RELIABILITY ASSESSMENT AND TEST METHODS FOR ANTI-COUNTERFEITING TECHNOLOGY

Ana Mafalda Monteiro Oliveira Cortez

RELIABILITY ASSESSMENT AND TEST METHODS FOR ANTI-COUNTERFEITING TECHNOLOGY

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. ir. K. C. A. M. Luyben,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op woensdag 4 november 2015 om 15:00 uur

door

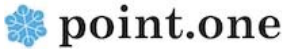
Ana Mafalda MONTEIRO OLIVEIRA CORTEZ

Master of Science in Electrical and Computers Engineering -
Telecommunications, Electronics and Computers
geboren te Porto, Portugal.

This dissertation has been approved by the
promotor: **Prof. dr. K.L.M. Bertels**
copromotor: **Dr. ir. S. Hamdioui**

Composition of the doctoral committee:

Rector Magnificus,	chairman
Prof. dr. K.L.M. Bertels,	Technische Universiteit Delft, promotor
Dr. ir. S. Hamdioui,	Technische Universiteit Delft, copromotor
Independent members:	
Prof. Dr. Ilia Polian,	University of Passau, Germany
Prof. Dr. Henk Corporaal,	Technische Universiteit Eindhoven
Prof. Dr. Eduardo Charbon,	Technische Universiteit Delft
Prof. Dr. ir. Geert Leurs,	Technische Universiteit Delft
Dr. Giorgio Di Natale,	Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier, France
Prof. Dr. ir. Alle-Jan van der Veen	Technische Universiteit Delft, reservelid



The work in this thesis was supported by the Dutch "Point One Program" under the RATE project (PNU09C09) and partially sponsored by COST action TRUDEVICE IC1204.

Keywords:	memory-based PUF systems, noise reduction, secure testing, scan-chain free testing, enhanced scan-chains
Published and distributed by:	Ana Mafalda Monteiro Oliveira Cortez (e-mail: mafalda.m.cortez@gmail.com)

ISBN 978-94-6186-529-8

An electronic version of this dissertation is available at
http://repository.tudelft.nl/.

Para as minhas duas Marias...

SUMMARY

In the Information Technology (IT) era, where valuable information is constantly stored, transferred and processed, there is a high incentive to hack IT systems. To prevent the success of such attacks, IT systems integrate a number of security schemes at the software level, from cryptographic algorithms to authentication protocols. However, as these security schemes become more and more sophisticated and harder to circumvent, attacks to the hardware components of IT systems, also denominated physical attacks, become more and more attractive. This poses major challenges in design, manufacturing, test and reliability of secure ICs. In this thesis two challenges are addressed: one related to reliability and one to test.

Reliability Characterization and Improvement for Secure ICs - Secure cryptographic key storage is a crucial design and implementation aspect to prevent the success of physical attacks. Traditional cryptographic methods use Non-Volatile Memories (NVMs) to permanently store the cryptographic key. However, NVMs are vulnerable to physical attacks as the stored information is permanently available, even when the power source is removed. Combined with proper post-processing, a Physical Unclonable Function (PUF) is able to generate secret keys of cryptographic strength, and reliably store them in a highly secure manner without the need for NVM. The key is derived from the device-intrinsic randomness which is evaluated by the silicon PUF. Post-processing is required to guarantee the reliability of PUF responses.

In this thesis, reliability (reproducibility and uniqueness) analysis was performed for memory PUFs by identifying the impact of different technological and non-technological parameters in the reproducibility and uniqueness of the PUF signatures. The results showed that the threshold voltage (V_{th}) is the technological parameter with the most significant impact, while the environment temperature is the non-technological parameter counterpart. Moreover, a new scheme for memory PUF reliability improvement is proposed; it is based on intelligently adapting the voltage ramp-up time to the environment temperature. The scheme is designed and analyzed in detail in order to evaluate its industrial attractiveness. The results show that the new system costs up to 82.1% less area while it delivers up to 3X higher reproducibility. Exploiting different memory designs (GP and LP) for PUF implementation is also studied in detail, both using circuit simulation and characterizing test chips. The results reveal that general purpose PUFs are up to 2X more reliable than low-power PUFs.

Testing Secure Devices - Testing digital ICs is an unavoidable endeavor to deliver high product quality. To enhance testability in digital ICs, Design-for-Testability (DFT) infrastructures are added. Scan-chains are the most commonly used DFT due to their high fault coverage and integration simplicity. However, they introduce security vulnerabilities, offering a back door for scan-based attacks. Conversely, Built-In Self-Test

(BIST) are less vulnerable to the most common physical attacks; however, realizing a high fault coverage is a major challenge.

In this thesis we propose three secure test solutions against scan-based attacks for digital ICs: one DFT based on Multi-Segment Secure Scan (MSSS) and two secure BIST for PUF based ICs. MSSS is a secure test scheme, it is generic (i.e., it can be integrated in any circuit), it leaks no information on attack progress, it has tunable (flexible) security segments, allowing secure DFT solution optimization depending on the targeted application, and it has no performance penalty in functional mode.

On the other hand, the two secure BIST schemes target PUF based circuits (in particular Fuzzy Extractor (FE) - circuit responsible for post-processing PUF responses and main component of PUF-based systems). The two secure BIST for FE target high stuck-at-fault (SAF) coverage by performing scan-chain free functional testing, to prevent scan-chain abuse for attacks. The first scheme reuses existing FE blocks (for pattern generation and compression) to minimize the area overhead, while the second scheme tests all the FE blocks simultaneously to minimize the test time. The results show that for the first test scheme, a SAF fault coverage of 95% can be realized with no more than 47.1k clock cycles at the cost of a negligible area overhead of only 2.2%; while for the second test scheme a SAF fault coverage of 95% can be realized with 3.5k clock cycles at the cost of 18.6% area overhead.

SAMENVATTING

In de tijdperk van Informatietechnologie (IT), waar waardevolle informatie voortdurend wordt opgeslagen, overgedragen en verwerkt, is er een grote prikkel om IT-systemen te hacken. Om het succes van dergelijke aanvallen te voorkomen, integreren IT-systemen een aantal beveiligingsschema's op software-niveau, van cryptografische algoritmes tot aan verificatie protocollen. Aangezien deze beveiligingsschema's steeds geraffineerder worden en moeilijker zijn te omzeilen, worden hardware-aanvallen die deze schema's implementeren, ook wel bekend als fysieke aanvallen, steeds aantrekkelijker. Dit levert grote uitdagingen op in het ontwerp, de fabricage, het testen en de betrouwbaarheid van beveiligde ICs. In dit proefschrift worden twee problemen geadresseerd: één verwant aan betrouwbaarheid en één aan test.

Betrouwbaarheidskarakterisatie en Verbetering voor Beveiligde ICs - Veilige opslag van cryptografische sleutels is een cruciale ontwerp- en implementatie-aspect om het succes van fysieke aanvallen te voorkomen. Traditionele cryptografische methoden gebruiken Non-Volatile Memories (NVMs) om de cryptografische sleutel permanent op te slaan. Echter zijn NVMs kwetsbaar voor fysieke aanvallen aangezien de opgeslagen gegevens permanent beschikbaar zijn, zelfs wanneer de voeding wordt verwijderd. In combinatie met geschikte nabewerking is een PUF in staat om geheime sleutels van cryptografische sterkte te genereren en op een uiterst veilige manier betrouwbaar op te slaan zonder NVM nodig te hebben. De sleutel wordt afgeleid van chip-intrinsieke willekeurigheid die geëvalueerd word door de silicium PUF. Nabewerking is noodzakelijk om de betrouwbaarheid van PUF reacties te garanderen.

In dit proefschrift werd een betrouwbaarheidsanalyse (reproduceerbaarheid en uniciteit) uitgevoerd voor PUF geheugens door het identificeren van de impact van verschillende technische en niet-technische parameters op de reproduceerbaarheid en de uniciteit van PUF signaturen. De resultaten toonden aan dat de threshold voltage (V_{th}) de technologische parameter is met het grootste effect, terwijl de parameter omgevingstemperatuur de niet-technische tegenhanger is. Bovendien is er een nieuwe schema voor geheugen PUF betrouwbaarheidsverbetering voorgesteld; het is gebaseerd op intelligente aanpassing van de spanningsaanlooptijd aan de omgevingstemperatuur. De schema is ontworpen en in detail geanalyseerd om zijn industriële aantrekkelijkheid te evalueren. De resultaten tonen aan dat het nieuwe systeem tot 82.1% minder aan ruimte kost terwijl het 3X een hogere reproduceerbaarheid levert. Het benutten van verschillende geheugen ontwerpen (GP en LP) voor PUF implementatie is ook in detail bestudeerd, zowel met behulp van circuit simulatie en het karakteriseren van test chips. Uit de resultaten blijkt dat de general purpose PUF's tot 2X betrouwbaarder zijn dan low-power PUF's.

Het Testen van Beveiligde Chips - Het testen van digitale IC's is een onvermijdelijk

inspanning om kwalitatief hoogwaardige producten te leveren. Om de testbaarheid van digitale IC's te verbeteren worden er Design for Testability (DFT) infrastructuren toegevoegd. Scan-chains zijn de meest gebruikte DFT vanwege hun hoge fault coverage en integratie eenvoud; echter introduceren ze beveiligingsproblemen door het aanbieden van een achterdeur voor scan-gebaseerde aanvallen. Daarentegen zijn Built-In Self-Test (BIST) minder kwetsbaar voor de meest voorkomende aanvallen; echter is het realiseren van een hoge fault coverage een grote uitdaging.

In dit proefschrift stellen we drie beveiligde testoplossingen voor tegen scan-gebaseerde aanvallen voor digitale IC's: een DFT op basis van Multi-Segment Secure Scan (MSSS) en twee beveiligde BIST voor PUF-gebaseerde IC's. MSSS is een beveiligde test schema, het is generiek (dat wil zeggen, het kan worden geïntegreerd worden in elke circuit), het lekt geen informaties betreffende de voortgang van de aanval, het heeft afstembare (flexibele) beveiligingssegmenten, daarmee staat het beveiligde DFT oplossing optimalisaties toe afhankelijk van de beoogde toepassing, en het heeft geen prestatie penalty in functionele modus.

Daarentegen beogen de twee veilige BIST schema's PUF gebaseerde circuits (met name Fuzzy Extractor (FE)- circuit verantwoorderlijk voor het nabewerken van PUF reacties en het hoofdonderdeel van PUF-gebaseerde systemen). De twee beveiligde BIST voor FE beogen hoge stuck-at-fault (SAF) dekking door het uitvoeren van functionele tests vrij van scan-chains om hun misbruik in aanvallen te voorkomen. De eerste schema hergebruikt bestaande FE blokken (voor pattern generation en compressie) om de oppervlakte overheid te minimaliseren, terwijl de tweede schema alle FE blokken gelijktijdig test om de test tijd te minimaliseren. De resultaten tonen aan dat voor de eerste testschema, een SAF fault coverage van 95% kan worden gerealiseerd met niet meer dan 47.1k klokcycli ten koste van een verwaarloosbare oppervlakte overhead van slechts 2.2%; terwijl voor het tweede testschema een SAF fault coverage van 95% kan worden gerealiseerd met 3.5k klokcycli ten koste van 18.6% oppervlakte overhead.

ACKNOWLEDGMENTS

Conjointly with chaos, life's greatest tragedy and beauty is our (current) incapability of experimenting with our decisions, to go back in time, and live the consequences of another decision. There are big decisions and small ones. In the perspective of my life, the decision I took about five years ago to commit myself to this project was, unquestionably, a big one! And, it impacted many lives... Isaac Newton once wrote: "If I have seen further it is by standing on the shoulders of giants". Here, I would like to thank my giants (both big and small)!

I would like to start by acknowledging the three persons who together gave me the opportunity to embrace this challenge: dr. Daniel Schobben, dr. Pim Tuyls and assoc. Prof. dr. ir. Said Hamdioui. Daniel, one of the founders of Intrinsic-ID moved on to other projects, shortly after I started. Though we did not spend a lot of time together, his entrepreneur vision and attitude were evident and admirable. Pim, thank you for providing a professional and cheerful working place. I wish you and Intrinsic-ID a very bright future. I am much obliged to Said, my co-promoter and daily supervisor. Said, thank you for the time invested in me, both in research and scientific writing, and for the enriching experiences you enabled such as, research collaborations and students supervision. Moreover, I want to extend my acknowledgments to my promoter and head of the Computer Engineering Lab, Prof. dr. ir. Koen Bertels. Koen, thank you for creating a healthy working environment and to always welcome new suggestions. You literally contributed to make me a more flexible person (yoga reference (:).

I would like to thank my co-authors Apurva Dargar, Gijs Roelofs, Said Hamdioui, Vincent van der Leest, Geert-Jan Schrijen, Roel Maes, Ryoichi Ishihara, Giorgio Di Natale, Marie-Lise Flottes, Bruno Rouzeyre, Ilia Polian and Ali Kaichouhi for all the effort invested in our research. A special thank you to Vincent van der Leest, my colleague at Intrinsic-ID and to Giorgio Di Natale, my host at LIRMM. Vincent, thank you for all the time you dedicated to our work and for the many productive discussions. Giorgio, thank you for your professionalism, friendship, hospitality and positive attitude!

Thank you to my two former master students Apurva Dargar and Gijs Roelofs. Working with you was a very enriching experience! I wish you very successful lives, both personal and professionally.

I would like to thank the committee members for accepting their role, reading this dissertation and providing feedback. Thank you for your effort!

To all the staff that makes our lives easier on a daily basis, I also like to say thank you! From Intrinsic-ID, I want to acknowledge Justine Kontou, Bernadette van Di-

jkhuizen and Femke van Nunen. From CE Lab at TU Delft, Lidwina Tromp, Erik de Vries and Eef Hartman. And finally, from LIRMM Ana Tacuri.

To my office mates, I would like to thank you for the lively and interesting discussions! From CE Lab at TU Delft, I would like to thank Nor Zaidi, Seyab, Motta, Cristi, Innocent and Mahroo. I would like to acknowledge Motta for translating both the propositions and summary in this thesis to Dutch. From Intrinsic-ID, I want to thank Vincent, Peter, Erik, Dipti, Geert-Jan, Olaf and Ilze. And finally, from LIRMM I would like to thank Mario Barbareschi (though not officially office mate), Khalid Latif, Charles Effiong and Stephan de Castro. In addition, I would like to thank prof. Sorin Cotofana, Razvan and Tina, Imran and Carmina for their friendship. To all the other colleagues, you are too many to be mentioned individually! Nonetheless, I would like to thank you all for the good working environment and the good company during our many social events.

During my time in Delft, I have been an enthusiastic student of some of the many cultural and sportive courses that TU Delft offers. I would like to thank both the instructors and the colleagues from fencing, capoeira, yoga, climbing, pole fitness, singing, theater, ballet and modern dance! With you, I got bruised, switched axis, challenged my mind and body in new ways and above all, I had a great time!

Giacomo, Haoxuan, Maja, Dejan, Mihai, Giovanni, Apostolos, Kostas, Katja, Vivi, Nick, Pepinho, Paolo, Mimi, Xuxu, Cati, Dori and Zoey, my friends of two and four legs... Thanks for making me laugh, for sharing food and life with me! Thank you for being there... :) An extra special thank you goes to Mihai, Maja and Haoxuan! Thank you for coming all the way from Dublin, Dresden, and Antwerp to share this day with me! :D mha Negra, you are my (soul) sister, my best friend since I can remember I exist! :) Thank you for all the good memories and for Rafinha (my tiny love)!

Finally, I would like to thank my dearest ones, my family! Thank you for your patience and understanding during my absence! To my unties, uncles and cousins, thank you for all the valuable life experiences and support. Thank you to my father and my (no longer so) little sister, for their encouragement and tenderness. A very special thank you goes to my two Marias, my mom and grandma, for their unconditional love! Last, I want to say thank you to Vlad for being an amazing partner! Thank you for being my most supportive friend and confidant! :)

Mafalda Cortez

CONTENTS

1

INTRODUCTION

Nowadays, human kind relies on Information Technology (IT) systems to store, process and transfer very sensitive and valuable information. The security of this information is enabled by cryptographic algorithms, which evolved in the last decades such that brute force attacks have become very hard. Due to this, attackers have become more creative and have started exploring methods to retrieve the wanted information from the hardware, therefore, circumventing the protection layers in the software. In this chapter, we first introduce the field of Hardware Security. Second, we present its challenges. Third, we describe the research directions of this dissertation. Fourth, we list the main contributions of this thesis. Finally, we provide the outline of the remainder of this dissertation.

1

1.1. INTRODUCTION TO HARDWARE SECURITY

THE aim of this section is to get the reader acquainted with Hardware Security (HS). Section 1.1.1 describes the motivation behind attacks. Section 1.1.2 classifies the attacks. Section 1.1.3 discusses physical attacks.

1.1.1. ATTACK MOTIVATION

In the context of IT systems, an attack is any attempt to destroy, expose, alter, disable, steal or gain unauthorized access to or make unauthorized use of an asset [18]. Attacks are illegal, cost money, time and expertise [2, 16]. Therefore, it is important to understand the different motivations that may lead attackers to hack a system. This information provides important indications about the security level a systems requires; e.g., a vending machine smartcard requires a lower level of security than a system controlling nuclear weapons. Moreover, it also provides information about the profile of the attacker (such as the attacker's expertise or access to high-tech equipment). Because high levels of security are expensive, it is critical that, before focusing on the design of a secure device, design engineers identify the possible attack motivations. The four main attack motivations are described next [3, 6, 24, 38, 44, 45, 50].

Theft of service aims at breaking into electronic devices that provide access to services or information. Successful attacks with this motivation can result in huge losses for the service provider, such as attacks carried out on smartcards used to charge for services (e.g., transportation or communication services). The losses can increase exponentially when the success is distributed over the attacker community or made available online.

Denial of service aims at damaging a product by, for example, launching a malicious update of a device firmware to either switching off the device or permanently damaging it. Competitors are likely attackers, as they are the ones to profit the most from such attack.

Cloning aims at obtaining a product at a negligible price. A wide range of attackers have interest in cloning; e.g., individuals who copy music and movies or companies who clone competitors products to reduce the development cost and therefore increase their profit.

Overbuilding aims at Intellectual Property (IP) theft. Potential overbuilding victims are fabless companies (large majority nowadays). The contract manufacturer can easily over produce the requested quantity of devices and sell them either on the back market or to a competitor.

Security investment is always a trade-off between losses of suffering an attack and the cost of preventing such an attack. This trade-off is very different from product to product. So are the attack types. These are discussed next.

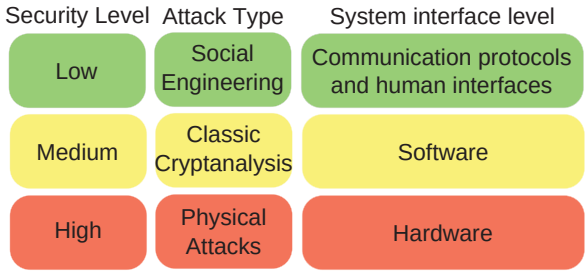


Figure 1.1: Relationship overview of security, attacks and system interface level

1.1.2. ATTACK CLASSIFICATION

Security schemes can be broken by a determined hacker giving it enough time and resources [45]. The goal is to design a system such that it is not attractive to attack; e.g, a system that requires an attack investment 10X when compared to the gain of the attack, can be considered secure. As security schemes and attacks methods are opposite sides of the same coin, they are discussed simultaneously.

IT systems integrate one or more security schemes. The systems' security is as strong as its weakest scheme. Security schemes, such as cryptographic algorithms, are public knowledge; the only secret is the key. Therefore, the goal of an attacker is to retrieve the secret key of the IT system. Figure 1.1 shows the way an attacker perceives how challenging it is to attack an IT system (i.e., security level) at each different system interface level and the associated attack types. IT system security levels are classified as low, medium and high, according to effort required to break them. To each security level there is an associated attack type, i.e., social engineering, classic cryptanalysis and physical attacks, respectively, according to the system interface level that the attack targets, i.e., communication protocols and human interfaces, software and hardware, respectively. Next, we briefly describe the three different attack types.

Social Engineering attacks are those that retrieve keys by involving humans [23, 27, 32, 45]. These attacks are performed by, e.g., directly threatening a human life or by tricking someone into sharing their personal key. This type of attack is associated with the system interface level of communication protocols and human interfaces. Security measures comprise all sorts of restrictions and access control to the building or room where the equipment is running [45].

Classic cryptanalysis tries to find a weakness in the cryptographic algorithm to extract the key [22]. This type of attack targets the software. The software level (operating system and application) supports all the external interfaces, communication protocols, as well as encryption and authentication.

Physical attacks are those that require direct contact with hardware module (e.g., memory) and either perform physical measurements or destroy parts of the hardware in order to succeed [2, 45]. Highly secure systems, such as those used in bank applications,

have hardware tamper resistant modules that keep the secret key inside and perform all critical cryptographic operations [1].

Given the fact that cryptographic algorithms are typically secure enough, it is hard to attackers to break them. Therefore, they are searching for good alternatives, such as physical attacks, to retrieve the key. Although physical attacks require attackers to possess the devices to be attacked, nowadays secret keys are stored in everyday objects such as smartcards. This way, physical attacks represent a real threat not just to elite applications, such as military or banks, but also to the everyday companies that either produce or use smartcard like technology. In this thesis, we are particularly interested in preventing the success of such attacks.

1.1.3. PHYSICAL ATTACKS

Physical attacks are those that require access to the hardware module being attacked. There are three main types of physical attacks, according to level of damage they inflict on the device under attack: non-invasive attacks, invasive attacks and semi-invasive attacks; they are discussed next.

Non-invasive attacks are those that do not damage the device under attack (DUA) [45, 47]. Typical non-invasive physical attacks include manipulation of the voltage supply and clock signal, probing the bus channel, performing power analysis and stimulation of the interface signals in order to break the security protocols and data remanence analysis [45, 47]. Figure 1.2 illustrates a set-up to perform a type of non-invasive attack, the power analysis attack. The set-up comprises a computer, a cryptographic device (i.e., DUA) and an oscilloscope. The computer is used to send cyphertexts to the cryptographic device, which leaks power traces during the cryptographic operations. The oscilloscope is used to read these power traces. Being the algorithm public knowledge, which implements simple logic functions (e.g., XOR) between the cryptographic key and the cyphertexts and given that the attacker has control over the cyphertexts, it is feasible to detect the key bits that are logic 0's and 1's. In short, an attacker can retrieve the secret key by analyzing the correlation between the power traces and the sent cyphertexts.
Non-invasive attacks are considered to be the most serious threat to hardware security of any device [45]. This is due to the lack of tamper evidence on an attacked device together with the easily scalability and low cost of these attacks. However, it is typically very time and effort consuming to succeed in an attack on any particular device.

Invasive attacks are the ones that do damage permanently the device under attack [45, 47]. These type of attacks have a much greater complexity than the ones of non-invasive attacks, as they require direct access to the internal components of the device. Such attacks normally require a well equipped and knowledgeable attacker to succeed. Nevertheless, in second-hand market of semiconductor equipment, one is able to find cheap solutions [45]. The first step of any invasive attack is to remove the chip package and to remove (e.g., by etching) the passivation layer (oxide layer) [45]. Once the chip is opened it is possible to perform probing, as shown in Figure 1.3 or to

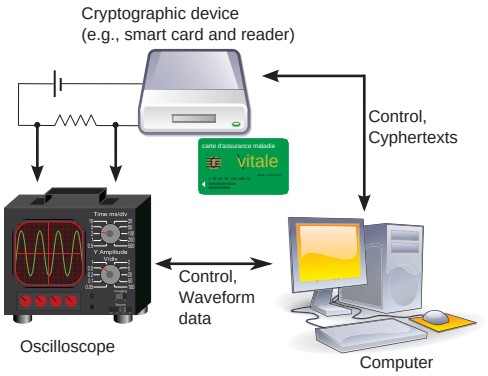


Figure 1.2: Schematic of a Power Analysis Set-up [36]

permanently modify the circuitry, and therefore its functionality as in [19]. The time, the cost and the required knowledge to perform invasive attacks are increasing with the decrease in technology node, resulting in increased attack complexity [45]. A very important fact to conclude is that normally invasive attacks are used as an initial step to understand the chip functionality and then develop cheaper and faster non-invasive attacks [45].

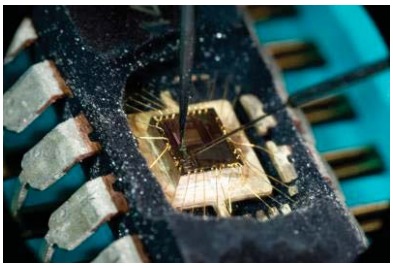


Figure 1.3: Microprobing a decapsulated IC [33]

Semi-invasive attacks [46] have the characteristics of both previous attack types. As invasive attacks, semi-invasive attacks require the removal of the chip package. Nevertheless, as this type of attack does not probe, there is no need of physical damage to the silicon. Similarly, from the non-invasive attacks, semi-invasive attack inherit their low cost and easiness to be reproduced. Examples of semi-invasive attacks include ultra violet (UV) attacks [45] and fault injection attacks [45, 46]. UV attacks target the reset fuses protection, setting the circuit into an unprotected state [45]. Fault-injection attacks aim at defining the state of any transistor in a circuit (by using for example a laser as in the set-up depicted in Figure 1.4) and to propagate its result to the output [45, 46].

1

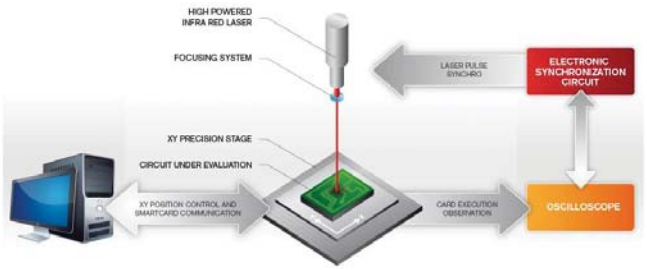


Figure 1.4: Schematic of a fault injection attack set-up [42]

1.2. CHALLENGES

HARDWARE security can be divided into three main areas as shown in Figure 1.5: (i) design and implementation, (ii) test and (iii) attacks and countermeasures. Each of these areas has its own challenges; they are presented next.

- **Design and implementation:** Secure IC design can be divided into three main areas: hardware security primitives, cryptographic algorithm implementation and integration. Hardware security primitives, such as true random number generators, are the foundations of cryptography. If these primitives fail, the entire system is compromised. Cryptographic algorithm implementation has additional requirements to prevent leaking information about the secrets being processed. Finally, securely integrating several cores into an IC is a challenge both from an architectural and placement point of view. Each area is equally important as an attacker needs only to find the weakest point of the system to success in the attack. Next, we briefly discuss the design and implementation challenges for each of the three main areas.

  – **Hardware security primitives** - are the essential hardware building blocks of a secure system. Examples of hardware security primitives include Trusted Monotonic Counters (TMCs) [41, 49], True Random Number Generators (TRNGs) [25, 26, 29, 48] and Physical Unclonable Functions (PUFs) [5, 7–10, 21, 28, 37, 40, 51]. TMCs are embedded counters with two main characteris-

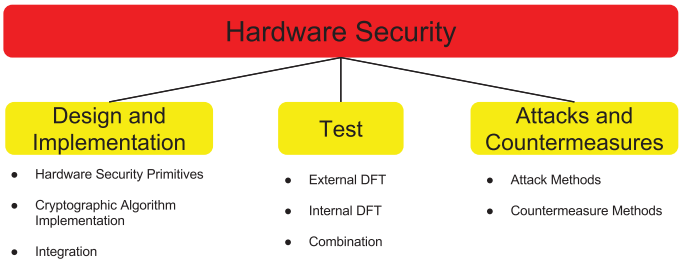


Figure 1.5: Hardware security main areas

tics; they are tamper-resistant (at least tamper-evident) and their value cannot be reverted, once incremented. TMCs main challenges are related to secure design. TRNGs are important security primitives used in cryptography. They use noise of statistically random noisy signals from physical sources. The main challenges are related to the design of TRNGs with required quality in terms of randomness (entropy). A PUF is a physical structure of a device that is hard to clone due to its inherent, device unique and deep-submicron process variations. Due to these characteristics, PUFs can be used to deploy a device unique key (when combined with appropriate post-processing), that is dependent on the physical characteristics of the device itself. PUF challenges include new PUF design, robustness improvement and development of post-processing algorithms (error correction and privacy amplification).

  - **Cryptographic algorithm implementation** - Security systems use cryptographic algorithms that are public knowledge. When implemented in hardware these algorithms comprise additions, multiplications, and other standard logic operations [30, 31, 34, 35]. For this reason, it is feasible to know which values are being processed by analyzing, e.g., the power consumption of the cryptographic core. Therefore, preventing information leakage is a major concern.
  - **Integration** - While integrating different cores in an IC, it is critical to architect the design in such a way that the communication between the different cores does not create attack opportunities [20]. In addition, the placement and routing of critical signals, from a security standpoint, should avoid easy access areas, such as IC pins and edges, to increase the challenge of manipulating such signals.

- **Test:** is an unavoidable task to deliver high quality circuits. Design For Testability (DFT) are all infrastructures added to the original circuit with the purpose to facilitate the test task as well as to make it more efficient. However, these infrastructures introduce a backdoor which malicious users can abuse to gain access to internal states of the circuit [4, 15, 17, 20, 43]. We can identify three main classes of DFT: external (i.e., those where an external source sends test patterns and receives the test results), internal (i.e., when the test patterns are stored or generated in the circuit, as well as the test results) and combined (i.e., when the test patterns and test results are of different types).

  - **External** - To rely on external sources, e.g., Automatic Test Pattern Generation (ATPG) machine, to deal with the test process has a number of advantages including a high fault coverage, diagnosability and low area overhead. However, as the circuit needs to communicate with the external source, extra pins are required. A malicious user can use these pins to gain access to the internal states of the circuit. Challenges include the development of secure test schemes preventing both; (a) the access to the DFT structure by unwanted users, and (b) easy interpretation of the test output results.
  - **Internal** - Conversely to external DFT, internal DFT, also known as Built-In-Self-Test (BIST), has an inherent higher security level. However, BIST can

significantly increase the overall area overhead of a design and it can be challenging to realize a high fault coverage. Methods to realize a high fault coverage and low area overhead are a major challenge.

  - **Combined** - Combined DFT aims at making use of the benefits of each type while realizing the required product quality. Obviously, some of external and internal DFT challenges apply here as well.

- **Attacks and countermeasures** - It is important that secure circuits are challenged with new (physical) attack methods, such as those presented in [14, 39]. Only by putting the security of the circuits to the test, its efficiency can be assured. In addition, typically each new proposed attack also provides countermeasure tips, contributing to the advancement of the state-of-the-art. As introduced previously, there are three main types of attacks: non-invasive attacks, invasive attacks and semi-invasive attacks. From an attacker standpoint, major challenges include the development of attack methods that require non-expensive equipment, that are fast, generic (not bound to a specific circuit design) and scalable. Here, scalability refers not only to the applicability of the method over a wide range of technology nodes, but as importantly, to the reusability of the attack over a family of devices; e.g., consider a production of smartcards all protected by the *same* secret key. Once the secret key of *one* device is discovered, the security of *all* devices is compromised. Contrastingly, countermeasures development target to invalidate a large number of attacks, to have a short development time, low area overhead and to be generic.

1.3. RESEARCH TOPICS

THE research carried out in this thesis addresses a number of challenges introduced in the previous section. The research is divided into two parts.

1. Reliability Characterization and Improvement for Secure ICs
2. Testing Secure Devices

RELIABILITY CHARACTERIZATION AND IMPROVEMENT FOR SECURE ICS

Physical Unclonable Functions (PUFs) are the embodiment of random and unique, but repeatable, mapping of challenges to responses in physical structures such as integrated circuits (ICs) [9, 47]. The uniqueness and repeatability of this mapping, known as fingerprint, enables unambiguous identification of ICs making PUFs efficient hardware security primitives. Moreover, PUFs are hard to clone due to their random, uncontrollable, inherent, device-unique and deep-submicron process variations. Combined with proper post-processing, a PUF is able to generate secret keys of cryptographic strength, and reliably store them in a highly secure manner without the need for conventional on-chip non-volatile memory [47]. However, PUF fingerprints have two main drawbacks. First, they are noisy; when the same challenge is consecutively applied to the same device, the mapped raw responses (i.e., PUF fingerprints) are slightly different even under the same operating conditions, resulting in reduced repeatability. Second, the fingerprints of any two random devices might be slightly correlated, resulting in

reduced uniqueness.

PUF robustness, i.e., repeatability and uniqueness, is a major concern for PUF-based systems. In this dissertation we investigate the impact that internal and external factors have on the robustness of memory-based PUFs (i.e., PUFs that have a memory cell at their core). Moreover, we investigate methods to increase robustness manipulating external factors, such as temperature.

TESTING SECURE DEVICES

As discussed in the previous section, a main concern of secure systems is the delivery of high quality products, guaranteed by the development and implementation of efficient DFT schemes, without jeopardizing the systems' security. In this thesis, we address this challenge proposing three different secure test schemes. First, we propose a generic enhanced DFT for secure ICs and thereafter, two secure BIST schemes for PUF-based ICs.

1.4. CONTRIBUTIONS

THIS thesis has the following contributions.

1.4.1. RELIABILITY CHARACTERIZATION AND IMPROVEMENT FOR SECURE ICS

We investigated the robustness of memory-PUF fingerprints and proposed a technique to enhance it. With respect to this work, the following contributions apply.

- **Robustness analysis of PUF-based secure ICs:** the contributions related to this topic are taken for the work publish in [7, 8].
    1. Analytical model of the start-up behavior of SRAM PUF and its validation of the model using silicon experiments.
    2. SRAM PUF sensitivity analysis, identifying the impact of different technology and non-technology parameters; examples are threshold voltage and temperature, respectively.
    3. Investigation of SRAM PUF robustness for two different designs (low-power and general purpose).
    4. Discussion of the pros and cons of each investigated design in terms of security, power consumption and area overhead.
- **Design for robustness for secure ICs:** these contributions are based on the work publish in [9, 10].
    1. A low-cost scheme to significantly improve the robustness of memory-based fingerprints, based on adapting the voltage-ramp up to the environment temperature.
    2. Validation of the scheme with both simulation and silicon experiments.
    3. Design and implementation of an adapter circuit to tune the voltage ramp-up to the environment temperature.

1

1.4.2. TESTING SECURE DEVICES

We present three solutions for testing secure devices; one DFT for secure ICs and two BIST solution for PUF-based ICs. Within the context of this work, the following contributions apply.

- **DFT:** the contributions below are based on [12].
  1. Novel Multi-Segment Secure Scan (MSSS) test scheme; the scheme is secure against brute force attacks, generic (can be integrated in any circuit) and tunable (flexible) security segments, allowing secure DFT solution optimization depending on the targeted application,
  2. Added countermeasure that leakage no information on attack progress.
  3. No performance penalty in functional mode and inherent low area overhead.
- **BIST:** the contributions below are based on [11, 13].
  1. Two efficient scan-chains free secure test schemes that realize a high test quality based on pattern generation for stuck-at-faults by performing functional testing. The first scheme reuses existing FE blocks (for pattern generation and compression) to minimize the area overhead, while the second scheme tests all the FE blocks simultaneously to minimize the test time.
  2. Fast and secure test methods with their inherent concept, methodology, results and discussion.
  3. Discussion of the results, including comparison between secure test methods, comparison with state-of-the-art, security analysis and list of recommendations on how to securely test FE.
  4. Classification of methods to improve test quality and implementation of one of these methods.

1.5. THESIS ORGANIZATION

THE remaining of this thesis is organized as follows. Chapter 2 presents the publications related with reliability characterization and improvement of secure ICs. Chapter 3 presents the publications related with testing secure devices. Chapter 4 presents the conclusions and future work.

2

RELIABILITY CHARACTERIZATION AND IMPROVEMENT FOR SECURE ICS

The content of this chapter includes the following research articles:

1. ***M. Cortez**, A. Dargar, S. Hamdioui, G.-J. Scrijen, Modeling SRAM Start-Up Behavior for Physical Unclonable Functions, IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFTS), pp. 1-6, 3-5 October 2012, Austin, TX, USA.*
2. ***M. Cortez**, S. Hamdioui, R. Ishihara, Design Dependent SRAM PUF Robustness Analysis, Latin-American Test Symposium (LATS), pp. 1-6, 25-27 March 2015, Puerto Vallarta, Mexico.*
3. ***M. Cortez**, S. Hamdioui, V. vd Leest, R. Maes, G.-J. Schrijen, Adapting Voltage Ramp-up Time for Temperature Noise Reduction on Memory-based PUFs, IEEE International Symposium on Hardware-Oriented Security and Trust (HOST), pp. 35-40, 2-3 June 2013, Austin, TX, USA.*
4. ***M. Cortez**, S. Hamdioui, A. Kaichouhi, V. vd Leest, R. Maes, G.-J. Schrijen, Intelligent Voltage Ramp-up Time Adaptation for Temperature Noise Reduction on Memory-based PUF Systems, IEEE Transactions on Computed Aided Design of Integrated Circuits and Systems (TCAD), pp. 1162-1175, volume 34, issue 7, July 2015.*

2

Modeling SRAM Start-Up Behavior for Physical Unclonable Functions

Mafalda Cortez Apurva Dargar Said Hamdioui

Delft University of Technology
Faculty of EE, Mathematics and CS
Mekelweg 4, 2628 CD Delft, The Netherlands
{A.M.M.O.Cortez, S.Hamdioui}@tudelft.nl

Geert-Jan Schrijen

Intrinsic-ID B.V.
High Tech Campus 9,
Eindhoven, The Netherlands
Geert.Jan.Schrijen@intrinsic-id.com

***Abstract*—One of the emerging technologies for cryptographic key storage is hardware intrinsic security based on *Physical Unclonable Functions* (PUFs); a PUF is a physical structure of a device that is hard to clone due to its inherent, device-unique and deep-submicron process variations. SRAM PUF is an example of such technology that is becoming popular. So far, only a little is published about modeling and analysis of their *start-up values* (SUVs). Reproducing the same start-up behavior every time the chip is powered-on is crucial to produce the same cryptographic key. This paper presents an analytical model for SUVs of an SRAM PUF based on *Static Noise Margin* (SNM), and reports some industrial measurements to validate the model. Simulation of the impact of different sensitivity parameters (such as variation in power supply, temperature, transistor geometry) has been performed. The results show that out of all sensitivity parameters, variation in threshold voltage is the one with the highest impact. Industrial measurements on real memory devices validate the simulation results.**

I. INTRODUCTION

The industry is recognizing the importance of hardware security to combat semiconductor device counterfeiting, theft of service and tampering, for which secure cryptographic key storage is an essential component. Traditional methods use *Non-Volatile Memories* (NVMs) to permanently store key/data, which are highly prone to physical attacks [1–3]; hence, the methods are no longer secure. Ideally, the cryptographic key would *not* be permanently stored in the system but generated *only* when required. One of the emerging technologies satisfying this requirement is hardware intrinsic security based on PUFs. A PUF is an inherent function that is embedded in a physical structure, such as an *Integrated Circuit* (IC). A PUF is hard to clone due to its inherent, device-unique and deep-submicron *process variations* (PVs). When challenged, a PUF generates a response based on the unique *fingerprint* inherent in an IC. There are several types of PUFs such as Optical PUF [4], Coating PUF [5], Silicon PUF [6], Flip-Flop PUF [7], Butterfly PUF [8] and SRAM PUF [9]. Because SRAM PUFs are standard components and easy to manufacture, no extra effort is invested for their implementation. Therefore, SRAM PUFs are one of the most popular PUF types today [6,10,11].

Although SRAM cells are symmetrical, small and random deviations during manufacturing process cause an intrinsic mismatch. SRAM PUF fingerprints are a consequence of the mismatch in SRAM cells. When powered-up, due to this mismatch, the cells take their preferred values - either a logic 0 or logic 1. Each SRAM cell provides one fingerprint bit. The SRAM cells *start-up values* (SUVs) together generate a fingerprint that uniquely identifies each device. This fingerprint is further processed to generate a unique cryptographic key. To be used as a source for key generation, the fingerprint needs to be reproducible over time, even under changing environmental conditions. Thus, it is crucial to understand the different parameters impact on the fingerprints robustness to design reliable SRAM PUF based systems.

Even though SRAM PUFs are becoming popular, very limited work has been published about modeling the robustness of its SUVs, not to mention actual silicon verification. In [10], the authors used soft decision information in helper data algorithms to correct the SUVs of non-robust cells. In [12] and [13], the authors proposed the use of SRAM for *Field Programmable Gate Array* (FPGA) Intellectual Property protection and studied SRAM PUF fingerprint statistical characteristics, such as entropy. However, their work was not directed towards the physical randomness source that causes fingerprints. In [14], the authors presented a technique called stable-PUF-marking to identify robust SRAM cells; only these cells are used for cryptographic key generation as an alternative for error correction. However, the authors assumed that the cells mismatch is based on the threshold voltage alone. In [15], the authors studied the impact of non-technology parameters (e.g., temperature) on the robustness of SRAM fingerprints. However, the work did not consider the impact of technology parameters such as transistor channel length. Understanding the impact of both technology and non-technology parameters on the SUVs enables the design of robust and reliable SRAM PUFs based systems. An appropriate model is therefore needed.

This paper presents an analytical model of start-up behavior of an SRAM. The model is further used to perform a sensitivity analysis to identify the impact of different technology and non-technology parameters. Validation of the model is done by comparing simulation results with silicon measurements.

The rest of this paper is organized as follows. Section II briefly reviews key storage based on PUFs, the *six transistors* (6Ts) SRAM cell and classifies it according to its ability to reproduce the same start-up behavior. Section III introduces the analytical model based on SNM. Section IV gives the simulation results. Section V reports silicon measurements and compares them with the obtained simulation results. Finally, Section VI concludes this paper.

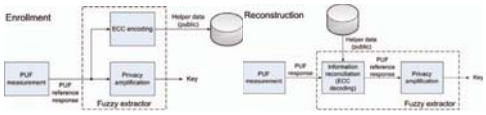
978-1-4673-3044-2/12/$31.00 © 2012 IEEE

II. Background on Secure Cryptographic Key Storage based on SRAM PUFs

This section provides background information of PUFs based systems and briefly gives an SRAM cell architecture and behavior overview. In addition, it proposes a classification of SRAM cells upon the reproducibility of their SUVs.

A. Key Storage System based on PUFs

PUFs in general, SRAM PUFs in particular, can be used as a secure cryptographic key storage mechanism [16]. Fig. 1 shows how such mechanism can be integrated to create a PUF based key storage system. Such a system performs two main operations; they are explained next.



(a) (b)

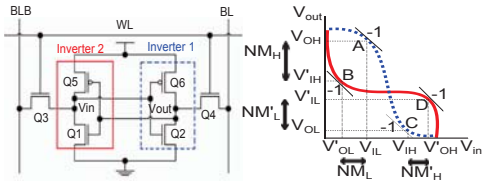
Fig. 1: Operations of a PUF based Key Storage System (a) Enrollment and (b) Reconstruction [17].

1) **Enrollment:** this operation generates a key based on a PUF fingerprint. This key is *programmed* into the device to be protected. This operation can be subdivided in three steps. First, the response of the targeted PUF is measured. This response is called *PUF reference response*. Second, this response is used as the input of the *Fuzzy Extractor* (FE) [18–20], which derives a cryptographic key and computes *Helper data* using ECC coding. Third, the Helper data is stored in a NVM attached to the device and is made as public information.
2) **Reconstruction:** this operation recovers the programmed key. It can be divided in two steps. First, the response of the targeted PUF is measured. This response is called *PUF response*; see Fig. 1(b). Second, this response is used as input of the FE; here, FE uses the stored Helper data and the new response to reconstruct the cryptographic key that was programmed during enrollment. If the measured PUF response is close enough to the PUF reference response (i.e., within the ECC correction capability, typically 25% [17]), the original key is successfully reconstructed.

