Secure Computing Systems using Emerging Technologies

Gokulnath Rajendran

College of Computing & Data Science

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

2025

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research, is free of plagiarised materials, and has not been submitted for a higher degree to any other University or Institution.

12/01/2025

Date

Gokulnath Rajendran

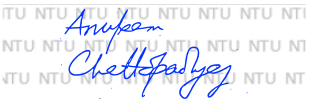
Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it is free of plagiarism and of sufficient grammatical clarity to be examined. To the best of my knowledge, the research and writing are those of the candidate except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

12/01/2025

.....

Date



.....

A/Prof. Anupam Chattopadhyay

Authorship Attribution Statement

This thesis contains material from 5 papers published in the following peer-reviewed journals or conferences in which I am listed as an author.

Chapters 2-7 are published as Rajendran, G., Ravi, P., D'anvers, J. P., Bhasin, S., Chattopadhyay, A. (2023). Pushing the limits of generic side-channel attacks on LWE-based KEMs-parallel PC oracle attacks on Kyber KEM and beyond. IACR Transactions on Cryptographic Hardware and Embedded Systems.

The contributions of the co-authors are as follows:

- I proposed the ideas and carried out all the simulations and experiments. Dr. Prasanna Ravi oversaw the project.
- Dr. D'anvers, J. P assisted with security strength calculation.
- I wrote the drafts of the manuscript together with Dr. Prasanna Ravi, Dr. Shivam Basin and A/Prof Anupam.

Chapter 8 is published as Rajendran, G., Banerjee, W., Chattopadhyay, A., Aly, M. M. S. (2021). Application of resistive random access memory in hardware security: A review. Advanced Electronic Materials, 7(12), 2100536.

The contributions of the co-authors are as follows:

- I prepared the manuscript drafts. The manuscript was revised by A/Prof Anupam, and Dr. Writam Banerjee

Chapter 9 is partially published as Rajendran, G., Zahoor, F., Singh, S., Merchant, F., Rana, V., Chattopadhyay, A. (2023, October). PR-PUF: A Reconfigurable Strong RRAM PUF. In 2023 IFIP/IEEE 31st International Conference on Very Large Scale Integration (VLSI-SoC) (pp. 1-6). IEEE.

The contributions of the co-authors are as follows:

- I proposed the ideas and carried out the simulations.
- Dr. Debajit Basak designed the comparator used for the analysis.
- I wrote the drafts of the manuscript. The manuscript was revised together with A/Prof Anupam.

Chapter 10 is published as Rajendran, G., Zahoor, F., Thakker, S. S., Singh, S., Merchant, F., Rana, V., Chattopadhyay, A. (2024, January). Harnessing Entropy: RRAM Crossbar-based Unified PUF and RNG. In 2024 37th International Conference on VLSI Design and 2024 23rd International Conference on Embedded Systems (VLSID) (pp. 560-564). IEEE.

The contributions of the co-authors are as follows:

- I proposed the ideas and carried out the simulations.
- I wrote the drafts of the manuscript. The manuscript was revised together with A/Prof Anupam.

Chapters 11 is published as published as Rajendran, G., Basak, D., Deb, S., Chattopadhyay, A. (2024, July). Securing Binarized Neural Networks via PUF-based Key Management in Memristive Crossbar Arrays. Accepted for publication in IEEE Embedded Systems Letters (ESL).

The contributions of the co-authors are as follows:

- I proposed the ideas and carried out the simulations.
- Dr. Debajit Basak designed the comparator used for the analysis.
- I wrote the drafts of the manuscript. The manuscript was revised together with A/Prof Anupam and Dr. Suman Deb.

27/06/2024

.....

Date

.....

Gokulnath Rajendran

Acknowledgements

I would like to express my deepest gratitude to my esteemed advisor, Professor Anupam Chattopadhyay, for his unwavering support and invaluable guidance throughout the course of my Ph.D. research. I am also deeply thankful to Dr. Prasanna Ravi, Dr. Furqan Zahoor, and Dr. Suman Deb for accompanying me on this academic journey. Their expertise and encouragement have played a crucial role in shaping the outcome of this thesis.

Abstract

The rapid progress in computational technology has led to the extensive generation and processing of massive datasets, driving accelerated advancements. Energy-efficient, high-capacity computational devices, integrated with intelligent algorithms, can now operate in any environment. These devices are interconnected using advanced communication technologies, contributing to the concept of the Internet of Things (IoT). Ongoing research is geared towards developing efficient devices and circuits to enhance computational speed while maximizing energy and spatial efficiency. However, the commercial deployment of these computational units necessitates significant financial investments and collaboration among vendors at various levels. As participants seek to safeguard their contributions from reverse engineering, the hardware systems cater to a diverse user base with varying security needs. The focus is currently on the Trusted Execution Environment (TEE) as a means to protect runtime states and stored data during execution, based on the Hardware Root of Trust (HROT) made up of hardware components that establish trust. Common threats include side and covert channel attacks, fault attacks, hardware trojans, reverse engineering techniques, unintentional design flaws, and bugs. HROT development requires specialized hardware cryptographic primitives and accelerators. Moreover, the exploration of emerging device technologies presents opportunities for the development of new kinds of hardware cryptographic primitives and accelerators for future intelligent hardware systems.

The first purpose of this thesis is to assess the potential security vulnerabilities stemming from the use of commercial hardware in the deployment of cryptographic schemes. The focus is particularly on the study of side-channel attacks on the implementation of post-quantum cryptography (PQC) in hardware. The decision by NIST to select CRYSTALS-Kyber as the exclusive candidate for standardizing Key Encapsulation Mechanisms (KEMs) in the third round of the PQC standardization process has paved the way for widespread integration of Kyber KEM across diverse computational platforms and applications. NIST has urged for a more thorough

evaluation of the security of PQC schemes against side-channel attacks, actively prompting the cryptographic community to investigate new attacks on lattice-based schemes like Kyber KEM and to devise effective side-channel protection methods. We present novel side-channel attacks that can efficiently uncover the secret key of the Kyber KEM using fewer queries. Our experiments demonstrated enhancements of around 2.89× and 7.65× in query counts compared to currently available binary PC oracle attacks. Furthermore, a determined attacker could achieve even greater improvements due to the adaptable nature of the proposed methods. Our proposed attacks are accompanied by a comprehensive discussion, analysis, and experimental validation.

The majority of contemporary device authentication and data transfer protocols predominantly utilize security measures implemented through software, which are resource-intensive and susceptible to persistent attackers with significant computational power. This underscores the inadequacy of software-based security mechanisms for resource-constrained IoT devices, emphasizing the imperative for robust hardware-based security solutions. The intricate and globally distributed nature of the semiconductor industry's supply chain necessitates rigorous scrutiny and assurance management throughout fabrication and system integration processes. Presently, research in hardware security predominantly focuses on the development of security primitives that exploit manufacturing variations to address these challenges. This pursuit has given rise to advancements in hardware security primitives such as True Random Number Generators (TRNG) and Physical Unclonable Functions (PUF). The utilization of CMOS technologies to establish PUFs requires the employment of error correction codes to guarantee reliable and unique outputs. Moreover, the generation of multi-bit responses using CMOS technologies poses a noteworthy technical challenge. In-memory computing architectures, which are based on emerging non-volatile memory technologies like Resistive Random Access Memory (RRAM), are currently under exploration due to their superior computational performance within constrained resource environments. Furthermore, these technologies hold promise in providing foundational security primitives. In this dissertation, we conduct an extensive examination of the most recent developments in device technologies, particularly RRAM, with the goal of utilizing them for the creation of hardware security primitives. We introduce novel implementations for PUF and RNG utilizing memristive crossbars. Our focus is not only on enhancing performance but also on addressing critical challenges in hardware security

attributes, such as writing-free reconfiguration and unified design. We undertook an in-depth analysis of the characteristics of our proposed designs, utilizing standard evaluation criteria, including the NIST SP 800-22 test suite. Furthermore, we systematically evaluated their resilience against machine learning attacks, ensuring comprehensive assessment of the effectiveness of the designs.

Binarized neural networks (BNNs) represent a specialized class of deep neural networks designed to operate with reduced computational and energy requirements. Recent research indicates that memristor-based in-memory computing architectures have the potential to enhance the performance of BNNs compared to conventional CMOS technologies. However, the non-volatile characteristics of memristors in in-memory computing give rise to concerns about potential security vulnerabilities, particularly in the event of physical access by malicious entities. Our final contribution in this thesis is on utilizing newly developed hardware cryptomodules to safeguard the trained model parameters of BNN when deployed in advanced RRAM in-memory computing accelerators. We explore various protection methodologies, detailing their respective circuit-level hardware designs and associated overheads.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

The recent proliferation of advanced computing is propelling the world towards constant generation and analysis of massive amounts of data, leading to accelerated new breakthroughs [19]. Our current technological progress allows for the deployment of energy-efficient, high computational capacity devices embedded with intelligent algorithms in virtually any setting. The progress in communication technologies has enabled the connection of such devices to establish device-to-device communication on a larger scale. We are categorizing this entire advancement as an emerging paradigm of Internet of Things (IoT). Current research efforts are primarily directed towards the development of devices and circuits aimed at accelerating computation speed, all the while maintaining energy and area efficiency. The commercialization of these computational units requires substantial financial investments, often leading to collaboration among vendors at various levels of trust in the manufacturing process [20]. As a result, vendors involved in this development view their contribution as assets and are compelled to protect their assets from reverse engineering.

The diverse user base of the hardware system spans various backgrounds and requirements, all unified in their demand for robust security measures to safeguard their critical data. This brings us to the notion of trusted computing. The current area of focus is the Trusted Execution Environment (TEE), which is being explored to safeguard both runtime states and stored data during execution [21]. In the development of TEE, the essential foundation is the Hardware Root of Trust (HROT), comprising hardware components that serve as the fundamental anchors

for establishing trust [22]. The general threats encompass commonly explored side and covert channel attacks, fault attacks, and hardware trojans [20]. Additionally, evolving reverse engineering techniques and the presence of unintentional design flaws and bugs also pose significant threats [23]. To develop HROT, we need various hardware cryptographic primitives and accelerators with specialized designs. The ongoing exploration of emerging device technologies for the design of future intelligent hardware systems presents numerous opportunities to develop new types of hardware cryptographic primitives and accelerators.

NIST has selected CRYSTALS-Kyber [24] as the exclusive candidate for Key Encapsulation Mechanism (KEM) in the third round of the Post-Quantum Cryptography (PQC) standardization process. Kyber's security is grounded in the established Module Learning With Errors (MLWE) problem [25]. Kyber not only provides robust theoretical security assurances but also outshines other PQC KEMS in implementation performance [26]. It is expected that Kyber KEM will experience widespread adoption across various platforms in the years to come.

The implementation of Kyber in embedded devices has been confirmed as a concrete development. This has sparked worries about the susceptibility of Kyber to physical attacks, specifically Side-Channel Attacks (SCA). NIST emphasized the need to assess PQC schemes SCA, prompting the cryptographic community to scrutinize the implementation details of different lattice-based schemes [26]. This led to the discovery of new SCA attacks [27–29] and the development of appropriate protection strategies [30, 31]. Our objective is to scrutinize the potential security vulnerabilities arising from the use of commercial hardware in the implementation of cryptographic schemes. Specifically, our focus is on investigating side-channel attacks on PQC in hardware. In this thesis, we introduce an novel approach known as parallel Plaintext-Checking (PC) oracle attacks tailored for LWE-based KEMs, with a specific emphasis on targeting the Kyber KEM. Although we showcase all our attacks on the Kyber KEM, we are confident that our methods can be adapted to work with other LWE/LWR-based KEMs, such as Saber and FrodoKEM.

The present authentication and data transfer methods for devices rely heavily on software-based cryptographic schemes. However, these security measures demand significant system resources, which may not be practical for resource-constrained IoT devices. Moreover, their dependence renders them susceptible to persistent attackers with substantial computational capabilities, who could exploit physical

information leaks or bypass security measures. Consequently, there is a growing need for dependable hardware security solutions that can be integrated into these devices [32]. The HROT is accountable for generating and maintaining all critical cryptographic keys, and it is designed to withstand tampering attempts. Key components of RoT comprise hardware security features such as Physical Unclonable Function (PUF) and True Random Number Generator (TRNG).

Non-volatile memory (NVM) devices have attracted significant interest as a potential avenue for advancing computing architectures beyond the traditional Von Neumann model. Notably, Resistive Random Access Memory (RRAM) has emerged as a promising two-terminal device due to its distinctive metal-insulator-metal (MIM) structure. The integration of RRAM is currently under active investigation for potential applications in areas such as storage, in-memory multivalued logic processing, neuromorphic computing, and security. The variability in RRAM array structures poses a key challenge. While ongoing efforts aim to address stochastic effects, it is noteworthy that these effects have been leveraged to establish hardware security primitives[32]. Our objective is to comprehensively review the latest advancements in device technologies, particularly RRAM, with the aim of employing them to develop hardware security features like Physical Unclonable Functions (PUF) and Random Number Generators (RNG). We propose innovative designs for PUF and RNG based on memristive crossbars. In addition to enhanced performance, we will address critical issues in hardware security features, such as writing-free reconfiguration and unified design.

The use of Deep Neural Networks (DNNs) is pervasive in numerous modern applications and holds considerable commercial value. However, the sensitive nature of the training data and the substantial computational resources required for developing high-precision models make DNNs a form of proprietary intellectual property that needs protection from unauthorized access. Deploying trained models to specialized inference hardware at the network edge can be achieved through conventional key exchange and encryption protocols to mitigate potential security threats. Nevertheless, the susceptibility of extracting model parameters from the inference hardware remains a significant concern in terms of security.

Binarized Neural Networks (BNNs) have emerged as a promising category of neural networks in which both the weights and activations are restricted to values of $+1$ and -1. BNNs have gained attention for edge computing applications due to the

binary computation during inference, which reduces computational complexities and conserves energy. Integrating BNNs with emerging NVM devices, particularly RRAM, is an active area of research aimed at enhancing inference performance. Despite the challenge of non-ideal RRAM behavior hindering multi-bit computing, BNNs require only two RRAM states, making them more resistant to variability. Leveraging RRAM crossbars to implement BNNs offers the advantage of utilizing a current comparator to eliminate the resource-intensive analog-to-digital converter, thereby necessitating significantly fewer peripheral resources[33, 34]. Consequently, RRAM-based BNN accelerators may be preferred over their multi-bit DNN counterparts for edge implementations. To expedite the inference process, the deployment of BNN accelerators is anticipated primarily in edge devices. However, the physical accessibility of accelerators to potential attackers renders them susceptible to theft-related security breaches. Therefore, the use of newly developed crypto-modules is proposed to safeguard neural network model parameters when deployed in state-of-the-art RRAM in-memory computing accelerators.

1.1 Major Contributions

Our main contributions can be stated as follows:

- *First part*: We present a novel side-channel-assisted P-way parallel PC oracle attack. These attacks can retrieve any P bits of secret key information per query simultaneously, whereas current PC oracle attacks can only obtain one bit per query. We found that current methods for developing chosen-ciphertext queries to achieve the best possible key recovery in binary PC oracle attacks may not always be the most effective for the suggested PC oracle attacks in the parallel environment. As a result, we suggest improved Binary Decision Trees (BDTs) to establish minimum limits for the number of queries for the suggested attacks. We have also modified the binary side-channel classifiers that were originally designed for the binary PC oracle attack. We transformed them into multi-class classifiers that can handle any 2^P number of classes, creating a practical P-way parallel PC oracle. We conducted experiments to confirm the effectiveness of our attacks using the most efficient implementation of the Kyber KEM in the *pqm4* library [35], which

is a widely recognized framework for evaluating and testing PQC schemes on the ARM Cortex-M4 microcontroller. We verified improvements ranging from 2.89× to 7.65× compared to binary PC oracle attacks. It is worth noting that further significant enhancements are feasible, as demonstrated later in this paper. These advancements offer specific guidance to designers for establishing secure key refresh rates in situations where an ephemeral setting for key-exchange is not feasible. We have also performed an extensive assessment of our attack's capabilities in various attack scenarios, considering (1) the existence or non-existence of a clone device, and (2) the potential for partial key recovery for attackers with diverse offline computation abilities. It is evident from our observations that our attack leads to significantly substantial enhancements in trace numbers, particularly in cases where the attacker can generate a large number of templates on the clone device. We also experimentally confirmed that our attacks can be applied to implementations safeguarded with low-cost countermeasures like shuffling. Our attack produces the fewest number of traces when compared to current chosen-ciphertext attacks involving side-channel assistance, which target the shuffled implementation of Kyber KEM.

- *Second part*: We present a novel RRAM PUF design that enables one-time reconfiguration without the need for write operations, allowing for the collection of reconfigured space prior to deployment. Furthermore, we propose the incorporation of a post-processing block into our RRAM PUF to enhance its attributes and resilience against machine learning attacks. Our approach includes a method for generating a RNG from the proposed PUF, leveraging a 1T1R crossbar to harness entropy derived from D2D HRS variations for both functionalities. Additionally, we assess the quality of the random bit-stream generated from the RNG using the NIST SP 800-22 test suite.

- *Third part*: We introduce novel weight transformation methods based on PUFs to protect the model parameters of BNNs employed in memristive crossbars. Our study includes hardware realizations of each technique, accompanied by in-depth discussions on their inference performance. Additionally, we perform power analyses to gauge the resulting additional overhead.

1.2 Outline of the Thesis

In Chapter 2, an overview and motivation of the research conducted in this thesis on SCA for PQC KEMs are presented. Additionally, significant contributions are highlighted, which will be further elaborated in the subsequent chapters.

In Chapter 3, a comprehensive analysis of the current literature on attacks on PQC KEMs is undertaken. Furthermore, the fundamentals of the Kyber KEM are established as a foundational framework.

In Chapter 4, we detail the attacker model and the proposed attack methodology aimed at recovering multiple bits of information about the secret key with each query.

Chapter 5 outlines the experimental setup used to validate the proposed attack. This chapter also provides details on the obtained results for validation, offering insight into the efficacy of the attack.

In Chapter 6, a comprehensive analysis is provided concerning the performance of the proposed attack. The chapter also articulates the advantages of the attack, emphasizing the heightened vulnerability it brings into focus.

In Chapter 7, we present a comprehensive summary of our findings on SCA in LWE-based schemes.

In Chapter 8, a comprehensive analysis is presented, delving into the progress made in hardware security primitives through the utilization of NVM device technologies, particularly focusing on RRAM.

In Chapter 9, a novel PUF is presented, leveraging memristive crossbar technology. The chapter also provides a detailed analysis of the PUF's characteristics and investigates potential machine learning attacks.

In Chapter 10, we present a methodology for building a unified PUF and RNG design. We also provide a detailed analysis of its characteristics.

In Chapter 11, we delve into the significance of safeguarding BNN model parameters and examine the current research on securing these parameters in memristive

crossbars. We then establish the groundwork for BNN and its integration into memristive crossbar architecture, and scrutinize the potential attacker model. Subsequently, we present our method for securing BNN model parameters in memristive crossbars, accompanied by a thorough performance analysis and validation.

Finally, Chapter 12 serves as a conclusion to the thesis work, summarizing our significant contributions and outlining the direction for future work.

Part I

Security beyond Classical: Quantum Era

Synopsis

In this work, we propose generic and novel adaptations to the binary Plaintext-Checking (PC) oracle based side-channel attacks for Kyber KEM. These attacks operate in a chosen-ciphertext setting, and are fairly generic and easy to mount on a given target, as the attacker requires very minimal information about the target device. However, these attacks have an inherent disadvantage of requiring a few thousand traces to perform full key recovery. This is due to the fact that these attacks typically work by recovering a single bit of information about the secret key per query/trace. In this respect, we propose novel *parallel PC oracle* based side-channel attacks, which are capable of recovering a generic P number of bits of information about the secret key in a single query/trace. We propose novel techniques to build chosen-ciphertexts so as to efficiently realize a parallel PC oracle for Kyber KEM. We also build a multi-class classifier, which is capable of realizing a practical side-channel based parallel PC oracle with very high success rate. We experimentally validated the proposed attacks (upto $P = 10$) on the fastest implementation of unprotected Kyber KEM in the *pqm4* library. Our experiments yielded improvements in the range of 2.89× and 7.65× in the number of queries, compared to state-of-the-art binary PC oracle attacks, while arbitrarily higher improvements are possible for a motivated attacker, given the generic nature of the proposed attacks. We further conduct a thorough study on applicability to different scenarios, based on the presence/absence of a clone device, and also partial key recovery. Finally, we also show that the proposed attacks are able to achieve the lowest number of queries for key recovery, even for implementations protected with low-cost countermeasures such as shuffling. Our work therefore, concretely demonstrates the power of PC oracle attacks on Kyber KEM, thereby stressing the need for concrete countermeasures such as masking for Kyber and other lattice-based KEMs.

Chapter 2

Overview

NIST very recently announced results for the third round of the Post-Quantum Cryptography (PQC) standardization process [25], in which CRYSTALS-Kyber [24] was selected as the sole candidate for standardization of Key Encapsulation Mechanisms (KEMs). The security of Kyber is based on the well known Module Learning With Errors (MLWE) problem, and served as one of the most promising candidates for KEMs in the NIST PQC process, owing to the confidence in its theoretical security guarantees, while also offering one of the best implementation performance compared to other PQC based KEMs [26]. Thus, one can expect Kyber KEM to be implemented and designed on a wide-variety of computational platforms and applications, in the coming years.

In this respect, security of Kyber against physical attacks such as Side-Channel Attacks (SCA) naturally arises as an immediate concern, particularly for applications involving embedded devices. NIST had also particularly encouraged more research on analysing the security of PQC schemes against SCA [26]. The cryptographic community has shown significant interest towards research on development of new attacks on several lattice-based schemes including Kyber KEM [27–29], as well as development of efficient side-channel protection techniques [30, 31]. While there exists a wide variety of SCA particularly on Kyber KEM, we observe that they can be broadly classified into two main categories.

Chapters 2 is published as Rajendran, G., Ravi, P., D'Anvers, J. P., Bhasin, S., & Chattopadhyay, A. (2023). Pushing the limits of generic side-channel attacks on LWE-based KEMs-parallel PC oracle attacks on Kyber KEM and beyond. IACR Transactions on Cryptographic Hardware and Embedded Systems.

The first category of attacks only require a single trace or very few traces to perform key recovery or message recovery [27, 36, 37]. However, these attacks typically target very precise leakages from targeted operations within the scheme. They typically require a fairly sophisticated setup, as well as a detailed knowledge of the target device. Moreover, exploitation of such leakages on different implementations and targets, is not very straightforward, and at the very least requires significant adaptations.

The second category of attacks are more generic, and exploit inherent vulnerabilities in the algorithm for key recovery, while remaining relatively somewhat agnostic to the target/implementation. These generic attacks typically work by querying the target device with chosen-ciphertexts, and subsequently utilizing leakage from the decapsulation of chosen-ciphertexts as an oracle, to recover information about the secret key [28, 38, 39]. However, such side-channel assisted chosen-ciphertext attacks typically suffer from a disadvantage of requiring a few thousand queries for key recovery. In particular, the attacks realizing such a Plaintext-Checking (PC) oracle typically exploit 1-bit of information about the secret key (i.e.) binary PC oracle, thereby requiring a few thousand queries/traces for full key recovery. Thus, we observe a clear trade-off for both the categories of attacks, based on the ease of mounting the attack versus number of traces for key recovery.

In this work, we attempt to *bridge this gap* through our proposal of *parallel* PC oracle attacks for LWE-based KEMs, with main focus on Kyber KEM. While all our attacks are demonstrated on Kyber KEM, we believe our attack can be adapted to similar LWE/LWR-based KEMs such as Saber [40], FrodoKEM [41] etc. The main contributions of our work are as follows:

1. We propose generic and novel adaptations of side-channel assisted binary PC oracle-based attacks, referred to as P-way parallel PC oracle attacks, which have the ability to simultaneously recover an arbitrary P number of bits of information about the secret key per query, while state-of-the-art PC oracle attacks are only capable of extracting one bit per query.

2. We identify that existing approaches to construct chosen-ciphertext queries for optimal key recovery in the case of binary PC oracle attacks, are not always optimal for the proposed PC oracle attacks in the parallel setting. We

therefore propose improved constructions of Binary Decision Trees (BDTs) to identify lower bounds for the number of queries for the proposed attacks.

3. We also adapt the binary side-channel classifiers used for the binary PC oracle attack, to develop multi-class classifiers for an arbitrary 2^P number of classes to realize a practical P-way parallel PC oracle. We practically validated that our multi-class classifier is able to uniquely classify between 1024 classes (for $P = 10$) with 100% success rate. While higher values of P are possible in theory, it is hard to estimate a bound on the highest value of P that is possible to achieve in practice.

4. We experimentally validated our attacks on the fastest implementation of Kyber KEM in the *pqm4* library [35], a well known benchmarking and testing framework for PQC schemes on the ARM Cortex-M4 microcontroller. We practically validated improvements in the range of 2.89× and 7.65×, compared to state-of-the-art binary PC oracle attacks. However, we note that significant improvements are possible as shown later in the paper. Such improvements provide concrete inputs to a designer for determining safe key refresh rates when ephemeral setting for key-exchange is not possible.

5. We also conduct a comprehensive analysis of the capabilities of our attack in different attack scenarios, based on (1) the presence/absence of a clone device and (2) partial key recovery for attackers with different capabilites for offline computation. We observe that our attack brings about arbitrarily high improvements in the number of traces, especially when the attacker can construct a very high number of templates on the clone device

6. We also practically validated the applicability of our attacks to implementations protected with low-cost countermeasures such as shuffling. Our attack yields the lowest number of traces compared to existing state-of-the-art side-channel assisted chosen-ciphertext attacks targeting the shuffled implementation of Kyber KEM.

$\approx 25\times$ improvement for an arbitrarily strong attacker capable of constructing 2^{32} templates on the clone device

Chapter 3

Preliminaries

3.1 Notation

We denote the ring of integers modulo $q \in \mathbb{Z}^+$ as $\mathbb{Z}_q$. Elements in $\mathbb{Z}_q$ are denoted using lower case letters (i.e.) $a \in \mathbb{Z}_q$, and the i^{th} bit of $a \in \mathbb{Z}_q$ is denoted as a_i. We use R_q to denote the polynomial ring $\mathbb{Z}_q[x]/(x^n + 1)$ and polynomials in R_q are denoted using bold lower case letters (i.e.) $\mathbf{a} \in R_q$. The i^{th} coefficient of $\mathbf{a} \in R_q$ is denoted as $\mathbf{a}[i]$. A vector of polynomials in R_q (i.e.) R_q^k with $k \in \mathbb{Z}^+$ is denoted using bold lower case letters, while a matrix of polynomials in $R_q^{k \times \ell}$ with $(k, \ell) \in \mathbb{Z}^+$ are denoted using bold upper case letters. The i^{th} polynomial of $\mathbf{a} \in R_q^k$ is denoted as $\mathbf{a}_i$. Matrices and vectors of polynomials in R_q are together referred to as *modules*. The product of two polynomials $\mathbf{a}$ and $\mathbf{b}$ in R_q is denoted as $\mathbf{c} = \mathbf{a} \cdot \mathbf{b} \in R_q$, while pointwise multiplication using $\circ$ (i.e.) $\mathbf{c} = \mathbf{a} \circ \mathbf{b} \in R_q$. Byte arrays of length n are denoted as $\mathcal{B}^n$. The transpose of a matrix $\mathbf{A}$ is denoted as $\mathbf{A}^T$. The NTT representation of $\mathbf{a} \in R_q$ is denoted as $\hat{\mathbf{a}} \in R_q$.

Chapters 3 is published as Rajendran, G., Ravi, P., D'Anvers, J. P., Bhasin, S., Chattopadhyay, A. (2023). Pushing the limits of generic side-channel attacks on LWE-based KEMs-parallel PC oracle attacks on Kyber KEM and beyond. IACR Transactions on Cryptographic Hardware and Embedded Systems.

3.2 Kyber KEM

Kyber is a chosen-ciphertext secure KEM (IND-CCA), built upon the hardness of the Module-LWE (MLWE) problem. It offers three parameter sets, hereby listed in increasing levels of security - (1) `Kyber-512` (NIST Security Level 1), (2) `Kyber-768` (Level 3) and (3) `Kyber-1024` (Level 5) with $k = 2, 3$ and 4 respectively. The CCA secure Kyber KEM is built upon a simpler chosen-plaintext secure PKE (IND-CPA), denoted as CPA.Kyber PKE. Refer to Algorithm 1 for a simplified description of the key-generation (CPA.KeyGen), encryption (CPA.Encrypt) and decryption procedures (CPA.Decrypt) of CPA.Kyber PKE. Sample_U is used to denote the operation sampling coefficients from a uniform distribution, Sample_B to denote sampling from a centered binomial distribution (CBD) and the function Expand, to denote expanding a small seed into a uniformly random matrix in $R_q^{k\times k}$. The function $\mathsf{Compress}(\mathbf{u}, d)$ lossily compresses $\mathbf{u} \in \mathbb{Z}_q$ into $v \in \mathbb{Z}_{2^d}$ with $q > 2^d$, while $\mathsf{Decompress}(\mathbf{v}, d)$ extrapolates $\mathbf{v} \in \mathbb{Z}_{2^d}$ into $u' \in \mathbb{Z}_q$.

3.2.1 IND-CCA Security

The IND-CPA secure PKE is transformed into an IND-CCA secure KEM using a post-quantum variant of well-known Fujisaki-Okamoto transformation [42]. It involves the use of two hash functions $(\mathcal{H}, \mathcal{G})$ and a key-derivation function KDF, forming a wrapper denoted as encapsulation (CCA.Encaps) and decapsulation (CCA.Decaps) procedures of CCA.Kyber KEM (Refer Alg.2).

Within this framework, the encryption procedure is deterministic and depends solely on the message m for a given public key pk. This is done by ensuring that the seed input r' to the encryption procedure is derived by hashing m with pk (Line 4-5 in CCA.Encaps). In the decapsulation procedure (CCA.Decaps), the decrypted message m is *re-encrypted* (Line 12-13) to generate the ciphertext ct'. The received ciphertext ct is compared with ct', and the valid shared key is generated (Line 17) only if $ct = ct'$, denoting a valid ciphertext, else a pseudo-random key is generated (invalid ciphertext). This enables to detect invalid/malicious ciphertexts, thereby offering concrete protection against chosen-ciphertext attacks. We refer the reader to [24] for more details on CCA secure Kyber KEM.

Algorithm 1 CPA Secure Kyber PKE (Simplified)

procedure CPA.KeyGen
  $(seed_A, seed_B) \in \mathcal{B}^* \leftarrow \mathsf{Sample}_U()$ ▷ Generate uniform seeds $seed_A, seed_B$
  $\hat{\mathbf{A}} \in R_q^{k \times k} \leftarrow \mathsf{Expand}(seed_A)$ ▷ Expand $seed_A$ into $\hat{\mathbf{A}}$
  $\mathbf{s}, \mathbf{e} \in (R_q^k \times R_q) \leftarrow \mathsf{Sample}_B(seed_B, coins)$ ▷ Sample $\mathbf{s}, \mathbf{e}$
  $\hat{\mathbf{s}} \in R_q^k \leftarrow \mathsf{NTT}(\mathbf{s})$; $\hat{\mathbf{e}} \in R_q^k \leftarrow \mathsf{NTT}(\mathbf{e})$ ▷ $\mathsf{NTT}(\mathbf{s})$, $\mathsf{NTT}(\mathbf{e})$
  $\hat{\mathbf{t}} = \hat{\mathbf{A}} \circ \hat{\mathbf{s}} + \hat{\mathbf{e}}$ ▷ $\mathbf{t} = \mathbf{A} \cdot \mathbf{s} + \mathbf{e}$ in NTT domain
  Return $(pk = (seed_A, \hat{\mathbf{t}}), sk = (\hat{\mathbf{s}}))$
end procedure

---

procedure CPA.Encrypt$(pk, m \in \{0,1\}^{256}, seed_R \in \{0,1\}^{256})$
  $\hat{\mathbf{A}} \in R_q^{k \times k} \leftarrow \mathsf{Expand}(seed_A)$
  $\mathbf{r}, \mathbf{e_1}, \mathbf{e_2} \in (R_q^k \times R_q^k \times R_q) \leftarrow \mathsf{Sample}_B(seed_R)$ ▷ Sample $\mathbf{r}, \mathbf{e_1}, \mathbf{e_2}$
  $\hat{\mathbf{r}} \in R_q^k \leftarrow \mathsf{NTT}(\mathbf{r})$ ▷ $\mathsf{NTT}(\mathbf{r})$
  $\mathbf{u} \in R_q^k \leftarrow \mathsf{INTT}(\mathbf{A}^T \circ \hat{\mathbf{r}}) + \mathbf{e_1}$ ▷ $\mathbf{u} = \mathbf{A}^T \cdot \mathbf{r} + \mathbf{e_1}$
  $\mathbf{v}' \in R_q \leftarrow \mathsf{INTT}(\hat{t}^T \circ \hat{\mathbf{r}}) + \mathbf{e_2}$ ▷ $\mathbf{v}' = \mathbf{t}^T \cdot \mathbf{r} + \mathbf{e_2}$
  $\mathbf{v} \in R_q \leftarrow \mathbf{v} + \mathsf{Decompress}(m, 1)$ ▷ $\mathbf{v} = \mathbf{v} + \mathsf{Encode}(m)$
  Return $ct = \mathsf{Compress}(\mathbf{u}, d_1), \mathsf{Compress}(v, d_2)$
end procedure

---

procedure CPA.Decrypt(sk, ct)
  $(\mathbf{u}, \mathbf{v}) \leftarrow \mathsf{Decompress}(ct, d_1, d_2)$
  $\hat{\mathbf{u}} = \mathsf{NTT}(\mathbf{u})$
  $\mathbf{m} = \mathbf{v} - \mathsf{INTT}(\hat{\mathbf{u}} \circ \mathbf{s})$ ▷ $\mathbf{m} = \mathbf{v} - \mathbf{u} \cdot \mathbf{s}$
  $m \in R_q \leftarrow \mathsf{Compress}(\mathbf{m}, 1)$ ▷ Decoding $\mathbf{m} \in R_q$ into $m \in \mathcal{B}^{32}$
  Return m
end procedure

3.3 Prior Side-Channel Attacks and Motivation

LWE/LWR-based KEMs including Kyber KEM have been subjected to a wide variety of side-channel attacks whose primary target is the long-term secret key $\mathbf{s}$ used in the decapulation procedure. Recovery of a single secret key $\mathbf{s}$ leads to compromise of all the corresponding session keys K, that were derived using $\mathbf{s}$.

In this respect, we can broadly classify existing attacks on Kyber KEM into two categories: (1) Target Operation Independent attacks (TO_Indep) and (2) Target

Algorithm 2 CCA secure Kyber KEM

procedure CCA.ENCAPS(pk)
  $m \leftarrow \{0,1\}^{256}$
  $m' = \mathcal{H}(m)$
  $(\bar{K}', r') = \mathcal{G}(m' \| \mathcal{H}(pk))$ ▷ Generate $\bar{K}'$ and r' using m and pk
  $ct = \mathsf{CPA.Encrypt}(pk, m', r')$
  $K = \mathsf{KDF}(\bar{K}' \| \mathcal{H}(ct))$
  Return (ct, K)
end procedure

---

procedure CCA.DECAPS(sk, ct)
  $(pk, \mathcal{H}(pk), z) \leftarrow \mathsf{UnpackSK}(sk)$
  $m = \mathsf{CPA.Decrypt}(sk, ct)$
  $(\bar{K}, r) = \mathcal{G}(m, \mathcal{H}(pk))$
  $ct' = \mathsf{CPA.Encrypt}(pk, m, r)$ ▷ $\mathsf{Re\_Encrypt}(m, pk)$ **if** $(ct' == ct)$ **then**
    Return $K = \mathsf{KDF}(\bar{K} \| \mathcal{H}(ct'))$ ▷ Ciphertext Comparison Success **else**
    Return $K = \mathsf{KDF}(z \| \mathcal{H}(ct'))$ ▷ Ciphertext Comparison Failure
end procedure

Operation Dependent (TO_Dep) attacks. TO_Dep attacks are those that exploit side-channel leakage from a specific targeted operation within the decapsulation procedure [37, 43, 44]. On the other hand, TO_Indep attacks are not limited to any single operation, but can collectively exploit leakage from several operations within the decapsulation procedure [28, 45, 46]. Moreover, TO_Indep attacks are generic and to a certain degree, agnostic to the target implementation, while TO_Dep attacks are more specific to the target device/implementation.

3.3.1 Target Operation Independent Attacks (TO_Indep)

Several works have shown that an adversary with the ability to query the decapsulation device with chosen-ciphertexts can amplify side-channel leakage related to the secret key[28, 45, 46]. The modus operandi of such attacks is as follows: The attacker submits malformed ciphertexts ct to the decapsulation device, whose corresponding decrypted message m has a close relation to the secet key $\mathbf{s}$. An attacker who can exploit side-channel leakage to recover information about m, in effect instantiates an *oracle* which leads to recovery of $\mathbf{s}$. These attacks can be further classified into two categories:

Plaintext-Checking (PC) Oracle-based SCA: D'Anvers *et al.* [45] demonstrated that chosen-ciphertexts can be constructed to restrict the decrypted message to a only two possible values (i.e.) $m = 0$ or $m = 1$, with the value of m depending on targeted single coefficients of $\mathbf{s}$. They utilized the timing side-channel from variable time error correcting codes, to recover m (i.e.) $\mathsf{Time}(\mathsf{ECC\_Decode}(0))! = \mathsf{Time}(\mathsf{ECC\_Decode}(1))$, thereby instantiating a Plaintext-Checking (PC) oracle. Ravi *et al.* [28] subsequently generalized the attack to multiple LWE/LWR-based KEMs including Kyber KEM through the EM side-channel. They showed that a single bit change between $m = 0$ and $m = 1$ results in vastly different computations in the re-encryption procedure, which can be easily distinguished in a single trace. Every chosen-ciphertext query only enables recovery of a single bit information about the secret $\mathbf{s}$, thus full key recovery is only possible in a few thousand queries ($\approx 2k - 3k$).

Decryption-Failure (DF) Oracle-based SCA: These attacks work by submitting perturbed ciphertexts $ct' = ct_{\mathsf{valid}} + \epsilon$ to the target, which induce a *decryption failure* depending on the value of the secret key $\mathbf{s}$. An attacker who can detect a decryption failure (i.e.) $m = m_{\mathsf{valid}}$ or $m = m_{\mathsf{invalid}}$ can construct *linear hints* about the secret key, enabling full key recovery in a few thousand queries ($5k - 6k$).

Guo *et al.* [46] exploited variable time implementations of the ciphertext comparison operation in the decapsulation procedure (Line 14 in $\mathsf{CCA.Decaps}$ of Alg.2) to instantiate a DF oracle, while subsequent works [29, 47] demonstrated exploitation of the EM side-channel for key recovery. While these attacks specifically targeted the ciphertext comparison operation for leakage, these attack can also exploit leakage from the entire re-encryption procedure of an unprotected implementation, to easily distinguish between $\mathsf{Re-Encrypt}(m_{\mathsf{valid}}, pk)$ and $\mathsf{Re-Encrypt}(m_{\mathsf{invalid}}, pk)$.

In essence, all the aforementioned attacks recover upto a single bit information about the secret key sk, thereby instantiating a *binary* oracle, which requires a few thousand queries for key recovery. These attacks are particularly attractive for their inherent *simplicity* in performing key recovery, where the attacker requires very minimal information about the underlying target.

3.3.2 Target Operation Dependent Attacks (TO_Dep)

These attacks target leakage from specific leaky operations in the decapsulation procedure for key recovery. We discuss two major types of attacks.

Targeting Message Encoding/Decoding Operation: Several attacks have targeted the message encoding (Line 15 of CPA.Encrypt in Alg.1) and message decoding (Line 22 of CPA.Decrypt) operations, enabling recovery of the entire message m in a single trace [27, 37, 48, 49]. These attacks mainly exploit very fine leakages from manipulations of single bits of the sensitive message m, enabling full message recovery. Xu *et al.* [43] showed that the aforementioned leakage can be exploited to instantiate a *Full Decryption* (FD) oracle in a chosen-ciphertext setting. This enabled full key recovery in only 6 queries for Kyber512. Adaptations of the same attack has also been demonstrated on masked and shuffled implementations of Kyber KEM [36, 49].

While these attacks consume very few queries for full key recovery, they suffer from inherent disadvantages. Firstly, they exploit very fine leakages from single bit manipulations of the message. Thus, leakage is limited to single clock cycles or very few samples for each message bit, thereby naturally being sensitive to acquisition noise (SNR), as shown in [27]. Moreover, it is not clear if similar leakages can be exploited on complex devices with features such as heavy parallelism, deep pipelining and inherent jitter, especially given the sensitivity of these attacks to noise.

Targeting NTT Operation: Several attacks have targeted the Number Theoretic Transform (NTT), used for polynomial multiplication in Kyber KEM [44, 50]. These attacks enable full key recovery in a single trace or very few traces, exploiting advanced algebraic side-channel techniques. However, they also suffer from the same disadvantages of exploiting fine leakages from multiple targeted instructions, while also requiring a sophisticated setup or a powerful side-channel adversary for key recovery. While improved attacks to counter noise have been demonstrated [51], the attacks still are relatively hard to implement. Moreover, the threat of the same attack on more sophisticated platforms is not clear.

3.3.3 Trade-off In TO_Indep and TO_Dep Attacks

We observe a very clear trade-off between the ease of attack and the number of traces/queries to perform full key recovery. While TO_Dep attacks only require a handful of traces for key recovery, these attacks rely on very delicate leakages from targeted operations/instructions for key recovery. However, TO_Indep attacks fall on the other side of the spectrum, with respect to ease of key recovery. The attacks can exploit leakage from practically the entire re-encryption procedure, but require traces/queries ranging in the few thousands for key recovery. The number of queries is particularly important since refreshing the key pair is a common strategy to reduce exposure of the key to possible classical/side-channel attacks [27]. Thus, a natural question that arises is "*whether it is possible to construct attacks that can obtain the best of both worlds?*" (i.e.) attacks that can exploit leakage independent of the target operation, as well as enable efficient key recovery in very few queries.

In this work, we answer this question positively by proposing generic and novel *parallel* Plaintext-Checking (PC) oracle based attacks, bringing about significant improvement in number of queries in key recovery, compared to the binary PC oracle attacks [28]. We lay main focus on improving the efficiency of simple and generic side-channel attacks on unprotected implementations. Thus, masking is naturally out of scope of this paper. Our work is motivated from the view point of a designer, contemplating the decision to use heavy countermeasures such as masking for SCA protection of Kyber [30, 31]. In this respect, our work attempts to answer the question: "*what is the simplest and most efficient attack on an unprotected implementation, with a basic SCA setup and very limited knowledge about the target?*.

The ephemeral setting for KEMs used for key-exchange is often recommended, where the public-private key pair (pk, sk) is refreshed for every new key-exchange. However, it is not always practical due to huge performance overhead of frequent key generation. Thus, the static key setting with regular refreshment of the key pair is a more preferable setting, where the public-private key pair (pk, sk) is refreshed once every X number of key-exchanges, where X is chosen by the designer. Here, the static secret key sk is more exposed to the attacker due to its use in multiple key-exchanges, compared to the ephemeral setting. If an attacker is able to recover the secret key in Y number of key-exchanges where $Y < X$, then the remaining $Z = X - Y$ number of key-exchanges using the same secret key sk with other

legitimate devices are compromised. This is because all corresponding session keys can be recovered with the knowledge of sk. Lower the value of Y, higher is the number of session keys that can be compromised by the attacker for a given secret key sk. In this context, our proposed attacks which improve upon the number of queries for key recovery provide concrete inputs to a designer for choosing an appropriate key refresh rate. As we show later in Section 6, the impact of our attacks depend upon different scenarios such as availability of clone device, ability to perform offline computations after partial key-recovery etc.

In this work, we primarily focus on unprotected implementations, but our proposed attacks also perform better than existing attacks on *lightly protected* implementations using countermeasures such as shuffling [27]. Refer to Fig.3.1 for an illustration of a qualitative comparison of reported SCA applicable to Kyber, with respect to target dependency and number of traces.

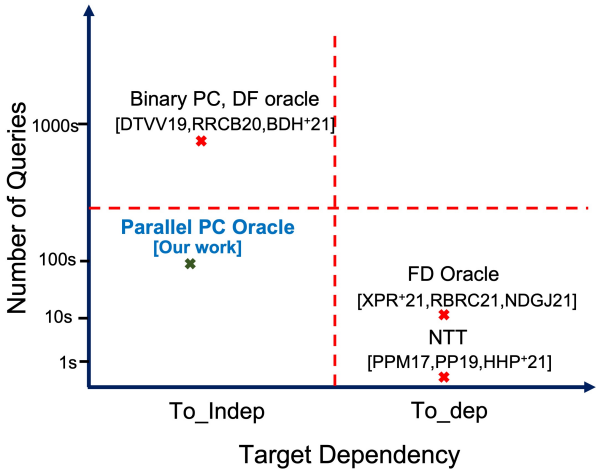


FIGURE 3.1: Qualitative comparison of reported SCA applicable to Kyber, with respect to target dependency and number of traces. Due to lack of space, we do not list all the attacks in the different categories

Chapter 4

Improved PC Oracle-based CCA

4.1 Attacker Model

We assume that the attacker has physical access to the target device implementing the decapsulation procedure of Kyber KEM. The attacker has the ability to query the target device with chosen-ciphertexts ct of his/her choice. Moreover, prior knowledge of the secret key of the DUT or detailed knowledge about the underlying implementation such as the source code or compiled executable is not required. The attacker also does not require the ability to profile the side-channel leakage of the Device Under Test (DUT) with known keys. In the following, we explain the binary PC oracle attack of Ravi *et al.* [28] on Kyber KEM, which serve as the basis of our improved attacks.

4.2 Binary PC Oracle-based CCA

The attack works by recovering the secret key one coefficient at a time. We therefore demonstrate recovery of a single coefficient $\mathbf{s}[0]$, while other coefficients can be

Chapters 4 is published as Rajendran, G., Ravi, P., D'Anvers, J. P., Bhasin, S., Chattopadhyay, A. (2023). Pushing the limits of generic side-channel attacks on LWE-based KEMs-parallel PC oracle attacks on Kyber KEM and beyond. IACR Transactions on Cryptographic Hardware and Embedded Systems.

recovered in a similar manner. For simplicity, we also assume that all the components are only polynomials in R_q, however the same technique can be extended to higher dimensions (i.e.) R_q^k.

4.2.1 Construction of Chosen-Ciphertexts

The attacker constructs chosen-ciphertexts $ct = (\mathbf{u}, \mathbf{v}) \in (R_q \times R_q)$ as $\mathbf{u} = k_u \cdot x^0$ and $\mathbf{v} = k_v \cdot x^0$ where $(k_u, k_v) \in \mathbb{Z}_q$. The corresponding decrypted message m is given as:

$$m_i = \begin{cases} \mathsf{Decode}(k_v - k_u \cdot \mathbf{s}[0]) & \text{for} \quad i = 0 \\ \mathsf{Decode}(-k_u \cdot \mathbf{s}[i]) & \text{for} \quad i \in \{1, n-1\} \end{cases} \tag{4.1}$$

for $i \in [0, n-1]$. Thus, every bit m_i of the decrypted message is only dependent on the corresponding secret coefficient $\mathbf{s}[i]$. Now, the attacker can choose values for (k_u, k_v) such that:

$$m_i = \begin{cases} \mathcal{F}(\mathbf{s}[0]), & \text{if } i = 0 \\ 0, & \text{for } 1 \leq i \leq n-1 \end{cases} \tag{4.2}$$

where m can only take two possible values (i.e.) $m = 0$ and $m = 1$ (all bits except LSB have a value of 0). Moreover, $m = 0/1$ for a given ct, solely depends upon the value of $\mathbf{s}[0]$. Here, $\mathcal{F}$ represents the relation between the secret coefficient $\mathbf{s}[0]$ and m_i such that multiple values of the tuple (k_u, k_v) can uniquely identify the value of $\mathbf{s}[0]$ based on the corresponding value of m_i. In other words, the attacker needs to identify the appropriate values for the tuple (k_u, k_v) such that the corresponding values for the message bit $m_i = 0/1$ serves as a binary distinguisher for $\mathbf{s}[0]$ based on $m = 0/1$. Thus, the value of $m_i = 0/1$ serves as a *binary plaintext checking (PC) oracle* for the attacker to obtain information about the secret coefficient $\mathbf{s}[0]$.

An attacker who can instantiate such a binary PC oracle through *side-channels* can uniquely recover the secret coefficient $\mathbf{s}[0]$. Similarly, other coefficients can be recovered by exploiting the rotational property of polynomial multiplication in the ring R_q. Multiplying $\mathbf{r} \in R_q$ with x^p rotates $\mathbf{r}$ by p positions in an anti-cyclic fashion [28]. Thus, for the chosen-ciphertext $\mathbf{u} = k_u \cdot x^p$ and $\mathbf{v} = k_v \cdot x^0$ with

$p \in \mathbb{Z}^+$, the first message bit m_0 is given as:

$$m_0 = \begin{cases} k_v - k_u \cdot \mathbf{s}[0], & \text{if } p = 0 \\ k_v - k_u \cdot (-\mathbf{s}[n-p]), & \text{for } 1 \leq p \leq n-1 \end{cases} \tag{4.3}$$

while the other bits are fixed to a value of 0. Thus, changing the parameter p ensures that m_0 depends upon different secret coefficients of $\mathbf{s}$, which can also be recovered in the same manner as $\mathbf{s}[0]$.

4.2.2 Instantiating Binary PC Oracle through SCA

A close observation of the decapsulation procedure (CCA.Decaps in Alg.2) reveals that the decrypted message m is hashed with the public-key ($\mathcal{G}$ in Line 12), and its result r along with the message m is fed into the re-encryption procedure (Line 13). For brevity, we denote these operations together as Re_Encrypt(m, pk). This re-encryption procedure is deterministic and solely depends upon m for a given public key pk. Thus, a single bit difference between $m = 0$ and $m = 1$ induces very different computations throughout the re-encryption procedure. This amounts to a few thousand leaky Points of Interest (PoI) which can be used to easily distinguish between $m = 0$ and $m = 1$, thereby instantiating a binary PC oracle.

