2 Source of the RRAM to RRAM variation

The impact of the RRAM-to-RRAM variation on other implementations cannot be simply observed from the resistance distribution, and hence it is required to first understand the source of RRAM-to-RRAM variation for further investigation. As described in the hourglass model, the resistance of a RRAM is related to the number of vacancies, N_C, in the filament constriction and the

shape of the filament constriction [1, 23]. In general, the more vacancies make up the constriction, the lower the resistance will be.

During the RESET operation, the vacancies are moved out of the constriction with an ion mobility determined by the reset voltage (V_{res}) [32]. This results in a higher resistance when applying a higher V_{res}. One possible cause of the resistance variation is hence the actual V_{res} applied across the RRAM devices. V_{res} can be varied due to the process variation of the controlling transistors, but the V_{res} variation is not likely to be significant, since the transistors are operated in the linear region, which has resistances several orders lower than the RRAMs. Therefore, it is not the main reason of the observed RRAM-to-RRAM variation.

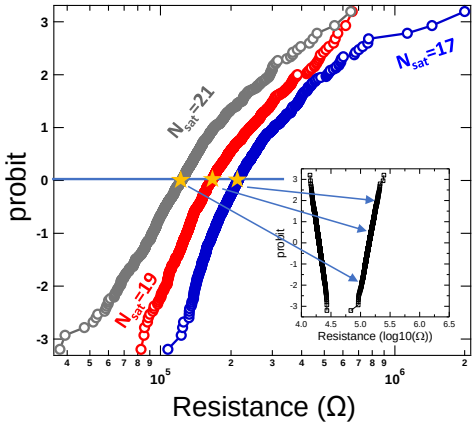


Figure 5.12: The simulated HRS distribution resulting from three different N_{sat} values. The median resistance stay within the measurement result in Figure 5.10.

Another possible cause of the RRAM-to-RRAM variation originates from the filament. It has been shown in the experiments that the HRS will saturate at a maximum value, when the reset voltage is further increased. This phenomenon can be modeled by a minimum saturation number of vacancies N_{C} inside the filament constriction. This number is denoted as N_{sat} in the hourglass model. While simulating the transient switching behavior, the number of vacancies cannot decrease beyond N_{sat}, in both the intermediate and final results.

Figure 5.12 shows the hourglass simulation of the HRS distributions for three different N_{sat} values. The HRS distributions are left-shifted with the increasing values of N_{sat}. Since the varying median resistance can be well described by N_{sat}, which has a real physical meaning, it can be concluded as the major contributor to the RRAM-to-RRAM variation. Although a discrete parameter cannot

fully describe the continuous-like distribution, as shown in Figure 5.10, it still provides a first order approximation enabling further analysis and comparison.

Following this simulation, the distribution of N_{sat} can be mapped onto the measurement data as also shown in Figure 5.12 (inset). The mapped median resistance and the corresponding probability of finding these N_{sat} values in a device are shown in Figure 5.13. Note that the discrete values of N_{sat} are parameterized to continuous values to enable intermediate fitting.

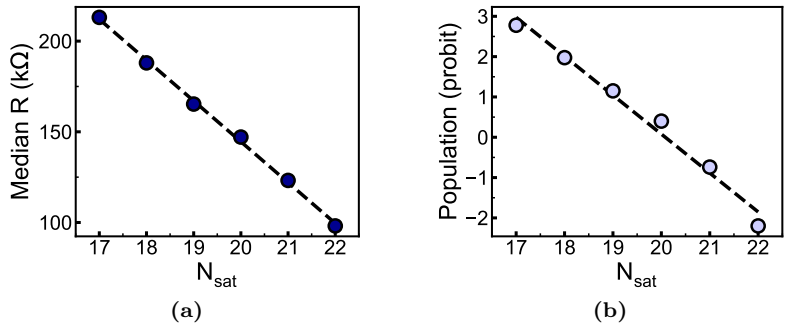


Figure 5.13: (a) The median of the HRS distributions starting from different N_{sat} values. (b) The population (percentage) of RRAM cells with a certain N_{sat} in an array, which is mapped by the real measurement data.

5.4.3 Relation between RRAM-to-RRAM variation and SET failure

Besides having direct impact on the resistance values, the RRAM-to-RRAM variation also has a strong impact on the switching probability, when applying SET with low voltage as is done in the procedures described in subsection 5.3.3 and subsection 5.3.4. Figure 5.14 shows the experimental results of applying SET pulses with different voltage to ten different devices, starting from the HRS with the same RESET condition. There are ten curves per color, and each represents the resistance distribution from a different RRAM device.

As expected, the probability of SET failure decreases with increasing $\mathrm{V_{set}}$. Moreover, there are also significant differences in the SET failure rate across devices. With the small population of only 10 devices, there is already a maximum difference of around 30-40%. In the real cases with thousands of RRAM devices, the deviation can be expected to be even more severe. Note that because of the large number of measurements (100k/condition), there is

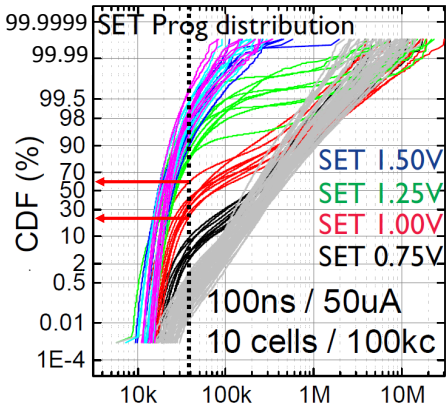


Figure 5.14: The measured SET failure probability of 10 devices with four different V_{set} (figure in [33]). The failure probabilities at 1V show a deviation about 30-40% within only 10 devices. The grey curves are the resistive state before SET operation.

no issue with error bar. Even though the absolute difference between failure probability decreases at the lower or higher voltages, this is not considered as an advantage since the failure probability is too far from 50%, making it infeasible to perform the half-SET algorithm.

The impact on SET failure can also be quantified by the simulations using the hourglass model. Figure 5.15 shows the simulated distributions for V_{set}=1V starting from three HRS distributions corresponding to three N_{sat} values. The probability of SET failure has a clear dependence on N_{sat}. This is in agreement with the experimental results shown in Figure 5.14 and confirms that N_{sat} can describe the RRAM-to-RRAM variation on the switching probability. Due to this mechanism, the ideal reconfigurability no longer holds for the half-SET and multi-SET PUF implementations discussed in subsection 5.3.3 and subsection 5.3.4.

5.4.4 Transition time

The underlying cause of SET failure can be traced back to the variations of the HRS-to-LRS transition time. Both the SET and RESET process are time dependent, i.e. once a SET or RESET voltage is applied, it requires a certain time to grow or shrink the conducting filament, which is referred as the transition

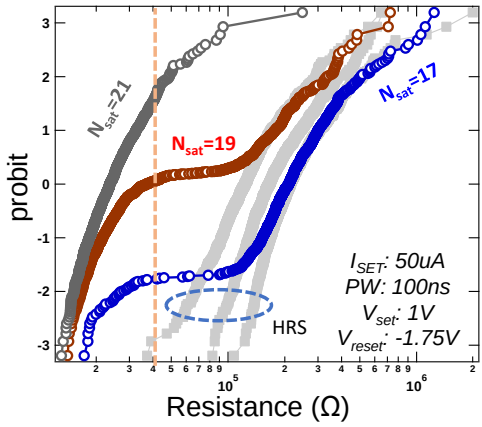


Figure 5.15: Simulation results of the resistance distribution after half-SET starting from different N_{sat} values. The SET failure rate decreases as N_{sat} increases, the threshold to determine a SET failure is 40 kΩ

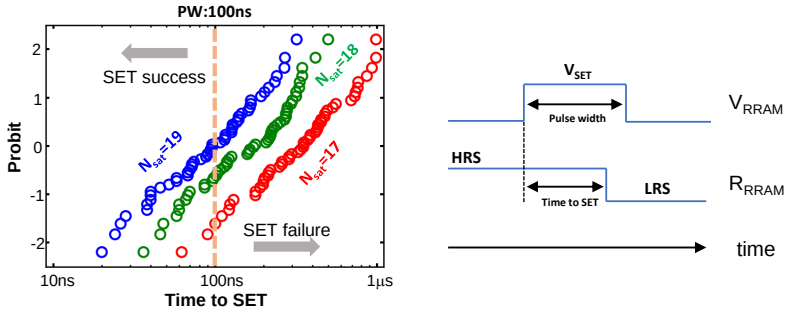


Figure 5.16: Simulated time of the SET transition time starting from different N_{sat} values and the relation to SET failure with a given pulse-width. The relation between the pulse-width and time-to-set is illustrated in the righthand-side. A set-failure can be observed when the time-to-set is longer than the pulse-width.

time. Similar to the time-to-breakdown in oxides [26], the transition times of RRAM are also statistically distributed and show strong voltage dependences. This explains the dependence of the switching probability to the set voltage and the pulse-width (not shown). If the pulse width is shorter than the projected transition time (the actual number is unknown when a transition does not happen), a SET failure will occur.

Moreover, the kinetic model described in [21, 23] shows that the transition time is determined by four time constants, that are inversely related to the initial number of vacancies in the constriction. That is, with more initial vacancies, the transition time is on average smaller. On the other hand, with the same pulse-width, the RRAM devices with more initial vacancies are more likely to successfully switch to LRS. The simulated distributions of transition time with different N_{sat} are plotted in Figure 5.16.

The transition time is the entropy source of the coupled RRAM PUF with the parallel SET mechanism as described in subsection 5.3.5. Ideally, if both RRAMs have identical transition time distributions, the probability to produce "0" or "1" will exactly equal to 0.5. In reality, the two RRAMs are not identical, e.g. one can have N_{sat}=18 and the other with N_{sat}=19, according to the distributions shown in Figure 5.12. Consequently, the one with N_{sat}=19 is more likely to switch faster during the next reconfiguration cycle. Therefore, the coupled RRAM PUF is also biased because of the existence of the RRAM-to-RRAM variation on the transition time.