It is then *crucial* to reproduce the same PUF reference response generated at enrollment during the key reconstruction phase within the error correction capabilities of the ECC.

B. SRAM cell and classification

The popular 6Ts SRAM cell (see Fig. 2(a)) consists of two cross-coupled CMOS inverters formed by four transistors (Q1 with Q5 and Q2 with Q6) and two pass transistors (Q3 and Q4). The pass transistors are used to access the cell for read and write operations. The bitline (BL), the compliment bitline (BLB) and the wordline (WL) are used to access the cell.



(a) 6Ts CMOS SRAM cell (b) VTCs of an SRAM cell

Fig. 2: SRAM cell (a) schematic and (b) VTCs

To be used for cryptographic key generation, it is required that the SUVs of the majority of the SRAM cells are reproducible, even under hostile conditions such as high temperature [15]. Therefore, SRAM cells are classified depending upon the sensitivity of its SUVs to stress conditions as follows:

1) **Non-skewed cell:** the cell has *no* measurable mismatch between its two inverters. This does not mean that PV did not occur in the cell, but just that the combined effects neutralize each other. A non-skewed cell generates randomly either a 0 or 1 at its output, depending mainly upon the noise present in the system.
2) **Partially-skewed cell:** the cell has a *little* mismatch between its two inverters. These kind of cells have a preferred state, depending upon the nature of the mismatch. Therefore, the cell can flip (hence, produce a different SUV) due to variation of external conditions such as the temperature.
3) **Fully-skewed cell:** the cell has a *high* mismatch between its two inverters in such a way that the cell always takes its preferred initial state regardless of the stress conditions. Ideally, SRAM PUFs have majority cells of this type.

III. Analytical Model for SRAM PUFs

In this section the concept of Static Noise Margin (SNM) is used to develop an analytical model for SRAM SUVs. First, the SNM is briefly reviewed. Then, a model is presented. Finally, a classification of parameters that could impact SRAM PUF SUVs is given.

A. SNM concept

SNM is the metric for quantifying the maximum noise voltage that an SRAM cell can tolerate before changing its state. SNM is calculated as the shortest side of the largest square that can fit inside the eyes of the *Voltage Transfer Curves* (VTCs) of the cross-coupled inverters that compose the cell; see Fig. 2(b). The dashed curve presents the VTC of Inverter 1 and the solid that of Inverter 2. The intersection of these lines forms two eyes. The side of the largest square that can fit inside *both* eyes is the SNM value [21]. To find the SNM value, the coordinates of four critical points A, B, C and D as shown in Fig. 2(b) have to be determined.

The traditional SNM model proposed by [21] takes all 6Ts into account as all of them affect the SRAM cell stability. The calculation is made for read-access mode as it is the worst case scenario. It is known that cell asymmetries are due

to PV affecting the size of the VTCs eyes [15,22]. Hence, by determining the relative size of the eyes, it is possible to determine the cell's preferred state. Perfectly symmetrical eyes indicate a non-skewed cell, small asymmetry between the eyes indicate a partially-skewed cell and a large asymmetry indicates a fully-skewed cell [23].

The traditional SNM model cannot be directly used to analyze the SUVs of SRAM cells because: (a) SUVs are generated during power-up and not during read-access mode, (b) the transistors that play a major role in determining the SUVs of SRAM cells are the ones forming the cross-coupled inverters, (c) pass transistors of SRAM PUF have no impact since the WL is not active and (d) SUVs are not only determined by the noise tolerance of the cell but also by the relative strength of SRAM cells inverters. Hence, a new *PUF SNM* (PSNM) is needed.

B. SRAM PUF Static Noise Margin (PSNM)

To determine the value of PSNM, we assume that only the noise and the mismatch of the cross-coupled inverters may impact the SUVs. As shown in Fig. 2(b), the PSNM square size depends on the coordinates of the four critical points denoted by A (V_{IL}, V_{OH}), B (V'_{OL}, V'_{IH}), C (V_{IH}, V_{OL}) and D (V'_{OH}, V'_{IL}). For each of the four points, the transistors involved are either in linear or saturation mode, assuming noise levels above the threshold voltage [24]. At point A, Q_2 is in saturation mode and Q_6 is in linear mode; at point B, Q_1 is in linear mode and Q_5 is saturation mode; at point C, Q_2 is in linear mode and Q_6 is in saturation mode, while at point D, Q_1 is in saturation mode and Q_5 is in linear mode. To calculate the coordinates of each of the critical points we performed the following steps. Due to space limitations we present the procedure and results only for point A; a similar approach is performed on points B, C and D [23].

1) Write the drain current equations for the transistors in their respective modes of operation. For point A, $I_{D_{Q2}} = I_{D_{Q6}}$. This results into:

$$\beta_2 \quad (V_{in} - V_{th_2})^2(1 + \lambda_2 V_{out}) = \beta_6[2(V_{in} - V_{dd} - V_{th_6})(V_{out} - V_{dd}) - (V_{out} - V_{dd})^2] \qquad (1)$$

where $\beta_{2,6}$ are the transconductances, $\lambda_{2,6}$ are the channel length modulation parameters, $V_{th_{2,6}}$ are the threshold voltages of $Q2$ and $Q6$ respectively, V_{out} and V_{in} are respectively the output and input voltage of Inverter 1 (see Fig. 2(a)), and V_{dd} is the supply voltage.
2) Differentiate the equations obtained in step 1 with respect to V_{in} and then replace the derivative with $\frac{dV_{out}}{dV_{in}} = -1$.
3) Utilize the equations in steps 1 and 2 to derive an expression for the coordinates of the critical point A; this results into:

$$V_{OH} = \frac{\frac{\beta_2}{\beta_6}(V_{IL} - V_{th_2}) - \frac{1}{2}\frac{\beta_2}{\beta_6}\lambda_2(V_{IL} - V_{th_2})^2}{2 - \frac{\beta_2}{\beta_6}\lambda_2(V_{IL} - V_{th_2})} \frac{+V_{IL} - V_{th_6} + V_{dd}}{2 - \frac{\beta_2}{\beta_6}\lambda_2(V_{IL} - V_{th_2})} \qquad (2)$$

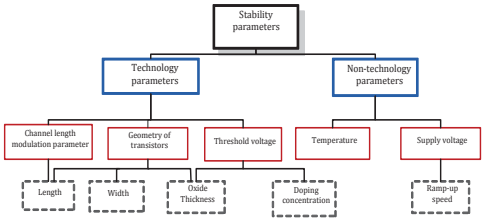


Fig. 3: Classification of sensitivity parameters for SRAM PUFs

V_{IL} is obtained by substituting V_{out} of Eq. 1 into Eq. 2.
4) Calculate the smallest of the noise margins (NM) per VTC of as:
   - NM = min $(NM_H = V_{OH} - V'_{IH}, NM_L = V_{IL} - V'_{OL})$
   - NM'= min $(NM'_H = V'_{OH} - V_{IH}, NM'_L = V'_{IL} - V_{OL})$.
5) Determine two metrics:
   a) $PSNM_{ratio}$ as NM/NM'. The preferred value of the SRAM cell is 1 if $PSNM_{ratio}$ is greater than 1 and 0 if $PSNM_{ratio}$ is smaller than 1. The higher or lower the $PSNM_{ratio}$ than 1, the higher the asymmetry within its cross-coupled inverters; hence, the more reproducible its SUVs.
   b) $PSNM_{noise}$ = min(NM, NM'). The higher the $PSNM_{noise}$ the higher the tolerance of the cell to the noise.

C. Classification of SRAM PUF stability parameters

Inspecting Eq. 1 and Eq. 2, used to calculate both PSNM metrics, reveal that the following parameters can impact the SUV:

- Channel length modulation λ; this parameter strongly depends on the transistor length L [25];
- MOSFET transconductance β; this parameter depends on the transistor length L, transistor width W and the gate oxide thickness t_{ox} [25];
- Threshold voltage V_{th}; this parameter is determined mainly by gate oxide thickness t_{ox}, intrinsic doping carrier concentration n_i, donor and acceptor doping carrier concentration $N_{D,A}$ and temperature T [25];
- Supply voltage V_{dd}. Note that *voltage supply ramp-up speed* t_r is also known to impact SUV stability [15]. Nevertheless, the proposed model does not deal with t_r; this needs a new model (ongoing work).

PSNM sensitivity parameters can be classified into two groups: technology and non-technology; see Fig. 3. We assume that technology parameters are the ones that are directly dependent upon the technology node such as L, and non-technology parameters are the ones that can be controlled externally such as T and V_{dd}. Note that the temperature is orthogonal to n_i and V_{th}.

The two previously defined metrics can be used to study the SUVs reproducibility. $PSNM_{ratio}$ can be used for technology parameters as these are the ones that cause the inverters' intrinsic mismatch; this metric provides the relative strength of one inverter as compared to the other. $PSNM_{noise}$ can be

TABLE I: Parameters for $65nm$ BSIM4 model

Parameter	NMOS	PMOS
Temperature T (in °C)	20	20
Supply voltage V_{dd} (in V)	1.2	1.2
Length L (in nm)	65	65
Width W (in nm)	195	130
Threshold voltage V_{th} (in V)	0.423	0.365
Gate Oxide Thickness t_{ox} (in nm)	1.85	1.95

used for the non-technology parameters as these are the ones that can vary the noise tolerance of the cell during operation (after manufacturing). Moreover, these parameters influence all the cell components in a homogeneous way.

IV. Simulation results

In this section, we analyze the impact of technology parameters and the combination of technology and non-technology on PSNM. First, the set-up and experiments are described. Thereafter, the results are presented and discussed.

A. Set-up

We simulate the start-up behavior of an SRAM cell using SPICE and BSIM4 $65nm$ models [26]. The CMOS parameters nominal values used in the simulations are listed in Table I. Note that analyzing the impact of non-technology parameters *alone* is not realistic as PV is always present.

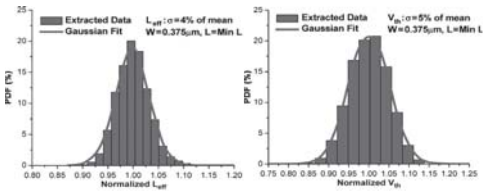
We perform two types of experiments: (1) We vary one technology parameter of one of the MOSFETS of Inverter 1 at a time and analyzed its impact on both $PSNM_{ratio}$ and $PSNM_{noise}$ and conclude about which parameter has the most impact on the reproducibility of the SUVs, (2) we introduce a mismatch on a cell by means of the most dominant parameter and determine the impact of each non-technology parameter on both $PSNM_{ratio}$ and $PSNM_{noise}$.

B. Impact of technology parameters

We performed four experiments in which we vary a single parameter per experiment; these are L, W, V_{th} or t_{ox}. The experiments reveal that the impact of technology parameters on $PSNM_{noise}$ is negligible; e.g., increasing the NMOS V_{th} by +10% increases $PSNM_{noise}$ by only 0.7%. The results on $PSNM_{ratio}$ are reported next.

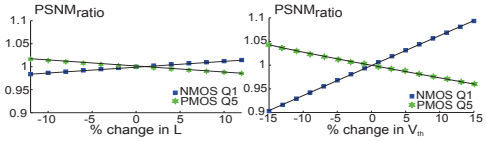
1) Impact of the transistor length L: We simulate the start-up behavior for different values of L, up to $\pm 12\%$ with a step of 2%. This variation corresponds to the worst case scenario for $65nm$ node, where the ratio of standard deviation to mean variation (σ) for L due to PV is $\pm 4\%$ [27]; see Fig. 4(a). The figure shows the PV *Probability Distribution Function* (PDF) of L for this technology. Note that the impact of λ is also reflected in L due to their interdependency.

Fig. 5(a) shows the results of the performed simulation; they reveal the following: (a) $PSNM_{ratio}$ is linearly dependent on L, (b) $PSNM_{ratio}$ indicates that the preferred value of the cell is 1 for an increasing in NMOS L or a decreasing PMOS L, (c) the preferred value of the cell is 0 for a decreasing NMOS L or an increasing PMOS L, and (d) the percentage change in $PSNM_{ratio}$ due to both PMOS and NMOS is similar for same variation in L; e.g., a variation of +10% in PMOS L varies $PSNM_{ratio}$ with 1.4%.



(a) length (b) threshold voltage

Fig. 4: Process variation PDF for $65nm$ [27]



(a) length (b) threshold voltage

Fig. 5: Impact of length and threshold voltage on $PSNM_{ratio}$

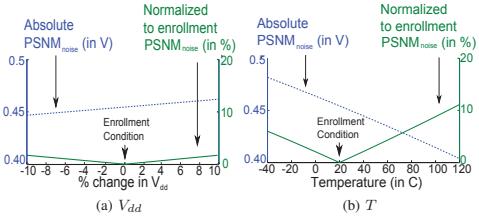
2) Impact of the transistor width W: We simulate the impact of W on start-up behavior in a similar way as we did for L. The results show the same trend as that observed for L, but with opposite effect, e.g., a decrease of W of NMOS results in a $PSNM_{ratio}$ above 1, hence, preferred value 1. Moreover, W has a similar impact as that of L variation.

3) Impact of the transistor threshold voltage V_{th}: We simulate the start-up behavior for different values of V_{th} up to $\pm 15\%$ with a step of 2%. This variation corresponds to the worst case scenario, where σ for V_{th} due to PV is $\pm 5\%$ [27]; see also Fig. 4(b). The simulation results are given in Fig. 5(b); based on the figure we can conclude that (a) the variation in V_{th} has a severe impact on $PSNM_{ratio}$ for both NMOS and PMOS, (b) the impact of NMOS V_{th} variation is the double of that of PMOS; e.g., +10% in NMOS V_{th} increases the $PSNM_{ratio}$ by 6%, (c) $PSNM_{ratio}$ indicates that the preferred value of cell is 1 for an increasing NMOS V_{th} or a decreasing PMOS V_{th}, and (d) $PSNM_{ratio}$ indicates that the preferred value of cell is 0 for a decreasing NMOS V_{th} or an increasing PMOS V_{th}.

4) Impact of the transistor gate oxide thickness t_{ox}: The t_{ox} for $65nm$ node is in the order of $2nm$, i.e., 4 to 5 atoms [28]. The roughness introduced by PV, although small between silicon and silicon dioxide, can be of one or two atomic layers [28]. For the given technology node, t_{ox} for both PMOS and NMOS is indicated in Table I. Since there was no available distribution function for t_{ox} for this technology node, we assumed the worst case variation up to $\pm 30\%$ with a step of 10% and analyzed its impact. The simulation results show similar trends as that of V_{th}; see Fig. 5(b). However, the impact of t_{ox} is $2\times$ less severe than that of V_{th}.

C. Combined impact of stability parameters

The objective of this experiment is to investigate the impact of different supply voltages (i.e., $\pm 10\% V_{dd}$) and temperatures (i.e., from -40°C up to 120°C) on the $PSNM_{noise}$ in a cell with a mismatch in the most dominanting technology



(a) V_{dd} (b) T

Fig. 6: Impact of $PSNM_{noise}$

parameter, which is NMOS V_{th} according to our simulation results. It is worth noting that when considering the non-technology parameters only, the impact on $PSNM_{ratio}$ is negligible; simulation results show that (a) a temperature decrease from 20°C to -40°C increases $PSNM_{ratio}$ by 0.07% and (b) that a supply voltage increase of +10% increases $PSNM_{ratio}$ by 0.01%. The impact results on $PSNM_{noise}$ are reported next.

1) Impact of V_{dd} variation for NMOS V_{th} mismatched cell: For these simulations, a mismatch is introduced on the SRAM cell by increasing the V_{th} of the NMOS transistor Q_1 by 5%. We simulate the start-up behavior for different voltage values up to $\pm 10\% V_{dd}$ with a step of 2% and determined its $PSNM_{noise}$. Fig. 6(a) presents the results; the variation in V_{dd} is represented on the x-axis whereas the y-axis represents the absolute (left) and normalized to enrollment (right) $PSNM_{noise}$ for a particular variation. The figure shows that the impct of V_{dd} on the $PSNM_{noise}$ is negligible for the considered range of values. Absolute $PSNM_{noise}$ increases linearly with V_{dd} increase; e.g., +10% increase in V_{dd} increases $PSNM_{noise}$ by 1.7%. Normalized to enrollment $PSNM_{noise}$ decreases linearly with V_{dd} increase/decrease; e.g., $\pm 10\% V_{dd}$ decreases $PSNM_{noise}$ by 1.7%.

2) Impact of T variation for NMOS V_{th} mismatched cell: For these simulations, the previously NMOS V_{th} mismatch is considered. Fig. 6(b) shows the simulation results for the range of T values considered. The variation T is represented on the x-axis whereas the y-axis represents the absolute (left) and normalized to enrollment (right) $PSNM_{noise}$ for a particular variation. The figure shows that the impact of T on the $PSNM_{noise}$ is severe. Absolute $PSNM_{noise}$ decreases linearly with T increase; e.g., an increase in T from -40°C to 120°C decreases the $PSNM_{noise}$ by 19.3%. Normalized to enrollment $PSNM_{noise}$ decreases linearly with T increase/decrease; e.g., -40°C decreases $PSNM_{noise}$ normalized to enrollment by 6%. However, small variations around enrollment T, e.g., $\pm 10\% T$ $PSNM_{noise}$, have negligible impact.

D. Discussion

An SRAM PUF must have a majority of fully-skewed cells to be reproducible (see Section II). Moreover, an SRAM fingerprint is considered to be reproducible if at least 75% of its SUVs are reproducible. In other words, if the maximum of its non-reproducible SUVs are within the error capability of its ECC, i.e., 25% [17].

Our simulation results showed that from all sensitivity parameters, NMOS V_{th} is the one with the most impact on $PSNM_{ratio}$ and therefore on the reproducibility of SRAM SUVs. To compute the minimum $PSNM_{ratio}$ between two SRAM PUF cell inverters that will reproduce the same SUV (to be fully-skewed), we consider the Gaussian distribution of V_{th}; see Fig. 4(b). From the figure we need to extract the NMOS V_{th} variation that corresponds to 25%. The Gaussian distribution equation is:

$$P(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2} \tag{3}$$

where σ is the standard deviation of V_{th}, x represents the variation in V_{th} and μ is the mean of V_{th}. The V_{th} variation a that corresponds to 25% of the cells is:

$$25\% = \int_{\mu-a}^{\mu+a} P(x)\,\mathrm{d}x \implies a = 1.6\% \tag{4}$$

where μ = 1 and σ = 0.05 [27]; see Fig. 4(b). The minimum $PSNM_{ratio}$ for which an SRAM PUF cell starts being fully-skewed is 1.005 if 1 skewed, or 0.995 if 0 skewed; see Fig. 5(b). This calculation is done by assuming the variation in one MOSFET parameter at a time. Although this count may vary when considering all sensitivity parameter variations, this calculation indicates that the cell has a high probability of being fully-skewed; hence, reproducible.

V. SILICON RESULTS AND VALIDATION

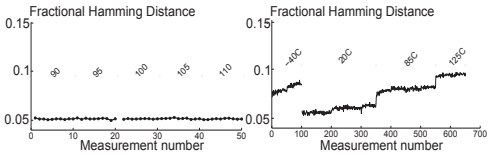
To validate the developed model and have better feeling about the reality, industrial experiments are performed on TSMC and NXP SRAM devices, 20 each, produced in $65nm$ node; all memory devices have a size of 65536 bits. Two experiments are performed to analyze the impact of supply voltage and temperature. In the rest of this section first the results of these experiments are presented and thereafter compared with the simulation results to validate the proposed model.

A. Supply voltage experiment

The SUVs of each of the above mentioned memory devices are measured for five V_{dd} values (i.e., $-10\% V_{dd}$, $-5\% V_{dd}$, V_{dd}, $+5\% V_{dd}$, and $+10\% V_{dd}$) at $20°C$. Each device is powered-up repeatedly ten times with intervals of one second; after each power-up, the SRAM SUVs are read and stored in a binary dump, which are then analyzed using MATLAB.

Fig. 7(a) shows the reproducibility analysis of the measurements performed on a single TSMC device at different V_{dd}. The remaining devices follow the same trend. The metric used to analyze the reproducibility is *Fractional Hamming Distance* (FHD); FHD gives a percentage of the total number of SUVs that have *different* values when compared to enrollment. Ideally, FHD should be zero. In our case, enrollment is performed at nominal V_{dd}. Fig. 7(a) shows that V_{dd} has a negligible impact on FHD.

The experiment was redone for NXP devices. The results show similar trends as those obtained for TSMC devices, but



(a) V_{dd} (b) T

Fig. 7: FHD for several enrollment conditions

with a FHD 1.5x higher. It can be concluded that the probability of reproducing the same SRAM SUVs is marginally impacted by V_{dd} variations.

B. Temperature cycle experiment

In this experiment SUVs are measured for different temperatures: $-40°C$, $20°C$, $80°C$ and $125°C$ with a V_{dd} of 1.2V. Each device is powered-up repeatedly 100 times for $-40°C$ and $125°C$, 250 times for $20°C$ and 200 times for $80°C$ with intervals of one second; after each power-up, the SUVs of the memories are read and stored in a binary dump.

Fig. 7(b) shows the FHD for a single TSMC device at different T; it reveals that FHD decreases for both higher and lower T as compared with enrollment; e.g., at $125°C$ FHD is 10% and $-40°C$ is 7%. As it can be seen a variation of $165°C$ in T results only in 10% variation in FHD. A similar experiment was performed on NXP devices and the results shows similar trends, but with a FHD 1.2x higher.

C. Comparison of measurements with simulation results

Both $PSNM$ metrics analysis are performed for a single cell. FHD analysis is performed for 65536 cells (bits) per SRAM device for 40 devices in total. $PSNM$ metrics are therefore a trend indicator of FHD. Next, simulation results are compared with silicon measurements.

1) For supply voltage experiment: Both simulation results and silicon measurements have shown that for 65nm technology node, a variation in the supply voltage V_{dd} has a negligible impact on the SUV reproducibility. Therefore, as silicon measurements follow the same trend as the simulation results, the correctness of the simulation results with regard to V_{dd} behavior can be concluded.

2) For temperature cycle experiment: Simulation results predict a $PSNM_{noise}$ when normalized to enrollment of 6% at $-40°C$ and of 11% for $120°C$. Silicon measurements have shown the same trend but $2\times$ less severe than the simulations predictions. This might be due to the consideration of a single parameter at a time during simulations. Note that the more variation within the SRAM cell technology parameters, the higher the $PSNM_{ratio}$ and hence the SRAM cell stability.

Note that the behavior of the silicon data matches with that of the simulation results from the analytical model, with regard to T.

VI. Conclusion

In this paper an SRAM start-up behavior model based on SNM is developed and the impact of both technology and non-technology parameters on the reproducibility of such behavior is analyzed. First, considering only variations on technology parameters, our model reveals that NMOS V_{th} has the most significant impact. Second, it shows that the reproducibility for combined variations in technology and non-technology parameters around enrollment conditions is more sensitive to V_{dd}. Furthermore, large T variations impact severely the reproducibility. These results are validated by silicon data.

References

[1] http://www.cl.cam.ac.uk/sps32/SG_talk_OSSC_a.pdf
[2] C.Y. Ng *et al.*, "RFID Privacy Models Revisited", *13th European Symp. on Research in Computer Security*, pp. 251-266, 2008.
[3] A.R Sadeghi and D. Naccache, "Towards Hardware-Intrinsic Security: Foundations and Practice", *Springer*, 2010.
[4] R. Pappu, "Physical one-way functions", *Massachusetts Institute of Tech.*, PhD thesis, 2001.
[5] P. Tuyls *et al.*, "Read-proof hardware from protective coatings", *Cryptographic Hardware and Embedded Systems Workshop*, 2006.
[6] http://homes.esat.kuleuven.be/ rmaes/puf.html, 2011.
[7] R. Maes, P. Tuyls and I. Verbauwhede, "Intrinsic PUFs from Flip-flops on Reconfigurable Devices", *3rd Benelux Workshop on Information and System Security*, 2008.
[8] S.S. Kumar *et al.*, "The butterfly PUF protecting IP on every FPGA", *IEEE Int. Workshop on Hardware-Oriented Security and Trust*, pp. 67-70, 2008.
[9] J. Guajardo *et al.*, "FPGA Intrinsic PUFs and Their Use for IP Protection", *Workshop on Cryptographic Hardware and Embedded Systems*, 2007.
[10] R. Maes, P. Tuyls and I. Verbauwhede, "A soft decision helper data algorithm for SRAM PUFs", *IEEE Int. Symp. on Information Theory*, pp. 2101-2105, 2009.
[11] R. Maes, P. Tuyls and I. Verbauwhede, "Low-Overhead Implementation of a Soft Decision Helper Data Algorithm for SRAM PUFs", *Workshop on Cryptographic Hardware and Embedded Systems*, 2009.
[12] J. Guajardo *et al.*, "Physical Unclonable Functions and Public-Key Crypto for FPGA IP Protection", *Field Programmable Logic and Applications*, pp.185-195, Aug. 2007.
[13] J. Guajardo *et al.*, "FPGA Intrinsic PUFs and Their Use for IP Protection", *Workshop on Cryptographic Hardware and Embedded Systems*, pp.63-80, Sept. 2007.
[14] M. Hofer and C. Boehm, "An Alternative to Error Correction for SRAM-Like PUFs", *Workshop on Cryptographic Hardware and Embedded Systems*, pp. 335-350, 2010.
[15] D.E. Holcomb *et al.*, "Power-Up SRAM State as an Identifying Fingerprint and Source of True Random Number", *IEEE Transactions on Computers*, vol. 58, no. 9, September 2009.
[16] B. Skoric, P. Tuyls and W. Ophey, "Robust key extraction from physical unclonable functions", *Applied Cryptography and Network Security, vol. 3531 of LNCS*, pages 99-135, Springer Berlin / Heidelberg, 2005.
[17] V. vd Leest, P. Simons and E. vd Sluis, "Buskeeper PUFs, a promising alternative to D Flip-Flop PUFs", *HOST*, 2012.
[18] X. Boyen, "Reusable cryptographic fuzzy extractors", *ACM Conference on Computer and Communications Security*, pages 82-91, 2004.
[19] Y. Dodis, L. Reyzin and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data", *LNCS 3027*, 2004.
[20] J.P. Linnartz and P. Tuyls, "New shielding functions to enhance privacy and prevent misuse of biometric templates", *Proceedings of AVBPA'03 Springer-Verlag*, pages 393-402, Berlin, Heidelberg, 2003.
[21] E. Seevinck, F.J. List and J. Lohstroh, "Static-Noise Margin Analysis of MOS SRAM Cells", *IEEE Journal of Solid-state Circuits*, vol. sc-22, no. 5, October 1987.
[22] A. Pavlov and M. Sachdev, "CMOS SRAM Circuit Design and Parametric Test in Nano-Scaled Technologies", *Springer*, 2008.
[23] A. Dargar, "Modeling SRAM Start-up Characteristics for Physical Unclonable Functions", *Delft Univ. of Tech.*, MSc. Thesis, 2011.
[24] A. Sedra and K. Smith, "Microelectronics Circuits", *Oxford*, 2004.
[25] S. Dimitrijev, "Principles of Semiconductor Devices", *Oxford Univ. Press*, 2006.
[26] http://ptm.asu.edu/, 2011.
[27] W. Zhao *et al.*, "Rigorous extraction of process variations for 65nm CMOS design", *European Solid State Device Research Conf.*, pp. 89-92, 2007.
[28] G.N. Silva and A. Chandrakasan, "Leakage in Nanometer CMOS Technologies", *Springer*, 2006.
[29] H. Qin *et al.*, "SRAM Leakage Suppression by Minimizing Standby Supply Voltage", *Int. Symp. on Quality Electronic Design*, pp. 55-60, 2004.

2

Design Dependent SRAM PUF Robustness Analysis

Mafalda Cortez Said Hamdioui

Delft University of Technology
Faculty of EE, Mathematics and CS
Mekelweg 4, 2628 CD Delft, The Netherlands
{A.M.M.O.Cortez, S.Hamdioui}@tudelft.nl

Ryoichi Ishihara

Delft University of Technology
Delft Institute of Microsystems and Nanoelectronics (DIMES)
Feldmannweg 17, 2600 GB Delft, The Netherlands
R.Ishihara@tudelft.nl

***Abstract*—In this paper we evaluate and compare the robustness (i.e., repeatability and uniqueness) of two SRAM PUF designs, General Purpose (GP) and Low-Power (LP), by means of both circuit simulations and industrial measurements. Circuit simulations are performed on both designs while considering two technology nodes (45nm and 32nm), three temperatures and three voltage ramp-up times. Industrial measurements are performed to validate the simulation results. The simulation results as well as industrial measurements demonstrate that GP devices provide better repeatability for all investigated cases (up to 4.5× better) while the uniqueness is design independent.**

Keywords: PUF-based system, SRAM PUF, Robustness

I. INTRODUCTION

Physical Unclonable Functions (PUFs) are the embodiment of random and unique, but repeatable, mapping of challenges to responses in physical structures such as integrated circuits (ICs) [1]. The *uniqueness* and *repeatability* of this mapping, known as *fingerprint*, enables unambiguous identification of ICs making PUFs efficient hardware security primitives. Moreover, PUFs are hard to clone due to their random, uncontrollable, inherent, device-unique and deep-submicron process variations. Combined with proper post-processing, a PUF is able to generate secret keys of cryptographic strength, and reliably store them in a highly secure manner without the need for conventional on-chip *non-volatile memory* [2, 3]. However, PUF fingerprints have two main drawbacks. First, they are noisy; when the same challenge is consecutively applied to the same device, the mapped responses are slightly different even under the same operating conditions, resulting in lower repeatability. Second, the fingerprints of any two random devices might be slightly correlated, resulting in lower uniqueness.

To tackle PUF challenges and to guarantee robustness, i.e., uniqueness and repeatability, PUF-based systems use fuzzy extractors for both privacy amplification (to improve uniqueness) and error correction (to improve repeatability) [4]. However, the usage of fuzzy extractors comes at a cost which scales up with less robust bare PUF responses; reduced uniqueness is compensated with larger PUF footprints and reduced repeatability is compensated with *error-correcting code* (ECC) having larger error correction capability, resulting in an overall larger silicon area overhead [4]. Numerous PUF constructions have been proposed and implemented (see [5] for an overview); however, SRAM PUFs are one of the most popular as they are standard IC components and CMOS technology compatible [6–8]. Our work focuses on the robustness analysis of this PUF type.

Much work has been published regarding the robustness of SRAM PUF technology [6–12]. In [6], the authors studied the mismatch root-cause in SRAM cells; the work validated only SRAM repeatability for 65nm node low-power. In [7], the authors theoretically analyzed the impact that external factors have on the fingerprint uniqueness and repeatability. In [8], [9] and [10] the authors addressed techniques to improve fingerprints' statistical characteristics, such as fuzzy extractors and helper data algorithms. In [11], the authors presented a technique called stable-PUF-marking to identify robust SRAM cells; they proposed to use only these cells for cryptographic key generation as an alternative for error correction code of those which are non-robust. They assumed that the cells mismatch is based on threshold voltage only. No silicon results were presented to validate the findings. In [12], the authors presented a comparison between SRAM PUFs and Flip-Flop PUFs repeatability and uniqueness based on 65nm silicon results. This study focused on a single technology node.

The state-of-the-art clearly shows that although the robustness of SRAM PUF is quite addressed, limited silicon results were reported. Moreover, robustness for different SRAM PUF designs in cryptographic systems is not investigated yet; understanding this will allow the integration of the best design in cryptographic systems, resulting in better robustness and overall reduced cost.

In this paper, we evaluate and compare the robustness of two different SRAM PUF designs, general purpose (GP) and low-power (LP), for different technology nodes and under varying operation conditions, such as temperature and voltage ramp-up time. The paper has the following contributions:

- Repeatability and uniqueness evaluation for two SRAM PUF design based on circuit simulations; the analysis is performed for two different technology nodes (45nm and 32nm) while considering three temperatures and three voltage ramp-up times.
- Repeatability and uniqueness measurements performed on silicon devices (both GP and LP), manufactured using different technology nodes. Also here, the impact of different stress conditions is investigated.

The rest of this paper is organized as follows. Section II provides some preliminaries on PUF-based systems, including the SRAM cell, the two different designs under consideration and introduces the metrics used to evaluate the PUF robustness. Section III discusses the simulation setup, the performed experiments and the simulation results. Section IV provides the industrial analysis, including the characteristics of the devices measured, the performed measurements, the results, a comparison with simulation results and a discussion. Finally, Section V concludes this paper.

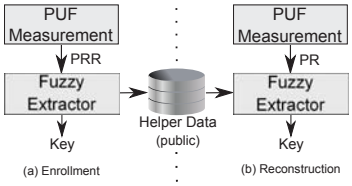


Fig. 1: PUF-based Key Generation and Storage System [6, 16]

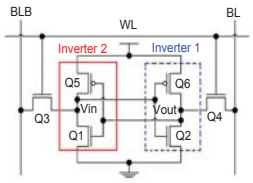


Fig. 2: 6T CMOS SRAM cell [6]

II. PUF-BASED SYSTEMS

In this section, we provide background information on PUF-based systems. First, we discuss their main operations. Second, we briefly provide some preliminaries on the basic operation of SRAM PUFs. Third, we summarize the main differences between general purpose and low-power devices. Finally, we introduce the robustness evaluation metrics.

A. PUF-based Key Generation and Storage

Fig. 1 shows the flow of a PUF-based key-storage system [3, 10] implemented with a *fuzzy extractor* (FE) [4, 13], which typically consists of two phases:

Enrollment: a key is generated from a *PUF Reference Response* (PRR). First, a PUF measurement produces the PRR. Next, the PRR is processed by the FE into a cryptographically strong *Key*, and helper data is generated as a FE byproduct. Finally, the helper data is stored in an external non-volatile memory (hence, becomes public information).
Reconstruction: the earlier enrolled *Key* is reliably recovered from a noisy *PUF Response* (PR) and the stored helper data. First, a PUF measurement produces the PR. Some bits of PR are different from original PRR; hence, PR is a noisy version of PRR. Next, PR is processed by the FE in combination with the helper data which is retrieved from the external memory. If the noisy PR is close enough to the PRR obtained during enrollment (i.e., the PUF response is repeatable up to a limited amount of noise), then the FE succeeds in reliably reconstructing the enrolled *Key*.

B. SRAM PUF

Fig. 2 shows the popular six-transistor SRAM cell. An SRAM cell is a bistable circuit, i.e., it has two possible states denoted as logic '0' and '1' and it comprises two cross-coupled inverters at its core, respectively formed by (*Q1*, *Q5*) and (*Q2*, *Q6*). The peripheral circuitry used to access the cell is comprised by two pass transistors (*Q3* and *Q4*), the bitline (*BL*), the complement bitline (*BLB*) and the wordline (*WL*).

When powered-up, the cross-coupled inverters start driving electric current, hence, increasing the voltages at their gates (V_{in} and V_{out}). The first inverter that builds enough gate voltage to drive its NMOS will pull-down its output, forcing the other inverter to pull-up and causing the SRAM cell to settle in one of both stable states. Since both inverters are designed to be nominally identical, the outcome (the states in which a cell settles) is entirely determined by the effect of random process variations. Hence, an SRAM power-up state, known as a *start-up value* (SUV), is a PUF response, and this construction is called an SRAM PUF.

C. SRAM Design

There are four designs that are optimized for different application requirements; they are *High-Performance* (HP) optimized for speed, *Low-Operating Dynamic Power* also known as *General Purpose* (GP) optimized for dynamic dissipation power, *Low-Standby Static Power* also known as *Low-Power* (LP) optimized for static power dissipation and *III-VGe* optimized for both low dynamic power and high speed operation. In our work, we focus on GP and LP devices, which are the ones manufactured for measurements. These designs differ on various parameters; a non-exhaustive list is summarized in Table I [14], comparing GP and LP devices. The table highlights the different configurations of both physical and electrical parameters for the same technology node, but different designs; V_{DD} is the supply voltage, L_{eff} is the effective gate length, t_{ox} is the thickness oxide, I_{on} is the leakage current during operation and I_{off} is the leakage current during idle. Comparing the two designs reveals that GP has a lower supply voltage, a shorter channel length and thickness oxide (hence, smaller threshold voltage), a similar operational current and two orders of magnitude higher leakage current. Our goal is to investigate the impact that these differences have on the robustness of SRAM PUFs, hence, on the overall cost of SRAM PUF-based systems.

D. PUF robustness metrics

PUF robustness in general, and of SRAM PUF in particular, can be evaluated by the repeatability and by the uniqueness of its fingerprint. Fingerprint's repeatability is the ability of a device to generate the same fingerprint every time it is powered-up. The higher the number of bits that always have the same SUV, the higher the repeatability of that device. Uniqueness is the ability of a fingerprint to be distinguished from other devices fingerprint. The higher the fingerprint randomness, the less the correlation between any two given devices (assuming same fingerprint lengths).

To evaluate our experiments impact on both repeatability and uniqueness, we rely on two widely used metrics; they are *Fractional Hamming Distance* (FHD) and *Fractional Hamming Weight* (FHW) [12]. A brief explanation of the metrics and how these are used to evaluate the robustness parameters is given next.

FHD is used to evaluate both repeatability and uniqueness. FHD calculates the percentage of bits that are different between two different PUF responses; e.g., the FHD between '0010' and '0110' is 25%. When FHD is used for repeatability, per device, each of the measured PUF responses (PR) is compared with the enrollment PUF response (PRR) and then

TABLE I: Different configurations of physical and electrical parameters

	45nm					32nm				
	V_{DD} (V)	L_{eff} (nm)	t_{ox} (nm)	I_{on} (μA/μm)	I_{off} (nA/μm)	V_{DD} (V)	L_{eff} (nm)	t_{ox} (nm)	I_{on} (μA/μm)	I_{off} (nA/μm)
GP	0.7	22	0.9	754.0	6.80E+0	0.6	16	0.8	750.3	9.53E+0
LP	1.0	25	1.3	752.2	5.78E-2	0.9	18	1.1	888.9	7.77E-2

normalized to the response length, see Fig. 1. An FHD per device close to 0% indicates a high repeatability. When FHD is used for uniqueness, the measured enrollment PUF response (PRR) of each PUF device is compared with that of all other PUF devices and then normalized to the response length. An FHD between devices close to 50% is a good indicator of uniqueness. Moreover, the metric' statistical significance increases with the number of PRs considered to calculate FHD per device (for repeatability) and with the number devices considered per FHD between devices (for uniqueness).

FHW is used to evaluate uniqueness; it computes the percentage of bits that are *not* zero; e.g. the FHW of '0010' is 25%. An FHW close to 50% indicates a balanced distribution of zeros and ones in the PUF responses. However, this metric is blind to logic values clustering.

III. SIMULATION BASED ANALYSIS

To analyze the repeatability and uniqueness of SRAM PUFs both for general purpose and low-power, a memory system comprising a cell and peripheral circuitry is synthesized and simulated using HSPICE and PTM models [18]. In this section, first, we present the PUF fingerprint generation. Thereafter, we describe the simulation experiments and results.

A. SRAM PUF Response Setup

Each bit of an SRAM PUF response is generated by an individual SRAM cell. Fig. 3 shows the SRAM fingerprint generation schematic used in our simulations. It has been shown in [6, 7] that the threshold voltage V_{th} of NMOS transistors is the technology parameter with the most impact on the start-up value of an SRAM cell. Hence, Monte Carlo is used to generate 300 random values of V_{th} for *Q1* (see Fig. 2) according to the distribution presented in [15], i.e., mean μ = standard NMOS V_{th} and deviation $\sigma = 9\% \cdot \mu$. These 300 SRAM cells combined create an SRAM cell array that generates a unique and random 300-bit response after power-up.

B. Performed Experiments

PUF-based systems are designed to reconstruct the enrolled key under extreme operation conditions. Hence, it is crucial to test the repeatability for extreme temperatures and voltage ramp-up times. However, it is only during enrollment that the uniqueness of the devices might be vulnerable due to the helper data. Therefore, uniqueness is only evaluated for enrollment condition.

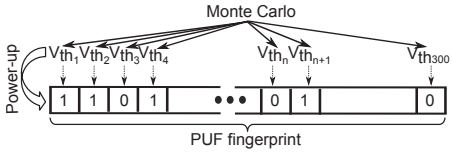


Fig. 3: SRAM PUF simulation [16]

To investigate the impact that technology scaling, temperature *Temp* and voltage ramp-up time t_{ramp} have on the repeatability and uniqueness, we simulated the power-up of the SRAM cell array for two experiment groups: repeatability experiments and uniqueness experiments.

Repeatability experiments: for each combination of design, technology node, *Temp* and t_{ramp}, we simulated the power-up of the SRAM cell array 20 times and evaluated its response. The transient noise during power-up is randomly generated by the simulation tool; hence, five variable parameters are used for the simulation:

- Design (GP, LP)
- Technology node (45nm, 32nm)
- Temperature (*Temp*) (-40^{o}C, 25^{o}C, 85^{o}C)
- Voltage ramp-up time (t_{ramp}) (10μs, 50μs, 90μs)
- Transient noise (different for each of the 20 power-ups)

Hence, a total of 720 simulations are performed (2 designs $\times$ 2 technology nodes $\times$ 3 *Temp* $\times$ 3 t_{ramp} $\times$ 20 transient noise).

Uniqueness experiments: at enrollment conditions (*Temp* = 25^{o}C and t_{ramp}= 10μs) for the considered designs and technology nodes, we analyzed a subset of the performed simulations results in the previous experiments with focus on uniqueness. Three variable parameters are used for the analysis:

- Design (GP, LP)
- Technology node (45nm, 32nm)
- Transient noise (different for each of the 20 power-ups)

Note that the simulations were carried out with the technology nodes mentioned above (45nm and 32nm) as the PTM models are available for both considered designs (GP and LP) and technology nodes.

C. Simulation Results

Fig. 4 and Table II show the simulation results for repeatability and uniqueness, respectively.

Repeatability: Fig. 4 shows the impact on FHD, hence, on SRAM fingerprint repeatability. The vertical axis (y axis) represents the FHD over the 20 measurements; the mean value is plotted in a gray box, the maximum value (max) over the 20 measurements in a white box and the standard deviation (std) as a black line in the center of the boxes. In the horizontal axis (x axis), the information is grouped first per *Temp* and thereafter per t_{ramp}. Each device is assigned a letter; a - 45nm GP, b - 32nm GP, c - 45nm LP and d - 32nm LP. Moreover, the enrollment conditions are highlighted in yellow (light gray if

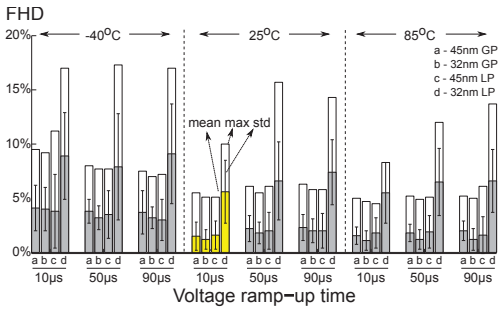


Fig. 4: Repeatability results - simulations

printed in black and white). From the figure, we can observe that GP and LP devices are impacted by *Temp*, t_{ramp} and technology scaling as follows:

Temperature impact at constant t_{ramp} (10μs): Regardless of the design, FHD (noise) is higher for extreme temperatures; e.g., for 45nm GP, mean FHD at -40^oC is 2.7× higher than that of 25^{o}C and 2.5× than that of 85^{o}C. Moreover, -40^{o}C has the highest max and std FHD.
Voltage ramp-up time at constant *Temp* (25^{o}C): GP devices are less sensitive to t_{ramp} variations than LP devices; e.g., 32nm GP FHD increases marginally for all different t_{ramp}, while 32nm LP max FHD increases 1.5×. Low-Power devices vulnerability to t_{ramp} variations can be explained by the slow response of these devices to frequency variation, as they have higher threshold voltage to minimize the leakage current.
Temperature and Voltage Ramp-Up Time Combined: Regardless of the design, for *Temp* lower than enrollment, a noise reduction is observed for t_{ramp} longer than enrollment; however, for *Temp* higher than enrollment, short t_{ramp} are the least noisy. For example, 45nm GP at -40^{o}C FHD is the lowest for 90μs, but for 85^{o}C FHD is lowest for 10μs. These results reveal a correlation between *Temp* and t_{ramp} that can be used to decrease noise by appropriate selection of t_{ramp} to the *Temp*. These results are in line with [16].
Technology scaling: LP devices are more sensitive to both *Temp* and t_{ramp} variations. Moreover, technology scaling reduces noise for GP designs, while it increases for LP designs. For example, GP FHD reduces by 0.93× with technology scaling while LP increases by 2×.

