Technische Universität München
Lehrstuhl für Sicherheit in der Informationstechnik
an der Fakultät für Elektrotechnik und Informationstechnik

Higher-Order Alphabet Physical Unclonable Functions

Constructions, Properties, and Applications

Vincent Charles Immler

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Informationstechnik der Technischen Universität München zur Erlangung des akademischen Grades eines **Doktor-Ingenieurs (Dr.-Ing.)** genehmigten Dissertation.

Vorsitzender der Kommission:	Prof. Dr. Sc. techn. Gerhard Kramer
Prüfer der Dissertation:	1. Prof. Dr.-Ing. Georg Sigl
	2. Prof. Dr. rer. nat. Christoph Kutter

Die Dissertation wurde am 04.04.2019 bei der Technischen Universität München eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am 25.10.2019 angenommen.

To my wife and children

Author's contact information:
vincent+phd@immler.us

Thesis Advisor:	**Prof. Dr.-Ing. Georg Sigl** Technical University of Munich (TUM)
Secondary Referee:	**Prof. Dr. rer. nat. Christoph Kutter** University of the German Federal Armed Forces (UniBW)
Thesis submitted:	April 04, 2019
Thesis defense:	November 29, 2019

Our greatest glory is not in never falling,
but in rising every time we fall.

CONFUCIUS

All human beings are born free and equal in dignity and rights. They are endowed with reason and conscience and should act towards one another in a spirit of brotherhood.

UNIVERSAL DECLARATION OF HUMAN RIGHTS

Abstract

Protecting secret information such as cryptographic keys and safeguarding physical integrity of a device are two related challenges when considering physical attacks, since the attacker may control the device in a hostile environment and carry out a wide range of sophisticated attacks. To detect physical intruders as part of a layered approach to security, it is common to implement an Access Denial System (ADS) on a board- or system-level, i.e., a mechanism that provides resistance towards physical attacks and that may also actively detect them and respond correspondingly. Most commonly, these mechanisms are based on a battery-backed continuous monitoring of a physical security boundary such as a finely patterned mesh that surrounds the protected components. Systems with this type of countermeasure typically store the cryptographic keys in battery-backed volatile memory such that upon detection of an intruder, this memory can be instantaneously erased.

Physical Unclonable Functions (PUFs) provide an alternative approach to cryptographic key storage. PUFs are based on the inherent manufacturing variations of a physical token that can be leveraged to create a kind of fingerprint, i.e., the key is no longer explicitly stored but represented by the physical characteristics of the token. The fingerprint's unique data can then be used as a seed for a cryptographic key generation. This process must be carried out upon each device startup. However, as the data is generated from physical measurements, it is inherently fuzzy which necessitates reliability enhancement techniques to ultimately obtain a reliable key.

This thesis focuses on a specific type of PUF with the property of tamper-evidence, i.e., a PUF that upon physical tampering provides sufficiently altered output data such that reconstruction of the designated key fails. More in particular, a new class of PUF is introduced where the output is no longer binary but instead comprised of symbols from a Higher-Order Alphabet (HOA). This new approach aligns well with the goals to achieve tamper-evidence and provides an alternative on how to construct an ADS without battery-backup and continuous sensing.

As part of the presented work, the full stack of this approach is investigated, ranging from the physical and architectural construction of the PUF, to specifics of the measurement circuit, and the algorithmic data processing as part of the reliability enhancement. The devised concept of a HOA PUF is practically demonstrated by two implementations with an in-depth assessment of their properties. Since the concept is generic, it is not limited to tamper-evident PUFs but could also be used to modify existing PUF designs. Since the assessment of the HOA PUF properties cannot be done with existing tools or criteria, they were adapted to reflect the PUF's behavior properly.

The results of this work are therefore manifold: a new class of PUF construction(s) with well-supported design rationale to achieve tamper-evidence, several contributions to the domain of reliability enhancement techniques in addition to quality metrics and tools to assess the newly created type of PUF. This is complemented by two practical implementations with a rigorous statistical analysis, environmental tests, and a practical security analysis. Overall, this work establishes a new branch of PUF research. Furthermore, it expands the state of the art by providing more efficient solutions w.r.t. some of the algorithmic data processing techniques involved.

Keywords cryptography, embedded security, FIPS 140-2, Anti-Tamper (AT), tamper-resistance, tamper-evidence, tamper-sensitivity, Physical Unclonable Function (PUF), fuzzy extractor, information theory, Higher-Order Alphabet PUF (HOA PUF), key derivation, Error-Correcting Codes (ECC), Access Denial System (ADS), volume protection.

Kurzfassung

Die Gewährleistung der Sicherheit geheimer Informationen wie etwa kryptographischer Schlüssel sowie der physikalischen Integrität eines Geräts sind zwei miteinander verknüpfte Herausforderungen im Kontext von physikalischen Angriffen. Dies ergibt sich aus der Tatsache, dass ein Angreifer das Gerät in einer feindseligen Umgebung betreiben und angreifen kann. Als Teil eines mehrstufigen Sicherheitskonzepts ist es daher üblich einen physikalischen Angreifer zu entdecken. Dies wird üblicherweise auf Basis eines geeigneten Zugriffsschutzsystems (Tamperschutz) geleistet, bspw. auf der Leiterplatten- oder Systemebene. Diese Schutzsysteme bieten Resistenz gegenüber physikalischen Angriffen und erlauben teilweise auch die Detektion eines Angreifers um proaktive Schutzmaßnahmen einzuleiten. Solche Mechanismen sind üblicherweise batterie-gepuffert und stellen die Sicherheit auf Basis einer zeitkontinuierlichen Überwachung eines engmaschigen Schutzgitters, welches das zu schützende Gerät umgibt, sicher. Bei Geräten dieser Schutzklasse wird der kryptographische Schlüssel in einem flüchtigen Speicher vorgehalten, so dass bei der Erkennung eines Angriffs eine sofortige Löschung des Schlüssels möglich ist.

Physical Unclonable Functions (PUFs) bieten eine Alternative zu dieser Schlüsselspeicherung an. Diese basiert auf den unvermeidbaren Toleranzen bei der Fertigung eines physikalischen Objekts, welche dann dazu genutzt werden können eine Art Fingerabdruck zu erzeugen. Der Schlüssel ist daher nicht mehr explizit gespeichert sondern wird durch die physikalischen Charakteristika des Objekts repräsentiert. Die dadurch vorhandenen eineindeutigen Daten können als Eingabe für eine kryptographische Schlüsselerzeugung genutzt werden. Dieser Prozess muss aber bei jedem Gerätestart wiederholt werden. Da die Daten Ergebnis eines physikalischen Messprozesses sind, sind diese jedes Mal teilweise leicht abweichend, so dass mit geeigneten Techniken zur Verbesserung der Ausfallsicherheit gearbeitet werden muss.

Die vorliegende Arbeit konzentriert sich auf eine bestimmte Art PUF, welche die Eigenschaft einer Unversehrtheits-Sicherung erfüllen, d.h. ein physikalischer Angriff verletzt diese Eigenschaft und erzeugt eine abweichende Ausgabe der PUF, so dass der ursprüngliche Schlüssel nicht rekonstruierbar ist. Insbesondere wird dabei eine neue Klasse von PUF eingeführt, wo die Ausgabe nicht mehr binär ist, sondern aus Symbolen eines höherwertigen Alphabets besteht. Dieser neue Ansatz erfüllt dabei die Anforderungen aus dem Bereich Tamperschutz besonders gut und stellt eine Konstruktion dar, wie ein Zugriffsschutz ohne Batteriepufferung realisiert werden kann.

Als Teil der Arbeit wird dabei das vollständige Spektrum dieses Ansatzes untersucht, beginnend mit der physikalischen Konstruktion und Architektur der PUF, über Eigenschaften der Messschaltung, sowie der algorithmischen Datennachverarbeitung als Teil der Verbesserung zur Ausfallsicherheit. Das entwickelte Konzept einer PUF mit höherwertigem Ausgabealphabet wird praktisch anhand von zwei Implementierungen demonstriert inklusive einer detailierten Bewertung der Eigenschaften. Da das Konzept generisch ist, ist es nicht auf PUFs mit der Eigenschaft einer Unversehrtheits-Sicherung beschränkt sondern könnte auch zukünftig dazu dienen andere PUFs anzupassen. Da die Bewertung des entwickelten PUFs nicht anhand existierender Kriterien oder Werkzeuge vorgenommen werden kann, mussten diese erweitert werden um die geänderten Begebenheiten widerzuspiegeln.

Das Ergebnis dieser Arbeit ist vielfältig: eine neue Klasse von PUF Konstruktion(en) mit klar begründetem Design zur Erreichung der Unversehrtheits-Sicherung, mehrere Beiträge zur Verbesserung der Ausfallsicherheit von PUFs, zuzüglich angepasster und neuer Metriken und Werkzeuge um die neuartige PUF zu bewerten. Dies wird vervollständigt durch zwei praktische Implementierung inklusive einer ausführlichen statistischen Analyse, Umgebungstests, und einer praktisch durchgeführten Sicherheitsanalyse.

Diese Arbeit begründet daher einen neuen Bereich der PUF Forschung. Darüber hinaus wird der Stand der Technik um neue und effizientere Methoden erweitert, bspw. in Bezug auf relevante algorithmische Datenverarbeitungsschritte.

Keywords Kryptographie, Eingebettete Sicherheit, FIPS 140-2, Anti-Tamper (AT), Tamperschutz, Tampersensitivität, Physical Unclonable Function (PUF), Fuzzy Extractor, Informationstheorie, Alphabet hörerer Ordnung, Schlüsselerzeugung, Fehlerkorrektur, physikalischer Zugriffsschutz.

Acknowledgements

This thesis describes the research that I conducted during my employment at Fraunhofer AISEC. I hope that the work presented here can help serve as an example of Fraunhofer's goal of creating innovative solutions for applied research. My colleagues both at AISEC and TUM, in addition to the spirit at work contributed significantly to the completion of this thesis. In particular, I wish to thank the following people for their support.

First of all, I am deeply grateful to my advisor Prof. Georg Sigl for his staunch support of my topic and making related research projects possible, both within the Fraunhofer Society and internationally, most importantly with DSO National Laboratories. These projects and corresponding collaborations were an enriching experience both on a personal and technical level. Moreover, I am thankful for his high expectations that motivated me to strive for the best solutions possible, his outstanding patience despite several setbacks, and his achievements towards a collaborative and good work atmosphere. I am indebted also to Prof. Christoph Kutter for his extremely encouraging and positive attitude, his unwavering support for our joint research project, and his guidance when needed. I would also like to thank my former superior Bartol Filipovic for always acting in my best interest, for providing the necessary degree of freedom to work on this highly interesting topic, and for paving the road for my later success. In addition, I am thankful for having the opportunity to join the newly founded Physical Security Technologies group headed by Matthias Hiller during the latter days of my employment. I am glad he was such a like-minded co-author with equal attention to detail and similar preferences in terms of writing. This made the whole paper writing process much easier and more pleasant. Moreover, his previous theoretical work on PUF key derivation provided new thought-provoking input for the tamper-evident PUF setting I was concerned with.

Regarding my coworkers, words are not enough to express the blessing of having Johannes Obermaier as such a hardworking teammate who was more than willing to participate in our sometimes extreme afterhour shifts and in particular for taking care of the discrete measurement circuit plus its related topics. Equally important to the discrete measurement circuit was the work done by Martin König of Fraunhofer EMFT, who went to great lengths to tailor and optimize the manufacturing processes to deliver the much-needed tamper-resistant PUF envelopes to confirm the overall design rationale. Other coworkers with whom I had the pleasure to work with on this topic and I would especially like to thank are Maxim and Oli, my former office mates, for not only getting me started on the topic of tamper-resistance but also for welcoming me to Fraunhofer and their resourceful teachings on how to succeed. Furthermore, I would like to thank Elischa Ferres, Alexander Utz, and Alexander Stanitzki, as well as the whole team of Fraunhofer IMS for their work on the developed integrated circuit for the PUF measurement.

My sincere thanks also go to my coworkers from the Hardware Security Department (HWS), including but not limited to: Robert Specht and Robert Hesselbarth for the great collaborations and interesting discussions. Further, I would like to thank Philipp Koppermann for his truly inspiring craftsmanship in creating presentation slides and our awesome trip to HOST 2017. Likewise are my recollections of HOST 2018 that I am lucky to share

with great people from both HWS and TUM. In addition, I would like to thank all people from TUM, most notable Michael Pehl for organizing the so called PUF cluster, the name of our regular meeting to discuss recent advances in this domain. This resulted in highly enjoyable collaborations with Lars Tebelmann and Michael Pehl, where we could jointly unleash our daddy superpowers (since being the only guys with children at the time).

There are also several students whom I head the pleasure to advise and who had a significant and positive impact on my work. Amongst others, I would like to name the following students with exceptional contributions: Qinzhi Liu, Karthik Uppund, Lukas Auer, and Aysun Önalan. Thank you! Special thanks also go to Ricarda Fedler for creating some of the artistic figures in this thesis, Viktor Deleski for being our entertainer and tireless advertiser, all our external partners I had the opportunity to work with, our administrative and technical staff at AISEC, and all the other helping hands.

Above all, I am eternally grateful to my wife. Her love and support carried me through the bumpy ride of pursuing a PhD. She endured more than I during this period and my achievements are no match to hers in the care and development of our children. The fortune and joy of having her and our children remind me of the truly important treasures in life.

Contents

Nomenclature

Abbreviations

3D Three-Dimensional

ADC Analog-to-Digital Converter

ADS Access Denial System

AES Advanced Encryption Standard

API Application Programming Interface

ASCII American Standard Code for Information Interchange

ASIC Application Specific Integrated Circuit

AT Anti-Tamper

ATEA Anti-Tamper Executive Agent

BBRAM Battery-Backed Random Access Memory

BCH Bose-Chaudhuri-Hocquenghem

BRAM Block Random Access Memory

CMOS Complementary Metal Oxide Semiconductor

COTS Commercial-Off-The-Shelf

CPS Cyber Physical System

CSP Critical Security Parameter

CTW Context-Tree-Weighting

DCT Discrete-Cosine-Transform

DEMA Differential Electro-Magnetic Analysis

DICE Device Identifier Composition Engine

DoD Department of Defense

DPA Differential Power Analysis

DPL Dual-Rail Precharge Logic

DSP Digital Signal Processor

DUT Device Under Test

ECC Error-Correcting Code

EMA Electromagnetic Analysis

ES Embedded System

EVP Enclosure for Volume Protection

FF Flip-Flop

FPGA Field Programmable Gate Array

GND Ground

HD Hamming Distance

HDL Hardware Description Language

HOA Higher-Order Alphabet

HSM Hardware Security Module

IC Integrated Circuit

ICS Industrial Control System

IP Intellectual Property

KEK Key-Encryption-Key

LDS Laser Direct Structuring

LFI Laser Fault Injection

LSB Least-Significant Bit

MAC Message-Authentication-Code

MCM Multiple-Chip Embedded Module

MCU Microcontroller Unit

ME Multiple Evaluation

MSB Most-Significant Bit

MUP Module Under Protection

NVM Non-Volatile Memory

PC Personal Computer

PCB Printed Circuit Board

PDF Probability Distribution Function

PIN Personal Identification Number

PUF Physical Unclonable Function

RAM Random Access Memory

RO Ring-Oscillator

ROM Read-Only Memory

RS Reed-Solomon

RX Receive

SCA Side Channel Analysis

SME Small and Medium-sized Enterprises

SNR Signal-to-Noise Ratio

SNVS Secure Non-Volatile Storage

SoC System on Chip

SPA Simple Power Analysis

SRAM Static Random Access Memory

SSE Systems Security Engineering

TCG Trusted Computing Group

TPM Trusted Platform Module

TRNG True Random Number Generator

TX Transmit

U.S. United States

VHDL VHSIC (Very High Speed Integrated Circuit) Hardware Description Language

VP Volume Protection

VT Varshamov-Tenengolts

XOR Exclusive OR

Symbols

$||$ Concatenation

$\oplus$ Binary XOR-Operation

Superscripts

$(\dot{\cdot})$ Denotes a noisy variable

Part I

Preliminaries

Chapter 1

Introduction and Preview

This chapter introduces basic aspects of physical security, cryptography, and summarizes the research contributions of this thesis. Since this thesis has been carried out as part of several projects at the Fraunhofer Institute AISEC, their scope, goals, and setting is briefly described in Section 1.5.

Contents

1.1 Motivation

Since the invention of modern electronics and computers, mankind has seen a rapid development in various technological areas like never before. Especially the performance gain in Integrated Circuits (ICs), as an indirect result of the observation known as Moore's law [144] has contributed to this remarkable growth, since faster machines employing more powerful ICs could carry out more complex tasks. At the same time, related technological advancements created new applications that could only succeed because of new forms of interaction, e.g., instead of computers with the size of a room we are now primarily exposed to Embedded Systems (ESs) or Cyber Physical Systems (CPSs) in everyday applications [123, 182]. Systems in this area are characterized by the following aspects: they often interact with their physical environment, i.e., by employing sensors and actuators, by being interconnected, i.e., via cyberspace or proximity based wireless communication, and they often no longer have the appearance of traditional Personal Computers (PCs) while mostly performing more dedicated functions. Devices adhering to these characteristics can be further classified based on their physical device architecture [147]: there are single-chip modules, e.g., a System on Chip (SoC) such as a smartcard, Multiple-Chip Embedded Modules (MCMs), e.g., Printed Circuit Boards (PCBs) with more than one IC as part of a carrier system, and multiple-chip standalone modules, e.g., a single device already providing all intended functions on its own. Most if not all of these systems contain at least

one Microcontroller Unit (MCU) to provide the necessary computing power. Considering all device architectures together, and taking into account that just a single well-equipped car already contains more than 50 MCU [44], it is apparent that ESs or CPSs outnumber traditional PC by orders of magnitude which makes them an even more critical building block of today's world.

Common concerns regarding such systems are safety, privacy, and security, whereas the latter will be the focus of this thesis. Ensuring the security of CPS has become increasingly more difficult due to their widespread use, a shorter time to market which is dictated by customer demand contradicting a thoughtful security-oriented development process, and due to the fact that sensitive data is stored more often in these devices nowadays, making them more rewarding for an attacker [114, 169]. Sensitive data can be Intellectual Property (IP), e.g., proprietary algorithms, or Critical Security Parameters (CSPs), e.g., cryptographic material and user credentials, or end-user data such as medical records and other data relevant to the user's privacy.

In general, there are software-based and hardware-based attacks [147]. Software-based attacks typically exploit logic errors in the software of a system. In contrast, hardware-based attacks exploit hardware interfaces or physical phenomena to interact with the system in ways not intended by its manufacturer or end-user, e.g., by observing the power consumption during a cryptographic operation [226]. Due to that, they are often referred to as physical attacks, too [186, 187]. To rule out the possibility of such physical tampering with a system it must be counteracted according to the required assurance level [111, 210]. This is a physical security objective to build security from the ground up [31, 28, 223] to avoid that an attacker extracts information from the device, adds or removes functionality, etc. Otherwise, information security objectives such as data Confidentiality, Integrity, and Availability (CIA) cannot be ensured which represent a cornerstone of trusted systems. This basic set of information security objectives is also commonly known as "CIA triad". A system is defined as *trustable* according to the Trusted Computing Group (TCG) [163] as long as "*if it always behaves in the expected manner for the intended purpose*". Without achieving basic security objectives, it is difficult to imagine how to establish trust in a system or achieving more complex security requirements [155].

Depending on the specific system and the attacker's intentions, it is likely that the targeted asset is different. However, independent of the attacker's strategy, there is a set of prudent engineering principles that minimize both the probability of a successful attack and the impact if it succeeds. This includes but is not limited to: minimizing design uncertainty, having multiple layers of security, limiting or restricting the critical interfaces of a system, controlling the information flow, etc. Hence, when developing countermeasures, it is of paramount importance to not only focus on the technical details of specific mechanisms but to follow a Systems Security Engineering (SSE) approach including technical and non-technical aspects.

In the early days of computers, i.e., the 1960s and 1970s, only trained personnel was allowed access to a computer. This was guaranteed by environmental and organizational security measures and provided the necessary assurance that no illegitimate user could access the system. Nowadays, we have ESs that often operate in a remote, unattended, and stationary environment, e.g., a smart-meter or Industrial Control Systems (ICSs). Another significant share of ESs are part of mobile applications and in proximity to the designated end-user, e.g., systems for autonomous driving or smartcards. Consequently, locking devices away to deny physical possession of the device is no longer a valid option for protecting these systems from malicious access.

To still meet the desired security objectives, it is common to use a set of interlaced security functions involving all fields of cryptography and systems security, e.g., to encrypt and authenticate data and to implement suitable software-based access control models. However, even when satisfactorily solved on a conceptual level, including schemes that are analytically secure, additional challenges arise from the practical implementation of the concept and its components. Unlike a software-based adversary that is restricted to given logical interfaces, is the physical adversary almost unconstrained in his access, i.e., within the constraints of the laws of physics and limitations of the equipment used, it is possible to carry out a large range of attacks. Developing corresponding countermeasures in hardware is a complex task, often depending on *fragile knowledge*, i.e., once it would be revealed how the countermeasure operates, it would be much less of a problem to circumvent it [82]. To a certain degree, this contradicts Kerckhoff's principle [110] which states that everything about a cryptographic system (in the sense of an algorithm) should be public, except its secret key, without diminishing the security.

Consequently, one of a system's most crucial assets to protect are the CSP. This is cryptographic key material such as secret keys of symmetric encryption algorithms but may also include the user's Personal Identification Numbers (PINs). Especially in Hardware Security Modules (HSMs), protecting the Key-Encryption-Key (KEK) is essential, i.e., a master key that is used to unlock other key material [147, 161, 162]. Unfortunately, storing CSP in Non-Volatile Memory (NVM) [63, 80] puts them at at risk, as memory contents can be extracted while the system is powered off [199, 176, 186], e.g., by delayering and optical analysis [203]. This is owed to the fact that the attacker can use every possible resource and time to slowly dissect the device and analyze its specifics to ultimately reveal the contained keys. Secure NVM technologies exist that provide a higher level of security but they are often not available in manufacturing processes outside of the smartcard industry. An alternative approach, as later on explained in more detail, is to store CSP in a Battery-Backed Random Access Memory (BBRAM), i.e., a volatile memory that can be erased instantaneously upon detection of a physical intruder. However, as can be deduced already, accommodating a battery and maintaining it in the system may not always be an option. Moreover, detection of a physical intruder requires active and continuous sensing of the device's physical integrity which hinders the shipping process. Hence, even better solutions are required that are secure and at the same time, do not entail the practical constraints of NVM or BBRAM-based approaches [151].

A promising approach to address this requirement are Physical Unclonable Functions (PUFs) [128, 67, 157, 48, 49, 158]. Once the device is powered-up, this security primitive derives a cryptographic key from the device's inherent manufacturing variations, i.e., the unique manufacturing variation of a device is leveraged as a fingerprint to create a kind of cryptographic seed. These manufacturing variations need to be measured and are thus subject to noise and environmental drift effects. Additional error-reducing and error-correcting schemes are necessary to remove these undesirable effects and yield a key of sufficient reliability and good cryptographic properties, e.g., a bit string with full entropy and insignificant failure rate. If based on Error-Correcting Codes (ECCs), this is typically called a *fuzzy extractor* but throughout this thesis, it is referred to as *key generation* to more generally include concepts that deviate from the original proposal of a fuzzy extractor [37].

PUFs are therefore considered a physical root of trust that supposedly provides a higher level of security when compared to permanent key storage in NVM or eFuses. This is based on the assumption that as long as the device employing a PUF is powered off, extracting its

minuscule manufacturing variations from the outside is not possible. A large body of work has been focusing on PUFs on a silicon level as a *component*, e.g., the PUF is a component of a SoC to store the key. In contrast, this thesis is directed towards a specific type of non-silicon PUF. More in particular, PUFs with the property of *tamper-evidence* [138], i.e., the property that evidence is left behind if it has been tampered with. Maybe somewhat surprisingly for readers not familiar with the topic can tamper-evident PUFs be used to not only store a key but at the same time limit physical access to a system, e.g., assuming the PUF is a *system-level* PUF that encloses the system either fully or to a larger degree, thereby obstructing physical access. Throughout this thesis, new concepts are being investigated associated with employing a *tamper-evident* PUF on a system-level, ranging from the physical construction, over sophisticated measurement techniques, to advanced algorithmic data processing algorithms.

Among other contributions, this resulted in the concept of Higher-Order Alphabet (HOA) PUFs, i.e., a PUF where the output is interpreted as symbols of a higher-order alphabet instead of bits that are often assumed independent and identically distributed (i.i.d.) in the PUF context. While these symbols will still be mapped to bits in typical computing architectures, their bits no longer fulfill the i.i.d. condition, thereby necessitating new techniques on how they can be used to derive a key and how the PUF output is evaluated. To the best of the author's knowledge, this is the first work on PUFs following this concept, as further detailed in the remainder of this thesis.

1.2 Problem Statement

Electronic products of low to medium quantity, e.g., in the range of up to 50 000 units per year, typically rely on Commercial-Off-The-Shelf (COTS) components. As a result, Small and Medium-sized Enterprisess (SMEs) and sometimes even governmental agencies are reluctant in developing Application Specific Integrated Circuits (ASICs) that include all the latest and greatest countermeasures. Moreover, the functionality of such a device often cannot be realized with just a single IC, i.e., several ICs contribute to the overall device functionality and ensure mandated performance if single-chip solutions, such as smartcards, lack performance or do not include necessary peripherals. Since hardware cannot be updated once deployed in the field, and considering the long development and manufacturing cycles of ICs, it is evident that once a new physical attack emerges, device security is at risk for several months to perhaps even years. Unfortunately, it is very difficult to counteract all possible threats by implementing various specifically-designed countermeasures at the IC-level. Even worse, implementing them in a new IC design requires verifying previous countermeasures yet again which is time consuming.

Hence, for highest security levels, additional countermeasures are required that limit an attacker's capability to perform attacks which often require physical proximity to the targeted device, e.g., as it is the case for advanced probing attacks and many other types of physical attacks [185, 148, 130, 64, 198, 129]. Numerous incidents, such as [101, 62] emphasize the strong need to develop countermeasures where no demonstrable way exists to bypass them. It is therefore common practice, in addition to IC-level countermeasures, to create a *physical security boundary* for MCMs that separates the insecure and secure domains of a device. This corresponds to the red/black concept, where classified plaintext information (red) is kept fully separated from ciphertexts (black). Several standards for security certification require this type of generic countermeasure to make follow-up attacks

more challenging to perform [147, 161, 162, 111]. As an example, to ensure compliance with FIPS 140-2 Level 4, a tamper-detection and response envelope with zeroization circuitry is mandatory that completely encloses the PCB in need of protection [147]. Systems protected by this type of countermeasure can be considered secure even when operated in a hostile environment.

Hence, the challenge addressed in this thesis is to devise methods of protecting MCMs from *physical* tamper attempts. This coincides with the goal of storing cryptographic keys or other data at rest securely but preferably, without battery-backed mechanisms. As sketched in Figure 1.1, creating a Three-Dimensional (3D) protected space, such as an MCM, requires considering attacks from arbitrary angles with any selection of tools. This is considered very challenging and as detailed in Chapter 2, very little public work is available on this topic compared to other fields in the security and cryptography domain. Specific PCBs security issues are discussed for example in [156] and include but are not limited to: in-field alternation, reverse-engineering or product-piracy, and hardware trojans. The designated mechanism to protect from tamper attempts therefore aims at preventing or slowing-down in-field alternation, distribution-chain interdiction attacks, and extraction of contained data/software to hinder reverse-engineering of the device.

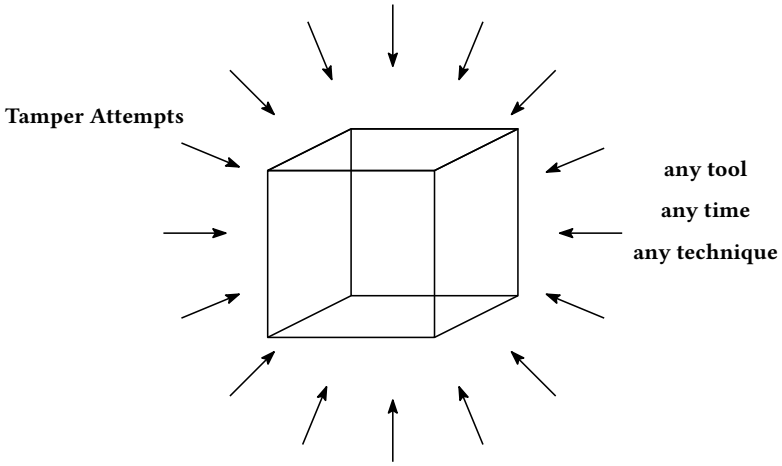


Figure 1.1: A 3D space in need of protection from tamper attempts. Throughout this thesis, this is considered an electronic "volume" such as a multiple-chip embedded module that must be protected from the adversary's attempts to operate, analyze, or exploit the module, i.e., tampering width the hardware and extraction of the contained data must be prevented or delayed significantly.

As indicated beforehand, PUFs could help to address some of these problems. However, since most PUFs are implemented in a newly made IC design, it is difficult to impossible to use them for aftermarket protection of COTS components, i.e., adding a PUF to an already manufactured IC usually cannot be done. Even when a fabric in the IC is available that can be exploited to also serve as a PUF, such as Static Random Access Memory (SRAM) [60], aging properties of such non-exclusively used SRAM is unclear or cannot be controlled properly. Furthermore, most silicon based PUFs typically do not have the property of ensuring a system-level tamper-evidence [138], i.e., once powered on, they cannot verify if an attack was executed on other parts of the system while powered off. Without additional countermeasures, such as IC-level meshes, they are incapable of detecting online attacks that extract values during runtime [65], e.g., as it would be the case when an SRAM-PUF

transfers values over a data bus that is being actively probed. With an increasing number of advanced probing attacks [185, 148, 130, 64, 198, 129] that often originate from IC failure analysis [189, 238], also via the backside of the IC, it is evident that applications requiring the highest level of security require these strong complementary countermeasures at the system-level. As direct physical access is then limited, susceptibility to side-channel attacks is effectively mitigated, which otherwise can be carried out on certain types of silicon-PUFs too [143, 197, 200]. Several more physical attacks such as Laser Fault Injection (LFI) would also be much more difficult to perform as gaining access is then more likely to result in an already sufficient destruction of the device.

Clearly, if there is some logic involved in evaluating either the PUF or any other type of physical security boundary, it must protect itself from attacks, too. If a PUF can be designed and manufactured appropriately such that it provides tamper-evidence, it can serve as such a physical security boundary. This is based on the following observation: if data contained in the device is encrypted using a key derived from the physical properties of its security boundary, then breaking or otherwise damaging the boundary will alter its properties, causing the key derivation ultimately to fail and rendering the data inaccessible. Hence, as long as the boundary is designed well-enough, accessing the contained data by physical means will be practically impossible, resulting in *read-proof* data [53]. Unlike previous approaches in the domain of physical security boundaries, they offer the intriguing benefit of not requiring a battery-backed evaluation logic as discussed in Chapter 2.

1.3 Definition of Terms

Protecting critical information of military equipment has been an important topic early on [109, 23, 82]. The United States (U.S.) Department of Defense (DoD), probably like many other countries, therefore maintains an organizational unit dedicated towards the protection of such information. This is the Anti-Tamper (AT) organization led by the Anti-Tamper Executive Agent (ATEA). As the required protection mechanisms include aforementioned hardware-based countermeasures to limit or restrict physical access, there appears to be a substantial amount of knowledge available in that community which however is inaccessible by the scientific community, as there are only very few publications on that topic. Apparently, this is in contrast to other topics such as cryptography and cryptanalysis, where even public competitions were organized to select follow-up encryption algorithms, e.g., as it was the case for the Advanced Encryption Standard (AES). Even proper definitions of some terms related to hardware-based countermeasures are often missing or incomplete. For the term "anti-tamper" we therefore refer to the definition of the ATEA which defines its own naming as follows:

> **Anti-tamper (AT):** "*Systems engineering activities intended to prevent or delay exploitation of Critical Program Information (CPI) in U.S. defense systems in domestic and export configurations to impede countermeasure development, unintended technology transfer, or alteration of a system due to reverse engineering. (DoDI 5200.39).*
>
> *Properly employed, AT will add longevity to CPI by deterring efforts to reverse-engineer, exploit, or develop countermeasures against a system or system component.*

> *AT is not intended to completely defeat such hostile attempts, but it should discourage exploitation or reverse-engineering or make such efforts so time-consuming, difficult, and expensive that even if successful, CPI will have been replaced by its next-generation version.*" [2]

In short, AT aims at deterrence, prevention, and detection of the threats caused by attempted exploration and exploitation of electronic systems in addition to response upon detection. Several white papers from the industry [5, 6, 16, 165] picked up this term to describe the set of interlaced countermeasures available in their commercial platforms. This often includes interaction with "tamper-detection and response" mechanisms [147, 161, 162, 111], or more generically tamper-protection mechanisms, whereas one of the earliest attempts to systematically define these and corresponding terms has been made in [224].

These tamper-detection and response mechanisms are referred to by many different names, mostly to describe a sophisticated mechanism that surrounds the actual device to detect physical intruders and initiate a suitable response, e.g., zeroization of data which is stored in BBRAM. The terms used to describe such mechanisms include but are not limited to: cryptographic or physical security boundary, enclosure, housing, shell, box, envelope, cover, volume protection, proximity sensor, proximity fuse, hardware access denial system, tamper-resistant barriers, etc. Unfortunately, authoritative definitions for these terms are often not available. The author of this thesis likes to think of it in the following way: Volume Protection (VP) is a security objective, whereas an Access Denial System (ADS) is the abstract superset of specific technical means to achieve that security objective. In the following is an attempt to define these terms:

Definition 1.3.1 (Volume Protection) *Defines the physical security objective to achieve protection from any adversarial physical alteration of a given electronic volume, e.g., an MCM. Here, protection is interpreted as the process of resisting or additionally of actively preventing such attempts. Moreover, VP specifically includes the aspect of hindering exploration of electronics contained in the volume, i.e., by hindering access and avoiding sensitive emanation.*

Definition 1.3.2 (Access Denial System) *Defines the technical means to resist or prevent physical intrusion and exploration attempts to counteract proximity based physical attacks. This may include the option to detect and respond to attacks.*

Both definitions are phrased such that the security objective of VP may be achieved by either active, passive, or hybrid ADSs. Hence, the scope of this definition is not constrained by specific or idealized implementations. Active ADSs could be based on a mesh that surrounds the Module Under Protection (MUP) and that is continuously monitored by a battery-backed evaluation circuit. Alternatively, it may only work when the device is powered on which would severely impact the scope of the provided protection. These types of ADSs are therefore likely to be based on some type of proximity sensor and depend on supplied energy. In contrast, a passive ADSs could be based on thick steel, coating, or potting material, i.e., a countermeasure that is independent of the operating state of the MUP and does not require an energy supply to provide protection.

Hybrid ADS, as the name implies, are somewhere in between. While the MUP is powered-off, they are not allowed to draw energy. Once the device is powered-on, they require energy to provide protection until the device is powered off again. Tamper-evident PUFs therefore fit this category, as they are designed to not require energy while the device is

powered off. However, once powered on, an evaluation logic is required to measure the PUF's physical parameters and to process the resulting data to yield a cryptographic key.

This classification may fall short when it comes to contact explosives, brittle or water soluble material, spring guns, and the like, as they would be considered a passive ADS. Hence, it should be added that independent of the chosen type of ADS, the desired goal is always to enable a *self-determination* of the MUP that it has not been tampered with while powered-off and that it is not actively under attack while powered-on. This rules out tamper-indicating mechanisms such as tamper-evident seals that also do not fit the given definition of VP. Consequently, they must *not* be considered a type of ADS. In the following, specific examples of ADSs are provided. An ADS may be …

- …based on a security enclosure, e.g., created from a housing, box, cover, or envelope.
- …tamper-resistant if it is a physical barrier such as potting material or thick steel.
- …providing tamper-detection and response when based on proximity sensors.

Please note that the proposed definition of an ADS deviates from the definition of a cryptographic boundary given in FIPS 140-2 [147] which is described as "*an explicitly defined continuous perimeter that establishes the physical bounds of a cryptographic module and contains all the hardware, software, and/or firmware components of a cryptographic module*". The differences are, e.g., in the architectural understanding of how the device is structured and the definition refers to HSMs only. Moreover, the definition of FIPS 140-2 does not explicitly state to what extend the boundary protects from physical tampering. At higher protection levels, a tamper detection and response envelope is required, which already implies that a battery-backed monitoring circuit is used. This is not surprising, since this standard has apparently been founded on ideas provided by the same work group who initially developed the so called GORE envelope (cf. Chapter 2). In the following, several more security concepts and basic terms are introduced. This is complemented with suitable references for the interested reader.

Basic Security Concepts: Well-written literature is available on various aspects of cryptography and security that are relevant to this thesis, e.g., [133, 155]. Figure 1.2 illustrates how some of the fundamental building blocks and concepts are linked together. From bottom to top, we have the physical world with physical security primitives such as secure logic styles to prevent leakage of processed secrets via the power side-channel [141]. For advanced key storage without explicitly storing the secret in a data memory, there are PUFs. As introduced beforehand, an ADS should be considered as yet another physical security primitive. These physical security primitives provide roots of trust based on physical phenomena, i.e., attackers trying to circumvent these mechanisms are therefore subject to the constraints of the physical world. By leveraging such physical roots of trust, physical security objectives can be achieved, i.e., the idea is to build security from the ground up by building upon security mechanisms deeply rooted in the physical domain that would require superior expertise and expensive tooling to overcome.

One physical security objective is VP to obtain a device which basically prevents any physical access that could turn out useful to an attacker. This covers the complete range of losing a device, theft, obtaining access with the help of defectors or a colluding party, etc. Additional objectives are *secure bootstrap*, i.e., the challenge of securely initializing a

device if it has been powered-off. Re-establishing trust in a device as part of the secure bootstrap is an extremely challenging and interesting topic, e.g., considering tightened border controls where physical control of the device cannot always be ensured.

Once physical security is ensured, it is possible to securely implement cryptographic primitives to achieve information security objectives, i.e., secure key storage and secure execution of these analytically secure cryptographic primitives would then be ensured. For example, by applying block or stream ciphers on data, confidentiality is achieved. This is however considered outside the scope of this thesis.

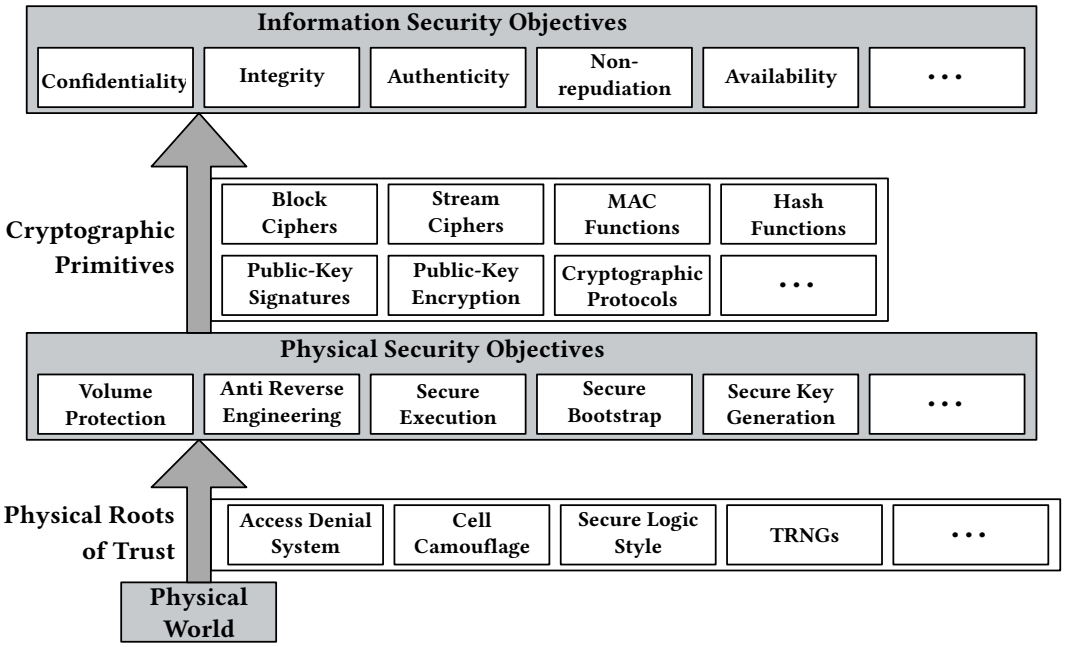


Figure 1.2: Relation between information security, cryptography, physical security and physical roots of trust. Figure adapted and extended from [133].

Taxonomy of Physical Attacks and Physical Security: Efforts to provide a systematization of attacks have been presented in [168, 189, 226]. In particular the work of Weingart [226] should be considered as influential, as the author presumably participated in the development of the draft that later resulted in the FIPS 140-1 security certification standard, the earlier version of the FIPS 140-2 standard [147]. In addition to that, the author was apparently involved in developing the solution known as "GORE envelope" (cf. Chapter 2), a formerly available commercial solution that was compliant with FIPS 140-1 level 4 overall and that should still be considered representative for the commercial state of the art in physical security design (despite the fact that it has been discontinued as a product). His overview [226] on possible attacks and corresponding laboratory equipment therefore still provides an excellent overview to get started on the topic. Regarding the terminology of tampering, this work slightly deviates from the definitions presented by Weingart et al. in [224]. In general, a device that counteracts physical attacks is called tamper-protected or tamper-resistant. Here in this work, tamper-resistance is considered a property on its own, e.g., if size of a device, its complexity, its weight, or a physical barrier such as potting material make tampering with the device more difficult. However, at the same time, it is

interpreted as a superset to other properties related to tampering, e.g., the properties of tamper-evident and tamper-respondent are a subset of tamper-resistance. Hence, unless specified in more detail, devices are simply called tamper-resistant which may also include tamper-detecting or tamper-responding features. With regard to tamper-evidence, it should be noted that the concept of tamper-evident PUFs exceeds older definitions of tamper-evidence [224], i.e., optical inspection as part of auditing the device is no longer required and they can actually be employed as part of a system that detects and prevents attacks. As a result are seals and bleeding paint considered as tamper-indicating mechanisms [28] that require periodic inspection. They are considered outside the scope of this work.

Basic Physical Unclonable Function (PUF) Terminology: In short, a PUF represents a physically-bound function that is easy to evaluate in a reproducible manner but hard to predict [49, 142]. To achieve this behavior, the uniqueness that stems from uncontrollable manufacturing variation of a physical object is leveraged. A more formal introduction is presented later in Chapter 3. Since a couple of notable authors worked in the domain of PUFs, a slightly deviating understanding exists of what a PUF is and correspondingly differ the terms, too. In this thesis, the term *construction* of a PUF primarily refers to the physical and analog-circuit level aspects that constitute the PUF, i.e., how the architectural physical design and corresponding manufacturing process is done such that the desired uniqueness can be expected from the measured output. Since the measured output is subject to noise and environmental drift effects, it must be processed to yield a stable cryptographic key. This is the algorithmic processing called key derivation. The resulting *properties* of both the physical construction and the algorithmic processing must then be analyzed. Since PUFs exist to serve a specific purpose, their intended *application* is important to consider, which is another part of this thesis. Hence, this thesis focuses on the constructions, properties, and applications of PUFs, in particular tamper-evident ones, in addition to their algorithmic processing to generate a cryptographic key.

1.4 Research Scope

The purpose of this research is to develop new concepts and techniques for tamper-evident PUFs to create an ADS at system-level. Ultimately, this is intended to overcome the practical limitations of previous PUF constructions and battery-backed mechanisms for volume protection [151]. We call the resulting concept a HOA PUF, as its output is no longer represented by a binary alphabet (i.e., zeros and ones) but instead, as symbols of a higher-order alphabet*. This necessitates the development of new PUF metrics beyond the scope of Hamming Distance, corresponding error-correcting schemes, and extending existing evaluation criteria, etc. Hence, the contribution of this thesis is primarily based on these generic concepts to successfully improve tamper-evident PUFs and not about the specifics of an implementation. They merely serve as an example to prove that the developed schemes are useful and relevant for real-world problems.

* While symbols of a higher-order alphabet will still be stored as binary data in commonly available computing architectures, their interpretation is solely based on the meaning of the symbols, not the bits.

1.4.1 Design Aspects of Access Denial Systems

In general, the assessment of any ADS, e.g., a security enclosure, is subject to three criteria, namely: producibility, usability, and security as illustrated in Figure 1.3. It is therefore not possible to fully separate these aspects and consider them on their own. They are explained in more detail as follows:

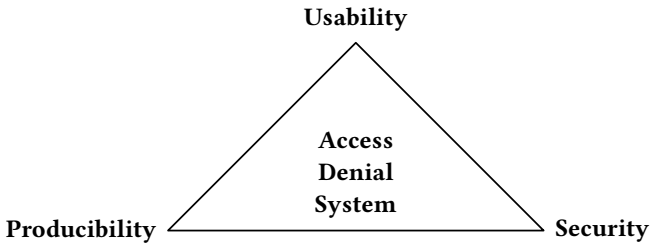


Figure 1.3: High-level design goals of an Access Denial System (ADS).

Producibility At some point in time, an ADS needs to be manufactured. In particular for a PUF-based security enclosure this can be a non-trivial task that is outside commonly available manufacturing technology and capabilities. This possibly entails higher costs and/or a lower yield, making it less desirable to use it in a real-world scenario. Hence, special attention should be paid to if the enclosure can be manufactured with a moderate effort by multiple independent parties to avoid single-source supplier problems such as trust issues, a price monopoly, etc.

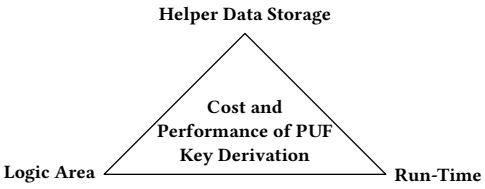
Usability Even prior to manufacturing the device, incorporating the ADS into the design should not cause much engineering overhead to ease adoption. After the ADS is manufactured, it should be easy to apply to the designated system, i.e., the assembly process should not require customized tooling or significantly increase production time. Moreover, once the device is fully assembled, there should be a mechanism to verify the integrity of the assembly process from inside of the device for security reasons. Once the system is armed, i.e., protected and/or enclosed by the ADS, it should withstand environmental conditions within the targeted operating window and survive prolonged storage, e.g., a total life span of 10 years may be considered a minimum for some applications.

Security Of utmost concern is the security of an ADS, as it must sufficiently protect the system from attacks it has been designed for and those which have not been anticipated. Moreover, it must withstand attacks on itself, i.e., attempts to attack or circumvent the ADS, its evaluation logic, or its link to the protected system must be prevented, too. For an enclosure of any type, this should ideally be based on strong and convincing reasoning with regard to the selected material composition, the overall physical construction such as geometric considerations, a stochastic model for the entropy (if applicable), estimated cost of tooling and expertise of the attacker, etc. This must specifically include attacks, attempted repairs, and an analysis on the possible limits of degradation in security due to undesired effects in the manufacturing process.

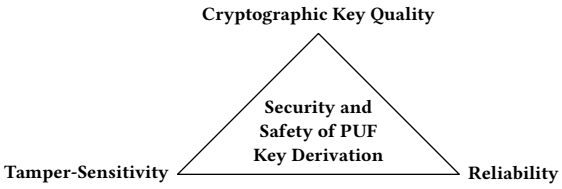
1.4.2 Design Aspects of PUF Key Derivation

Since the type of ADS considered in this thesis is based on a PUF, special attention needs to be paid to the algorithmic part that processes the raw physical output data of the PUF up to the point where a cryptographic key is generated. This process is called PUF key derivation. With regard to binary PUFs, a significant amount of work was done by Maes [133] and Hiller [72], in particular with a strong focus on implementation aspects (cf. Figure 1.4a) of PUF primitives and their corresponding algorithmic part. This is in contrast to Tajik [196] where the physical (in)security of silicon-based PUFs was analyzed.

The design goals of the PUF key derivation are illustrated in Figure 1.4. In Figure 1.4a, the implementation aspects of the PUF key derivation are illustrated. They mostly focus on the efficiency of the implemented scheme with respect to the utilization of hardware resources such as logic area and the resulting performance, e.g., run-time and energy consumption. In contrast, Figure 1.4b, focuses on the safety and security of the key derivation, e.g., how reliable the derived key is to not inadvertently cause device failures, how good its cryptographic quality is, and most importantly for a tamper-evident PUF, how sensitive the key responds to attempts of tampering with the PUF. In particular the latter is a newly developed aspect that is addressed in full detail in Part III of this thesis. These aspects are additionally summarized in the following:



(a) Implementation aspects of PUF key derivation.



(b) Security and safety aspects of PUF key derivation.

Figure 1.4: Design goals of PUF key derivation algorithms and corresponding trade-offs.

Logic Area For hardware implementations of PUFs and corresponding key derivation schemes is the hardware resource utilization important, e.g., how many logic gates and Flip-Flops (FFs) are required to implement the scheme and the PUF primitives. This covers both ICs and Field Programmable Gate Arrays (FPGAs). For software-based implementations this would be interpreted as register and memory usage of the program code.

Helper Data Storage To enable error-correction and algorithmic error-reduction schemes, additional non-volatile data needs to be stored. This *helper data* is stored permanently and adds to the implementation complexity of the key derivation scheme. Preferably, the memory requirement of this data is limited to keep the implementation resource efficient and avoid security risks associated with storing the data, i.e., so called helper data leakage [89, 35, 91] and helper data manipulation attacks [36]. However, helper data leakage within the context of implementation cost is primarily an efficiency issue, *not* a security issue, as the loss in entropy can be accounted for by more PUF cells from which the entropy is drawn.

Run-Time Depending on functional or security requirements, there may be run-time constraints that limit the possible choices of how the key derivation is implemented, e.g.,

with respect to timing, energy consumption, etc. For example, the security policy of the Utimaco HSM CryptoServer Se-Series Gen2 [208] states that the battery-backed tamper-detection and zeroization circuitry responds within just 4 ms to tampering with the device, i.e., taking into account some time for carrying out the zeroization, the PUF key derivation would need to be performed within an even shorter duration than 4 ms.

Cryptographic Key Quality The key quality mainly depends on the number of effective bits with full entropy. This is controlled by the raw entropy that can be extracted from the PUF and the secrecy leakage caused by *helper data leakage*, i.e., depending on the structure of the key derivation and the type of helper data stored, it is possible to deduce information from the helper data to gain knowledge about the key derived from the PUF. Hence, properly designed key derivation schemes should not diminish the raw entropy extracted from the PUF.

Reliability As PUFs are based on fuzzy data, i.e., data that is slightly different for each read-out due to noise and environmental drift effects, it is important to consider the probability of a device to fail. Even traditional NVM entails a certain failure probability and ideally, a PUF provides a similar failure rate. In general, the odds of a failing device should be smaller than 10^{-6} which is a common baseline in PUF literature. Error-reduction techniques and ECCs are commonly applied to counteract effects that would otherwise cause device failures.

Tamper-Sensitivity Within the context of physical attacks, deficiencies in the PUF data may not only be caused by the lack of reliability but also by attempted tampering. The capability of a system to carry out a self-determination that it has been tampered with is called tamper-detection. If reliability enhancing mechanisms have been made too powerful, then damage from physical attacks could be mistaken as errors from insufficient reliability. To describe the quality of the tamper-detection while still ensuring sufficient reliability, we define the term tamper-sensitivity with a corresponding metric as later introduced in this thesis. Intuitively, it is not possible to maximize all given design goals at the same time.

1.5 Thesis Setting and Project Background

The Fraunhofer Society is a German research organization focused on *applied* sciences and the work presented in this thesis has been carried out by its author in collaboration with several colleagues at the Fraunhofer Institute for Applied and Integrated Security (AISEC) and in cooperation with Fraunhofer IMS and EMFT between 2013 and 2018. It is based on preliminary ideas envisioned by former AISEC coworkers, as presented in [66]. Back in 2013, the initial idea was to develop a foil to prevent extraction of protected data from embedded systems by means of a wrappable film or flexible sheet that reacts to tampering in a sensitive manner such that a previously derived unique fingerprint of the foil can no longer be reconstructed, i.e., a tamper-evident PUF contained in the foil representing the Access Denial System (ADS).

However, it quickly turned out that a sufficient level of protection could only be achieved with a more thorough R&D effort of each individual aspect of the targeted system. Shortly after publishing [66], my colleague Maxim and I jointly started this task, in parallel to acquiring project funding and working on other projects. Over the course of two years,

several preliminary designs of such a foil and corresponding system architectures were envisioned, ultimately resulting in more specific ideas for the project acquisition and a requirements engineering process. In parallel, we were already approached by prospective customers seeking a replacement for former battery-backed tamper-respondent envelopes, indicating the strong need for a generic high-security enclosure that is mostly independent from the security features implemented in the contained FPGAs or ICs.

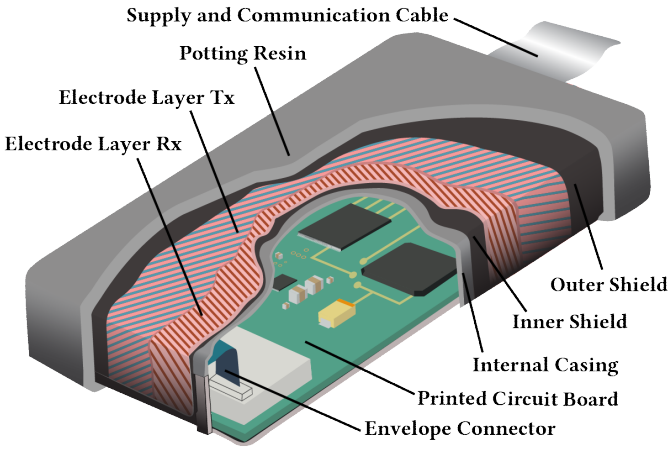


Figure 1.5: Drawing of the design goal of this thesis: a batteryless tamper-resistant enclosure to protect multiple-chip modules from physical tampering.

The major building blocks of the then designated system as illustrated in Figure 1.5 are: a physical enclosure made from an envelope (or cover), a measurement circuit, and tailored algorithmic processing. With this idea and as a result of a competitive process within the Fraunhofer Society, we were then able to acquire an internally-funded project named "COPYCAT", a multi-year effort from 2015 to early 2018 that was carried out collaboratively by Fraunhofer IMS, Fraunhofer EMFT, and Fraunhofer AISEC. The majority of academic work as part of this thesis has been a direct result of the COPYCAT project*. Its structure with its main topics and corresponding project responsibilities is sketched in Figure 1.6. It was funded by the Fraunhofer internal programs called "MAVO", a German acronym for "MArktorientierte VOrlaufforschung" which translates to "market-oriented preliminary research". "MAVO" aims at enabling research in areas that require a joint effort of different research domains and thus, at least two Fraunhofer institutes. Moreover, the conducted research should target specific business cases with a mandatory Return-On-Investment (ROI). While projects of this type are designated to minimize the gap between academic ideas and a later productization, they are still mostly geared towards solving the underlying research challenges rather than creating a final product.

To outline the basic system concept and overall architecture within the scope of COPYCAT, I then authored 64 out of 71 requirements for the initial device specification. In addition to my technical contributions throughout the project, I was promoted to be the project lead for COPYCAT at Fraunhofer AISEC in late 2015, a role that I then carried out successfully until the end of the project. Beyond the scope of this internally funded project, I participated in various other industry projects some of which were related to the topic of tamper-resistance.

* This work was supported by the Fraunhofer Internal Programs under Grant No. MAVO 828 432.

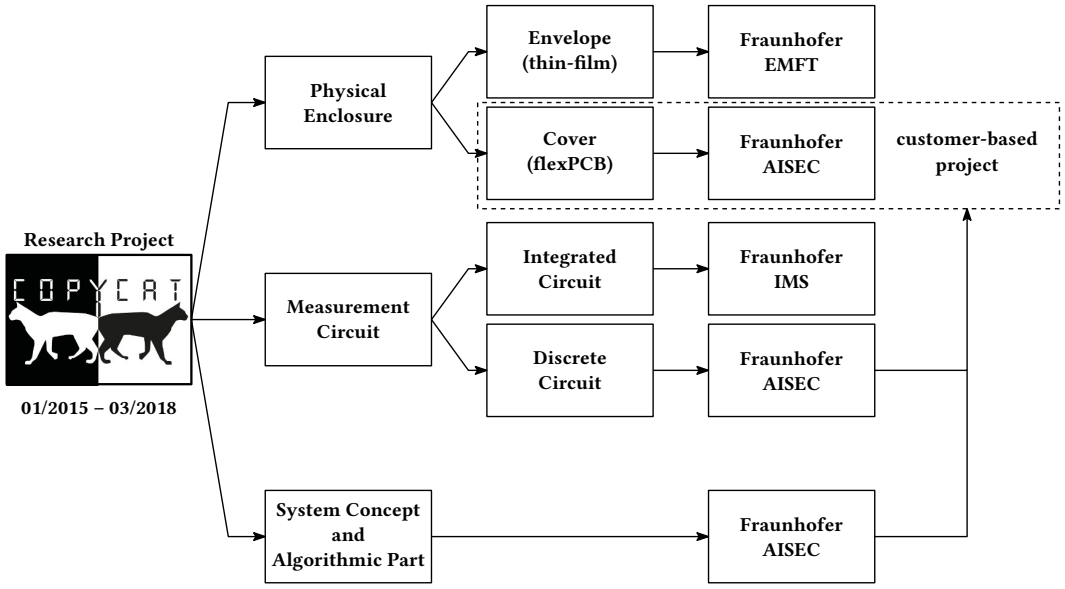


Figure 1.6: COPYCAT project structure outlining the collaboration and topics.

This joint effort by Fraunhofer EMFT, Fraunhofer IMS, and Fraunhofer EMFT resulted in several manufactured batches of envelopes, custom made ICs, more than 10 peer-reviewed publications on the topic of tamper-protection, and several patent filings. The PhD thesis by Johannes Obermaier, who joined the project around April 2016, in particular his work on the measurement circuit significantly contributed to the project and this thesis. Complementary work on PUFs at the IC-level was carried out by Fraunhofer IMS and Fraunhofer AISEC and added additional value to the project.

Since this thesis is only a starting point for research on tamper-evident, system-level PUFs, future advancements can be expected from the continuation of this line of work. Ultimately, this will help replacing formerly battery-backed anti-tamper mechanisms.

1.6 Thesis Outline and Summary of Research Contributions

Additional preliminaries of this thesis are presented in the remainder of Part I, in particular the application context and previous work on tamper-resistant enclosures that are based on battery-backed sensing mechanisms to detect tampering. All other parts of this thesis then cover the full stack of how to build and analyze a tamper-evident, system-level PUF. This ranges from the architectural construction presented in Part II, to the algorithmic data processing in Part III, followed by analyzed properties in Part IV, and conducted case studies in Part V. All parts follow a mostly data-centric view based on the concept of a Higher-Order Alphabet PUF. This is owed to the fact that the underlying concept for the designated PUF was developed by the thesis author at AISEC, while a substantial amount of the engineering and technological effort was carried out by Fraunhofer EMFT and IMS. Eventually, this work is concluded in Part VI. In the following, each part is summarized in more detail in addition to the overview provided in Table 1.1.

Table 1.1: Outline of this thesis, its topics, and summary of research contributions.

Part	Topic	Related Publications
I	Preliminaries	[151], [96]
II	Higher-Order Alphabet PUF Construction	[95], [97], [152], [42]
III	Reliability Enhancement Techniques for PUFs	[91], [92], [100], [93]
IV	Properties of Higher-Order Alphabet PUFs	[94], [97], [100], [164]
V	Case Studies and Applications	[95], [97]
VI	Conclusion	

Part I: Preliminaries The preliminaries include two chapters, whereas Chapter 1 includes the motivation to deal with the presented topics, the addressed problem statement, a basic definition of terms, and explains the research scope of this thesis. Moreover, the thesis project setting and outline are explained. Since the Fraunhofer Society focuses on applied sciences, special attention is paid to the application context in Chapter 2. This covers a brief overview on the history of publicly known tamper-resistant enclosures. In addition to that, a selection of commercial HSMs products and corresponding tamper-resistant enclosures is studied. This is complemented by a description of standards for security certification that mandate this kind of countermeasure. If a product is not compliant to these security certifications, it may not be used for the intended purpose due to legal restrictions, requirements imposed by insurance companies, or industry associations.

Part II: Constructions Here, first an overview and analysis of existing PUF constructions is presented in Chapter 3. This includes common PUF definitions and a classification of PUF constructions based on certain design principles and features shared among all constructions. Atferwards, in Chapter 4, a construction is proposed for a tamper-evident, system-level PUF. This is later used for an envelope and a cover to carry out a case study. The proposed construction includes the physical, analog, digital, and application domain, i.e., the full stack from bottom to top. Each of these domains entails solving several challenges on its own which is why the focus is on the core principles of the proposed approach in this chapter only. Some of these aspects are further detailed in the referenced work. Since the

development of the digital data processing was the sole responsibility of the thesis author, it is presented in full detail in Part III. The developed principles applied within the context of tamper-evident PUFs can also be used for other types of PUFs and are of general nature.

Part III: ECC Schemes This part builds upon the previously presented PUF construction in Part II. In Chapter 5, an overview of reliability enhancement techniques is presented, as ensuring reliability is a major design challenge for any PUF implementation. As part of this overview, existing approaches for this task are reviewed and reasoning is provided as to why they operate inefficiently on the resulting data of the designated PUF construction. A strong focus is then put on the type of quantization scheme as a first error-reducing technique prior to an ECC in Chapter 6. Afterwards, different approaches are investigated for ECCs that are a common building block of fuzzy extractors. One of the approaches is based on a symbol to variable-length bit mapping as detailed in Chapter 7. An even better approach that continues operating on symbols is then presented in Chapter 8. All schemes are then compared based on simulated data in Chapter 9. Conclusions on ECC schemes are then drawn in Chapter 10.

Part IV: Properties and Evaluation Criteria In Chapter 11, various properties of a PUF construction are identified and studied to ensure sufficient confidence in the PUF design. Two basic properties include the popular PUF metrics named *Uniqueness* and *Reliability*, i.e., each PUF device must be sufficiently different from others of the same type but at the same time, they must be robust over time towards environmental influence to ensure proper PUF functionality. These two basic metrics previously have been solely used in combination with the Hamming Distance (HD), i.e., differences between devices (inter-device distance) and over time due to environmental influence (intra-device distance) are counted in terms of the bit differences in binary representation. To better reflect the ECC approaches presented in the previous part of this thesis, new distance metrics must be used and the definitions of these two basic properties updated correspondingly.

Part V: Case Studies and Applications This thesis has been driven by project work resulting in Proof-of-Concept (PoC) implementations of the design principles presented in Part II. These PoC implementations are named B-TREPID [95], SPECTRE [97]*, and FORTRESS† (*to be published*). In this part, the practical results are presented, covering a thorough statistical assessment of the contained PUF, a practical physical security analysis, and environmental testing.

Part VI: Conclusion This part summarizes the results of this thesis and concludes it. Moreover, since the case studies support the overall design rationale of the chosen approach, they facilitate future research. Ideas and left-over work from this thesis are therefore briefly introduced as future work.

* The acronym "SPECTRE" was later dropped from the title of the corresponding publication, due to coinciding with the timing side-channel attacks based on speculative execution branded under the same name.

† FORtified Tamper-Resistant Envelope with Embedded Security Sensor (FORTRESS)

Chapter 2

Application Context

This chapter elaborates the application context of this thesis, i.e., why this work is useful and of practical relevance. As part of that, we provide an overview on physical security enclosures, i.e., protection mechanisms that are designated to slow down or prevent a physical intruder. In addition to that, corresponding standards for security certifications are briefly described to address the regulatory needs and certification requirements. This chapter is based on unpublished project work by the thesis author and joint work published in [151] as co-principal author. For that work, Johannes Obermaier primarily analyzed the IBM Crypto Coprocessor. The analysis of HP Atalla Cryptographic Subsystem was primarily done by the thesis author. Writing the paper was done in a highly collaborative manner.

Contents

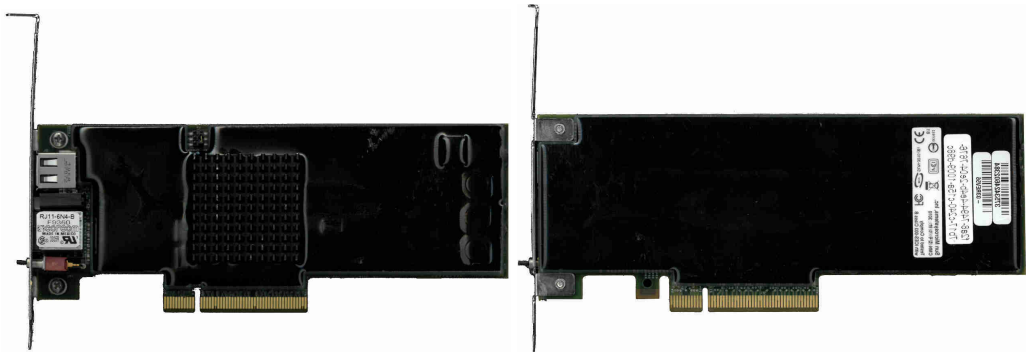
2.1 Protection From Physical Attacks

Early examples to prevent physical tampering and ensuring unattended operation of electronics date back at least until the early 1970s. Back then, safeguarding of special nuclear material and installation of corresponding tamper-resistant instrumentation was one of the applications driving the development [28]. The subsequent advancements with some of the notable public records (as perceived by the thesis author) are then covered in Section 2.1.1. Afterwards, in Section 2.1.2, real-world physical security examples are presented, i.e., formerly commercially available MCMs that include different levels of tamper protection. With respect to the general design goals of an ADS, as stated earlier, should these mechanisms provide sufficient fragility towards tampering to enable detection of attacks, provide a thorough scope of protection and not leave parts of the system unprotected, ensure a high reliability and ruggedness of the assembled system while at the same time not being susceptible to environmental effects. Moreover, complexity of device assembly, i.e., level of automation, time and cost, should be low. If possible, brief statements are included

in the description of these commercial mechanisms with regard to these aspects. This is followed by a summary of the findings and general drawbacks of the presented examples in Section 2.1.3.

2.1.1 History of Tamper-Resistant Enclosures

This section follows the timeline of Figure 2.2 and starts with the publication [28], outlining basic design concepts of tamper-responding hardware. The included high-level concept describes an aggregate layer of detectors, protectors, and barrier particles. Clearly, the proposed concept focuses on material properties that, based on the previous classification of ADSs, fall into the category of passive systems. Another passive system is illustrated in Figure 2.1, showing a HSM by the former company Sun Microsystems. It serves as a practical example for many other devices where protection is mainly based on opaque encapsulation material/potting, as required by FIPS 140-2 Level 2. Gaining access to the device then requires tailoring solvents to the chemical properties of the potting, using appropriate drills, milling, or other types of mechanical machining, etc. However, given sufficient time and expertise, it is difficult to fully deny physical access to the system by using this method alone.



(a) Front side of Sun Microsystems HSM with opaque encapsulation material.

(b) Back side view (mirrored) of Sun Microsystems HSM with encapsulation material.

Figure 2.1: Example of a passive ADS.