5.5 Simulation-based reconfigurability assessment

In order to assess the reconfigurability of all of the implementations discussed in section 5.3, different types of 1024-bit RRAM PUFs are simulated. The simulation parameters are calibrated by the real measurement using a RRAM array, as shown in Figure 5.10 and Figure 5.11. Typically, the main performance metrics for PUFs are randomness, uniqueness, and data stability. In particular for assessing a reconfigurable PUF, uniqueness is the most important feature. In other words, the randomness and data stability should always be satisfied no matter a PUF is reconfigurable or not. It has been shown in prior work [10,57,89] that RRAM PUFs have both good randomness and data stability.

For a typical PUF implementation without reconfigurability, the uniqueness is assessed by computing the hamming distance between different chips, so called the inter-chip hamming distance ($\mathrm{HD}_{\mathrm{inter}}$). This type of uniqueness for the RRAM PUFs has also been proven by earlier work, and therefore is not discussed here. For the reconfigurable PUF implementations, the uniqueness is calculated between different configurations, referred to as the inter-configuration hamming distance ($\mathrm{HD}_{\mathrm{config}}$). To maintain both the forward and backward secrecy, an ideal reconfigurable PUF need to have a $\mathrm{HD}_{\mathrm{config}}$ distribution identical to the ideal $\mathrm{HD}_{\mathrm{inter}}$ distribution (50% with normalization), i.e. *reconfiguring the PUF gives the same security as replacing it with a new chip.*

5.5.1 Reconfigurable RRAM PUF modeling

To construct a RRAM PUF array for simulation, it is required to know how the properties of each individual RRAM cell distribute in the array. As characterized in the previous section, the population of RRAM cells with a certain N_{sat} value is characterized, as shown in Figure 5.13 (b). For example, based on the data shown in this figure, it can be calculated that there are around 2.5% of RRAM cells that have N_{sat} below 18.

As the relation between RRAM-to-RRAM variation and N_{sat} is well characterized in subsection 5.4.2, it can serve as the basis of RRAM PUF construction for reconfigurability assessment. For RRAM cells with different N_{sat} values, there are different resistance distributions, set probabilities, and transition times. The N_{sat} dependence of these parameters is first simulated using the hourglass model. As an example, Figure 5.17 (a) shows how N_{sat} impacts the PUF implementation using resistance variation, in which the PUF bits are determined by a fixed resistance threshold. The ratio of "1"-bits has a clear positive dependence on N_{sat}.

Combining the results in Figure 5.13 (b) and Figure 5.17 (a), the corresponding population of RRAM cells with a certain ratio of "1"-bits can be mapped, as shown in Figure 5.17 (b). Similar results for other RRAM PUF implementations can also be computed and used to emulate the actual RRAM PUF arrays for reconfigurability assessment.

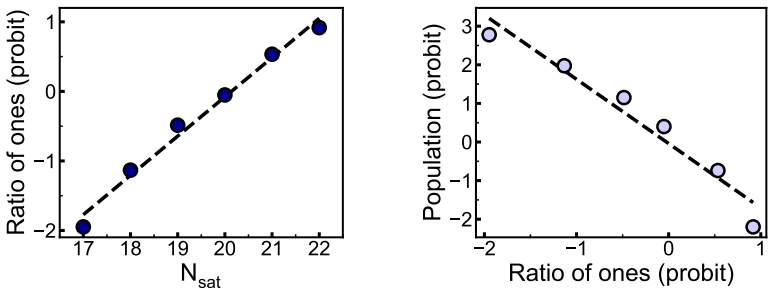


Figure 5.17: The biased probability of producing "0" or "1" bits by comparing the resistance (described in subsection 5.3.1 and subsection 5.3.2) for different N_{sat} values. The bias is mapped to the probability of finding N_{sat} (Figure 5.13), as shown in the right figure. The fitted probability of the RRAM-to-RRAM bias is used for RRAM array simulation.

5.5.2 Reconfigurability using HRS distribution

As shown in Figure 5.12, the HRS distribution is different for different N_{sat}, resulting in biased probabilities to produce "1" and "0" when a fixed threshold resistance is applied. Figure 5.17 (a) shows the ratio of generated "1"-bits with the threshold of 150 kΩ in multiple reconfiguration cycles, as a function of N_{sat}. As described in the previous subsection, this dependency is then mapped to the population of RRAM, as shown in Figure 5.17(b). Using the result of linear fitting, a 1024-bit RRAM array is constructed in the simulation environment.

The inter configuration hamming distance of the simulated RRAM PUF is shown in Figure 5.18, together with the ideal result which does not consider RRAM-to-RRAM variation. It can be clearly seen that the $\mathrm{HD}_{\mathrm{config}}$ distribution is left-shifted, showing a degradation from the ideal case. *An ideal configuration-to-configuration uniqueness is therefore not achievable* using this implementation. It should be noted that the "split" procedure for the RRAM PUFs described in subsection 5.3.2 only has impact on the stability, and thus the resulting $\mathrm{HD}_{\mathrm{config}}$ can be well represented by the same simulation results.

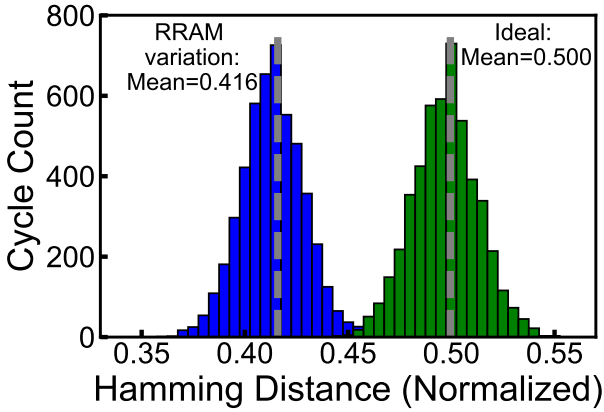


Figure 5.18: The simulated inter-configuration hamming distance with and without RRAM-to-RRAM variability. The $\mathrm{HD}_{\mathrm{config}}$ is shifted lower as more devices are likely to produce the same result after reconfiguration.

5.5.3 Reconfigurability using half-SET

To assess the reconfigurability for implementations using half-SET, the SET success rate with the half-SET voltage of 1V for different N_{sat} is simulated and plotted in Figure 5.19 (a). The threshold resistance to distinguish success and

failure is set as 40 kΩ. Following the same method described in subsection 5.5.1, the population of RRAM cells with a certain success rate can be derived, as shown in Figure 5.19 (b). The linear fitting is used to construct a RRAM PUF array which emulates the actual behavior.

The simulated distribution of inter-configuration hamming distance is shown in Figure 5.20, as well as the ideal case. Similar to the previous case, the mean value is also left-shifted, indicating an even more severe uniqueness degradation as the mean value is more apart from the ideal value of 0.5. The result shows that the half-SET method does not have good reconfigurability, and is more sensitive to RRAM-to-RRAM variation than directly using the HRS variation.

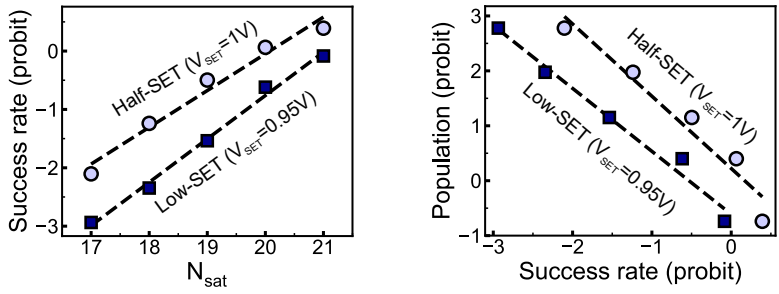


Figure 5.19: The biased probability of producing "0" or "1" bits by the half-SET and multiple-SET algorithms (described in subsection 5.3.3 and 5.3.4) for different N_{sat} values. The fitted probability of the RRAM-to-RRAM bias in the right figure is used for RRAM array simulation.

5.5.4 Reconfigurability using multiple SET

The PUF generation method using multiple SET, described in subsection 5.3.4, uses the same mechanism as the half-SET discussed previously. By applying a low SET pulse, the probability of having a SET failure is increased comparing to the results shown in Figure 5.19. Following the algorithm described in subsection 5.3.4, the PUF data are reproduced for 100 cycles, and the resulting $\text{HD}_{\text{config}}$ with and without RRAM-to-RRAM variation are shown in Figure 5.21.

The $\text{HD}_{\text{config}}$ is the most biased of the three studied cases with 1T1R configurations. This is because the RRAM-to-RRAM variation is amplified by the repeated SET. Note that the mean of the ideal $\text{HD}_{\text{config}}$ is not equal to 0.5, due to the inherent entropy loss discussed in [56].

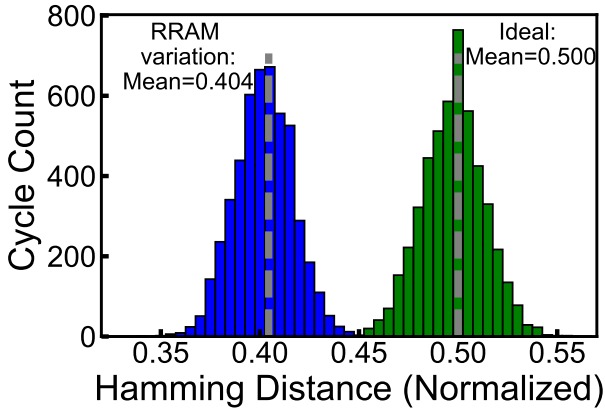


Figure 5.20: The simulated inter-configuration hamming distance with and without the RRAM-to-RRAM variation. The HD_{config} is shifted even lower than the one in Figure 5.18, implies that the RRAM-to-RRAM variation has more impact on the SET failure.