Uniqueness: Table II shows the impact that design type and technology node have on the uniqueness of an SRAM fingerprint. Note that uniqueness is evaluated only at enrollment condition. The table shows the FHW average over 20 measurements with its respective standard deviation. FHD between devices is not calculated as only one device per combination of design and technology node is simulated. The table reveals that, regardless of the technology node, both designs have a balanced distribution of 0s and 1s, indicating good uniqueness; e.g., 45nm GP has an FHW of 48% ± 2.3%.

IV. Industrial based analysis

The simulation results are validated using silicon devices. For this purpose, we perform measurements on two SRAM designs manufactured in two technology nodes (SRAM module providers not disclosed due to IP constrains). In this section, first, we introduce the devices and thereafter the measurements carried out. Finally, we present the repeatability and uniqueness results, compare them with the simulation results and discuss the differences between the two designs.

TABLE II: Uniqueness results - simulations

	45nm GP	32nm GP	45nm LP	32nm LP
FHW	48% ± 2.3%	53% ± 2.9%	49% ± 2.6%	54% ± 3.5%

A. Devices under consideration

To study the variation of repeatability and uniqueness of SRAM fingerprints, 100 SRAM devices distributed over two designs (65nm GP, 45nm GP, 65nm LP and 40nm LP), are evaluated. This information is summarized in Table III. Note that the simulated technology nodes and measured ones are not exactly the same; this because the available simulation models are limited. Nevertheless, the correlation and trends between simulations and silicon data can still be derived from them.

B. Performed Measurements

Similarly to the simulation experiments, we evaluate the repeatability and uniqueness of each of the SRAM devices (GP and LP), considering various temperatures, voltage ramp-up times and temperature and voltage ramp-up combined, as follows. Note that we consider only the first 2k bits of each device, as this is the typical size required to deploy a secure *Key* [21].

Temperature: we evaluate SRAM PUF repeatability for a wide range of *Temp* (-40^{o}C, 25^{o}C and 85^{o}C). The devices are placed in a climate chamber at 25^{o}C. Then, *Temp* is decreased to -40^{o}C. Next, *Temp* is increased to 25^{o}C and 85^{o}C. Finally, *Temp* is decreased to 25^{o}C. This is repeated twice. When the devices reach the temperatures of interest, 50 measurements are performed per device, making a total of 250 measurements per device. Between measurements, there is a power-off time of 1 sec, to make sure that all capacitances are discharged and hence prevent the impact on the new PUF response. Every temperature increase/decrease is performed at

TABLE III: SRAM devices under test characteristics

Technology	#	Geometry	Size (Bytes)
65nm	30	128×128	2048
65nm LP	6	2048×16	4096
	4	1024×160	20480
	4	4096×64	32768
	4	4096×320	163840
	4	4096×80	40960
	4	128×128	2048
	4	512×128	8096
45nm	5	16384×8	16384
	5	32×320	1280
	5	128×128	2048
	5	1024×128	16384
40nm LP	5	16384×8	16384
	5	32×320	1280
	5	1024×8	1024
	5	8×256	256

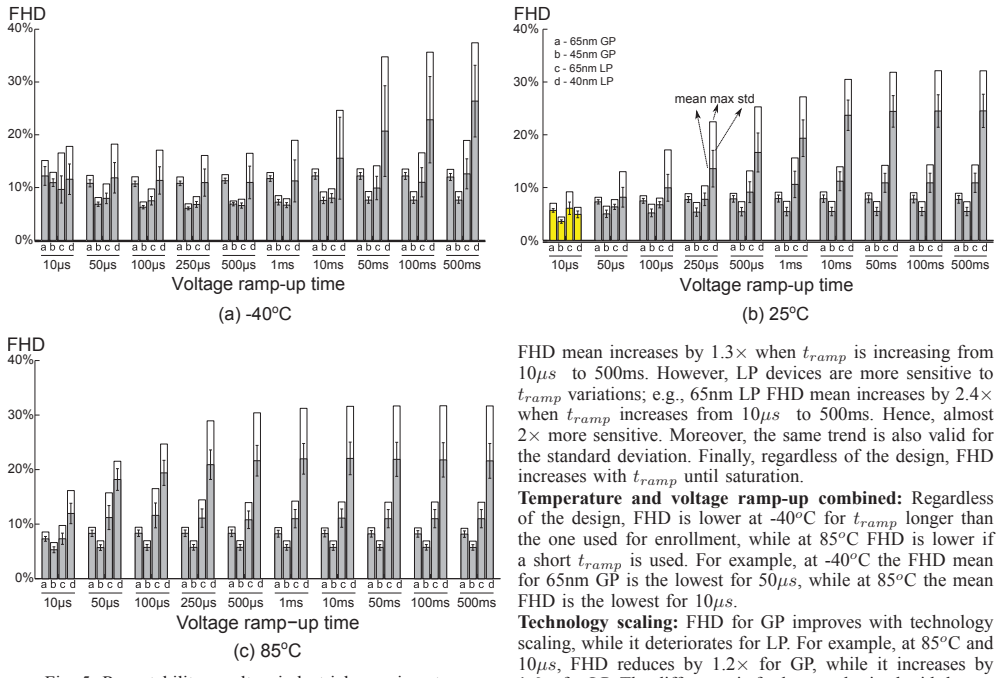


Fig. 5: Repeatability results - industrial experiments

a speed of 3°C/min. The settling time on desired temperature is 5 mins.

Voltage ramp-up time: we evaluate SRAM PUF repeatability for a wide range of t_{ramp} (10μs, 50μs, 100μs, 250μs, 500μs, 1ms, 10ms, 50ms, 100ms, 500ms). The devices are placed in a test set-up which is suitable for varying ramp-up time of the IC core voltage. At each t_{ramp} the devices are powered-up, read, powered-off for 1 sec and powered on again. There are 20 measurements taken per t_{ramp}.

Temperature and voltage ramp-up time combined: we repeat the voltage ramp-up test for each of the considered temperatures.

All measurement results are stored in a binary dump. These binary dump files are analyzed using MATLAB.

C. Measurement Results

Repeatability: Fig. 5 shows the repeatability results. From the figure, the following observations can be made.

Temperature impact at constant t_{ramp} (10μs): Regardless of the design, FHD is higher for extreme temperatures, in particular for -40°C; e.g., FHD at -40°C for 65nm GP is 2.2× higher than that of enrollment and 1.7× higher than that of 85°C.

Voltage ramp-up impact at constant *Temp* (25°C): GP devices are negligibly impacted by t_{ramp}; e.g., 65nm GP FHD mean increases by 1.3× when t_{ramp} is increasing from 10μs to 500ms. However, LP devices are more sensitive to t_{ramp} variations; e.g., 65nm LP FHD mean increases by 2.4× when t_{ramp} increases from 10μs to 500ms. Hence, almost 2× more sensitive. Moreover, the same trend is also valid for the standard deviation. Finally, regardless of the design, FHD increases with t_{ramp} until saturation.

Temperature and voltage ramp-up combined: Regardless of the design, FHD is lower at -40°C for t_{ramp} longer than the one used for enrollment, while at 85°C FHD is lower if a short t_{ramp} is used. For example, at -40°C the FHD mean for 65nm GP is the lowest for 50μs, while at 85°C the mean FHD is the lowest for 10μs.

Technology scaling: FHD for GP improves with technology scaling, while it deteriorates for LP. For example, at 85°C and 10μs, FHD reduces by 1.2× for GP, while it increases by 1.6× for LP. The difference is further emphasized with longer t_{ramp}; e.g., at 85°C and 500ms, HD reduces by 1.4× with GP design, while it increases by 2.0× with LP design.

Uniqueness: Table IV shows the uniqueness results. From the table, two main observations can be made:

1) Regardless of the design, FHD between devices shows a good distance between fingerprints at enrollment for all devices; e.g., 65nm GP has an FHD of 50% ± 0.42% between devices.

2) Both designs and technology node present a good distribution of 0's and 1's, which is a good uniqueness indicator; e.g, 65nm GP has an FHW of 50% ± 1%.

D. Comparison: Simulation vs. Silicon

Repeatability: simulation results show that GP devices are less sensitive to varying operation conditions, keeping FHD virtually constant, and that technology scaling reduces its FHD by 0.93×. Silicon measurements show the same trend, however with even more severe values; varying operation conditions impact FHD up to 1.3× while technology scaling reduces FHD by 1.4×. Overall, GP is up to 4.5× better than LP; e.g., at 25°C for 500ms max FHD is 4.5× higher for 40nm LP than for 45nm GP.

Uniqueness: simulation results indicate a good FHW, however the other metrics were not conclusive due to the limited amount of simulated PUF bits and number of devices simulated. The industrial measurements show good uniqueness values for for all investigated devices, therefore, revealing that uniqueness is not impacted by design (GP or LP).

TABLE IV: Uniqueness results - industrial experiments

	65nm GP	45nm GP	65nm LP	40nm LP
FHD (between devices)	50% ± 0.42% (min = 48.73%)	49.85% ± 0.63% (min = 48.54%)	49.90% ± 0.51% (min = 47.30%)	49.23% ± 1.05% (min = 46.74%)
FHW (at enrollment)	50% ± 1%	50% ± 0.5%	50% ± 2.5%	50% ± 0.5%

E. Discussion

The superior performance of GP devices over LP devices for PUF purpose can be explained as follows. GP devices have lower threshold voltages to enhance speed when compared with LP devices. This lower threshold voltage makes them more vulnerable to process variation. While in most applications vulnerability to process variation is a concern, in PUF applications it enhances the asymmetry of the cross-coupled inverters of the SRAM cell; hence putting the SRAM cell in a repeatable state rather than in a random state on power-up. Therefore making the SRAM PUF cell more robust. However, with technology scaling, despite at enrollment condition the previous statement holding true, for the remaining stress conditions LP devices become more vulnerable. Increased LP vulnerability is particularly evident with respect to t_{ramp} variations, as LP devices respond poorly to frequency variations.

The validity of our results holds across the investigated designs, despite the discrete number of devices available. However, the absolute noise values may vary when other technology nodes and manufacture process are considered, the trends are solid. This is due to the very nature of the designs. GP devices, when compared with LP devices will always perform better with varying voltage ramp-up times, due to their intrinsic lower V_{th}.

Regarding the cost of each design in terms of power consumption and area overhead. As PUF-based systems are active only during the start-up of a device to generate the key, delay and power consumption play very minor roles. Therefore, we consider the area overhead to be the main design constrain. With this respect, GP devices have a lower footprint when compared with LP devices (see Table I) [14]. Moreover, the lower the noise in PUF responses, the smaller the PUF size required to design a robust PUF system [17]. As SRAM PUF GP devices are less noisy and more robust when compared with its LP counterparts, we can predict that PUF systems based on GP SRAMs will result in a significant overall smaller area. More specifically, according to [8], when the system is designed to correct a maximum noise of 15%, SRAM corresponds from 585B (or 77% out of 6.1k gates) up to 1kB (or 92% out of 9.3k gates) of the overall area overhead of a PUF-based system, depending on the ECC type. When a higher error correction capability is required, e.g., error correction up to 30%, SRAM footprint increases from 585B to 2kB and from 1kB up to 3.5kB, depending on the ECC type. This simple analysis show the potential of choosing appropriate memory design, e.g., 45nm GP over 40nm LP, as it reduces the footprint of the PUF-based system circa 3×.

From a security point of view, due to higher electrical currents, GP devices might be more vulnerable to being cloned according to [20]. However, the complexity and required tools to perform the attack are not accessible for the average attacker.

V. Conclusion

In this paper we demonstrated the robustness of SRAM PUF for both general purpose and low-power design, by evaluating their repeatability and uniqueness for a wide range of temperatures and voltage ramp-up times, using circuit simulations and industrial measurements. The results show that general purpose design are up to 2× less sensitive to varying operating conditions. Moreover, general purpose designs improve their robustness with technology scaling by 1.4× while low-power deteriorates by 2.0×. Overall, using general purpose SRAM PUFs will result in more robust and cheaper PUF-based systems.

Acknowledgments

The authors would like to thank Intrinsic-ID B.V. for providing the silicon data used in this work.

References

[1] R. Pappu, *Physical One-Way Functions*, Ph.D. Thesis, 2001.
[2] J. Guajardo et al., *FPGA Intrinsic PUFs and Their Use for IP Protection*, CHES, 2007.
[3] B. Skoric, P. Tuyls, and W. Ophey, *Robust Key Extraction from Physical Unclonable Functions*, Applied Cryptography and Network Security, 2005.
[4] Y. Dodis, L. Reyzin, and A. Smith, *Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data*, Advances in Cryptology - Eurocrypt, 2004.
[5] R. Maes and I. Verbauwhede, *Physically Unclonable Functions: A Study on the State of the art and Future Research Directions*, Towards Hardware-Intrinsic Security, Information Security and Cryptography, 2010.
[6] M. Cortez et al., *Modeling SRAM Start-Up Behavior for Physical Unclonable Functions*, DFT, 2012.
[7] D.E. Holcomb et al., *Power-up SRAM State as an Identifying Fingerprint and Source of True Random Number*, IEEE Trans. on Computers **vol.58** (2009), no. 9, 1198–1210.
[8] R. Maes, P. Tuyls, and I. Verbauwhede, *A Soft Decision Helper Data Algorithm for SRAM PUFs*, IEEE Int. Symp. on Information Theory, 2009.
[9] J. Guajardo et al., *Physical Unclonable Functions and Public-key Crypto for FPGA IP Protection*, FPL, 2007.
[10] J. Guajardo et al., *FPGA Intrinsic PUFs and Their Use for IP Protection*, CHES, 2007, pp. 63–80.
[11] M. Hofer and C. Boehm, *An Alternative to Error Correction for SRAM-like PUFs*, CHES, 2010.
[12] M. Claes, V. vd Leest, and A. Braeken, *Comparison of SRAM and FF PUF in 65nm Technology*, NordSec, 2011.
[13] J.-P. Linnartz and P. Tuyls, *New Shielding Functions to Enhance Privacy and Prevent Misuse of Biometric Templates*, AVBPA, 2003.
[14] ITRS, *MASTAR*, 2013.
[15] W. Zhao et al., *Rigorous Extraction of Process Variations for 65nm CMOS Design*, European Solid State Device Research, 2007.
[16] B. Preneel V. vd Leest E. vd Sluis, *Soft Decision Error Correction for Compact Memory-Based PUFs using a Single Enrollment*, CHES, 2012.
[17] M. Cortez et al., *Adapting voltage ramp-up time for temperature noise reduction on memory-based PUFs*, HOST, 2013.
[18] Y. Cao W. Zhao, *New generation of Predictive Technology Model for sub-45nm early design exploration*, IEEE Trans. on Electron Devices, 2006.
[19] Y. Dodis et al., *Fuzzy extractors: How to generate strong keys from biometrics and other noisy data*, SIAM Journal on Computing, 2008.
[20] C. Helfmeier et al., *Cloning Physically Unclonable Functions*, HOST, 2013.
[21] R. Maes et al., *Low-Overhead Implementation of a Soft Decision Helper Data Algorithm for SRAM PUFs*, CHES, 2009.

2

Adapting Voltage Ramp-up Time for Temperature Noise Reduction on Memory-based PUFs

Mafalda Cortez Said Hamdioui
Delft University of Technology
Faculty of EE, Mathematics and CS
Mekelweg 4, 2628 CD Delft, The Netherlands
{A.M.M.O.Cortez, S.Hamdioui}@tudelft.nl

Vincent van der Leest Roel Maes Geert-Jan Schrijen
Intrinsic-ID B.V.
High Tech Campus 9,
Eindhoven, The Netherlands
{Vincent.van.der.Leest, Roel.Maes, Geert.Jan.Schrijen}@intrinsic-id.com

***Abstract*—The efficiency and cost of silicon PUF-based applications, and in particular key generators, are heavily impacted by the level of reproducibility of the bare PUF responses under varying operational circumstances. Error-correcting codes can be used to achieve near-perfect reliability, but come at a high implementation cost especially when the underlying PUF is very noisy. When designing a PUF-based key generator, a more reliable PUF will result in a less complex ECC decoder and a smaller PUF footprint, hence an overall more efficient implementation. This paper proposes a novel insight and resulting technique for reducing noise on memory-based PUF responses, based on adapting supply voltage ramp-up time to ambient temperature. Circuit simulations on 45nm Low-Power CMOS, as well as actual silicon measurements are presented to validate the proposed methods. Our results demonstrate that choosing an appropriate voltage ramp-up for enrollment and adapting it according to the ambient temperature at key-reconstruction is a powerful method which makes memory-based PUF response noise up to three times smaller.**

I. INTRODUCTION

In recent years, silicon *Physically Unclonable Functions* (PUFs) [1] have been well established as innovative hardware security primitives. Numerous constructions have been proposed and implemented (see, e.g., [2] for an overview), and their interesting properties are being extensively investigated in large scale experiments [3–5]. A silicon PUF's ability to generate device-unique fingerprints based on deep-submicron silicon process variations makes it a highly practical tool for device identification. In addition, the intriguing and unparalleled property of *physical unclonability* is a strong foundation for deploying a silicon PUF as a security primitive.

Combined with proper post-processing, a PUF is able to generate secret keys of cryptographic strength [6,7], and reliably store them in a highly secure manner without the need for conventional on-chip *Non-Volatile Memory* (NVM). The key is derived from the device-intrinsic randomness which is evaluated by the silicon PUF. The main purpose of a PUF-based key generator is twofold: *i)* increasing the *reproducibility* of a typically noisy PUF evaluation to near-perfect reliability, and *ii)* accumulating sufficient *unpredictability* of possibly low-entropic PUF responses into a highly unpredictable cryptographic key. It is evident that the natural reproducibility and unpredictability of a bare silicon PUF implementation have a strong impact on the efficiency, and hence on the cost of a PUF-based key generator as a whole. A PUF with less noisy and more random responses will result in a key generator which requires less "PUF material", and hence less silicon area, to produce a reliable cryptographic key.

To produce a key with a practically acceptable reliability level (e.g., failure rate $\leq 10^{-6}$), a PUF-based key generator based on a fuzzy extractor [8,9] uses *Error-Correcting Codes* (ECC) to correct noisy PUF responses.These ECC techniques are very effective in boosting the reliability but tend to be computationally intensive. Moreover, the helper data, which is an unavoidable byproduct of the fuzzy extractor, will partially disclose the unpredictability of the bare PUF responses. This needs to be compensated for by using more PUF material and hence a larger PUF. Both the complexity of the ECC decoder, and the amount of randomness loss due to the helper data, scale with the required error correction capability of the ECC, i.e. less reliable PUF responses will result in a more complex decoder and a larger silicon PUF footprint. Hence, there is a strong incentive to use a PUF construction with an as high as possible reproducibility of its bare responses. This objective is seriously complicated by the reproducibility deterioration of silicon PUFs when subjected to varying operating conditions, like temperature and supply voltage variations.

Substantial research effort has been put into reliability enhancement of PUF-based key generators. Careful selection of the right ECC algorithms minimizes the helper data loss and decoder implementation cost [10,11]. On a physical level, construction improvements have been proposed to decrease the noise level of the bare silicon PUF responses directly, by modifying the PUF circuit [12,13] or the wafer mask set [14]. Analyzing a silicon PUF's susceptibility to its operating conditions has been explored for reliability enhancement [15,16].

In this work, we take this one step further by considering the combined effect of different operating parameters, in particular temperature and supply voltage ramp-up time, and their impact on the reproducibility of SRAM memory-based PUF responses. It is well known that temperature impacts the switching speed of electronic devices and contributes to electronic noise [3], whereas the voltage ramp-up time (i.e., the time it takes to reach the operational supply voltage after power-on) influences the power-up state of an SRAM [17,19]. This paper shows that intelligent matching of voltage ramp-up time to ambient temperature significantly improves the reproducibility of PUF responses at extreme temperatures,

978-1-4799-0601-7/13/$31.00 ©2013 IEEE

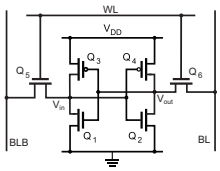


Fig. 1: SRAM Cell transistor level schematic.

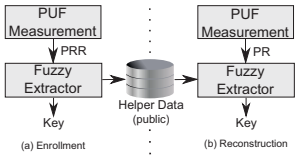


Fig. 2: Operations of a PUF based Key Storage System.

with noise levels up to $3\times$ smaller than without matching. Moreover, this effective technique requires only a small number of additional building blocks and does not impose any modifications to the actual standard memory cell circuit. These effects are demonstrated, both in simulation and actual silicon measurements for SRAM PUFs [6,17], and in silicon only for other memory-based PUF types [20–23].

The remainder of this paper is organized as follows. Section II provides a brief background on memory-based PUFs and PUF-based key storage. Section III discuses the simulation setup, including the noise metric, and the simulation results. Section IV details the silicon measurement setup, including the optimization algorithms used and the achieved improvements. The obtained results are discussed in more detail in Section V, and Section VI provides possible implementation options. Finally, Section VII concludes the paper.

II. Background: PUFs and Key Generation

This section first briefly provide some preliminaries on the basic operation of memory-based PUFs. Then, it shows how PUFs are deployed in a key storage system, and thereafter it gives the PUF's main quality metrics.

A. Memory-based PUFs

Memory-based PUFs [6,20–23] comprise bistable circuits, i.e., having two possible stable states denoted as logic '0' and '1'. Fig. 1 shows a typical six-transistor SRAM cell with at its core a basic bistable circuit consisting of two cross-coupled inverters, respectively formed by (Q_1, Q_3) and (Q_2, Q_4). The peripheral circuitry used to access the cell is comprised by two pass transistors (Q_5 and Q_6), the bitline (BL), the complement bitline (BLB) and the wordline (WL). When powered-up, the cross-coupled inverters start driving electric current, hence increasing the voltages at their gates (V_{in} and V_{out}). The first inverter that builds enough gate voltage to drive its NMOS will pull-down its output, forcing the other inverter to pull-up and causing the SRAM cell to settle in one of both stable states. Since both inverters are designed to be nominally identical, the outcome (in which of both states a cell settles) is entirely determined by the effect of random process variations. Hence, an SRAM power-up state is a PUF response, and this construction is called an SRAM PUF [6].

B. PUF-based Key Generation and Storage

Fig. 2 shows the basic flow of a PUF-based key generation and storage system [6,7] based on a fuzzy extractor [8,9], which typically consists of two phases:

(a) **Enrollment:** a key is generated from a *PUF Reference Response* (PRR) as shown in Fig. 2(a). First, the PUF is evaluated and produces the PRR. Next, the PRR is processed by the fuzzy extractor into a cryptographically strong key, and helper data is generated as a byproduct of the fuzzy extractor's internal ECC method. Finally, the helper data is stored in an external NVM (and hence becomes public information).

(b) **Reconstruction:** the earlier enrolled key is reliably recovered from a noisy *PUF Response* (PR) and the stored helper data as shown in Fig. 2(b). First, the PUF is evaluated again and produces the noisy PR. Next, PR is processed by the fuzzy extractor in combination with the helper data which is retrieved from the external NVM. If the noisy PR is close enough to the PRR obtained during enrollment (i.e. the PUF response is reproducible upto a limited amount of noise), then the extractor succeeds in reliably reconstructing the enrolled key.

C. PUF Properties

The two most basic quality measures of a PUF implementation are *reproducibility*: expressing how reliable a response can be reproduced on a single device, and *uniqueness*: expressing the difference between responses coming from distinct devices.

1) Reproducibility: A fuzzy extractor needs to be designed to cope with the worst-case expected difference between PRR at enrollment and PR at reconstruction in order to obtain a reliable key generation. The noise on a PUF response is typically expressed as the relative number of bit flips between the enrollment PRR and the PR during reconstruction. The smaller the expected noise, and hence the higher the *reproducibility* of the PUF responses, the more efficient the overall PUF-based key generation system can be implemented.

2) Uniqueness: To generate a secure key, a fuzzy extractor requires that a PUF response is unpredictable, even when other responses on the same PUF or access to other PUFs are given. This entails that:

- The probability that two different PUFs have responses close to each other should be negligible, i.e., PUF responses are highly *unique* and the expected amount of differing bits is close to 50%.
- The bits in a specific PUF response should be highly random and independent, i.e., each bit provides a negligible amount of information about the remaining response bits, and the relative entropy of each response is large.

2

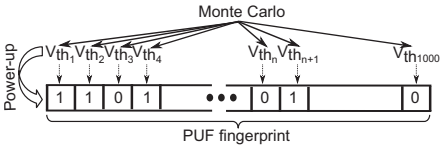


Fig. 3: SRAM PUF simulation.

III. SIMULATIONS

A memory system comprising a cell and peripheral circuitry is synthesized and simulated using SPICE, to analyze the reproducibility of memory-based PUFs by adapting the voltage ramp-up time to the environmental temperature. In this section, first, the PUF fingerprint generation is presented. Second, the metric used to evaluate noise is discussed. Finally, simulation experiments and results are described.

A. SRAM PUF Response Simulation

Each bit of an SRAM PUF response is generated by an individual SRAM cell. Fig. 3 shows the SRAM fingerprint generation schematic used in our simulations. It has been shown in [17,18] that the threshold voltage V_{th} of NMOS transistors is the technology parameter with the most impact on the start-up value of an SRAM cell. Hence, Monte Carlo method is used to generate 1k random values of V_{th} for Q_1 (see Fig. 1) according to the distribution presented in [24], i.e., mean μ = standard NMOS V_{th} and deviation $\sigma = 9\% \cdot \mu$. These 1k SRAM cells combined create an SRAM cell array that generates a unique and random 1k-bit response after power-up.

B. Noise Metric

To analyze the noise we read the PR of the simulated SRAM cell array for different voltage ramp-up times (t_{ramp}) and different temperatures (*Temp*). Then, the *Fractional Hamming Distance* (FHD) of each measured response compared to the enrollment response (PRR) is calculated; this is the number of differing bits normalized to the response length.

C. Simulation Experiments

To investigate the impact of the voltage ramp-up time t_{ramp} on the noise at different temperatures *Temp*, we consider a range of values for both t_{ramp} and *Temp* for 45nm *Low Power* (LP) [25]. For each combination of *Temp* and t_{ramp} we simulated the power-up of the SRAM cell array 20 times and read its response. The transient noise during power-up is randomly generated by the simulation tool, hence three variable parameters are used for the simulation:

- **Voltage ramp-up time:** $3 \times t_{ramp}$ ($10\mu s$, $50\mu s$ and $90\mu s$),
- **Temperature:** $3 \times$ *Temp* (-40°C, $+25$°C, $+85$°C) and,
- **Measurements:** $20 \times$ *Meas* (each with a random seed).

Hence, a total of $(3 \times t_{ramp}) \times (3 \times Temp) \times (20 \times Meas) \times (1000 \times V_{th}) = 180{,}000$ simulations are performed.

TABLE I: Description of devices used in validation.

Technology	# ICs	# PUF inst. / IC			Total # PUF inst.		
		BK	DFF	SRAM	BK	DFF	SRAM
40nm LP	5	-	-	3	-	-	15
65nm LP	50	2	4	4	100	200	200
130nm LP	16	-	1	1	-	16	16

D. Simulation Results

Fig. 4 shows the results of *maximum FHD* ($\max$FHD) calculations per t_{ramp} and $Temp$ considering enrollment performed at $+25$°C with t_{ramp} of (a) 10μs, (b) 50μs and (c) 90μs.

From Fig. 4(a) it can be seen that for *Temp* below the enrollment ($+25$°C), $\max$FHD is lower if t_{ramp} is longer than the one used for enrollment. However, at *Temp* above the enrollment, the opposite is true, e.g., at $+85$°C, key-reconstruction with $10\mu s$ generates the lowest $\max$FHD while at -40°C, that is true for $90\mu s$.

Fig. 4(b) and (c) report similar results but now for other t_{ramp} at enrollment ($50\mu s$ and $90\mu s$). Following the trend observed previously, for *Temp* below enrollment ($+25$°C), $\max$FHD is lower if t_{ramp} is longer than the one used during enrollment; e.g., considering Fig. 4(b), at $+85$°C, key-reconstruction with $10\mu s$ generates the lowest $\max$FHD while at -40°C, that is true for $90\mu s$.

IV. SILICON VALIDATION

The theoretical results from the simulations are validated in an experiment using silicon devices. For this purpose, measurements are performed on three different types of memory-based PUFs: the SRAM PUF [6,17], the D flip-flop (DFF) PUF [21] and the buskeeper (BK) PUF [22].

A. Test Set-up

The considered memory-based PUF types are manufactured in three different LP technology nodes. Table I provides an overview of all devices citing the technology node, the number of available integrated circuits (ICs), the number of PUF instances per IC in the given technology (if any), and the total number of tested instances of each PUF type. Note that each IC contains one or more PUF instances.

Measurements are performed at three different temperatures (-40°C, $+25$°C and $+85$°C)[1] and for ten different t_{ramp} varying from $10\mu s$ to $500ms$. In case of the 40nm SRAM, the shortest possible t_{ramp} is $50\mu s$ due to specific capacitive load. The measurements flow is as follows:

1) The ICs are placed in a climate chamber and connected to a programmable power supply.
2) Climate chamber is set to one of the test temperatures.
3) ICs are powered with a t_{ramp} from the test set.
4) Each PUF device response is read and stored in a file.
5) The ICs are powered down for 1 second.

[1]Industrial standard for temperature testing of ICs ranges from -40°C to $+85$°C, which are therefore part of the test as worst case temperatures in comparison to the enrollment temperature of $+25$°C.

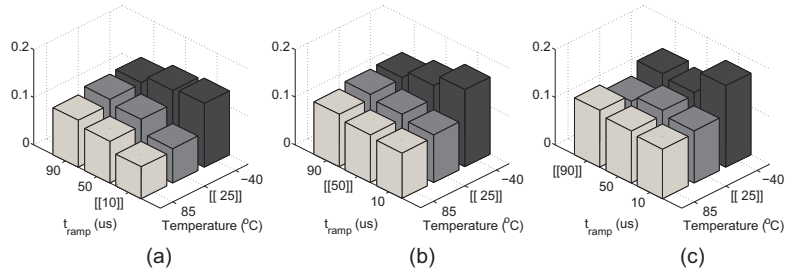


Fig. 4: $\max$FHD; enrollment performed at $+25°$C with t_{ramp} of (a) 10μs, (b) 50μs and (c) 90μs.

6) Steps 3 to 5 are repeated 9 times (i.e. 10 measurements per PUF per temperature per t_{ramp}).
7) Change t_{ramp} and repeat steps 3 to 6 (until all values of t_{ramp} have been tested for this temperature).
8) Change temperature and repeat steps 3 to 7.

B. Evaluation Metrics

1) Reproducibility: For calculating FHD, first an enrollment response of each PUF instance is measured. Thereafter, each reconstruction measurement is compared to this enrollment by counting the number of flipped bits and dividing it by the response length. A key based on the PUF response (as described in Section II) is reliable if the worst-case FHD under any stress condition is below the error correction capability of the ECC. Hence, the smaller FHD (noise), the lower the required error correction.

2) Uniqueness: We evaluate the uniqueness of the different PUF implementations by considering *(a)* the *average between-class Hamming distance* (μ-BCHD), and *(b)* the estimated *min-entropy* (H_∞) of the measured responses. Note that the uniqueness is analyzed only at enrollment. In the key storage use case (as described in Section II) only the uniqueness of the enrollment PUF response is critical, as it is from this response that the cryptographic key is derived.

μ-BCHD provides an indication of uniqueness. This value is calculated as follows:

1) The enrollment response of each PUF is measured.
2) The Hamming distance between each pair of enrollment responses coming from different PUF instances of the same type is determined (e.g. between all pairs of enrollment responses of 65nm LP SRAM PUFs are computed).
3) The distribution of these *between-class* distances is determined and the obtained mean value, normalized to the response length, is μ-BCHD.

Optimally, the obtained distribution should be approximately Gaussian and μ-BCHD should be very close to 50% [17].

Min-entropy is used to evaluate the intrinsic unpredictability of the PUF responses. Min-entropy is a pessimistic measure of the unpredictability of a random variable [8]. We estimate the min-entropy of the responses of a particular PUF type by considering the following model: each PUF response bit is independent of the other bits in the same response and has an individual probability p_1 of being '1' for a random PUF instance. This model is particularly reasonable for memory-based PUFs, as each response bit originates from an individual memory cell. Under the assumption of this model, the min-entropy of a single response bit is calculated as $H_\infty = -\log_2 \max\{p_1, 1-p_1\}$. The value for p_1 of a bit is estimated by counting the number of enrollment responses for which this bit is '1' and dividing by the total number of enrollment responses. The min-entropy of the entire response is simply the summation of the min-entropy of each bit. We express H_∞ as the average min-entropy per bit in a response value, by dividing the total min-entropy of the response by its length. Optimally, H_∞ of a PUF response bit should be close to 1. Note that, due the limited number of measured PUF instances, the obtained estimations of H_∞ could be smaller than the actual min-entropy of these PUF responses.

C. Optimization Algorithms

The silicon test analyses have the objective to investigate the use of t_{ramp} as a technique for increasing memory-based PUF response reproducibility (noise reduction). As a side effect, the impact on PUF uniqueness is also investigated. For this purpose, two optimization algorithms are used:

1) Reproducibility optimization: This algorithm identifies for each value of t_{ramp} at enrollment the t_{ramp} configuration per temperature that leads to the highest reproducibility (lowest maximum noise) at extreme temperatures.

2) Uniqueness optimization: This algorithm identifies the enrollment t_{ramp} that provides the highest H_∞. After this first step the values of t_{ramp} at other temperatures are determined, which minimize the noise.

D. Measurement Results

In order to evaluate the performance of the optimization algorithms, the original PUF measurements (without optimization) need to be analysed first. Table II shows the original measured maximum noise values for the considered temperatures as well as uniqueness indicators. These values are obtained using the shortest possible t_{ramp} for each PUF. As stated before, the noise is determined using 10 response measurements per PUF per temperature.

TABLE II: Measurement results without optimization.

Technology	PUF	t_{ramp}	Maximum noise FHD			μ-BCHD	H_∞
			$-40°$C	$+25°$C	$+85°$C		
40nm LP	SRAM	$50\mu s$	23%	6%	20%	0.50	0.73
65nm LP	SRAM	$10\mu s$	8%	6%	8%	0.50	0.87
	DFF	$10\mu s$	28%	8%	25%	0.37	0.40
	BK	$10\mu s$	10.5%	4.5%	20%	0.48	0.75
130nm LP	SRAM	$10\mu s$	13%	6%	12%	0.47	0.66
	DFF	$10\mu s$	16.5%	5%	28%	0.43	0.61

Table II reveals that overall the maximum noise measured at $-40°$C is 28% (for the 65nm DFF PUF), at $+25°$C is 8% (for the 65nm DFF PUF), and at $+85°$C is 28% (for the 130nm DFF PUF). Regarding uniqueness, although a truly fair comparison is not possible due to the different number of devices available per technology node and PUF type, the 65nm DFF PUF has the lowest μ-BCHD $= 0.37$ and $H_\infty = 0.40$.

1) Reproducibility optimization: Table III presents the results of the reproducibility optimization algorithm; it shows the t_{ramp} configuration that minimizes the noise (maximizes the reproducibility) per temperature in comparison to the enrollment. The results show that for all tested PUFs, adapting t_{ramp} to the ambient temperature has a major impact on the maximum noise. For low temperatures, noise reduction is realized with longer t_{ramp}; whereas for high temperatures, this is realized with shorter t_{ramp}; e.g., the maximum noise for the 65nm LP DFF PUF at $-40°$C with $t_{ramp} = 10\mu s$ for both enrollment and reconstruction was originally 28%. If the optimized t_{ramp} is used both at the enrollment ($500\mu s$ at $+25°$C) and at reconstruction ($50ms$ at $-40°$C), then the maximum noise is reduced to merely 11.5%. Note that all results in Table III demonstrate the *same trend* as predicted by the simulation results of Section III-D. Since this algorithm does not optimize the uniqueness, μ-BCHD and H_∞ decrease for some PUFs (e.g. the 130nm SRAM PUF), while they increase for others (e.g. the 65nm DFF PUF).

2) Uniqueness optimization: Table IV reports the results of the uniqueness optimization algorithm; it shows *(a)* the t_{ramp} at enrollment that maximizes uniqueness and *(b)* the t_{ramp} for the other temperatures that results in the lowest maximum noise (with respect to the t_{ramp} selected for enrollment). Uniqueness indicators μ-BCHD and H_∞ are at least as high as the originals for 40nm and 130nm SRAMs, and for the remaining devices these indicators are higher than the original indicators. The uniqueness optimization algorithm clearly leads to significant improvements in μ-BCHD and H_∞ for the tested DFF and buskeeper PUFs. Improvements for the SRAM PUFs from all tested nodes are negligible. Since this algorithm does not select the enrollment t_{ramp} optimized for reproducibility, it is natural that the noise resulting from this algorithm is worse than that of reproducibility optimization algorithm. In case of the 65nm SRAM PUF, the maximum noise at $-40°$C is even worse than the measurements without optimization. Reason for this is that the t_{ramp} at enrollment ($+25°$C) is very long and the algorithm is unable to find a corresponding longer t_{ramp} at $-40°$C.

TABLE III: Results after reproducibility optimization.

Technology	PUF	t_{ramp}			Maximum noise FHD			μ-BCHD	H_∞
		$-40°$C	$+25°$C	$+85°$C	$-40°$C	$+25°$C	$+85°$C		
40nm LP	SRAM	10ms	1ms	$50\mu s$	14%	4.5%	17%	0.49	0.71
65nm LP	SRAM	50ms	$250\mu s$	$10\mu s$	7%	5.5%	7%	0.50	0.89
	DFF	50ms	$500\mu s$	$25\mu s$	11.5%	5%	9%	0.49	0.84
	BK	500ms	1ms	$25\mu s$	6.5%	4%	6.5%	0.46	0.69
130nm LP	SRAM	500ms	10ms	1ms	5.5%	2%	5%	0.37	0.42
	DFF	500ms	10ms	$500\mu s$	12.5%	2.5%	8.5%	0.45	0.67

TABLE IV: Results after uniqueness optimization.

Technology	PUF	t_{ramp}			Maximum noise FHD			μ-BCHD	H_∞
		$-40°$C	$+25°$C	$+85°$C	$-40°$C	$+25°$C	$+85°$C		
40nm LP	SRAM	1ms	$100\mu s$	$50\mu s$	16%	6%	19%	0.50	0.73
65nm LP	SRAM	50ms	100ms	$50\mu s$	13%	2%	8%	0.50	0.89
	DFF	500ms	10ms	$250\mu s$	18.5%	2.5%	8%	0.50	0.90
	BK	100ms	$250\mu s$	$10\mu s$	7%	5%	9%	0.50	0.88
130nm LP	SRAM	1ms	$10\mu s$	$10\mu s$	7.5%	6%	12%	0.47	0.66
	DFF	50ms	$500\mu s$	$10\mu s$	10%	4.5%	9.5%	0.47	0.67

V. DISCUSSION

SPICE simulations show that using long t_{ramp} at low temperatures and short t_{ramp} at high temperatures results in reduced SRAM PUF response noise when compared to enrollment. The observation is validated using silicon measurement, and regardless of the technology node and memory PUF type. Hence, choosing appropriate t_{ramp} according to ambient temperature, including enrollment, can be used as an efficient scheme to reduce noise and increase reproducibility.

Moreover, the silicon measurements have also indicated that varying the voltage ramp-up time can have a significant impact on the uniqueness of memory-based PUFs. By choosing the appropriate optimization algorithm according to the PUF type, noise can be reduced while either maintaining or increasing the uniqueness indicators. Inspecting the silicon results with regard to reproducibility and uniqueness we conclude the following:

- SRAM PUFs benefit from applying the reproducibility optimization algorithm, but the uniqueness optimization algorithm is not very effective as there is very little margin for improvement. Furthermore, the uniqueness optimization algorithm does not minimize the noise well for the tested SRAMs.
- Buskeeper and DFF PUFs benefit from applying the uniqueness optimization algorithm, since the original silicon results show that there is a lot of room for improvement. Besides increasing the PUF response uniqueness, the proposed algorithm also decreases the noise at extreme temperatures. Hence, this algorithm works very well for these PUF types.

VI. IMPLEMENTATION CONSIDERATIONS

The proposed scheme can be implemented by a simple circuit consisting of a voltage regulator and a temperature sensor. Fig. 5 shows an example of such a circuit, comprising five blocks, an SRAM PUF, a voltage ramp-up regulator, an embedded temperature sensor, an ADC and a controller.

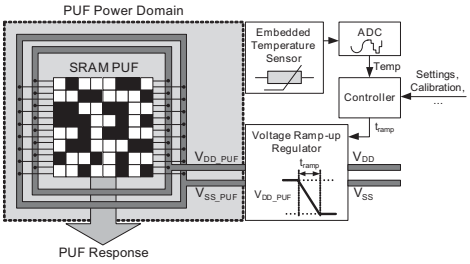


Fig. 5: Schematic of an extended SRAM PUF design.

The circuit performs five main steps. First, the temperature sensor senses the ambient temperature. Second, this temperature is used as an input to the ADC that converts the given temperature to the closest digital temperature *Temp*. Third, according to *Temp* the Controller is calibrated and the t_{ramp} that minimizes the FHD (noise) is produced. Fourth, the voltage ramp-up regulator powers-up the SRAM PUF with the assigned t_{ramp}, generating finally a PUF response.

One of the main advantages of the proposed optimization technique, besides its evident effectiveness, is that its implementation demands no adaptations of the memory-based PUF circuit itself. In fact, the basic PUF comprises only standard library memory cells, but needs to be placed in its own power domain and extended with an embedded temperature sensor and a voltage ramp-up regulator. A small controller regulates the optimal ramp-up time of the memory-based PUF to the sensed temperature, based on a prior calibration. The general design of these extensions is schematically shown for an SRAM PUF in Fig. 5. Since the concerned building blocks are all rather standard, the implementation effort of the proposed optimization technique is considered minimal, in particular in relation to the large obtained gain in PUF reproducibility as demonstrated in Section IV.

VII. Conclusion

In this paper, we proposed a method based on adapting the voltage ramp-up time to the ambient temperature for enhancing the reproducibility of memory-based PUFs. The combined effect on PUF reproducibility has been evaluated using both circuit simulation (in 45nm LP CMOS) and actual silicon measurements (in 45nm, 65nm and 130nm LP CMOS). The results are highly effective, showing a major decrease in worst-case PUF noise (up to $3\times$ lower for particular PUFs) at extreme temperatures. A significant advantage of the proposed noise-reduction technique is that it can be implemented without altering existing memory-based PUF circuits, but merely by extending them with standard building blocks. The application of the proposed techniques will result in a significantly reduced complexity and a smaller footprint of a PUF-based key generator. The reproducibility enhancement is achieved while either maintaining or increasing the uniqueness.