A physical security mechanism based on active sensing was later presented as μABYSS in [223] by Weingart. It is based on a MCM that is wire-wrapped using four layers of fine, thinly insulated nichrome wire which is then potted. The resistance of wire strands is measured and how they connect to the circuit can be configured, thereby enabling a wire-layout configuration that results in a physically permuted ordering of the strands while retaining the same electrical configuration. Fragility of the sensing element upon attempts to physically tamper with the package is guaranteed by the chosen approach. Furthermore, a high density of the wire-wrapping leaves no spot or hole of the MCM unprotected. The development of this physical security mechanism has been complemented by a whole system architecture, as described in [224, 3, 38]. This concept apparently has been the predecessor of the GORE envelope that is covered in more detail in Section 2.1.2. At the same time around 1987, other authors covered the topic of tamper-resistant hardware and basic engineering considerations related to it [31].

Afterwards, several other publications and patents can be found describing tamper-resistant or tamper-responding hardware mechanisms. Typically, these physical intrusion

detection mechanisms exceeding the scope of simple micro-switches fall in either one of the following categories: *i*) mesh or node based, i.e., a boundary is constructed that is made from wires or other types of nodes that directly connect to the measurement circuit; *ii*) backscatter based, i.e., physical material with reflective properties either directly covers the circuitry or is used as a lining for the inside of an enclosure or box.

Selected approaches to detect tampering and that are of different physical nature and *not* related to PUFs are: a piezoelectric token [83], a planar waveguide optical tamper-detection system [26], several examples of mesh/resistance based solutions including but not limited to [25, 20, 46, 179] and in particular the GORE envelope [131, 218, 102, 132, 27], PCB-internal arrangements of vias or other structures [43, 166, 153], a fringe-effect capacitive proximity sensor [40], and detection of PCB-level tampering with the tracks to prevent "mod-chip" insertion between ICs [156].

Let us briefly consider some of these approaches in slightly more detail. The fringe-effect capacitive proximity sensor presented in [40] is one solution that claims conformance to FIPS 140-2 level 4. Its security concept is based on electrodes whose capacitive coupling is analyzed by a monitoring circuit. As long as the system is not under attack, the capacitive coupling remains constant. In case the enclosure is tampered with, the intruding object interferes with the electric field and causes a change in capacitance. This is detected by the monitoring circuit which consequently triggers the zeroization of all CSPs. For retaining the CSPs and continuously providing protection, a battery is incorporated into the system.

Most other approaches almost exclusively monitor the ohmic resistance of traces. One such example is the "Security Housing" by Bourns Inc. [25, 20]. It is based on plastic or ceramic covers that contain one or more layers of conductive traces. This cover is then mounted on top of a PCB after manufacturing to enclose the components underneath. Their former commercial brochure [20] states a protection from drills down to 500 µm. To resist other types of tampering, the traces are reportedly manufactured such that it is not easily possible to electrically contact them. Please note that covers of similar type are still available from several other manufacturers today.

In parallel to advancements in the domain of traditional tamper-resistant technology, PUFs where conceived. Early versions of PUFs include [116, 128, 108], i.e., concepts clearly following the concept of a PUF without explicitely using this term since it was not established as such at the time [157]. Of particular interest are [116] and [108] since they aim at a system-level anti-tamper capability which exceeds the scope of circuit identification as targeted in [128] or cryptographic key generation as primarily the case in [49, 48]. Later publications then focusing on tamper-evident PUFs are: most notably the "Coating PUF" [206, 184, 178, 172, 52], the "Cocoon PUF" [119, 118], an optical security check box called UMABASA [41, 120], and the "Polymer Waveguide PUF" [190, 209, 211, 51].

In the following, the focus is on academic publications rather than patents. The "Coating PUF" [206] protects a whole IC by covering its top with a randomized coating material, which is measured to extract its unique properties and to derive a secret key. Reconstructing this key is infeasible if the coating has been damaged due to an attack. A similar approach using an optical backscatter PUF is presented in [41]. Both approaches do not specifically address attacks during runtime, i.e., how to avoid a repetitive key generation from the unique properties to determine physical integrity of the PUF. Furthermore, covering *every* IC of an embedded device with a coating requires a costly, fully customized sourcing of its components. Moreover, direct access to the PCB would still be possible and therefore simplify various attacks, e.g., voltage glitch or power side-channel attacks.

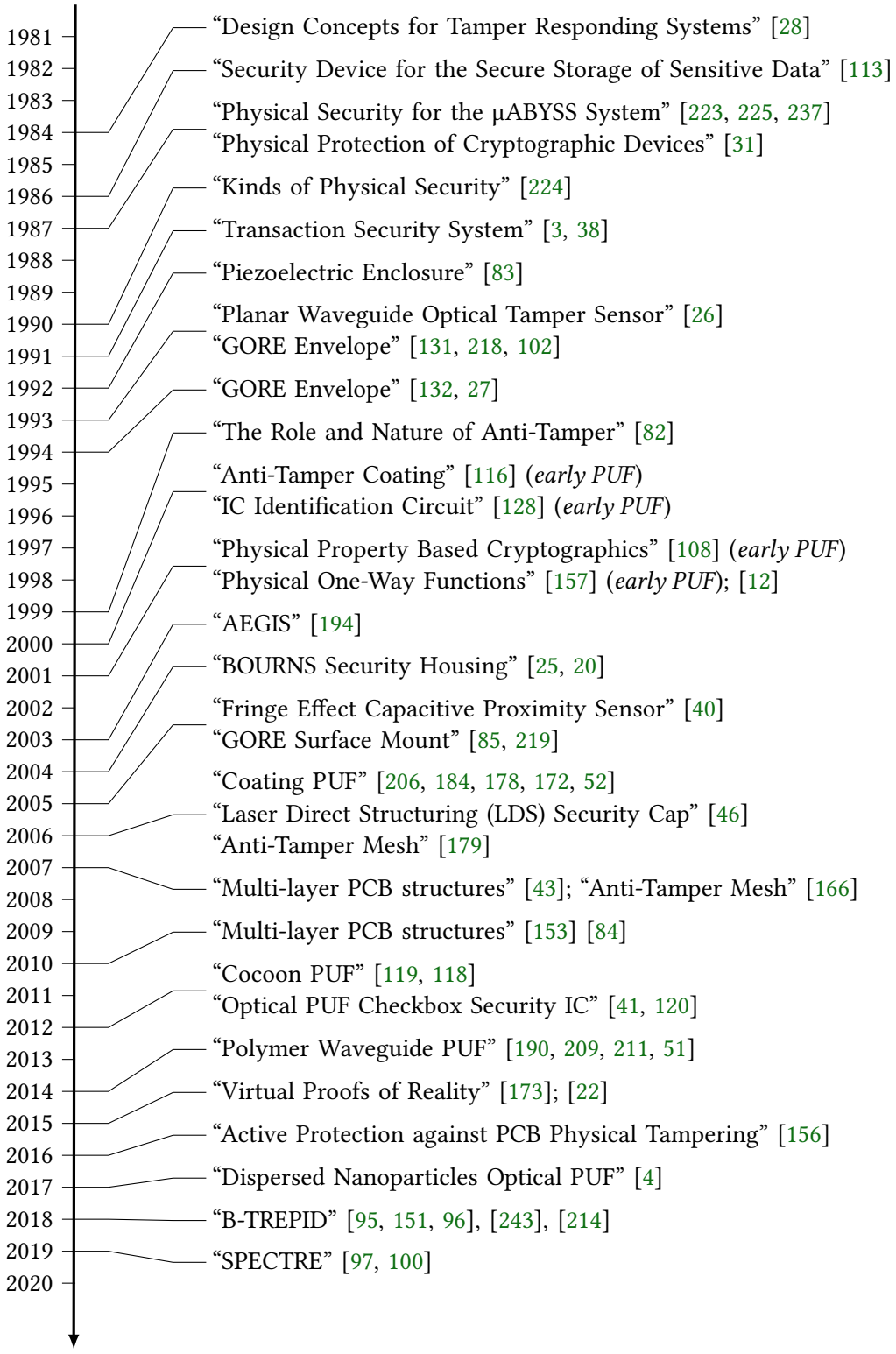


Figure 2.2: Timeline of noteworthy publications and inventions in the domain of tamper-protection (this list is not exhaustive, please let me know if I missed yours).

Based on the requirement to protect a system as a whole, Vai et al. present an optical "Polymer Waveguide PUF" [190] with a corresponding system architecture in [209]. This appears to follow similar considerations as originally envisioned in [26]. Since the waveguide only covers the top of a PCB, its edges and bottom remain unprotected. In general, combining optical approaches that do not fully enclose the PCB suffer from the challenge of how to securely assemble the optical token, e.g., the waveguide, with the PCB that typically provides only electrical means to make contact, e.g., pads, vias, or copper tracks. In addition, generic shortcomings of backscatter based systems such as inhomogeneity of the "illumination" and relative shift of the PUF token to the sensor, e.g., due to vibration, were not addressed. Yet another aspect is protection of such a system during runtime which is vital to protect its keys that are temporarily stored in volatile memory. Therefore, implementing a runtime tamper detection that monitors the system after power-on is essential to detect possible tampering attempts which is not mentioned in [209]. Unfortunately, no statistical assessment of the PUF properties and no practical security analysis was carried out.

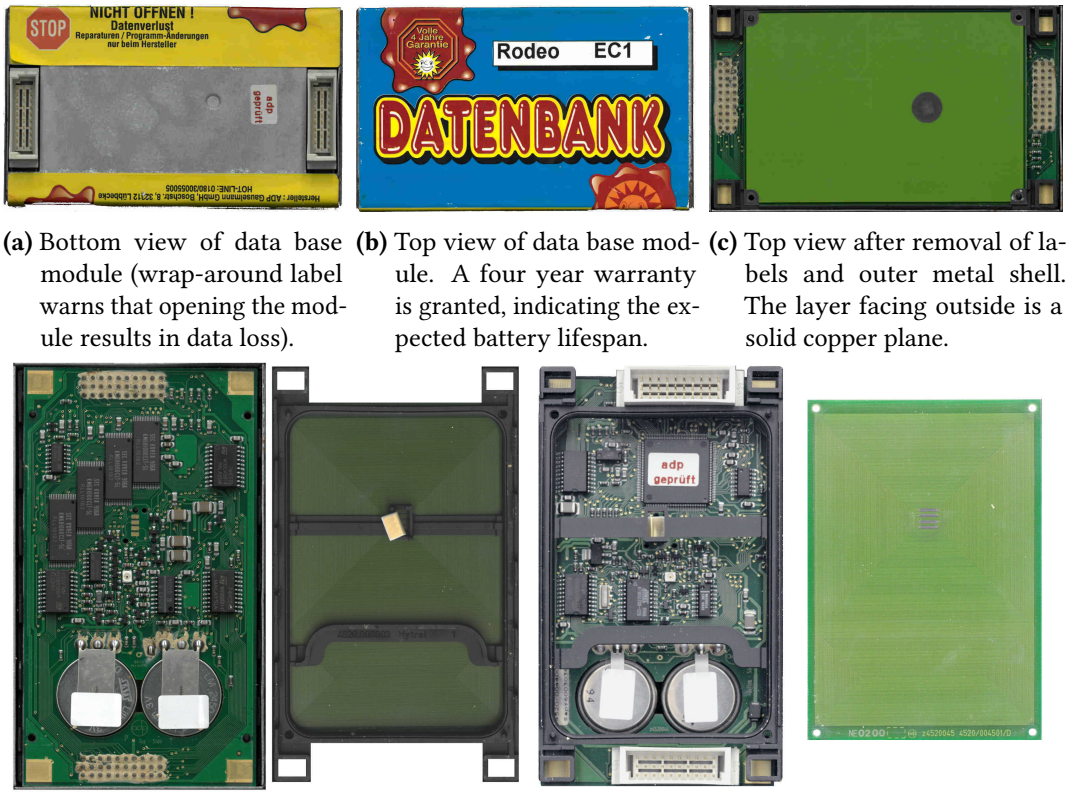
2.1.2 Real-World Physical Security Examples

In general, HSM documentation typically does not include specific information about their tamper-respondent enclosures. This might be owed to the fact that the Joint Interpretation Library (JIL) [105] to assess the attack potential of a device grants points based on the lack of public information of the security mechanisms. As scoring these points is essential in passing the security certification process, it implies that such countermeasures are at least partially founded on a "security-by-obscurity" principle.

Due to the unavailability of public information for battery-backed tamper-responding approaches, we provide selected details of related work for the reader's convenience. For that purpose, we acquired three types of HSMs to carry out an analysis by means of destructive disassembly. We first analyze the Gauselmann Data Base Module, a module used in slot machines to ensure integrity of financial transactions to prevent money laundering and tax fraud. Afterwards, the IBM Crypto Coprocessor [87] is studied which is protected by the GORE envelope. This is complemented by a brief analysis of the HP Atalla Cryptographic Subsystem [69, 70]. This choice is based on the high level of craftsmanship of the mechanisms employed by IBM and HP, their ease of availability via an online market place at a reasonable cost, and their representative features that are exemplary for many other devices of this class. Similar or slightly less sophisticated tamper-responding solutions can be found in Point-of-Sales (PoS) terminals adhering to the standards of the Payment Card Industry (PCI) [161, 162]. The findings on the IBM Crypto Coprocessor and HP Atalla Cryptographic System have been published in [151].

Disassembly of Gauselmann Data Base Module

The device shown in Figure 2.3 is a dated version of a module made by Gauselmann. It is part of slot-machines and is a subsystem for processing financial transactions to ensure that the odds of winning while gambling adhere to German government regulations which are enforced by a government body called the "Physikalisch Technische Bundesanstalt" (PTB). Its metal casing is made from two shells that, once fitted together, form an enclosure with no direct angle to access the enclosure's inside, i.e., only some connectors are directly accessible from the outside. The top and bottom view is shown in Figure 2.3a and Figure 2.3b respectively. On a mechanical level, disassembly is only prevented by standard screws and



(a) Bottom view of data base module (wrap-around label warns that opening the module results in data loss).

(b) Top view of data base module. A four year warranty is granted, indicating the expected battery lifespan.

(c) Top view after removal of labels and outer metal shell. The layer facing outside is a solid copper plane.

(d) Top view after removing frame with protective PCB containing a set of meander traces.

(e) Bottom view after removing outer shell and protective PCB, leaving frame on other side.

Figure 2.3: Step-wise dissassembly of a (dated) Gauselmann data base module.

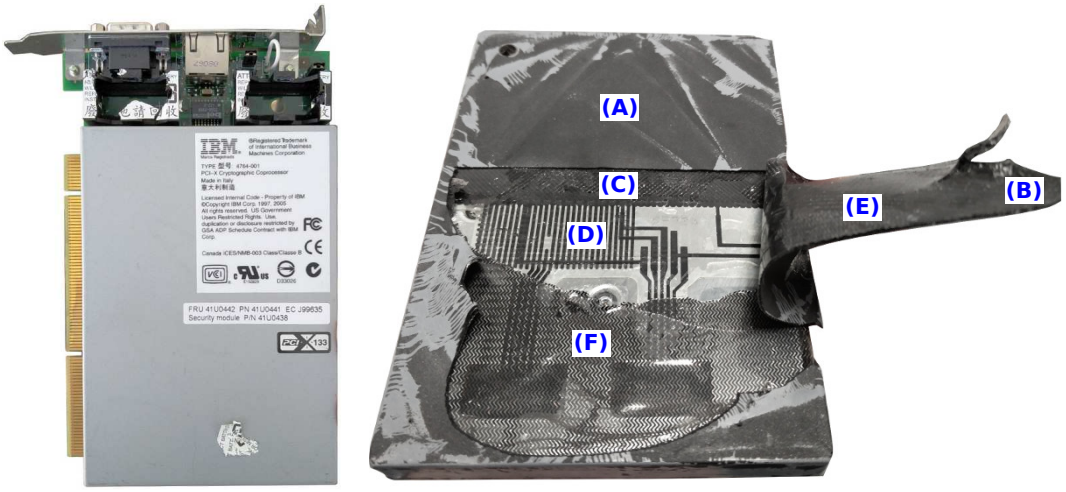
a label warning of data-loss upon attempted opening. Once the shells are removed, the same cover comprising meander-tracks is revealed (cf. Figure 2.3c) protecting the top and bottom of the PCB in the center of the PCB sandwich. Both covers are positioned by a plastic frame as seen in Figure 2.3d where this frame including its cover have been removed, and Figure 2.3e, where just the cover was removed while leaving the frame in place.

The PCB apparently includes the batteries to sustain the tamper-responding monitoring mechanism, i.e., servicing the battery is only possible when the device is disassembled. Small electrically conductive sponges connect the covers and their tracks to the MCM, i.e., removing the covers creates an open circuit that is detected easily. If the covers are left in place, drilling or otherwise penetrating the meander-tracks is likely to break the tracks and thereby cause detection, too. Light sensors on the top and bottom of the MCM additionally detect intruding light and presumably raise an alarm when light intensity is above a certain threshold. The CSPs are apparently stored in the BBRAM which is a low-power SRAM that can be sustained by the batteries and erased instantaneously upon detection. No attempts were made to circumvent the security mechanisms of the module.

Disassembly of IBM Crypto Coprocessor / GORE Envelope

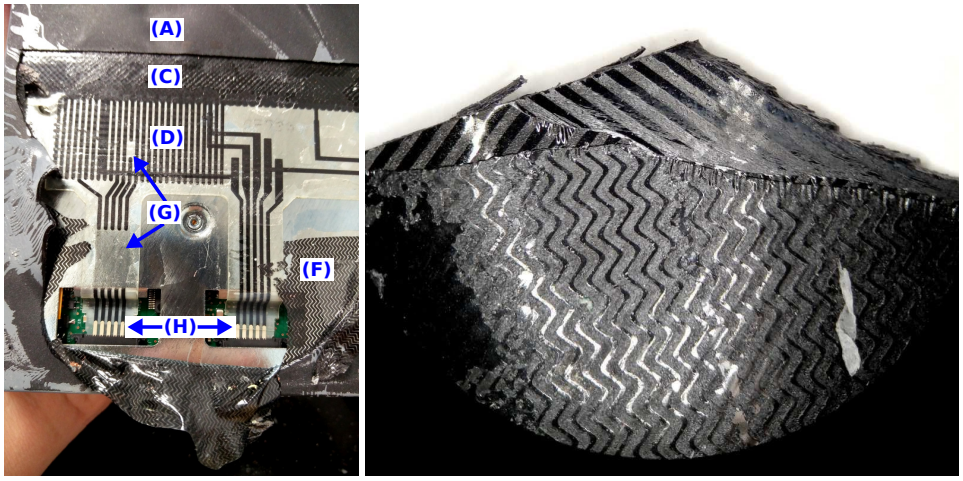
IBM's HSM, shown in Figure 2.4a, is based on a PCB that is enclosed in a case which is then enwrapped by the GORE tamper-respondent envelope, as described in part in [102,

224, 3, 38]. The envelope comprises a complex mesh that is monitored by a battery-backed tamper-detection system. This circuit verifies the mesh's integrity by measuring its ohmic resistance. Since our analysis is based on destructive reverse-engineering, we focus only on the most striking countermeasures we stumbled upon. Our findings therefore do not necessarily represent the full scope of the implemented security mechanisms. Please note, this brief analysis is only intended to support the strong need for batteryless security enclosures and does not necessarily suggest a fully successful attack on the system.



(a) Selected IBM HSM with GORE Envelope for analysis.

(b) Wrapped and potted module, partially opened to show the various layers of defense.



(c) Close-up of connectors and randomization.

(d) Pattern of three (out of four) layers, enabling a detection of 300 µm holes.

Figure 2.4: Selected aspects of the tamper-respondent envelope and packaging of [102] and related devices with envelope by GORE.

Packaging and physical properties. Figure 2.4 shows basic elements of the GORE envelope together with the protected HSM. In Figure 2.4b and 2.4c, the outer metal shell is already removed to gain access to the potted HSM. The potting (A) is a dark, rubbery, and opaque material with an awful smell, completely surrounding and concealing the wrapped envelope. The only objects breaking the otherwise closed surface are flexPCB connector

cables and a minuscule air vent, presumably to prevent gases from building up on the inside while applying the potting. When peeling off potting material at a specific position, it was possible to loosen a large section (B) and by further applying force, to tear the envelope apart, thereby revealing the underlying structures, such as the beginning of the envelope's sensoric area (C), where the envelope overlaps itself after the packaging is finalized.

Beyond a pattern-level physical randomization, e.g., changing the zig-zag pattern to curves, the envelope and its circuit provides additional means to permute the internal structure for each individual envelope without changing the electrical parameters as seen by the measurement circuit. This is implemented in region (D) that consists of *vertical* traces which lead in and out of the envelope's sensoric region. Region (E) contains *horizontal* traces that span the whole width of region (D). Since (D) and (E) are separated by a carrier substrate, connections are created by vias in the insulating layer.

In general, the envelope is quite fragile in response to attempts of physically tampering with it. Regions (D) and (E) were designed to separate from each other under mechanical force, i.e., they provide only very low tensile resistance which is confirmed by corresponding patents. Hence, any attempts of unfolding or partially unwrapping the envelope are highly probable to permanently destroy it. Other regions, such as (C) are designed similarly, such that (F) remained on the HSM while the upper three layers stuck together. When the bottom layer (F) was torn off, parts of the traces of (G) were removed also. Clearly, this material-based property in combination with the continuous sensing of the monitoring circuit is a crucial element for the envelope's security.

A close-up of the envelope's sensoric region is shown in Figure 2.4d. It shows three out of four layers of the mesh, whereas the two innermost layers are a zig-zag pattern and the two outer layers show a diagonal structure. The traces are made from carbon-ink material with a substantial ohmic resistance. The material properties and assembly steps are such that the carbon-ink material is easily scrapped off with very little force. Moreover, its chemical properties closely resemble the potting material such that tailoring solvents to only remove the potting is assumed difficult. Attempts to solder to these carbon-ink traces failed, instead, conductive silver or similar materials must be used. Likewise, directly probing these traces is difficult, as they become easily damaged using standard multimeter probes. However, the mesh structure is rather coarse compared to the solution B-TREPID [95], as also presented in Chapter 4.

Monitoring circuit. Two cables that are part of the envelope itself connect the battery-backed circuit to the envelope's mesh. This is marked with (H) in Figure 2.4c. Seven signals are present on the connector: ground (GND), the supply voltage (V_{CC}), and five voltage sense signals (V_{S1} to V_{S5}). The envelope's traces are configured as five voltage dividers. The output of each voltage divider is connected to GND and V_{CC}, whereas the center of the trace outputs V_S. As long as the envelope is intact, V_S is $V_{CC}/2$. The voltage of all five voltage dividers is sensed and checked by the evaluation circuit.

The evaluation circuit is powered from two redundant 3 V lithium batteries residing outside the HSM. Due to the limited energy available, the system was apparently built with a minimized power consumption in mind. Consequently, it uses a low-power MSP430 microcontroller in addition to low-current operational amplifiers. The envelope itself is estimated to have resistances in the range of several megaohms which is attributed to the carbon-ink material. This is necessary to minimize the energy drawn from the battery. However, such a high-ohmic voltage divider is designated to be susceptible to external influences that might erroneously trigger the tamper detection. This statement is supported

by the fact that the envelope's output voltages V_S are stabilized with 10 nF capacitors. Please note, low-power designs containing large capacitances and resistances are known to be comparably slow and as a result, limit the circuit's response time to physical tampering.

The voltages V_S are buffered using voltage followers and the five resulting signals are combined using diodes to determine the minimum and maximum voltage of the five voltages. Evaluating the signals independently from a microcontroller is motivated by the simplicity of the approach which reduces security pitfalls and preserves energy. A subsequent comparator checks if any of the considered signals exceed the lower or upper bound of the specified operating range. If this comparison fails, an alarm is raised to cause zeroization. As the system works with static voltage levels, the envelope's output remains constant as long as it has not been tampered with, thereby avoiding unnecessary switching activity. While this minimizes power consumption, it opens up a conceptual weakness, as an attacker could force the expected voltage from an external source into the circuit. Alternatively, the same could be achieved by using minimalist holes and/or attempted repairs with conductive silver. In either case, creating a suitable contact to the envelope's connector pads is relatively easy, as they are not made from carbon-ink and relatively large compared to the size of the mesh. Since there is no dynamic signal used to monitor the envelope, the attacker does not need to synchronize to it, further simplifying the attack.

The remainder of the evaluation circuit performs complementary system-level checks to detect other adversarial operating conditions, e.g., it verifies that the battery's voltage level is within a specified range. The same is true for the temperature, as the evaluation unit also comprises an internal temperature sensor. Having a sophisticated attacker in mind, the circuit additionally employs a large-area photodiode that senses even smallest amounts of light inside the enclosure. Bypassing this sensor would, e.g., require either a suitable hole to inject an opaque material onto the sensor (while operating in the dark) or a remote-controlled apparatus for the whole attack.

The results of all checks are combined using diode logic. Hence, even if the microcontroller is not active, the alarm as part of the tamper-response is triggered anyway. In such a case the power supply to the BBRAM, storing the CSPs, is shut down and its power supply pin is pulled to ground using a crowbar circuit, thereby zeroizing all data.

Analysis summary. Taking into account the previously described countermeasures, it is evident that only the most sophisticated attackers would attempt to break into the system. Despite being discontinued, it is therefore understandable that this physical security enclosure has been the de facto standard for many years throughout the industry. Still, we identified *potential* drawbacks of the system's measurement setup, i.e., static signals and the need for a battery, resulting in a limited responsiveness of the design. In addition to its restricted operating temperature range this strongly supports our argument for batteryless tamper-resistant enclosures.

Disassembly of HP Atalla Cryptographic Subsystem

HP's HSM as shown in Figure 2.5a is enclosed by a cover on top and bottom of the PCB. Hence, from an attacker's point of view, it is necessary to consider cover removal and/or cover penetration. While fully assembled, no apparent opening is present to get inside the secure compartment. Since no potting or adhesives are used, we expect that as part of an authorized servicing, disassembly and refitting the covers is possible. Likewise, attacking and reverse-engineering the device is simplified when compared to the previously analyzed device. Both covers comprise the following layers: outer metal shell or heatsink, thermal pad, flexible sheet with mesh, inner shell. Please note that removing the outer metal shell

and heatsink is possible without causing the zeroization, e.g., by first creating holes in the heatsink and then using a small jig frame to hold pressure on the inner shell.

Cover removal. On both top and bottom, cover removal sensors are implemented by pads on the PCB that are shortened by a conductive tape that is attached to the rim on the inside of the covers. Hence, when the pressure holding the covers together is relieved, then the conductive tape no longer short-circuits the pads on the PCB. Once this is detected, this causes the zeroization of CSPs that are again stored in a BBRAM. Please note, for a successful removal, the bottom cover must be removed prior to the top cover, as the top cover's screws directly feed into the PCB. Hence, access to these screws is prevented as long as the bottom cover is attached.

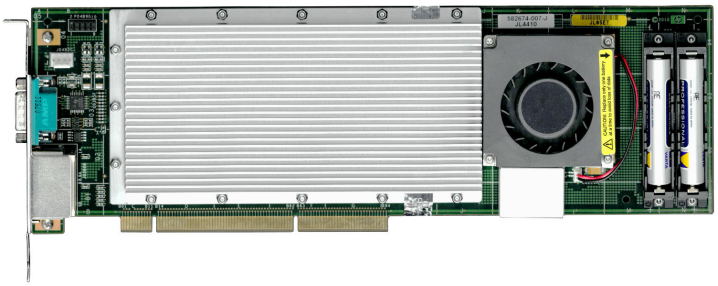
In contrast to the top features the bottom cover an additional thick conductive foam in its center that creates a connection between the PCB through the bottom cover's metal shell to the cover's connector, representing the signal MeshGND which is one out of three signals present at the connector of the cover. Hence, a step-wise and thoughtful disassembly without tampering the HSM needs to focus on this specific countermeasure, as the cover removal sensors are exposed once the outer shell (bottom) or heatsink (top) are removed.

Cover penetration. This cover's mesh is relatively crude with approx. 1 mm traces and an equally sized space in between, as shown in Figure 2.5c. Its structure size is therefore three to ten times larger than the mesh contained in the GORE envelope or the implementation in [95]. Only a single loop is present in the cover, i.e., one long track in a serpentine pattern that goes from the connector's MeshSigIn to MeshSigOut. Since, it lacks the strong material properties of the solution described in the previous subsection, we could disassemble the device without changing its physical parameters. Hence, we could determine the loop's resistance to be 300 Ω. Given the fact of the relatively large separation of the traces, allowing drill diameters of up to 1 mm to go undetected, and the overall track length that adds uncertainty to the exact resistance value due to manufacturing variation, it is evident that resistance against physical penetration is limited.

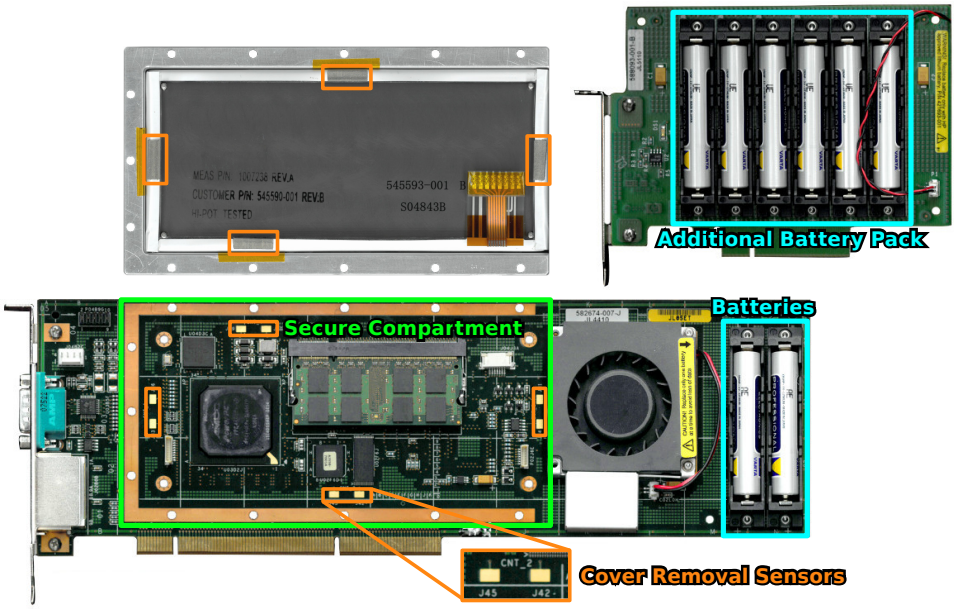
Analysis summary. Tailoring tools for attacking the device only require a moderate level of sophistication, owed to the less-complex set of countermeasures and the crude mesh. This may either result in attempts to directly disable, e.g., the alarm signal, or to disable the individual countermeasures step-by-step. For this step-by-step approach to succeed, defeating the bottom cover's connection through the conductive foam in its center appears most challenging. Taking into account the previous comments and the lack of, e.g., a light sensor, it is evident that a successful attack is more probable to succeed when compared to the GORE envelope of IBM's HSM.

2.1.3 Drawbacks of Battery-Backed Access Denial Systems

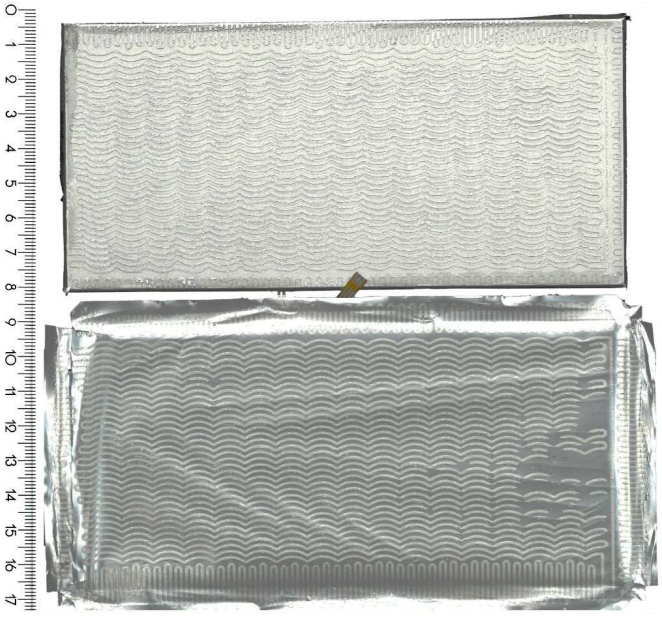
Battery-backed mechanisms in general entail several drawbacks that are beyond the specifics of the aforementioned HSMs. This is due to the perpetual monitoring of the enclosure, even if the device is powered off. Practical challenges arise from added bulk and weight which limits the use of such systems for mobile applications. Moreover, it clearly increases cost in addition to the enclosure itself. On a technical level, batteries are subject to self-discharge over time. Prolonged storage may fully discharge them, causing loss of CSPs which in turn leads to inoperable devices, as physical security can no longer be guaranteed. This is a severe limitation, requiring maintenance personnel to regularly inspect and replace batteries, thereby adding to the cost of these solutions.



(a) HP Atalla Cryptographic Subsystem (assembled).



(b) HP Atalla Cryptographic Subsystem (disassembled).



(c) Dismantled bottom cover showing the resistive sensor mesh (scale as reference in cm). Its connector only carries the three signals MeshGND, MeshSigIn, and MeshSigOut.

Figure 2.5: Physical security of the HP Atalla Cryptographic Subsystem.

However, insufficient battery power is not the only reason the tamper detection mechanism and response mechanisms are initiated. The shipping process is yet another significant obstacle, since the mechanism must be armed at a trusted facility and is subsequently exposed to uncontrolled temperature, mechanical shocks, and vibration. As these devices implement Environmental Failure Protection (EFP) to limit physical tampering with the materials, e.g., melting potting away, their environmental operating window is typically small. For example, the upper ambient temperature limit for a similar IBM HSM is 60 °C during shipping, already taking into account that the HSM is in a "thermally insulated box with gel packs" [87]. Since the monitoring circuit is permanently in operation it is possible that these environmental conditions inadvertently result in the detection of an attack.

In contrast to this behavior are PUF-based security enclosures not prone to this problem, as they are fully powered down when not in use. Hence, they are relatively unaffected by the shipping process. Due their batteryless design, their lifetime is also not inherently limited by a battery. This clearly emphasizes the potential benefit of PUF-based security enclosures over current battery-backed approaches and therefore motivates investigating alternative concepts to achieve the same level of protection, or possibly even exceed previous levels of protection.

2.2 Standards for Security Certification

HSMs are an indispensable tool to secure the root of trust for many security-aware applications and digital infrastructure. Due to how several domains evolved, different standards exist to control how HSMs are designed, manufactured, and operated throughout their lifetime. Major standards for security certification include: FIPS 140-2 [147], a standard that applies solely to HSMs and their specifics; PCI-HSM [161] and PCI-PTS and their related standards that are relevant for devices processing payment transactions; Common Criteria [202], a security certification framework covering a multitude of security aspects and device classes by instantiating Protection Profiles (PP) [111]; the German Banking Industry Committee (GBIC), a standard for devices in the German banking sector.

In general, a security certification process ensures that the security relevant functionality has been reviewed by an independent third-party and that they adhere to requirements mandated by the corresponding authority of the standard. Just to name two examples where this is relevant: baseline requirements for the issuance and management of publicly-trusted certificates, i.e., guidelines applying to Certificate Authorities (CA) explicitly state that [45]: "*The CA SHALL protect its Private Key in a system or device that has been validated as meeting at least FIPS 140 level 3 or an appropriate Common Criteria Protection Profile or Security Target, EAL 4 (or higher), which includes requirements to protect the Private Key and other assets against known threats.*". A practice statement for the Root Zone Key Signing Key Operator of DNSSEC similarly states [86]: "*For RZ KSK generation and RZ KSK private component operations and storage, the RZ KSK Operator uses hardware security modules that are validated at FIPS 140-2 level 4 overall.*"

A certification process typically follows a procedure as described hereafter. A device manufacturer files a request for certification at the national body responsible for the standard. An accredited and independent testing lab is then selected to carry out an analysis and corresponding tests to confirm the claimed security functionality. Upon completing the review, a report is sent to both the manufacturer and the national certification body. Based on the provided outcome of the test, the certification body then grants the certificate.

Since the standards vary quite significantly, let us have a brief look at the different levels of FIPS 140-2 *physical* security, i.e., the following list does *not* cover other aspects of this standard. Please note that the higher the level, security is considered accumulative including all aspects of the lower levels.

- **FIPS 140-2 Level 1:** Lowest security level. At least one approved cryptographic algorithm must be implemented and there are *no* physical security controls.
- **FIPS 140-2 Level 2:** Level 1 plus basic physical security controls. CSPs are protected with tamper-evident coatings or seals, i.e., passive systems.
- **FIPS 140-2 Level 3:** Level 2 with enhanced physical security controls. CSPs are deleted if a potential breach is detected, e.g., typically active systems that detect if covers or lids have been removed or opened.
- **FIPS 140-2 Level 4:** Level 3 with additional physical security controls, providing the highest level to make the HSM usable in physically unprotected environments, i.e., there must be no demonstrable way to defeat the physical security mechanism outside of accredited testing labs. The standard explicitly states that a "tamper-detection and response envelope with zeroization circuitry" must be used.

Clearly, from a security perspective, the goal should always be to achieve a Level 4 protection to provide the best level of protection possible. However, only very few devices are known to have successfully passed this type of certification (while some parties may have chosen not to undergo such a certification at all for reasons of discretion). Guidelines for developers adhering to Common Criteria [183] additionally list four useful properties that are relevant for the security functionality:

- **Non-bypassability:** "*The developer shall design and implement the TOE so that the security features of the TSF can not be bypassed. He shall design and implement the TSF so that it is able to protect itself from tampering by untrusted active entities. The developer shall provide a security architecture description of the TSF.*" [183]
- **Security Domains:** "*The security architecture description shall describe the security domains maintained by the TSF.*" [183]
- **Security Function Initialization Process:** "*The security architecture description shall demonstrate that the TSF initialisation process preserves security. This portion of the security architecture description should list the system initialization components and describe the processing that occurs in transitioning from the down state to the initial secure stage (i.e. when all parts of the TSF are operational) when power-on or a reset is applied.*" [183]
- **Self-Protection:** "*The security architecture description shall demonstrate that the TSF protects itself from tampering.*" [183]

When developing the tamper-resistant enclosure presented in Chapter 4, the specifics of these security standards where taken into account to support a later certification process, e.g., the duality of integrity detection and capacitance-based PUF is a direct result of the PCI security standard, where physical integrity mechanisms of a different nature are mandated

such that if one of them would fail, the other would still provide a reasonable level of security. Moreover, since previous approaches provide a well-defined rationale for the achievable security level, we tried to follow similar principles and ideas, even though it was designated to be a PUF.

2.3 Conclusions on Application Context

Developing physical security countermeasures exceeding the scope of IC-level techniques is clearly motivated by the previous examples. This is additionally supported by the provided background information on the application context and the industry standards in existence to ensure a uniform level of protection across the whole range of products and manufacturers. Upcoming applications such as Unmanned Arial Vehicles (UAVs) or self-driving cars are only going to increase the need for such countermeasures.

Similarly to topics other than access denial systems, it is difficult to impossible to exhaustively cover all publications and patents. However, compared to other hardware security topics, e.g., power side-channel analysis, it is rather striking how scarce the information on tamper-resistant enclosures and related mechanisms is, i.e., a single timeline as provided in Figure 2.2 is sufficient to cover a majority of public references, even when complemented with patents which is not common for academic work.

As seen in other domains that had been subject to secrecy and utmost discretion beforehand, e.g., design and analysis of cryptographic algorithms, transitioning this into an openly debated matter has helped to come up with important advancement and ensured that the greater public benefits from these developments. It is the author's opinion that a more open and competitive process is more than due for access denial systems. The work following in Part II is a humble attempt to aid this process and encourage others to join in.

Part II

Higher-Order Alphabet PUF Construction

Chapter 3

Previous Work on PUF Constructions

As briefly introduced in the previous chapter, PUFs evaluate manufacturing variation to provide a physical root of trust, e.g., by using its unpredictable output data as a kind of seed. This seed can then be used for various purposes such as secret key derivation, tamper-detection, or challenge-response authentication protocols. In Section 3.1, basic PUF definitions are covered. Afterwards, in Section 3.2, the state of the art in PUF constructions is surveyed, i.e., how the architectural hardware concept is designed to leverage physical phenomena to yield a PUF with the desired properties. In addition, differences of the various PUF constructions are detailed and, including the differences of binary PUFs when compared to HOA PUFs.

Contents

3.1 PUF Definitions and Exemplary Constructions

As seen in the timeline presented in Figure 2.2, PUFs evolved from concepts that had not been named PUF at the time. Early terms include but are not limited to: Physical Property Based Cryptographics [108], Physical One-Way Functions [157, 158], Physical Random Functions [48, 49], Physical(ly) Unclonable Function (PUF) [60], and Physically Obfuscated Key [48, 49]. Correspondingly different definitions have been formulated to describe some of these sometimes slightly varying concepts. Additional works to formalize and summarize PUFs are [67, 7, 8, 175]. In the following, the term PUF is used, as it has proven itself to be the most commonly accepted term. The earliest definition of a PUF is by Gassend et al:

Definition 3.1.1 (Physical Random Function (PUF), quoted from [48, 49]) *A physical random function (PUF) is a function that maps challenges to responses, that is embodied by a physical device, and that verifies the following properties:*

1. *Easy to evaluate: The physical device is easily capable of evaluating the function in a short amount of time.*
2. *Hard to predict: From a polynomial number of plausible physical measurements (in particular, determination of chosen challenge-response pairs), an attacker who no longer has the device, and who can only use a polynomial amount of resources (time, matter, etc.) can only extract a negligible amount of information about the response to a randomly chosen challenge.*

Reflecting the idea of mathematical one-way functions is this definition based on the two complementary properties that a PUF shall be "easy to evaluate" but at the same time "hard to predict". An extended concept by the same author is named a Controlled PUF (CPUF) which states that the PUF shall only be accessible via an algorithm that is physically linked to the PUF in an inseparable way, i.e., the PUF *cannot* be a separate physical token that is measured with an external measurement circuit that would only be connected to the token when needed. This concept in addition to the previous two properties form the definition by Guajardo et al. [60]:

Definition 3.1.2 (Physical Unclonable Function (PUF), quoted from [60]) *PUFs consist of inherently unclonable physical systems. They inherit their unclonability from the fact that they consist of many random components that are present in the manufacturing process and can not be controlled. When a stimulus is applied to the system, it reacts with a response. Such a pair of a stimulus C and a response R is called a Challenge-Response Pair (CRP). In particular, a PUF is considered as a function that maps challenges to responses. The following assumptions are made on the PUF:*

1. *It is assumed that a response R_i (to a challenge C_i) gives only a negligible amount of information on another response R_j (to a different challenge C_j) with $i \neq j$.*
2. *Without having the corresponding PUF at hand, it is impossible to come up with the response R_i corresponding to a challenge C_i, except with negligible probability.*
3. *Finally, it is assumed that PUFs are tamper-evident. This implies that when an attacker tries to investigate the PUF to obtain detailed information of its structure, the PUF is destroyed. In other words, the PUF's challenge-response behavior is changed substantially.*

We distinguish between two different situations. First, we assume that there is a large number of challenge response pairs (C_i, R_i), $i = 1, \ldots, N$, available for the PUF; i.e., a strong PUF has so many CRPs such that an attack (performed during a limited amount of time) based on exhaustively measuring the CRPs only has a negligible probability of success and, in particular, $1/N \approx 2^{-k}$ for large $k \approx 100$. We refer to this case as strong PUFs. If the number of different CRPs N is rather small, we refer to it as a weak PUF. Due to noise, PUFs are observed over a noisy measurement channel, i.e., when a PUF is challenged with C_i a response R'_i which is a noisy version of R_i is obtained.

The latter definition includes the aspect that data retrieved from a PUF must be considered *fuzzy*, i.e., each read-out results in data that is slightly different due to circuit noise and environmental drift effects such as voltage, temperature, and humidity. To mitigate these effects, they must be counteracted which is detailed in Part III. Some of the techniques involved are based on algorithmic processing requiring *helper data*. Typically, this is done by a two-staged approach: at the factory, the *enrollment* derives the PUF key for the first time and helper data is created to enable later *reconstruction* of the same key from a noisy PUF response in the field.

Typically, this helper data is assumed to be a *public* parameter of the system, i.e., the attacker would know this data and may attempt to deduce information from this data about the derived secret of the PUF. In general, this is called "helper data leakage" and since partial recovery of the PUF's secret is attempted, it is also coined *secrecy leakage*. If a PUF

system has been properly designed, storing the helper data in NVM of a system should be possible without jeopardizing the PUF's security. Some popular PUF designs that have also been marketed by companies such as Intrinsic-ID B.V. and Verayo Inc. are:

- **SRAM-PUF** [60]**:** Upon power-up, if left uninitialized, the state of an SRAM cell is defined by the threshold voltages of the involved transistors in the inverters. Hence, a unique fingerprint is present in the SRAM that can subsequently be used as a fuzzy random seed. Since SRAM is available in many microcontrollers, it is a natural candidate to serve a dual-purpose of a PUF at device start-up, in addition to storage of volatile data during runtime.

  Due to the decade-old concept of an SRAM cell, these circuits can be considered highly optimized throughout the whole chain of designing and manufacturing them, i.e., resulting in a low area due to being available in most recent technology nodes, robustness towards temperature and voltage drift, etc. With regard to error-correction, a typical Bit-Error-Rate (BER) of $\sim 15\%$ is assumed [107]. Publicly available data sets include for example [230].

- **RO-PUF** [193]**:** The Ring-Oscillator (RO) PUF comprises a closed-loop with an odd number of inverting elements. Constant switching occurs in this loop once the circuit is enabled, resulting in a continuous switching activity leading to a manufacturing variation dependent frequency. This is caused by the timing differences of the gates, causing a unique timing for the signal propagation when comparing spatially separated instances of ROs on the same device, or across different devices.

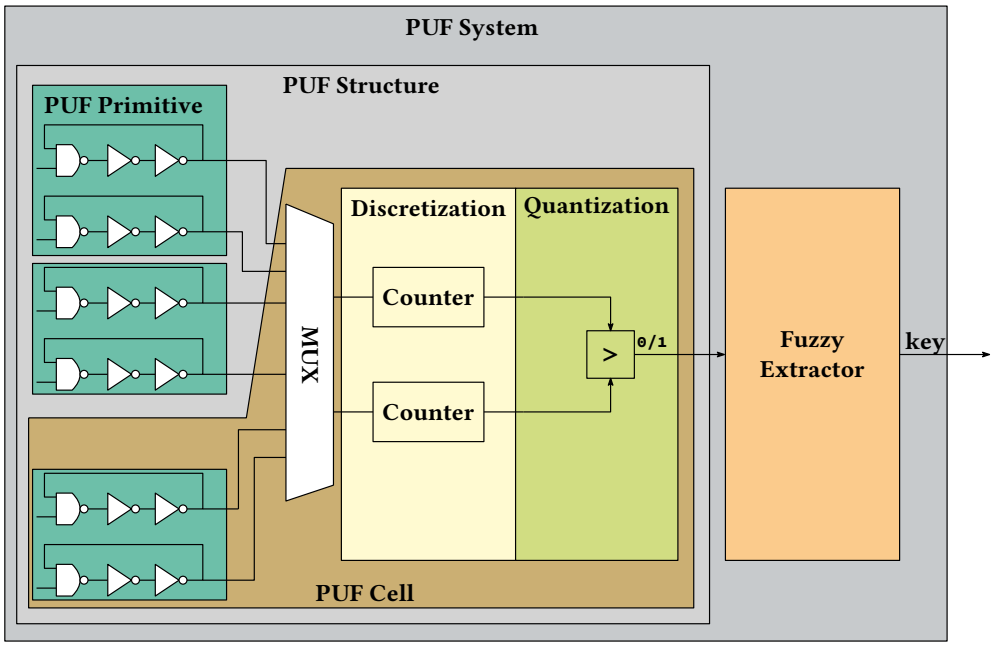


Figure 3.1: Structure of the RO-PUF as proposed by [193].

Typically, several ROs are combined to an RO-PUF as illustrated in Figure 3.1 Alternatively, on FPGAs, different routings can be explored by means of Dynamic Partial Reconfiguration [50]. To measure the randomized oscillation frequencies of the inverter chains, typically a counter is used. Subsequently, the most common choice of further processing the obtained counter-values is by performing a

pairwise-comparison. In general, the RO-PUF has been a highly favored variant in academia, since they can be implemented well in FPGAs, leading to several publications with accompanying public data sets [139, 227, 68]. A related RO variant is the SUM-PUF [242].

The most noteworthy difference between Definition 3.1.1 and 3.1.2 is the property of tamper-evidence in addition to a more formal treatment of the other two basic PUF requirements. While these two seminal definitions have served as excellent reference for many publications in the domain of PUFs, they should not be considered complete. For example, these definitions fall short with regard to the physical scope of a PUF, i.e., a PUF could be instantiated simply as a component in a larger system without having its property of tamper-evidence "propagate" to the remainder of the system. This might be owed to the fact that most PUFs have been implemented in silicon, and not on a system-level, as explained in Chapter 3. As a result are PUFs often perceived as a "black-box" where physical security is implicitly assumed for the inside of the PUF and only its helper data may be accessed by an attacker. However, several publications have practically proven that PUFs lacking tamper-evidence can indeed be attacked with moderate resources available in standard testing labs [65, 148, 64, 197, 129] such that this assumption was already proven invalid. Additional aspects related to tamper-evidence that are only poorly captured by the current definition of a PUF are: allowed extent of "repairability" of the otherwise tamper-evident structure, formalized sensitivity towards physical attacks, tamper-evidence based on operating state of the PUF, i.e., whether the PUF device is powered-off or powered-on. In part, this is later addressed by the proposed definition of *tamper-sensitivity* in Section 9.1.

In the following, basic terms of a PUF construction from an engineer's point of view are introduced. As such, they differ from formalization attempts for examle in [7] and are intended to provide a more specific guideline for PUF implementations. These terms are:

- **PUF Construction:** The following components are also illustrated in Figure 3.1 for the RO-PUF and can often be found in several PUF designs.
  - **Primitive:** A PUF is rooted in a physical primitive that provides *entropy*, i.e., a capacitor, a Ring-Oscillator (RO), etc. This is considered a physical object comprising the physical parameters that are subject to uncontrollable and therefore random manufacturing variations.
  - **Discretization:** At some point in time of the subsequent processing stage, an analog-to-digital conversion needs to be carried out, i.e., making the otherwise continuous physical parameters available in a discretized form such that further processing in a digital system is possible. In a practical system, the resulting value is often an *integer*, e.g., a counter or the output of an ADC.
  - **Quantization:** A quantization scheme helps to convert a high-resolution value with m bits to a decimated value with n bits, i.e., $n < m$ (cf. bullet point on alphabet). This allows processing the values that are reduced in bit complexity more efficiently. At the same time, based on the specifics of the quantization scheme, the influence of noise and/or environmental drift is reduced.
  - **Cell:** The conceptual combination of PUF primitive, discretization, and quantization is termed PUF cell. Depending on the chosen area trade-off, this may be fully replicated, or partially replicated to enable a resource sharing of elements such as discretization and quantization among multiple PUF primitives.

  - **Structure:** In all known PUF constructions to date, a single PUF primitive as entropy source is not sufficient to gather enough random data for cryptographic purposes. Hence, multiple PUF primitives and cells must be instantiated, often resulting in a grid or array of PUF primitives. All PUF cells combined form the PUF structure.
  - **System:** The whole PUF construction, including the subsequent ECC, in addition to any other component necessary to create a PUF that is ready to be used in an application, is called PUF system.

- **Related Terms:** The following terms are not illustrated in the Figure 3.1 but are of relevance for several other PUF designs.
  - **Compensation:** To account for environmental drift effects, some of the PUF constructions make use of a dedicated technique that is referred to as *compensation* (in contrast to error-correction). For example, additive and multiplicative errors as a result from drift effects can be counteracted with a circuit-level technique called "3-signal" approach [206], describing a linear transformation based on the knowledge provided by a known reference.
  - **Raw Data:** Depending on the PUF construction, *raw* data may refer to data at different stages of the PUF. In general, this term refers to *unprocessed* data, i.e., the earliest data available from the PUF. In case of the SRAM-PUF this is the already quantized data from the SRAM cells. In contrast, for the RO-PUF this is the data generated by the counters.
  - **Normalization:** Manufacturing a PUF structure typically entails desired manufacturing variation in the sense of entropy but also undesirable effects such as structural bias that represent a deviation from the expected result and as such do not provide entropy. Ensuring homogeneity of the raw data by means of a suitable PUF construction is called normalization, since the ideal outcome for further processing is data fitting in the same parameter window, i.e., following the same distribution. In many PUF constructions, the measurement and subsequent comparison of two directly neighboring PUF primitives is used as ad-hoc normalization technique. This approach often intersects with the concept of a differential measurement targeting a more robust measurement with regard to environmental drift effects and noise.
  - **Alphabet:** In this work, the alphabet $\mathcal{L}$ of a PUF refers to the resulting values *after* the quantization right *before* the subsequent processing of an ECC. Most PUFs, as detailed in the next section, are designed to provide a single binary output bit per PUF cell, i.e., $\mathcal{L} = \{0, 1\}$. This is different to the PUF construction later presented, generating a higher-order alphabet per PUF cell, i.e., $\mathcal{L} = \{a, b, c, \ldots, |\mathcal{L}|\}$.

3.2 Classification of PUF Constructions

Independent of the property of tamper-evidence, it is of interest to survey existing PUF constructions and study their architecture. In this section, a classification of PUF constructions is provided, based on the overview shown in Table 3.1. It is primarily based on how the PUF output of a single primitive or cell has been designed. Additionally, the table lists

identifying properties of each construction with regard to the components leading to the output of the PUF structure.

The first class of PUFs is based on primitives that naturally output binary data directly, such as the SRAM-PUF [60] or Arbiter-PUF [49]. Each of their PUF primitives provides a single bit and the output provided by all primitives is directly fed into the subsequent ECC (part of the fuzzy extractor). Other processing steps such as a dedicated quantization scheme, compensation, or a normalization are therefore not part of their construction.

Another important class of PUFs is introduced based on the example of the RO-PUF. The basic idea of an RO-PUF and a first design was proposed by Gassend in [48]. Here, the RO-PUF is a single configurable oscillating circuit combined with a counter that tracks the number of oscillating cycles during a fixed time interval. This counter yields an integer which is the discretized representative of the RO's continuous frequency that is different for each RO and configuration due to manufacturing variations. All referenced RO PUFs make use of this integer-based counter for the discretization. In general, the concepts for compensation and quantization are independent of the type of discretization which could also be based on an Analog-to-Digital Converter (ADC), e.g., for mixed-signal PUFs.

Gassend identified that the environmental influences exceed the observed manufacturing variation of the RO and suggested to use a *compensated* measuring technique. In this case, he proposed to use the *ratio* of two neighboring ROs, assuming that drift effects are primarily based on a multiplicative factor. However, owed to the early stage of PUF research in 2003, he did not propose a specific quantization or alphabet.

An influential follow-up work by Suh and Devadas [193] proposed an FPGA-based implementation of the RO that included the quantization step by means of a comparator. Therefore, we term this RO(CMP). Unfortunately, at the same time the idea of compensated measurement was lost. Several other notable works are based on [193], e.g., such as the large-scale characterization of this type of PUF in [139] by Maiti et al. , [227], and [68].

Please note: determining the ratio of a pair of ROs, comparing their frequency with a comparator, or computing their difference can all be interpreted as a *differential* measurement. While the choice of differential operator might coincide with the dominant error type due to environmental drift, it should *not* necessarily be considered a dedicated compensating technique on its own. Part of the observed error-reduction is a side-effect of reducing the bit-complexity, i.e., the environmental effects simply become less visible in the output. Instead, we argue that a differential measurement primarily helps to extract the *local* variation of a PUF structure, e.g., by only considering directly neighboring ROs it can be assumed that an RO's Probability Distribution Function (PDF) is indistinguishable from the PDF of its neighboring RO, i.e., a structural bias in the discretized data is avoided. This was shown in [94] and provides a strong rationale to *only* use exclusive, directly neighboring pairs for the differential measurement of the RO instead of allowing all possible permutations of pairwise comparisons as a suggested option in [193]. Another work illustrating the structural bias in the RO(CMP) of [139] can be found in [229].

An example of a compensated measurement *without* a differential measurement was shown in [61] and is named RO(DCT). Here, the raw frequencies of the ROs are processed by the DCT and a subsequent coefficient selection, i.e., the DC offset is removed and only the most relevant data is used for the subsequent quantization.

Another example of a compensated measurement without a differential measurement is part of the Coating PUF [206]. This PUF is based on a randomized coating on the top of the IC that shows a unique capacitive behavior when measured using distinct capacitive sensors.

Table 3.1: Selected PUF designs and their respective structural properties.

PUF Construction	Entropy	Discretization	Normalization	Compensation	Quantization	Alphabet
Single-bit/primitive						
SRAM [60, 79]	memory		*not applicable / indivisible from cell*			binary
TwoStage [17]	memory		*not applicable / indivisible from cell*			binary
Butterfly [121]	memory		*not applicable / indivisible from cell*			binary
FlipFlop [136]	memory		*not applicable / indivisible from cell*			binary
Arbiter [49]	delay	latch	differential-pairs	*not applicable*		binary
PUFKY [134]	delay	counter	stored offsets	*by-product*	Lehmer-Gray	binary
RO (CMP) [193, 139]	delay	counter	differential-pairs	*by-product*	comparator	binary
HELP [29, 1]	delay	counter	modulus	linear transform	threshold	binary
Multiple-bit/primitive						
RO (RATIO) [115]	delay	counter	differential-pairs	ratio	bit decimation	binary
TERO [19]	transitions	counter	differential-pairs	difference	bit decimation	binary
RO (DCT) [61]	delay	counter	DCT	DCT	equiprobable	binary
MEMS PUF [231]	MEMS	not part of design	?	?	equiprobable	binary
Coating PUF [206]	capacitance	counter	*missing*	linear transform	equiprobable	binary
Symbol/primitive						
this thesis	capacitance	ADC	differential-pairs	various	*equidistant*	**higher**

As result of environmental influences, the basic assumption is that each capacitance is subject to *multiplicative and additive errors*. To compensate them, the authors make use of the "3-signal" measurement technique, a type of circuit-level linear transform with known reference. This approach is based on acquiring three measurements in a short sequence: one of a reference capacitance, one for the circuit's offset, and another one for the unknown capacitance. The circuit's offset is subtracted from the unknown capacitance and the known reference is then used to determine the multiplicative factor and scale the measurement accordingly. In [206], the resulting value is called *stabilized* and only carries remaining circuit noise. These stabilized values, i.e., discretized and compensated, are then further processed using an equiprobable quantization scheme, as also done for the MEMS-PUF in [231]. During enrollment, this scheme is used to compute the offsets of the noise-free stabilized measurements (obtained by averaging) to the center of the corresponding quantization interval. These offsets are then stored and represent helper data. Due to the unequal width of these intervals, it is easily possible that large offsets are generated for values in the outermost intervals which cannot occur in any of the smaller intervals [91]. Hence, severe helper data leakage occurs that is independent from the subsequent ECC.

Other approaches to equiprobable quantization include [215, 192, 24] using a partitioning scheme to avoid helper data leakage. However, two fundamental problems of equiprobable quantization remain. First of all, the necessity of precisely knowing the PDF which is assumed to be difficult for some practical scenarios, e.g., for an FPGA-based PUF where little control and knowledge of the underlying hardware is available to the PUF designer. Secondly, the quantization error is largely determined by the innermost (smallest) intervals which either results in relatively large number of errors or in a diminished entropy output when increasing the width of the innermost interval (assuming a constant noise level across the range of stabilized values). In contrast, an equidistant quantization as introduced in Chapter 6 is relatively insensitive to, e.g., shifts of the PDF and also provides a constant quantization error probability across the range of values. It is therefore an attractive choice for practitioners at the downside of a biased PUF output which needs to be carefully considered in subsequent processing steps. Please note: unlike previous approaches, the output of an equidistant quantization is based on symbols from a higher-order alphabet, i.e., interpretation of a symbol is based on the symbol's meaning and not its binary representation.

Going back to RO PUFs, the benefit of computing the ratio of RO frequencies to achieve a compensated measurement was rediscovered in [115]. Here, the authors choose to apply a "bit-decimation" to the discretized and compensated integers such that the most- and least-significant bits are ignored. By following this methodology, they ideally obtain multiple bits per RO pair based on a binary alphabet. A similar approach is found in the TERO-PUF [19]. Clearly, the effectiveness of the bit-decimation is rather limited in comparison to an equiprobable quantization, as the obtained bits are not i.i.d., would still suffer from burst-errors when low-magnitude changes in the integer representation cause multiple bits in the binary domain to flip, and also does not specifically include an error-correcting aspect. Moreover, designing an equiprobable or equidistant quantization scheme allows to take the noise standard deviation σ_N of the physical measurements into account, i.e., it can be naturally used as a design parameter to maximize the efficiency of this processing step [206, 91] which is in contrast to the bit decimation process.

Considering the aforementioned RO-PUF designs, we notice that neither one of them

investigated the idea of using a reference RO in the design to determine the multiplicative error similarly to the Coating PUF. Please also note that the sequence of discretization, compensation, and quantization could be slightly different based on the specifics of the implementation, i.e., the compensation could also possibly be done on the analog level using suitable circuitry even prior to the discretization. Additional work optimizing the properties of RO were done in [235, 10], in particular with regard to the measurement technique and compensation aspects. Unfortunately, a specific quantization approach was not used.

Another notable design is the HELP-PUF [29, 1]. It is an FPGA-oriented design that enables reuse of an AES S-box design by measuring its path delays. By appropriate selection criteria of the paths and suitable algorithmic processing, the authors manage to avoid the pitfalls of an ECC-based fuzzy extractor altogether, i.e., their design is primarily based on a linear transform as compensation technique in addition to a coarse-grained quantization. Compared to other PUF designs, apparently more effort was spent towards a well-designed PUF primitive. Vast amounts of empirical data support the chosen design rationale.

Taking into account all PUFs listed in Table 3.1, we notice that previous PUF designs aimed at obtaining a binary alphabet as early as possible, sometimes at the cost of omitting a more sophisticated compensation or quantization, and assessing their quality using well-known PUF metrics based on the fractional hamming distance, i.e., Uniqueness and Reliability. While this allowed to reuse all concepts from memory-based PUFs, they may have not fully exploited the potential of the implemented PUF primitives.

To wrap up this discussion, we would like to reference some other concepts relevant for PUF constructions, such as encoding the ordering of RO frequencies [239], as done for example by Maes et. al. in [134]. This technique could still be combined with some of the approaches presented here in this thesis, i.e., instead of using "normalization offsets" that must be stored to remove a structural bias, an ad-hoc differential measurement with subsequent compensation could be used to improve the PUF or provide better design trade-offs, as storing normalization offsets is deemed impractical for real-world applications.

Finally, we point out that methods such as *1-out-of-k masking* [193] are often applied to the output of a comparator-based output of the RO-PUF to only select pairs with a sufficiently large difference in their frequencies, thereby making the output more robust but also further decreasing the number of possible output bits. Moreover, as the uncertainty over the structural bias increases due to a larger spatial distance of the compared ROs, this gain in robustness could be owed to the structural bias and adversely effect the entropy. As this applies only to single-bit per primitive PUFs and cannot be applied to the output of an equiprobable or equidistant quantization, we do not take this scheme or related ones into further consideration.

Beyond the scope of the PUF output alphabet, only very few PUF designers have actually attempted resisting physical attacks with their designs. Among these very few publications are [220, 206, 209, 118, 41] (not including the thesis author's contributions). Clearly, much more work is necessary to create truly tamper-evident PUFs withstanding even the most sophisticated attacks.

Additional Remarks on Binary vs. HOA PUF Output. The difference between a binary PUF and a HOA PUF can be further explained by the following analogy: suppose that a text file contains an English text, then the distribution of letters in this file would follow the distribution of letters of the English language. This statement is independent

from how these letters are encoded. Assuming they are encoded as one byte per letter, then the Extended American Standard Code for Information Interchange (ASCII) could be used to represent them, e.g., the letter *a* would then be represented by the byte `0x61` in hexadecimal notation. However, it would also be possible to come up with a completely different mapping where *a* is represented by `0xAA`. Consequently would be the bit representation completely different to the ASCII case.

Interpretation of the English text must therefore be solely based on the interpretation of the letters, not their binary representation. Now, when considering the text file as the output of a HOA PUF, it is evident that studying its properties with previously existing binary-oriented methods is not an adequate approach. Furthermore, for PUFs with a binary output typically an i.i.d. assumption is made, i.e., all bits have been generated from the same source and therefore have the same statistical properties. This is in contrast to HOA PUFs where the i.i.d. assumption cannot be made at a bit level, as the individual bits of an encoded byte do not fulfill this property. Moreover, the mapping from symbols to bits is only a by-product of representing the symbols on common processing architectures. In fact, the binary representation of the symbols may be completely separated from the symbol's meaning. Hence, while applying methods designed for binary PUFs to the output of a HOA PUF may generally be done, the resulting output would be completely misleading and of no use. In the context of PUFs, this necessitates a new approach to ECC for the PUF output (cf. Part III) and complementary assessment tools (cf. Part IV). The construction leading to this type of PUF is explained in Part II.

Chapter 4

Higher-Order Alphabet PUF from Tamper-Resistant Enclosures

Based on the previous analysis, a complementary approach of how to construct a PUF is proposed where the output is no longer interpreted as a binary alphabet but as symbols of a higher-order alphabet. This has been a by-product of developing a tamper-resistant enclosure and the techniques involved are of general nature, i.e., the same principles and ideas could be used to construct higher-order alphabet PUFs from other primitives. At first, an overview of the intended architecture is introduced which is based on the publication in [95]. It comprises four domains that are crucial for the overall functionality: physical domain, analog domain, digital domain, and application domain. However, the focus is on how the structure of this PUF is constructed, i.e., up to the point of the quantization. This chapter is primarily based on joint work published in [95, 97] with the thesis author as principal author, whereas [95] presents an envelope and [97] a cover as tamper-resistant enclosure. For both publications, Johannes Obermaier was primarily concerned with the simulation of the enclosure's physical structure, the specifics of the measurement circuit, and how to integrate its proof-of-concept implementation into FreeRTOS. In contrast, the thesis author conceived the overall system architecture, ranging from the physical layout and its stochastic model, over to the required processing such that a suitable input for the subsequent quantization and ECCs is created, the specifics of these schemes, and secure bootstrap mechanisms of such a device.

Contents

4.1 Architecture Overview

In the following, we briefly discuss the simplified attacker model which we first had in mind when designing the system-level security architecture. This is done in Section 4.1.1. Afterwards, we introduce the components of the designated architecture as shown in Figure 4.2. To protect a host system, e.g., a HSM, two building blocks are required: an enclosure with capacitive sensors obstructing physical access to the system and its corresponding evaluation unit.

4.1.1 Simplified Attacker Model

Since system-level tamper-resistant architectures based on PUFs have not been well investigated yet, our goal is not to present a final solution for absolute security (which may not be possible anyway) but to investigate concepts that may ultimately lead in that direction. Hence, we first and foremost want to achieve a reasonable and verifiable level of security for MCMs to possibly pass certification without battery-backed mechanisms. Due to that and also to limit the complexity of this work, we focus primarily on attempts to physically penetrate the enclosure, i.e., its mesh. More specifically, we assume penetrations to be at least 300 µm in diameter. Other attacks such as removing the enclosure are deemed impractical and would result in severe damage, as explained when introducing the packaging concept in Section 4.2.1. As result of an attack above the given diameter, the system needs to be able to ensure that it becomes immediately inoperable and recovery of its sensitive data must be infeasible. In the following, we briefly justify our reasoning for this diameter.