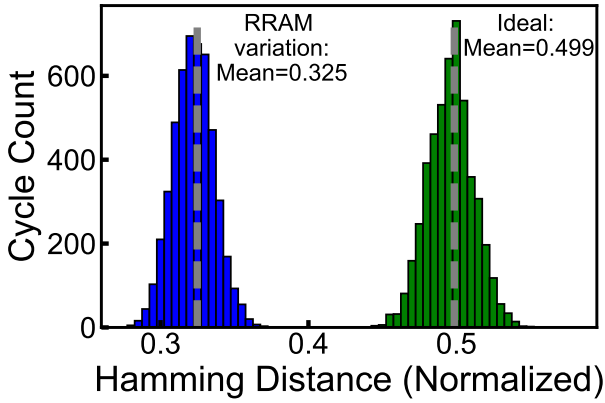


Figure 5.21: The simulated inter-configuration hamming distance with and without the RRAM-to-RRAM variation using the multiple SET algorithm.

5.5.5 Reconfigurability using parallel SET

For the coupled RRAM PUF using parallel SET, as described in subsection 5.3.5, the analysis becomes more complicated since the hourglass model is not simulating the interaction between two RRAMs. In order to simulate the transient of two RRAMs in parallel, we first generate an 1024-bit array of

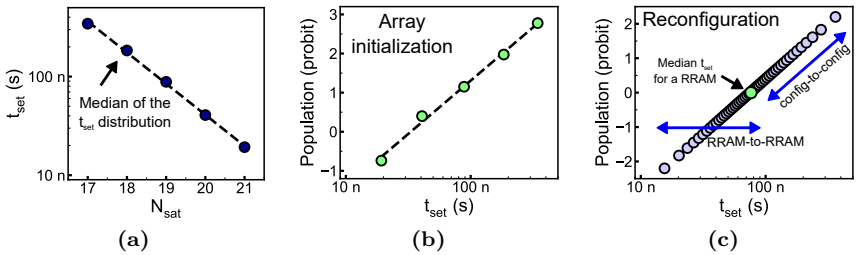


Figure 5.22: (a) The median SET transition time (t_{set}) for different N_{sat} values and (b) the mapping to the probability of finding N_{sat} (Figure 5.13). The median transition time is then used to construct the configuration-to-configuration transition time distribution for each RRAM elements used in the simulation, as shown in (c).

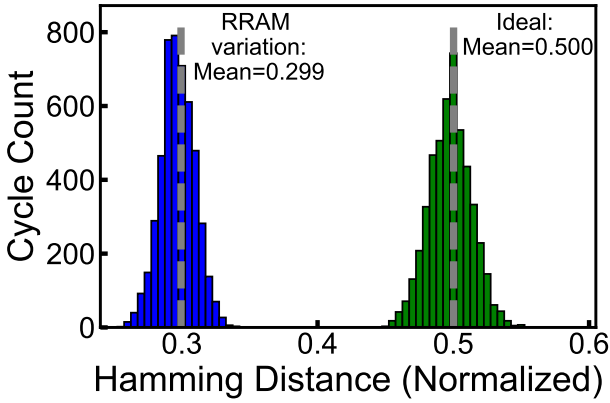


Figure 5.23: The simulated inter-configuration hamming distance with and without the RRAM-to-RRAM variation using the parallel SET algorithm.

coupled RRAMs, where each RRAM device has a median set transition time sampled from the distribution shown in Figure 5.22. In each reconfiguration, the transition time of each RRAM device is sampled from the configuration-to-configuration distribution and compared. If the transition time of the R_P or R_N (Figure 5.9) is lower, the output will be "1" or "0" respectively. With this method we are able to simulate the HD_{config} for 100 configurations with and without RRAM-to-RRAM variation, as shown in Figure 5.23. As observed, this implementation shows the most severe degradation among all implementations.

5.5.6 Effect of correlation

Besides the bias of "1" and "0" bits, the bit-wise correlation is also a major concern while generating random numbers. In the simulations shown earlier, each configuration cycle is assumed to be independent and identically distributed (i.i.d.), i.e., the correlation between the configuration cycles is not considered. For example, if there is a positive correlation, by knowing the current PUF bit is "1", it will be more likely to produce a "1" in the next configuration, comparing to the overall probability.

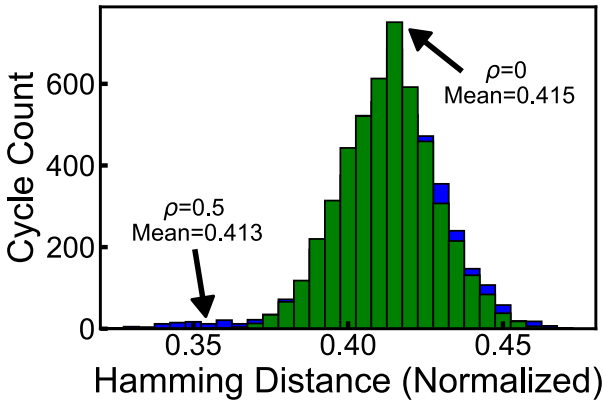


Figure 5.24: The simulated inter-configuration hamming distance based on HRS resistance, with and without the correlation between the resistance of adjacent configurations.

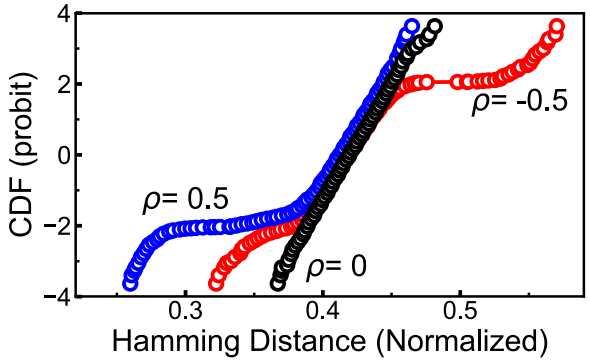


Figure 5.25: The CDF of the inter-configuration hamming distance based on HRS resistance, with three different correlation coefficients.

For RRAMs, there could be a weak correlation between the resistance at the current state and the resistance at the next state. This correlation may as well have impact on the reconfigurability. A similar simulation as discussed in subsection 5.5.2 is performed and shown in Figure 5.24, with correlation coefficients $\rho = 0$ and $\rho = 0.5$. The mean values of the two distributions are almost the same and there is a tail in the distribution with correlation, which is not clear to see in a histogram. By plotting the cumulative distribution function in a probit scale, as shown in Figure 5.25, this tail becomes more clearly noticeable and the discrepancy for different correlation coefficients can be clearly seen. The result shows that the correlation between configurations has also negative impact on the uniqueness, and can be easily overlooked if only given the histogram and mean value of inter-configuration HDs.

5.6 Min-entropy estimation

To describe the difficulty of predicting a random sequence, the term of min-entropy is widely used in cryptographic researches. For reconfigurable PUFs, the min-entropy H_∞ is defined as in Equation 5.1, in which PUF_i is the PUF data of the i_{th} configuration and $p_1, ..., p_n$ are the n possible outcomes of the generated PUF data.

$$H_\infty = -\log_2 \{\max [\Pr(PUF_i = p_1), ..., \Pr(PUF_i = p_n)]\} \tag{5.1}$$

Assuming the PUF cells are independent to each other, for a reconfigurable PUF with N cells, the probability of the i_{th} configuration to be equal to the most probable outcome $p_{\max}$ can be derived as in Equation 5.2, in which $PUF_{i,k} \in \{0, 1\}$ represent the data of the k_{th} PUF cell in the i_{th} configuration and $p_{\max,k}$ is the k_{th} element of the most probable outcome $p_{\max}$.

$$\Pr(PUF_i = p_{\max}) = \prod_{k=1}^{N} \Pr(PUF_{i,k} = p_{\max,k}) \tag{5.2}$$

Since all RRAM cells are independent, the most probable outcome $p_{\max}$ must comprise all the most probable outcome of each RRAM cells. Therefore, the probability of the k_{th} PUF cell can be derived as Equation 5.3.

$$\Pr(PUF_{i,k} = p_{\max,k}) = \max [\Pr(PUF_{i,k} = 0), \Pr(PUF_{i,k} = 1)] \tag{5.3}$$

By substituting the results of Equation 5.2 and 5.3 into Equation 5.1, the min-entropy of the i_{th} configuration can be represented by the maximum probability of having a "1" or a "0" in each RRAM cells, as shown in Equation 5.4.

$$H_\infty = -\sum_{k=1}^{N} \log_2 \{\max [\Pr(PUF_{i,k} = 0), \Pr(PUF_{i,k} = 1)]\} \tag{5.4}$$

With the knowledge on how each RRAM cell will behave, the min-entropy can be simply computed using this equation. For an ideal case, in which all the RRAM cells have equal probability of 0.5 to generate "1" or "0" bits in each configuration, it can be found that $H_\infty = N$. Table 5.1 shows the computed min-entropy using the four algorithms discussed in this chapter, the results are normalized by the number of cells. Comparing to the ideal case of the normalized $H_\infty = 1$, these results show that there is more than 40% entropy loss for the best case.

Table 5.1: Min-entropy of each RRAM PUF implementations

	HRS	half-SET	multi-SET	parallel-SET
H_∞	0.583	0.568	0.468	0.389

5.6.1 Predicting the reconfiguration results

The parameters used in the previous subsection requires detailed characterization, which is unfeasible for a realistic attack scenario. Assuming an adversary only knows there are device dependent biases in each PUF cells, and knows the PUF data of every existing configurations, the best prediction this adversary can make is based on the number of existing "1"s and "0" for each PUF cells. This optimal prediction of knowing l cycles, x_i, can be defined by Equation 5.5.

$$x_l = (x_{l,k})_{k=1}^{N} = (\text{Majority } [(PUF_{i,k})_{i=1}^{l}])_{k=1}^{N} \tag{5.5}$$

Based on the probability of correctly predicting the i_{th} configuration, after knowing the first l configurations. The remaining entropy, H_l, can then be defined by Equation 5.6, which is then used as an index to show entropy loss with respect to the number of revealed configuration.