Acknowledgment

The work performed by Intrinsic-ID for this publication has been supported by the European Commission through the FP7 programme under contract 284833 PUFFIN. The authors would also like to thank all partners from the FP7 project UNIQUE. In this project the 65nm ASICs and test boards used in this publication were designed and manufactured.

References

[1] B. Gassend, D. Clarke, M. van Dijk and S. Devadas, "Silicon physical random functions", *ACM CCS*, pp. 148–160, 2002.
[2] R. Maes and I. Verbauwhede, "Physically Unclonable Functions: A Study on the State of the Art and Future Research Directions", *Towards Hardware-Intrinsic Security*, A.-R. Sadeghi and D. Naccache (Eds.), pp. 3–37, 2010.
[3] S. Katzenbeisser, Ü. Kocabas, V. Rozic, A.-R. Sadeghi, I. Verbauwhede and C. Wachsmann, "PUFs: Myth, Fact or Busted? A Security Evaluation of Physically Unclonable Functions (PUFs) Cast in Silicon", *CHES*, pp. 283–301, 2012.
[4] A. Maiti, J. Casarona, L. McHale and P. Schaumont, "A large scale characterization of RO-PUF", *HOST*, pp. 94–99, 2010.
[5] T. Yoshida, T. Katashita and A. Satoh, "Quantitative and Statistical Performance Evaluation of Arbiter Physical Unclonable Functions on FPGAs", *ReConFig*, pp. 298–303, 2010.
[6] J. Guajardo, S.S. Kumar, G.-J. Schrijen and P. Tuyls, "FPGA Intrinsic PUFs and Their Use for IP Protection", *CHES*, pp. 63–80, 2007.
[7] B. Skoric, P. Tuyls and W. Ophey, "Robust key extraction from Physical Unclonable Functions", *ACNS*, pg. 99–135, 2005.
[8] Y. Dodis, R. OstrovskyL. Reyzin and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data", *SIAM Journal on Computing*, pp. 97–139, 2008.
[9] J.-P. Linnartz and P. Tuyls, "New shielding functions to enhance privacy and prevent misuse of biometric templates", *AVBPA*, pp. 393–402, 2003.
[10] R. Maes, A. Van Herrewege and I. Verbauwhede, "PUFKY: A Fully Functional PUF-Based Cryptographic Key Generator", *CHES*, pp. 302–319, 2012.
[11] V. van der Leest, B. Preneel and E. van der Sluis, "Soft Decision Error Correction for Compact Memory-Based PUFs Using a Single Enrollment", *CHES*, pp. 268-282, 2012.
[12] M. Hofer and C. Boehm, "An Alternative to Error Correction for SRAM-Like PUFs", *CHES*, pp. 335–350, 2010.
[13] V. Vivekraja and L. Nazhandali, "Circuit-level techniques for reliable Physically Uncloneable Functions", *HOST*, pp. 30–35, 2009.
[14] D. Forte and A. Srivastava, "On improving the uniqueness of silicon-based physically unclonable functions via Optical Proximity Correction", *DAC*, pp. 96–105, 2012.
[15] M. Bhargava, C. Cakir and K. Mai, "Attack Resistant Sense Amplifier based PUFs (SA-PUF) with Deterministic and Controllable Reliability of PUF Responses", *HOST*, pp. 106–111,2010.
[16] R. Kumar, H.K. Chandrikakutty and S. Kundu, "On improving reliability of delay based Physically Unclonable Functions under temperature variations", *HOST*, pp. 142–147, 2011.
[17] D.E. Holcomb, W.P. Burleson and K. Fu, "Power-Up SRAM State as an Identifying Fingerprint and Source of True Random Number", *IEEE Transactions on Computers*, vol. 58, no. 9, 2009.
[18] M. Cortez, A. Dargar, S. Hamdioui and G.-J. Schrijen, "Modeling SRAM start-up behavior for Physical Unclonable Functions", *IEEE Int. Symp. on Defect and Fault Tolerance in VLSI and Nanotechnology Systems*, pp. 1-6, 2012
[19] M. Claes, V. van der Leest and A. Braeken, "Comparison of SRAM and FF PUF in 65nm technology", *NordSec*, pp. 47–64, 2011.
[20] S.S. Kumar, J. Guajardo, R. Maes, G.-J. Schrijen and P. Tuyls, "The Butterfly PUF protecting IP on every FPGA", *HOST*, pp. 67–70, 2008.
[21] R. Maes, P. Tuyls and I. Verbauwhede, "Intrinsic PUFs from Flip-Flops on Reconfigurable Devices", *WISSec*, 2008.
[22] P. Simons, V. van der Leest and E. van der Sluis, "Buskeeper PUFs, a promising alternative to D Flip-Flop PUFs", *HOST*, pp. 7–12, 2012.
[23] Y. Su, J. Holleman and B. Otis, "A 1.6pJ/bit 96% Stable Chip-ID Generating Circuit using Process Variations", *ISSCC*, pp. 406–611, 2007.
[24] W. Zhao, F. Liu, K. Agarwal, D. Acharyya, S.R. Nassif, K.J. Nowka and Y. Cao, "Rigorous extraction of process variations for 65nm CMOS design", *ESSDERC*, pp. 89–92, 2007.
[25] "http://ptm.asu.edu/", 2012.

1162 IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, VOL. 34, NO. 7, JULY 2015

2

Intelligent Voltage Ramp-Up Time Adaptation for Temperature Noise Reduction on Memory-Based PUF Systems

Mafalda Cortez, *Student Member, IEEE*, Said Hamdioui, *Senior Member, IEEE*, Ali Kaichouhi, Vincent van der Leest, Roel Maes, and Geert-Jan Schrijen

***Abstract*—The efficiency and cost of silicon physically unclonable function (PUF)-based applications, and in particular key generators, are heavily impacted by the level of reproducibility of the bare PUF responses (PRs) under varying operational circumstances. Error-correcting codes (ECCs) can be used to achieve near-perfect reliability, but come at a high implementation cost especially when the underlying PUF is very noisy. When designing a PUF-based key generator, a more reliable PUF will result in a less complex ECC decoder and a smaller PUF footprint, and hence, an overall more efficient implementation. This paper proposes novel insight and resulting method for reducing noise on memory-based PRs, based on adapting supply voltage ramp-up time to ambient temperature. Circuit simulations on 45 nm low-power CMOS, as well as silicon measurements are presented to validate the proposed method. Our results demonstrate that choosing an appropriate voltage ramp-up for enrollment and adapting it according to the ambient temperature at key-reconstruction is a powerful method which makes memory-based PR noise up to 3× smaller. In addition, this paper investigates the competitiveness of integrating the proposed method in a commercial product; the investigation is done in two phases. First by determining the saved area, and second by implementing a circuit that maps the ambient temperature into an appropriate voltage ramp-up. The results show that the new system costs up to 82.1% less area while it delivers up to 3× higher reproducibility.**

***Index Terms*—Adapter circuit, memory-based physically unclonable function (PUF), noise reduction, voltage ramp-up time.**

I. INTRODUCTION

IN RECENT years, silicon physically unclonable functions (PUFs) [1] have been well established as innovative hardware security primitives. Numerous constructions have been proposed and implemented (see [2] for an overview), and their interesting properties are being extensively investigated in large scale experiments [3]–[5]. A silicon PUFs ability to generate device-unique fingerprints based on deep-submicron silicon process variations makes it a highly practical tool for device identification. In addition, the intriguing and unparalleled property of physical unclonability is a strong foundation for deploying a silicon PUF as a security primitive.

Combined with proper post-processing, a PUF is able to generate secret keys of cryptographic strength [6], [7], and reliably store them in a highly secure manner without the need for conventional on-chip nonvolatile memory (NVM). The key is derived from the device-intrinsic randomness which is evaluated by the silicon PUF. The main purpose of a PUF-based key generator is twofold: 1) increasing the reproducibility of a typically noisy PUF evaluation to near-perfect reliability and 2) accumulating sufficient unpredictability of possibly low-entropic PUF responses (PRs) into a highly unpredictable cryptographic key. It is evident that the natural reproducibility and unpredictability of a bare silicon PUF implementation have a strong impact on the efficiency, and hence on the cost of a PUF-based key generator as a whole. A PUF with less noisy and more random responses will result in a key generator which requires less "PUF material," and hence less silicon area, to produce a reliable cryptographic key.

To produce a key with a practically acceptable reliability level (e.g., failure rate $\leq 10^{-6}$), a PUF-based key generator based on a fuzzy extractor (FE) [8], [9] uses error-correcting codes (ECC) to correct noisy PRs. These ECC techniques are very effective in boosting the reliability but tend to be computationally intensive. Moreover, the helper data, which is an unavoidable FE byproduct, will partially disclose the unpredictability of the bare PRs. This needs to be compensated for by using more PUF material and hence, a larger PUF. Both complexity of the ECC decoder and the amount of randomness loss due to the helper data scale with the required error correction capability (ECCap) of the ECC; i.e., less reliable PRs will result in a more complex decoder and a larger silicon PUF footprint. Hence, there is a strong incentive to use a PUF construction with an as high as possible reproducibility of its bare responses. This objective is seriously complicated by the reproducibility deterioration of silicon PUFs when subjected to varying operating conditions, such as temperature and supply voltage variations.

Substantial research effort has been put into reliability enhancement of PUF-based key generators. Careful selection

Manuscript received July 5, 2014; revised November 5, 2014; accepted February 16, 2015. Date of publication April 14, 2015; date of current version June 16, 2015. This work was supported in part by the European Commission through the FP7 Programme under Contract 284833 PUFFIN, and in part by the Dutch "Point One Program" under the RATE Project PNU09C09. This paper was recommended by Associate Editor R. Karri.

M. Cortez and S. Hamdioui are with the Computer Engineering Group, Delft University of Technology, Delft 2628CD, The Netherlands (e-mail: a.m.m.o.cortez@tudelft.nl).

A. Kaichouhi is with the Circuits and Systems Group, Delft University of Technology, Delft 2628CD, The Netherlands.

V. van der Leest, R. Maes, and G.-J. Schrijen are with Intrinsic-ID B.V., Eindhoven 5656AE, The Netherlands.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCAD.2015.2422844

of the right ECC algorithms to minimize the helper data loss and decoder implementation cost have been reported [10], [11]. On a physical level, construction improvements to directly decrease the bare silicon PRs noise level have been proposed, either by modifying the PUF circuit [12], [13], or the wafer mask set [14]. Analyzing a silicon PUFs susceptibility to its operating conditions has been explored for reliability enhancement [15], [16].

In this paper, an extension of this paper presented in [26], we take this one step further by considering the combined effect of different operating parameters, in particular temperature and supply voltage ramp-up time, and their impact on the reproducibility of memory-based PRs. It is well known that temperature impacts the switching speed of electronic devices and contributes to electronic noise [3], whereas the voltage ramp-up time (i.e., the time it takes to reach the operational supply voltage after power-on) influences the power-up state of an static random-access memory (SRAM) [17]–[19]. This paper shows that intelligent matching of voltage ramp-up time to ambient temperature significantly improves the reproducibility of PRs at extreme temperatures, with noise levels up to 3× smaller than without matching. Moreover, this effective technique requires only a small number of additional building blocks and does not impose any modifications to the actual standard memory cell circuit. These effects are demonstrated, both using circuit simulation and actual silicon measurements for SRAM PUFs, and only silicon measurements for other memory-based PUF types such as [20]–[23].

In addition, we investigate the competitiveness of integrating the proposed technique in a commercial product. The competitiveness is evaluated first, by investigating the relation between memory-based PUF noise and area overhead, determining the saved area for various technology nodes for various PUF-technologies. Second, by proposing and implementing a circuit that maps the ambient temperature into an adequate voltage ramp-up that minimizes the noise. Comparing the saved area against the area of the circuit that enables the noise reduction, we demonstrate that adapting the voltage ramp-up time to the ambient temperature is a very powerful and industrially attractive technique for memory-based PUFs.

The remainder of this paper is organized as follows. Section II provides a brief background on memory-based PUFs and PUF-based key storage. Section III discusses the simulation setup, including the noise metric and the simulation results. Section IV details the silicon measurement setup, including the optimization algorithms used, the achieved improvements and their discussion. Section V reviews the various FE constructions, makes the link between area overhead and noise, describes the setup to analyze the impact of noise reduction on the area overhead and presents the results. Section VI provides the requirements, the implementation details, and the results of the circuit that maps the temperature to the voltage ramp-up time. Section VII evaluates the proposed system competitiveness by combining and discussing the previous sections results. Finally, Section VIII concludes this paper.

II. Background: PUFs and Key Generation

This section first briefly provides some preliminaries on the basic operation of memory-based PUFs. Then, it shows how PUFs are deployed in a key storage system, and thereafter it gives the PUFs main quality metrics.

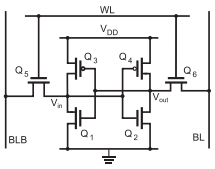


Fig. 1. SRAM cell transistor level schematic.

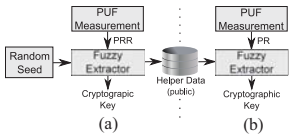


Fig. 2. Operations of a PUF-based key storage system. (a) Enrollment. (b) Reconstruction.

A. Memory-Based PUFs

Memory-based PUFs [6], [20]–[23] comprise bistable circuits, i.e., having two possible stable states denoted as logic "0" and "1." Fig. 1 shows a typical six-transistor SRAM cell with at its core a basic bistable circuit consisting of two cross-coupled inverters, respectively, formed by (Q_1, Q_3) and (Q_2, Q_4). The peripheral circuitry used to access the cell is comprised by two pass transistors (Q_5 and Q_6), the bitline, complement bitline, and wordline. When powered-up, the cross-coupled inverters start driving electric current, hence, increasing the voltages at their gates (V_{in} and V_{out}). The first inverter that builds enough gate voltage to drive its nMOS will pull-down its output, forcing the other inverter to pull-up and causing the SRAM cell to settle in one of both stable states. Since both inverters are designed to be nominally identical, the outcome (in which of both states a cell settles) is entirely determined by the effect of random process variations. Hence, an SRAM power-up state is a PR, and this construction is called an SRAM PUF [6].

B. PUF-Based Key Generation and Storage

Fig. 2 shows the basic flow of a PUF-based key generation and storage system [6], [7] based on an FE [8], [9], which typically consists of two phases.

1) *Enrollment:* A cryptographic key is generated from a PUF. First, a PUF measurement is taken and used as PUF reference response (PRR). Next, PRR and an external Random Seed are processed by the FE into a cryptographically strong cryptographic key, and helper data is generated as an FE byproduct. Finally, the helper data is stored in an external NVM; hence, it becomes public information.
2) *Reconstruction:* The earlier enrolled cryptographic key is reliably recovered. First, a PUF measurement is taken and used as PR. Typically, some bits of PR are different from the original PRR; hence, PR is a noisy version of PRR. Next, FE processes PR in combination with the

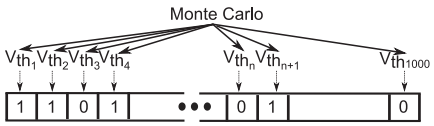


Fig. 3. SRAM PUF simulation.

helper data (retrieved from the external NVM). If PR is close enough to PRR (i.e., PRR is reproducible up to a limited noise amount), then the FE succeeds in reliably reconstructing the enrolled cryptographic key.

C. PUF Properties

The two most basic PUF implementation quality measures are reproducibility (expressing how reliably a response can be reproduced on a single device), and uniqueness (expressing the difference between responses coming from distinct devices).

1) Reproducibility: A FE needs to be designed to cope with the worst-case expected difference between enrollment PRR and reconstruction PR, to reliably generate a key. PR noise is typically expressed as the relative number of bit-flips between the enrollment PRR and the reconstruction PR. The smaller the expected noise, and hence, the higher the reproducibility of the PRs, the more efficient the overall PUF-based key generation system can be implemented.

2) Uniqueness: To generate a secure key, an FE requires PR unpredictability, even if other responses on the same PUF or access to other PUFs are given. This entails the following.

a) The probability that two different PUFs have responses close to each other should be negligible, i.e., PRs are highly unique and the expected amount of differing bits is close to 50%.
b) The bits in a specific PR should be highly random and independent, i.e., each bit provides a negligible amount of information about the remaining response bits, and the relative entropy of each response is large.

III. SIMULATIONS

To analyze the reproducibility of memory-based PUFs when adapting the voltage ramp-up time to the environmental temperature, a memory system comprising a cell and peripheral circuitry is synthesized and simulated using SPICE. In this section, first, the PUF fingerprint generation is presented. Second, the metric used to evaluate noise is discussed. Third, simulation experiments are described. Finally, results are presented and discussed.

A. SRAM PUF Response

Each bit of an SRAM PR is generated by an individual SRAM cell. Fig. 3 shows the SRAM fingerprint generation schematic used in our simulations. Holcomb *et al.* [17] and Cortez *et al.* [18] showed that the threshold voltage V_{th} of nMOS transistors is the technology parameter with the most impact on the start-up value of an SRAM cell. Hence, the Monte Carlo method is used to generate 1K random values of V_{th} for Q_1 (see Fig. 1) according to the distribution presented in [24], i.e., mean μ = standard nMOS V_{th} and deviation $\sigma = 9\% \cdot \mu$. These 1K SRAM cells combined create an SRAM cell array that generates a unique and random 1K-bit response after power-up.

B. Noise Metric

To analyze the noise we read the PR of the simulated SRAM cell array for different voltage ramp-up times (t_{ramp}) and different temperatures (Temp). Then, the fractional Hamming distance (FHD) [17] of each measured response compared to the enrollment response (PRR) is calculated; this is the number of differing bits normalized to the response length.

C. Simulation Experiments

To investigate the impact of the voltage ramp-up time t_{ramp} on the noise at different temperatures Temp, we consider a range of values for both t_{ramp} and Temp for 45 nm low power (LP) [25]. For each combination of Temp and t_{ramp} we simulate the power-up of the SRAM cell array 20 times and read its response. The transient noise during power-up is randomly generated by the simulation tool, hence, three variable parameters are used for the simulation.

1) *Voltage Ramp-Up Time:* $4 \times t_{\text{ramp}}$ (10, 50, 90, and 130 μs).
2) *Temperature:* $3 \times$ Temp (-40, 25, and 85 °C).
3) *Measurements:* $20 \times$ Meas (each with a random seed).

Hence, a total of $(4 \times t_{\text{ramp}}) \times (3 \times \text{Temp}) \times (20 \times \text{Meas}) \times (1000 \times \text{V}_{\text{th}}) = 240\,000$ simulations are performed.

D. Simulation Results and Analysis

Fig. 4 shows the results of maximum FHD (max FHD) calculations per t_{ramp} and Temp. PUF-based systems are designed to correct up to the worse reconstruction conditions. For this reason, we present the worse (highest) FHD out of the 20 measurements for each of the evaluated conditions. Note that, enrollment is performed at 25 °C with t_{ramp} of Fig. 4(a)–(d) is 10, 50, 90, and 130 μs, respectively; the enrollment conditions are given between "[]" in the figure. From Fig. 4(a), it can be seen that for Temp below the enrollment, max FHD is lower if t_{ramp} is longer than the one used for enrollment. However, at Temp above the enrollment, the opposite is true, e.g., at 85 °C, key-reconstruction with 10 μs generates the lowest max FHD while at -40 °C, that is true for 90 μs. Fig. 4(b)–(d) report similar results but now for enrollment at 50, 90, and 130 μs. Following the trend observed previously, for Temp below enrollment, max FHD is lower if t_{ramp} is longer than the one used during enrollment; e.g., Fig. 4(b) shows that the lowest max FHD at 85 °C is achieved with 10 μs while at -40 °C this is realized with 90 μs.

The simulation results revealed a negative correlation between the temperature and the voltage ramp-up time with respect to noise during key reconstruction on memory-PUF fingerprints. The main components of memory-PUFs are MOSFETs; these are vulnerable to three main types of noise: 1) thermal noise (ThN); 2) flicker noise (FN); and 3) shot noise (SN) [34], [35]. During the enrollment phase, we are in fact establishing a noise level reference, that is

$$\text{TN} = \text{ThN} + \text{FN} + \text{SN} \tag{1}$$

where TN is the total noise. First, ThN is related to the scattering of carrier charges in thermal motion, and is directly

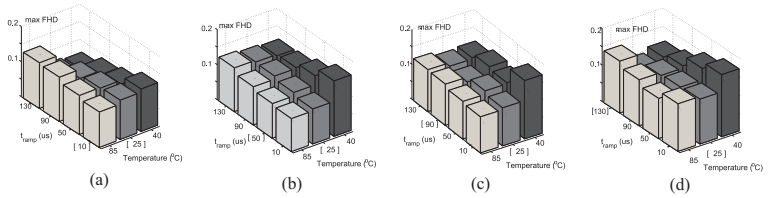


Fig. 4. max FHD; enrollment performed at 25 °C with t_{ramp} of (a) 10 μs, (b) 50 μs, (c) 90 μs, and (d) 130 μs.

proportional to the temperature, i.e., the higher the temperature, the higher the noise. In addition, in short-channel devices, ThN increases with increase in gate-to-source and drain-to-source voltages [34], [35]. Second, FN, also known as $1/f$ noise, is related to trapping and releasing charges near the $Si–SiO_2$ interface (silicon–silicon dioxide), and is inversely proportional to the frequency. For short-channel devices, a special case of FN occurs—the random telegraph noise (RTN). In fact, FN is the sum of a large amount of RTN [34], [35]. The fingerprints of memory-based PUFs are determined during one single power-up with a certain t_{ramp}; such t_{ramp} can be seen as a part of a periodic signal (e.g., sawtooth signal), and therefore different t_{ramp} corresponds to different signal frequencies influencing FN in different ways. Finally, SN is related to charges overcoming potential barriers, such as moving from the source to the channel; this type of noise is directly proportional to the electrical current [34], [35]. It is worth noting that ThN and FN have much larger impact than SN in the frequency range considered [36]. At higher temperatures, the PUF suffers from higher ThN as compared with enrollment done at lower temperature. To compensate for such noise and get the overall noise close to that of the enrollment, we can reduce the FN at higher temperature reducing the t_{ramp}. At lower temperature, the impact is opposite.

IV. Silicon Validation

To validate the simulation results, we performed silicon measurements on three different types of memory-based PUFs: the SRAM PUF [6], [17], the D flip-flop (DFF) PUF [21], and the buskeeper (BK) PUF [22].

A. Test Setup

The considered memory-based PUF types are manufactured in three different technology nodes. Table I provides an overview of all devices, including the technology node, the number of available integrated circuits (ICs), the number of PUF instances per IC in the given technology (if any), and the total number of tested instances of each PUF type. Note that, each IC contains one or more PUF instances.

Measurements are performed at three different temperatures (−40, 25, and 85 °C) and for ten different t_{ramp} varying from 10 μs up to 500 ms. In case of 40 nm SRAM, the shortest possible t_{ramp} is 50 μs due to specific capacitive load. The measurements flow is as follows.

1) The ICs are placed in a climate chamber and connected to a programmable power supply.
2) Climate chamber is set to one of the test temperatures.
3) ICs are powered with a t_{ramp} from the test set.
4) Each PUF device response is read and stored in a file.
5) The ICs are powered down for 1 s.
6) Steps 3–5 are repeated nine times (i.e., ten measurements per PUF per temperature per t_{ramp}).
7) Change t_{ramp} and repeat steps 3–6 (until all values of t_{ramp} have been tested for this temperature).
8) Change temperature and repeat steps 3–7.

TABLE I
Description of Devices Used in Validation

Technology	# ICs	# PUF inst. / IC			Total # PUF inst.		
		BK	DFF	SRAM	BK	DFF	SRAM
40nm LP	5	-	-	3	-	-	15
65nm LP	50	2	4	4	100	200	200
130nm LP	16	-	1	1	-	16	16

B. Evaluation Metrics

1) Reproducibility: To calculate FHD, first an enrollment response of each PUF instance is measured. Thereafter, each reconstruction measurement is compared to this enrollment by counting the number of flipped bits and dividing it by the response length. A key based on the PR (as described in Section II) is reliable if the worst-case FHD under any stress condition is below the ECCap of the ECC. Hence, the smaller FHD, the lower the required error correction.

2) Uniqueness: We evaluate the uniqueness at enrollment of the different PUF implementations by using: 1) the average between-class Hamming distance (μ-BCHD) [17] and 2) the estimated min-entropy (H_∞) [17] of the measured responses. Note that, for key storage application (as described in Section II) only the uniqueness of the enrollment PR is critical, as it is from this response that the cryptographic key is derived. μ-BCHD is calculated as follows.

1) The enrollment response of each PUF is measured.
2) The Hamming distance between each pair of enrollment responses (i.e., between-class) coming from different PUF instances of the same type is determined (e.g., between all pairs of enrollment responses of 65 nm LP SRAM PUFs are computed).
3) The distribution of these between-class distances is determined and the obtained mean value, normalized to the response length, is μ-BCHD.

Optimally, the obtained distribution should be approximately Gaussian and μ-BCHD should be very close to 50% [17].

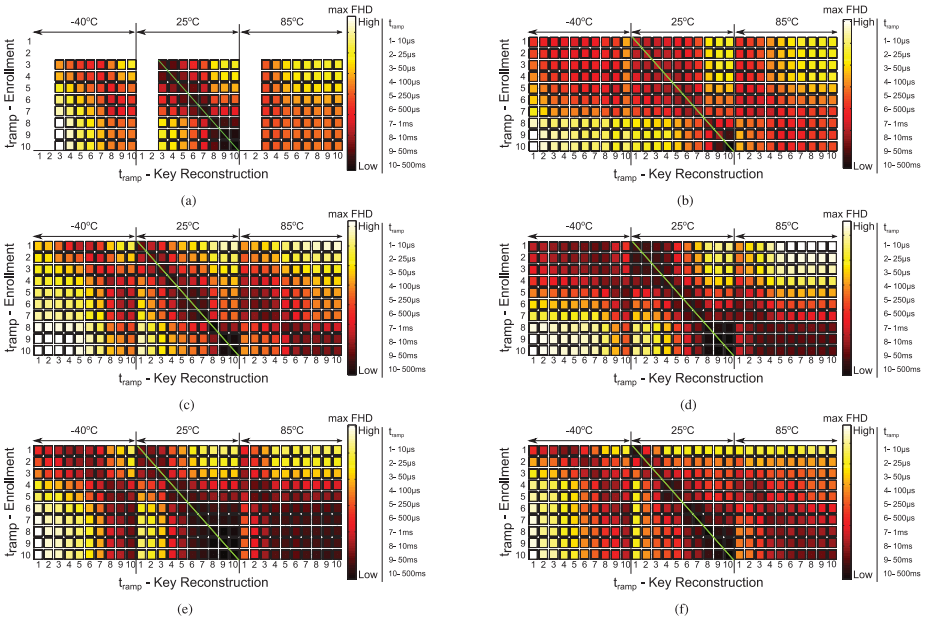


Fig. 5. max FHD for various t_{ramp} enrollment (green line) and key reconstruction. (a) 40 nm, (b) 65 nm, and (e) 130 nm SRAM PUF. (c) 65 nm and (f) 130 nm DFF PUF. (d) 65 nm BK PUF.

H_∞ is used to evaluate the intrinsic unpredictability of PRs. H_∞ is a pessimistic measure of a random variable unpredictability [8]. We estimate H_∞ of the responses of a particular PUF type by considering the following model: each PR bit is assumed to be independent of the other bits in the same response, and that it has an individual probability p_1 of being 1 for a random PUF instance. This model is particularly reasonable for memory-based PUFs, as each response bit originates from an individual memory cell. Under the assumption of this model, $H_\infty = -\log_2 \max\{p_1, 1-p_1\}$ for a single response bit [6]. The value for p_1 of a bit is estimated by counting the number of enrollment responses for which this bit is 1 and dividing by the total number of enrollment responses. The entire response H_∞ is simply the summation of H_∞ of each bit. We express H_∞ as the average H_∞ per bit in a response value, by dividing the total H_∞ of the response by its length. Optimally, H_∞ of a PR bit should be close to 1. Note that, due to the limited number of measured PUF instances, the obtained estimations of H_∞ could be lower than the actual PRs H_∞.

C. Optimization Algorithms

The silicon measurements have the objective to investigate the use of t_{ramp} as a technique for increasing memory-based PR reproducibility (noise reduction). As a side effect, the impact on PUF uniqueness is also investigated. For this purpose, two optimization algorithms are used.

1) Reproducibility Optimization: This algorithm identifies the enrollment t_{ramp} that leads to the highest reproducibility (lowest maximum noise).

2) Uniqueness Optimization: This algorithm identifies the enrollment t_{ramp} that provides the highest H_∞. After this first step the values of t_{ramp} at other temperatures are determined, which minimize the noise.

D. Measurement Results

In order to evaluate the performance of the optimization algorithms, first we analyzed the max FHD for all t_{ramp} enrollment key reconstruction combinations. Fig. 5 shows the results; the t_{ramp} used for enrollment (at 25 °C, also highlighted by a green line) and key reconstruction (at −40, 25, and 85 °C) are represented on the y- and x-axis, respectively, whereas the max FHD is represented by color. These values are obtained using t_{ramp} from 10 μs, which is the shortest feasible t_{ramp} for each PUF, except for 40 nm LP SRAM PUF where the shortest feasible t_{ramp} is 50 μs, up to 500 ms. The max FHD (noise) is determined using ten response measurements per PUF per temperature per t_{ramp}.

Fig. 5(a) reveals a clear convergence pattern toward a local minimum max FHD for each temperature/t_{ramp} enrollment combination. The local minimum max FHD is achieved for t_{ramp} longer than that of enrollment for −40 °C, the same as that of enrollment for 25 °C and shorter than that of enrollment for 85 °C. For example, considering enrollment

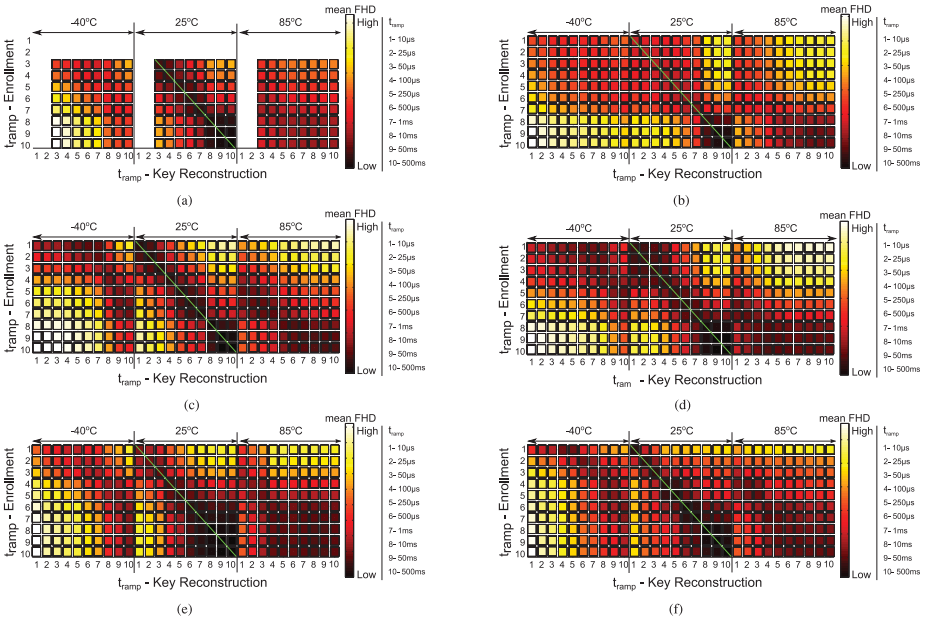


Fig. 6. Mean FHD for various t_{ramp} enrollment (green line) and key reconstruction. (a) 40 nm, (b) 65 nm, and (e) 130 nm SRAM PUF. (c) 65 nm and (f) 130 nm DFF PUF. (d) 65 nm BK PUF.

TABLE II
MEASUREMENT RESULTS WITHOUT OPTIMIZATION

Technology	PUF	t_{ramp}	Maximum noise FHD			μ-BCHD	H_∞
			$-40°C$	$+25°C$	$+85°C$		
40nm LP	SRAM	$50\mu s$	23%	6%	20%	0.50	0.73
65nm LP	SRAM	$10\mu s$	8%	6%	8%	0.50	0.87
	DFF	$10\mu s$	28%	8%	25%	0.37	0.40
	BK	$10\mu s$	10.5%	4.5%	20%	0.48	0.75
130nm LP	SRAM	$10\mu s$	13%	6%	12%	0.47	0.66
	DFF	$10\mu s$	16.5%	5%	28%	0.43	0.61

TABLE III
RESULTS AFTER REPRODUCIBILITY OPTIMIZATION

Technology	PUF	t_{ramp}			Maximum noise FHD			μ-BCHD	H_∞
		$-40°C$	$+25°C$	$+85°C$	$-40°C$	$+25°C$	$+85°C$		
40nm LP	SRAM	10ms	1ms	$50\mu s$	14%	4.5%	17%	0.49	0.71
65nm LP	SRAM	50ms	$250\mu s$	$10\mu s$	7%	5.5%	7%	0.50	0.89
	DFF	50ms	$500\mu s$	$25\mu s$	11.5%	5%	9%	0.49	0.84
	BK	500ms	1ms	$25\mu s$	6.5%	4%	6.5%	0.49	0.81
130nm LP	SRAM	500ms	10ms	1ms	5.5%	2%	4%	0.37	0.42
	DFF	500ms	10ms	$500\mu s$	9.0%	3.0%	9.0%	0.46	0.63

3 (i.e., t_{ramp} at 50 μs), for -40 °C max FHD decreases until t_{ramp} 5, increasing thereafter, while for both 25 and 85 °C max FHD increases with t_{ramp} increase. Similar trends are observed for the other PUF types and technology nodes.

Table II summarizes the information of Fig. 5 for the shortest enrollment t_{ramp} per PUF type; i.e., it shows the original measured maximum noise values for the considered temperatures. Moreover, it shows the uniqueness indicators. Table II is used as reference to compare the results of the proposed optimization algorithms against, as the enrollment conditions are the standard ones.

Table II reveals that, overall, the maximum noise measured is 28% at -40 °C (for the 65 nm DFF PUF), 8% at 25 °C (for the 65 nm DFF PUF), and 28% at 85 °C (for the 130 nm DFF PUF). Regarding uniqueness, although a truly fair comparison is not possible due to limited available devices per technology node and PUF type, the 65 nm DFF PUF has the lowest μ-BCHD $= 0.37$ and $H_\infty = 0.40$.

In addition, to investigate whether the observed convergence toward a local minimum holds for the mean FHD, we perform a similar analysis as for max FHD. Fig. 6 shows the results. The mean FHD (noise) is determined using ten response measurements per PUF per temperature per t_{ramp}. Fig. 6 reveals the same convergence trend observed in Fig. 5.

1) Reproducibility Optimization: Table III presents the reproducibility optimization algorithm results; it shows the t_{ramp} configuration that minimizes the noise (maximizes reproducibility) per temperature in comparison to enrollment. The results reveal that for all tested PUFs, adapting t_{ramp} to the ambient temperature has a major impact on the maximum noise. For low temperatures, noise reduction is realized with

TABLE IV
RESULTS AFTER UNIQUENESS OPTIMIZATION

Technology	PUF	t_{ramp}			Maximum noise FHD			μ-BCHD	H_∞
		$-40°C$	$+25°C$	$+85°C$	$-40°C$	$+25°C$	$+85°C$		
40nm LP	SRAM	1ms	100μs	50μs	16%	6%	19%	0.50	0.73
65nm LP	SRAM	50ms	100ms	50μs	13%	2%	8%	0.50	0.89
	DFF	500ms	10ms	250μs	18.5%	2.5%	8%	0.50	0.90
	BK	100ms	250μs	10μs	7%	5%	9%	0.50	0.88
130nm LP	SRAM	1ms	10μs	10μs	7.5%	6%	12%	0.47	0.66
	DFF	50ms	500μs	10μs	10%	4.5%	9.5%	0.47	0.67

longer t_{ramp}; whereas for high temperatures this is realized with shorter t_{ramp}. For example, the maximum noise for 65 nm LP DFF PUF at -40 °C with $t_{\text{ramp}} = 10$ μs for both enrollment and reconstruction is originally 28%. However, if the optimized t_{ramp} is used both at enrollment (500 μs at 25 °C) and at reconstruction (50 ms at -40 °C), then the maximum noise is reduced to merely 11.5%. Note that, all results of Table III follow the same trend, as predicted by the simulation results of Section III-D. Since this algorithm does not optimize uniqueness, μ-BCHD and H_∞ are deteriorated for some PUFs (e.g., 130 nm SRAM PUF), while they are significantly improved for others (e.g., 65 nm DFF PUF).

2) Uniqueness Optimization: Table IV reports the uniqueness optimization algorithm results; it shows: 1) the t_{ramp} at enrollment that maximizes uniqueness and 2) the t_{ramp} for the other temperatures that results in the lowest maximum noise (with respect to the t_{ramp} selected for enrollment). Uniqueness indicators μ-BCHD and H_∞ are at least as high as the originals for 40 and 130 nm SRAMs, and for the remaining devices these indicators are higher than the original indicators. The uniqueness optimization algorithm clearly leads to significant improvements in μ-BCHD and H_∞ for DFF and BK PUFs. However, this improvement is negligible for the SRAM PUFs for all tested nodes. Since this algorithm does not select the enrollment t_{ramp} optimized for reproducibility, it is natural that the noise resulting from this algorithm is worse than that of reproducibility optimization algorithm. In case of the 65 nm SRAM PUF, the maximum noise at -40 °C is even worse than the measurements without optimization. Reason for this is that t_{ramp} at enrollment (25 °C) is very long and the algorithm is unable to find a corresponding longer t_{ramp} at -40 °C.

E. Discussion

SPICE simulations show that using long t_{ramp} at low temperatures and short t_{ramp} at high temperatures results in reduced SRAM PR noise when compared to enrollment. The observation is validated using silicon measurements, and holds for all technology nodes and memory PUF type investigated. Hence, choosing appropriate t_{ramp} according to ambient temperature, including enrollment, can be used as an efficient scheme to reduce noise and increase reproducibility.

Moreover, the silicon measurements have also indicated that varying t_{ramp} can have a significant impact on the uniqueness of memory-based PUFs. We can conclude from our measurements that t_{ramp} can slightly bias the fingerprints of memory-based PUFs. The bias is visible by the uniqueness metrics, as these represent the correlation between fingerprints during enrollment. When selecting a certain t_{ramp} we are either enhancing this bias behavior (for reliability optimization) or neutralizing it (for uniqueness optimization); e.g., a PUF device that would generate a response of only 1s would be 100% reliable (FHD = 0), however, it would not be unique.

By choosing the proper optimization algorithm according to the PUF type, noise can be reduced when compared to the original results in Table II while either maintaining or increasing the uniqueness indicators. Inspecting the silicon results with respect to reproducibility and uniqueness reveals the following.

1) The 40 and 65 nm SRAM PUFs benefit from applying the reproducibility optimization algorithm, but the uniqueness optimization algorithm is not very effective as there is very little margin for improvement. Furthermore, the uniqueness optimization algorithm does not significantly minimize the noise for the tested SRAMs.
2) The 130 nm SRAM PUFs benefit from applying the uniqueness optimization algorithm, as the noise is reduced while the uniqueness is maintained.
3) BK and DFF PUFs benefit from applying the uniqueness optimization algorithm, since the original silicon results show that there is a lot of room for improvement. Besides increasing the PR uniqueness, the proposed algorithm also decreases the noise at -40 and 85 °C temperatures. Hence, this algorithm works very well for these PUF types.

V. NOISE REDUCTION IMPACT ON AREA OVERHEAD

In this section, we investigate the noise reduction impact on the area overhead of memory-based PUFs by means of adapting the voltage ramp-up time to the temperature. First, we briefly describe the FE and its possible configurations. Then, we relate noise with area overhead. Thereafter, we define a set of experiments to investigate the impact noise reduction has on the area overhead. And finally, we show and discuss the results of the experiments.

A. Types of Fuzzy Extractor Constructions

An FE is a fundamental component of a PUF-based key storage system (see Fig. 2); it has two main functions.

1) *Information Reconciliation:* It uses the helper data to correct errors on the measured PR.
2) *Privacy Amplification:* Considering that the helper data contains information on the PRR, privacy amplification is needed to make sure that the helper data does not reveal any information on the derived cryptographic key.

The FE compresses the resulting data into a cryptographic key with maximum entropy making it impossible for an attacker to guess the key [8], [9]; it also removes any biasing (unequal distribution of zeros and ones) in the error-corrected PR.

Information reconciliation is enabled by error correction blocks, while privacy amplification is enabled by hash function, see Fig. 7. The number and type of error correction blocks depends on both noise and application of each PUF-based system. Encoder blocks are used to add redundancy to the original data during the enrollment phase, while decoder blocks aim at recovering the original data during the key reconstruction phase. The hash function concludes this phase.

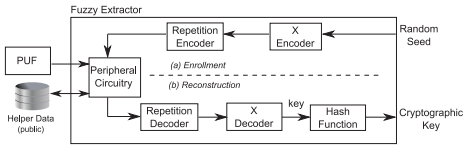


Fig. 7. FE.

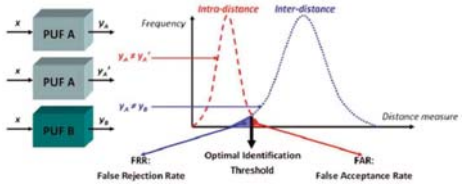


Fig. 8. FRR and FAR [27].

There are several popular constructions with respect to the type of error correction blocks and their parameters. Error correction blocks can be classified into block codes and convolutional codes. Block codes are memoryless, i.e., the encoder's output at any given time depends only on the input at that time. They are easy to implement, efficient with small data, and have low area overhead. However, they suffer from lower ECCap when compared to the convolutional codes. On the other hand, convolutional codes have memory, i.e., the encoder's outputs at any given time (t) depends not only on the inputs at that time unit but also on some of previous inputs. They have higher ECC capabilities. However, convolutional codes require long data streams to work efficiently, are complex to implement and have higher area overhead. For these reasons, block codes are the most used in FE for PUF-based systems.

There are various types of FE constructions using linear block codes for error correction; typical constructions comprise repetition code followed by either Golay code or Reed–Muller code [27]. The aforementioned FE constructions owe their popularity to their area overhead efficiency when compared with their Bose Chaudhuri Hocquenghem counterparts, while delivering the same error correction efficiency [27], [28]. For this reason this paper focus on these FE constructions. Fig. 7 depicts a generic FE; it comprises a repetition code and a generic X code representing either a Golay [24, 12, 8] code, or a Reed–Muller16 [16, 5, 8] code, or a Reed–Muller8 [8, 4, 4] (note that, the used codes have n length, k secret bits, and d minimum Hamming distance, resulting in $[n, k, d]$).

B. Linking Noise Reduction to Area Overhead

A high quality PUF-based system is the one which: 1) efficiently reconstructs a valid cryptographic key from a true PUF device (the one used for enrollment) under various conditions and 2) does not reconstruct a valid cryptographic key from a false PUF device (any device different than the one of enrollment being illegally used to reconstruct the key of the true device). Common quality metrics used for PUF-based systems are false rejection rate (FRR) and false acceptance rate (FAR) [28]. FRR is the probability that the noise of a PR of true device A is above the error correction capabilities of the PUF-system and therefore, the authentication of true device A is rejected. FAR is the probability that the noise of a PR of device B is such that it is mistakenly corrected to the PRR of device A and therefore, device B is falsely authenticated as A. FRR and FAR are exemplified in Fig. 8 [27]; the figure shows two FHD histograms: the intra-FHD (i.e., noise) on the left side and the inter-FHD (i.e., the FHD among different devices) on the right side. When designing a PUF-based system, ideally all intra-FHD would be corrected and at the same time each device would be perfectly distinguishable from others. However, in reality the two histograms overlap, resulting into two different areas FRR and FAR. The optimal identification threshold is when FRR = FAR.