Standards for Security Certification. The Derived Test Requirement (DTR) A1 of [162]* demands a "Minimum width/separation (of active traces) of 6 mil" for an enclosure's mesh which translates to 300 µm based on geometrical considerations as illustrated in Figure 4.6a. The same principles must be adhered to for other layouts, as shown in Figure 4.6b.

Please note that within the context of security certifications, just making a hole is not considered an attack [105]. Instead, holes *and* subsequent attacks that lead to *successful* exploitation of a system are rated on the scorecards. It is crucial that the determined attack potential (in points) is above a certain threshold to pass certification, i.e., there will always be some attack possible, the only question is how much effort needs to be spent. Hence, we consider attempts of only making a hole as an evaluation-level analysis only without real-world significance. For practical exploitation, we assume that the underlying system has been designed such that either multiple smaller holes of 300 µm would be required or an increased drill diameter of $3 - 6$ mm for a single hole, e.g., to allow decapsulation of an IC which appears impractical through a 300 µm hole of several millimeters depth.

Commercial Products. Another approach to limit the relevant diameter is to look at previous products and commercial brochures. According to our findings regarding the

* This document is officially available only under Non-Disclosure Agreement, nevertheless it can be found, e.g., on Baidu. It must not be confused with the *public* document "PCI PTS POI SR v4" on the Security Requirements (SR) of PCI PTS POI which does not include such detailed information.

GORE envelope [151], a track width of 300 µm and spacing of 300 µm was used, i.e., the smallest diameter to detect *at best* is of the same size. Another security housing that was previously available [20] was advertised to detect drills of 500 µm. Other solutions such as the one employed in HP Atalla's HSM have an even larger track width and spacing of 1 mm [151]. Hence, to the best of the authors' knowledge, there are currently no commercial products offering a smaller mesh structure other than what is required to meet the 300 µm at best. There are likely solutions in place that are not available on a commercial basis.

Available Tools. Mostly the diameter of drills and *shaft*[*] diameter of micro-probing needles matters. While a micro-needle's tip is usually very small (~ 1 µm), it is also very short and not suited to reach far inside an enclosure. In contrast, the shaft is often several millimeters long but also much larger, e.g., [55] offers tungsten needles with a copper shaft of 500 µm in diameter, i.e., a shaft already larger than the considered hole diameter. This shaft diameter does not account for a small gap around it, i.e., the hole itself would need to be slightly larger than 500 µm since a perfect alignment and insertion angle of 90° are difficult to achieve in practice.

Mechanical drills are easily available down to 100 µm as later illustrated in Figure 13.13c. However, as a rule of thumb [221], a micro-drill's diameter versus its effective drill length – determined by its flute length – is a ratio of 1:15, e.g., a drill with 0.3 mm in diameter has an effective drill length of 4.5 mm at best. Therefore, such drills must be considered as part of the later security analysis in Part V. In contrast, we consider laser ablation or laser drilling not as a viable option as it typically creates cone shaped holes, i.e., the top hole will be typically larger than the bottom hole as illustrated in Figure 4.1 for a layer of 50 µm polyimide. Considering the aspect ratio of hole diameter to material thickness, in addition to the aspect ratio of top to bottom hole, it appears impractical to use laser drilling for the layer stack-up later presented in Table 4.1, i.e., even when assuming an overly optimistic hole diameter to material thickness ratio of 1:1 this still would create a hole of within the range of the considered 300 µm. Regarding chemical solvents, we cannot make an educated statement as it would exceed our own expertise.

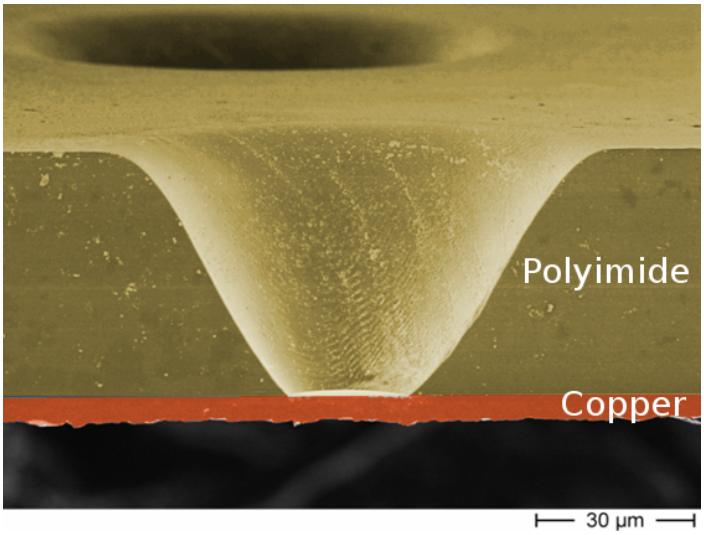


Figure 4.1: Cone-shaped hole as result of a laser ablation process (with courtesy of Fraunhofer EMFT). Specific shape and ratio depend on laser and material used.

[*] In some data sheets this part of a micro-probe is called shank instead of shaft.

Conclusions on Attacker Model. Following these arguments, we are of the opinion that a 300 µm hole diameter is a reasonable choice for most practical applications and in accordance to current industry standards. As long as the enclosed system follows best design practices in this domain, such as routing all signal layers on the inner layers of a PCB, only using Ball Grid Array (BGA) components, buried vias, etc. it is difficult to foresee a successful exploitation with only few points on the score card of a security certification process, if it is possible at all when not deactivating some countermeasures for the evaluation process. This is particularly true as such an enclosure is only one layer of a thorough Defense-in-Depth (DiP) concept. Therefore, we still require countermeasures at the appropriate level to counteract follow-up attacks such as LFI or EMA. Defeating these countermeasures would then in turn require more rework, requiring a larger degree of freedom to access the targeted IC which however is hindered by the enclosure. Other types of attacks are later briefly discussed in Section 13.2.3.

4.1.2 System Overview

The overall system is depicted in Figure 4.2, following the data processing concept in Figure 4.3. The envisioned enclosure, e.g., an envelope or cover, comprises capacitive sensors that act as a PUF and provide the basis for a cryptographic key. During each device start-up, the same key can only be extracted if the enclosure has not been tampered with. While manufacturing the device, this key is used as key-encryption-key (KEK) to encrypt and authenticate CSPs or other sensitive data of the enclosed device. The thusly protected data is stored in non-volatile memory, since an attacker can neither gain information from it nor change it in a useful way without damaging the enclosure, thereby destroying its key.

Upon power-on, the system self-authenticates and is decrypted. Once the device is running, the same sensors that extracted the PUF properties from the enclosure now continuously monitor it. In case of an attack during runtime, an alarm is raised to trigger the zeroization of sensitive data which is temporarily stored in volatile memory for processing it. In addition to that, several more countermeasures should be used to make recovery from such an event even more difficult, e.g., by blowing fuses and attempting zeroization of permanent data such as the PUF's helper data.

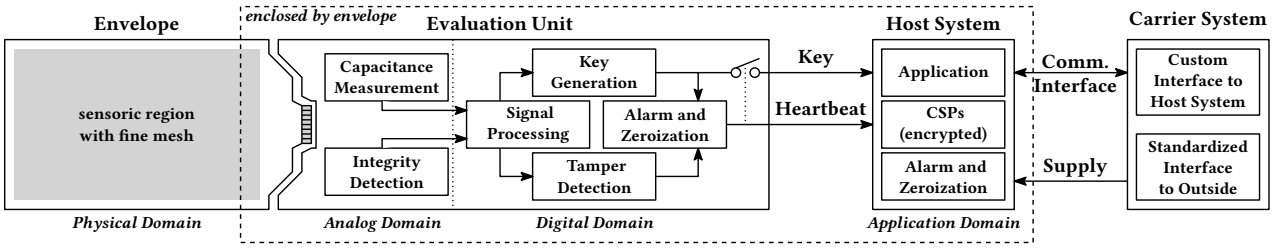


Figure 4.2: Host system protected by a tamper-resistant enclosure. For the given example, the enclosure is assumed to be an envelope.

Enclosure. For the given example in Figure 4.2, the enclosure is created from an envelope that is comprised of a foil containing a mesh of fine conductive tracks. The mesh represents the PUF to derive a cryptographic key by evaluating the capacitance measurements over the entire sensoric region. It also acts as an opaque barrier around the fully enclosed device. The envelope's sensoric region contains overlapping tracks that represent the electrodes which work as capacitive sensors. These tracks are subject to minuscule manufacturing

variations in terms of surface roughness and physical dimension due to etching and related manufacturing processes [21, 222]. As a result, each overlap between electrodes represents a capacitance that cannot be accurately predetermined. Therefore, this concept relies on the intrinsic variation of a standard manufacturing process in contrast to artificially introduced randomness of, e.g., the Coating PUF [206].

Evaluation Unit. This unit connects the enclosure to the host system. We refer to this as a separate unit primarily out of the reason for clarity of the explanations. In fact, it could be integrated into the host system which appears as the most secure but also least flexible approach in terms of development, as the process of incorporating it most likely results in changing the design of the host, too. Therefore, we later implemented the evaluation unit in a dedicated microcontroller, controlling the PUF data processing concept, including but not limited to:

- Analog domain: a single analog front-end that unifies distinct measurement concepts for the capacitance and integrity detection, i.e., they are sharing the same circuitry
- Digital domain: signal processing, PUF key generation, and runtime tamper detection logic including zeroization upon detection of physical intruders
- Data interface: to exchange information with the host, e.g., to serve as a decryption oracle, i.e., encrypted data is transferred to the evaluation unit and decrypted data is returned. Please note that this interface is within the physical security boundary, i.e., enclosed and protected by the enclosure.
- Heartbeat interface: with two independent alarm signals that are monitored by the host system during runtime to thwart "one-shot" intrusion attempts. This is for example, a Pulse-Width Modulated (PWM) signal with randomized frequency to which the host synchronizes and a static alarm signal which is active high.

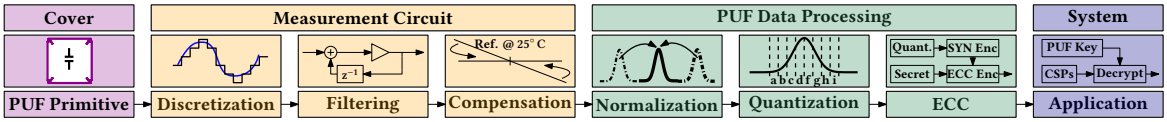


Figure 4.3: PUF data processing concept of the evaluation unit.

Host System. After each power-on, the host system synchronizes to the heartbeat signals and only then starts the interaction with the evaluation unit, e.g., to request the decryption of its firmware or additional CSPs using the key derived from the enclosure. Direct access to the key is denied to prevent software-based extraction. If the alarm signals indicate a tampering attempt, a zeroization is carried out. Following this generic approach, it is possible to implement a wide range of applications that may be unaware of their physically protected execution environment.

4.2 Physical Domain

We foresee an enclosure design based on an envelope or cover, as illustrated in Figure 4.4, such that its surface which is exposed to an attacker is fully covered by the sensoric region of the enclosure, i.e., the portion containing the tamper-detecting sensors. This provides

a comprehensive resistance to attacks, as any direct line of attack is obstructed by the sensoric mesh. Moreover, wrapping an envelope around a case has the least impact on the design of the enclosed PCB. In contrast, a cover-based enclosure has benefits w.r.t. improved heat dissipation and less complex assembly, while at the same time having a disadvantage concerning its security, as the cover's seams are more difficult to protect. Please note that unwrapping the envelope or removing the cover in real-world designs is prevented by potting it.

4.2.1 Packaging Concept

The enclosure is either based on an envelope as illustrated in Figure 1.5 with a wrapping technique similar to [102] or a cover-based solution as sketched in Figure 4.4. In the following, we focus on a cover-based enclosure to protect the PCB's top which is for active and the PCB's bottom which is for passive components. A corresponding top and bottom cover are used such that the majority of the surface which is exposed to an attacker is fully covered by the sensoric region contained in the covers. Both covers and their auxiliary mounting components such as the stiffener frame are attached to the PCB by at least two different mechanisms: firstly, by adhesives with high mechanical strength and good chemical resistance, secondly, by mechanical means such as screws. The covers themselves are additionally connected to the PCB using a secure seam which is beyond the scope of this thesis and the simplified attacker model. Since the physical assembly of the covers is intertwined, removing them or prying them open without causing severe damage to one or the other is unlikely. To further harden the design and increase damage upon cover removal, we intend on using a conformal coating or potting resin for real-world designs which we omitted for our study.

As illustrated in Figure 4.4, there is sufficient space beneath the top cover to internally mount a heatsink to dissipate the heat. Moreover, the heatsink acts as an additional physical barrier once the attacker gets passed the cover itself. Since the distance between the top cover's surface to the PCB is 7.4 mm, we assume that at least a drill diameter of 0.5 mm must be used for practical exploitation, i.e., a perfect attacker would know the best spot to attack, drill a hole to fully reach inside, decapsulate the area of the IC where the PUF data processing takes place, and extract its raw measurement data to reconstruct the PUF key. Such attacks must therefore be counteracted at the IC-level, too. However, in contrast to previous battery-backed solutions, it is no longer possible to only tamper with PCB-level tracks to defeat the security mechanism [151]. Instead, it is highly probable that the advanced evaluation logic at the IC-level must be attacked, too. This is a significant advantage of PUF-based enclosures over battery-backed approaches and their relatively crude but energy saving determination of the enclosure's physical integrity.

To complement the security provided by the covers, a vertical protection structure inside the PCB was designed to prevent attacks via its sides. Hence, any direct line of attack is obstructed either by the capacitive sensoric mesh or requires difficult angles to attack from which in turn are obstructed by the vertical protection structure. The packaging concept therefore already provides a comprehensive resistance towards attacks on a practical level.

Aside from the physical assembly which is designed to resist physical attacks, we still envision to use various other sensors, e.g., light, voltage, pressure contacts, and brittle components such as vias that easily get torn apart, to detect adversarial operating conditions upon power-on. An actual exploitation of the whole system therefore not only relies on defeating the tamper-resistant covers but also on successfully disabling additional

layers of physical security on the inside which would require multiple holes to be made, thereby necessitating further damage to the covers and/or requiring a more advanced effort. Hence, an attacker will likely require more than one device to first design the best attack (identification) before attempting an actual attack (exploitation). This aspect is reflected on the scorecards during a certification process [105].

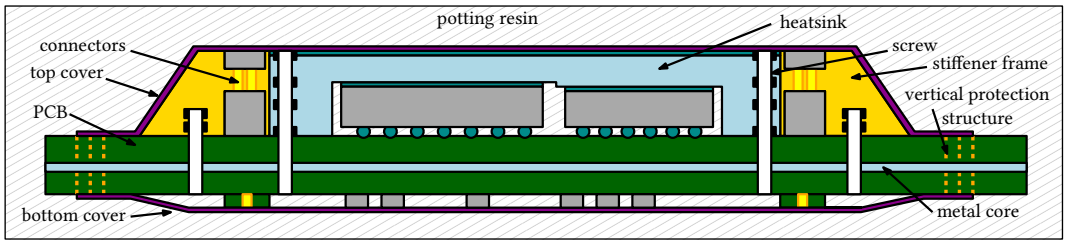


Figure 4.4: Packaging concept for tamper-resistant enclosure based on cover.

4.2.2 Layer Stack-Up of the Enclosure

Designing a layer stack-up depends on the limitations of the manufacturing technology and the targeted sensor type, i.e., using non-standard manufacturing technology helps to tailor the materials towards security, while standard manufacturing processes are designated to provide a more economic solution. Thus far, tamper-respondent enclosures are primarily based on resistive sensors that are manufactured by a silk-screen printing process, i.e., fine tracks are printed on a flexible sheet and the resulting mesh is considered as resistors in the corresponding evaluation circuit. However, this has several disadvantages when compared to capacitive sensors, especially for devices that can be fully powered off, as the resistance of a track could be measured and replaced with a matched resistor which would result in a bypass difficult to detect. Moreover, resistive sensors only detect changes within their own tracks. In contrast, capacitive sensoric regions are conceptually less prone to bypassing of their tracks due to the small capacitances in the range of femtofarads. Furthermore, parasitic capacitances towards surrounding objects influence the measurement. Hence, not only are tracks considered part of the measurement but so are nearby layers and objects.

For both cover and envelope, we aim at a self-contained capacitive sensor to sense the intrinsic manufacturing variations of the mesh. This is achieved by implementing two layers of electrodes that are enclosed with a grounded shield to provide a defined boundary condition and prevent interference from the inside or outside, as listed in Table 4.1. One layer of electrodes is named “Tx” while the other layer contains the corresponding “Rx” electrodes. As will be detailed in Section 4.3, the “Tx” electrodes are driven by an excitation signal and the “Rx” electrodes act as receivers. The capacitance between each Tx and Rx electrode is quantified as the “mutual capacitance”, as noted in Table 4.1. Since the parasitic capacitance towards the shield is rather large compared to the mutual capacitance, partially removing or not grounding the shield already degrades the measurement up to the point that it no longer works. For connectivity to the measurement circuit, the cover requires an additional layer for connectors, resulting in a total of five conductive layers. This is not the case for the envelope. For our implementation of the cover, we later exemplarily use flexPCB technology which is a lithographic process and therefore allows a much smaller track width when compared to silk-screen printing. In case of the cover, all electrically conductive tracks are therefore made from copper providing an inherently lower security level due

to improved reparability when compared to more customized materials. In contrast, the envelope is based on a fully customized technology that allows to mix materials and layers, e.g., a carbon-paste based shield and copper tracks, or a fully tailored solution with PEDOT tracks [160] and a carbon-paste shield. This material mix including the carbon-paste based shield has been proposed by the thesis author to overcome limitations regarding flexibility of the copper shield and to enhance integration with potting material for security reasons. Alternative manufacturing processes that are of interest include [171]. Here, a wave-like surface structure is created that is designated to improve flexibility of the carrier material.

The manufacturing process of this layer concept can still be done differently. Assuming an envelope-based stack-up with four layers, two distinct options are available. In Figure 4.5a, an adhesiveless carrier is used as a start, i.e., a patterning process is used on its top and bottom to first create the electrode structure, and subsequently additional bonding sheets are used to add a shielding layer. This approach requires a via technology to interconnect both electrode structures on the top and bottom of the adhesiveless carrier. Since vias are inherently limited in their size due to the holes that can be created (cf. Section 4.1.1), it is best to avoid them altogether.

By using a different manufacturing process as depicted in Figure 4.5b, it is indeed possible to avoid traditional via technology. Here, a carrier is used and subsequent layers are printed on top, possibly using materials that are doped with significantly opposing particles as done also for the Coating PUF [206]. Vias can now be created by a slope/incline of printed material, such that a natural crossover from top to bottom or vice-versa exists. This is the preferred method but requires more advanced manufacturing capabilities which could not be accessed as part of the project.

Table 4.1: Exemplary layer stack-up for tamper-resistant PUF enclosures (based on flexPCB).

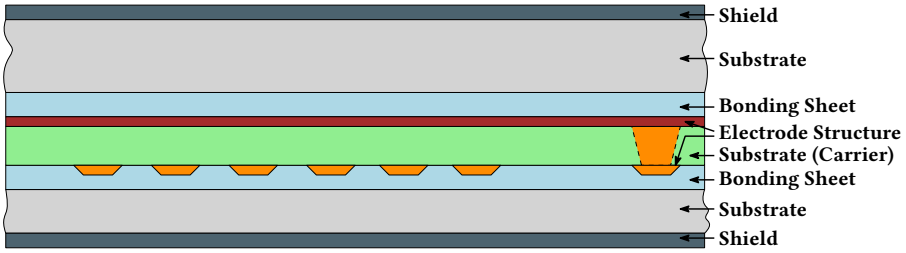
Layer	Height	Description	Comment
1	27 µm	Shield	Facing to environment
	52.5 µm	Bonding/Insulation	⇳ Parasitic capacitance C^{P}
2	24 µm	Tx electrodes	Driven electrodes
	12 µm	Polyimide substrate (carrier)	⇳ Mutual capacitance C^{M}
3	24 µm	Rx electrodes	Receiving electrodes
	52.5 µm	Bonding/Insulation	⇳ Parasitic capacitance C^{P}
4	12 µm	Shield	} Facing inside (to PCB)
	12 µm	Polyimide substrate	
5	27 µm	Connectors and routing	

4.2.3 Sensor Design (Physical Layout)

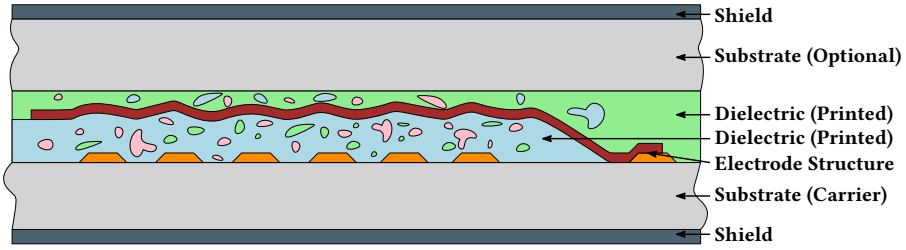
The following requirements were considered in order to design a suitable sensor layout:

(i) The layers comprising the electrodes must be covered completely with the intended sensor structure, thereby avoiding blind spots where attacks would go undetected.

(ii) If the enclosure is damaged in one spot, this should result in more than one destroyed sensor, i.e., to make this attack more easily detectable, e.g., by realizing an interconnected sensor arrangement.



(a) Stack-up with fixed layers (requires vias).

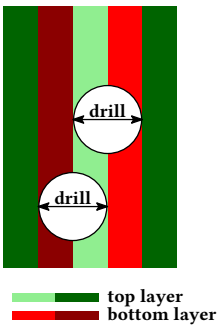


(b) Stack-up with printed layers (vialess approach).

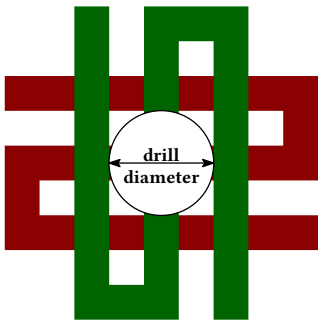
Figure 4.5: Comparison of different manufacturing technologies (in cooperation with Fraunhofer EMFT). The variant shown in Figure 4.5a was chosen as a start, since relying mostly on tested processes.

(iii) The sensor structure of "track-space-track" (or vice-versa) must be smaller than the diameter of expected attacks (cf. Figure 4.6).

(iv) A layout randomization must be available in terms of the enclosure's electrical configuration (physical randomization is deemed too complex due to various other requirements).



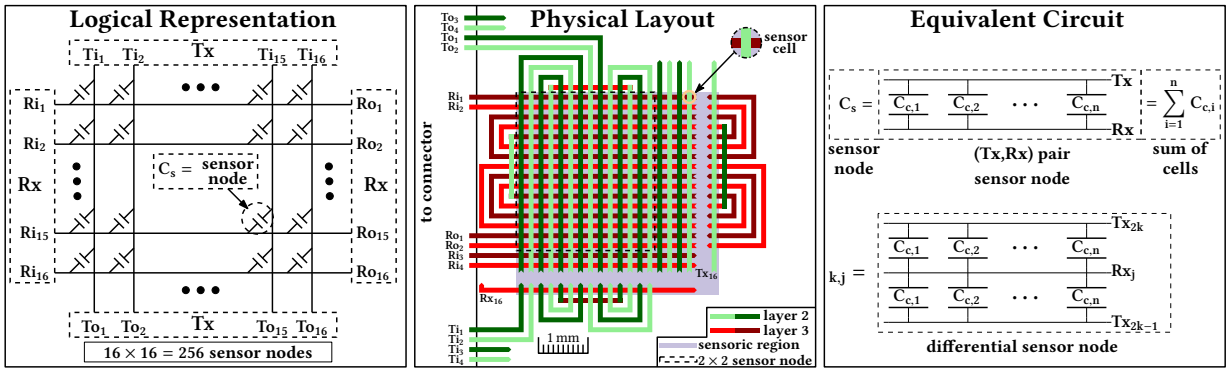
(a) Tracks with gapless design across layers.



(b) Layout variant with visible gaps (chosen layout).

Figure 4.6: Geometrical considerations of track width vs. drill and laser diameter.

To address these, we envision a sensor layout with a structure size of 100 µm line and space as shown in Figure 4.7b, i.e., $3 \cdot 100\,\mu m \leq 300\,\mu m$. Creating small structures increases the difficulty of attacks and improves manufacturing variations, as shown in Figure 4.9. However, since the structure size is small, contamination during manufacturing is possible, resulting in short circuits. Moreover, some manufacturing steps may break electrode



(a) Logical layout. **(b)** Physical layout. **(c)** Equivalent circuit.

Figure 4.7: Different representations of the chosen layout.

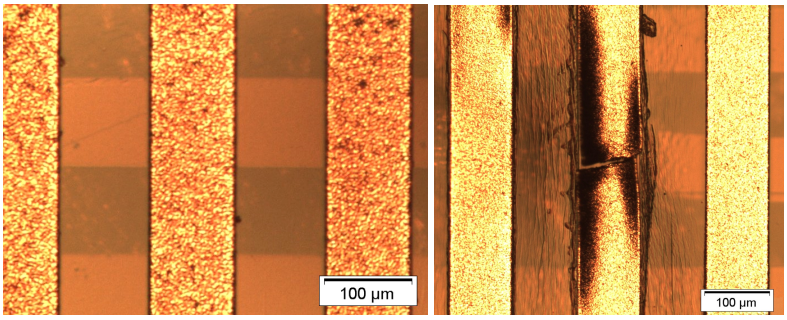
tracks, resulting in open circuits. Unfortunately, both effects sometimes occur as shown in Figure 4.8b and 4.8c. At the time of device assembly, it is therefore critical to verify that each enclosure is free of such defects. This is considered as a mesh with "full integrity" which provides assurance that the whole sensoric surface contributes to the PUF.

To detect open circuits, the layout in Figure 4.7b allows checking the electrode's continuity by forming a loop, i.e., both *input* and *output* of an electrode are routed to the connector, denoted as Ri/Ro for Rx and Ti/To for Tx electrodes. To also check for short circuits, the electrodes are interleaved such that each neighboring track can be driven independently. Figure 4.7 shows the resulting advanced layout and its various representations, which can easily be scaled to cover a larger area by increasing the number of windings and/or electrodes. The layout is essentially a grid of overlapping electrodes, whereas the routing of the electrodes is bifilar such that the smallest unit is a 2×2 *node square*, i.e., two Tx electrodes overlapping with two Rx electrodes. Two criteria are later important to assess the layout, they are named:

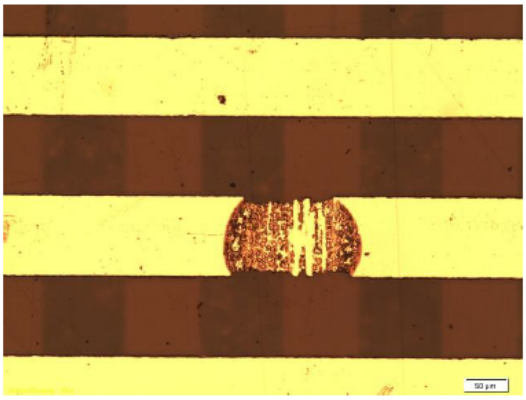
Definition 4.2.1 (Fine-grained drill sensitivity) *The minimum drill diameter guaranteed to destroy at least one track of a mesh on at least two separate signal layers is termed Fine-Grained Drill Sensitivity (FGDS).*

Definition 4.2.2 (Coarse-grained drill sensitivity) *The maximum drill diameter to not exceed the limits of a single node square upon successful repair is termed Coarse-Grained Drill Sensitivity (CGDS).*

To ensure an optimized CGDS later on, layout randomization is a necessity. The basic idea for layout randomization is as follows: Given a sensoric region with an existing number of electrodes and corresponding connectors, then each electrode is split in half and the number of connectors is doubled accordingly (this process may be repeated several times). These new electrodes must be connected to the evaluation unit. Here, depending on the level of sophistication of the implementation, the electrodes can be either statically (fixed assignment) or dynamically (based on a challenge) reconnected again prior to the actual measurement process, i.e., as long as previous design rules are followed, it is possible to connect *arbitrary* pairs of electrodes across the whole sensoric surface while ensuring the same electrical parameters for the previously designed measurement circuit. Thereby, not only a layout randomization is realized but also an improved CGDS is obtained, as each



(a) Electrodes with full integrity. **(b)** Electrodes with open circuit.



(c) Electrodes with manufacturing defect.

Figure 4.8: Magnified sections of the mesh with courtesy of Fraunhofer EMFT (here: envelope). Clearly visible is also the minuscule manufacturing variation.

2×2 node square will be much smaller. Hence, this is jokingly called a "Puzzle PUF", as different node squares from across the sensoric region are assembled and put together as one electrode pair for the measurement. Providing a challenge to select the specific randomized layout configuration is therefore a natural extension of this PUF.

4.2.4 Stochastic Model of a Sensor Node

To determine the physical parameters of the sensor layout, we analyze the capacitance C_s of a single sensor node (cf. Figure 4.7a) based on its simplified equivalent circuit in Figure 4.7c. Each of the n overlaps (sensor cells) between the electrode tracks represents a tiny capacitor in parallel. C_s is therefore the sum over the capacitances $C_{\mathrm{c},i}$. This representation is simplified since it ignores the resistance in series between each sensor cell. However, as long as track resistance is matched, this is a valid initial estimate based on our practical experience.

In the following, we assume $C_{\mathrm{c},i} \sim \mathcal{N}(\mu_\mathrm{c}, \sigma_\mathrm{c}^2)$ as i.i.d. Recall that adding two Gaussian random variables results in a Gaussian distribution with the sum of means and sum of variances. Therefore $C_\mathrm{s} \sim \mathcal{N}(n \cdot \mu_\mathrm{c}, n \cdot \sigma_\mathrm{c}^2)$, i.e., $\mu_\mathrm{s} = n \cdot \mu_\mathrm{c}$ and $\sigma_\mathrm{s}^2 = n \cdot \sigma_\mathrm{c}^2$. According to the weak law of large numbers we then compute the respective means of the sensor cell

$$\overline{C_\mathrm{c}} = \frac{C_\mathrm{s}}{n} \;,\; \overline{\mu_\mathrm{c}} = \frac{\mu_\mathrm{s}}{n} \;,\; \overline{\sigma}_\mathrm{c}^2 = \frac{\sigma_\mathrm{s}^2}{n} \tag{4.1}$$

and obtain an equation that depends on n which is the number of parallel cells combined to a sensor node, i.e., $C_\mathrm{s} = n \cdot \overline{C_\mathrm{c}}$.

Validating the Assumptions. Independence of variables: Other publications such as [222] and [21] show that besides of local variation there is also global variation across manufacturing panels of PCBs. This results in a capacitance gradient and therefore a global bias. This applies to the technologies selected in Part V, too. To counteract this effect that would result in a varying nominal capacitance, we use a differential measurement as detailed in Section 4.3. Measuring the difference between two pairs of nodes in close vicinity isolates the local variation and minimizes the global effects. This local variation is illustrated in Figure 4.9. With regard to having normally distributed variables, we refer to the central limit theorem, i.e., the sum of many independent cells combined to a node tends towards a normal distribution.

Estimating the Entropy. To estimate the entropy of the thus far continuous PDF of a sensor node, we need to consider the resolution Δ_M of the measurement circuit. As security objective, we target $\Delta_\mathrm{M} \leq \overline{C_\mathrm{c}}$, i.e., removing a *single cell* from the capacitance C_s of a sensor node would be detected with high probability. Please note, if only considering attacks above the targeted diameter to protect against, removing a single cell is impossible since an attack always cuts off multiple overlaps in the layout. Based on later results, we select $\Delta_\mathrm{M} = 1\,\mathrm{fF} \leq \overline{C_\mathrm{c}}$.

Measuring the capacitances is only a first step. Subsequent processing includes a quantization with bin size Δ_Q [91]. Since we are interested in the fundamental properties of the design only, we proceed with Δ_M and do not take the specifics of Δ_Q into account. According to [33], the Shannon entropy H^Δ of a discretized Gaussian random variable is given by

$$\mathrm{H}^\Delta = \mathrm{ld}\left(\frac{\sigma_\mathrm{s}}{\Delta_\mathrm{M}} \cdot \sqrt{2\pi\mathrm{e}}\right) \tag{4.2}$$

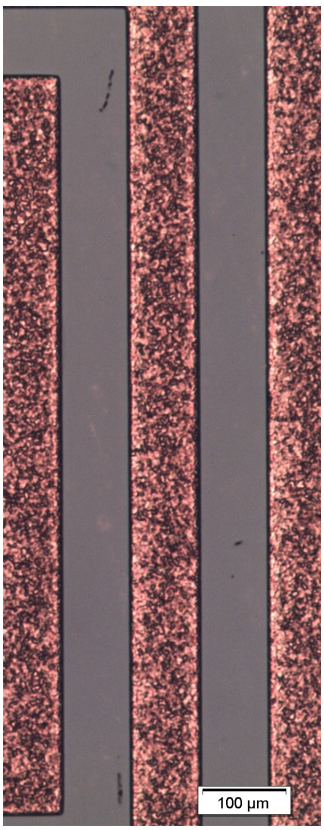


Figure 4.9: Close-up of the bumpy tracks illustrating manufacturing variation (with courtesy of Fraunhofer EMFT). This is the Rx electrode layer, whereas the Tx electrode layer was not manufactured yet.

As design target for our later implementations, we aim at $H^{\Delta} = 5$ bit for the given Δ_M and then solve for σ_s which is 7.7 fF. This value can be verified empirically once a statistically relevant number of samples is available. Using Equation 4.1, the minimum sensor cell count for a design is $\min(n) = \frac{\sigma_s^2}{\sigma_c^2}$. However, this can only be calculated if σ_c is known, i.e., empirical data is already available. Alternatively, the cell capacitance may be determined using a simulation tool, as done by Johannes Obermaier. Additionally, a reasonable assumption for the expected variation needs to be made. In our case, we used: $\overline{C_c} = 18.18$ fF and $\sigma_c = 1.6\%$. For the same H^{Δ} and Δ_M this yields $\min(n) = 713$ which allows partitioning the enclosure accordingly, i.e., selecting the number of Tx/Rx electrodes.

4.3 Analog Domain

In the following, we focus on the capacitance measurement that incorporates C_s and its PDF as illustrated in Figure 4.10. Here, C^N is the nominal capacitance and C^V the variation from the manufacturing process. One goal of selecting a measurement technique is to optimize its sensitivity towards C^V. This is mainly controlled by two parameters: first by the number of steps the capacitance measurement system resolves, expressed by ENOB, 2^{ENOB}; secondly by the maximum of the capacitance, denoted as C_{max}. The lower bound ΔC_{min} is then defined as $\Delta C_{min} = \frac{C_{max}}{2^{ENOB}}$. Subsequently, we assume ENOB is constant and analyze C_{max} in more detail. Let $C^M_{i,j} = C^N + C^V_{i,j}$ be the mutual capacitance between Tx_i/Rx_j and $C_{max} = \max_{i,j}(C^M_{i,j})$.

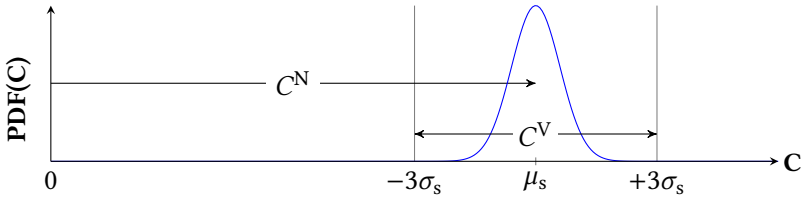


Figure 4.10: Exemplary PDF of C_s with mean μ_s.

As C^V is small compared to C^N, this causes $C_{max} \approx C^N$. As a consequence, ΔC_{min} primarily depends on C^N which leads to $\Delta C_{min} > C^V$ for even a small number of sensor cells, as C^N increases linearly in n, while the variation increases only by $\sqrt{n} \cdot \sigma_c$. Thus, no variation could be measured. We solve this using a differential measurement. For an even i, the electrodes Tx_{i-1} and Tx_i are routed differentially. They form the Tx pair (Tx_{2k-1}, Tx_{2k}), for $k \in \{1, 2, \ldots, N_{Tx}/2\}$. All Rx are used as single electrodes with (Rx_j), for $j \in \{1, 2, \ldots, N_{Rx}\}$. Hence, the differential capacitance is $\gamma_{k,j} = C^M_{(2k-1),j} - C^M_{(2k),j} = C^V_{(2k-1),j} - C^V_{(2k),j}$.

Accordingly, the resolution no longer depends on C^N which ensures an improved sensitivity where also the dynamic range is well-adjusted to C^V. Extracting only C^V coincides with the assumption that C^N is the same for neighboring differentially-routed electrodes, i.e., global variation causing different C^N over larger distances are ignored. Unfortunately, improving the sensitivity comes at the price of halving the number of measured capacitances to extract information from, i.e., $N_{Diff} = N_{Rx} \cdot (N_{Tx}/2) = 16 \cdot (16/2) = 128$. However, the resulting PDF of the differential capacitance γ is $\mathcal{N}_\gamma(0, \sqrt{2} \cdot \sigma_s)$ and therefore Equation 4.2 can be rewritten as

$$\mathrm{H}^{\Delta} = \mathrm{ld}\left(\sqrt{2}\right) + \mathrm{ld}\left(\frac{\sigma_{\mathrm{s}}}{\Delta_{\mathrm{M}}} \cdot \sqrt{2\pi\mathrm{e}}\right) \tag{4.3}$$

Hence, the maximum theoretical entropy of the overall enclosure in our case is $128 \cdot 5.5\,\mathrm{bit} = 704\,\mathrm{bit}$. Please note, this is only an example to illustrative how the design process of such an enclosure is done. It does not account for degradation due to bias in the data, environmental conditions, noise, etc.

Additional Considerations. Critical in the design process is the propagation effect upon tampering vs. the entropy variation per measurement node. For attacks that are not based on repairs, the effect of a cut off is rather severe, since C^N is removed, i.e., there is little doubt that such an attack would go unnoticed. However, if repairs are done, then it is likely that C^N can be approximately restored and security depends more on the irrecoverable destruction of C^V. The optimal trade-off between C^V and C^N therefore must be investigated in the future, possibly by using a layer stack-up as presented in Figure 4.5b that allows to specifically tune these parameters which is not possible when using standardized manufacturing processes.

Practical Implementation. A sophisticated measurement system is required to later resolve the minuscule C^V component of the mutual capacitance. Two potential measurement principles were identified that could be used for this task, both of which have been practically tested by the thesis author [42, 152]. Based on ideas rooted in [126], a customized security sensor IC was developed [42] to measure the differential capacitance by integrating over the charge when applying suitable Tx excitation pulses, i.e., a single Tx electrode pumps charge into all Rx electrodes and a pairwise-differential evaluation is used on the Rx side, allowing for full parallelization with only few hardware resources. Another technique attempts to create an in-situ differential capacitance, also called on-cell capacitance [236]. Following the complementary excitation idea of [236], a pair of Tx electrodes is excited with an antiphasic signal such that a complex current representing the differential capacitance is created at the evaluated Rx electrode. This approach has been followed in our publication in [152] with the notable difference being that the resulting signal is evaluated in the frequency domain instead of the time domain, thereby avoiding the pitfalls of using too many analog components. Moreover, the difference is created within the enclosure as opposed to circuit components.

Both approaches have their pros and cons regarding their implementation, e.g., the method of [42] can be realized in an IC with relatively moderate resources while the method of [152] is more generic and could be implemented with discrete components, as well as within an FPGA, but also an IC at the expense of a more complex engineering process. In our practical experiments, having both implementations at hand, the solution of [152] turned out to provide a better performance. However, this does not account yet for the results of ongoing iteration processes and subsequent tests which is why a final verdict on either solution is not possible, as both approaches are designated to provide better results in the future.

4.4 Digital Domain

Several additional processing steps are required to yield a cryptographic key that is reliable, provides full entropy, and in addition to that offers the property of *tamper-sensitivity*, i.e., even small physical changes should result in a significant change of the PUF's output.

4.4.1 Compensation and Normalization

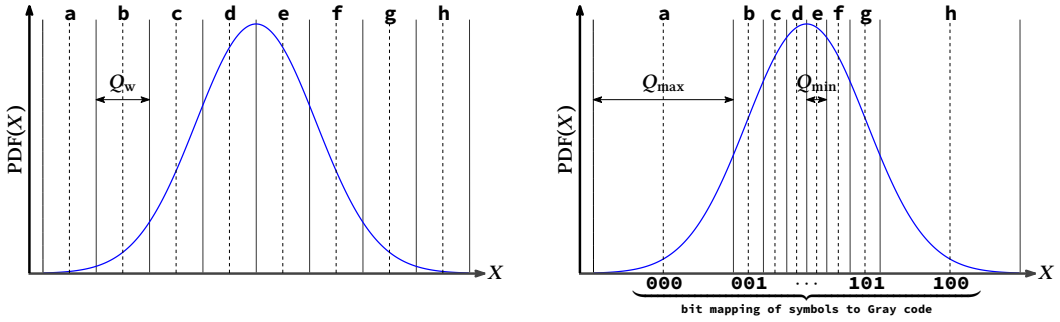
The output of the previous stages is considered as raw differential capacitance data that must be adjusted to account for structural bias and environmental changes such as temperature drift. Removing structural bias is also called "normalization", as for example in [134]. Typically, this would require additional helper data to mitigate the effects of a structural bias. However, as seen later on in the case studies, the structural bias in our case is mostly in such a way that removing the mean of each Tx group also removes the structural bias, i.e., all Rx electrodes measured in parallel are subject to the same bias. Since a shift in these means is the predominant effect of temperature drift this serves as a *simplified* temperature compensating step*, too. Hence, the values prior to the quantization are computed by the following equation

$$X_i = X_{k,h} = \gamma'_{k,h} - \left(\frac{1}{N_{\mathrm{Rx}}} \sum_{r=1}^{N_{\mathrm{Rx}}} \gamma'_{r,h}\right) \qquad h = 1, \ldots, N_{\mathrm{Tx}}/2 \quad \text{and} \quad k = 1, \ldots, N_{\mathrm{Rx}} \tag{4.4}$$

whereas $\gamma'_{k,h}$ is a representative of the previously obtained noisy differential capacitance. The output $X_{k,h}$ is created by subtracting each Tx group's mean. To simplify the notation, the result is reshaped to X_i with $i = 1, \ldots, k \cdot h$.

4.4.2 Quantization and Error-Correcting Code (ECC)

The previously compensated and normalized data is now further processed by an equidistant quantization [91], as explained in full detail in Chapter 6. This is an error-*reduction* technique to mitigate the remaining circuit noise σ_{N} that would otherwise cause frequent changes in the output data. Alternatives would have been, e.g., an equiprobable quantization as applied to the output of the Coating PUF [206] which is typically based on a Gray code, as illustrated in Figure 4.11b. However, the unequal width of equiprobable intervals causes helper data leakage in addition to an uneven tamper-sensitivity as explained in [91] and later on in this thesis. Other approaches to equiprobable quantization include [215, 192, 24] using a partitioning scheme to avoid helper data leakage.



(a) Exemplary equidistant quantization.

(b) Exemplary equiprobable quantization.

Figure 4.11: Different quantization approaches with assignment of *symbols* for the equidistant quantization and a Gray code (bits) in case of equiprobable quantization.

* Developing more advanced temperature compensating schemes are the most crucial step in the design of a tamper-evident PUF.

However, two problems of equiprobable quantization remain. First of all, it is mandatory to precisely know the PDF in addition to its preferred symmetry. This is difficult for some practical scenarios, e.g., within the context of low volume manufacturing as it is typically the case for tamper-resistant enclosures. Secondly, the quantization error is mainly determined by the innermost intervals as illustrated in Figure 4.11b which either results in a relatively high error rate or in a diminished entropy when increasing the width of the two innermost intervals (assuming a relatively uniform noise level across the range of values).

In contrast, an equidistant quantization as illustrated in Figure 4.11a is relatively insensitive to, e.g., shifts of the PDF and also provides a constant quantization error probability across the range of values. It is therefore an attractive choice for practitioners at the downside of a biased PUF output *at the stage of quantization* which needs to be considered in subsequent processing steps. The equidistant quantization works as follows. The width Q_w of the quantization intervals is determined by $Q_\mathrm{w} = 2 \cdot y \cdot \sigma_N$ whereas y is a parameter of choice according to the required reliability. To obtain m-bit PUF responses, PDF(X) is divided into $L = 2^m$ intervals of the form $(\mu + l \cdot Q_\mathrm{w}, \mu + (l+1) \cdot Q_\mathrm{w}]$ where $l = -L/2, \ldots, -1, 0, 1, \ldots, L/2$. Aligning $l = 0$ and μ of the Gaussian distribution leads to the highest entropy output while it is slightly decreased by misalignment depending on the choice of y and the shift. However, due to symmetry reasons of the equidistant quantization this decrease is well-bounded and therefore a robust scheme.

Figure 4.11a exemplarily illustrates the quantization intervals for $L = 8$ and an optimal alignment. Each interval is represented by a *symbol* Q_l in $[0, L-1]$ from a higher-order alphabet. As the measurement of the PUF values X'_i is non-ideal, i.e., affected by noise of the measurement process, values could move to a different interval compared to the time of enrollment. To additionally reduce such errors, the offsets between each value X_i and their corresponding interval center are stored as helper data ${}^{\mathrm{Q}}W$. By following this approach, the probability of a quantization error can be significantly reduced, e.g., by choosing $y = 3.29$ the symbol error-rate is at 0.1% for each node [91]. During PUF reconstruction, this value is then mapped to the quantized PUF response Y'_i, i.e., $(X'_i - {}^{\mathrm{Q}}W_i \in Q_{l_i} \rightarrow Y'_i)$ for $i = 1, \ldots, N_{,\mathrm{nodes}}$.

To obtain a fully robust device, a subsequent error-correction scheme is still required. This is explained in Part III and focuses on the subsequent processing in terms of symbols from a higher-order alphabet as opposed to bits (cf. Figure 4.11).

4.5 Application Domain

In the following, we briefly explain the secure boot process on a conceptual level that was conceived in the years of 2014 and 2015. In addition to that, we describe how an example application could leverage the system's capabilities.

Boot process. The overall system's boot process is depicted in Figure 4.12. Immediately after power-up, two independent heartbeat signals are generated by the evaluation unit to which the host system synchronizes, in particular if the two units are two different ICs. This should prevent rapid "one-shot" attempts to directly interrupt the alarm later on. As a first line of defense, an integrity detection is carried out to verify if the electrodes contain short or open circuits.

We name this Tamper Detection A (TD-A) which is then followed by a capacitive measurement. Both are continuously repeated during runtime, i.e., they take turns. The first differential capacitance measurement after power-up is considered a reference value and

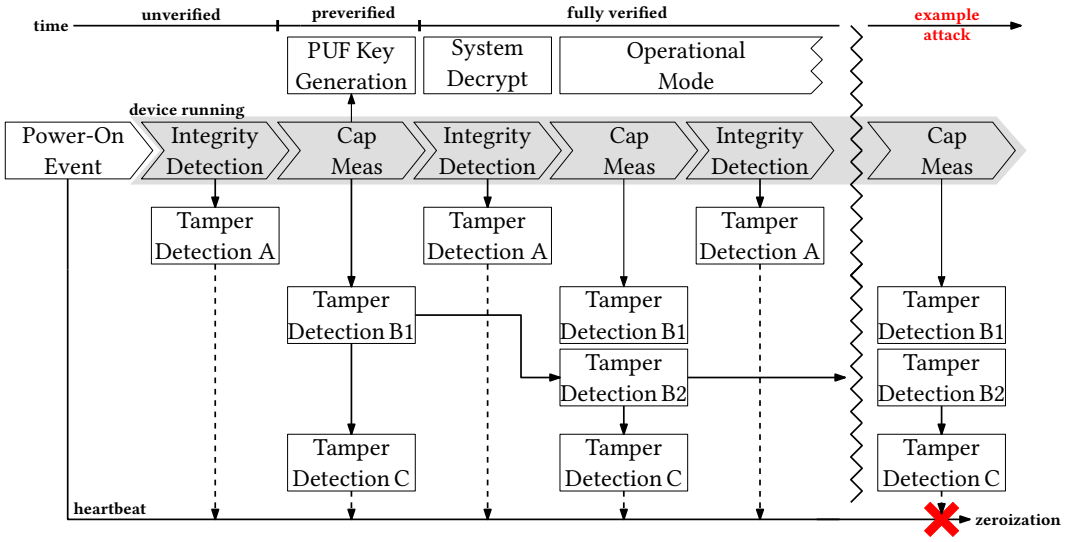


Figure 4.12: Secure boot process of the enclosed system.

used for the PUF key reconstruction. Simultaneously, the *same* differential capacitance values are used to start another TD, termed TD-B1 and TD-B2. TD-B1 limits the valid range of each individual capacitance relative to its reference value, i.e., at $t = T_0$ boundaries for each sensor node are computed *once* based on the reference value $\pm p$, whereas p is a constant guard parameter. For each subsequent measurement, the then current capacitance value is checked against the computed boundaries: $|\gamma(t) - \gamma(T_0)| < p_1$. As additional precaution, TD-B2 limits the discrete rate of change, i.e., by computing $|(\gamma(t) - \gamma(t-1))| < p_2$, for a second security parameter p_2. Both parameters p_1 and p_2 must be tuned to the specific application profile of the device and are strongly related to the width Q_w of the equidistant quantization.

The output of the absolute capacitance measurement serves as input for TD-C. Here, zeroization is caused if any of the absolute capacitance values significantly deviates from the then-current mean of all absolute capacitance nodes. This approach is relatively insensitive to temperature drift in absolute capacitances as supported by our practical results. As later illustrated in Part V, a deviation due to tampering can be assumed if the value is outside a ±15% range of the mean. Please note that tinkering with TD-A, TD-B1, TD-B2, and TD-C cannot be done easily without violating some of their properties.

By successfully generating the PUF key, the proper initialization of the TD-B mechanisms is ensured. Evaluating TD-C complements this approach. This PUF key can then be used to decrypt the firmware of the host or some of its CSPs. In an actual implementation, the PUF key is combined with IC-level roots-of-trust* to form a compound device identifier within the Device Identifier Composition Engine (DICE) framework for the secure boot process of the device. If either during power-up or runtime any of these checks fail, a tamper-event is caused that triggers the zeroization and stops the heartbeat signals. All mechanisms have been designed in an intertwined way to have a layered approach to security; individually disabling them is considered very challenging.

* This could be another tamper-evident PUF at the IC level or keys stored in Secure Non-Volatile Storage (SNVS) in COTS microcontrollers, i.e., the cover basically extends the physical trust domain of the IC to the whole enclosed area.

Firmware level. A custom firmware was developed mainly by Johannes Obermaier for testing the operating concept following the ideas of [150], based on a security-enhanced fork of FreeRTOS that serves as operating system for the measurement setup and PUF data processing chain. Additionally, it implements an Embedded Key Management System (EKMS) which operates similar to the software of a Hardware Security Module (HSM). This system ensures real-time behavior of the measurement process while protecting and operating on sensitive data, i.e., PUF data and derived keys. The host system can request cryptographic operations to be performed on data using a handle to the key material. Thereby, the PUF key material itself is not exposed and never leaves the measurement system. To achieve these goals, FreeRTOS has been extended with a secure syscall interface that allows a userspace task, e.g., the communication interface, to only execute well-defined operations. The Memory Protection Unit (MPU) provides hardened data protection such that an attacker cannot gain access to key material by taking over a single userspace task. However, the described approach does not address the critical issue of how to securely bootstrap the device. This was done in collaboration with Lukas Auer [9], where the PUF key is combined with existing roots-of-trust as part of the Device Identifier Composition Engine (DICE) [204] which provides functionality similar to a Trusted Platform Module (TPM). Please note, the application domain was not specifically the focus of this work. The given example is only intended to point out how such a PUF-based enclosure could be incorporated into a larger system.

4.6 Summary on Higher-Order Alphabet Constructions

We presented a way of how to construct a higher-order alphabet PUF. It naturally arises from the requirement of implementing an enclosure-based PUF, i.e., a tamper-evident system-level structure that can be used as a PUF. This is apparently the first construction to derive symbols from the PUF output as opposed to binary data. Several other practical design goals such as layout randomization align well with existing PUF theory. While other authors have been using similar processing steps prior to the quantization, e.g., normalization, compensation, they only received little attention in widely referenced works. It is evident that the approach presented here is not limited to specific tamper-resistant enclosures but could be applied to other types of PUFs where the underlying raw data can be accessed, e.g., the RO or TERO PUF. Hence, it can be expected that transferring these concepts to exisiting PUF designs may lead to new area improved silicon PUFs, too.

Part III

Reliability Enhancement Techniques for PUFs

Chapter 5

Previous Work on Reliability Enhancement Techniques for PUFs

This chapter provides an overview of reliability enhancement techniques which includes error-correction and error-reduction techniques. Moreover, a model for tamper-evident PUFs is presented that serves as a reference to develop quantization and error-correcting schemes in the following chapters. With respect to these two topics, a more detailed survey of the existing work is presented in Section 5.3 and Section 5.4. The work and ideas included in this chapter are primarily based on the publications in [92, 91, 100, 93] with the thesis author as principal author.

Contents

5.1 Overview: Reliability Enhancement Techniques

In the following, an overview of the various techniques of how to enhance a PUF's reliability is presented. This is based on Figure 5.1. In general, two complementary approaches are common: error-correction by means of *syndrome coding* followed by an ECC [241, 133, 72] and error-reduction either by improving the physical process or algorithmic techniques such as multiple evaluations. While a single technique from any of these domains may be sufficient to ultimately result in a reliable device, it is most likely that from an engineering point of view, it may be more desirable to apply a selection of techniques for reasons of a more efficient (or more secure) implementation. For example, while an overly sophisticated and powerful ECC scheme could be used to correct an incredible number of errors, it may be a better approach to prevent these errors from happening by using error-reduction techniques prior to selecting the parameters of an ECC.

The left part of Figure 5.1 covers the ECC part. Generic and well-known constructions are the *fuzzy commitment* [106, 205], *fuzzy extractor* [37], and *parity construction* [34].

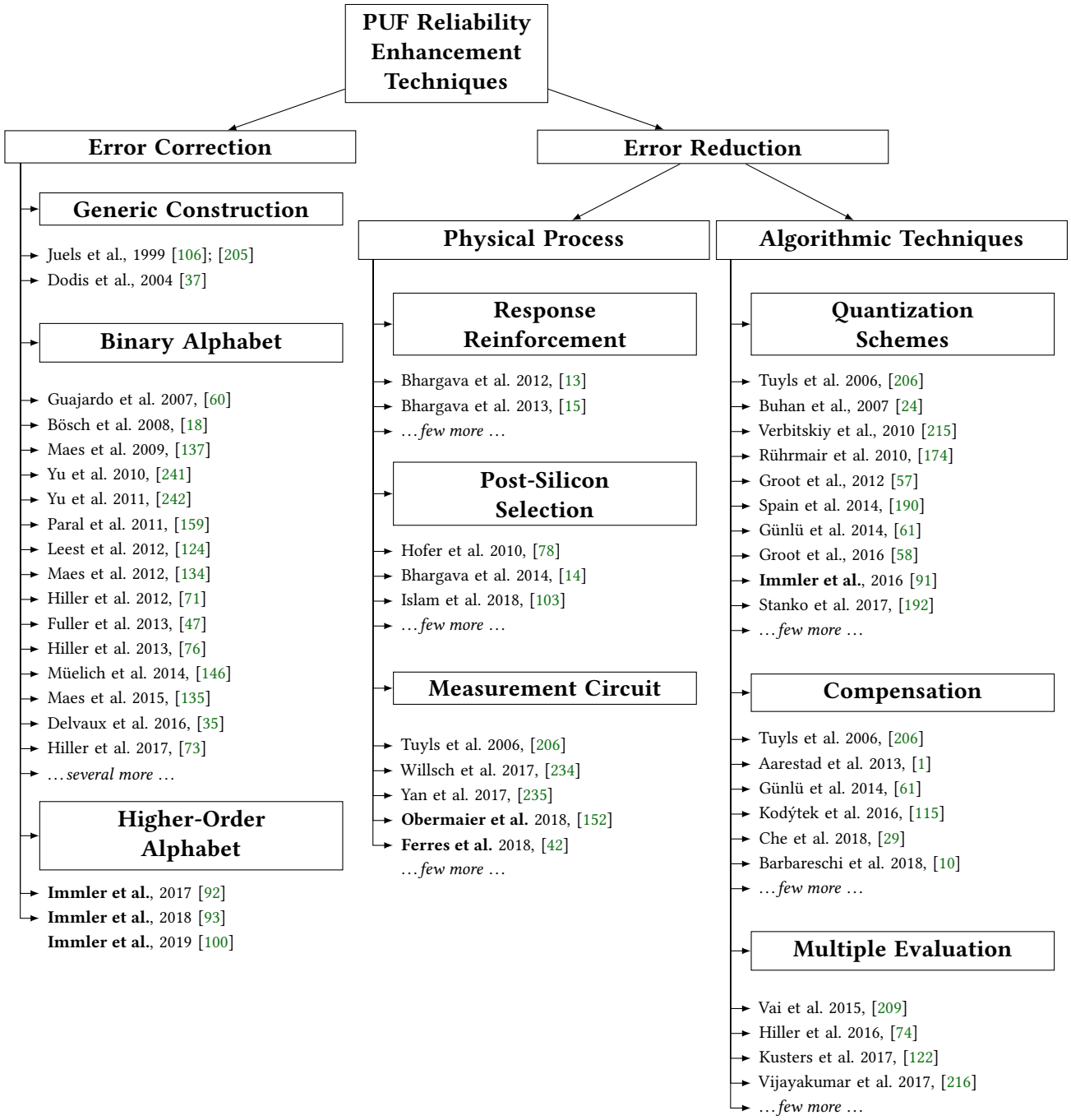


Figure 5.1: Overview of PUF reliability enhancement techniques with selected publications. Bold font is used to indicate contributions by thesis author.

Specific instances of these constructions are typically based on *linear* ECC schemes for the syndrome coding and the ECC itself. This is in contrast to *pointer*-based schemes where the linear dependencies between secret and helper data are removed by only selecting specific PUF response bits (or symbols). However, only partially taking into account the information provided by the PUF response neglects the requirements of tamper-detection which is why pointer-based schemes such as Index-Based Syndrome coding (IBS) [241] by Yu and Devadas, Complementary IBS [71] by Hiller et al., and Maximum Likelihood Symbol Recovery [240] by Yu, Hiller, and Devadas are not considered in this work. They may be a suitable solution for other PUFs that are not tamper-evident.

Most, if not all, practical implementations following the ideas of either a fuzzy extractor, fuzzy commitment, or pointer schemes were thus far based on a binary PUF alphabet, i.e., the PUF responses comprise *bits* that are typically assumed to be i.i.d. (independent and identically distributed) across the PUF responses [60, 18, 137, 241, 242, 159, 124, 134, 71, 47, 76, 146, 135, 35, 73]. In contrast, the thesis author has been working on *symbol*-based ECCs that meet the specific requirements of tamper-sensitivity, as explained in Chapter 7 and Chapter 8. Moreover, even when these symbols are encoded by a fixed-length bit sequence, these individual bits are no longer i.i.d.

The right part of Figure 5.1 covers the error-reduction domain, i.e., instead of correcting errors, they attempt to prevent errors from happening. This class of reliability enhancement techniques is further divided into the physical process and algorithmic part. Improving the physical process during the manufacturing process can already rule out the necessity of using ECCs, e.g., by using Response Reinforcement (RR) [13, 15] or Post-Silicon Selection (PSS) [78, 14, 103]. Clearly, these techniques sometimes cannot be used, as control over the manufacturing process is not always possible, i.e., manufacturers tend to avoid non-standard manufacturing processes to ensure consistency and cost-efficiency. Another option to improve the physical process is to optimize the measurement circuit, e.g., improving the Signal-to-Noise Ratio (SNR) or adding certain circuit-level compensation techniques to mitigate environmental drift effects [206, 234, 235, 152].

Another line of work towards error-reduction are algorithmic techniques, including but not limited to quantization schemes, algorithmic compensation, and different approaches for multiple evaluation. In general, quantization schemes aim at reducing the bit complexity of the considered values, e.g., n bit values are mapped to k bit values with $k < n$. Most attempts in this domain are based on scalar quantizers, i.e., values reside in a single dimension. This is in contrast to vector quantization where multi-dimensional values are considered which is often the case for biometric systems and some optical PUFs. Here, the quantization attempts to reduce dimensionality in addition to bit complexity. For the PUFs considered in this work, the input to the quantizer is conceptually viewed as a PDF, i.e., a (quasi-)continuous distribution, and the goal is to extract as many bits from it as possible, while ensuring sufficient reliability and tamper-sensitivity, as explained in Chapter 6. To obtain a decent size of the input alphabet to the subsequent ECC from an engineering point of view, quantization is an indispensable processing step. Works covering the concept of quantization in the domain of PUFs include but are not limited to [206, 24, 215, 174, 57, 190, 61, 58, 91, 192]. As detailed in Section 5.3, there are primarily two orthogonal approaches to quantization, namely equiprobable and equidistant quantization. The work of this thesis is solely based on equidistant quantization, as it outputs symbols of a higher-order alphabet and thereby provides a fundamentally different approach of how to construct and evaluate a PUF. Equidistant quantization is of particular importance to guarantee tamper-sensitivity in the quantized values.

Other algorithmic techniques to error reduction are geared towards compensation of environmental drift effects. Considering the PUF response as a signal, these attempts typically aim at removing the signal's DC-offset while preserving the AC component representing the PUF. Simple approaches to achieve this are by removing the mean of a group of values [10], while more sophisticated approaches are based on applying the Discrete Cosine Transform (DCT) which is then followed by selecting DCT-coefficients carrying the most information [61]. Another class of compensation is based on a linear transform to scale the values with respect to a measured or stored reference, as done in [206] or [29, 1]. In all cases, compensation must target the type of error induced by the environmental drift, i.e., additive or multiplicative errors must be reduced by selecting appropriate processing steps. Well-made solutions such as the HELP PUF [29, 1] even no longer require a dedicated ECC after compensation and quantization of its values.

An additional way to optimize reliability of the PUF response is to consider Multiple Evaluations (MEs). This has been done for example in [209, 74, 122, 216]. One of the most straightforward options to do this is based on oversampling while more sophisticated techniques incorporate the obtained information from oversampling in the subsequent ECC. Most of the time, the obtained reliability is in direct relation to the additional time spent for performing the MEs.

Depending on the targeted type of PUF and permissible iterations of hardware engineering, the combination of several of these techniques appears as the most promising approach to implement the most efficient and secure PUF. In the following, a simplified model for tamper-evident PUFs is presented to further study specifics of error-reducing and error-correcting processing steps.

5.2 Model for Tamper-Evident PUFs

To describe the system model that is relevant for all subsequent chapters, we first briefly introduce the notation used in the following chapters. Afterwards, the model itself is described in Section 5.2.2. Since the goal is to study this model as part of different key derivation techniques, corresponding safety and security aspects are described in Section 5.2.3.

5.2.1 Notation

In the following, unless specifically noted otherwise, random *variables* and their distributions are represented by capital italic letters, whereas numbers and specific realizations of random variables are denoted as small italic letters. Subscripts refer to indices of vectors, and right superscripts show the length of vectors (in either symbols or bit). Constants and operators are always in upright font, e.g., left superscripts differentiate subtypes of an otherwise shared variable letter. C is the ECC and c stands for an n-bit codeword with k information bits and p parity bits (or symbols).

Throughout this part of the thesis, we make use of several distance metrics, namely: $\mathrm{d_E}$ for Euclidean distance, $\mathrm{d_{Lev}}$ for Levenshtein distance, $\mathrm{d_{Lee}}$ for Lee distance, $\mathrm{d_{Man}}$ for Manhattan distance, $\mathrm{d_{H|2}}$ for Hamming distance applied to bit strings, and $\mathrm{d_{H|S}}$ for Hamming distance applied to strings with symbols of a higher-order alphabet.

5.2.2 PUF System Model

The PUF system model that is relevant to this work is illustrated in Figure 5.2 and represents the practical work of, e.g., [206, 95, 97] in sufficient detail to discuss their PUF key derivation specifics that are of general nature and relevant for future proposals of tamper-evident PUFs, too. From left to right, it comprises the tamper-evident PUF and illustrates all necessary steps to generate a key. The upper part represents the enrollment of the PUF, i.e., the point in time when the PUF is initialized in a secure environment and helper data is created to enable later error correction. The lower part depicts the reconstruction in the field where the PUF key is extracted again to serve as secret input for cryptographic applications. Both branches of the figure merge at the very right. This is the determination whether the designated values Z^v match those from the reconstruction $\hat{Z}^v$. Should this comparison succeed, then the device can use the derived values to generate a key with an additional privacy amplification step. However, should the comparison fail, then this is the result of either insufficient reliability or a physical attack.

Each single PUF value denoted as X is drawn from its corresponding physical PUF *node*. In both [206] and [95], the node from which X is drawn is a capacitor C that is subject to manufacturing variation, i.e., X_1 corresponds to a capacitor C_1, X_2 to C_2, and so on. We specifically refer to this as PUF *node* as opposed to bits, to point out that symbols comprised of multiple bits per node are extracted. This underlying element of a PUF is sometimes also called a PUF primitive and this model is not limited to a specific type of node/primitive. X follows a quasi-continuous PDF as illustrated in Figure 5.3 and is the digital representation of the capacitance obtained by a compensated* measurement and subsequent conversion by an ADC. These compensating techniques, such as [206, 10] depend on the specifics of the PUF and are considered outside the scope of this work. Here, we use the term quasi-continuous since in the actual application we do not know the real value (in the sense of continuous) of the PUF nodes and can only practically measure it using a high-resolution measurement circuit. Therefore, X would be typically represented by an *integer* with its number of bits in binary representation equivalent to the number of bits of the ADC. In total, there are v nodes (i.e., v distinct capacitors) in the PUF and all their values combined are termed PUF *device* and written as X^v, i.e., $X^v = \{X_1, X_2, ..., X_v\}$ with $X \in \mathbb{Z}$.

As part of the data acquisition, the PUF values X are always affected by remaining circuit noise $N \in \mathcal{N}(0, \sigma_N)$ during reconstruction which makes it necessary to account for this influence by suitable mechanisms, e.g., a combination of quantization scheme and ECC. Noise is assumed to be Gaussian following $\mathcal{N}(0, \sigma_N)$, i.e., it is mean free. Moreover, the noise standard deviation σ_N is considered equally distributed across all PUF nodes. If the system has *not* been tampered with, then the noisy PUF response is $\hat{X}^v = X^v + N^v$. This noise modeling is equivalent to [206] and also relevant for other systems, such as [243].