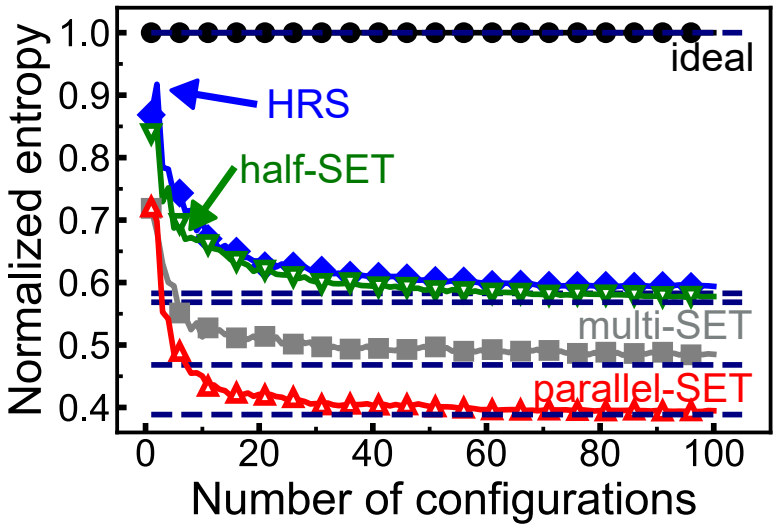


Figure 5.26: The simulated entropy loss v.s. number of reconfiguration cycles. The dashed lines are the asymptotes with the values corresponding to the min-entropy shown in Table 5.1.

$$H_l = -\sum_{k=1}^{N} \log_2 \left[\Pr(PUF_{i,k} = x_{l,k})\right], \ i > l \tag{5.6}$$

Using the same PUF array setup for uniqueness assessment, the PUF data $PUF_1, ..., PUF_{100}$ are generated by applying reconfiguration algorithm. The entropy H_l computed based on Equation 5.6 and plotted in Figure 5.26. The value is normalized by the number of PUF cells, N, which will then have a range of [0, 1]. For all the cases, the remaining entropy will approach to the theoretical min-entropy values. Consequently, after revealing sufficient reconfigured PUF data, an adversary may have the same ability of predicting the next one as if it knows the characteristic of each RRAM cells.

5.7 Comparison and discussion

Following the results of the previous section, it is clear that all the discussed RRAM PUF implementations have non-ideal reconfigurability. Using the same device and modeling methods, these implementations all have different amounts

of uniqueness degradation, which will be discussed and compared with other advantages and disadvantages. We will also introduce recent PUF work which has claimed reconfigurability with measurement data using real circuits.

5.7.1 Uniqueness degradation of different RRAM PUFs

First, we can observe that there is less uniqueness degradation when the comparison is based on the resistance variation, compared to the implementations based on the switching behavior. It shows that the switching instability is more sensitive to the physical parameters of a device, comparing to the resistance variability. In terms of reconfigurability, the RRAM PUF with thresholding method is indeed a better choice compared to the one using half-SET. It, however, necessitates a more precise current comparator to distinguish the resistance, while there will be less requirement for the implementation using half-SET since the current difference is much larger.

For the three implementations based on transient switching behaviors, the uniqueness degradation is also worse for the implementations with more complicated algorithms. Regardless of reconfigurability, these algorithms have other advantages. For the multi-SET algorithm, there is no need to find an optimized SET condition, a lower SET probability may only affect the number of required pulses but not the final results. On the other hand, the SET condition must be carefully chosen for the PUFs using half-SET. The same argument can be applied for the coupled RRAM PUF using parallel SET algorithm, it can work in a wider operating range, e.g. V_{set}=1 or V_{set}=1.5 both will work.

5.7.2 Non-ideal reconfigurability in real PUF implementations

After our research paper [17] about the reconfigurability of RRAM PUFs was published, there are already two papers claiming to have reconfigurable RRAM PUF. In these two work, the configuration-to-configuration uniqueness are both not ideal as expected.

The PUF work introduced in [53] use RRAM cells with multiple current levels to generate multiple bits per cell. The reconfigurability is claimed in this work, however the measurement results show that inter-configuration hamming distance has a mean value of *0.38*, which is far from the ideal case. This problem has been recognized, and a method called "bit-shuffling" is proposed as a solution, which shuffles the generated PUF bits into a different order. The HD_{config} after bit-shuffling indeed has a mean value close to 0.5, it however

does not provide much extra protection, since the bit-shuffling procedure is reversible, which makes it possible to retrieve the original results.

Another RRAM PUF implementation is proposed in [69], which uses coupled RRAM cells as in the "parallel-SET" algorithm. The algorithm to generate PUF data is similar to the "split-resistance" method, it first compares the resistance between two RRAM cells both in the HRS, and then apply a SET to the RRAM cell with lower resistance. This design shows robust results, but the reconfigurability is still not ideal. The mean value of the HD_{config} is around *0.47*, which still shows a non-ideal inter-configuration uniqueness degradation.

5.8 Methods to use non-ideal reconfigurability

From our analysis as well as from other work, it can be concluded that RRAM PUF implementations lead to non-ideal reconfigurability. In the use cases that reconfigurability is a requirement, a reconfigurable RRAM PUF is still a viable candidate, which needs a certain technique to compensate the degraded configuration-to-configuration uniqueness.

5.8.1 Entropy extraction

The first possible solution is rather straightforward, which relies on entropy extraction techniques, such as using a fuzzy-extractor [28] as in the conventional post-processing circuit for PUFs. For example, the RRAM PUF using HRS variation has a min-entropy of 0.583 for each cell, which implies that it need at least $1/0.583 - 1 \approx 72\%$ more cells to extract the same amount of entropy as the ideal case. In summary, this method requires an increased amount of PUF cells and an additional entropy extraction circuit.

5.8.2 Bias masking

The second possible solution is similar to the bit-shuffling method in [53], which shuffles the PUF bits into a different order. In this so-called bias-masking method, instead of shuffling the PUF bits, one can feed the reconfigured PUF data into a cryptographic one-way function, e.g. a hash function, and then store the output of this function back to the RRAM cells. The purpose of this method is not to solve the problem of device dependent biases, but is meant to hide them. Due to the property of the one-way function, each cell has equal probability of storing a "1" or "0", no matter of how the input is biased. By

applying this algorithm, an adversary without prior knowledge on the properties of each RRAM cells, cannot guess the most probable outcome by the prediction method defined in Equation 5.5, the prediction x_l will be uncorrelated to the most probable outcome p_{max}.

This method is tested using the RRAM PUF array based on HRS variation, the generated 1024-bit PUF data are sliced into four 256-bit sequences and are fed through a SHA256 function [77], which is a widely used cryptographic hash algorithm. The subsequent mask function can be defined as Equation 5.7. The resulting 1024-bit data is then stored in the RRAM array as the reconfigured PUF data. In this case, the most probable outcome is $m_{\max} = \text{Mask}(p_{\max})$ instead of the original one.

$$\text{Mask}(PUF_i) = \text{SHA256}(PUF_{i,1:256}), ..., \text{SHA256}(PUF_{i,769:1024}) \tag{5.7}$$

Using the same method for predicting the PUF outcome as described in Equation 5.5, the predicted PUF outcome at the l_{th} cycle with masking is defined in Equation 5.8.

$$\chi_l = (\chi_{l,k})_{k=1}^{N} = (\text{Majority}\left[(\text{Mask}(PUF_i)_k)_{i=1}^{l}\right])_{k=1}^{N} \tag{5.8}$$

The correlation coefficient between the prediction χ_l or x_l and the most probable PUF outcome $m_{\max}$ or $p_{\max}$ can be computed based on Equation 5.9 and Equation 5.10 respectively.

$$\rho_{\text{masked,l}} = \frac{\text{cov}(\chi_l, m_{\max})}{\sigma_{\chi_l}\sigma_{m_{\max}}} \tag{5.9}$$

$$\rho_{\text{unmasked,l}} = \frac{\text{cov}(x_l, p_{\max})}{\sigma_{x_l}\sigma_{p_{\max}}} \tag{5.10}$$

The simulated correlation coefficient with the two cases are plotted in Figure 5.27. It can be clearly seen that the prediction results for the unmasked case are moving towards the most probable outcome, while the masked case shows no correlation as the number of configurations increases. Consequently, this method can hide the information about device dependent bias, providing it the potential of preserving reconfigurability. Although this method dose not require more PUF cells, an additional hardware for the one-way function is needed, and it is still risky once an adversary can directly examine the properties of RRAM cells.

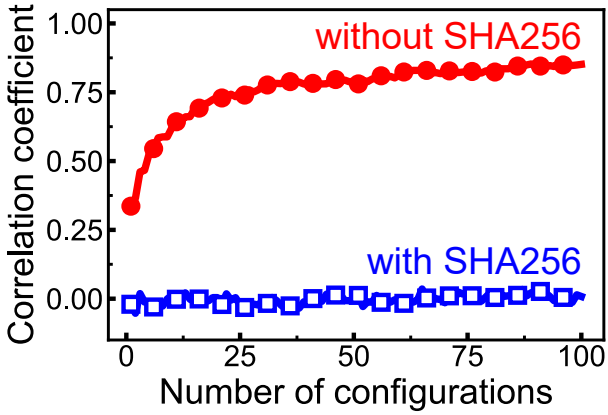


Figure 5.27: The correlation coefficient between the PUF data prediction and the most probable outcome with and without masking.

5.8.3 Use unbiased single cell

The last method is to use only one RRAM cell to generate the bits to be stored in other RRAM cells. With only one cell, the algorithm can be adjusted to generate unbiased random bits, and hence there will be no issue with biases. For a 1024-bit RRAM array, one RRAM cell will be reconfigured 1024 times to generate the PUF data. In order to prevent overuse of a particular cell, a different cell should be chosen to generate output for the next reconfiguration cycle.

This method is more lightweight than the others, since no additional PUF cells or data processing circuits are required. It, however, introduces longer latency while generating the PUF bits. Moreover, as it is actually equivalent to a TRNG embedded in a NVM block, it may lose the advantages against the conventional solutions.

5.9 Conclusion

We have discussed the possible methods to implement reconfigurable PUFs using oxygen vacancy-based RRAM. Even though in theory all the individual RRAM devices can be completely reconfigured using a set/reset cycle, there remains a noticeable configuration-to-configuration similarity when the variation inside a full array of devices is considered. The inherent RRAM-to-RRAM

variation, originating from the structural discrepancy, will create different optimal operating conditions for each RRAM PUF cell. Using a single algorithm to reconfigure all devices will introduce different biases to each device. The impact can be quantified by the inter-configuration hamming distance and min-entropy, and both show that all the discussed algorithms cannot achieve true reconfigurability.