4.3 Optimizing the Number of Queries for Key-Recovery

Ravi *et al.* [28] targeted the Round 2 specification of Kyber512 with secret coefficients in $[-2, 2]$. They utilized a *non-adaptive* approach to query the decapsulation device with chosen-ciphertexts, thereby using 5 ciphertexts to recover a single coefficient in $[-2, 2]$, which is clearly suboptimal. Thus, subsequent works [52, 53] proposed improved adaptive attacks to reduce the number of queries for key recovery. The most recent work of Qin *et al.* [54] proposed a systematic approach to find the lower bounds for the number of queries. They propose to build a Binary Decision Tree (BDT), which can be traversed based on the oracle's responses $m = 0/1$ to efficiently recover the correct coefficient.

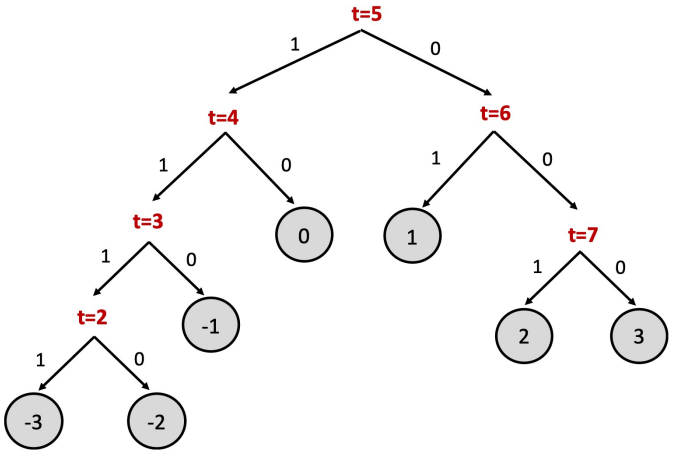


FIGURE 4.1: Optimal BDT for Kyber512 to minimise $\mathcal{Q}_{bin}$ for binary PC oracle-based CCA

4.3.1 Construction of an Optimal BDT

The core idea to construct an optimal BDT is based on the observation that the secret coefficients of Kyber follow a *non-uniform* CBD distribution. Thus, the optimal minimum for queries can be attained by constructing a non-uniform distinguisher with the following strategy: *higher the frequency of a secret coefficient candidate, lower should be the number of queries for unique distinguishability.* Thus, the number of queries to uniquely recover a candidate is inversely proportional to the probability of its occurrence.

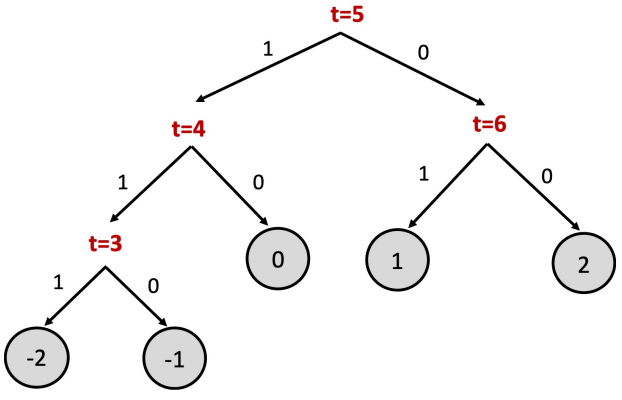


FIGURE 4.2: Optimal BDT for Kyber768, Kyber1024 to minimise $\mathcal{Q}_{bin}$ for binary PC oracle-based CCA

Let q_x denote the number of queries to uniquely distinguish $\mathbf{s}[i] = x$, and $\mathsf{Pr}(x)$ denote the probability that a secret coefficient $\mathbf{s}[i] = x$. Then, the objective is to

build a BDT that yields a minimum for $\mathcal{Q}_{bin}$ which is given as:

$$\mathcal{Q}_{bin} = \sum_{i=-\eta}^{i=\eta} q_x \cdot \mathsf{Pr}(x) \tag{4.4}$$

We adopted the technique of Qin *et al.* [54] to construct BDTs for all parameter sets of Kyber. For our chosen-ciphertexts, we choose $(k_u, k_v) = (208, 208 \cdot t)$ where $t \in \mathbb{Z}^+$. Refer to Figure.4.1 for the corresponding optimal BDT for Kyber512 with $\mathcal{Q}_{bin} = 2.5625$, distinguishing every candidate in $[-3, 3]$ in not more than 4 queries. A node, edge and leaf of the BDT denotes a chosen-ciphertext query, oracle response and a recovered secret coefficient respectively. The optimal BDT for Kyber768 and Kyber1024 (with coefficients in $[-2, 2]$) is shown in Figure.4.2, with $\mathcal{Q}_{bin} = 2.3125$. Thus, the average number of queries for full key recovery is given as:

$$\mathcal{Q}_{attack} = 2^8 \cdot k \cdot \mathcal{Q}_{bin} \tag{4.5}$$

Thus, $\mathcal{Q}_{attack}$ amounts to 1312, 1776 and 2368 for Kyber512, Kyber768 and Kyber1024 respectively. We refer to the constructed trees as $\mathsf{BDT}_{\mathsf{min\_ent}}$, since the minimum average number of queries is very similar to computation of a certain Shannon entropy.

4.3.2 Critical Observations on the Binary PC Oracle-based CCA

We make two critical observations on the binary PC oracle CCA.

1. Observation-1: The decrypted message m for the chosen-ciphertexts only contains a single secret dependent message bit (i.e.) $m_0 = 0/1$, while all other bits $m_i = 0 \; \forall \, i = \{1, n-1\}$.
2. Observation-2: Leakage from the entire re-encryption procedure has only been exploited to recover a single bit of m, and therefore the secret key $\mathbf{s}$, especially when there are a few thousand leakage points [28].

This motivates us to investigate if it is possible to recover multiple bits of information about the secret key, exploiting leakage from the re-encryption procedure

in a generic manner. In the following, we propose novel extensions of the binary PC oracle attack, which parallelize secret key recovery in a generic and configurable manner. This yields significant improvements in the possible lower bounds achievable for key recovery.

4.4 Parallel PC Oracle-based CCA

The core idea of our attack lies in constructing ciphertexts, such that multiple targeted bits of the message m (i.e.) m_i for $i \in \{0, P-1\}$ ($P \in \mathbb{Z}^+$) depend upon the P corresponding coefficients of the secret key. To achieve the same, we choose $\mathbf{u} = 208 \cdot x^0$ (i.e.) $k_u = 208$ and $\mathbf{v} = 208 \cdot t \cdot (\sum_{i=0}^{i=(P-1)} x^i)$ where $t \in \mathbb{Z}^+$ (i.e.) $k_v = 208 \cdot t$. We explain our choice for the exact value for $(\mathbf{u}, \mathbf{v})$ later in this section. Thus, the decrypted message m is given as:

$$m_i = \begin{cases} \mathsf{Decode}(208 \cdot t - 208 \cdot \mathbf{s}[i]), & \text{if } i \in [0, P-1] \\ \mathsf{Decode}(-208 \cdot \mathbf{s}[i]), & \text{for } i \in [P, n-1] \end{cases} \tag{4.6}$$

For the same values of t used to build the optimal BDT (Fig.4.1-4.2), the decrypted message m is given as:

$$m_i = \begin{cases} \mathcal{F}(\mathbf{s}[i]), & \text{if } i \in [0, P-1] \\ 0, & \text{for } i \in [P, n-1] \end{cases} \tag{4.7}$$

Thus, the first P bits of m are now dependent on the corresponding coefficients of $\mathbf{s}$, while all the other bits are fixed to 0. This technique is subtly different from attacks exploiting the *Full-Decryption* (FD) oracle [27, 43], where all the bits of m are dependent on the corresponding coefficients of $\mathbf{s}$. However, our technique allows us to control the number and position of the secret dependent message bits. As seen later in Sec.5, this nuanced difference in approach allows to exploit leakage from the re-encryption procedure, in a parallel as well as generic manner.

An adversary able to recover the correct value of the message (i.e.) $m \in [0, 2^P - 1]$ can simultaneously recover a configurable P bits of information about $\mathbf{s}$. While the binary PC oracle attack traverses one BDT in a single query (Fig.4.1-4.2), our attack allows us to simultaneously traverse P distinct BDTs (BDT_i for $i \in [0, P-1]$)

in a single query, where each BDT_i is traversed using the corresponding message bit m_i. This simultaneous traversal of P BDTs for P message bits m_i for $i \in [0, P-1]$ is made possible due to the choice of our chosen-ciphertexts. In particular, the value of the ciphertext component $\mathbf{u}$ is fixed to $\mathbf{u} = 208 \cdot x^0$, while the coefficients of the ciphertext component $\mathbf{v}$ (i.e.) the value of t for $\mathbf{v}[i] = 208 \cdot t$ with $i \in [0, P-1]$, can be decided based on the traversed node of the corresponding BDT (i.e.) BDT_i and the corresponding message bit m_i. We exhaustively searched for a single value for the non-zero coefficient of $\mathbf{u}$ (i.e.) k_u such that, simply changing $\mathbf{v}$ can yield different oracle responses to uniquely identify all possible values for the secret coefficients. In this manner, the traversal of all the P BDTs for the message bits m_i for $i \in [0, P-1]$ can be made completely independent of one-another.

Thus, a generic number of P secret coefficients can be simultaneously recovered in not more than 4 queries for Kyber512 using $\mathsf{BDT}_{\mathsf{min\_ent}}$ (Figure.4.1). Similarly, P secret coefficients can be simultaneously recovered in not more than 3 queries for Kyber768, Kyber1024 using $\mathsf{BDT}_{\mathsf{min\_ent}}$ (Figure.4.2). Similar to the binary attack, the rotational property of polynomial multiplication can be used to recover different P coefficients at a time, thereby recovering the full key. We refer to P as the parallelization factor and our attack as the P-way parallel PC oracle attack.

4.5 Optimal Key Recovery for P-way Parallel Attack

While the approach of Qin *et al.* [54] yields the lower bound for queries for the binary attack, it is not clear if it is optimal in the P-way parallel attack. Our intuition is based on the following observation. If we utilize $\mathsf{BDT}_{\mathsf{min\_ent}}$ (Figure.4.1) for a 2-way parallel attack on Kyber512, then recovering two coefficients with a value of $(0, 0)$ simultaneously, only requires two queries. However, pairs with values of $(-2, 0)$ or $(-3, 0)$, can only be recovered in 4 queries, since -2 and -3 occur at a higher depth in the BDT. Thus, the number of queries to recover a given set of P random coefficients depends upon that coefficient in the set, with the maximum depth in the BDT. Moreover, increasing the parallelization factor P, only increases the probability of observing coefficients with a higher depth (e.g.) $-2, -3$, leading to increase in average number of queries to recover a given set of P coefficients.

4.5.1 More efficient BDTs for the P-way Parallel Attack

This leads us to hypothesize if BDTs with a lower depth yield lower average number of queries for the P-way parallel attack, in particular for higher values of P. We refer to Eqn.4.6 to construct our chosen-ciphertexts. After careful consideration of all possible values of t and corresponding m_i, we construct a BDT with a depth of 3 (Figure.4.3), which is the lowest achievable depth for unique distinguishability of coefficients in $[-3, 3]$. We also verified that it is not possible to build a BDT, with lower entropy and a depth of 3. We refer to this alternate tree as $\mathsf{BDT}_{\mathsf{min\_depth}}$.

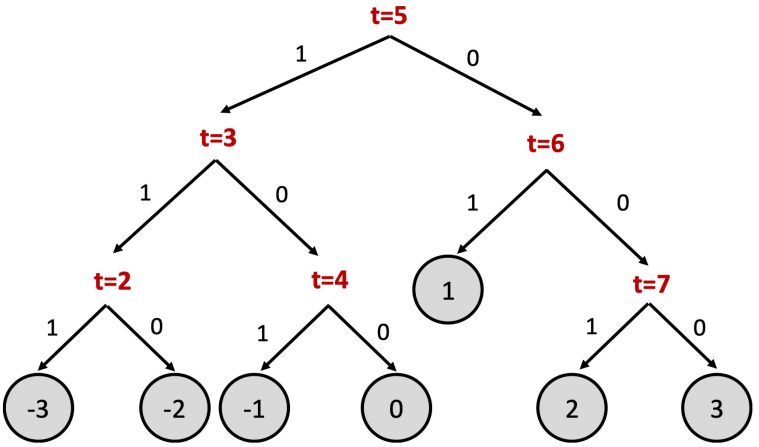


FIGURE 4.3: Optimal BDT ($\mathsf{BDT}_{\mathsf{min\_depth}}$) for P-way parallel oracle attack on Kyber512

We now derive an expression to compute the average number of queries to recover a set of P random coefficients for a generic BDT. We introduce some notation to explain our analysis. Refer to Figure.4.4 for the corresponding illustration of the same. We use *subtree* to denote a tree with smaller depth starting from the root. We use the notation st_d to denote a subtree with depth d where the depth of the root node is 0. So, for any BDT with maximum depth d, there are $d+1$ such subtrees (i.e.) $\{st_0, st_1, \ldots, st_d\}$. We denote the set of subtrees that contain at least a single leaf (recovered coefficient) as $\mathcal{V}$. The set of leaves in a given sub-tree st_i is denoted as $\mathcal{L}_{st_i}$ and specifically the leaves in the last layer of the subtree are denoted as $\mathcal{M}_{st_i}$. The average number of queries required to recover all P coefficients using the tree BDT denoted as $\mathcal{Q}_{set}$ is therefore given as:

$$\mathcal{Q}_{set} = \sum_{\forall st_i \in \mathcal{V}} i \cdot \mathsf{R}_i \tag{4.8}$$

where R_i denotes the probability that a set of random P coefficients belong to $\mathcal{L}_{st_i}$, with at least one coefficient in the set $\mathcal{M}_{st_i}$. With knowledge about the apriori distribution of the secret coefficients, it is possible to compute $\mathcal{Q}_{set}$ for any given BDT.

Note that it might be possible to obtain a more efficient tree using all possible P-tuples and not splitting into 2 branches after each node, but instead splitting into 2^P branches. While this approach might slightly further reduce the number of queries needed during the attack, it requires a much higher number of templates (one for each node). As such, trying to reach the optimal BDT-mindepth in the $P > 1$ scenario is not always desirable.

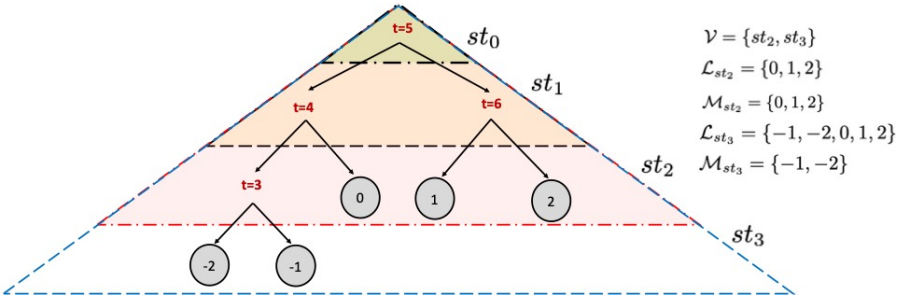


FIGURE 4.4: Illustration for calculation of $\mathcal{Q}_{set}$ (i.e.) average number of queries to recover P coefficients for a given BDT

Refer to Figure.4.5 for the plot of $\mathcal{Q}_{set}$ for different values of P for Kyber512, for both $\mathsf{BDT}_{\mathsf{min\_ent}}$ (Figure.4.1) and $\mathsf{BDT}_{\mathsf{min\_depth}}$ (Figure.4.3). We can clearly see that $\mathcal{Q}_{set}$ for our proposed $\mathsf{BDT}_{\mathsf{min\_depth}}$ graph is lower than that of $\mathsf{BDT}_{\mathsf{min\_ent}}$ for $P \geq 3$,

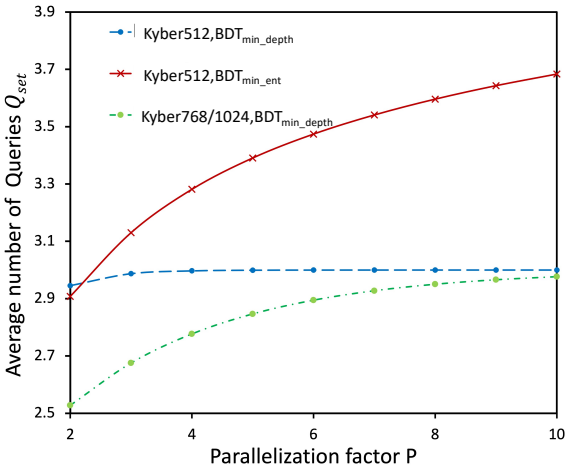


FIGURE 4.5: Average number of queries to recover P coefficients (i.e.) $\mathcal{Q}_{set}$ versus the parallelization factor P, for all parameter sets of Kyber

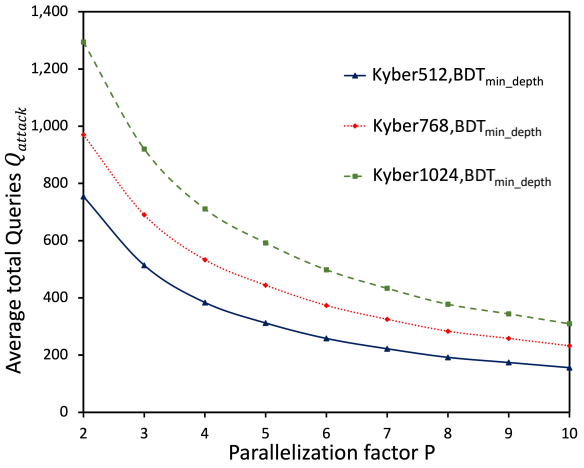


FIGURE 4.6: Average number of queries $\mathcal{Q}_{attack}$ for full key recovery, versus parallelization factor P for all parameter sets of Kyber

thereby confirming our hypothesis. Thus, our proposed BDT (i.e.) $\mathsf{BDT}_{\mathsf{min\_depth}}$ clearly yields fewer number of queries for Kyber512, compared to $\mathsf{BDT}_{\mathsf{min\_ent}}$ for $P \geq 3$. For the case of Kyber768 and Kyber1024, the secret coefficients lie in the span of $[-2, 2]$. Moroever, the same BDT used for the binary PC oracle attack (i.e.) $\mathsf{BDT}_{\mathsf{min\_ent}}$ in Fig.4.2 can be used for both Kyber768 and Kyber1024. The BDT has the minimum possible achievable depth of 3. Since the BDT already has the lowest entropy as well as minimum depth, $\mathsf{BDT}_{\mathsf{min\_ent}}$ yields the lowest number of queries for the P-way parallel attack, for both Kyber768 and Kyber1024 (i.e.) $\mathsf{BDT}_{\mathsf{min\_ent}} = \mathsf{BDT}_{\mathsf{min\_depth}}$. This therefore yields the same value for $\mathcal{Q}_{set}$, for both Kyber768 and Kyber1024 as shown in Fig.4.5.

Putting it all together, if an adversary has access to a perfect P-way PC oracle, then the average number of queries for full key recovery, denoted as $\mathcal{Q}_{attack}$ is given as:

$$\mathcal{Q}_{attack} = \lceil \frac{2^8}{P} \rceil \cdot k \cdot \mathcal{Q}_{set} \tag{4.9}$$

Refer to Figure.4.6 for $\mathcal{Q}_{attack}$ versus the parallelization factor P, for all parameter sets of Kyber. Our experimental simulations assuming a perfect P-way parallel PC oracle yielded a 100% success rate in recovering the secret key for a generic value of P. In the following, we demonstrate that an attacker can realize a very efficient and practical P-way parallel PC oracle through exploitation of side-channel leakage from the re-encryption procedure.

Chapter 5

Realizing a Side-Channel based P-way Parallel PC Oracle

5.1 Experimental Setup

Our Device Under Test (DUT) is the STM32F407VG microcontroller, mounted on the STM32F4DISCOVERY evaluation board. We target the fastest implementation of Kyber KEM (m4speed version), taken from the public *pqm4* library [35], a benchmarking and testing framework for PQC schemes on the 32-bit ARM Cortex-M4 microcontroller. The target is clocked at 24 MHz. We utilize the Electromagnetic Emanation (EM) side-channel for our experiments. We obtain EM leakage using a near-field EM probe mounted on top of the chip, with the measurements collected on a Lecroy HD6104 oscilloscope, using a sampling rate of 250 MSam/sec, amplified 30dB with a pre-amplifier.

5.2 Side-Channel Methodology

Our task is to build a side-channel classifier for $m \in [0, 2^P - 1]$, using leakage from the re-encryption procedure (i.e.) $\mathsf{Re\_Encrypt}(m, pk)$. While prior works [28, 54]

Chapters 5 is published as Rajendran, G., Ravi, P., D'Anvers, J. P., Bhasin, S., Chattopadhyay, A. (2023). Pushing the limits of generic side-channel attacks on LWE-based KEMs-parallel PC oracle attacks on Kyber KEM and beyond. IACR Transactions on Cryptographic Hardware and Embedded Systems.

utilized the same leakage to distinguish between $m = 0$ and $m = 1$, we demonstrate that it is possible to classify an arbitrary number of values for m with a very high accuracy.

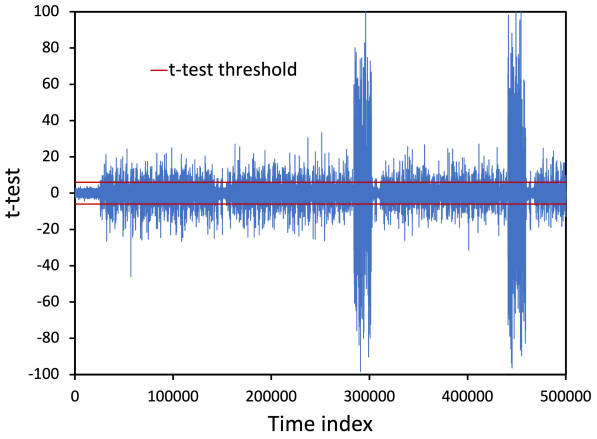


FIGURE 5.1: Welch's t-test plot computed for Tr_0 and Tr_1 for Kyber768

5.2.1 Building a Multi-Class Side-Channel Classifier

Our approach for multi-class classification, builds upon the binary classification approach using the well-known Welch's t-test [28]. We will briefly explain the binary classification method, and subsequently explain our generic extensions to arbitrary 2^P number of classes.

The binary classification is done in two phases: (1) Pre-Processing Phase and (2) Classification Phase. The pre-processing phase involves construction of side-channel templates for each class 0 and 1. The subsequent classification phase uses the templates to classify a given side-channel trace into one of the 2 classes. At no point during any of the two phases, does the attacker require to operate the target device with known secret keys.

Pre-Processing Phase: The adversary obtains T repeated measurements corresponding to $\mathsf{Re\_Encrypt}(m, pk)$ for both $m = 0$ and $m = 1$. This is done by repeatedly querying the decapsulation device with valid ciphertexts for $m = 0$ and $m = 1$ (T times each). We denote the trace set for $m = i$ as Tr_i, and the complete trace set as $\mathsf{Tr} = \cup_{i=0}^{i=1}\mathsf{Tr}_i$. The number of repeated measurements T is a parameter of the experimental setup.

- Every trace tr_j in Tr is normalized by removing the mean and dividing by its standard deviation to obtain t'_j.
- The Welch's t-test is computed between Tr_0 and Tr_1 to detect univariate leakage, based on Equation (5.1).

$$\mathsf{t\text{-}value} = \frac{\mu_1 - \mu_2}{\sqrt{\frac{\sigma_1^2}{T} + \frac{\sigma_2^2}{T}}} \tag{5.1}$$

  where μ_i and σ_i are the mean and standard deviation of trace set Tr_i.

  Refer to Figure.5.1 for the t-test plot between Tr_0 and Tr_1 for $\mathsf{Kyber768}$ ($T = 20$ traces). The plot shows several peaks about the t-test threshold of ± 5, clearly indicating significant difference between the two computations. In particular, there are two distinct peaks (over multiple samples) with very high t-test values. Upon inspection, we identified it to be the sampling of polynomials of $\mathbf{r}$ from the CBD distribution (Line 11 of $\mathsf{CPA.Encrypt}$ in Alg.1). The attacker however does not require this information to perform the attack.

- Those features/points whose *absolute t-test value* is greater than a chosen threshold $\mathsf{Th}_{\mathsf{PoI}}$ are selected as the Points of Interest (PoI) set, denoted as $\mathcal{P}$. We do not have other criteria, apart from the t-test value to select the PoI. The threshold value $\mathsf{Th}_{\mathsf{PoI}}$ is also a parameter of the experimental setup and is empirically determined.
- The set $\mathcal{P}$ is used to derive a reduced trace set for each class, which we denote as Tr'_i for $i \in \{0, 1\}$, and the mean of the reduced trace set Tr'_i is the reduced template $m_{(i,\mathcal{P})}$ for class i with $i \in \{0, 1\}$.

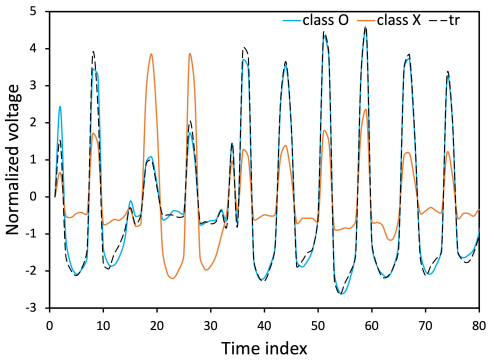
Thus, the reduced templates $m_{(i,\mathcal{P})}$ for $i \in \{0, 1\}$ are the output of the pre-processing phase. Since the target operation $\mathsf{Re\_Encrypt}(m, pk)$ depends upon both the message m and the public key pk, the pre-processing phase is not *one-time*, and therefore has to be carried out for every new public key.

Classification Phase: The reduced templates obtained from the pre-processing phase are now used to classify a given trace tr for a chosen-ciphertext, into either $m = 0/1$. The trace tr is first normalized, and the reduced trace $t'_{\mathcal{P}}$ is obtained.

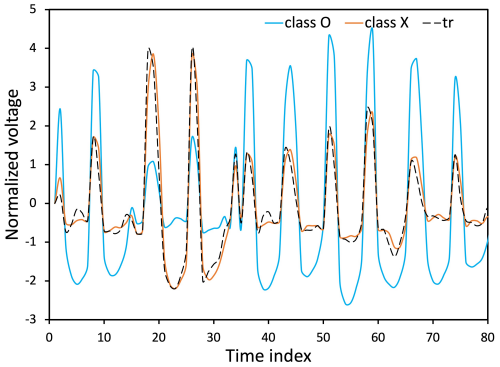
Then, the sum-of-squared difference Γ_* is computed with the reduced template of each class $m_{(i,\mathcal{P})}$ for $i \in \{0, 1\}$ as follows:

$$\Gamma_0 = (t'_{\mathcal{P}} - m_{0,\mathcal{P}})^\top \cdot (t'_{\mathcal{P}} - m_{0,\mathcal{P}}) \text{ and } \Gamma_1 = (t'_{\mathcal{P}} - m_{1,\mathcal{P}})^\top \cdot (t'_{\mathcal{P}} - m_{1,\mathcal{P}}). \tag{5.2}$$

The trace tr belongs to the class with the least sum-of-squared difference (i.e.) $\mathsf{Class}(tr) = 0$ if $\Gamma_0 < \Gamma_1$, else $\mathsf{Class}(tr) = 1$. Thus, a single side-channel trace can be used to distinguish between the two classes $m = 0$ and $m = 1$, thereby instantiating a binary PC oracle. Refer to Figure.5.2 which shows clear distinguishability of a sample trace tr into either of the two classes. The figures only show a short segment of the reduced trace for better visual distinguishability, while the reduced templates used for our experiments span a few hundreds to few thousand points. We were able to obtain a 100% success rate for the binary classification $m = 0/1$, as also shown in Figure.5.2.



(a) $\mathsf{Class}(tr) = 0$



(b) $\mathsf{Class}(tr) = 1$

FIGURE 5.2: Matching the reduced attack trace tr' with the reduced templates of the two classes $m = 0$ and $m = 1$

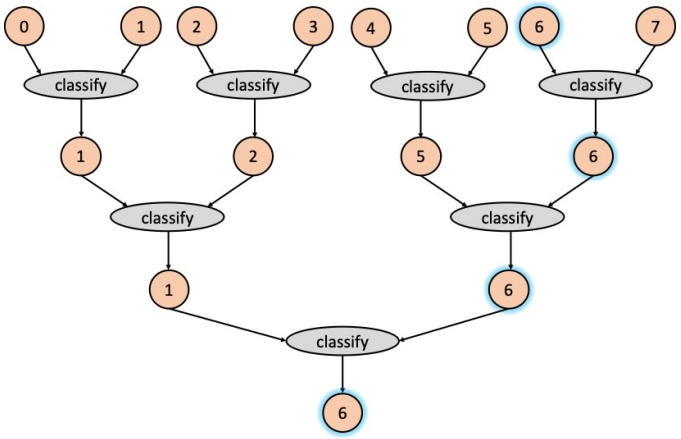


FIGURE 5.3: Illustration to classify the attack trace tr among 8 classes, m = [0,7]

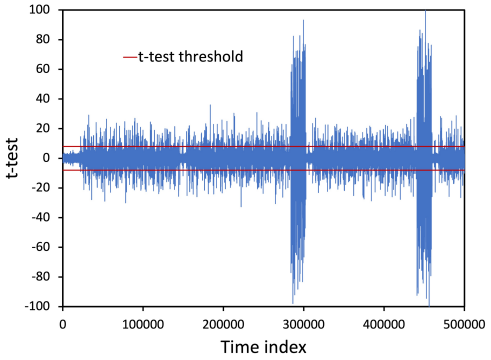
5.2.2 Towards Multi-Class Classification

Our approach towards multi-class classification is based on the observation that it is possible to classify any two random values of m in the same manner, as $m = 0/1$. This is rendered possible due to the diffusion property of hash functions used in the re-encryption procedure. For an illustration, refer to Figure.5.4 for the t-test based binary classification between $m = 330$ and $m = 559$, as illustration. It is well known that unique identification of a particular candidate within a group is possible, if there exists a pairwise classifier for every possible pair of candidates [55]. For a P-way parallel PC oracle attack, there are 2^P possible classes for m. Thus, if we are able to classify between any two pairs of m with $m \in [0, 2^P-1]$, it is also possible to uniquely identify the value of m. This applies for any generic value of P. Thus, the P-way parallel PC oracle can be realized in two phases in the following manner.

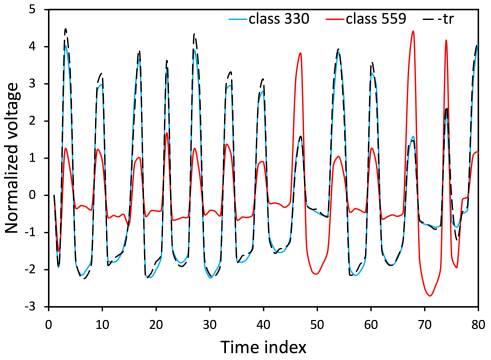
Pre-Processing Phase: The adverary collects T repeated measurements corresponding to $\mathsf{Re\_Encrypt}(m, pk)$ for all 2^P values of $m \in [0, 2^P - 1]$. The complete trace set for all classes is denoted as $\mathsf{Tr} = \cup_{i=0}^{i=2^P-1} \mathsf{Tr}_i$.

Classification Phase: Given an attack trace tr, the adversary uses pairwise binary classification similar to a knock-out tournament with 2^P players. The correct class (resp. winner) is selected after $(2^P - 1)$ pairwise classifications (resp. matches). A match in this context is nothing but binary classification of a given trace tr between two classes $m = x$ and $m = y$, which is denoted as $\mathsf{Classify}(x, y)$.

Refer to Figure.5.3 for an illustration for 8 classes (i.e.) $m = [0, 7]$, where the attack trace tr corresponds to $m = 6$.



(a) t-test



(b) $\mathsf{Class}(tr) = 330$

FIGURE 5.4: t-test plot and matching a given reduced attack trace tr' corresponding to class 330, against reduced templates for classes 330 and 559

This approach only requires $h - 1$ pairwise binary classifications for h classes, and is optimal in terms of the number of pairwise classifications necessary for unique distinguishability. There is another costlier approach, of doing pairwise classification of all possible pairs of classes and adopting a majority voting approach to select the correct class. However, this yields a much higher h^2 binary classifications for h classes. Thus, we adopt the former and more efficient approach for classification. The aforementioned technique yields the correct candidate as long as the correct candidate for m is correctly classified, when paired with any other value of $m \in [0, 2^P - 1]$. This is similar to the case of having a player, who is capable of winning against any other player in the tournament, and therefore emerges as the winner.

5.2.3 Experimental Validation

We validated our proposed attack on the speed optimized implementation of Kyber768 from the *pqm4* library (Refer Sec.5.1 for the experimental setup). We were able to achieve full key recovery with 100% success rate for a parallelization factor of $P = 10$ (1024 classes). We utilize $T = 5$ traces to build templates for each class, which amounts to 5520 traces. While this is the maximum value of P used for our experiments, it is possible to increase P to any arbitrary value. In this respect, we also verified the success of binary classification of several pairs of messages with $P = 12$ (i.e.) which amounts to 4096 classes. We were able to classify all the collected pairs correctly with 100% accuracy.

The achieved success rate significantly depends upon the the experimental setup, and particularly the SNR of the collected traces. For success rates below 100% due to the effect of random noise, it is possible to utilize majority voting based on multiple traces or utilize error correcting codes to encode the oracle responses as shown in [36, 39] to enhance the success rate. Nevertheless, our experiments provide sufficient evidence that there is enough information in the leakage available from the re-encryption procedure, that allows to recover a generic P bits of information about the secret key, while prior works [28] underutilized leakage from the re-encryption procedure to only recover a single bit. This is also due to the fact that there are several hundred PoIs for classification, to distinguish any pair of values for m.

We also believe that an upper limit for P for perfect classification if exists, is challenging to determine through experiments. It depends upon a variety of factors such as the target device and target implemenation, Signal to Noise Ratio (SNR) etc. Thus, it can only be determined empirically by attempting key recovery for different values of P, which we leave for future work. However, we show later in Section 6, that arbitrarily increasing P also exponentially increases the number of traces for the pre-processing phase, thereby decreasing the relevance of the attack. Moreover, we can see that our proposed attack is generic and clearly agnostic to the target implementation, and requires almost no information about the design of the target.

Chapter 6

Evaluating Total Cost for Key Recovery

From an attacker's perspective, the number of queries is the primary cost of the attack, as he/she looks for key recovery with minimum possible interaction with the target device. The cost of key recovery, therefore includes the number of queries for the pre-processing phase denoted as $\mathcal{Q}_{template}$, as well as for the classification phase, denoted as $\mathcal{Q}_{attack}$. The pre-processing phase requires T queries for each of the 2^P classes, which amounts to $2^P \cdot T$ queries. The number of queries in the classification phase is nothing but the total number of chosen-ciphertext queries for key recovery (Refer Eqn.4.8 and Fig.4.6 in Sec.4.4). Thus, the total number of queries for key recovery denoted as $\mathcal{Q}_{total}$ is given as:

$$\mathcal{Q}_{total} = \mathcal{Q}_{template} + \mathcal{Q}_{attack} \tag{6.1}$$

$$= 2^P \cdot T + \lceil \frac{2^8}{P} \rceil \cdot k \cdot \mathcal{Q}_{set} \tag{6.2}$$

6.1 Analysis for Partial Key Recovery

In a bid to reduce the number of traces, an attacker can also resort to recovering $m < (k \cdot n)$ coefficients of Kyber, and recovering the remaining coefficients using

Chapters 6 is published as Rajendran, G., Ravi, P., D'Anvers, J. P., Bhasin, S., Chattopadhyay, A. (2023). Pushing the limits of generic side-channel attacks on LWE-based KEMs-parallel PC oracle attacks on Kyber KEM and beyond. IACR Transactions on Cryptographic Hardware and Embedded Systems.

TABLE 6.1: Number of coefficients to be recovered, for scenarios considering attackers with different offline computational capabilities

	Full_Recovery	Partial_Recovery_2^{32}	Partial_Recovery_2^{64}
Kyber512	512	354	184
Kyber768	768	667	463
Kyber1024	1024	1010	782

suitable lattice-based solvers in an offline manner. In this respect, we consider three possible cases for an attacker with differing capabilities to perform offline computations:

1. Full_Recovery - Full Key Recovery with 0 remaining offline computations.
2. Partial_Recovery_2^{32} - Partial Key Recovery with 2^{32} remaining offline computations.
3. Partial_Recovery_2^{64} - Partial Key Recovery with 2^{64} remaining offline computations.

We utilized the leaky LWE estimator developed by Dachman-Soled *et al.* [56] to estimate the number of coefficients to be recovered, to reduce the security strength of Kyber to 2^{32} and 2^{64} respectively. The tool allows us to include exact or approximate hints and estimate the remaining cost to recover the secret. Refer to Table.6.1 for the exact number of coefficients to be recovered for the aforementioned attacker scenarios.

6.2 On the Presence of Clone Device

A close observation of Eqn.6.2 to calculate the total number of queries $\mathcal{Q}_{total}$ reveals that the cost of pre-processing phase to generate templates cannot be ignored, especially given that the pre-processing phase is required to be done for every new public key. In this respect, we identify two possible scenarios, with respect to whether or not the adversary has access to a clone device.

6.2.1 With Clone Device

In this scenario, the adversary has access to a clone device. Thus, he/she can generate templates for $\mathsf{Re\_Encrypt}(m, pk)$ from the clone device, since the computations only depend upon known values to the attacker (i.e.) m and pk. Thus, the pre-processing phase is completely taken offline. By offline, we mean that templates can be captured on the clone device, and the attacker only requires to capture traces from the target device for the key recovery phase. In this case, the number of queries to the target device, denoted as $\mathcal{Q}_{target}$ is nothing but:

$$\mathcal{Q}_{target} = \mathcal{Q}_{attack} \tag{6.3}$$

$$= \lceil \frac{2^8}{P} \rceil \cdot k \cdot \mathcal{Q}_{set} \tag{6.4}$$

Here, $\mathcal{Q}_{template} = 0$ since the pre-processing phase is carried out on the clone device.Thus, $\mathcal{Q}_{target}$ simply scales inversely with the parallelization factor P. Thus, the lower bound for $\mathcal{Q}_{target}$ is only limited by the parallelization factor P achievable on the given target device. We however recall that finding the exact limit on P is very hard to achieve in practice.

Refer to Figure.6.2(a) for the plot of number of queries to the target versus P for Kyber768, considering both full key recovery and partial key recovery. We can clearly see that the number of queries scales inversely with increase in P. For the experimentally verified case of $P = 10$, the attacker requires ≈ 232 queries for full key recovery, which improves over the state-of-the-art binary PC oracle attack [54] by a factor of $\approx 7.6\times$.

(Arbitarily) Best Case Scenario: We also consider an arbitarily strong attacker capable of building 2^{32} templates on the clone device. In this case, full key recovery is possible in just 72 queries, which is an improvement by a factor of ≈ 24.6 compared to the binary PC oracle attack. Naturally, we also observe better improvements for the case of partial key recovery as seen in Table.6.2.

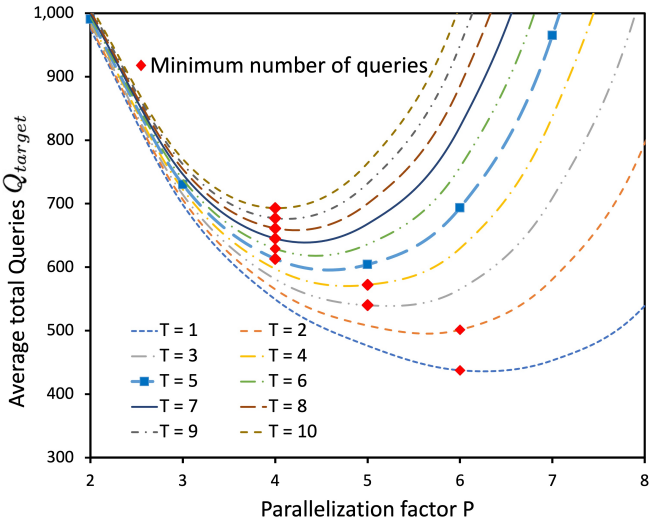


FIGURE 6.1: Total number of queries required for full key recovery for Kyber768 in the Scenario_Without_Clone versus the parallelization factor P, for different values of T, where T is the number of traces per template

6.2.2 Without Clone Device

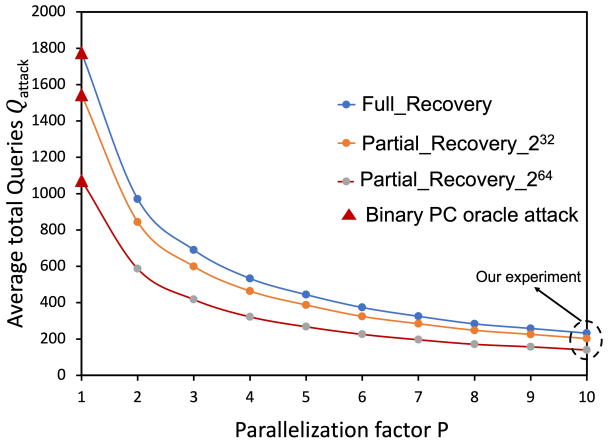
In this scenario, the adversary does not have access to a clone device. Recall that our attack is possible even without knowledge of the key in pre-processing phase. Thus, both the pre-processing as well as classification phase has to be carried out directly on the target device. Here $\mathcal{Q}_{target}$ is nothing but:

$$\mathcal{Q}_{target} = \mathcal{Q}_{template} + \mathcal{Q}_{attack} \tag{6.5}$$

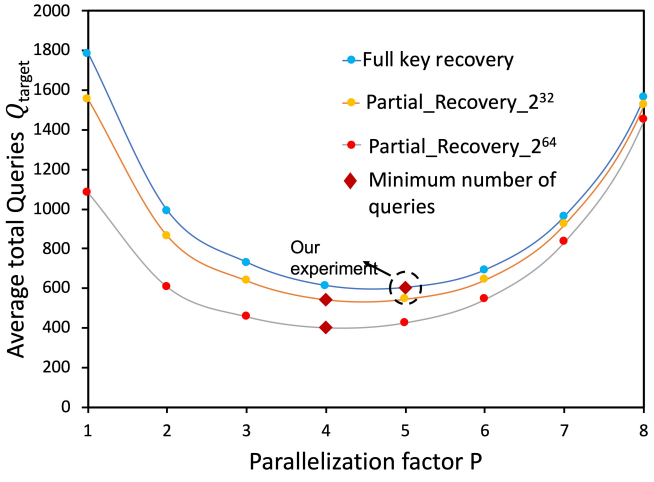
$$= 2^P \cdot T + \lceil \frac{2^8}{P} \rceil \cdot k \cdot \mathcal{Q}_{set} \tag{6.6}$$

We can observe that $\mathcal{Q}_{template}$ scales exponentially with P (i.e.) 2^P and also increases linearly with T (number of traces per template), while $\mathcal{Q}_{attack}$ scales inversely with P. Thus, there exists a fine trade-off between the cost of pre-processing and classification phase. It is not possible to arbitrarily increase P to improve $\mathcal{Q}_{target}$, as the cost of the pre-processing phase outweighs the cost of the classification phase for higher values of P, unlike when attacker has access to clone device. Refer to Figure.6.1 for the plot of number of queries versus the parallelization factor P, for different values of T. As expected, we observe a certain minima for number of queries for each value of $T \in [1, 10]$.

We experimentally verified that full key recovery is possible with $T = 5$. For $T = 5$, the parallelization factor $P = 4$ yields the lowest number of queries (i.e.) 613. This is an improvement by a factor of $\approx 2.89\times$ compared to the binary PC oracle attack. However, a lower value for T could be achieved with a better experimental setup with low acquisition noise. For the best possible scenario of $T = 1$ (single trace per template), the parallelization factor $P = 6$ yields the lowest number of queries (i.e.) 437, an improvement factor of $\approx 4\times$ compared to the binary PC oracle attack. Refer to Figure.6.2(b) for the number of queries versus P, for $T = 5$, for both full key recovery and partial key recovery.



(A)



(B)

FIGURE 6.2: Estimates for the total number of queries to the target versus the parallelization factor P for Kyber768 (a) With clone device and (b) Without clone device, and also considering partial key recovery and full key recovery

TABLE 6.2: Tabulation of the total number of queries for key recovery for Kyber768, considering attack scenarios with respect to clone device, as well as the attacker's offline computational capability.

	Parallelization Factor P					
	With Clone				Without Clone	
	1	10	12	32	4 (T=5)	6 (T =1)
Full_Recovery	1776	232	197	72	613	437
Partial_Recovery_2^{32}	1545	202	170	63	544	388
Partial_Recovery_2^{64}	1073	140	120	45	402	290

6.3 Extensions to Lightly Protected Implementations

Given the heavy performance penalty of masking countermeasures for LWE/LWR-based KEMs with upto 3.1× in runtime as shown in [31], there is significant interest in low-cost countermeasures to offer protection against known side-channel attacks. One such approach is the shuffling countermeasure, which was proposed to protect the message encoding procedure against single trace message recovery attacks [37, 48]. Moreover, leakage from the message encoding procedure was also shown to be exploitable for key recovery through the Full-Decryption (FD) oracle attack in [43]. These attacks are capable of recovering 256 bits of information from a single trace, thereby capable of message recovery in potentially less than 10 traces on unprotected implementations of Kyber KEM, especially in presence of sufficiently high SNR.

In this respect, shuffling was proposed as a concrete countermeasure against attacks targeting the message encoding procedure. Shuffling ensures that the attacker can still recover all the 256 bits of the message, but not its correct order, thereby offering protection against message recovery and also removing the presence of the FD oracle. However, Ravi *et al.* [27] showed an attack on shuffling countermeasure in a chosen-ciphertext setting and recover 1 targeted message bit per query. While shuffling does not offer concrete protection, it at least prevents single trace message recovery, and reduces the attacker's capability to only recover a single bit per query, which is equivalent to a binary PC oracle attack. This is the best known attack on such a lightly protected shuffled implementation of Kyber KEM. Thus, a simple shuffling countermeasure on the message encoding operation is able to increase the

attacker's effort from recovering the secret key in < 10 traces to a few thousand traces.

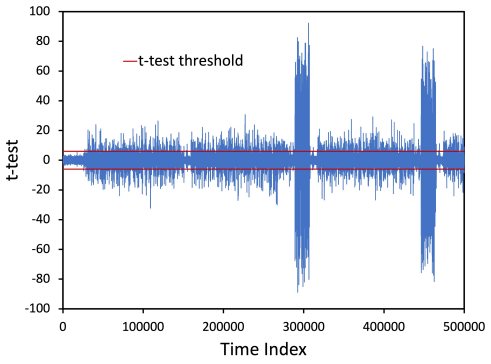
However, we observe that our proposed parallel PC oracle attack is independent of leakage from the shuffled message encoding operation. In fact, for our experiments on the unprotected implementation, we utilized leakage only until sampling of the ephemeral secret module $\mathbf{r}$ during the re-encryption procedure (Line 11 of CPA.Encrypt in Alg.1). Thus, we do not utilize leakage from the message encoding operation for our practical experiments. Thus, we hypothesize that our attack can defeat the lightly protected implementation with the shuffled message encoding operation, in the same manner as that of the unprotected implementation.

6.3.1 Experimental Results

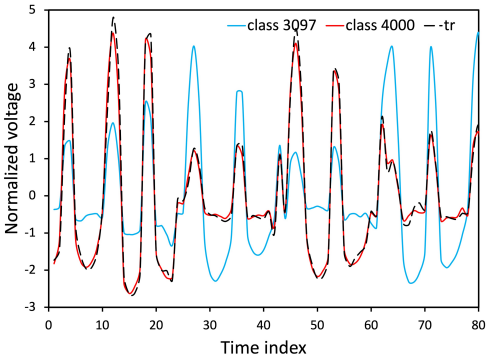
We mounted our parallel PC oracle attack on the decapsulation procedure of Kyber KEM, with a shuffled message encoding procedure. Confirming our hypothesis, we were able to successfully recover the secret key in the same manner as the unprotected implementation. The number of traces for key recovery, remains the same as that of our attack on the unprotected implementation. We also experimentally validated the capability to exhaustively recover all 2^P possible values of the decrypted message for a parallelization factor of $P = 10$, and we were able to correctly distinguish all classes only using single trace, thereby concretely demonstrating the ability to distinguish between 2^P possible values of the decrypted message. Figure.6.3 shows the t-test based classification between $m = 3097$ and $m = 4000$, as illustration for our attack on the shuffling countermeasure. Thus, our parallel PC oracle attack also serves an effective TO_Indep attack on the lightly protected implementation with the shuffling countermeasure.

6.4 Differences from [1]

In a recent and independently developed work by Tanaka *et al.* [1], authors also propose to recover multiple bits in parallel from a PC oracle attack. While the objectives of [1] are aligned with our work, we few subtle differences.



(a) t-test



(b) $\mathsf{Class}(tr) = 4000$

FIGURE 6.3: t-test plot and matching a given reduced attack trace tr corresponding to class 4000, against reduced templates for classes 3997 and 4000 with shuffling countermeasure

Number of Oracle Queries: We observe that the BDTs for optimal binary PC oracle attack are not always optimal for the parallel PC oracle attack. We demonstrate that BDTs with minimum depth (i.e.) BDT_{min_depth} are optimal for a high parallelization factor P, compared to BDTs with minimum entropy (i.e.) BDT_{min_ent} (Refer Section 4.5.1). However, [1] use the same BDTs that were used for the binary PC oracle attack, to also perform their attack in the parallel setting. Refer Tab.3(a) of [1] for the BDT used to attack Kyber512 in the parallel setting and Tab.9(a) of [57] for the BDT used to attack Kyber512 in the binary setting. Thus, our approach to construct optimal BDTs for the parallel PC oracle attack yields lower number of queries for Kyber, even in the presence of a perfect parallel PC oracle.