To investigate the impact of PUF noise reduction on the area of PUF system, we need to estimate the quality metrics as a function of the raw PR and ECCap. Let us consider the system shown in Fig. 7. During key reconstruction, an FE is able to successfully reconstruct the cryptographic key only when the output of the X decoder is correct for all decoding iterations (note that, the successful reconstruction tolerates errors at the output of the repetition decoder, as long as these errors are corrected by the X decoder); i.e., when the number of errors are within error correction capabilities of the PUF-system. Assume that the hash function needs an input key with a length "l" to produce the required cryptographic key; the key is generated by multiple iterations of the decoding path (i.e., repetition decoder combined with X decoder). In addition, assume that the number of secret bits per decoding iteration is k [28]; these bits reflect the original information coming from the PUF and the random seed (see Fig. 7), and not the redundant bits introduced by the encoding and decoding. To generate key with a length l, we need $\lceil l/k \rceil$ decoding iterations. The probability that a true key is not reconstructed can be expressed as [28]

$$\mathrm{FRR} = 1 - (1 - \mathrm{PE}_{\mathrm{Xcode}})^{\mathrm{iterations}} \tag{2}$$

where $\mathrm{PE}_{\mathrm{Xcode}}$ is the probability that one or more errors occur above the error correction capabilities of X decoder. Note that, $(1 - \mathrm{PE}_{\mathrm{Xcode}})^{\mathrm{iterations}}$ denotes the probability that all errors are corrected for all the decoding iterations. $\mathrm{PE}_{\mathrm{Xcode}}$ can be expressed as [28]

$$\begin{aligned}\mathrm{PE}_{\mathrm{Xcode}} &= \sum_{i=t+1}^{s} \binom{s}{i} \mathrm{PE}_{\mathrm{rep}}^{i} (1 - \mathrm{PE}_{\mathrm{rep}})^{s-i} \\ &= 1 - \sum_{i=0}^{t} \binom{s}{i} \mathrm{PE}_{\mathrm{rep}}^{i} (1 - \mathrm{PE}_{\mathrm{rep}})^{s-i}\end{aligned} \tag{3}$$

where t and s are X decoder ECCap and code length, respectively, and $\mathrm{PE}_{\mathrm{rep}}$ is the probability that one or more errors occur above the error correction capabilities of repetition decoder (see Fig. 7). $\mathrm{PE}_{\mathrm{rep}}$ can be estimated as [28]

$$\begin{aligned}\mathrm{PE}_{\mathrm{rep}} &= \sum_{i=\lceil n/2 \rceil}^{n} \binom{n}{i} \epsilon^{i} (1 - \epsilon)^{n-i} \\ &= 1 - \sum_{i=0}^{\lfloor n/2 \rfloor} \binom{n}{i} \epsilon^{i} (1 - \epsilon)^{n-i}\end{aligned} \tag{4}$$

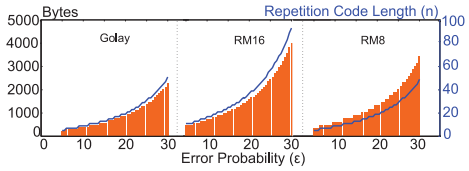


Fig. 9. PUF size (bars) and repetition code length (line) versus error probability (ϵ).

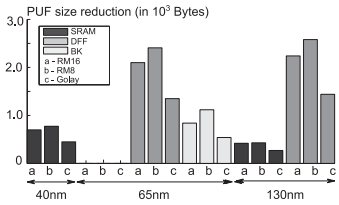


Fig. 10. Absolute PUF size reduction.

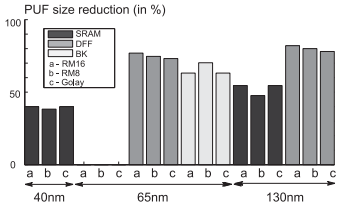


Fig. 11. Relative PUF size reduction.

where n is the repetition decoder code length and ϵ is the PUF-response error probability, see Fig. 7. Note that, the repetition encoder is not used during Key Reconstruction. Using the previous equations, one can easily determine n as a function of ϵ for a given FRR and s; say $n = f(\epsilon)$.

The size of the required PUF data can be estimated [28]

$$\begin{aligned} \text{PUF}_{\text{bits}} &= (\text{code length X decoder}) \\ &\quad \times (\text{code length repetition decoder}) \\ &\quad \times (\text{number iterations}) \\ &= s \times n \times \text{iterations} \\ &= s \times \text{iterations} \times f(\epsilon). \end{aligned} \tag{5}$$

C. Simulation Setup

To estimate the noise reduction impact on PUF-based systems area for several FE construction types, we use the equations introduced in the previous section with the following set of values.

1) FRR $= 10^{-6}$ [28].
2) The key (input of hash function) has a length $l = 171$ bits; here, we assume that we want to generate a key of 128 bits of entropy and we consider a secrecy rate (minimal amount of compression that needs to be applied to a PUF fingerprint by the hash function) of 0.75 [6], [27], hence, $\lceil 128/0.75 \rceil = 171$ bits are required [27].

In addition, we perform the simulation for the following scenarios.

1) Fifty-one different ϵ; we sweep ϵ from 5% up to 30% with a step of 0.5%.
2) Three different combinations of s and k; these reflect three FE constructions: a) Golay-based with $\{s, k\} = \{24, 12\}$; b) RM16-based with $\{s, k\} = \{16, 5\}$; and c) RM8-based $\{s, k\} = \{8, 4\}$.

D. Results and Analysis

Fig. 9 shows the results for each of the three FE constructions investigated; the left y-axis (bars) depicts the required memory (in bytes), the right y-axis (line) the required repetition code length, and the x-axis the PR error probability ϵ. From the figure we can make the following conclusions.

1) Reduction in noise ϵ significantly reduces the required PUF size and n. Regardless of the FE construction, the lower ϵ, the lower the PUF size and the lower the repetition code length. For example, when $\epsilon = 15\%$, a RM16-based PUF system requires 910 PUF bits and a repetition code of length $n = 13$, while when $\epsilon = 5\%$ only 350 PUF bits and $n = 5$ are required to realize the same quality (FRR); hence, a noise reduction of $3\times$ causes a $2.6\times$ reduction in both PUF size and n.
2) Golay-based and RM16-based PUF systems are the ones benefiting the most from our technique; their PUF size and n reduces by $2.6\times$ when ϵ reduces from 15% to 5%. However, this is only $2.3\times$ for RM8-based. Moreover, overall, Golay-based PUF system is the one with smaller PUF size and n for any given ϵ.

Now that we have determined the PUF size as a function of the noise, we can estimate the saved PUF size based on our method by first estimating the PUF size of the PUFs shown in Table II (without voltage ramp-up optimization) and thereafter for those shown in Table III (with voltage ramp-up optimization). This will be done as follows.

1) For each of the PUFs in Table II, select the maximum noise FHD (=ϵ), and use Fig. 9 to calculate the required PUF size.
2) For each of the PUFs in Table III, select the maximum noise FHD, and use Fig. 9 to calculate the required PUF size.
3) Determine the savings in PUF size by subtracting the PUF size values found in 2) from those found in 1).

The results are plotted in Figs. 10 and 11. Fig. 10 shows the absolute PUF size reduction while Fig. 11 shows the relative PUF size reduction. The results show that the area savings are strongly PUF type and FE construction dependent. DFF PUFs are the ones benefiting the most; e.g., 130 nm DFF RM16-based requires 2.24K bytes less of PUF material, i.e., a reduction of 82.1%. On the other hand SRAM PUFs are the ones benefiting the least; although that for 40 and 130 nm a quite saving is achieved for all FE constructions, almost no saving is realized for 65 nm irrespective of the FE construction. This is due to the small improvement that the optimization algorithm has on this PUF type, see Tables II and III.

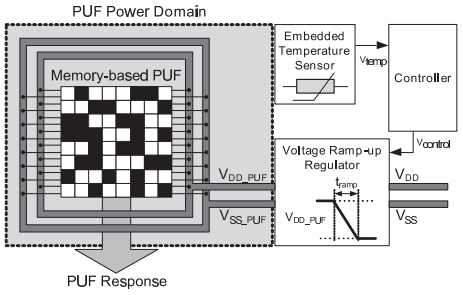


Fig. 12. Schematic of an extended memory-based PUF design.

VI. Adapter-Circuit Implementation

The proposed noise reduction scheme can be implemented by a simple circuit consisting of a temperature sensor, a controller and a voltage regulator. In this section, first we define the requirements of such a circuit. Then, we propose and implement our solution. Finally, we extract the circuit characteristics and discuss them.

A. Requirements

We divide the requirements into design requirements and functional requirements. From design perspective the proposed noise reduction scheme has an added value only if the area of the circuit (which enables various t_{ramp} according to the sensed temperature) is less than the area of the memory it saves. As seen in the previous section, the saved area varies with technology node, memory-PUF type, and FE construction. Due to this, we have different area budgets for the different scenarios, ranging from virtually 0 GE (for 65 nm SRAM PUF) up to 20 kGE (for the 65 nm DFF PUF); gate equivalent (GE) is a technology node independent metric of area that denotes the area of NAND2 with standard drive strengths. Note that, 1 GE is considered as a reasonable estimate of a single SRAM, DFF or BK cell for any of the investigated technologies according to [29]–[31].

In addition, as PUF-based systems are active only during the start-up of a device to generate the key, delay, and power consumption play very minor roles. Therefore, we consider the area overhead to be our main design requirement.

With respect to functional requirements, a set of targets is defined. Table III shows that the optimal t_{ramp} per sensed temperature varies with technology node and memory-PUF type. Hence, as there are several possible configurations, we decided to target the extreme values of t_{ramp}; i.e., t_{ramp} 10 μs at 85 °C, 1 ms at 25 °C, and 500 ms at −40 °C.

In short, the requirements are as follows.

1) Low area overhead (up to budget).
2) Output $t_{ramp} = 10$ μs at 85 °C, $t_{ramp} = 1$ ms at 25 °C, and $t_{ramp} = 500$ ms at −40 °C.

B. Adapter-Circuit

Fig. 12 shows the block diagram of a memory-based PUF extended with the adapter circuit. This system comprises four blocks: a memory-based PUF, an embedded temperature sensor, a controller, and a voltage ramp-up regulator. It performs five main steps. First, the temperature sensor senses the ambient temperature and outputs V_{temp}. Second, V_{temp} is used as the input to the controller, which accordingly, generates a calibration voltage $V_{control}$. Third, $V_{control}$ is used as an input to the voltage ramp-up regulator, which outputs a t_{ramp} that minimizes the FHD (noise). Finally, the memory-based PUF is powered-up with the assigned t_{ramp}, generating a PR.

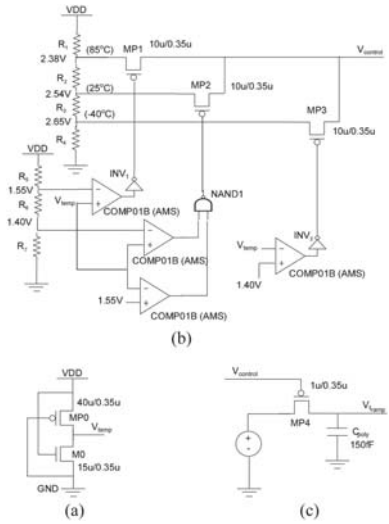


Fig. 13. Adapter circuit schematic. (a) Temperature sensor. (b) Controller. (c) Voltage ramp-up regulator.

One of the main advantages of the proposed optimization technique, besides its evident effectiveness, is that its implementation demands no adaptations of the memory-based PUF circuit itself. In fact, the basic PUF comprises only standard library memory cells, but needs to be placed in its own power domain and extended with an embedded temperature sensor, a voltage ramp-up regulator and controller. The general design of these extensions is schematically shown in Fig. 12. Since the concerned building blocks are all rather standard, the implementation effort of the proposed optimization technique is considered minimal.

C. Implementation

Fig. 13 shows the schematic of the circuit; where Fig. 13(a) depicts the embedded temperature sensor, Fig. 13(b) the controller, and Fig. 13(c) the voltage ramp-up regulator. The circuit is implemented in 0.35 μm, due to lack of availability of smaller technologies, and with AMS technology. We implement a temperature sensor comprising two MOSFETs (MP0 and M0). The sensor outputs a voltage (V_{temp}) that is proportional to the sensed temperature.

The controller, Fig. 13(b), is an intermediary circuitry that maps its input voltage V_{temp} to its output voltage V_{contol}. Each one of the three pMOS (one pMOS per voltage ramp-up time) has at its drain the specific voltage that is required for the voltage ramp-up regulator to deliver the specific t_{ramp}; MP1 for 85 °C, MP2 for −40 °C and MP3 for 25 °C. When a certain temperature is sensed, only the pMOS transistor that represents the closest temperature should drive. The selection of the driving transistor is done via the operational amplifiers, which are used as comparators in this configuration. The voltage outputted by the temperature sensor is compared against the reference values for each temperature. For the extreme temperatures (i.e., −40 and 85 °C) only one comparison is required as we only need to make sure that the V_{temp} is either above (for 85 °C) or below (for −40 °C) the reference voltage of the respective temperature. For intermediary temperatures (i.e., 25 °C) two comparisons are required (hence, two operational amplifiers) as we need to make sure that the received V_{temp} is above a reference and below another. The output of the comparisons for the extreme temperatures needs to be inverted (INV0 and INV1) as pMOS are active for low-voltage at their gates. With the output of the two comparisons of the intermediary temperature we perform an AND (AND0) operation as MP3 should be driven only when both comparisons are true. Finally, two networks of voltage dividers (one comprised by R1, R2, R3, and R4, and the second comprised by R5, R6, and R7) are used to define the reference voltages at the drain of the pMOS and at the inputs of the operational amplifiers, respectively.

The voltage ramp-up regulator, Fig. 13(c), is a basic RC circuit, where the resistor has been replaced by an MOSFET. By varying the voltage at the gate of the MOSFET MP4 we can tune its resistance such that the time constant of the circuit is the one of our specifications (i.e., 10 μs at 85 °C, 1 ms at 25 °C, and 500 ms at −40 °C).

It is worth emphasizing that the proposed circuit generates more than just the three specified voltage ramp-up times for enrollment and extreme temperature corners. The voltage ramp-up time decreases monotonically from 500 ms down to 10 μs, as the temperature increases from −40 °C up to +85 °C; from the continuous range of voltage ramp-up times, we fix the values for the enrollment and extreme corners. The voltage ramp-up times for the remaining temperatures are intrinsically generated by the change in the resistance of the MOSFET MP4 of the voltage ramp-up regulator. This feature is a big plus of the design as it provides larger voltage ramp-up time granularity while not increasing the area overhead of the circuit.

D. Results

The results show that the circuit successfully maps the ambient temperature into the required voltage ramp-up time.

Fig. 14 shows the results for the voltage ramp-up regulator circuit; the circuit outputs at −40 °C a t_{ramp} of 500 ms, at 25 °C a t_{ramp} of 1 ms, and at 85 °C a t_{ramp} of 10 μs, as required. Moreover, as predicted, the voltage ramp-up time decreases continuous and monotonically from 500 ms down to 10 μs, as the temperature increases from −40 °C up to 85 °C; e.g., at −30 °C the circuit outputs a t_{ramp} of 358 ms, while at 75 °C it outputs a t_{ramp} of 12.6 μs. These results

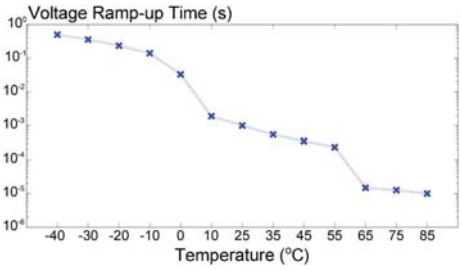


Fig. 14. Voltage ramp-up time versus temperature.

reveal the extra resolution of the circuit, which is realized for free (i.e., with no extra area overhead). The voltage ramp-up regulator has an area of 563.36 μm^2 (22.4 × 25.15 μm) which is fixed regardless of the resolution of the system and it is easily implementable in other technology nodes.

The controller, as designed, outputs one of the three reference voltages (Vm40 = 2.65 V, V25 = 2.53 V or V85 = 2.38 V). It has an area of 0.014 mm^2 (71.5 × 204.5 μm), which 90% corresponds to the area of the operational amplifiers (area of one operational amplifier 0.0034 mm^2). The controller is easily implementable in other technology nodes.

The temperature sensor outputs a voltage with a linear relation with the temperature; V_{temp} is 1.32 V at −40 °C, 1.47 V at 25 °C, and 1.61 V at 85 °C, which results in a resolution of 2.5 mV/°C. The temperature sensor has an area of 169.035 μm^2 (8.85 × 19.1 μm). Moreover, the sensor has a fixed area regardless of the resolution of the system and it is easily implementable in other technology nodes.

Overall, the circuit has an area overhead of 0.015 mm^2 (70.9 × 214.75 μm).

VII. DISCUSSION AND COMPARISON

In this section, first we discuss the impact of our scheme on area overhead, second that of on the delay and finally we discuss the procedure for investigating the temperature/voltage ramp-up time for other PUFs.

A. Impact on Area Overhead

To evaluate the attractiveness of integrating the adaptive circuit when compared with the classic approach, we need to determine the overall area before and after the optimization and compare them. As the adaptive circuit and the investigated memory-PUFs are implemented in different technology nodes, we cannot directly compare the areas; we need a fair comparison metric. Therefore, we convert the area of the adaptive circuit to GE according to [33]; 0.015 mm^2 corresponds to 275 GE (= $\lceil 0.015\ \text{mm}^2/54.6\ \mu m^2 \rceil$, where 54.6 μm^2 corresponds to the area of NAND2 cell in 0.35 nm [33]). We can determine the overall reduction in area overhead as follows. Add the 275 GE of the adaptive circuit to that of the PUF-system after the optimization and compare it with the PUF-system before the optimization. The results are depicted in Figs. 15 and 16. Fig. 15 shows the area overhead,

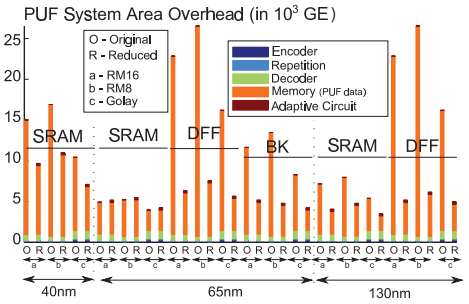


Fig. 15. Absolute area overhead without (O) and with (R) noise reduction.

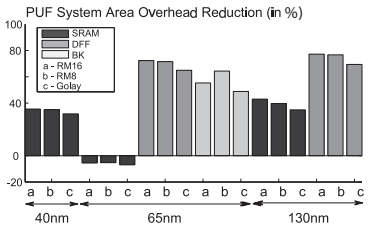


Fig. 16. Relative area overhead reduction.

before (Original) and after (Reduction) the noise reduction, for the different PUF-systems constructions investigated. The area overhead values of encoders, repetition and decoders, for the various constructions, were extracted from [28]. Fig. 16 shows the relative area overhead reduction in percentage. From Fig. 15, we can conclude the following. First, for all memory-PUF system constructions, the block that impacts the most the area overhead is the memory (PUF data size). Therefore, methods targeting noise reduction (resulting in memory reduction), such as the one proposed in this paper, are good allies to reduce the overall cost of the system. Second, the area overhead of Golay, RM16 or RM8 is not impacted by the noise reduction; the implementation of these blocks is independent from PUF noise (ϵ) as these encode/decode a standard number of bits per iteration.

Note that, in the figure we assumed the area overhead of the repetition code as constant. This is a conservative assumption, as in truth, the area overhead of this block is reduced as the noise decreases. As seen in [32], the repetition code hardware implementation comprises a counter, which counts up to n (length of the repetition code). The higher the n the higher the area overhead of the counter, hence, the higher the area overhead of the repetition code. We have seen in Fig. 9 that n decreases with noise, and so decreases the area overhead of the repetition code. Therefore, the overall area reduction is slightly greater than the one presented.

Considering both figures reveals that, overall, integrating the adapter circuit in a memory-based PUF system is an attractive solution. Five out of the six investigated PUF memories have their area overhead reduced, ranging from a minimum of 31.6% (40 nm SRAM) up to a maximum of 82.1% (130 nm DFF). The memory-PUF benefiting the most from this technique is the 130 nm DFF-PUF; not only its area overhead reduction ranges from a minimum of 78% up to 82.1% (depending on the FE construction) but also its noise reduces from 28% down to 9%, its μ-BCHD increases from 0.43 up to 0.46 and its H_∞ increases from 0.61 up to 0.63, see Tables II and III. Similar improvements are obtained for both 65 nm DFF-PUF and 65 nm BK-PUF. Applying the noise reduction method for SRAM-based PUF systems reduces its area overhead ranging from a minimum of 31.6% up to 35.2% for 40 nm, while this range is 34.9% up to 43.1% for 130 nm. For 65 nm SRAM there is an increase in area ranging from of 5.2% up to 6.9%; however, both noise and min entropy are improved. The results show that the proposed noise reduction solution is attractive for all memory-based PUFs, particularly DFF and BK PUFs.

Regarding the cost of adding extra specific temperature/voltage ramp-up pairs, we estimate the following. Each new specific temperature/voltage ramp-up pair impacts only the controller design. Per new pair, a similar set of components as those used for 25 °C are required; i.e., one pMOS, one NAND, and two operational amplifiers. The areas of the pMOS and the NAND are very small when compared with that of the operational amplifiers. Therefore, we can estimate that the cost of adding a new specific temperature/voltage ramp-up pair is roughly the area overhead of two operational amplifiers; i.e., 125 GE = $\lceil (2 \times 0.0034\ \text{mm}^2)/(54.6\ \mu m^2) \rceil$, see Section VI-D. To have a better feeling of the number of extra temperature/voltage ramp-up pairs that make the noise optimized solution achieve the same area overhead of the nonoptimized, we carry out the following steps. First, from Fig. 15, we identify the PUF-system construction that has the least absolute area overhead reduction, i.e., 130 nm SRAM Golay-based, and calculate this value. Second, we divide the value from the first step by the GE of the extra components, i.e., 125 GE. We estimate that up to 12 new pairs can be added to the least reduced PUF system (in absolute terms), i.e., a total of 15 (12 plus the three pairs implemented in the previous section) fixed temperature/voltage ramp-up pairs. Therefore, we conclude that the proposed noise reduction solution is advantageous for a wide range of fixed temperature/voltage ramp-up pairs.

Finally, we would like to mention that any PUF size variation is mirrored by the helper data; helper data and PUF have the same size, see Fig. 7, hence, any increase or decrease in the PUF size due to noise reduction is intrinsically followed by the helper data. However, in this paper, we consider the helper data as being stored off-chip, and therefore, our results do not reflect its area reduction with PUF noise optimization.

B. Impact on Delay

In this type of industry we can easily tradeoff delay over higher reproducibility and higher uniqueness. Nonetheless, a delay analysis reveals the following. The total computational time, from power-up up to key reconstruction can be expressed by $\text{Total}_{\text{Delay}} = \text{Delay}_{\text{Sensors}} + \text{Delay}_{t_{\text{ramp}}} + \text{Delay}_{\text{Decoding}}$. The delay introduced by the sensors is negligible. The delay introduced by the t_{ramp} when compared to the original construction can be significant (depending on the temperature at which the

reconstruction is performed). However, with less noise, less PUF data is required. Therefore, the delay of the decoding is reduced. The number of iterations is constant for any given temperature and/ or ramp-up combination. The outcome of the tradeoff between the increase in t_{ramp} and decrease in decoding time is highly dependent on the frequency applied (as the t_{ramp} is fixed). However, as the key reconstruction phase is typically performed during power-up only, the overall impact of the method on the overall delay of the circuit is negligible. In other words, the area savings compensate for an eventual and discrete delay increase.

C. Generic Procedure

To investigate the noise reduction we performed measurements on ten voltage ramp-up times widely distributed (10 μs, 25 μs, 50 μs, 100 μs, 250 μs, 500 μs, 1 ms, 10 ms, 50 ms, and 500 ms). For any new technology node, type or architecture, new measurements would need to be performed (as an analytical model is too complex and unfeasible; among other issues, one would need to accurately describe the asymmetry between each memory cell). Obviously, a wider range of values with even more granularity would present more accurate results, however, it is more time consuming. Once the measurements are taken, they are analyzed by one of the proposed algorithms, hence, determining which temperature/voltage ramp-up time is optimal.

VIII. CONCLUSION

In this paper, we proposed a method for enhancing the reproducibility of memory-based PUFs based on adapting the voltage ramp-up time to the ambient temperature. The combined effect on PUF reproducibility has been evaluated using both circuit simulation and actual silicon measurements. The results are highly effective, showing a major decrease in worst-case PUF noise (up to 3× lower for particular PUFs) at extreme temperatures. The reproducibility enhancement is achieved while either maintaining or increasing the uniqueness. Furthermore, we investigated the relation between PUF noise and area overhead both for several types of memory-based PUFs and several memory-based PUF systems constructions. Our results show that when the PUF noise is reduced, the PUF size decreases up to 3× and that the footprint of the error correction system is also slightly reduced. Finally, we implemented a small and scalable circuit that adapts the voltage ramp-up time to the sensed ambient temperature. Overall, the implementation of the proposed method will result in a PUF-based key generator significantly smaller. The proposed solution is particularly attractive for less robust memory-PUFs, such as DFF and BK, boosting their competitiveness.

REFERENCES

[1] B. Gassend, D. Clarke, M. van Dijk, and S. Devadas, "Silicon physical random functions," in *Proc. CCS*, Denver, CO, USA, 2002, pp. 148–160.

[2] R. Maes and I. Verbauwhede, "Physically unclonable functions: A study on the state of the art and future research directions," *Towards Hardware-Intrinsic Security*, A.-R. Sadeghi and D. Naccache, Eds. Berlin, Germany: Springer, 2010, pp. 3–37.

[3] S. Katzenbeisser *et al.*, "PUFs: Myth, fact or busted? A security evaluation of physically unclonable functions (PUFs) cast in silicon," in *Proc. CHES*, Leuven, Belgium, 2012, pp. 283–301.

[4] A. Maiti, J. Casarona, L. McHale, and P. Schaumont, "A large scale characterization of RO-PUF," in *Proc. HOST*, Anaheim, CA, USA, 2010, pp. 94–99.

[5] T. Yoshida, T. Katashita, and A. Satoh, "Quantitative and statistical performance evaluation of arbiter physical unclonable functions on FPGAs," in *Proc. ReConFig*, Quintana Roo, Mexico, 2010, pp. 298–303.

[6] J. Guajardo, S. S. Kumar, G.-J. Schrijen, and P. Tuyls, "FPGA intrinsic PUFs and their use for IP protection," in *Proc. CHES*, Vienna, Austria, 2007, pp. 63–80.

[7] B. Skoric, P. Tuyls, and W. Ophey, "Robust key extraction from physical unclonable functions," in *Proc. ACNS*, New York, NY, USA, 2005, pp. 99–135.

[8] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," *SIAM J. Comput.*, vol. 38, no. 1, pp. 97–139, 2008.

[9] J.-P. Linnartz and P. Tuyls, "New shielding functions to enhance privacy and prevent misuse of biometric templates," in *Proc. AVBPA*, Guildford, U.K., 2003, pp. 393–402.

[10] R. Maes, A. Van Herrewege, and I. Verbauwhede, "PUFKY: A fully functional PUF-based cryptographic key generator," in *Proc. CHES*, Leuven, Belgium, 2012, pp. 302–319.

[11] V. van der Leest, B. Preneel, and E. van der Sluis, "Soft decision error correction for compact memory-based PUFs using a single enrollment," in *Proc. CHES*, Leuven, Belgium, 2012, pp. 268–282.

[12] M. Hofer and C. Boehm, "An alternative to error correction for SRAM-like PUFs," in *Proc. CHES*, Santa Barbara, CA, USA, 2010, pp. 335–350.

[13] V. Vivekraja and L. Nazhandali, "Circuit-level techniques for reliable physically unclonable functions," in *Proc. HOST*, San Francisco, CA, USA, 2009, pp. 30–35.

[14] D. Forte and A. Srivastava, "On improving the uniqueness of silicon-based physically unclonable functions via optical proximity correction," in *Proc. DAC*, San Francisco, CA, USA, 2012, pp. 96–105.

[15] M. Bhargava, C. Cakir, and K. Mai, "Attack resistant sense amplifier based PUFs (SA-PUF) with deterministic and controllable reliability of PUF responses," in *Proc. HOST*, Anaheim, CA, USA, 2010, pp. 106–111.

[16] R. Kumar, H. K. Chandrikakutty, and S. Kundu, "On improving reliability of delay based physically unclonable functions under temperature variations," in *Proc. HOST*, San Diego, CA, USA, 2011, pp. 142–147.

[17] D. E. Holcomb, W. P. Burleson, and K. Fu, "Power-up SRAM state as an identifying fingerprint and source of true random number," *IEEE Trans. Comput.*, vol. 58, no. 9, pp. 1198–1210, Sep. 2009.

[18] M. Cortez, A. Dargar, S. Hamdioui, and G.-J. Schrijen, "Modeling SRAM start-up behavior for physical unclonable functions," in *Proc. IEEE Int. Symp. Defect Fault Toler. VLSI Nanotechnol. Syst.*, Austin, TX, USA, 2012, pp. 1–6.

[19] M. Claes, V. van der Leest, and A. Braeken, "Comparison of SRAM and FF PUF in 65nm technology," in *Proc. NordSec*, Tallinn, Estonia, 2011, pp. 47–64.

[20] S. S. Kumar, J. Guajardo, R. Maes, G.-J. Schrijen, and P. Tuyls, "The butterfly PUF protecting IP on every FPGA," in *Proc. HOST*, Anaheim, CA, USA, 2008, pp. 67–70.

[21] R. Maes, P. Tuyls, and I. Verbauwhede, "Intrinsic PUFs from flip-flops on reconfigurable devices," in *Proc. WISSec*, Eindhoven, The Netherlands, 2008, pp. 1–17.

[22] P. Simons, V. van der Leest, and E. van der Sluis, "Buskeeper PUFs, a promising alternative to D flip-flop PUFs," in *Proc. HOST*, San Francisco, CA, USA, 2012, pp. 7–12.

[23] Y. Su, J. Holleman, and B. Otis, "A 1.6pJ/bit 96% stable chip-ID generating circuit using process variations," in *ISSCC Dig. Tech. Papers*, San Francisco, CA, USA, 2007, pp. 406–611.

[24] W. Zhao *et al.*, "Rigorous extraction of process variations for 65nm CMOS design," in *Proc. ESSDERC*, Munich, Germany, 2007, pp. 89–92.

[25] Predictive Technology Model. (2012). [Online]. Available: http://ptm.asu.edu/

[26] M. Cortez, S. Hamdioui, V. van der Leest, R. Maes, and G.-J. Schrijen, "Adapting voltage ramp-up time for temperature noise reduction on memory-based PUFs," in *Proc. HOST*, Austin, TX, USA, 2013, pp. 35–40.

[27] C. Bösch, J. Guajardo, A. R. Sadeghi, J. Shokrollahi, and P. Tuyls, "Efficient helper data key extractor on FPGAs," in *Proc. CHES*, vol. 5154. Washington, DC, USA, 2008, pp. 181–197.

[28] G. D. Forney, Jr., *Concatenated Codes* (Research Monograph), vol. 37. Cambridge, MA, USA: MIT Press, 1966.

[29] Taiwan Semiconductor Manufacturing Company Limited. (Oct. 2014). *65nm Technology Overview*. [Online]. Available: http://www.tsmc.com/english/dedicatedFoundry/technology/65nm.htm

[30] Taiwan Semiconductor Manufacturing Company Limited. (Oct. 2014). *40nm Technology Overview*. [Online]. Available: http://www.tsmc.com/tsmcdotcom/PRListingNewsAction.do?action=detail&language=E&newsid=2561

[31] Europractice. (Oct. 2014). *0.13um Technology Overview*. [Online]. Available: http://www.europractice-ic.com/technologies_TSMC.php?tech_id=013um

[32] M. Cortez, G. Roelofs, S. Hamdioui, and G. Di Natale, "Testing PUF-based secure key storage circuits," in *Proc. DATE*, Dresden, Germany, 2014, pp. 1–6.

[33] AMS. *0.35µ CMOS Technology Selection Guide*. [Online]. Available: http://www.ams.com/eng/Products/Full-Service-Foundry/Process-Technology/CMOS/0.35-m-CMOS-Technology-Selection-Guide

[34] C. Böhm and M. Hofer, *Physical Unclonable Functions in Theory and Practice*. New York, NY, USA: Springer, 2013

[35] Y. Tsividis and C. McAndrew, *Operation and Modeling of the MOS Transistor*, 3rd ed. New York, NY, USA: Oxford, 2011.

[36] J. Chang, A. A. Abidi, and C. R. Viswanathan, "Flicker noise in CMOS transistors from subthreshold to strong inversion at various temperatures," *IEEE Trans. Electron Devices*, vol. 41, no. 11, pp. 1965–1971, Nov. 1994.

Mafalda Cortez (S'12) received the M.Sc. degree in electrical and computers engineering—telecommunications, electronics and computers from the Faculdade de Engenharia da Universidade do Porto, Porto, Portugal. She is currently pursuing the Ph.D. degree with the Computer Engineering Laboratory, Delft University of Technology, Delft, The Netherlands, in collaboration with Intrinsic-ID B.V., Eindhoven, The Netherlands. During the M.Sc. degree, she did her graduation thesis at NXP Semiconductors Research, Eindhoven, the Netherlands, entitled "Electrical Characterization and Interpretation of Micro-Electro-Mechanical Systems Microphones With Spring Suspended Backplates.

She was an Invited Researcher with Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier Laboratory, Montpellier, France, to research novel secure design-for-testability schemes. Her current research interests include circuit design and modeling, hardware security, and secure IC test.

Said Hamdioui (M'99–SM'11) received the M.S.E.E. and Ph.D. degrees (both with Hons.) from Delft University of Technology (TUDelft), Delft, The Netherlands.

He was with Intel, Santa Clara, CA, USA, Philips Semiconductors Research and Development, Crolles, France, and Philips/NXP Semiconductors, Nijmegen, The Netherlands. He is currently co-leading dependable-nano computing research activities with the Computer Engineering Laboratory, TUDelft. He has consulted for several semiconductor companies. His current research interests include testability and design-for-test, reliability, hardware security, and emerging computation paradigms based on memristor technology. He published one book and co-authored over 130 conference and journal papers.

Dr. Hamdioui serves on the Editorial Board of the IEEE DESIGN AND TEST and the *Journal of Electronic Testing: Theory and Applications*. He is an Associate Editor of the IEEE TRANSACTIONS ON VERY LARGE-SCALE INTEGRATION (VLSI) SYSTEMS. He is strongly involved in the international test technology community and has delivered dozens of keynote speeches, distinguished lectures, and invited presentations and tutorials at major international forums/conferences and leading semiconductor companies. He is a member of Association for European Nanoelectronics Activities/ENIAC Scientific Committee Council.

Ali Kaichouhi is currently pursuing the M.Sc. degree in electrical engineering, track of microelectronics with the Delft University of Technology (TUDelft), Delft, The Netherlands.

He was with several companies as a Hardware Design/Network Engineer. He is a Support Engineer for IC-Design and Measurement with the TUDelft, where he researches on electronic design, support Cadence IC design kit technologies, Cadence layout design, electrostatic discharge, high voltage IC design, mixed signal IC design, RF IC design, layout verification in Cadence Assura, Mentor Graphics Calibre, Cadence Skill Programming, and Verilog-AMS.

Vincent van der Leest received the master's degree in electrical engineering from Eindhoven University of Technology, Eindhoven, The Netherlands.

He is a Senior Project Leader with Intrinsic-ID B.V., Eindhoven, responsible for research and subsidy projects. He is regularly invited to teach lectures on physically unclonable functions (PUFs). His current research interests include PUFs, coding theory, and hardware security implementations. He has co-authored around 20 scientific publications in different security conferences and journals.

Roel Maes received the M.E.E. and Ph.D. degrees from Katholieke Universiteit Leuven, Leuven, Belgium, in 2007 and 2012, respectively.

He is a Hardware Security Engineer with Intrinsic-ID B.V., Eindhoven, The Netherlands, developing and integrating security architectures and solutions for innovative applications. He has co-authored over 25 papers in high-ranking security venues and has published a book on the topic of physically unclonable functions (PUFs). His current research interests include PUFs, information and coding theory, and security architectures in general.

Dr. Maes regularly performs reviews for the IEEE TRANSACTIONS ON COMPUTER AIDED DESIGN and the *Journal of Cryptographic Engineering*. He is a Recurring Program Committee Member of Cryptographic Hardware and Embedded Systems and Design, Automation & Test in Europe Conference & Exhibition.

Geert-Jan Schrijen received the master's degree in electrical engineering from the University of Twente, Enschede, The Netherlands, in 2000.

In 2001, he joined the Security Group of Philips Research, Eindhoven, The Netherlands, where he researched on digital rights management, low-power authentication protocols, private biometrics, and physical unclonable functions. He was a Senior Algorithm Designer with Intrinsic-ID B.V., Eindhoven, where he focused on the development of signal processing algorithms and security architectures for hardware-intrinsic key storage systems. In 2011, he was appointed as a VP Engineer with Intrinsic-ID B.V., where he is currently the Head of the Engineering Team that is responsible for hardware and software development.

3

TESTING SECURE DEVICES

The content of this chapter includes the following research articles:

1. ***M. Cortez**, G. Roelofs, S. Hamdioui, G. Di Natale, Testing PUF-Based Secure Key Storage Circuits, Design, Automation & Test in Europe (DATE), pp. 1-6, 24-28 March 2014, Dresden, Germany.*
2. ***M. Cortez**, G. Roelofs, S. Hamdioui, G. Di Natale, Testing Methods for PUF-Based Secure Key Storage Circuits, Journal of Electronic Testing: Theory and Applications (JETTA), pp. 581-594, volume 30, issue 5, October 2014.*
3. ***M. Cortez**, S. Hamdioui, G. Di Natale, M.-L. Flottes, B. Rouzeyre, I. Polian, Multi-segment Attack-resistant DFT for Secure ICs, to be submitted.*

3

Testing PUF-Based Secure Key Storage Circuits

Mafalda Cortez Gijs Roelofs Said Hamdioui
Delft University of Technology
Faculty of EE, Mathematics and CS
Mekelweg 4, 2628 CD Delft, The Netherlands
{A.M.M.O.Cortez, S.Hamdioui}@tudelft.nl

Giorgio di Natale
LIRMM, Universit Montpellier II
161 Rue Ada, 34392 Montpellier Cedex 5, France
diNatale@lirmm.fr

***Abstract*—Design for test is an integral part of any VLSI chip. However, for secure systems extra precautions have to be taken to prevent that the test circuitry could reveal secret information. This paper addresses secure test for Physical Unclonable Function based systems. In particular it provides the testability analysis and a secure Built-In Self-Test (BIST) solution for Fuzzy Extractor (FE) which is the main component of PUF-based systems. The scheme targets high stuck-at-fault (SAF) coverage by performing scan-chain free functional testing, to prevent scan-chain abuse for attacks. The scheme reuses existing FE sub-blocks (for pattern generation and compression) to minimize the area overhead. The scheme is integrated in FE design and simulated; the results show that a SAF fault coverage of 95.1% can be realized with no more than 50k clock cycles at the cost of a negligible area overhead of only 2.2%. Higher fault coverage is possible to realize at extra cost.**

Keywords: PUF-based systems, Fuzzy Extractor, Secure Testing, Scan-chain free test

I. INTRODUCTION

Physical Unclonable Functions (PUFs) based systems are becoming popular solutions for secure key storage against physical attacks; they use the unique, random, uncontrollable and intrinsic physical properties of *Integrated Circuits* (ICs) to derive a cryptographic key. The robustness of such a system is evaluated by means of its reproducibility, i.e., ability of the system to recover the cryptographic key from the same IC, and its uniqueness; i.e., the ability of the system to generate a unique cryptographic key for each IC [1,2]. A *Fuzzy Extractor* (FE) is one of the main components of a PUF-based system; its responsibility is to assure the system's reproducibility and uniqueness [3,4]. Hence, FE flawless operation is essential for the robustness of PUF-based systems. Testing a PUF-based system, and FE in particular, is a challenge. Testability demands excellent accessibility and observability, while security demands poor/no accessibility and observability to the chip, especially during the operation mode where an attacker could easily retrieve partial or complete cryptographic key. The trade-off between testability and security is the main challenge.

Design-for-Test and testability of secure devices have recently gained a lot of attention [5–11]. Overall, the published schemes can be classified into two classes: enhanced scan-chains [5–8] and functional based *Built-In Self Test* (BIST) [10,11].

Enhanced scan-chains target the protection of chains from being misused by attackers. In [5], B. Yang *et al.* developed a test solution for crypto cores based on a type of register that cannot be scanned out during test mode until being reset. In [6], A. Das *et al.* developed a test wrapper for secure test that authenticates legitimate testers. In [7], D. Hely *et al.* introduced spy flip-flops in the scan-chain that detect malicious shifts. In [8], J. Lee *et al.* applied a technique that makes the scan-chain operate unpredictably for untrusted users. However, the industry strongly believes that enhanced scan-chains cannot provide 100% secure IC and therefore they are reluctant to include them in designs targeting secure applications [9].

On the other hand, functional test based BIST targets the enhancement of security, although reaching a very high fault coverage with these schemes is a major challenge. In [10], M. Doulcier *et al.* presented a technique to reuse an *Advanced Encryption Standard* (AES) for self-testing. The work showed that AES cores have enough randomness to be used as test pattern generators and used this property to seft-test the AES core in a loop fashion. In [11], di Natale *et al.* proposed a generic self-test scheme for crypto cores. The work is an extension of the work presented in [10]; it performs the same analysis but for a *Data Encryption Standard* (DES). However, both [10] and [11] are not suited for testing PUF-based systems for two main reasons. First, AES/DES crypto cores are not available in all PUF-based systems and second, PUF-based systems comprise, on top of the crypto cores, error correction blocks which make it more challenging to test functionally.

Although the research in hardware security including test is getting more attention due to the importance of the field, there is almost nothing published on testing PUF-based systems. This topic is addressed in this paper. In particular, it targets testing and testability analysis of Fuzzy Extractors (FEs), which are the main blocks of such systems; FEs are challenging to test as they comprise not only a crypto core, but also error correction blocks, which are typically hard to test functionally. The paper proposes an efficient *scan-chains free* secure test scheme that realizes a high test quality based on pattern generation for stuck-at-faults by performing functional testing. The proposed solution reuses FE existing sub-blocks (for pattern generation and compression) to minimize the area overhead.

The rest of the paper is organized as follows: Section II

978-3-9815370-2-4/DATE14/©2014 EDAA

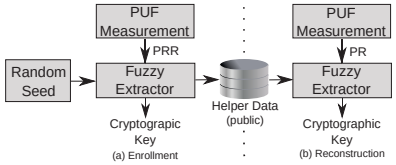


Fig. 1: PUF-based Key Storage System

briefly reviews the background on PUF based-systems. Section III analyzes the testability of FE sub-blocks. Section IV defines the test requirements, proposes a secure test method, presents and discusses the results and provides a generic list of step-by-step instructions on FE secure testing. Finally, Section V concludes the paper.

II. PUF-based Secure Systems

In this section, we first briefly show how PUFs are deployed in a key storage system. Thereafter, we briefly describe the Fuzzy Extractor, which is the main block of such a system and the primary focus of this work.