Using a measurement-calibrated physics-based model, these observations can be understood and explained. We have shown that the RRAM-to-RRAM variation not only affects the optimal threshold to determine 0/1 in a single resistance distribution, but also changes the probability of SET failure and the SET transition time. Consequently, PUFs using oxygen vacancy-based RRAM are not fully reconfigurable, regardless of the algorithm used. The results shown in two recent RRAM PUF papers provide experimental support on this conclusion.

Provided these facts, a RRAM PUF can be considered as a viable candidate when looking for a reconfigurable PUF solution, but the non-ideal reconfigurability needs to be carefully handled. Designers should be aware of the degraded uniqueness and entropy loss, and have to choose a proper solution to compensate this non-ideal phenomenon.

Chapter 6

Conclusion and future work

By studying the two practical examples of active PUFs, the advantages of using this concept have been clearly shown. Besides these advantages, we have also addressed some non-ideal phenomena that need to be taken into account for a proper usage of such PUF solutions. The main contributions of this thesis will be listed in this chapter, followed by some suggestions on the future development of active PUFs.

6.1 Summary of contributions

Generally speaking, there are three main contributions of this thesis. First, we have summarized the challenges for eliminating instability and brought up the concept of active PUFs, which distinguish them from the conventional PUFs relying on process variations. Second, we have proposed a novel PUF solution based on the soft oxide breakdown positions in MOSFETs. With experimental validation and detailed analysis, we have proven it to be a PUF solution that is highly reliable and has good security. Last but not least, we have performed detailed studies on the RRAM PUFs, showing that the reconfigurability of these implementations is not ideal.

6.1.1 Introducing active PUFs

Data stability has been considered as a main issue for PUFs for a long time. Various stabilization techniques and PUF implementations have been proposed

as a solution towards the ideal data stability. The incompleteness of these stabilization techniques has been clearly stated, and it shows the necessities of looking for new PUF concepts.

Limitations on data stability

We studied the properties of several popular PUF implementations, showing that instability is inherently present due to the relatively small process-induced variability. We also studied the most commonly used techniques for stability improvement, including temporal majority voting, dark-bit masking and burn-in enhancement. These techniques are shown to be effective but each technique has its own disadvantages, and hence they are not ideal solutions for improving data stability. A new PUF design concept needs to be introduced, in order to overcome these limitations and achieving an ideal data stability.

Active PUFs

As proposed to enlarge the mismatches within PUFs, time-dependent variability is used as an auxiliary or the main entropy source in some PUF work. The operating principle of these PUFs differs from the conventional PUFs based on process variations. Since the PUF behavior of these implementations are actively generated, they are referred as the active PUFs. The active PUFs, such as the anti-fuse base PUFs, can achieve excellent data stability without any post-processing. For some of the active PUFs, such as the RRAM PUFs, they can be potentially reconfigurable.

6.1.2 Soft-BD PUF

We have proposed a new PUF design based on the position of a soft oxide breakdown in MOSFETs. The self-limiting breakdown generating procedure can successfully produce exactly one soft breakdown spot in a PUF cell, which is the key for both good randomness and good data stability. We have implemented two test chips for soft-BD PUFs in 40nm CMOS, for both device level and circuit level characterization.

Device-level characterization

Using the first test chip, we have demonstrated the feasibility of generating soft breakdown spots in the PUF cells efficiently and reliably. The binarized current

characteristic predicts an excellent readout stability once introducing an on-chip readout interface. The analysis on probability of having a 2_{nd} breakdown spot also predicts a good long-term reliability. Apart from the three-transistor (3T) structure with binary breakdown positions, the two-transistor (2T) structure with analog breakdown positions is also tested. As a result, we concluded that the 3T structure is a more promising solution in terms of data stability.

Circuit validation

In the second test chip, an up-sized soft-BD PUF array is implemented with all the essential peripheral circuits. A custom-designed sense-amplifier is utilized as the readout interface, and we have shown that it has a good current resolution of a few nA. The data stability is tested over a wide voltage/temperature space, showing excellent results. We also found that data stability is worse at lower operating voltage, due to the voltage dependence of the breakdown current. At higher temperatures, the data stability is also degraded, and we have found that the main reason is the sense-amplifier offsets.

Security analysis

The statistics of the PUF data have shown good randomness and uniqueness. The statistical indices including the hamming weight, hamming distance and the auto-correlation functions cannot be clearly distinguished from the ideal results. By passing the NIST 800-22 statistical test suite, we have demonstrated that the resulting PUF data has the statistical properties as a random sequence. We have also performed side-channel analysis on the PUF chips, and in this first trial we did not discover any vulnerability against the chosen attack.

Benchmarking

By comparing with prior PUF work, we have demonstrated that the soft-BD PUF has a very good overall performance. Despite it cannot be considered as the most robust implementation, this implementation has a good balance between stability and other aspects, including the visual attack immunity and energy efficiency. In summary, we have proposed and realized a promising PUF solution that can help solving the upcoming challenges of IoT security.

6.1.3 Analysis on Reconfigurable RRAM PUFs

We have made a detailed analysis on the reconfigurability of the RRAM PUFs. The reconfigurable RRAM PUFs are widely discussed but there is usually lack of deeper understanding on the reconfigurability, neither qualitatively nor quantitatively. By surveying recent RRAM PUF work, five possible implementations of reconfigurable RRAM PUFs are listed. The studied RRAM PUFs are based on two main reconfiguration mechanisms: the resistance variations and the stochastic switching events. Our main contribution is analyzing the reconfiguration procedure of these PUF implementations, showing how the actual reconfigurability is apart from the ideal case.

Understanding non-ideal reconfigurability

From the measurement results of RRAMs, we identified that the RRAM-to-RRAM variation can harm the reconfigurability of RRAM PUFs. We studied further on this issue and have identified the main cause of the RRAM-to-RRAM variation is that there are a fixed number of vacancies in each filament constriction. This effect can be simulated using the physics-based hourglass model. According to measurements and simulations, each RRAM has different resistance characteristics and switching behaviors. As a result, the same operating principle may not suit all the RRAMs, and it will cause device-dependent biases that degrades the reconfigurability.

Reconfigurability assessment

Based on a good characterization of RRAMs, we have constructed different RRAM PUFs in the simulation environment. Simulation results clearly show the reconfigurability is not ideal since the inter-configuration hamming distance has a mean value apart from 0.5. We also shown that it is easier to predict the result of a new configuration, if the number of configurations increases. Finally, based on these observations, several possible solutions to use the non-ideal reconfigurability are proposed.

6.2 Future work

In our opinion, PUF is still a relatively new multi-disciplinary research field, which can benefit from the development of semiconductor devices, reliability

physics, ASIC designs, FPGA designs and cryptographic algorithms. As a result, there are many different aspects to work towards better PUF solutions.

Following the same direction on active PUFs, we are listing here the three most interesting research challenges and opportunities in this field. The first one is to develop new active PUFs in novel technologies. The second one is to optimize the existing active PUFs, in particular on improving the immunity against imaging attacks. The third one is the entropy evaluation method for PUFs, which is expected to be more feasible for active PUFs.

6.2.1 PUFs in novel technologies

Following the Moore's law, the semiconductor industry is still pushing the trend of CMOS scaling, bringing up more variability that can benefit PUF designs. In addition, there is another direction towards *Beyond CMOS* solutions which could also keep the increasing trend of the computation power without shrinking the device dimensions. Both trends are providing new opportunities for developing new PUFs, since the newly proposed device structures and materials can bring up new PUF solutions.

For example, in the planar CMOS technologies, we care mainly about the damages on the gate oxides, while in the FinFET technologies, there are also concerns on the spacers. In this case, it would be possible to exploit the variability in the spacers as the PUF entropy, which might give different advantages comparing to the case uses gate oxide BDs. As the technology development continues, there will always be new PUF concepts to be explored, and we can expect even more to happen in newer technologies. Consequently, it would always be a good idea to pay attention to novel technologies, and we might have some ground breaking results.

6.2.2 Optimization of active PUFs

For both the antifuse-based PUFs or the RRAM PUFs, there are doubts on the physical attack immunity. One advantage of using SRAM-like PUFs, is that the PUF data is removed once the chip is powered down. Due to the non-volatile property of the active PUFs, the data remains within the chip even when there is no power supply. The major difference between these two cases is clear once the chip cannot be properly powered, e.g., the chip is physically broken, the data of a SRAM PUF is permanently lost, while there are still chances to retrieve the data stored in antifuses or RRAM cells.

This type of threat cannot be overlooked while designing or using an active PUF, and this is the main reason why we propose to use soft-BD rather than hard-BD. By comparing these two cases, we also found that there is a trade-off between the physical attack immunity and the robustness of the PUF functions (mainly the data stability). Consequently, the main challenge in this direction is how to make proper trade-offs or even find an ideal solution that can optimize the two properties simultaneously.

6.2.3 Novel entropy evaluation methods

As a root of trust for security applications, it is important to have better understandings on the entropy source of the PUFs. Until now, the entropy evaluation methods for PUFs are still relying on pure statistical analysis, which is not sufficient to well qualify the entropy sources. In the research field of TRNGs, the latest qualification standards (recommendations) [48, 83] require long data sequences and stochastic models to have a good evaluation on the entropy sources. To apply a similar concept to PUFs, we need to develop new evaluation techniques and stochastic models for PUFs.

For the conventional PUFs relying on process variations, the entropy is given in the processing steps that the designers have very limited knowledge. Since the foundries usually do not provide details on their processes, it is almost impossible to construct a verifiable stochastic model for the entropy induced by process variations. On the other hand, the active PUFs mainly rely on the entropy generated after fabrication, and these generating processes can be well monitored and analyzed. Consequently, it is feasible to construct stochastic models for active PUFs, and based on these models, we can have deeper understandings on the PUF entropy and can improve the ways of using it.

Bibliography

[1] Akinaga, H., and Shima, H. Resistive random access memory (ReRAM) based on metal oxides. *Proceedings of the IEEE 98*, 12 (2010), 2237–2251.