Side-Channel based Oracle: The authors of [1] utilized a multi-class classification neural network (NN) to realize a parallel PC oracle, while we utilize a

simple t-test based classifier. We observe that the attacker needs to carry out the pre-processing phase to create templates for every new public key, *it is important to minimize the sum of traces for the pre-processing phase and key recovery phase.* This is especially applicable in the scenario where the attacker does not have access to a clone device. Thus, we chose t-test based classifier with an objective to reduce the no. of traces during pre-processing phase, typically $T < 10$ for each class. On the other hand, NN-based classifiers are typically known to require a very high number of traces for training. [1] utilize 1000 traces for training and 500 traces for validation for each of the 2^P classes. While NN-based classifiers are suitable for training on the clone device (offline), they are sub-optimal when there is no access to a clone device.

Moreover, we perform experiments on a full implementation of Kyber to realize the parallel PC oracle, while [1] perform experiments on implementations of SHAKE,SHA3 and AES. Thus, we are able to exploit leakage from several operations within the re-encryption procedure which enable us to yield a 100 % success rate over single traces to realize a practical parallel PC oracle.

6.5 Applicability to other PQC schemes

While we present our parallel PC oracle attack for Kyber KEM, we also believe that our attack can be adapted in a straightforward manner to other lattice-based schemes such as Saber and Frodo. This is because our technique to construct chosen-ciphertexts not only applies to Kyber, but to the broader framework of the LPR encryption scheme [58], which forms the core of several lattice-based KEMs such as Kyber, Saber [40], NewHope [59], Round5 [60], LAC [61] and Frodo [41]. We recall that the binary PC oracle attacks proposed in several prior works [28, 38] have been shown to be adaptable to several LWE/LWR-based schemes including Kyber and Saber, albeit with appropriate changes in the actual value of the chosen-ciphertexts and the number of queries for key recovery. We also experimentally validated our attack on Saber through simulations of a perfect parallel PC oracle. We would like to briefly sketch the idea to perform parallel PC oracle attack on Saber.

Similar to the chosen-ciphertexts for Kyber, we choose $\mathbf{u} = k_u \cdot x^0$ and $\mathbf{v} = k_v \cdot (\sum_{i=0}^{i=(P-1)} x^i)$ where $t \in \mathbb{Z}^+$. Thus, the decrypted message m is given as:

$$m_i = \begin{cases} \mathsf{Decode}(k_v - k_u \cdot \mathbf{s}[i]), & \text{if } i \in [0, P-1] \\ \mathsf{Decode}(-k_u \cdot \mathbf{s}[i]), & \text{for } i \in [P, n-1] \end{cases} \tag{6.7}$$

For the recommended parameters of Saber, the secret coefficients are in a slightly larger range of $[-4, 4]$ compared to Kyber768 with coefficients in $[-2, 2]$. Please refer to Fig.6.4 for the BDT with minimum depth we were able to achieve for the recommended parameter sets of Saber.

For ($k_u = 57$) and different values for $k_v \in [1, 7]$, we were able to uniquely distinguish candidates for the secret coefficient in the range $[-2, 4]$ in not more than 4 queries. However, we were not able to distinguish between candidates -3 and -4 using $k_u = 57$. Thus, we have to utilize an additional query of $(k_u, k_v) = (54, 1)$ to distinguish between -3 and -4. However, in case of Kyber, we were able to distinguish all candidates between $[-2, 2]$ using the same value of k_u, in not more than 3 queries (Refer Fig.4.2). We estimate that, for parallelization factor $P = 10$, we would require approximately 390 queries in the key recovery phase for full key recovery, compared to 232 for Kyber768 (Refer Tab.6.2). Thus, we can see that our technique can be easily adapted to Saber, albeit with differences in the number of traces for key recovery. Similarly, we believe our attack can also be adapted to other LWE/LWR-based KEMs such as NewHope, Round5, LAC and Frodo which are based on the LPR encryption scheme. However, adapting our attack to other lattice-based schemes based on the NTRU paradigm such as NTRU [62], NTRU Prime [63] is not trivial, and thus consider them as potential future work.

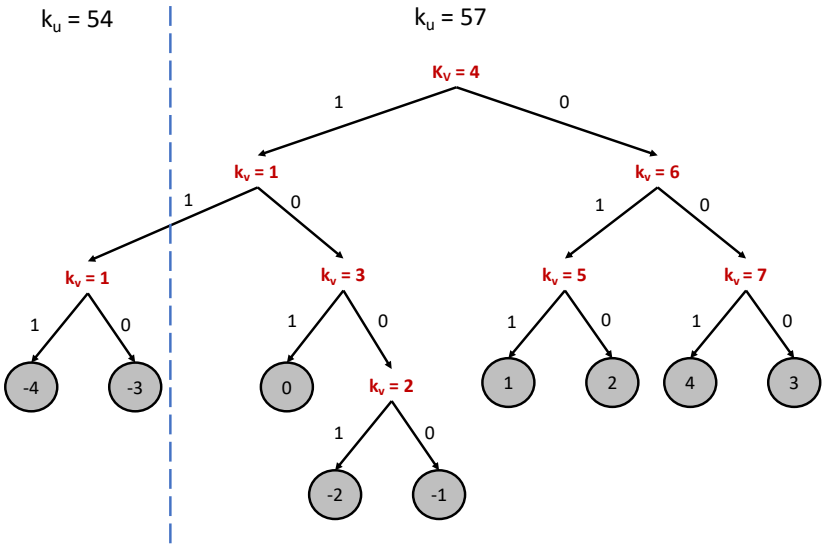


FIGURE 6.4: Optimal BDT for Parallel PC oracle based CCA on Saber (recommended parameters)

Chapter 7

Summary

In this work, we propose novel *parallel PC oracle* based side-channel attacks, which are capable of recovering an arbitrary P number of bits of information about the secret key in a single trace. We experimentally validated our attacks on the fastest implementation of unprotected Kyber KEM in the *pqm4* library. Our experiments yielded improvements in the range of 2.89× and 7.65× in the number of queries, compared to state-of-the-art binary PC oracle attacks, while arbitrarily high improvements are possible given the generic nature of the attack. We also show that our proposed attacks are able to achieve the lowest number of queries for key recovery, even over implementations protected with low-cost countermeasures such as shuffling. Masking serves as a concrete countermeasure against our proposed attacks, and therefore we believe our work stresses the strong need to implement masking countermeasures for lattice-based schemes, particularly for embedded applications. The future directions of our presented work are finding theoretical optimal attacks in the context of parallel PC oracle and applicability in the hardware targets.

Chapters 7 is published as Rajendran, G., Ravi, P., D'Anvers, J. P., Bhasin, S., Chattopadhyay, A. (2023). Pushing the limits of generic side-channel attacks on LWE-based KEMs-parallel PC oracle attacks on Kyber KEM and beyond. IACR Transactions on Cryptographic Hardware and Embedded Systems.

Part II

Security beyond CMOS: Emerging NVM

Chapter 8

Literature Review

Synopsis

Nowadays advancements in the design of trusted system environment are relying on security provided by hardware-based primitives, while replacing resource-hungry software security measures. Emerging non-volatile memory devices are promising candidates to provide the required hardware security functionalities at very low area-energy-runtime budget. Resistive random access memory (RRAM) offers high-density integration with outstanding performance among the state-of-the-art NVM devices. The RRAM device technology is currently getting significant attention from both academia and industry for constructing beyond Von Neumann architectures. This technology's strength is its scalable two-terminal structure, the availability of wide range of functional materials, and multi-bit storage capability. The fluctuations in switching resistances, random telegraph noise, and sneak path current are detrimental characteristics of RRAM integrations for storage and in-memory computing applications, and more research focuses on alleviating these effects. Interestingly, these characteristics make them suitable for designing security hardware. In recent times, there has been significant progress in the design and analysis of RRAM-based security primitives such as Physical Unclonable Function (PUF), True Random Number Generator (TRNG) and hash function. This review

Chapter 8 is published as Rajendran, G., Banerjee, W., Chattopadhyay, A., Aly, M. M. S. (2021). Application of resistive random access memory in hardware security: A review. Advanced Electronic Materials, 7(12), 2100536.

discusses the detailed developments in RRAM security and presents the demanded security requirement and available opportunities to be explored.

8.1 Overview

The rapid digitalization and advancements in computational intelligence have propelled the world towards a data-driven society in the last decade. Ubiquitous computing is a reality today, with the low power embedded devices deployed for sensing, actuating and running intelligent decision-making algorithms. The advancement in wireless communication technologies such as WiFi, Bluetooth low energy (BLE), 4G-LTE, LoRaWAN and recently 5G millimetre wave communication coupled with huge boost in the computing capabilities enable these devices to exchange and process data both at the edge and cloud. This intelligent environment is now widely explored as the Internet of Things (IoT). One of the primary concerns in this development is to security. Most of the available protocols and encryption techniques for device authentication and data transfer are purely software measures, thus presenting an opportunity for a determined attacker to bypass those with higher computing power or uncover the secrets through information leakages in physical forms. Furthermore, software-driven security demand high system resources, which is unavailable in resource-constrained IoT devices, therefore necessitating robust hardware security solutions that can be integrated into those devices. On the other hand, globalization of the semiconductor industry supply chain mandates a careful auditing and trust management during the fabrication and system integration process, which can very well benefit from the inherent randomness in the manufacturing processes. Hence, hardware security research today predominantly focuses on designing security primitives that tap onto the manufacturing variations, thereby constructing primitives like True Random Number Generator (TRNG), Physical Unclonable Function (PUF) and cryptographic hash functions.

The emerging non-volatile memory (NVM) devices such as resistive random-access memory (RRAM), spin-transfer torque magnetic random-access memory (STT-MRAM), spin-orbit torque magnetic random-access memory (SOT-MRAM) and ferroelectric field-effect transistors (FeFET) are currently explored for building beyond Von Neumann architectures. Among them, RRAM is well established today.

It is a two-terminal device commonly known for its metal-insulator-metal (MIM) structure. This device technology's strength is the available wide range of functional materials that can be engineered for reliable resistive switching. It includes binary transition metal oxides, 2D materials like graphene, perovskites and several organic materials. The RRAM devices are demonstrated to be scalable below 10 nm[64] with an achievable switching time of fewer than 1 ns[65] and multibit storage per cell[66]. These are the attributes commonly expected in the commercial NVM market, and consequently, today's major chip manufacturers are investing in RRAM.

The RRAM integrations are being studied for storage, realizing in-memory multi-valued logic processing, neuromorphic computing, and security applications. One of the primary concerns in RRAM array structures is variability. The switching resistances are not uniform between the devices and to the same device for every programming cycle. The sneak path current through the unselected cells and random telegraph noise are other major issues that disturb the array operation. Interestingly, even though most research work focuses on alleviating these stochastic effects, yet these effects turn out to be valuable for constructing hardware security primitives such as PUF, TRNG and hash function. There is significant progress recently in this direction, and the RRAM security primitives are promising to be integrated into resource constraint IoT systems. Most of the reviews till now lack detailed developments in RRAM security in a systematic manner. This survey aims to cover those aspects and present available opportunities. The rest of this manuscript is organised as following. We introduce the RRAM cell operations and their performance in section II. In section III, we first discuss the concepts of TRNG, its implementations in cryptographic applications and the performance benchmarking criteria. The later part of that section focuses on RRAM TRNG developments, comparison to other technologies and future opportunities. Similarly, the concepts of PUF, implementations, benchmarking criteria, research progress, comparison and future outlook are discussed in section IV. We discuss the hash function, its requirements and the RRAM implementations in section V. Finally, we present the general challenges in hardware security applications and summarise the discussions.

8.2 Basics of resistive random access memory

There are different mechanisms that govern the resistive switching of the memristor devices. Nonetheless, the basic operation is still the same. Various types of resistive switching, their design methodologies, and applications are summarized in Fig.8.1. In general, a resistive switching device can switch from the high resistance state (HRS) to the low resistance state (LRS) and vice-versa depending on the applied set and reset voltages, respectively. The simple stack of RRAM can be designed in diverse ways, essentially running permutations and combinations of structures and materials. Detail about the engineering methodologies of RRAM is reviewed previously by many research groups.[2, 67–71] Here we are focusing on the two most common memristor types i.e. conductive bridge random access memory (CBRAM) [66, 72–77] and oxide random access memory (OxRAM). [78–88] The CBRAM also can be referred to as electrochemical metallization (ECM) cell, is based on the metal cation migration process, and the OxRAM is based on the oxygen or oxygen vacancy (Vo) migration process.

8.2.1 Metal-filament based Conductive bridge random access memory

The cells based on CBRAM consists of electrochemically active and inert electrodes separated by a solid electrolyte or insulator. The active electrode (AE) gets oxidised and undergoes dissolution on the applied set bias voltage as detailed in Fig.8.2. The high electric field now drifts the metal cations formed at the AE through the separating layer. At the inert electrode surface, the cations get reduced, and the nucleation process begins with the metal filament formation towards the AE. When the filament reaches close to the AE, electron tunnelling begins, and the cell is switched to the low resistance state (LRS) [72, 89, 90]. The opposite polarity reset voltage is applied to rupture the conductive filament to switch back to the high resistance state (HRS) [91, 92]. A higher voltage, called forming process, is required to switch the virgin cell for the first time. It is often explained by nanochannel formation in the insulating layer of CBRAM, which persists for subsequent switching, [72, 93] and there are also studies in developing forming-free CBRAM cells [94, 95]. In CBRAM devices, AE is the most crucial part to dominate the switching. Due to faster electrochemically dissolving probability Ag and

Cu are widely used for the AE. Other materials such as Al, Au, Zn, also has been studied [96–98]. Lübben et al. investigated selecting the AE material based on Gibbs free energy [99].

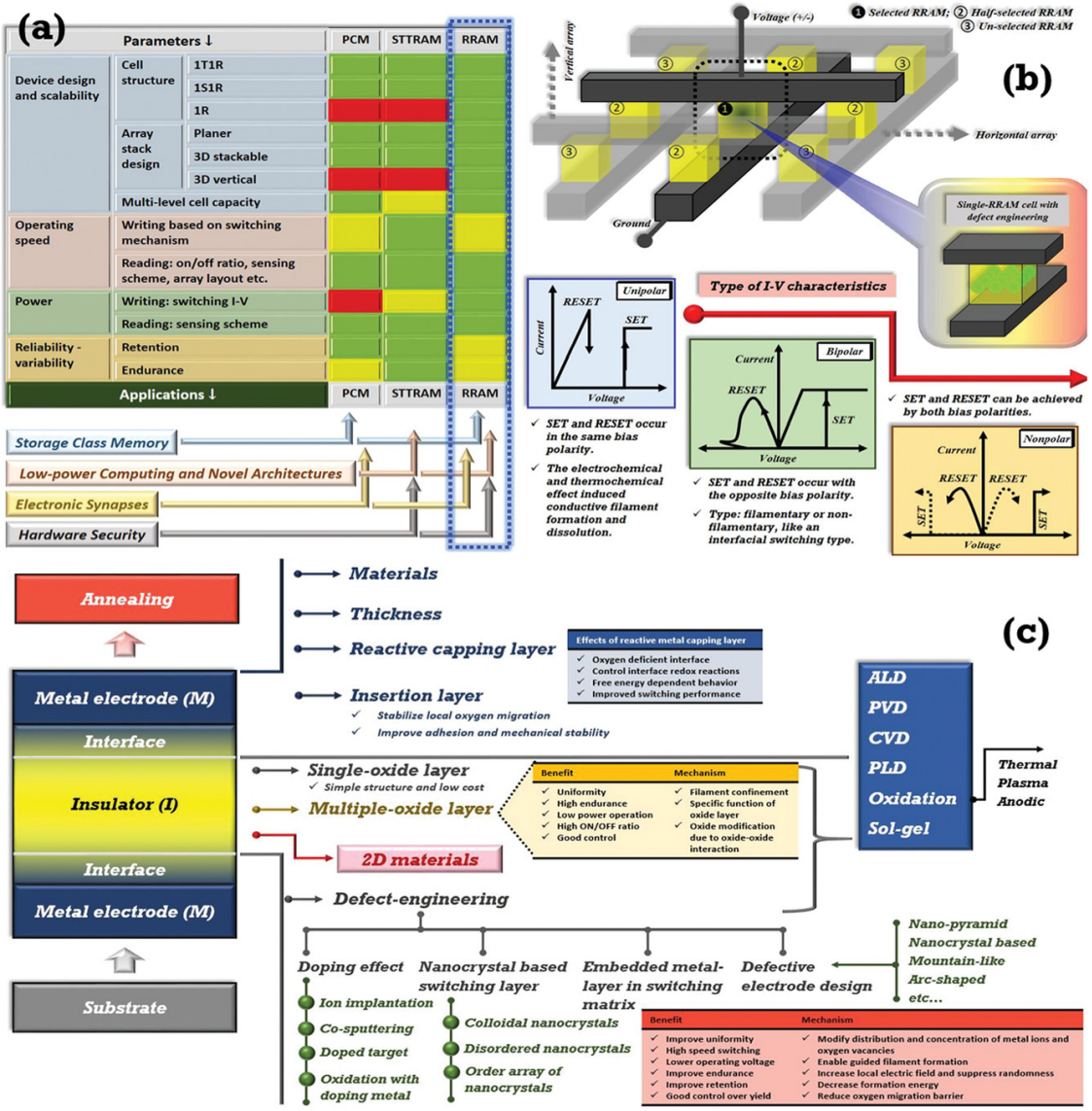


FIGURE 8.1: a) Comparison of different emerging devices, their cell structure, and applications. The RRAM is the most promising candidate for all applications. Especially, the variability issue of resistive switching arises from uncontrolled defect, makes it highly attractive for hardware security applications. b) Schematic illustration of a 3 × 3 crossbar array of RRAM and different types of I-V characteristics. The array can be extended in the horizontal and vertical directions by increasing the density of devices. c) Materials and design of the RRAM devices. Interestingly the defect engineering has the tremendous potential to modulate the entropy sources in RRAM devices for hardware security applications.[2]

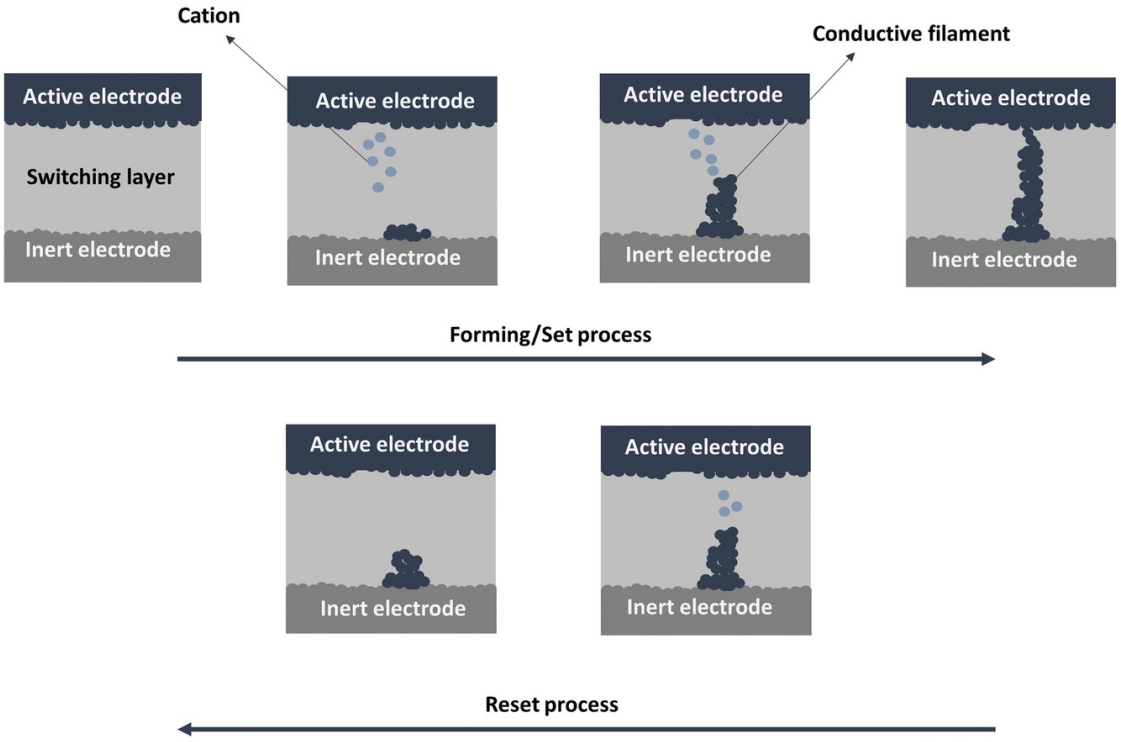


FIGURE 8.2: Resistive switching mechanism of conductive bridge random access memory.

The W and Pt are commonly used as the inert counter electrode, and the former is well established at the industrial level [72, 100]. Some of the widely studied switching layer materials are solid electrolytes such as chalcogenides of Ge, including GeSx [101, 102], GeSex[103] and their Ag-doped [104], and other common insulators including SiO2, TiO2, HfO2. In the electrolytes containing cations, the reduction can happen readily at the counter electrode [105]. The traditional CBRAM type devices are suffering with current-retention dilemma. The minimum size of filament is limited to atomic dimension i.e. 1-atom conductance, comparable to Go = 2e2/h, is not suitable for low power automotive applications. Recently, subquantum CBRAM devices are in the focus of academia and industry research.[106, 107] Several materials including Te-based alloys are used to fabricate subquantum CBRAM devices. Jameson et al.[108] reported that the reliability of subquantum CBRAM can be improved from ZrTe/Al2O3 to HfTe/SiO2 devices. However, Jiang et al.[109] have been used the randomness of subquantum-Ag/SiO2 based threshold switching devices for security applications. Dang et al.[110] have reported Ag/MgOx/Ag based threshold switching devices for TRNG applications with 1400 bits. Hence, depending on the applications and its requirements the CBRAM type devices can be designed.

8.3 Vacancy-filament based Oxide random access memory

As the name suggests the switching in vacancy-filament based oxide random access memory is dominated by the vacancy filament formation through the bias driven movements of oxygen ions from the lattice sites. There are different explanations given to the defects assisted switching mechanism involved in OxRAM. Menzel et al. have reviewed some of the conduction mechanisms[111]. Here we refer to a commonly accepted description of their operation. In bipolar switching devices, the electric field drifts the oxygen anions towards the anode by oxide layer soft breakdown on the applied set bias. The conductive filament formed now, switching the device to LRS, is by the resultant oxygen vacancy defects created. Hence, it is also called the valence change mechanism (VCM) cell. The top electrode is usually an oxidizable material such as Ti[112], TiN[113], and the inert material such as Pt, Au is used as the bottom electrode. Some of the commonly investigated switching layer materials are HfOx[114, 115],TiOx[116],AlOx[117],TaOx[118].The virgin state cell's available defects are not adequate for the conduction mechanism though it can be tailored with material engineering. Hence, the forming process with higher voltage is required to increase the defects, and it is utilized for subsequent cycles[69, 119]. During the set operation, the migrated oxygen anions react with anode material and form the interfacial layer, often referred to as the oxygen-exchange layer [120, 121]. During reset operation with the opposite bias, the electric field helps overcome the diffusion barrier at the interface oxide layer to mobilize anions for partial rupturing of conductive filament,[122] and the device now switches to HRS. If both the electrodes are inert, then the resistive switching is generally unipolar type, and the switching is explained with the thermochemical mechanism (TCM). Nickel oxide (NiO)[123–125] is most widely investigated for this unipolar behaviour. During the forming process, the dielectric breakdown is induced thermally, and the resulting conductive filament formed is dominated by the metallic phase transition of the oxide material. As with other mechanisms, the set voltage required for subsequent cycles is lower compared to this forming step, and in both the set and forming step, the current is limited by the current compliance. This is not followed during the reset process, and the high current induced Joule heating ruptures the conductive filament [126, 127]. Apart from the filamentary switching there are several other type of resistive switching devices which usually

refer as non-filamentary switching. Different types of resistive switching devices are reviewed previously [2, 67–71]. Nevertheless, due to the filament formation in the filamentary switching, it usually suffers with higher randomness as compared to non-filamentary switching devices. Depending on the design of the RRAM devices it is also possible to alter the switching behavior[86]. The performance of the VCM type of devices can be controlled through the control of the oxygen vacancies or defects. The uncontrolled vacancy based devices produces randomness which are extremely useful to design VCM-type systems for security applications. Previously Lin et al.[128] have discussed the TaOx/HfOx-based 1T1R devices for PUF applications. Apart from the typical memory switch devices, oxygen vacancy based ovonic threshold switching devices are also verified for TRNG applications. Recently, Kim et al.[129] have used stochastic self-oscillation behavior of Ti/NbOx/Pt based ovonic threshold switching device for TRNG with 130 Mbits, which passed all NIST 800-22 random number tests.

8.3.1 Impact of hybrid filament in resistive switching

Recently several researchers [3, 130–132] have pointed out that instead of having a pure metallic filament based ECM type device or Vo filament based VCM type system, the RRAM devices having a mixed metal ions and Vo combined filament has the potential to show highly efficient resistive switching performance. The hybrid type mixed filament-based devices are generally termed as hybrid-RRAM. Sassine et al.[132] has investigated the coupling of Vo's and metal ions. To identify the impact of Cu and Vo in Cu/Ta2O5 based RRAM devices during switching process, the ToF-SIMS measurements were performed as shown in Fig.8.3(a,b) for Cu and oxygen, respectively. In the ON state (LRS), the amount of Cu decreases in the CuTe2Ge/Ti electrode side and increases inside Ta2O5. However, the situation is different in the case of oxygen anions. The amount of oxygen is increases in the CuTe2Ge/Ti electrode side and decrease inside Ta2O5. The Cu content diffusion towards the bottom electrode side is a typical CBRAM type behavior. On the other hand, the decreasing oxygen content in the dielectric layer is leading to the formation of Vo and behaves like OxRAM. Here we must mention that both the Cu and oxygen profile evolves with sputtering time under the application of external bias. This type of device known as hybrid-RRAM as the filament is constructed by diffused metal guided by formed Vo's. Atomistic simulations combined with defect

formation enthalpy and migration energy barrier were performed to understand the most favorable form of the filament in the Cu-based hybrid-RRAM with various dielectrics such as HfO2, Ta2O5, Al2O3, GdOx, etc... The formation energy i.e. the total energy difference between an initial and a final state after introducing a new defect is the lowest when the filament consisted with Cu interstitial next to Vo i.e. CuiVo. During the memory operation, only CuiVo defect or both the CuiVo and Vo defects can move. In the case of HfO2 dielectric, the formation of Vo and CuiVo are more favorable. However, the formation of Cui and CuiVo are favorable in the case of Al2O3, Ta2O5. Nevertheless, the formation of hybrid filament (CuiVo) is energetically the most suitable in hybrid-RRAM devices. A comparative study of diffusion barrier of Cui, Vo, and CuiVo defects in different dielectrics are shown in Fig.8.3(c,d) for Cu/Al2O3 and Cu/HfO2, respectively. Depending on the dielectric layer and device structure the construction of the filament can change. In Cu/Al2O3 based devices the hybrid CuiVo is the most favorable filament structure whereas, in the case of Cu/HfO2, both the Vo and CuiVo are most favorable.

In another work Banerjee et al.[131] have reported that AgiVo based hybrid filament in Ag/HfO2 device is the most suitable to extend the device performance as compared to any other type of device. Fig.8.3(e) shows the binding energy variation depending on the distance between Agi-Vo and on charge state. High binding energy is estimated when the distance between Agi-Vo is 0.2 nm with similar binding energy for charge state 3. The binding energy calculations with an excess electron in Fig.8.3(f) shows that at least 2 excess electrons are required to maintain the positive binding energy i.e. high ON current when the filament is AgiVo whereas, for a filament with only Agi the high ON current can be maintained with lower excess electron. Therefore, to design a volatile threshold switch type device AgiVo filament is the most suitable candidate. Here we must mention that due to the higher diffusion probability of Ag ions as compared to Cu ions, Ag electrode is a common electrode material to design volatile threshold switch. The AgiVo based filament model is shown in Fig.8.3(g). The binding energy of such filament is negative in the ionized state and defines the insulating density of state whereas the positive binding energy is estimated when the filament neutralized with excess electron. The binding energy difference during ionized and metallic filament is shown in Fig.8.3(h,i), respectively. The hybrid-filament of AgiVo can produce highly efficient volatile threshold switching devices with ultra-low OFF current $<$

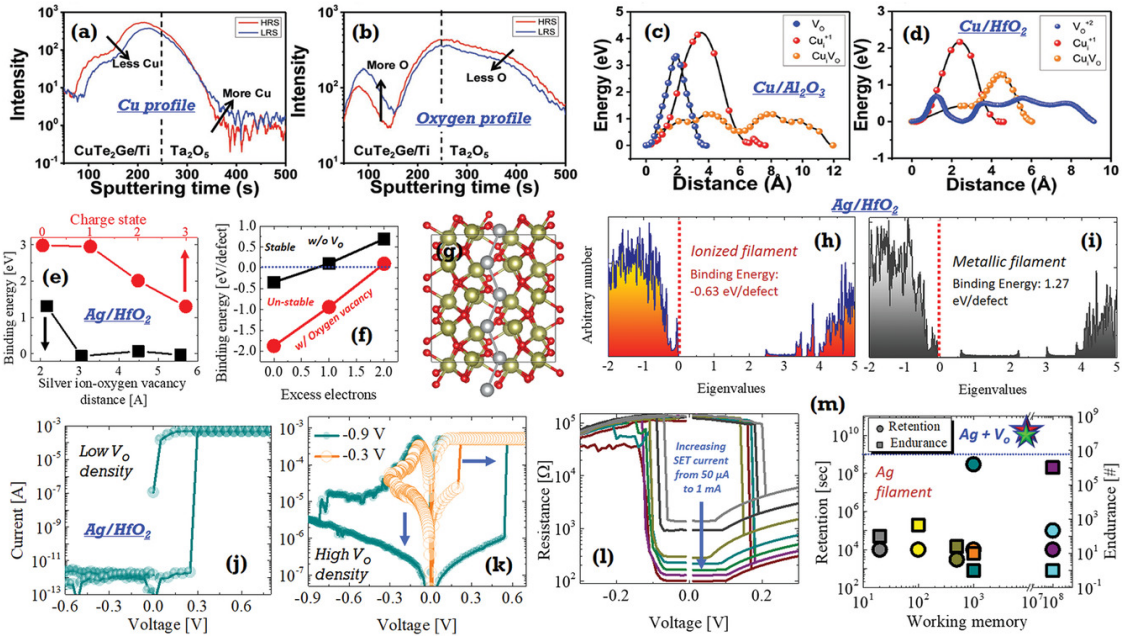


FIGURE 8.3: ToF-SIMS measurement for a) Cu profile and b) oxygen profile in CuTe2Ge/Ta2O5 based devices. Energetically CuiVo filament is favorable in c) Cu/Al2O3 and d) Cu/HfO2 devices. a–d) Reproduced with permission.[71] Copyright 2021, Wiley-VCH. e) Binding energy variation with Agi-Vo distance and charge state. f) Binding energy variation with excess electron. g) The AgiVo based hybrid-filament model in Ag/HfO2/Pt RRAM device. The binding energy variation for h) ionized filament and i) metallic filament. Vacancy density and hybrid-filament structure dependent j) threshold switching, k) memory switching, l) resistance variation. m) The performance comparison for Ag-filament and AgiVo-filament. j–m) Reproduced with permission.[3]

1 pA, high ON current 1 mA, ultra-high selectivity 109, steep-slop $<$ 2 mV/dec with excellent endurance $>$109 cycles. Point to be noted, the maximum endurance of the devices is dependent of maximum selectivity and conductance. The interdependency of selectivity and endurance is reported recently.[133] A selectivity $>$1010 can be achieved but a selectivity of 108 is suitable to switch for longer cycles. In the quest for a hybrid filament, the amount of Vo can play an important role. Unlike a low Vo density which can produce a volatile threshold switch (shown in Fig.8.3(j)), a high Vo density can produce tunable non-volatile memory switching (shown in Fig.8.3(k)) behaviour having the same Ag/HfO2 based structure. [3] In general, the tuning of resistance states in Ag-based memory switching devices is challenging due to highly diffusive Ag electrode materials. But in the presence of modulated Vo in HfO2 matrix, several shades of storage can be achieved even in Ag/HfO2 based memory devices. As shown in Fig.8.3(l), the resistance states can be controlled with the variation of the set current compliance from 50 μA to 1 mA. A

comparative study (Ag-based devices) Fig.8.3(m) shows that AgiVo based hybrid-filament devices can achieve the best data retention >1010 sec, best endurance >109 with a moderately high working memory >103. A similar for Cu-based HfO2 devices shows that high endurance of 1010 can be achieved with a working memory of 103 for CuiVo based filament as compare to the normal CBRAM or OxRAM devices.[132] Note that, the variability of the switching in hybrid-RRAM devices can be controlled with tuning of defect density.

8.3.2 Emerging two-dimensional (2D) materials based resistive switching devices

In the past several years, the researchers are rigorously focusing on the development of two-dimensional (2D) materials-based RRAM devices. For detail about the development and mechanism of 2D materials-based RRAM devices, the readers can go through the following references [134–137]. The actual journey of 2D materials in the RRAM system were started from graphene and then hexagonal boron nitride (h-BN).[138–145] After that several other 2D materials have investigated for RRAM such as MoS2, Si2Te3, WTe, etc.[5, 146, 147]. Interestingly graphene has been used in RRAM devices in many ways such as electrode material, as the active layer, a barrier layer, and so on. Lee et al.[148] have successfully scaled down the electrode thickness to 0.3 nm by using an atomically thin graphene layer in the vertically stacked 3D RRAM structure. The lowest power loss of those atomically thin electrode devices attributed to the ultra-thin nature of the graphene electrode. As an active layer in RRAM, graphene has been used with a monolayer or multi-layer structure. Using a single graphene sheet, Wu et al.[149] have reported a switching ratio > 106. Zhao et al.[4] has used nanopore graphene layer as a barrier layer to Cu diffusion in Cu/nanohole-graphene/HfO2/Pt structure. The graphene nanoholes were fabricated using electron-beam lithography followed by a plasma etching process. The conductive atomic force microscopy (C-AFM) image under voltage ramping measurement mode is shown in Fig.8.4(a). After forming process, the conductive filament can be observed at a current state of 1.2 µA whereas after the reset operation the conductive filament has not existed, confirming the filamentary switching process in the fabricated devices. Further investigation of the I-V performance shows the impact of nanohole-graphene over the control devices (without nanohole-graphene) in Fig.8.4(b). The minimum operating current level

of >50 µA is needed for the control devices to show memory switching. However, the nanohole-graphene devices can show memory switching at a low current level of 200 nA. The reason behind this improvement is the localization of the conducting filament formation. In control devices the Cu ion can diffuse abruptly from the top Cu electrode, resulting in poor control over the filament formation. Unlike the control devices, filament formation in nanohole-graphene devices is only possible through the nanohole regions, which can make a stronger filament even at low current levels. Furthermore, Wu et al. [139] have reported the possibility to utilize graphene as a barrier layer without going through the costly nanohole patterning process. The Ag/graphene/HfO2/Pt devices can perform as memory switching at several µA current levels with good data retention as compared to the non-graphene devices. Zhao et al.[143] have reported that using graphene barrier layer it is possible to control switching at ultra-low power of femto-joule.

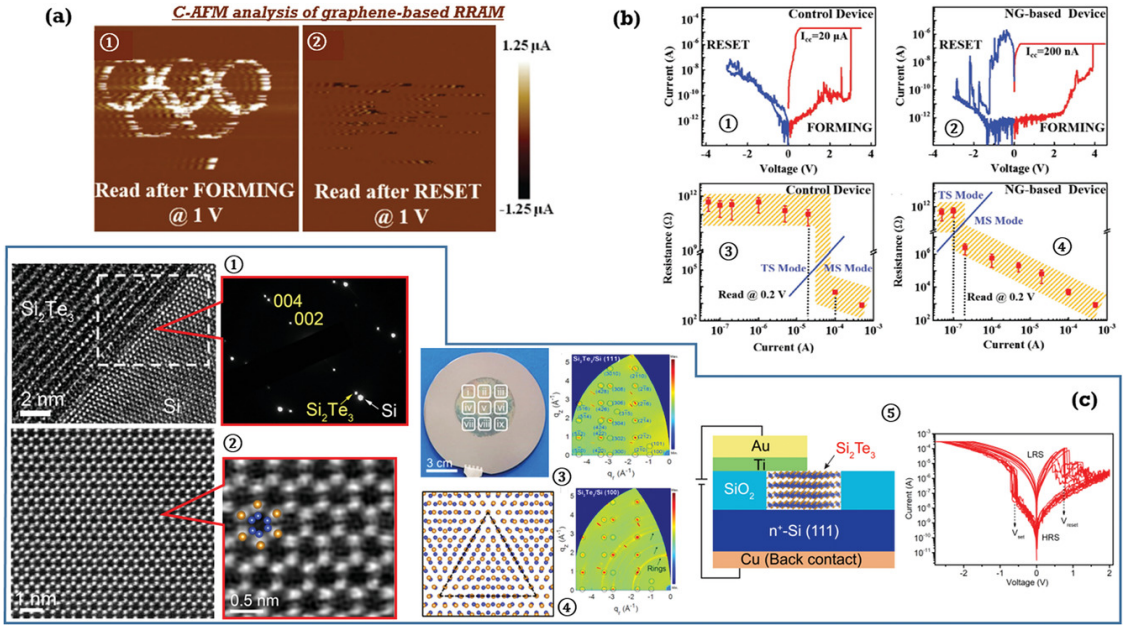


FIGURE 8.4: a) The C-AFM analysis of filament forming in nanohole-graphene based RRAM devices. b) A comparative study of electrical performance variation using nanohole-graphene in RRAM devices. [4] c) The wafer scale integration of Si2Te3 for RRAM applications.[5]

Apart from graphene-based devices, h-BN is also studied extensively in RRAM research. Shi et al.[140] has reported h-BN-RRAM devices. The multi-layered h-BN was confirmed through HRTEM analysis and the cross-sectional HRTEM image of 5-7 layers and 15-18 layers based multi-layered h-BN devices were investigated. The thickness of CVD grown h-BN layer can be modulated by process conditions

like growth time, temperature, pressure, and gas flow. The physical analysis identifies that the h-BN stack contains native defect states with lattice disorders and thickness fluctuations. Moreover, the lattice disorders generate defects across the h-BN layer. The h-BN based devices can show volatile threshold switching and nonvolatile memory switching at 1 nA and 1 µA, respectively. Recently Nikam et al. [150] pointed out the possibility to control single atomic contact through h-BN based RRAM devices. Wafer-scale single grain Si2Te3 based RRAM devices are reported by Giri et al. [5] as shown in Fig.8.4(c). The HRTEM image of a Si2Te3 thin film has a nice interface with Si, which is further confirmed by selected area electron diffraction pattern indicates a perfect alignment of Si2Te3 to the [00l] direction (①). A cross-section high-angle annular-dark-field STEM images of the 3.6 nm thick Si2Te3 the film shows that all the Te atoms (bright spots) are hexagonally-close packed (②). The fabrication was done in wafer-scale where the diffraction spots from the Si2Te3 are labeled (③), which indicates the single grain crystal Si2Te3 film over the whole wafer. The 9-unit mesh of Te coincides with the 10-unit mesh of Si, shows the perfect epitaxial alignment of the lattice (④). The schematic of the fabricated Si2Te3 based RRAM devices are shown with the resistive switching I-V characteristics (⑤). In this case, the vacancy filament or metal filament is not the dominating switching method, however, structural changes may be the origin of the high-speed resistive switching $<$100 ns. The switching variability is the major concern of this kind of device. The progress of 2D material-based RRAM device is also extended in security applications. Recently, Wen et al.[151] have reported a few-atom-wide defect based h-BN-based TRNG with high degree of randomness and high throughput of 1Mbit/s.

8.3.3 Design of resistive switching array

There are different RRAM array integration forms, such as 0T1R, 1T1R and 1D1R/1S1R. There are many reviews available on this, and hence, we present here an overview of them[152, 153]. In the 0-transistor 1-resistor structure, the RRAM cell is directly sandwiched between the perpendicular metal wires, and hence the cell size can be as small as 4F2. Here F is the technology node, i.e., the minimum feature size. The main drawback of this integration is the sneak path current which is the current that flows through the unselected cells and significantly affects

the read operation. This can be overcome by transistor gating, which is the 1-transistor 1-resistor (1T1R) integration, and the access transistor now controls the cell. Although this structure is robust, the access transistor's size highly impacts the cell area limiting density. The other solution with better cell size is diode gating, where the diode is connected in series to the memristor. The 1D1R structure is more suitable for unipolar memristors. However, for bipolar memristors, where the opposite polarity voltage is needed for setting and resetting, bi-directional, or zener diode is required, often referred to in the literature as 1S1R structure. This structure is also currently being investigated for the 3D memristor crossbars for high-density integrations [154].

8.3.4 Reliability of resistive switching

The reliability of the NVM devices is commonly assessed with data retention, endurance and random telegraph noise. Data retention signifies the NVM device's capability to maintain the stored data over time, and in RRAM devices, it is the programmed resistance states. The data retention evaluations are generally carried-out at higher operating conditions to accelerate the involved events[155]. Few statistical modelling studies are available on RRAM retention[156], and improvement techniques such as HfO2/Al2O3 multilayer structures[157] and electrical tuning[158] are now being widely explored. Endurance is another essential metric that denotes the number of program/erase cycles, which can be performed on the NVM device before failure occurs. It is often measured in commercial devices as drive writes per day (DWPD). The RRAM devices are also showing significant endurance improvement in comparing to commercial flash memories[159]. The endurance of RRAM is largely discussed with its stack and conductive filament related mechanism. Chen et al. have investigated endurance degradation in RRAM and discussed the possible failure mechanisms based on the loss of RHRS/RLRS ratio, which is the window margin[160]. There are also some studies available now in modelling RRAM endurance and the techniques to improve it [161–165].

Random telegraph noise (RTN) is one of the concerns of traditional semiconductor technologies, and it is a significant issue with storage devices [166, 167], especially threshold-voltage instability in flash memories [168, 169]. Several statistical studies were conducted to characterize and understand the physical origin of RTN,

primarily based on the carrier trapping and de-trapping events at the defects [170–172]. The RTN is also now more pronounced in RRAM affecting the read current stability. Most of the available studies in modelling the RTN in RRAM are primarily focused on two-level fluctuations and RTN dependence on the resistance level [173–176]. The noise origin is generally explained by electron trapping and de-trapping events and the fluctuations in oxygen vacancy movements close to the filament [175, 177, 178]. Belmonte et al. showed through modelling and experiments with Cu filament based CBRAM that RTN impact is lesser in CBRAM than the OxRAM, which is attributed to the denser packaging of Cu than the oxygen vacancy defects in the conductive filament. [179, 180]

8.3.5 Variability in RRAM

The common issues in RRAM integrations are the device to device (D2D) and cycle to cycle (C2C) variations in switching resistance[181]. The major contributing factor to this non-uniformity is the stochastic nature of the conductive filament, which is also impacted by the fabrication variations. In CBRAM devices, the conductive filament growth has been observed experimentally[182], and several investigations were conducted to analyse the variability during switching[183, 184]. Solutions were explored to suppress this, such as alloying active electrodes[185], nanoindentation to introduce the concentrated electric field[186], adding barrier layers[187], and so on. In anion based devices, the stochasticity in the vacancy generation and recombination also introduces the non-uniform switching resistance[188]. Degraeve et al. has analysed some of the sources of this variability[189]. Several techniques were also proposed to improve the switching uniformity, such as doping [190], embedding nanocrystals in the oxide layer[191], introducing buffer layers[192], and so on.

The above-discussed RTN, variability in switching resistances and sneak path current are the major challenges currently being investigated in implementing RRAM devices for storage and in-memory computing applications. However, this associated stochastic nature makes RRAM notably suitable for security applications such as TRNG, PUF and hash function. Most of the reviews till now discuss the RRAM device level implementations and their performance[193–195]. Here, we also

present insight on the security application requirements and discuss the detailed development in RRAM devices-based security architectures.

8.4 True Random Number Generator

Random Number Generation is the essential building block a security protocol, and certainly contributes to the establishment of trust in evolving IoT systems. It is often required to generate keys for security protocols to set up secure communication channels, for example, in Wireless Sensor Networks (WSN), and for intelligent applications running at the edge [6, 196, 197]. True Random Number Generator (TRNG), which relies on the physical entropy source, can generate these required random numbers. There are a variety of analog and digital circuits that can harness the physical processes, and therefore, natural entropy, for generating randomness. Since some environmental/physical variation usually controls the source of natural entropy, the TRNG is known to be 'rate-limited or 'blocking'. It hinders utilising it directly for the applications and further demands high system resources, which is unavailable in low power IoT devices [198]. This limitation is alleviated by the deterministic algorithm based Pseudo-Random Number Generator (PRNG) to generate the random bitstream. Cryptographically secure PRNG (CSPRNG) requires high entropy input seed meeting its security strength, and this seed determines its internal state making the random number generation unpredictable [199]. The TRNG's role is now limited in generating only the seed, and PRNG subsequently handles the random number generation.

8.4.1 TRNG in cryptographic applications

TRNGs are useful for various general-purpose and cryptographic applications, and here, we focus on its security purpose implementations. Today, most of the TRNGs are only employed in generating high entropy input seed for the deterministic CSPRNG algorithms, as mentioned earlier [199]. The required length of the seed depends on the deterministic algorithm mechanism and the targeted security strength that determines the number of operations needed to break the algorithm. For example, random number generation based on the hash function SHA-224 with a security strength of 192 bits, requires a seed of length 440 bits in order to meet the

entropy requirement [198, 200, 201]. The input seed determines the internal state of the CSPRNG, and once initiated, the algorithm can generate pseudo-random sequences upon requests. It is recommended to regularly reseed the algorithm, primarily to restore the forward and backward secrecy as it otherwise could be compromised [6, 201, 202]. Thus, each seed has a finite period length, and it is determined by the algorithm used and the number of random output sequences generated [201]. Generally, the number of requests made is monitored with a counter, and when it reaches the set value, then reseeding is done. For the SHA-224 mentioned above, the recommended maximum number of requests between reseeding is 248 [201]. The seed once used is not recommended to be used again for reseeding when the generator is used cryptographic application. Hence, a truly independent random sequence is required for reseeding. An example of random number generation implementation for the WSN nodes is shown in Fig.8.5[6]. Here, the received packets bit error is used as the source of randomness and the initial seeding is carried out before deployment. The major reasons for the CSPRNG assembly are first, entropy harvesting often involves high energy consumption and second, it has poor throughput due to several long processing steps such as harvesting and conditioning.

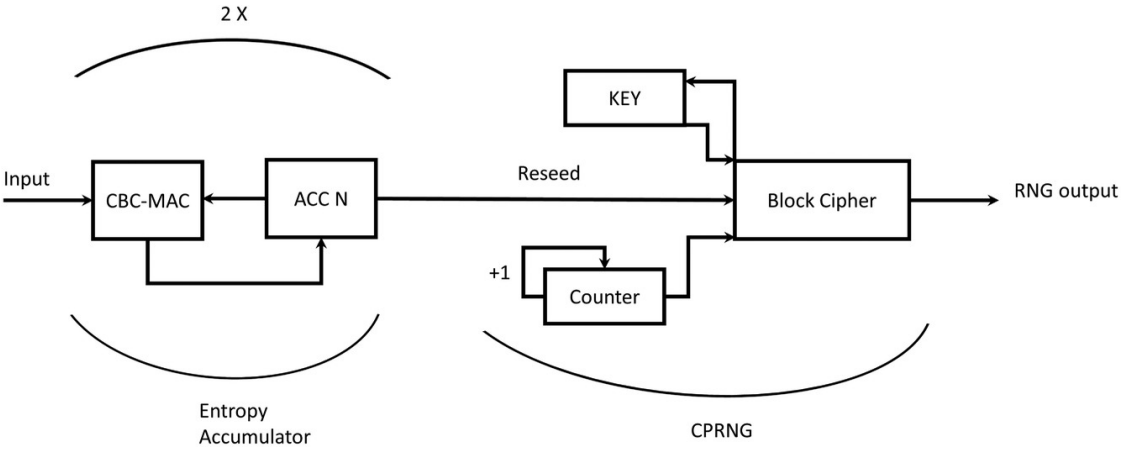


FIGURE 8.5: Block diagram of TinyRNG for wireless sensor nodes. Reproduced with permission.[6]

The random numbers are used for cryptographic key derivations, such as symmetric key and generating asymmetric (public/private) key pairs. The symmetric keys are utilised in data encryption with the popular Advanced Encryption Standard (AES) and in Message Authentication Code (MAC), and asymmetric keys are widely used for digital signature schemes. National Institute of Standards and Technology (NIST) has also provided guidelines for cryptographic key generation[203].

Cryptographic algorithms like AES also demand considerable system resource, and hence, there is growing interest now in the development of the lightweight cryptographic algorithms. High throughput and energy-efficient TRNGs are required in most security applications other than low power IoT systems like in vehicular systems shown in the Fig.8.6[7]. Today, many commercial microcontrollers used for resource-constrained applications, such as PSoC 64Bx[204] and STM32F412[205], already support TRNG and can be deployed directly for applications.