A. Key-storage based on PUFs

Fig. 1 shows the flow of a PUF-based key storage system [1,2] implemented with a *Fuzzy Extractor* (FE) [3,4], which typically consists of two phases:

(a) **Enrollment:** a cryptographic key is generated from a PUF. First, a PUF measurement is taken and used as *PUF Reference Response* (PRR). Next, PRR and *Random Seed* are processed by the FE into a cryptographically strong *Cryptographic Key*, and helper data is generated as a FE byproduct. Finally, the helper data is stored in an external *Non-Volatile Memory* (NVM); hence, it becomes public information.

(b) **Reconstruction:** the earlier enrolled *Cryptographic Key* is reliably recovered. First, a PUF measurement is taken and used as *PUF Response* (PR). Typically, some bits of PR are different from the original PRR; hence, PR is a noisy version of PRR. Next, PR is processed by the FE in combination with the helper data which is retrieved from the external NVM. If the noisy PR is close enough to the PRR measured during enrollment (i.e., the PUF response is reproducible up to a limited amount of noise), then the FE succeeds in reliably reconstructing the enrolled *Cryptographic Key*.

B. Fuzzy Extractor

A Fuzzy Extractor (FE) is the fundamental component of a PUF-based key storage system; it has two main functions. (a) Error correction: it uses the helper data to correct errors on the measured PUF response; and (b) Privacy amplification: considering that the helper data contains information on the PRR, privacy amplification is needed to make sure that the helper data does not reveal any information on the derived cryptographic key; the FE compresses the resulting data into a cryptographic key with maximum entropy making it hard for the attacker to retrieve the key [3,4]; it also removes any biasing (unequal distribution of zeros and ones) in the error-corrected PUF response.

III. Fuzzy Extractor and its testability analysis

Fig. 3 shows the six main blocks of a Fuzzy Extractor; this implementation is based on the one used for the UNIQUE project [17]. The Peripheral Circuitry has two main functions; it selects between both functional modes and it performs an XOR function between either the PUF and the output of the Repetition Encoder (*RE_O*) generating the Helper Data, if during enrollment or, between the stored Helper Data and PUF generating the input of the Repetition Decoder (*RD_I*), if during reconstruction. The other five blocks are mostly computation intensive and are responsible for enrollment and reconstruction. Each of the five blocks is explained next.

1) Golay Encoder: first block of the enrollment phase. Its responsibility is to prepare the data for the error correction. This block maps the input *Random Seed* (12 bits) to *GE_O* (24 bits) by appending twelve parity bits used for error correction. This makes it feasible for the Golay Decoder in Reconstruction phase to correct up to three bits [14,15]. The main core of this block comprises a loop that generates the Golay space (space of perfect code words). Our implementation of the Golay Encoder has a latency of two clock cycles and it comprises 6.5% of the total number of FE gates.

2) Repetition Encoder: second block of the enrollment phase. It adds extra robustness to the error capabilities of the Golay Encoder. The block replicates each of the *GE_O* 24 bits 11 times resulting in 264 bits serial output *RE_O*; it enables error correction up to five bits for the Repetition Decoder in reconstruction phase. The enrollment phase completes after 86 rounds, i.e., the computations described above are performed 86 times. At each time, the 12 bits of the 1032 bit *Random Seed* are used with a 264 bits fraction of the PUF to generate Helper Data. The total size of both PUF and Helper Data equals 2.8kB (264×86). The main part of the block comprises two counters. The first counter loops over the 24 bits of *GE_O*, while the second counter replicates each bit of *GE_O* 11 times. Our implementation of the Repetition Encoder has a latency of 267 clock cycles and it comprises 7.3% of FE gates.

The reconstruction phase starts with performing a new measurement of the PUF and XORing it with the Helper Data. The result of this operation is the serial input of the Repetition Decoder block.

3) Repetition Decoder: first block of the reconstruction phase. It is also the first stage of error correction. The block performs majority voting on each of the 24 groups (each of 11 bits) scanned serially via *RD_I*, and produce 24 bits at the output *GD_I*. This sub-block performs the inverse operation of the Repetition Encoder and its main core comprises three counters:

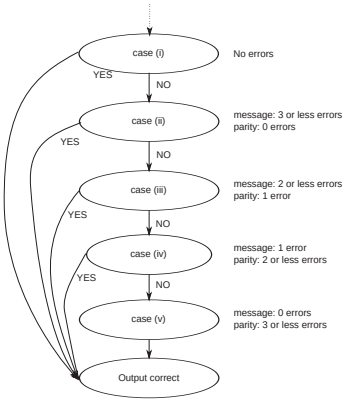


Fig. 2: Golay Decoder state-machine

one counter, *repetition counter* and *destination counter*. The one counter counts the number of ones in a chunk of input *RD_I* (see Fig. 3) and its value is reset after the repetition counter processed n=11 input bits. Next, a single output bit is written on the index provided by the destination counter which is subsequently incremented. The written output bit presents the majority voting result of the processed input chunk derived from the one counter. Our implementation of the Repetition Decoder has a latency of 290 clock cycles and comprises 6.5% of FE gates.

4) Golay Decoder: second block of the reconstruction phase responsible for error correction. The block recovers *Random Seed*, i.e., *HF_I* (12 bits), as long as the provided input *GD_I* is within the error capabilities of the error correction system. Also during the reconstruction phase, the Repetition Decoder and the Golay Decoder repeat their operations 86 times; each time, they serially process 264 bits generated based on PUF and Helper Data. The results of each iteration is a 12 bits buffered inside the Hash Function block. The Golay Decoder is the most complex block in the circuit. It contains a *Finite State-Machine* (FSM), with nine states for vector decoding. As stated previously, a Golay Decoder can correct up to three errors. Its input *GD_I* comprises 12 message bits and 12 parity bits (as the outcome of the Golay Encoder). Fig. 2 shows the states dedicated to error correction; these are selected depending on the location and number of errors in *GD_I*. Error wise, five different cases are possible, denoted in Fig. 2 as case (i) till (v).

(i) *GD_I* is error-free; thus, the four states where the error correction takes place, i.e., case (ii) to case (v), are skipped.

(ii) there are three or less errors in the message bits of *GD_I* and none in the parity bits.

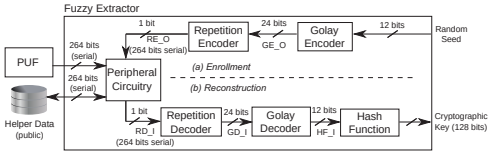


Fig. 3: Example of a Fuzzy Extractor

(iii) there are one or two errors in the message bits of *GD_I* and exactly one in the parity bits.

(iv) there is exactly one error in the message bits of *GD_I* and two or less in the parity bits.

(v) there are no errors in the message bits of *GD_I* and three or less in the parity bits.

The Golay Decoder has a variable latency depending on its input, with a maximum of 10 clock cycles and it comprises 61.5% of FE gates.

5) Hash Function: last block of the reconstruction phase. It performs privacy amplification. This block concatenates the 1032 bits (12×86 iterations) received from the Golay Decoder and applies the hash function on it to calculate the 128 bit Cryptographic Key.

Our Hash Function comprises three main components: an input buffer, a *Linear Feedback Shift Register* (LFSR) and an accumulator register. First, the input *HF_I* is copied to the input buffer. The input buffer is then analyzed bit per bit: if a bit is one, the current LFSR output (which updates itself each cycle based on its polynomial function) is added (XORed) with the accumulator; however, if the bit is zero, the accumulator keeps its value. When all input bits are analyzed the value of the accumulator register is propagated to the output. The Hash Function has a latency of 32 clock cycles and it comprises 18.2% of FE gates.

It is worth noting that the Fuzzy Extractor presented here is a *generic* and simplified construction of an industrial implementation [17]; therefore, any test method developed for this circuit can be applied also to any other implementation.

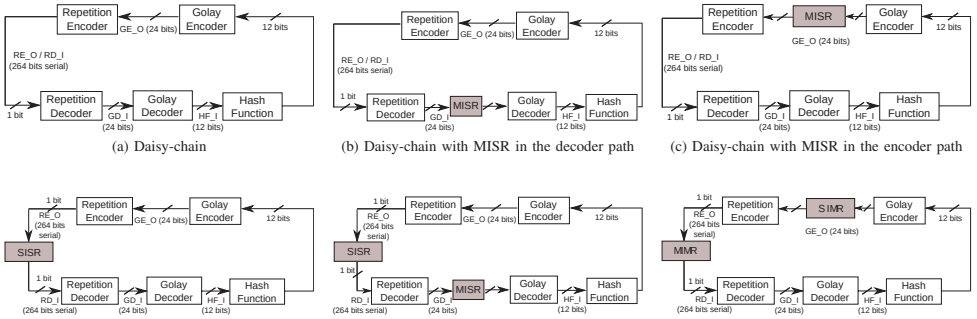
IV. EXPERIMENTS RESULTS

In this section, first, we define test and security requirements considered for the development of our test solution. Then, we present our test method and the experiments. Thereafter, we present the results. Finally, we provide a list of recommendations for secure testing of FE.

A. Test versus Security Requirements

Efficient test solutions for FE must prevent compromising the system security. The following requirements and assumptions apply:

(a) The signals *PUF*, *Random Seed* and *HF_I* (see Fig. 3) shall not be revealed at any time, partially nor fully.



(a) Daisy-chain

(b) Daisy-chain with MISR in the decoder path

(c) Daisy-chain with MISR in the encoder path

(d) Daisy-chain with SISR

(e) Daisy-chain with SISR and MISR in the decoder path

(f) Daisy-chain with SISR and MISR in the encoder path

Fig. 4: Daisy-chain for a FE - different constructions

An attacker learning this information might derive the *Cryptographic Key*, breaking the systems security.

(b) Helper data is assumed to be public knowledge and does not have to be secured.

(c) Reverse engineering the Fuzzy Extractor is not an issue. The Fuzzy Extractor uses algorithms that are standardized and publicly known.

(d) The PUF circuitry has its own internal test method, therefore it is outside the scope of this work.

B. Daisy-chain secure test method

Next, we propose and discuss a test method for FE, a daisy-chain based test method. The method is scan-chain free, which is a requirement in our case.

In this method, we propose to reuse the *Linear Feedback Shift Register* (LFSR) of the Hash Function block to create a random generator and test the FE in a loop-chain fashion, i.e., the outputs of each block are directly provided as inputs to next block as depicted in Fig. 4(a). This approach results in a negligible area overhead. However, a high fault coverage for the Golay Decoder cannot be guaranteed. This is because the Golay Decoder receives *always* error free input messages as provided by the Golay Encoder, which prevents the correct checking of all the decoder' states. Hence, reusing LFSR of hash function with daisy-chain approach *alone* will not provide the required test quality for Golay Decoder. To solve this problem, the randomness of the patterns provided at the Golay Decoder inputs (generated by the Golay Encoder) have to be improved, in order to trigger all states of the Golay Decoder FSM. This can be done by inserting a *Multiple-Input-Shift-Register* (MISR) at the input of the Golay Decoder as seen in Fig. 4(b). However, as the blocks are connected in a loop, the desired effect of randomness improvement can also be achieved by placing a MISR in any location between the Golay Encoder output and the Golay Decoder input (such as at the output of Golay Encoder in Fig. 4(c)), or a *Single-Input-Shift-Register* (SISR) in case the location is just a serial line; see Fig. 4(d). Moreover, a combination of MISR and SISR can be also used as shown in Fig. 4(e) and Fig. 4(f).

Comparing the cost (area overhead) and randomness of the several scenarios presented in Fig. 4 reveals that:

1) Scenario (d) results in the smallest area overhead.
2) Scenarios (b) and (c) as well as (e) and (f) are equivalent, reducing the number of scenarios to four.
3) Scenarios (e) and (f) could lead to higher fault coverage, as the combination of using SISR and MISR could improve the randomness.

C. Experiments performed

We synthesize a Fuzzy Extractor described in VHDL in 0.35μm technology node with Synopsys Design Compiler. The design compiler outputs a verilog netlist that is used to extract a fault list with the *Automated Test Pattern Generation* (ATPG) tool Synopsys TetraMAX. We use LIFTING fault simulator optimized for functional BIST to analyze the fault coverage [25]. The results are analyzed with MATLAB.

To evaluate the quality of the proposed solutions in Fig. 4 in terms of fault coverage and test time, we perform the following experiments, as described next.

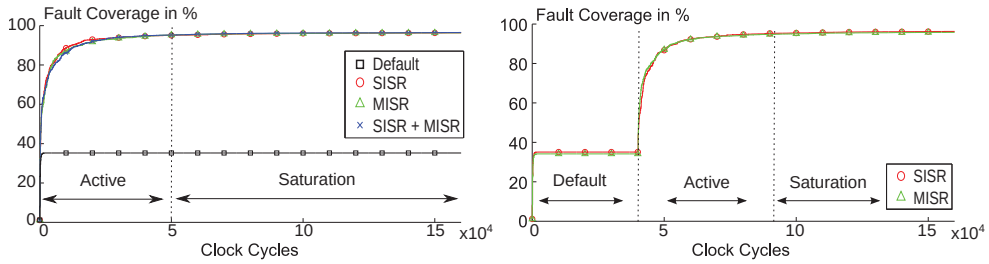
(i) **Default:** in this experiment, we simulate the circuit as in Fig. 4(a) for 15×10^4 clock cycles and analyze the fault coverage. This number of clock cycles is assumed to be our test time budget for all remaining experiments.

(ii) **MISR:** in this experiment, we simulate the circuit as in Fig. 4(b) (equivalent to Fig. 4(c)).

(iii) **SISR:** in this experiment, we simulate the circuit as in Fig. 4(d).

(iv) **SISR + MISR:** in this experiment, we simulate the circuit combining SISR and MISR as in Fig. 4(e) (equivalent to Fig. 4(f)).

(v) **Default + SISR:** in this experiment, we simulate the circuit in two stages. First, as in Fig. 4(a), we simulate



(a) Without default stage

(b) With default stage

Fig. 5: Fault coverage of Fuzzy Extractor Daisy-chain test method versus clock cycles

the FE using the default loop-chain for 25% of the test time budget. Second, as in Fig. 4(d), we include the SISR on the chain flow (between the Repetition Encoder and Repetition Decoder blocks) and analyze its fault coverage over the remaining 75% of the time. The goal of this experiment is to analyze the impact of *combining* the default scenario in Fig. 4(a) with the SISR in Fig. 4(d) on the fault coverage.

(vi) **Default + MISR:** in this experiment, we repeat the procedure of (v), but replacing the SISR with a MISR.

D. Results

Fig. 5 shows the fault coverage (y-axis) versus number of clock cycles (x-axis) of the experiments. Part (a) of the figure shows four first experiments, while part (b) shows the remainder experiments that include the default stage.

From Fig. 5 (a) we can observe the following.

1) During the default stage the Fuzzy Extractor realizes a fault coverage of only 36%. This fault coverage is realized quickly in the first 2k clock cycles. The figure clearly shows that the fault coverage remains stable at 36% during the remaining clock cycles.
2) For the remaining schemes, after 5×10^4 clock cycles the fault coverage reaches 95.1%. The remaining 10×10^4 clock cycles lead to an increment of only 1.2% (from 95.1% up to 96.2%) fault coverage.
3) There is no major difference in fault coverage between SISR, MISR and their combination. All the techniques realize the same fault coverage in similar test time, i.e., 95.1% is achieved at 5×10^4 (50k) clock cycles.

From Fig. 5 (b) we can observe the following.

1) Switching to SISR/MISR after the default stage strongly increases the fault coverage, i.e, the fault coverage increases from 36% up to 95.1% in 6.4×10^4 clock cycles.
2) After switching to SISR/MISR, it takes 5.8×10^4 clock cycles to reach a fault coverage of 95.1%. Extending the test time to 15×10^4 increases the fault coverage to 96.3%.
3) The impact of combining the default stage with either a SISR or MISR on the fault coverage is negligible.

The area overhead is measured in 0.35μm technology with the following results: 2.2% for SISR, 6.80% for MISR and 8.0% for SISR and MISR combined.

Analyzing the results, we can see that (i) it is critical to randomize the output of the Golay Encoder block. (ii) the final obtained fault coverage in both figures is similar. From this we conclude that the default stage is superfluous. (iii) in terms of fault coverage and test time, there is no difference between a SISR, MISR and a combination of both. (iv) in addition, if we consider the area overhead, the most efficient solution is to use the SISR only.

The fault coverage of our method is in line with the fault coverage reported in other self-test methods [10] and [11]. However, due to the nature of the extra FE components were required (such as SISR/MISR) to increase the fault coverage. The area overhead of the proposed method is negligible, which is intrinsic to methods that reuse hardware.

E. Recommendations

We provide a generic step-by-step procedure for secure testing of FE based on our findings. These steps are:

- The first step is to identify each block and to deeply understand its functionality (which is critical for successful functional testing).
- Second, the characteristics of each block need to be assessed (e.g., its area overhead, if the block comprises a state-machine or not, state-machine complexity, etc).
- Third, perform testability analysis of each block by considering a random input source. Identify eventual challenges by testing a certain block with this method.
- Fourth, identify if there is need of extra components in order to increase the fault coverage, such as a SISR. If so, analyze the trade-off of such components in terms of its possible locations and of its impact on security, fault coverage, test time and area overhead.
- Fifth, optimize the test solution; e.g., by using a SISR or MISR to activate uncovered paths in a state-machine.
- Sixth, analyzed the FC test time and area overhead of

the test method separately and when combined with complementary schemes.
- Finally, determine and select the best test method combined with complementary schemes (such as SISR/MISR) to meet the design requirements.

Following these steps will enable a secure test scheme with low area overhead and high test quality for any construction of Fuzzy Extractor.

V. Conclusion

In this paper we demonstrated a secure test method for a Fuzzy Extractor, scan-chain free to make sure that secret information stays inside the module and cannot be read out. The secure test method is based on daisy-chains; it reuses Fuzzy Extractor blocks for test pattern generation and output compression. The results show that the method has an inherent low area overhead 2.2%, while it realizes a fault coverage of 95.1% using only 50k clock cycles. Finally, we provided a generic step-by-step procedure to test any given Fuzzy Extractor based on our findings.

Acknowledgments

The authors would like to thank Peter Simons and Geert-Jan Schrijen from Intrinsic-ID B.V. for all the useful discussions.

References

[1] J. Guajardo *et al.* , "FPGA Intrinsic PUFs and Their Use for IP Protection", *Workshop on Cryptographic Hardware and Embedded Systems (CHES)*, pp.63-80, Sept. 2007.
[2] B. Skoric, P. Tuyls, and W. Ophey, "Robust key extraction from Physical Unclonable Functions", *Applied Cryptography and Network Security*, vol.3531 of LNCS, pg.99135, Springer Berlin / Heidelberg, 2005.
[3] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data", *vol. 3027*, LNCS, 2004.
[4] J.-P. Linnartz and P. Tuyls, "New shielding functions to enhance privacy and prevent misuse of biometric templates", *Proc. of AVBPA03*, pp.393-402, Springer Berlin / Heidelberg, 2003.
[5] B. Yang, K. Wu and R. Karri, "Secure Scan: A Design-for-Test Architecture for Crypto Chips", *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol.25, no.10, pp.2287-2293, Oct. 2006.
[6] A. Das and Ünal Kocabaş and Ahmad-Reza Sadeghi and Ingrid Verbauwhede, "PUF-based Secure Test Wrapper Design for Cryptographic SoC Testing", Design, Automation and Test in Europe, 2012
[7] D. Hely, F. Bancel, M.-L. Flottes and B. Rouzeyre, "A secure Scan Design Methodology", Design, Automation and Test in Europe, 2006.
[8] J. Lee, M. Tehranipoor, C. Patel and J. Plusquellic, "Securing Scan Design Using Lock and Key Technique", *IEEE Int. Symp. on Defect and Fault Tolerance in VLSI System*, 2005
[9] J. da Rolt, G di Natale, M.-L. Flottes and B. Rouzeyre, "New security threats against chips containing scan chain structure", *IEEE Int. Symp. on Hardware-Oriented Security and Trust*, 2012
[10] M. Doulcier, M.L. Flottes and B. Rouzeyre, "AES-based BIST: self-test, test pattern generation and signature analysis", *IEEE International Symposium on Electronic Design, Test & Applications*, 2008
[11] G. di Natale, M. Doulcier, M.-L. Flottes and B. Rouzeyre "Self-Test Techniques for Crypto-Devices", *IEEE Trans. on VLSI Systems*, vol. 18, no. 2, Feb. 2010
[12] D.K. Pradhan and M. Chatterjee, "GLFSR-a new test pattern generator for built-in-self-test", *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol.18, no.2, pp.238-247, Feb. 1999.
[13] S. Boubezari and B. Kaminska, "A deterministic built-in self-test generator based on cellular automata structures", *IEEE Trans. on Computers*, vol.44, no.6, pp.805-816, Jun. 1995.
[14] "http://mathworld.wolfram.com/GolayCode.html"
[15] V. Pless, "Decoding the golay codes", *IEEE Trans. on Information Theory*, vol.32, no.4, pp.561-567, July 1986.
[16] C.V. Krishna, A. Jas and N.A. Touba, "Test vector encoding using partial LFSR reseeding", Int. Test Conference, pp.885-893, 2001.
[17] "www.unique-project.eu"
[18] A.A. Al-Yamani and E.J. McCluskey, "Built-in reseeding for serial BIST", *VLSI Test Symposium*, pp.63-68, 2003.
[19] S. Hellebrand, *et al.* , "Built-in test for circuits with scan based on reseeding of multiple-polynomial linear feedback shift registers" *IEEE Trans. on Computers*, vol.44, no.2, pp.223-233, Feb. 1995.
[20] L. Lei and K. Chakrabarty, "Test set embedding for deterministic BIST using a reconfigurable interconnection network", *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol.23, no.9, pp.1289-1305, Sept. 2004.
[21] N.A. Touba and E.J. McCluskey, "Bit-fixing in pseudorandom sequences for scan BIST", *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, vol.20, no.4, pp.545-555, Apr. 2001.
[22] H.-J. Wunderlich and G. Kiefer, "Bit-flipping BIST", IEEE/ACM Int. Conf. on Computer-Aided Design, pp.337-343, 1996
[23] M. Arai *et al.* , "Seed selection procedure for LFSR-based BIST with multiple scan chains and phase shifters", *Asian Test Symposium*, 2004
[24] M. Bellos, D. Kagaris and D. Nikolos, "Test Set Embedding Based on Phase Shifters", *European Dependable Computing Conf.*, 2002.
[25] "http://www.lirmm.fr/"

J Electron Test (2014) 30:581–594
DOI 10.1007/s10836-014-5471-7

Testing Methods for PUF-Based Secure Key Storage Circuits

3

Mafalda Cortez · Gijs Roelofs · Said Hamdioui · Giorgio Di Natale

Received: 30 April 2014 / Accepted: 21 August 2014 / Published online: 17 September 2014

Abstract Design for test is an integral part of any VLSI chip. However, for secure systems extra precautions have to be taken to prevent that the test circuitry could reveal secret information. This paper addresses secure test for Physical Unclonable Function based systems. It investigates two secure Built-In Self-Test (BIST) solutions for Fuzzy Extractor (FE) which is the main component of PUF-based systems. The schemes target high stuck-at-fault (SAF) coverage by performing scan-chain free functional testing, to prevent scan-chain abuse for attacks. The first scheme reuses existing FE blocks (for pattern generation and compression) to minimize the area overhead, while the second scheme tests all the FE blocks simultaneously to minimize the test time. The schemes are integrated in FE design and simulated; the results show that for the first test scheme, a SAF fault coverage of 95 % can be realized with no more than 47.1k clock cycles at the cost of a negligible area overhead of only 2.2 %; while for the second test scheme a SAF fault coverage of 95 % can be realized with 3.5k clock cycles at the cost of 18.6 % area overhead. Higher fault coverages are possible to realize at extra cost (i.e., either by extending the test time, or by adding extra hardware, or a combination of both).

Responsible Editor: M. Tehranipoor

M. Cortez (✉) · G. Roelofs · S. Hamdioui
Faculty of EE, Mathematics and CS, Delft University of Technology, Mekelweg 4, 2628 CD Delft, The Netherlands
e-mail: mafalda.m.cortez@gmail.com

G. Di Natale
Giorgio Di Natale LIRMM, Université Montpellier II, 161 Rue Ada, 34392 Montpellier Cedex 5, France
e-mail: DiNatale@lirmm.fr

Keywords PUF-based systems · Fuzzy Extractor · Secure testing · Scan-chain free testing

1 Introduction

Physical Unclonable Functions (PUFs) based systems are becoming popular solutions for secure key storage against physical attacks [12, 13, 25]; they use the unique, random, uncontrollable and intrinsic physical properties of *Integrated Circuits* (ICs) to derive a cryptographic key. The robustness of a such system is evaluated by means of its *reproducibility* (i.e., ability of the system to recover the cryptographic key from the same IC) and its *uniqueness* (i.e., the ability of the system to generate a unique cryptographic key for each IC) [12, 25]. A *Fuzzy Extractor* (FE) is one of the main components of a PUF-based system; its responsibility is to assure the system's reproducibility and uniqueness [10, 21]. Hence, FE flawless operation is critical for the robustness of PUF-based systems. Testing a PUF-based system, and FE in particular, is a challenge. Testability demands excellent accessibility and observability, while security demands poor/no accessibility and observability to the chip, especially during the operation mode where an attacker could easily retrieve partial or complete cryptographic key. The trade-off between testability and security is a major challenge.

Design-for-Test and testability of secure devices have recently gained a lot of attention [8, 9, 11, 15, 19, 29]. Overall, the published schemes can be classified into two classes: enhanced scan-chains [6, 8, 15, 19, 29] and functional based *Built-In Self Test* (BIST) [9, 11].

Enhanced scan-chains target the protection of chains from being misused by attackers. In [29], B. Yang et al. developed a test solution for crypto cores based on a type

of register that cannot be scanned out during test mode until being reset. In [8], A. Das et al. developed a test wrapper for secure test that authenticates legitimate testers. In [15], D. Hely et al. introduced spy flip-flops in the scan-chain that detect malicious shifts. In [19], J. Lee et al. applied a technique that makes the scan-chain operate unpredictably for untrusted users. In [6], J. Da Rolt et al. designed a smart test controller that automatically discerns scan shift operations and blocks any scan-out leakage. However, industry strongly believes that enhanced scan-chains cannot provide 100 % secure IC, as many researches have showed that scan-based DFTs can be hacked [2, 4, 7, 23]. Therefore, industry is reluctant to include them in designs targeting secure applications.

On the other hand, functional test based BIST targets the enhancement of security, although reaching a very high fault coverage with these schemes is a major challenge. In [11], M. Doulcier et al. presented a technique to reuse an *Advanced Encryption Standard* (AES) for self-testing; the work shows that AES cores have enough randomness to be used as test pattern generators and built on this property to seft-test the AES core in a loop fashion. In [9], Di Natale et al. proposed a generic self-test scheme for crypto cores; the work is an extension of the work presented in [11]. It performs the same analysis but for *Data Encryption Standard* (DES). However, both [11] and [9] are not suited for testing PUF-based systems for two main reasons. First, AES/DES crypto cores are not available in all PUF-based systems and second, PUF-based systems comprise, on top of the crypto cores, error correction blocks which make it more challenging to test functionally.

Although the research in hardware security including test is getting more attention due to the importance of the field, there is almost nothing published on testing PUF-based systems. This topic is addressed in this paper. In particular, this work targets testing of FEs, which are the main blocks of such systems; FEs are challenging to test as they comprise not only a crypto core, but also error correction blocks, being typically hard to test functionally.

This paper is an extension of our previous work presented in [5]; it proposes two efficient *scan-chains free* secure test schemes that realize high test quality based on pattern generation for stuck-at-faults using functional testing. The first proposed solution reuses FE existing blocks (for pattern generation and compression) to minimize the area overhead [5], while the second solution tests all comprising FE blocks simultaneously to minimize the test time. In addition, optimization techniques to even further reduce the test time and increase fault coverage are proposed. In addition to the main contribution of [5], i.e.,

- a low area overhead secure test method with its inherent concept, methodology, results and discussion,

this paper has the following contributions

- fast and secure test method with its inherent concept, methodology, results and discussion;
- in depth discussion of the results, including comparison between secure test methods, comparison with state-of-the-art, security analysis and list of recommendations on how to securely test FE;
- and classification of methods to improve test quality and implementation of one of these methods.

The rest of the paper is organized as follows. Section 2 briefly reviews the background on PUF based-systems and analyzes FE in detail. Section 3 defines the test requirements, proposes the two secure test methods and gives means to further improve the quality of proposed methods. Section 4 defines the experiments and presents the results. Section 5 discusses them, compares both test methods, analysis the methods security and provides a generic list of step-by-step instructions to securely test FE. Finally, Section 6 concludes the paper.

2 PUF-Based Secure Systems

We first briefly show how PUFs are deployed in a key storage system. Thereafter, we describe the Fuzzy Extractor in detail; it is the main block of such a system and the primary focus of this work.

2.1 Key-Storage based on PUFs

Figure 1 shows the flow of a PUF-based key-storage system [12, 25] implemented with a *Fuzzy Extractor* (FE) [10, 21], which typically consists of two phases:

(a) **Enrollment** : a cryptographic key is generated from a PUF. First, a PUF measurement is taken and used as *PUF Reference Response* (PRR). Next, PRR and *Random Seed* (provided externally) are processed by the FE into a cryptographically strong *Cryptographic Key*, and helper data is generated as an FE byproduct.

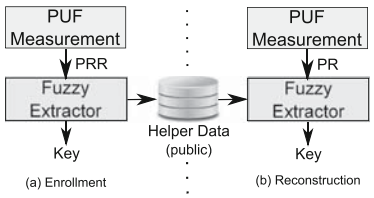


Fig. 1 PUF based Key Storage System

J Electron Test (2014) 30:581–594 583

Finally, the helper data is stored in an external *Non-Volatile Memory* (NVM); hence, it becomes public information.

(b) **Reconstruction** : the earlier enrolled *Cryptographic Key* is reliably recovered. First, a PUF measurement is taken and used as *PUF Response* (PR). Typically, some bits of PR are different from the original PRR; hence, PR is a noisy version of PRR. Next, PR is processed by the FE in combination with the helper data which is retrieved from the external NVM. If the noisy PR is close enough to the PRR measured during enrollment (i.e., the PUF response is reproducible up to a limited amount of noise), then the FE succeeds to reliably reconstruct the enrolled *Cryptographic Key*.

2.2 Fuzzy Extractor

A Fuzzy Extractor (FE) is the fundamental component of a PUF-based key storage system; it has two main functions. (a) Error correction: it uses the helper data combined with error correction to correct errors in the measured PUF response; and (b) Privacy amplification: considering that the helper data contains information on the PRR, privacy amplification is needed to make sure that the helper data does not reveal any information on the derived cryptographic key; the FE compresses the resulting data into a cryptographic key with maximum entropy making it hard for the attacker to retrieve the key [10, 21]. It also removes any biasing (unequal distribution of zeros and ones) in the error-corrected PUF response. Privacy amplification is realized with Hash Function.

Figure 2 shows the six main blocks of a Fuzzy Extractor; this implementation is based on the one used for the UNIQUE project [27]. The Peripheral Circuitry has two main functions: (a) it selects between both functional modes (enrollment versus reconstruction), and (b) it performs XOR function either between the PUF response and the output of the Repetition Encoder (*RE_O*) to generate the Helper Data during the enrollment or, between the stored Helper Data and PUF response to generate the input of the Repetition Decoder (*RD_I*) during the reconstruction. The other five blocks are mostly computation intensive and are responsible for the enrollment and the reconstruction. Each of the five blocks is explained next.

1) *Golay Encoder:* first block of the enrollment phase. Its responsibility is to prepare the data for the error correction. This block maps the input *Random Seed* (12 bits per iteration × 86 iterations) to *GE_O* (24 bits per iteration) by appending twelve parity bits used for error correction. This makes it feasible for the Golay Decoder in Reconstruction phase to correct up to three bits [16, 24]. The main core of this block comprises a loop that generates the Golay space (space of perfect code words). Our implementation of the Golay Encoder has a latency of two clock cycles and it comprises 6.5 % of the total number of FE gates.

Repetition Encoder second block of the enrollment phase. It adds extra robustness to the error capabilities of the Golay Encoder. The block replicates each of the *GE_O* 24 bits 11 times resulting in 264 bits serial output *RE_O*; this enables the error correction up to five bits for the Repetition Decoder in the reconstruction phase. The enrollment phase completes after 86 iterations, i.e., the computations described above are performed 86 times. At each time, the 12 bits fraction of the 1032 bit *Random Seed* are processed with a 264 bits fraction of the PUF response to generate Helper Data. The total size of both PUF and Helper Data are each 2.8kB (264×86). The main part of the block comprises two counters. The first counter loops over the 24 bits of *GE_O*, while the second counter replicates each bit of *GE_O* 11 times. Our implementation of the Repetition Encoder has a latency of 267 clock cycles and it comprises 7.3 % of FE gates.

Repetition Decoder first block of the reconstruction phase and also the first stage of error correction. The

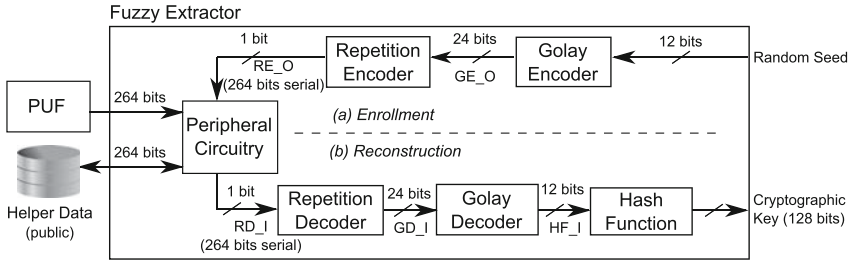


Fig. 2 Example of a Fuzzy Extractor

reconstruction phase starts with performing a new measurement of the PUF and XORing it with the Helper Data. The result of this operation is the serial input of the Repetition Decoder block. The block performs majority voting on each of the 24 groups (each of 11 bits) scanned serially via *RD_I*, and produce 24 bits at the output *GD_I*. This block performs the inverse operation of the Repetition Encoder and its main core comprises three counters: *one counter*, *repetition counter* and *destination counter*. The one counter counts the number of ones in a chunk of input *RD_I* (see Fig. 2) and its value is reset after the repetition counter processed n=11 input bits. Next, a single output bit is written on the index provided by the destination counter which is subsequently incremented. The written output bit presents the majority voting result of the processed input chunk derived from the one counter. Our implementation of the Repetition Decoder has a latency of 290 clock cycles and comprises 6.5 % of FE gates.

Golay Decoder second block of the reconstruction phase responsible for error correction. The block recovers *Random Seed*, i.e., *HF_I* (12 bits), as long as the provided input *GD_I* is within the error capabilities of the error correction system. Also during the reconstruction phase, the Repetition Decoder and the Golay Decoder repeat their operations 86 times; each time, they serially process 264 bits generated based on PUF and Helper Data. The results of each iteration is a 12 bits buffered inside the Hash Function block. The Golay Decoder is the most complex block of the FE; it contains a *Finite State-Machine* (FSM) with nine states for vector decoding. As stated previously, a Golay Decoder can correct up to three errors. Its input *GD_I* comprises 24 bits (12 message bits combined with 12 parity bits). Figure 3 shows the states dedicated to error correction; these are selected depending on the location and number of errors in *GD_I*. Error wise, five different cases are possible, denoted in Fig. 3 as case (i) till (v).

(i) *GD_I* is error-free; thus, the four states where the error correction takes place are skipped.
(ii) there are three or less errors in the message bits of *GD_I* and none in the parity bits.
(iii) there are one or two errors in the message bits of *GD_I* and exactly one in the parity bits.
(iv) there is exactly one error in the message bits of *GD_I* and two or less in the parity bits.
(v) there are no errors in the message bits of *GD_I* and three or less in the parity bits.

The Golay Decoder has a variable latency depending on its input, with a maximum of 10 clock cycles and it comprises 61.5 % of FE gates.

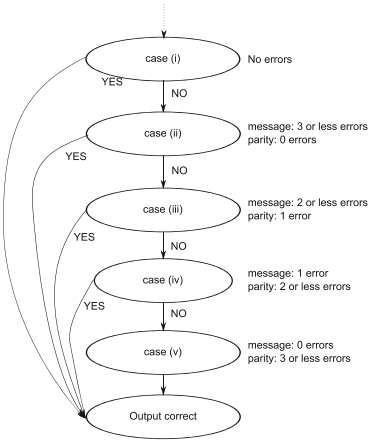


Fig. 3 Golay Decoder state-machine

2.2.1 Hash Function

last block of the reconstruction phase. It performs privacy amplification. This block concatenates the 1032 bits (12×86 iterations) received from the Golay Decoder and applies the hash function on it to calculate the 128 bit Cryptographic Key. Our Hash Function implementation comprises three main components: an input buffer, a *Linear Feedback Shift Register* (LFSR) and an accumulator register. First, the input *HF_I* is copied to the input buffer, which is then analyzed bit per bit. If the bit is one, the current LFSR output (which updates itself each cycle based on its polynomial function) is added (XORed) with the accumulator. However, if the bit is zero, the accumulator keeps its value. When all input bits are analyzed the value of the accumulator register is propagated to the output. The Hash Function has a latency of 32 clock cycles and it comprises 18.2 % of FE gates.

It is worth noting that the Fuzzy Extractor presented here is a *generic* construction of industrial implementations [27]; therefore, any test method developed for this circuit can be applied also to any other implementation.

3 Test Methods

First we define test and security requirements considered for the development of our test solutions. Then, we present our test methods and thereafter give means that can be used to further improve the quality of the proposed methods.

J Electron Test (2014) 30:581–594

3.1 Test versus Security Requirements

Efficient test solutions for FE must prevent compromising the system security. The following requirements and assumptions apply:

(a) The signals of *PUF* measurement, *Random Seed* and *HF_I* (see Fig. 2) shall not be revealed at any time, partially nor fully. An attacker learning this information might derive the *Cryptographic Key*, breaking the systems security.
(b) Helper data is assumed to be public knowledge and does not have to be secured.
(c) Reverse engineering the Fuzzy Extractor is not an issue. The Fuzzy Extractor uses algorithms that are standard and publicly known.
(d) The PUF circuitry has its own internal test method, therefore it is outside the scope of this work.
(e) Minimum fault coverage of 95 %. Extended test times combined with methods to increase fault coverage are supposed to compensate for the remaining 5 %.

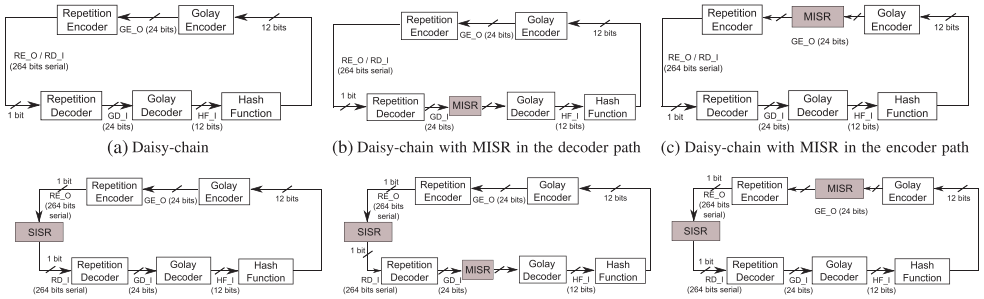
3.2 Secure Test Methods for FE

Next, we propose two secure test methods for FE: daisy-chain based and parallel test based methods. Both of them are scan-chain free, which is a security requirement. The two proposed methods will enable a good profiling of the maximum and minimum test time and area overhead. In an industrial application, a hybrid solution between these methods might be preferential.

3.2.1 Daisy-Chain Secure Test Method

We propose to (a) reuse the *Linear Feedback Shift Register* (LFSR) of the Hash Function block to create a random generator and, (b) test the FE in a loop-chain fashion, i.e., the outputs of each block are directly provided as inputs to next (connected) block as depicted in Fig. 4a. This approach results in a negligible area overhead. However, a high fault coverage for the Golay Decoder cannot be guaranteed. This is because the Golay Decoder receives error free input messages as provided by the Golay Encoder, which prevents the correct checking of all the decoder' states (see Fig. 3); e.g., in case the input vector of the Golay Decoder is error free as in case (i) of Fig. 3, the remaining four cases will be skipped. Hence, reusing LFSR of hash function with daisy-chain approach *alone* will not provide the required test quality for Golay Decoder. To solve this problem, the randomness of the patterns provided at the Golay Decoder inputs have to be improved in order to trigger all states of the Golay Decoder FSM. This can be done by inserting a *Multiple-Input-Shift-Register* (MISR) at the input of the Golay Decoder as shown in Fig. 4b. However, as the blocks are connected in a loop, the desired effect of randomness improvement can also be achieved by placing a MISR in any location between the Golay Encoder output and the Golay Decoder input (such as at the output of Golay Encoder in Fig. 4c), or a *Single-Input-Shift-Register* (SISR) if the location is just a serial line as it is the case in Fig. 4d). Moreover, a combination of MISR and SISR could be also used as shown in Fig. 4e and Fig. 4f. Comparing the area overhead and randomness of the several constructions presented in Fig. 4 reveals that:

1 Construction (d) results in the smallest area overhead.
2 Constructions (e) and (f) could lead to higher fault coverage, as the combination of using SISR and MISR could improve the randomness.
3 Constructions (b) and (c) as well as (e) and (f) are equivalent, reducing the number of constructions to four.



(a) Daisy-chain (b) Daisy-chain with MISR in the decoder path (c) Daisy-chain with MISR in the encoder path

(d) Daisy-chain with SISR (e) Daisy-chain with SISR and MISR in the decoder path (f) Daisy-chain with SISR and MISR in the encoder path

Fig. 4 Daisy-chain for a Fuzzy Extractor - different constructions

3.2.2 Parallel Secure Test Method

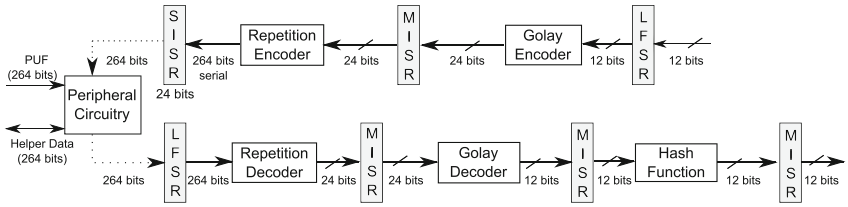
One way that can be used to reduce the overall test time and potentially increase the fault coverage of each FE block is to add a *Random Test Pattern Generator* (RTPG) at the input of each FE block. Increasing the randomness of the input patterns of each block will *potentially* increase the fault coverage as well. We will explore this approach while testing all FE blocks simultaneously (parallel test); the total test time is then the test time required by the largest block (in this case the Golay Decoder). In addition, a SISR/MISR is inserted at the output of each block for test data compression; see Fig. 5a. Starting with the first block, Golay Encoder, an LFSR is added at its input for random pattern generation and a MISR is added at its output for output test compression. The second block, Repetition Encoder, can reuse the MISR for pattern generation, reducing area overhead; a SISR is used for output compression as the output of the Repetition Encoder is a serial line. The same idea is applied for the remaining blocks. An LFSR is used as RTPG for the Repetition Decoder and a MISR is used to compress its test response. This MISR is reused as pattern generator for the Golay Decoder; a new MISR is added at the output of the Golay Decoder for output compression and reused for test pattern generation for the Hash Function. Finally, a last MISR is added at the output of the Hash Function for output compression.