[2] Alam, M. A., Varghese, D., and Kaczer, B. Theory of breakdown position determination by voltage- and current-ratio methods. *IEEE Transactions on Electron Devices 55*, 11 (Nov 2008), 3150–3158.

[3] Alvarez, A., Zhao, W., and Alioto, M. 15fJ/b static physically unclonable functions for secure chip identification with <2% native bit instability and 140× inter/intra PUF hamming distance separation in 65nm. In *2015 IEEE International Solid-State Circuits Conference - (ISSCC) Digest of Technical Papers* (Feb 2015), pp. 1–3.

[4] Balasch, J., Gierlichs, B., Reparaz, O., and Verbauwhede, I. DPA, bitslicing and masking at 1 GHz. In *International Workshop on Cryptographic Hardware and Embedded Systems* (2015), Springer, pp. 599–619.

[5] Balatti, S., Ambrogio, S., Carboni, R., Milo, V., Wang, Z., Calderoni, A., Ramaswamy, N., and Ielmini, D. Physical unbiased generation of random numbers with coupled resistive switching devices. *IEEE Transactions on Electron Devices 63*, 5 (2016), 2029–2035.

[6] Beckers, A., Gierlichs, B., Balasch, J., and Verbauwhede, I. Comparison of two setups for contactless power measurements for side-channel analysis. In *2018 IEEE International Symposium on Electromagnetic Compatibility and 2018 IEEE Asia-Pacific Symposium on Electromagnetic Compatibility (EMC/APEMC)* (2018), IEEE, pp. 739–744.

[7] Bhargava, M., and Mai, K. A high reliability PUF using hot carrier injection based response reinforcement. In *International Workshop on Cryptographic Hardware and Embedded Systems* (2013), Springer, pp. 90–106.

[8] Blech, I., and Herring, C. Stress generation by electromigration. *Applied Physics Letters 29*, 3 (1976), 131–133.

[9] Brier, E., Clavier, C., and Olivier, F. Correlation power analysis with a leakage model. In *International workshop on cryptographic hardware and embedded systems* (2004), Springer, pp. 16–29.

[10] Che, W., Plusquellic, J., and Bhunia, S. A non-volatile memory based physically unclonable function without helper data. In *Computer-Aided Design (ICCAD), 2014 IEEE/ACM International Conference on* (2014), IEEE, pp. 148–153.

[11] Chen, A. Utilizing the variability of resistive random access memory to implement reconfigurable physical unclonable functions. *IEEE Electron Device Letters 36*, 2 (2015), 138–140.

[12] Chen, N. The benefits of antifuse OTP, 2016. Available at http://semiengineering.com/the-benefits-of-antifuse-otp/.

[13] Chuang, K.-H., Bury, E., Degraeve, R., Kaczer, B., Groeseneken, G., Verbauwhede, I., and Linten, D. Physically unclonable function using CMOS breakdown position. In *2017 IEEE International Reliability Physics Symposium (IRPS)*, pp. 4C–1.1–4C–1.7.

[14] Chuang, K.-H., Bury, E., Degraeve, R., Kaczer, B., Kallstenius, T., Groeseneken, G., Linten, D., and Verbauwhede, I. A multi-bit/cell PUF using analog breakdown positions in CMOS. In *2018 IEEE International Reliability Physics Symposium (IRPS)* (March 2018), pp. P–CR.2–1–P–CR.2–5.

[15] Chuang, K.-H., Bury, E., Degraeve, R., Kaczer, B., Linien, D., and Verbauwhede, I. A physically unclonable function with 0% BER using soft oxide breakdown in 40nm CMOS. In *2018 IEEE Asian Solid-State Circuits Conference (A-SSCC)* (2018), IEEE, pp. 157–160.

[16] Chuang, K.-H., Bury, E., Degraeve, R., Kaczer, B., Linten, D., and Verbauwhede, I. A physically unclonable function using soft oxide breakdown featuring 0% native BER and 51.8fJ/bit in 40nm CMOS. *IEEE Journal of Solid-State Circuits 54*, 10 (Oct 2019).

[17] Chuang, K.-H., Degraeve, R., Fantini, A., Groeseneken, G., Linten, D., and Verbauwhede, I. A cautionary note when looking for a truly reconfigurable resistive RAM PUF. *IACR Transactions on Cryptographic Hardware and Embedded Systems 2018*, 1 (Feb. 2018), 98–117.

[18] Claes, M., van der Leest, V., and Braeken, A. Comparison of SRAM and PUF in 65nm technology. In *Nordic Conference on Secure IT Systems* (2011), Springer, pp. 47–64.

[19] Crupi, F., Kauerauf, T., Degraeve, R., Pantisano, L., and Groeseneken, G. A novel methodology for sensing the breakdown location and its application to the reliability study of ultrathin Hf-silicate gate dielectrics. *IEEE Transactions on Electron Devices 52*, 8 (Aug 2005), 1759–1765.

[20] Das, J., Scott, K., Rajaram, S., Burgett, D., and Bhanja, S. RRAM PUF: A novel geometry based magnetic PUF with integrated CMOS. *IEEE Transactions on Nanotechnology 14*, 3 (May 2015), 436–443.

[21] Degraeve, R., Fantini, A., Clima, S., Govoreanu, B., Goux, L., Chen, Y. Y., Wouters, D., Roussel, P., Kar, G. S., Pourtois, G., et al. Dynamic 'hour glass' model for set and reset in HfO_2 RRAM. In *VLSI Technology (VLSIT), 2012 Symposium on* (2012), IEEE, pp. 75–76.

[22] Degraeve, R., Fantini, A., Gorine, G., Roussel, P., Clima, S., Chen, C. Y., Govoreanu, B., Goux, L., Linten, D., Jurczak, M., and Thean, A. Quantitative model for post-program instabilities in filamentary RRAM. In *2016 IEEE International Reliability Physics Symposium (IRPS)* (April 2016), pp. 6C–1–1–6C–1–7.

[23] Degraeve, R., Fantini, A., Raghavan, N., Chen, Y., Goux, L., Clima, S., Cosemans, S., Govoreanu, B., Wouters, D., Roussel, P., et al. Modeling RRAM set/reset statistics resulting in guidelines for optimized operation. In *VLSI Technology (VLSIT), 2013 Symposium on* (2013), IEEE, pp. T98–T99.

[24] Degraeve, R., Fantini, A., Raghavan, N., Goux, L., Clima, S., Chen, Y.-Y., Belmonte, A., Cosemans, S., Govoreanu, B., Wouters, D., et al. Hourglass concept for RRAM: a dynamic and statistical device model. In *Physical and Failure Analysis of Integrated Circuits (IPFA), 2014 IEEE 21st International Symposium on the* (2014), IEEE, pp. 245–249.

[25] Degraeve, R., Govoreanu, B., Kaczer, B., Van Houdt, J., and Groeseneken, G. Measurement and statistical analysis of single trap current-voltage characteristics in ultrathin SiON. In *Reliability Physics Symposium, 2005. Proceedings. 43rd Annual. 2005 IEEE International* (2005), IEEE, pp. 360–365.

[26] Degraeve, R., Groeseneken, G., Bellens, R., Ogier, J. L., Depas, M., Roussel, P. J., and Maes, H. E. New insights in the relation

between electron trap generation and the statistical properties of oxide breakdown. *IEEE Transactions on Electron Devices 45*, 4 (1998), 904–911.

[27] Delvaux, J. Machine-learning attacks on PolyPUFs, OB-PUFs, RPUFs, LHS-PUFs, and PUF–FSMs. *IEEE Transactions on Information Forensics and Security 14*, 8 (Aug 2019), 2043–2058.

[28] Delvaux, J., Gu, D., Schellekens, D., and Verbauwhede, I. Helper data algorithms for PUF-based key generation: Overview and analysis. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 34*, 6 (2015), 889–902.

[29] Delvaux, J., Peeters, R., Gu, D., and Verbauwhede, I. A survey on lightweight entity authentication with strong PUFs. *ACM Comput. Surv. 48*, 2 (Oct. 2015), 26:1–26:42.

[30] Delvaux, J., and Verbauwhede, I. Fault injection modeling attacks on 65 nm arbiter and RO sum PUFs via environmental changes. *IEEE Transactions on Circuits and Systems I: Regular Papers 61*, 6 (2014), 1701–1713.

[31] E. R. Hsieh, H. W. Wang, C. H. Liu, S. Chung, T. P. Chen, S. A. Huang, T. J.Chen, and O. Cheng. Embedded PUF on 14nm HKMG FinFET platform: A novel 2-bit-per-cell OTP-based memory feasible for IoT secuirty solution in 5G era. In *2019 Symposium on VLSI Circuits. Digest of Technical Papers (IEEE Cat. No.03CH37408)* (June 2019).

[32] Fantini, A., Gorine, G., Degraeve, R., Goux, L., Chen, C.-Y., Redolfi, A., Clima, S., Cabrini, A., Torelli, G., and Jurczak, M. Intrinsic program instability in HfO_2 RRAM and consequences on program algorithms. In *Electron Devices Meeting (IEDM), 2015 IEEE International* (2015), IEEE, pp. 7–5.

[33] Fantini, A., Goux, L., Redolfi, A., Degraeve, R., Kar, G., Chen, Y. Y., and Jurczak, M. Lateral and vertical scaling impact on statistical performances and reliability of 10nm TiN/Hf(Al)O/Hf/TiN RRAM devices. In *VLSI Technology (VLSI-Technology): Digest of Technical Papers, 2014 Symposium on* (2014), IEEE, pp. 1–2.

[34] Franco, J., Kaczer, B., and Groeseneken, G. Poly-si heaters for ultra-fast local temperature control of on-wafer test structures. *Microelectronic Engineering 114* (2014), 47–51.

[35] Gao, Y., Ma, H., Al-Sarawi, S. F., Abbott, D., and Ranasinghe, D. C. PUF-FSM: A controlled strong PUF. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 37*, 5 (May 2018), 1104–1108.