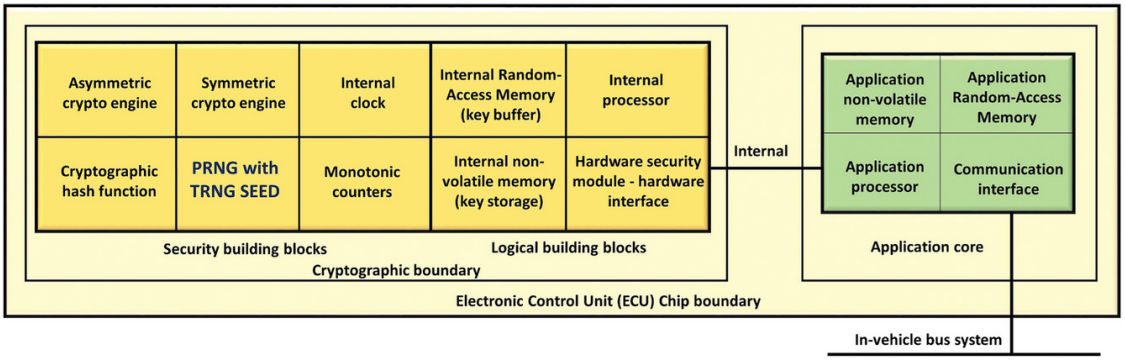


FIGURE 8.6: Overall vehicular full security hardware module architecture. Adapted with permission.[7]

8.4.2 Performance benchmarking

NIST and Bundesamt für Sicherheit in der Informationstechnik (BSI) have provided clear guidelines for designing the random number generators[201, 202]. Designers can use these guidelines to design and validate their generator for cryptographic application usage from the statistical front [206]. The revised NIST test suite includes 15 statistical tests to study the entropy in the generated stream of bits. It is recommended to run all the tests on the bitstream to identify if there are any local non-randomness. Each test computes a P-value that implies the associated closeness of randomness towards the null hypothesis, and ideally, it should be equal to 1. A significance level, often a value in the range between 0.001 and 0.01, is chosen as the minimum acceptance mark, and any P-value below this is considered unacceptable. NIST has further provided the entropy source model shown in Fig.8.7, which includes noise source to derive the randomness, optional conditioning components to remove the bias in the derived bits and health tests to ensure the generator continues to work as intended [8]. The addition of more post-processing components worsen the throughput and also increase energy consumption. Hence,

other than statistical analysis, the TRNG designers should also consider benchmarking throughput and energy consumption. When deployed, the TRNGs can be exposed to extreme operating conditions. The quality of the random bit stream generated is still expected to meet the statistical requirement as it otherwise could compromise the strength of the security protocols that rely on them. Therefore, it is also essential to report the robustness of the entropy source at various operating conditions. An analysis with the combination of the criteria mentioned above can signify the economy of the TRNG for system-level integrations. The throughput in several implementations is enhanced by operating devices in parallel [9], which could significantly impact the area and energy budget. Hence, TRNG is characterized by metrics such as throughput per area, energy efficiency over throughput as we report in the following.

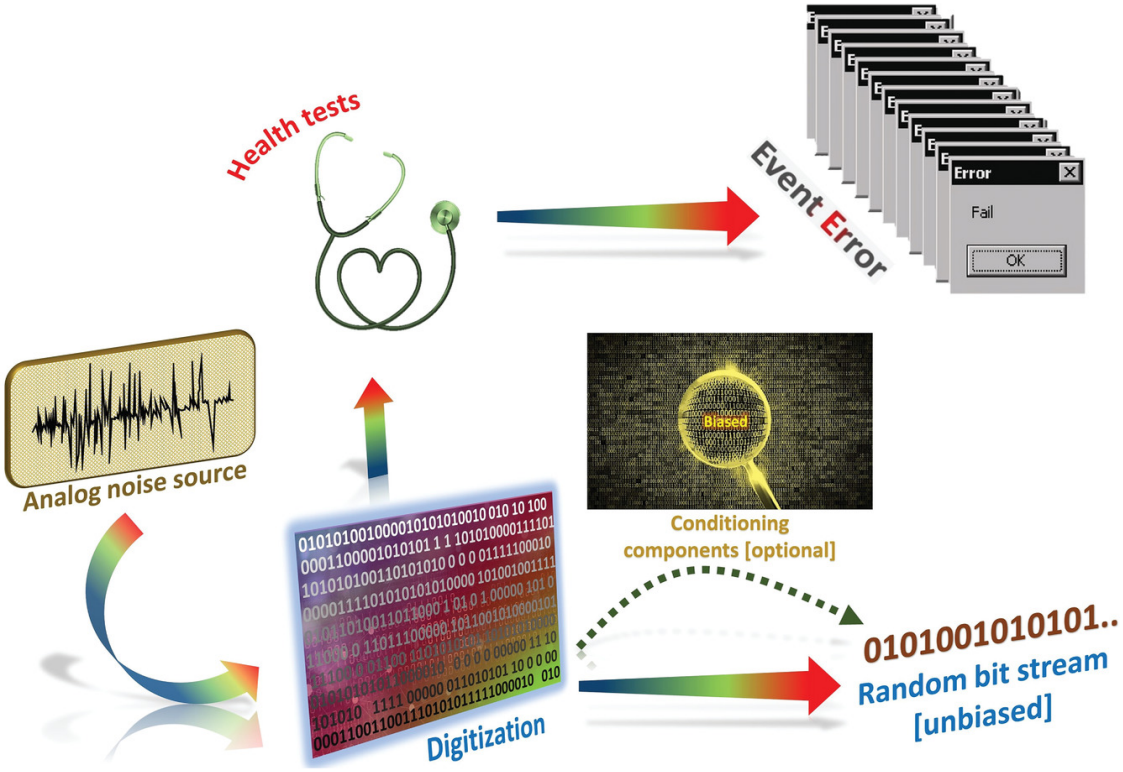


FIGURE 8.7: Entropy source model to design TRNG.[8]

8.4.3 RRAM TRNG implementations

The cycle-to-cycle variations in the switching of RRAM devices and the noise in the read current are the two catetories of entropy source that were commonly explored

for the generation of random numbers, as shown in Fig.8.8. Wei et al [9]. demonstrated a TRNG based on TaOx RRAM in 1T1R configuration utilising 1/f noise. The current fluctuation in 1/f noise and the corresponding power spectral densities for different resistance states were studied. The difference in noise current at the LRS was used to generate the random number as it had wider current difference distribution than HRS. Thus, the random bitstream is generated by comparing the digitized adjacent current values, as shown in Fig.8.9a). The demonstrated TRNG can successfully pass the NIST SP800-22 randomness tests for a temperature range from -40^{o}C to 125^{o}C.

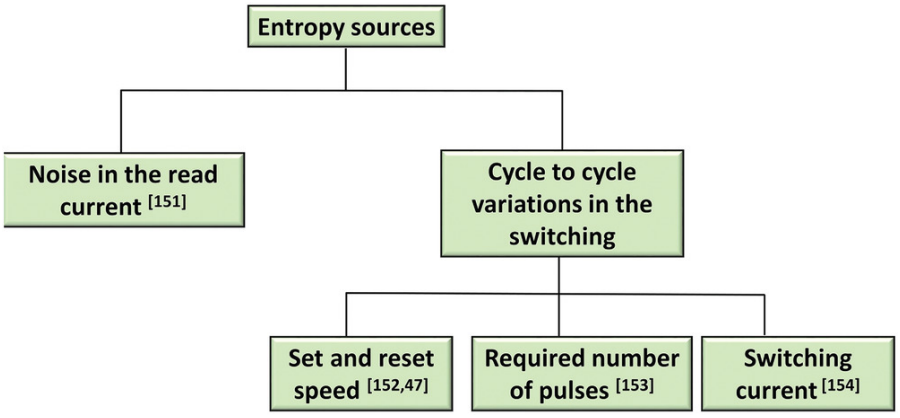


FIGURE 8.8: Harvested entropy sources in RRAM for random number generation.

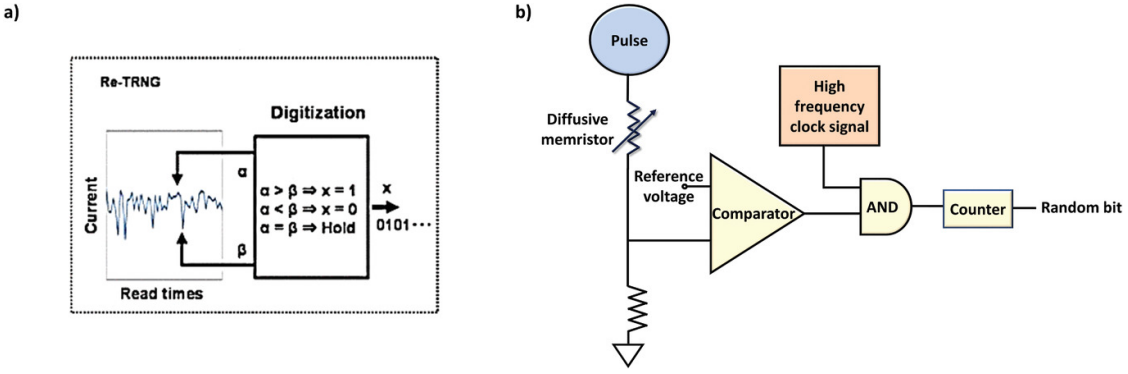


FIGURE 8.9: a) Random bitstream generation by comparing adjacent noise current values at LRS. Reproduced with permission.[9] b) Standalone volatile RRAM TRNG based on switching delay time.

The SET and RESET speed variation caused due to the fluctuation in trapping and de-trapping of oxygen vacancies was explored to generate the random bitstream. Yang et al. [207] exploited the cycle to cycle variation in reset speed of AlOx/WOx bilayer stack RRAM and demonstrated the TRNG with the throughput of 8 Mb/s.

The reset speed variation is calculated by clock cycles with the 20-bit counter, and its value after the reset endpoint is captured and truncated to output the random bit stream. A standalone TRNG based on Ag: SiO2 diffusive memristor utilising the stochasticity in the switching delay time was also reported 144[47]. One of the claimed merits of this structure is the self-reset process due to the device's volatile nature, which could reduce energy consumption. The circuit complexity of this TRNG is comparatively low, and it relies mainly on the comparator, logic AND gate and counter other than the memristor itself, as shown in Fig.8.9 b). The series resistor voltage increases after the stochastic delay time on the applied voltage pulse to the memristor, which affects the comparator's output pulse width. This variation in pulse width is utilised to generate the random bit by logic AND operation with the high-frequency clock signal that flips the counter's binary state at the rising edge.

Lin et al.[10] presented the HfOx based TRNG by exploiting the cycle to cycle variation in the number of pulses required of the SET and RESET process. The random bit is generated based on the parity of the number of pulses needed for one SET and RESET switching cycle. The one bit counter captures the parity of the pulse number, and the D flip flop generates the random number at the end of the cycle by sampling the final state of the counter, as shown in Fig.8.10. The TRNG passed all the 15 NIST tests with the throughput per cell of 1 Mb/s and claimed better endurance. Recently, the TRNG design utilizing the entire HfOx RRAM memory array to generate the random bit stream was also demonstrated [208]. Here, all the RRAM cells were reset to HRS by individually addressing each cell and then the entire array is activated with the limited input current. Because of this limitation in the input current and the intrinsic device variations, analogue resistance states were randomly distributed across the whole array. The distribution is finally digitized by comparing it with the selected threshold value. The XOR gate based post-processing is also implemented to improve the entropy, and this TRNG passed 12 out of 15 NIST tests.

Table 8.1 presents a comparison among the state-of-the-art RRAM based TRNGs. Most of the work done on RRAM TRNG is based on standalone and 1T1R integration. The fluctuations in noise current, variations in speed, pulse number, and delay time of switching and analogue resistance variations by limited input current were experimentally demonstrated as an entropy source to generate random

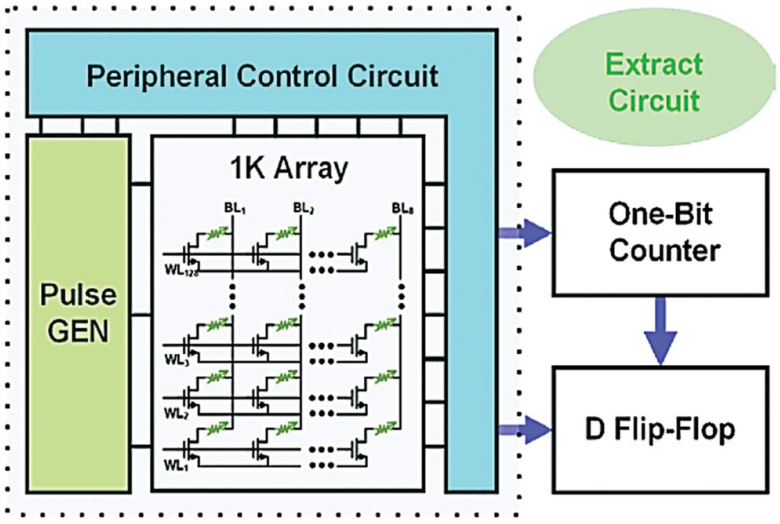


FIGURE 8.10: RRAM TRNG exploiting cycle to cycle pulse number variations.[10]

bits. Few TRNGs were also able to generate high entropy random bitstream without conditioning circuits. The bit rate of RRAM TRNGs can be further viewed in terms of the harvested entropy source, as shown in Fig.8.11, and it is already comparable to other matured technologies.

TABLE 8.1: Comparison of Demonstrated RRAM TRNGs

Year	2016 [9]	2017 [207]	2017 [109]	2019 [10]	2020 [208]
Technology [nm]	40	180	-	-	130
Entropy source	$1/f^{\beta}$ noise	C2C reset speed variation	RRAM switching delay time	C2C pulse number variation	RRAM switching current
RRAM integration	1T1R	1T1R	Standalone (volatile device)	1T1R	1T1R
Switching material	TaO_x	AlO_x/WO_x bilayer stack	Ag: SiO_2	HfO_x	HfO_x
Bit rate	32 Mb/s	8 Mb/s	6 kb/s	1.1 Mb/s (per cell)	-
Energy Efficiency	0.04 nJ/bit	-	-	3.51 pJ/bit	138 pJ/bit
NIST tests	All	All	All	All	12
Test conditions [°C]	-40 to 125	-	25 to 85	-40 to 125	-
Post process	-	No	No	No	Yes (XOR operation)

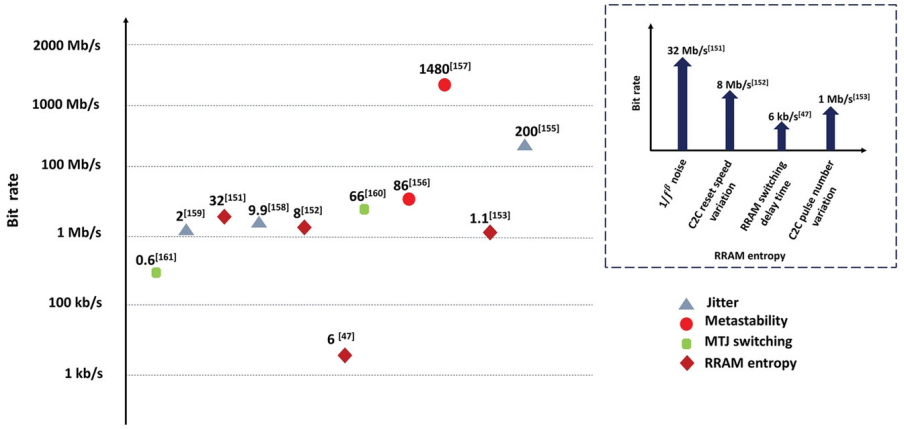


FIGURE 8.11: Demonstrated TRNG bit rate based on harvested entropy source.

8.4.4 Comparison with other technologies

There are multiple practically demonstrated TRNG architectures based on conventional as well as various emerging technologies. The TRNG based on jitter as an entropy source is widely explored because of its feasibility to integrate into digital circuits [11, 209–211]. The basic architecture of such TRNG consists of low and high-frequency oscillators, which can be MOS compatible ring oscillators discussed later in section 4.4 and a flip-flop for sampling, as shown in Fig.8.12a)[11]. The high-frequency oscillator signal is sampled with the low frequency jittered oscillator to generate the random bitstream. There are different schemes proposed for jitter amplification and post-processing to remove bias and produce the truly random bitstream, and they all require custom design circuits [11, 209, 211, 212]. However, it is still challenging to pass all the NIST randomness tests [211, 213]. Another commonly explored TRNG utilising jitter is based on the beat frequency detection (BFD) mechanism. Here, two identical ring oscillators are connected to the D flip-flop. However, due to the influence of manufacturing variations, one oscillator oscillates slightly faster than the other. Hence, after certain intervals capturing events occur that results in the change in the logic state of the flip-flop. The associated random jitter affects the interval every time, making it an entropy source that can be utilised for random bit generation by integrating a counter to the flip flop. The major challenges with this method are selecting the ring oscillators themselves, and the requirement of post-processing as the generated bitstream is not truly random. Several calibration methods based on mathematical modelling and design

Please note that the references in the figure should be identified from the published paper[32] rather than using the numbering from the bibliography in this thesis.

improvements were also explored to overcome these issues recently. [214–216] The TRNG based on metastability of cross-coupled inverters or latches were also explored to generate the random bit stream [217], as shown in Fig.8.12b). However, the major limitation of this method is the device mismatches caused by the fabrication variations resulting in biased output. Hence, calibration and post-processing techniques such as XOR operation, von Neumann corrector are needed to reduce the bias [12, 218, 219]. Other than these discussed implementations, few discrete and analog-time chaos based TRNGs were also explored [220–222]. The primary concern with multiple application scenarios with security as a requirement is tight constraints on the area and energy budget, such as edge computing platforms, IoT platforms and mobile handheld devices. Such devices are battery-driven, and the deployed environment is also highly volatile. These necessitate the TRNG to be equipped with a robust entropy source while offering high throughput and energy efficiency in the smallest form factor. Notable advantages of RRAM in this context are low power consumption and simultaneous usage as a storage application in contrast to the CMOS-based implementations. These advantages facilitate the TRNG design with RRAM to be tailored to meet the system constraints.[223]

Among the emerging NVM technologies, STT and SOT-MRAM and FeFET were also investigated for implementing TRNG. In STT-MRAM, entropy associated with switching probability and switching time were exploited for random number generation [224, 225]. A complex circuit to precisely control the current pulse's amplitude and duration is required in the former while the latter is affected by the temperature variations. The stochasticity in the switching of SOT - MRAM is also currently being explored for TRNG [226–228], and similar to STT-MRAM, the robustness of such devices is still a downside. By applying a voltage pulse for which the probability of switching is 50%, the TRNG was demonstrated with FeFET [229]. However, ferroelectric switching is again influenced by temperature, which affects stability. RRAM based TRNG is more robust than these technologies with the required high entropy, making it suitable for practical implementations. Furthermore, the RRAM TRNG demonstrations show comparable improvement in energy efficiency recently, as seen in Fig.8.13. More investigations are still needed in designing the energy-efficient RRAM random bit generator that can potentially be integrated into low power IoT systems.

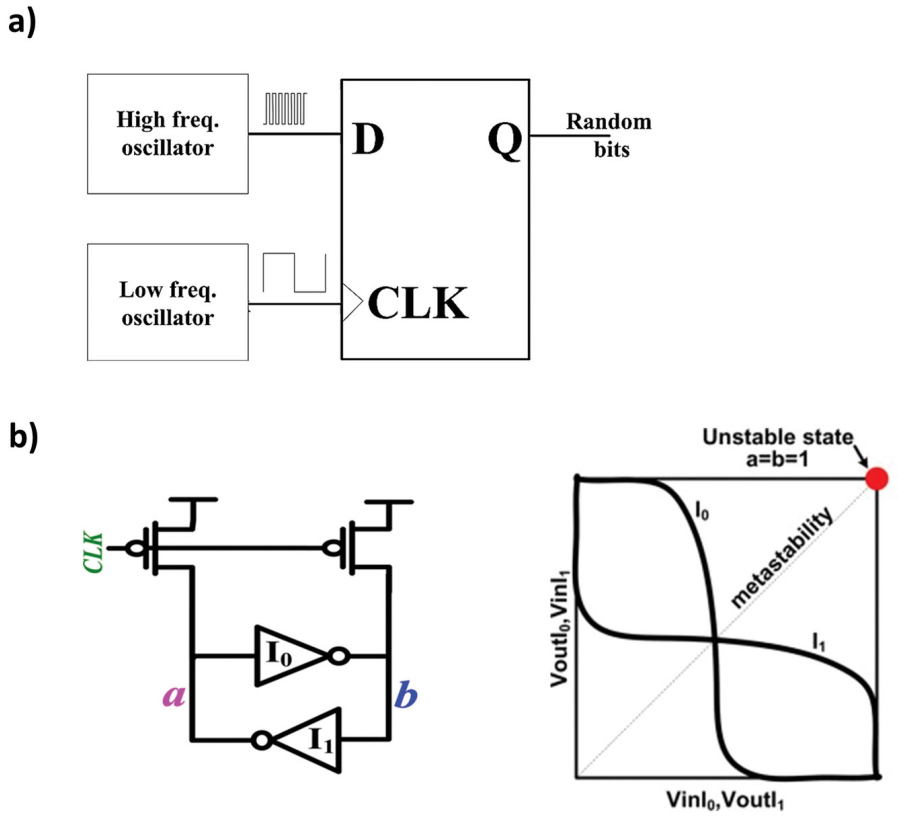


FIGURE 8.12: TRNG based on a) Jitter as the entropy source. Reproduced with permission.[11] b) Metastability of cross-coupled inverters. [12]

8.4.5 Future outlook

The availability of a robust high entropy source is the primary requirement for constructing TRNG. It is not adequately available with conventional MOS technologies where the bitstream generated is biased and require postprocessing. However, these technologies still offer decent throughput, and hence, it is used for seed generation currently. The emerging NVM technologies such as RRAM inherently have high entropy source, and it is comparatively robust and scalable for high-density integrations. The RRAM TRNGs can pass the NIST tests even without any post-processing schemes. Nevertheless, throughput is still the hurdle when it comes to practical applications. As discussed in the earlier sections, the TRNGs are resource hungry as the entropy is harvested each time with the subsequent process for random bit generation. If the TRNGs can overcome these issues, they can potentially be an alternative for CSPRNGs, provided they are secured from attacks. One of the future direction of RRAM TRNG is towards designing high throughput architectures. RRAM integration architectures such as 0T1R, 1TnR, 1D1R and entropy

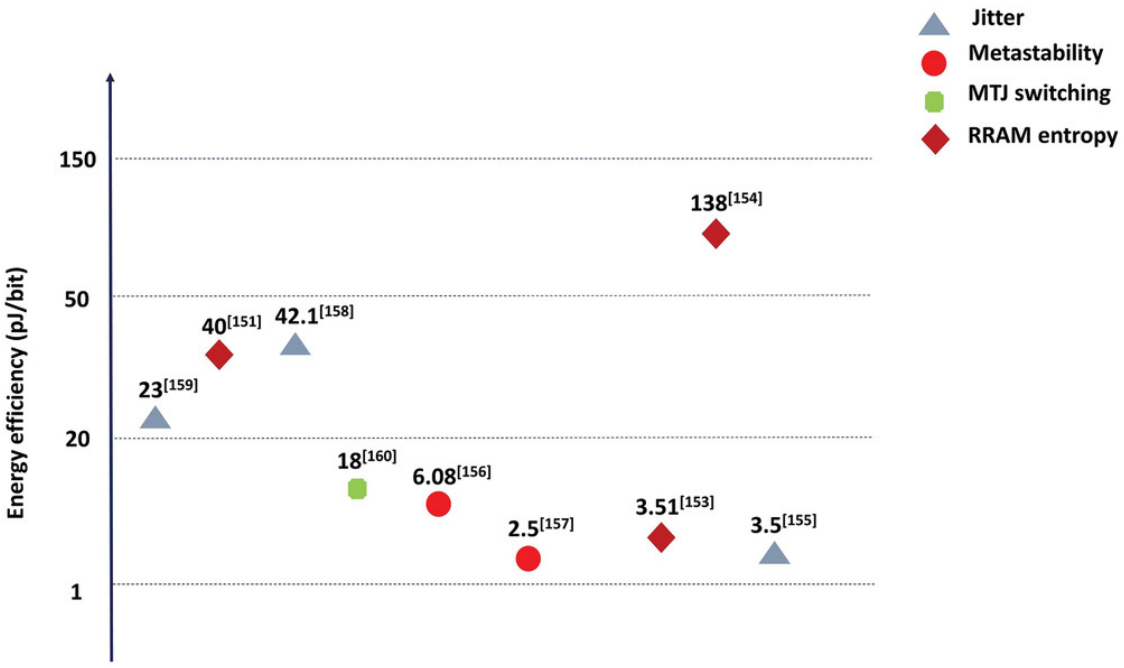


FIGURE 8.13: Demonstrated energy efficiency of TRNG based on different entropy sources.

source associated with it, including sneak path current, still need to be experimentally investigated for random number generation. There is also a growing interest in emerging monolithic 3D integration of RRAM array, which could be explored to generate high throughput and energy-efficient random bitstream. Health tests such as startup, continuous and on-demand tests are needed to be integrated into the RRAM TRNG to monitor its operation, and it is currently unexplored.

8.5 Physical Unclonable Function

Physical Unclonable Function (PUF) is gaining significant attention for hardware security applications in recent years. The concept of PUF was proposed as a Physical Random Function to authenticate Integrated Circuits (ICs) [230]. The basic idea is that a device-specific unique key can be derived each time rather than storing a set of keys in the non-volatile memory. This method is more secure in preventing the adversary from accessing the keys and also lightens the expense of creating a safe storage space [231]. The working principle of PUF is that to

Please note that the references in the figure should be identified from the published paper[32] rather than using the numbering from the bibliography in this thesis.

compute the output response, which is the key, for the applied input challenge, and it is referred to as the challenge-response pairs (CRPs), as shown in Fig.8.14. It is often explained with the black-box model where the challenge vector from the input space is mapped to the response vector in the output space, and the mapping function here is completely unknown, one-way and unique to the device. The function is associated with the device's intrinsic properties, and the fabrication variation can also induce such behaviour. For example, in RRAM devices, as mentioned before, the stochastic nature of the conductive filament, which is also influenced by fabrication variations, can result in different switching resistance value to each device for the same programming pulse. These device variations can be utilised in generating the key.

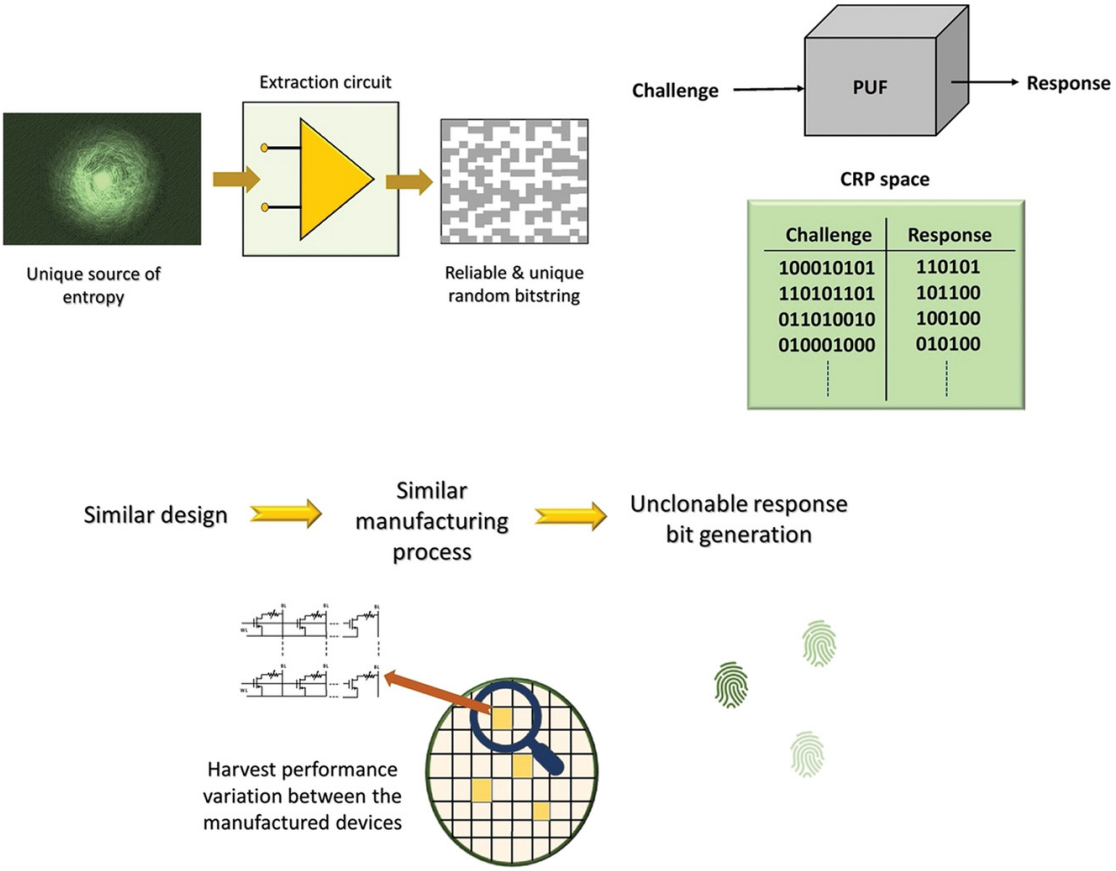


FIGURE 8.14: Working principle—the PUF computes the response for the given input challenge. The response is unique to the device, similar to the fingerprint associated with the variation in the device's performance characteristics.

The application of PUF is based on the available number of CRPs. We can classify the PUF into two types based on the CRP size - weak and strong. The number of CRPs supported by weak PUF is minimal, or one can loosely say that it is linearly

dependent on the number of components subjected to the intrinsic variation. On the other hand, strong PUF has a larger CRP space, which is ideally exponential to the size of challenge bits [232, 233]. Hence, weak PUF must be secured, and access restricted, which is not the case with strong PUF. In most circumstances, weak PUFs are used as a substitute for storing the keys in NVM to restrict extraction, and the derived key can be used for device identifications and some cryptographic applications. Strong PUF is more versatile due to the larger CRP space and can be used for secure authentication. The CRP is computed each time with a new challenge in strong PUF, and generally, it is not reused.

8.5.1 PUF in cryptographic applications

PUF is explored for various security applications, of which, we discuss some of the common implementations of strong and weak PUF. One of the applications of strong PUF is authentication. The basic scheme is as follows. The verifier collects the considerable or whole CRP space of the PUF and stores it in the secured database with the entity's identity. This is referred to as the enrolment phase. During authentication, the entity sends its identity to the verifier. If the identity matches the one in the database, the verifier chooses a random challenge from the previously computed CRP space and sends it to the entity. The entity computes the response with the PUF and presents it to the verifier. If the response matches the stored one, the authentication is successful, and the session is established. The used challenge is then removed from the database to avoid reply attacks. The major difficulties of this basic scheme are that the verifier needs a more extensive database to store many CRPs and method for mutual authentication. These problems are even pronounced in the IoT environment [234]. To overcome this issue, several solutions are being explored [13, 235–237] currently, such as the one proposed for IoT systems shown in Fig.8.15 [13]. Instead of storing many CRPs, a single response is computed before deployment here. During authentication, the server uses this response to encrypt the message for verification and present it with a new challenge to the deployed device. The device verifies the message using the response computed from the PUF, computes the new response to the given challenge. This generated response is then sent to the server as an encrypted message using the previously known response. The server verifies and stores the new response for

future authentication, and the session is established. This method reduces the complexity of storing large CRPs and also facilitates mutual authentication.

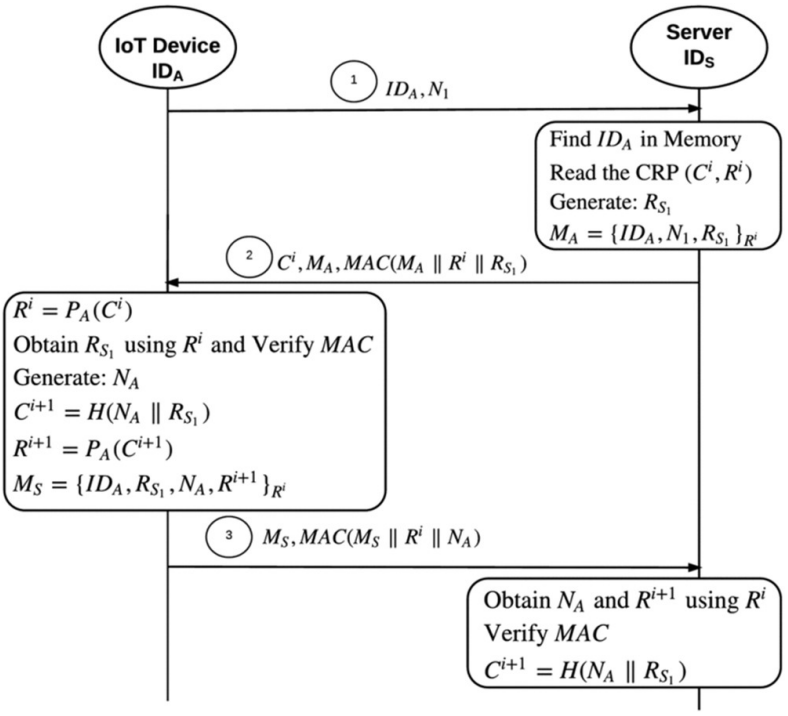


FIGURE 8.15: Authentication based on PUF for IoT system.[13]

The weak PUF is also useful for practical implementations, especially in preventing counterfeiting, tampering, and ensuring device integrity [238–240]. Islam et al. have proposed an IC traceability solution with PUF and blockchain technology [239]. Recently privacy-preserving mutual authentication (PPMA) scheme shown in Fig.8.16 is also gaining attention [14, 241, 242]. This implementation requires both TRNG and PUF and hence, often explored as a unified design. Here, the server generates a random string R1 through its TRNG and encrypts it with the secret PUF secret ID and transmits it to the device. The IoT device decrypts the message with its PUF and adds the entropy string R2 generated by its TRNG to the string received. Then again, it encrypts similarly and sends the messages to the server. The server decrypts and derives the original string from the encrypted response. If the derived string matches with the original string R1, then the authentication is successful. The advantage of this scheme is that a CRP can be used multiple times.

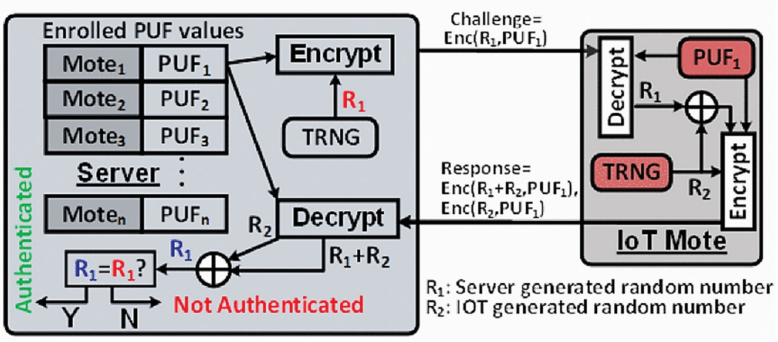


FIGURE 8.16: Privacy-preserving mutual authentication (PPMA) scheme. Reproduced with permission.[14]

8.5.2 Performance benchmarking

The PUF is expected to meet specific requirements defined as uniformity, bit-aliasing, uniqueness and reliability to be deployed for applications. These metrics can be utilized to analyse the quality of the PUF statistically.

Uniformity: The PUF should generate an equal number of 0's and 1's in its responses.

$$Uniformity = \frac{1}{n}\sum_{i=1}^{n}(R_i) \times 100 \tag{8.1}$$

Here, R_i is the bit at the index "i" of the n-bit response. The expected value of mean uniformity is 50% for truly random responses, and any value less and greater than 50% shows that the generated responses are biased to 0 and 1, respectively.

Bit-aliasing: It denotes the inclination of different PUFs to produce a similar response bit or, in other words, whether the particular response bit generated by PUFs is stuck to either '0' or '1', making it more predictable for the applied challenge and the expected value is 50%.

$$\text{Bit} - \text{aliasing}(\text{i}^{\text{th}}\text{bit}) = \frac{1}{k}\sum_{j=1}^{k}(R_j) \times 100 \tag{8.2}$$

Here, R_j is the i_{th} bit response of the PUF device "j" of "k" devices.

Uniqueness: The PUF should generate a highly independent response compared to other PUFs for the same applied challenge. The ideal value of uniqueness is 50%.

$$\text{Uniqueness } = \frac{2}{k(k-1)} \sum_{j=1}^{k-1} \sum_{l=j+1}^{k} \frac{HD(R_i, R_j)}{n} \times 100 \tag{8.3}$$

Here, "k" is the total number of PUFs, "j" and "l" refer to two different PUFs with responses R_i and R_j respectively, "n" is the response bit length and HD(.) is the hamming distance which denotes the number of bit positions that differs between two binary strings.

Reliability: It is one of the essential criteria of PUF for practical implementations. It denotes the capability of PUF to reproduce the response at different trials for the same applied challenge. Generally, the reliability is calculated at different operating conditions and the ideal value is 100%.

$$\text{Reliability} = 1 - \frac{1}{p} \frac{\sum_{t=1}^{p} HD(R_i, R_i^t)}{n} \times 100 \tag{8.4}$$

Here, "p" refers to the number of trials or total number of collected response samples, R_i is the reference response at ideal operating condition, R_i^t is the response at the trial "t", "n" is the response bit length and HD(.) is the hamming distance. The reliability of PUF is an essential metric in deciding the error correction schemes. PUF's reliability is affected by various factors such as environment, characteristic of entropy source, variations in voltage and so on. It is a best practice to analyse the Bit Error Rate, BER = 1 - reliability, at different operation conditions to benchmark the performance of the PUF. The worst-case BER indicates the tolerance of PUF at extreme conditions. The simplest approach to decrease the BER is to mask the unstable bits, often referred to as dark bits, at the expense of shortened response length [243, 244]. The majority voting technique can also be used where the same challenge is applied to the PUF repeatedly, and the majority in the response is finally selected [245]. Ganji et al. [246] has discussed computing the minimum number of repetitions needed in majority voting to improve the PUF's reliability. Other than these approaches, helper data generation and error correction code (ECC) is widely implemented, and there are several studies available on it [247, 248]. Biased PUF is also vulnerable to attacks [249]. Therefore, XOR operation [250], Von Neumann debiasing [251], and other related techniques [252, 253] are available utilized to remove the bias in PUF response. The implementation of any of these post-processing techniques increases the system overhead. Hence,

PUF designers should need to target very low native and worst-case BER with better uniformity to minimise the requirement of these techniques. Some designers also report NIST test suite results on their PUF's response bitstream as it covers the necessary randomness behaviour.

8.5.3 RRAM PUF implementations

Among the emerging devices based PUF, RRAM based PUF has attained significant attention due to the lower BER and high-density integration. The RRAM devices' HRS variations are generally used as the entropy source in 1T1R integration as they have a wider distribution range than LRS [254]. After the forming process, a reset pulse is applied to all the devices in the array. It results in the analogue distribution of resistance states due to the inherent device to device variation. The response bit is generated by comparing the RRAM cell read current in two ways now, as shown in Fig.8.17. One way is to compare with the reference cell current or constant current source[255], and the other is to compare with another RRAM cell read current. The latter is usually referred to as the differential mode[15]. The split resistance technique is also implemented to improve the sensing window and overcome reliability degradation in both operations [254, 256]. Yang et al. also extended the RRAM TRNG based on the reset speed variations discussed in section 3 for PUF implementation. The digitized response bitstream is written back to two symmetrical cell arrays as inverted and non-inverted bits, and the cell addresses are used as the challenge. The advantage of implementing the symmetrical cell array structure here is to avoid side-channel attacks based on power analysis. [257]

One of the primary concerns of the 0T1R structure for memory application is the sneak path current. Though several solutions [258] have been proposed to supress the influence of sneak path current, it is still useful for PUF implementation. The challenge vector uniquely determines the rows and columns to be activated, and inactivated lines are left floating as shown in Fig.8.18[16]. The resultant current through the activated devices and the sneak path current determines the response bit. The Al2O3/TiO2-x RRAM passive crossbar array PUF with native BER of 0.7% was experimentally demonstrated [259]. The devices' conductances were tuned once for the PUF instance to improve the uniformity and noise margin.

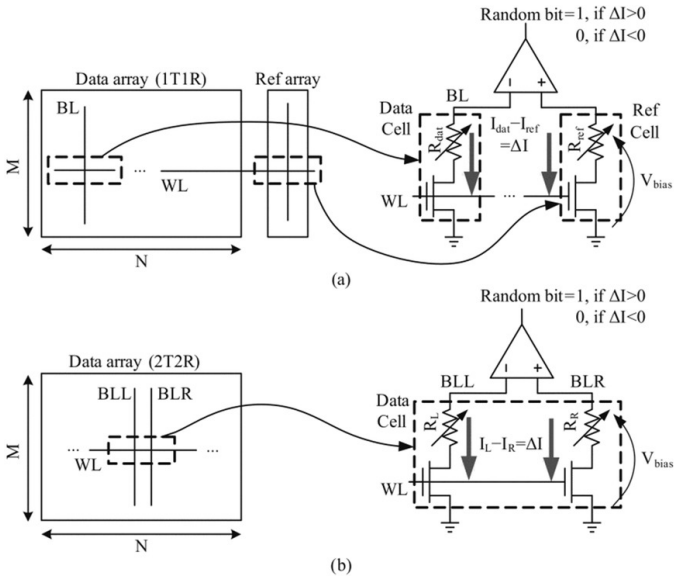


FIGURE 8.17: RRAM read cell current compared with (a) common reference cell current (left) and (b) another selected RRAM cell (right). [15]

An XOR operation is adapted further in the design to minimise the bias in the output vector. The passive crossbar PUF implementations were also explored with unformed devices. The conductance of the unformed devices is much lower than that of the formed ones. Hence, operating the devices in the unformed state reduces power consumption and peripheral overhead. The downside of this approach is the high native and worst-case BER as the signal is already in the noise range [260]. Mahmoodi et al. implemented the key booking technique to overcome this issue. The golden key at different temperatures is stored in the server during the enrollment process and later used during the authentication [261]. All these discussed PUF demonstrations could be classified as strong PUF as the number of CRPs generated is exponentially in the challenge bits' size. One of the key benefits of this RRAM technology is the 3D stacking of the array. Nili et al. illustrated the RRAM PUF with two-level monolithic 3D integrated 20 $\times$10 effective crossbar arrays. The structural improvement grants high integration density and enhances throughput as multiple bits can be generated in parallel [262]. Yang et al. also demonstrated a 3D vertical RRAM PUF with an 8 $\times$32 crossbar array. The elevated entropy in the vertical integration due to the IR drop and the sneak path current is exploited to generate a highly random response bit that can subdue the modelling attacks. They also proposed a scheme to detect the unstable bits and stabilise them by reprogramming to improve the BER.[263]

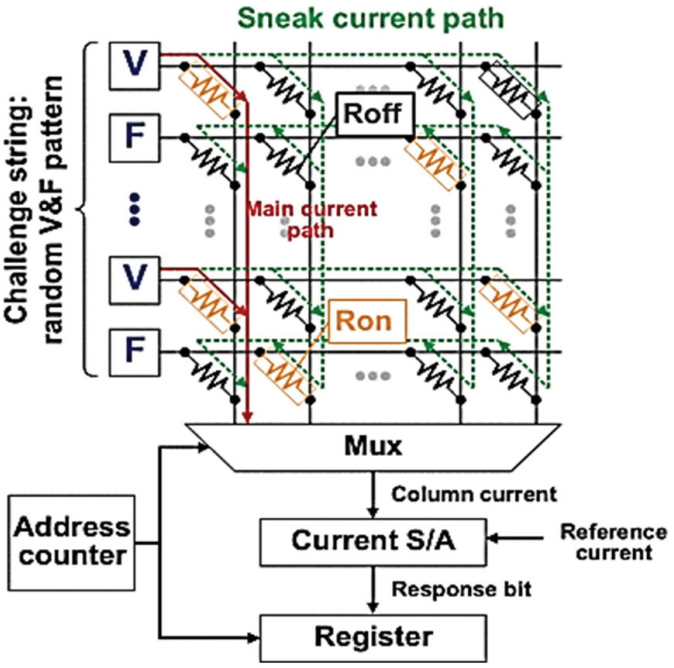


FIGURE 8.18: Passive crossbar RRAM PUF exploiting sneak path current to generate response bit. Reproduced with permission.[16]

Recently, a variant of RRAM PUF with reconfigurability is proposed. The reconfigurable mechanism completely overwrites the existing CRP space of the PUF. Lin et al. demonstrated reconfigurable PUF with the lowest native BER of around 6 × 10-6 for a 128-bit response bit length [255]. The cycle-to-cycle variation in device resistance is exploited for reconfigurability. The CRP space is cleared first by setting all the devices to LRS. Then by applying the RESET pulse, the device resistances were again stochastically distributed over the entire array. The response bit can now be generated by the resistance comparison scheme discussed before [255, 264]. A reconfigurable design between storage and PUF functions was also explored based on the variations of switching voltages across the RRAM devices [265]. The intermediate switching voltage, where the probability of switching is 50%, is applied during SET operation. The standard addressing peripheral circuit can then be used to read out the distributed binary resistance states. Zhao et al. [266] also proposed a scheme that can achieve similar reconfigurable behaviour. Instead of using the RRAM device characteristic as the entropy source, the random bitstream generated by the RO TRNG is utilised in setting each RRAM cell state to either HRS or LRS. Flexible electronics are finding several applications these days, and one such flexible structure PUF design based on Halide perovskite

memristor was demonstrated recently. [267] The strategy is similar to the discussions before, where after forming/set process, the stochasticity in the resistance distribution by the reset process is utilised for response generation due to its wider probability distribution. A 32 × 32 dot-point devices are used for the illustration, which is interpreted as the crossbar array logically. The demonstrated structure can be configured as weak and strong PUF and further reconfigurable utilising the cycle-to-cycle variations. A recurrent scheme, the response is XORed again with the input challenge and iteratively applied to the same PUF as the intermediate challenge generating the final response, is implemented to circumvent the modelling attacks.

TABLE 8.2: Comparison of Demonstrated RRAM PUFs

Year	2018 [259]	2019 [264]	2019 [260]	2020 [265]	2020 [261]	2020 [266]	2020 [263]	2021 [255]
Technology [nm]	250	130	250	130	250	130 and 65	-	130
RRAM integration	0T1R	1T1R	0T1R	1T1R	0T1R	1T1R	3D VRRAM	1T1R
Bit cell area	≈4×4 F^2	≈169F	$4F^2$	$108F^2$	$4F^2$	$108F^2$	≈$1F^2$ (8-layer stacking)	≈$169F^2$ / ≈$108F^2$
Entropy source	D2D variations, sneak path current	D2D variations, write C2C variations	Unformed RRAM D2D variations	RRAM switching voltage	D2D variations, Sneak path current	Dynamic Jitter noise	D2D variations, sneak path current	D2D variations, write C2C variation
Switching material	Al_2O_3 / TiO_{2-x}	HfO_x	Al_2O_3 / TiO_{2-x}	-	Al_2O_3/ TiO_{2-x}	-	HfO_x / TiO_x	HfO_x
Energy efficiency	41–213 fJ b^{-1} (0.1–0.3 V)	3.028 pJ b^{-1}	4 fJ b^{-1}	-	-	-	-	3.028 pJ b^{-1}/ 2.404 pJ b^{-1}
Native BER (RT)	0.7%	≈0%	≈4%	0.03%	-	0.03%	≈2%	≈0%
Worst case BER (without post processing)	4.2% (85^oC)	≈0% (till 125^oC)	11.5% (85^oC)	≈0.13% (150^oC)	1.4%/ 0.7% (85^oC)	0.13% (150^oC)	5.10% (85^oC)	≈0% (till 2^oC)
Reconfigurability demonstration	No	Yes	No	Yes	No	yes	No	Yes

Table 8.2 compares the state-of-the-art experimentally demonstrated RRAM PUF. 0T1R and 1T1R integrations were popularly investigated and demonstrated to have larger CRP space classifying under strong PUF. In addition to the inherent device to device variations and associated I-V non-linearities, sneak path current was also utilized as an entropy source in passive crossbar structures. One of the notable advantages of RRAM PUF over other PUFs is its record low native and worst-case BER. This significantly lowers the requirement of post-processing which in turn reduces the system overhead. The unformed RRAM devices with a 4F2 bit cell area were also explored, and it is reported to have a better energy efficiency of 4 fJ/bit. However, its BER is comparatively higher than that of other structures. The cycle-to-cycle variation in device resistance states and RO TRNGs' random binary stream-based switching mechanism are utilized to demonstrate the RRAM PUF's reconfigurability. The performance can also be interpreted in terms of the type of integration, as shown in Fig.8.19. The 0T1R integrations offer better bit cell area and energy efficiency, and on the other hand, 1T1R integrations generally demonstrated to have low BER. It should be noted that PUF cell area equivalent to 1 F2 has been demonstrated with 3D VRRAM integration.[263]

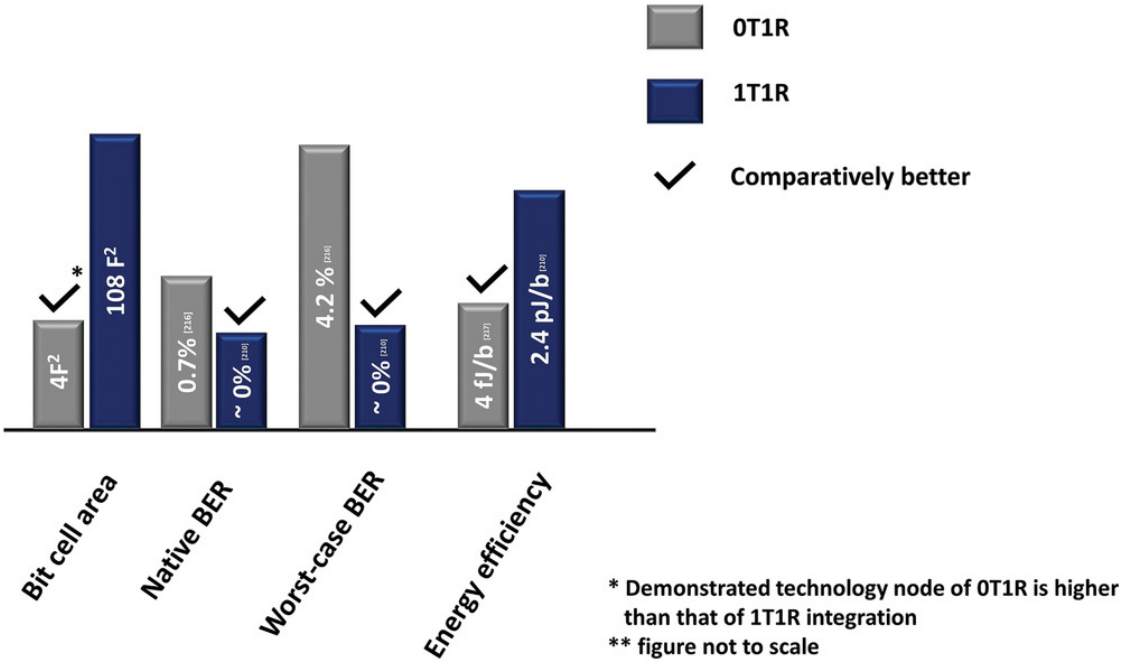


FIGURE 8.19: Comparison of RRAM PUF based on the integration.