Now, in the event of tampering with the PUF, the physical PUF nodes from which values are drawn are additionally altered. This effect is denoted as ${}^{A}W^v \in \mathbb{Z}$, i.e., $\hat{X}^v = X^v + N^v + {}^{A}W^v$ as indicated in Figure 5.2. We note that ${}^{A}W$ does not follow a stochastic model or is otherwise formally constrained. This is owed to the fact that a designer of a tamper-evident PUF will not know (*i*) which nodes will be affected by tampering, or how many (*ii*) what the resulting magnitude of the attack is. Hence, regarding the magnitude of ${}^{A}W$, we need to implicitly assume that $\sigma_N < {}^{A}W$ which is supported by the practical

* The term *compensated* measurement refers to circuit-level techniques to remove temperature and voltage drift effects. An exemplary compensated technique is the 3-signal approach mentioned in [206]. For other PUF designs, such as the RO-PUF, similar concepts were presented in [10].

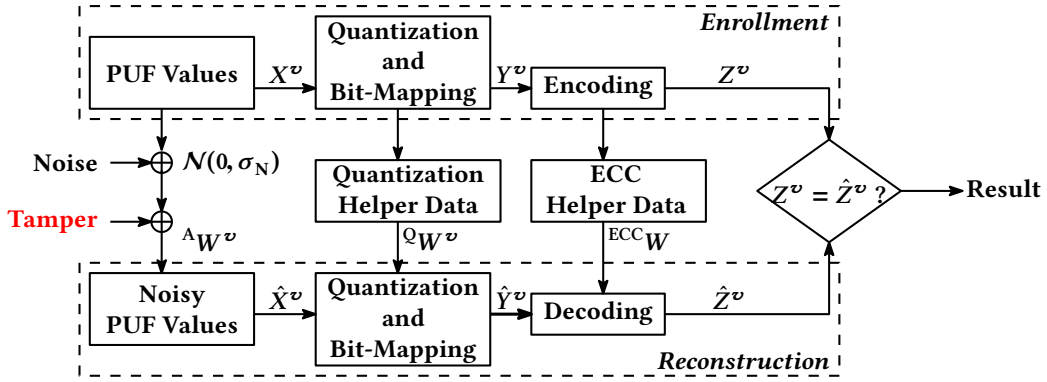


Figure 5.2: PUF system model with enrollment and reconstruction. Y is the quantized PUF response and Z the secret bit sequence. Added noise is denoted as ($\hat{\cdot}$).

attacks in [206, 95, 97]. Since the smallest physical quantity in the system is one node, the best approach approach will enable tamper detection even in cases when only one resulting symbol is tampered with. $^A W$ is often referred to as shift, and in the noiseless but tampered case, the Euclidean distance $d_E(X, \hat{X})$ is termed the *tamper magnitude*.

Based on this noise model, it is evident that instances in time may occur where $N = 0$ for a specific $\hat{X}$ and at the same time $^A W \approx \sigma_N$, i.e., tampering would go undetected as its magnitude would essentially be mistaken as noise only. Since the noiseless scenario allows for the maximum tamper magnitude to possibly go undetected, this is the scenario we later choose for analysis purposes. In all other cases, practically speaking, it is similarly difficult to distinguish the noise from the effects of tampering, as an unexpectedly large magnitude may either be the result of a relatively unlikely noise event, or the result of tampering. Hence, the challenge is to devise a scheme that provides a clear Tamper Detection Threshold (TDT) of whether the error magnitude should be treated as noise, or as tampering, while not impeding typical PUF reliability requirements. This is achieved by schemes where TDT $= u \cdot \sigma_N$, with u being as small as possible.

5.2.3 Safety and Security Aspects of Key Derivation

The implementation of a PUF is characterized by several aspects that ensure basic properties of the PUF-based key generation. This includes but is not limited to the cryptographic quality of the derived key (security), its reliability (safety), and tamper-sensitivity (security), as illustrated in Figure 1.4. In particular, the comparison of $Z^v = \hat{Z}^v$ within the PUF system model, as illustrated in Figure 5.2, is essential for a PUF-based device which allows it to behave in the expected manner for the intended purpose.

Successful tamper-detection is the result of sufficient tamper-sensitivity and is the self-determination by a device that $Z^v \neq \hat{Z}^v$ and in that sense no different to the case when the device fails because of insufficient reliability. The interesting result of this work is that ECC schemes effectively working under $\hat{X}^v = X^v + N^v$ are *not* automatically the same to effectively detect the effects of $^A W$, i.e., despite providing more entropy, their TDT is sometimes worse compared to schemes providing less entropy but a better TDT, as later practically demonstrated in Chapter 9. As additional constraints, we aim at schemes with superior detection of $^A W$ while ensuring the following two requirements regarding the reliability and cryptographic quality of the key:

- The reliability or device failure rate, written as the mismatch probability $P_e(Z^v) = \Pr[Z^v \neq \hat{Z}^v]$ shall be $< 10^{-6}$ in the presence of noise (without tampering).
- The effective number of secret bits that are extracted from the tamper-evident PUF should be sufficiently large, e.g., $\tilde{H}_\infty(Y^v|W) > 128$ bits (preferably more). Hence, the loss in entropy caused by information leakage via the helper data must be considered.

The information leakage is measured by the mutual information between quantized PUF response and helper data, i.e., $I(Y^v; W)$. The min-entropy definition for $\tilde{H}_\infty(Y^v|W)$ is given in [37]:

$$I(Y^v; W) = H(Y^v) - H(Y^v|W) \leq v \cdot \log_2(q) - \tilde{H}_\infty(Y^v|W), \tag{5.1}$$

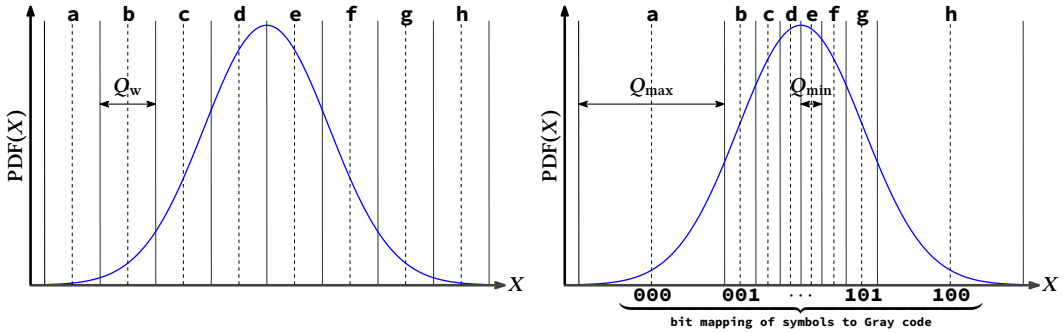
$$\tilde{H}_\infty(Y^v|W) = -\log_2\left(\underset{w}{E}\left[\max_{y^v} \Pr_{Y^v|W}[y^v|w]\right]\right). \tag{5.2}$$

Please note, in these equations W is instantiated in a generic manner, independent from the fact that it could be quantization helper data ${}^{Q}W$ and/or ECC helper data ${}^{ECC}W$, as seen in Figure 5.2.

5.3 Quantization Schemes and Bit Mappings

Thus far, there are two predominant schemes to quantize normally distributed PUF data. Both schemes are based on subdividing the quasi-continuous PDF based on the distribution of X into intervals. In case of equiprobable quantization [206], the intervals are chosen such that the intervals occur with equal probability. In contrast, equidistant quantization [91] divides it into intervals of equal width. In order to decrease the probability of an erroneous quantization value Y, an offset is stored as helper data ${}^{Q}W$ during enrollment that shifts the PUF value X to the center of its corresponding quantization interval. For reasons of clarity of explanations, we always assume that symbols of a Higher-Order Alphabet (HOA) are assigned to these intervals as a first processing step even though this was not necessarily included in the original publication, i.e., the PUF output alphabet $\mathcal{L}$ is not $\mathcal{L} = \{0, 1\}$ but $\mathcal{L} = \{a, b, c, d, \ldots\}$, whereas $|\mathcal{L}|$ is the size of the alphabet which is equivalent to the number of quantization intervals L. Hence, this is referred to as HOA PUF. Both quantization approaches and the assignment of symbols are sketched in Figure 5.3.

Equiprobable quantization of PUF data was first introduced in [206] for the tamper-evident Coating PUF and later used for example in [61, 231], too. As proposed in [206], each interval is assigned a multi-bit binary representation by means of a Gray code, i.e., neighboring intervals are designed such that their binary representation differs by a one bit substitution error only. Hence, the Hamming distance in binary denoted as $d_{H|2}$ is 1 for directly neighboring intervals. Please note that for this approach, both symbols *and* their corresponding binary bit mapping are i.i.d. *and* uniformly distributed. The processed output prior to the ECC is therefore a binary alphabet and in that sense highly similar to, e.g., the output of an SRAM-PUF. However, for the specific scheme presented in [206], it was later shown that the length of each individual helper data offsets ${}^{Q}W$ stored for the quantization during enrollment leaks significant amounts of information on the PUF key [91]. In addition, ensuring uniformity of bits requires precise knowledge of the underlying PUF distribution and therefore limits the practical relevance of this scheme.



(a) Equidistant quantization.

(b) Equiprobable quantization.

Figure 5.3: Visualization of equiprobable and equidistant quantization schemes processing PDF(X) which follows $\mathcal{N}(\mu_X, \sigma_X)$ based on the parameters given in [206].

Other equiprobable quantization schemes implement a partitioning scheme to avoid helper data leakage but again require precise knowledge of the distribution [215, 192]. Furthermore, as pointed out in [91] and detailed later as part of this work, equiprobable quantization is ineffective to ensure good tamper-sensitivity in all scenarios due to the size of the outermost intervals of width $Q_{\max}$.

Equidistant quantization apparently mitigates these effects due to the evenly sized intervals with only minor leakage from the sign of its helper data ${}^{Q}W$. Moreover, a suboptimal assignment of the interval boundaries relative to the PDF only has an insignificant impact on the resulting entropy of the quantized output. However, it comes at the downside of a biased quantized PUF output, i.e., when mapping the symbols to bits, it is evident that the individual positions of the resulting bit string are neither i.i.d. nor uniform. As a result, any fixed-length binary bit mapping of the symbols is heavily biased. Correspondingly, when combining equidistant quantization with a fixed-length binary output and a linear fuzzy extractor scheme, significant amounts of secret information would be leaked by the helper data due to the induced bias [89, 35].

To overcome some of the limitations of equidistant quantization, the thesis author proposed a variable-length bit mapping of the symbols [92], as explained in Chapter 7. Hence, as a kind of debiasing step, this follows the information theoretic intuition of assigning shorter binary representations to intervals that occur more often, while assigning longer bit representations to intervals that occur less often. However, the quantized sequence comprised of the values Y is no longer of fixed length which necessitates Varshamov-Tenengolts (VT) codes operating in Levenshtein distance $\mathrm{d}_{\mathrm{Lev}}$, accounting not only for substitution errors but also insertions and deletions [201, 213]. This is due to the fact that more commonly known codes such as Bose-Chaudhuri-Hocquenghem (BCH) and Reed Solomon (RS) codes are not designed to work on variable-length inputs. While VT-codes are well-suited to operate in Levenshtein metric, their overall capability in terms of error-correction is still quite limited. The specific values of each quantization interval are chosen such that neighboring intervals differ by $\mathrm{d}_{\mathrm{Lev}} = 1$ in [92], i.e., the bit mapping of symbols to binary is similar to a Gray code such that directly neighboring intervals differ by only one substitution or insertion/deletion error. Again, a rather precise knowledge and symmetry of the PDF is required to ensure proper behavior of this scheme.

Unfortunately, as later demonstrated as part of the evaluation in Chapter 9, the scheme based on equidistant quantization and VT-codes falls short when it comes to tamper-

sensitivity when compared to a scenario only based on equidistant quantization without ECC. To overcome the limitations of this new scheme and previous approaches, the aspect of tamper-sensitivity is formalized to tailor a scheme specifically for tamper-evident PUFs. The resulting scheme is presented in Chapter 8 and based on an equidistant quantization, too. It represents a better alternative when compared to the approach in Chapter 7. For the solution presented in Chapter 8 based on Limited Magnitude Codes (LMCs), the subsequent ECC is based on the quantized PUF output Y which is based on symbols with aforementioned properties of an equidistant quantization. Please note that the overall setting in this work deviates quite significantly from scenarios commonly assumed, e.g., for the SRAM PUF.

5.4 Error-Correcting Codes for PUFs

A significant amount of work was carried out in the domain of PUFs ranging from formalizing PUFs [7] to generic ECCs constructions, and protocols [32] in addition to analyses in terms of implementation and information efficiency [133, 77, 35]. As indicated beforehand, previous work is mostly specifically tailored towards PUFs based on a binary alphabet with only very few exceptions covered by the thesis author [91, 92]. The strong focus on these binary-only PUFs has been a valid requirement due to their ease of physical construction in silicon and widespread availability. While generally being suitable to provide a sufficient reliability even for other scenarios than their intended purpose, the shortcoming of most ECC schemes is related to helper data leakage in ${}^{\text{ECC}}W$ that is caused by biased PUF data and/or insufficiencies of the ECC construction, as detailed in [89, 75, 35]. If not considered at all, helper data leakage is a severe security threat, as the anticipated security level is not present in the design. If not systematically counteracted on an algorithmic level, helper data leakage impacts the cost/size of the PUF implementation, as demonstrated for example in [73], where – depending on the chosen ECC construction – the corresponding PUF size would differ by a factor of ~ 2 to achieve the same security level. Hence, the problem of bias in PUF data and ECC helper data leakage is not completely new and the same is true for ideas of counteracting it. Therefore, when considering new ECC approaches for tamper-evident PUFs and higher-order alphabets, these known effects and existing concepts must be taken sufficiently into account as done in the following.

To remove PUF induced leakage, various debiasing schemes were proposed. Index-Based Syndrome coding (IBS) [241] is a pointer-based debiasing technique that also improves the reliability by indexing only reliable PUF response bits. However, the symbols of an equidistant quantization as later used in our scheme all have the same reliability such that IBS is not applicable to the discussed scenario. Moreover, not considering certain bits of the PUF output counteracts the idea of detecting tamper attempts.

The scheme presented in [135] improves the von Neumann (VN) corrector [217]. For i.i.d. PUF response bits (which is different to the considered scenario), pairs of consecutive zeros or ones occur with different probabilities, while pairs (1,0) and (0,1) have the same probability. However, the approach is intended for PUFs with small output alphabets. It evaluates groups of elements that occur with the same probability but differ in their sequence, such that an increasing number of elements decreases the probability of these equiprobable events. In [195], it was extended to ternary outputs using reliability information. However, it cannot be efficiently applied to higher-order alphabets. The multi-bit symbol approach in [240] is especially suited for PUFs with high bit error probabilities $> 20\%$. It is not explicitly designed for bias reduction but can also handle biased inputs efficiently as well.

Additional recent debiasing work includes [73] where again the PUF bits are assumed i.i.d. and coset coding is applied to mitigate the leakage. This idea could be interpreted as combining different equidistant quantization intervals to create a more uniform occurrence of the symbols. However, this again would contradict the idea of tamper-sensitivity as will become evident by the remainder of this work.

As a result, none of the discussed techniques provide a promising foundation to efficiently derive keys from PUFs with biased symbols of a higher-order alphabet which has motivated the development of the solutions presented in Chapter 7 and Chapter 8. To the best of the author's knowledge, the case of Levenshtein or Lee metric as distance metric for PUFs has never been considered beforehand. Please note, the thesis author is are aware of the threat of helper data manipulation attacks [36]. However, for the presented work, only fundamental properties of quantization and ECC schemes are discussed. In addition to that, it is assumed that access to the helper data is also obstructed by the tamper-evident PUF, i.e., attempts to change the helper data would cause the partial destruction of the PUF as any other physical access to the underlying system. An additional privacy amplification step for the resulting output is always advised but considered out of scope.

Chapter 6

Error-Reduction by Quantization

A well-chosen quantization scheme is a necessity for the designated system due to several reasons. First of all, it helps to significantly reduce the errors caused by noise, e.g., assuming a Gaussian noise source as done in the previously shown PUF model. Moreover, the post-quantization error-rate can be tuned according to the application specifics and the subsequent ECC scheme. In addition to that, the unprocessed measurement output is typically a high-resolution integer that is not well-suited for direct processing by an ECC. Hence, a quantization scheme helps translating from a high-resolution integer to a smaller set of finite symbols that can be processed more efficiently. This can be typically achieved with a relatively low effort from an engineering point of view. However, with an increasing number of PUF output symbols to consider, the higher is the probability for a single node to be in error, thereby causing the device to fail. Hence, this diminishes the error-reducing effects of a quantization scheme per symbol which is why it must still be combined with an ECC for better performance. Two predominant quantization schemes were proposed for PUFs based on equidistant and equiprobable intervals and they are analyzed as part of this chapter. This chapter is based on joint work published in [91] with the thesis author as principal author.

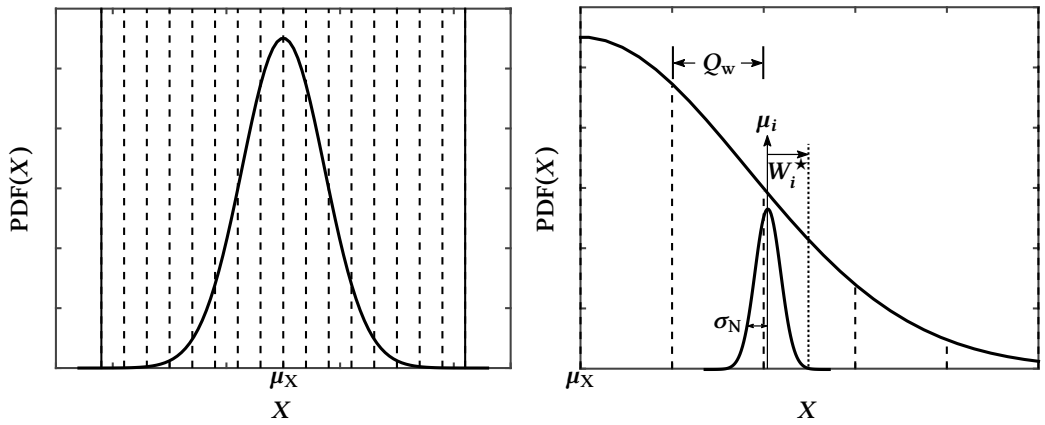
Contents

6.1 Introduction to Quantization

Based on the PUF system model in Figure 5.2, we study different approaches at the stage of quantization, namely equidistant quantization in Section 6.2 and equiprobable quantization in Section 6.3. This also follows the notation of Section 5.2.

6.2 Equidistant Quantization

Let ρ be the PDF of the values over all physical nodes as introduced above. It is now subject to an equidistant quantization, i.e., a quantization that uses evenly-spaced intervals of the same width as shown in Figure 6.1.



(a) Equidistant quantization. **(b)** Annotated close-up of quantization.

Figure 6.1: Exemplary equidistant quantization.

Based on an empirically determined noise-level σ_N of the physical measurement process, the interval width is chosen as $Q_w = 2 \cdot y \cdot \sigma_N$. The choice of y depends on the required reliability and thereby determines the number of L intervals that can be used (see below).

Enrollment: The domain of ρ (its X-axis) is divided into L intervals of the form

$$]l \cdot Q_w, (l+1) \cdot Q_w], \quad l = 0, \ldots, L-1 \tag{6.1}$$

During enrollment, each measured node X_i for $i = 1, \ldots, v$ is assigned to one of these intervals by computing

$$Y_i = \lfloor |X_i / Q_w - 0.5| \rfloor \tag{6.2}$$

with $Y_i \in \{0, 1, \ldots, L-1\}$. This is considered the quantized *symbol*, as illustrated in Figure 5.3a. In practice, the value X_i is measured multiple times and averaged to obtain μ_i as shown in Figure 6.1b, i.e., the expected value for the physical node i without noise. Since working on a conceptual level, we suppose that $X_i = \mu_i$. While this approach helps to determine the interval as part of the enrollment, it is impractical to use it in the field, as measuring sufficiently often may not be possible under given time constraints. Therefore, helper data is required to account for the noise during the quantization as part of the reconstruction. This helper data ${}^{Q}W$ is computed as

$${}^{Q}W_i = X_i - (Y_i + 0.5) \cdot Q_w \tag{6.3}$$

and represents the offset between X_i and the center of the quantization interval the current value resides in. This is additionally illustrated in Figure 6.1b.

Reconstruction: In the field, the device reconstructs the values from noisy measurements denoted as $\hat{X}_i$ $(i = 1, \ldots, v)$ with

$$\hat{Y}_i = \lfloor\, |(\hat{X}_i - {}^{\mathrm{Q}}W_i)/Q_{\mathrm{w}} - 0.5| \,\rfloor \tag{6.4}$$

Here, ${}^{\mathrm{Q}}W_i$ is used to shift noisy values $\hat{X}_i$ towards the center of those intervals used during enrollment.

Reliability: By carrying out the previous operations, one aligns the center μ_i of the PDF(X_i) (with standard deviation σ_{N}) of the noisy measurements of a single measurement location with the center of a quantization interval. To determine the symmetric confidence interval CI = $[-y\sigma_{\mathrm{N}}, y\sigma_{\mathrm{N}}]$, i.e., the percentage of measurements that will be successfully assigned to the correct quantization interval, one can refer to commonly available tables for this purpose or use the erf$(\cdot)$ function. As an example, selecting $y \approx 3.9$ causes 99.99% of the values to be within the CI of a single node. Please note that for computing the device failure rate, one must consider the unreliability of all nodes.

Key quality: Considering the amount of information H(Y) that is extracted by this method, no general statement can be made, as it is dependent on ρ and L. However, it is evident that by increasing the number of intervals, the extracted information converges towards the differential entropy of the underlying PDF.

While at this processing stage, no equiprobability of the obtained bits can be achieved (due to the chosen quantization method), it is clear that several functions exist to compress the entropy into a smaller bitstring and thereby achieve uniform entropy, e.g., by using one of the approved conditioning functions of NIST 800-90b. This is typically part of the privacy amplification step.

Tamper-Sensitivity at Quantization Level: Considering a physical attack, one must analyze its effect on the quantization. Hence, the reconstructed value $\hat{Y}_i$ no longer is the result of Equation 6.4 but

$$\hat{Y}_i = \lfloor\, |(\hat{X}_i - {}^{\mathrm{Q}}W_i + {}^{\mathrm{A}}W_i)/Q_{\mathrm{w}} - 0.5| \,\rfloor \tag{6.5}$$

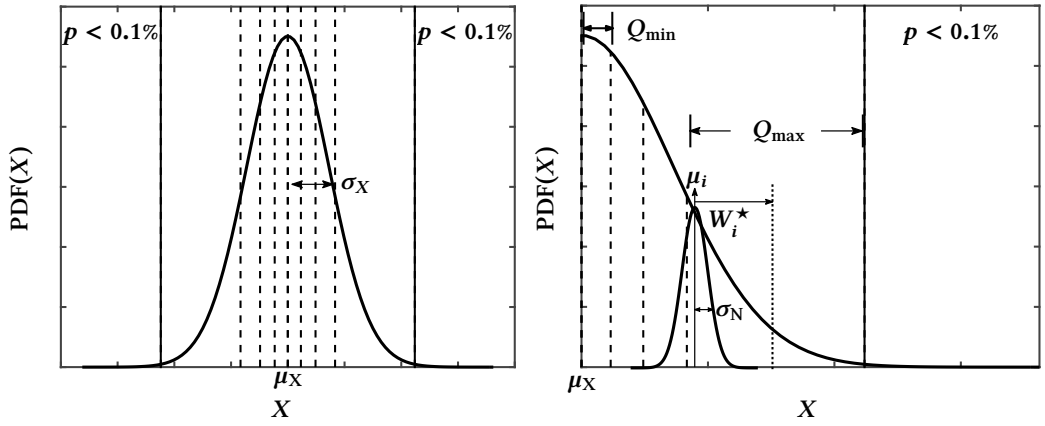
with ${}^{\mathrm{A}}W_i$ being the shift induced by the attacker which is either in positive or negative direction. Since all intervals are of the same width, the maximum shift possible (not considering the noise) is $Q_{\mathrm{w}}/2$. Thus, any ${}^{\mathrm{A}}W_i$ exceeding this value causes $\hat{Y}_i \neq Y_i$ and will therefore be detected.

Considering Information Leakage I(Y^{υ}, ${}^{\mathrm{Q}}W^{\upsilon}$)*:* As stated in Section 5.2.3, the information leakage caused by the helper data ${}^{\mathrm{Q}}W^{\upsilon}$ must be studied. Here, ${}^{\mathrm{Q}}W^{\upsilon}$ does not cause a significant information leakage by the length of the offset since each interval is equidistant and any value ${}^{\mathrm{Q}}W^{\upsilon}$ could occur in any of the intervals. Only the sign of the offset and the probability gradient creates minor leakage. Hence, an attacker attempting to exploit the leakage in ${}^{\mathrm{Q}}W^{\upsilon}$ to help determine the value of X_i or Y_i does not gain a significant advantage by accessing the helper data.

Limitations of this approach: Since there will be a bias in the quantized data, it is no longer advised to use one of the commonly available ECCs as the bias would create a severe helper data leaking with regard to ${}^{\mathrm{ECC}}W$ of a subsequent ECC, as demonstrated in Chapter 9.

6.3 Equiprobable Quantization

We briefly recapitulate the necessary equations of the Coating PUF [206] since they are subject to further analysis. This is also intended to point out a possible naming mismatch in the referenced paper concerning the variables: W_i and w (as given in the original paper).



(a) Equiprobable quantization. **(b)** Annotated close-up of quantization.

Figure 6.2: Exemplary equiprobable quantization.

The equiprobable quantizer defines equiprobable intervals on ρ which can be considered as a histogram equalization. This can be done by considering the respective CDF $\mathrm{q}(\cdot)$ of ρ denoted as

$$\mathrm{q}(X) = \int_0^X \rho(x)\,\mathrm{d}x \tag{6.6}$$

The number L of equiprobable intervals with boundaries at t_j, $j = 0, \ldots, L$ are computed by $t_j = q^{-1}(j/L)$, whereas q^{-1} is the inverse function of q. It follows by the definition of the normal distribution that these intervals are different in size to achieve equiprobability. Let $Q_{\min}$ be the width of the smallest interval and $Q_{\max}$ be the width of the largest interval. We suppose that L is even, then $Q_{\min}$ is the width of the two intervals closest to μ_x (innermost intervals). The two intervals with maximum distance to μ_x are those with the width $Q_{\max}$ (outermost intervals). This is also depicted in Figure 6.2a and Figure 6.2b respectively. The size of the smallest interval may be chosen as $Q_{\min} = 2 \cdot y \cdot \sigma_{\mathrm{N}}$ with the same considerations as beforehand. This determines the size of $Q_{\max}$ based on ρ and L.

Enrollment: For the enrollment, the quantized value $Y_i \in 0, \ldots, L-1$ is determined based on the measured values X_i for $i = 1, \ldots, v$. Again, ${}^{\mathrm{Q}}W^{v}$ is computed.

$$Y_i = \lfloor L\,\mathrm{q}(X_i) \rfloor \quad , \quad {}^{\mathrm{Q}}W_i = Y_i + 1/2 - L\,\mathrm{q}(X_i) \tag{6.7}$$

Reconstruction: In the field, the device reconstructs the data from noisy measurements denoted as $\hat{X}_i, i = 1, \ldots, v$.

$$\hat{Y}_i = \left\lfloor L\,\mathrm{q}(\hat{X}_i) + {}^{\mathrm{Q}}W_i \right\rfloor \tag{6.8}$$

Key quality: Concerning the entropy, it can be seen that as long as a suitable set of equiprobable intervals can be defined, the Shannon entropy is $\mathrm{H}(Y) = \log_2(L)$. Moreover, the quantization already results in equiprobable bits.

Tamper-Sensitivity at Quantization Level: As before, the reconstructed value $hat Y_i$ no longer is Equation 6.8 but

$$\hat{Y}_i = \left\lfloor L\,\mathrm{q}(\hat{X}_i) + {}^{\mathrm{Q}}W_i + {}^{\mathrm{A}}W_i \right\rfloor \tag{6.9}$$

For the magnitude of ${}^{\mathrm{A}}W^{v}$ we must distinguish the following cases (at first, only considering a single measurement without noise and ignoring a subsequent error correction):

- Attack case 1: ${}^{\mathrm{A}}W < Q_{\min}/2$: Attack goes undetected.
- Attack case 2: $Q_{\min}/2 \leq {}^{\mathrm{A}}W \leq Q_{\max}/2$: Attack may be detected depending on which value is attacked.
- Attack case 3: ${}^{\mathrm{A}}W > Q_{\max}/2$: Attack is detectable.

Since the intervals have been adjusted to occur with same probability, it is evident that attacks on any of the quantization intervals also occur with equal likelihood. This leads to the undesired situation when larger quantization intervals are equally likely attacked as smaller quantization intervals.

Considering Information Leakage $\mathrm{I}(Y^{v}, {}^{\mathrm{Q}}W^{v})$*:* As designed does ${}^{\mathrm{Q}}W^{v}$ express the offset to the middle of the interval the current value resides in. However, since no limitation in the range of ${}^{\mathrm{Q}}W^{v}$ is given, one can conclude a value of ${}^{\mathrm{Q}}W$ may exceed $Q_{\min}/2$. As a consequence, it is certain that a measured value which is to be shifted by any value ${}^{\mathrm{Q}}W > Q_{\min}/2$ has not been quantized to the innermost interval as part of the enrollment.

Depending on the actual distribution of ρ and values of ${}^{\mathrm{Q}}W^{v}$ this may result in a situation where some of the measured values can only reside in the outermost interval. We therefore consider the statement of the Coating PUF [206] authors that no information leakage of in equiprobable quantization scheme still to be valid but to reflect the properties of ${}^{\mathrm{ECC}}W^{v}$ instead of ${}^{\mathrm{Q}}W^{v}$ as claimed (assuming it is based on a fuzzy commitment).

Combining both weaknesses: By considering the practical case of $L = 8$ intervals in total [206], then with a chance of $1/4$ an attack will occur in one of the two largest intervals. Since ρ follows a normal distribution, it is easy to see that whenever the left- or rightmost interval is hit, the probability for influencing a value close to the border of the inner next quantization interval is the highest. As a result, the fact that values with larger ${}^{\mathrm{Q}}W^{v}$ are more likely to be attacked can be used – after carrying out the attack and extracting the helper data – to ascertain that the previous value was indeed quantized to the largest interval.

Additional thoughts on tamper-sensitivity: By directly applying the equiprobable quantization as proposed one does not (mathematically) limit the range of the outer intervals. Instead, the limits of the outer intervals are constrained by the measurement range of the circuit. This enables an attacker to always shift a value from within such an interval towards the limit of the measurement range, thereby causing no change in the quantized value itself. This is supported by considering Figure 6.2b without the solid interval lines. One should therefore restrict the valid range of values used for the key generation by limiting the range of the outer intervals (as shown). In addition to that, one should be able to measure an additional range of $Q_{\max}/2$ beyond the interval limits used for the key generation to distinguish noise from tamper attempts (thus, requiring a large measurement range). These considerations were not included in [206].

Reducing the information leakage: Since $Q_{\min}$ must be chosen according to the measurement noise σ_{N}, it is safe to assume that measurement noise does not increase in values being more distant to μ_X. Hence, by limiting any value ${}^{\mathrm{Q}}W^{v}$ by ${}^{\mathrm{Q}}W \leq Q_{\min}/2$ one still achieves robustness with regard to the measurement and thereby reduces the information leakage caused by ${}^{\mathrm{Q}}W^{v}$. This still leads to a certain information leakage, as the value ${}^{\mathrm{Q}}W = Q_{\min}/2$ is much more likely to occur. Moreover, at the same time one further increases the space left to ${}^{\mathrm{A}}W$ which increases towards $Q_{\max} - Q_{\min}/2$ for the largest intervals.

Table 6.1: Comparison of several design parameters for different quantization profiles.

Parameter	$\mathbf{P}_1$	$\mathbf{P}_2$	$\mathbf{P}_3$	$\mathbf{P}_4$
Quantizer	equiprobable	equiprobable	equiprobable	equidistant
$\mathrm{P_e}(Y^v) \lesssim 10^{-6}$	yes	yes	yes	yes
$\mathrm{H}(Y)$ in bit	3	3	3	~ 2.9
$Q_{\min}\,[2\,\sigma_\mathrm{N}]$	2.9	2.9	2.9	5.3
$Q_{\max}\,[2\,\sigma_\mathrm{N}]$	∞	17.5	17.5	5.3
$\max(^{\mathrm{A}}W)\;[\sigma_\mathrm{N}]$	∞	17.5	29.2	5.3
n bits	90	90	90	120
k bits [a]	66.4	66.4	66.4	60
t bits [b]	4	4	4	–

[a] For equiprobable quantization, k is based on an optimal error correcting code [206], e.g., a code with parameters $[n, k, 2\,t + 1]$. For equidistant quantization, k is half the size of n due to requirements stated in NIST 800-90b.
[b] t bits an error correcting code corrects. Considered as negative impact on tamper-sensitivity.

6.4 Comparison of Quantization Schemes

To make a fair comparison of the quantization approaches, both techniques were applied to the empirical data of the Coating PUF [206] which is based on a system with $v = 30$ nodes. Moreover, variants of the equiprobable approach are considered to address the identified issues.

The results of this case study are listed in Table 6.1. The comparison includes the following quantization profiles $\mathbf{P}_p$, with $p = 1, \ldots, 4$.:

- $\mathbf{P}_1$: Equiprobable quantization, as originally proposed for the Coating PUF in [206] and as described in Section 6.3, i.e., $Q_{\max}$ is not limited.
- $\mathbf{P}_2$: Modified equiprobable quantization approach as outlined above to limit size of $Q_{\max}$, thereby representing the practically relevant case where this limit imposed is by the limits of the measurement circuit.
- $\mathbf{P}_3$: In addition to the modification of Profile 2, still based on the equiprobable quantization, the leakage of the helper data ${}^{\mathrm{Q}}W^v$ is reduced by limiting the length of each offset to ${}^{\mathrm{Q}}W \leq Q_{\min}/2$.
- $\mathbf{P}_4$: The proposed equidistant quantization of Section 6.2.

To define the equiprobable intervals of P_1, the PDF was partitioned with $L = 2^3 = 8$ intervals. This initially resulted in quantization interval boundaries indicated by a dashed line in Figure 6.2a. $Q_{\max}$ is therefore infinitely large and only limited by the measurement range. The number of intervals determines $Q_{\min} = y \cdot 2 \cdot \sigma_\mathrm{N}$ with $y = 2.9$. The entropy is 3 bit and also represents the number of extracted bits per physical node. Hence, a total of $n = 3 \cdot v = 90$ bit is extracted

Because of the negative impact on the security of P_1, the interval boundaries were adjusted to restrict the valid range used for the key generation by not considering values that occur with less than 0.1%. These boundaries are illustrated by the solid lines in

Figure 6.2b. This limits the width of $Q_{\max}$ but does not restrict the information leakage (Profile 2).

In addition to the properties of P_2, the information leakage was now reduced for P_3 by limiting the range of values of ${}^{\mathrm{Q}}W$. This does not cause any change to the width of the intervals. The interval $Q_{\max}$ is therefore still significantly larger than $Q_{\min}$, resulting in a poor detection-capability within these intervals.

In contrast, the equidistant approach of $\mathbf{P}_4$ resulted in $L = 2^4 = 16$ intervals that could be used for key-generation (dashed interval boundaries of Figure 6.1b). The parameter y was chosen as 5.3 to give sufficiently stable results even without ECC. Each of the intervals has a width of $Q_{\mathrm{w}} = 5.3 \cdot 2 \cdot \sigma_{\mathrm{N}} = Q_{\min} = Q_{\max}$. Due to the chosen approach and number of intervals are 4 bit necessary to encode the value of each node, yielding $n = 4 \cdot 30 = 120$ bit. However, these only contain an entropy $\mathrm{H}(Y)$ which is close to 3 bit.

Observations and Results Concerning the reliability and the extracted entropy, both approaches offer reasonable results for a key mismatch probability of $\mathrm{P_e}(Y^{\upsilon}) \lesssim 10^{-6}$. However, by considering a worst-case attacker, the tamper-sensitivity of the equidistant approach is at least three-times better than the equiprobable variants at the stage of quantization. For an actual system with equiprobable quantization, one must also consider the additional error-correction of the ECC, since this would allow to completely destroy a single physical node without being detected. This is later done in Chapter 9 by introducing a more advanced notion of tamper-sensitivity.

Hence, if attacks on the design succeed because of insufficient tamper-sensitivity, it is possible to improve this by using the equidistant approach which also reduces the information leakage by the quantization helper data ${}^{\mathrm{Q}}W^{\upsilon}$. Alternatively, one can still consider using Profile 3 based on equiprobable quantization (or the partitioning method [215, 192]) but should be aware of the reduced tamper-sensitivity.

6.5 Conclusions on Quantization

In this chapter, we analyzed how to quantize a continuous range entropy source that represents a tamper-evident PUF. One of the results is that at the stage of quantization, achieving optimal tamper-sensitivity and equiprobability of bits is a conflicting requirement. Considering this, one should always take the worst-case tamper-sensitivity into account and prioritize this metric once sufficient entropy has been extracted.

Another part of this work analyzed equiprobable quantization as one possible step of an overall key derivation process from a tamper-evident structure. It has been discovered that a certain information leakage in this scheme is present and how to reduce it. Moreover, by bridging the gap between the formal description of this approach and the practical realization, we indicated that optimal tamper-sensitivity should also consider a certain range outside of the actual quantization intervals to better detect attacks within the outermost intervals.

Further building upon the obtained insight, we developed a new approach to derive a key from a tamper-evident PUF which is based on equidistant quantization intervals. This leads to an improved tamper-sensitivity without significant information leakage in ${}^{\mathrm{Q}}W^{\upsilon}$.

In the following chapters, two possible follow-up ECC schemes are analyzed, i.e., either one or the other can be used to further process the resulting symbols of the equidistant quantization. In Chapter 7, symbols are mapped to a variable-length bit representation such that the bias in the data is reduced. As an alternative approach in Chapter 8, the symbols

are interpreted as is and processed by a Limited Magnitude Code. Hence, two different schemes are investigated that are both designated to continue the data processing within the scope of the presented PUF model of Figure 5.2 and the equidistant quantization.

Chapter 7

ECC for Variable-Length Bit Mappings of Higher-Order Alphabet PUFs

This chapter briefly presents the concepts that form the foundation of our proposed scheme for a variable-length bit mapping of higher-order alphabet symbols as a new approach for PUF-based ECC. First, the Levenshtein distance is discussed and its applicability to quantify the distortion by insertion/deletion errors. Afterwards, VT codes are covered as a code class to counteract errors of this type. Then, we introduce the specifics of our variable-length bit mapping scheme in Section 7.3. This chapter is based on preliminary ideas proposed by the thesis author in 2014 (back then without knowing that VT codes even existed). Later on, Matthias Hiller and the thesis author jointly supervised a master's thesis carried out by Qinzhi Liu [167] that was essential to create a working approach which was later published in [92, 93] with the thesis author as principal author. Antonia Wachter-Zeh and Andreas Lenz provided valuable guidance on this topic, in particular the code construction to also correct substitution errors.

Contents

7.1 Introduction to Variable-Length ECC

Based on the obtained insight in the previous chapter, we select equidistant quantization as processing step prior to applying an ECC. However, as a result of equidistant quantization is the binary sequence heavily biased when using a fixed-length mapping from symbols to bits. To address this issue, we follow the information-theoretical intuition of quantizing values with different probabilities of occurrence to binary sequences of varying length, i.e., values that occur more often are assigned a shorter binary representation and vice-versa.

Therefore, the output binary data will be nearly unbiased and the underlying equidistant quantization is less prone to leak secret information due to stored helper data of the ECC.

Unfortunately, following this idea comes at the expense that a large body of previous work on error correction can no longer be applied to the quantized bit sequence of a PUF. This is owed to the fact that if noise exceeds the tolerance of the quantization scheme, the *length* of the considered sequence changes. A change in length is either called an *insertion* if it gets longer, or a *deletion* if it gets shorter. If the length remains the same but an error occurs this is called an *substitution* error, i.e., in the binary case this is a bit flip.

Commonly known ECCs are directed towards correcting *substitution* errors, typically by taking into account the Hamming distance of sequences. Since one insertion or deletion does not only affect the erroneous symbol itself, but also shifts all subsequent symbols, codes in the Hamming metric are not able to efficiently correct insertion or deletion errors.

The challenge therefore is to use codes capable of correcting errors that stem from variable-length bit mappings within the context of ECCs, i.e., they must address common design issues of PUF key derivation schemes such as reliability and secrecy leakage in the helper data. To do so, we leverage the properties of Varshamov-Tenengolts (VT) codes [125] that are able to correct insertion and deletion errors. In fact, we use a variation of the original VT codes that also covers substitution errors.

Let us briefly consider the following practical example to further motivate this topic: let $Y = [1, 0, 1, 0, 1, 0, 1]$ be the designated bit sequence and $\hat{Y} = [1, 1, 0, 1, 0, 1]$ a shorter received sequence where a deletion occurred at the second position of Y. Since the Hamming distance is not defined between vectors of unequal length, one could artificially pad $\hat{Y}$ with a zero which results in $\mathrm{d}_{\mathrm{H}|2}(Y, [\hat{Y}, 0]) = 6$, i.e., 6 substitution errors. This large distance highlights that it is impractical to rate deletions (and similarly, insertions) by the Hamming metric which is only suited for substitution errors, i.e., bit flips occurring between bit sequences of equal length.

To better reflect the nature of the error, Levenshtein [125] defined the distance $\mathrm{d}_{\mathrm{Lev}}(Y, \hat{Y})$ as the smallest number of insertions, deletions, and substitutions that are required to transform $\hat{Y}$ into Y. Hence, $\mathrm{d}_{\mathrm{Lev}}(Y, \hat{Y}) = 1$ for the given example. In the following, we review VT codes that form a class of codes that can correct errors in the Levenshtein metric which are thus able (including minor adjustments) to operate on our proposed custom variable-length bit mapping of the symbols.

7.2 VT Codes for Insertion/Deletion Error Correction

Varshamov-Tenengolts (VT) codes have been introduced to address insertion and deletion errors and correct a single insertion or deletion [213, 188]. For a fixed integer $a \in \{0, \ldots, n\}$, a binary VT code of length n is defined as the set of all vectors $C^n = (c_1, c_2, \ldots, c_n) \in \{0, 1\}^n$ such that

$$\sum_{i=1}^{n} i \cdot c_i \equiv a \pmod{M}, \tag{7.1}$$

where $M \geq n + 1$. The integer a is called the checksum (or syndrome). VT codes with $M = n+1$ are conjectured to be optimal in the sense that they have the largest cardinality of all single-deletion correcting codes [188]. The highest code rates are obtained for $M = n+1$ and $a = 0$. Based on the pigeonhole principle, for every M, there exists a checksum a, such that size of the code is at least $\frac{2^n}{M}$ and its redundancy therefore at most $\log_2(M)$ bits.

However, this basic construction with $M = n + 1$ is unable to correct substitutions and only works when the type of error is already known, i.e., the length of the received word must be provided. We will therefore use $M = 2n + 1$ for our constructions, since in this case, the VT code is able to correct a single insertion, deletion or substitution [213].

The procedure to construct *systematic* VT codes according to [177] is as follows: For a binary input sequence $(y_1, \ldots, y_m)$, the corresponding codeword has the form $(c_1, \ldots, c_n)$ where $y_1 = c_{i_1}, y_2 = c_{i_2}, \ldots, y_m = c_{i_m}$, $1 \leq i_1 < i_2 < \cdots < i_m \leq n$. The remaining bits c_k, where $k \notin \{i_1, i_2, \cdots, i_m\}$ are called parity bits and are located at positions $k = 2^l$, for $l \in \mathbb{N}$ and $k \leq n$, and additionally at position n. For a codeword of length n, the number of parity-check bits is therefore $r = \lceil \log_2 n \rceil + 1$.

For M such that $2n \leq M \leq \min(n + 2^{r-1}, 2^r)$, the parity-check bits $(p_1, \ldots, p_r)$ are chosen according to

$$\sum_{l=1}^{r-1} p_l \cdot 2^{l-1} + p_r \cdot n + \sum_{j=1}^{m} i_j \cdot y_j \equiv 0 \pmod{2n}, \tag{7.2}$$

such that the constructed codeword C^n has checksum 0. Note, that “systematic" in this setting does not imply that the *first* m bits contain the information, instead they are distributed to positions which are not a power of 2 or equal to n. Extending this systematic encoding with the capability to also correct one substitution error comes at the expense of storing one additional redundancy bit.

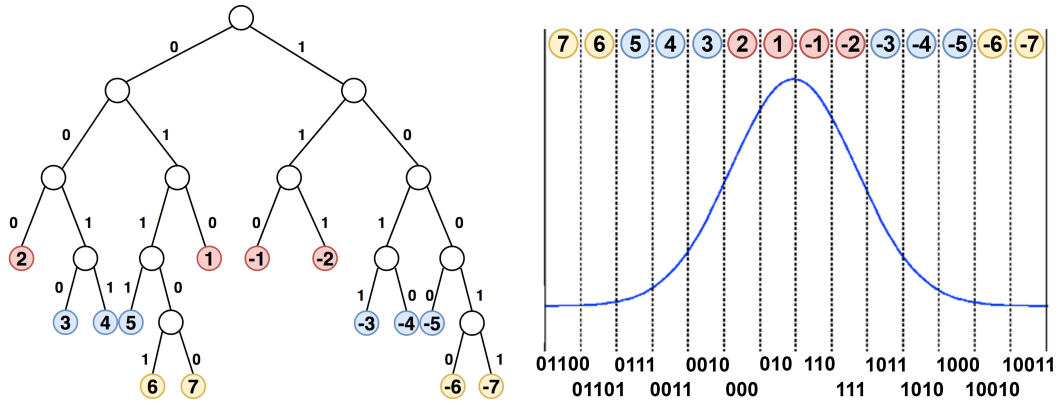
In the considered PUF scenario, only parts of the codewords are transmitted since parity bits are stored as public helper data. The helper data is assumed not to be corrupted, so we can retrieve it without errors, similarly to [34]. However, message bits may contain errors at unknown positions as they are drawn from the noisy PUF.

Consequently, the standard systematic VT code cannot be employed in PUFs because when recovering the response from the PUF, the positions where to insert the parity-check bits cannot be determined. It is therefore necessary to fully separate parity-check bits from the message containing secret information such that parity bits and codeword bits are no longer interleaved. This is explained in Section 7.4.1.

7.3 Variable-Length Bit Mapping for Higher-Order Alphabet Symbols from Equidistant Quantization

Ideally, the mapping of higher-order alphabet symbols to bits is such that the obtained sequence is not biased, i.e., the ones and zeros are uniformly distributed at the stage of quantization already. In addition, the mapping should support the subsequent error correction in terms of low distance changes from one to another quantization interval. At the same time this improves tamper-sensitivity, as errors resulting in a large distance to the designated value are almost certainly caused by a physical attack and should – as intended – cause the device to fail.

To achieve low distance changes for neighboring quantization intervals in Hamming distance, i.e., $d_H = 1$, one would use a Gray code [56]. However, it cannot be applied in our case, since this scheme only works for fixed-length bit mappings as opposed to variable-length bit mappings. These variable-length bit mappings are required to overcome the bias of fixed-length bit mappings, i.e., certain patterns of ones and zeros are more likely to occur in a fixed-length bit mapping of symbols, thereby causing the bias. To overcome these limitations, we propose a new variable-length bit mapping scheme (cf. Figure 7.1).



(a) Tree for variable-length bit mapping. **(b)** Resulting bit assignment.

Figure 7.1: Proposed variable-length bit mapping for equidistant quantization.

In order to preserve the entropy at the stage of quantization, when mapping its symbols to the binary domain, a uniquely decodable code is required, e.g., it should be prefix-free. Therefore, we build a binary tree to explicitly assign symbols to a variable-length bit mapping that differs only in $d_{Lev} = 1$ for neighboring intervals. Hence, it is the Levenshtein counterpart to the Gray code. Notice that a Huffman code, a standard construction for a variable-length prefix-free code, is not an eligible candidate here as it neither ensures a debiasing characteristic due to the lack of equiprobability of zeros and ones, nor is the constraint of $d_{Lev} = 1$ for neighboring intervals considered.

In contrast, our construction follows the principle of a prefix-free code, where each leaf in a tree is connected to only one parent node. For the resulting symbols of adjacent quantization intervals, the desired distance of $d_{Lev} = 1$ is achieved. By traversing the graph either to the left or right, bit 1 or 0 is incorporated in the pattern. Unfortunately, there is no way yet to generalize this construction yet. The resulting mapping for 14 intervals is represented by Figure 7.1. It is well-suited for the application based on the following perspective:

- As long as the input distribution is symmetric, 0s and 1s are balanced, since equally probable intervals have an equal number of 1s and 0s.
- It fulfills the requirement that adjacent intervals only differ by one insertion/deletion/substitution error, i.e., adjacent intervals have $d_{Lev} = 1$.
- It is prefix-free, i.e., there is no whole code word in the bit mapping that is a prefix (initial segment) of any other code word in the bit map (cf. Figure 7.1b). This makes it uniquely decodeable and preserves the information provided by the quantization while requiring less redundant bits when compared to a fixed-length bit mapping of the symbols.
- It has a debiasing property, i.e., more probable symbols are assigned shorter bit mappings and less probable symbols are assigned longer bit mappings.

To substantiate our claims, we simulated 1000 devices with 128 nodes each, based on the PUF system model in Figure 5.2, using the PUF distribution given in Chapter 9. The resulting output data was then analyzed by the NIST 800-90b [149] test suite, a framework

to assess the properties of entropy sources. This approach leads to the results presented in Table 7.1. While the obtained bit string of the variable-length encoded symbols is (as expected) shorter compared to fixed-length symbols obtained by applying a Gray code, the per-bit *min*-entropy of the variable-length encoding is much higher. This fits well the analytical results later shown in Table 7.2, i.e., dividing the *min*-entropy per symbol by the expected bits per node should only result in a slightly *lower* number when compared to the output of the NIST 800-90b test. The results of this experiment also show that the overall entropy output is the highest for the variable-length encoding. Please note that Table 7.2 studies the effects of a different number of quantization interval with regard to the extracted entropy, obtained variable-length bit sequence, and the error-rate after quantization *prior to* applying the VT-like ECC.

Table 7.1: NIST 800-90b test results for variable-length and fixed-length bit mapping using Gray code (4 bit per symbol). The tested data was generated by simulating the output of 1000 devices with 128 physical nodes each.

Setting: L= 14 ($y = 4.24$)	Variable-length code	Gray code
average output length [bit per device]	431	512
min-entropy [bit]	0.79	0.56
min-entropy [bit per device]	$0.79 \cdot 431 = 340.5$	$0.56 \cdot 512 = 286.7$

Unfortunately, since the PUF device comprises multiple nodes from which values are drawn, the probability for an error to occur increases quickly the more PUF nodes contribute to a single codeword. This may lead to the situation that the error-correcting capability of the VT-codes is exceeded, as they typically correct only one error. To counteract this effect, it is necessary to develop a segmentation strategy, i.e., how to efficiently group fewer nodes together without compromising reliability or security. Let us consider two different segmentation strategies, namely Case 1 and Case 2, that form the input of Section 7.4 and the publication in [93] respectively. In either case, and in accordance to the previous PUF model of Section 5.2, a single PUF device is assumed to have v nodes that are subject to the quantization, whereas Y^v is the output quantized response*.

Case 1. Fixed-Number of Nodes per Segment. Here, one segment is chosen to contain u nodes, i.e., the output of one segment is Y^u, where $Y \in C_{\mathrm{VT}}$. The overall output sequence is therefore divided into $z = \lceil \frac{v}{u} \rceil$ segments.

Case 2. Fixed Bit-Length per Segment. Here, a fixed segment bit-length m is set as a parameter for the whole system. This m sets the upper bound for the bit length of one segment. Subsequently, the variable-length symbols are assigned to the first segment. For as long as a symbol's bit sequence fits inside the first segment, the next symbol will be considered. Once the upper bound m is reached, a new segment is created and the process repeated. If the bit sequence of a single symbol does not fully fit into the remaining bit positions of a segment, a padding of 0s is inserted at the end of the segment such that the length m is reached. The symbol which could not be put into the previous segment is then inserted into the subsequent segment. This strategy has been covered in [93].

* For example, the element Y_1 is the output of a single node which is a symbol of the variable-length bit-mapping.

Table 7.2: Effect of equidistant quantization under different parameters and resulting data for entropy (per node), length of bit mapping, and reliability.

Number of Intervals	min Entropy	Shannon Entropy	Bits per Node	Bits per Device	97% Confidence Interval	$P_e(Y^v)$ (before ECC)
12 ($y = 4.95$)	2.26	2.92	3.27	419	[406, 430]	9.5×10^{-5}
14 ($y = 4.24$)	2.47	3.13	3.36	430	[417, 443]	2.8×10^{-3}
16 ($y = 3.71$)	2.65	3.33	3.51	449	[433, 466]	2.6×10^{-2}
18 ($y = 3.30$)	2.81	3.49	3.73	478	[457, 500]	1.2×10^{-1}
20 ($y = 2.97$)	2.96	3.64	3.92	502	[482, 517]	3.1×10^{-1}

7.4 VT-like Code and Fixed-Number of Nodes Segmentation

In the following, we present our systematic VT-like code construction for PUFs. This is based on the fixed-number of nodes per segmentation case (Case 1).

7.4.1 Systematic VT-Like Code Construction for PUFs

This section introduces a code to address a single insertion, deletion or substitution error that originates from a quantization error and subsequently stems from the bit mapping as introduced in Section 7.3. We propose a VT-like code construction for the situation that the parity-check bits are not transmitted within the input bit stream and are thus error-free, i.e., they are stored in a non-volatile memory. Our construction is as follows:

$$C_{\mathrm{VT}} := \Big\{(y_1, \cdots, y_m, p_1, \cdots, p_r) : \sum_{i=1}^{m} i y_i + \sum_{j=1}^{r} 2^{j-1} p_j \equiv 0 \pmod{2m+1}\Big\}, \tag{7.3}$$

where m information bits and r parity-check bits together form a codeword of length $n = m + r$. The number of check bits of this code construction is $r = \lceil \log(2m+1) \rceil$ and smaller than the redundancy of the systematic construction from [177] where the redundancy depends on n. In the following, we show how C_{VT} can correct one deletion, insertion, or substitution error. The decoding procedure is similar to the decoding of classical VT codes [188]. Let us consider an example with a single deletion based on the following notation that is also used in Table 7.3.

- π: the location of the error, indicating that x_π is corrupted; $\pi = \lambda_1 + \lambda_0 + 1$,
- ω: number of 1s in received bit stream, i.e., the Hamming weight
- λ_1: number of 1s left of position π
- λ_0: number of 0s left of position π
- ρ_1: number of 1s right of position π
- m: number of encoded information bits

Assume that the π-th bit in the original bit sequence was deleted, which has λ_0 zeros to the left of it, ρ_0 zeros to the right of it, λ_1 ones left of it and ρ_1 ones right of it. Therefore, $\pi = 1 + \lambda_0 + \lambda_1$. Let ω be the Hamming weight the received bit stream, i.e., $\omega = \lambda_1 + \rho_1$. Evaluating the sums in Equation 7.3, the deficiency Δ of the new checksum compared to the original one is

$$\Delta = -(\pi \cdot y_\pi + \sum_{i=\pi+1}^{m} y_i) \pmod{(2m+1)} \tag{7.4}$$

When a 1 was deleted, the checksum deficiency is

$$\begin{aligned}
\Delta &= -(\pi + \rho_1) && (7.5)\\
&= -(1 + \lambda_0 + \lambda_1 + \rho_1) && (7.6)\\
&= -(1 + \lambda_0 + \omega) && (7.7)\\
&\equiv 2m + 1 - (1 + \lambda_0 + \omega) \pmod{2m+1} && (7.8)
\end{aligned}$$

To recover the initial input, one needs to insert a one at the right side of λ_0 zeros in the received sequence. When a zero was deleted, the new checksum is ρ_1 less than the original, i.e., $\Delta = 2m + 1 - \rho_1$. To recover, one needs to insert a zero on the left side of ρ_1 ones. The case for insertion errors can be solved in a similar manner.

For substitution errors, the error pattern where a 0 flips to 1 gives a deficiency Δ of the position number, i.e., π. Vice-versa, if 1 changes to 0, the deficiency Δ is the value of $2m + 1 - \pi$. The range of values for the checksum deficiency Δ for insertion, deletion, and substitution errors is given in Table 7.3.

Table 7.3: Checksum Deficiency Δ vs. Error Pattern.

Error Type	Error Pattern	Δ	Range of Δ
Insertion	insert 0	ρ_1	$[0, \omega]$
Insertion	insert 1	$\pi + \rho_1 = \omega + \lambda_0$	$[\omega, m+1]$
Deletion	delete 0	$-\rho_1 + 2m + 1$	$[2m + 1 - \omega, 2m] \cup \{0\}$
Deletion	delete 1	$-\rho_1 - \pi + 2m + 1$	$[m+1, 2m - \omega]$
Substitution	flip 0 to 1	π	$[1, m]$
Substitution	flip 1 to 0	$2m + 1 - \pi$	$[m+1, 2m]$

The table shows that the range of the two cases of insertions overlap in ω. The error correction here can be explained as follows: for an insertion error, if $\Delta = \omega$, there is either a 0 or 1 inserted in the beginning. For this case, we delete the first bit to correct the insertion error. Algorithm 7.4.1 shows the decoding procedure for our proposed VT-like code construction. It generalizes the systematic decoding process of the discussed example.

In Algorithm 7.4.1, l_I denotes the length information $m \pmod 3$ which is stored as helper data. It allows to identify the error type. Recall that $\hat{X}$ is the output of the measured PUF values, $\hat{Y}$ is the quantized output, and $\hat{Z}$ the secret bit sequence, as illustrated in Figure 5.2. Hence, we propose the following theorem.

Theorem 7.4.1 *If $p_1, \ldots, p_r$ are chosen according to construction C_{VT} from (7.3) and known to the decoder, it is possible to correct one insertion, deletion, or substitution error in $(y_1, \ldots, y_m)$ by using Algorithm 7.4.1.*

Algorithm 7.4.1: VT-like Systematic Decoding Algorithm for PUFs

Data:
l_{I} = (Length information)
Δ = (Checksum deficiency)
$\hat{Y}$ = (noisy quantized PUF response)
$\hat{m}$ = (bit length for reference PUF response)
Result: $\hat{Z}$ = (corrected secret bit sequence)

if $\hat{m} \equiv l_{\mathrm{I}} \pmod 3$ **then**
  /* substitution error or error-free,i.e., $\hat{m} = m$ */
  if $\Delta = 0$ **then**
    No error ; // $\hat{Z} \leftarrow \hat{Y}$
  else
    if $\Delta > \hat{m}$ **then**
      $\hat{Y}[2\hat{m} + 1 - \Delta] = 1$; // substitution error from 1 to 0
    else
      $Y[\Delta] = 0$; // substitution error from 0 to 1
    end
  end
  $\hat{Z} \leftarrow \hat{Y}$
else if $\hat{m} + 1 \equiv l_{\mathrm{I}} \pmod 3$ **then**
  /* deletion error, i.e., $\hat{m} = m - 1$ */
  if $\Delta = 0$ **then**
    $\hat{Z} \leftarrow \hat{Y}$ with 0 inserted at the end
  else
    if $\Delta > 2 \cdot \hat{m} + 3 - \omega$ **then**
      insert 0 at left side of ρ_1 1's on the right ; // $\rho_1 = 2\hat{m} + 3 - \Delta$
    else
      insert 1 at right side of λ_0 0's on the left ; // $\lambda_0 = 2\hat{m} + 2 - \omega - \Delta$
    end
    $\hat{Z} \leftarrow \hat{Y}$
  end
else
  /* insertion error, i.e., $\hat{m} = m + 1$ */
  if $\Delta = 0$ **then**
    $\hat{Z} \leftarrow \hat{Y}$ with 0 deleted at the end
  else
    if $\Delta > \omega$ **then**
      delete 1 at the right side of λ_0 0's on the left ; // $\lambda_0 = \Delta - \omega$
    else
      delete 0 at the left side of ρ_1 1's on the right ; // $\rho_1 = \Delta$
    end
    $\hat{Z} \leftarrow \hat{Y}$
  end
return $\hat{Z}$

To also guarantee correction of substitution errors, we increased the argument of the modulo operation to $2m + 1$. If we only have an insertion or deletion error, we use the following code definition which has one bit less redundancy:

$$\left\{(y_1 \cdots y_m, p_1 \cdots p_r) : \sum_{i=1}^{m} i \cdot y_i + \sum_{j=1}^{r} 2^{j-1} \cdot p_j \equiv 0 \pmod{m+1}\right\}. \tag{7.9}$$

7.4.2 Reliability of VT-like Scheme

After error-correction using the VT-like code, the noise tolerance has tripled to $3 \cdot Q_w$ for one node in comparison to just using an equidistant quantization scheme as presented in Chapter 6. Therefore, same values of the safety parameter y now offer a much better reliability compared to a pure quantization.

However, for each segment of nodes still only one error can be corrected due to the properties of the constructed code. This limitation is preferred, as a physical attack that causes a large increase in Levenshtein distance from the reference value should *not* be corrected. Heavily distorted measurement values occur from noise only with small probability, so multiple errors outside of the CI $[-y \cdot \sigma_N, +y \cdot \sigma_N]$ interval should cause the system to fail, thereby improving tamper-sensitivity.

We first calculate the error probability $P_e(Y)$ of a node by integrating over the PDF of the noise. Then we apply the VT-like code for error correction to obtain the corresponding error probability for a segment, if more than one node is corrupted with $d_{Lev} = 1$. Finally, for an error-free device, all of its segments must be correct. The node error probability $P_e(Y)$ *before* applying the VT-like ECC is calculated by the PDF of a Gaussian distribution with $\mathcal{N}(\mu, \sigma)$ as follows:

$$P_e(Y) = 1 - \int_{-y \cdot \sigma_N}^{+y \cdot \sigma_N} \mathcal{N}(0, \sigma_N). \tag{7.10}$$

Without error correction, i.e., $Z = Y$ and $\hat{Z} = \hat{Y}$, a segment with u nodes will pass comparison of $Z \stackrel{?}{=} \hat{Z}$ only if all its nodes are quantized correctly. This corresponds to a segment error probability P_s of

$$P_s(Y^u) = 1 - (1 - P_e(Y))^u. \tag{7.11}$$

Here, the aim is to correct the error when the encoded value shifts into adjacent intervals. Hence, per segment, only one node with $d_{Lev} = 1$ must be corrected. The error probability $P_e(Z)$ that a single node is *not* correct *after* applying the VT-like ECC is:

$$P_e(Z) = 1 - \int_{-3 \cdot y \cdot \sigma_N}^{+3 \cdot y \cdot \sigma_N} \mathcal{N}(0, \sigma_N). \tag{7.12}$$

This is based on the fact that the variable-length bit mapping has been designed such that neighboring intervals are of distance $d_{Lev} = 1$ and therefore will be corrected by the VT-like code.

The error probability $P_e(Z^u)$ after VT error correction is

$$\mathrm{P_e}(Z^u) \leq 1 - \left(u(1-\mathrm{P_e}(Y))^{u-1}(\mathrm{P_e}(Y)-\mathrm{P_e}(Z)) + (1-\mathrm{P_e}(Y^u))\right) \tag{7.13}$$
$$= 1 - \left(u(1-\mathrm{P_e}(Y))^{u-1}(\mathrm{P_e}(Y)-\mathrm{P_e}(Z)) + (1-\mathrm{P_e}(Y))^u\right) \tag{7.14}$$

This equation is structured as follows: the first part describes a device that had one erroneous segment before applying the ECC but it is corrected afterwards times the probability that only one node was in error (which is a direct result of the previous assumption of one error per segment) minus the probability that the device had no error at all even without ECC.

We additionally note that the probability in Equation (7.12) assumes that only adjacent intervals differ in one bit, i.e., a single insertion/deletion/substitution error. However, in the process of building the codebook, one cannot avoid that nearby intervals other than the adjacent ones also differ in one bit.

Hence, the probability of the analytically computed error rate upper bounds the error probability and simulated results should slightly outperform the calculations. This difference can be practically observed, whereas the margin is larger for a higher error-rate and smaller for a lower error-rate. For a device with z segments, the overall device error probability after error-correction $\mathrm{P_e}(Z^v))$ is finally given by

$$\mathrm{P_e}(Z^v)) = 1 - (1-\mathrm{P_e}(Z^u))^z. \tag{7.15}$$

As listed in Table 7.2, we observe for a device with 128 nodes that increasing y leads to an improved reliability at the expense of loss in entropy and shortened length of the bit sequence. Therefore, a designer's goal is to maximize the number of secret bits while meeting the reliability requirement. The performance numbers including the VT-like code are presented in Table 9.1 alongside several other constructions for comparison reasons.

7.4.3 Information Leakage caused by VT-like ECC

To determine the amount of leakage between encoded sequence Y^v helper data ${}^{\mathrm{ECC}}W = (L_\mathrm{I}, P^*)$, we select one of our later results (first entry of Profile 4) from Table 9.2 that meets the reliability requirements and has the largest number of effective secret bits. For other selected parameters, the calculation is similar.

The first source of leakage is caused by the stored length information l_I. It is stored for each segment and may have 3 possible values only. Therefore $\mathrm{I}(Y^v; L_\mathrm{I})$ is considered as worst-case if rounded-up, i.e.,

$$\mathrm{I}(Y^v; L_\mathrm{I}) \leq \mathrm{H}(\mathrm{L_I}) \leq \lceil \log_2(3) \rceil = 2\,\mathrm{bits}$$

The second source of leakage is based on the parity bits P^* of the VT code. For a segment with $v = 128$ node values, the maximum entropy of these parity bits is therefore considered as information leakage $\mathrm{I}(Y^v; P^*)$. Please note, for the subsequent calculation, the maximum length of the segment is used as upper bound for the leaked bits. For the specific example, the code size determines the maximum entropy, i.e., here, resulting in the size of P^*. The remaining multiplicative factor of 2 and additive component + 1 is due to the structure of the code, cf. Equation (7.3):

$$\begin{aligned} \mathrm{I}(Y^{128}; P^*) &\leq \mathrm{H}(P^*) && (7.16)\\ &\leq \lceil \log_2(2m+1) \rceil && (7.17)\\ &= \lceil \log_2(2 \cdot 5 \cdot 128 + 1) \rceil && (7.18)\\ &= 11 \,\text{bits} && (7.19) \end{aligned}$$

Hence, the overall number of leaked bits based on a worst-case assumption is

$$\mathrm{I}(Y^{128}; {}^{\mathrm{ECC}}W) \leq 2 + 11 = 13 \,\text{bits} \tag{7.20}$$

Concerning the min-entropy that is extracted on average from a device, we consider each node with $y = 4.95$ (resulting in 12 quantization intervals) which leads to a *min*-entropy of 2.26 bit per node, according to Table 7.2. This gives

$$\tilde{\mathrm{H}}_\infty(Y^v) = 2.26 \cdot 128 = 289.3 \,\text{bits} \tag{7.21}$$

Hence, for a device with 128 nodes, the number of overall effective secret bits is

$$\tilde{\mathrm{H}}_\infty(Y^v) - \mathrm{I}(Y^{128}; {}^{\mathrm{ECC}}W) = 289.3 - 13 = 276.3 \,\text{bits} \tag{7.22}$$

7.4.4 VT-like Code Example

In the following toy example, we demonstrate the encoding and decoding of our VT-like code. Based on PUF nodes with $y^8 = [5, 4, -3, -6, 7, -1, 2, 4]$. The symbols are encoded according to the bit mapping presented in Section 7.3, i.e.,

$$\mathrm{enc}(y^8) = [(0111), (0011), (1011), (10010), (01100), (110), (000), (0011)]. \tag{7.23}$$

Afterwards, 4 symbols are combined to one VT codeword. The first 4 symbols are encoded to a binary sequence of length 17. Therefore $l_{\mathrm{I}}(y^4) = 17 \equiv 2 \pmod 3$. The left half of Equation 7.3 is

$$\sum_{i=1}^{17} i\, y_i = 2 + 3 + 4 + 7 + 8 + 9 + 11 + 12 + 13 + 16 = 85 \equiv 15 \pmod{35}. \tag{7.24}$$

The parity bits are a binary representation of $35 - 15 = 20$, so $p^6 = (010100)$. For the second part of the PUF response, we analogously calculate the helper data $l_{\mathrm{I}} = 15 \equiv 0 \pmod 3$ and $p^6 = (001111)$.