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

[37] Gilbert, E. N. Gray codes and paths on the n-cube. *The bell system technical journal 37*, 3 (1958), 815–826.

[38] Govoreanu, B., Kar, G. S., Chen, Y., Paraschiv, V., Kubicek, S., Fantini, A., Radu, I. P., Goux, L., Clima, S., Degraeve, R., Jossart, N., Richard, O., Vandeweyer, T., Seo, K., Hendrickx, P., Pourtois, G., Bender, H., Altimime, L., Wouters, D. J., Kittl, J. A., and Jurczak, M. 10×10nm2 Hf/HfOx crossbar resistive RAM with excellent performance, reliability and low-energy operation. In *2011 International Electron Devices Meeting* (Dec 2011), pp. 31.6.1–31.6.4.

[39] Grasser, T., Kaczer, B., Goes, W., Reisinger, H., Aichinger, T., Hehenberger, P., Wagner, P.-J., Schanovsky, F., Franco, J., Luque, M. T., et al. The paradigm shift in understanding the bias temperature instability: From reaction–diffusion to switching oxide traps. *IEEE Transactions on Electron Devices 58*, 11 (2011), 3652–3666.

[40] Guajardo, J., Kumar, S. S., Schrijen, G.-J., and Tuyls, P. FPGA intrinsic PUFs and their use forr IP protection. In *International workshop on cryptographic hardware and embedded systems* (2007), Springer, pp. 63–80.

[41] Herrewege, A. V. *Lightweight PUF-based Key and Random Number Generation.* PhD thesis, KU Leuven, 2015. Ingrid Verbauwhede (promotor).

[42] Hospodar, G., Maes, R., and Verbauwhede, I. Machine learning attacks on 65nm arbiter PUFs: Accurate modeling poses strict bounds on usability. In *2012 IEEE International Workshop on Information Forensics and Security (WIFS)* (Dec 2012), pp. 37–42.

[43] Hu, Chenming, and Lu, Qiang. A unified gate oxide reliability model. In *1999 IEEE International Reliability Physics Symposium Proceedings. 37th Annual (Cat. No.99CH36296)* (March 1999), pp. 47–51.

[44] Hutter, M., and Schmidt, J.-M. The temperature side channel and heating fault attacks. In *International Conference on Smart Card Research and Advanced Applications* (2013), Springer, pp. 219–235.

[45] Jinbong Kim, and Kwyro Lee. 3-transistor antifuse OTP ROM array using standard CMOS process. In *2003 Symposium on VLSI Circuits. Digest of Technical Papers (IEEE Cat. No.03CH37408)* (June 2003), pp. 239–242.

[46] Kasper, T., Oswald, D., and Paar, C. EM side-channel attacks on commercial contactless smartcards using low-cost equipment. In *International Workshop on Information Security Applications* (2009), Springer, pp. 79–93.

[47] Katzenbeisser, S., Koçabas, Ü., Van Der Leest, V., Sadeghi, A.-R., Schrijen, G.-J., Schröder, H., and Wachsmann, C. Recyclable PUFs: logically reconfigurable PUFs. *Cryptographic Hardware and Embedded Systems–CHES 2011* (2011), 374–389.

[48] Killmann, W., and Schindler, W. A proposal for: Functionality classes for random number generators, 2011, BSI, Bonn.

[49] Kocher, P., Jaffe, J., and Jun, B. Differential power analysis. In *Annual International Cryptology Conference* (1999), Springer, pp. 388–397.

[50] Kuhn, K. J., Giles, M. D., Becher, D., Kolar, P., Kornfeld, A., Kotlyar, R., Ma, S. T., Maheshwari, A., and Mudanai, S. Process technology variation. *IEEE Transactions on Electron Devices 58*, 8 (2011), 2197–2208.

[51] Kumar, D. S., Beckers, A., Balasch, J., Gierlichs, B., and Verbauwhede, I. An in-depth and black-box characterization of the effects of laser pulses on ATmega328P. In *International Conference on Smart Card Research and Advanced Applications* (2018), Springer, pp. 156–170.

[52] Kursawe, K., Sadeghi, A.-R., Schellekens, D., Skoric, B., and Tuyls, P. Reconfigurable physical unclonable functions-enabling technology for tamper-resistant storage. In *Hardware-Oriented Security and Trust, 2009. HOST'09. IEEE International Workshop on* (2009), IEEE, pp. 22–29.

[53] Lee, G. S., Kim, G., Kwak, K., Jeong, D. S., and Ju, H. Enhanced reconfigurable physical unclonable function based on stochastic nature of multilevel cell RRAM. *IEEE Transactions on Electron Devices 66*, 4 (April 2019), 1717–1721.

[54] Lee, J. W., Daihyun Lim, Gassend, B., Suh, G. E., van Dijk, M., and Devadas, S. 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)* (June 2004), pp. 176–179.

[55] Liu, M., Zhou, C., Tang, Q., Parhi, K. K., and Kim, C. H. A data remanence based approach to generate 100% stable keys from an

SRAM physical unclonable function. In *2017 IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*, pp. 1–6.

[56] Liu, N., Hanson, S., Sylvester, D., and Blaauw, D. Oxid: On-chip one-time random id generation using oxide breakdown. In *VLSI Circuits (VLSIC), 2010 IEEE Symposium on* (2010), IEEE, pp. 231–232.

[57] Liu, R., Wu, H., Pang, Y., Qian, H., and Yu, S. A highly reliable and tamper-resistant RRAM PUF: Design and experimental validation. In *2016 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)* (May 2016), pp. 13–18.

[58] Maes, R. *Physically unclonable functions: Constructions, properties and applications*. PhD thesis, KU Leuven, Aug. 2012.

[59] Maes, R., Rozic, V., Verbauwhede, I., Koeberl, P., van der Sluis, E., and van der Leest, V. Experimental evaluation of physically unclonable functions in 65 nm CMOS. In *2012 Proceedings of the ESSCIRC (ESSCIRC)* (Sep. 2012), pp. 486–489.

[60] Maes, R., and van der Leest, V. Countering the effects of silicon aging on SRAM PUFs. In *Hardware-Oriented Security and Trust (HOST), 2014*, IEEE, pp. 148–153.

[61] Maes, R., Van Herrewege, A., and Verbauwhede, I. PUFKY: A fully functional PUF-based cryptographic key generator. In *International Workshop on Cryptographic Hardware and Embedded Systems* (2012), Springer, pp. 302–319.

[62] Mathew, S. K., Satpathy, S. K., Anders, M. A., Kaul, H., Hsu, S. K., Agarwal, A., Chen, G. K., Parker, R. J., Krishnamurthy, R. K., and De, V. A 0.19pJ/b PVT-variation-tolerant hybrid physically unclonable function circuit for 100In *2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*, pp. 278–279.

[63] Merli, D., Schuster, D., Stumpf, F., and Sigl, G. Side-channel analysis of PUFs and fuzzy extractors. In *International Conference on Trust and Trustworthy Computing* (2011), Springer, pp. 33–47.

[64] Miranda, E., Falbo, P., Nafría, M., and Crupi, F. Electron transport through electrically induced nanoconstrictions in HfSiON gate stacks. *Applied Physics Letters 92*, 25 (2008), 253505.

[65] Miranda, E., Suñé, J., Rodríguez, R., Nafria, M., Aymerich, X., Fonseca, L., and Campabadal, F. Soft breakdown conduction in ultrathin (3-5 nm) gate dielectrics. *IEEE Transactions on Electron Devices 47*, 1 (2000), 82–89.

[66] Na, T., Song, B., Kim, J. P., Kang, S. H., and Jung, S. Offset-canceling current-sampling sense amplifier for resistive nonvolatile memory in 65 nm CMOS. *IEEE Journal of Solid-State Circuits 52*, 2 (Feb 2017), 496–504.

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

[68] Nigam, T., Kerber, A., and Peumans, P. Accurate model for time-dependent dielectric breakdown of high-k metal gate stacks. In *2009 IEEE International Reliability Physics Symposium* (April 2009), pp. 523–530.

[69] Pang, Y., Gao, B., Wu, D., Yi, S., Liu, Q., Chen, W.-H., Chang, T.-W., Lin, W.-E., Sun, X., Yu, S., Qian, H., Chang, M.-F., and Wu, H. A reconfigurable RRAM physically unclonable function utilizing post-process randomness source with <6×10-6 native bit error rate. In *2019 IEEE International Solid-State Circuits Conference (ISSCC)* (2019), IEEE, pp. 402–404.

[70] Pey, K. L., Tung, C. H., Radhakrishnan, M. K., Tang, L. J., and Lin, W. H. Dielectric breakdown induced epitaxy in ultrathin gate oxide - a reliability concern. In *Digest. International Electron Devices Meeting,* (Dec 2002), pp. 163–166.

[71] Rényi, A., et al. On measures of entropy and information. In *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics* (1961), The Regents of the University of California.

[72] Rhee, K., Kwak, J., Kim, S., and Won, D. Challenge-response based RFID authentication protocol for distributed database environment. In *Security in Pervasive Computing* (Berlin, Heidelberg, 2005), D. Hutter and M. Ullmann, Eds., Springer Berlin Heidelberg, pp. 70–84.

[73] Rührmair, U., Jaeger, C., and Algasinger, M. An attack on PUF-based session key exchange and a hardware-based countermeasure: Erasable PUFs. In *International Conference on Financial Cryptography and Data Security* (2011), Springer, pp. 190–204.

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

[75] Rukhin, A., Soto, J., Nechvatal, J., Smid, M., Barker, E., Leigh, S., Levenson, M., Vangel, M., Banks, D., Heckert, A., et al. A statistical test suite for random and pseudorandom number generators for cryptographic applications. *NIST special publication 800*, 22-r1a (2010).

[76] Satpathy, S., Mathew, S. K., Suresh, V., Anders, M. A., Kaul, H., Agarwal, A., Hsu, S. K., Chen, G., Krishnamurthy, R. K., and De, V. K. A 4-fJ/b delay-hardened physically unclonable function circuit with selective bit destabilization in 14-nm trigate CMOS. *IEEE Journal of Solid-State Circuits 52*, 4 (April 2017), 940–949.