8.5.4 Comparison with other technologies

Most of the investigated standard MOS technology PUF architectures are based on either propagation delay mechanism, such as ring oscillator (RO) and arbiter PUF or intrinsic variations in the embedded memory device such as SRAM PUF. The RO based PUFs are widely explored, especially with FPGA implementations [17, 268, 269]. A basic RO consists of an odd number of chain of inverters, with the output signal fed back to the input resulting in oscillation. The frequency of oscillation changes slightly between different ROs influenced by the fabrication variations, and this property can be exploited for constructing PUF. The simplest method is to use the challenge bit to select the RO circuits, and the response bit is generated based on the frequency comparison of the chosen circuits as shown in Fig.8.20a). The CRP space of RO PUF is limited, and hence, it is classified under weak PUF, and its performance is also susceptible to operating conditions [186]. There are different proposed RO PUF architectures and post-processing methods to increase the CRP space, uniqueness, reliability and reconfigurability [270–274]. However, the working principle of RO limits the scaling, and its performance is also susceptible to operating conditions.

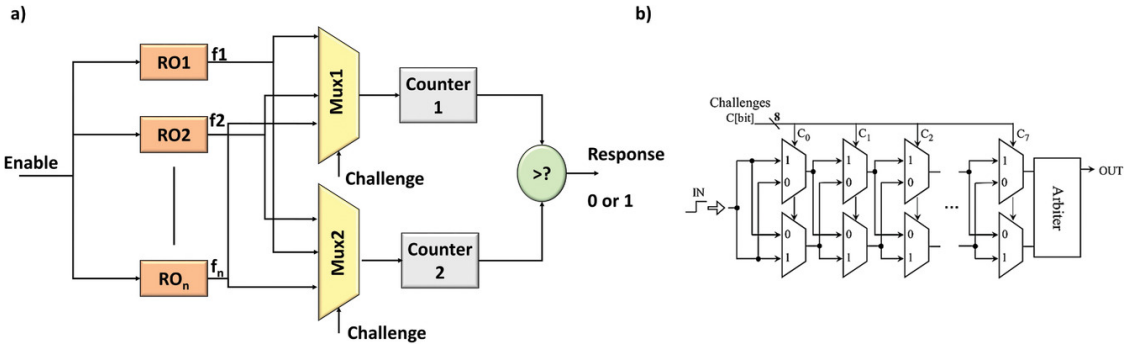


FIGURE 8.20: a) Ring oscillator PUF. Adapted with permission.[17] b) Schematic of arbiter PUF. Reproduced with permission. [18]

Another common PUF based on the delay mechanism is the arbiter PUF. The strategy of response bit generation in this implementation is based on the race between the signals. One of the basic construction of arbiter PUF is that the input pulse is applied to the common end of the first stage, and the challenge bits are given to the select lines of the multiplexors that determine the path in which the signal propagates, as shown in the Fig.8.20b). Ideally, both the signals should reach the other end simultaneously as the layout is the same. However, due to the fabrication variations and the associated changes in the gate delays, one signal

reaches before the other. The arbiter at the end, usually the D flip-flop, generates the response bit based on the signal reaches first [275]. The CRP space of the arbiter PUF is larger, and therefore, it is classified under strong PUF. To overcome the modelling attacks and to improve uniqueness, different approaches were also explored such as response bits XORing[276], challenge bit transformation[277] and so on[278–281]. All these approaches impact the system overhead. It should also be noted that here the n bit challenge is mapped to a single-bit response, and hence, multiple such assemblies are needed to generate the multi-bit response penalising the area.

The memory element based PUF is often preferred as it lessens the addition of dedicated hardware. The Static Random Access Memory (SRAM) consists of cross-coupled inverters, which holds the written logic as long as it is powered up. During initial power-up, the SRAM cell has to be at the metastable state ideally. However, due to the threshold voltage mismatch between the cross-coupled inverters induced by the fabrication variations, the cell state moves to either logic 0 or 1, amplified by the positive feedback loop, and it is utilised for key generation. In this case, the response generated is limited to the available cells, and hence, it is placed under weak PUF. Again, different schemes such as aging injection[282], eliminating unstable bits with calibration[283] were proposed for reliability and uniformity improvement and several others[284, 285] to resist modelling attacks. The variant of the concept mentioned above is based on cross-coupled latches. This is utilised for PUF implementation in FPGA, referred to as butterfly PUF, where the initial SRAM memory cell is forced to the known state. The major challenges of all the discussed above MOS architectures are the necessity of post-processing steps due to the high BER, scalability, size of the CRP space and presenting resistance to modelling attacks. The CRP space demonstrated with RRAM is comparable to these matured CMOS technologies, as seen in Fig.8.21, and hence it can be potentially more robust and can easily meet the resource constraints.

Among the emerging NVM technologies, STT and SOT-MRAM are also widely being explored for PUF implementations, and both field-assisted and field-free stochastic switching mechanisms were demonstrated for response bits generation[286–289]. There are few proposals based on quantum-dot cellular automata (QCA), and however, it still needs experimental investigations [290, 291]. The RRAM has demonstrated higher reliability and lower BER for a wider operating range than

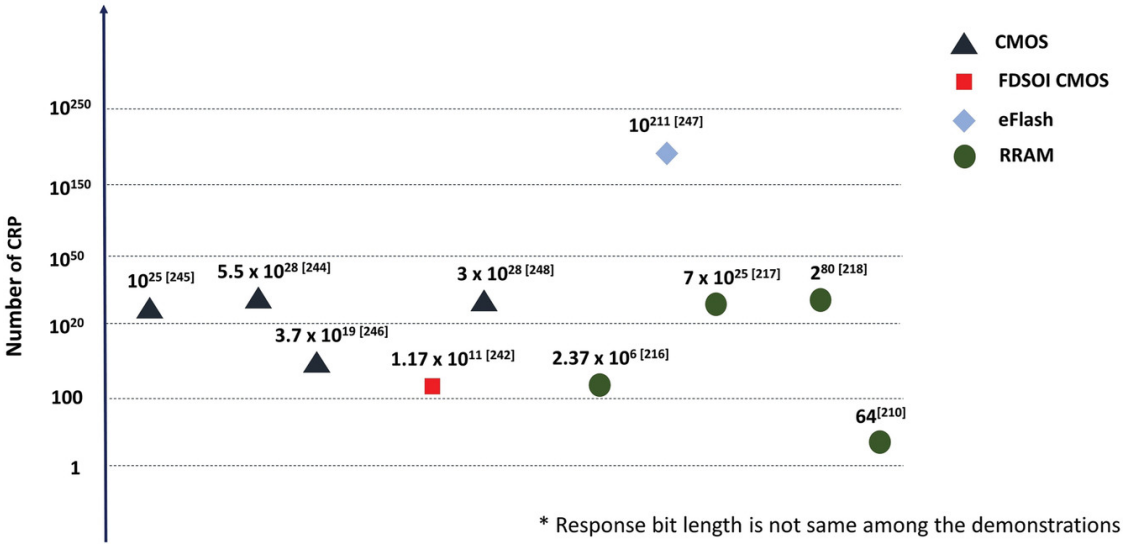


FIGURE 8.21: Demonstrated CRP space of the PUF using different device technologies.

all these technologies, significantly reducing the necessity of the additional error correction schemes, as mentioned before. The RRAM PUF architectures are also demonstrated with larger CRP space classified under strong PUF. The larger resistance distribution window makes RRAM technology more robust for practical implementations, and the passive crossbar structure further comparatively offers high-density integration.

8.5.5 Future outlook

The chip makers are looking for a robust method of secured key storage for hardware obfuscation to prevent IC piracy and adversary from reverse-engineering the chip design and expose them to vulnerability. The method of deriving keys each time rather than storing them in NVM make PUF ideal for these applications. Many low power embedded devices are deployed for various purposes in the IoT environment, and they are handling critical data, including healthcare and finance. It necessitates secure ways of authentication and data transfer with resilience to attacks. Strong PUF with larger CRP space can be used as a security primitive to construct the lightweight protocols that could run economically in these resource-constrained devices while meeting the demanded requirements. RRAM

Please note that the references in the figure should be identified from the published paper[32] rather than using the numbering from the bibliography in this thesis.

PUF is demonstrated to have a larger CRP space in the smallest form factor and inherently has unique characteristics for unclonable, which is the hurdle with conventional MOS technologies as most of them are susceptible to modelling attacks. Hence, RRAM PUF is expected to be integrated into most of the devices for all these applications. The investigations till now are predominantly proof of concept studies. More research focuses on how these devices can be integrated into resource-constrained systems, and with security protocols are needed. Several reported RRAM PUF circuits are capable of thwarting the passive side-channel attacks based on power analysis. The information leakage in unformed RRAM based PUF is notably minimal despite relatively high BER as the signals are comparable to the noise level. [260] The careful tailoring of unselected electrodes in the passive crossbar array for the sneak path current can further improve the complexity. [259, 261] The utilisation of two symmetrical array structures for generating the PUF response by writing back the inverted response bit in one of the arrays can complicate the power analysis, especially for active crossbar structures. [257] However, investigations are needed on RRAM PUF designs to overcome the active side-channel attacks where the adversary could strike to overwrite the resistance states of the PUF device. The concept of reconfigurability is unique in these NVM devices. The reconfigurability in RRAM PUF has been demonstrated in recent years, and however, the new CRP space's secure enrollment process still needs to be thoroughly investigated. There are few studies in utilising the same array of RRAM devices for both PUF and regular memory functions. Detailed studies, including security issues associated with switching between PUF and memory function, possible robust entropy sources, readout schemes, and endurance, are still needed for realising it.

8.6 Hash Function

Hash functions are another important cryptographic function commonly used in message integrity check, digital signatures, error detection and so on, and today, many security standards are based on it. We can define it as a one-way mathematical function that maps arbitrary-sized input data to a fixed-sized bitstring. Hence, given the functions' output, it must be computationally infeasible for the adversary to find the original message, often referred to as preimage resistance. The

strength of the hash function is further denoted based on its collision resistance. It indicates that it should be infeasible to find a second available input that maps the same output. Though, the output should remain the same for a particular input maintaining determinism. One of the common approaches to designing a hash function is from the random oracle model [292–294]. It is based on the theoretical belief that an oracle can provide a truly random output for every given input with the above-discussed deterministic and collision resistance properties. The traditional methodology adopted by most secure hash function algorithms (MD5, SHA-1) builds on compression function based on Merkle–Damgård construction. The input message of arbitrary size is divided into blocks of equal length, and the compression function iteratively processes it. The padding scheme is usually adapted to make the input message into multiples of fixed size. The compression function takes two inputs at each iteration, one from the input message block and the other is the output from the previous iteration, as shown in Fig.8.22. It is also recommended to use the secret key as one of the inputs at the first iteration [294, 295].

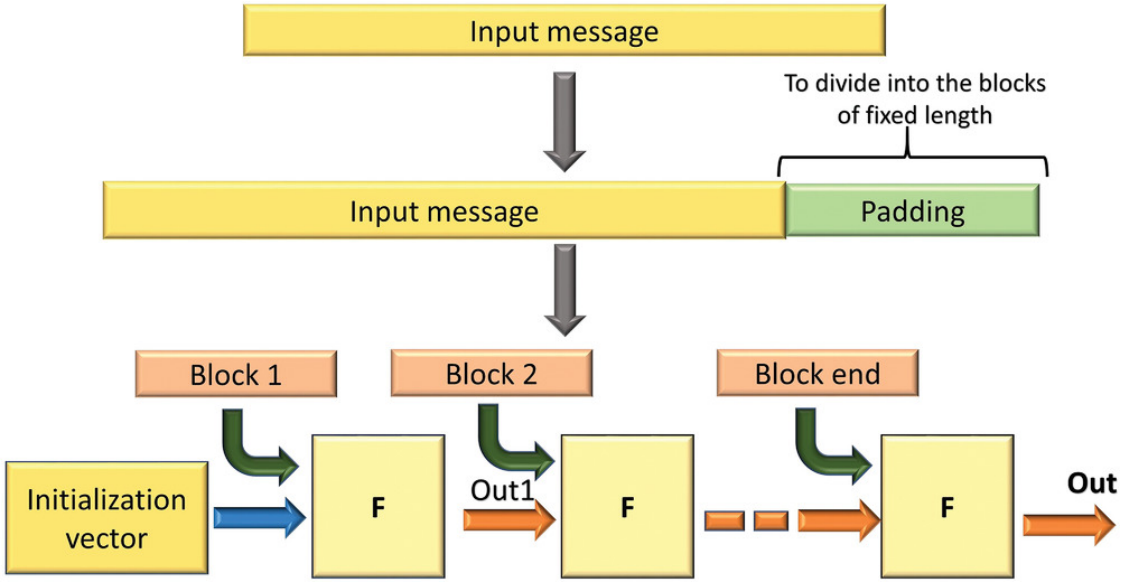


FIGURE 8.22: Merkle–Damgård construction. Here F is the compression function.

In the context of IoT, the lightweight hash functions are a promising solution for device authentication and establishing integrity [296–298]. Other than security strengths, system overhead and energy consumption are the common challenges in integrating it in resource constrained systems. Hardware hash functions implementation could be one alternative meeting both security and resource requirements.

Most of the works till now focus on building accelerators to improve the throughput and energy efficiency of established hash function algorithms [299, 300]. Though hash functions based on chaotic systems were explored [301–303], experimental and theoretical background in realising the physical hash functions still need to be appropriately investigated. Emerging NVM devices can be utilised for implementing a hash function. Azriel et al. proposed a passive crossbar RRAM based embedded key hash function exploiting sneak path current [304]. The proposed algorithm updates the devices' resistance states on top of their current state with the input bit utilising sneak path current on each step iteratively. Hence, one of the input bits is processed at each iteration. The hash value is retrieved at the end of all the iterations by the differential read circuit. Strachan et al. also proposed the RRAM based hash computation for network security applications [305]. The hash function is stored as the conductance matrix in the memristors at the crossbar intersection. The input string is mapped to the analog voltages, and it is applied to the rows of the crossbars. The output hash value is determined by the dot product of the applied input voltage to each row, and the conductance of the memristor devices at the crossbar intersection sensed along the column. Though few RRAM based hash functions were proposed, they still need to be experimentally validated. Some of the requirements for realising such architectures are the robust entropy source, the methodology to map the arbitrary sized input to the fixed-sized output, high throughput, and other hash function properties, including preimage and collision resistance and determinism.

8.7 Challenges for Security Applications

Security protocols rely on cryptographic primitives. Rapid proliferation of Internet-of-Things (IoT) and various other low-resource edge computing platforms led to an increasing demand of such primitives that can be fit under tight energy, area and timing budgets. This is being explored, from the theoretical perspective, within the sub-domain called lightweight cryptography [306]. However, the implementation of such lightweight cryptographic primitives in traditional and emerging computing technologies is still in a nascent stage. Moreover, there is a growing threat of physical cryptanalysis, which permits an adversary to infer the secret key of an operation by observing the physical manifestations of the computation,

e.g., power consumption, and electro-magnetic radiation. By introducing a new platform, i.e., in-memory computing, for cryptographic operations, designer may prepare with a fresh perspective striking the right balance between efficiency and security. Recent studies reveal that non-CMOS devices are not immune to such attacks either [307]. Considering these aspects, we suggest exploration of the following open research directions to fully unlock the potential of RRAM for security applications. • Mathematical Analysis: Cryptographic primitives need to provide provable robustness against a computationally well-equipped adversary. This requires one to study the presented construction by applying known cryptanalysis techniques, both by using extensive simulation as well as by constructing a mathematical model starting from the underlying device behavior [308]. • Implementation Study and Performance Benchmarking: The acceptability of RRAM based security application largely hinges on its improved performance numbers. While it is imperative that bulk encryption/decryption/hash functions stand to benefit from the data locality provided by in-memory computing [309], it is still not studied across a large set of representative benchmarks. Early results are promising [310]. • Side-Channel Attacks (Passive, Active): New techniques for side-channel attacks are emerging on regular basis. This not only includes attacks based on observation but, could be more invasive in nature, e.g., by implanting a Trojan hardware circuit that leaks sensitive information. The resilience of RRAM-based cryptographic primitives against such attacks can only be studied at implementation level. • Integrated Platform-level Security Analysis: Traditional system-level security protocols are based on the model, where memory, and computing blocks are clearly separated. Naturally, security analysis on the basis of such protocols do not apply to the scenario where computing happens close to the storage blocks. Consequently, a rigorous platform-level protocol design and analysis is required when RRAM-based cryptographic primitives are utilized.

8.8 Summary

We have reviewed the detailed developments in RRAM security and also addressed requirements in designing such security hardware. The larger resistance window, high-density integration passive crossbar structures and the availability of a wide range of functional materials for engineering are the primary advantages of this

RRAM technology. The switching resistance variability, sneak path current and random telegraph noise are some of the entropy sources that are particularly beneficial for hardware security applications. The demonstrated RRAM TRNGs are robust in maintaining the demanded entropy requirements over the wide range of operating conditions and have better throughput and energy efficiency, making them promising to be integrated into resource constrained IoT and edge computing platforms. One of the significant challenges of any PUF implementations generally is the native and worst-case BER as it necessitates the post-processing schemes. The RRAM PUFs can generate highly reliable responses, with the lowest BER even at extreme operating conditions than other technologies. Hence, they can be potentially integrated into resource constraint environments and further strengthen the system's security from attacks. The hardware hash functions could be a better replacement for the available complex software only protocols. The RRAM based hash function is still in preliminary development and needs appropriate experimental and theoretical investigations. We have also presented the future outlook of these security solutions and discussed several aspects that still need to be explored for each application separately, potentially leading to novel developments in the field.

Chapter 9

Design of Parity-Based Strong Reconfigurable RRAM Physical Unclonable Function

Synopsis

Physical Unclonable Functions (PUFs) have emerged as a highly promising security mechanism for authentication and secret key generation, specifically within the Root of Trust context. The emerging non-volatile memory device technologies such as memristor are widely studied for constructing PUFs. Reconfiguration stands as an indispensable functionality of PUFs. Once a PUF's challenge and response pairs have been exhausted or compromised, it is advisable to initiate reconfiguration. In the case of memristor PUFs, the resistance states of memristor devices are currently rewritten for reconfiguration. Although the complete set of challenge and response pairs of PUFs is generally collected and stored in a trusted server prior to deployment, securely collecting and storing the new CRP space of the memristor PUF during reconfiguration after deployment is proven to be a challenging task. Additionally, introducing on-chip test modules for quality checks is often a costly affair. To address this challenge, we propose a memristor PUF that

Chapter 9 is partially published as Rajendran, G., Zahoor, F., Singh, S., Merchant, F., Rana, V., Chattopadhyay, A. (2023, October). PR-PUF: A Reconfigurable Strong RRAM PUF. In 2023 IFIP/IEEE 31st International Conference on Very Large Scale Integration (VLSI-SoC) (pp. 1-6). IEEE.

allows writing-free at least one-time reconfiguration where both the original and reconfigured CRP space can be collected and stored securely in the trusted server before deployment. We also propose a post-processing module that can be attached to the proposed memristor PUF architecture to improve its characteristics. Our memristor PUF has shown a near-ideal PUF characteristic with a uniformity of 51.1%, bit-aliasing of 48.9%/53.4%, and uniqueness of 50.2% after post-processing. Furthermore, we have tested the robustness of our PUF against machine learning attacks and have found it to be reasonably resilient.

9.1 Overview

Nearly all electronic devices nowadays require a root of trust (RoT) component. This is necessitated because of the increased use of connected devices, and we expect them to handle all sensitive data securely. All essential cryptographic keys are generated and maintained inside the RoT and are likely to be tamper-resistant. The security primitives such as Physical Unclonable Function (PUF) and True Random Number Generator (TRNG) form the building blocks of RoT. A PUF is a hardware that can generate a secret key, which we call a response based on the applied input, referred to as a challenge. A single PUF can yield from one to many challenge-response pairs (CRPs). There are many applications of PUF, including authentication and secret key generation. The PUF utilizes inherent variations in the electronic components to generate the responses. The PUF can be broadly divided into weak and strong PUF based on the number of CRPs it can generate[32]. Weak PUF has a smaller CRP space which is linearly related to the number of noise-generating components. Strong PUF has a larger CRP space exponentially proportional to the size of the challenge bit.

Initially, CMOS technologies are studied to construct PUFs. But most of the time, it requires error correction codes to generate reliable and unique responses. Additionally, it is challenging to generate multi-bit responses with them [32]. Currently, in-memory computing architectures based on emerging non-volatile memory technologies such as resistive random access memory (RRAM), spin-transfer torque magnetic random-access memory (STT MRAM) are explored to achieve superior computation performance with less resource budget. This emerging device technology can also be used to construct security primitives such as PUF. It has been

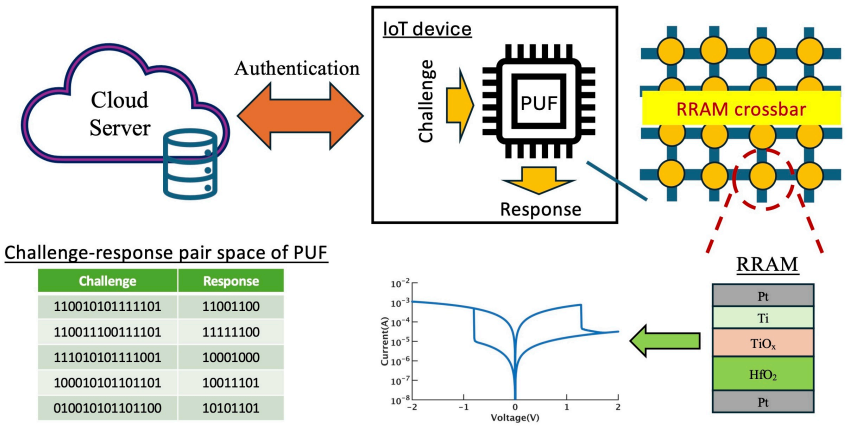


FIGURE 9.1: RRAM PUF-based authentication of IoT device

found that they are able to offer high reliability and a huge CRP space compared to CMOS PUFs, and can further generate multi-bit responses economically. We specifically focus on RRAM or memristor in this study. However, our discussion may be extended to other emerging device technologies.

The PUF is also expected to be reconfigurable to produce a different reliable response for the same applied challenge [32]. The reconfigured CRP space of the PUF is completely unique from its original CRP space. The reconfiguration is especially useful when the CRP space of the PUF is compromised or when expanding the CRP space after full utilization. Currently, the reconfiguration is achieved by triggering a new write operation on the RRAM devices after deployment [311]. This method has several drawbacks. First, it is costly to enroll the reconfigured CRP space in a trusted server securely. The standard methodology for authentication is to collect the entire CRP space of the PUF before deployment on the trusted server and during runtime after deployment, compare the responses generated by the PUF for the same applied challenge with that stored in the server as shown in Fig.9.1. Now, after reconfiguring the deployed PUF, collecting and storing the new CRP space on the trusted server is challenging. Second, on-chip testing is needed to assess the quality of the responses generated after reconfiguration. To overcome this issue, we propose an RRAM PUF construction that allows at least one-time reconfiguration, where both the original and reconfigured spaces can be collected into the trusted server before deployment.

The construction of RRAM PUF involves creating the entropy source in the crossbar and harvesting the generated entropy with a dedicated scheme. There are different procedures to generate the entropy in the RRAM crossbar, and it is fundamentally based on the Device to Device (D2D) variations[312, 313]. The first method involves applying a voltage pulse for which the switching probability is 50% to change the state from high resistance state (HRS) to low resistance state (LRS)[314]. A random bit map is created in the crossbar from this probability switching using D2D variations. The second method directly utilises the distribution in the D2D HRS in the fabricated devices[312]. The final method involves writing the RRAM crossbar with the random bit stream from the TRNG[315]. The RRAM PUF proposed in this work can be constructed from any of the discussed entropy sources. In this study, we utilize the D2D HRS variations as the randomness source in the crossbar.

The main contributions of this work are:

- We propose an RRAM PUF construction that allows write-free one-time reconfiguration where the reconfigured space can be collected before deployment.
- We propose a post-processing block that can be integrated into our RRAM PUF to improve its characteristics and robustness to machine learning attacks.

9.2 Construction of the proposed RRAM PUF

We have used the RRAM JART VCM model from [316] for our simulation study in Cadence, and this model has been experimentally validated. The model has four variable parameters, $N_{disc,min}$, $N_{disc,max}$, R_{var} and l_{var}, to set the D2D variations. All these four parameters are sampled from the truncated Gaussian distribution with the ranges shown in Table 9.1. We have constructed a 32×32 1T1R crossbar using CMOS 180nm technology. We have utilized the variations in resistance values of RRAM devices in the crossbar to create PUF. We selected high-resistance state variations as they exhibited a wider distribution range compared to low-resistance state variations. Fig.9.2(a) shows the voltage-current characteristics of RRAM

TABLE 9.1: RRAM model parameters

Parameter	**Range**
$N_{disc,min}[10^{23}m^{-3}]$	4 / 8 / 16
$N_{disc,max}[10^{26}m^{-3}]$	18 / 20 / 22
$r_{var}[nm]$	40.5 / 45 / 49.5
$l_{var}[nm]$	0.36 / 0.4 / 0.44

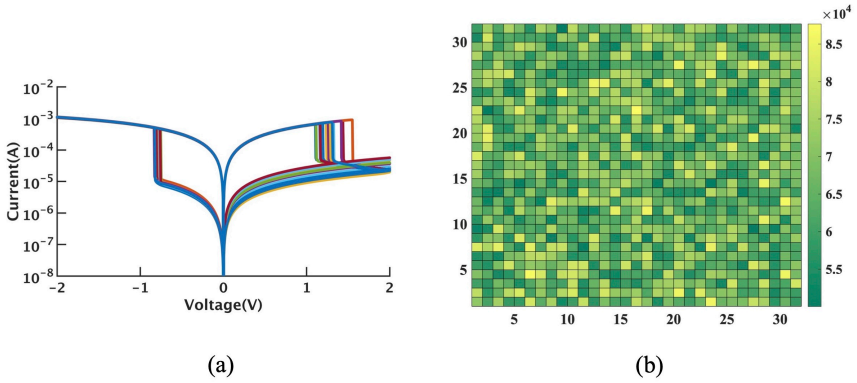


FIGURE 9.2: (a) Voltage-current characteristics of RRAM with device-to-device variations; (b) HRS distribution in the 32x32 RRAM crossbar utilised to construct PUF

devices when the variations are set according to Table.9.1. In our simulation study, the HRS variations are distributed across a 32×32 crossbar, as shown in Fig.9.2(b).

The basic schematic of our proposed RRAM PUF is detailed in Fig.9.3. The input challenge bits are applied to select the rows in the crossbar. If the challenge bit is 0, that row is unselected; if it is 1, then it is selected. The total current I_j through the individual columns results from the matrix-vector multiplication property of the RRAM crossbar, $I_j = \sum_{i=1}^{N} V_i G_{i,j}$, where $1 \leq i \leq N$ and $1 \leq j \leq N$ represents the row and column, respectively. The column selection is based on the parity of the applied challenge. Hence, the parity of the challenge is first calculated, and it is used to select the unique set of columns compared to generate the response. For example, if parity bit (P) = 0, an odd set of columns is selected and compared to create the response bit. On the other hand, if P = 1, an even set of columns is selected and compared. There are two types of parity schemes possible - odd and even. The even and odd parity schemes generate P = 1 when odd and even numbers of 1s are in the challenge, respectively and generate P = 0 otherwise.

The size of the CRP space of our RRAM PUF is 2^N, where N is the length of

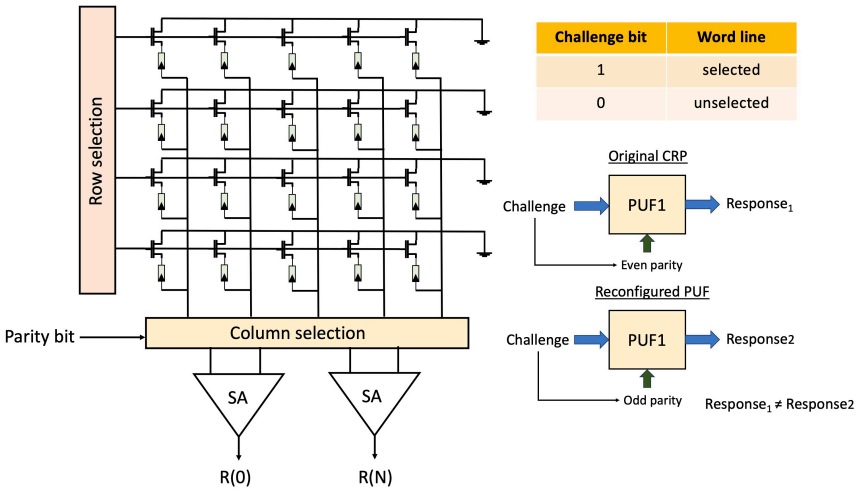


FIGURE 9.3: Schematic of the proposed one-time writing free reconfigurable RRAM PUF

the challenge. Thus, our RRAM PUF can be classified as strong PUF since the size of the CRP space is exponentially huge in the length of the challenge. The proposed PUF maps the N-bit input challenge to the (N/4) bit output response. For example, in this study, we utilise a 32×32 crossbar to construct PUF, which takes 32-bit input challenges and produces 8-bit responses. The advantage of the proposed PUF is a novel writing-free reconfiguration mechanism and the resistance to machine learning attacks, which will be discussed later here.

9.3 Characteristics of the proposed RRAM PUF

The PUF is expected to meet specific statistical requirements, described as uniformity, bit-aliasing and uniqueness [32]. All these statistical properties analyse the response generated by the PUF to the applied input challenge. We used 50K CRPs collected from 5 different RRAM PUF instances for this analysis.

9.3.1 Uniformity

The responses generated by a PUF should contain an equal number of 0's and 1's. The ideal value of uniformity is 50%. For the proposed RRAM PUF, we analysed

this property when operating under both odd and even parity-based schemes. As shown in Fig.9.4, the average uniformity is 51.4% for both parity schemes.

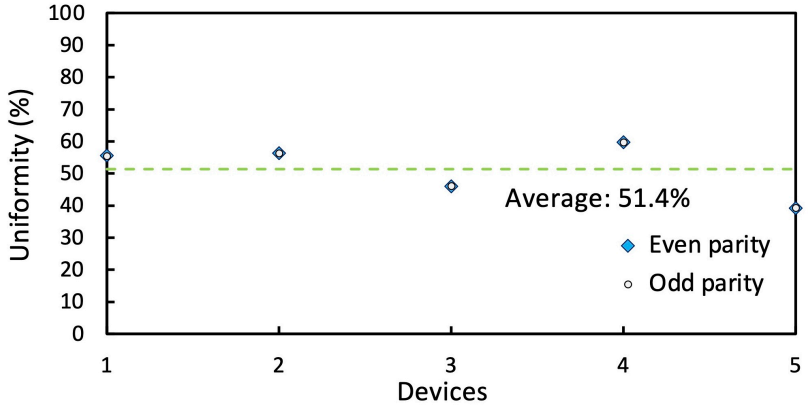


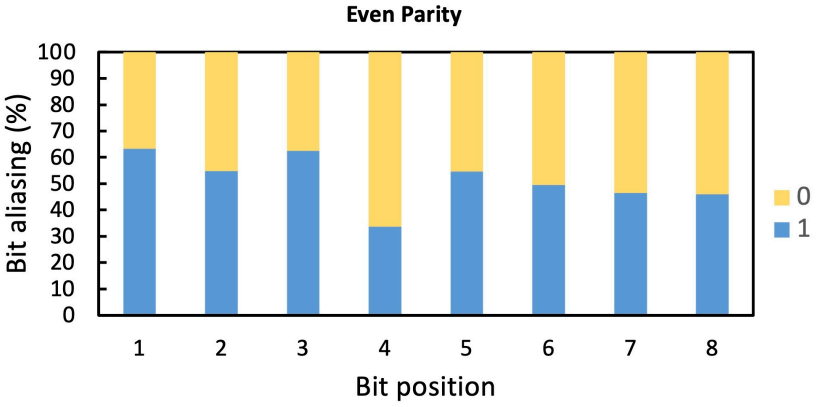
FIGURE 9.4: Uniformity of the proposed RRAM PUF

9.3.2 Bit-aliasing

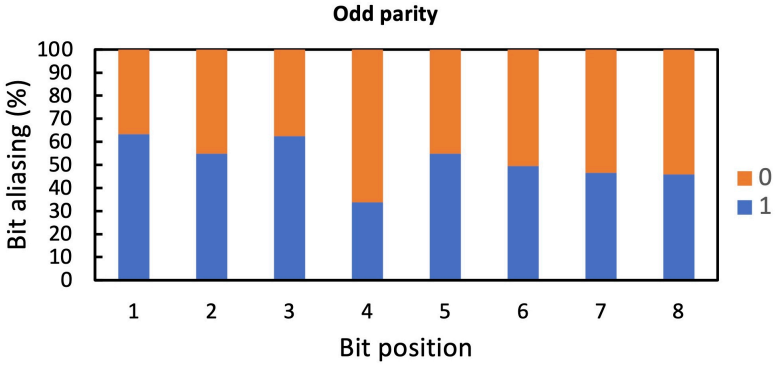
One of the notable merits of NVM PUFs is their capability to produce a multi-bit response. Hence, bit-aliasing is one of the important properties that is studied. It denotes that different PUFs for the same applied challenge are not inclined to produce 0s and 1s at a particular response bit location. The ideal value of bit-aliasing is 50%. For the proposed RRAM PUF, we found that bit-aliasing ranges from 33.6% to 63.3% for all 8 bit-positions when calculating for even and odd parity schemes, as shown in Fig. 9.5.

9.3.3 Uniqueness

Each PUF should generate distinct responses, which is referred to as uniqueness. The ideal uniqueness value is 50%. For our PUF, we computed an average uniqueness of 50.4% when operating in both odd and even parity schemes.



(a)



(b)

FIGURE 9.5: Bit-aliasing of the proposed RRAM PUF calculated for (a) odd and (b) even parity schemes

9.3.4 Reconfiguration

The response generation of the proposed PUF is based on the parity of the applied challenge. Hence, by changing the type of parity from odd to even or vice versa, the entire CRP space of the PUF can be reconfigured, as detailed in Fig. 9.3. The parity scheme also allows the service provider to collect and store both the original and reconfigured space before deployment on the trusted server. The reconfiguration quality of the proposed PUF based on the parity scheme is tested by calculating the uniqueness. It is given in Fig. 9.6, and we calculated the average uniqueness to be 50.5%. It shows that our PUF generates entirely different responses after reconfiguration for the same applied challenges. As discussed in the previous sections, the statistical quality of the generated responses is also unaffected by the type of

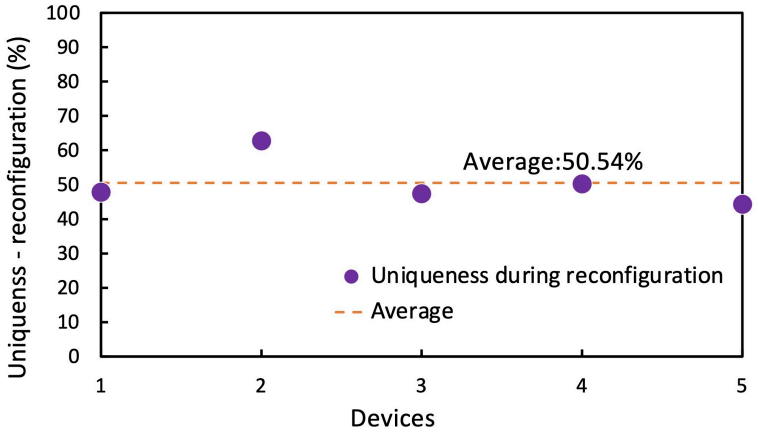


FIGURE 9.6: Uniqueness of the proposed RRAM PUF after reconfiguring between even and odd parity schemes

parity. Hence, the proposed parity-based scheme is an effective solution. Notably, reconfiguration can be achieved in our proposed PUF without altering the states of RRAM devices through a new write operation, thereby providing a significant advantage over previously demonstrated schemes.

9.4 Post-processing scheme

The quality of the proposed RRAM PUF can be further improved by integrating a post-processing module. We developed a post-processing block derived from the Blowfish cipher, as shown in Fig. 9.7. The raw response bit length of 8 is downsized to 2 with the post-processing block. However, it improves non-linearity, uniformity distribution, and ensures one-wayness. The quality of the response after post-processing is given in Table 9.2. It is evident that post-processing improves all the characteristics overall, especially bit-aliasing. As observed before, the type of parity does not impact the PUF characteristics. The post-processing block consists of basic digital components, such as XOR gate, and 2-bit adders. These elements contribute less than 1% to the total power consumed by the PUF module.

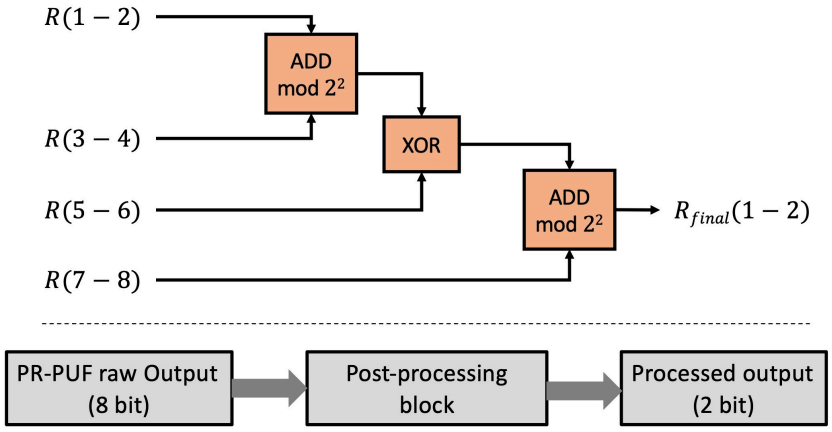


FIGURE 9.7: Proposed optional post-processing block

TABLE 9.2: Characteristics of the proposed RRAM PUF after post-processing

Metrics	**Even parity**	**Odd parity**
Uniformity	51.2%	51.1%
Bit-aliasing	Bit 1 - 48.9% Bit 2 - 53.4%	Bit 1 - 48.9% Bit 2 - 53.3%
Uniqueness (within PUFs)	50.2%	50.2%
Uniqueness (Reconfigured)	49.5%	

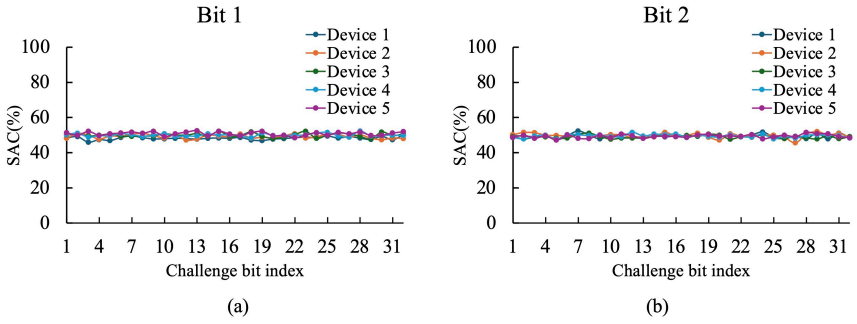


FIGURE 9.8: SAC of the proposed PUF operating in an even parity scheme

9.4.1 Strict-Avalanche criteria

A PUF is considered to have satisfied the Strict-Avalanche criteria (SAC) property - if a single bit in the challenge is flipped, the output has a 50% probability of changing. The objective of SAC is to confirm that every output bit is dependent on all the input bits. We have analyzed the SAC property of the proposed PUF after post-processing and found that it is nearly equal to the ideal 50% for both

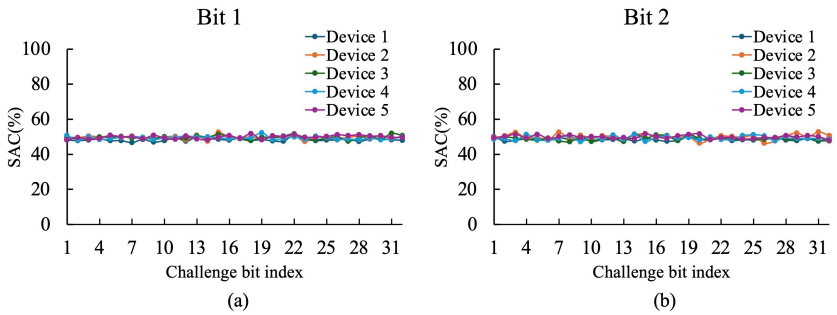


FIGURE 9.9: SAC of the proposed PUF operating in an odd parity scheme

parity schemes. Fig. 9.8 and 9.9 show the SAC of even and odd parity schemes, respectively, in response to a single bit flip in the challenge.

9.4.2 Machine learning attacks

We tested post-processed 2-bit responses of the proposed PUF with machine learning attacks as it is most likely used for practical implementations. The objective of this analysis is to ascertain the requisite number of CRPs needed by a determined attacker to effectively train a machine learning model. This model aims to predict the response generated by the PUF based on a given challenge. All the analyses are performed using Python packages, and the results are presented in Fig.9.10. The maximum learning accuracy of 75% is reached with multi-Layer perceptron. For the decision tree, k-nearest neighbors and random forest algorithm, we obtained the maximum learning accuracy of 55%, 61% and 64%, respectively. These results show that the proposed PUF and its novel response-deriving mechanism are robust to machine learning attacks.

9.4.3 Comparison

We have conducted a thorough comparison of our proposed RRAM PUF with other RRAM PUFs previously demonstrated, which has been presented in Table 9.3. Our proposed RRAM PUF is a strong PUF with a total size of 2^{2N} CRPs, which includes reconfiguration. Furthermore, our RRAM PUF exhibits standard PUF characteristics, with a near-ideal value of 50%. Importantly, we can securely enroll the reconfigured CRP space into the trusted server before deployment, unlike other

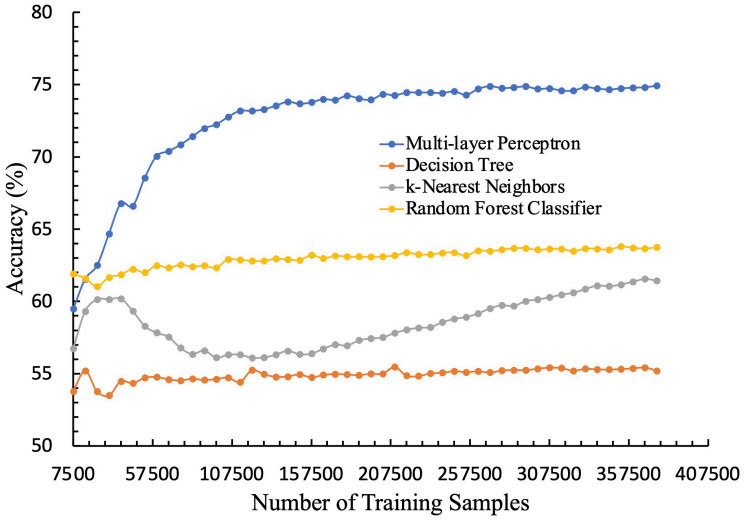


FIGURE 9.10: Machine learning studies on proposed RRAM PUF

demonstrated RRAM PUFs. We have also subjected our RRAM PUF to machine learning attacks, and it has proven to be reasonably robust.

TABLE 9.3: Comparison with other proposed RRAM PUF constructions

	VLSIT'18 [312]	IEDM'19[317]	SSC-L'20[318]	TCAS-I'20[315]	JSSC'21[313]	NANOARCH'23 [319]	**This Work**
Integration	0T1R	0T1R	0T1R	1T1R	1T1R	1T1R	1T1R
Entropy Source	D2D Variations	D2D Variations	D2D Variations	Jitter	D2D Variations	D2D Variations	D2D Variations
PUF type	Strong	Strong	Strong	Weak	Weak	Strong	Strong
Number of CRPs	$\backsim$ 2.37M	$\backsim 7 \times 10^{25}$	$\backsim 2^{80}$	-	64	1k/523.7k	$2^{32} + 2^{32}$
Uniformity	50.04%	49.95%	49.95%	50.01%	49.53%	50.146/49.956	51.1%
Uniqueness	50.12%	50.03%	49.25%	50%	49.9%	50.054/50.026	50.2%
Reconfigurability	Yes	No	No	Yes	Yes	Yes	Yes
Secure Reconfigured CRP Enrolment	No	No	No	No	No	No	Yes
Tested Modelling Attacks	Yes	Yes	Yes	No	No	No	Yes

9.5 Summary

In this work, we proposed a strong RRAM PUF design that allows for one-time writing-free reconfiguration. The selection of crossbar columns is dynamically determined based on the calculated parity bit of the input challenge, leading to the generation of response bits. Additionally, we proposed a post-processing block that can be integrated with our PUF architecture. Our analysis demonstrated that the proposed RRAM PUF meets the standard statistical requirements, including uniformity, bit-alising, and uniqueness. Furthermore, our findings indicated that the proposed RRAM PUF is capable of generating highly unique responses following reconfiguration. We also assessed the resilience of our RRAM PUF against machine learning attacks and further confirmed that it satisfies the Strict-Avalanche Criteria with nearly ideal 50% performance.

Chapter 10

Harnessing Entropy: RRAM Crossbar-based Unified PUF and RNG

Synopsis

Physical Unclonable Functions (PUF) and Random Number Generators (RNG) are two fundamental security components with applications in various tasks of a security protocol, such as key establishment, identity management, and authentication. In recent times, various PUF and RNG constructions have been proposed in the literature using RRAMs, demonstrating excellent properties, low energy consumption, and resistance to modelling attacks. Underlying these constructions, the source of entropy remains the same, and therefore, it is natural to design a unified architecture using an RRAM crossbar to function as a PUF and RNG. This is accomplished in the current paper. We propose a unified Parity-RRAM PUF (PR-PUF) and RNG construction and validate it for well-known PUF characteristics and RNG randomness properties with the NIST test suite SP 800-22. We present detailed design analysis and simulation-based studies to substantiate our ideas.

Chapter 10 is published as Rajendran, G., Zahoor, F., Thakker, S. S., Singh, S., Merchant, F., Rana, V., Chattopadhyay, A. (2024, January). Harnessing Entropy: RRAM Crossbar-based Unified PUF and RNG. In 2024 37th International Conference on VLSI Design and 2024 23rd International Conference on Embedded Systems (VLSID) (pp. 560-564). IEEE.

10.1 Overview

Root of Trust (RoT) is a necessity in the modern hardware environment. Cryptographic operations, device authentication, and secure boot processes are typical tasks performed by RoT, and they fundamentally require hardware security primitives such as Random Number Generators (RNG) and Physical Unclonable Functions (PUF) for their operations while ensuring integrity. Over the years, there have been several research and commercial implementations of PUF and RNG [320]. The concept of PUF is to derive the keys each time queried with a challenge instead of storing them immutably in a Non-Volatile Memory (NVM). The input challenge vector is mapped to the output response, and mapping is considered hidden inside the PUF and kept completely unknown, dictated by the entropy source of the physical components [321]. Based on the available number of Challenge-Response Pairs (CRPs), the PUF can be classified into Strong and Weak PUFs. Weak PUFs have a smaller CRP space linearly related to the number of entropy components, and are typically used for device-integrity and IC traceability solutions. Strong PUFs have a huge CRP space exponentially in the size of the challenge bits and entropy components, and hence, it can be used for secure authentication purposes. The PUF is expected to meet specific statistical requirements, as presented later in this paper.

RNGs are critical to producing a high-quality random bit stream in a security module. The security of the crypto algorithms heavily relies on the RNG as the initialization of security keys is from the RNG. Hence, it is always expected to meet the specific statistical requirement of randomness, and the standard procedure is to test the quality with the NIST test suite SP 800-22 [322]. Conventionally, when physical entropy sources are utilized to generate the random-bit stream, it is referred to as True RNG (TRNG), and when a proven mathematical framework is used to generate the random-bit stream, it is referred to as Pseudo RNG (PRNG). PRNGs are usually constructed from cipher blocks, and a TRNG is used to seed and initiate the PRNG [323]. The RNG in this work utilizes a physical entropy source for each cycle of 96-bit random number generation and also relies on a post-processing AES block and, hence, is outside this classification.