To demonstrate deletion and insertion error correction, let us assume that during reconstruction one quantization error occurred in the third symbol and another one in the seventh symbol, such that $\hat{y}^8 = [5, 4, \mathbf{-2}, -6, 7, -1, \mathbf{3}, 4]$. Therefore the third symbol is encoded to (111) instead of (1**0**11), which corresponds to one deletion error. Computing $l_{\mathrm{I}}(\hat{y}^4) = 1 \equiv 16 \pmod 3$ shows that the one bit was deleted:

$$\Delta = \sum_{i=1}^{m} i\, \hat{y}_i + \sum_{j=1}^{r} 2^{j-1} p_j = 81 + 20 = 101 \equiv 31 \pmod{33 + 2}. \tag{7.25}$$

$\Delta = 2 \cdot (16 + 1) + 1 - \rho_1$, therefore we have $\rho_1 = 4$ and insert 0 on the left of 4 1s in the right. Thus, we were able to detect the position of the deletion and correct the error. For the second half, let us assume that the third symbol shifted from 2 to 3 such that (00**1**0) is forwarded instead of (000). Now $l_{\mathrm{I}}(\hat{y}^4) = 1$. Since $l_{\mathrm{I}}(y^4) = 0$, one insertion occurred. $\Delta = 13$, so according to line 28 of Algorithm 7.4.1, we delete the 1 at the right side of $13 - 7 = 6$ 0s.

Chapter 8

ECC for Fixed-Length Bit Mappings of Higher-Order Alphabet PUFs

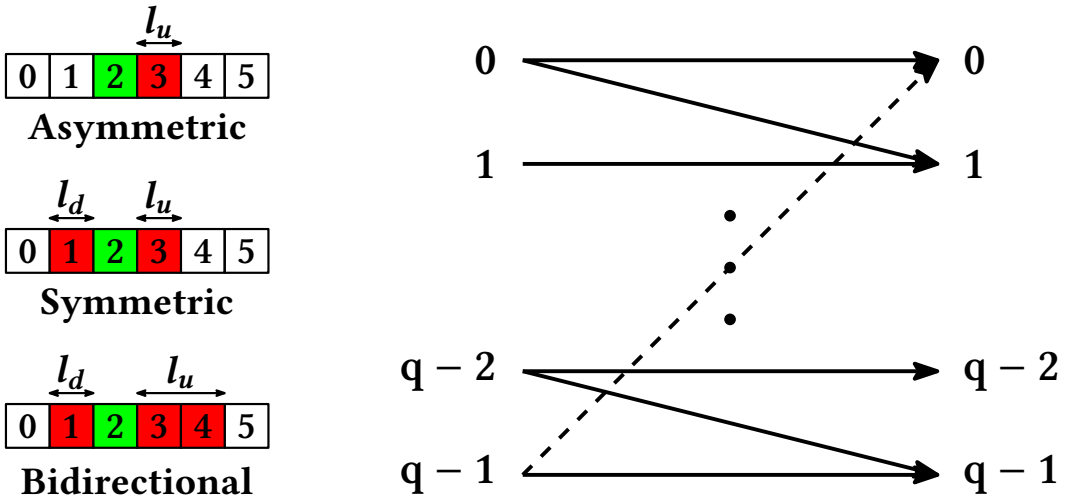
This chapter introduces Limited Magnitude Codes (LMC) as an optimized ECC to continue operating on symbols that are represented by a fixed-length bit mapping. This is an alternative to the approach presented in Chapter 7 and avoids the possible pitfalls of variable-length encoding such as difficulties in achieving a time-constant implementation. Moreover, LMCs turned out to be much more easily scalable and efficient. The work on LMCs emerged from a master's thesis by Karthik Uppund [207] that resulted in the publication in [100] with the thesis author as principal author.

Contents

8.1 Limited Magnitude Codes (LMC)

One of the problems by previous approaches is that mapping higher-order alphabet symbols to an alphabet of lower degree diminishes tamper-sensitivity by causing an unevenly spread TS in the codebook, as supported by our findings in Chapter 9. However, also building upon inappropriate distance metrics such as Hamming distance over symbols degrades tamper-sensitivity, as the Euclidean distance $\mathrm{d_E}(X, \hat{X})$ is not be well reflected by the Hamming distance over symbols $\mathrm{d_{H|}}_S(Y, \hat{Y})$. To solve these problems, we model the outcome of the equidistant quantization as a q-ary channel as depicted in Figure 8.1b, i.e., we continue operating on the symbols directly. In contrast to previous works, we rate errors in this channel by the *Lee* metric $\mathrm{d_{Lee}}$, i.e., symbols of neighboring intervals will have a distance of 1 whereas symbols of larger distance l will have distance l. This is also called the *magnitude*. Different possible types of magnitude errors are illustrated in Figure 8.1a. These are classified as *asymmetric* when unidirectional, *symmetric* when of equal magnitude in either direction, or *bidirectional* when in either direction but of unequal magnitude.

Elarief et al. [39] first proposed a code to correct all asymmetric and symmetric errors of limited magnitude in a q-ary channel. While the code proposed by [39] corrects *all* magnitude errors, it does not allow to limit the number of magnitude errors corrected by the ECC which does not match the exact requirements of the targeted application, where an attacker physically tampers with a subset of the PUF nodes. Correspondingly, the



(a) LMC error types for $q = 5$ and the designated symbol 2.

(b) q-ary asymmetric channel model for $l = 1$ between symbols. Non-wrap around (solid lines only) and wrap-around channel (solid lines and dashed line as wrap-around).

Figure 8.1: LMC error types and q-ary channel model.

designated code should only correct a subset of the PUF nodes. To address this shortcoming, Myeongwoon et al. [104] proposed a modified version of this code called Limited-Magnitude Error Correction Code (LMC). This is based on an RS Encode/Decode step that is additionally introduced to limit the number of correctable errors as later described. Hence, this can be considered as a concatenated code construction of LMC and RS codes, whereas we are not limited to RS codes but could have selected any other code operating on higher-order alphabet symbols. Although the new code by [104] was intended for bidirectional errors, it is equally applicable to asymmetric and symmetric errors.

The error correction capability of these codes is as follows (cf. Figure 8.1a): In Asymmetric LMC (A-LMC), a symbol is correctable if the possible error occurs in only one direction. For example, if the symbol is *2* then in A-LMC ($l_\mathrm{u} = 1$) the symbol is corrected only if it changes to *3* (error = +1). If the symbol changes to any other value, it is not corrected. Similarly for Symmetric LMC (S-LMC), the error magnitude can be ±1 i.e. $l_\mathrm{u} = |l_\mathrm{d}| = 1$. This implies that even if symbol *2* becomes *1*, it is corrected. Bidirectional LMC (B-LMC) is a generic case of S-LMC where $|l_\mathrm{u}| \neq |l_\mathrm{d}|$.

These error types can be considered within the scope of two different q-ary channel models. They are called wrap-around and non-wrap-around channel. In Figure 8.1b the wrap-around is indicated by a dashed line, whereas all other lines are solid and represent the only valid transitions for the non-wrap-around channel. Hence, for the wrap-around channel, $\mathrm{d}_\mathrm{Lee}(q-1, 0) = 1$, whereas for the non-wrap-around channel $\mathrm{d}_\mathrm{Lee}(q-1, 0) = q-1$. Since the underlying application is based on a physical measurement process, the wrap-around is not desirable and counteracts the aspect of tamper-sensitivity. Therefore, to best reflect $\mathrm{d}_\mathrm{E}(X, \hat{X})$ in the quantized symbols $\hat{Y}^v$, we only make use of the non-wrap-around channel model. The Lee metric in the non-wrap around channel is sometimes also termed *Manhattan* distance d_Man.

For encoding and decoding, the corresponding steps are listed in Algorithm 8.1.1 and Algorithm 8.1.2 that are described by [104]. The parameters of an LMC are $q\prime$, q, p and t. $q\prime$

represents the number of values a symbol can take under the influence of an error, while still being within the LMC boundary. q represents the number of quantization intervals. p is the RS code field size GF(p) and t is the error correction capability of the RS code. While constructing any LMC, Equation (8.1) must always hold.

$$q\prime = l_{\mathrm{u}} + |l_{\mathrm{d}}| + 1 \quad \text{and} \quad q\prime \leq q \leq p \tag{8.1}$$

The Encode Algorithm 8.1.1 and Decode Algorithm 8.1.2 are complemented by Algorithm 8.1.3 which is instantiated by both LMC Encode and Decode and helps translating an array of elements from one base to another, especially for the description presented here, assuming that q' is a power of 2, allowing for a very efficient implementation as demonstrated by the LMC examples in Section 8.3.

The basic idea of Algorithm 8.1.1 is that only a subset of the input message is effectively operated on. This can be thought of as only considering the Least-Significant Bits (LSBs) of a binary encoded integer. However, even low magnitude changes may cause a dramatic effect in the binary representation of an integer, e.g., from $7|_{10} = 0111|_2$ to $8|_{10} = 1000|_2$ which is why only correcting the LSBs would in fact not work. In contrast, LMCs generalize the idea of this approach and make it applicable to any alphabet which is why they are also called *base codes*. As a result, the code rate of these codes is larger than conventional ECCs, and this is their main advantage [104]. The LMC Encode Algorithm 8.1.1 is executed for the PUF enrollment (cf. Figure 5.2). Its inputs are the symbols Y as result of the equidistant quantization of Chapter 6 of field size q. The resulting outputs are the helper data ${}^{\mathrm{ECC}}W$ and the secret Z which is however not stored as part of the enrollment. The complementary operation that is performed during PUF reconstruction is the LMC Decode Algorithm 8.1.2 that operates on the noisy quantized input symbols $\hat{Y}$ and additionally requires the stored helper data ${}^{\mathrm{ECC}}W$. The result is the corrected output $\hat{Z}$. Please note the comments for each step inside the algorithm listings.

Algorithm 8.1.1: LMC Encode

Data: $Y = [y_1, y_2, ...y_v] \in [0, q-1]$
Result: $Z = [z_1, z_2, ...z_v] \in [0, q-1]$, ${}^{\mathrm{ECC}}W$
/* Step 1: Calculate remainder of $\frac{Y}{q\prime}$ */
1 $\eta = Y \pmod{q\prime}$
/* Step 2: Generate p-ary message symbols using η and encode it using RS(n,t) encoder. */
2 $\eta_p = \mathrm{baseChange}(\eta, q\prime, p)$
3 $C = \mathrm{RS}_{\mathrm{Enc}}(\eta_p, n, t)$
/* Step 3: Convert $2t$ p-ary parity symbols to q-ary */
4 ${}^{\mathrm{ECC}}W = \mathrm{baseChange}(C[n-2t+1:n], p, q)$
/* Step 4: Since this is the enrollment, no error correction is required and the output Z is set to Y */
5 $Z = Y$

The algorithms for encoding and decoding can be used for A-LMC and S-LMC as well, by changing $q\prime$ as in Equation (8.2). If we correct t times a p-ary error, then the maximum number of $q\prime$-ary errors potentially corrected by LMC is given by $t_{\max}$ as defined in Equation 8.3. Since the minimum number of errors corrected is t, we use t as the number of

Algorithm 8.1.2: LMC Decode

Data: $\hat{Y} = [\hat{y}_1, \hat{y}_2, ...\hat{y}_v] \in [0, q-1], {}^{\text{ECC}}W, e \in \{\text{TRUE}, \text{FALSE}\}$
Result: $\hat{Z} = [\hat{z}_1, \hat{z}_2, ...\hat{z}_v] \in [0, q-1]$
/* Step 1: Calculate remainder of $\frac{\hat{Y}}{q\prime}$ */
$\varphi = \hat{Y} \pmod{q\prime}$
/* Step 2: Convert φ and ${}^{\text{ECC}}W$ to p-ary and form a codeword. */
$\varphi_p = \text{baseChange}(\varphi, q\prime, p)$
$P = \text{baseChange}({}^{\text{ECC}}W, q, p)$
$C' = [\, \varphi_p \,||\, P \,]$
/* Step 3: Correct the codeword using RS(n,t) decoder. */
$\hat{C} = \text{RS}_{\text{Dec}}(C', n, t)$
/* Step 4: Convert the message part of $\hat{C}$ to $q\prime$-ary and estimate the error */
$\varphi\prime = \text{baseChange}(\hat{C}[1 : n-2t], p, q\prime)$
$\varepsilon\prime = \varphi - \varphi\prime = [\varepsilon_1\prime, \varepsilon_2\prime ... \varepsilon_v\prime]$
/* Step 5: Refine error to lie in [l_d l_u] bound */
for $i \leftarrow 1$ **to** v **do**
  if $\varepsilon'_i < l_\text{d}$ **then**
    $\varepsilon''_i = \varepsilon'_i + q\prime$
  else if $\varepsilon'(i) > l_\text{u}$ **then**
    $\varepsilon''_i = \varepsilon'_i - q\prime$
  if $\varepsilon''_i \neq 0$ **then**
    count = count + 1 // required only for Early Termination
/* Optional: Early Decoding Termination */
if e == TRUE & count > t **then**
  return
/* Step 6: Subtract ϵ'' from $\hat{Y}$ to get the corrected output */
$\hat{Z} = \hat{Y} - \epsilon''$

Algorithm 8.1.3: LMC baseChange

Data: $D^{\text{In}} = [d_1, d_2, ...d_n]$, baseIn, baseOut
Result: $D^{\text{Out}} = [d_1, d_2, ...d_m]$
1 baseInBits = $\lceil \log_2(\text{baseIn}) \rceil$
2 baseOutBits = $\lceil \log_2(\text{baseOut}) \rceil$
/* Step 1: Represent each array element of D^{In} in binary using dec2bin() */
3 **for** $i \leftarrow 1$ **to** n **do**
4 $\quad D_b[i \cdot \text{baseInBits} : (i+1) \cdot \text{baseInBits}] = \text{dec2bin}(D^{\text{In}}[i], \text{baseInBits})$
/* Step 2: Estimate number of elements in D^{Out} */
5 $m = \lceil n \cdot \text{baseInBits}/\text{baseOutBits} \rceil$
/* Step 3: Combine each baseOutBits elements of D_b to form one symbol using bin2dec() */
6 **for** $i \leftarrow 1$ **to** m **do**
7 $\quad D^{\text{Out}}[i] = \text{bin2dec}(D_b[i \cdot \text{baseOutBits} : (i+1) \cdot \text{baseOutBits}], \text{baseOutBits})$

errors corrected by LMC for notation purposes and also computation of the reliability. However, for *max*-TS, we indeed use $t_{\max}$. This could be even further improved by making use of the early decoding termination, as introduced in the subsequent section.

$$q\prime = \begin{cases} l_{\text{u}} + |l_{\text{d}}| + 1, & \text{B-LMC} \\ 2l_u + 1, & \text{S-LMC} \\ l_{\text{u}} + 1, & \text{A-LMC} \end{cases} \tag{8.2}$$

$$t_{\max} = \frac{t \cdot \log_2(p)}{\log_2(q')} \tag{8.3}$$

8.2 LMC Reliability and Secrecy Leakage

In the following, we briefly discuss additional properties of LMCs.

Early Decoding Termination: We introduce an additional check on the number of non-zero elements in ε'' (cf. Algorithm 8.1.2) to limit the maximum number of $q\prime$-ary errors that get corrected. If the number exceeds the threshold t, then a decoding failure is triggered. (cf. lines 13 – 16 of Algorithm 8.1.2). Once a decoding error occurs, the device enters a permanent failure mode from which recovery is difficult, e.g., by blowing fuses or zeroization of data. This is required to not introduce an obvious timing side-channel in the decoding process and adheres to the principles of tamper-detection and response.

Secrecy Leakage by Helper Data: The leakage caused by LMC helper data ${}^{\text{ECC}}W$ is upper bounded using Equation 8.4, since it is essentially a Code-Offset construction where only the parity is stored. If the block length of the underlying code does not match the block of the message, then z segments are created. Therefore, $P = z \cdot 2\,t \cdot \log_2(p)$ is the total number of parity bits P generated for z segments of LMCs, based on the RS code operating in the p-ary domain.

$$\mathrm{I}(X^v; W) = \lceil P \rceil = \left\lceil z \cdot 2\,t \cdot \log_2(p) \right\rceil \mathrm{bit} \tag{8.4}$$

The leakage calculation for the first entry of Profile 6 in Table 9.1 is provided as an example in the following. First, we compute the secrecy leakage.

$$\mathrm{I}(X^v; W) = \left\lceil z \cdot 2\,t \cdot \log_2(p) \right\rceil = \left\lceil 1 \cdot 2 \cdot 10 \cdot \log_2(64) \right\rceil = 120\,\mathrm{bit}$$

Concerning the *min*-entropy that is extracted on average from a device, we consider each node with parameter $y = 2.1$ for the equidistant quantization which leads to a *min*-entropy $\tilde{\mathrm{H}}_\infty(Y)$ of 3.4325 bit per node, resulting in an overall *min*-entropy for a device $\tilde{\mathrm{H}}_\infty(Y^v)$ with $v = 128$ nodes of

$$\tilde{\mathrm{H}}_\infty(Y^v) = v \cdot \tilde{\mathrm{H}}_\infty(Y) = 3.4325 \cdot 128 = 439.36\,\mathrm{bit}$$

Hence, the effective number of secret bits, i.e., when accounting for the previously computed helper data leakage, is

$$\mathrm{H}_\infty^{\mathrm{eff}} = \tilde{\mathrm{H}}_\infty(Y^v) - \mathrm{I}(Y^v; W) = 439.36 - 120 \approx 319\,\mathrm{bit}$$

Failure Probability: Based on the presented LMC properties, decoding fails if one of the following conditions is met:

1. The magnitude of error ε exceeds $[l_\mathrm{d}\ l_\mathrm{u}]$ of the LMC
2. The number of p-ary errors is greater than t, i.e., too many magnitude errors in total

To provide a generic description of the failure probability, let r parts constitute a symbol (cf. Figure 8.2), i.e., the number of unique digits to represent the symbol (radix). Let $\mathrm{P}_{\mathrm{part}}$ be the error probability of one part and the symbol error probability be $\mathrm{P}_{\mathrm{symb}}$. Then the error probability of a symbol is computed from the error probabilities of its parts as follows:

$$\mathrm{P}_{\mathrm{symb}}(r, \mathrm{P}_{\mathrm{part}}) = \sum_{i=1}^{i=r} \binom{r}{i} \cdot {\mathrm{P}_{\mathrm{part}}}^i \cdot (1 - \mathrm{P}_{\mathrm{part}})^{r-i} \tag{8.5}$$

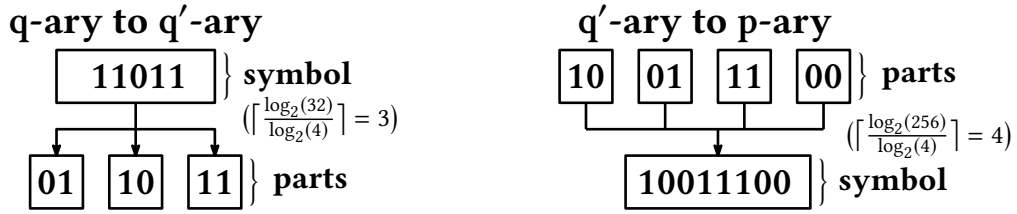
In the opposite direction, i.e., computing error probabilities for a part given a symbol error probability, we use the following equation:

$$\begin{aligned} \mathrm{P}_{\mathrm{symb}} + \binom{r}{0} \cdot {\mathrm{P}_{\mathrm{part}}}^0 \cdot (1 - \mathrm{P}_{\mathrm{part}})^{r-0} &= 1 \\ \mathrm{P}_{\mathrm{symb}} + (1 - \mathrm{P}_{\mathrm{part}})^r &= 1 \\ \implies \mathrm{P}_{\mathrm{part}}(r, \mathrm{P}_{\mathrm{symb}}) &= 1 - (1 - \mathrm{P}_{\mathrm{symb}})^{1/r} \end{aligned} \tag{8.6}$$

For example, if $\mathrm{P}_{\mathrm{part}} = 0.05, r = 4$ then using Equation (8.5) we get $\mathrm{P}_{\mathrm{symb}} = 0.18549$. Similarly for $\mathrm{P}_{\mathrm{symb}} = 0.18549, r = 4$ using Equation (8.6) we get $\mathrm{P}_{\mathrm{part}} \approx 0.05$. If the incorporated ECC corrects up to t errors then the error probability after ECC is given by Equation (8.7). Should LMCs be combined with RS codes, then $\mathrm{P} = \mathrm{P}_{\mathrm{symb}}$. Alternatively, when combined with BCH codes, then $\mathrm{P} = \mathrm{P}_{\mathrm{bit}}$. P_e is the error probability of one segment (block) of RS code.

$$\mathrm{P}_\mathrm{e}(n, t, \mathrm{P}) = \sum_{i=t+1}^{i=n} \binom{n}{i} \cdot \mathrm{P}^i \cdot (1 - \mathrm{P})^{n-i} \tag{8.7}$$

For the error probability calculation of LMC, we assume that after LMC decode, the q-ary symbol error probability ($P_e(Z^v)$) depends only on q'-ary errors. The errors of magnitude $> q'$ are not used in the calculation since this is considered as tampering.



(a) Exemplary q to q'-ary change.

(b) Exemplary q' to p-ary change.

Figure 8.2: Example for the terms *symbol* and *part* when determining error probabilities.

Based on the previous equations, Algorithm 8.2.1 provides the approach on how to compute the error probabilities. Please note that it provides an upper bound for the failure probability for LMC cases where $\log_2(q)/\log_2(q')$ and $\log_2(p)/\log_2(q')$ are not integers. The resulting performance numbers for the considered parameters are presented in Table 9.1, alongside all other profiles. In the following, the notion of tamper-sensitivity is introduced.

Algorithm 8.2.1: LMC Error Probability

Data: $P_e(Y^v), q', q, p, z$
Result: $P_e(Z^v)$

/* Step 1: Calculate q'-ary symbol error probability before RS Decoder using Equation (8.6). */
1 $P_{q'\_symb} = P_{part}(\lceil \log_2(q)/\log_2(q') \rceil, P_e(Y))$

/* Step 2: Calculate p-ary symbol error probability before RS Decoder using Equation (8.5). */
2 $P_{p\_symb} = P_{symb}(\lceil \log_2(p)/\log_2(q') \rceil, P_{q'\_symb})$

/* Step 3: Calculate p-ary block error probability after RS Decoder using Equation (8.7). */
3 $P_{e\_block\_rs} = P_e(n, t, P_{p\_symb})$

/* Step 4: Calculate p-ary symbol error probability after RS Decoder using Equation (8.6). */
4 $P_{p\_symb\_rs} = P_{part}(n, P_{e\_block\_rs})$

/* Step 5: Calculate q'-ary symbol error probability after LMC Decoder using Equation (8.6). */
5 $P_e(Z) = P_{part}(\lceil \log_2(p)/\log_2(q') \rceil, P_{p\_symb\_rs})$

/* Step 6: Calculate q-ary block error probability after LMC Decoder using Equation (8.5). Note, there are $\lceil k \cdot \log_2(p)/\log_2(q') \rceil$ q-ary symbols in one segment of LMC. */
6 $P_e(Z_z) = P_{symb}(\lceil k \cdot \log_2(p)/\log_2(q') \rceil, P_e(Z))$

/* Step 7: Calculate q-ary device error probability after LMC Decoder using Equation (8.5). There are z segments of LMC per device. */
7 $P_e(Z^v) = P_{symb}(z, P_e(Z_z))$

8.3 LMC Examples

For convenience reasons and clarity of the algorithmic descriptions, we provide examples of several LMC calculations for the interested reader. They follow the notation of Algorithm 8.1.1 and Algorithm 8.1.2. Please note that these calculations are based on $q' = 4$, i.e., the base/radix is a power of two, allowing for a very efficient hardware implementation.

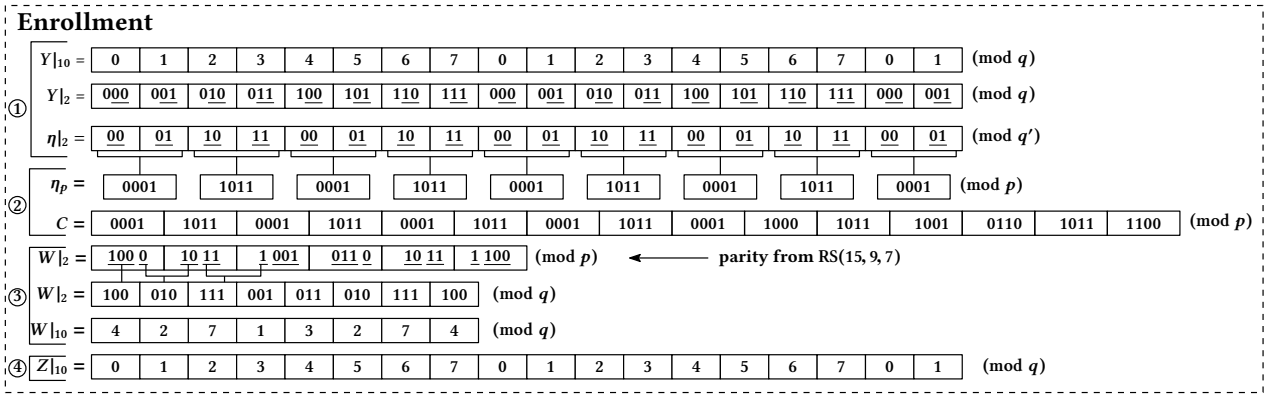


Figure 8.3: LMC encode example (q=8, $q' = 4$, $l_u = 2, l_d = -1$, p=16).

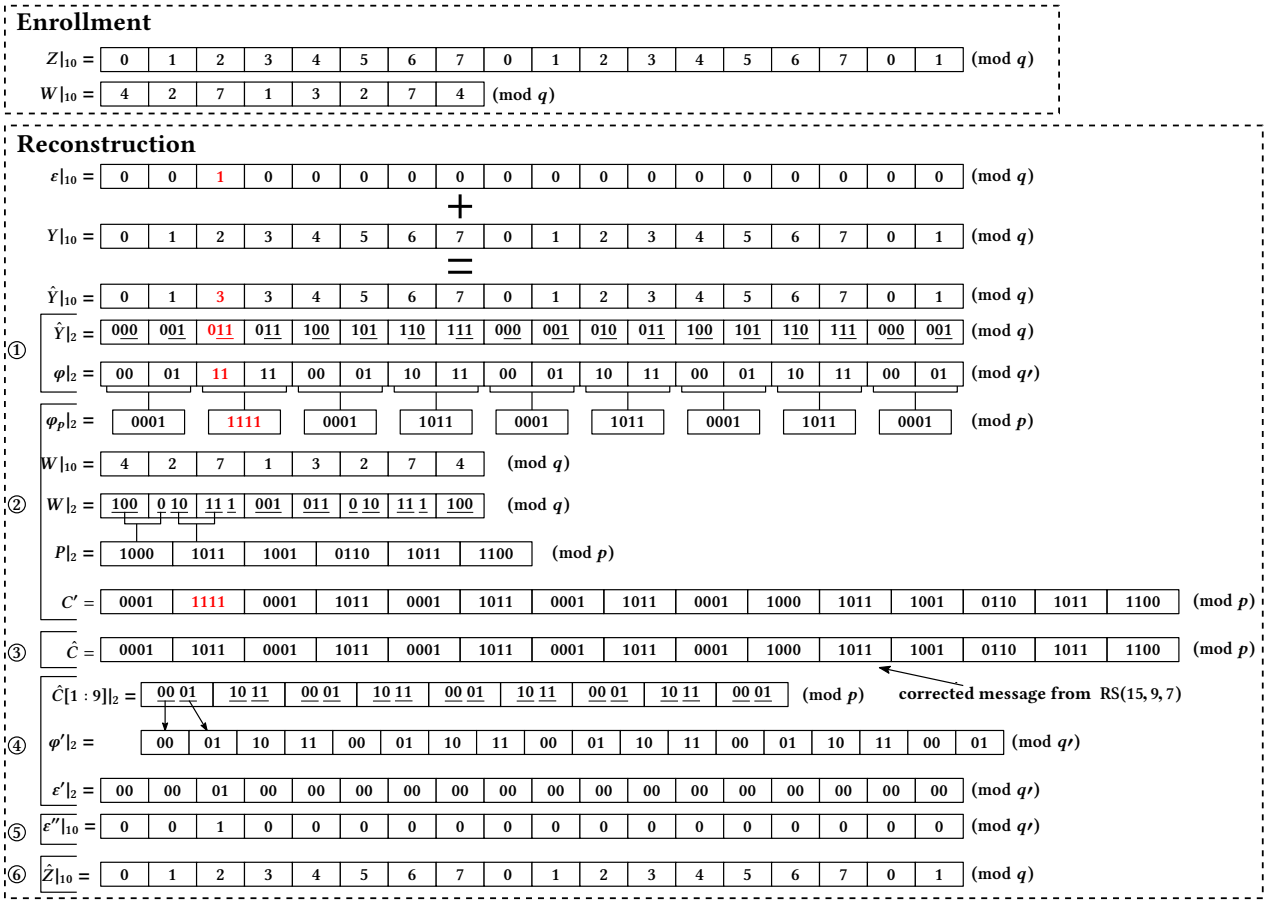


Figure 8.4: LMC example for successful decoding (q=8, $q' = 4$, $l_u = 2, l_d = -1$, p=16).

Figure 8.5: LMC example for decoding failure (q=8, $q' = 4$, $l_u = 2, l_d = -1$, p=16).

Chapter 9

Comparison of ECC Schemes for Higher-Order Alphabet PUFs

This chapter formalizes the aspect of tamper-sensitivity for PUF-based key derivation schemes. The initial idea is rooted in the thesis author's paper on quantization schemes [91] and was later extended when working on LMCs, a paper published in [100] with the thesis author as principal author. All considerations and comparisons still follow the PUF system model presented in Figure 5.2. Furthermore, a thorough evaluation of relevant key derivation schemes is carried out.

Contents

9.1 Tamper-Sensitivity for PUF-based Key Derivation

To further substantiate this topic, let us briefly discuss an introductory example that hints at the strong need to formalize tamper-sensitivity (TS). When comparing Figure 5.3a with Figure 5.3b, then it is striking that the intervals for equidistant quantization are of constant width, whereas the intervals of equiprobable quantization are of unequal width. Consequently, when arbitrarily selecting a value X and subsequently shifting it to the left or right (mimicking an attack), it is easy to see that the magnitude by which we can shift X *without* changing the obtained symbol varies between these two different approaches. Clearly, the permissible magnitude of the shift without causing $Z^v \neq \hat{Z}^v$ reflects the system's (in)capacity to detect adversarial tampering within $\hat{X}$. Therefore, when a system provides good tamper-sensitivity, it is able to detect even the smallest magnitude changes as a result of the tampering ${}^{\mathrm{A}}W$.

Here, we deliberately describe the term tamper-sensitivity informally without making any assumptions on the processing of $\hat{X}$ to include processing variants other than those mentioned in this thesis, such as [190] or [61]. Furthermore, while we are of the opinion that expressing TS in multiples of the noise standard deviation σ_{N} of the underlying measurement circuit is a reasonable choice for the presented work, it may be too limiting for other models or distributions w.r.t. to the noise. Depending on the type of PUF and specifics of the key derivation scheme, TS should be analyzed for a single measured node

as $\mathrm{TS}_{\mathrm{node}}$ (corresponding to one symbol) or for the whole device as $\mathrm{TS}_{\mathrm{device}}$. For detecting tamper attempts, the property of TS appears to be much more important than effective number of secret bits, as later demonstrated. Based on this generic introduction to tamper-sensitivity, we derive two definitions to more precisely capture a system's capability to detect the tampering ${}^{\mathrm{A}}W$.

Definition 9.1.1 (*max*-TS – Maximum Magnitude Tamper Insensitivity) *Defines the* maximum *magnitude of* ${}^{\mathrm{A}}W$ *that goes* undetected, *i.e.,* $\max({}^{\mathrm{A}}W)$ *for which* $Z = \hat{Z}$ *(or* $Z^v = \hat{Z}^v$*) still holds. The corresponding notation for a PUF node and device are* $\mathrm{TS}^{\mathrm{max}}_{\mathrm{node}}$ *and* $\mathrm{TS}^{\mathrm{max}}_{\mathrm{device}}$.

max-TS therefore is a *worst*-case scenario from a defender's point of view. Hence, *max*-TS should be minimized to enable better detection of an attacker regardless of the circumstances, i.e., independent for the probability of occurrence of the affected PUF symbols or specifics of the attack. We note that for TS on a device level, either the accumulated per-node TS is considered, or it is normalized by the number of nodes in that system to support comparisons across devices with different number of nodes, as detailed later. In contrast, we define *min*-TS as follows:

Definition 9.1.2 (*min*-TS – Minimum Magnitude Tamper Sensitivity) *Defines the* minimum *magnitude of* ${}^{\mathrm{A}}W$ *that is* detected, *i.e.,* $\min({}^{\mathrm{A}}W)$ *for which* $Z \neq \hat{Z}$ *(or* $Z^v \neq \hat{Z}^v$*) is achieved. The corresponding notation for a PUF node and device are* $\mathrm{TS}^{\mathrm{min}}_{\mathrm{node}}$ *and* $\mathrm{TS}^{\mathrm{min}}_{\mathrm{device}}$.

It therefore reflects the *best*-case scenario from the defender's point of view to enable earliest detection of an attacker. Within practical limits of applications such as [206, 95], it is evident that a system performs best when *min*-TS equals *max*-TS and approaches the measurement's noise standard deviation σ_{N}, i.e., the smaller the value for TS is, the better is the sensitivity.

These definitions have been formulated such that a hierarchy across different PUF key derivation schemes can be created in a meaningful way, e.g., if *min*-TS(Scheme1) > *max*-TS(Scheme2) is given, then Scheme2 *always* provides a better tamper-sensitivity than Scheme1 and thus, a better detection of attempts to physically tamper with the PUF. Similarly to *min*-entropy as a worst-case scenario for entropy, we are mostly interested in *max*-TS, as it represents the worst-case for the defender.

9.2 Tamper-Sensitivity Equations of Key Derivation Schemes

Let us put the previous definitions to practical use, survey existing schemes, and derive corresponding equations to describe their tamper-sensitivity more analytically. All evaluated schemes have been targeting the scenario of the tamper-evident Coating PUF [206]. However, specific performance numbers will only be shown later in Section 9.4.

In the following, we refer to these schemes as profiles to have a semantic difference between the underlying theoretical scheme and its tested instance based on specific parameters. In total, we selected five profiles, whereas Profile 1,2,3,4 and 6 are based on an equidistant quantization. In case of Profile 1, *only* equidistant quantization is applied *without* subsequent ECC. Profile 2, 3, 4, and 6 then employ an additional ECC after the equidistant quantization. In contrast, Profile 5 is based on an equiprobable quantization and subsequent ECC. These profiles are further detailed hereafter.

TS of Profile 1 based on equidistant quantization *without* ECC (Chapter 6,[91]): As a baseline, we evaluate the performance of a system that only relies on equidistant quantization without any further processing steps as introduced in Chapter 6. Following [91], the equidistant quantization is applied to the PUF outputs X. The width Q_{w} of the evenly sized quantization intervals is determined by

$$Q_{\mathrm{w}} = 2 \cdot y \cdot \sigma_{\mathrm{N}} \tag{9.1}$$

whereas y is a parameter of choice according to the required reliability, i.e., the Confidence Interval (CI) is $[-y \cdot \sigma_{\mathrm{N}}; +y \cdot \sigma_{\mathrm{N}}]$. To obtain m-bit PUF responses, PDF(X) is divided into $L = 2^m$ intervals of the form $(\mu_X + l \cdot Q_{\mathrm{w}}, \mu_X + (l+1) \cdot Q_{\mathrm{w}}]$ where $l = -L/2, \ldots, -1, 0, 1, \ldots, L/2$. Aligning $l = 0$ and μ_X of the Gaussian distribution leads to the highest entropy output while it is slightly decreased by misalignment depending on the choice of y and the relative shift to μ_X. However, due to symmetry reasons of the equidistant quantization this decrease is well-bounded and therefore a robust scheme.

Figure 9.1 illustrates the quantization intervals for $m = 4$ and $L = 16$ and an optimal alignment. Each interval is represented by a symbol Q_l in $[0, L-1]$. As the compensated measurement of the PUF response is non-ideal, i.e., affected by noise of the measurement process, values could move to a different interval compared to the time of enrollment. To counteract this, the offsets between each PUF response X_i and their corresponding interval center are stored as helper data ${}^{\mathrm{Q}}W$. Upon reconstruction, this offset is applied to the noisy value $\hat{X}_i$ to shift it towards its formerly considered interval center, i.e., $(\hat{X}_i - {}^{\mathrm{Q}}W_i \in Q_{l_i} \rightarrow \hat{Y}_i)$ for $i = 1, \ldots, v$.

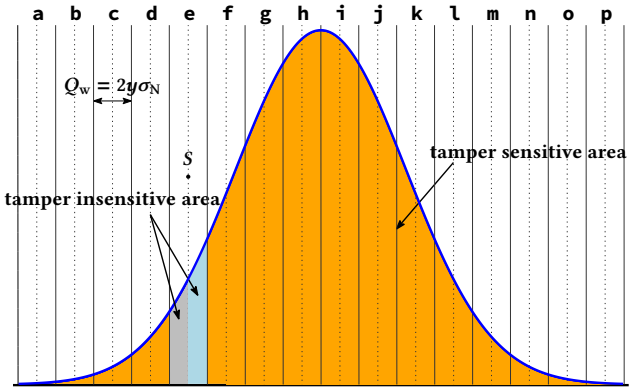


Figure 9.1: $\mathrm{TS}^{\mathrm{max}}_{\mathrm{node}}$ of Profile 1. Any shift outside of the indented quantization interval causes the detection of a tamper attempt which causes the device to fail (as desired).

When assessing this profile with respect to its tamper-sensitivity, it is best to start with $\mathrm{TS}^{\mathrm{max}}_{\mathrm{node}}$ and visualize its properties as done in Figure 9.1. Assuming a symbol S at a specific location of the range of values, it is evident that by exceeding its designated quantization interval limits, an erroneous symbol is obtained. The difference between $\mathrm{TS}^{\mathrm{max}}_{\mathrm{node}}$(P1) and $\mathrm{TS}^{\mathrm{min}}_{\mathrm{node}}$(P1) is therefore only rooted in a small ϵ that represents the smallest possible resolution step of the underlying measurement circuit. For $\mathrm{TS}^{\mathrm{max}}_{\mathrm{device}}$(P1), the accumulated tampering that goes undetected on a device-level is therefore the result of $\mathrm{TS}^{\mathrm{max}}_{\mathrm{node}}$(P1) times the number of nodes v in the system. In contrast, $\mathrm{TS}^{\mathrm{min}}_{\mathrm{device}}$(P1) is limited by $\mathrm{TS}^{\mathrm{min}}_{\mathrm{node}}$(P1), i.e., a single erroneous node allows detection of physical tampering. The resulting equations are therefore:

$$\mathrm{TS}^{\max}_{\mathrm{node}}(P1) = Q_{\mathrm{w}}/2 = y \cdot \sigma_{\mathrm{N}} \qquad \mathrm{TS}^{\max}_{\mathrm{device}}(P1) = v \cdot \mathrm{TS}^{\max}_{\mathrm{node}}(P1) \tag{9.2}$$

$$\mathrm{TS}^{\min}_{\mathrm{node}}(P1) = \mathrm{TS}^{\max}_{\mathrm{node}}(P1) + \epsilon \qquad \mathrm{TS}^{\min}_{\mathrm{device}}(P1) = \mathrm{TS}^{\min}_{\mathrm{node}}(P1) \tag{9.3}$$

TS of Profile 2 based on Fuzzy Commitment and RS codes [106, 92]: Fuzzy commitment is a well-investigated scheme for PUFs and therefore should be considered within the context of this work, too. While the choice of ECC operating on a higher-order alphabet is not limited to RS codes, we chose them to replicate the results of [92]. The basic idea when combining equidistant quantization with an additional ECC is that by making y of Q_{w} smaller, more entropy can be extracted from the PDF which however does not take into account yet the effects of secrecy leakage by the helper data. At the same time when making y smaller, the failure probability increases and must be counteracted by an ECC which is designated to provide a more flexible approach of counteracting errors when compared to a quantization scheme alone.

Here, we make use of a symbol-based RS code with parameters RS(n, t), i.e., n as block length in symbols and t as errors to be corrected. RS codes belong to a class of codes called Linear Block Codes. They are represented as RS(n, k), where k is the number of message symbols and n the block length. A primitive RS Code is defined by a $k \times n$ generator matrix G_{RS} as given in Equation (9.4). RS Codes are Maximum Distance Seperable (MDS), which makes $\mathrm{d}_{\mathrm{H}|S}(\mathrm{RS}(n,k)) = d = n - k + 1$. Hence they can detect and correct up to $d - 1$ errors and $t = \lfloor (d-1)/2 \rfloor$ errors respectively.

$$\mathrm{G}_{\mathrm{RS}} = \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \alpha & \alpha^2 & \cdots & \alpha^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{k-1} & \alpha^{2\cdot(k-1)} & \cdots & \alpha^{(n-1)\cdot(k-1)} \end{pmatrix} \tag{9.4}$$

where $\alpha \in \mathrm{GF}(2^m)$. The ECC input symbols Y are assumed to be of size $q = L$ and their distance is rated by the Hamming distance $\mathrm{d}_{\mathrm{H}|S}$ which states that any substitution error between $\mathrm{d}_{\mathrm{H}|S}(Y^v, \hat{Y}^v)$ and their symbols, *regardless of their actual distance in the underlying domain of* X, is counted as $\mathrm{d}_{\mathrm{H}|S}(Y, \hat{Y}) = 1$. As an example, $(Y, \hat{Y}) = (a, p)$ yields $\mathrm{d}_{\mathrm{H}|S} = 1$ as shown in Figure 9.2.

Hence, the scheme operates independently from the actual binary representation of the symbols similar to Profile 1. Consequently, when considering $\mathrm{TS}^{\max}_{\mathrm{node}}(\mathrm{P2})$, the largest magnitude of ${}^{\mathrm{A}}W$ without causing detection may span from the very left to the very right side of the range of values. This corresponds to $L \cdot Q_{\mathrm{w}}$ for $\mathrm{TS}^{\max}_{\mathrm{node}}(\mathrm{P2})$ and indicates already that the detection of ${}^{\mathrm{A}}W$ is rather limited when compared to Profile 1.

Since the number of nodes v and symbols derived thereof may not necessarily be equal to the ECC's block length n, it must be divided by a number of segments z for separate processing. This is often owed to the fact that codes with substantial block length are often impractical to implement, especially in hardware implementations. The equation describing $\mathrm{TS}^{\max}_{\mathrm{device}}(\mathrm{P2})$ therefore covers the tampering corrected by the code in its first summand and the remaining tampering that goes undetected by the quantization is contained in the second summand. For $\mathrm{TS}^{\min}_{\mathrm{node}}(\mathrm{P2})$, tampering cannot be detected within a single node for as long as the error threshold t has not been exceeded. To properly define $\mathrm{TS}^{\min}_{\mathrm{device}}(\mathrm{P2})$, we therefore take into account the first summand of $\mathrm{TS}^{\max}_{\mathrm{device}}(\mathrm{P2})$ but then only add $\mathrm{TS}^{\min}_{\mathrm{node}}(\mathrm{P1})$

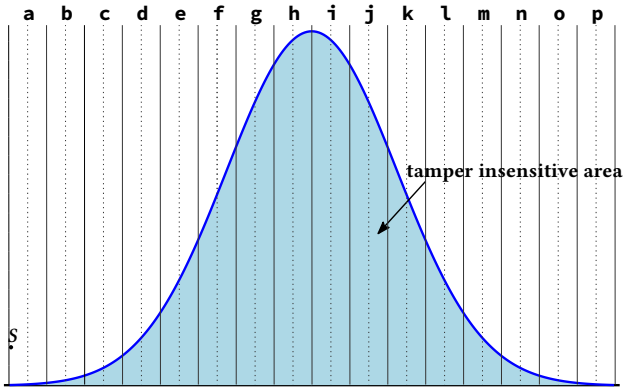


Figure 9.2: $\mathrm{TS}^{\max}_{\mathrm{node}}$ of Profile 2. Based on a single value X of a node, it is not possible to detect tampering, since any magnitude changes result in $\mathrm{d}_{\mathrm{H}|S}(Y, \hat{Y}) = 1$, due to how Hamming distance is defined over symbols.

causing the minimum error to just exceed the scope of the quantization scheme. The resulting equations for TS of P2 are:

$$\mathrm{TS}^{\max}_{\mathrm{node}}(P2) = L\,Q_{\mathrm{w}} \qquad \mathrm{TS}^{\max}_{\mathrm{device}}(P2) = z\,t\,\mathrm{TS}^{\max}_{\mathrm{node}}(P2) + (v - z\,t) \cdot \mathrm{TS}^{\max}_{\mathrm{node}}(P1) \tag{9.5}$$

$$\mathrm{TS}^{\min}_{\mathrm{node}}(P2) = \infty \qquad \mathrm{TS}^{\min}_{\mathrm{device}}(P2) = z\,t\,\mathrm{TS}^{\max}_{\mathrm{node}}(P2) + \mathrm{TS}^{\min}_{\mathrm{node}}(P1) \tag{9.6}$$

TS of Profile 3 based on Code-Offset and BCH codes [37]: Another well-investigated scheme for PUFs is the Code-Offset method. Similar to Profile 2, equidistant quantization is applied. However, this time, the resulting symbols are mapped to bits using a Gray code, i.e., the binary representation of neighboring quantization intervals differs by a hamming distance of 1 only, as it was done also in [206] for equiprobable quantization, as later considered in Profile 5. After this bit mapping to Gray coded symbols, a BCH code is applied. BCH codes can also be described as binary RS codes, i.e., they are represented as $\mathrm{BCH}[n, k, d]_{\mathrm{GF}(2^m)}$. Correspondingly, the distance between codewords is counted by the Hamming distance $\mathrm{d}_{\mathrm{H}|2}$.

The basic idea of this scheme is as follows: Errors close to the designated value result in a small Hamming distance, while a larger shift will increase the Hamming distance. We observe that $L = 2^m$, i.e., m as number of bits to encode the intervals. Since $m < L$, it follows that there exists only one case of the codebook where $\mathrm{d}_{\mathrm{H}|2}$ per node is maximized, i.e., $\mathrm{d}_{\mathrm{H}|2}(Y, \hat{Y}) = m$. This is the case when the all null bit sequence derived from a node is flipped to the all one bit sequence. In all other cases, $\mathrm{d}_{\mathrm{H}|2}(Y, \hat{Y}) \leq m - 1$ which degrades the tamper-sensitivity of the device. Even worse, some very extreme magnitude shifts may result in only $\mathrm{d}_{\mathrm{H}|2}(Y, \hat{Y}) = 1$ due to how a Gray code is constructed. For the example given in Figure 9.3, when assuming a Gray code as follows: $(a \leftarrow 0000)$, $(b \leftarrow 0001)$, $(c \leftarrow 0011)$, $\ldots$, $(p \leftarrow 1000)$, then the largest possible shift while ensuring a Hamming distance of 1 is from the symbol a to the symbol p. Correspondingly, *max*-TS for this profile results in

$$\begin{aligned} \mathrm{TS}^{\max}_{\mathrm{node}}(P3) &= L \cdot Q_{\mathrm{w}} \\ \mathrm{TS}^{\max}_{\mathrm{device}}(P3) &= z\,t\,\mathrm{TS}^{\max}_{\mathrm{node}}(P3) + (v - z\,t) \cdot \mathrm{TS}^{\max}_{\mathrm{node}}(P1) \end{aligned} \tag{9.7}$$

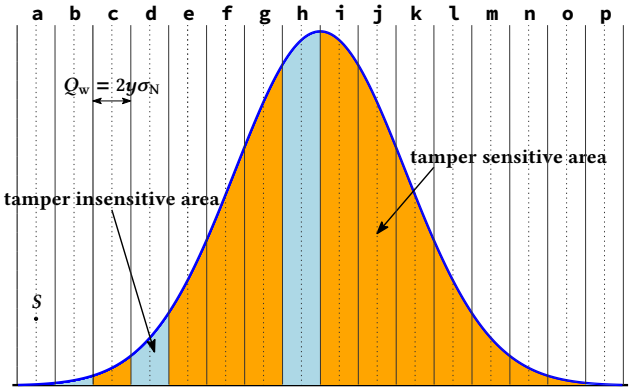


Figure 9.3: TS^{max}_{node} of Profile 3. Please note that for Gray encoded symbols, the resulting distance $d_{H|2}(a, p) = 1$, due to how a Gray code is typically constructed.

To write a closed form of $TS^{min}_{node}(P3)$ and $TS^{min}_{device}(P3)$, we assume that the attacker can divide and distribute ${}^{A}W^{v}$ such that indeed only the smallest detectable change in $d_{H|2}$ per node occurs. For equiprobable quantization this is a symbol residing in any interval with width Q_w and shifting to its directly neighboring intervals, thereby causing a single bit substitution error. When $t > 1$ the ECC is capable of correcting more bits, then multiple nodes with a single bit error within a segment z could be corrected, or larger magnitude shifts within a node (which is not desired with regard to tamper-sensitivity). However, to adhere to the definition of *min*-TS, we assume that for larger t, indeed t-times the smallest detectable change occured. While this is unlikely to reflect a real-world scenario, this assumption is useful to assess the conceptual tamper-sensitivity of the scheme. The resulting equations are therefore:

$$\begin{aligned} TS^{min}_{node}(P3) &= 3 \cdot Q_w/2 + \epsilon \qquad \text{iff } t = 1 \\ TS^{min}_{device}(P3) &= z\, t\, TS^{min}_{node}(P3) + TS^{min}_{node}(P1) \end{aligned} \tag{9.8}$$

TS of Profile 4 based on VT-like codes (Chapter 7,[92]): This profile again is based on an equidistant quantization but this time with a variable-length mapping of the symbols Y to bits, as described in Chapter 7. The corresponding code is a VT-like code denoted as $VT(\cdot, t)$ with t as number of errors in d_{Lev}. Due to the limitations of VT-codes, $t = 1$ always, as multiple insertion/deletion errors can only be corrected when considering multiple segments z.

We briefly recall basic properties of the VT-like codes. They are founded on the *Levenstein Distance* metric d_{Lev}. Each derived symbol Y corresponding to a quantization interval is bit mapped to a variable number of bits. Since these bit maps should be uniquely decodable, they are generated using a binary tree, resulting in a prefix-free code, while ensuring $d_{Lev} = 1$ between neighboring intervals. For a PUF device with v nodes, z number of VT-like code segments can be generated. Since each segment can correct only 1 symbol error, the total number of correctable symbol errors is z. The systematic code construction is described in Equation (9.9) following the notation of Section 7.4.

$$C_{VT} := \left\{ (b_1, b_2, ..., b_m, p_1, ..., p_r) : \sum_{i=1}^{m} i \cdot b_i + \sum_{j=1}^{r} 2^{j-1} \cdot p_j \equiv 0 \pmod{2m+1} \right\} \tag{9.9}$$

$$\mathrm{P} = 2m + 1 - \left(\sum_{i=1}^{m} i \cdot b_{\mathrm{i}} \pmod{2m+1} \right) \tag{9.10}$$

where $(b_1, b_2, \ldots, b_{\mathrm{m}})$ is the bit map of $(y_1, y_2, \ldots, y_{\mathrm{v/z}})$ PUF nodes and $(p_1, \ldots, p_{\mathrm{r}})$ is the binary representation of P given by Equation (9.10).

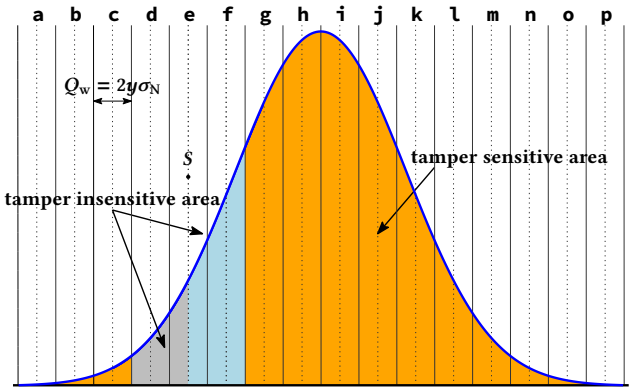


Figure 9.4: $\mathrm{TS}^{\mathrm{min}}_{\mathrm{node}}$ of Profile 4.

When analyzing the tamper-sensitivity of this profile, it is evident that writing a closed form for $\mathrm{TS}^{\mathrm{max}}_{\mathrm{node}}(\mathrm{P4})$ and $\mathrm{TS}^{\mathrm{max}}_{\mathrm{device}}(\mathrm{P4})$ is difficult, as it depends on the number of quantization intervals and the codebook used to create the variable-length bit mapping*. This statement is based on the observation that $\mathrm{d}_{\mathrm{Lev}}(Y, \hat{Y}) = 1$ is ensured for directly neighboring intervals but larger magnitude changes may still result in distance $\mathrm{d}_{\mathrm{Lev}} = 1$, i.e., the attacker may be even encouraged to cause larger magnitude changes that would still be accounted for by the error-correcting capability of the code. We therefore directly compute *max*-TS values for the parameters later considered. In contrast, stating *min*-TS equations is straightforward and visualized in Figure 9.4. The corresponding equations are

$$\mathrm{TS}^{\mathrm{min}}_{\mathrm{node}}(P4) = 3\,Q_{\mathrm{w}}/2 + \epsilon \qquad \mathrm{TS}^{\mathrm{min}}_{\mathrm{device}}(P4) = \mathrm{TS}^{\mathrm{min}}_{\mathrm{node}}(P4) \tag{9.11}$$

owed to the fact that the minimum error to detect is the one just exceeding the error-correcting capability of the VT-like code. Similarly to Profile 1, the overall $\mathrm{TS}^{\mathrm{min}}_{\mathrm{device}}(\mathrm{P4})$ is again the same as $\mathrm{TS}^{\mathrm{min}}_{\mathrm{node}}(\mathrm{P4})$, i.e., a single erroneous node triggers the tamper-detection which is a beneficial behavior for improved tamper-sensitivity.

TS of Profile 5 based on Equiprobable Quantization and BCH-based Code-Offset [206]: Unlike before, we make use of an equiprobable quantization and refer to Section 6.3 for its formal description. As illustrated in Figure 5.3b, this approach is characterized by its innermost intervals of width Q_{min} and outermost intervals of width Q_{max}. As described in Section 5.3, the symbols are mapped to a binary representation using a Gray code. A BCH(n, t) code is applied to the resulting output, whereas both n and t are in bits.

One of the challenges in this profile are defining the outermost intervals properly when considering a practical implementation, as equal proability of intervals needs to be ensured

* For the specific case later considered: $\mathrm{TS}^{\mathrm{max}}_{\mathrm{node}}(P4) = 6 \cdot Q_{\mathrm{w}} + Q_{\mathrm{w}}/2$ for 12 intervals; $\mathrm{TS}^{\mathrm{max}}_{\mathrm{node}}(P4) = 10 \cdot Q_{\mathrm{w}} + Q_{\mathrm{w}}/2$ for 14 intervals; and on a device-level: $\mathrm{TS}^{\mathrm{max}}_{\mathrm{device}}(P4) = z\,t \cdot \mathrm{TS}^{\mathrm{max}}_{\mathrm{node}}(P4) + (v - z\,t) \cdot \mathrm{TS}^{\mathrm{max}}_{\mathrm{node}}(P1)$

also for the outermost intervals that are however limited by the measurement range of the underlying implementation. Hence, at some point in the range of X, the tails of the PDF need to be cut off. To balance the properties of this profile and provide a fair comparison, we chose to neglect the part of the tails when the probability of occurrence drops below 0.1%. The same has been done in [91] or [231].

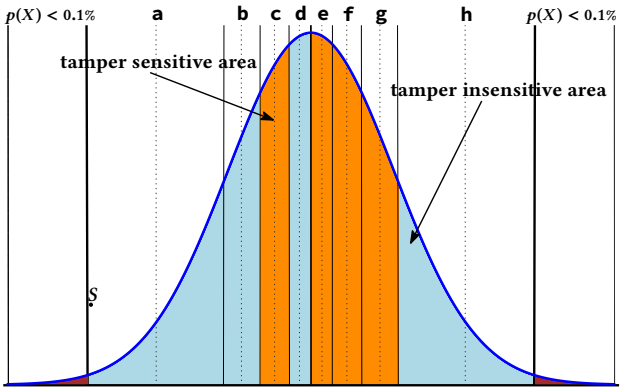


Figure 9.5: $\mathrm{TS}^{\max}_{\mathrm{node}}$ of Profile 5 for the symbol S as indicated. Based on the Gray code bit mapping as illustrated in Figure 5.3b.

Regarding the tamper-sensitivity of this profile, we observe similarly to P3 that the specifics of the Gray code significantly affect the tamper-sensitivity. For example, for the scenario presented in [206], a *shift* from the left outermost interval to the right would only result in distance 1, as illustrated in Figure 9.5 when assuming the bit mapping of Figure 5.3b. *max*-TS for this profile therefore results in

$$\begin{aligned} \mathrm{TS}^{\max}_{\mathrm{node}}(P5) &= \sum_{i=1}^{L} \mathrm{width}(Q_i) \\ \mathrm{TS}^{\max}_{\mathrm{device}}(P5) &= z\,t\,\mathrm{TS}^{\max}_{\mathrm{node}}(P5) + (v - z\,t)\cdot Q_{\max}/2 \end{aligned} \tag{9.12}$$

To write a closed form of $\mathrm{TS}^{\min}_{\mathrm{node}}$(P5) and $\mathrm{TS}^{\min}_{\mathrm{device}}$(P5), we again assume that the attacker can divide and distribute ${}^{\mathrm{A}}W$ such that indeed only the smallest detectable change in $\mathrm{d}_{\mathrm{H}|2}$ per node occurs. For equiprobable quantization this is a symbol residing in $Q_{\min}$ and shifting to its neighboring intervals. When $t > 1$, then multiple nodes could be corrected or larger magnitude shifts within a node. To adhere to the definition of *min*-TS, we again assume that for larger t, indeed t-times the smallest detectable change occured. The resulting equations are therefore:

$$\begin{aligned} \mathrm{TS}^{\min}_{\mathrm{node}}(P5) &= 3\cdot Q_{\min}/2 + \epsilon \qquad \text{iff } t = 1 \\ \mathrm{TS}^{\min}_{\mathrm{device}}(P5) &= z\,t\,\mathrm{TS}^{\min}_{\mathrm{node}}(P5) + Q_{\min}/2 + \epsilon \end{aligned} \tag{9.13}$$

Regarding the fairness of comparison and the design trade-off made w.r.t. $Q_{\max}$, i.e., where to cut off the range of values, we point out that by defining *min*-TS as given, it is independent from the size of the outermost interval. Hence, it only affects *max*-TS whereas excluding more values would make $Q_{\max}$ smaller but increase excess during the manufacturing process, thereby reducing yield.

TS of Profile 6 based on Equidistant Quantization and LMC (Chapter 8,[100]): Following the descriptions of Chapter 8, LMCs correct t errors within the $[l_d \quad l_u]$ boundary. Hence $\mathrm{TS}^{\max}_{\mathrm{node}}(\mathrm{P6})$ is defined using Equation 9.14. Its first summand is based on the error correction capability of the LMC and the second summand caused by the equidistant quantization. Hence, to cause detection, an additional ϵ is required for $\mathrm{TS}^{\min}_{\mathrm{node}}(\mathrm{P6})$. Since LMC decoding fails even if the number of errors is less than t but the magnitude exceeds $[l_d \quad l_u]$, $\mathrm{TS}^{\min}_{\mathrm{device}}(\mathrm{P6})$ is equivalent $\mathrm{TS}^{\min}_{\mathrm{node}}(\mathrm{P6})$. This already indicates a significant advantage over the other profiles discussed earlier. Calculating $\mathrm{TS}^{\max}_{\mathrm{device}}(\mathrm{P6})$ then follows similar principles of the other ECC-based profiles, i.e., if the block length n does not match the input length of symbols v, then multiple segments z must be created.

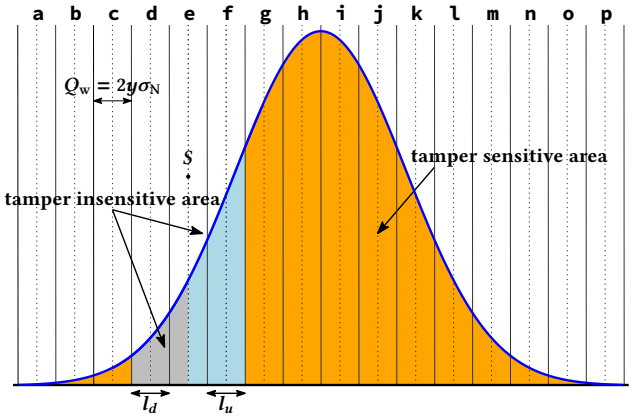


Figure 9.6: $\mathrm{TS}^{\max}_{\mathrm{node}}$ of Profile 6. Please note the difference to Figure 9.4 where $\mathrm{TS}^{\min}_{\mathrm{node}}(\mathrm{P4})$ is illustrated, i.e., TS-*max* vs. TS-*min*. In this figure, neighboring intervals of magnitude l_u and l_d are corrected.

$$\mathrm{TS}^{\max}_{\mathrm{node}}(P6) = \max(l_{\mathrm{u}}, |l_{\mathrm{d}}|) \cdot Q_{\mathrm{w}} + \mathrm{TS}^{\max}_{\mathrm{node}}(P1) \tag{9.14}$$

$$\mathrm{TS}^{\max}_{\mathrm{device}}(P6) = z\,t \cdot \mathrm{TS}^{\max}_{\mathrm{node}}(P6) + (v - z\,t) \cdot \mathrm{TS}^{\max}_{\mathrm{node}}(P1) \tag{9.15}$$

$$\mathrm{TS}^{\min}_{\mathrm{node}}(P6) = \min(l_{\mathrm{u}}, |l_{\mathrm{d}}|) \cdot Q_{\mathrm{w}} + \mathrm{TS}^{\min}_{\mathrm{node}}(P1) \tag{9.16}$$

$$\mathrm{TS}^{\min}_{\mathrm{device}}(P6) = \mathrm{TS}^{\min}_{\mathrm{node}}(P6) \tag{9.17}$$

9.3 Discussion of Tamper-Sensitivity

All presented TS equations have in common that they describe a noise-free scenario for analysis purposes only, as motivated beforehand. This simplifies the equations without affecting their accuracy in describing the fundamental TS property of the scheme. Moreover, we neglect the challenges that arise when trying to define TS for the outermost intervals of a specific profile, i.e., independent of the actual measurement range of the PDF that could be covered and the number of quantization intervals to sample it. We assume that TS is not affected by these practical constraints and instead is purely based on the properties of the underlying scheme.

We point out that our definition of TS assumes unidirectional *shifts*, i.e., a change in value cannot be in both directions at the same time. This is of particular relevance for Profile 3, 4, and 5, where a shift may move values over intervals that are considered tamper-sensitive. Hence, *not* the tamper-sensitive area of a PDF is taken into account but indeed the magnitude of the shifts only. Since Profile 5 deviates already at the point of quantization from the other profiles, neither $\mathrm{TS}^{\mathrm{min}}_{\mathrm{node}}$ nor $\mathrm{TS}^{\mathrm{max}}_{\mathrm{node}}$ will reflect the perceived tamper-sensitivity in a practical setting as it will be based on the *average* tamper-sensitivity that takes into account the probability of occurrence of an affected quantization interval, i.e., it would be necessary to weigh the tamper-sensitivity per interval by its probability of occurrence. However, *min*-TS and *max*-TS already provide a quality assessment to sufficiently compare Profile 5 against the other profiles.

As can be derived from the given equations, all schemes behave differently when considering $\mathrm{TS}^{\mathrm{max}}_{\mathrm{node}}$ and $\mathrm{TS}^{\mathrm{max}}_{\mathrm{device}}$. This already supports the argument that a property is being addressed that otherwise cannot be captured by entropy or failure rate. Please note that while some of the given equations appear highly similar, e.g., $\mathrm{TS}^{\mathrm{min}}_{\mathrm{node}}$ of Profiles 1, 3, 4, and 5, their actual value will still be different when considered under a specific set of parameters. The interested reader may already proceed to Table 9.2 to see the resulting numbers for the tested profiles. Correspondingly will the visual appearance of the presented Figures for the actual parameters be different, e.g., smaller but more intervals.

As can be derived from the equations also, Profile 6 enables $\mathrm{TS}^{\mathrm{min}}_{\mathrm{node}}$ to be equal to $\mathrm{TS}^{\mathrm{min}}_{\mathrm{device}}$, as it is the case for Profile 1. However, since LMCs are used, this allows to be almost twice as tamper-sensitive on a device-level in addition to extracting more entropy, as detailed in the following evaluation.

Late Tamper Evaluation. This work focuses on improving the combination of quantization and ECC *without* additional processing steps. Alternatively, it may be possible for some profiles to further improve tamper detection by studying the magnitude of errors *after successful decoding* was done, i.e., by computing the Eucledian distance $\mathrm{d_E}(\hat{Z}, \hat{X})$ and validating that the result is of a reasonable magnitude, e.g., by requiring $\mathrm{d_E}(\hat{Z}, \hat{X}) \leq \mathrm{TDT}$, e.g., $\mathrm{TDT} = 3 \cdot Q_{\mathrm{w}}/2 = 3 \cdot y \cdot \sigma_{\mathrm{N}}$ for Profile 4. By following this approach for Profile 4, it is possible to limit the error magnitude to $\mathrm{TS}^{\mathrm{min}}_{\mathrm{node}}$(Profile 4). This is possible since only one error per segment z is covered by the scheme. For other Profiles though, such as Profile 3 and Profile 5, this quickly leads to inconsistencies in how errors are treated. This argument is based on the observation that within a block of length n (in bits), up to t errors (in bits) are corrected. Assuming that m bits per node are derived and $t > m$ (which is the case for the practical scenarios considered), then it is becoming increasingly difficult to formulate a valid late tamper evaluation approach, since the late tamper evaluation will impede with how the ECC operates. Hence, even if such an approach can be successfully applied to any of the existing profiles, the obtained result will still not exceed the *min*-TS level due to how it has been defined. This is in addition to the potential security threat of first reconstructing the valid secret, before discarding it based on the result of the late tamper evaluation. To the best of the author's knowledge, there are no publications discussing the specifics of such a late tamper evaluation with regard to tamper detection.