[77] Sklavos, N., and Koufopavlou, O. On the hardware implementations of the SHA-2 (256, 384, 512) hash functions. In *Proceedings of the 2003 International Symposium on Circuits and Systems, 2003. ISCAS'03.* (2003), vol. 5, IEEE, pp. V–V.

[78] Skorobogatov, S. Flash memory 'bumping'attacks. In *International Workshop on Cryptographic Hardware and Embedded Systems* (2010), Springer, pp. 158–172.

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

[80] Takeda, E., and Suzuki, N. An empirical model for device degradation due to hot-carrier injection. *IEEE electron device letters 4*, 4 (1983), 111–113.

[81] Taneja, S., Alvarez, A. B., and Alioto, M. Fully synthesizable PUF featuring hysteresis and temperature compensation for 3.2% native BER and 1.02 fJ/b in 40 nm. *IEEE Journal of Solid-State Circuits 53*, 10 (Oct 2018), 2828–2839.

[82] Tiri, K., and Verbauwhede, I. A VLSI design flow for secure side-channel attack resistant ICs. In *Design, Automation and Test in Europe* (2005), IEEE, pp. 58–63.

[83] Turan, M. S., Barker, E., Kelsey, J., McKay, K. A., Baish, M. L., and Boyle, M. Recommendation for the entropy sources used for random bit generation. *NIST Special Publication 800* (2018), 90B.

[84] Ulreich, S., and Zwerger, W. Where is the potential drop in a quantum point contact? *Superlattices and microstructures 23*, 3-4 (1998), 719–730.

[85] Wicht, B., Nirschl, T., and Schmitt-Landsiedel, D. Yield and speed optimization of a latch-type voltage sense amplifier. *IEEE Journal of Solid-State Circuits 39*, 7 (July 2004), 1148–1158.

[86] Wu, E. Y., Vayshenker, A., Nowak, E., Sune, J., Vollertsen, R.-P., Lai, W., and Harmon, D. Experimental evidence of t_{BD} power-law for voltage dependence of oxide breakdown in ultrathin gate oxides. *IEEE Transactions on Electron Devices 49*, 12 (2002), 2244–2253.

[87] Wu, E. Y., and Vollertsen, R. . On the weibull shape factor of intrinsic breakdown of dielectric films and its accurate experimental determination. Part I: theory, methodology, experimental techniques. *IEEE Transactions on Electron Devices 49*, 12 (Dec 2002), 2131–2140.

[88] Wu, M. Y., Yang, T. H., Chen, L. C., Lin, C. C., Hu, H. C., Su, F. Y., Wang, C. M., Huang, J. P. H., Chen, H. M., Lu, C. C. H., Yang, E. C. S., and Shen, R. S. J. A PUF scheme using competing oxide rupture with bit error rate approaching zero. In *2018 IEEE International Solid - State Circuits Conference - (ISSCC)*, pp. 130–132.

[89] Yoshimoto, Y., Katoh, Y., Ogasahara, S., Wei, Z., and Kouno, K. A ReRAM-based physically unclonable function with bit error rate $< 0.5\%$ after 10 years at 125°C for 40nm embedded application. In *2016 IEEE Symposium on VLSI Technology* (June 2016), pp. 1–2.

[90] Yu, H., Leong, P. H. W., and Xu, Q. An FPGA chip identification generator using configurable ring oscillator. In *2010 International Conference on Field-Programmable Technology* (Dec 2010), pp. 312–315.

[91] Zeitouni, S., Oren, Y., Wachsmann, C., Koeberl, P., and Sadeghi, A. Remanence decay side-channel: The PUF case. *IEEE Transactions on Information Forensics and Security 11*, 6 (June 2016), 1106–1116.

[92] Zhang, L., Fong, X., Chang, C., Kong, Z. H., and Roy, K. Highly reliable spin-transfer torque magnetic RAM-based physical unclonable function with multi-response-bits per cell. *IEEE Transactions on Information Forensics and Security 10*, 8 (Aug 2015), 1630–1642.

[93] Zhang, L., Fong, X., Chang, C.-H., Kong, Z. H., and Roy, K. Feasibility study of emerging non-volatile memory based physical unclonable functions. In *Memory Workshop (IMW), 2014 IEEE 6th International* (2014), IEEE, pp. 1–4.

[94] Zhang, L., Kong, Z. H., Chang, C.-H., Cabrini, A., and Torelli, G. Exploiting process variations and programming sensitivity of phase change memory for reconfigurable physical unclonable functions. *IEEE Transactions on Information Forensics and Security 9*, 6 (2014), 921–932.

[95] Zhou, C., Parhi, K. K., and Kim, C. H. Secure and reliable xor arbiter PUF design: An experimental study based on 1 trillion challenge response pair measurements. In *2017 54th ACM/EDAC/IEEE Design Automation Conference (DAC)* (June 2017), pp. 1–6.

Curriculum Vitae

Kai-Hsin Chuang was born on May 21st 1992 in Yuanlin, Taiwan. He received B.S. degree in Electrical Engineering and Computer Science and M.S. degree in Electronics Engineering from the National Tsing Hua University, Hsinchu, Taiwan, in 2014 and 2015, respectively. In 2015, he joined the Computer Security and Industrial Cryptography (COSIC) group at KU Leuven and the Device Reliability and Electrical Characterization (DRE) group at imec, where he starts working towards the PhD degree.

His main research interest is about design and characterization of physically unclonable functions in CMOS and emerging memory devices.

List of publications

International Journals

1. **K.-H. Chuang**, R. Degraeve, A. Fantini, G. Groeseneken, D. Linten, and I. Verbauwhede, "A Cautionary Note When Looking for a Truly Reconfigurable Resistive RAM PUF," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2018, no. 1, pp. 98-117, 2018.

2. **K.-H. Chuang**, E. Bury, R. Degraeve, B. Kaczer, D. Linten, and I. Verbauwhede, "A Physically Unclonable Function Using Soft Oxide Breakdown Featuring 0% Native BER and 51.8fJ/bit in 40nm CMOS," *IEEE Journal of Solid-State Circuits*, vol. 54, no. 10, 2019.

3. P. Wang, E. Zhang, **K.-H. Chuang**, W. Liao, H. Gong, P. Wang, C. N. Arutt, K. Ni, M. W. McCurdy, I. Verbauwhede, E. Bury, D. Linten, D. M. Fleetwood, R. D. Schrimpf, and R. A. Reed, "X-Ray and Proton Radiation Effects on 40-nm CMOS Physically Unclonable Function Devices," *IEEE Transactions on Nuclear Science*, vol. 65, no. 8, pp. 1519-1524, Aug. 2018.

International Conferences

1. **K.-H. Chuang**, E. Bury, R. Degraeve, B. Kaczer, G. Groeseneken, I. Verbauwhede and D. Linten, "Physically Unclonable Function Using CMOS Breakdown Position," in *2017 IEEE International Reliability Physics Symposium (IRPS)*, April, 2017, pp. 4C–1.1–4C–1.7.

2. **K.-H. Chuang**, E. Bury, R. Degraeve, B. Kaczer, T. Kallstenius, G. Groeseneken, I. Verbauwhede and D. Linten, "A Multi-bit/cell PUF Using Analog Breakdown Positions in CMOS," in *2018 IEEE International*

Reliability Physics Symposium (IRPS), March, 2018, pp. P-CR.2-1-P-CR.2-5. (*Best Poster Award*)

3. **K.-H. Chuang**, E. Bury, R. Degraeve, B. Kaczer, D. Linten, and I. Verbauwhede, "A Physically Unclonable Function with 0% BER Using Soft Oxide Breakdown in 40nm CMOS," in *2018 IEEE Asian Solid-State Circuits Conference (A-SSCC)*, Nov, 2018, pp. 157–160.

4. E. Bury, B. Kaczer, **K.-H. Chuang**, J. Franco, P. Weckx, A. Chasin, M. Simicic, D. Linten and G. Groeseneken, "Statistical Assessment of the Full V_G/V_D Degradation Space Using Dedicated Device Arrays," in *2017 IEEE International Reliability Physics Symposium (IRPS)*, April, 2017, pp. 2D-5.1-2D-5.6.

5. E. Bury, A. Chasin, B. Kaczer, **K.-H. Chuang**, J. Franco, M. Simicic, P. Weckx and D. Linten, "Self-heating-aware CMOS Reliability Characterization Using Degradation Maps," in *2018 IEEE International Reliability Physics Symposium (IRPS)*, March, 2018, pp. 2A.3-1-2A.3-6.

6. E. Bury, A. Chasin, B. Kaczer, **K.-H. Chuang**, J. Franco, M. Simicic, P. Weckx and D. Linten, "Recent Insights in CMOS Reliability Characterization by the Use of Degradation Maps (invited)," in *2018 14th IEEE International Conference on Solid-State and Integrated Circuit Technology (ICSICT)*, Qingdao, 2018, pp. 1-4.

7. E. Bury, A. Chasin, **K.-H. Chuang**, M. Vandemaele, S. Van Beek, J. Franco, B. Kaczer and D. Linten "Array-based Statistical Characterization of CMOS Degradation Modes and Modeling of the Time-Dependent Variability Induced by Different Stress Patterns in the $\{V_G,V_D\}$ bias space," in *2019 IEEE International Reliability Physics Symposium (IRPS)*, April, 2019, pp. 1-6.

8. I. Verbauwhede and **K.-H. Chuang**, "Security and Reliability: Friend or Foe (invited)," in *2019 IEEE International Electron Devices Meeting (IEDM)*, December, 2019, pp. 13.4.1-13.4.4.

9. M. Vandemaele, **K.-H. Chuang**, E. Bury, S. Tyaginov, G. Groeseneken and B. Kaczer, "The Infuence of Gate Bias on the Anneal of Hot-Carrier Degradation", in *2020 International Reliability Physics Symposium (IRPS)*, in press.

FACULTY OF ENGINEERING SCIENCE
DEPARTMENT OF ELECTRICAL ENGINEERING
COSIC
Kasteelpark Arenberg 10, box 2452
B-3001 Leuven
kent@esat.kuleuven.be
https://www.esat.kuleuven.be/cosic/