Related Works and Contributions: Though initial research focuses on constructing PUF and RNG from CMOS technologies, the emerging NVM device

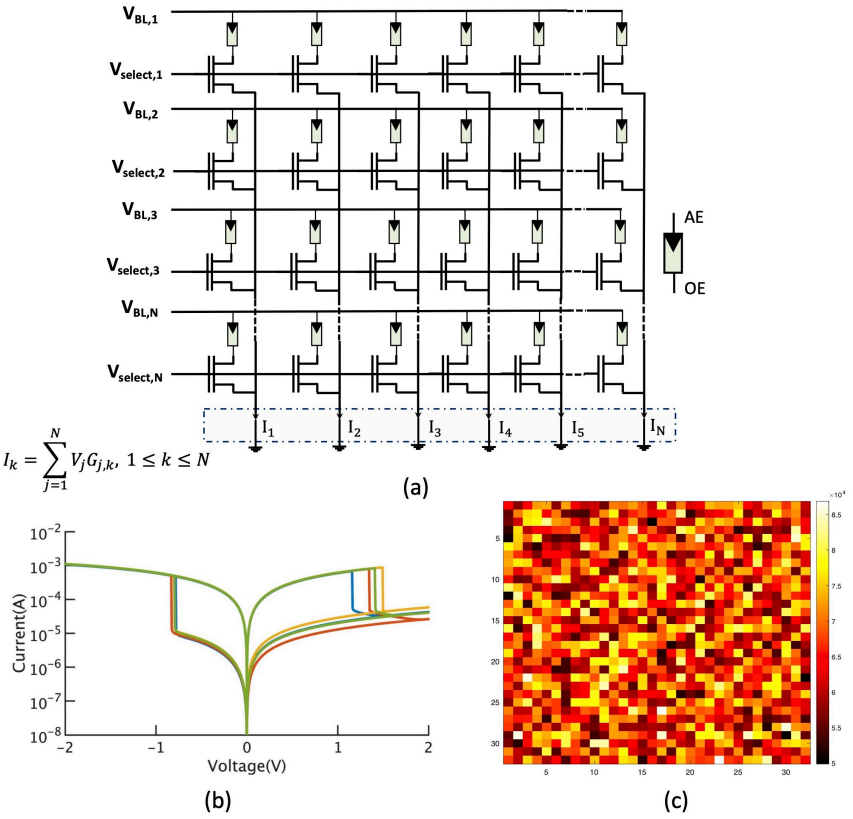


FIGURE 10.1: (a) 1T1R crossbar performing matrix-vector multiplication, (b) I-V characteristics of RRAM with D2D variations (c) HRS distribution in the 32×32 crossbar

technologies deliver the required high entropy sources and offer robust constructions against machine learning attacks that are difficult to achieve [321]. The Resistive Random Access Memory (RRAM) is one of the most studied emerging NVM device technologies. The basic operation of RRAM is based on a resistive switching mechanism - High Resistance State (HRS) to Low Resistance State (LRS) and vice versa. The PUF construction, based on RRAM, is prominently constructed from the variations in Device-to-Device (D2D) HRS / LRS [312, 313, 324, 325] and threshold in switching voltages [314]. Both weak and strong PUF can be constructed with RRAM. The basic construction methodology involves two steps - the first is selecting the targeted entropy sources, such as D2D variations or switching voltage, and generating it in the crossbar. The second is using the challenges to dynamically select the RRAM cells and generate the response bit based on the comparison results, which is the function of the entropy source. One of the essential features of the RRAM PUF is reconfiguration. When the CRP space is fully

utilized or compromised by the adversary, the original PUF is rewritten to create a new entropy referred to as reconfiguration, and the CRP space after reconfiguration is entirely different from the original.

RNG constructions from RRAM are based on $1/f^{\beta}$ noise of the read current [326] and cycle-to-cycle variations in the switching [327, 328]. There are also few attempts at constructing a unified structure, and most of the structures use the same basic construction but rely on different entropy sources for PUF and RNG operations [329]. Since PUF and RNG construction commonly rely on an entropy source and digitization produces a random bit stream, a more compact design can be constructed to harvest a single entropy source for both operations. In our previous study, we explored Parity-RRAM PUF (PR-PUF) [330] that offers to store both original and reconfigured space on the server before deployment. PR-PUF is a strong PUF with a huge CRP space, and we showed that it is robust against machine learning attacks. In this study, we utilized the PR-PUF to construct RNG to establish a unified construction. The major contributions of this work are as follows:

- We propose a methodology to construct an RNG from the PR-PUF based on a 1T1R crossbar that harvests the same entropy source based on D2D HRS variations for both operations.
- We discuss the characteristics of PR-PUF when constructed with HRS D2D variations and analyze the quality of random bit-stream generated from RNG with NIST SP 800-22 test suite.

The rest of the paper is organized as follows: Section II discusses the architecture of the PR-PUF based on the 1T1R crossbar and the characteristics of the discussed PR-PUF. Section III details the RRAM RNG design extended from PUF with NIST randomness test results, and Section IV concludes the paper.

10.2 Architecture of PR-PUF

In this study, we have utilized a 32×32 1T1R crossbar performing matrix-vector multiplication to construct the PUF. The 1T1R cell consists of an NMOS transistor from cadence GPDK180 and RRAM from the JART VCM variability model [316].

The D2D variations of the RRAM are only considered, and other crossbar non-idealities are ignored in the simulations. The model parameters for the RRAM device variations such as $N_{min,var}$,$N_{max,var}$, r_{var} and l_{var} are sampled from the truncated Gaussian distributions as recommended in [316]. We have already mentioned in a separate study that the discussed PR-PUF can be constructed from any entropy scheme and is not attached to a specific type [330]. The entropy source harvested and digitized in this study is the raw D2D HRS variations of the RRAM cells in the crossbar shown in Fig. 10.1.

Our PUF design is based on the matrix-vector multiplication property of the crossbar. If the challenge bit is one, that row is selected; otherwise, it is unselected. The response generation is based on comparing the total current I_k generated along the columns of the crossbar. Let $G_{j,k}$ represent the conductance state of the 1T1R cell, where $1 \leq j \leq N$ and $1 \leq k \leq N$ represents the row and column, respectively, and V_j be the applied read voltage across that cell, then $I_{j,k} = V_j G_{j,k}$ gives the current through that cell. Due to the D2D variations, $G_{j,k}$ differs for each cell, as seen as HRS variations in Fig.10.1. The total current I_k at each column for the applied challenge C_{state} can be calculated as, $I_k = \sum_{j=1}^{N} C_j I_{j,k}$ where C_j represents the challenge bit at position j. We only computed the $I_{j,k}$ through Cadence and used it further as the algorithmic analysis similar to [330, 331] in Python and MATLAB.

The working of the PR-PUF is as follows: for the applied challenge, its parity is first calculated and used to select the set of columns to be compared. For example, the parity bits 1 and 0 can select the even and odd number of columns in the crossbar, respectively. Then, response bits can be generated by comparing the total current at the selected columns adjacently, as shown in Fig.10.2 (Example: for Parity bit = 0, if $I_1 < I_3$, $R_1 = 0$; else $R_1 = 1$). The advantage of this scheme is the reconfiguration. By changing the type of parity from even to odd or vice versa, the set of the columns compared can be changed, thereby remapping the entire CRP space. Hence, the original and reconfigured CRP space can be collected and stored before deployment.

10.3 Study of PUF Characteristics:

Our previous study analyzed this PR-PUF with the entropy source from RNG. In this study, we use D2D HRS variations of the RRAM cells in the crossbar to

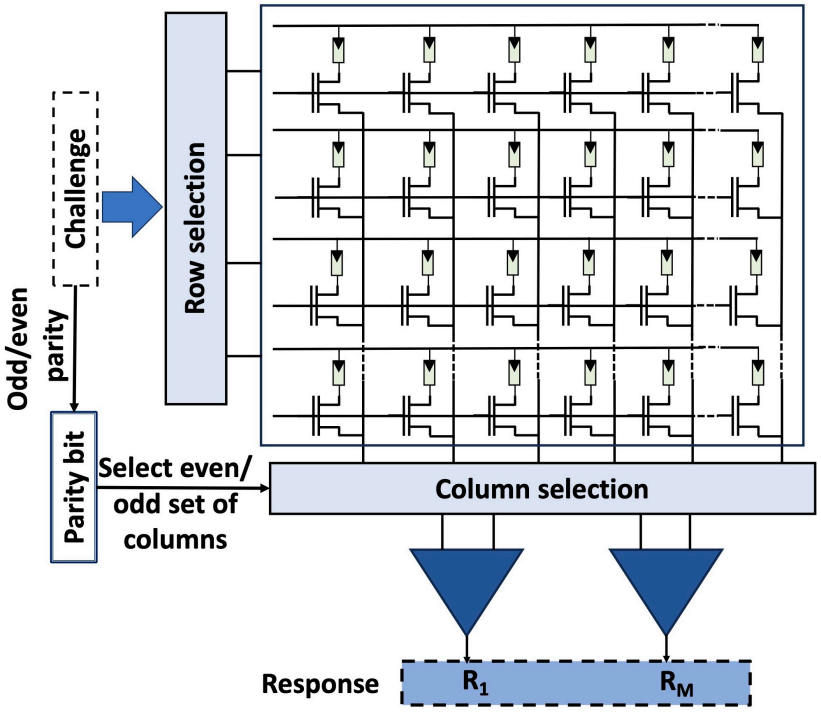


FIGURE 10.2: Schematic of PR-PUF based on 1T1R crossbar

create the PR-PUF. We tested the quality against the standard PUF characteristics mentioned in [332] and reported it below. We generated 50K CRPs, and all the characteristics were tested for 5 instances.

10.3.1 Uniformity

It signifies that PUF should generate an equal number of zeros and ones in its responses. As seen in Fig.10.3, the PR-PUF can meet this requirement with an average uniformity of 52.24% and 52.26% when analyzing with even and odd parity, respectively. It should be noted that the odd and parity-based schemes have similar uniform properties since, even after reconfiguration, the responses are generated from the same PUF schematic utilizing the same entropy source.

10.3.2 Bit-aliasing

It is analyzed for the same applied challenge between different PUF instances, whether they are inclined to produce ones and zeros at the specific response bit-position. We analyzed the bit-aliasing for all the 8-bit positions of the responses

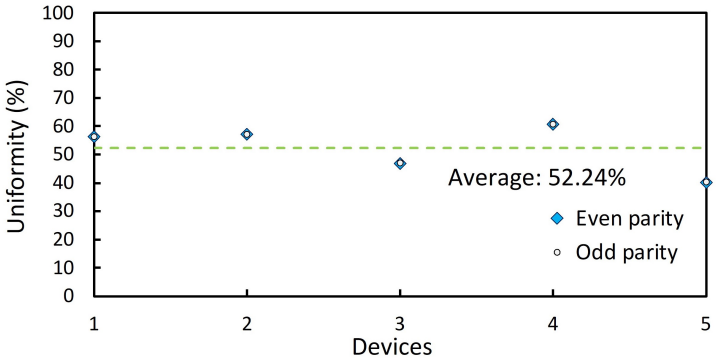


FIGURE 10.3: Uniformity of PR-PUF with reconfiguration

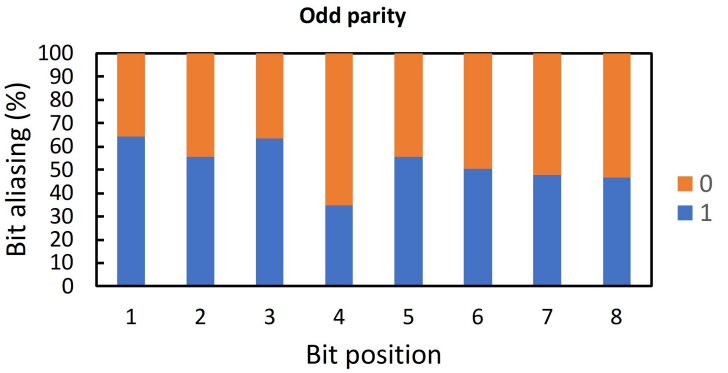
for PR-PUF. We found that bit-aliasing is within the accepted range of 34.51% to 64.17% for the even parity scheme, and a similar range is observed for the odd-parity scheme as seen in Fig.10.4. The post-processing block proposed in our previous study can further improve this characteristic [330].

10.3.3 Uniqueness

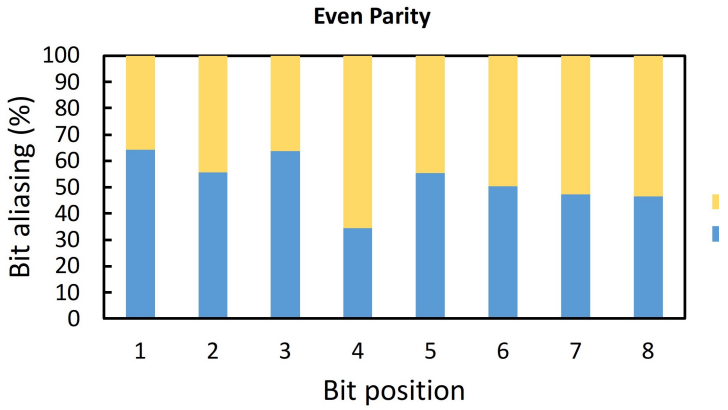
Each PUF instance should generate a distinct response for the same applied challenge. We calculated uniqueness for the PR-PUF and found that it meets this requirement with an average uniqueness of 50.32% and 50.34% for odd and even parity schemes, respectively. We have also checked the uniqueness after reconfiguration for individual PR-PUF instances and found that PR-PUF can generate unique responses in this case with an average uniqueness of 50.46% as given in Fig.10.5.

10.3.4 Reliability

PUF must produce the exact response when the same challenge is applied multiple times. There are different techniques to improve the reliability [331, 333], and we have also reported in our previous study that PR-PUF can be constructed with high reliability from any entropy source written generated in the crossbar. Hence, we report that this property can be satisfied for PR-PUF. It should also be noted that this property does not contribute to the statistical distribution of CRP space but rather the robustness of the PUF device itself in reproducing the responses.



(a)



(b)

FIGURE 10.4: Bit-aliasing of PR-PUF for (a) odd and (b) even parity based response generation

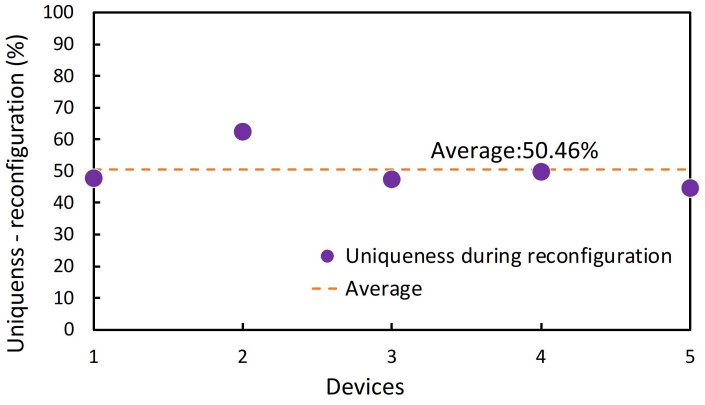


FIGURE 10.5: Uniqueness of PR-PUF after one time reconfiguration by changing the type of Parity

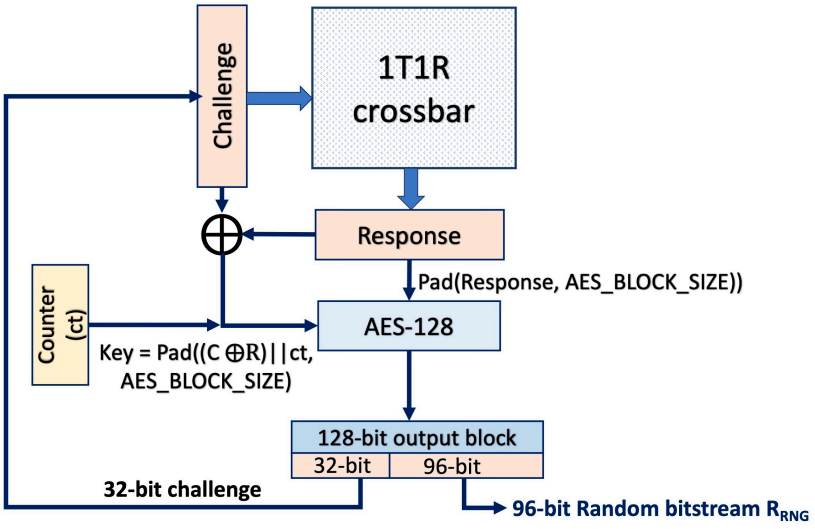


FIGURE 10.6: Schematic of RNG based on PR-PUF with AES-128

10.4 Proposed RRAM RNG design

An input challenge is required to generate the response in PUF, and the challenge here controls the entropy. In PR-PUF, the challenge vector dictates which rows of the RRAM crossbar are selected, and the randomness in the HRS variations of the RRAM devices generates different total currents along the columns. When digitized using a parity-based scheme, a unique response is generated for the applied challenge. On the other hand, there is no user-given input challenge for the RNG, and when it is just activated, the entropy is digitized and post-processed to generate a high-quality random bit stream. The common element in both primitives is the requirement of a randomness source. Hence, we propose using the same randomness from the RRAM PUF to construct RNG.

10.4.1 Extension with RNG Construction

One of the recommended building blocks by NIST is using the AES as a conditioning component to generate a high-quality random bit stream [334]. We used AES with the PR-PUF to operate as an RNG, and the proposed schematic is shown in Fig.10.6. During the operation as RNG, the PR-PUF is connected directly to AES-128. The response of the PR-PUF is fed as a message to the AES after padding it to the required block size of 128 bits. The challenge for PR-PUF is now fed back internally from the RNG output collected from AES. When the RNG is initialized

for the first time, a challenge of 0xFFFFFFFF is used to derive the response. The subsequent challenges are retrieved internally from the portion of RNG output, the first 32 output bits of the AES, as detailed in Fig.10.6. These challenge bits are hidden inside the device and not revealed to the user.

Algorithm 1. details the process of deriving the RNG bit streams from PR-PUF. It should be noted that the proposed methodology is generic to work with any PUF. The input to the AES is the key and message. The response is fed directly as a message after padding. However, for the key, the challenge is XORed with the response, and a counter output is padded to it. In this study, we have used the 8-bit counter that resets every time it reaches its maximum value. The counter prevents the architecture from repeating the same output in cycles. The AES padding mechanism utilized for the message is also used to pad the key to meet the 128-bit block size. The AES output is divided into 32-bit and 96-bit blocks; the former is used as a new challenge, and the latter is released as a random bit stream. Thus, the proposed RNG can produce 96 random bits every cycle.

Algorithm 3 Random Number Generation

Output: Random bit stream R_{RNG}
Initialisation : $C_{state}, counter$ ▷ Restore from last state
$R_{PUF} = \text{PUF}(C_{state})$
$R_{PUF,key} = [\,]$ ▷ Expand R_{PUF} to size of C_{state}
$i = 1$ to $\frac{N}{M}$
$R_{PUF,key} = R_{PUF,key} || R_{PUF}$
$\text{key} = \text{PAD}(C_{state} \oplus R_{PUF,key} \parallel counter, 128 - bit)$
$\text{M} = \text{PAD}(R_{PUF}, size = 128 - bit)$
$R_{AES} = \text{AES(M,key)}$
$counter < 255$
$counter = counter + 1$ $counter = 0$ ▷ Reset counter
$C_{state} = R_{AES}[1:32]$
$R_{RNG} = R_{AES}[33:128]$
return R_{RNG}

10.4.2 Randomness Evaluation

We tested the generated bit-streams from the proposed RNG with the NIST SP 800-22 test suite. We generated 50M random bits, divided into 10 sequences, and used them for the analysis with the test suite. We tested for all 5 constructed PUF instances, and one of the analyses is shown in Table.10.1, and it is clear that the

TABLE 10.1: NIST SP 800-22 test results

Tests	**P-Value**	**Result**
Frequency	0.122325	Pass
Block Frequency	0.534146	Pass
Cumulative Sums	0.991468	Pass
Cumulative Sums	0.213309	Pass
Runs	0.739918	Pass
Longest Run	0.739918	Pass
Rank	0.213309	Pass
FFT	0.350485	Pass
NonOverlapping Template	0.739918	Pass
Overlapping Template	0.017912	Pass
Universal	0.911413	Pass
Approximate Entropy	0.350485	Pass
Random Excursions	0.350485	Pass
Random Excursions Variant	0.534146	Pass
Serial	0.534146	Pass
Serial	0.911413	Pass
Linear Complexity	0.534146	Pass

RNG output is a high-quality random bit stream passing all NIST tests. Also, one test result is given as a sample from all the passed Non-overlapping template, Random Excursions, and Random Excursions Variant tests.

TABLE 10.2: Comparison with state-of-the-art RRAM PUF-RNG constructions

	2016[326]	2019[328]	2019[314]	2020[335]	2021[313]	2022[329]	**This Work**
RRAM Integration	1T1R	1T1R	1T1R	1T1R	1T1R	1T1R	1T1R
Entropy Source	$1/f^{\beta}$ noise	Cycle-to-cycle pulse number	RRAM Switching voltage	RRAM switching current	D2D Variations	D2D Variations, Read noise	D2D Variations
PUF	✖	✖	✔	✖	✔	✔	✔
RNG	✔	✔	✖	✔	✖	✔	✔
Unified structure	✖	✖	✖	✖	✖	✔	✔
NIST pass	All	All	NA	12	NA	All	All

We explained constructing RNG from PUF using the 1T1R crossbar harvesting the same HRS D2D randomness of the RRAM devices. The major advantage of the proposed methodology is that we can now have a unified architecture that acts like PUF and RNG, introducing compactness in design. During PUF operation, the user input challenge is applied to the PUF and the derived response is shared outside directly or after a post-processing block discussed in our previous work. However, when operated as RNG, the challenges are derived internally, and only the

random bit-stream R_{RNG} is shared outside. It is noted that careful consideration should be given to storing the C_{state} and counter, which is invoked and used in subsequent RNG calls. Table.10.2 compares the state-of-the-art RRAM PUF and RNG studies. The construction studied in this work utilizes a single entropy source, unlike separate entropy sources for PUF and RNG in other works, and importantly, passes all the NIST randomness tests. Furthermore, AES is included in commercial crypto-coprocessors [336] because of its vast use cases. Hence, designing RNG with AES is an effective solution as it is included irrespective of the RNG construction in the cryptoprocessors. Therefore, the overhead cost of our proposed AES utilization is negligible.

10.5 Summary

In this work, we presented a unified PR-PUF and RNG with 1T1R crossbar performing matrix-vector multiplication. The HRS D2D variations of the RRAM devices are the primary source of entropy utilized to generate the responses of the PUF. The discussed PR-PUF is the strong PUF, and the main advantage of its construction is its reconfigurability, which is achieved by changing the parity type during response generation. We mainly proposed and analyzed an RNG from the PR-PUF construction as a unified structure. The generated high-quality random bit streams from the RNG passed all the NIST SP 800-22 tests. We invite researchers to study the PR-PUF/RNG and identify opportunities to improve and analyze new attacks that could be mounted with corresponding countermeasures.

Chapter 11

Securing Binarized Neural Networks via PUF-based Key Management in Memristive Crossbar Arrays

Synopsis

Binarized neural networks (BNNs) are a subset of deep neural networks proposed to consume less computational resources with a smaller energy budget. Recent studies showed that memristor-based in-memory computing architectures can be constructed to accelerate BNNs, with better performance compared to traditional CMOS technologies. The memristor non-volatility utilized for in-memory computing poses a notable threat to theft attacks in the presence of adversaries with physical access. This motivates us to introduce two novel protection methodologies to safeguard the model parameters of BNNs in the memristive crossbar. We propose to take advantage of Physical Unclonable Functions (PUFs), which can be implemented using memristor-based crossbars for protecting BNN. This feature provides superior security compared to the traditional stored-key-based schemes. We provide circuit-level hardware designs to implement our methodologies with negligible additional overhead compared to an unprotected design and detailed supporting analysis to validate our security claims.

Chapters 11 is published as published as Rajendran, G., Basak, D., Deb, S., Chattopadhyay, A. (2024, July). Securing Binarized Neural Networks via PUF-based Key Management in Memristive Crossbar Arrays. IEEE Embedded Systems Letters (ESL).

11.1 Overview

Deep neural networks (DNN) are used in many applications today and have tremendous business value. The data used for training often concerns privacy, and training a high-accuracy model requires considerable resources. Hence, the DNN model is considered intellectual property and a valuable asset to protect from theft. The deployment of the trained model to the dedicated inference hardware in the edge can be handled well through the standard key exchange and encryption schemes. Hence, man-in-the-middle attacks can be prevented. However, the vulnerability of retrieving model parameters from the inference hardware still poses a considerable threat.

Binarized neural networks (BNN) are an emerging class of neural networks in which weights and activations are constrained to take values +1 and -1. BNN is attractive for edge applications because its binary computation during inference reduces computational complexities and saves energy. Implementing BNN with emerging non-volatile memory (NVM) devices, especially resistive random access memory (RRAM), is one of the primary research areas, and several methodologies and optimizations have been proposed to achieve better inference performance [33, 34]. Accelerating DNN with RRAM-based in-memory computing is challenging because RRAM non-idealities hinder multi-bit computing. However, for BNN, only two RRAM states are required and are more robust to variability. Implementing BNN in RRAM crossbars has the advantage that a current comparator can replace the resource-hungry analog-to-digital converter[33, 34]. Hence, the BNN crossbar arrays also necessitate significantly fewer peripheral resources. Consequently, RRAM-based BNN accelerators could be preferred over their multi-bit DNN counterparts for edge implementations.

Other than for NN accelerators, emerging NVM devices are also studied for hardware security primitives such as Physical Unclonable Function (PUF) and True Random Number Generator (TRNG) [32]. A PUF works by producing a response when a challenge is given to it. The response can be either used as a key or to derive new key(s). A single PUF device can take many input challenges and produce corresponding responses, making it flexible to tailor for applications after deployment. PUFs are more secure than stored keys that are prone to theft by probing. Unlike the stored-key paradigm, PUFs can also be reconfigured after deployment.

After reconfiguration, the PUF will produce a different response for the same applied challenge. Hence, if the keys are compromised, PUFs can still be used after reconfiguration [32].

11.2 Related works

To avoid communication overhead, the BNN accelerators are often deployed in edge devices, which makes those physically accessible to the attackers. There are few previous works on securing multi-bit DNN in RRAM crossbars against theft attacks but not on BNN. They essentially transform the weight matrix before mapping it into the crossbar using a stored key. The first technique proposed is shuffling, in which the weight matrix is shuffled along the rows, columns, or both and the shuffled weights are mapped into the crossbar [337–339]. The shuffling details, such as the actual position of the weights, are used as the key. This method has several drawbacks: i) it is not PUF-friendly because the weight matrix indices are used as keys. Since the PUF output is a binary response, it cannot be directly utilized as a key for this method. Also, the index-based key is very long, which is unnecessary even if it gives factorial security strength in brute-force attacks. ii) The peripheral overhead is very high, penalizing runtime, area and energy. The second technique proposed with regular DNN is the weight matrix transformation by 1's complement [340] along the column. This scheme is attractive compared to the shuffling technique due to its low resource overhead. However, this technique is unexplored for BNNs and optimized architectures, such as integrating the batch normalization (BN) layer with the matrix-vector multiplication (MVM) layer [34]. The BN-integrated architectures of BNNs differ significantly from standard DNN architectures. Hence, schemes proposed for standard DNN cannot be directly extended for BNNs. The third technique proposed is to pad the actual weight matrix with extra weights [339]. This technique has no quantifiable security benefits except in hiding the network topology.

The main contributions of this work are as follows. *First*, we propose two novel PUF-based weight transformation techniques for BNNs deployed in memristive crossbars: inversion and swapping. The row inversion and swapping techniques are discussed for the first time, even from the context of RRAM-based multi-bit DNN

with stored keys. *Second*, we present hardware implementations of each methodology with detailed discussions on the inference performance. We also conducted power analyses to estimate the additional overhead incurred.

11.3 Binarized Neural Network in RRAM crossbars

The trained binarized weights $W^b_{j,k}$ of the BNN are directly mapped to the RRAM crossbar as resistance states of the memristors. The standard mapping uses two RRAM cells to represent a single binary state arranged along the columns. The binary weight value +1 is implemented by programming the top cell to LRS and the bottom cell to HRS. For -1, it is implemented oppositely. It is equivalent to specifying +1 as (1,0) and -1 as (0,1). The binary inputs also follow the same convention for +1 and -1. The output of the MVM operation y_k is given as input to the activation unit, which is the sign function of the BNN. One of the principal strategies for implementing BNN is to integrate the BN and the MVM layers in a single crossbar and use the BN layer as a threshold B_k in calculating the neuron output y^b_k with a comparator [34]. It is achieved by rewriting the BN layer parameters through the sign function as given in equations 11.1 and 11.2. Here, k refers to the column index, and β, σ, ϵ, γ, and μ are the parameters of the BN layer.

$$B_k = -\beta \frac{\sqrt{\sigma^2 + \epsilon}}{\gamma} + \mu \tag{11.1}$$

$$y^b_k = \begin{cases} -1, & \text{if } y_k < B_k \\ 1, & \text{otherwise} \end{cases} \tag{11.2}$$

The B_k can take decimal values. Hence, it cannot be mapped to the RRAM crossbar directly. One of the notable properties of BNN is that if the weight matrix is of even number size, then its corresponding MVM output y_k will always be an even number. Hence, utilizing this property, the B_k can be modified to $2\lceil \frac{B_k}{2} \rceil$ and mapped to the RRAM devices in the crossbar. There are two ways to implement it. The first way is to have two separate columns for MVM and BN and compare their output currents using the comparator to generate the output bit y^b_k. We refer to this architecture as a double-column BN in this paper. The second way is to

integrate the BN and MVM in a single column and compare it to a reference value -1. The B_k is sign inverted before integrating into a single column. We refer to this architecture as a single-column BN.

Attacker model: The attacker can access the edge hardware with an RRAM crossbar, in which the trained model has been deployed for execution. The attacker knows the topology of the deployed BNN. The attacker can read the conductance states of RRAM devices using micro-probing, which has been identified as a viable method [337–342]. The attacker can also employ other physical attacks, such as elemental analysis using energy dispersive x-ray spectroscopy and imaging, as described in earlier studies with experimental validation, to find the conductance state of RRAM devices [342, 343]. However, the PUF and its response in the edge device are always hidden and protected from the attacker at any point in time. Hence, the attacker cannot access PUF and collect its responses making machine learning attacks not applicable. Here, it should be noted that the responses are never stored as memory states with PUF. The PUF always computes them during runtime. Our goal is to prevent the attacker from correctly reading the BNN model parameters from the RRAM crossbar.

11.4 Proposed security schemes

For the BNN inference, the trained weight-matrix $W^b_{j,k}$ of each layer is directly mapped to the RRAM crossbar. We propose transforming the $W^b_{j,k}$ into a new matrix $W^{*b}_{j,k}$ with a PUF response and then storing it in the crossbar. Hence, when $W^{*b}_{j,k}$ is used for inference without the PUF response used for transformation, the classification will be incorrect, and the model will be useless. We previously proposed a reconfigurable Parity RRAM PUF with a 1T1R crossbar and a procedure to extend it to a random number generator [344]. During the inference stage, the keys generated by the PUF are loaded only once at runtime. They are then stored in registers, such as D flip-flops, for use in subsequent inferences involving multiple inputs. As a result, the power consumption will be reduced to D flip-flops. Our analysis shows that the additional power overhead from using D flip-flops accounts for less than 1% of the total power consumption. We used a similar PUF construction for analysis in this paper. We propose two methodologies with PUF to protect the trained BNN parameters of each layer. We analyse our schemes using the BNN architecture shown in Fig.11.1 and train it on the MNIST dataset. In this study,

we used the RRAM model discussed in [316] for the simulations. We need to determine the level of difficulty an attacker would face when attempting to recover $W^b_{j,k}$ from the $W^{*b}_{j,k}$ - after gathering $W^{*b}_{j,k}$ from the RRAM crossbar. We estimate this security strength for our proposed methodologies in terms of the brute-force approach and the effectiveness of weight matrix transformation in inference accuracy following similar state-of-the-art works [337–340].

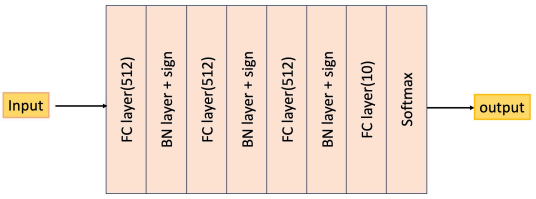


FIGURE 11.1: Architecture of Neural Network trained used for MNIST dataset

11.4.1 Transformation by Inversion

We propose two inversion strategies for $W^b_{j,k}$ - one is along the row j, which we refer to as the row inversion, and the other is along the column k, which we refer to as a column inversion. The length of the PUF response required for row and column inversion equals the row size, l_j and column size, l_k of $W^b_{j,k}$, respectively. The security strength of the $W^{*b}_{j,k}$ against brute force attack depends on the length of the response. For a response size of length l, it is 2^l.

11.4.1.1 Row inversion

- This transformation can be completed using equation (11.3), where R_j and $W^b_{j,1:l_k}$ refer to the response bit value and weight matrix elements at row index j, respectively. Only the transformed matrix $W^{*b}_{j,k}$ is mapped to the crossbar. During inference runtime, the same PUF response used for transformation is used to transform input signals X^b_j applied to the crossbar as given in equation (11.4). Both double-column and single-column BN architectures are implemented with the same transformation flow as detailed in Fig 11.2. The inversion logic required at the input is the XOR gate that takes X^b_j and R_j as inputs and outputs the transformed input signal X^{*b}_j to the crossbar, as detailed in Fig.11.2.

$$W^{*b}_{j,1:l_k} = \begin{cases} -W^b_{j,1:l_k} & \text{if } R_j = 1 \\ W^b_{j,1:l_k} & \text{otherwise} \end{cases} \tag{11.3}$$

$$X^{*b}_j = \begin{cases} -X^b_j & \text{if } R_j = 1 \\ X^b_j & \text{otherwise} \end{cases} \tag{11.4}$$

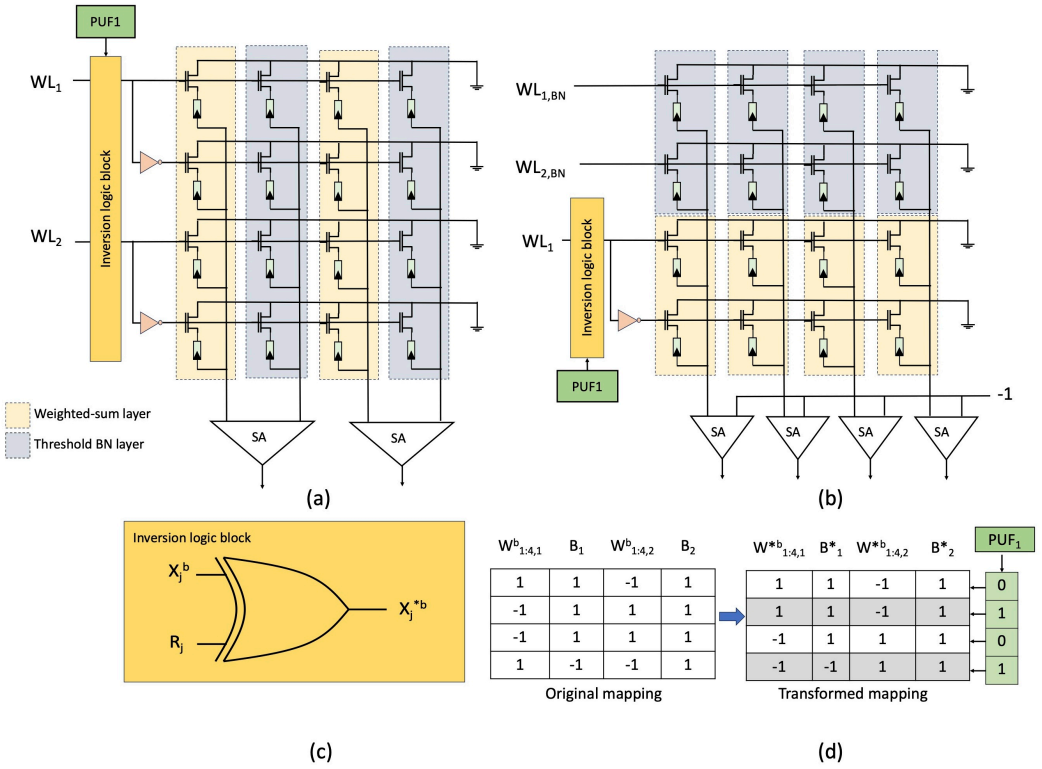


FIGURE 11.2: Layout of PUF-protected row inversion of (a) double-column and (b) single-column BN architectures, (c) inversion logic to recover the inference output and (d) illustration of transformation including BN layer

11.4.1.2 Column inversion

- The response bit value R_k of PUF is directly coupled to the column k of the weight matrix. In row-inversion, we transformed only the weight matrix $W^b_{j,k}$. However, both $W^b_{j,k}$ and B_k are transformed in column inversion. For double-column BN architectures, $W^b_{j,k}$ and B_k are transformed using the mappings in equations (11.5) and (11.6), respectively. Similarly, the mappings in equations (11.5) and (11.7) transform $W^b_{j,k}$ and B_k of single-column BN architectures, respectively. The comparator output y^{*b}_k is checked for sign and inverted during inference runtime with

the PUF response as given in equation (11.8) for both the architectures before passing it to the next inference stage. The required inversion logic is the XOR gate that takes the y_k^{*b} and R_k as inputs and gives y_k^b as output, as detailed in Fig.11.3(c).

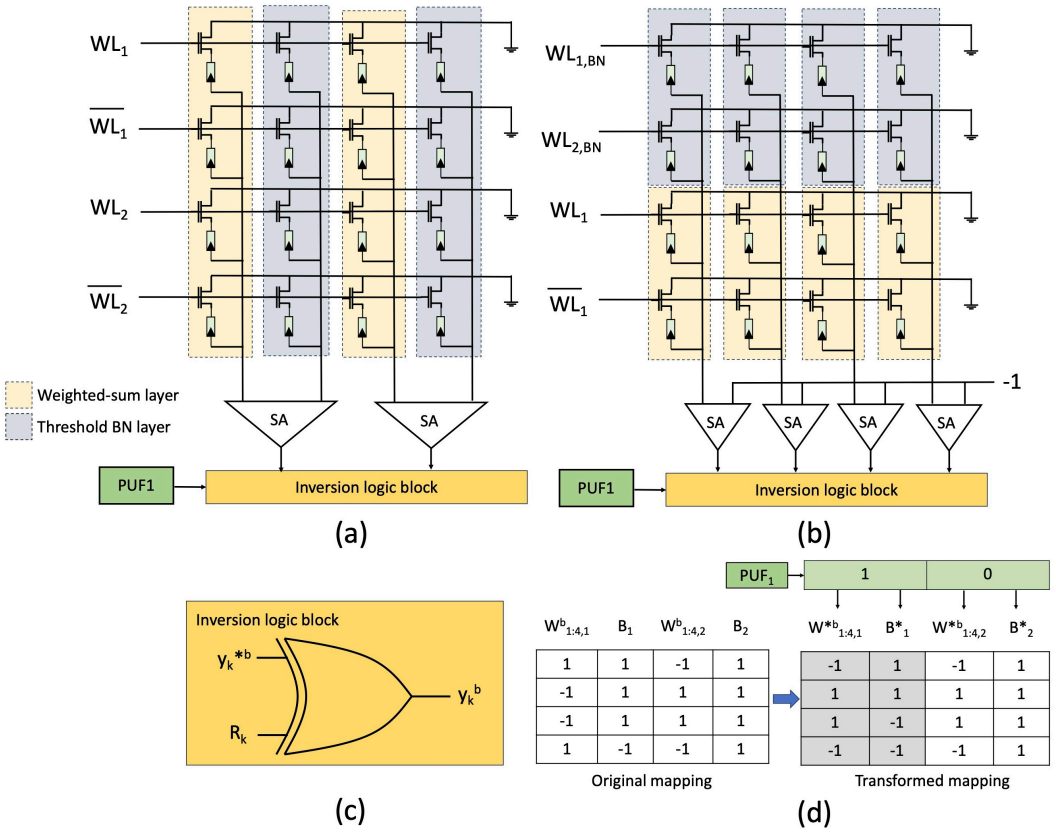


FIGURE 11.3: Layout of PUF-protected column inversion of (a)double-column and (b) single-column BN architectures, (c) inversion logic to recover the inference output and (d) illustration of transformation including BN layer

$$W_{1:l_j,k}^{*b} = \begin{cases} -W_{1:l_j,k}^{b} & \text{if } R_k = 1 \\ W_{1:l_j,k}^{b} & \text{otherwise} \end{cases} \tag{11.5}$$

$$B_k^* = \begin{cases} -B_k + 2 & \text{if } R_k = 1 \\ B_k & \text{otherwise} \end{cases} \tag{11.6}$$

$$B_k^* = \begin{cases} B_k - 2 & \text{if } R_k = 1 \\ B_k & \text{otherwise} \end{cases} \tag{11.7}$$

$$y_k^b = \begin{cases} -y_k^{*b} & \text{if } R_k = 1 \\ y_k^{*b} & \text{otherwise} \end{cases} \tag{11.8}$$

Different PUF responses are used to transform each layer. Hence, recovering the transferred layer without the actual response is tricky, improving security. We used 10 different PUF responses for each layer to analyze the impact of weight transformation and calculated the average inference accuracy. In consideration of our simulation time constraints, we have empirically chosen the value of 10 to ensure a realistic analysis by utilizing multiple secret keys rather than relying solely on a single key. Tables 11.1 and 11.2 show the classification accuracy loss with the MNIST data set when individual layers are transformed and not recovered with PUF response during inference. We have also presented the response key length required for each transformation case. The original accuracy of the model without transformation is 96.74%. When all the layers are transformed using the inversion scheme, the accuracy is reduced to 11.13% and 10.2% for row and column inversion, respectively. The impact of the transformation of the individual layers is also presented in Tables 11.1 and 11.2. We performed a power analysis for the overhead incurred because of the XOR logic. We found that the percentage increase in power is negligible and significantly less than 1%.

TABLE 11.1: Inference accuracy with row weight inversion transformation for MNIST BNN

Transformed layers	**PUF key length**	**Inference accuracy without right PUF key**
None (without PUF protection)	0	96.74%
FC 1	784	9.83%
FC 2	512	8.28%
FC 3	512	13.49%
FC 1,2,3	$784 + 2 \times 512$	11.13%

11.4.2 Transformation by swapping

A trained BNN model is valid only when all the individual weight values are used for inference in trained order. We propose transforming the weight matrix by locally swapping the values along the rows or columns based on the PUF response. We

TABLE 11.2: Inference accuracy with column weight inversion transformation for MNIST BNN

Transformed layers	**PUF key length**	**Inference accuracy without right PUF key**
None (without PUF protection)	0	96.74%
FC +BN 1	512	10.71%
FC +BN 2	512	7.35%
FC +BN 3	512	13.8%
FC + BN 1,2,3	3×512	10.2%

discuss transforming with column swapping here; the same procedure applies to row swapping. To accomplish swapping with BN-integrated architectures, we need to swap the column values of the weight matrix together with the BN layer values as given in equations (11.9) and (11.10). The new $W^{*b}_{j,k}$ and B^{*}_{k} are mapped to the crossbar. The layout of such implementation is given in Fig.11.4. Based on the PUF response R_k, the output y^{*b}_{k} is swapped during inference runtime. The required swapping logic block is also given in Fig.11.4. Similar to the previous section, we analysed the impact of swapping transformation on inference accuracy with PUF responses. With this method, transformation on individual layers alone does not significantly affect the overall classification accuracy. However, when all the layers are transformed, the accuracy drops to 52.96%. The key point is that this accuracy does not reflect the security strength, and the PUF response still determines the security. The accuracy drops only show the performance without the key. Three PUF keys of 256-bit length are required in this transformation, which is half the column size of the weight matrix. Hence, the security against brute force attack is 2^{256} for the individual layer, with the total security being $2^{3\times256}$. Similar to the inversion scheme, we estimated power for the overhead incurred by the MUX. We found that the percentage increase in power is insignificant and well below 1%.

$$R_k = \begin{cases} 1 & \text{then } W^{*b}_{j,k} = W^{b}_{j,k+1} \; and \; W^{*b}_{j,k+1} = W^{b}_{j,k} \\ 0, & \text{then } W^{*b}_{j,k} = W^{b}_{j,k} \; and \; W^{*b}_{j,k+1} = W^{b}_{j,k+1} \end{cases} \tag{11.9}$$

$$R_k = \begin{cases} 1 & \text{then } B^{*}_{k} = B_{k+1} \; and \; B^{*}_{k+1} = B_{k} \\ 0, & \text{then } B^{*}_{k} = B_{k} \; and \; B^{*}_{k+1} = B_{k+1} \end{cases} \tag{11.10}$$

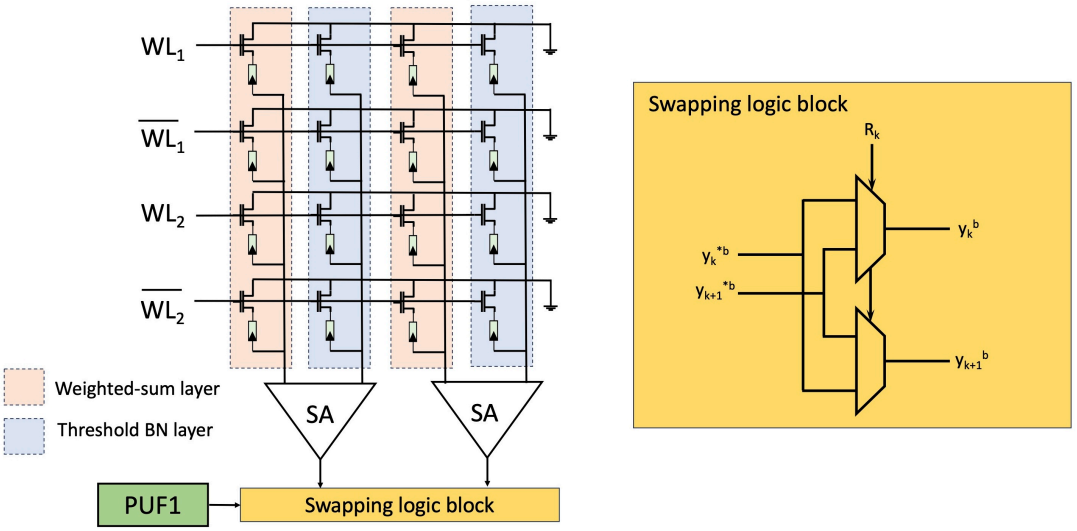


FIGURE 11.4: Layout of PUF-protected swapping scheme with its logic block

TABLE 11.3: Inference accuracy with swapped columns

Transformed layers	**PUF key length**	**Inference accuracy without right PUF key**
None (without PUF protection)	0	96.74%
FC +BN 1	256	79.79%
FC +BN 2	256	88.4%
FC +BN 3	256	94.57%
FC + BN 1,2,3	3×256	52.96%

11.5 Summary

We presented two methods with their hardware design and analysis to protect BNN in RRAM crossbars using PUF. The security strength of the stolen transformed weight matrix $W^{*b}_{j,k}$ against brute force attacks for both schemes is directly related to the length of the response key. The response key length required in the swapping scheme is half that required in the inversion scheme for the same weight matrix. Hence, the inversion scheme is more secure in terms of key length. However, this difference in key length in the two schemes may not be significant if we transform the larger matrix. For example, the key length of (3×256) discussed in this paper for column swapping can provide increased security against brute force attacks. Hence, the swapping scheme is also robust. We conclude that the choice of the

scheme is based on the required security strength and the availability of the PUF response length. In situations where the secret key generated through PUF is not of adequate length for transforming the weight matrix by the inversion scheme, the designer may choose to use the swapping scheme. Furthermore, to achieve a better trade-off between security strength, the inversion scheme may be preferred to transform $W^b_{j,k}$ with a smaller dimension, and a swapping scheme may be preferred to transform $W^b_{j,k}$ with a larger dimension. The proposed row inversion and swapping techniques can also be extended to RRAM-based multi-bit DNN implementations, which can be future work.

Chapter 12

Conclusion and Future works

The recent advancements in computing and communication technologies have led to the emergence of the IoT paradigm, driving the development of energy-efficient, high computational capacity devices integrated with intelligent algorithms. The commercialization of these computational units requires substantial financial investments and collaboration among vendors to protect their assets from reverse engineering. As a response to the diverse user base's demand for robust security measures, current research efforts are focused on the development of TEE and HROT to safeguard critical data from various threats, including side and covert channel attacks, fault attacks, hardware trojans, and unintentional design flaws.

Notably, NIST's selection of CRYSTALS-Kyber as the exclusive candidate for KEM in the PQC standardization process reflects its robust theoretical security assurances and implementation performance. However, concerns about the susceptibility of Kyber to physical attacks, such as SCA, have prompted the cryptographic community to scrutinize the implementation details and develop protection strategies.

Our research focused on investigating side-channel attacks on PQC in hardware, introducing an new approach of parallel PC oracle attacks tailored for LWE-based KEMs, specifically targeting the Kyber KEM. While our attacks are demonstrated on the Kyber KEM, we believe that our methods can be adapted to work with other LWE/LWR-based KEMs, such as Saber and FrodoKEM. This research aimed to address potential security vulnerabilities arising from the use of commercial hardware in the implementation of cryptographic schemes.

The current methods for authenticating devices and transferring data heavily rely on software-based cryptographic schemes, but these measures require significant

system resources, making them impractical for resource-constrained IoT devices. Additionally, their dependence makes them vulnerable to persistent attackers with substantial computational capabilities. Therefore, there is a growing need for dependable hardware security solutions. The HROT plays a crucial role in generating and maintaining critical cryptographic keys and is designed to resist tampering attempts. Key components of the Root of Trust include hardware security primitives such as PUF and TRNG.

We have thoroughly reviewed the recent progress in non-volatile memory (NVM) device technologies, with a specific focus on RRAM. Our review aimed to explore how advancements in RRAM technology can be used to create hardware security primitives like PUF and RNG. The research on RRAM has actively looked into its potential applications in storage, in-memory multivalued logic processing, neuromorphic computing, and security. The variability in RRAM array structures has been identified as a significant challenge, and ongoing efforts are being made to address stochastic effects. Interestingly, we can use these stochastic effects to design hardware security primitives. We proposed new designs for PUF and RNG based on memristive crossbars with a focus on improving performance and resolving critical issues in hardware security, including writing-free reconfiguration and unified design.

DNNs are commonly used in modern applications, but due to the sensitive nature of the training data and the substantial computational resources required for developing high-precision models, they are considered a form of proprietary intellectual property that needs protection from unauthorized access. BNNs, where both the weights and activations are restricted to values of +1 and -1, have garnered attention for edge computing applications due to their reduced computational complexities and energy conservation during inference. Accelerating BNNs with emerging NVM devices, particularly RRAM, is an active area of research aimed at enhancing inference performance. Despite the challenge of non-ideal RRAM behavior hindering multi-bit computing, BNNs require only two RRAM states, making them more resistant to variability. We have proposed the deployment of newly developed cryptomodules to safeguard BNN model parameters when deployed in state-of-the-art RRAM in-memory computing accelerators to mitigate potential security threats.

In summary, we addressed the evolving challenges in hardware security. The higher-level overview of contributions can be categorized into three parts. Firstly, the study introduced novel adaptations of side-channel-assisted binary PC oracle-based

attacks, significantly improving the efficiency of extracting information about the secret key. The proposed attacks demonstrate superiority in terms of the required number of queries to retrieve the secret key compared to existing methods, as validated through experimental analysis. Secondly, the research explored the latest advancements in device technologies, such as RRAM, for developing hardware security primitives like PUF and RNG. The study introduced new constructions for PUF and RNG based on memristive crossbars, addressing critical issues in hardware security primitives. Lastly, we leveraged the newly developed crypto-modules to protect BNN model parameters when deployed in advanced RRAM in-memory computing accelerators. These contributions are significant steps towards advancing the field of hardware security and have the potential to impact future developments in secure hardware systems.