9.4 Evaluation of Key Derivation Profiles

In this section we discuss the results listed in Table 9.1 and Table 9.2. All former profiles have been tested based on the empirical data of [206]. The corresponding parameters are:

$\mu_X = 1.8 \cdot 10^{-13}$ and $\sigma_X = 3.6 \cdot 10^{-15}$. Individual measurements of the nodes are affected by Gaussian distributed, mean-free noise with $\sigma_N = 2 \cdot 10^{-16}$.

Starting with Profile 1 in Table 9.1, it can be seen that even a basic equidistant quantization scheme *without* subsequent ECC is sufficient to create a workable solution. This is achieved by values y of 5.4 or larger, i.e., the width of the quantization intervals needs to be relatively large to account for the assumed noise. We note that the extracted *min*-entropy is only determined by the innermost intervals closest to μ_X, i.e., an increasing number of quantization intervals does *not* increase the *min*-entropy. The extracted entropy ranges between 267 bits and 231 bits for a reliability in the range of 10^{-6} to 10^{-9}. As described by Equation 9.2, $\mathrm{TS}^{\max}_{\mathrm{node}}(P1)$ is equivalent to $y \cdot \sigma_N$, whereas $\mathrm{TS}^{\max}_{\mathrm{device}}(P1)$ simply scales this number by the number of nodes v in the system. The corresponding numbers for Profile 1 with the best *max*-TS are therefore 5.4 on a node-level and 692 on a device-level. If we would be considering an increasing number of nodes beyond $v = 128$, it is clear that the increasing numbers of nodes in the exponent of the error probability computation demand an over-excessively wide quantization interval to be counteracted. Hence, this cannot be considered a flexible engineering solution and should only be considered as a baseline for subsequent comparisons. For all subsequent profiles, we investigate whether a smaller y with an additional ECC can perform better than this.

In Profile 2, a fuzzy commitment based on RS codes is used. While y can be lowered to 2.3 resulting in much smaller and more intervals, the helper data leakage caused by the ECC completely counteracts the gain in *min*-entropy such that the effective entropy $\mathrm{H}^{\mathrm{eff}}_{\infty}$ (accounting for the leakage) extracted from the PUF is less than that of Profile 1. In general, this scheme can be adapted easily to different requirements by adjusting t. However, as the distance metric is based on $\mathrm{d}_{\mathrm{H}|S}$, tamper-sensitivity is relatively poor as supported by the obtained results. For both *min*-TS and *max*-TS, the results are actually much worse when compared to a scheme based on equidistant quantization only.

With the help of Profile 3, entropy levels reach a similar amount when compared to Profile 1. This is owed to the differences in the underlying Code-Offset construction when compared to the Fuzzy Commitment scheme, as the leakage is upper bounded by the parity, resulting in a reduced leakage when compared to Profile 2. However, extracting more entropy is at the cost of losing tamper-sensitivity. Moreover, $\mathrm{TS}^{\min}_{\mathrm{node}}$ is *only* defined for $t = 1$ and therefore represents a strong assumption regarding the attacker as in a practical scenario, the attacker would not be able to divide and distribute the resulting errors to keep them small. Hence, even while the numbers for $\mathrm{TS}^{\min}_{\mathrm{node}}$ indicate a tamper-sensitivity performance close to Profile 1, it cannot be considered a feasible alternative.

For Profile 4, VT-like codes were used with variable-length bit mapping of the symbols. Due to the limitations of these codes, t cannot be chosen arbitrarily and is limited to 1. Consequently, it is not surprising that y cannot be made smaller than 4.24 to still obtain a reliable device. In contrast to Profile 2, a similar *max*-TS is obtained when compared to Profile 1, while performing worse on a node-level. The extracted entropy is marginally better than Profile 1 but we are of the opinion that the added complexity of carrying out the computation for an ECC does not justify this gain.

In contrast to all previous profiles, we applied an equiprobable quantization in Profile 5 which cannot be used as a standalone solution under the given simulation parameters. The given y of 2.87 in the table applies to $Q_{\min}$ only. All other intervals towards $Q_{\max}$ are therefore significantly larger. To provide a fair comparison, we chose to exclude values of X with probability of occurrence less than 0.1%, otherwise, tamper-sensitivity in the

Table 9.1: Comparison of key derivation schemes for higher-order alphabet PUFs. Profile settings are shared among publications [92, 91, 206] and as follows: $\mu_X = 1.8 \cdot 10^{-13}$ and $\sigma_X = 3.6 \cdot 10^{-15}$. Individual measurements of the nodes are affected by Gaussian distributed, mean-free noise with $\sigma_\mathrm{N} = 2 \cdot 10^{-16}$.

Profile [a]	y	L	z	ECC(n,t)	$\mathrm{P_e}(Y)$ (before ECC)	$\mathrm{P_e}(Y^{\boldsymbol{v}})$ (before ECC)	$\mathrm{P_e}(Z^{\boldsymbol{v}})$ (after ECC)	$\mathrm{H}_\infty^\mathrm{eff}$ [bit]	$\mathrm{TS}_\mathrm{node}^\mathrm{max}$ [σ_N]	$\mathrm{TS}_\mathrm{device}^\mathrm{max}$ [σ_N]	Distance Metric
P1 [b]	5.4	8	128	–	6.7×10^{-8}	8.5×10^{-6}	(*id.*)	267	5.4	692	none
	6.6	16	128	–	4.1×10^{-11}	5.3×10^{-9}	(*id.*)	231	6.6	845	
P2 [c]	2.3	32	4	RS(31, 7)	1.2×10^{-2}	7.9×10^{-1}	6.1×10^{-8}	122	148	4352	$\mathrm{d_{H\|}}_S$
	3	32	4	RS(31, 4)	2.7×10^{-3}	2.9×10^{-1}	3.4×10^{-7}	193	192	3408	
	5	16	8	RS(15, 1)	5.7×10^{-7}	7.3×10^{-5}	4.8×10^{-10}	185	160	1880	
P3 [d]	2.3	32	4	BCH(255, 8)	2.1×10^{-2}	9.4×10^{-1}	8.9×10^{-6}	166	148	4932	$\mathrm{d_{H\|2}}$
	2.7	32	7	BCH(127, 4)	6.9×10^{-3}	5.9×10^{-1}	1.1×10^{-6}	197	173	5109	
	3.6	16	5	BCH(127, 2)	3.1×10^{-4}	4.0×10^{-2}	1.7×10^{-7}	265	116	1577	
P4 [e]	4.95	12	1	VT$(\cdot, 1)$	7.4×10^{-7}	9.5×10^{-5}	4.5×10^{-9}	276	65	693	$\mathrm{d_{Lev}}$
	4.24	14	4	VT$(\cdot, 1)$	2.2×10^{-5}	2.8×10^{-3}	1.0×10^{-6}	271	90	828	
P5 [f]	2.87	8	2	BCH(255, 7)	1.3×10^{-3}	1.6×10^{-1}	1.2×10^{-12}	272	112	3558	$\mathrm{d_{H\|2}}$
			2	BCH(255, 4)			2.8×10^{-7}	320		2994	
P6 [g]	2.1	64	1	LMC(63, 10)	3.6×10^{-2}	9.9×10^{-1}	9.1×10^{-6}	319	6.3	395	$\mathrm{d_{Man}}$
	2.3	32	1	LMC(63, 9)	2.1×10^{-2}	9.4×10^{-1}	3.3×10^{-6}	314	6.9	419	
	2.7	32	1	LMC(63, 10)	6.9×10^{-3}	5.9×10^{-1}	3.7×10^{-12}	273	8.1	508	
	2.7	16	1	LMC(63, 6)	6.9×10^{-3}	5.9×10^{-1}	3.5×10^{-6}	321	8.1	443	

[a] Neglecting leakage from quantization helper data ${}^{\mathrm{Q}}W$ for computation of $\mathrm{H}_\infty^\mathrm{eff}$, i.e., only leakage by ECC helper data ${}^{\mathrm{ECC}}W$ is considered.

[b] **Profile 1 (P1):** Equidistant quantization *without* ECC (independent of symbol's bit mapping)

[c] **Profile 2 (P2):** Equidistant quantization and RS-based Fuzzy Commitment scheme (independent of symbol's bit mapping, n in symbols, t in $\mathrm{d_{H|}}_S$)

[d] **Profile 3 (P3):** Equidistant quantization and BCH-based Code-Offset scheme (n in bits, t in $\mathrm{d_{H|2}}$)

[e] **Profile 4 (P4):** Equidistant quantization, variable-length bit mapping of symbols, VT-like codes (t in $\mathrm{d_{Lev}}$)

[f] **Profile 5 (P5):** Equiprobable quantization, Gray code bit mapping of symbols, BCH-based Code-Offset scheme (n in bits, t in $\mathrm{d_{H|2}}$)

[g] **Profile 6 (P6):** Equidistant quantization, LMC ($l_\mathrm{u} = 1, l_\mathrm{d} = -1$) with concatenated RS code (n in symbols, t in $\mathrm{d_{Man}}$)

Table 9.2: This table complements the tamper-sensitivity results of Table 9.1 regarding *min*-TS and also provides the numbers for *max*-TS normalized by the number of nodes v (last column) with $v = 128$, therefore representing the on-average per-node sensitivity. These numbers enable a comparison across different tamper-evident PUF system designs with varying number of PUF nodes v.

Profile	y	L	z	ECC(n,t)	$P_e(Z^v)$	H_∞^{eff} [bit]	TS_{node}^{min} [σ_N]	TS_{node}^{max} [σ_N]	TS_{device}^{min} [σ_N]	TS_{device}^{max} [σ_N]	TS_{device}^{max}/v [σ_N]
P1	5.4	8	128	–	8.5×10^{-6}	267	5.4	5.4	5.4	692	5.4
	6.6	16		–	5.3×10^{-9}	231	6.6	6.6	6.6	845	6.6
P2	2.3	32	4	RS(31, 7)	6.1×10^{-8}	122	∞	148	4124	4352	34
	3	32	4	RS(31, 4)	3.4×10^{-7}	193	∞	192	3075	3408	27
	5	16	8	RS(15, 1)	4.8×10^{-10}	185	∞	160	1285	1880	15
P3	2.3	32	4	BCH(255, 8)	8.9×10^{-6}	166	6.9	148	224	4932	39
	2.7	32	7	BCH(127, 4)	1.1×10^{-6}	197	8.1	173	230	5109	40
	3.6	16	5	BCH(127, 2)	1.7×10^{-7}	265	10.8	116	112	1577	13
P4	4.95	12	1	VT(·, 1)	4.5×10^{-9}	276	15	65	15	693	5.4
	4.24	14	4	VT(·, 1)	1.0×10^{-6}	271	13	90	13	882	6.9
P5	2.87	8	2	BCH(255, 7)	1.2×10^{-12}	272	8.7	112	141	3558	30
				BCH(255, 5)	2.8×10^{-7}	320			72	2994	24
P6	2.1	64	1	LMC(63, 10)	9.1×10^{-6}	319	6.3	6.3	6.3	395	3.1
	2.3	32	1	LMC(63, 9)	3.3×10^{-6}	314	6.9	6.9	6.9	419	3.3
	2.7	32	1	LMC(63, 10)	3.7×10^{-12}	273	8.1	8.1	8.1	508	4.0
	2.7	16	1	LMC(63, 6)	3.5×10^{-6}	321	8.1	8.1	8.1	443	3.5

outermost intervals would not be bounded which would exaggerate the numbers for *max*-TS unnecessarily. *When neglecting the significant quantization helper data leakage by* ${}^{\mathrm{Q}}W$, the effective entropy after accounting for the ECC helper data is quite significant, as the equiprobable quantization extracts 3 bits of full entropy per node under this simulated scenario. Regarding tamper-sensitivity, interesting properties are observed. Since the innermost intervals of $Q_{\min}$ are relatively small, the earliest possible detection which translates to *min*-TS on a node-level, is almost within the range of Profile 1. However, most errors that occur are also at or within the range of the innermost intervals. As a result, t must be chosen sufficiently large to account for these errors. This already leads to a suboptimal $\mathrm{TS}^{\min}_{\mathrm{device}}$ behavior. When further analyzing $\mathrm{TS}^{\max}_{\mathrm{node}}$ and $\mathrm{TS}^{\max}_{\mathrm{device}}$, then the obtained tamper-sensitivity performance is clearly worse when compared to Profile 1 and sometimes equally poor when compared to Profile 2 or Profile 4.

Let us now consider our proposal based on equidistant quantization and LMC under the name Profile 6. It can be seen right away that y is the smallest for all considered profiles. For equidistant quantization, this leads to the best-case in terms of entropy that can be extracted from the PUF PDF. Since the equidistant quantization is quite effective in removing a significant portion of the noise influence, only a fraction of nodes need further correction by the LMC. Mainly due to the transformation of q' to p, the overall construction is more efficient when compared to, e.g., Profile 3. This results in a total of ~ 320 effective number of secret bits, the maximum of all previously considered profiles. In addition to that, it can be seen in Table 9.1 that the per-node *max*-TS is similar to Profile 1 while drastically outperforming all other Profiles. However, the most important result is that *max*-TS on a device level is almost only half of Profile 1. When normalizing $\mathrm{TS}^{\max}_{\mathrm{device}}$(P6) by the number of nodes as done in Table 9.2, i.e., $395/v = 3.1\,[\sigma_{\mathrm{N}}]$, then this can be interpreted as the on-average tamper detection threshold per node, $\mathrm{TDT} = 3.1 \cdot \sigma_{\mathrm{N}}$. This is a significant gain in terms of tamper-sensitivity *and* effective number of bits, for various different levels of reliability and alphabet sizes/number of quantization intervals. Taking into account that $\mathrm{TS}^{\min}_{\mathrm{node}}$(P6) is equal to $\mathrm{TS}^{\max}_{\mathrm{node}}$(P6) and $\mathrm{TS}^{\min}_{\mathrm{device}}$(P6) is bounded by the *min*-TS per node of Profile 6 (cf. Table 9.2), it is evident that the general behavior of LMC mimics the behavior of Profile 1 with regard to the detection of tampering, while performing more effectively which allows to choose a smaller y, resulting in a better entropy and tamper-sensitivity. Overall, this clearly demonstrates the superiority of this scheme and optimized detection of tampering.

Chapter 10

Conclusions on Reliability Enhancement Techniques for PUFs

This chapter briefly wraps up the work towards reliability enhancement techniques considered in this thesis. Moreover, an outlook is presented to indicate future work.

Contents

10.1 Summary on Reliability Enhancement Techniques

This part of the thesis considered different techniques for reliability enhancement of PUFs, in particular quantization schemes and ECCs operating on a higher-order alphabet. One of the results is that equidistant quantization is likely to be the most desirable type of quantization within the context of tamper-sensitivity. Beyond that, it is a robust scheme that can be incorporated into systems with little to no prior knowledge of the PDF, unlike equiprobable quantization where exact knowledge of the PDF is a necessity. Moreover, shifts in the PDF, e.g., due to manufacturing degradation or insufficient control over some of the manufacturing parameters as generally assumed for PUFs, may completely eradicate equiprobability of bits obtained from an equiprobable quantization. Assuming an equidistant quantization and a higher-order alphabet as output, it is evident that existing works in the domain of ECCs thus far have not been tailored for this scenario. The author of this thesis proposed two schemes, namely a VT-like ECC based on a variable-length mapping of symbols to bits, and a specific type of LMCs based on arbitrary fixed-length mapping of the symbols. As part of the comparison, the superiority of the latter scheme was demonstrated w.r.t. the newly established notion of tamper-sensitivity but also the existing performance criteria reliability and effective number of secret bits. In all cases, previously existing schemes where outperformed.

10.2 Outlook on Reliability Enhancement Techniques

Several aspects are likely to improve in the future. First of all, while a preliminary analysis was carried out, the minor impact of helper data leakage via the equidistant quantization helper data may still need a closer consideration. In addition, working on hybrid

quantization schemes, i.e., a combination of equidistant and equiprobable may provide other more desirable design trade-offs. Another topic not receiving sufficient attention is compensating techniques. Here, they have not been explored systematically, i.e., while simplified versions of such a technique have been applied while working on this topic, there is still a substantial potential left to explore much improved solutions, e.g., how to better separate multiplicative and additive errors in the presence of noise and structural bias, account for effects due to bending the tamper-resistant envelope, etc. This may require updating the measurement circuit accordingly. An advanced compensation technique is essential for the overall performance of a tamper-evident PUF and faces similar issues when compared to quantization schemes and ECCs, as they must operate in an ad-hoc manner, i.e., *without* storing helper data or reference objects, as done for the 3-signal approach [206]. Future proposals of temperature compensating schemes may be based on the combined measurement of absolute and differential values, whereas the absolute values are of significant less resolution such that no to little information on the differential values can be deduced but the drift effects identified and counteracted [97]. Other approaches may be based on further improved measurement circuit techniques, as the proposed solutions in [152, 42] only take additive errors into account, i.e., a combined differential measurement of difference *and* ratio of capacitances may further support compensating efforts.

Yet another line of work is the continuation of the ECCs and the scenario of a higher-order alphabet. Here, debiasing techniques for higher-order alphabets have not been investigated at all. One of the options could be, e.g., the combination of symbols via secret-sharing to provide a well-defined threshold for which no leakage occurs. Preliminary work in that direction was carried out by the thesis author but did not reach a level of sophistication that could have been published. Certainly, better debiasing techniques combined with, e.g., hierarchically structured LMCs, may further improve overall ECC performance without compromising tamper-sensitivity.

Part IV

Properties of Higher-Order Alphabet PUFs

Chapter 11

Performance Metrics

This chapter provides a brief overview of PUF performance metrics, i.e., tools on how to assess a PUF's properties and quality with regard to certain criteria. Furthermore, since HOA PUFs were not considered beforehand, it is necessary to extend existing metrics correspondingly. This is based on the publications [97, 100] with the thesis author as principal author and additional contributions from Aysun Önalan. Another line of work is the invention of new tests to capture properties which thus far, have not been investigated beforehand in the PUF context. Since this thesis is concerned with tamper-evident PUFs, their spatial *properties are of particular importance, e.g., when drilling a hole through the PUF structure, then it is natural to assume that the whole structure behaves the same with regard to tamper-sensitivity and loss in entropy. This homogeneity translates to equality of the PDFs derived from the physical nodes of the PUF with a corresponding test proposed by the thesis author in [94]. However, equality of PDFs may still not be sufficient, as intra-PDF deficiencies could lead to undesirable effects such as correlation of values due to spatial proximity. To also analyze these effects, a spatial extension of the Context Tree Weighting (CTW) method is proposed which is however outside the scope of this thesis. This is based on preliminary work on this topic by the thesis author in close collaboration with Michael Pehl from TU Munich [164]. In particular, the thesis author among other contributions provided the part covering higher-order alphabets together with Daniel Becker [11].*

Contents

11.1 Overview: PUF Performance Metrics

Assessing a PUF's quality is typically based on its output data or some of the intermediate processing steps. In general, there are two classes of tests to assess the quality of this data.

One class is based on an information-theoretic approach, i.e., the targeted outcome of the test is a value that describes the contained entropy. Different definitions of entropy are known in the literature, e.g., *min*-entropy or Shannon entropy [59]. Moreover, determining

the specific value of the targeted entropy for a given empirical data set can also be done differently. There are estimator-based techniques, such as the first work within the PUF context based on Context Tree Weighting [88]. This was later extended by the thesis author for specifics of the physical structures in [164]. An additional publication making use of Context Tree Weighting is the often referenced paper by Katzenbeisser et al. [107]. Other works in that domain include [127]. As alternative to estimator-based techniques, it is possible to estimate the contained entropy directly from the properties of the (fitted) distribution, as for example done in [212].

Another class of tests is based on statistical moments of the PUF distribution. They can be further distinguished in either hypothesis-based or direct techniques. For hypothesis-based tests, as the name implies, a hypothesis with corresponding significance threshold is used to assess the data. This can be based on a non-parametric setting as done in [112] using the Kolmogorov–Smirnov test, or the Welch's t-test as done by the thesis author in [94]. Alternatively, parametric tests may be used such as the Anderson Darling test [229] or Fisher Yates test. One of the main issues of hypothesis-based tests is in properly selecting the significance threshold which often cannot be done in an ad-hoc manner, i.e., only when previous empirical data is available the chosen significance can be properly justified. However, this represents a major disadvantage when evaluating a new PUF design, which is why this class of tests has not been included in this thesis.

Direct techniques can be based either on correlations, multivariate statistics, or raw moments. Examples for correlation-based evaluations are the works by Willsch et al. [233] and Wilde, Gammel, and Pehl in [228]. In both cases, a specific type of correlation-based evaluation is done that is called spatial correlation. Here, the physical structure of a PUF is taken into account and therefore provides valuable insight for the PUF designer. Another approach based on multivariate statistics uses Principle-Component-Analysis (PCA) [229] and a technique called Hierarchical Median-Polish [232]. In both cases, components across the PUF structure can be identified, whereas the latter specifically allows to separate systematic components that would result in a biased PUF output from random components. A limitation that all these tests have in common is that bridging the gap between the test outcome and the actual degradation in entropy of the PUF is difficult. Hence, it is difficult to assess the severity of correlations w.r.t. the performance criteria that matter for a PUF. This is an advantage of the previously mentioned information-theoretic tests, where the result directly represents the entropy.

The last class of performance metrics has also been the most popular, as they include *Uniqueness* and *Reliability*, the most commonly used criteria to assess a PUF [81, 139, 140, 59]. They are also called inter-device distance (Uniqueness) and intra-device distance (Reliability) and are typically used to complement entropy-oriented tests. They operate on the raw moments and/or raw empirical data of the PUF and attempt to answer the intuitive questions whether a PUF is sufficiently different from other instances of the overall PUF population and if it is sufficiently reliable. It has been shown in [59] by Gu et al. that *min*-entropy over binary data is closely linked to the ideal outcome of the Uniqueness. Other work in a similar direction includes [117].

In the following sections, both Uniqueness and Reliability are considered within the context of HOA PUFs. Moreover, an extension of an information-theoretic test, namely Context Tree Weighting, is proposed to provide a tighter entropy bound for PUFs as result of the estimation process. Most importantly though, it is particularly useful for the assessment of a tamper-evident PUF.

11.2 Extension of Uniqueness and Reliability for Higher-Order Alphabet PUFs

Originally defined by Maiti et al. in [140, 139], the metrics *Uniqueness* and *Reliability* have become the de facto standard for PUF publications. While various other metrics have been proposed, they can still be considered as a starting point to assess the fundamental PUF properties, i.e., if PUF values sufficiently differ from each other and if they can be reconstructed reliably. While we recommend to always complement them with additional tests, we nevertheless focus on these two most common metrics with regard to higher-order alphabets which is owed to their popularity. Since the definition of *Uniqueness* and *Reliability* is inherently bound to the distance metric used for the subsequent ECC, we first study the behavior of Uniqueness when defined over the Hamming distance in Section 11.2.1. For example, this would be relevant when choosing Profile P2 of Table 9.1 as designated ECC scheme. Alternatively, when choosing an ECC scheme based on Profile P6 of Table 9.1, as introduced in Chapter 8, then Uniqueness must be defined based on the Lee or Manhattan distance. This is done in Section 11.2.2.

11.2.1 Uniqueness and Reliability based on Hamming Distance

In the following, the (quantized) PUF responses Y_i (cf. Figure 5.2) are considered as symbols within the context of Hamming Distance for the Uniqueness and Reliability, as this represents the interface to the subsequent ECC. Hence, this is an important step to study (and possibly adjust) the outcome of the quantization and select the ECC parameters accordingly, i.e., to systematically study the trade-off between Uniqueness and Reliability. Considering the classical definition of Uniqueness in Equation (11.1) according to [139] with k being the number of PUF devices and v the length of the PUF response in number of bits (or later symbols) of each PUF

$$\text{Uniqueness}_{\text{d}_\text{H},\,\text{non-weighted}} = \frac{2}{k(k-1)} \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \frac{\text{d}_\text{H}(Y_i^v, Y_j^v)}{v} \cdot 100\% \tag{11.1}$$

it is evident that it is based on the Hamming Distance d_H as metric to rate how many *substitutions* are necessary to change one fixed-length string into the other. Please note, the definition of Hamming Distance not only holds true for binary-strings *but also* for strings from a higher-order alphabet, i.e., it is possible to substitute the bits in Equation (11.1) with symbols from a higher-order alphabet.

In general, this equation shows how many percent of the bits differ between PUF responses on average. Assuming an ideal binary PUF that provides i.i.d. bits that are uniform, i.e., it provides an output alphabet $\mathcal{L} = \{l_1, l_2\}$ with $l_1 = 0$ and $l_2 = 1$, the optimum for the Uniqueness is 50 % which is based on the following observation:

$$\begin{aligned}
\text{ExpectedUniquenessBinary} &= 100\% \cdot \sum_{i=1}^{2} \Pr(l_i) \cdot (1 - \Pr(l_i)) && (11.2)\\
&= 100\% \cdot [\Pr(1) \cdot (1 - \Pr(1)) + \Pr(0) \cdot (1 - \Pr(0))] && (11.3)\\
&= 100\% \cdot [0.5 \cdot 0.5 + 0.5 \cdot 0.5] && (11.4)\\
&= 50\% && (11.5)
\end{aligned}$$

i.e., it is expected that half of the bits change when comparing one PUF device to another which is based on a uniform distribution of a binary-PUF. However, as the alphabet size increases from binary to more symbols, the expected output of this metric changes. Assuming an ideal HOA PUF that provides uniform symbols with an alphabet $\mathcal{L} = \{l_1, l_2, \ldots l_q\}$ of size q, i.e., $\Pr(l_i) = 1/q$ for $i = 1, \ldots, q$, the expected Uniqueness becomes

$$\text{ExpectedUniqueness} = 100\% \cdot \sum_{i=1}^{q} \Pr(l_i)(1 - \Pr(l_i)) \tag{11.6}$$

As an example, for an alphabet size of 4 which is equal to having 4 quantization intervals, it is expected that 75 % of the symbols differ between PUF responses, again assuming a uniform distribution. This already increases to 87.5 % for 8 symbols. For non-uniform distributions, e.g., Gaussian, the expected number of symbols to change decreases in comparison to the uniform distribution but Equation 11.6 would still hold true, as it operates on the actual probabilities of the symbols.

In the case of HOA PUFs, when employing Equation 11.1 to compute the Uniqueness and interpreting the result, we choose a lower bound of 50% and the upper bound as [50%, ExpectedUniqueness], i.e., the resulting histogram must be in this range to consider the PUF as sufficiently unique. Alternatively, the lower bound could be chosen based on a stochastic model to provide a stronger rationale. Since ExpectedUniqueness is the best value a PUF can achieve given a distribution without noise, we expect that most empirical data will fail to actually reach that bound. Unlike Uniqueness for binary-PUFs, we now have a metric that better complements the entropy contained in the PUF, as a range of values is acceptable to consider a PUF as unique. This nicely complements entropy-based assessments of the PUF.

If desired, it is still possible to adapt the metric of Equation 11.1 to mimic the behavior of the binary uniqueness scenario in the sense that 50% will be the ideal outcome. This is done by introducing appropriate scaling factors, as also discussed in [140]. For example, ExpectedUniqueness can directly be incorporated into the normalization factor, as done in Equation (11.7).

$$\text{Uniqueness}_{\text{d}_\text{H},\,\text{weighted,Gaussian}} = \frac{1}{k(k-1)\text{ExpectedUniqueness}} \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \frac{\text{d}_\text{H}(Y_i^v, Y_j^v)}{v} \times 100\% \tag{11.7}$$

In the binary case, this normalization factor of $k(k-1)/2$ represents the total number of all possible unique pairwise combinations of PUF responses. Here with the modified normalization factor, the optimum for a given distribution, e.g., Gaussian, now again is 50%. The same approach is followed in Equation (11.8) but for a uniform distribution, therefore allowing a universal comparison independent of the actual distribution present in the PUF system. Note that Equation (11.7) and Equation (11.8) compute the same result in the idealized case of a uniform distribution.

$$\text{Uniqueness}_{\text{d}_\text{H},\,\text{weighted,Uniform}} = \frac{q}{k(k-1)(q-1)} \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \frac{\text{d}_\text{H}(Y_i^v, Y_j^v)}{v} \times 100\% \tag{11.8}$$

As natural extension of the previous equations, we suggest to use a logarithmic representation, as done in Equation (11.9). Now, the best result of a uniform distribution

is unanimously 1, whereas any other result lower than that represents a degradation in Uniqueness. This reflects more naturally the intuition that the more Uniqueness is present, the better is the result. However, at the same time, the outcome is more difficult to interpret.

$$\text{Uniqueness}_{\text{d}_\text{H},\text{weighted},\log,\text{Uniform}} = log_q \left(1 - \frac{2}{k(k-1)} \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \frac{\text{d}_\text{H}(Y_i^v, Y_j^v)}{v}\right) \tag{11.9}$$

For Reliability, we adhere to the previous definition of [139], as presented in Equation (11.10), i.e., a change of a symbol to any other (no matter its distance) is counted as one. A suitable ECC could then either be based on Reed-Solomon (RS) codes [170] employed in a fuzzy commitment scheme or insertion/deletion codes as explained in Chapter 7 prior to the variable-length bit mapping of the symbols.

$$\text{Reliability}_{\text{d}_\text{H}} = \frac{1}{m} \sum_{t=1}^{m} \frac{\text{d}_\text{H}(Y_i^v, Y_{i,t}^v)}{v} \times 100\% \tag{11.10}$$

Practical results for these definitions are illustrated in Section 13.2.2.

11.2.2 Uniqueness and Reliability based on Lee/Manhattan Distance

If Limited Magnitude Codes as introduced in Chapter 8 are to be used for the quantized PUF output (cf. Figure 5.2), then the definition of *Uniqueness* and *Reliability* must be adapted to work with the Lee/Manhattan distance that is relevant for these codes. Hence, this is in contrast to the previous section and in accordance to the q-ary channel model of Section 8.1.

In Equation (11.1), we observe that the PUF metric Uniqueness over Hamming distance is normalized by the length v of the considered response. This must be done in an appropriate manner also for responses over Lee/Manhattan distance, as their length is different due to how the distance metrics d_Lee and d_Man are defined. Lee distance d_Lee between two quantized PUF responses, with a field size of q, is defined below in Equation (11.11). It is circular i.e., $\text{d}_\text{Lee}(0, q-1) = 1$.

$$\text{d}_\text{Lee}(Y_1^v, Y_2^v) = \sum_{i=1}^{v} \min((y_i^1 - y_i^2), q - (y_i^1 - y_i^2)) \tag{11.11}$$

Similar to before, Manhattan distance d_Man between two words is defined below in Equation (11.12). It is non-circular, i.e., $\text{d}_\text{Lee}(0, q-1) = q-1$.

$$\text{d}_\text{Man}(Y_1^v, Y_2^v) = \sum_{i=1}^{v} |y_i^1 - y_i^2| \tag{11.12}$$

where $Y_j^v = \{y_i^j; 1 \leq i \leq v\}$, $j = 1, 2$ and $0 \leq y_i^j \leq q-1$. For LMCs in order to normalize, we apply Plotkin's low rate average distance bound defined in Equation (11.13) for the wrap-around channel [30].

$$\text{d}_\text{Lee} \leq \frac{v\overline{D}}{(1 - K^{-1})} \tag{11.13}$$

where K is the cardinality of C and $\overline{D}$ is the average Lee weight [30] given by Equation (11.14).

$$\overline{D} = \begin{cases} \frac{(q^2-1)}{4q}, & \text{odd q} \\ \frac{q}{4}, & \text{even q} \end{cases} \tag{11.14}$$

For a practical scenario of $v = 128$ nodes in a PUF device with field size $q = 16$, $K = q^{128}$ this leads to $K^{-1} \approx 0$. Thus Equation (11.15) holds which makes it compatible to previous definitions of Uniqueness for binary PUFs [140].

$$\frac{\mathrm{d}_{\mathrm{Lee}}}{v\overline{D}} \leq 1 \tag{11.15}$$

Uniqueness using Lee or Manhattan distance is defined in Equation (11.16) and Equation (11.17) respectively.

$$\mathrm{Uniqueness}_{\mathrm{d_{Lee}}} = \frac{2}{k(k-1)} \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \frac{\mathrm{d}_{\mathrm{Lee}}(Y_i^v, Y_j^v)}{v\,\overline{D}} \times 100\% \tag{11.16}$$

$$\mathrm{Uniqueness}_{\mathrm{d_{Man}}} = \frac{2}{k(k-1)} \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \frac{\mathrm{d}_{\mathrm{Man}}(Y_i^v, Y_j^v)}{v\,q} \times 100\% \tag{11.17}$$

where k is the number of devices and v the number of nodes in a device which is equivalent to its length in symbols. Please note that the computed outcome of these definitions is quite different to the ones based on Hamming Distance, as the magnitude becomes part of the Uniqueness which is no longer just a change in symbol. Hence, to improve Uniqueness, not only the symbols as such would have to change, but also the occurred magnitude. Additionally note that the computed outcome of Equation (11.16) compared to Equation (11.17) is quite different, since the normalization factor in front of the sum remains unchanged.

In particular, the result of Equation (11.17) is such that the Uniqueness is bounded to 100% which would be achieved only when for each symbol a maximum magnitude change is observed. In contrast, empirical results for Equation (11.16) may exceed 100% of Uniqueness when the average Lee weight of Equation (11.14) is exceeded. This confirms that interpreting Uniqueness for PUFs based on a higher-order alphabet is completely different to binary PUFs.

To complement the previous Uniqueness definitions, Reliability is defined for both metrics in Equation (11.18) and Equation (11.19)

$$\mathrm{Reliability}_{\mathrm{d_{Lee}}} = \frac{1}{m} \sum_{i=1}^{m} \frac{\mathrm{d}_{\mathrm{Lee}}(Y_i^v, Y_{i,t}^v)}{v\,\overline{D}} \times 100\% \tag{11.18}$$

$$\mathrm{Reliability}_{\mathrm{d_{Man}}} = \frac{1}{m} \sum_{i=1}^{m} \frac{\mathrm{d}_{\mathrm{Man}}(Y_i^v, Y_{i,t}^v)}{v\,q} \times 100\% \tag{11.19}$$

where m is the number of measurements of same PUF device at different times. Practical results for these definitions are illustrated in Section 13.2.2.

Chapter 12

Conclusions on Properties of Higher-Order Alphabet PUFs

This chapter briefly summarizes the previous contests on the properties of higher-order alphabet PUFs. In addition, a brief outlook is presented to indicate future work.

Contents

12.1 Summary on Properties of Higher-Order Alphabet PUFs

To fill the gap for tests and metrics regarding higher-order alphabet PUFs, this work proposed corresponding extensions of common PUF metrics, namely Uniqueness and Reliability. Adapting these metrics was primarily motivated by the fact that different ECC distance metrics are needed and the considered alphabet is no longer binary. Based on the results of Chapter 9, it was already substantiated that choosing an appropriate metric is essential for the targeted application of a tamper-evident PUF. Similarly, when assessing a PUF's entropy, the higher-order alphabet must be taken into account. This is done in [164] based on an extension of Context Tree Weighting (CTW) that additionally considers spatial effects that locally degrade the entropy contained in the PUF. For tamper-evident PUFs, this is of particular relevance, as an attacker could drill a hole and with the help of the obtained raw measurement values, attempt to reconstruct the missing values that were destroyed as part of the attack. For the purpose of a security certification, it is evident that appropriate thresholds for the tests must be chosen, depending on the level of confidence required and the designated security level. Taking all previous statements into account, it is difficult to imagine how a higher-order alphabet PUF would be assessed without the proposed metrics and tests.

12.2 Outlook on Properties of Higher-Order Alphabet PUFs

Since this work was concerned with PUFs that provide key storage ("weak PUF"), it is evident that these tests and metrics need to be further adapted to better reflect specifics of challenge-response PUFs ("strong PUF"). Moreover, as further detailed in our upcoming work on Spatial CTW [164], there are several specifics of PUF-based statistics that have

not been well investigated yet, e.g., considering the sequence generated by a PUF as a stationary source may not be the best option. Of course, this applies to all PUFs and is not limited to the specifics of a HOA PUF. A completely different direction is to create better metrics for PUF-specific properties, as Uniqueness and Reliability based on their current definition provide only a rather coarse-grained picture of the PUF behavior, e.g., if few positions remain constant in the output data this is not detected by these tests. Ideally, a more complete set of tests would be available to address these issues and other specifics of the PUF, e.g., identify certain distribution errors.

Part V

Case Studies and Applications

Chapter 13

Enclosures: Envelopes and Covers

This chapter is the direct result of the work published in [97, 95] with the thesis author as principal author. The work in [97] has been performed in close collaboration with DSO National Laboratories. In addition to that, the work on FORTRESS is based on thus far unpublished project work. All Proof-of-Concept (PoC) implementations are the result of project work to investigate the underlying principle of mesh-based tamper-resistant enclosures that are leveraged as a tamper-evident PUF. The notable difference between [97] and [95] is the manufacturing process, as the former is based on a standard flexPCB manufacturing process and the latter a fully customized process by Fraunhofer EMFT. For each of these implementations, basic design parameters are presented, corresponding measurement results, and the test results of the robustness towards environmental drift and vulnerability towards drilling attacks. In addition, the work of [97] features a thorough statistical evaluation of 115 flexPCB covers, confirming the overall design rationale of a HOA PUF.

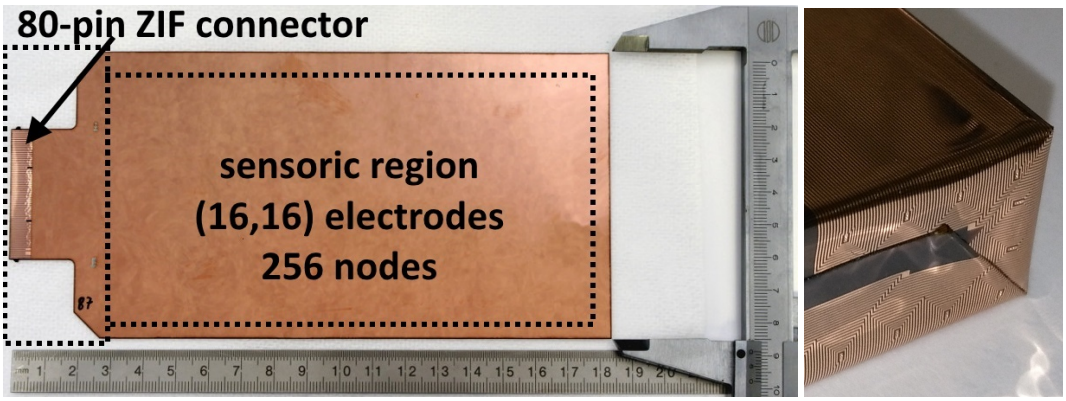
Contents

13.1 B-TREPID and FORTRESS

To demonstrate the feasibility of our approach presented in Part II, we present a case study of 50 manufactured envelopes that contain the tamper-evident PUF and selected technological properties. An early sample with a metallic shield as shown in Figure 13.1a is used for design validation of the concepts. Figure 13.1b demonstrates the envelope wrapping concept around a case that contains the protected module. For better visualization of the

mesh, the shielding was not attached and the sensoric region does not fully overlap as intended. The design properties of the evaluated proof-of-concept envelope are:

- Physical dimension: 185 mm × 90 mm
- 16 × 16 electrodes; 256 sensor nodes; n=990 cells each
- 16 × 16/2 = 128 differential sensor nodes

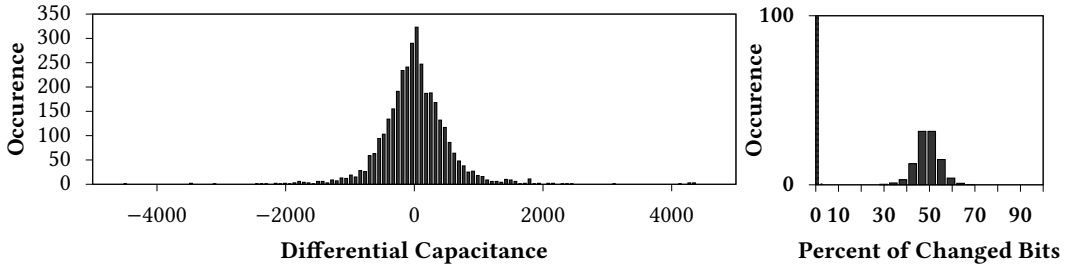


(a) Early technology sample (scale as reference in cm). **(b)** Wrapped envelope.

Figure 13.1: Various aspects of the envelope. 13.1b Exemplary wrap around a corner without attached shield to show mesh.

Manufacturing Process: The mesh is based on lithographic patterning to have a scalable technology that allows even smaller structures in the future. Using a reel-to-reel process with an infinite-length substrate, we deposit copper (Cu) on the first electrode layer by sputtering on a polyimide (PI) substrate. Subsequently, this layer with Rx electrodes is reinforced by an additional semi-additive galvanic process, resulting in a Cu layer of 7 µm. This is necessary to have a defined stop interface while processing the blind vias in the PI substrate by laser ablation. Afterwards, the Tx layer is only sputtered, resulting in a Cu thickness of just 500 nm, while at the same time creating the conductive interconnection between the electrodes on both sides of the PI. The carrier substrate with electrodes is enclosed in a shield on both sides. The resulting height of the layer stack-up is approx. 200 µm, which is important for the flexibility when mounting the envelope. This work has been performed by Fraunhofer EMFT.

Measurement Circuit: A custom discrete measurement circuit was used for testing as described in [152]. Its basic operating principle is to use two antiphasic excitation signals for each Tx pair while the other Tx electrodes remain inactive, thereby creating an in-situ differential capacitance inside the envelope. The resulting current on the Rx electrodes is then further processed by analog circuitry before being sampled, filtered, and evaluated by an STM32 microcontroller. The resulting full-scale range is ±73 fF, which corresponds to −10 000 to +10 000 (in points) in the plots, at a theoretical digital resolution of $\Delta_M = 7.3$ aF (equivalent to 1 point) which is however limited by circuit noise of $\sigma_N = 0.19$ fF when the envelope is connected. Performing a single differential measurement can be done in 0.6 ms. Since it can be parallelized on the Rx side for each TX pair, this results in a theoretical $(16/2) \cdot 0.6\,\text{ms} = 4.8\,\text{ms}$ for the overall envelope, e.g., when implemented fully parallelized in an IC.



(a) Histogram of the obtained differential capacitances. **(b)** PUF Uniqueness.

Figure 13.2: Various results from early technology samples.

13.1.1 Practical Results

A total of 50 envelopes have been manufactured to confirm our design rationale. Since all measurements were performed with the same circuit, the variation observed in the data is only rooted in the variation of the envelopes. To evaluate the statistical properties, we acquired 200 samples over time for each sensor node to compute its noise-free mean. To ensure conservative results regarding the entropy and only assess the manufacturing variation, the envelopes were measured laying straight, such that only the variation inside the electrodes is captured. This leads to the following preliminary results:

- Nominal mutual capacitance of sensor node: $C^{\mathrm{N}} \approx 18\,\mathrm{pF}$
- PDF of differential capacitance γ: $\mu_{\mathrm{s}} = 0.13\,\mathrm{fF}$, $\sigma_{\mathrm{s}} = 6.25\,\mathrm{fF}$, $\sigma_{\mathrm{N}} = 0.19\,\mathrm{fF}$
- Quantization interval width: $\Delta_{\mathrm{Q}} \approx 1.25\,\mathrm{fF} = 2 \cdot y \cdot \sigma_{\mathrm{N}}$ $(y = 3.29)$
- Cell capacitance: $\overline{C_{\mathrm{c}}} = 18\,\mathrm{pF}/990 = 18.18\,\mathrm{fF} > \Delta_{\mathrm{Q}}$

Entropy and Key Generation: Figure 13.2a shows the PDF of γ and contains all sensor nodes from all envelopes. To analyze the entropy of this empirical distribution, we select an equidistant quantization for reasons of a uniform tamper-sensitivity across the measurement range (cf. Chapter 6). Its bin size Δ_{Q} is chosen as multiples of the noise deviation σ_{N}, thereby making the result more robust. For $\Delta_{\mathrm{Q}} \approx 1.25\,\mathrm{fF}$, the computed Shannon entropy yields 4.4 bit per node. Hence, a total of $128 \cdot 4.4\,\mathrm{bit} \approx 560\,\mathrm{bit}$ can be expected from the PUF under ideal conditions. Using the given Δ_{Q}, we experience an average error rate of $\leq 0.1\%$ per sensor node after quantization at room temperature.

However, to compensate for environmental effects such as temperature drift, we need to lower the number of quantization intervals, causing the entropy to drop to 2.5 bit per node. When both envelope and measurement circuit are subject to these environmental influences, this typically results in less than 3 erroneous nodes (out of 128) over the range of −20°C to +60°C, i.e., in addition to the quantization, an error-correcting code is required*. As described in Part III, a well-tailored choice is made by considering the result of each quantization interval as a symbol from a higher-order alphabet. Correspondingly, the best performing ECC known to date would be LMCs, as presented in Chapter 8.

Uniqueness and Reliability At the time of publication in [95], Uniqueness was never considered beforehand for higher-order alphabets. Hence, to compute the Uniqueness

* Please note, this already exceeds the operating temperature range of the IBM 4765 PCIe crypto coprocessor which is *only* +10°C to +35°C [87].

based on previous definitions, we carried out the quantization of Chapter 6 and variable-length bit mapping of symbols proposed in Chapter 7. The obtained variable-length bit strings are then truncated to the shortest output and the Uniqueness computed based on its previous binary definition which results in the plot as shown in Figure 13.2b. It is well-centered around 0.5 and indicates a good PUF behavior. The plot includes the result of the reliability at room temperature, too. However, we emphasize the significant differences of our approach compared to binary-only PUFs, such as the SRAM-PUF, which leads to a much better reliability of the quantized data already. Please note, as this work presents basic research and *not* a final product, we omitted tests based on humidity, vibration, altitude, electromagnetic-compatibility, etc., as they would also depend on the specifics of the overall system that were not considered as part of this work, e.g., potting, specific heat distribution of components, total area.

13.1.2 Drilling Attack

To verify the tamper-evident properties of our enclosure, we attacked one of the envelopes using a 0.3 mm drill. As guaranteed by the chosen structure size, we destroyed one Tx and Rx electrode, here, resulting in open-circuits of Tx13 and Rx10 which is based on the matrix-like layout as described in Chapter 3 (cf. Figure 4.7a). *Independent* of the PUF-properties, this already allows the system to determine that an attack has taken place. Hence, to study the effects on the PUF, we needed to *disable* the integrity check first. The resulting plot in Figure 13.3 shows the *difference* of the capacitances from before and after the attack. As the Tx pair Tx13 and Tx14 is no longer balanced, a dramatic change for the whole group of nodes is observed, i.e., Tx13 and Tx14 towards Rx1 to Rx16. Since Rx10 is destroyed also, it shows up as a peak in all the other Tx groups, e.g., Tx1 and Tx2 (TX group 1) towards Rx10 with x-value 10 and y-value of $\sim$ 1000 in Figure 13.3. As result of our attack, we also see changes in the directly neighboring Tx pairs due to the fact that Tx13 can no longer be properly grounded. In total, we observe > 32 nodes that shift by ≥ 1000 points and are therefore considered destroyed, causing $32 \cdot 2.5\,\text{bit} = 80\,\text{bits}$ of entropy to be lost. This loss is not covered by the ECC of the key generation and imposes a significant computational complexity on the attacker.

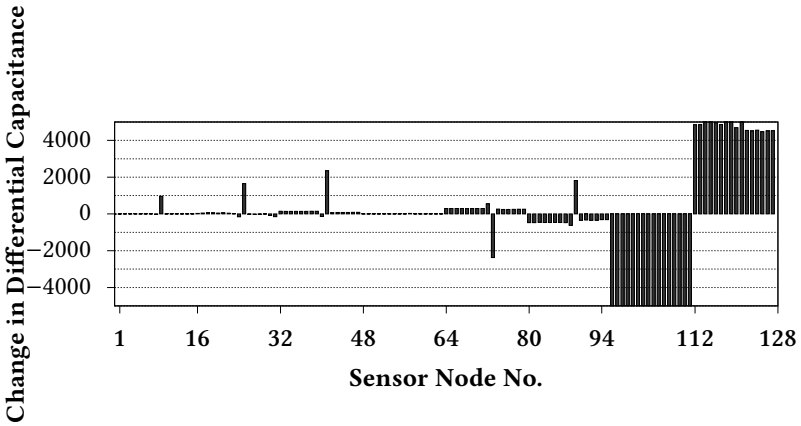


Figure 13.3: Difference of capacitance as result of attack.

13.1.3 Conclusions and Outlook on FORTRESS

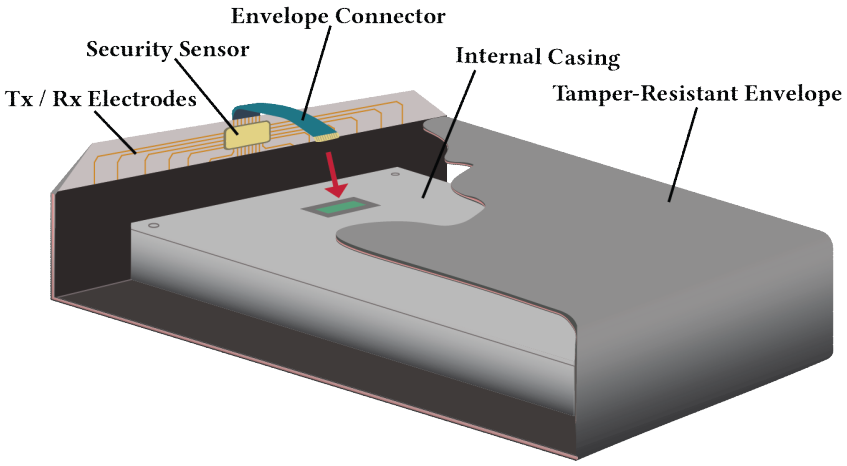
We introduced B-TREPID, a holistic approach to protect embedded devices from the ground-up. It is the first PUF-based envelope that should allow to replace formerly available tamper-detection and response envelopes based on a battery-backed monitoring concept. Both envelope and security architecture have been developed to meet the highest levels of the existing security standards. The envelope is based on the concept of using a dual sensor, i.e., an advanced sensor layout that provides the capability to determine its integrity and, at the same time, use it as a capacitive sensor array to create a tamper-evident PUF. A stochastic model as outlined earlier in this thesis complements the sensor layout and provides a practical guideline for designing envelopes of varying size.

We point out, that the concepts presented here are generic, scalable, and could be implemented using other enclosure technologies. Our tests provide initial evidence that this concept fulfills the desired criteria. As future work, it is evident that the material properties of the envelopes need to be improved, e.g., with a carbon-paste shield for added security and improved integration with the potting. Moreover, as the yield was quite low due to the miniature-sized vias, a vialess layer stack-up based on printed dielectrics must be realized. This aligns with the goal of creating even smaller structures in the range of 10 µm to 20 µm that can be manufactured using available lithographic processes already, i.e., the vias are currently the limiting factor in terms of structure size. Furthermore, we plan to carry out more thorough statistical tests, and a more detailed analysis of the envelope's entropy when it is wrapped.

Beyond these incremental improvements, as part of the designated TAMPERSEC project, we are currently working on embedding the measurement circuit into the envelope, such that the former 80-pin ZIF cable is replaced by an all digital 20-pin ZIF connector cable, as illustrated in Figure 13.4. The designated solution will feature a *thinned* security sensor IC [42] that will be completely integrated in the envelope such that it is no longer visible to the human eye from the outside, as the subsequently applied shielding layer would conceal the IC. This will be accompanied by additional scaling tests to further increase the entropy per node and enclose even larger PCBs. Unfortunately, this was not possible with the given manufacturing possibilities at the time of the COPYCAT project.

The designated combination of envelope and integrated security sensing capability could then be combined with an FPGA-based host system. In that case, the digital signal processing could also be done within the FPGA and the security sensor would merely serve as an analog sensor front-end. Alternatively, the security sensor itself would be realized as a smartcard-like processor to enable a cryptographically secured communication between microcontroller-based host systems (with their own trust anchor), thereby completely avoiding the risk of having an interface that could be probed or eavesdropped by an attacker to obtain signals that carry information on the envelope's integrity.

Right now, the current size of the envelope and case has been designed to enable later compatibility to the PC Card Type III standard in terms of physical dimension. This may open up projects based on EOMA68, the Embedded Open Modular Architecture Standard (68 pin connector variant).



(a) Conceptual drawing of FORTRESS including Security Sensor [42].

(b) Early manufacturing sample of FORTRESS in April 2018.

Figure 13.4: Outlook on FORTRESS, the follow-up implementation of B-TREPID. This will include an improved material composition and enhanced layer stack-up of the envelope, more advanced circuit capabilities, and a designated TRL level of 6 as part of the TAMPERSEC project. Note: envelope wrapping is inside-out to illustrate electrode structure and show the embedded IC that would additionally be thinned for the final version.

13.2 SPECTRE: Secure Physical Enclosures from Covers with Tamper-Resistance

This section is based on [97] where the construction presented in Part II is realized in two covers for the top and bottom of a PCB. We present a case study that is based on the statistical evaluation of 115 of these top covers with a physical dimension of 140 mm × 140 mm and the test vehicle design as shown in Figure 13.5. It is primarily based on an STM32F303 Cortex-M4F microcontroller running at 72 MHz for the evaluation unit. The cover design properties and the resulting capacitive behavior are listed hereafter. Please note the significant difference in the order of magnitude between the capacitances.

- 16×16 electrodes resulting in 256 sensor nodes with $n = 1800$ sensor cells each
- $16/2 \times 16 = 128$ differential sensor nodes due to how the measurement circuit operates
- Parasitic capacitance: $C^{\mathrm{P}} \sim 1.8\,\mathrm{nF}$; mutual capacitance: $C^{\mathrm{M}} \sim 50\,\mathrm{pF}$; variation of differential capacitance: $C^{\mathrm{V}} < \pm 132\,\mathrm{fF}$; on average per-cell capacitance: $\overline{C_{\mathrm{c}}} = 50\,\mathrm{pF}/1800 = 27\,\mathrm{fF}$

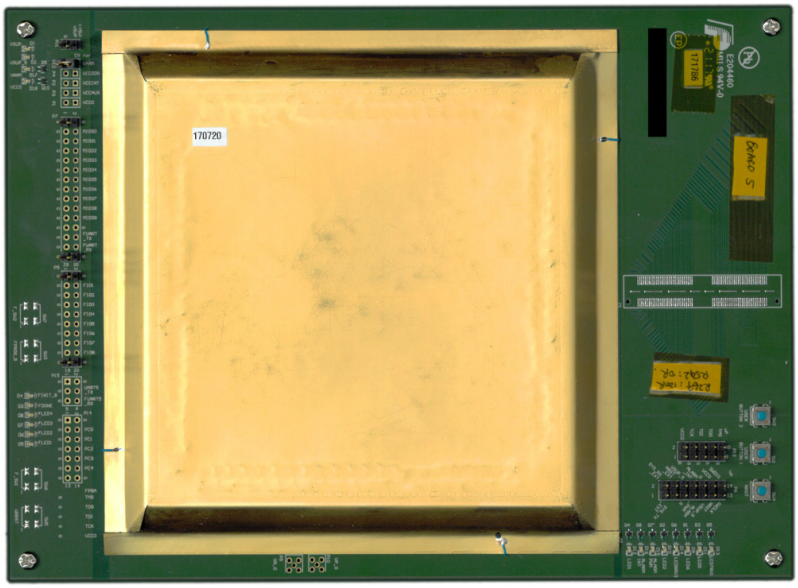


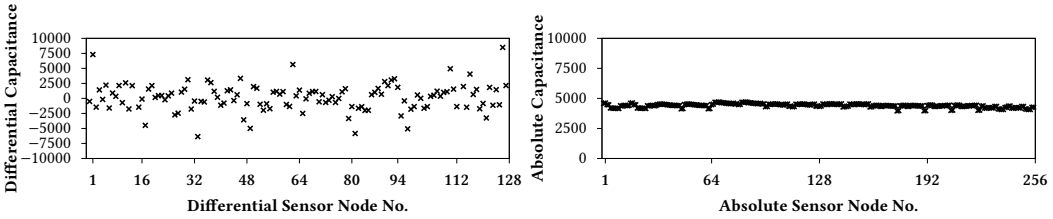
Figure 13.5: Test vehicle implementation with flexPCB cover of size 140 mm × 140 mm.

13.2.1 Statistical Evaluation

In the following, let us consider basic statistics obtained from the measurement of 115 top covers. This is done for both the differential and absolute measurement of the capacitance. The absolute capacitance measurement provides an even more complete picture of the PUF properties inside the cover.

Exemplary Measurement Output. In Figure 13.6, an exemplary output of a *single* measured cover is shown. The output of the differential output is plotted in Figure 13.6a. Clearly visible is the scattered distribution of values in the range of −10 000 to +10 000

(in points), indicating that there is randomness in the covers (otherwise it would be a flat line along the y-value 0). This is in contrast to Figure 13.6b which shows the output of the absolute capacitance measurement. Clearly visible is a rather straight line that indicates much less variation when compared to the differential measurement. A structural bias becomes visible when zooming into the range of 4000 to 5000. This is best analyzed when considering the overall set of 115 covers as visualized in Figure 13.10d as part of the statistical evaluation of the absolute capacitance. It well reflects the expectation that directly neighboring electrodes have about the same nominal capacitance C^{N}, i.e., absolute capacitance values always occur in pairs.



(a) Output of a differential measurement. **(b)** Output of an absolute measurement.

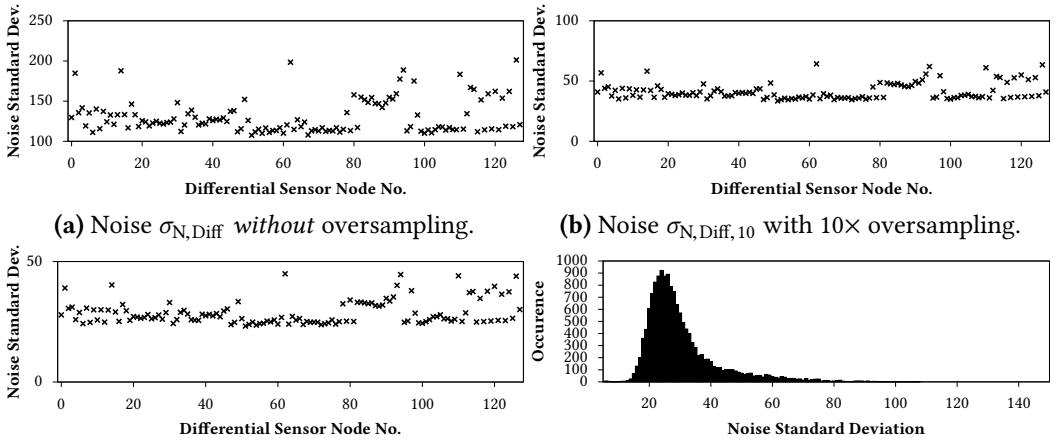
Figure 13.6: Exemplary measurement output of a *single* cover to illustrate basic properties of the system. 200 samples over time were averaged to create a noise-free representation.

Statistical Evaluation of Differential Capacitance Measurement

The statistical evaluation of the differential measurement concentrates on the noise, the manufacturing variation, and the resulting entropy. This comprehensive evaluation strongly supports the chosen design rationale based on the provided data.

Measurement noise. In Figure 13.7a, the noise standard deviation $\sigma_{\mathrm{N,Diff}}$ of the differential measurement is plotted for each individual sensor node over the set of all 115 covers. Clearly visible is a mostly uniform behavior across the whole range of nodes and an expected value of $\bar{\sigma}_{\mathrm{N,Diff}} = 130$. Only Tx-group 6 (Tx11 and Tx12) shows a slightly degraded noise performance which may require further investigation. Without further adjusting the number of measurement periods, a direct oversampling of the values by a factor of 10 leads to the plot in Figure 13.7b with a reduced noise level of $\bar{\sigma}_{\mathrm{N},10} = 39$. This increases the measurement duration to 384 ms in our proof of concept implementation. Further increasing this to a 20× oversampling only reduces the noise to $\bar{\sigma}_{\mathrm{N},20} = 29$ at the cost of 768 ms for the measurement (cf. Figure 13.7c). Even with this tremendous oversampling, resulting in an extremely low noise behavior, we would still be at an equal performance level compared to the solution of [209] whose authors state a measurement duration of 620 ms to 930 ms. To minimize the time for device start-up, we choose an oversampling of 10× while still reducing the noise.

The distribution of the occurring noise per node deviation (*not* of the noise itself which is Gaussian) is shown in Figure 13.7d and illustrates that the higher the noise is, the fewer occurrences are seen. Overall, this ensures a high level of confidence in the low noise behavior of the design which is essential for PUF-based tamper-evident applications. Of course, with a fully parallelized implementation of the circuit in an ASIC, both noise level and measurement duration are likely to be further improved.



(a) Noise $\sigma_{\mathrm{N,Diff}}$ *without* oversampling.

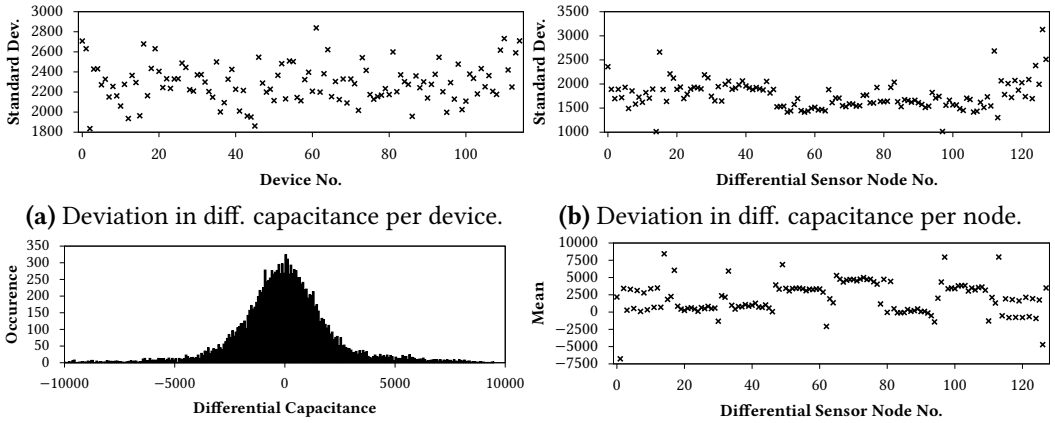
(b) Noise $\sigma_{\mathrm{N,Diff},10}$ with 10× oversampling.

(c) Noise $\sigma_{\mathrm{N,Diff},20}$ with 20× oversampling.

(d) Noise distribution of all $\sigma_{\mathrm{N,Diff},20}$.

Figure 13.7: Statistical evaluation of 115 flexPCB covers (noise behavior).

Manufacturing variation. In Figure 13.8a, the device-specific standard deviation of the observed capacitance values is plotted with an expected value of $\bar{\sigma} = 2290$. To investigate the question whether there are "weak" spots of little deviation, we created Figure 13.8b which shows the standard deviation of the capacitance values per sensor node.



(a) Deviation in diff. capacitance per device.

(b) Deviation in diff. capacitance per node.

(c) PDF of normalized capacitance of nodes.

(d) Normalization offsets (structural bias).

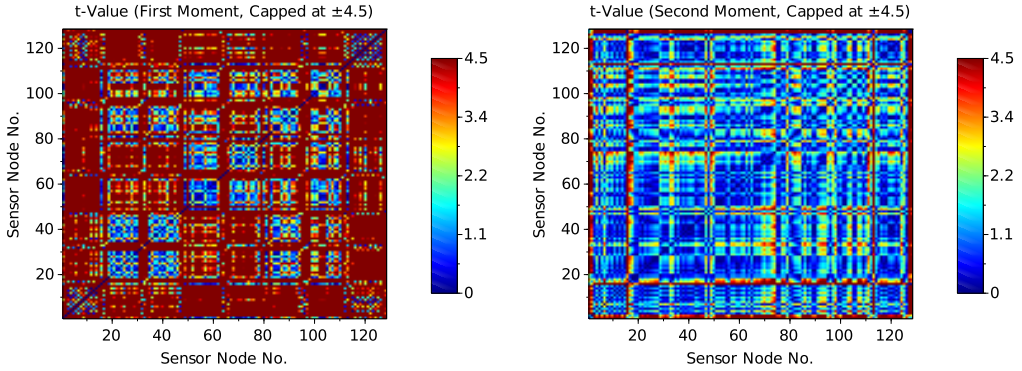
Figure 13.8: Statistical evaluation of 115 flexPCB covers (differential capacitance).

This is of particular importance within the context of physical attacks, since we assume that the PUF entropy is spatially distributed. If this would not be the case, an attacker may characterize the PUF by gaining partial knowledge of its distribution from previously analyzed devices and then use this knowledge to attack that specific location of the cover where the standard deviation is the smallest, thereby minimizing the damage. As supported by the plot in Figure 13.8b is in terms of variation the differential measurement indeed a suitable approach to prevent such structural bias or imperfections, thereby avoiding the risk of the aforementioned attack scenario. There are only two nodes that appear to have a rather low manufacturing variation. However, as seen in Figure 13.8d this stems from the fact that the corresponding sensor nodes are affected by a structural bias in their expected value, causing some of the variation to hit the limit of the measurement range. This is an

imperfection of the layout due to the irregular shape of the top cover and will be addressed in the next hardware revision.

To complement these tests, we applied Welch's t-test as proposed in [94] to create Figure 13.9a and Figure 13.9b. This is essentially a hypothesis-based comparison of each per-node PDF, e.g., of the differential capacitance node 1 to all the other per-node PDFs, resulting in a matrix where both x and y-axis refer to a sensor node and the corresponding value as indicated by the color is the output value of the t-test. Once the test value exceeds $|t| > 4.5$, the PDFs are statistically distinguishable with very high probability. As indicated by Figure 13.8d differ the means across Tx groups. Figure 13.9a clearly supports that this difference is statistically relevant, i.e., the considered PDFs are distinguishable by their first statistical moment, indicating a structural bias that is present across different Tx groups.

In contrast, Figure 13.9b compares the PDFs in their second statistical moment, i.e., only the variation is considered. The result shows that only few comparison exceed the threshold of $|t| > 4.5$, i.e., the differences in variation of Figure 13.8b are not statistically relevant most of the time. As our data processing attempts to extract only the variation by removing first-order bias, this confirms the good PUF behavior at the stage of the raw data already.



(a) Welch's t-test (1st moment). **(b)** Welch's t-test (2nd moment).

Figure 13.9: Statistical evaluation of 115 flexPCB covers based on Welch's t-test [94].