Figure 5b shows the scheme implementation details for the Golay Encoder block; the remaining blocks have similar implementations. First, a MUX selects between the functional mode and the test mode by forwarding either the functional input (i.e., Random Seed) or the output of the RTPG (i.e., LFSR) to the input of block under test (i.e., Golay Encoder). In the functional mode, the output of the Golay Encoder is forwarded to the Repetition Encoder, while in the test mode the output of the Golay Encoder (test response) is compressed (MISR) and at the same time being sent as input test stimuli for the next block (Repetition Encoder). Once the test is concluded, the results are compared against a hardwired golden reference by means of XOR gates. The result of this comparison is a pass/fail signal. This approach results in a small test time, as the maximum test time is the test time of the most time consuming block to be tested. However, when compared with the daisy-chain approach, it has a larger area overhead.

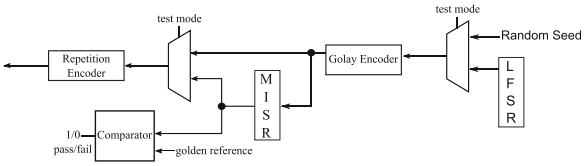
Considering a golden reference with defects, one of two cases may happen: either (a) a faulty device is not detected or (b) a good device is rejected. While both cases are costly, case (a) is more damaging but also very improbable. For a faulty device to pass the test, the faulty circuit would need to generate a MISR signature such that would match perfectly the also faulty golden reference. Moreover, the faulty MISR signature would serve as input to other blocks, which would cause the fault to be detected. However, if we want to increase the robustness of the golden references, some options are; e.g., comparing the test signatures against not one but two golden references (costly in area) or including a parity bit comparison of the test signature (cheaper as only 1 bit per golden reference is required).

3.3 Test Quality Improvement Using RTPG

Random-pattern-resistant faults might increase the test time; due to the specificity of the input test vector that detects such



(a) Block diagram of test mode



(b) Implementation details for Golay Encoder

Fig. 5 Parallel secure test method for a Fuzzy Extractor

J Electron Test (2014) 30:581–594 587

faults, random test pattern generators might take a large number of clock cycles to generate the required test vector. In literature many methods targeting the reduction of the test time by means of reconfiguration of the random test pattern generator parameters can be found. We identify three main classes of reconfigurable parameters that affect random test pattern generation: output, state and structure. Moreover, these methods can be combined. They are explained next.

- **Output reconfigurability:** are methods where one or more original generated random test patterns being not able to detect any fault are mapped into other test vectors that detect one or more faults. Known examples of this method are *bit-flipping/fixing* [28][26] and *reconfigurable network* [22].
- **State reconfigurability:** are methods where one or more states (seed) of the random test pattern generator LFSR are changed to skip a sequence of random patterns that do not detect additional faults. Known examples of this method are *full reseeding*, *partial reseeding* [18] and *encoded reseeding* [3].
- **Structure reconfigurability:** are methods where one or more random test pattern generator feedback networks are dynamically reconfigured; hence, the generated random test pattern sequence is also changed. A known example of this method is *multi-polynomial* [14].
- **Combined reconfigurability:** to even optimize further the results, some of the previous methods can be combined together; e.g., in [14] the authors combined state (reseeding) and structural (multi-polynomial) reconfigurabilities.

Note that optimizing the Daisy-chain secure test method using the above scheme is not allowed as the circuitry of the random test pattern generator (i.e., the circuitry of the hash function) cannot be manipulated; this is because otherwise an attacker could use this feature to gain access to the cryptographic key during operation mode.

4 Experiments Results

We first define the experiments. Thereafter, we present and discuss the results. Finally, we investigate the impact of reconfigurable RTPG on test quality improvement.

4.1 Experiments Performed

We synthesized the Fuzzy Extractor, described in VHDL, using 0.35μm technology node and Synopsys Design Compiler. The design compiler outputs a verilog netlist that is used to extract a fault list with Synopsys *Automated Test Pattern Generation* (ATPG) tool TetraMAX. We used LIFTING fault simulator optimized for functional BIST to analyze the fault coverage [17]. The results are analyzed thereafter with MATLAB. The experiments performed to evaluate each of the proposed schemes are described next.

Daisy-chain secure test method: To evaluate the quality of the proposed solutions in Fig. 4 in terms of fault coverage (FC), test time and area overhead, we performed the following six experiments:

1) **Default:** we simulated the circuit as in Fig. 4a for 15×10^4 clock cycles and analyzed the FC. This number of clock cycles is assumed to be our test time budget for all remaining experiments.
2) **MISR:** we simulated the circuit as in Fig. 4b (equivalent to Fig. 4c).
3) **SISR:** we simulated the circuit as in Fig. 4d.
4) **SISR + MISR:** we simulated the circuit combining SISR and MISR as in Fig. 4e (equivalent to Fig. 4f).
5) **Default + SISR:** we simulated the circuit in two stages. First, as in Fig. 4a, we simulated the FE using the default loop-chain for 25 % of the test time budget. Second, as in Fig. 4d, we included the SISR in the chain flow (between the Repetition Encoder and Repetition Decoder blocks) and analyzed the FC over the remaining 75 % of the test time. The goal of this experiment is to analyze the impact of *combining* the default scenario as in Fig. 4a with that of SISR in Fig. 4d.
6) **Default + MISR:** in this experiment, we repeated the procedure (5), but replacing the SISR with a MISR.

Parallel secure test method: To determine the impact that the state (seed) and the structure (polynomial) reconfigurability have on the FC and test time, we performed nine experiments per FE block; each FE is tested using three polynomials, each combined with three seeds. Each of the three used seeds and polynomials are described next. Note that conceptually, output reconfigurability and state reconfigurability are very similar; hence, the influence of the output reconfigurability can be easily derived from the results of the experiments carried out for the state reconfigurability.

(1) **State reconfigurability:** we tested each FE block with a random test pattern generator (primitive polynomial) using three different initial seeds; these are:
   (i) **Seed 0:** all bits are zero except the last bit, which is a one (e.g., '0...00001').
   (ii) **Seed 1:** a randomly chosen starting state (e.g., '10...0110').
   (iii) **Seed 2:** a string of alternating zeros and ones (e.g., '01...10101').

(2) **Structure reconfigurability:** we tested each FE block with a random test pattern generator using three different polynomials; one primitive ('Poly 0') and two non-primitive polynomials ('Poly 1' and 'Poly 2'). The size of the polynomials depends of the number of input bits of the block being tested. The polynomials used for Golay Encoder, Repetition Decoder and Hash Function are:

(i) **Poly 0:** $x^{16}+x^{14}+x^{13}+x^{11}+1$.
(ii) **Poly 1:** $x^{16}+x^{14}+x^{12}+x^{11}+x^{10}+x^{7}+x^{6}+x^{5}+1$.
(iii) **Poly 2:** $x^{16}+x^{13}+x^{12}+x^{8}+x^{4}+x^{3}+1$.

while those used for Repetition Encoder and Golay Decoder are:

(i) **Poly 3:** $x^{24}+x^{23}+x^{22}+x^{17}+1$.
(ii) **Poly 4:** $x^{24}+x^{22}+x^{17}+x^{16}+x^{15}+x^{14}+x^{10}+x^{6}+x^{5}+x^{1}+1$.
(iii) **Poly 5:** $x^{24}+x^{22}+x^{21}+x^{19}+x^{17}+x^{14}+x^{12}+x^{11}+x^{10}+x^{9}+x^{8}+x^{7}+x^{6}+x^{4}+x^{3}+x^{2}+1$.

All polynomials were used with 'Seed 0'.

4.2 Results for Daisy-Chain Secure Test Method

Figure 6 shows the FC (y-axis) versus the number of clock cycles (x-axis) of the experiments; part (a) gives the results for the first four experiments, and part (b) for the remaining two experiments. From Fig. 6a we can observe the following.

1) Default experiment realizes a FC of only 35.26 %. This FC is quickly realized in the first 2.3k clock cycles (ccs). The figure clearly shows that the FC remains constant for the remaining clock cycles.
2) For the remaining three schemes, the targeted FC of 95.00 % is achieved after 4.71×10^4 ccs for SISR, after 4.56×10^4 ccs for MISR and after 4.33×10^4 ccs for SISR+MISR. The remaining clock cycles until the end of the experiment lead to an additional FC increment of 1.29 % for SISR, 1.34 % for MISR and 1.43 % for SISR+MISR.
3) Using SISR+MISR is relatively the best method in terms of FC and test time; however, it has the largest area overhead.

From Fig. 6b we can observe the following.

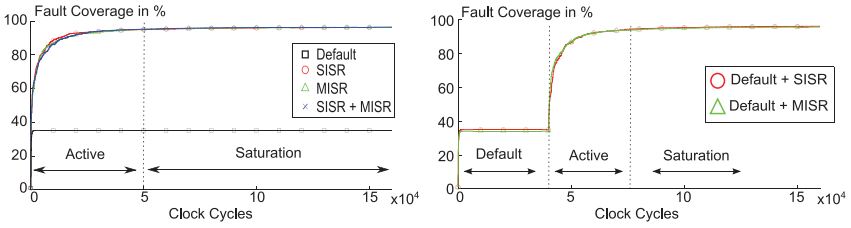
1) During the first stage of Experiment 5 ('Default + SISR') and Experiment 6 ('Default + MISR'), a FC of 35.07 % respectively 34.16 % is realized. This is a little less than the FC achieved with Experiment 1 ('Default') due to the insertion of the extra hardware (SISR/MISR).
2) In the second stage, the FC is significantly increased; Experiment 5 realizes the targeted 95 % FC after 8.57×10^4 clock cycles, while Experiment 6 does this after 9.80×10^4 clock cycles (first stage included).
3) Making use of the entire test time budget of 15×10^4 results in a FC of 96.24 % for 'Default + SISR' and 95.77 % for 'Default + MISR'.

Obviously the inserted blocks required additional area overhead; this is 2.2 % w.r.t. the FE for SISR, 6.80 % w.r.t. the FE for MISR and 9.0 % w.r.t. the FE for SISR combined with MISR.

Inspecting the obtained simulation results clearly reveals that testing FE in a loop chain fashion (Fig. 6a) will never realize the required product quality; the FC realized in our case study does not exceed 35 %. Additional DFT to increase the randomness of the test patterns is essential. E.g., introducing a SISR in the loop can increase the FC up to 96.29 % at the cost of 2.2 % area overhead for the predefined test budget.

4.3 Results for Parallel Secure Test Method

The target of this method is to optimize the test time while realizing the targeted FC. The test time will be then



(a) Without default stage (b) With default stage

Fig. 6 Fault coverage of Fuzzy Extractor Daisy-chain secure test method versus clock cycles

J Electron Test (2014) 30:581–594 589

defined by the Golay Decoder which comprises 61.5 % of the FE. The simulation results show that the test time is 3.5k ccs. Moreover, the realized FC by can be obtained by combining the weighted FC of each block. In our case, the FC can be calculated as follows. $FC_{total} = 4.5\,\% \times FC_{GolayEncoder} + 5.2\,\% \times FC_{RepetitionEncoder} + 5.2\,\% \times FC_{RepetitionDecoder} + 53.1\,\% \times FC_{GolayDecoder} + 32.0\,\% \times FC_{HashFunction}$.

4.3.1 Golay Encoder

Figure 7 shows the FC versus the number of clock cycles for the nine experiments. Part (a) of the figure shows the results of 'Poly 0' for the three different seeds and part (b) and (c) present similar results but then for 'Poly 1' and 'Poly 2', respectively. The figure clearly reveals that all experiments realize the targeted FC ($FC_{GolayEncoder}$). Nevertheless, the impact of varying the polynomial and/or the seed on both test time and FC cannot be ignored; e.g., after only six ccs 'Poly 0' with 'Seed 2' reaches FC of 82.30 %, while with 'Seed 0' or with 'Seed 1' this does not exceed 57.67 % and 53.27 %, respectively. Note that 'Poly 1' with 'Seed 1' is the most efficient combination in realizing the targeted FC.

4.3.2 Repetition Encoder

Figure 8 shows that all the nine experiments result in a constant FC of maximum 90.7 % after circa 540 ccs. The impact of the polynomials and the seeds is significant. 'Poly 4' with 'Seed 0' is the best combination realizing FC of 90.7% after 542 ccs.

To reach the targeted FC_{total} with no increase in the test time there are two options. First, investigate and apply the specific test vectors that detect the remaining faults. Second, increase the FC of the other blocks, such that it compensates for the lower FC of Repetition Encoder. The first option is very expensive, as it requires storing the extra vectors on the die. However, the second option is cost-free; simply by extending the test time of one or more of the other blocks, as long as still below the test time budget.

In our case, we chose to extend the FC of the Golay Encoder to 99.8 %, which is realized in 27 ccs when using 'Poly 3' combined with 'Seed 1'.

4.3.3 Repetition Decoder

Figure 9 shows the simulation results. The figure reveals that all experiments realize the targeted FC in no more than 600 ccs. Also here varying the polynomial and/or the seed has a clear impact. 'Poly 2' combined with 'Seed 0' is the most efficient pair realizing the targeted FC.

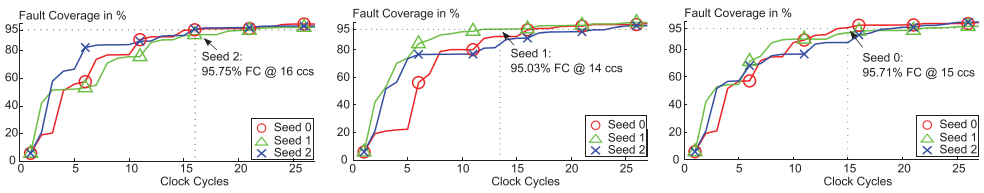
4.3.4 Golay Decoder

Figure 10 shows the simulation results. All the nine experiments realize the targeted FC. 'Poly 4' combined with 'Seed 2' is the most efficient combination.

4.3.5 Hash Function

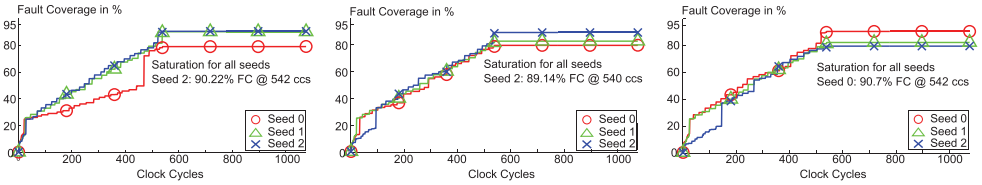
Figure 11 shows the simulation results for all the nine experiments. The impact of varying the polynomial and or seed is marginal. After circa 700 ccs, the FC ($FC_{HashFunction}$) for all cases is 95 %.

The results clearly reveal that an appropriate selection of polynomial and or seed significantly increases the FC and/or decreases the test time per FE block, except for the Hash Function. This is due to the specificity of the remaining test patterns required to detect the last remaining faults. Additionally, the results reveal that the required test time per FE block varies significantly (from 14 ccs up to 3.5k ccs) as well as the realized FC. Finally, the results also show that the remaining test time budget is a useful resource to further increase the targeted FC. The additional area overhead of this method is 18.6 % w.r.t. the FE.



(a) Poly 0 (b) Poly 1 (c) Poly 2

Fig. 7 FC of Golay Encoder with different polynomials and different seeds



(a) Poly 3 (b) Poly 4 (c) Poly 5

Fig. 8 Fault coverage of Repetition Encoder with different polynomials and different seeds

4.4 Impact of Reconfigurable LFSR/RTPG Parameters

To demonstrate the potential of reconfiguring RTPG parameters in improving FC and reducing the test time, we use the Golay Encoder as a case study. The idea is to skip test patterns without additional FC. For example, Figure 7a shows that during clock cycles 9 and 10 no faults are detected. We skip all the clock cycles that do not contribute to FC but that do consume test time by reseeding, i.e., by changing the seed of the registers of the RTPG. We stop the test when the 95 % FC target is realized. Figure 12a shows the number of faults that each test vector detects (obtained using LIFTING tool [17]), while Fig. 12b shows the impact of skipping the test vectors that do not contribute to FC such that the overall test time is minimized. Figure 12b shows that a speedup of 1.5× can be realized by reseeding three times (speedup from 16 ccs to 11 ccs).

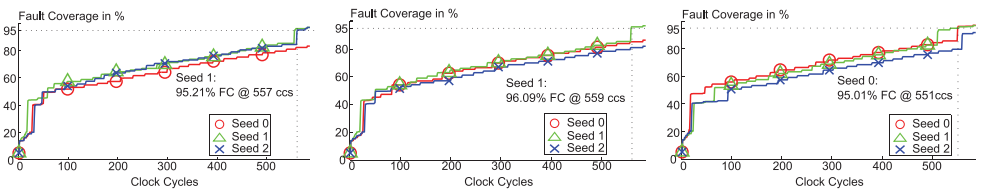
On the downside, the area overhead required for the implementation of the method is very large when compared to the test without reseeding. Each seed requires 12 bits, hence, a total of 36 bits (12 bits×3 seeds) has to be stored on the die. Due to the small size of the Golay Encoder, the extra area overhead that would be needed makes the optimization methods prohibitively expensive for our case study circuit. Moreover, depending on the implementation, extra test time to load the seeds might be required; however, in [18] the authors present a reseeding solution that does not increase the total test time. However, the goal is to find trends to apply in larger circuits. When considering applying one of the test quality improvement methods, the test designer must take into consideration the following parameters: (i) the number of test vectors to anticipate, (ii) the number of bits that need to be flipped from the original RTPG to generate test vectors that detect faults and (iii) the number of times that this operation needs to be performed.

5 Discussion

First we compare both the efficiency of both secure test methods. Second we compare our results with the prior work. Thereafter we make a security analysis of the proposed secure test methods. Finally we provide a list of recommendations to secure test an FE.

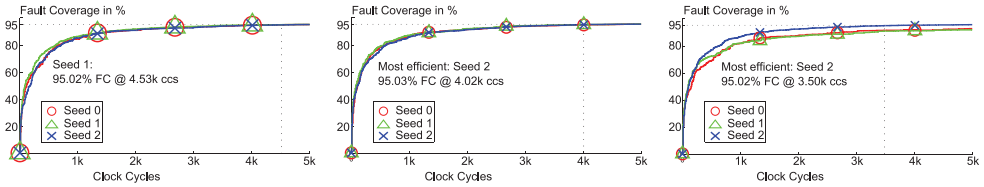
5.1 Comparison Between the Secure Test Methods

Table 1 summarizes the main features of the two secure test methods previously proposed. Testing the FE using the parallel secure test method with dedicated RTPGs per block for test pattern generation and for result compression has an area overhead of 18.6 % w.r.t the FE while the area overhead of the daisy-chain secure test method is of only 2.2 % w.r.t. the FE, i.e., the parallel secure test method has 8.5× larger area overhead. Note that the additional area overhead represents a small value, as a typical



(a) Poly 0 (b) Poly 1 (c) Poly 2

Fig. 9 Fault coverage of Repetition Decoder with different polynomials and different seeds



(a) Poly 3 (b) Poly 4 (c) Poly 5

Fig. 10 Fault coverage of Golay Decoder with different polynomials and different seeds

FE has a overall area of 1.4kGE, where GE (Gate Equivalent) is a measure of area in any given technology; 1 GE is the area of a NAND gate [20]. However, it realizes the same fault coverage in 3.5k clock cycles, i.e., 13.46× faster when compared with the 47.1k clock cycles of the daisy-chain secure test method. In addition, having a dedicated RTPG per block enables optimizing methods to achieve higher FC in shorter test time, but again at the expense of area overhead. Optimizing the daisy-chain secure test solution is not feasible because the circuitry of the random test pattern generator (i.e., the circuitry of the hash function) cannot be manipulated as an attacker could use this feature to gain access to the cryptographic key during operation mode.

5.2 Comparison with Prior Work

No prior work focus on the same problem addressed by this paper; hence, a quantitative comparison with prior work is not possible. However, we compare the work qualitatively of the Daisy-chain secure test method to [11] and [9]. The fault coverage of the method is in line with the FC reported in other self-test methods [11] and [9]. The area overhead of the proposed method is negligible, which is intrinsic to methods that reuse hardware. The parallel secure test method can be seen as a test time optimization of the daisy-chain method; realizing the same fault coverage but in 14.3× less time at the cost of 8.5× more area, which is a common trade-off.

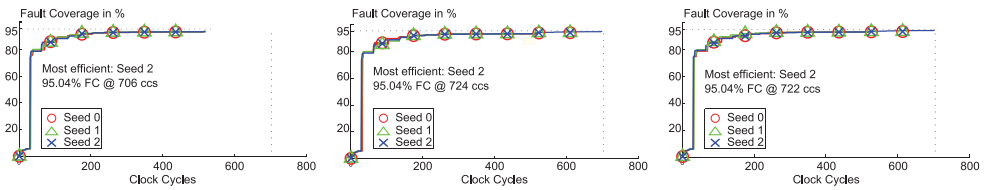
5.3 Security Analysis

Ideally, the design of a secure device begins with identifying which class(es) of attacks it should prevent. Several countermeasure methods must be combined to deliver the required level of security. In other words, no method alone is secure. In this work we aim at the prevention of side-channel attacks by means of the test infrastructure. Our BIST methods do not allow data to be scanned-in nor do they leak information on the test results (only pass/fail).

We make a brief analysis of the security of the methods proposed considering the following vulnerabilities. We consider that an attacker a) might use Hardware Trojan to gain access to the switch between functional and test modes to get knowledge on the full or partial value of either PUF or key and b) might try to attack the stored seeds to, e.g., decrease the fault coverage.

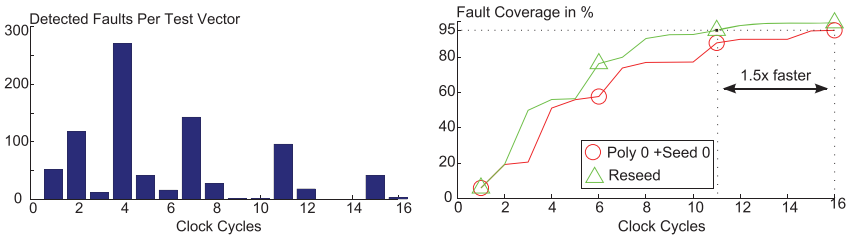
With respect to attack a), during test mode, regardless if activated by a legitimate source or by a Hardware Trojan, all registers are reset. Therefore, any traces sensitive information are destroyed. However, considering that the inserted Hardware Trojan can circumvent this protection measure, we would need to combine/enhance our methods with one or several countermeasures proposed in the literature; e.g., [1].

Considering an attack on the stored seeds. Attacks aiming the stored seeds would need to be invasive attacks. Typically, invasive attacks require the depackaging of the device



(a) Poly 0 (b) Poly 1 (c) Poly 2

Fig. 11 Fault coverage of Hash with different polynomials and different seeds



(a) Detected faults per test vector

(b) FC before and after reseeding

Fig. 12 Reseeding results for Golay Encoder

under attack and other destructive measures. Destroying a device to change its test seeds to lower its fault coverage it is not profitable. Therefore, stored seeds do not seem vulnerable to this type of attack, as no information could be gained from it.

5.4 Recommendations

We provide a generic step-by-step procedure for secure testing of FE based on our findings. These steps are:

1) Identify each block and deeply understand its functionality (which is critical for successful functional testing).
2) Assess the characteristics of each block (e.g., its area overhead, if the block comprises a state-machine or not and state-machine complexity).
3) Identify possible challenges by testing a certain block with a random input source.
4) Implement parallel secure test method for the shortest test time and implement Daisy-chain secure test method for lowest area overhead.
5) Identify if extra components are needed in order to increase the fault coverage, such as a SISR. If so, analyze the trade-off of such components in terms of their possible locations and of its impact on security, fault coverage, test time and area overhead.
6) Explore further the two secure test methods; optimize parallel secure test method by choosing an optimal polynomial and seed or optimize Daisy-chain secure test method by using a SISR to activate uncovered paths in the state-machine.
7) Analyze the FC, test time and area overhead of all test methods separately and when combined with complementary schemes.
8) Determine and select the best test method and complementary schemes to meet the design requirements.

Table 1 Test methods' results

Test Method	Area overhead	Test time	FC
Daisy-chain (+ SISR)	2.2%	47.1k clock cycles	95 %
Parallel (+ DFT)	18.6 %	3.5k clock cycles	95 %

6 Conclusion

We demonstrated two secure test methods for a Fuzzy Extractor, both are scan-chain free. The first secure test method is based on daisy-chains; it reuses Fuzzy Extractor blocks for test pattern generation and output compression. The second method tests all the Fuzzy Extractor blocks simultaneously by adding dedicated test pattern generation blocks. The results show that the first method has an inherent low area overhead 2.2 %, while it realizes a fault coverage of 95 % using only 47.1k clock cycles. The second method realizes a similar fault coverage with 8.5× more area overhead but 13.46× faster, when compared with the first method. In addition, we identified and analyzed techniques to optimize the previous methods, and provided a generic step-by-step procedure to test any given Fuzzy Extractor based on our findings.

This case study considers a small FE. Nonetheless, the proposed approaches are still valid for any FE construction. Moreover, the two proposed methods provide upper and lower bounds for both test time and area overhead. A real system would benefit from a hybrid solution between the two proposed methods.

Acknowledgments The authors would like to thank Geert-Jan Schrijen and Peter Simons from Intrinsic-ID B.V. for the useful discussions on the architecture of the Fuzzy Extractor used in this publication. It is worth noting that the work presented in this publication was partially sponsored by COST action TRUDEVICE IC1204.

References

1. Agrawal D, Baktir S, Karakoyunlu D, Rohatgi P, Sunar B (2007) Trojan Detection using IC Fingerprinting, IEEE Symposium on Security and Privacy (SP) pp 296–310
2. Ali SS, Said SM, Sinanoglu O, Karri R (2013) Scan Attack in Presence of Mode-Reset Countermeasures. IEEE International on-line testing symposium (IOLTS) 230:231
3. Al-Yamani AA, McCluskey EJ (2003) Built-in reseeding for serial BIST. VLSI Test Symp:63–68
4. Bo Y, Kaijie W, Karri R (2004) Scan-based Side-Channel Attack on Dedicated Hardware Implementations of Data Encryption Standard, Proceedings of International Test Conference, pp 339–344
5. Cortez M, Roelofs G, Hamdioui S, Di Natale G (2014) Testing PUF-Based Secure Key Storage Circuits Design, Automation and Test in Europe Conference and Exhibition (DATE), pp 1–6
6. Da Rolt J, Di Natale G, Flottes ML, Rouzeyre B (2013) A Smart test controller for scan-chains in secure circuits. IEEE International On-line Testing Symposium (IOLTS):228–229
7. Da Rolt J, Di Natale G, Flottes ML, Rouzeyre B (2012) New security threats against chips containing scan chain structure. IEEE International Symposium on Hardware-Oriented Security and Trust (HOST) p 110
8. Das A, Kocabaş Ú, Sadeghi AR, Verbauwhede I, Sadeghi AR, Verbauwhede I (2012) PUF-based Secure Test Wrapper Design for Cryptographic SoC Testing Design, Automation and Test in Europe Conference and Exhibition pp 866–869
9. Di Natale G, Doulcier M, Flottes ML, Rouzeyre B (2010) Self-test techniques for crypto-devices. IEEE Trans VLSI Syst 18:2
10. Dodis Y, Reyzin L, Smith A (2004) Fuzzy Extractors: How to Generate Strong Keys from Biometrics and other Noisy Data, Advances in Cryptology-EUROCRYPT vol. 3027, LNCS, Springer Berlin Heidelberg, pp 523–540
11. Doulcier M, Flottes ML, Rouzeyre B (2008) AES-based BIST: self-test, test pattern generation and signature analysis. IEEE International symposium on electronic design, Test & Applications, pp 314–321
12. Guajardo J, Kumar SS, Schrijen GJ, Tuyls P (2007) FPGA Intrinsic PUFs and their Use for IP Protection. Workshop on Cryptographic Hardware and Embedded Systems (CHES): 63–80
13. Hamdioui S, Di Natale G, van Battum G, Danger JL, Smailbegovic F, Tehranipoor M (2014) Hacking and Protecting IC Hardware Design, Automation and Test in Europe Conference and Exhibition, pp 1–7
14. Hellebrand S, Rajski J, Tarnick S, Venkataraman S, Courtois B (1995) Built-in test for circuits with scan based on reseeding of multiple-polynomial linear feedback shift registers. IEEE Trans Comput 44(2):223–233
15. Hely D, Bancel F, Flottes ML, Rouzeyre B (2006) A Secure Scan Design Methodology Design, Automation and Test in Europe Conference and Exhibition, pp. 1–2
16. http://mathworld.wolfram.com/GolayCode.html
17. http://www.lirmm.fr/
18. Krishna CV, Jas A, Touba NA (2001) Test vector encoding using partial LFSR reseeding. Int Test Conf:885–893
19. Lee J, Tehranipoor M, Patel C, Plusquellic J Tehranipoor M, Patel C, Plusquellic J (2005) Securing scan design using lock and key technique. IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems (DFT), pp 51–62
20. Leest Vvd, Preneel B, Sluis Evd (2012) Soft Decision Error Correction for Compact Memory-Based PUFs using a Single Enrollment, Workshop on Cryptographic Hardware and Embedded Systems (CHES), pp 268–282
21. Linnartz JP, Tuyls P (2003) New shielding functions to enhance privacy and prevent misuse of biometric templates. In: Proceedings of Audio- and Video-Based Biometric Person Authentication (AVBPA) 03, pp 393–402, Springer Berlin / Heidelberg
22. Lei L, Chakrabarty K (2004) Test set embedding for deterministic BIST using a reconfigurable interconnection network. IEEE Trans Comput Aided Des Integr Circ Syst 23(9):1289–1305
23. Liu Y, Wu K, Karri R (2011) Scan-based attacks on linear feedback shift register based stream ciphers. ACM Trans Design Autom Electr Syst 16(2):20
24. Pless V (1986) Decoding the golay codes. IEEE Trans Inf Theory 32(4):561–567
25. Skoric B, Tuyls P, Ophey W (2005) Robust key extraction from physical unclonable functions, Applied cryptography and network security, vol. 3531 of LNCS, pp 99135, Springer Berlin / Heidelberg
26. Touba NA, McCluskey EJ (2001) Bit-fixing in pseudorandom sequences for scan BIST. IEEE Trans Comput Aided Des Integr Circ Syst 20(4):545–555
27. www.unique-project.eu
28. Wunderlich HJ, Kiefer G (1996) Bit-flipping BIST. IEEE/ACM Int Conf Comput Aided Des:337–343
29. Yang B, Wu K, Karri R (2006) Secure Scan: A Design-for-Test Architecture for Crypto Chips. IEEE Trans Comput Aided Des Integr Circ Syst 25(10):2287–2293

Mafalda Cortez received her M.Sc. degree in Electrical and Computers Engineering - Telecommunications, Electronics and Computers from Faculdade de Engenharia da Universidade do Porto (FEUP), Portugal. She is currently pursuing her PhD at the Computer Engineering Lab from the Delft University of Technology in collaboration with Intrinsic-ID B.V., a company on Hardware Intrinsic Security. Her research interests include circuit design and modelling, hardware security and secure IC test.

Gijs Roelofs received his M.Sc. degree in Embedded Systems from the Delft University of Technology in 2013. Currently he is a Security Evaluator at Brighsight B.V., an IT security lab, which tests Integrated Circuits, Smart Card Software, including crypto libraries and payment terminals. He specialized in Semi Invasive Attacks, namely Laser Attacks and Electro Magnetic Fault Injecting. Over the years he has gained expertise in testing state-of the-art secure systems. His current research is focused on attacks of products with Near Field Communication and Single Wire Protocol.

Said Hamdioui received the MSEE and PhD degrees (both with honors) from Delft University of Technology (TUDelft), Delft, The Netherlands. He is currently co-leading dependable-nano computing research activities within the Computer Engineering Laboratory of TUDelft. Prior to joining TUDelft, Hamdioui worked for Intel Corporation (in Santa Clara and Folsom, Califorina), for Philips Semiconductors R&D (Crolles, France) and for Philips/ NXP Semiconductors (Nijmegen, The Netherlands). His research interests include Nano-Computing, Dependability, Reliability, Hardware Security, Memristor Technology, Test Technology & Design-for-Test, 3D stacked IC, etc. Professor Hamdioui published one book and co-authored over 130 conference and journal papers. He has consulted for many semiconductor companies (such as Intel, ST Microelectronics, Altera, Atmel,

Renesas, DS2, ST Microelectronics, etc). He is strongly involved in the international test technology community as a member of organizing committees or as a member of technical program committees of the leading conferences, as a reviewer for major journals, etc. He delivered dozens of keynote speeches, distinguished lectures, and invited presentations and tutorials at major international forums/conferences and leading semiconductor companies. Hamdioui is a Senior member of the IEEE; he serves on the editorial board of the Journal of Electronic Testing: Theory and Applications (JETTA), on that of Design and Test. He is also a member of AENEAS/ENIAC Scientific Committee Council (AENEAS =Association for European NanoElectronics Activities).

Giorgio Di Natale received the PhD in Computer Engineering from the Politecnico di Torino (Italy) in 2003 and the HDR (Habilitation Diriger les Recherches) in 2014 from the University of Montpellier II (France). He is currently a researcher for the National Research Center of France at the LIRMM laboratory in Montpellier. He has published publications spanning diverse disciplines, including VLSI Testing, Memory Testing, Fault Tolerance, Reliability, Hardware Security and Trust. He is the Action Chair of the COST Action IC1204 (TRUDEVICE) on Trustworthy Manufacturing and Utilization of Secure Devices. He is the chair of the European group of the TTTC, Golden Core member of the Computer Society and Senior member of the IEEE.

Multi-segment Attack-resistant DFT for Secure ICs

3

Mafalda Cortez Said Hamdioui
Delft University of Technology
Delft, the Netherlands
{A.M.M.O.Cortez,S.Hamdioui}@tudelft.nl

Giorgio Di Natale Marie-Lise Flottes Bruno Rouzeyre
LIRMM, Universit Montpellier II
Montpellier, France
{Giorgio.DiNatale, Marie-Lise.Flottes, Bruno.Rouzeyre}@lirmm.fr

Ilia Polian
University of Passau
Passau, Germany
Ilia.Polian@uni-passau.de

***Abstract*—Scan-chains are the most commonly used Design-for-Testability (DFT) infrastructure to enhance testability and test pattern generation for digital ICs. However, scan-chains introduce security vulnerabilities, offering a back door for scan-based attacks. This paper introduces the Multi-Segment Secure Scan (MSSS) test scheme to prevent the success of scan-based attacks. MSSS has three main advantages: it is generic (not standard specific), it integrates a number of features to prevent leaking attack progress information and it provides tunable (flexible) security levels. This flexibility is important as it allows for the optimization of secure DFT solutions depending on the target application. To illustrate the scheme, MSSS is implemented on a crypto core with three security levels. The results show that MSSS does not only go beyond the state-of-the-art by being flexible, but also by having an inherently very low area overhead and no impact on the circuit performance. Moreover, Intellectual Property (IP) suppliers can use MSSS to provide their customers with restricted access to different parts of a core; hence, enhancing the confidentiality of their IPs.**

Keywords: Hardware Security, Secure Testing, Enhanced Scan-Chains, IP Protection

I. INTRODUCTION

Testing digital Integrated Circuits (ICs) is essential to identify faulty devices. Design-for-Testability (DFT) structures are added to enhance testability and respective diagnosis. The most common DFT technique is the insertion of scan-chains, which allows shifting-in test vectors and shifting-out test responses to and from storage elements. These structures increase the observability and controllability of IC internal nodes, by facilitating accessibility to internal states; making test vector generation easier. However, scan-chains open a back door that malicious users can exploit to gain access to sensitive information; these attacks are known as scan-based attacks [1–7]. Scan-based attacks typically aim at retrieving the secret encryption key of secure devices by using the scan-chains to shift out the information stored in the flip-flops (FF).

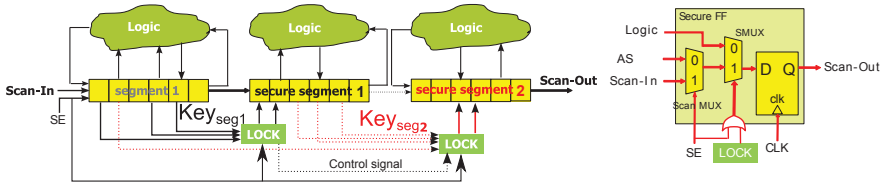
Several countermeasures against scan-based attacks have been proposed in literature [7–21]. These countermeasures can be divided into scan control methods [7–10], unauthorized scan-shift detection methods [7,9,11–16] and data confusion methods [17–21]. Scan control methods involve power-off or reset of scan FF when switching from mission to test mode; unauthorized scan-shift methods involve reset scan FF when an unauthorized shift is detected, and they include scan pattern watermarking [7,9,11–13], scan-enable tree monitoring [14] and spy FF [15–17]; while data confusion methods scramble the stream shifted out of the scan-chain by shuffling the output according to a specific function. However, a common main weakness of prior solutions is that they leak information to the attacker, at least on the test procedure and thus, on possible attack progress.

In this work we introduce the Multi-Segment Secure Scan (MSSS) test scheme against scan-based attacks targeting secure applications; within this context the following contributions apply:

- generic solution that can be integrated in any circuit,
- no leakage information on attack progress,
- tunable (flexible) security segments, allowing secure DFT solution optimization depending on the targeted application,
- no performance penalty in functional mode.

Segmentation of the serial test access mechanism was proposed in [22] to isolate the instrument during the test procedure of a IEEE 1687 compliant circuit. It is important to emphasize that our work, though compatible with, has three main distinguishable characteristics from work presented in [22]: (i) in [22] the proposed architecture and implementation of the security features are dedicated to the Segment Insertion Bits (SIBs) of the IEEE 1687 network (iJTAG), while the work developed in this paper is dedicated to scan-chains in general. (ii) Preventing Power Analysis is beyond the scope of [22]. Conversely, our generic secure DFT includes prevents the disclosure of the test mechanism through power analysis. (iii) Finally, the methodology in [22] proposes a SIB chain that varies its length according to the key guess. A variation in the SIB chain leaks information that an attacker may use to retrieve the system's key. Moreover, this construction relies on the assumption that an attacker does not have knowledge on the design of the SIB chain. To prevent the leakage of information to an attacker, in this work, we keep the length on the scan-chain constant and accessible all the time.

The rest of the paper is organized as follows: Section II introduces MSSS architecture. Section III discusses the implementation details. Section IV show how to apply MSSS on a crypto core with three security segments and reports the results. Section V compares the proposed method with the state-of-the-art, discusses the pros and cons of the solution and investigates how to further increase the security of the system. Finally, Section VI concludes the paper.



(a) Overview of a core with L (=2) SCSS (b) Secure Flip-Flop of a SCSS

Fig. 1: Multi-Segment Secure Scan

II. Multiple Segment Secure Scan Architecture

First, we briefly define the requirements for the development of secure scan-chains. Second, we present MSSS test concept including a discussion of its strengths and weaknesses.

A. Goals and Assumptions

Our goal is to design a secure test method being able to protect scan-chains. The solution needs to satisfy the following requirements:

- Generic; i.e., integrable in any circuit.
- Secure; i.e., exponentially increase the invested amount of resources required to successfully hack the device.
- High fault coverage.
- Short test time.
- Minimum impact on the circuit performance.
- Flexible; i.e, tunable number of secure segments according to targeted application.

B. MSSS Concept

Fig. 1(a) shows the schematic of Multi-Segment Secure Scan (MSSS). MSSS comprises a series of L *Scan-Chain Secure Segments* (SCSS); each secure segment is protected by a different secret key, with a total length K. To unlock a given SCSS, all precedent SCSS must be unlocked. Once all SCSS are unlocked, the circuit can be tested using the classic scan-chain approach. The key bits to unlock SCSS are taken from the FF of the previous unlocked segments; e.g., in Fig. 1(a), segment 2 is unlocked when the correct key is scanned into segment 1, while segment 3 is unlocked when the correct key is scanned into segments 1 and 2. Only a subset of the scan-chain FFs are used as key bits for two main reasons: (a) cost (area overhead and MSSS unlocking time) and (b) to (potentially) increase the attack complexity; they are addressed next.

The required area to either store or deploy the reference key and perform the comparison increases with larger key size and effects MSSS unlocking time negatively. We consider that the reference key can be either hardwired, Physical Unclonable Function (PUF) based or handled by a key manager. Each option has its trade-offs between the cost and security; this study is out of the scope of our work.

Attack complexity can be increased by combining the secret key security with the uncertainty provided by the key bits unknown number and location in the scan-chain. Considering a brute force attack, we identify three scenarios (see Table I) according to the available information to the attacker regarding the number and location of the key bits in the scan-chain. The total number of trials necessary to find the key of length K in a scan-chain of length N, for the different scenarios is:

TABLE I: Different attack scenarios

		Location of Key Bits	
		KNOWN	UNKONWN
Number of Key Bits	KNOWN	A	B
	UNKNOWN		C

- **A** - (classic); it is assumed that the attacker knows the system design (i.e., the number and the location of key bits in the scan-chain). A brute force attack requires 2^K trials.
- **B** - the attacker knows the key bits number, but not their location. In this scenario, a brute force attack results in:

$$\#Trials = min(2^N, C_K^N \times 2^K) \quad (1)$$

  where, C_K^N describes the number of combinations of K key bits from N bits (scan size) and 2^K is the number of possible key; e.g., consider a scan-chain with length N=100 and two keys scenarios (i) K=30 and (ii) K=20. Then, the total number of trials needed will be 2^{100}=1.3×10^{30} and C_{20}^{100}x2^{20}=5.6×10^{26}, respectively.
- **C** - the attacker does not know the key bits number nor their location. In this scenario, a brute force attack results in:

$$\#Trials = min(2^N, \sum_{K=1}^{N} C_K^N \times 2^K) \quad (2)$$

  Clearly, the effort required to perform a brute force attack increases exponentially when hiding the key length and bit location.

III. Implementation Aspects

First, we detail the implementation of MSSS and Alternative Source (an additional feature to reduce attack information leakage). Second, we investigate how to securely partition a scan-chain, which is an important aspect for MSSS method. Thereafter, we detail all the countermeasures MSSS method offers. Finally, we discuss MSSS plus and cons.

A. MSSS Implementation

Fig. 1(b) presents the main component of each SCSS; i.e., a secure FF. As Fig. 1(a) shows, a SCSS is a chain of secure and

scannable FF (in short, secure FFs) unlocked by the same key. In each secure FF, an extra multiplexer (*Secure Multiplexer* SMUX) is inserted between the scan multiplexer and the FF; see Fig. 1(b). The SMUX selects between unlocked (0 - forward the logic output) and locked (1 - forward the output of the scan MUX) signals. The selection is determined by the output of a *Finite-State-Machine* (FSM) LOCK ORed with SE (scan enable). Furthermore, when SE is 0, the scan MUX forwards the output of an *Alternative Source* (AS), while when SE is 1 (test mode), scan MUX forwards the Scan-In signal. The secure FF has two operation modes; normal (functional mode) and scan mode. The operation modes are selected via SE signal (SE = 0 selects normal mode and SE = 1 selects scan mode). Besides test operations, the scan mode is used to unlock the secure segments of the scan-chain and thereafter to perform the classic test. As will be discussed later, AS is used to further increase the security.

The procedure to unlock the scan-chain secure segments is:

1) Set SE signal to 1.
2) Wait N clock cycles to flush out all FFs, as will be later explained.
3) Scan in pattern (i.e, potential key).
4) Set SE signal to 0 (i.e., capture).
5) LOCK FSM verifies whether the pattern scanned in step 3 (i.e, potential key) is correct. If incorrect, the segment is permanently locked, while if correct, the segment is unlocked until the circuit goes back to mission mode.
6) Repeat until all secure segments of interested are unlocked.
7) Proceed to classic testing.