Consider the following open challenges presented for future research.

- The thesis primarily focused on the investigation of SCA on lattice-based PQC KEMs. However, it's important to note that other PQC KEMs, such as Classic McEliece and HQC, are currently under consideration for standardization and require thorough investigation into their susceptibility to SCA. Furthermore, the extent of SCA leakage is reliant on the specifics of the hardware being used. Consequently, there is a crucial need for a tool capable of evaluating the vulnerability of various types of hardware to SCA in the context of PQC.

- In this study, we have introduced new PUF designs utilizing RRAM crossbars. We have tackled key challenges such as enabling reconfiguration without the need for writing, and creating a unified design with a RNG. Moving forward, we suggest integrating this cryptographic module to develop a comprehensive HROT and evaluating its performance across various applications.

- Our analysis centered on safeguarding the BNN model parameters within RRAM crossbars using PUF. The techniques we introduced are adaptable and can be utilized in other implementations, including STT/SOT-MRAM crossbars or FPGAs, which may be potential areas for future exploration.

List of Publications

- Rajendran, G., Ravi, P., D'anvers, J. P., Bhasin, S., Chattopadhyay, A. (2023). Pushing the limits of generic side-channel attacks on LWE-based KEMs-parallel PC oracle attacks on Kyber KEM and beyond. IACR Transactions on Cryptographic Hardware and Embedded Systems.

- Rajendran, G., Banerjee, W., Chattopadhyay, A., Aly, M. M. S. (2021). Application of resistive random access memory in hardware security: A review. Advanced Electronic Materials, 7(12), 2100536.

- Rajendran, G., Zahoor, F., Singh, S., Merchant, F., Rana, V., Chattopadhyay, A. (2023, October). PR-PUF: A Reconfigurable Strong RRAM PUF. In 2023 IFIP/IEEE 31st International Conference on Very Large Scale Integration (VLSI-SoC) (pp. 1-6). IEEE.

- Rajendran, G., Zahoor, F., Thakker, S. S., Singh, S., Merchant, F., Rana, V., Chattopadhyay, A. (2024, January). Harnessing Entropy: RRAM Crossbar-based Unified PUF and RNG. In 2024 37th International Conference on VLSI Design and 2024 23rd International Conference on Embedded Systems (VLSID) (pp. 560-564). IEEE.

- Rajendran, G., Basak, D., Deb, S., Chattopadhyay, A. (2024, July). Securing Binarized Neural Networks via PUF-based Key Management in Memristive Crossbar Arrays. IEEE Embedded Systems Letters (ESL).

- Singh, S., Zahoor, F., Rajendran, G., Patkar, S., Chattopadhyay, A., Merchant, F. (2023, January). Hardware security primitives using passive rram crossbar array: Novel trng and puf designs. In Proceedings of the 28th Asia and South Pacific Design Automation Conference (pp. 449-454).

- Singh, S., Zahoor, F., Rajendran, G., Rana, V., Patkar, S., Chattopadhyay, A., Merchant, F. (2023, June). Integrated architecture for neural networks and security primitives using rram crossbar. In 2023 21st IEEE Interregional NEWCAS Conference (NEWCAS) (pp. 1-5). IEEE.

The superscript * indicates joint first authors

Bibliography

[1] Yutaro Tanaka, Rei Ueno, Keita Xagawa, Akira Ito, Junko Takahashi, and Naofumi Homma. Multiple-valued plaintext-checking side-channel attacks on post-quantum kems. Cryptology ePrint Archive, Paper 2022/940, 2022. URL `https://eprint.iacr.org/2022/940`. `https://eprint.iacr.org/2022/940`. x, 44, 45, 46

[2] Writam Banerjee. Challenges and applications of emerging nonvolatile memory devices. *Electronics*, 9(6):1029, 2020. xiii, 54, 55, 58

[3] Writam Banerjee, Seong Hun Kim, Seungwoo Lee, Donghwa Lee, and Hyunsang Hwang. An efficient approach based on tuned nanoionics to maximize memory characteristics in ag-based devices. *Advanced Electronic Materials*, 7(4):2100022, 2021. xiii, 58, 60

[4] Xiaolong Zhao, Jun Ma, Xiangheng Xiao, Qi Liu, Lin Shao, Di Chen, Sen Liu, Jiebin Niu, Xumeng Zhang, Yan Wang, et al. Breaking the current-retention dilemma in cation-based resistive switching devices utilizing graphene with controlled defects. *Advanced materials*, 30(14):1705193, 2018. xiv, 61, 62

[5] Anupam Giri, Manish Kumar, Jaeseon Kim, Monalisa Pal, Writam Banerjee, Revannath Dnyandeo Nikam, Junghyeok Kwak, Minsik Kong, Seong Hun Kim, Kaliannan Thiyagarajan, et al. Surface diffusion and epitaxial self-planarization for wafer-scale single-grain metal chalcogenide thin films. *Advanced Materials*, 33(35):2102252, 2021. xiv, 61, 62, 63

[6] Aurélien Francillon and Claude Castelluccia. Tinyrng: A cryptographic random number generator for wireless sensors network nodes. In *2007 5th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks and Workshops*, pages 1–7. IEEE, 2007. xiv, 66, 67

[7] Marko Wolf and Timo Gendrullis. Design, implementation, and evaluation of a vehicular hardware security module. In *Information Security and Cryptology-ICISC 2011: 14th International Conference, Seoul, Korea, November 30-December 2, 2011. Revised Selected Papers 14*, pages 302–318. Springer, 2012. xiv, 68

[8] Meltem Sönmez Turan, Elaine Barker, John Kelsey, Kerry A McKay, Mary L Baish, Mike Boyle, et al. Recommendation for the entropy sources used for

random bit generation. *NIST Special Publication*, 800(90B):102, 2018. xiv, 68, 69

[9] Z Wei, Y Katoh, S Ogasahara, Y Yoshimoto, K Kawai, Y Ikeda, K Eriguchi, K Ohmori, and S Yoneda. True random number generator using current difference based on a fractional stochastic model in 40-nm embedded reram. In *2016 IEEE International Electron Devices Meeting (IEDM)*, pages 4–8. IEEE, 2016. xiv, 69, 70, 72

[10] Bohan Lin, Bin Gao, Yachuan Pang, Peng Yao, Dong Wu, Hu He, Jianshi Tang, He Qian, and Huaqiang Wu. A high-speed and high-reliability trng based on analog rram for iot security application. In *2019 IEEE International Electron Devices Meeting (IEDM)*, pages 14–8. IEEE, 2019. xiv, 71, 72

[11] Yan Yang, Guoqiang Bai, and Hongyi Chen. A 200mbps random number generator with jitter-amplified oscillator. In *Fifth International Conference on Computing, Communications and Networking Technologies (ICCCNT)*, pages 1–5. IEEE, 2014. xiv, 73, 75

[12] Sanu K Mathew, Suresh Srinivasan, Mark A Anders, Himanshu Kaul, Steven K Hsu, Farhana Sheikh, Amit Agarwal, Sudhir Satpathy, and Ram K Krishnamurthy. 2.4 gbps, 7 mw all-digital pvt-variation tolerant true random number generator for 45 nm cmos high-performance microprocessors. *IEEE Journal of Solid-State Circuits*, 47(11):2807–2821, 2012. xiv, 74, 75

[13] Muhammad Naveed Aman, Kee Chaing Chua, and Biplab Sikdar. Mutual authentication in iot systems using physical unclonable functions. *IEEE Internet of Things Journal*, 4(5):1327–1340, 2017. xiv, 78, 79

[14] Sudhir K Satpathy, Sanu K Mathew, Raghavan Kumar, Vikram Suresh, Mark A Anders, Himanshu Kaul, Amit Agarwal, Steven Hsu, Ram K Krishnamurthy, and Vivek De. An all-digital unified physically unclonable function and true random number generator featuring self-calibrating hierarchical von neumann extraction in 14-nm tri-gate cmos. *IEEE Journal of Solid-State Circuits*, 54(4):1074–1085, 2019. xiv, 79, 80

[15] Le Zhang, Xuanyao Fong, Chip-Hong Chang, Zhi Hui Kong, and Kaushik Roy. Feasibility study of emerging non-volatilememory based physical unclonable functions. In *2014 IEEE 6th international memory workshop (IMW)*, pages 1–4. IEEE, 2014. xiv, 82, 83

[16] Rui Liu, Pai-Yu Chen, Xiaochen Peng, and Shimeng Yu. X-point puf: Exploiting sneak paths for a strong physical unclonable function design. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 65(10):3459–3468, 2018. xiv, 82, 84

[17] Abhranil Maiti and Patrick Schaumont. Improved ring oscillator puf: An fpga-friendly secure primitive. *Journal of cryptology*, 24:375–397, 2011. xiv, 88

[18] Kota Fruhashi, Mitsuru Shiozaki, Akitaka Fukushima, Takahiko Murayama, and Takeshi Fujino. The arbiter-puf with high uniqueness utilizing novel arbiter circuit with delay-time measurement. In *2011 IEEE international symposium of circuits and systems (ISCAS)*, pages 2325–2328. IEEE, 2011. xiv, 88

[19] Shiqiang Zhu, Ting Yu, Tao Xu, Hongyang Chen, Schahram Dustdar, Sylvain Gigan, Deniz Gunduz, Ekram Hossain, Yaochu Jin, Feng Lin, et al. Intelligent computing: The latest advances, challenges, and future. *Intelligent Computing*, 2:0006, 2023. 1

[20] Wei Hu, Chip-Hong Chang, Anirban Sengupta, Swarup Bhunia, Ryan Kastner, and Hai Li. An overview of hardware security and trust: Threats, countermeasures, and design tools. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 40(6):1010–1038, 2020. 1, 2

[21] Mohamed Sabt, Mohammed Achemlal, and Abdelmadjid Bouabdallah. Trusted execution environment: What it is, and what it is not. In *2015 IEEE Trustcom/BigDataSE/Ispa*, volume 1, pages 57–64. IEEE, 2015. 1

[22] Hardware Root of Trust | Security IP | Silicon IP | Synopsys — synopsys.com. `https://www.synopsys.com/designware-ip/technical-bulletin/understanding-hardware-roots-of-trust-2017q4.html`. [Accessed 31-05-2024]. 2

[23] Paolo Prinetto, Gianluca Roascio, et al. Hardware security, vulnerabilities, and attacks: A comprehensive taxonomy. In *ITASEC*, pages 177–189, 2020. 2

[24] Roberto Avanzi, Joppe W. Bos, Leo Ducas, Eike Kiltz, Tancrede Lepoint, Vadim Lyubashevsky, John Schanck, Peter Schwabe, Gregor Seiler, and Damien Stehlé. CRYSTALS-Kyber (version 3.02): Algorithm specifications and supporting documentation (Auguest 4, 2021). 2021. 2, 10, 14

[25] Gorjan Alagic, Daniel Apon, David Cooper, Quynh Dang, Thinh Dang, John Kelsey, Jacob Lichtinger, Carl Miller, Dustin Moody, Rene Peralta, et al. Status report on the third round of the NIST post-quantum cryptography standardization process. Technical report, National Institute of Standards and Technology, 2022. 2, 10

[26] Gorjan Alagic, Jacob Alperin-Sheriff, Daniel Apon, David Cooper, Quynh Dang, John Kelsey, Yi-Kai Liu, Carl Miller, Dustin Moody, Rene Peralta, et al. Status report on the second round of the NIST post-quantum cryptography standardization process. *US Department of Commerce, NIST*, 2020. 2, 10

[27] Prasanna Ravi, Shivam Bhasin, Sujoy Sinha Roy, and Anupam Chattopadhyay. On exploiting message leakage in (few) nist pqc candidates for practical message recovery attacks. *IEEE Transactions on Information Forensics and Security*, 2021. 2, 10, 11, 18, 19, 20, 26, 43

[28] Prasanna Ravi, Sujoy Sinha Roy, Anupam Chattopadhyay, and Shivam Bhasin. Generic side-channel attacks on cca-secure lattice-based pke and kems. *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, 2020(3):307–335, 2020. 11, 16, 17, 19, 21, 22, 23, 25, 31, 32, 37, 46

[29] Shivam Bhasin, Jan-Pieter D'Anvers, Daniel Heinz, Thomas Pöppelmann, and Michiel van Beirendonck. Attacking and defending masked polynomial comparison for lattice-based cryptography. 2021(3):334–359, 2021. doi: 10.46586/tches.v2021.i3.334-359. URL `https://tches.iacr.org/index.php/TCHES/article/view/8977`. 2, 10, 17

[30] Daniel Heinz, Matthias J Kannwischer, Georg Land, Thomas Pöppelmann, Peter Schwabe, and Daan Sprenkels. First-order masked kyber on arm cortex-m4. *Cryptology ePrint Archive*, 2022. 2, 10, 19

[31] Joppe W Bos, Marc Gourjon, Joost Renes, Tobias Schneider, and Christine van Vredendaal. Masking kyber: First-and higher-order implementations. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pages 173–214, 2021. 2, 10, 19, 43

[32] Gokulnath Rajendran et al. Application of resistive random access memory in hardware security: A review. *Advanced Electronic Materials*, 7(12):2100536, 2021. 3, 73, 76, 90, 97, 98, 101, 123, 124

[33] Xiaoyu Sun et al. Xnor-rram: A scalable and parallel resistive synaptic architecture for binary neural networks. In *2018 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pages 1423–1428. IEEE, 2018. 4, 123

[34] Hyungjun Kim et al. and Kim. In-memory batch-normalization for resistive memory based binary neural network hardware. In *Proceedings of the 24th Asia and South Pacific Design Automation Conference*, pages 645–650, 2019. 4, 123, 124, 125

[35] Matthias J. Kannwischer, Joost Rijneveld, Peter Schwabe, and Ko Stoffelen. PQM4: Post-quantum crypto library for the ARM Cortex-M4, 2019. `https://github.com/mupq/pqm4`. 4, 12, 31

[36] Kalle Ngo, Elena Dubrova, Qian Guo, and Thomas Johansson. A side-channel attack on a masked ind-cca secure saber kem implementation. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pages 676–707, 2021. 11, 18, 37

[37] Dorian Amiet, Andreas Curiger, Lukas Leuenberger, and Paul Zbinden. Defeating NewHope with a single trace. In *International Conference on Post-Quantum Cryptography*, pages 189–205. Springer, 2020. 11, 16, 18, 43

[38] Rei Ueno, Keita Xagawa, Yutaro Tanaka, Akira Ito, Junko Takahashi, and Naofumi Homma. Curse of re-encryption: A generic power/em analysis on

post-quantum kems. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pages 296–322, 2022. 11, 46

[39] Muyan Shen, Chi Cheng, Xiaohan Zhang, Qian Guo, and Tao Jiang. Find the bad apples: An efficient method for perfect key recovery under imperfect sca oracles–a case study of kyber. *Cryptology ePrint Archive*, 2022. 11, 37

[40] Jan-Pieter D'Anvers, Angshuman Karmakar, Sujoy Sinha Roy, Frederik Vercauteren, Jose Maria Bermudo Mera, Michiel Van Beirendonck, and Andrea Basso. Saber. *NIST Round 3 Submissions*, 2020. 11, 46

[41] Erdem Alkim, Joppe W. Bos, Leo Ducas, Patrick Longa, Ilya Mironov, Michael Naehrig, Valeria Nikolaenko, Chris Peikert, Ananth Raghunathan, and Douglas Stebila. Frodo : Algorithm Specifications And Supporting Documentation (June 4, 2021). *Submission to the NIST post-quantum project*, 2020. 11, 46

[42] Eiichiro Fujisaki and Tatsuaki Okamoto. Secure integration of asymmetric and symmetric encryption schemes. In *Annual international cryptology conference*, pages 537–554. Springer, 1999. 14

[43] Zhuang Xu, Owen Michael Pemberton, Sujoy Sinha Roy, David Oswald, Wang Yao, and Zhiming Zheng. Magnifying side-channel leakage of lattice-based cryptosystems with chosen ciphertexts: The case study of kyber. *IEEE Transactions on Computers*, 2021. 16, 18, 26, 43

[44] Robert Primas, Peter Pessl, and Stefan Mangard. Single-trace side-channel attacks on masked lattice-based encryption. In *International Conference on Cryptographic Hardware and Embedded Systems*, pages 513–533. Springer, 2017. 16, 18

[45] Jan-Pieter D'Anvers, Marcel Tiepelt, Frederik Vercauteren, and Ingrid Verbauwhede. Timing attacks on error correcting codes in post-quantum schemes. In *Proceedings of ACM Workshop on Theory of Implementation Security Workshop*, pages 2–9, 2019. 16, 17

[46] Qian Guo, Thomas Johansson, and Alexander Nilsson. A key-recovery timing attack on post-quantum primitives using the fujisaki-okamoto transformation and its application on frodokem. In *Annual International Cryptology Conference*, pages 359–386. Springer, 2020. 16, 17

[47] Jan-Pieter D'Anvers, Daniel Heinz, Peter Pessl, Michiel Van Beirendonck, and Ingrid Verbauwhede. Higher-order masked ciphertext comparison for lattice-based cryptography. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2022(2):115–139, Feb. 2022. doi: 10.46586/tches.v2022.i2.115-139. URL `https://tches.iacr.org/index.php/TCHES/article/view/9483`. 17

[48] Bo-Yeon Sim, Jihoon Kwon, Joohee Lee, Il-Ju Kim, Tae-Ho Lee, Jaeseung Han, Hyojin Yoon, Jihoon Cho, and Dong-Guk Han. Single-trace attacks on message encoding in lattice-based KEMs. 8:183175–183191, 2020. 18, 43

[49] Kalle Ngo, Elena Dubrova, and Thomas Johansson. Breaking masked and shuffled cca secure saber kem by power analysis. In *Proceedings of the 5th Workshop on Attacks and Solutions in Hardware Security*, pages 51–61, 2021. 18

[50] Peter Pessl and Robert Primas. More practical single-trace attacks on the number theoretic transform. In *International Conference on Cryptology and Information Security in Latin America*, pages 130–149. Springer, 2019. 18

[51] Mike Hamburg, Julius Hermelink, Robert Primas, Simona Samardjiska, Thomas Schamberger, Silvan Streit, Emanuele Strieder, and Christine van Vredendaal. Chosen ciphertext k-trace attacks on masked cca2 secure kyber. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pages 88–113, 2021. 18

[52] Ciprian Băetu, F Betül Durak, Loïs Huguenin-Dumittan, Abdullah Talayhan, and Serge Vaudenay. Misuse attacks on post-quantum cryptosystems. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 747–776. Springer, 2019. 23

[53] Loïs Huguenin-Dumittan and Serge Vaudenay. Classical misuse attacks on nist round 2 pqc. In *International Conference on Applied Cryptography and Network Security*, pages 208–227. Springer, 2020. 23

[54] Yue Qin, Chi Cheng, Xiaohan Zhang, Yanbin Pan, Lei Hu, and Jintai Ding. A systematic approach and analysis of key mismatch attacks on lattice-based nist candidate kems. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 92–121. Springer, 2021. 23, 25, 27, 31, 40

[55] Boonserm Kijsirikul and Nitiwut Ussivakul. Multiclass support vector machines using adaptive directed acyclic graph. In *Proceedings of the 2002 International Joint Conference on Neural Networks. IJCNN'02 (Cat. No. 02CH37290)*, volume 1, pages 980–985. IEEE, 2002. 35

[56] Dana Dachman-Soled, Léo Ducas, Huijing Gong, and Mélissa Rossi. Lwe with side information: attacks and concrete security estimation. In *Annual International Cryptology Conference*, pages 329–358. Springer, 2020. 39

[57] Keita Xagawa, Akira Ito, Rei Ueno, Junko Takahashi, and Naofumi Homma. Fault-injection attacks against nist's post-quantum cryptography round 3 kem candidates. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 33–61. Springer, 2021. 45

[58] Vadim Lyubashevsky, Chris Peikert, and Oded Regev. On ideal lattices and learning with errors over rings. In *EUROCRYPT*, pages 1–23, 2010. 46

[59] Erdem Alkim, Roberto Avanzi, Joppe W. Bos, Leo Ducas, Antonio de la Piedra, Thomas Poppelmann, Peter Schwabe, and Douglas Stebila. NewHope (Version 1.1): Algorithm Specifications And Supporting Documentation (April 10, 2020). *Submission to the NIST post-quantum project*. 46

[60] Hayo Baan, Sauvik Bhattacharya, Scott Fluhrer, Oscar Garcia-Morchon, Thijs Laarhoven, Rachel Player, Ronald Rietman, Markku-Juhani O. Saarinen, , Ludo Tolhuizen, José Luis Torre-Arce, and Zhenfei Zhang. Round5 : Algorithm Specifications And Supporting Documentation (10th April, 2020). *Submission to the NIST post-quantum project*. 46

[61] Xianhui Lu, Yamin Liu, Dingding Jia, Haiyang Xue, Jingnan He, Zhenfei Zhang, Zhe Liu, Hao Yang, Bao Li, and Kunpeng Wang. LAC: Practical Ring-LWE Based Public-Key Encryption with Byte-Level Modulus (19th Dec, 2019). 46

[62] Cong Chen, Oussama Danba, Jeffrey Hoffstein, Andreas Hülsing, Joost Rijneveld, John M Schanck, Peter Schwabe, William Whyte, and Zhenfei Zhang. NTRU: Algorithm specifications and supporting documentation (March 20, 2019). *Submission to the NIST post-quantum project*, 2019. 47

[63] Daniel J. Bernstein, Billy Bob Brumley, Ming-Shing Chen, Chitchanok Chuengsatiansup, Tanja Lange, Adrian Marotzke, Bo-Yuan Peng, Nicola Tuveri, Christine van Vredendaal, and Bo-Yin Yang. NTRU Prime: Round 3 (October 7, 2020). *Submission to the NIST post-quantum project*, 2020. 47

[64] Shuang Pi, Can Li, Hao Jiang, Weiwei Xia, Huolin Xin, J Joshua Yang, and Qiangfei Xia. Memristor crossbar arrays with 6-nm half-pitch and 2-nm critical dimension. *Nature nanotechnology*, 14(1):35–39, 2019. 53

[65] Byung Joon Choi, Antonio C Torrezan, John Paul Strachan, PG Kotula, AJ Lohn, Matthew J Marinella, Zhiyong Li, R Stanley Williams, and J Joshua Yang. High-speed and low-energy nitride memristors. *Advanced Functional Materials*, 26(29):5290–5296, 2016. 53

[66] Writam Banerjee and Hyunsang Hwang. Quantized conduction device with 6-bit storage based on electrically controllable break junctions. *Advanced Electronic Materials*, 5(12):1900744, 2019. 53, 54

[67] Mohammed A Zidan, John Paul Strachan, and Wei D Lu. The future of electronics based on memristive systems. *Nature electronics*, 1(1):22–29, 2018. 54, 58

[68] Mario Lanza, H-S Philip Wong, Eric Pop, Daniele Ielmini, Dimitri Strukov, Brian C Regan, Luca Larcher, Marco A Villena, J Joshua Yang, Ludovic Goux, et al. Recommended methods to study resistive switching devices. *Advanced Electronic Materials*, 5(1):1800143, 2019.

[69] Writam Banerjee, Qi Liu, and Hyunsang Hwang. Engineering of defects in resistive random access memory devices. *Journal of Applied Physics*, 127(5), 2020. 57

[70] Hong Wang and Xiaobing Yan. Overview of resistive random access memory (rram): Materials, filament mechanisms, performance optimization, and prospects. *physica status solidi (RRL)–Rapid Research Letters*, 13(9): 1900073, 2019.

[71] Writam Banerjee, Qi Liu, Shibing Long, Hangbing Lv, and Ming Liu. Crystal that remembers: Several ways to utilize nanocrystals in resistive switching memory. *Journal of Physics D: Applied Physics*, 50(30):303002, 2017. 54, 58

[72] Ilia Valov, Rainer Waser, John R Jameson, and Michael N Kozicki. Electrochemical metallization memories—fundamentals, applications, prospects. *Nanotechnology*, 22(25):254003, 2011. 54, 56

[73] Byung Chul Jang, Sungkyu Kim, Sang Yoon Yang, Jihun Park, Jun-Hwe Cha, Jungyeop Oh, Junhwan Choi, Sung Gap Im, Vinayak P Dravid, and Sung-Yool Choi. Polymer analog memristive synapse with atomic-scale conductive filament for flexible neuromorphic computing system. *Nano letters*, 19(2):839–849, 2019.

[74] Hangbing Lv, Xiaoxin Xu, Pengxiao Sun, Hongtao Liu, Qing Luo, Qi Liu, Writam Banerjee, Haitao Sun, Shibing Long, Ling Li, et al. Atomic view of filament growth in electrochemical memristive elements. *Scientific Reports*, 5(1):13311, 2015.

[75] Qiaoling Tian, Xiaohan Zhang, Xiaoning Zhao, Zhongqiang Wang, Ya Lin, Haiyang Xu, and Yichun Liu. Dual buffer layers for developing electrochemical metallization memory with low current and high endurance. *IEEE Electron Device Letters*, 42(3):308–311, 2020.

[76] Haitao Sun, Qi Liu, Shibing Long, Hangbing Lv, Writam Banerjee, and Ming Liu. Multilevel unipolar resistive switching with negative differential resistance effect in ag/sio2/pt device. *Journal of Applied Physics*, 116(15), 2014.

[77] Fekadu Gochole Aga, Jiyong Woo, Jeonghwan Song, Jaehyuk Park, Seokjae Lim, Changhyuck Sung, and Hyunsang Hwang. Controllable quantized conductance for multilevel data storage applications using conductive bridge random access memory. *Nanotechnology*, 28(11):115707, 2017. 54

[78] C Chen, S Gao, F Zeng, GY Wang, SZ Li, C Song, and F Pan. Conductance quantization in oxygen-anion-migration-based resistive switching memory devices. *Applied Physics Letters*, 103(4), 2013. 54

[79] Chengqing Hu, Martin D McDaniel, Agham Posadas, Alexander A Demkov, John G Ekerdt, and Edward T Yu. Highly controllable and stable quantized conductance and resistive switching mechanism in single-crystal tio2 resistive memory on silicon. *Nano letters*, 14(8):4360–4367, 2014.

[80] Naga Raghavan, Andrea Fantini, Robin Degraeve, PJ Roussel, Ludovic Goux, B Govoreanu, DJ Wouters, G Groeseneken, and M Jurczak. Statistical insight into controlled forming and forming free stacks for hfox rram. *Microelectronic Engineering*, 109:177–181, 2013.

[81] Writam Banerjee, Wu Fa Cai, Xiaolong Zhao, Qi Liu, Hangbing Lv, Shibing Long, and Ming Liu. Intrinsic anionic rearrangement by extrinsic control: Transition of rs and crs in thermally elevated tin/hfo 2/pt rram. *Nanoscale*, 9(47):18908–18917, 2017.

[82] Spyros Stathopoulos, Ali Khiat, Maria Trapatseli, Simone Cortese, Alexantrou Serb, Ilia Valov, and Themis Prodromakis. Multibit memory operation of metal-oxide bi-layer memristors. *Scientific reports*, 7(1):17532, 2017.

[83] Sk Ziaur Rahaman, Yu-De Lin, Heng-Yuan Lee, Yu-Sheng Chen, Pang-Shiu Chen, Wei-Su Chen, Chien-Hua Hsu, Kan-Hsueh Tsai, Ming-Jinn Tsai, and Pei-Hua Wang. The role of ti buffer layer thickness on the resistive switching properties of hafnium oxide-based resistive switching memories. *Langmuir*, 33(19):4654–4665, 2017.

[84] Writam Banerjee, Xumeng Zhang, Qing Luo, Hangbing Lv, Qi Liu, Shibing Long, and Ming Liu. Design of cmos compatible, high-speed, highly-stable complementary switching with multilevel operation in 3d vertically stacked novel hfo2/al2o3/tiox (hat) rram. *Advanced Electronic Materials*, 4 (2):1700561, 2018.

[85] Writam Banerjee, Xiaoxin Xu, Hangbing Lv, Qi Liu, Shibing Long, and Ming Liu. Complementary switching in 3d resistive memory array. *Advanced Electronic Materials*, 3(12):1700287, 2017.

[86] Siddheswar Maikap and Writam Banerjee. In quest of nonfilamentary switching: a synergistic approach of dual nanostructure engineering to improve the variability and reliability of resistive random-access-memory devices. *Advanced Electronic Materials*, 6(6):2000209, 2020. 58

[87] Writam Banerjee, Xiaoxin Xu, Hangbing Lv, Qi Liu, Shibing Long, and Ming Liu. Variability improvement of tio x/al2o3 bilayer nonvolatile resistive switching devices by interfacial band engineering with an ultrathin al2o3 dielectric material. *ACS omega*, 2(10):6888–6895, 2017.

[88] Writam Banerjee, Qi Liu, Hangbing Lv, Shibing Long, and Ming Liu. Electronic imitation of behavioral and psychological synaptic activities using tio x/al 2 o 3-based memristor devices. *Nanoscale*, 9(38):14442–14450, 2017. 54

[89] Stephan Menzel, Stefan Tappertzhofen, Rainer Waser, and Ilia Valov. Switching kinetics of electrochemical metallization memory cells. *Physical Chemistry Chemical Physics*, 15(18):6945–6952, 2013. 54

[90] Michael N Kozicki, Maria Mitkova, and Ilia Valov. Electrochemical metallization memories. *Resistive switching: from fundamentals of nanoionic redox processes to memristive device applications*, pages 483–514, 2016. 54

[91] Fei Zhuge, Kang Li, Bing Fu, Hongliang Zhang, Jun Li, Hao Chen, Lingyan Liang, Junhua Gao, Hongtao Cao, Zhimin Liu, et al. Mechanism for resistive switching in chalcogenide-based electrochemical metallization memory cells. *AIP Advances*, 5(5), 2015. 54

[92] Stephan Menzel and Rainer Waser. Analytical analysis of the generic set and reset characteristics of electrochemical metallization memory cells. *Nanoscale*, 5(22):11003–11010, 2013. 54

[93] Sven Dirkmann and Thomas Mussenbrock. Resistive switching in memristive electrochemical metallization devices. *AIP Advances*, 7(6), 2017. 54

[94] Xiaoning Zhao, Mengyao Li, Haiyang Xu, Zhongqiang Wang, Cen Zhang, Weizhen Liu, Jiangang Ma, and Yichun Liu. Forming-free electrochemical metallization resistive memory devices based on nanoporous tioxny thin film. *Journal of Alloys and Compounds*, 656:612–617, 2016. 54

[95] JQ Huang, LP Shi, EG Yeo, KJ Yi, and R Zhao. *IEEE electron device letters*, 33(1):98–100, 2011. 54

[96] Christopher Pearson, Leon Bowen, Myung-Won Lee, Alison L Fisher, Katharine E Linton, Martin R Bryce, and Michael C Petty. Focused ion beam and field-emission microscopy of metallic filaments in memory devices based on thin films of an ambipolar organic compound consisting of oxadiazole, carbazole, and fluorene units. *Applied physics letters*, 102(21), 2013. 55

[97] Chung-Nan Peng, Chun-Wen Wang, Tsung-Cheng Chan, Wen-Yuan Chang, Yi-Chung Wang, Hung-Wei Tsai, Wen-Wei Wu, Lih-Juann Chen, and Yu-Lun Chueh. Resistive switching of au/zno/au resistive memory: an in situ observation of conductive bridge formation. *Nanoscale research letters*, 7: 1–6, 2012.

[98] Zheng Wang, Peter B Griffin, Jim McVittie, Simon Wong, Paul C McIntyre, and Yoshio Nishi. *IEEE electron device letters*, 28(1):14–16, 2006. 55

[99] Michael Lübben and Ilia Valov. Active electrode redox reactions and device behavior in ecm type resistive switching memories. *Advanced Electronic Materials*, 5(9):1800933, 2019. 55

[100] Stefan Tappertzhofen, Rainer Waser, and Ilia Valov. Impact of the counter-electrode material on redox processes in resistive switching memories. *ChemElectroChem*, 1(8):1287–1292, 2014. 56

[101] Jan van den Hurk, Ann-Christin Dippel, Deok-Yong Cho, Joshua Straquadine, Uwe Breuer, Peter Walter, Rainer Waser, and Ilia Valov. Physical origins and suppression of ag dissolution in ges x-based ecm cells. *Physical Chemistry Chemical Physics*, 16(34):18217–18225, 2014. 56

[102] Jan Van Den Hurk, Ilia Valov, and Rainer Waser. Preparation and characterization of gesx thin-films for resistive switching memories. *Thin Solid Films*, 527:299–302, 2013. 56

[103] SZ Rahaman, S Maikap, H-C Chiu, C-H Lin, T-Y Wu, Y-S Chen, P-J Tzeng, F Chen, M-J Kao, and M-J Tsai. Bipolar resistive switching memory using cu metallic filament in ge0. 4se0. 6 solid electrolyte. *Electrochemical and Solid-State Letters*, 13(5):H159, 2010. 56

[104] Michael N Kozicki and Maria Mitkova. Mass transport in chalcogenide electrolyte films–materials and applications. *Journal of non-crystalline solids*, 352(6-7):567–577, 2006. 56

[105] Ilia Valov and Michael N Kozicki. Cation-based resistance change memory. *Journal of Physics D: Applied Physics*, 46(7):074005, 2013. 56

[106] Writam Banerjee and Hyunsang Hwang. Understanding of selector-less 1s1r type cu-based cbram devices by controlling sub-quantum filament. *Advanced Electronic Materials*, 6(9):2000488, 2020. 56

[107] John R Jameson and Deepak Kamalanathan. Subquantum conductive-bridge memory. *Applied Physics Letters*, 108(5), 2016. 56

[108] John R Jameson, John Dinh, Nathan Gonzales, S Hollmer, Sue Hsu, David Kim, Foroozan Koushan, Derric Lewis, Ed Runnion, Jeffrey Shields, et al. Towards automotive grade embedded rram. In *2018 48th European Solid-State Device Research Conference (ESSDERC)*, pages 58–61. IEEE, 2018. 56

[109] Hao Jiang, Daniel Belkin, Sergey E Savel'ev, Siyan Lin, Zhongrui Wang, Yunning Li, Saumil Joshi, Rivu Midya, Can Li, Mingyi Rao, et al. A novel true random number generator based on a stochastic diffusive memristor. *Nature communications*, 8(1):882, 2017. 56, 72

[110] Bingjie Dang, Jing Sun, Teng Zhang, Saisai Wang, Mo Zhao, Keqin Liu, Liying Xu, Jiadi Zhu, Caidie Cheng, Lin Bao, et al. Physically transient true random number generators based on paired threshold switches enabling monte carlo method applications. *IEEE Electron Device Letters*, 40(7):1096–1099, 2019. 56

[111] Stephan Menzel and Rainer Waser. Mechanism of memristive switching in oxram. In *Advances in Non-Volatile Memory and Storage Technology*, pages 137–170. Elsevier, 2019. 57

[112] Felix Cüppers, S Menzel, C Bengel, A Hardtdegen, M Von Witzleben, U Böttger, R Waser, and S Hoffmann-Eifert. Exploiting the switching dynamics of hfo2-based reram devices for reliable analog memristive behavior. *APL materials*, 7(9), 2019. 57

[113] Nuo Xu, Lifeng Liu, Xiao Sun, Xiaoyan Liu, Dedong Han, Yi Wang, Ruqi Han, Jinfeng Kang, and Bin Yu. Characteristics and mechanism of conduction/set process in tin/ zno/ pt resistance switching random-access memories. *Applied Physics Letters*, 92(23), 2008. 57

[114] Umberto Celano, Yang Yin Chen, Dirk J Wouters, Guido Groeseneken, Malgorzata Jurczak, and Wilfried Vandervorst. Filament observation in metal-oxide resistive switching devices. *Applied Physics Letters*, 102(12), 2013. 57

[115] Elisa Vianello, Philippe Blaise, Boubacar Traoré, Kanhao Xue, Leonardo Fonseca, Gabriel Molas, Barbara de Salvo, Luca Perniola, and Yoshio Nishi. Investigation of frenkel-pair formation in hfo2 and its influence on oxram memory reliability. *ECS Transactions*, 64(8):141, 2014. 57

[116] BJ Choi, Doo Seok Jeong, SK Kim, C Rohde, S Choi, J Hl Oh, HJ Kim, CS Hwang, K Szot, R Waser, et al. Resistive switching mechanism of tio2 thin films grown by atomic-layer deposition. *Journal of applied physics*, 98 (3), 2005. 57

[117] Shimeng Yu, Yi Wu, Yang Chai, J Provine, and H-S Philip Wong. Characterization of switching parameters and multilevel capability in hfo x/alo x bi-layer rram devices. In *Proceedings of 2011 International Symposium on VLSI Technology, Systems and Applications*, pages 1–2. IEEE, 2011. 57

[118] Jen-Chieh Liu, I-Ting Wang, Chung-Wei Hsu, Wun-Cheng Luo, and Tuo-Hung Hou. Investigating mlc variation of filamentary and non-filamentary rram. In *Proceedings of Technical Program-2014 International Symposium on VLSI Technology, Systems and Application (VLSI-TSA)*, pages 1–2. IEEE, 2014. 57

[119] H-S Philip Wong, Heng-Yuan Lee, Shimeng Yu, Yu-Sheng Chen, Yi Wu, Pang-Shiu Chen, Byoungil Lee, Frederick T Chen, and Ming-Jinn Tsai. Metal–oxide rram. *Proceedings of the IEEE*, 100(6):1951–1970, 2012. 57

[120] Ludovic Goux. Oxram technology development and performances. *Advances in Non-volatile Memory and Storage Technology*, pages 3–33, 2019. 57

[121] Sergiu Clima, Yang Yin Chen, Chao Yang Chen, Ludovic Goux, Bogdan Govoreanu, Robin Degraeve, Andrea Fantini, Malgorzata Jurczak, and Geoffrey Pourtois. First-principles thermodynamics and defect kinetics guidelines for engineering a tailored rram device. *Journal of Applied Physics*, 119(22), 2016. 57

[122] Shimeng Yu, Byoungil Lee, and H. S.Philip Wong. *Metal oxide resistive switching memory*, pages 303–335. Number 1 in Springer Series in Materials Science. 1 edition, 2012. ISBN 9781441999306. doi: 10.1007/978-1-4419-9931-3_13. 57

[123] Ludovic Goux, JG Lisoni, M Jurczak, DJ Wouters, Lorene Courtade, and Ch Muller. Coexistence of the bipolar and unipolar resistive-switching modes in nio cells made by thermal oxidation of ni layers. *Journal of Applied Physics*, 107(2), 2010. 57

[124] Ludovic Goux, Robin Degraeve, Johan Meersschaut, Bogdan Govoreanu, DJ Wouters, Stefan Kubicek, and Malgorzata Jurczak. Role of the anode material in the unipolar switching of tin\nio\ni cells. *Journal of Applied Physics*, 113(5), 2013.

[125] Hyung Dong Lee and Yoshio Nishi. Reduction in reset current of unipolar nio-based resistive switching through nickel interfacial layer. *Applied Physics Letters*, 97(25), 2010. 57

[126] Daniele Ielmini, Rainer Bruchhaus, and Rainer Waser. Thermochemical resistive switching: materials, mechanisms, and scaling projections. *Phase Transitions*, 84(7):570–602, 2011. 57

[127] Rainer Waser, Regina Dittmann, Georgi Staikov, and Kristof Szot. Redox-based resistive switching memories-nanoionic mechanisms, prospects, and challenges. *Advanced Materials (Deerfield Beach, Fla.)*, 21(25-26):2632–2663, 2009. 57

[128] Bohan Lin, Yachuan Pang, Bin Gao, Jianshi Tang, Dong Wu, Ting-Wei Chang, Wei-En Lin, Xiaoyu Sun, Shimeng Yu, Meng-Fan Chang, et al. A highly reliable rram physically unclonable function utilizing post-process randomness source. *IEEE Journal of Solid-State Circuits*, 56(5):1641–1650, 2021. 58

[129] Gwangmin Kim, Jae Hyun In, Young Seok Kim, Hakseung Rhee, Woojoon Park, Hanchan Song, Juseong Park, and Kyung Min Kim. Self-clocking fast and variation tolerant true random number generator based on a stochastic mott memristor. *Nature communications*, 12(1):2906, 2021. 58

[130] G Molas, E Vianello, F Dahmani, M Barci, P Blaise, J Guy, A Toffoli, M Bernard, A Roule, F Pierre, et al. Controlling oxygen vacancies in doped oxide based cbram for improved memory performances. In *2014 IEEE International Electron Devices Meeting*, pages 6–1. IEEE, 2014. 58

[131] Writam Banerjee, Ilya V Karpov, Ashish Agrawal, Seonghun Kim, Seungwoo Lee, Sangmin Lee, Donghwa Lee, and Hyunsang Hwang. Highly-stable ($<$ 3% fluctuation) ag-based threshold switch with extreme-low off current of 0.1 pa, extreme-high selectivity of 10 9 and high endurance of 10 9 cycles. In *2020 IEEE International Electron Devices Meeting (IEDM)*, pages 28–4. IEEE, 2020. 59

[132] Gilbert Sassine, Cecile Nail, Philippe Blaise, Benoit Sklenard, Mathieu Bernard, Rémy Gassilloud, Aurélie Marty, Marc Veillerot, Christophe Vallee, Etienne Nowak, et al. Hybrid-rram toward next generation of nonvolatile memory: coupling of oxygen vacancies and metal ions. *Advanced Electronic Materials*, 5(2):1800658, 2019. 58, 61

[133] Writam Banerjee, Seong Hun Kim, Seungwoo Lee, Sangmin Lee, Donghwa Lee, and Hyunsang Hwang. Deep insight into steep-slope threshold switching with record selectivity ($> 4\times$ 1010) controlled by metal-ion movement through vacancy-induced-percolation path: quantum-level control of hybrid-filament. *Advanced Functional Materials*, 31(37):2104054, 2021. 60

[134] Fei Hui, Enric Grustan-Gutierrez, Shibing Long, Qi Liu, Anna K Ott, Andrea C Ferrari, and Mario Lanza. Graphene and related materials for resistive random access memories. *Advanced Electronic Materials*, 3(8):1600195, 2017. 61

[135] Nicholas R Glavin, Rahul Rao, Vikas Varshney, Elisabeth Bianco, Amey Apte, Ajit Roy, Emilie Ringe, and Pulickel M Ajayan. Emerging applications of elemental 2d materials. *Advanced Materials*, 32(7):1904302, 2020.

[136] Qianlong Zhao, Zhongjian Xie, Ya-Pei Peng, Kaiyang Wang, Huide Wang, Xiangnan Li, Hongwei Wang, Jingsheng Chen, Han Zhang, and Xiaobing Yan. Current status and prospects of memristors based on novel 2d materials. *Materials Horizons*, 7(6):1495–1518, 2020.

[137] Xiang Hou, Huawei Chen, Zhenhan Zhang, Shuiyuan Wang, and Peng Zhou. 2d atomic crystals: a promising solution for next-generation data storage. *Advanced Electronic Materials*, 5(9):1800944, 2019. 61

[138] Sen Liu, Nianduan Lu, Xiaolong Zhao, Hui Xu, Writam Banerjee, Hangbing Lv, Shibing Long, Qingjiang Li, Qi Liu, and Ming Liu. Eliminating negative-set behavior by suppressing nanofilament overgrowth in cation-based memory. *Advanced Materials (Deerfield Beach, Fla.)*, 28(48):10623–10629, 2016. 61

[139] Zuheng Wu, Xiaolong Zhao, Yang Yang, Wei Wang, Xumeng Zhang, Rui Wang, Rongrong Cao, Qi Liu, and Writam Banerjee. Transformation of threshold volatile switching to quantum point contact originated nonvolatile switching in graphene interface controlled memory devices. *Nanoscale advances*, 1(9):3753–3760, 2019. 62

[140] Yuanyuan Shi, Xianhu Liang, Bin Yuan, Victoria Chen, Haitong Li, Fei Hui, Zhouchangwan Yu, Fang Yuan, Eric Pop, H-S Philip Wong, et al. Electronic synapses made of layered two-dimensional materials. *Nature Electronics*, 1 (8):458–465, 2018. 62

[141] Shaochuan Chen, Mohammad Reza Mahmoodi, Yuanyuan Shi, Chandreswar Mahata, Bin Yuan, Xianhu Liang, Chao Wen, Fei Hui, Deji Akinwande,

Dmitri B Strukov, et al. Wafer-scale integration of two-dimensional materials in high-density memristive crossbar arrays for artificial neural networks. *Nature Electronics*, 3(10):638–645, 2020.

[142] Chengbin Pan, Yanfeng Ji, Na Xiao, Fei Hui, Kechao Tang, Yuzheng Guo, Xiaoming Xie, Francesco M Puglisi, Luca Larcher, Enrique Miranda, et al. Coexistence of grain-boundaries-assisted bipolar and threshold resistive switching in multilayer hexagonal boron nitride. *Advanced functional materials*, 27 (10):1604811, 2017.

[143] Huan Zhao, Zhipeng Dong, He Tian, Don DiMarzi, Myung-Geun Han, Lihua Zhang, Xiaodong Yan, Fanxin Liu, Lang Shen, Shu-Jen Han, et al. Atomically thin femtojoule memristive device. *Advanced Materials*, 29(47):1703232, 2017. 62

[144] Francesco Maria Puglisi, Luca Larcher, C Pan, N Xiao, Y Shi, F Hui, and M Lanza. 2d h-bn based rram devices. In *2016 IEEE International Electron Devices Meeting (IEDM)*, pages 34–8. IEEE, 2016.

[145] Fei Hui, Marco A Villena, Wenjing Fang, Ang-Yu Lu, Jing Kong, Yuanyuan Shi, Xu Jing, Kaichen Zhu, and Mario Lanza. Synthesis of large-area multilayer hexagonal boron nitride sheets on iron substrates and its use in resistive switching devices. *2D Materials*, 5(3):031011, 2018. 61

[146] Feng Zhang, Huairuo Zhang, Sergiy Krylyuk, Cory A Milligan, Yuqi Zhu, Dmitry Y Zemlyanov, Leonid A Bendersky, Benjamin P Burton, Albert V Davydov, and Joerg Appenzeller. Electric-field induced structural transition in vertical mote2-and mo1–x w x te2-based resistive memories. *Nature materials*, 18(1):55–61, 2019. 61

[147] Chia-Hui Lee, Eduardo Cruz Silva, Lazaro Calderin, Minh An T Nguyen, Matthew J Hollander, Brian Bersch, Thomas E Mallouk, and Joshua A Robinson. Tungsten ditelluride: a layered semimetal. *Scientific reports*, 5 (1):10013, 2015. 61

[148] Seunghyun Lee, Joon Sohn, Zizhen Jiang, Hong-Yu Chen, and H-S Philip Wong. Metal oxide-resistive memory using graphene-edge electrodes. *Nature communications*, 6(1):8407, 2015. 61

[149] Chaoxing Wu, Fushan Li, Yongai Zhang, and Tailiang Guo. Recoverable electrical transition in a single graphene sheet for application in nonvolatile memories. *Applied Physics Letters*, 100(4), 2012. 61

[150] Revannath Dnyandeo Nikam, Krishn Gopal Rajput, and Hyunsang Hwang. Single-atom quantum-point contact switch using atomically thin hexagonal boron nitride. *Small*, 17(7):2006760, 2021. 63

[151] Mario Lanza, C Wen, X Li, T Zanotti, FM Puglisi, Y Shi, F Saiz, A Antidormi, S Roche, W Zheng, et al. Advanced data encryption using two-dimensional materials. *Adv. Mater*, 33(2100185.10):1002, 2021. 63

[152] Geoffrey W Burr, Rohit S Shenoy, Kumar Virwani, Pritish Narayanan, Alvaro Padilla, Bülent Kurdi, and Hyunsang Hwang. Access devices for 3d crosspoint memory. *Journal of Vacuum Science & Technology B*, 32(4), 2014. 63

[153] Hong-Yu Chen, Stefano Brivio, Che-Chia Chang, Jacopo Frascaroli, Tuo-Hung Hou, Boris Hudec, Ming Liu, Hangbing Lv, Gabriel Molas, Joon Sohn, et al. Resistive random access memory (rram) technology: From material, device, selector, 3d integration to bottom-up fabrication. *Journal of Electroceramics*, 39:21–38, 2017. 63

[154] Pengxiao Sun, Nianduan Lu, Ling Li, Yingtao Li, Hong Wang, Hangbing Lv, Qi Liu, Shibing Long, Su Liu, and Ming Liu. Thermal crosstalk in 3-dimensional rram crossbar array. *Scientific reports*, 5(1):13504, 2015. 64

[155] R Djenadi, G Micolau, J Postel-Pellerin, P Chiquet, R Laffont, J-L Ogier, A Regnier, F Lalande, and J Melkonian. Data retention under gate stress on a nvm array. *Solid-state electronics*, 78:80–86, 2012. 64

[156] Nagarajan Raghavan, Daniel D Frey, Michel Bosman, and Kin Leong Pey. Statistics of retention failure in the low resistance state for hafnium oxide rram using a kinetic monte carlo approach. *Microelectronics Reliability*, 55 (9-10):1422–1426, 2015. 64

[157] Xueyao Huang, Huaqiang Wu, Bin Gao, Deepak C Sekar, Lingjun Dai, Mark Kellam, Gary Bronner, Ning Deng, and He Qian. Hfo2/al2o3 multilayer for rram arrays: a technique to improve tail-bit retention. *Nanotechnology*, 27 (39):395201, 2016. 64

[158] Yang Yin Chen, Masanori Komura, Robin Degraeve, Bogdan Govoreanu, Ludovic Goux, Andrea Fantini, Naga Raghavan, Sergiu Clima, Leqi Zhang, Attilio Belmonte, et al. Improvement of data retention in hfo 2/hf 1t1r rram cell under low operating current. In *2013 IEEE International Electron Devices Meeting*, pages 10–1. Ieee, 2013. 64

[159] Myoung-Jae Lee, Chang Bum Lee, Dongsoo Lee, Seung Ryul Lee, Man Chang, Ji Hyun Hur, Young-Bae Kim, Chang-Jung Kim, David H Seo, Sunae Seo, et al. A fast, high-endurance and scalable non-volatile memory device made from asymmetric ta2o5- x/tao2- x bilayer structures. *Nature materials*, 10(8):625–630, 2011. 64

[160] B Chen, Y Lu, B Gao, YH Fu, FF Zhang, P Huang, YS Chen, LF Liu, XY Liu, JF Kang, et al. Physical mechanisms of endurance degradation in tmo-rram. In *2011 International electron devices meeting*, pages 12–3. IEEE, 2011. 64

[161] Gilbert Sassine, Diego Alfaro Robayo, Cecile Nail, Jean-Francois Nodin, Jean Coignus, Gabriel Molas, and Etienne Nowak. Optimizing programming energy for improved rram reliability for high endurance applications. In *2018 IEEE International Memory Workshop (IMW)*, pages 1–4. IEEE, 2018. 64

[162] C Nail, G Molas, P Blaise, G Piccolboni, B Sklenard, C Cagli, M Bernard, A Roule, M Azzaz, E Vianello, et al. Understanding rram endurance, retention and window margin trade-off using experimental results and simulations. In *2016 IEEE International Electron Devices Meeting (IEDM)*, pages 4–5. IEEE, 2016.