Entropy (Global Analysis). Figure 13.8c shows the PDF of $\Delta C = \gamma$ and contains all sensor nodes from all covers. Its standard deviation σ is 2241 points which equals 29.58 fF. To compute the entropy, we apply an equidistant quantization as presented in Chapter 6. Its bin size Δ_Q (equivalent to Q_w) is chosen as multiples of the noise deviation $\sigma_{N,Diff}$, thereby making the result more robust. For $\Delta_Q = 2 \cdot 3.29 \cdot \bar{\sigma}_{N,Diff} \approx 11.3$ fF, the computed Shannon entropy yields 3.45 bit per node. With 10× oversampling, this changes to $\Delta_Q = 2 \cdot 3.29 \cdot \bar{\sigma}_{N,Diff,10} \approx 3.4$ fF resulting in 5.2 bit per node. Hence, a total of $128 \cdot 5.2$ bit ≈ 665 bit can be expected from the PUF under ideal conditions. Using the given $y = 3.29$ for the quantization, we experience an average error rate of $\leq 0.1\%$ per differential sensor node at room temperature. For the full temperature range of −20 °C to +60 °C, the results are presented in Section 13.2.4.

Entropy (Spatial Analysis). To further investigate inter-dependencies of neighboring nodes from an information-theoretic point of view, we developed an extension of the Context Tree Weighting (CTW) method [164, 88] which we call Spatial CTW (or SCTW in short). Due to how we interpret the PUF output, this spatial extension is based on q-ary symbols as opposed to bits. Hence, the differences to the classical CTW are: instead of

considering the successive bit of a context does our approach operate on the successive higher-order alphabet symbols, in addition, we consider a context comprising all nodes within a certain spatial radius around the targeted node, whereas a radius of 1 corresponds to a tree depth of 8 and a radius of 3 to a tree depth of 48. This analysis can be interpreted as follows: if an attacker would be able to destroy one node only and obtain all values of the surrounding nodes, what is the remaining conditional entropy left to reconstruct the single destroyed node.

In our case, for a total of 32 quantization intervals, the obtained results of the SCTW analysis were 3.1 bit for a radius of 3, the same for a radius of 2, and 3.7 bit for a radius of 1, i.e., lower than the Shannon entropy, indicating a minor degradation in entropy due to inter-dependency of values. Still, the results support the properties of the overall design. With designated improvements in the future, e.g., layout randomization, this behavior is expected to improve as the size of the node square will be smaller, and several distributed pieces of the enclosure jointly measured, thereby mitigating a local bias in the data.

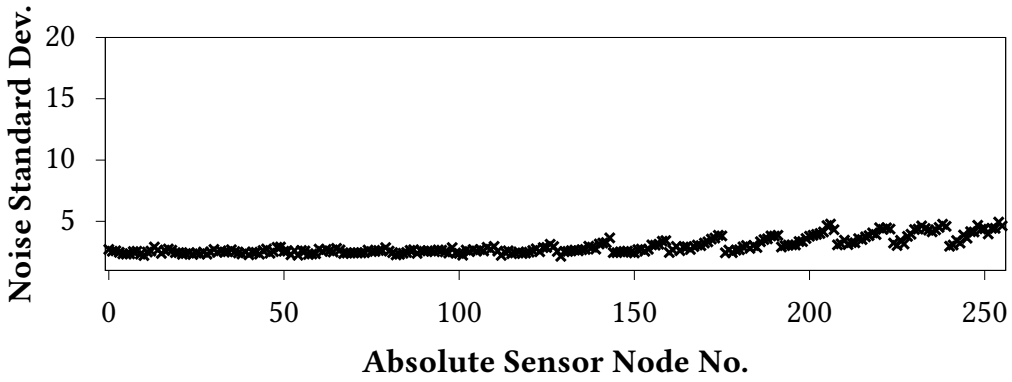
Statistical Evaluation of Absolute Capacitance Measurement

For the sake of completeness, we include the statistical properties of the absolute capacitance measurement. While they are by far less critical for the contained PUF, they are nevertheless important for the overall design to provide consistency with our assumptions regarding the differential measurement. The statistical evaluation of the absolute capacitance measurement is done on the same data set of 115 flexPCB covers. In Figure 13.10a, the noise standard deviation per node is shown. Clearly visible is that the noise of the absolute capacitance measurement only has a minor impact on the data acquisition, i.e., $\bar{\sigma}_{\mathrm{N,Abs}} = 3$ which is equivalent to ±30 fF.

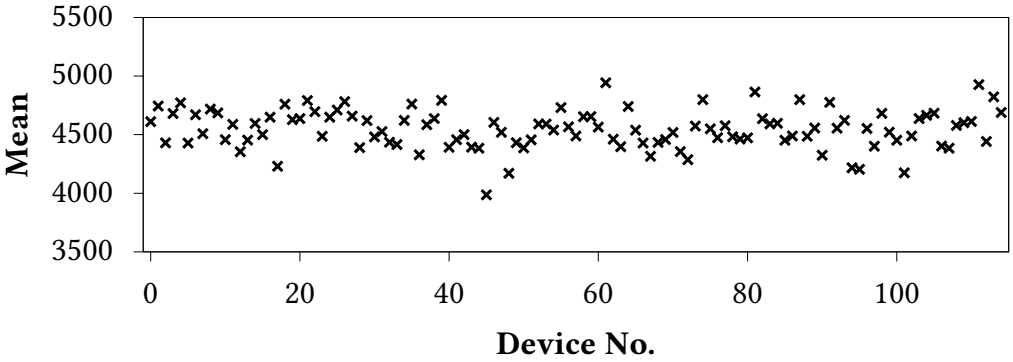
To analyze the absolute capacitance variation, we provide Figures 13.10b and 13.10c that show a per-device average absolute capacitance varying in the range of 40 pF to 50 pF while the per-node standard deviation is approx. at 4 pF. Few outliers are observed that are attributed to bending the flaps which induces mechanical stress resulting in miniature cracks in the copper tracks, as the bending radius is rather tight. In Figure 13.10d is the per-node mean of the capacitance shown. While there is a distinct pattern, it is also visible that data points occur in pairs, i.e., directly neighboring absolute capacitance nodes indeed have a highly similar nominal capacitance. This supports our previous arguments regarding the differential measurement and chosen pairwise electrode layout.

13.2.2 PUF Properties – Uniqueness and Reliability

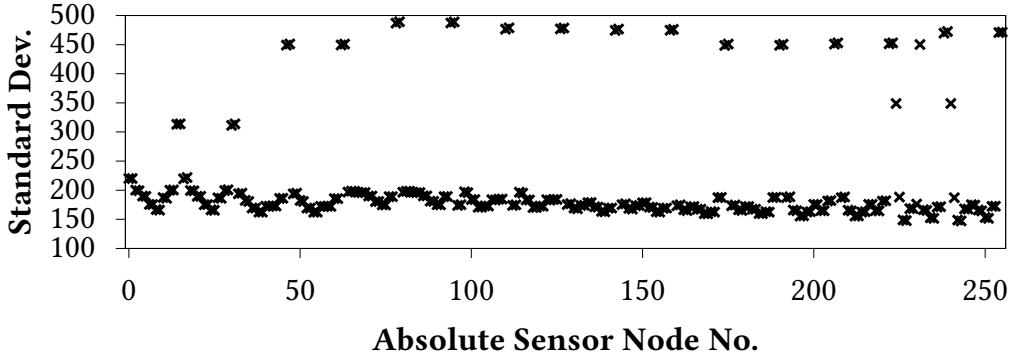
The resulting behavior for both Uniqueness and Reliability over Hamming Distance d_H according to Section 11.2.1 based on our data set is illustrated in Figure 13.11a and Figure 13.11b (*without* using oversampling). The minimum boundary of 50% is illustrated as a solid vertical line, while ExpectedUniqueness as a dotted line. For 16 quantization intervals, the reliability is very high while the Uniqueness is centered between the two defined boundaries. Now, when increasing the number of intervals this increases the entropy we can extract from the PDF and the histogram of the Uniqueness moves closer to the dotted line which by itself also moves towards 100%. At the same time, since the width of the quantization interval reduces, the effect of the noise becomes more dominant, thereby clearly affecting the Reliability. Overall, Uniqueness defined over Hamming Distance shows



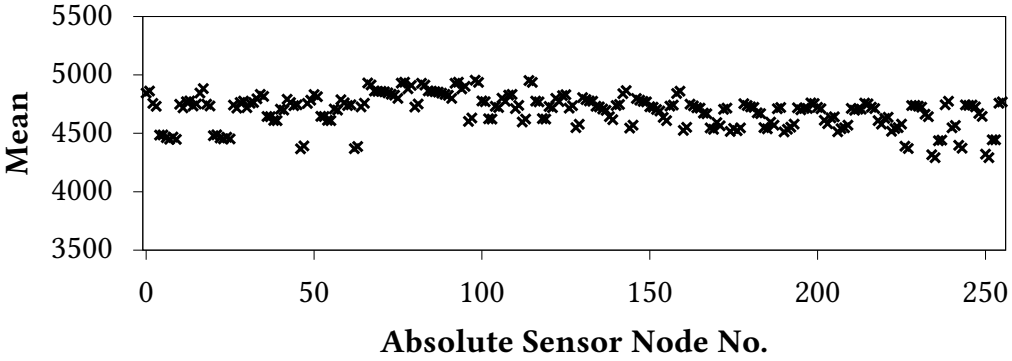
(a) Noise $\sigma_{\mathrm{N,Abs}}$ *without* oversampling.



(b) Mean of absolute capacitance per device.



(c) Deviation of absolute capacitance per node.

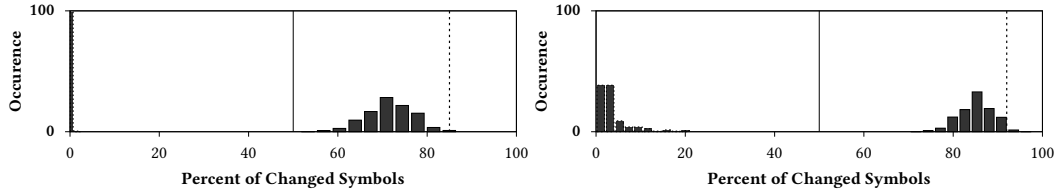


(d) Mean of absolute capacitance per node.

Figure 13.10: Statistical evaluation of 115 flexPCB covers (absolute capacitance).

that a majority of the symbols change when comparing one PUF response from one cover with the PUF response of a different cover.

In addition to the previous plots, we illustrate Uniqueness and Reliability over Manhattan Distance d_{Man} in Figure 13.12. In contrast to the previous figures, Uniqueness appears relatively low which is owed to the fundamentally different definition of Uniqueness over Manhattan Distance that combines changes in symbols and magnitude at the same time. To put the outcome into perspective note that for the given parameters, a change in 3.125% corresponds to the case when all comparisons between symbols result in a magnitude of $d_{Man} = 1$. For the given data, the average Uniqueness is 21.897% which is very close to the case that every compared symbol has a distance $d_{Man} = 7$ which corresponds to 21.875% of Uniqueness. Considering the fact that this is the first such implementation which differs quite significantly from other PUFs, Uniqueness appears at a reasonable level which could be improved though to make it more unique. In contrast, Reliability is at a very high level. Overall, the results show that interpreting the PUF output as a higher-order alphabet nicely complements previous works in this domain, while opening up a new path for ECCs, i.e., working on higher-order alphabets instead of a binary PUF output.



(a) Uniqueness and Reliability: 16 quantization intervals (*without* oversampling).

(b) Uniqueness and Reliability: 32 quantization intervals (*without* oversampling).

Figure 13.11: Statistical evaluation of 115 flexPCB covers (Uniqueness/Reliability) based on Equation (11.1), Equation (11.6), and Equation (11.10) of Section 11.2.1.

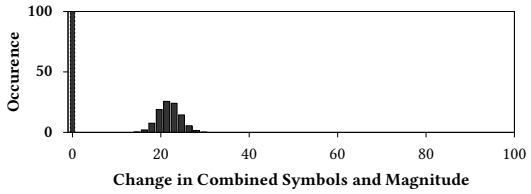


Figure 13.12: Statistical evaluation of 115 flexPCB covers (Uniqueness/Reliability) based on Equation (11.17) and Equation (11.19) of Section 11.2.2. The corresponding data is obtained with a 10× oversampling and $L = 32$ quantization intervals which translates to a field size of $q = 32$

13.2.3 Practical Security Analysis

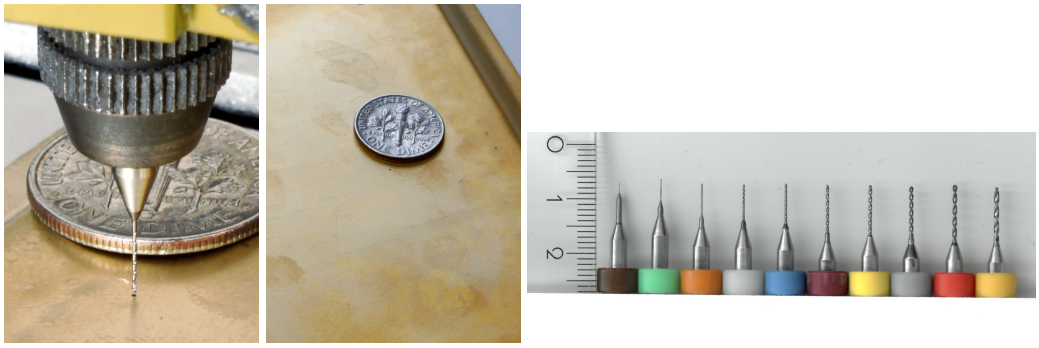
In the following, we provide practical evidence for the difficulty of tampering with the cover without causing detection. Clearly, it is not possible to exhaustively cover all possible attacks within the limited scope of a research oriented project. Hence we do not claim a complete protection against all attacks. Instead, it should be considered as a study on the presented enclosure concepts to demonstrate that practically carrying out a successful attack would be challenging, in particular when considered as a black-box design with

limited prior knowledge, i.e., a real-world design would include additional obfuscation techniques to increase uncertainty for the attacker.

The choice of parameters for the quantization is based on the results of Section 13.2.4 and accounts for possible changes in the environment, too. Hence, realistic parameters are chosen to assess the intrusion detection. These parameters are $\Delta_Q \approx 500$ as quantization width Q_w which leads to 40 quantization intervals and a Shannon entropy of 4.18 bit per differential node. This corresponds to a *min*-entropy of 3.46 bit per differential node. We note that this is based on a 10× oversampling for system startup.

Invasive Attacks: Drilling

To investigate the tamper-evident properties of our enclosure with respect to the assumed attacker model, we attacked several covers by drilling various types of holes and carrying out attempted repairs. Thus, our focus is on open-circuits and corresponding repairs, as short-circuits, especially on the Tx side are prone to cause damage to the circuit. There is no plausible benefit for the attacker to deliberately cause such short-circuits. For drilling, we used a multifunction rotary tool (a "dremel") with corresponding workstation as shown in Figure 13.13a. High revolutions per minute (RPM) are required to not break the fragile drill bits illustrated in Figure 13.13c. Since the structure size is chosen with respect to the assumed minimum drill diameter of 0.3 mm, it is guaranteed that at least one Tx *and* Rx electrode are cut-off. Therefore, larger drill sizes will cause even more damage.



(a) Attack close-up. **(b)** Hole from distance. **(c)** Drill bits from 0.1 mm to 1.0 mm.

Figure 13.13: Exemplary attack on cover with 300 µm drill and a US dime as reference showcasing the disproportion of attack size to overall size of cover.

For smaller drill sizes than 0.3 mm that are outside of the assumed attacker model, there is still a reasonable chance of sufficient damage to cause detection, e.g., a diameter of 0.2 mm is still guaranteed to break at least one Tx *or* Rx electrode. Even for 0.1 mm, there is still a chance left to break electrodes based on the position of the drill hole. Please note, for all drilling attempts that severed electrodes, we had to *disable* the integrity check first, i.e., without attempted repairs and *independent* from the PUF-properties this would already allow the system to determine that an attack was carried out.

In the following, we study several attack profiles that we chose based on our understanding of the system*. Please note that the attacked layout follows the logical representation shown in Figure 13.14. Therefore, when attacking the beginning of an electrode, this refers

* Here, we want to emphasize that for some of these attacks, it took us several attempts in carrying out the attack strategy as intended, even though the text neglects this fact.

to the input side of an electrode denoted as either Ti for a Tx electrode or Ri for an Rx electrode. In total, the following profiles/attacks were carried out:

- Attack Profile 1 (P1): Single 0.3 mm Hole. Beginning of Tx.
- Attack Profile 2 (P2): Single 0.3 mm Hole. Center of Tx.
- Attack Profile 3 (P3): Two-Holes of 0.3 mm. Additional Tx Damage.
- Attack Profile 4 (P4): Single Hole of 0.33 mm, Symmetric Rx Cut-Off.
- Attack Profile 5 (P5): Single Hole of 0.33 mm, Symmetric Tx Cut-Off.
- Attack Profile 6 (P6): Advanced Attack with Attempted Repair.
- Attack Profile 7 (P7): Advanced Attack with Attempted Repair.

In general, these profiles have been created to systematically study the effects of different attacks. Other than the mentioned criteria in the profiles, the selection which Tx or Rx electrode to attack was done randomly.

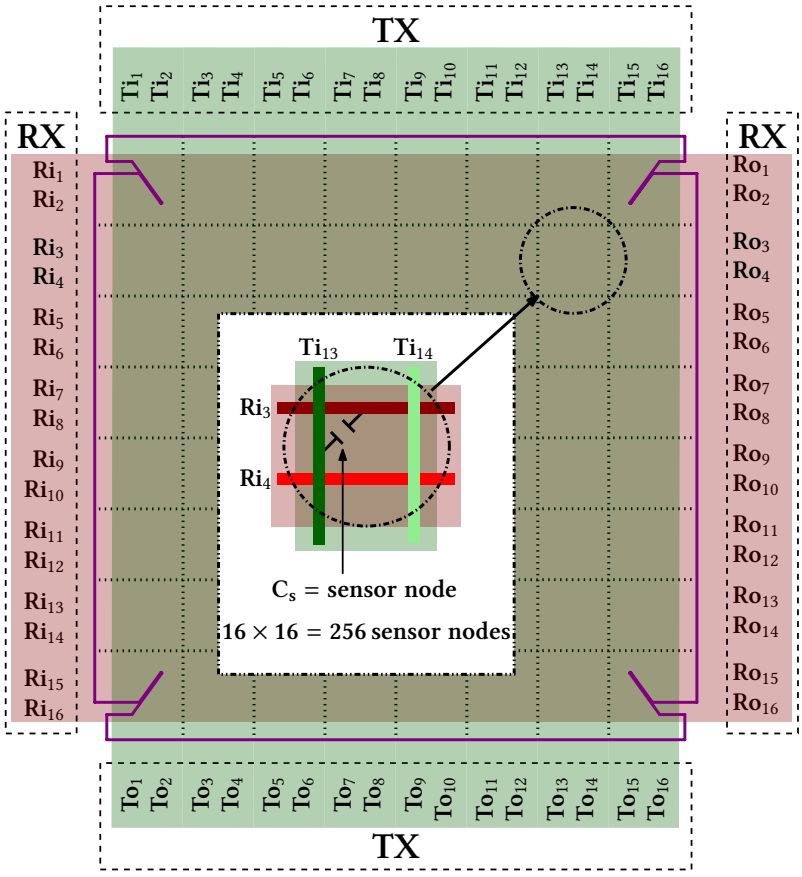


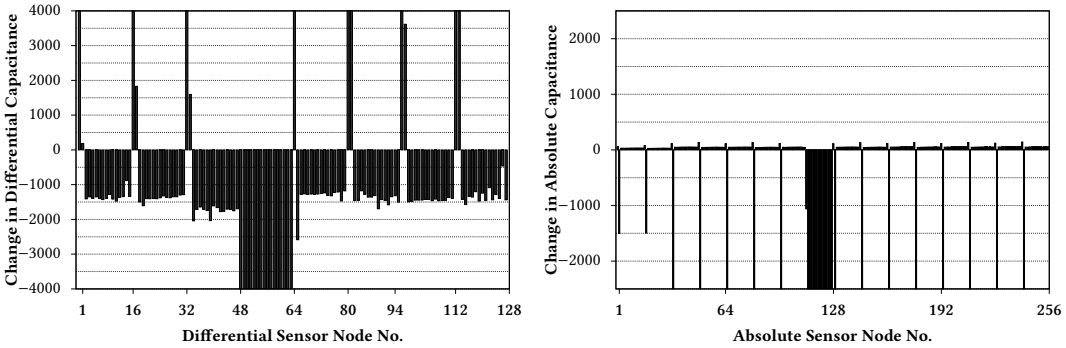
Figure 13.14: Logical layout of the PUF-based covers.

Attack Profile 1 (P1): Single 0.3 mm Hole. Beginning of Tx. As a start, we created a single hole of 0.3 mm relatively close to the beginning of a Tx electrode. In this case, the affected electrodes were Tx8 and Rx2. The resulting plot in Figure 13.15a shows the

noise-free *difference* of the differential capacitances from before and after the attack, i.e., the nodes were measured 200 times and averaged to remove the noise.

As the Tx pair consisting of electrodes Tx7 and Tx8 is no longer balanced, a dramatic change for the whole group of differential nodes is observed. Since Rx2 is destroyed also, it shows up as significant change in all the other Tx groups. Rx1 also appears to have taken damage but is not flagged by the integrity check as broken. Moreover, cut-off electrode parts lead to improper grounding, creating a changed coupling behavior which in turn results in additional shifts for a majority of the other nodes at the stage of the discretized PUF data. For the specific attack considered, all but one of the nodes have significantly moved away from their enrollment such that they would have had a different value during reconstruction. Hence, recovery of the key either by direct measurement of the cover or extracting the circuit's data would have been infeasible.

To complement the differential measurements, we show the result of the difference in absolute capacitance in Figure 13.15b. The significant change in values is easily detectable by Tamper Detection C, i.e., the change is larger than 15% of the absolute capacitances' mean. By computing the difference to the mean, drift effects such as temperature would be accounted for even under different environmental conditions (see Section 13.2.4).



(a) Change in differential capacitance (P1). **(b)** Change in absolute capacitance (P1).

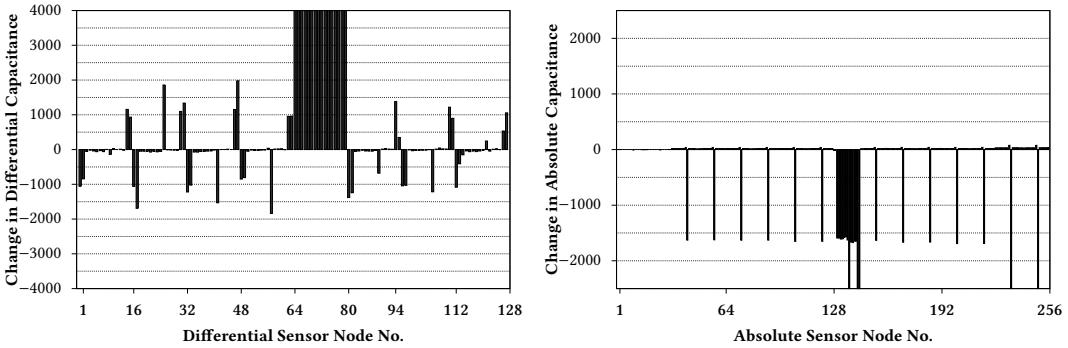
Figure 13.15: Attack Profile 1 (P1): result of a single hole of 0.3 mm in diameter, severing electrode Tx8 and Rx2. Clearly visible is the significant change in values.

Attack Profile 2 (P2): Single 0.3 mm Hole. Center of Tx. As next step, we started over with a new cover. This time we created a single hole of 0.3 mm in the center of a Tx electrode to balance the cut-off parts of both Tx and Rx electrodes. The affected electrodes were Tx9 and Rx10. Figure 13.16a shows the resulting plot of the change in differential capacitance.

Since Tx9 does no longer create a balanced Tx pair with Tx10, again a severe change for the whole corresponding group of differential values is observed. As Rx10 is destroyed also, it shows up as significant change in all the other Tx groups. Due to a more centered destruction of Tx and Rx is the global change in the coupling behavior not as significant when compared to P1. Still, some additional shifts in several other nodes occur.

In general, experimental results support the argument that for a hole of 0.3 mm, the whole Tx group (one column) and the affected Rx group (one row) are always sufficiently altered, resulting in at least $8 + 16 - 1 = 23$ destroyed nodes, i.e., nodes that shift by ≥ 500 points. Hence, we expect that at least $23 \cdot 3.46\,\text{bit} = 80\,\text{bits}$ of *min*-entropy are destroyed by a single hole without attempted repairs. Taking into account that only a fraction of differential nodes happen to reside on the center of a quantization interval, it is likely that for most practical experiments more nodes differ from the quantized value of

their enrollment even for smaller shifts. For the specific cover of Figure 13.16, we observed a total of 47 differential nodes that would have moved away from their enrollment. This is still idealized in the sense that we are considering noise-free values, i.e., an attacker would need to deal with noisy values which increases the difficulty of an attack. Hence, the actual loss in entropy would have been even higher under real-world conditions. Moreover, results for the difference in absolute capacitance in Figure 13.16b again provide strong evidence that in addition to the loss in entropy, the attack would have been detected upon power-on prior to generating the key.



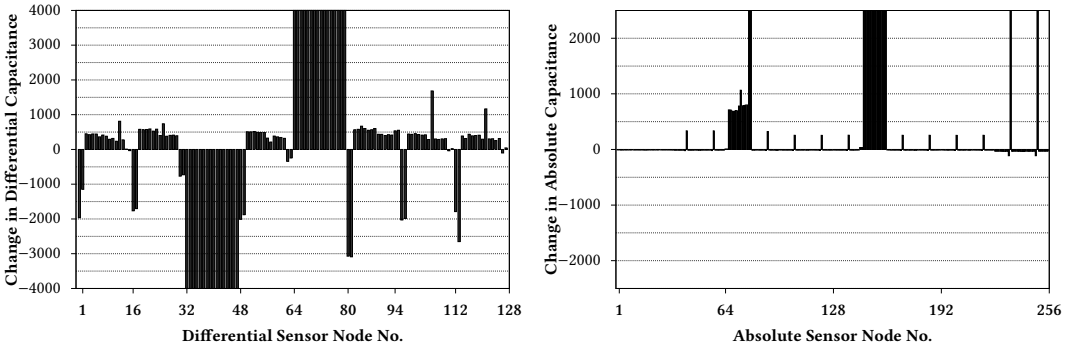
(a) Change in differential capacitance (P2). **(b)** Change in absolute capacitance (P2).

Figure 13.16: Attack Profile 2 (P2): result of a single hole of 0.3 mm in diameter, severing electrode Tx9 and Rx10. Clearly visible is the significant change in values.

Attack Profile 3 (P3): Two-Holes of 0.3 mm. Additional Tx Damage. For the next step of the analysis, again a new cover was used. This time, two holes of 0.3 mm in diameter were created while aiming at shorter cut-offs of the Tx electrodes which corresponds to attacking Rx electrodes with a higher number (cf. Figure 13.14). The first hole severed Tx5 and Rx10. To minimize the damage of the overall attack, we created the second hole such that only Tx10 was additionally cut-off. This is possible by penetrating the cover at a spot where Rx10 is cut-off once more. The resulting damage of the differential capacitance measurement is shown in Figure 13.17a. As expected, we see two devastating shifts in two Tx groups. Moreover, we see a result that is consistent with P1, i.e., a global shift occurs which indicates a severely degraded behavior within the cover due to improper grounding of unused signals. This would again render almost all capacitive nodes destroyed. From this result, we deduce that the more damage to Tx electrodes is done, the worse is the global shift. We confirmed this behavior for other attacks causing more damage. Hence, even when aiming at shortest cut-offs, it is improbable for an attacker to succeed without attempted repairs.

In the plot of Figure 13.17b showing the difference in absolute capacitance we again see severe changes in the capacitive behavior, too. Clearly visible are the two groups as result of the two broken Tx electrodes. Moreover, when comparing the data between the first and second hole, we see a difference in the change of the Rx10 electrode which is owed to the two different points where it was damaged (plot omitted). Hence, by using the information drawn from the absolute capacitance measurement it is possible to provide a spatial estimate of where the attack took place.

Attack Profile 4 (P4): Single Hole of 0.33 mm, Symmetric Rx Cut-Off. For the analysis of the next attack profile, again a new cover was used. This time, an uncommon drill bit of 0.33 mm in diameter was used to create a hole of approximately the same diameter. The affected Tx electrodes were Tx2 and Rx1 and Rx2. Based on geometrical considerations,

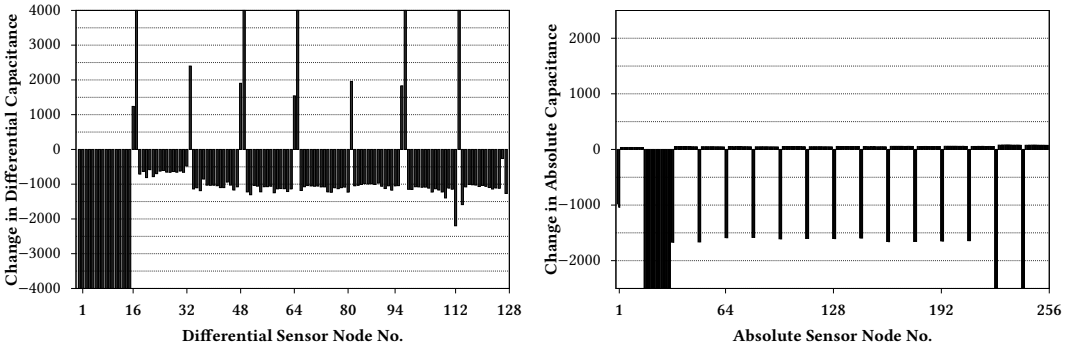


(a) Change in differential capacitance (P3). **(b)** Change in absolute capacitance (P3).

Figure 13.17: Attack Profile 3 (P3): attack with two holes of diameter 0.3 mm severing Tx5, Tx10, and Rx10.

we consider this as a perfect symmetric cut-off of the electrodes Rx1 and Rx2. This attack leads to the change in differential capacitance as shown in Figure 13.18a. Again, we hit the Tx electrode more towards its beginning, resulting in a severe shift in all values due to a much larger portion of the electrode that has been cut-off. Hence, if an attacker would not be able to repair any damage, the best strategy for the current circuit implementation (e.g., when not measuring from both sides) would be to attack electrodes such that the cut-off parts are the shortest and farthest away from the excited input.

Clearly visible is the overall severe damage that does not justify a more detailed analysis. Furthermore, the change in absolute capacitance as shown in Figure 13.18b also indicates an attack. Hence, there is no advantage in attempting a symmetric Rx cut-off.



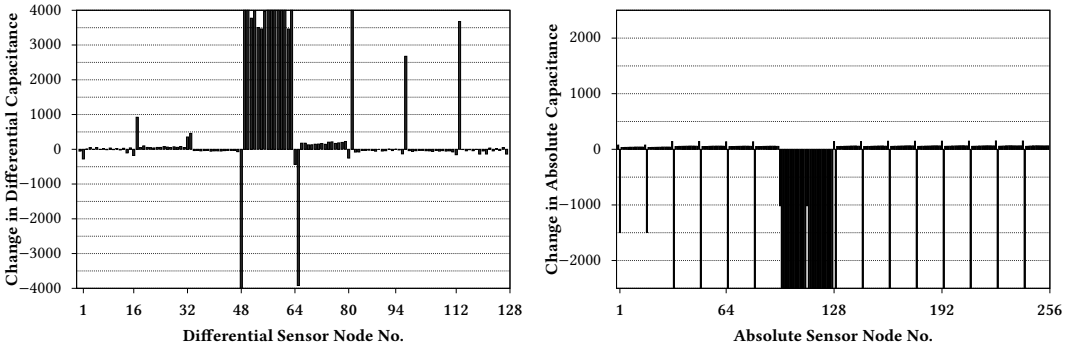
(a) Change in differential capacitance (P4). **(b)** Change in absolute capacitance (P4).

Figure 13.18: Attack Profile 4: attack with a single hole of diameter 0.33 mm and symmetric Rx cut-off. Here, severing electrodes Tx2, Rx1, and Rx2.

Attack Profile 5 (P5): Single Hole of 0.33 mm, Symmetric Tx Cut-Off. For this attack, we continued with the cover used in P1. To do so, we hit the previous 0.3 mm hole with our 0.33 mm drill bit. This caused the additional destruction of Tx7, creating a symmetric cut-off with Tx8. The resulting change in capacitance of Figure 13.19 should be compared to Figure 13.15 of P1. It is interesting to see that the previously assumed damage of Rx1 is now mostly gone in addition to the observed global shift in the values. However, the damage in Rx2 remains, as expected from the result of the failed integrity check. Moreover, while the damage in the Tx group was significantly lowered from more

than 10 000 points to slightly less than ~ 4 000, it is still present, clearly indicating an attack. Taking the results of this attack and previous attack profiles into account, it is highly improbable to succeed in attacking the device without doing attempted repairs.

An attacker may still want to aim for symmetric Tx cut-offs to minimize the effects due to imbalanced Tx pairs. However, when aligning these results with the absolute capacitance measurement of Figure 13.19b, it is evident that the attack would have been detected both by the differential and absolute capacitance measurement. Hence, the absolute capacitance measurement provides additional assurance to detect attacks that aim at tricking the behavior of the differential measurement.



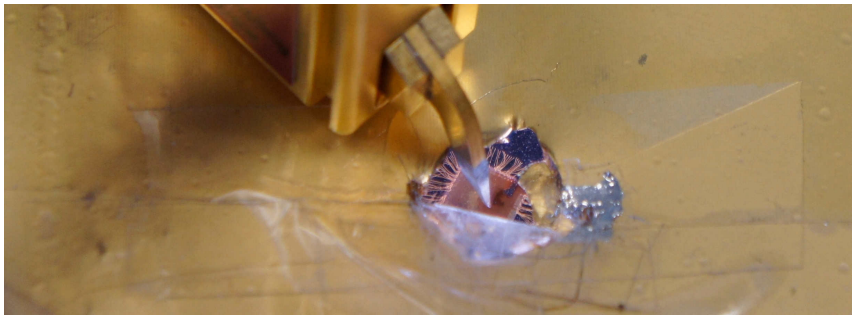
(a) Change in differential capacitance (P5). **(b)** Change in absolute capacitance (P5).

Figure 13.19: Attack Profile 5 (P5): result of a single hole of 0.33 mm in diameter, severing electrode Tx7, Tx8, and Rx2. Due to having a single hole is the cut-off of Tx7 and Tx8 considered symmetric.

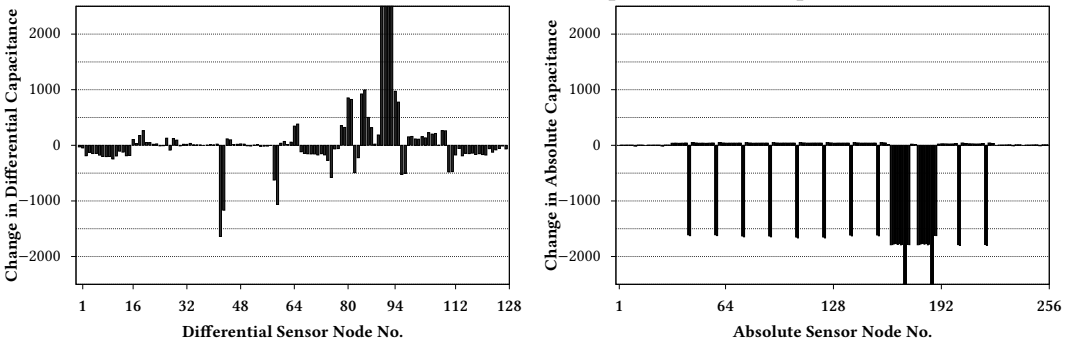
Attack Profile 6 (P6): Advanced Attack with Attempted Repair. As a next step, we push the concept to its limits by first drilling a hole with 5 mm and then simulating a real attack by means of analyzing the localized electromagnetic emanation (EM) of an IC as shown in Figure 13.20a. We chose the position for the hole such that the attacker would minimize the cut-off parts of the electrodes and at the same time, allow for the largest hole possible without exceeding a 2×2 node square. Moreover, we repaired the damage caused by the attack by reconnecting the severed electrodes, namely Tx11, Tx12, Rx11, and Rx12 using ultra-thin copper wire. A larger hole would have affected more electrodes and make this attack more complex in terms of repair.

To account for attackers exceeding our own capabilities and to simulate tasks we consider practically extremely challenging, we simplified the following steps as part of the attack. Prior to mounting the cover and carrying out the attack, the IC was decapsulated. No heatsink was mounted such that between the drilled hole and the IC no material had to be removed. While the repair of the affected Tx electrodes was done from the outside, we reconnected the broken Rx electrodes *on the inside* prior to mounting the cover. Since the finalized assembly prevents a non-destructive cover removal this is a noticeable simplification to not consider the effort required of reaching the Rx layer through the Tx layer and performing a miniature repair. Alternatively, a hole would need to be made to pull the bottom layer of the cover outwards and do the same (without breaking the remainder of the electrodes).

The resulting differential capacitance is shown in Figure 13.20b. While the damage is quite significant, it can be seen also that it is not as devastating due to the repairs. Still, a total of 18 nodes would have been destroyed, i.e., exceeding the threshold of the subsequent ECC scheme and causing a total of $18 \cdot 3.46\,\text{bit} = 62\,\text{bits}$ of *min*-entropy to be destroyed. While



(a) Photo of the advanced attack with field probe above decapsulated IC.



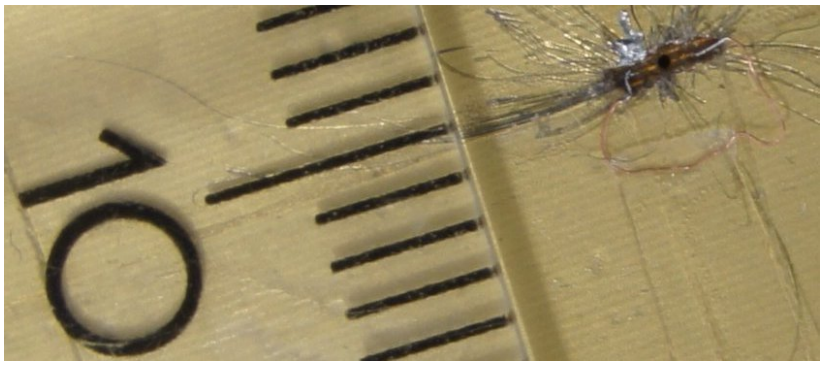
(b) Change in differential capacitance (P6). **(c)** Change in absolute capacitance (P6).

Figure 13.20: Attack Profile 6 (P6): Using drill of 5 mm with subsequent Tx *and* Rx repair.

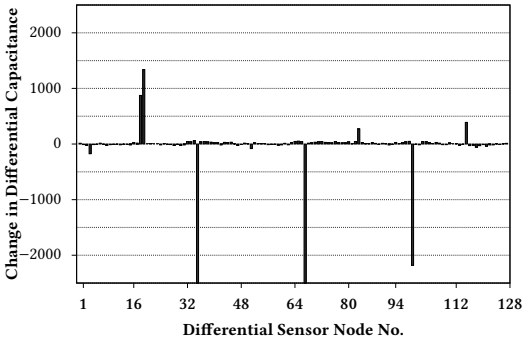
the loss in entropy drops to a level that is no longer considered computationally infeasible, we need to emphasize that the practical complexity of carrying out the attack in addition to the computational effort is still high, especially when considering the corresponding amount of Shannon entropy. Moreover, there is no doubt that based on the results of the absolute capacitance measurement as shown in Figure 13.20c would raise an alarm, too.

Attack Profile 7 (P7): Advanced Attack with Attempted Repair. We performed another advanced attack by testing the limits of this concept with holes of 300 µm in diameter and attempted repairs. The corresponding attack is shown in Figure 13.21a. As stated beforehand, we are of the opinion that compromising the enclosed system by making one hole only is not practically feasible due to the complex IC-level checks made. Instead, multiple such holes would need to be made at several strategic positions, necessitating more rework which in turn increases the likelihood for an attacker to make mistakes.

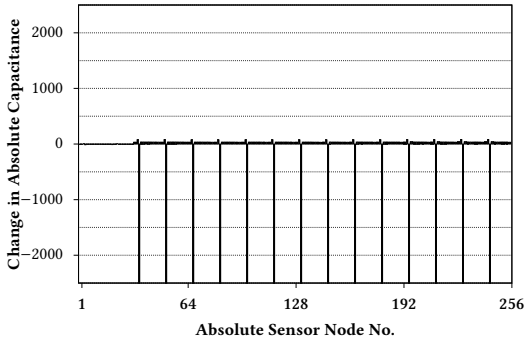
The drilled hole of 300 µm in diameter destroyed the integrity of Tx3 and Rx4 as result of the attack (before the repair). Figure 13.21b presents the change in differential capacitance from before the attack to after the attack including the attempted repair. Clearly visible is that the imbalance in the Tx pair due to the repair is insufficient to cause a shift in the values across the group. What remains is the Rx damage in all Tx excitation groups. To take advantage of the specific behavior of such attacks which we derived from previous analyses, we chose the specific location for the attack based on our knowledge of the actual values. Still, a total of 8 nodes would have moved away from their designated values, allowing for the attack to be detected but no longer representing an effort considered computationally infeasible, assuming the attacker would be able to obtain the measurement data just by using this hole alone. While we are unaware of how such a small hole with attempted repair could be used to compromise the underlying system, we fairly show the limits of



(a) 300 µm hole and attempted repair. Same ruler as in Figure 13.13c as reference (ticks in mm). Please note the disproportion of the hole's diameter vs. the overall size of the cover.



(b) Change in differential capacitance (P7).



(c) Change in absolute capacitance (P7).

Figure 13.21: Attack Profile 7 (P7): Using drill of 300 µm with subsequent repair.

our concept when using commercially available manufacturing technology only, i.e., a customized technology limiting the repairability of holes will help mitigate the risk of such attacks, e.g., by doping the carrier substrate with randomized dielectric particles and/or customized material for the electrode tracks [206, 160].

When considering the results of the absolute capacitance measurement in Figure 13.21c, we again see a striking difference in the capacitive behavior, allowing the detection of the attack. This emphasizes the importance of combining different measurement principles to make physical attacks more difficult to perform.

Conclusions on Attack Profiles. We practically and fairly evaluated the security of the cover based on the assumed attacker model under various drilling attacks including attempted repairs. The overall result is that attacks without attempted repairs are detected with very high probability. By carrying out more advanced attacks with attempted repairs while allowing some simplifications to be made, we have also openly shown the limits of the concept that cannot be fully overcome without more advanced manufacturing technology for the enclosure. Still, the combined use of differential and absolute capacitance measurement is a promising approach to detect a majority of physical intruders even when repairs are attempted. Moreover, during our white-box testing, we could disable countermeasures at will and focus on effects seen in the measurement data. In other situations this was also helpful, e.g., when reconnecting electrodes, as this is a laborious task and alignment errors are easily made such that the wrong electrodes would be mistakenly connected. Since the PUF data acquisition and tamper detection is done in a complex IC, disabling the detection logic while not destroying more entropy appears challenging.

Non-Invasive Attacks: Optical Inspection and Probing

One of the other possible threats of PUF-based enclosures is that an attacker may learn the PUF by means of optical inspection, i.e., contactless techniques that are non-invasive and therefore impossible to detect after attempted use when the device is powered on. As part of a more detailed analysis, we studied drill holes with the help of a Shimadzu SMX 6000 scanning system which is intended for PCB failure analysis and allows 2D and 3D X-ray imaging. The resulting 2D X-ray image of a drill hole with 200 µm and its surrounding mesh is shown in Figure 13.22.

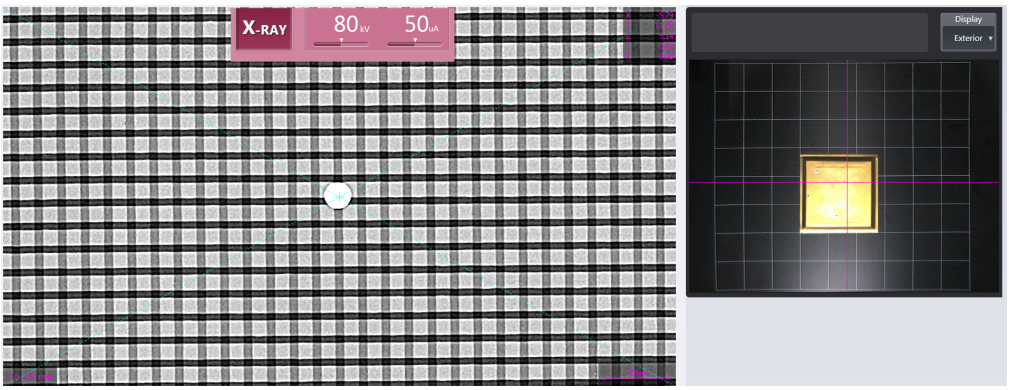


Figure 13.22: X-ray based two-dimensional (2D) optical inspection of cover with 200 µm hole.

It was *not* necessary to remove the cover's shield, i.e., it is possible to see through the

solid copper plane. The same applies when considering the resulting 3D X-ray image as shown in Figure 13.23. Neither in 2D, nor in 3D, it is possible to identify locations from which specific information on the PUF could be derived, i.e., other than a highly regular structure of the mesh there is no revealing information visible. This is within the scope of our expectation due to the following reasons:

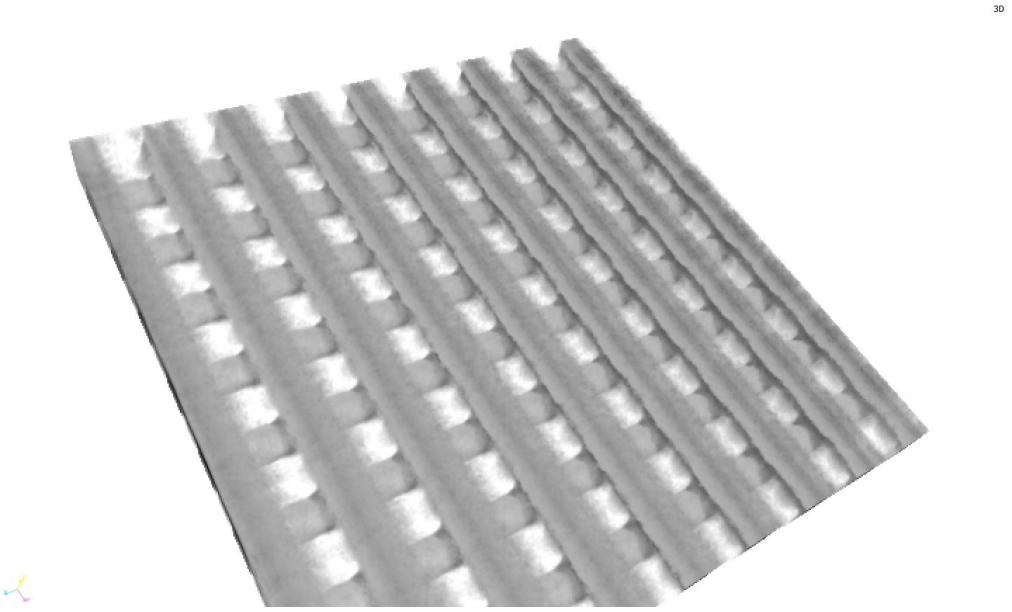


Figure 13.23: X-ray based three-dimensional (3D) optical inspection of mesh.

- For the 2D case, the obtained image is from a bird's eye view, i.e., the 3D structure of the fuzzy edges of the PCB tracks cannot be resolved. Likewise it is not possible to analyze the surface roughness in between the tracks from the outside, even for the 3D case, at least with the imaging technology we had at hand.
- While the 3D structure of the mesh becomes visible under 3D imaging technology, we still could not derive useful information from these images about the PUF values.
- Assuming the PUF deviation could be observed to a certain degree, it is still mandatory to look at the accumulated deviation over all sensor cells per node, i.e., an automated tool would need to extract the deviation per sensor cell which entails a certain error due to limited resolution, etc. This error accumulates over the sum of all cells per node and would severely falsify the obtained value.
- Upon manual inspection of the images, there are no obvious patterns or marks visible (aside from manufacturing defects) that would justify further analysis with regard to optical inspection.

Other optical attacks include Laser Voltage Probing (LVP), as for example used in [238]. To the best of our knowledge is this technique designed for IC analysis only, as it requires a p-n junction to work correctly. Moreover, it is beyond our own expertise if a *current* (as opposed to voltage) signal in the lower nanoampere range could be optically probed. We are currently unaware of other analysis techniques in this domain that could help to optically probe signals on bare tracks inside the flexPCB.

Discussion of Additional Attacks

Since it is not possible to exhaustively cover all possible attacks, let us briefly consider a selection of other attacks and how they have been considered in the design. Note that some attacks require additional countermeasures which are outside the scope of the cover itself, e.g., preventing data remanence or having a sufficiently internally buffered supply to enable zeroization even if an attacker pulls the power during runtime.

Bending/Prying Open the Cover. In general, there are two types of flexPCB offered. One type is for static flexing, i.e., a one-time bending to fit the flexPCB in the packaging design. When targeting this application, it is common to choose an adhesiveless carrier, i.e., the same we use. In contrast, for dynamic flexing where the flexPCB must be bent multiple times as part of the functionality, it is common to choose carriers with flexible adhesives to minimize strain when bending the flexPCB. Since our flexPCB is intended for one-time bending and has been manufactured correspondingly, it is difficult to not create cracks when bending it in reverse direction of the previous assembly process. As prying open the cover causes severe mechanical stress, either breaking it or creating cracks in the copper tracks. Moreover, without X-rays, such cracks cannot be located through the solid copper plane of the shield which makes it difficult to repair them, too.

Careful Cover Disassembly and Measurement with Attacker's Circuit. The goal of this attack would be to extract the cover's PUF key without the actual device, i.e., to carefully disassemble it without destroying the PUF behavior. Since the packaging concept including its potting have been specifically designed to thwart such attempts of an easy cover removal, it is not possible to remove the cover without severely damaging it. The whole unit has *not* been designed to allow servicing of its components, even by its legitimate device owner.

Assuming the cover could be removed, the attacker would still need to replicate the measurement circuit with utmost care. Due to the specifics of the electrode setup, e.g., its massively parallel structure, disproportion of different capacitances contributing to the measurement, and the small-scale differential capacitance, it is highly unlikely to use a standard LCR-meter to carry out the measurement of C^{V} in a useful way. In a certification process, this would add to the complexity of the attack even despite the fact that this is not theoretically impossible.

Imposter Attack. The goal of this attack would be an undetected disassembly, successful tampering with an IC on the inside, and re-assembly. Due to the same reasons stated above, we consider cover disassembly not as a well-founded choice for the attacker. In addition to these difficulties related to that would an imposter attack imply that an attacker has not only been able to secretly circumvent all countermeasures that are checked by the device itself but also tamper-evident properties that are visible to the human eye. For example, optical inspection of the unit prior to putting it in the field would notice differences in the particle-mix of the potting, possible damage, etc.

"Frankenstein" Attack. Since our laser intended for IC failure analysis could not be used to cut or drill flexPCB material, we could not carry out attacks where pieces of one flexPCB cover are used to repair damage done to another flexPCB cover. This requires a precision setup to not violate the underlying design rules of the electrodes, i.e., a matched cell-overlap of differential electrode pairs. We point out that cutting and putting back pieces of flexPCB material entails a significant amount of work for reconnecting each of the cut lines, adding to the complexity of the attack the larger the piece is. A better approach would be knowing the size of the piece targeted for removal and to manufacture

a corresponding piece where the wiring is done internally, such that only the outside connectors would need to be reconnected appropriately, thereby reducing the work of reconnecting lines. However, our findings for batches from two different manufacturers and even batch-to-batch differences indicate that there will still be noticeable differences in the PUF behavior, making this attack still reasonably difficult to perform.

Physically Probing Electrodes. An attacker might try to probe electrodes directly to measure their capacitance or eavesdrop signals. This requires access to all electrodes, as properly connecting unused ones is mandatory for the measurement. This claim is not only supported by our practical experience but also the plots presented as part of the attack profiles as unconnected parts of an electrode degrade the measurement. At the same time, the shield would need to be partially removed at multiple spots, causing the surrounding field to change, thereby falsifying the results. Repeatedly carrying out these steps without making errors along the way is considered challenging. Moreover, even state-of-the-art micro probes [54] add a capacitive load of $> 20\,\mathrm{fF}$ which exceeds the observed standard deviation in differential capacitance. Customized circuitry to investigate the feasibility of such an attack has been developed by Johannes Obermaier and corresponding results can be found in his dissertation.

Side-Channel Attacks. Emanations of the system are prevented by the heatsink, shielding layers, and the supply lines are additionally protected with filters. Moreover, the Tx layer carries only insensitive excitation signals, i.e., the attacker would only see the 33.3 kHz of the excitation signal without the possibility to derive useful information from it. In contrast, the Rx layer carries sensitive signals in the lower nano-ampere range, making it difficult to eavesdrop on them. The measurement itself is otherwise time-constant.

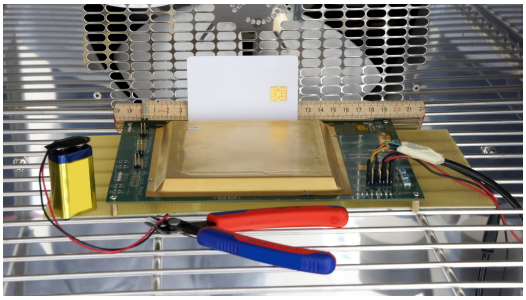
13.2.4 Environmental Tests

To analyze the robustness of our approach, we carried out tests in the temperature range of $-20\,^{\circ}\mathrm{C}$ to $+60\,^{\circ}\mathrm{C}$ using a VT 4011 temperature chamber by Vötsch as illustrated in Figure 13.24a. We tested this with a single board and three top covers, i.e., the assembly was not finalized and no potting was used to enable the measurement of different covers using the same circuit. Overall, we observed a highly similar behavior for the covers.

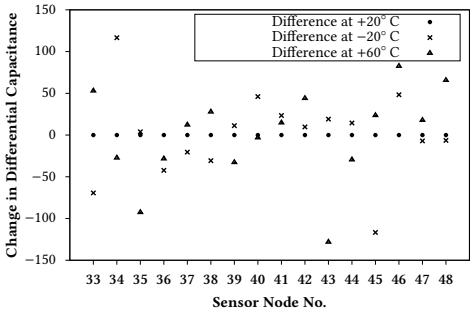
When both cover and measurement circuit are subject to these environmental influences this causes a certain temperature drift in the values as shown in Figure 13.24c for the absolute capacitance measurement. The plateau regions illustrate the differences in temperature with steps of $10\,^{\circ}\mathrm{C}$. Clearly visible is the direct relation of temperature to change in value and that the spread of values relative to the overall mean per sample point in time is relatively constant. In fact, the absolute capacitance measurement could be exploited as a coarse-grained temperature sensor for Environmental Failure Protection (EFP), too.

This behavior is incomparable to the raw differential capacitance prior to compensation, as shown in Figure 13.24d. Here, we see a much weaker pattern from the temperature cycle which is only barely visible. Moreover, as the differential nodes have different values, they behave slightly different. For a constant temperature level, the lines would be going straight from left to right. Here, we do see that larger differential capacitances tend to have a larger drift when compared to smaller differential capacitances that are apparently less affected by temperature. For a representative group of values with larger and smaller capacitances which is based on one Tx pair, the maximum drift d_{E} after compensation is less than 130 points as illustrated in Figure 13.24b.

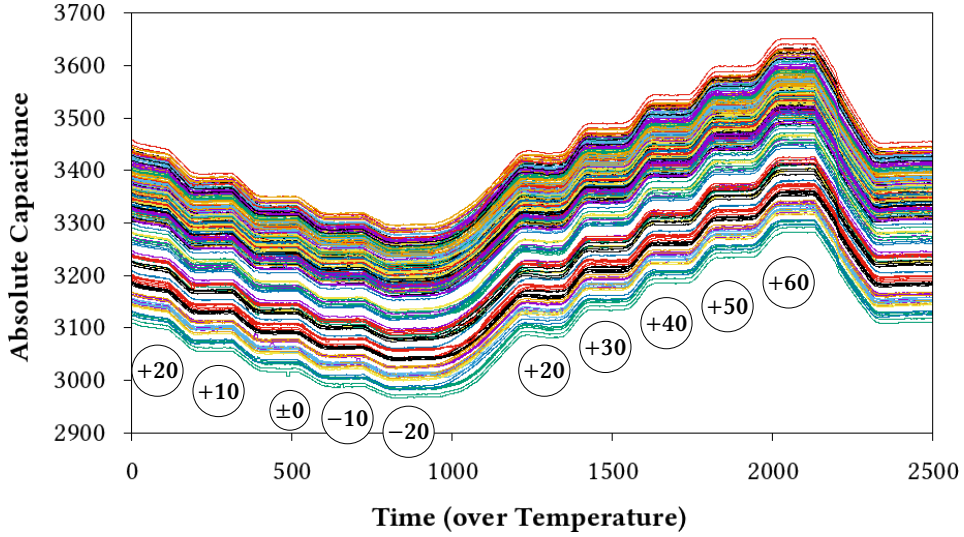
To counteract this remaining drift effect, we need to lower the number of quantization



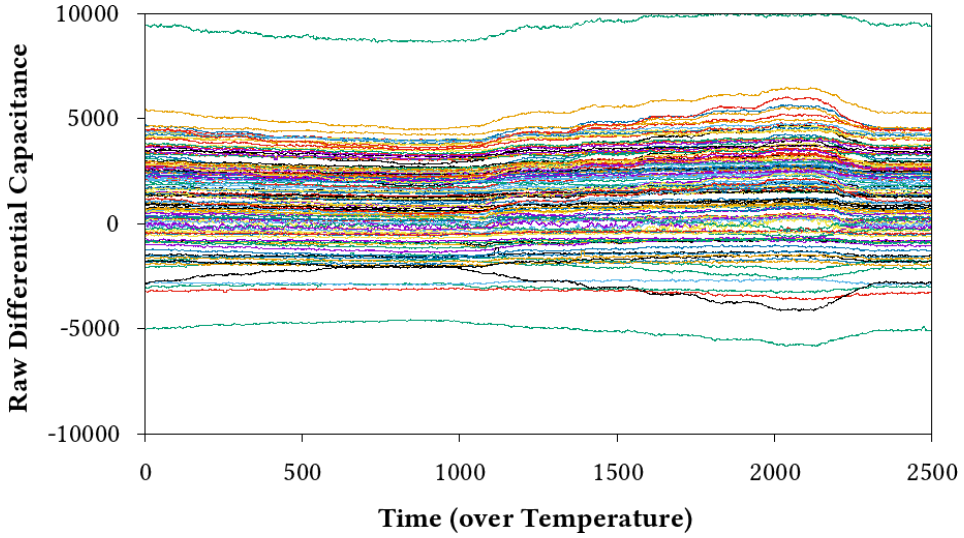
(a) Temperature chamber.



(b) Compensated differential capacitance.



(c) Absolute capacitance over time and temperature for nodes of a cover.



(d) *Raw* (unprocessed) differential capacitance over time and temperature.

Figure 13.24: Environmental tests and results. Plots in Figure 13.24c and Figure 13.24d have the identical time axis, i.e., they both cover the temperature range from +20 °C to −20 °C, then to +60 °C, and back to room temperature during the same test cycle.

intervals, i.e., increase their width to $Q_w = 2 \cdot y \cdot \sigma_{N,\mathrm{Diff},10} + 2 \cdot d_E$. Accordingly, for $y = 3.29$, the number of quantization intervals is reduced to 40 while the Shannon entropy drops to 4.17 bit per node. Even for drifts that are much higher, e.g., up to $d_E = 400$ points, we conveniently stay above 3 bit of Shannon entropy per node. Without implementing more advanced compensating techniques, the temperature drift is fully accounted for by the increased width of the quantization interval. Hence, there is no need to improve the error-correcting capability of the subsequent ECC scheme. As the quantization interval width is only ~ 500 points (based on 40 intervals), it is still possible to reliably detect the damage of the physical attacks as presented beforehand. Erroneous differential nodes as result from an erratic behavior under temperature effects is typically less than three to five nodes such that a sufficient gap is ensured to destroyed nodes from physical attacks. Hence, an attacker would try to attack the system at the temperature of enrollment to exploit the ECC to possibly correct damage made by the attack.

Aging. We also performed tests for accelerated aging of the foils, i.e., heating up to +110 °C for drying at a relative humidity of $< 10\%$, then exposing the covers to +90 °C at a relative humidity of 85% with another drying cycle afterwards. This procedure was repeated several times. In between each step, we measured the values to determine their behavior, i.e., the measurement circuit was *not* subject to this accelerated aging to assess the properties of the covers independently of a possible aging in circuit components. After this test, the majority of values returned to their designated values of the enrollment with very small error margin (typically much less than 30 points). This is not unexpected, since flexPCB is typically rated for much worse conditions. The only nodes with critical behavior were located in the flaps, as they were not mechanically secured by a conformal coating or potting for our tests, resulting in mechanical stress due to expansion of the material. This is owed to the fact that for the purpose of measuring the covers, we needed to mount and unmount the covers which would not have been possible when finalizing their assembly, i.e., applying the potting and securing the seams would have prevented this. In the future, a measurement IC is developed which is why the aging behavior of the chosen COTS components was not of relevance. We additionally point out that for aging, it is always an option to re-enroll the device in the field if necessary.

13.2.5 Conclusions and Outlook

Here, we analyzed how to enclose a device with a cover that is evaluated using a batteryless security concept while still detecting a majority of physical intruders. We implemented our proposed full-stack approach and experimentally verified the PUF-behavior based on the statistical measurements of 115 covers. In addition to the work for B-TREPID, we implemented a "full scope" measurement by means of a differential *and* absolute capacitance measurement which makes it practically impossible to tailor an attack that is able to trick both measurements at the same time.

Our comprehensive tests provide initial evidence that this concept fulfills the targeted requirements, i.e., statistical results in addition to attacks and environmental tests confirm the chosen design rationale. However, when comparing our academic study with previous industrial solutions, it is evident that our material properties should be further improved to provide an even higher level of security by making attempted repairs more difficult. This is difficult to achieve within a standard flexPCB manufacturing process.

Moreover, a layout randomization is currently not implemented, due to the limitation of using COTS components for the measurement circuit based on discrete components.

Further improvements could be the measurement from both sides of the electrodes. Hence, our results clearly facilitate future research as the presented concepts are generic and do not depend on the chosen manufacturing technology or circuit implementation. Hence, it should be considered as a hint of what could be achieved with different manufacturing technologies such as panel level integration [154] or more advanced manufacturing technologies with custom tailored materials for either the tracks [160] or the carrier materials. Furthermore, we plan to update the physical design such that the outcome results in a bimodal distribution, i.e., a double-peaked PDF with a local minimum in the center which is aligned with the value 0 of the differential measurement. This has the benefit of increased value shift upon attacks and consequently makes them more difficult to perform.

Part VI

Conclusion

Chapter 14

Conclusion and Future Work

The following sections conclude this thesis and provide a small assessment of what has been achieved. In addition to that, specific ideas for future work are presented.

Contents

14.1 Conclusion

This work is one of the very few attempts on the topic of creating a system-level, tamper-evident PUF. As such, only its surface could be scratched (pun intended). It is evident that replacing former battery-backed security enclosures by solutions not requiring a battery is a challenging task requiring additional research in the future.

Background information on these former solutions was presented in Chapter 2. While previous solutions have been successfully used in the past, sometimes even for decades based on a single technology, advancements in the domain of imaging technology and the lack of prospective updates in the manufacturing technology w.r.t. silk-screen printing have made these previous solutions obsolete. This is in addition to the disadvantages of a battery-backed monitoring concept.

Afterwards, we surveyed PUF constructions and identified the potential of a PUF concept exceeding previous constructions, namely HOA PUFs that fully leverage the entropy contained in the raw PDF. As part of that, equidistant quantization has been favored as a first building block to improve tamper-sensitivity of the resulting implementation. Mainly two different classes of ECCs have been investigated. One is based on a variable-length bit mapping of the symbols and VT-like codes. The other is based on LMCs and continues operating on symbols directly. The latter has proven to be superior in all aspects relevant for the key derivation process, i.e., entropy, reliability, and tamper-sensitivity.

Due to the fact that the data processing is fundamentally different to other PUFs, we had to extend existing metrics to properly assess the PUF properties. This included updated definitions of Uniqueness and Reliability, the two most well-known performance metrics for PUFs.

Afterwards, PoC implementations were presented to substantiate the chosen design rationale. In addition, a brief example was proposed how the previous concepts could be incorporated into the secure boot process of the device.

Overall, this thesis investigated the full-stack of a prospective tamper-evident PUF, ranging from its physical representation, over analog measurement, to digital data processing, and subsequent application domain. As result, the concept of HOA PUFs was created and preliminary implementations of a tamper-resistant envelope and a pair of tamper-resistant covers that may serve as a cornerstone of follow-up developments to ultimately replace battery-backed enclosures. While a large range of topics could be addressed, it is evident that this thesis was carried out within the scope of two specific projects, i.e., limited in time and funding. Hence, it was not possible to investigate each aspect in full detail. In the following, several topics are addressed that may be of interest in the future to complement this work.

14.2 Future Work

Considering our previous analyses and related work, we identified several open challenges. This scope for future work is briefly outlined in the following.

Designated Type of Distribution. In this thesis, based on the assumed statistical model, the designated outcome of the PUF was a normal distribution. However, some of the processing steps regarding normalization and temperature compensation interfere with the differential measurement, the default outcome of a normal distribution, and result in a degradation in tamper-sensitivity. As an alternative, a bimodal distribution should be investigated, i.e., a double-peaked PDF where the zero-value of the differential measurement aligns with the middle between the two peaks. This has the benefit of improved value-shift upon tampering and leads to a decision problem for the attacker, as the most likely values are further separated from each other. While this may be difficult to achieve as part of the implementation itself, it could also be possible to implement a physical random bias unit within the measurement circuit. This unit would then shift the results on a physical level such that the targeted distribution is obtained. While this is not a perfect solution and should be considered an obfuscation technique, it would represent another practical challenge for the attacker to overcome it, i.e., only attacking the enclosure would no longer be possible but the values of the physical random bias unit would have to be extracted, too. This somewhat reflects the idea commonly seen in key agreement protocols that no single party should be in control of the key generation. Here, this corresponds to the enclosure and circuit as parties, whereas neither one of them should be in control of the key.

Incorporating Absolute Measurement Values. Here, we mostly focused on the differential capacitance values as they are the primary source of entropy in the system. However, based on the results of Section 13.2, the importance of the absolute measurement values was shown. Unfortunately, directly including these values into the key generation process is not an idea based on a proper theoretical reasoning. Again, obfuscation techniques could be used to help strengthen this part of the implementation. For example, by impregnating a key into the absolute measurement values by having selectable (or even tunable) matched offsets for C^N at the time of manufacturing the circuit/enclosure. Hence, the result would be that for a differential measurement, nothing would have changed, but for an absolute measurement, the impregnated values could be measured. Again, we point out that this is only an obfuscation technique that may not be necessarily practically feasible in an actual implementation.

Ratio of Variation in Values vs. Nominal Component. For the PoC implementations, we could not investigate how doping a printed dielectric would turn out. As indicated beforehand, reparability and possibility to probe electrode tracks without causing damage is most essential for this type of application. Unfortunately, there was no opportunity to investigate this line of work which is why it must be considered in more detail in the future. Eventually, a careful trade-off must be found, as a greater C^{V} improves complexity of local repairs, while possibly limiting the propagation effect due to having a less significant impact of a smaller C^{N}.