This procedure is controlled by the LOCK FSM, as shown in Fig. 2. The FSM comprises six states (Initial, Flushing, Key Insertion, Key Verification, Segment Locked and Segment Unlocked) and nine control signals

- **clock** - clock signal;
- **reset** - forces the FSM to Initial state and initializes all signals;
- **counter** - decrements from N (chain total FF number) down to zero; controls the flushing of all FF scan-chain contents;
- **SOEn** - scan output enable signal is used to block the output of the scan-chain until flushing operation is completed;
- **SE** - scan enable, enables shifting-in into the scan-chain;
- **flag_prev** - flag that signals that the previous secure segment SCSS is unlocked;
- **key** - signals if the inserted key is correct;
- **flag_next** - signals that the next secure segment SCSS is ready to receive potential key;
- **lock** - output of the LOCK FSM.

At power-up the system starts in the initial state. When scan enable signal SE = 1 for N clock cycles, the contents of all the FF in all the segments are flushed out. A counter keeps track of number of clock cycles passed. When counter = 0 and flag_prev = 1, i.e., when the chain is flushed and when the previous secure segment is unlocked (signal disabled for the first SCSS), the FSM jumps to the key insertion state. In this state the scan-out enable SOEn is set 1 as no longer

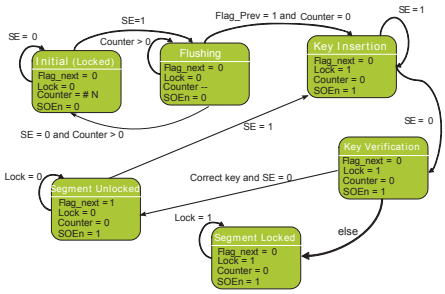


Fig. 2: LOCK Finite State Machine

sensitive information is stored. Once SE = 0, i.e, the desired key is inserted, the FSM jumps to the key verification state. If the key is correct, it unlocks all the secure FF of the SCSS and it signals that the next SCSS can be unlocked. Else, it permanently locks the segment (and therefore, blocking the unlocking possibility of the next SCSS). Once the circuit is in mission mode all the SCSS are locked once again. Regarding secure implementations of the LOCK FSM such that no information is leaked via side-channels, it would be very challenging to perform power analysis as few power traces are available. Nonetheless, if required, side-channel attack resistant FSM can be implemented [24]. Note that (a) FFs comprising the LOCK state machine are not part of the scan-chain. To test the few logic gates of the LOCK FSM we use a combination of sequential ATPG and functional test. (b) only the first LOCK FSM integrates the flushing state. The remaining FSMs skip the flushing state as the operation was already carried out.

B. Alternative Source (AS)

To make it harder for the attacker to collect any feedback from the design regarding the success of an attack, an additional feature is integrated in MSSS scheme where patterns are shifted in via AS during the attack. When a segment is locked, no data coming from its upstream logic must be stored in the FFs to avoid any possible exploitation of the secret data. Another source of data must be used instead, which depends on non-secret information only. Considering that an attacker could shift in the same test vector two or more times consecutively, a random source would decrease the overall security of the proposed method. Indeed, observing the responses he/she would conclude that when a segment is locked the output is always different (due to the AS) or constant if the introduced key is correct. To guarantee that the AS always outputs the same response for identical inputs, and therefore preventing feedback on the success of the attack, we implement the solution in Fig. 3. The proposed AS solution XORes bits from key and non-key FFs.

C. Segments Partitioning / Selecting Secure FF

Clearly, one of the key questions when designing MSSS, is how to select the FFs to be included in the segment.

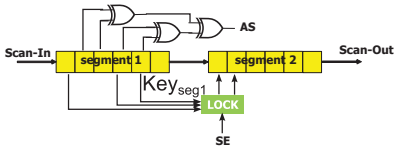


Fig. 3: AS based on key and non-key bits

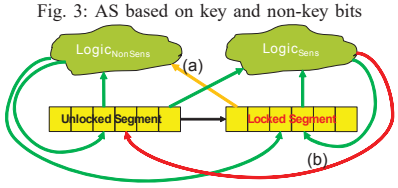


Fig. 4: Logic Dependency

Fig. 4 depicts the schematic of all network scenarios of our secure test scheme. The figure shows two scan-chain segments, one unlocked segment and one locked segment, and two combinational logic blocks, one processing non-sensitive data ($Logic_{NonSens}$) and one processing sensitive data ($Logic_{Sens}$). Considering this network, there are two unwanted scenarios. (a) an authorized user of the unlocked segment cannot test $Logic_{NonSens}$ because its inputs are fed from locked segment FF. (b) Logic that processes sensitive data and stores its outputs (even if partially) into FFs of an unlocked segment.
Scenario (a) results in reduced controllability, while scenario (b) results in reduced security. To prevent these scenarios the security engineer must carefully attribute a security level to each segment to be protected and simultaneously enforce that the inputs of combinational logic of each unlocked phase do not depend on higher security level segments.

Any commercial tool, can be used to determine the fan-in /fan-out cones, thus, highlighting any logic dependencies w.r.t. the several MSSS segments.

D. Potential Attacks & Countermeasures

A malicious user may attack our design in two ways. Next, we describe the attacks and the respective countermeasure implemented to prevent the attack.

- Shift out mission mode sensitive data. Countermeasures: allow shifting out operations only after flushing the contents of all FFs in all segments of the chain.
- Determine the segments key via brute force. Countermeasures: do not provide feedback about the success of the attack. The attacker has the entire scan-chain completely available for shifting operations. This way, the attacker does not know when he/she successfully unlocked a new segment. Moreover, we keep all the logic active via an AS. By keeping the logic active we prevent power analysis that could reveal when new segment is unlocked. In addition, if the same test vector is introduced two or more time consecutively, investigating the output, an attacker would observe that the responses would be (a) always different when a segment is locked due to the AS or (b) constant if the introduced key is correct. As a countermeasure, we implement the solution in Fig. 3, where the new AS generates its output based a combination of bits taken for the upstream scan-chain. They can equally be key bits or non-key bits. This guarantees that the same response is obtained for identical inputs, when a wrong secret key is used. Finally, as only a subset of the total FFs available is used as key bits, an attacker first needs to identify which FFs are used as key. Though this countermeasure provides security by obfuscation, it can increase the attack time.

E. Discussion

MSSS has four main advantages.

- First, it does not provide the attacker with any feedback regarding the success of an attack. When an attacker tries to unlock a certain segment, there are in principal two available ways to derive if the attack is successful: (1) shift-in a known pattern and count the number of clock cycles until the pattern is observed at the output; however, as the scan-chain has a constant length irrespective of the number of locked segments, this procedure reveals nothing on the success of the attack. (2) assume that when a segment is locked and the logic is not active, then perform power analysis to determine when the attack is successful; however, because of the patterns shifted via AS and forward to logic, the latter will be active.
- Second, the proposed scheme has a low impact on the area overhead. We add only one SMUX per scan FF, one LOCK FSM and one OR gate per SCSS and few XOR gates to exercise the logic (AS).
- Third, it is generic, it can be applied with low effort on any given design by replacing regular scan FFs by secure FFs. Moreover, it does not change the interface (i.e, does not include extra ports) nor the test procedure (once the SCSS are unlocked).
- Fourth, it has no impact on the circuit performance when compared with a classic scan-chain, as the additional multiplexer SMUX is not in the delay path. On the downside an attacker can circumvent the AS function if discovering its implementation.

IV. Experimental Analysis

In this section we investigate the proposed method in a real circuit and evaluate the method's effectiveness. First, we introduce our case-study circuit and its experimental set-up. Thereafter, we discuss the results.

A. Experimental Set-up

To evaluate MSSS effectiveness we implement the system depicted in Fig. 5. The system comprises AES and RSA crypto cores and a FSM.
Consider the following scenario where a server and a client want to securely exchange data. The server generates a RSA key (i.e., a public and a private key), handing the public key to the client. The client encrypts (a) the data with its own AES key and (b) the used AES key with the public RSA key

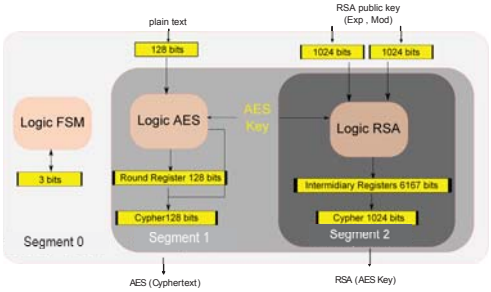


Fig. 5: Case study

received by the server. The client sends to server the AES encrypted data and the RSA encrypted AES key. The server decrypts client data by decrypting the clients AES key using its RSA private key.

We want to protect the AES key and the RSA private key. We divide our system into three segments. As the leakage of RSA private key would also compromise the AES key, we define the access to RSA as the top segment (most secure). Middle segment grants access to AES. Finally, at down segment (with no secure FFs) is the FSM, which any user can access.

We synthesize the client circuitry in VHDL in 0.35μm technology node from AMS with Synopsys Design Compiler. From the netlist of the circuit we determine the logic output cones of each FF. This information allows identifying FFs that might not be suitable to integrate the SCSS (see Section II D). Furthermore, we develop a tool that given a netlist and a technology node library outputs a new netlist including the following five additions. (1) Two new ports to enable classic scan test (scan enable and scan-in). (2) Either a new MUX (scan MUX) or two new MUXes (scan MUX and SMUX) prior to each unsecured or secured FF, respectively. (3) Connect the FFs in a scan-chain. (4) Insert AS and connect them to each SMUX input. (5) Insert LOCK per SCSS.

We implement the circuitry in Fig. 5 with:

- Key length to unlock AES secure segment: 128 bits (taken from segment 0 registers, i.e, FSM, AES plain text and RSA public key).
- Key length to unlock RSA secure segment: 128 bits (taken from segment 0 and 1 registers, i.e, in addition to the registers listed before, round and cypher register from AES).
- 10 ASs of 16 bits each (5 ASs connected to AES segment and 5 ASs connected to RSA segment).

We extract the area overhead of our solution. Note that we do not perform fault simulation as the results would be equivalent to that of a classic scan-chain. As mentioned before, the combinational logic of the LOCK FSM can be easily tested functionally by inserting the correct key into the SCSS.

TABLE II: Area results

	# FF	# Combinational	# additional gates
FSM	3	24	75
AES	387	143	364
RSA	9239	2078	63

B. Results

Our case study has no logic dependency issues. Table II shows the number of FFs and combinational logic elements each segment comprises and the number of gates added by implementing our method; e.g., FSM comprises 3 FFs, 24 logic gates and an additional 75 gates. The last row indicates the additional cost of MSSS implementation. MSSS has an area overhead of 502 gates, i.e., 4% when compared with the original circuit area overhead. Analyzing the relative impact, MSSS increases significantly the FSM area overhead, as the FSM has an inherent low area overhead. This area corresponds to the AS. The relative impact on the AES area overhead is significant. This is mainly due to the FSM counter to flush the entire scan chain. The larger the scan chain, the larger the counter. Finally, RSA has low area overhead as it comprises mainly a small FSM (not that the LOCK FSM is smaller than the LOCK FSM for the AES, as no flushing is required).

V. COMPARISON & DISCUSSION

In this section first, we compare MSSS with prior work. Thereafter, we discuss its the pros and cons. Finally, we propose solutions to further increase the security.

A. Comparison with prior work

Security can be evaluated by identifying and defining potential attacks against a design. Once a potential attack is defined, the resistance to the attack can be characterized by five main different criteria [23]:

- Time; i.e., required time to successfully perform the attack.
- Tool expertise; i.e, the required tool expertise.
- Design knowledge; i.e., device under attack knowledge.
- Samples; i.e, the required number of device samples.
- Equipment; i.e, the required equipment to set up the attack.

Table III shows a relative comparison of MSSS method against state-of-the-art methods presented in [7–21] according to the security metrics defined previously. In addition, two design metrics (area overhead and performance) are also included for comparison. In the table, MSSS is used as baseline/reference (with value 0) for comparison, while '++', '+', '−' and '−−' illustrate the relative comparison; e.g., '+' indicates 'more than' baseline (e.g., longer attack time), '−' indicates 'less than' baseline (e.g., shorter attack time).

The table shows that scan control methods, when compared with MSSS, are less secure but have less area overhead. MSSS, which is also a scan control method, in addition to a FSM to control the switching from secure mode to insecure mode, comprises several keys to gain access to the chain. As a result MSSS is more secure, however, its area overhead is larger. The impact on the circuit performance is similar. To successfully attack unauthorized shift detection methods an attacker requires a similar expertise in tools, knowledge

TABLE III: Comparison against state-of-the-art scan attacks countermeasure methods

	Security metrics					Design metrics	
	Attack time	Tool expertise	Design knowledge	# samples	Equipment	Area overhead	Impact on performance
MSSS	0	0	0	0	0	0	0
Scan control [7–10]	-	-	-	-	-	-	0
Un. shift detection [7,9,11–16]	-	0	0	-	0	0	0
Data confusion [17–21]	- -	- -	- -	- -	- -	+	0

regarding the design of the circuit to hack and equipment as to attack out method. However, the attack is carried out in less time and requiring less devices as shifting detection infrastructures are less complex, making our solution more secure. Area overhead and circuit performance are similarly impacted. Finally, data confusion methods protect via obfuscation, i.e., they assume the attacker does not know the scan-chain mapping. Therefore, MSSS provides higher security in all metrics. With respect to design metrics, data confusion methods require a large look-up-tables but they do not impact the circuit performance.

With respect to flexibility, all the state-of-the-art methods offer a binary security solution, i.e., either it is secure (according to the method) or not. MSSS is the only which enables tuning security according to the requirements.

B. Pros and cons of the solution

MSSS has several advantages; (i) flexibility to tune the security, (ii) an attacker has no feedback w.r.t. the attack success, as the scan-chain is entirely available for shifting operations and the protected logic is always active (preventing power analysis). (iii) MSSS has no impact of the circuit performance when compared to the classic scan-chain implementation and has low impact on the area overhead. (iv) MSSS is generic and does not change the standard interface nor the test methodology (after unlocking the required segments). However, MSSS must take into account logic dependency.

C. Further increasing security

MSSS security can be enhanced by limiting the key insertion attempts. This can be realized by storing the maximum number of attempts in a non-volatile memory and decrementing this number each LOCK key verification state (Fig. 2).
Moreover, MSSS can be combined with other state-of-the-art methods [7–9,11–21]. For instance, combining MSSS with an unauthorized shift detection method (e.g., spy FF [16]) would protect the SE signal.

VI. Conclusion & Future Work

We proposed a new generic multi-segment secure method to access scan-chains (MSSS). MSSS enable flexibility to tune design'security. In particular, MSSS secures the access to each scan-chain segment with a key. The segments have precedence, i.e., to gain access to a segment, all precedent segments must be unlocked. In addition, we implement a number of countermeasures to prevent leaking information with respect to the success of attack. Results show that our method is secure against brute force attacks. Moreover, the method has an inherent low area overhead and no impact on the circuit performance. We propose combining MSSS with other state-of-the-art methods to further increase the security. As future work, we will extend our solution for IEEE 1500 and JTAG standards, covering the entire hierarchy of circuit design.

References

[1] S. Hamdioui *et al.* , "Hacking and protecting IC hardware", *DATE*, pp.1-7,2014
[2] Y. Liu, K. Wu, and R. Karri, "Scan-based Attacks on Linear Feedback Shift Register Based Stream Ciphers", *ACM Trans. on Design Automation of Electronic Systems (TODAES)*, vol. 16, no. 2, Mar 2011.
[3] R. Nara *et al.* , "Scan-Based Side-Channel Attack against RSA Cryptosystems Using Scan Signatures", *IEICE Trans. on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E93-A, no.12, pp. 2481-2489, 2010.
[4] R. Nara *et al.* , "Scan-based Attack Against Elliptic Curve Cryptosystems", *ASP-DAC*, pp. 407 - 412, 2010.
[5] J. Dorolt *et al.* , "Are advanced DfT structures sufficient for preventing scan-attacks?", *VLSI Test Symposium*, pp. 246 - 251, 2012.
[6] B. Yang, K. Wu, and R. Karri, "Scan based side channel attack on dedicated hardware implementations of Data Encryption Standard", *International Test Conference*, pp. 339 - 344, 2004.
[7] B. Yang, K. Wu, and R. Karri, "Secure Scan: A Design-for-Test Architecture for Crypto Chips", *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol.25, no. 10, pp. 2287 - 2293, 2006.
[8] A. Das and Ünal Kocabaş and Ahmad-Reza Sadeghi and Ingrid Verbauwhede, "PUF-based Secure Test Wrapper Design for Cryptographic SoC Testing", Design, Automation and Test in Europe, 2012
[9] D. Hely, F. Bancel, M.-L. Flottes, B. Rouzeyre, "Securing Scan Control in Crypto Chips", *JETTA*, 23, 5, pp. 457-464, 2007
[10] J. Dorolt, G. Di Natale, M.-L. Flottes, and B. Rouzeyre, "Thwarting Scan-based Attacks on Secure-ICs with on-chip comparison", *IEEE Trans. on VLSI Systems*, 2013
[11] D. Hely et al., "Scan Pattern Watermarking, *LATW*, 2006
[12] J. Lee, M. Tebranipoor, and J. Plusquellic, "A low-cost solution for protecting IPs against scan-based side-channel attacks", *Proc. VTS*, pp.93-99, 2006.
[13] S. Paul, R.S. Chakraborty, S. Bhunia, "VIm-Scan: A Low Overhead Scan Design Approach for Protection of Secret Key in Scan-Based Secure Chips", *Proc. VTS*, pp.455-460, 2007.
[14] D. Hely, F. Bancel, M.-L. Flottes, B. Rouzeyre, "Test Control for Secure Scan Designs", *Proc. ETS*, pp. 190-195, 2005.
[15] D. Hely, "Testability of Secure ICs", *PhD report University of Montpellier 2*, 2005
[16] D. Hely, F. Bancel, M.-L. Flottes and B. Rouzeyre, "A secure Scan Design Methodology", Design, Automation and Test in Europe, 2006.
[17] G. Sengar, D. Mukhopadhyay, and D.R. Chowdhury, "Secured Flipped Scan-Chain Model for Crypto-Architecture", *IEEE Trans. on CAD*, vol. 26, no.11, pp.2080-2084, Nov. 2007.
[18] D. Mukhopadhyay *et al.* "CryptoScan: A Secured Scan Chain Architecture", *Proc. 14th ATS*, pp. 348-353, 2005.
[19] D. Hely *et al.* ,"Scan design and secure chip [secure IC testing]", *Proc. IOLTS*, pp. 219-224, 2004.
[20] J. Lee, M. Tehranipoor, C. Patel, and J. Plusquellic, "Securing Designs against Scan-Based Side-Channel Attacks", *IEEE Trans. on Dependable and Secure Computing*, vol.4, no.4, pp.325-336, 2007.
[21] J. Lee, M. Tehranipoor, C. Patel and J. Plusquellic, "Securing Scan Design Using Lock and Key Technique", *IEEE Int. Symp. on Defect and Fault Tolerance in VLSI System*, 2005
[22] A. Zygmontowicz, J. Dworak, A. Crouch, and J. Potter, "Making it harder to unlock an LSIB: Honeytraps and misdirection in a P1687 network", *DATE*, pp.1 - 6, 2014
[23] Application of Attack Potential to Smart-Card, Common Criteria, "www.commoncriteriaportal.org"
[24] M. Borowczak and R. Vemuri, "S*FSM: A Paradigm Shift for Attack Resistant FSM Designs and Encodings", *ASE/IEEE International Conference on BioMedical Computing (BioMedCom)*, 2012.

4

CONCLUSION

This chapter summarizes the overall achievements of this dissertation and highlights some future research directions. Section 4.1 presents a summary of the main conclusions of the presented in this dissertation. Section 4.2 provides the future research directions.

4.1. SUMMARY

Chapter 1, "Introduction", briefly introduced the field of Hardware Security. It described the motivation behind IT attacks and provided a classification of such attacks. In addition, it focused on physical attacks, which are those targeting the hardware, providing a classification of these type of attacks. Hardware Security is the foundation of secure devices; it is an ever changing field due to the constant dynamic between the new proposed attacks and their respective countermeasures. The field's main challenges include the design, manufacturing, test and reliability of secure devices. This thesis focused on two of such challenges: one related to reliability and one to test.

Chapter 2, "Reliability Characterization and Improvement for Secure ICs", investigated the parameters impacting memory-PUF robustness (i.e., fingerprint repeatability or stability and uniqueness) and proposed schemes to improve it. First, considering the SRAM PUF, a classification based on the asymmetry of the SRAM cells was proposed. Furthermore, an SRAM PUF stability mathematical model based on the SRAM cell's transistor electrical equations was developed. Based on the equations, the parameters impacting the stability was determined and further classified into technology (e.g., transistor threshold voltage) and non-technology parameters (e.g., temperature). In addition, the mathematical model was first used to perform a sensitivity analysis determining the most dominant parameters, and thereafter used to statistically determine the probability of SRAM devices being stable. The model was validated with silicon experiments. Complementary, the impact of the design, more specifically low-power and general purpose designs, was investigated via both simulations and silicon experiments. Based on the obtained results combined with prior work, the pros and cons of the two considered design is discussed including robustness, power consumption, area overhead and security.

Finally, a scheme to improve memory-PUF robustness is proposed. Based on the before mentioned work, it is known that temperature and voltage ramp-up time are the non-technology parameters impacting the most memory-PUF robustness. Therefore, SPICE circuit simulations were carried out varying these parameters for a wide range of values. The results revealed a negative correlation between temperature and voltage ramp-up time on the repeatability of the memory fingerprint; i.e., it was observed that fingerprints were more repeatable when either a the fingerprint was reconstructed at low temperature combined with long voltage ramp-up time or at high temperature with a short voltage ramp-up time. In addition, silicon measurements were carried out on three different types of memory-PUFs distributed over three technology nodes; their analysis validate the observed simulation results. To evaluate the added value of integrating these findings into a commercial product, a cost analysis based on area overhead is performed, revealing that, depending of the PUF type, major area savings can be realized. Furthermore, a circuit which outputs a voltage ramp-up time according to the sensed environment temperature is designed. Finally, the overall impact of implementing the proposed solution is evaluated; considering the saved memory area overhead together with the additional area overhead introduced by the adapter circuit

and the increase in robustness, it was proven that the scheme is both efficient and commercially attractive.

Chapter 3, "Testing Secure Devices", proposed test schemes for secure circuits, initially targeting a generic solution and later with focus on PUF-based systems. The test scheme proposes a generic and flexible (i.e., tunable security) secure solution. The idea is to tune the security level according to the requirements of the application. Conducted work focus on the solution at the core level. The security is enabled by Multiple-Segment Secure Scan (MSSS), which comprises L levels on scan-chain secure segments (SCSS). Each SCSS is protected by a secret key and it can be unlocked only when all its precedents SCSS are unlocked. The requirements that a flip-flop needs to fulfill to be suitable to comprise a SCSS are investigated. In addition, several countermeasures against known attacks are integrated. A case study comprising two crypto cores (AES and RSA) is implemented. The results show that the test scheme delivers a higher level of security than the state-of-the-art at the cost of 4% increase of area overhead when compared with the original circuit.

The development of secure test schemes for PUF-based systems, more specifically for Fuzzy Extractor (FE), involved a number of steps. First, a requirement and assumption list was defined. Second, two secure test schemes were proposed; both schemes use functional BIST, are scan-chain free (preventing scan-attacks), use RTPGs and target high stuck-at-fault coverage. The first scheme is designed to minimize area overhead by reusing FE building blocks as RTPG, while the second scheme is designed to minimize the test time by inserting dedicated RTPGs per FE block. Third, techniques targeting test quality improvement using RTPGs were studied and some of them were deployed to further increase the fault coverage, which was limited by the error correcting capabilities of the FE. Thereafter, the system was implemented and simulated in order to investigate the impact of various parameters on both test time and fault coverage. Finally, the results were compared with the state-of-the-art, and a security analysis was made in order to provide appropriate recommendations.

4.2. FUTURE RESEARCH DIRECTIONS

Several recommendations are suggested to further research some aspects of topics addressed in this dissertation. They are given next.

- **PUF Technology:** with the long predicted end of the classic CMOS technology (Moore's law), many alternative technologies are being proposed, such as 3D CMOS and memristors. These new technologies will deeply impact the design of hardware security, in particular, of hardware security primitives such as PUFs. Next, we briefly highlight a none exhaustive list of future work possibilities with respect to PUF technology.

    1. Investigate the quality of 3D memory-PUF. There have been reported attacks cloning SRAM PUFs which require access to the memory substrate. A possible countermeasure is to use 3D stacked memories as PUF. A clear advantage

is that middle layers are intrinsically protected. To tamper them, the attacker would need to destroy either the top or bottom layers, destroying the PUF.

2. Assess and validate the quality of memristor PUFs. A lot of investigation needs to be carried out before this technology can be declared a suitable for PUFs.
3. Delay-based PUFs are impacted by the surrounding logic. However, this phenomena is not yet investigated for memory-based PUFs. In particular the impact of the memory geometry when combined with the surrounding logic; the geometry of the memory impacts the distance of the surrounding logic to the memory cell array, as well as peripheral circuitry.

- **Security Design:** is an ever evolving field, increasingly complex due to the rising number of attack possibilities. Next, we suggest some future work possibilities within security design.

  1. Development of Electronic Design Automation (EDA) tools. Currently, secure circuits are designed by hand. The industry would benefits from tools to support secure design with such tools, to at least, prevent the success of known attacks.
  2. Improve diagnosability in BIST test schemes. When no additional security measures are implemented, BIST schemes are more secure than external schemes; however, the latter has higher diagnosability. A generic scheme to improve BIST diagnosability without compromising its security is needed.

Other than the open challenges discussed in this chapter, there are many others related with Hardware Security (see Section 1.2).

REFERENCES

[1] R.N. Akram, K. Markantonakis, and K. Mayes. User Centric Security Model for Tamper-Resistant Devices. In *e-Business Engineering (ICEBE), 2011 IEEE 8th International Conference on*, pages 168–177, Oct 2011.

[2] S.S. Ali, O. Sinanoglu, S.M. Saeed, and R. Karri. New scan attacks against state-of-the-art countermeasures and DFT. In *IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, pages 142–147, May 2014.

[3] Kwang-Hyun Baek and S.W. Smith. Preventing theft of quality of service on open platforms. In *Security and Privacy for Emerging Areas in Communication Networks, 2005. Workshop of the 1st International Conference on*, pages 246–257, Sept 2005.

[4] R. Baranowski, M.A. Kochte, and H.-J. Wunderlich. Securing Access to Reconfigurable Scan Networks. In *Test Symposium (ATS), 2013 22nd Asian*, pages 295–300, Nov 2013.

[5] M. Bhargava, C. Cakir, and K. MAI. Attack resistant sense amplifier based PUFs (SA-PUF) with deterministic and controllable reliability of PUF responses. In *Hardware-Oriented Security and Trust (HOST), 2010 IEEE International Symposium on*, pages 106–111, June 2010.

[6] R.K.C. Chang. Defending against flooding-based distributed denial-of-service attacks: a tutorial. *Communications Magazine, IEEE*, 40(10):42–51, Oct 2002.

[7] M. Cortez, A. Dargar, S. Hamdioui, and G.-J. Schrijen. Modeling SRAM Start-Up Behavior for Physical Unclonable Functions. In *Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), 2012 IEEE International Symposium on*, pages 1–6, Oct 2012.

[8] M. Cortez, S. Hamdioui, and R. Ishihara. Design Dependent SRAM PUF Robustness Analysis. In *16th Latin-American Test Symposium*, 2015.

[9] M. Cortez, S. Hamdioui, A. Kaichouhi, V. vd Leest, R.Maes, and G.-J. Schrijen. Intelligent Voltage Ramp-up Time Adaptation for Temperature Noise Reduction on Memory-based PUF Systems. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2015.

[10] M. Cortez, S. Hamdioui, V. van der Leest, R. Maes, and G.-J. Schrijen. Adapting Voltage Ramp-up Time for Temperature Noise Reduction on Memory-based PUFs. In *Hardware-Oriented Security and Trust (HOST), 2013 IEEE International Symposium on*, pages 35–40, June 2013.

[11] M. Cortez, G. Roelofs, S. Hamdioui, and G. Di Natale. Testing PUF-based Secure Key Storage Circuits. In *Design, Automation and Test in Europe Conference and Exhibition (DATE), 2014*, pages 1–6, March 2014.

[12] Mafalda Cortez, Said Hamdioui, Giorgio Di Natale, Marie-Lise Flottes, Bruno Rouzeyre, and Ilia Polian. Multi-segment Attack-resistant DFT for Secure ICs. In *to be submitted*, 2015.

[13] Mafalda Cortez, Gijs Roelofs, Said Hamdioui, and Giorgio Di Natale. Testing Methods for PUF-Based Secure Key Storage Circuits. *Journal of Electronic Testing*, 30(5):581–594, 2014.

[14] J. Da Rolt, A. Das, G. Di Natale, M. Flottes, B. Rouzeyre, and I. Verbauwhede. A scan-based attack on Elliptic Curve Cryptosystems in presence of industrial Design-for-Testability structures. In *Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT), 2012 IEEE International Symposium on*, pages 43–48, Oct 2012.

[15] J. Da Rolt, G. Di Natale, M.-L. Flottes, and B. Rouzeyre. A smart test controller for scan chains in secure circuits. In *On-Line Testing Symposium (IOLTS), 2013 IEEE 19th International*, pages 228–229, July 2013.

[16] Jean Da Rolt, Amitabh Das, Giorgio Di Natale, Marie-Lise Flottes, Bruno Rouzeyre, and Ingrid Verbauwhede. A new scan attack on rsa in presence of industrial countermeasures. In Werner Schindler and SorinA. Huss, editors, *Constructive Side-Channel Analysis and Secure Design*, volume 7275 of *Lecture Notes in Computer Science*, pages 89–104. Springer Berlin Heidelberg, 2012.

[17] A. Das, U. Kocabas, A. Sadeghi, and I. Verbauwhede. PUF-based secure test wrapper design for cryptographic SoC testing. In *Design, Automation Test in Europe Conference Exhibition (DATE), 2012*, pages 866–869, March 2012.

[18] International Organization for Standarization (ISO) and International Electrotechnical Commission (IEC). ISO/IEC 27000 - Information technology - Security techniques - Information security management systems - Overview and vocabulary. Technical Report ISO/IEC 27000:2009(E), ISO/IEC, May 2009.

[19] C. Helfmeier, C. Boit, D. Nedospasov, and J.-P. Seifert. Cloning Physically Unclonable Functions. In *Hardware-Oriented Security and Trust (HOST), 2013 IEEE International Symposium on*, pages 1–6, June 2013.

[20] D. Hely, M.-L. Flottes, F. Bancel, B. Rouzeyre, N. Berard, and M. Renovell. Scan design and secure chip [secure IC testing]. In *On-Line Testing Symposium, 2004. IOLTS 2004. Proceedings. 10th IEEE International*, pages 219–224, July 2004.

[21] S.U. Hussain, S. Yellapantula, M. Majzoobi, and F. Koushanfar. BIST-PUF: Online, Hardware-based Evaluation of Physically Unclonable Circuit Identifiers. In *Computer-Aided Design (ICCAD), 2014 IEEE/ACM International Conference on*, pages 162–169, Nov 2014.

[22] A. Kaminsky, M. Kurdziel, and S. Radziszowski. An overview of cryptanalysis research for the advanced encryption standard. In *MILITARY COMMUNICATIONS CONFERENCE, 2010 - MILCOM 2010*, pages 1310–1316, Oct 2010.

[23] I. Kotenko, M. Stepashkin, and E. Doynikova. Security Analysis of Information Systems Taking into Account Social Engineering Attacks. In *Parallel, Distributed and Network-Based Processing (PDP), 2011 19th Euromicro International Conference on*, pages 611–618, Feb 2011.

[24] F. Koushanfar. Provably secure active ic metering techniques for piracy avoidance and digital rights management. *Information Forensics and Security, IEEE Transactions on*, 7(1):51–63, Feb 2012.

[25] I. Koyuncu, A.T. Ozcerit, I. Pehlivan, and E. Avaroglu. Design and implementation of chaos based true random number generator on FPGA. In *Signal Processing and Communications Applications Conference (SIU), 2014 22nd*, pages 236–239, April 2014.

[26] S.H.M. Kwok and E.Y. Lam. FPGA-based High-speed True Random Number Generator for Cryptographic Applications. In *TENCON 2006. 2006 IEEE Region 10 Conference*, pages 1–4, Nov 2006.

[27] L. Laribee, D.S. Barnes, N.C. Rowe, and C.H. Martell. Analysis and Defensive Tools for Social-Engineering Attacks on Computer Systems. In *Information Assurance Workshop, 2006 IEEE*, pages 388–389, June 2006.

[28] T. Machida, D. Yamamoto, M. Iwamoto, and K. Sakiyama. A new mode of operation for arbiter PUF to improve uniqueness on FPGA. In *Computer Science and Information Systems (FedCSIS), 2014 Federated Conference on*, pages 871–878, Sept 2014.

[29] Abhranil Maiti, Raghunandan Nagesh, Anand Reddy, and Patrick Schaumont. Physical Unclonable Function and True Random Number Generator: A Compact and Scalable Implementation. In *Proceedings of the 19th ACM Great Lakes Symposium on VLSI*, GLSVLSI '09, pages 425–428, New York, NY, USA, 2009. ACM.

[30] B. Mazumdar, D. Mukhopadhyay, and I. Sengupta. Design for Security of Block Cipher S-Boxes to Resist Differential Power Attacks. In *VLSI Design (VLSID), 2012 25th International Conference on*, pages 113–118, Jan 2012.

[31] R. Modugu, Yong-Bin Kim, and Minsu Choi. Design and performance measurement of efficient IDEA (International Data Encryption Algorithm) crypto-hardware using novel modular arithmetic components. In *Instrumentation and Measurement Technology Conference (I2MTC), 2010 IEEE*, pages 1222–1227, May 2010.

[32] F. Mouton, M.M. Malan, L. Leenen, and H.S. Venter. Social engineering attack framework. In *Information Security for South Africa (ISSA), 2014*, pages 1–9, Aug 2014.

[33] TU Munchen. http://www.sec.ei.tum.de/uploads/pics/microprobing_02.jpg. [Online; accessed 27-February-2015].

[34] Radu Muresan and S. Gregori. Protection Circuit against Differential Power Analysis Attacks for Smart Cards. *Computers, IEEE Transactions on*, 57(11):1540–1549, Nov 2008.

[35] Cuong Nguyen, Lai Tran, and Khoa Nguyen. On the resistance of Serpent-type 4 bit S-boxes against differential power attacks. In *Communications and Electronics (ICCE), 2014 IEEE Fifth International Conference on*, pages 542–547, July 2014.

[36] Mark Pellegrini. Differential power analysis. http://commons.wikimedia.org/wiki/File:Differential_power_analysis.svg#/media/File:Differential_power_analysis.svg, 2009. [Online; accessed 27-February-2015].

[37] M.T. Rahman, D. Forte, J. Fahrny, and M. Tehranipoor. ARO-PUF: An aging-resistant ring oscillator PUF design. In *Design, Automation and Test in Europe Conference and Exhibition (DATE), 2014*, pages 1–6, March 2014.

[38] M. Rostami, F. Koushanfar, J. Rajendran, and R. Karri. Hardware security: Threat models and metrics. In *Computer-Aided Design (ICCAD), 2013 IEEE/ACM International Conference on*, pages 819–823, Nov 2013.

[39] S.M. Saeed, S.S. Ali, O. Sinanoglu, and R. Karri. Test-mode-only scan attack and countermeasure for contemporary scan architectures. In *Test Conference (ITC), 2014 IEEE International*, pages 1–8, Oct 2014.

[40] D.P. Sahoo, S. Saha, D. Mukhopadhyay, R.S. Chakraborty, and H. Kapoor. Composite PUF: A new design paradigm for Physically Unclonable Functions on FPGA. In *Hardware-Oriented Security and Trust (HOST), 2014 IEEE International Symposium on*, pages 50–55, May 2014.

[41] Luis F. G. Sarmenta, Marten Van Dijk, Charles W. Odonnell, Jonathan Rhodes, and Srinivas Devadas. Virtual Monotonic Counters and Count-Limited Objects using a TPM without a Trusted OS. In *In Proceedings of the 1st ACM CCS Workshop on Scalable Trusted Computing (STC06*, pages 27–42, 2006.

[42] Kudelski Security. http://www.kudelskisecurity.com/sites/val/files/lab-setup-full.jpg. [Online; accessed 27-February-2015].

[43] Y. Shi, Nozomu Togawa, M. Yanagisawa, and T. Ohtsuki. Design-for-secure-test for crypto cores. In *Test Conference, 2009. ITC 2009. International*, pages 1–1, Nov 2009.

[44] J. Singh, R. Ruhl, and D. Lindskog. GSM OTA SIM Cloning Attack and Cloning Resistance in EAP-SIM and USIM. In *Social Computing (SocialCom), 2013 International Conference on*, pages 1005–1010, Sept 2013.

[45] Sergei Skorobogatov. Semi-invasive attacks - A new approach to hardware security analysis. Technical Report UCAM-CL-TR-630, University of Cambridge, Computer Laboratory, April 2005.

[46] Sergei P. Skorobogatov and Ross J. Anderson. Optical Fault Induction Attacks. In *Revised Papers from the 4th International Workshop on Cryptographic Hardware and Embedded Systems*, CHES '02, pages 2–12, London, UK, UK, 2003. Springer-Verlag.

[47] Pim Tuyls, Boris Skoric, and Tom Kevenaar. *Security with Noisy Data: Private Biometrics, Secure Key Storage and Anti-Counterfeiting*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2007.

[48] M. Varchola, M. Drutarovsky, and V. Fischer. New universal element with integrated PUF and TRNG capability. In *Reconfigurable Computing and FPGAs (ReConFig), 2013 International Conference on*, pages 1–6, Dec 2013.

[49] V.M. Weaver and S.A. McKee. Can hardware performance counters be trusted? In *Workload Characterization, 2008. IISWC 2008. IEEE International Symposium on*, pages 141–150, Sept 2008.

[50] A. Wood and J.A. Stankovic. Denial of service in sensor networks. *Computer*, 35(10):54–62, Oct 2002.

[51] Jiliang Zhang, Qiang Wu, Yongqiang Lyu, Qiang Zhou, Yici Cai, Yaping Lin, and Gang Qu. Design and Implementation of a Delay-Based PUF for FPGA IP Protection. In *Computer-Aided Design and Computer Graphics (CAD/Graphics), 2013 International Conference on*, pages 107–114, Nov 2013.

LIST OF PUBLICATIONS

International Journals

2. **M. Cortez**, S. Hamdioui, A. Kaichouhi, V. vd Leest, R. Maes, G.-J. Schrijen, *Intelligent Voltage Ramp-up Time Adaptation for Temperature Noise Reduction on Memory-based PUF Systems*, IEEE Transactions on Computed Aided Design of Integrated Circuits and Systems (TCAD), pp. 1162-1175, volume 34, issue 7, July 2015.

1. **M. Cortez**, G. Roelofs, S. Hamdioui, G. Di Natale, *Testing Methods for PUF-Based Secure Key Storage Circuits*, Journal of Electronic Testing: Theory and Applications (JETTA), pp. 581-594, volume 30, issue 5, October 2014.

International Symposiums and Conferences

5. **M. Cortez**, S. Hamdioui, G. Di Natale, M.-L. Flottes, B. Rouzeyre, I. Polian, *Multi-segment Attack-resistant DFT for Secure ICs*, to be submitted.

4. **M. Cortez**, S. Hamdioui, R. Ishihara, *Design Dependent SRAM PUF Robustness Analysis*, Latin-American Test Symposium (LATS), pp. 1-6, 25-27 March 2015, Puerto Vallarta, Mexico.

3. **M. Cortez**, G. Roelofs, S. Hamdioui, G. Di Natale, *Testing PUF-Based Secure Key Storage Circuits*, Design, Automation & Test in Europe (DATE), pp. 1-6, 24-28 March 2014, Dresden, Germany.

2. **M. Cortez**, S. Hamdioui, V. vd Leest, R. Maes, G.-J. Schrijen, *Adapting Voltage Ramp-up Time for Temperature Noise Reduction on Memory-based PUFs*, IEEE International Symposium on Hardware-Oriented Security and Trust (HOST), pp. 35-40, 2-3 June 2013, Austin, TX, USA.

1. **M. Cortez**, A. Dargar, S. Hamdioui, G.-J. Schrijen, *Modeling SRAM Start-Up Behavior for Physical Unclonable Functions*, IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFTS), pp. 1-6, 3-5 October 2012, Austin, TX, USA. **Best Student Paper Award**

Workshops

5. **M. Cortez**, S. Hamdioui, G. Di Natale, M.-L. Flottes, B. Rouzeyre, I. Polian, *Multi-segment Enhanced Scan-chains for Secure ICs*, Workshop on Trustworthy Manufacturing and Utilization of Secure Devices (TRUDEVICE) co-located with Workshop on Cryptographic Hardware and Embedded Systems (CHES), 17 September 2015, Saint-Malo, France.

4. **M. Cortez**, S. Hamdioui, G. Di Natale, M.-L. Flottes, B. Rouzeyre, *Hierarchical Secure DFT*, Workshop on Trustworthy Manufacturing and Utilization of Secure Devices (TRUDEVICE) co-located with Design, Automation & Test in Europe (DATE), 13 March 2015, Grenoble, France.

3. **M. Cortez**, G. Roelofs, S. Hamdioui, G. Di Natale, *Secure Test Method for Fuzzy Extractor*, Joint MEDIAN-TRUDEVICE Open Forum, 30 September 2014, Amsterdam, the Netherlands.

2. **M. Cortez**, S. Hamdioui, V. van der Leest, R. Maes, G.J. Schrijen, *Noise Reduction on Memory-based PUFs*, 1st Workshop on Trustworthy Manufacturing and Utilization of Secure Devices (TRUDEVICE) co-located with IEEE European Test Symposium (ETS), 30-31 May 2013, Avignon, France.

1. **M. Cortez**, V. van der Leest, G.-J. Schrijen, S. Hamdioui, *Investigation of Voltage Ramp-up Time for Temperature Noise Reduction on Memory-based PUFs*, Sentinels - Security and Privacy at ICT.OPEN 2012, 22-23 October 2012, Rotterdam, the Netherlands.

CURRICULUM VITÆ

Ana Mafalda Monteiro Oliveira Cortez was born in the city of Porto, Portugal in June 1984. She concluded her M.Sc. in Microelectronics and Embedded Systems (5 years degree with a major in Electrical and Computers Engineering - Telecommunications) from the *Faculdade de Engenharia da Universidade do Porto* (FEUP) in 2009. During her master's education, she did her graduation thesis at NXP Semiconductors Research, in Eindhoven, the Netherlands, entitled "Electrical Characterization and Interpretation of MEMS Microphones with Spring Suspended Backplates". In 2010 she joined the Computer Engineering Lab at the Faculty of Electrical Engineering, Mathematics, Computer Science at the Delft University of Technology, to pursue the Ph.D. degree under the supervision of dr. ir. Said Hamdioui. Her Ph.D. project was a collaboration with Intrinsic-ID and supported by the Dutch "Point One Program" under the RATE (Reliability Assessment and test methods for anti-counterfeiting TEchnology) project (project number PNU09C09). In addition, within the framework of the European ICT COST Action TRUDEVICE (IC1204), she received an STSM (Short Term Scientific Mission) grant to further explore secure DFT schemes in collaboration with the Department of Microelectronics at the *Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier* (LIRMM), in Montpellier, France, where she spent 4 months. Her research interests include Circuit Design and Modeling, Hardware Security and Secure IC Test. Currently, Mafalda is a Security Analyst at Riscure B.V., in Delft, the Netherlands.