[163] P Huang, B Chen, YJ Wang, FF Zhang, L Shen, R Liu, L Zeng, G Du, X Zhang, B Gao, et al. Analytic model of endurance degradation and its practical applications for operation scheme optimization in metal oxide based rram. In *2013 IEEE International electron devices meeting*, pages 22–5. IEEE, 2013.

[164] Nagarajan Raghavan, Kin Leong Pey, Daniel D Frey, and Michel Bosman. Stochastic failure model for endurance degradation in vacancy modulated hfo x rram using the percolation cell framework. In *2014 IEEE International Reliability Physics Symposium*, pages MY–9. IEEE, 2014.

[165] Diego Alfaro Robayo, Gilbert Sassine, Quentin Rafhay, Gerard Ghibaudo, Gabriel Molas, and Etienne Nowak. Endurance statistical behavior of resistive memories based on experimental and theoretical investigation. *IEEE Transactions on Electron Devices*, 66(8):3318–3325, 2019. 64

[166] Jason P Campbell, Jin Qin, KP Cheungl, Liangchun Yu, JS Suehlel, A Oates, and Kuang Sheng. The origins of random telegraph noise in highly scaled sion nmosfets. In *2008 IEEE International Integrated Reliability Workshop Final Report*, pages 105–109. IEEE, 2008. 64

[167] Naoki Tega, Hiroshi Miki, Masanao Yamaoka, Hitoshi Kume, Toshiyuki Mine, Takeshi Ishida, Yuki Mori, Renichi Yamada, and Kazuyoshi Torii. Impact of threshold voltage fluctuation due to random telegraph noise on scaled-down sram. In *2008 IEEE International Reliability Physics Symposium*, pages 541–546. IEEE, 2008. 64

[168] Christian Monzio Compagnoni, Riccardo Gusmeroli, Alessandro S Spinelli, Andrea L Lacaita, Mauro Bonanomi, and Angelo Visconti. Statistical model for random telegraph noise in flash memories. *IEEE Transactions on electron devices*, 55(1):388–395, 2007. 64

[169] Hideaki Kurata, Kazuo Otsuga, Akira Kotabe, Shinya Kajiyama, Taro Osabe, Yoshitaka Sasago, Shunichi Narumi, Kenji Tokami, Shiro Kamohara, and Osamu Tsuchiya. Random telegraph signal in flash memory: Its impact on scaling of multilevel flash memory beyond the 90-nm node. *IEEE Journal of Solid-State Circuits*, 42(6):1362–1369, 2007. 64

[170] MJ Uren, DJ Day, and MJj Kirton. 1/f and random telegraph noise in silicon metal-oxide-semiconductor field-effect transistors. *Applied physics letters*, 47 (11):1195–1197, 1985. 65

[171] Gérard Ghibaudo, O Roux, Ch Nguyen-Duc, Francis Balestra, and J Brini. Improved analysis of low frequency noise in field-effect mos transistors. *physica status solidi (a)*, 124(2):571–581, 1991.

[172] Maurício Banaszeski da Silva, Hans Tuinhout, Adrie Zegers-van Duijnhoven, Gilson I Wirth, and Andries Scholten. A physics-based rtn variability model for mosfets. In *2014 IEEE International Electron Devices Meeting*, pages 35–2. IEEE, 2014. 65

[173] Stefano Ambrogio, Simone Balatti, Antonio Cubeta, Alessandro Calderoni, Nirmal Ramaswamy, and Daniele Ielmini. Statistical fluctuations in hfo x resistive-switching memory: Part ii—random telegraph noise. *IEEE Transactions on Electron Devices*, 61(8):2920–2927, 2014. 65

[174] Daniele Ielmini, Federico Nardi, and Carlo Cagli. Resistance-dependent amplitude of random telegraph-signal noise in resistive switching memories. *Applied Physics Letters*, 96(5), 2010.

[175] Shinhyun Choi, Yuchao Yang, and Wei Lu. Random telegraph noise and resistance switching analysis of oxide based resistive memory. *Nanoscale*, 6 (1):400–404, 2014. 65

[176] Naga Raghavan, Robin Degraeve, Andrea Fantini, Ludovic Goux, Sebastiano Strangio, Bogdan Govoreanu, DJ Wouters, Guido Groeseneken, and Malgorzata Jurczak. Microscopic origin of random telegraph noise fluctuations in aggressively scaled rram and its impact on read disturb variability. In *2013 IEEE International Reliability Physics Symposium (IRPS)*, pages 5E–3. IEEE, 2013. 65

[177] Tiancheng Gong, Qing Luo, Xiaoxin Xu, Jie Yu, Danian Dong, Hangbing Lv, Peng Yuan, Chuanbing Chen, Jiahao Yin, Lu Tai, et al. Classification of three-level random telegraph noise and its application in accurate extraction of trap profiles in oxide-based resistive switching memory. *IEEE Electron Device Letters*, 39(9):1302–1305, 2018. 65

[178] Naga Raghavan, Robin Degraeve, Ludovic Goux, Andrea Fantini, DJ Wouters, Guido Groeseneken, and Malgorzata Jurczak. Rtn insight to filamentary instability and disturb immunity in ultra-low power switching hfo x and alo x rram. In *2013 Symposium on VLSI Technology*, pages T164–T165. IEEE, 2013. 65

[179] Attilio Belmonte, Robin Degraeve, Andrea Fantini, W Kim, Michel Houssa, M Jurczak, and Ludovic Goux. Origin of the current discretization in deep reset states of an al2o3/cu-based conductive-bridging memory, and impact on state level and variability. *Applied Physics Letters*, 104(23), 2014. 65

[180] Ludovic Goux and Ilia Valov. Electrochemical processes and device improvement in conductive bridge ram cells. *physica status solidi (a)*, 213(2):274–288, 2016. 65

[181] An Chen and Ming-Ren Lin. Variability of resistive switching memories and its impact on crossbar array performance. In *2011 International Reliability Physics Symposium*, pages MY–7. IEEE, 2011. 65

[182] Yuchao Yang, Peng Gao, Siddharth Gaba, Ting Chang, Xiaoqing Pan, and Wei Lu. Observation of conducting filament growth in nanoscale resistive memories. *Nature communications*, 3(1):732, 2012. 65

[183] MB Gonzalez, JM Rafí, O Beldarrain, M Zabala, and F Campabadal. *IEEE Transactions on Device and Materials Reliability*, 14(2):769–771, 2014. 65

[184] Umberto Celano, Ludovic Goux, Attilio Belmonte, Karl Opsomer, Christophe Detavernier, Malgorzata Jurczak, and Wilfried Vandervorst. Conductive filaments multiplicity as a variability factor in cbram. In *2015 IEEE International Reliability Physics Symposium*, pages MY–11. IEEE, 2015. 65

[185] Leilei Qiao, Yiming Sun, Cheng Song, Siqi Yin, Qin Wan, Jialu Liu, Rui Wang, Fei Zeng, and Feng Pan. Performance improvement of conductive bridging random access memory by electrode alloying. *The Journal of Physical Chemistry C*, 124(21):11438–11443, 2020. 65

[186] Yiming Sun, Cheng Song, Jun Yin, Xianzhe Chen, Qin Wan, Fei Zeng, and Feng Pan. Guiding the growth of a conductive filament by nanoindentation to improve resistive switching. *ACS applied materials & interfaces*, 9(39): 34064–34070, 2017. 65

[187] Rongrong Cao, Sen Liu, Qi Liu, Xiaolong Zhao, Wei Wang, Xumeng Zhang, Facai Wu, Quantan Wu, Yan Wang, Hangbing Lv, et al. Improvement of device reliability by introducing a beol-compatible tin barrier layer in cbram. *IEEE Electron Device Letters*, 38(10):1371–1374, 2017. 65

[188] Shimeng Yu, Ximeng Guan, and H-S Philip Wong. On the stochastic nature of resistive switching in metal oxide rram: Physical modeling, monte carlo simulation, and experimental characterization. In *2011 International Electron Devices Meeting*, pages 17–3. IEEE, 2011. 65

[189] Robin Degraeve, Andrea Fantini, Nagarajan Raghavan, Ludovic Goux, Sergiu Clima, Bogdan Govoreanu, Attilio Belmonte, Dimitri Linten, and M Jurczak. Causes and consequences of the stochastic aspect of filamentary rram. *Microelectronic Engineering*, 147:171–175, 2015. 65

[190] B Gao, HW Zhang, Shimeng Yu, B Sun, LF Liu, XY Liu, Y Wang, RQ Han, JF Kang, B Yu, et al. Oxide-based rram: Uniformity improvement using a new material-oriented methodology. In *2009 Symposium on VLSI Technology*, pages 30–31. IEEE, 2009. 65

[191] Wen-Yuan Chang, Kai-Jung Cheng, Jui-Ming Tsai, Hung-Jen Chen, Frederick Chen, Ming-Jinn Tsai, and Tai-Bor Wu. Improvement of resistive switching characteristics in tio2 thin films with embedded pt nanocrystals. *Applied Physics Letters*, 95(4), 2009. 65

[192] Shimeng Yu, Bin Gao, Haibo Dai, Bing Sun, Lifeng Liu, Xiaoyan Liu, Ruqi Han, Jinfeng Kang, and Bin Yu. Improved uniformity of resistive switching behaviors in hfo2 thin films with embedded al layers. *Electrochemical and Solid-State Letters*, 13(2):H36, 2009. 65

[193] Nan Du, Heidemarie Schmidt, and Ilia Polian. Low-power emerging memristive designs towards secure hardware systems for applications in internet of things. *Nano Materials Science*, 3(2):186–204, 2021. 65

[194] Yachuan Pang, Bin Gao, Bohan Lin, He Qian, and Huaqiang Wu. Memristors for hardware security applications. *Advanced Electronic Materials*, 5(9): 1800872, 2019.

[195] Shubham Sahay and Manan Suri. Recent trends in hardware security exploiting hybrid cmos-resistive memory circuits. *Semiconductor Science and Technology*, 32(12):123001, 2017. 65

[196] Kyle Wallace, Kevin Moran, Ed Novak, Gang Zhou, and Kun Sun. Toward sensor-based random number generation for mobile and iot devices. *IEEE Internet of Things Journal*, 3(6):1189–1201, 2016. 66

[197] Benjamin Jun and Paul Kocher. The intel random number generator. *Cryptography Research Inc. white paper*, 27:1–8, 1999. 66

[198] Luca Baldanzi, Luca Crocetti, Francesco Falaschi, Matteo Bertolucci, Jacopo Belli, Luca Fanucci, and Sergio Saponara. Cryptographically secure pseudo-random number generator ip-core based on sha2 algorithm. *Sensors*, 20(7): 1869, 2020. 66, 67

[199] Yevgeniy Dodis, David Pointcheval, Sylvain Ruhault, Damien Vergniaud, and Daniel Wichs. Security analysis of pseudo-random number generators with input: /dev/random is not robust. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, pages 647–658, 2013. 66

[200] Elaine B Barker, William C Barker, William E Burr, W Timothy Polk, and Miles E Smid. Recommendation for key management: Part 1: General, 2007. 67

[201] Elaine B Barker, John Michael Kelsey, et al. *Recommendation for random number generation using deterministic random bit generators (revised).* US Department of Commerce, Technology Administration, National Institute of . . . , 2007. 67, 68

[202] Wolfgang Killmann and Werner Schindler. A proposal for: Functionality classes for random number generators. *ser. BDI, Bonn*, 2011. 67, 68

[203] Elaine Barker, Elaine Barker, Allen Roginsky, and Richard Davis. Recommendation for cryptographic key generation. 2012. 67

[204] Infineon Technologies AG. Psoctm 64 - secured mcu. URL `https://www.cypress.com/products/psoc-64-microcontrollers-arm-cortex-m4m0`. 68

[205] `https://www.st.com/en/microcontrollers-microprocessors/stm32f412.html`. [Accessed 06-04-2021]. 68

[206] Lawrence E Bassham III, Andrew L Rukhin, Juan Soto, James R Nechvatal, Miles E Smid, Elaine B Barker, Stefan D Leigh, Mark Levenson, Mark Vangel, David L Banks, et al. Sp 800-22 rev. 1a. a statistical test suite for random and pseudorandom number generators for cryptographic applications, 2010. 68

[207] Jianguo Yang, Yinyin Lin, Yarong Fu, Xiaoyong Xue, and BA Chen. A small area and low power true random number generator using write speed variation of oxidebased rram for iot security application. In *2017 IEEE international symposium on circuits and systems (ISCAS)*, pages 1–4. IEEE, 2017. 70, 72

[208] Hassen Aziza, Jeremy Postel-Pellerin, Hussein Bazzi, Pierre Canet, Mathieu Moreau, Vincenzo Della Marca, and Adnan Harb. True random number generator integration in a resistive ram memory array using input current limitation. *IEEE Transactions on Nanotechnology*, 19:214–222, 2020. 71, 72

[209] Takehiko Amaki, Masanori Hashimoto, and Takao Onoye. An oscillator-based true random number generator with jitter amplifier. In *2011 IEEE International Symposium of Circuits and Systems (ISCAS)*, pages 725–728. IEEE, 2011. 73

[210] Boyan Valtchanov, Alain Aubert, Florent Bernard, and Viktor Fischer. Modeling and observing the jitter in ring oscillators implemented in fpgas. In *2008 11th IEEE Workshop on Design and Diagnostics of Electronic Circuits and Systems*, pages 1–6. IEEE, 2008.

[211] Marco Bucci, Lucia Germani, Raimondo Luzzi, Alessandro Trifiletti, and Mario Varanonuovo. A high-speed oscillator-based truly random number source for cryptographic applications on a smart card ic. *IEEE transactions on computers*, 52(4):403–409, 2003. 73

[212] Takehiko Amaki, Masanori Hashimoto, and Takao Onoye. Jitter amplifier for oscillator-based true random number generator. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 96 (3):684–696, 2013. 73

[213] Kaiyuan Yang, David Fick, Michael B Henry, Yoonmyung Lee, David Blaauw, and Dennis Sylvester. 16.3 a 23mb/s 23pj/b fully synthesized true-random-number generator in 28nm and 65nm cmos. In *2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*, pages 280–281. IEEE, 2014. 73

[214] Anju P Johnson, Rajat Subhra Chakraborty, and Debdeep Mukhopadyay. An improved dcm-based tunable true random number generator for xilinx fpga. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 64(4): 452–456, 2016. 74

[215] Miguel A Prada-Delgado, Cristina Martínez-Gómez, and Iluminada Baturone. Auto-calibrated ring oscillator trng based on jitter accumulation. In *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–4. IEEE, 2020.

[216] Qianying Tang, Bongjin Kim, Yingjie Lao, Keshab K Parhi, and Chris H Kim. True random number generator circuits based on single-and multi-phase beat frequency detection. In *Proceedings of the IEEE 2014 Custom Integrated Circuits Conference*, pages 1–4. IEEE, 2014. 74

[217] Chaoyang Li, Qin Wang, Jianfei Jiang, and Nin Guan. A metastability-based true random number generator on fpga. In *2017 IEEE 12th international conference on ASIC (ASICON)*, pages 738–741. IEEE, 2017. 74

[218] Chaoyang Li, Qin Wang, Jianfei Jiang, and Nin Guan. A metastability-based true random number generator on fpga. In *2017 IEEE 12th international conference on ASIC (ASICON)*, pages 738–741. IEEE, 2017. 74

[219] Vikram B Suresh and Wayne P Burleson. Entropy and energy bounds for metastability based trng with lightweight post-processing. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 62(7):1785–1793, 2015. 74

[220] Myunghwan Park, John C Rodgers, and Daniel P Lathrop. True random number generation using cmos boolean chaotic oscillator. *Microelectronics Journal*, 46(12):1364–1370, 2015. 74

[221] Siva Nishok Dhanuskodi, Arunkumar Vijayakumar, and Sandip Kundu. A chaotic ring oscillator based random number generator. In *2014 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, pages 160–165. IEEE, 2014.

[222] H Moqadasi and MB Ghaznavi-Ghoushchi. A new chua's circuit with monolithic chua's diode and its use for efficient true random number generation in cmos 180 nm. *Analog Integrated Circuits and Signal Processing*, 82:719–731, 2015. 74

[223] Borja Martinez, Marius Monton, Ignasi Vilajosana, and Joan Daniel Prades. The power of models: Modeling power consumption for iot devices. *IEEE Sensors Journal*, 15(10):5777–5789, 2015. 74

[224] Kaiyuan Yang, Qing Dong, Zhehong Wang, Yi-Chun Shih, Yu-Der Chih, Jonathan Chang, David Blaauw, and Dennis Svlvester. A 28nm integrated true random number generator harvesting entropy from mram. In *2018 IEEE Symposium on VLSI Circuits*, pages 171–172. IEEE, 2018. 74

[225] Won Ho Choi, Yang Lv, Jongyeon Kim, Abhishek Deshpande, Gyuseong Kang, Jian-Ping Wang, and Chris H Kim. A magnetic tunnel junction based true random number generator with conditional perturb and real-time output probability tracking. In *2014 IEEE International Electron Devices Meeting*, pages 12–5. IEEE, 2014. 74

[226] Huiming Chen, Shuai Zhang, Nuo Xu, Min Song, Xin Li, Ruofan Li, Yi Zeng, Jeongmin Hong, and Long You. Binary and ternary true random number generators based on spin orbit torque. In *2018 IEEE International Electron Devices Meeting (IEDM)*, pages 36–5. IEEE, 2018. 74

[227] Min Song, Wei Duan, Shuai Zhang, Zhenjiang Chen, and Long You. Power and area efficient stochastic artificial neural networks using spin–orbit torque-based true random number generator. *Applied Physics Letters*, 118(5), 2021.

[228] Yang Liu, Zhaohao Wang, Zuwei Li, Xiaoxiao Wang, and Weisheng Zhao. A spin orbit torque based true random number generator with real-time optimization. In *2018 IEEE 18th International Conference on Nanotechnology (IEEE-NANO)*, pages 1–4. IEEE, 2018. 74

[229] Halid Mulaosmanovic, Thomas Mikolajick, and Stefan Slesazeck. Random number generation based on ferroelectric switching. *IEEE Electron Device Letters*, 39(1):135–138, 2017. 74

[230] Blaise Gassend, Dwaine Clarke, Marten Van Dijk, and Srinivas Devadas. Silicon physical random functions. In *Proceedings of the 9th ACM Conference on Computer and Communications Security*, pages 148–160, 2002. 76

[231] G Edward Suh, Charles W O'Donnell, and Srinivas Devadas. Aegis: A single-chip secure processor. *IEEE Design & Test of Computers*, 24(6):570–580, 2007. 76

[232] Charles Herder, Meng-Day Yu, Farinaz Koushanfar, and Srinivas Devadas. Physical unclonable functions and applications: A tutorial. *Proceedings of the IEEE*, 102(8):1126–1141, 2014. 78

[233] Thomas McGrath, Ibrahim E Bagci, Zhiming M Wang, Utz Roedig, and Robert J Young. A puf taxonomy. *Applied physics reviews*, 6(1), 2019. 78

[234] Roel Maes and Roel Maes. *Physically unclonable functions: Concept and constructions*. Springer, 2013. 78

[235] Mehrdad Majzoobi, Masoud Rostami, Farinaz Koushanfar, Dan S Wallach, and Srinivas Devadas. Slender puf protocol: A lightweight, robust, and secure authentication by substring matching. In *2012 IEEE Symposium on Security and Privacy Workshops*, pages 33–44. IEEE, 2012. 78

[236] Jeroen Delvaux, Roel Peeters, Dawu Gu, and Ingrid Verbauwhede. A survey on lightweight entity authentication with strong pufs. *ACM Computing Surveys (CSUR)*, 48(2):1–42, 2015.

[237] Ghaith Hammouri and Berk Sunar. Puf-hb: A tamper-resilient hb based authentication protocol. In *Applied Cryptography and Network Security: 6th International Conference, ACNS 2008, New York, NY, USA, June 3-6, 2008. Proceedings 6*, pages 346–365. Springer, 2008. 78

[238] Blaise Gassend, Dwaine Clarke, Marten Van Dijk, and Srinivas Devadas. Silicon physical random functions. In *Proceedings of the 9th ACM Conference on Computer and Communications Security*, pages 148–160, 2002. 79

[239] Md Nazmul Islam and Sandip Kundu. Enabling ic traceability via blockchain pegged to embedded puf. *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, 24(3):1–23, 2019. 79

[240] Jorge Guajardo, Sandeep S Kumar, Geert-Jan Schrijen, and Pim Tuyls. Physical unclonable functions and public-key crypto for fpga ip protection. In *2007 International Conference on Field Programmable Logic and Applications*, pages 189–195. IEEE, 2007. 79

[241] Sudhir Satpathy, Sanu Mathew, Vikram Suresh, Mark Anders, Himanshu Kaul, Amit Agarwal, Steven Hsu, Ram Krishnamurthy, and Vivek De. An all-digital unified static/dynamic entropy generator featuring self-calibrating hierarchical von neumann extraction for secure privacy-preserving mutual authentication in iot mote platforms. In *2018 IEEE Symposium on VLSI Circuits*, pages 169–170. IEEE, 2018. 79

[242] Bohan Lin, Bin Gao, Yachuan Pang, Jianshi Tang, He Qian, and Huaqiang Wu. A unified memory and hardware security module based on the adjustable switching window of resistive memory. *IEEE Journal of the Electron Devices Society*, 8:1257–1265, 2020. 79

[243] Frederik Armknecht, Roel Maes, Ahmad-Reza Sadeghi, Berk Sunar, and Pim Tuyls. Memory leakage-resilient encryption based on physically unclonable functions. *Towards Hardware-Intrinsic Security: Foundations and Practice*, pages 135–164, 2010. 81

[244] Sudhir Satpathy, Sanu Mathew, Jiangtao Li, Patrick Koeberl, Mark Anders, Himanshu Kaul, Gregory Chen, Amit Agarwal, Steven Hsu, and Ram Krishnamurthy. 13fj/bit probing-resilient 250k puf array with soft darkbit masking for 1.94% bit-error in 22nm tri-gate cmos. In *ESSCIRC 2014-40th European Solid State Circuits Conference (ESSCIRC)*, pages 239–242. IEEE, 2014. 81

[245] Mehrdad Majzoobi, Farinaz Koushanfar, and Srinivas Devadas. Fpga puf using programmable delay lines. In *2010 IEEE international workshop on information forensics and security*, pages 1–6. IEEE, 2010. 81

[246] Fatemeh Ganji, Domenic Forte, and Jean-Pierre Seifert. Pufmeter a property testing tool for assessing the robustness of physically unclonable functions to machine learning attacks. *IEEE Access*, 7:122513–122521, 2019. 81

[247] Sven Puchinger, Sven Müelich, Martin Bossert, Matthias Hiller, and Georg Sigl. On error correction for physical unclonable functions. In *SCC 2015; 10th International ITG Conference on Systems, Communications and Coding*, pages 1–6. VDE, 2015. 81

[248] Matthias Hiller, Ludwig Kürzinger, and Georg Sigl. Review of error correction for pufs and evaluation on state-of-the-art fpgas. *Journal of cryptographic engineering*, 10(3):229–247, 2020. 81

[249] Robbert Van Den Berg, Boris Skoric, and Vincent van der Leest. Bias-based modeling and entropy analysis of pufs. In *Proceedings of the 3rd international workshop on Trustworthy embedded devices*, pages 13–20, 2013. 81

[250] Patrick Koeberl, Jiangtao Li, Roel Maes, Anand Rajan, Claire Vishik, and Marcin Wójcik. Evaluation of a puf device authentication scheme on a discrete 0.13 um sram. In *Trusted Systems: Third International Conference, INTRUST 2011, Beijing, China, November 27-29, 2011, Revised Selected Papers 3*, pages 271–288. Springer, 2012. 81

[251] Roel Maes, Vincent Van Der Leest, Erik Van Der Sluis, and Frans Willems. Secure key generation from biased pufs. In *Cryptographic Hardware and Embedded Systems–CHES 2015: 17th International Workshop, Saint-Malo, France, September 13-16, 2015, Proceedings 17*, pages 517–534. Springer, 2015. 81

[252] Jeroen Delvaux, Dawu Gu, Ingrid Verbauwhede, Matthias Hiller, and Meng-Day Yu. Efficient fuzzy extraction of puf-induced secrets: Theory and applications. In *International Conference on Cryptographic Hardware and Embedded Systems*, pages 412–431. Springer, 2016. 81

[253] Aydin Aysu, Ye Wang, Patrick Schaumont, and Michael Orshansky. A new maskless debiasing method for lightweight physical unclonable functions. In *2017 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, pages 134–139. IEEE, 2017. 81

[254] Rui Liu, Huaqiang Wu, Yachuan Pang, He Qian, and Shimeng Yu. Experimental characterization of physical unclonable function based on 1 kb resistive random access memory arrays. *IEEE Electron Device Letters*, 36 (12):1380–1383, 2015. 82

[255] Bohan Lin, Yachuan Pang, Bin Gao, Jianshi Tang, Dong Wu, Ting-Wei Chang, Wei-En Lin, Xiaoyu Sun, Shimeng Yu, Meng-Fan Chang, et al. A highly reliable rram physically unclonable function utilizing post-process randomness source. *IEEE Journal of Solid-State Circuits*, 56(5):1641–1650, 2021. 82, 84, 86

[256] Yachuan Pang, Huaqiang Wu, Bin Gao, Ning Deng, Dong Wu, Rui Liu, Shimeng Yu, An Chen, and He Qian. Optimization of rram-based physical unclonable function with a novel differential read-out method. *IEEE Electron Device Letters*, 38(2):168–171, 2017. 82

[257] Jianguo Yang, Xing Li, Tao Wang, Xiaoyong Xue, Zhiliang Hong, Yuanyuan Wang, David Wei Zhang, and Hongliang Lu. A physically unclonable function with ber< 0.35% for secure chip authentication using write speed variation of rram. In *2018 48th European Solid-State Device Research Conference (ESSDERC)*, pages 54–57. IEEE, 2018. 82, 91

[258] Lingyun Shi, Guohao Zheng, Bobo Tian, Brahim Dkhil, and Chungang Duan. Research progress on solutions to the sneak path issue in memristor crossbar arrays. *Nanoscale Advances*, 2(5):1811–1827, 2020. 82

[259] Mohammad Reza Mahmoodi, Hussein Nili, and Dmitri B Strukov. Rx-puf: Low power, dense, reliable, and resilient physically unclonable functions based on analog passive rram crossbar arrays. In *2018 IEEE Symposium on VLSI Technology*, pages 99–100. IEEE, 2018. 82, 86, 91

[260] MR Mahmoodi, H Nili, Z Fahimi, S Larimian, H Kim, and D Strukov. Ultra-low power physical unclonable function with nonlinear fixed-resistance crossbar circuits. In *2019 IEEE International Electron Devices Meeting (IEDM)*, pages 30–1. IEEE, 2019. 83, 86, 91

[261] Hussein Nili, Gina C Adam, Brian Hoskins, Mirko Prezioso, Jeeson Kim, M Reza Mahmoodi, Farnood Merrikh Bayat, Omid Kavehei, and Dmitri B Strukov. Hardware-intrinsic security primitives enabled by analogue state and nonlinear conductance variations in integrated memristors. *Nature Electronics*, 1(3):197–202, 2018. 83, 86, 91

[262] Hussein Nili, Gina C Adam, Brian Hoskins, Mirko Prezioso, Jeeson Kim, M Reza Mahmoodi, Farnood Merrikh Bayat, Omid Kavehei, and Dmitri B Strukov. Hardware-intrinsic security primitives enabled by analogue state and nonlinear conductance variations in integrated memristors. *Nature Electronics*, 1(3):197–202, 2018. 83

[263] Jianguo Yang, Dengyun Lei, Deyang Chen, Jing Li, Haijun Jiang, Qingting Ding, Qing Luo, Xiaoyong Xue, Hangbing Lv, Xiaoyang Zeng, et al. A machine-learning-resistant 3d puf with 8-layer stacking vertical rram and 0.014% bit error rate using in-cell stabilization scheme for iot security applications. In *2020 IEEE International Electron Devices Meeting (IEDM)*, pages 28–6. IEEE, 2020. 83, 86, 87

[264] Yachuan Pang, Bin Gao, Dong Wu, Shengyu Yi, Qi Liu, Wei-Hao Chen, Ting-Wei Chang, Wei-En Lin, Xiaoyu Sun, Shimeng Yu, et al. 25.2 a reconfigurable rram physically unclonable function utilizing post-process randomness source with< 6× 10- 6 native bit error rate. In *2019 IEEE International Solid-State Circuits Conference-(ISSCC)*, pages 402–404. IEEE, 2019. 84, 86

[265] Xiaojin Zhao, Qiang Zhao, Yongpan Liu, and Feng Zhang. An ultracompact switching-voltage-based fully reconfigurable rram puf with low native instability. *IEEE Transactions on Electron Devices*, 67(7):3010–3013, 2020. 84, 86

[266] Qiang Zhao, Wenhan Zheng, Xiaojin Zhao, Yuan Cao, Feng Zhang, and Man-Kay Law. A 108 f 2/bit fully reconfigurable rram puf based on truly random dynamic entropy of jitter noise. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 67(11):3866–3879, 2020. 84, 86

[267] Rohit Abraham John, Nimesh Shah, Sujaya Kumar Vishwanath, Si En Ng, Benny Febriansyah, Metikoti Jagadeeswararao, Chip-Hong Chang, Arindam Basu, and Nripan Mathews. Halide perovskite memristors as flexible and reconfigurable physical unclonable functions. *Nature Communications*, 12 (1):3681, 2021. 85

[268] G Edward Suh and Srinivas Devadas. Physical unclonable functions for device authentication and secret key generation. In *Proceedings of the 44th annual design automation conference*, pages 9–14, 2007. 88

[269] Abhranil Maiti, Jeff Casarona, Luke McHale, and Patrick Schaumont. A large scale characterization of ro-puf. In *2010 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, pages 94–99. IEEE, 2010. 88

[270] Mingze Gao, Khai Lai, and Gang Qu. A highly flexible ring oscillator puf. In *Proceedings of the 51st annual design automation conference*, pages 1–6, 2014. 88

[271] Shohreh Sharif Mansouri and Elena Dubrova. Ring oscillator physical unclonable function with multi level supply voltages. In *2012 IEEE 30th International Conference on Computer Design (ICCD)*, pages 520–521. IEEE, 2012.

[272] Yuan Cao, Le Zhang, Chip-Hong Chang, and Shoushun Chen. A low-power hybrid ro puf with improved thermal stability for lightweight applications. *IEEE Transactions on computer-aided design of integrated circuits and systems*, 34(7):1143–1147, 2015.

[273] Sajid Khan, Ambika Prasad Shah, Neha Gupta, Shailesh Singh Chouhan, Jai Gopal Pandey, and Santosh Kumar Vishvakarma. An ultra-low power, reconfigurable, aging resilient ro puf for iot applications. *Microelectronics journal*, 92:104605, 2019.

[274] Yuan Cao, Xiaojin Zhao, Wenbin Ye, Qingbang Han, and Xiaofang Pan. A compact and low power ro puf with high resilience to the em side-channel attack and the svm modelling attack of wireless sensor networks. *Sensors*, 18(2):322, 2018. 88

[275] Urbi Chatterjee, Rajat Subhra Chakraborty, Hitesh Kapoor, and Debdeep Mukhopadhyay. Theory and application of delay constraints in arbiter puf. *ACM Transactions on Embedded Computing Systems (TECS)*, 15(1):1–20, 2016. 89

[276] Fukui Dan, Yehan Xu, Zheng Li, Jing Wen, Ben Liu, Shuai Chen, and Bing Li. A modeling attack resistant r-xor apuf based on fpga. In *2018 IEEE 3rd International Conference on Signal and Image Processing (ICSIP)*, pages 577–581. IEEE, 2018. 89

[277] Mehrdad Majzoobi, Farinaz Koushanfar, and Miodrag Potkonjak. Lightweight secure pufs. In *2008 IEEE/ACM International Conference on Computer-Aided Design*, pages 670–673. IEEE, 2008. 89

[278] Durga Prasad Sahoo, Debdeep Mukhopadhyay, Rajat Subhra Chakraborty, and Phuong Ha Nguyen. A multiplexer-based arbiter puf composition with enhanced reliability and security. *IEEE Transactions on Computers*, 67(3): 403–417, 2017. 89

[279] Siarhei S Zalivaka, Alexander A Ivaniuk, and Chip-Hong Chang. Reliable and modeling attack resistant authentication of arbiter puf in fpga implementation with trinary quadruple response. *IEEE Transactions on Information Forensics and Security*, 14(4):1109–1123, 2018.

[280] Siarhei S Zalivaka, Alexander A Ivaniuk, and Chip-Hong Chang. Low-cost fortification of arbiter puf against modeling attack. In *2017 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–4. IEEE, 2017.

[281] Takanori Machida, Dai Yamamoto, Mitsugu Iwamoto, Kazuo Sakiyama, et al. A new arbiter puf for enhancing unpredictability on fpga. *The Scientific World Journal*, 2015, 2015. 89

[282] Design of sram puf with improved uniformity and reliability utilizing device aging effect. 89

[283] Yizhak Shifman, Avi Miller, Osnat Keren, Yoav Weizmann, and Joseph Shor. A method to improve reliability in a 65-nm sram puf array. *IEEE Solid-State Circuits Letters*, 1(6):138–141, 2018. 89

[284] Supreet Jeloka, Kaiyuan Yang, Michael Orshansky, Dennis Sylvester, and David Blaauw. A sequence dependent challenge-response puf using 28nm sram 6t bit cell. In *2017 Symposium on VLSI Circuits*, pages C270–C271. IEEE, 2017. 89

[285] Vikash Kumar Rai, Somanath Tripathy, and Jimson Mathew. 2spuf: Machine learning attack resistant sram puf. In *2020 Third ISEA Conference on Security and Privacy (ISEA-ISAP)*, pages 149–154. IEEE, 2020. 89

[286] DY Wang, YC Hsin, KY Lee, GL Chen, SY Yang, HH Lee, YJ Chang, IJ Wang, YC Kuo, YS Chen, et al. Hardware implementation of physically unclonable function (puf) in perpendicular stt mram. In *2017 International Symposium on VLSI Technology, Systems and Application (VLSI-TSA)*, pages 1–2. IEEE, 2017. 89

[287] Yu-Sheng Chen, Ding-Yeong Wang, Yu-Chen Hsin, Kai-Yu Lee, Guan-Long Chen, Shan-Yi Yang, Hsin-Han Lee, Yao-Jen Chang, I-Jung Wang, Pei-Hua Wang, et al. On the hardware implementation of mram physically unclonable function. *IEEE Transactions on Electron Devices*, 64(11):4492–4495, 2017.

[288] Zhen Cao, Shuai Zhang, Jian Zhang, Nuo Xu, Ruofan Li, Zhe Guo, Jijun Yun, Min Song, Qiming Zou, Li Xi, et al. Reconfigurable physical unclonable function based on spin-orbit torque induced chiral domain wall motion. *IEEE Electron Device Letters*, 42(4):597–600, 2021.

[289] Giovanni Finocchio, T Moriyama, Raffaele De Rose, Giulio Siracusano, M Lanuzza, V Puliafito, S Chiappini, F Crupi, Z Zeng, T Ono, et al. Spin–orbit torque based physical unclonable function. *Journal of Applied Physics*, 128(3), 2020. 89

[290] MM Abutaleb. Qcapuf: Qca-based physically unclonable function as a hardware security primitive. *Semiconductor Science and Technology*, 33(4): 045011, 2018. 89

[291] Mohammad Hadi Valavi and Ghassem Jaberipur. Physically unclonable functions based on small delay defects in qca. *Semiconductor Science and Technology*, 35(3):035024, 2020. 89

[292] Mihir Bellare and Phillip Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In *Proceedings of the 1st ACM Conference on Computer and Communications Security*, pages 62–73, 1993. 92

[293] Ran Canetti. Towards realizing random oracles: Hash functions that hide all partial information. In *Advances in Cryptology—CRYPTO'97: 17th Annual International Cryptology Conference Santa Barbara, California, USA August 17–21, 1997 Proceedings 17*, pages 455–469. Springer, 1997.

[294] Jean-Sébastien Coron, Yevgeniy Dodis, Cécile Malinaud, and Prashant Puniya. Merkle-damgård revisited: How to construct a hash function. In *Advances in Cryptology–CRYPTO 2005: 25th Annual International Cryptology Conference, Santa Barbara, California, USA, August 14-18, 2005. Proceedings 25*, pages 430–448. Springer, 2005. 92

[295] Mihir Bellare, Ran Canetti, and Hugo Krawczyk. Keying hash functions for message authentication. In *Advances in Cryptology—CRYPTO'96: 16th Annual International Cryptology Conference Santa Barbara, California, USA August 18–22, 1996 Proceedings 16*, pages 1–15. Springer, 1996. 92

[296] Sumit Singh Dhanda, Brahmjit Singh, and Poonam Jindal. Lightweight cryptography: a solution to secure iot. *Wireless Personal Communications*, 112 (3):1947–1980, 2020. 92

[297] Vidya Rao and KV Prema. Light-weight hashing method for user authentication in internet-of-things. *Ad Hoc Networks*, 89:97–106, 2019.

[298] Amrita Roy Chowdhury, Tanusree Chatterjee, and Sipra DasBit. Locha: a light-weight one-way cryptographic hash algorithm for wireless sensor network. *Procedia Computer Science*, 32:497–504, 2014. 92

[299] Ricardo Chaves, Georgi Kuzmanov, Leonel Sousa, and Stamatis Vassiliadis. Cost-efficient sha hardware accelerators. *IEEE transactions on very large scale integration (VLSI) Systems*, 16(8):999–1008, 2008. 93

[300] Janaka Deepakumara, Howard M Heys, and R Venkatesan. Fpga implementation of md5 hash algorithm. In *Canadian Conference on Electrical and Computer Engineering 2001. Conference Proceedings (Cat. No. 01TH8555)*, volume 2, pages 919–924. IEEE, 2001. 93

[301] Xun Yi. Hash function based on chaotic tent maps. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 52(6):354–357, 2005. 93

[302] Mahdi Nouri, Ali Khezeli, Alireza Ramezani, and Azita Ebrahimi. A dynamic chaotic hash function based upon circle chord methods. In *6th International Symposium on Telecommunications (IST)*, pages 1044–1049. IEEE, 2012.

[303] Kristine Jean Diane A Virtudez and Reggie C Gustilo. Fpga implementation of a one-way hash function utilizing hl11-1111 nonlinear digital to analog converter. In *TENCON 2012 IEEE Region 10 Conference*, pages 1–5. IEEE, 2012. 93

[304] Leonid Azriel and Shahar Kvatinsky. Towards a memristive hardware secure hash function (memhash). In *2017 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, pages 51–55. IEEE, 2017. 93

[305] John Paul Strachan, Catherine Graves, and Suhas Kumar. Hash computation using memristor-implemented dot product engine. *Google Patents*, 2019. 93

[306] Information Technology Laboratory Computer Security Division. Lightweight cryptography: Csrc. URL `https://csrc.nist.gov/projects/lightweight-cryptography`. 93

[307] Mohammad Nasim Imtiaz Khan, Shivam Bhasin, Alex Yuan, Anupam Chattopadhyay, and Swaroop Ghosh. Side-channel attack on sttram based cache for cryptographic application. In *2017 IEEE International Conference on Computer Design (ICCD)*, pages 33–40. IEEE, 2017. 94

[308] Alex Pappachen James. An overview of memristive cryptography. *The European Physical Journal Special Topics*, 228(10):2301–2312, 2019. 94

[309] Kunal Korgaonkar, Ronny Ronen, Anupam Chattopadhyay, and Shahar Kvatinsky. The bitlet model: Defining a litmus test for the bitwise processing-in-memory paradigm. *arXiv preprint arXiv:1910.10234*, 2019. 94

[310] Karthikeyan Nagarajan, Sina Sayyah Ensan, Mohammad Nasim Imtiaz Khan, Swaroop Ghosh, and Anupam Chattopadhyay. Shine: A novel sha-3 implementation using reram-based in-memory computing. In *2019 IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*, pages 1–6. IEEE, 2019. 94

[311] Jiang Li, Yijun Cui, Chenghua Wang, Chongyan Gu, and Weiqiang Liu. A fully configurable puf using dynamic variations of resistive crossbar arrays. *IEEE Transactions on Nanotechnology*, 21:737–746, 2022. 98

[312] Mohammad Reza Mahmoodi, Hussein Nili, and Dmitri B Strukov. Rx-puf: Low power, dense, reliable, and resilient physically unclonable functions based on analog passive rram crossbar arrays. In *2018 IEEE Symposium on VLSI Technology*, pages 99–100. IEEE, 2018. 99, 108, 112

[313] Bohan Lin, Yachuan Pang, Bin Gao, Jianshi Tang, Dong Wu, Ting-Wei Chang, Wei-En Lin, Xiaoyu Sun, Shimeng Yu, Meng-Fan Chang, et al. A highly reliable rram physically unclonable function utilizing post-process randomness source. *IEEE Journal of Solid-State Circuits*, 56(5):1641–1650, 2021. 99, 108, 112, 120

[314] Xiaojin Zhao, Qiang Zhao, Yongpan Liu, and Feng Zhang. An ultracompact switching-voltage-based fully reconfigurable rram puf with low native instability. *IEEE Transactions on Electron Devices*, 67(7):3010–3013, 2020. 99, 112, 120

[315] Qiang Zhao, Wenhan Zheng, Xiaojin Zhao, Yuan Cao, Feng Zhang, and Man-Kay Law. A 108 f 2/bit fully reconfigurable rram puf based on truly random dynamic entropy of jitter noise. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 67(11):3866–3879, 2020. 99, 108

[316] Christopher Bengel, Anne Siemon, Felix Cüppers, Susanne Hoffmann-Eifert, Alexander Hardtdegen, Moritz von Witzleben, Lena Hellmich, Rainer Waser, and Stephan Menzel. Variability-aware modeling of filamentary oxide-based bipolar resistive switching cells using spice level compact models. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 67(12):4618–4630, 2020. 99, 113, 114, 127

[317] MR Mahmoodi, H Nili, Z Fahimi, S Larimian, H Kim, and D Strukov. Ultra-low power physical unclonable function with nonlinear fixed-resistance crossbar circuits. In *2019 IEEE International Electron Devices Meeting (IEDM)*, pages 30–1. IEEE, 2019. 108

[318] Mohammad Reza Mahmoodi, Zahra Fahimi, Shabnam Larimian, Hussein Nili, Hyugin Kim, and Dmitri B Strukov. A strong physically unclonable function with$>$ 2 crps and$<$ 1.4% ber using passive reram technology. *IEEE Solid-State Circuits Letters*, 3:182–185, 2020. 108

[319] Yijun Cui, Jiang Li, Chongyan Gu, Chenghua Wang, and Weiqiang Liu. An rram-based puf with adjustable programmable voltage and multi-mode operation. In *Proceedings of the 18th ACM International Symposium on Nanoscale Architectures*, pages 1–5, 2023. 108

[320] Intrinsic ID. Sram puf. `https://www.intrinsic-id.com/`. 111

[321] Gokulnath Rajendran, Writam Banerjee, Anupam Chattopadhyay, and Mohamed M Sabry Aly. Application of resistive random access memory in hardware security: A review. *Advanced Electronic Materials*, 7(12):2100536, 2021. 111, 112

[322] Nist sp 800-22: Documentation and software - random bit generation: Csrc, . URL `https://csrc.nist.gov/projects/random-bit-generation/documentation-and-software`. 111

[323] Boaz Barak and Shai Halevi. A model and architecture for pseudo-random generation with applications to/dev/random. In *Proceedings of the 12th ACM conference on Computer and communications security*, pages 203–212, 2005. 111

[324] Rohit Abraham John, Nimesh Shah, Sujaya Kumar Vishwanath, Si En Ng, Benny Febriansyah, Metikoti Jagadeeswararao, Chip-Hong Chang, Arindam Basu, and Nripan Mathews. Halide perovskite memristors as flexible and reconfigurable physical unclonable functions. *Nature Communications*, 12 (1):3681, 2021. 112

[325] Pai-Yu Chen, Runchen Fang, Rui Liu, Chaitali Chakrabarti, Yu Cao, and Shimeng Yu. Exploiting resistive cross-point array for compact design of physical unclonable function. In *2015 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, pages 26–31, 2015. doi: 10.1109/HST.2015.7140231. 112

[326] Z Wei, Y Katoh, S Ogasahara, Y Yoshimoto, K Kawai, Y Ikeda, K Eriguchi, K Ohmori, and S Yoneda. True random number generator using current difference based on a fractional stochastic model in 40-nm embedded reram. In *2016 IEEE International Electron Devices Meeting (IEDM)*, pages 4–8. IEEE, 2016. 113, 120

[327] Jianguo Yang, Yinyin Lin, Yarong Fu, Xiaoyong Xue, and BA Chen. A small area and low power true random number generator using write speed variation of oxidebased rram for iot security application. In *2017 IEEE international symposium on circuits and systems (ISCAS)*, pages 1–4. IEEE, 2017. 113

[328] Bohan Lin, Bin Gao, Yachuan Pang, Peng Yao, Dong Wu, Hu He, Jianshi Tang, He Qian, and Huaqiang Wu. A high-speed and high-reliability trng based on analog rram for iot security application. In *2019 IEEE International Electron Devices Meeting (IEDM)*, pages 14–8. IEEE, 2019. 113, 120

[329] Bin Gao, Bohan Lin, Xueqi Li, Jianshi Tang, He Qian, and Huaqiang Wu. A unified puf and trng design based on 40-nm rram with high entropy and robustness for iot security. *IEEE Transactions on Electron Devices*, 69(2): 536–542, 2022. 113, 120

[330] Gokulnath Rajendran, Furqan Zahoor, Simranjeet Singh, Farhad Merchant, Vikas Rana, and Anupam Chattopadhyay. Pr-puf: A reconfigurable strong rram puf. In *2023 IFIP/IEEE 31th International Conference on Very Large Scale Integration (VLSI-SoC)*. IEEE, 2023. 113, 114, 116

[331] Le Zhang, Xuanyao Fong, Chip-Hong Chang, Zhi Hui Kong, and Kaushik Roy. Feasibility study of emerging non-volatilememory based physical unclonable functions. In *2014 IEEE 6th international memory workshop (IMW)*, pages 1–4. IEEE, 2014. 114, 116

[332] Abhranil Maiti, Vikash Gunreddy, and Patrick Schaumont. A systematic method to evaluate and compare the performance of physical unclonable functions. *Embedded systems design with FPGAs*, pages 245–267, 2013. 115

[333] Rui Liu, Huaqiang Wu, Yachun Pang, He Qian, and Shimeng Yu. A highly reliable and tamper-resistant rram puf: Design and experimental validation. In *2016 IEEE international symposium on hardware oriented security and trust (HOST)*, pages 13–18. IEEE, 2016. 116

[334] Recommendation for the entropy sources used for random bit generation, . URL `https://csrc.nist.gov/pubs/sp/800/90/b/final`. 118

[335] Hassen Aziza, Jeremy Postel-Pellerin, Hussein Bazzi, Pierre Canet, Mathieu Moreau, Vincenzo Della Marca, and Adnan Harb. True random number generator integration in a resistive ram memory array using input current limitation. *IEEE Transactions on Nanotechnology*, 19:214–222, 2020. 120

[336] Crypto coprocessor. URL `https://www.nxp.com/products/processors-and-microcontrollers/legacy-mpu-mcus/crypto-coprocessors/crypto-coprocessor:C29x`. 121

[337] Melvin Estuardo Galicia et al. " s3cure": Scramble, shuffle and shambles-secure deployment of weight matrices in memristor crossbar arrays. In *Proceedings of the 2023 International Conference on Neuromorphic Systems*, pages 1–8, 2023. 124, 126, 127

[338] Minhui Zou et al. Security enhancement for rram computing system through obfuscating crossbar row connections. In *DATE*, pages 466–471. IEEE, 2020.

[339] Yuhang Wang et al. A low cost weight obfuscation scheme for security enhancement of reram based neural network accelerators. In *Asia and South Pacific Design Automation Conference*, pages 499–504, 2021. 124

[340] Minhui Zou et al. Enhancing security of memristor computing system through secure weight mapping. In *2022 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, pages 182–187. IEEE, 2022. 124, 127

[341] Minhui Zou, Nan Du, and Shahar Kvatinsky. Review of security techniques for memristor computing systems. *Frontiers in Electronic Materials*, 2:1010613, 2022.

[342] Chintan Chavda et al. Vulnerability analysis of {On-Chip}{Access-Control} memory. In *9th USENIX Workshop on Hot Topics in Storage and File Systems (HotStorage 17)*, 2017. 126

[343] Nhu Huynh et al. Hardware security of emerging non-volatile memory devices under imaging attacks. In *2021 International Conference on Applied Electronics (AE)*, pages 1–4. IEEE, 2021. 126

[344] Gokulnath Rajendran et al. Harnessing Entropy: RRAM Crossbar-based Unified PUF and RNG. In *International Conference on VLSI Design (VLSID)*. IEEE, 2024. 126