PUF-adhesives and Securing Seams. Two fundamentally different enclosure concepts have been investigated, namely envelopes and covers. The latter suffers from the limitation of how to securely bond it to a PCB such that prying it open is sufficiently challenging and also causes destruction of the PUF. Ideally, securing these seams could be done with a PUF that is both part of the cover and the PCB. While preliminary ideas have been conceived by the thesis author how this could be done, it is evident that this requires a different approach when compared to the previous PUF concepts and thus, opens up a new line of work.

Improved Data Processing. To further improve robustness towards environmental effects, the optimal combination of processing steps still needs to be determined. For example, the specifics of a differential approach based on either ratio or difference need to be investigated more carefully, in addition to proper normalization and drift compensation. Ideally, a normalization can be avoided completely, as some of the most straightforward processing steps, such as subtracting a group-wise average, impedes the tamper-sensitivity of the system.

Smarter Materials, More Scalable, and Cheaper Solutions. Since all discussed solutions have been designed to meet the highest security levels, the cost of such a solution was not the primary concern. However, this hinders adoption in a wider range of products and therefore results in an overall lower security level. Ideally, smarter materials with a strong physical avalanche effect are developed that cause a propagation effect upon tampering, such that the contained entropy is thoroughly destroyed, while at the same time, being scalable and considerably cheap.

Part VII

Appendix

Codebooks of Key Derivation Profiles

Since some of the *min*-TS and *max*-TS results depend on the specific codebook chosen to carry out the mapping of symbols to bits, we provide this necessary information to replicate our results. Unfortunately, no systematic could be found on how to construct the codebook for VT-like codes that are relevant for Profile 4. All codebooks represent the assigned values to the quantization intervals from "left to right" (cf. Figure 5.3).

Codebook of Profile 4 [92] and $|\mathcal{L}| = 12$

$\mathcal{L} = [0110; 0111; 0011; 0010; 000; 010; 110; 111; 1011; 1010; 1000; 1001]$
The maximum magnitude shift while ensuring $\mathrm{d}_{\mathrm{Lev}}(Y, \hat{Y}) = 1$ may occur for the following values, all of which describe a shift by 6 quantization intervals:

- $0110 \leftrightarrow 110$ (insertion/deletion of 0)
- $0111 \leftrightarrow 111$ (insertion/deletion of 0)
- $000 \leftrightarrow 1000$ (insertion/deletion of 1)
- $0011 \leftrightarrow 1011$ (substitution of 0/1 in left most position)
- $0010 \leftrightarrow 1010$ (substitution of 0/1 in left most position)

Codebook of Profile 4 [92] and $|\mathcal{L}| = 14$

$\mathcal{L} = [01100; 01101; 0111; 0011; 0010; 000; 010; 110; 111; 1011; 1010; 1000; 10010; 10011]$
The maximum magnitude shift while ensuring $\mathrm{d}_{\mathrm{Lev}}(Y, \hat{Y}) = 1$ may occur for the following values which describe a shift by 10 quantization intervals:

- $0011 \leftrightarrow 10011$ (insertion/deletion of 1)

Codebook of Profile 5 [206] and $|\mathcal{L}| = 8$

$\mathcal{L} = [000; 001; 011; 010; 110; 111; 101; 100]$
Larger magnitude shifts exceeding the range of one quantization interval while still ensuring $\mathrm{d}_{\mathrm{H}|2}(Y, \hat{Y}) = 1$ may occur for the following values:

- $000 \leftrightarrow 010$ (shift by 3 quantization intervals of unequal size)
- $011 \leftrightarrow 111$ (shift by 3 quantization intervals of unequal size)
- $110 \leftrightarrow 100$ (shift by 3 quantization intervals of unequal size)
- $001 \leftrightarrow 101$ (shift by 5 quantization intervals of unequal size)
- $000 \leftrightarrow 100$ (shift is across the full range of values)

List of Figures

List of Tables

List of Algorithms

Bibliography

[1] J. Aarestad, J. Plusquellic, and D. Acharyya. "Error-Tolerant Bit Generation Techniques for Use with a Hardware-Embedded Path Delay PUF." In: *2013 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*. June 2013, pp. 151–158.

[2] *About Us | DoD Anti-Tamper Executive Agent*. https://at.dod.mil/content/about-us.

[3] Dennis G. Abraham, George M. Dolan, Glen P. Double, and James V. Stevens. "Transaction Security System." In: *IBM Systems Journal* 30.2 (1991), pp. 206–229.

[4] Benjamin R. Anderson, Ray Gunawidjaja, and Hergen Eilers. "Initial Tamper Tests of Novel Tamper-Indicating Optical Physical Unclonable Functions." In: *Applied Optics* 56.10 (Apr. 1, 2017), pp. 2863–2872.

[5] *Anti-Tamper Capabilities in FPGA Designs*. White Paper WP-01066-1.0. Altera, July 2008, pp. 1–9.

[6] *Anti-Tamper Technology: Safeguarding Today's COTS Platforms*. White Paper A-WP-865A. Abaco Systems, Feb. 2014, pp. 1–9.

[7] Frederik Armknecht, Roel Maes, Ahmad-Reza Sadeghi, Francois-Xavier Standaert, and Christian Wachsmann. "A Formalization of the Security Features of Physical Functions." In: *IEEE Symposium on Security and Privacy (S&P)*. 2011, pp. 397–412.

[8] Frederik Armknecht, Daisuke Moriyama, Ahmad-Reza Sadeghi, and Moti Yung. "Towards a Unified Security Model for Physically Unclonable Functions." In: *Topics in Cryptology - CT-RSA 2016 - The Cryptographers' Track at the RSA Conference 2016, San Francisco, CA, USA, February 29 - March 4, 2016, Proceedings*. 2016, pp. 271–287.

[9] Lukas Auer. "Verification Is Power: Secure Bootstrap of Tamper-Protected Devices." A master's thesis advised by Vincent Immler. Munich: Technical University Munich, 2017.

[10] M. Barbareschi, G. Di Natale, L. Torres, and A. Mazzeo. "A Ring Oscillator-Based Identification Mechanism Immune to Aging and External Working Conditions." In: *IEEE Transactions on Circuits and Systems I: Regular Papers* 65.2 (Feb. 2018), pp. 700–711.

[11] Daniel Becker. "Analysis of Spatial Entropy of Higher-Order Alphabet PUFs." A bachelor's thesis advised by Vincent Immler. Bochum: Ruhr-University Bochum, 2018.

[12] Justin H. Benson, John I. Daspit, and Charles McCown. "Security Module System, Apparatus and Process." U.S. pat. 7054162B2. SafeNet Inc. May 30, 2006.

[13] M. Bhargava, C. Cakir, and K. Mai. "Reliability Enhancement of Bi-Stable PUFs in 65nm Bulk CMOS." In: *2012 IEEE International Symposium on Hardware-Oriented Security and Trust*. June 2012, pp. 25–30.

[14] M. Bhargava and K. Mai. "An Efficient Reliable PUF-Based Cryptographic Key Generator in 65nm CMOS." In: *2014 Design, Automation Test in Europe Conference Exhibition (DATE)*. Mar. 2014, pp. 1–6.

[15] Mudit Bhargava and Ken Mai. "A High Reliability PUF Using Hot Carrier Injection Based Response Reinforcement." In: *Cryptographic Hardware and Embedded Systems - CHES 2013*. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, Aug. 20, 2013, pp. 90–106.

[16] Nisarga Bhargavi and Eric Peeters. *System-Level Tamper Protection Using MSP MCUs*. Application Report SLAA715. Texas Instruments, Aug. 2016, pp. 1–13.

[17] Christoph Böhm and Maximilian Hofer. "Two Stage PUF." In: *Physical Unclonable Functions in Theory and Practice*. Ed. by Christoph Böhm and Maximilian Hofer. New York, NY: Springer New York, 2013, pp. 221–226.

[18] Christoph Bösch, Jorge Guajardo, Ahmad-Reza Sadeghi, Jamshid Shokrollahi, and Pim Tuyls. "Efficient Helper Data Key Extractor on FPGAs." In: *Workshop on Cryptographic Hardware and Embedded Systems (CHES)*. Ed. by Elisabeth Oswald and Pankaj Rohatgi. Vol. 5154. LNCS. Springer Berlin / Heidelberg, 2008, pp. 181–197.

[19] L. Bossuet, X. T. Ngo, Z. Cherif, and V. Fischer. "A PUF Based on a Transient Effect Ring Oscillator and Insensitive to Locking Phenomenon." In: *IEEE Transactions on Emerging Topics in Computing* 2.1 (Mar. 2014), pp. 30–36.

[20] BOURNS INC. "Application Note – Security Housing." In: (2007). http://application-notes.digchip.com/176/176-48205.pdf.

[21] Gary A. Brist. "Design Optimization of Single-Ended and Differential Impedance PCB Transmission Lines." In: *PCB West Conference Proceedings*. 2004.

[22] William L. Brodsky, John R. Dangler, Zachary T. Dreiss, David C. Long, Michael T. Peets, William Santiago-Fernandez, and Thomas Weiss. "Enclosure with Inner Tamper-Respondent Sensor(s)." U.S. pat. 9591776B1. International Business Machines Corp. Mar. 7, 2017.

[23] Barbara J. Brymer, Edward J. Kapp, and Frank Z. Keister. "Anti-Compromise Microelectronic Circuit." U.S. pat. 3860835A. US Secretary of Navy. Jan. 14, 1975.

[24] Ileana Buhan, Jeroen Doumen, Pieter Hartel, and Raymond Veldhuis. "Fuzzy Extractors for Continuous Distributions." In: *Proceedings of the 2Nd ACM Symposium on Information, Computer and Communications Security*. ASIACCS '07. New York, NY, USA: ACM, 2007, pp. 353–355.

[25] Ray Burke and Karl Queen. "A Security Enclosure for a Circuit." European pat. 1462907A1. Bourns Inc. Sept. 29, 2004.

[26] Richard F. Carson and Stephen A. Casalnuovo. "Integrated Optical Tamper Sensor with Planar Waveguide." U.S. pat. 5177352A. US Department of Energy. Jan. 5, 1993.

[27] Mario Leonardo Cesana and Roberto Antonio Zavatti. "Tamper Resistant Card Enclosure with Improved Intrusion Detection Circuit." U.S. pat. 6957345B2. International Business Machines Corp. Oct. 18, 2005.

[28] David Chaum. "Design Concepts for Tamper Responding Systems." In: *Advances in Cryptology*. Springer, Boston, MA, 1984, pp. 387–392.

[29] W. Che, F. Saqib, and J. Plusquellic. “Novel Offset Techniques for Improving Bitstring Quality of a Hardware-Embedded Delay PUF.” In: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 26.4 (Apr. 2018), pp. 733–743.

[30] J. Chung-yaw Chiang and Jack K. Wolf. “On Channels and Codes for the Lee Metric.” In: *Information and Control* 19.2 (Sept. 1, 1971), pp. 159–173.

[31] Andrew J. Clark. “Physical Protection of Cryptographic Devices.” In: *Advances in Cryptology — EUROCRYPT’ 87*. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, Apr. 13, 1987, pp. 83–93.

[32] B. Colombier, L. Bossuet, V. Fischer, and D. Hély. “Key Reconciliation Protocols for Error Correction of Silicon PUF Responses.” In: *IEEE Transactions on Information Forensics and Security* 12.8 (Aug. 2017), pp. 1988–2002.

[33] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Second. New York: John Wiley & Sons, 2006.

[34] G. I. Davida, Y. Frankel, and B. J. Matt. “On Enabling Secure Applications through Off-Line Biometric Identification.” In: *Proceedings. 1998 IEEE Symposium on Security and Privacy (Cat. No.98CB36186)*. May 1998, pp. 148–157.

[35] Jeroen Delvaux, Dawu Gu, Ingrid Verbauwhede, Matthias Hiller, and Meng-Day (Mandel) Yu. “Efficient Fuzzy Extraction of PUF-Induced Secrets: Theory and Applications.” In: *Cryptographic Hardware and Embedded Systems - CHES 2016 - 18th International Conference, Santa Barbara, CA, USA, August 17-19, 2016, Proceedings*. Ed. by Benedikt Gierlichs and Axel Y. Poschmann. Vol. 9813. Lecture Notes in Computer Science. Springer, 2016, pp. 412–431.

[36] Jeroen Delvaux and Ingrid Verbauwhede. “Key-Recovery Attacks on Various RO PUF Constructions via Helper Data Manipulation.” In: *Proceedings of the Conference on Design, Automation & Test in Europe*. DATE ’14. 3001 Leuven, Belgium, Belgium: European Design and Automation Association, 2014, 72:1–72:6.

[37] Yevgeniy Dodis, Leonid Reyzin, and Adam Smith. “Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data.” In: *Advances in Cryptology (EUROCRYPT)*. Ed. by Christian Cachin and Jan L. Camenisch. Vol. 3027. LNCS. Springer Berlin / Heidelberg, 2004, pp. 523–540.

[38] Glen P. Double and Steve H. Weingart. “Data Protection by Detection of Intrusion into Electronic Assemblies.” U.S. pat. 5159629A. International Business Machines Corp. Oct. 27, 1992.

[39] N. Elarief and B. Bose. “Optimal, Systematic,q-Ary Codes Correcting All Asymmetric and Symmetric Errors of Limited Magnitude.” In: *IEEE Transactions on Information Theory* 56.3 (Mar. 2010), pp. 979–983.

[40] H. Eren and L.D. Sandor. “Fringe-Effect Capacitive Proximity Sensors for Tamper Proof Enclosures.” In: *Sensors for Industry Conference*. 2005.

[41] Thomas Esbach, Walter Fumy, Olga Kulikovska, Dominik Merli, Dieter Schuster, and Frederic Stumpf. “A New Security Architecture for Smartcards Utilizing PUFs.” In: *ISSE Conference*. 2012.

[42] E. Ferres, V. Immler, A. Utz, A. Stanitzki, R. Lerch, and R. Kokozinski. “Capacitive Multi-Channel Security Sensor IC for Tamper-Resistant Enclosures.” In: *2018 IEEE SENSORS*. Oct. 2018, pp. 1–4.

[43] Markus Fischer and Günther Froschermeier. "Elektronik-Sicherheits-Modul." European pat. 1804557A1. EL-ME AKTIENGESELLSCHAFT, EL ME AG. July 4, 2007.

[44] B. Fleming. "Microcontroller Units in Automobiles [Automotive Electronics]." In: *IEEE Vehicular Technology Magazine* 6.3 (Sept. 2011), pp. 4–8.

[45] CA/Browser Forum. *Baseline Requirements for the Issuance and Management of Publicly-Trusted Certificates*. Oct. 14, 2018.

[46] Jörg Franke. *Three-Dimensional Molded Interconnect Devices (3D-MID): Materials, Manufacturing, Assembly and Applications for Injection Molded Circuit Carriers*. Carl Hanser Verlag GmbH Co KG, Apr. 3, 2014. 375 pp.

[47] Benjamin Fuller, Xianrui Meng, and Leonid Reyzin. "Computational Fuzzy Extractors." In: *Advances in Cryptology - ASIACRYPT 2013*. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, Dec. 1, 2013, pp. 174–193.

[48] Blaise Gassend. "Physical Random Functions." Massachusetts Institute of Technology, Jan. 2003.

[49] 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*. ACM, 2002, pp. 148–160.

[50] Stefan Gehrer. "Highly Efficient Implementation of Physical Unclonable Functions on FPGAs." Dissertation. München: Technische Universität München, 2017.

[51] M. Geis, K. Gettings, and M. Vai. "Optical Physical Unclonable Function." In: *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*. Aug. 2017, pp. 1248–1251.

[52] Johannes A. J. Van Geloven, Robertus A. M. Wolters, and Nynke Verhaech. "Sensing Circuit for Devices with Protective Coating." U.S. pat. 8138768B2. NXP BV. Mar. 20, 2012.

[53] Rosario Gennaro, Anna Lysyanskaya, Tal Malkin, Silvio Micali, and Tal Rabin. "Algorithmic Tamper-Proof (ATP) Security: Theoretical Foundations for Security against Hardware Tampering." In: *Theory of Cryptography Conference (TCC)*. Ed. by Moni Naor. 2004.

[54] GGB Industries Inc. "Picoprobe Model 19C." In: (2004).

[55] GGB Industries Inc. "T-4 Series Tungsten Probe Tips." In: (2004). Available online: http://www.ggb.com/t-4.html, as of October 10, 2016.

[56] Frank Gray. *Pulse Code Communication*. 1953.

[57] Joep de Groot, Boris Škorić, Niels de Vreede, and Jean-Paul Linnartz. *Quantization in Continuous-Source Zero Secrecy Leakage Helper Data Schemes*. 566. 2012.

[58] Joep de Groot, Boris Škorić, Niels de Vreede, and Jean-Paul Linnartz. "Quantization in Zero Leakage Helper Data Schemes." In: *EURASIP Journal on Advances in Signal Processing* 2016.1 (Dec. 1, 2016), p. 54.

[59] C. Gu, W. Liu, N. Hanley, R. Hesselbarth, and M. O'Neill. "A Theoretical Model to Link Uniqueness and Min-Entropy for PUF Evaluations." In: *IEEE Transactions on Computers* (2018), pp. 1–1.

[60] Jorge Guajardo, Sandeep S. Kumar, Geert-Jan Schrijen, and Pim Tuyls. "FPGA Intrinsic PUFs and Their Use for IP Protection." In: *Cryptographic Hardware and Embedded Systems - CHES 2007*. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, Sept. 10, 2007, pp. 63–80.

[61] O. Günlü and O. İşcan. "DCT Based Ring Oscillator Physical Unclonable Functions." In: *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. May 2014, pp. 8198–8201.

[62] *Hainan Island Incident*. In: *Wikipedia*. Page Version ID: 849618441. July 10, 2018.

[63] H. Handschuh and E. Trichina. "Securing Flash Technology." In: *Workshop on Fault Diagnosis and Tolerance in Cryptography (FDTC 2007)*. Sept. 2007, pp. 3–20.

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

[65] Clemens Helfmeier, Dmitry Nedospasov, Christopher Tarnovsky, Jan Starbug Krissler, Christian Boit, and Jean-Pierre Seifert. "Breaking and Entering Through the Silicon." In: *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security*. CCS '13. New York, NY, USA: ACM, 2013, pp. 733–744.

[66] Maxim Hennig, Oliver Schimmel, Philipp Zieris, and Georg Sigl. "Manipulationssensible Kopierschutzfolie (*Translated from German: Tamper-Sensitive Foil for Copy Protection*)." In: *D-A-CH Security 2013*. 2013.

[67] C. Herder, M. Yu, F. Koushanfar, and S. Devadas. "Physical Unclonable Functions and Applications: A Tutorial." In: *Proceedings of the IEEE* 102.8 (Aug. 2014), pp. 1126–1141.

[68] R. Hesselbarth, F. Wilde, C. Gu, and N. Hanley. "Large Scale RO PUF Analysis over Slice Type, Evaluation Time and Temperature on 28nm Xilinx FPGAs." In: *2018 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*. Apr. 2018, pp. 126–133.

[69] Hewlett-Packard Company. "Atalla Cryptographic Subsystem (ACS) Security Policy (Compliant to FIPS 140-2 Level 4)." In: (July 2009).

[70] Hewlett-Packard Company. "Atalla Cryptographic Subsystem (ACS) Security Policy (Compliant to FIPS 140-2 Level 3)." In: (Oct. 2010).

[71] M. Hiller, D. Merli, F. Stumpf, and G. Sigl. "Complementary IBS: Application Specific Error Correction for PUFs." In: *2012 IEEE International Symposium on Hardware-Oriented Security and Trust*. June 2012, pp. 1–6.

[72] Matthias Hiller. "Key Derivation with Physical Unclonable Functions." Dissertation. München: Technische Universität München, 2016.

[73] Matthias Hiller and Aysun Gurur Önalan. "Hiding Secrecy Leakage in Leaky Helper Data." In: *Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings*. Ed. by Wieland Fischer and Naofumi Homma. Vol. 10529. Lecture Notes in Computer Science. Springer, 2017, pp. 601–619.

[74] Matthias Hiller, Aysun Gurur Önalan, Georg Sigl, and Martin Bossert. "Online Reliability Testing for PUF Key Derivation." In: *Proceedings of the 6th International Workshop on Trustworthy Embedded Devices*. TrustED '16. New York, NY, USA: ACM, 2016, pp. 15–22.

[75] Matthias Hiller, Michael Pehl, Gerhard Kramer, and Georg Sigl. "Algebraic Security Analysis of Key Generation with Physical Unclonable Functions." In: (2016). https://eprint.iacr.org/2016/854.

[76] Matthias Hiller, Michael Weiner, Leandro Rodrigues Lima, Maximilian Birkner, and Georg Sigl. "Breaking Through Fixed PUF Block Limitations with Differential Sequence Coding and Convolutional Codes." In: *Proceedings of the 3rd International Workshop on Trustworthy Embedded Devices*. TrustED '13. New York, NY, USA: ACM, 2013, pp. 43–54.

[77] Matthias Hiller, Meng-Day (Mandel) Yu, and Michael Pehl. "Systematic Low Leakage Coding for Physical Unclonable Functions." In: *Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security, ASIA CCS '15, Singapore, April 14-17, 2015*. Ed. by Feng Bao, Steven Miller, Jianying Zhou, and Gail-Joon Ahn. ACM, 2015, pp. 155–166.

[78] Maximilian Hofer and Christoph Boehm. "An Alternative to Error Correction for SRAM-Like PUFs." In: *Cryptographic Hardware and Embedded Systems, CHES 2010*. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, Aug. 17, 2010, pp. 335–350.

[79] Daniel E. Holcomb, Wayne P. Burleson, and Kevin Fu. *Initial SRAM State as a Fingerprint and Source of True Random Numbers for RFID Tags*. 2007.

[80] Linh Hong. *Comparison of Embedded Non-Volatile Memory Technologies and Their Applications*. White Paper. Kilopass, May 2009, pp. 1–8.

[81] Y. Hori, T. Yoshida, T. Katashita, and A. Satoh. "Quantitative and Statistical Performance Evaluation of Arbiter Physical Unclonable Functions on FPGAs." In: *2010 International Conference on Reconfigurable Computing and FPGAs*. Dec. 2010, pp. 298–303.

[82] I. I. Huber, F. Arthur, and Jennifer M. Scott. *The Role and Nature of Anti-Tamper Techniques in US Defense Acquisition*. DEPARTMENT OF THE AIR FORCE WASHINGTON DC, 1999.

[83] Gerardus Tarcisius Maria Hubert. "Device with Protection against Access to Secure Information." European pat. 0509567A2. Koninklijke Philips NV. Oct. 21, 1992.

[84] S.B. Hunter, J.A. Voltz, B.W. Lewis, and H.S. Wylie. "Tamper Respondent Sensor and Enclosure." In: (2010). US Patent 7,760,086.

[85] Stephen B. Hunter. "Tamper Respondent Enclosure." U.S. pat. 7978070B2. Gore W L and Associates (UK) Ltd. July 12, 2011.

[86] IANA. *DNSSEC Practice Statement for the Root Zone KSK Operator*. Oct. 1, 2016.

[87] IBM. "IBM 4765 Cryptographic Coprocessor Security Module Security Policy (Compliant to FIPS 140-2 Level 4)." In: (Dec. 2012).

[88] T. Ignatenko, G. Schrijen, B. Skoric, P. Tuyls, and F. Willems. "Estimating the Secrecy-Rate of Physical Unclonable Functions with the Context-Tree Weighting Method." In: *2006 IEEE International Symposium on Information Theory*. July 2006, pp. 499–503.

[89] T. Ignatenko and F. M. J. Willems. “Information Leakage in Fuzzy Commitment Schemes.” In: *IEEE Transactions on Information Forensics and Security* 5.2 (June 2010), pp. 337–348.

[90] Vincent Immler. “Breaking Hitag 2 Revisited.” In: *Security, Privacy, and Applied Cryptography Engineering*. Ed. by Andrey Bogdanov and Somitra Sanadhya. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2012, pp. 126–143.

[91] Vincent Immler, Maxim Hennig, Ludwig Kürzinger, and Georg Sigl. “Practical Aspects of Quantization and Tamper-Sensitivity for Physically Obfuscated Keys.” In: *Proceedings of the Third Workshop on Cryptography and Security in Computing Systems*. CS2 ’16. New York, NY, USA: ACM, 2016, pp. 13–18.

[92] Vincent Immler, Matthias Hiller, Qinzhi Liu, Andreas Lenz, and Antonia Wachter-Zeh. “Variable-Length Bit Mapping and Error-Correcting Codes for Higher-Order Alphabet PUFs.” In: *Security, Privacy, and Applied Cryptography Engineering (SPACE)*. 2017.

[93] Vincent Immler, Matthias Hiller, Qinzhi Liu, Andreas Lenz, and Antonia Wachter-Zeh. “Variable-Length Bit Mapping and Error-Correcting Codes for Higher-Order Alphabet PUFs—Extended Version.” In: *Journal of Hardware and Systems Security*. Journal of Hardware and Systems Security (Dec. 2018).

[94] Vincent Immler, Matthias Hiller, Johannes Obermaier, and Georg Sigl. “Take a Moment and Have Some t: Hypothesis Testing on Raw PUF Data.” In: *IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*. May 2017, pp. 128–129.

[95] Vincent Immler, Johannes Obermaier, Martin König, Matthias Hiller, and Georg Sigl. “B-TREPID: Batteryless Tamper-Resistant Envelope with a PUF and Integrity Detection.” In: *IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*. Apr. 2018, pp. 49–56.

[96] Vincent Immler, Johannes Obermaier, Martin König, Matthias Hiller, and Georg Sigl. “Next-Generation Anti-Tamper Envelopes for Cyber Physical Defense Systems - Extended Abstract.” In: *SCI-300 Specialists’ Meeting Proceedings on Cyber Physical Security of Defense Systems*. Vol. STO-MP-SCI-300. Florida: NATO Science and Technology Organization (STO), May 2018, p. 8.

[97] Vincent Immler, Johannes Obermaier, Kuan Kuan Ng, Fei Xiang Ke, JinYu Lee, Yak Peng Lim, Wei Koon Oh, Keng Hoong Wee, and Georg Sigl. “Secure Physical Enclosures from Covers with Tamper-Resistance.” In: *IACR Transactions on Cryptographic Hardware and Embedded Systems* (2019), pp. 51–96.

[98] Vincent Immler, Robert Specht, and Florian Unterstein. “Your Rails Cannot Hide from Localized EM: How Dual-Rail Logic Fails on FPGAs.” In: *Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings*. 2017, pp. 403–424.

[99] Vincent Immler, Robert Specht, and Florian Unterstein. “Your Rails Cannot Hide from Localized EM: How Dual-Rail Logic Fails on FPGAs—Extended Version.” In: *Journal of Cryptographic Engineering* 8.2 (June 1, 2018), pp. 125–139.

[100] Vincent Immler and Karthik Uppund. “New Insights to Key Derivation for Tamper Evident Physical Unclonable Functions.” In: *IACR Transactions on Cryptographic Hardware and Embedded Systems* (2019).

[101] *Iran–U.S. RQ-170 Incident*. In: *Wikipedia*. Page Version ID: 848313996. July 1, 2018.

[102] Phil Isaacs, Thomas Morris Jr, Michael J. Fisher, and Keith Cuthbert. "Tamper Proof, Tamper Evident Encryption Technology." In: *Pan Pacific Symposium*. SMTA, 2013.

[103] M. N. Islam, V. C. Patil, and S. Kundu. "On Enhancing Reliability of Weak PUFs via Intelligent Post-Silicon Accelerated Aging." In: *IEEE Transactions on Circuits and Systems I: Regular Papers* 65.3 (Mar. 2018), pp. 960–969.

[104] M. Jeon and J. Lee. "On Codes Correcting Bidirectional Limited-Magnitude Errors for Flash Memories." In: *2012 International Symposium on Information Theory and Its Applications*. Oct. 2012, pp. 96–100.

[105] Joint Interpretation Library. *Application of Attack Potential to Hardware Devices with Security Boxes*. SOGIS, Dec. 2015.

[106] Ari Juels and Martin Wattenberg. "A Fuzzy Commitment Scheme." In: *Proceedings of the 6th ACM Conference on Computer and Communications Security*. CCS '99. New York, NY, USA: ACM, 1999, pp. 28–36.

[107] Stefan Katzenbeisser, Ünal Kocabaş, Vladimir Rožić, Ahmad-Reza Sadeghi, Ingrid Verbauwhede, and Christian Wachsmann. "PUFs: Myth, Fact or Busted? A Security Evaluation of Physically Unclonable Functions (PUFs) Cast in Silicon." In: *Cryptographic Hardware and Embedded Systems – CHES 2012*. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, Sept. 9, 2012, pp. 283–301.

[108] Kenji Kawano, Masahiro Taguchi, Masaki Hirota, Junji Okada, Masao Funada, and Takashi Ozawa. "Physical Property Based Cryptographics." U.S. pat. 6233339B1. Fuji Xerox Co Ltd. May 15, 2001.

[109] F. Keister and J. Rust. "Pyrotechnic Eradication of Microcircuits." U.S. pat. 3725671A. US Secretary of Navy. Apr. 3, 1973.

[110] *Kerckhoffs's Principle*. In: *Wikipedia*. Page Version ID: 844528755. June 5, 2018.

[111] Wolfgang Killmann and Kerstin Lemke-Rust. "Common Criteria Protection Profile - Cryptographic Modules, Security Level "Enhanced"." In: (July 2008).

[112] Inyoung Kim, Abhranil Maiti, Leyla Nazhandali, Patrick Schaumont, Vignesh Vivekraja, and Huaiye Zhang. "From Statistics to Circuits: Foundations for Future Physical Unclonable Functions." In: *Towards Hardware-Intrinsic Security*. Information Security and Cryptography. Springer, Berlin, Heidelberg, 2010, pp. 55–78.

[113] Theodoor A. Kleijne. "Security Device for the Secure Storage of Sensitive Data." U.S. pat. 4593384A. NCR Corp. June 3, 1986.

[114] Paul Kocher, Ruby Lee, Gary McGraw, Anand Raghunathan, and Srivaths Moderator-Ravi. "Security as a New Dimension in Embedded System Design." In: *Proceedings of the 41st Annual Design Automation Conference*. ACM, 2004, pp. 753–760.

[115] F. Kodýtek, R. Lórencz, J. Bucek, and S. Buchovecká. "Temperature Dependence of ROPUF on FPGA." In: *2016 Euromicro Conference on Digital System Design (DSD)*. Aug. 2016, pp. 698–702.

[116] Oliver Kömmerling and Fritz Kömmerling. "Anti Tamper Encapsulation for an Integrated Circuit." U.S. pat. 7005733B2. Koemmerling Oliver, Koemmerling Fritz. Feb. 28, 2006.

[117] G. Kömürcü and G. Dündar. “Determining the Quality Metrics for PUFs and Performance Evaluation of Two RO-PUFs.” In: *10th IEEE International NEWCAS Conference*. June 2012, pp. 73–76.

[118] H. Kreft and W. Adi. “Cocoon-PUF, a Novel Mechatronic Secure Element Technology.” In: *2012 NASA/ESA Conference on Adaptive Hardware and Systems (AHS)*. June 2012, pp. 227–232.

[119] Heinz Kreft. “Tamper-Protected Hardware and Method for Using Same.” U.S. pat. 9461826B2. EMSYCON GmbH. Oct. 4, 2016.

[120] Olga Kulikovska, Manfred Paeschke, Walter Fumy, and Frank Morgner. “Identity Card with Physical Unclonable Function.” U.S. pat. 20150286914A1. Bundesdruckerei GmbH. Oct. 8, 2015.

[121] S. S. Kumar, J. Guajardo, R. Maes, G. Schrijen, and P. Tuyls. “Extended Abstract: The Butterfly PUF Protecting IP on Every FPGA.” In: *2008 IEEE International Workshop on Hardware-Oriented Security and Trust*. June 2008, pp. 67–70.

[122] L. Kusters, T. Ignatenko, F. M. J. Willems, R. Maes, E. van der Sluis, and G. Selimis. “Security of Helper Data Schemes for SRAM-PUF in Multiple Enrollment Scenarios.” In: *2017 IEEE International Symposium on Information Theory (ISIT)*. June 2017, pp. 1803–1807.

[123] E. A. Lee. “Cyber Physical Systems: Design Challenges.” In: *2008 11th IEEE International Symposium on Object and Component-Oriented Real-Time Distributed Computing (ISORC)*. May 2008, pp. 363–369.

[124] Vincent van der Leest, Bart Preneel, and Erik van der Sluis. “Soft Decision Error Correction for Compact Memory-Based PUFs Using a Single Enrollment.” In: *Cryptographic Hardware and Embedded Systems – CHES 2012*. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, Sept. 9, 2012, pp. 268–282.

[125] V. I. Levenshtein. “Binary Codes Capable of Correcting Deletions, Insertions and Reversals (in Russian).” In: *Doklady Akademii Nauk SSR* 163.4 (1965), pp. 845–848.

[126] K. Lim, K. Jung, C. Jang, J. Baek, and I. Kang. “A Fast and Energy Efficient Single-Chip Touch Controller for Tablet Touch Applications.” In: *Journal of Display Technology* 9.7 (July 2013), pp. 520–526.

[127] H. Liu, W. Liu, Z. Lu, Q. Tong, and Z. Liu. “Methods for Estimating the Convergence of Inter-Chip Min-Entropy of SRAM PUFs.” In: *IEEE Transactions on Circuits and Systems I: Regular Papers* 65.2 (Feb. 2018), pp. 593–605.

[128] K. Lofstrom, W. R. Daasch, and D. Taylor. “IC Identification Circuit Using Device Mismatch.” In: *2000 IEEE International Solid-State Circuits Conference. Digest of Technical Papers (Cat. No.00CH37056)*. Feb. 2000, pp. 372–373.

[129] Heiko Lohrke, Shahin Tajik, Christian Boit, and Jean-Pierre Seifert. “No Place to Hide: Contactless Probing of Secret Data on FPGAs.” In: *Cryptographic Hardware and Embedded Systems – CHES 2016*. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, Aug. 17, 2016, pp. 147–167.

[130] Heiko Lohrke, Shahin Tajik, Thilo Krachenfels, Christian Boit, and Jean-Pierre Seifert. “Key Extraction Using Thermal Laser Stimulation: A Case Study on Xilinx Ultrascale FPGAs.” In: (2018). https://eprint.iacr.org/2018/717.

[131] Hugh MacPherson. "Security Enclosure Manufacture." U.S. pat. 5539379A. Gore W L and Associates (UK) Ltd. July 23, 1996.

[132] Hugh MacPherson. "Tamper Respondent Enclosure." U.S. pat. 5858500A. Gore W L and Associates Inc. Jan. 12, 1999.

[133] Roel Maes. "Physically Unclonable Functions: Constructions, Properties and Applications." KU Leuven, 2012.

[134] Roel Maes, Anthony Van Herrewege, and Ingrid Verbauwhede. "PUFKY: A Fully Functional PUF-Based Cryptographic Key Generator." In: *Cryptographic Hardware and Embedded Systems – CHES 2012*. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, Sept. 9, 2012, pp. 302–319.

[135] 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*. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, Sept. 13, 2015, pp. 517–534.

[136] Roel Maes, Pim Tuyls, and Ingrid Verbauwhede. *Intrinsic PUFs from Flip-Flops on Reconfigurable Devices*. 2008.

[137] Roel Maes, Pim Tuyls, and Ingrid Verbauwhede. "Low-Overhead Implementation of a Soft Decision Helper Data Algorithm for SRAM PUFs." In: *Cryptographic Hardware and Embedded Systems - CHES 2009, 11th International Workshop, Lausanne, Switzerland, September 6-9, 2009, Proceedings*. Ed. by Christophe Clavier and Kris Gaj. Vol. 5747. Lecture Notes in Computer Science. Springer, 2009, pp. 332–347.

[138] Roel Maes and Ingrid Verbauwhede. "A Discussion on the Properties of Physically Unclonable Functions." In: *TRUST 2010 Workshop, Berlin*. 2010.

[139] A. Maiti, J. Casarona, L. McHale, and P. Schaumont. "A Large Scale Characterization of RO-PUF." In: *2010 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*. June 2010, pp. 94–99.

[140] Abhranil Maiti, Vikash Gunreddy, and Patrick Schaumont. *A Systematic Method to Evaluate and Compare the Performance of Physical Unclonable Functions*. 657. 2011.

[141] Stefan Mangard, Elisabeth Oswald, and Thomas Popp. *Power Analysis Attacks: Revealing the Secrets of Smart Cards*. Springer US, 2007.

[142] Thomas McGrath, Ibrahim E. Bagci, Zhiming M. Wang, Utz Roedig, and Robert J. Young. "A PUF Taxonomy." In: *Applied Physics Reviews* 6.1 (Feb. 12, 2019), p. 011303.

[143] Dominik Merli, Dieter Schuster, Frederic Stumpf, and Georg Sigl. "Semi-Invasive EM Attack on FPGA RO PUFs and Countermeasures." In: *Proceedings of the Workshop on Embedded Systems Security*. WESS '11. New York, NY, USA: ACM, 2011, 2:1–2:9.

[144] *Moore's Law*. In: *Wikipedia*. Page Version ID: 852551659. July 29, 2018.

[145] Amir Moradi and Vincent Immler. "Early Propagation and Imbalanced Routing, How to Diminish in FPGAs." In: *Proceedings of the 16th International Workshop on Cryptographic Hardware and Embedded Systems — CHES 2014 - Volume 8731*. Berlin, Heidelberg: Springer-Verlag, 2014, pp. 598–615.

[146] Sven Müelich, Sven Puchinger, Martin Bossert, Matthias Hiller, and Georg Sigl. "Error Correction for Physical Unclonable Functions Using Generalized Concatenated Codes." In: (July 30, 2014).

[147] National Institute of Standards and Technology (NIST). *FIPS PUB 140-2: Security Requirements for Cryptographic Modules*. Gaithersburg, MD, USA: NIST, May 2002.

[148] D. Nedospasov, J. Seifert, C. Helfmeier, and C. Boit. "Invasive PUF Analysis." In: *2013 Workshop on Fault Diagnosis and Tolerance in Cryptography*. Aug. 2013, pp. 30–38.

[149] NIST. "Recommendation for the Entropy Sources Used for Random Bit Generation." In: (2012).

[150] Johannes Obermaier, Florian Hauschild, Matthias Hiller, and Georg Sigl. "An Embedded Key Management System for PUF-Based Security Enclosures." In: *2018 7th Mediterranean Conference on Embedded Computing (MECO)*. June 2018.

[151] Johannes Obermaier and Vincent Immler. "The Past, Present, and Future of Physical Security Enclosures: From Battery-Backed Monitoring to PUF-Based Inherent Security and Beyond." In: *Journal of Hardware and Systems Security*. Journal of Hardware and Systems Security (Aug. 15, 2018), pp. 1–8.

[152] Johannes Obermaier, Vincent Immler, Matthias Hiller, and Georg Sigl. "A Measurement System for Capacitive PUF-Based Security Enclosures." In: *Proceedings of the 55th Annual Design Automation Conference*. DAC '18. New York, NY, USA: ACM, 2018, 64:1–64:6.

[153] Stefano S. Oggioni, Vincenzo Condorelli, and Claudius Feger. "Multilayer Securing Structure and Method Thereof for the Protection of Cryptographic Keys and Code." U.S. pat. 20120117666A1. International Business Machines Corp. May 10, 2012.

[154] A. Ostmann, C. Boehme, K. Schrank, and K. Lang. "Development of a Microcamera with Embedded Image Processor Using Panel Level Packaging." In: *2015 European Microelectronics Packaging Conference (EMPC)*. Sept. 2015, pp. 1–4.

[155] Christof Paar and Jan Pelzl. *Understanding Cryptography: A Textbook for Students and Practitioners*. Berlin Heidelberg: Springer-Verlag, 2010.

[156] S. Paley, T. Hoque, and S. Bhunia. "Active Protection against PCB Physical Tampering." In: *2016 17th International Symposium on Quality Electronic Design (ISQED)*. Mar. 2016, pp. 356–361.

[157] Ravikanth Pappu. "Physical One-Way Functions." Massachusetts Institute of Technology, 2001.

[158] Ravikanth Pappu, Ben Recht, Jason Taylor, and Neil Gershenfeld. "Physical One-Way Functions." In: *Science* 297.5589 (2002), pp. 2026–2030.

[159] Zdenek Paral and Srinivas Devadas. "Reliable and Efficient PUF-Based Key Generation Using Pattern Matching." In: *Hardware-Oriented Security and Trust (HOST), 2011 IEEE International Symposium On*. IEEE, 2011, pp. 128–133.

[160] P. Paul, S. Moore, and S. Tam. "Tamper Protection for Security Devices." In: *2008 Bio-Inspired, Learning and Intelligent Systems for Security*. Aug. 2008, pp. 92–96.

[161] Payment Card Industry Security Standards Council. *Payment Card Industry PTS HSM Security Requirements, v2.0*. Wakefield, MA, USA: PCI, May 2012.

[162] Payment Card Industry Security Standards Council. *Payment Card Industry PTS POI Modular Derived Test Requirements, v4.0*. Wakefield, MA, USA: PCI, May 2013.

[163] Siani Pearson and Boris Balacheff. *Trusted Computing Platforms: TCPA Technology in Context*. Prentice Hall Professional, 2003. 358 pp.

[164] Michael Pehl, Tobias Tretschok, Daniel Becker, and Vincent Immler. "Spatial CTW for Physical Unclonable Functions." In: *To Be Published*. 2019.

[165] Ed Peterson. *Developing Tamper Resistant Designs with Xilinx Virtex-6 and 7 Series FPGAs*. Application Note XAPP1084. Xilinx, June 2017.

[166] Cuong V. Pham, David E. Chubin, Robert A. Clarke, and Aaron D. Kuan. "Anti-Tamper Mesh." U.S. pat. 7947911B1. Teledyne Technologies Inc. May 24, 2011.

[167] Liu Qinzhi. "Error Correction For Variable-Length PUF Quantization." A master's thesis co-advised by Vincent Immler. Aachen: RWTH Aachen University, 2017.

[168] Andrew Rae and Luke Wildman. "A Taxonomy of Attacks on Secure Devices." In: *Proceedings of the Australia Information Warfare and Security Conference 2003*. York, 2003, pp. 251–264.

[169] Srivaths Ravi, Anand Raghunathan, Paul Kocher, and Sunil Hattangady. "Security in Embedded Systems: Design Challenges." In: *ACM Transactions on Embedded Computing Systems (TECS)* 3.3 (2004), pp. 461–491.

[170] Irving Reed and Golomb Solomon. "Polynomial Codes over Certain Finite Fields." In: *Journal of the Society of Industrial and Applied Mathematics* 8.2 (June–1960), pp. 300–304.

[171] Samuel Rosset and Herbert R. Shea. "Flexible and Stretchable Electrodes for Dielectric Elastomer Actuators." In: *Applied Physics A* 110.2 (Feb. 1, 2013), pp. 281–307.

[172] D. Roy, J. H. Klootwijk, N. A. M. Verhaegh, H. H. A. J. Roosen, and R. A. M. Wolters. "Comb Capacitor Structures for On-Chip Physical Uncloneable Function." In: *IEEE Transactions on Semiconductor Manufacturing* 22.1 (Feb. 2009), pp. 96–102.

[173] U. Rührmair, J. L. Martinez-Hurtado, X. Xu, C. Kraeh, C. Hilgers, D. Kononchuk, J. J. Finley, and W. P. Burleson. "Virtual Proofs of Reality and Their Physical Implementation." In: *2015 IEEE Symposium on Security and Privacy*. May 2015, pp. 70–85.

[174] Ulrich Rührmair, Christian Jaeger, Christian Hilgers, Michael Algasinger, György Csaba, and Martin Stutzmann. "Security Applications of Diodes with Unique Current-Voltage Characteristics." In: *International Conference on Financial Cryptography and Data Security*. Springer, 2010, pp. 328–335.

[175] Ulrich Rührmair, Jan Sölter, and Frank Sehnke. "On the Foundations of Physical Unclonable Functions." In: *IACR Cryptology ePrint Archive* 2009 (2009), p. 277.

[176] D. Samyde, S. Skorobogatov, R. Anderson, and J.- Quisquater. "On a New Way to Read Data from Memory." In: *First International IEEE Security in Storage Workshop, 2002. Proceedings*. Dec. 2002, pp. 65–69.

[177] K. Saowapa, H. Kaneko, and E. Fujiwara. "Systematic Deletion/Insertion Error Correcting Codes with Random Error Correction Capability." In: *Proceedings 1999 IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems (EFT'99)*. Nov. 1999, pp. 284–292.

[178] Geert Schrijen and Boris Skoric. "On-Chip Estimation of Key-Extraction Parameters for Physical Tokens." European pat. 1972090B1. NXP BV. Aug. 19, 2015.

[179] Gary Schwenck, Mark Corio, and Keith Alexander Harrison. "Tamper-Evident/Tamper-Resistant Electronic Components." U.S. pat. 7065656B2. Hewlett-Packard Development Co LP. June 20, 2006.

[180] Johanna Sepulveda, Daniel Florez, Vincent Immler, Guy Gogniat, and Georg Sigl. "Efficient Security Zones Implementation through Hierarchical Group Key Management at NoC-Based MPSoCs." In: *Microprocessors and Microsystems* 50 (2017), pp. 164–174.

[181] Johanna Sepulveda, Daniel Flórez, Vincent Immler, Guy Gogniat, and Georg Sigl. "Hierarchical Group-Key Management for NoC-Based MPSoCs Protection." In: *Journal of Integrated Circuits and Systems* 11.1 (2016), pp. 38–48.

[182] L. Sha, S. Gopalakrishnan, X. Liu, and Q. Wang. "Cyber-Physical Systems: A New Frontier." In: *2008 IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (Sutc 2008)*. June 2008, pp. 1–9.

[183] Bundesamt für Sicherheit in der Informationstechnik. *Guidelines for Developer Documentation According to Common Criteria Version 3.1*. 2007.

[184] B. Skoric, S. Maubach, T. Kevenaar, and P. Tuyls. "Information-Theoretic Analysis of Capacitive Physical Unclonable Functions." In: *Journal of Applied Physics* 100.2 (2006).

[185] S. Skorobogatov. "How Microprobing Can Attack Encrypted Memory." In: *2017 Euromicro Conference on Digital System Design (DSD)*. Aug. 2017, pp. 244–251.

[186] Sergei Skorobogatov. "Physical Attacks and Tamper Resistance." In: *Introduction to Hardware Security and Trust*. Springer, 2012, pp. 143–173.

[187] Sergei Petrovich Skorobogatov. "Semi-Invasive Attacks: A New Approach to Hardware Security Analysis." PhD Thesis. Citeseer, 2005.

[188] Neil James Alexander Sloane. "On Single-Deletion-Correcting Codes." In: *Codes and Designs*. de Gruyter, 2002, pp. 273–292.

[189] J. M. Soden and R. E. Anderson. "IC Failure Analysis: Techniques and Tools for Quality Reliability Improvement." In: *Proceedings of the IEEE* 81.5 (May 1993), pp. 703–715.

[190] M. Spain, B. Fuller, K. Ingols, and R. Cunningham. "Robust Keys from Physical Unclonable Functions." In: *2014 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*. May 2014, pp. 88–92.

[191] Robert Specht, Vincent Immler, Florian Unterstein, Johann Heyszl, and Georg Sigl. "Dividing the Threshold: Multi-Probe Localized EM Analysis on Threshold Implementations." In: *IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*. Apr. 2018, pp. 33–40.

[192] Taras Stanko, Fitria Nur Andini, and Boris Skoric. "Optimized Quantization in Zero Leakage Helper Data Systems." In: *IEEE Transactions on Information Forensics and Security* (2017).

[193] G. E. Suh and S. Devadas. "Physical Unclonable Functions for Device Authentication and Secret Key Generation." In: *2007 44th ACM/IEEE Design Automation Conference*. June 2007, pp. 9–14.

[194] G. Edward Suh, Dwaine Clarke, Blaise Gassend, Marten van Dijk, and Srinivas Devadas. "AEGIS: Architecture for Tamper-Evident and Tamper-Resistant Processing." In: *Proceedings of the 17th Annual International Conference on Supercomputing*. ICS '03. New York, NY, USA: ACM, 2003, pp. 160–171.

[195] Manami Suzuki, Rei Ueno, Naofumi Homma, and Takafumi Aoki. "Multiple-Valued Debiasing for Physically Unclonable Functions and Its Application to Fuzzy Extractors." In: *Constructive Side-Channel Analysis and Secure Design*. Ed. by Sylvain Guilley. Lecture Notes in Computer Science. Springer International Publishing, 2017, pp. 248–263.

[196] Shahin Tajik. *On the Physical Security of Physically Unclonable Functions*. T-Labs Series in Telecommunication Services. Springer International Publishing, 2019.

[197] Shahin Tajik, Enrico Dietz, Sven Frohmann, Helmar Dittrich, Dmitry Nedospasov, Clemens Helfmeier, Jean-Pierre Seifert, Christian Boit, and Heinz-Wilhelm Hübers. "Photonic Side-Channel Analysis of Arbiter PUFs." In: *J. Cryptol.* 30.2 (Apr. 2017), pp. 550–571.

[198] Shahin Tajik, Heiko Lohrke, Jean-Pierre Seifert, and Christian Boit. "On the Power of Optical Contactless Probing: Attacking Bitstream Encryption of FPGAs." In: *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. CCS '17. New York, NY, USA: ACM, 2017, pp. 1661–1674.

[199] Christopher Tarnovsky. "Hacking the Smartcard Chip." In: Blackhat USA (2010).

[200] Lars Tebelmann, Michael Pehl, and Vincent Immler. "Side-Channel Analysis of the TERO PUF." In: *Constructive Side-Channel Analysis and Secure Design - 10th International Workshop, COSADE 2019, Darmstadt, Germany, April 3-5, 2019*. 2019, pp. 43–60.

[201] G. Tenengolts. "Nonbinary Codes, Correcting Single Deletion or Insertion (Corresp.)" In: *IEEE Transactions on Information Theory* 30.5 (Sept. 1984), pp. 766–769.

[202] The Common Criteria Recognition Agreement Members. "Common Criteria for Information Technology Security Evaluation." In: (Sept. 2006).

[203] Randy Torrance and Dick James. "The State-of-the-Art in IC Reverse Engineering." In: *Cryptographic Hardware and Embedded Systems - CHES 2009*. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, Sept. 6, 2009, pp. 363–381.

[204] Trusted Computing Group. *Trusted Platform Architecture Hardware Requirements for a Device Identifier Composition Engine*. May 19, 2017.

[205] Pim Tuyls, Anton H. M. Akkermans, Tom A. M. Kevenaar, Geert-Jan Schrijen, Asker M. Bazen, and Raimond N. J. Veldhuis. "Practical Biometric Authentication with Template Protection." In: *Audio- and Video-Based Biometric Person Authentication*. Ed. by Takeo Kanade, Anil Jain, and Nalini K. Ratha. Red. by David Hutchison, Takeo Kanade, Josef Kittler, Jon M. Kleinberg, Friedemann Mattern, John C. Mitchell, Moni Naor, Oscar Nierstrasz, C. Pandu Rangan, Bernhard Steffen, Madhu Sudan, Demetri Terzopoulos, Dough Tygar, Moshe Y. Vardi, and Gerhard Weikum. Vol. 3546. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 436–446.

[206] Pim Tuyls, Geert-Jan Schrijen, Boris Škorić, Jan van Geloven, Nynke Verhaegh, and Rob Wolters. "Read-Proof Hardware from Protective Coatings." In: *Cryptographic Hardware and Embedded Systems - CHES 2006*. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, Oct. 10, 2006, pp. 369–383.

[207] Karthik Uppund. "Improving Tamper-Sensitivity of Physical Unclonable Functions." A master's thesis advised by Vincent Immler. Munich: Technical University Munich, 2019.

[208] UTIMACO. "UTIMACO CryptoServer Se-Series Gen2 Security Policy (Compliant to FIPS 140-2 Level 3)." In: https://csrc.nist.gov/CSRC/media/projects/cryptographic-module-validation-program/documents/security-policies/140sp2814.pdf. Jan. 2018.

[209] M. Vai, B. Nahill, J. Kramer, M. Geis, D. Utin, D. Whelihan, and R. Khazan. "Secure Architecture for Embedded Systems." In: *2015 IEEE High Performance Extreme Computing Conference (HPEC)*. Sept. 2015, pp. 1–5.

[210] M. Vai, D. Whelihan, J. Leemaster, H. Whitman, W. Wan, Y. Fei, R. Khazan, I. Lebedev, K. Hogan, and S. Devadas. "Mission Assurance: Beyond Secure Processing." In: *2018 IEEE International Conference on Software Quality, Reliability and Security Companion (QRS-C)*. July 2018, pp. 593–598.

[211] Michael Vai, David J. Whelihan, Benjamin R. Nahill, Daniil M. Utin, Sean R. O'Melia, and Roger I. Khazan. "Secure Embedded Systems." In: *Lincoln Laboratory Journal* 22.1 (2016), pp. 110–122.

[212] 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*. TrustED '13. New York, NY, USA: ACM, 2013, pp. 13–20.

[213] R. R. Varshamov and G. M. Tenengolts. "Codes Which Correct Single Asymmetric Errors (in Russian)." In: *Automatika i Telemkhanika* 161.3 (1965), pp. 288–292.

[214] D. C. Vasile and P.M. Svasta. "Antitamper Conductive Mesh Used for Securing Cryptographic Modules." In: *2018 IEEE 24th International Symposium for Design and Technology in Electronic Packaging (SIITME)*. Oct. 2018, pp. 230–233.

[215] E. A. Verbitskiy, P. Tuyls, C. Obi, B. Schoenmakers, and B. Skoric. "Key Extraction From General Nondiscrete Signals." In: *IEEE Transactions on Information Forensics and Security* 5.2 (June 2010), pp. 269–279.

[216] Arunkumar Vijayakumar, Vinay C. Patil, and Sandip Kundu. "On Improving Reliability of SRAM-Based Physically Unclonable Functions." In: *Journal of Low Power Electronics and Applications* 7.1 (2017), p. 2.

[217] John von Neumann. "Various Techniques Used in Connection With Random Digits." In: *Applied Math Series* (1951).

[218] W.L. GORE & Associates Inc. "GORE Secure Encapsulated Module (Commercial Brochure)." In: (2007).

[219] W.L. GORE & Associates Inc. "GORE Tamper Respondent Surface Enclosure (Commercial Brochure)." In: (2007).

[220] M. Wan, Z. He, S. Han, K. Dai, and X. Zou. "An Invasive-Attack-Resistant PUF Based On Switched-Capacitor Circuit." In: *IEEE Transactions on Circuits and Systems I: Regular Papers* 62.8 (Aug. 2015), pp. 2024–2034.

[221] Hidehito Watanabe, Hideo Tsuzaka, and Masami Masuda. "Microdrilling for Printed Circuit Boards – Influence of Radial Run-out of Microdrills on Hole Quality." In: *Precision Engineering Journal of The International Societies for Precision Engineering and Nanotechnology – PRECIS ENG* (2008).

[222] Lingxiao Wei, Chaosheng Song, Yannan Liu, Jie Zhang, Feng Yuan, and Qiang Xu. "BoardPUF: Physical Unclonable Functions for Printed Circuit Board Authentication." In: *IEEE/ACM ICCAD*. 2015.

[223] S. H. Weingart. "Physical Security for the uABYSS System." In: *1987 IEEE Symposium on Security and Privacy*. Apr. 1987, pp. 52–52.

[224] S. H. Weingart, S. R. White, W. C. Arnold, and G. P. Double. "An Evaluation System for the Physical Security of Computing Systems." In: *[1990] Proceedings of the Sixth Annual Computer Security Applications Conference*. Dec. 1990, pp. 232–243.

[225] Steve H. Weingart. "Tamper-Resistant Packaging for Protection of Information Stored in Electronic Circuitry." U.S. pat. 4860351A. International Business Machines Corp. Aug. 22, 1989.

[226] Steve H. Weingart. "Physical Security Devices for Computer Subsystems: A Survey of Attacks and Defenses." In: *Cryptographic Hardware and Embedded Systems — CHES 2000*. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, Aug. 17, 2000, pp. 302–317.

[227] A. Wild, G. T. Becker, and T. Güneysu. "A Fair and Comprehensive Large-Scale Analysis of Oscillation-Based PUFs for FPGAs." In: *2017 27th International Conference on Field Programmable Logic and Applications (FPL)*. Sept. 2017, pp. 1–7.

[228] F. Wilde, B. M. Gammel, and M. Pehl. "Spatial Correlation Analysis on Physical Unclonable Functions." In: *IEEE Transactions on Information Forensics and Security* 13.6 (June 2018), pp. 1468–1480.

[229] F. Wilde, M. Hiller, and M. Pehl. "Statistic-Based Security Analysis of Ring Oscillator PUFs." In: *2014 International Symposium on Integrated Circuits (ISIC)*. Dec. 2014, pp. 148–151.

[230] Florian Wilde. "Large Scale Characterization of SRAM on Infineon XMC Microcontrollers As PUF." In: *Proceedings of the Fourth Workshop on Cryptography and Security in Computing Systems*. 2017.

[231] Oliver Willers, Christopher Huth, Jorge Guajardo, and Helmut Seidel. "MEMS Gyroscopes As Physical Unclonable Functions." In: *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. CCS '16. New York, NY, USA: ACM, 2016, pp. 591–602.

[232] B. Willsch, J. Hauser, S. Dreiner, A. Goehlich, H. Kappert, and H. Vogt. "Analysis of Semiconductor Process Variations by Means of Hierarchical Median Polish." In: *2017 Austrochip Workshop on Microelectronics (Austrochip)*. Oct. 2017, pp. 1–5.

[233] B. Willsch, J. Hauser, S. Dreiner, A. Goehlich, and H. Vogt. "Statistical Tests to Determine Spatial Correlations in the Response Behavior of PUF." In: *2016 12th Conference on Ph.D. Research in Microelectronics and Electronics (PRIME)*. June 2016, pp. 1–4.

[234] B. Willsch, K. Müller, Q. Zhang, J. Hauser, S. Dreiner, A. Stanitzki, H. Kappert, R. Kokozinski, and H. Vogt. "Implementation of an Integrated Differential Readout Circuit for Transistor-Based Physically Unclonable Functions." In: *2017 Austrochip Workshop on Microelectronics (Austrochip)*. Oct. 2017, pp. 58–63.

[235] W. Yan, C. Jin, F. Tehranipoor, and J. A. Chandy. "Phase Calibrated Ring Oscillator PUF Design and Implementation on FPGAs." In: *2017 27th International Conference on Field Programmable Logic and Applications (FPL)*. Sept. 2017, pp. 1–8.

[236] I. Yang and O. Kwon. "A Touch Controller Using Differential Sensing Method for On-Cell Capacitive Touch Screen Panel Systems." In: *IEEE Transactions on Consumer Electronics* 57.3 (Aug. 2011), pp. 1027–1032.

[237] Bennet Yee. "Using Secure Coprocessors." PhD Thesis. IBM, 1994.

[238] Wai Mun Yee, M. Paniccia, T. Eiles, and V. Rao. "Laser Voltage Probe (LVP): A Novel Optical Probing Technology for Flip-Chip Packaged Microprocessors." In: *Proceedings of the 1999 7th International Symposium on the Physical and Failure Analysis of Integrated Circuits*. 1999, pp. 15–20.

[239] C. E. D. Yin and G. Qu. "LISA: Maximizing RO PUF's Secret Extraction." In: *2010 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*. 2010, pp. 100–105.

[240] M. Yu, M. Hiller, and S. Devadas. "Maximum-Likelihood Decoding of Device-Specific Multi-Bit Symbols for Reliable Key Generation." In: *2015 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*. May 2015, pp. 38–43.

[241] Mandel Yu and Srinivas Devadas. "Secure and Robust Error Correction for Physical Unclonable Functions." In: *IEEE Design & Test of Computers* 27.1 (2010), pp. 48–65.

[242] Meng-Day (Mandel) Yu, David M'Raihi, Richard Sowell, and Srinivas Devadas. "Lightweight and Secure PUF Key Storage Using Limits of Machine Learning." In: *Cryptographic Hardware and Embedded Systems – CHES 2011*. Lecture Notes in Computer Science. Springer, Berlin, Heidelberg, Sept. 28, 2011, pp. 358–373.

[243] Christian Zenger, David Holin, and Lars Steinschulte. "Enclosure PUF – Tamper Proofing Commodity Hardware and Other Applications." In: *35th Chaos Communication Congress (35c3)*. Dec. 29, 2018.

About the Author

Vincent Immler was born on June 11th, 1987 in Neuenbürg, Germany. He received the Master's degree in IT-Security/Information Technology from Ruhr-Universität Bochum, Germany in 2013. During his studies, he did a 7-month internship at escrypt Inc., Michigan, USA as a Security Engineer and was part of the 3-month "Extreme Blue" program of IBM Research & Development GmbH, Böblingen, Germany. He joined the Fraunhofer Institute for Applied and Integrated Security (AISEC) in Garching, Germany as a full-time employee in 2013. In May 2014, he additionally enrolled as PhD student at Technical University Munich (TUM). The research for his PhD studies led to numerous publications in peer-reviewed, international conferences and journals. His work was accepted at several top venues such as CHES. For his contribution to HOST'18, he was awarded the best paper award. Additionally, his work was featured in other media such as IEEE Spectrum. He reviewed several articles for academic conferences such as HOST and DAC as sub-reviewer.

List of Publications

The author of this thesis has worked in several research areas. His thesis is a monograph, containing unpublished material, but is based on the following contributions to the hardware security and cryptographic community until March 2019. All publications are listed in chronological order and sorted in international conferences/workshops and journal papers.

International Conferences and Workshops

- Vincent Immler. "Breaking Hitag 2 Revisited." In: *Security, Privacy, and Applied Cryptography Engineering*. Ed. by Andrey Bogdanov and Somitra Sanadhya. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2012, pp. 126–143
- Amir Moradi and Vincent Immler. "Early Propagation and Imbalanced Routing, How to Diminish in FPGAs." In: *Proceedings of the 16th International Workshop on Cryptographic Hardware and Embedded Systems — CHES 2014 - Volume 8731*. Berlin, Heidelberg: Springer-Verlag, 2014, pp. 598–615
- Vincent Immler, Maxim Hennig, Ludwig Kürzinger, and Georg Sigl. "Practical Aspects of Quantization and Tamper-Sensitivity for Physically Obfuscated Keys." In: *Proceedings of the Third Workshop on Cryptography and Security in Computing Systems*. CS2 '16. New York, NY, USA: ACM, 2016, pp. 13–18
- Vincent Immler, Matthias Hiller, Johannes Obermaier, and Georg Sigl. "Take a Moment and Have Some t: Hypothesis Testing on Raw PUF Data." In: *IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*. May 2017, pp. 128–129
- Vincent Immler, Robert Specht, and Florian Unterstein. "Your Rails Cannot Hide from Localized EM: How Dual-Rail Logic Fails on FPGAs." In: *Cryptographic Hardware and Embedded Systems - CHES 2017 - 19th International Conference, Taipei, Taiwan, September 25-28, 2017, Proceedings*. 2017, pp. 403–424
- Vincent Immler, Matthias Hiller, Qinzhi Liu, Andreas Lenz, and Antonia Wachter-Zeh. "Variable-Length Bit Mapping and Error-Correcting Codes for Higher-Order Alphabet PUFs." In: *Security, Privacy, and Applied Cryptography Engineering (SPACE)*. 2017
- Vincent Immler, Johannes Obermaier, Martin König, Matthias Hiller, and Georg Sigl. "B-TREPID: Batteryless Tamper-Resistant Envelope with a PUF and Integrity Detection." In: *IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*. Apr. 2018, pp. 49–56
- Robert Specht, Vincent Immler, Florian Unterstein, Johann Heyszl, and Georg Sigl. "Dividing the Threshold: Multi-Probe Localized EM Analysis on Threshold Implementations." In: *IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*. Apr. 2018, pp. 33–40

- E. Ferres, V. Immler, A. Utz, A. Stanitzki, R. Lerch, and R. Kokozinski. "Capacitive Multi-Channel Security Sensor IC for Tamper-Resistant Enclosures." In: *2018 IEEE SENSORS*. Oct. 2018, pp. 1–4

- Vincent Immler, Johannes Obermaier, Martin König, Matthias Hiller, and Georg Sigl. "Next-Generation Anti-Tamper Envelopes for Cyber Physical Defense Systems - Extended Abstract." In: *SCI-300 Specialists' Meeting Proceedings on Cyber Physical Security of Defense Systems*. Vol. STO-MP-SCI-300. Florida: NATO Science and Technology Organization (STO), May 2018, p. 8

- Johannes Obermaier, Vincent Immler, Matthias Hiller, and Georg Sigl. "A Measurement System for Capacitive PUF-Based Security Enclosures." In: *Proceedings of the 55th Annual Design Automation Conference*. DAC '18. New York, NY, USA: ACM, 2018, 64:1–64:6

- Vincent Immler, Johannes Obermaier, Kuan Kuan Ng, Fei Xiang Ke, JinYu Lee, Yak Peng Lim, Wei Koon Oh, Keng Hoong Wee, and Georg Sigl. "Secure Physical Enclosures from Covers with Tamper-Resistance." In: *IACR Transactions on Cryptographic Hardware and Embedded Systems* (2019), pp. 51–96

- Vincent Immler and Karthik Uppund. "New Insights to Key Derivation for Tamper Evident Physical Unclonable Functions." In: *IACR Transactions on Cryptographic Hardware and Embedded Systems* (2019)

- Lars Tebelmann, Michael Pehl, and Vincent Immler. "Side-Channel Analysis of the TERO PUF." in: *Constructive Side-Channel Analysis and Secure Design - 10th International Workshop, COSADE 2019, Darmstadt, Germany, April 3-5, 2019*. 2019, pp. 43–60

- Michael Pehl, Tobias Tretschok, Daniel Becker, and Vincent Immler. "Spatial CTW for Physical Unclonable Functions." In: *To Be Published*. 2019

International Journals

- Johanna Sepulveda, Daniel Flórez, Vincent Immler, Guy Gogniat, and Georg Sigl. "Hierarchical Group-Key Management for NoC-Based MPSoCs Protection." In: *Journal of Integrated Circuits and Systems* 11.1 (2016), pp. 38–48

- Johanna Sepulveda, Daniel Florez, Vincent Immler, Guy Gogniat, and Georg Sigl. "Efficient Security Zones Implementation through Hierarchical Group Key Management at NoC-Based MPSoCs." In: *Microprocessors and Microsystems* 50 (2017), pp. 164–174

- Vincent Immler, Matthias Hiller, Qinzhi Liu, Andreas Lenz, and Antonia Wachter-Zeh. "Variable-Length Bit Mapping and Error-Correcting Codes for Higher-Order Alphabet PUFs—Extended Version." In: *Journal of Hardware and Systems Security*. Journal of Hardware and Systems Security (Dec. 2018)

- Vincent Immler, Robert Specht, and Florian Unterstein. "Your Rails Cannot Hide from Localized EM: How Dual-Rail Logic Fails on FPGAs—Extended Version." In: *Journal of Cryptographic Engineering* 8.2 (June 1, 2018), pp. 125–139

- Johannes Obermaier and Vincent Immler. “The Past, Present, and Future of Physical Security Enclosures: From Battery-Backed Monitoring to PUF-Based Inherent Security and Beyond.” In: *Journal of Hardware and Systems Security*. Journal of Hardware and Systems Security (Aug. 15, 2018), pp. 1–8